LISTADO DE PUBLICACIONES Y PRESENTACIONES EN CONGRESOS DERIVADAS DE LA PRESENTE TESIS

[1] L. Panizo, M. Gascó, D.Y. Marcos del Blanco, J.A. Hermida, H. Aláiz. "E-voting system evaluation based on the Council of Europe recommendations: Helios Voting". IEEE Transactions on Emerging Topics in Computing, ISSN: 2168-6750, IMPACT FACTOR 2016: 4.017, Q1, JIF Percentile: 92.123. (changes pending), 2018.

[2] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "Review of Cryptographic Schemes applied to Remote Electronic Voting systems: remaining challenges and the upcoming post-quantum paradigm". Open Mathematics, ISSN 2391-5455, IMPACT FACTOR 2016: 0.682, Q2, 2017.

[3] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "Ciberseguridad y e-democracia: Un estudio protocolizado del sistema Helios Voting". In Proceedings of JNIC 2017: Jornadas Nacionales de Investigación en Ciberseguridad. Madrid, 31 de mayo, 1 y 2 de junio de 2017. ISBN: 978-84-608-4659-8. pp.: 9-16, 2017.

[4] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "The need for harmonization in the on-line e-voting field: Towards and European Standard for e-democracy". In: Proceedings of the International Conference on Electronic Voting E-Vote-ID 2016. 18-21 October 2016, Bregenz, Austria. Eds.: R. Krimmer, M. Volkamer, J. Barrat, J. Benaloh, N. Goodman, P.Y.A. Ryan. O. Spycher, V. Teague, G. Wenda. ISBN 978-9949-83-022-0. pp.: 339, 2016.

[5] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "Desarrollo de una metodología avanzada de evaluación de sistemas de voto electrónico remoto". In Proceedings of JNIC 2016: Jornadas Nacionales de Investigación en Ciberseguridad. Granada, 15-17 de junio de 2016. ISBN: 978-84-608-8070-7. pp.: 65-72, 2016.

[6] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "Development of a Holistic Methodology for the Evaluation of Remote Electronic Voting Systems". In: International Journal of Complex Systems in Science (IJCSS) Vol 6 (1), ISSN: 2174-6036, pp 37-57, 2016.

[7] D.Y. Marcos del Blanco, L. Panizo, J.A. Hermida. "Development of an Advanced Methodology for the Evaluation of Remote Electronic Voting Systems". In: Proceedings of ALAMA 2016, Leon 20-22 junio 2016, pp 47, 2016.

[8] D.Y. Marcos del Blanco. "I-voting opportunities and challenges of the ICT applied to e-democracy". In: Proceedings of the First European Cybersecurity Seminar, Astorga 14,15 julio 2017, 2017.

CIBERSEGURIDAD APLICADA A LA E-DEMOCRACIA: ANÁLISIS CRIPTOGRÁFICO Y DESARROLLO DE UNA METODOLOGÍA PRACTICA DE EVALUACIÓN PARA SISTEMAS DE VOTO ELECTRÓNICO REMOTO Y SU APLICACIÓN A LAS SOLUCIONES MÁS RELEVANTES

David Yeregui Marcos del Blanco

Tesis propuesta como cumplimiento parcial de
los requisitos para el doctorado en

Ingeniería de Producción y Computación

Escuela de Ingenierías Industrial, Informática y Aeroespacial

UNIVERSIDAD DE LEÓN

Directores de tesis:

DR. D. JOSÉ ÁNGEL HERMIDA ALONSO

DR. D. LUIS PANIZO ALONSO

León, febrero de 2018

Palabras clave: criptografía, ciberseguridad, e-democracia, e-voting.

DEDICATORIA

A mi familia. Tanto a los que estáis como a los que os fuisteis demasiado pronto.

A mi compañera de viaje. Sin tí nada sería posible.

Os llevo siempre a todos en el corazón.

Esta tesis es vuestra.

Tanto si crees que puedes, como si crees que no puedes, estás en lo cierto

-Henry Ford

πάντα ῥεῖ

Todo fluye

-Heráclito

Agradecimientos

El inmenso esfuerzo que conlleva la realización de una tesis doctoral es únicamente comparable al júbilo y orgullo que se siente al completarla de una manera en la que se sienten orgullosas todas las partes implicadas.

En mi caso concreto y debido a mi desempeño profesional, el presente trabajo se ha escrito a caballo entre España y Japón, entre un despacho en el Campus de Vegazana y el OpenSource Cafe de Shimokitazawa en Tokio. Entre trenes, aviones, jet-lags y obligaciones profesionales varias, esta tesis hubiese sido a todas luces imposible de no ser por la inestimable ayuda y comprensión recibida por las siguientes personas, a las que agradezco de todo corazón su aportación:

En primer lugar, a mis directores de tesis Dr. D. José Ángel Hermida Alonso y Dr. D. Luis Panizo Alonso. Sois mi referente tanto en el plano profesional como sobre todo en el personal. El principal objetivo de esta tesis es que esté mínimamente a vuestra altura como docentes/investigadores como así como sobre todo en el aspecto humano. Para mí ha sido un auténtico honor y placer que hayáis aceptado dirigirme la tesis y espero y deseo que nuestra ya longeva colaboración y amistad perdure por muchos años.

De la misma manera, me gustaría extender mi gratitud a mi tutor Dr. D. Andrés Sáez Schwedt. Su disponibilidad y presteza a la hora de abordar cualquier cuestión o contratiempo ha sido fundamental para la conclusión de la tesis.

En lo referente a las aportaciones de forma, contenido y enfoque, me gustaría expresar mi más profundo agradecimiento al Dr. D. Justo Carracedo, Dr. D. Jorge Ramió, Dr. D. Miguel Carriegos, Dr. D. Ramón Ángel Fernández, Dr. D. Héctor Aláiz, Dra. Dña. Mila Gascó y Dra. Dña. Oksana Kulyk así como a las empresas Scytl y *n*Votes por su absoluta disponibilidad y transparencia a la hora de contestar a todas mis (insistentes) preguntas.

Todo lo anterior, aún siendo indispensable, es únicamente relevante gracias a que mis pilares fundamentales son sólidos: en primer lugar, mis padres David y Soledad y mi hermana Clara. Todo lo que aquí escriba no alcanza para corresponder siquiera mínimamente al inquebrantable apoyo y ánimo que siempre me habéis brindado, sobre todo en los momentos más oscuros de zozobra. Sois mi alfa y mi omega. Esta tesis es vuestra.

Finalmente, a mi compañera de viaje. Nadie me comprende como tú, nadie cree en mí más que tú. Eres mi faro, eres mi brújula, eres mi vida. Tu sonrisa ilumina mis días más aciagos. Somos el mejor equipo del mundo.

A todos GRACIAS. Dicen que quien tiene un amigo tiene un tesoro. En ese caso, debo ser de las personas más ricas del mundo. Espero de corazón que todos los que habéis hecho posible esta tesis la encontréis digna de vuestra categoría. Con ello, mi objetivo estaría más que cumplido.

Resumen

Las Tecnologías de la Información y la Comunicación (TIC) han tenido un enorme impacto en multitud de facetas de nuestra vida cotidiana en los últimos años.

A principios del siglo XXI se pensaba que su influencia sería mayoritaria también en todo tipo de procesos electorales, como parte de lo que se ha venido en llamar la *e-democracia*.

Dicho escenario no se ha producido, o al menos no al ritmo anticipado. Las causas son numerosas y se detallan en la presente tesis. No obstante, destacan una serie de características que hacen del Voto Electrónico Remoto (VER) una disciplina especialmente exigente respecto al resto de aplicaciones de las TIC:

- La necesidad de asegurar simultáneamente la integridad y la privacidad reforzadas, antagónicas entre sí.
- El requerimiento de conseguirlo a largo plazo (en el caso de ataques descubiertos tras concluir los comicios).
- Lo que está en juego es la legitimidad misma de los comicios, que otorga un amplio poder al vencedor. Sus resultados pueden ser difícilmente revertibles en caso de fraude.
- La gran diversidad de criterios, teorías, definiciones y legislaciones en función del país.
- La existencia de un sistema tradicional fácil, intuitivo y verificable que funciona razonablemente bien.

A ello se añaden no uno sino tres vectores de ataque:

- El dispositivo del votante, con una estimación de entre un 30 y un 40% de ellos infectados por algún tipo de *malware*. Además, se encuentran en entornos no controlados, dificultando la protección de la privacidad.
- La red, existiendo numerosos ataques que han tenido como objetivo los protocolos criptográficos asociados.
- El propio sistema de Voto Electrónico Remoto.

En resumen, a los peligros habituales de cualquier actividad *on-line* se suman unos requerimientos de seguridad mayores y un fuerte efecto llamada para potenciales atacantes por la trascendencia de lo que está en juego.

Lo arriba comentado explica en buena parte la dificultad en la implementación de sistemas de VER en elecciones vinculantes en el ámbito político.

Por otra parte, los beneficios potenciales de la introducción del VER son muy notables:

- Mejor acceso general al voto, especialmente para colectivos como: residentes en el extranjero, personas con discapacidad motora, visual, temporalmente ausentes etc.

- Ahorro de costes con respecto al procedimiento tradicional.
- Mayor implicación y participación ciudadana en la vida pública.
- Desarrollado correctamente, podría incrementar la seguridad y transparencia.

En ese sentido, la presente tesis trata de contribuir a la materia desarrollando una metodología práctica de evaluación de sistemas de voto electrónico remoto transversal, para después aplicarla a los esquemas más relevantes hasta la fecha.

Para ello, se ha seguido un exhaustivo proceso que consta de los siguientes pasos:

1. Estudio y armonización del estado del arte y la seguridad del VER.
2. Definición de requisitos tradicionales homogéneos del VER, utilizando la metodología KORA, las pautas de CC, ISO 27001-IT *Grundschutz,* su integración por Simic-Draws et al., las recomendaciones del Consejo de Europa y los trabajos de la Dra. Volkamer y el Dr. Neumann.
3. Estudio cualitativo y cuantitativo de las principales experiencias de VER a nivel nacional e internacional en todo tipo de elecciones.
4. Basándose en las conclusiones del punto anterior, definición de criterios adicionales para la metodología no cubiertos por los requisitos tradicionales.
5. Consulta a expertos tanto de la industria como de la comunidad científica para revisar la metodología y añadir factores de ponderación a los criterios.
6. Definición formal de la metodología, compuesta por 2 requisitos *sine-qua-non* y 73 puntos de evaluación.
7. Aplicación de la metodología a los 5 esquemas del VER más relevantes hasta la fecha: Helios, Scytl, Agora/nVotes, Civitas, BeleniosRF.

Tras concluir con todo el proceso anterior, se está en disposición de contestar a las dos principales cuestiones que dan sentido a la presente disertación:

¿Existe en la actualidad algún sistema/tecnología de Voto Electrónico Remoto lista para ser implantada en procesos electorales?

y de ser así,

¿Bajo qué condiciones y hasta qué punto en términos de nivel de uso, tecnología y tipología de elecciones sería suficientemente segura su introducción?

Con ello, se pretende contribuir a crear una base común en el VER en forma de metodología de evaluación sobre la que cada país pueda incorporar su idiosincrasia particular para analizar las propuestas que esté valorando. El objetivo último es una introducción del VER de una manera gradual, basada en criterios técnicos y sobre todo segura.

Por último, indicar que la presente tesis supone una continuación al trabajo del Profesor Dr. Luis Panizo desde el año 2002 sobre el desarrollo de una metodología para el análisis y la clasificación de sistemas de Voto Electrónico, en su caso en entornos controlados.

Abstract

The Information and Communications Technologies (ICT) have had a huge impact in the day-to-day lives of billions of citizens around the globe in recent years.

Back in the early 2000s, it was anticipated that its range would also include public elections, as an integral part of what has been labeled as *e-democracy*.

Several years later, that forecast has not turned into a reality. Certainly, the Remote Electronic Vote (REV) shows several features making it an especially demanding discipline within the plethora of ICT applications:

- The need to simultaneously comply with two antagonistic properties: Integrity and Privacy.
- Furthermore, the necessity to assure them also in the long term (for the potential risk of attacks discovered after the elections concluded).
- What is at stake is the very legitimacy of the elections, which grants broad power to the winner. On the top of that, the results can be very difficult to revert in the event of a fraud discovered after the elections ended.
- A wide variety of theories, definitions, laws and criteria depending on the country.
- The existence of a traditional system which is simple, intuitive, verifiable which works considerably well.

On the top of that, there are 3 attack vectors:

- The voter device, with 30-40% of them infected by *malware* and located in non-controlled environments, thus contributing to an increased difficulty in privacy-control.
- The network itself; oftentimes target of attacks against its cryptographic protocols.
- The Remote Electronic Voting system.

In short, the REV has higher demands in terms of security while attracting sophisticated attackers because of the relevance of what is at stake compared to other on-line activities.

All of the aforementioned highlights some of the hurdles in the implementation of REV solutions for public binding elections.

On the other hand, the potential benefits of implementing REV systems are noteworthy:

- Better overall access to the polls, especially for vulnerable population groups.
- Great cost savings.
- Better citizen engagement.
- If properly developed, it could potentially increase safety.

This dissertation aims at contributing to a much needed harmonization by developing a practical methodology for the evaluation of REV systems and its application to the most relevant solutions to date.

To that end, the following process has been designed and executed:

1. Study and harmonization of the state-of-the-art mathematical and cryptographic principles and theories applied to the REV and its security.
2. Definition of an homogeneous set of traditional REV requirements based on: the KORA methodology, the CC and ISO 27001-IT *Grundschutz* guidelines, their integration by Simic-Draws et al., the Council of Europe Recommendations as well as Dr. Volkamer's and Dr. Neumann's research on the matter.
3. Qualitative and Quantitative research on the most relevant REV experiences both at a national and international level in all kinds of elections.
4. Based on the previous item's conclusions, definition of the practical additional criteria to be included in the evaluation methodology.
5. Consultation to more than 30 international experts in REV (both from academia and industry) to review the methodology and add weighting factors.
6. Formal definition of the methodology, comprised of 2 *sine-qua-non* requirements and 73 evaluation items.
7. Methodology application to the 5 most relevant REV solutions to date: Helios, Scytl, Agora/nVotes, Civitas and BeleniosRF.

After the above mentioned process has been successfully applied, we are ready to answer the following questions, ultimate reason for this Ph.D. dissertation:

¿Is there any Remote Electronic Voting System/Technology ready to be deployed in binding elections?

and if that is the case,

¿Under what circumstances and to what extent in terms of usage, technology and election type would its implementation be safe enough?

With those answers, the author tries to contribute to the establishment of a *"common ground"* in the field of REV, shaped as a practical evaluation methodology. From there, every country/territory can adapt it to its particular idiosyncrasy and the systems they are evaluating. The ultimate goal is to enhace a gradual, tech-based and safe introduction of the REV in elections.

Last but not least, it is relevant to point out that this Ph. D. thesis takes up the baton from Prof. Dr. Luis Panizo's ongoing research since 2002 on the development of REV evaluation methodologies.

Índice

Parte II. Análisis práctico y definición formal de la metodología

3. Antecedentes, experiencias previas y estado del arte

4. Criterios adicionales de evaluación y metodología completa

Parte III. Aplicación de la metodología a los sistemas de VER más relevantes

5. Análisis y comparativa de los sitemas de voto remoto electrónico más relevantes

INDICE DE FIGURAS

Número *Página*

INDICE DE TABLAS

Número *Página*

Anexos

GLOSARIO Y ACRÓNIMOS

VER: En la presente tesis, se entiende por Voto Electrónico Remoto *"Aquel sistema de votación que se produce en un entorno remoto y no controlado, por medios electrónicos y en el que el voto es enviado total o parcialmente a través de una conexión a internet desde un ordenador personal o dispositivo móvil que no es una máquina construida al efecto de votar".*

BB: Bulletin Board
BOEE: Washington D.C. Board of Election and Ethics
CDH: Computational Diffie-Hellman
Ciphersuite: Dícese de una combinación de autenticación, encripción, código de autenticación de mensaje (MAC) y algoritmos de intercambio de claves usada para negociar el marco de seguridad de una conexión de red.
Ciphertext, cifertexto o Texto cifrado: Es el resultado de aplicar un algoritmo de encriptación denominado cipher a un texto sin codificar.
Criptosistema: Un conjunto de procedimientos dirigido a garantizar la seguridad de la información (usualmente la confidencialidad) mediante técnicas criptográficas.
CSO: Chief Scientific Officer
Dark Web: Parte de internet no indexada con los motores de búsqueda habituales y que requiere de algún tipo de software específico, configuración o autorización para acceder.
DDH: *Decisional Diffie Hellman*
DDoS: *Distributed Denial of Service*
DES: *Data Encryption Standard*
DH: *Diffie-Hellman Key Exchange Protocol*
DHCP: *Dynamic Host Configuration Protocol*
DLP: *Discrete Logarithm Problem*
DNS: *Domain Name System*
DoS: *Denial of Service*
DRE: *Direct-Recording Electronic*
DVBM: *D.C. Digital Vote-by-Mail Service*
ECC: *Elliptic Curve Cryptography*
ECC: *European Citizen Card*
ECDLP: *Elliptic Curve Discrete Logarithm Problem*
EVAP: Elecciones vinculantes en el ámbito político
FEC: *Federal Election Commission*
GCHQ: *Government Communications Headquarters*
GNFS: *General Number Field Sieve*
HAVA: *Help America Vote Act*
HMAC: *Hash message authentication code*
IACR: *International Association for Cryptology Research*

IETF: *Internet Engineering Task Force*
IFP: *Integer Factorization Problem*
IP: *Internet Protocol*
IPSec: *Internet Protocol Security*
MOVE: *Military and Overseas Voter Empowerment Act*
MPC: *Multi Party Computation*
NDA: *Non-Disclosure agreement*
NIST: *National Institute of Standards and Technology*
NIZKP: *Non-Interactive Zero-Knowledge Proof*
NSA: *National Security Agency*
NTP: *Network Time Protocol*
OCR: *Optical Character Recognition*
OSCE: *Office for Democratic Institutions and Human Rights*
OSDV: *Open Source Digital Voting*
OSET: *Open Source Elections Technology*
OSS: *Open Source Software*
OVWP: *Online Voting Working Party*
PKC: *Public Key Cryptography*
PKE: *Public Key Encryption*
PKI: *Public Key Infrastructure*
PPT: *Probabilistic Polynomial Time*
SRA: *Supervised Registration Authorithm*
SSH: *Secure Shell*
SVP: *Senior Vice-President*
TCP/IP: *Transmission Control Protocol/Internet Protocol*
TIC: Tecnologías de la Información y las Comunicaciones
TLS: *Transport Layer Security*
TTP: *Trusted Third Party*
TV: Tarjeta de Votación
UOCAVA: *Uniformed and Overseas Citizens Absentee Voting Act*
VAP: Vinculante en el ámbito político
VI: Vector de inicialización
VSS: *Voting Systems Standards*
WCAG: *Web Content Accessibility Guidelines*
ZKP: *Zero-Knowledge Proof*

C a p í t u l o 1

JUSTIFICACIÓN, OBJETIVOS Y ESTRUCTURA

It's not the voting that's democracy, it's the counting

-Tom Stoppard

1.1 Introducción

Antes de comenzar con la presente tesis, el autor de la misma quiere manifestar que ha sido realizada de manera totalmente autónoma sin aportaciones de ninguna de las empresas, partidos políticos o partes interesadas que aparecen. Por tanto, las ideas y análisis aquí desarrollados y vertidos son personales, independientes y apolíticos, basados únicamente en criterios científicos y técnicos y pensados con el único objetivo de ofrecer una metodología de análisis lo más rigurosa y útil posible a los intereses democráticos de quien quiera utilizarla.

Las tecnologías de la información y la comunicación (TIC) han supuesto una revolución y un avance incontestable en multitud de ámbitos de la sociedad.

Su influencia es palpable en infinidad de actividades cotidianas: desde las relacionadas con las comunicaciones o el transporte, pasando por el ocio, el entorno laboral, la sanidad e incluso las transacciones económicas y demás actividades financieras.

Es indiscutible que todo ciudadano disfruta en mayor o menor medida de las facilidades y ventajas que paulatinamente han aportado las TIC a nuestras vidas.

Ello nos llevaría a pensar que, de la misma manera que las TIC han contribuido a una mayor accesibilidad y mejora en los ámbitos arriba enumerados, su aplicación a procesos de votación debería conformar un avance en el uso de las nuevas tecnologías.

No obstante, dicha evolución no se ha producido, o al menos no hasta los niveles esperados. Las razones son múltiples y se abordarán en detalle en esta disertación, no obstante hay dos que destacan:

Por una parte y como es bien conocido, la base de un Estado de Derecho es el conjunto de leyes que lo conforman y su cumplimiento. En consecuencia, la elección de los representantes que van a discutir, redactar, promulgar y aprobar dichas leyes constituyen la piedra angular y la base de la legitimidad de una sociedad democrática.

Es por ello que asegurar la transparencia y seguridad de los procesos de elección es una tarea irrenunciable y de la máxima importancia, sujeta a presiones e incluso ataques de partes interesadas en poner dichas leyes al servicio de sus intereses particulares. En ese sentido, los procesos electorales con utilización de TIC no han sido una excepción, como se verá con más detalle en el apartado 2.4 sobre seguridad del Voto Electrónico Remoto.

En segundo lugar, la necesidad de satisfacer en los sistemas de voto electrónico simultáneamente integridad y privacidad, al menos parcialmente antagónicas entre sí como apuntan [23, 1, 260, 388, 438]. Es necesario estar seguro que el resultado de la votación es el correcto SIN comprometer en ningún momento el anonimato del votante.

Es más, el sistema de Voto Electrónico Remoto debería estar en disposición de garantizar que un votante no pudiese obtener prueba alguna de qué opción eligió ni aun queriendo colaborar con un atacante.

El proceso tradicional de votación presenta una serie de características que lo hacen sólido y fiable, manteniendo las dos características arriba enunciadas. El problema es que su "digitalización" o "informatización" requiere la utilización de dispositivos, redes y programas cuya comprensión no está al alcance de la gran mayoría de los votantes y además constituyen potenciales objetivos de ataques. Todos estos particulares se analizan en detalle en los puntos 2.1, 2.2 y 2.3.

En ese sentido, la presente tesis doctoral supone un paso adelante natural en la misma dirección, al desarrollar una metodología de evaluación de sistemas de Voto Electrónico Remoto (VER) de aplicación práctica basada por una parte en los requerimientos tradicionales en el campo junto con los criterios adicionales obtenidos del estudio metodológico de las experiencias más relevantes en el campo en más de 15 países. Para concluir, se aplica la metodología desarrollada a los sistemas de VER más relevantes hasta la fecha.

Con todo ello, se pretende estar en una mejor disposición de responder a la gran pregunta: ¿Es el Voto Electrónico Remoto lo suficientemente seguro para ser utilizado?

La presente disertación recoge el resultado del proceso de revisión, análisis, implementación y aplicación llevado a cabo por el autor en su contribución al campo.

1.2 Justificación

La RAE define la democracia como la *"Doctrina política según la cual la soberanía reside en el pueblo, que ejerce el poder directamente o por medio de representantes."*

En ese sentido, parece legítimo el uso de las TIC para procesos de elecciones democráticas (algunas veces denominado *e-democracy* o *i-democracy* [59, 60, 61] siempre que comporten una mejora al sistema vigente sin comprometer ni la verificabilidad del resultado final ni la privacidad de los votantes.

Dicha mejora puede ser en la accesibilidad para personas con movilidad reducida o que viven en lugares más remotos, en eficiencia energética, en el proceso de recuento, en ahorro de recursos materiales, humanos, de seguridad etc. Después de todo, pese a que cada vez se tiene un mayor acceso a los gastos de numerosas partidas del Estado, sigue siendo relativamente desconocido el coste de un proceso electoral.

En el caso de España y tomando como fuente la partida destinada en los Presupuestos Generales del Estado para las Elecciones Generales de 2011, el coste de dichas elecciones fue de 124 millones de euros. Puediera parecer una cantidad muy importante; si bien hay que tener en cuenta que se movilizaron más de 100.000 efectivos de las Fuerzas y Cuerpos de Seguridad del Estado, hubo 59.876 mesas electorales y estaban llamados a votar casi 36 millones de españoles.

En las más recientes Elecciones Municipales y Autonómicas de 2015, el coste total fue de 127.99 millones de euros [76] y para el conjunto de las elecciones que se produjeron en ese año, el coste superó los 430 millones de euros.

A nivel internacional, y según el informe *"A Global Survey on the Cost of Registration and Elections*" del *Bureau for Development Policy* de las Naciones Unidas [5], el coste electoral por votante varía bastante de país a país. No obstante, existe una importante correlación entre la experiencia del país en elecciones democráticas y el coste per cápita (a menor experiencia mayor coste). Asimismo, un entorno político inestable o de democracia amenazada encarece muy notablemente el gasto electoral.

A modo de orientación, el coste por votante en países avanzados europeos oscila entre 1 y 4 USD. En países en vías de desarrollo el coste varía entre 5 y 10 USD y en países con misiones de paz y observadores internacionales presentes se suelen superar los 10 USD por votante.

El caso de Estados Unidos es particular; puesto que su sistema electoral con procesos de elección de los candidatos muy prolongados encarece de manera muy notable el coste total hasta más de 4.000 millones de USD o 18 USD por votante.

No obstante y en guarismos absolutos, las elecciones más costosas a nivel mundial son las de la India (la mayor democracia del planeta), con un coste total de 5.000 millones de USD para un censo de más de 1.300 millones de personas.

Se puede concluir por tanto que el ejercicio del derecho democrático acarrea unos costes muy notables, sujetos a beneficiarse al igual que muchos otros ámbitos de la introducción de las TIC para incrementar su seguridad, eficiencia y fiabilidad.

Es también relevante su contribución a la accesibilidad al voto. En países con un mayor nivel de desarrollo se trata de una cuestión menor. Ello no es así en otras áreas geográficas con infraestructuras menos desarrolladas (África subsahariana, Sudeste Asiático etc.). En esos casos, la implementación de sistemas de VER podría conllevar un importante ahorro en recursos, muy limitados de por sí.

Se podría comparar con el avance que supuso la telefonía móvil en los países más desfavorecidos, ya que pudieron desarrollar enormemente sus comunicaciones sin tener que realizar la enorme inversión en infraestructura de telefonía convencional (inversión que sí se hizo en países más avanzados y que hoy en día está mayormente en desuso). Pudieron "saltarse" una generación intermedia de desarrollo en infraestructuras que unos años después quedaron obsoletas, con el consiguiente ahorro.

Todo lo arriba comentado justifica la necesidad de una evaluación en profundidad de los sistemas de VER actualmente en uso. A tal fin, en la presente tesis se desarrolla una metodología práctica de evaluación para soluciones de VER y posteriormente se aplica a las más relevantes hasta la fecha. El objetivo último es tener un mejor conocimiento práctico sobre el Voto Electrónico y decidir hasta qué punto y a qué ritmo sería conveniente su implantación.

1.3 Objetivos

La progresiva introducción de la tecnología en los procesos democráticos constituye un escenario que va a ir implantándose paulatinamente, ya sea con mayor o menor intensidad y a una mayor o menor velocidad.

Por tanto, la finalidad primordial es que dicha implantación se produzca cuando la tecnología esté madura, a un ritmo prudente y sobre todo basada en criterios técnicos y objetivos, sin prisas ni presiones externas.

En consecuencia, el principal reto de la presente tesis es tratar de contestar, al menos en parte, a la pregunta:

¿Existe en la actualidad algún sistema/tecnología de Voto Electrónico Remoto lista para ser implantada en procesos electorales?

y de ser así,

¿Bajo qué condiciones y hasta qué punto en términos de nivel de uso, tecnología y tipología de elecciones sería suficientemente segura su introducción?

Para responder a dichas cuestiones con las mayores garantías, en la presente tesis se ha diseñado una metodología de evaluación gradual y procedimental, partiendo del estudio y comparación de las principales primitivas y conceptos criptográficos usados en el VER, concluyendo con la aplicación de la misma a los principales sistemas desarrollados.

En concreto, los pasos seguidos, cada uno de los cuales constituye un sub-objetivo, son:

1. Estudio y armonización del estado del arte y la seguridad del VER. (Capítulo 2)
2. Definición de requisitos tradicionales homogéneos del VER, utilizando parcialmente la metodología KORA [461], las pautas de CC [460], ISO 27001-IT Grundschutz [462], su integración por Simic-Draws et al. [457], así como las recomendaciones del Consejo de Europa [456] y los trabajos de Volkamer [456] y Neumann [458]. (Apartado 2.3)
3. Estudio cualitativo y cuantitativo de las principales experiencias de Voto Electrónico Remoto a nivel nacional e internacional de elecciones vinculantes en el ámbito político así como en otros ámbitos, sumando más de 500 elecciones y 6 millones de votos analizados desde todas sus vertientes. (Capítulo 3)
4. Basándose en las conclusiones del punto anterior, definición de criterios adicionales para la metodología que no cubren los requisitos tradicionales. (Capítulo 4)

5. Consulta a expertos tanto de la industria como de la comunidad científica para revisar la metodología y añadir factores de ponderación a los criterios utilizando las metodologías de sampleado *snowball* [454] y *judgement* [455] para su selección y cribado. (Apartado 4.2)

6. Definición formal de la metodología, compuesta finalmente por 2 requisitos *sine-qua-non* (garantes de las 5 propiedades intrínsecas a una votación democrática [66, 358]) y otros 10 cuantificables, que suman un total de 73 puntos de evaluación, detallados en el anexo A. (Apartado 4.2)

7. Aplicación de la metodología a los 5 esquemas del VER más relevantes hasta la fecha: Helios, Scytl, Agora/nVotes, Civitas, BeleniosRF. (Capítulo 5)

Una vez completados los puntos anteriores, se está en disposición de responder a la pregunta principal de la tesis expuesta al comienzo del presente apartado.

Adicionalmente, se indican una serie de propuestas y líneas de investigación futuras para el mantenimiento y la mejora de la metodología desarrollada, así como de los esquemas analizados y del Voto Electrónico Remoto en su conjunto.

1.4 Estructura de la tesis

La tesis se divide en cuatro partes y seis capítulos principales:

El presente capítulo 1 comienza con una introducción al Voto Electrónico Remoto (VER), seguido de una justificación al trabajo desarrollado en la presente tesis y una enumeración de los objetivos de la misma.

La primera parte de la tesis comienza con el capítulo 2, donde se realiza un pormenorizado desglose de los principales conceptos, consideraciones, definiciones y primitivas matemáticas y criptográficas (*building blocks*) que van a ser utilizadas en apartados posteriores.

Además, se definen los requerimientos armonizados que debe de cumplir un sistema de Voto Electrónico Remoto. Dichos requerimientos se extraen tras un exhaustivo análisis y comparación del estado del arte y las distintas corrientes de investigación junto con la aplicación de las metodologías más aceptadas en el campo: KORA [461], ISO 27001 [462], CC [460], su implementación conjunta [457] así como las recomendaciones del Consejo de Europa [456] y el trabajo de Volkamer et al [459] y la tesis doctoral de Neumann [458].

El último apartado del capítulo 2 está reservado a una cuestión de vital importancia: la seguridad del Voto Electrónico Remoto. Se detallan además las principales tipologías de ataques y se explican dos de los casos más recientes: FREAK y Logjam.

La segunda parte de la tesis se inicia en el capítulo 3 con un repaso en profundidad de los precedentes históricos de la democracia y los distintos tipos de sufragio y tipologías de voto, incluyendo el voto electrónico remoto.

Posteriormente se dividen las experiencias previas de voto electrónico remoto en dos grandes grupos: las experiencias en elecciones públicas vinculantes en el ámbito político (VAP en adelante) y las que han tenido lugar en otros ámbitos tales como el universitario, corporativo o simplemente consultivo.

En el caso de las primeras (vinculantes en el ámbito político), se realiza un exhaustivo análisis cualitativo y cuantitativo de los casos más destacados en los países que más decididamente han apostado por el voto electrónico remoto: Estonia, Noruega, Canadá, Estados Unidos de América, Australia, Suiza, Austria, Nueva Zelanda, Francia, España, Finlandia y Grecia, acabando con una tabla resumen del estado actual del Voto Electrónico Remoto en cada uno de dichos países.

Para cerrar el capítulo 3, se detallan las conclusiones extraídas de todas las experiencias de elecciones políticas vinculantes, así como en los demás ámbitos y que suman más de 500 elecciones y 6 millones de emitidos con sistemas de VER.

Una vez expuestos los principales conceptos, consideraciones, definiciones y requerimientos, habiendo también analizado la seguridad, los ataques, la historia y las experiencias previas de voto electrónico remoto con las metodologías más aceptadas en la disciplina, se está en disposición de definir en el capítulo 4 los criterios que conformarán la base de la metodología de evaluación: un total de 10 apartados más dos criterios *sine-qua-non*. En total 75 aspectos a evaluar, cada uno de ellos con su definición y codificación única asociada como se detalla en el anexo A.

Como mejora adicional, se ha introducido un factor de ponderación en cada criterio de evaluación. Para decidir el coeficiente específico de cada uno de ellos, el autor ha realizado una encuesta técnica que se ha enviado a 31 expertos en el campo del VER tanto en su vertiente académica como empresarial. Se ha aplicando la metodología de sampleado *snowball* [454] junto con la *judgement* [455] en aras de conseguir la selección más objetiva y completa posible.

Finalmente, 21 de los expertos contactados ofrecieron su *feedback*, el cual sirvió para definir la ponderación final de cada apartado, junto con el criterio de los directores de la presente tesis. En el anexo D se ofrece el desglose completo de las respuestas recibidas de la encuesta técnica.

A continuación, el capítulo 5 da comienzo a la tercera parte de la tesis, en la que se aplica la metodología desarrollada a las soluciones de voto electrónico remoto más relevantes (Helios, Scytl, Agora/nVotes, Civitas y nVotes). Adicionalmente se incluye un apartado sobre el sistema Votescript, desarrollado por el grupo de pioneros del VER en España.

Finalmente, la cuarta y última parte de la disertación se desarrolla en el capítulo 6. En él, se detallan las conclusiones de la tesis con respecto a los objetivos descritos en el punto 1.3, incluyendo las tablas y la comparativa final. Finalmente, se introducen las limitaciones de la metodología, las recomendaciones finales y se sugieren una serie de líneas futuras de trabajo.

De una manera más gráfica, la figura siguiente detalla la estructura de la tesis:

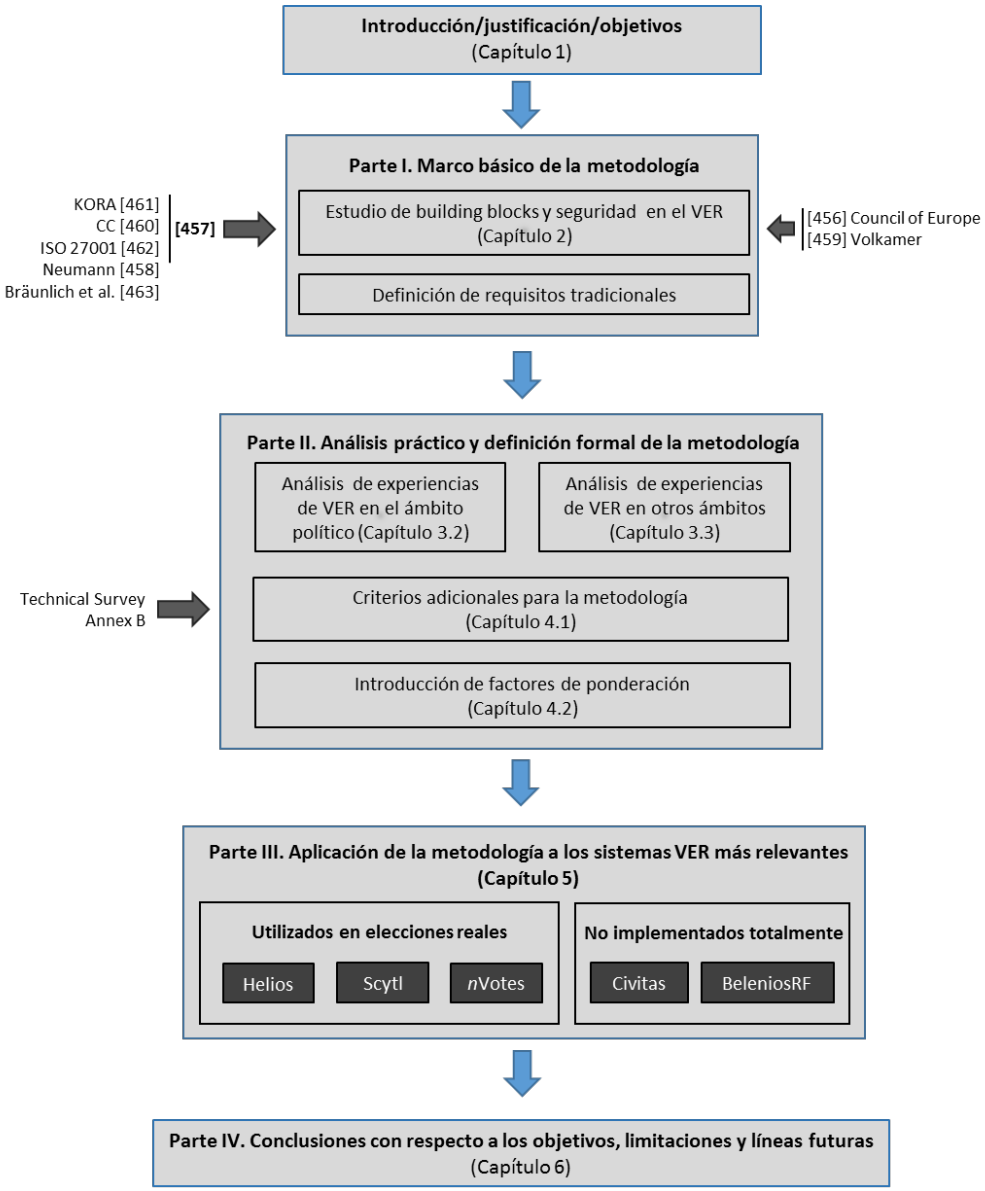


Figura 1. Estructura de la tesis

Todos los links de la presente tesis han sido comprobados y se encuentran activos a 28 de enero de 2018.

Parte I

Marco básico de la metodología

Per aspera ad astra
A través del esfuerzo, el triunfo
A través de las dificultades, hacia las estrellas
-Anónimo

Capítulo 2

CONCEPTOS, DEFINICIONES, REQUERIMIENTOS Y SEGURIDAD DEL VOTO ELECTRÓNICO REMOTO

Ex nihilo nihil fit
Nada sale de la nada
-Parménides

2.1 Conceptos y consideraciones

Según el Consejo Europeo en [54, 55], voto electrónico es aquel en el cuál, *"al menos el voto es emitido por medios electrónicos"*.

Dicha definición es demasiado ambigua para el propósito de esta disertación y por tanto es necesario acotar lo que se entiende por Voto Electrónico Remoto (VER) en la manera en la que se va a usar el término en esta tesis:

"Aquel sistema de votación que se produce en un entorno remoto y no controlado, por medios electrónicos y en el que el voto es enviado total o parcialmente a través de una conexión a internet desde un ordenador personal o dispositivo móvil que no es una máquina construida al efecto de votar".

2.1.1 Tipologías de voto

El siguiente diagrama detalla las distintas modalidades de voto en función de si es remoto o presencial, con las consiguientes subdivisiones. Los tipos de voto considerados electrónicos según la definición del Consejo Europeo se encuentran marcados en verde.

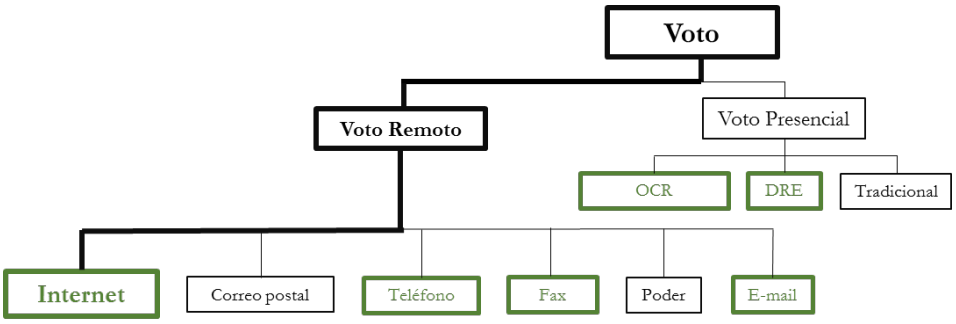


Figura 2: Tipología de voto

El objeto de estudio de esta tesis está resaltado en negrita y en verde.
Es por tanto el voto electrónico remoto a través de internet y se abreviará en todo el documento como VER.

Sin entrar en profundidad al quedar fuera del ámbito de la presente tesis, se va a definir brevemente cada uno de los tipos de voto de la figura previa.

La precondición común a todas las modalidades es que para ser considerado democrático un proceso de votación, debe cumplir con TODOS los atributos siguientes: universal, libre, igual, directo (en el sentido de que el ciudadano ejerce su derecho a voto de manera directa, sin intermediarios a no ser que la ley permita autorizar a otra persona bajo determinadas circunstancias concretas. Por norma general el derecho de voto no es un derecho transferible) y secreto, según establece la Constitución Española en su artículo 68 [66].

Voto presencial tradicional o en mesa

Es la modalidad más común en la que el votante acude al recinto electoral que tiene asignado a ejercer su derecho de voto mediante la elección de una(s) papeletas(s) física(s) con la opción seleccionada, que posteriormente introduce en las urnas habilitadas a tal fin tras haber sido identificado por el personal de la mesa electoral.

Voto (electrónico) presencial OCR

En esta tipología, a diferencia de la anterior, sí se produce un uso de tecnología en el procedimiento de votación. En concreto, el votante realiza su voto manualmente pero en la fase de recuento se utilizan técnicas de OCR (*Optical Character Recognition*) o reconocimiento óptico de caracteres.

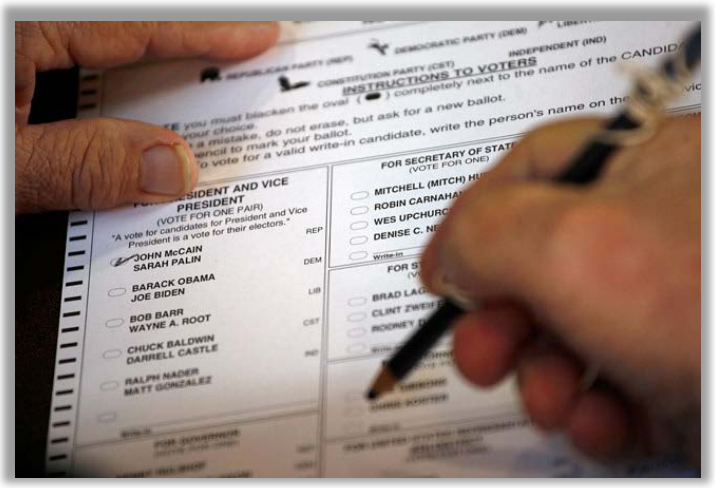


Figura 3: Ejemplo de voto presencial OCR. *Fuente: Minnesota Public Radio*

Voto (electrónico) presencial DRE (*direct-recording electronic*)

Esta modalidad de voto electrónico va un paso más allá en el uso de la tecnología, llegando al proceso de selección del/los candidato/s.

Se define como el proceso de votación a través del uso de máquinas de votación DRE (direct-recording electronic) o máquinas electrónicas de registro directo.

En ellas, se introduce el voto a través de una papeleta electrónica que aparece en un display que se controla de una manera mecánica (teclado) o electro-óptica (táctil) activada por el votante. Posteriormente, se almacenan los datos en unidades de memoria y se procesan los resultados a través de un software desarrollado a tal efecto.

Los registros de los votos se almacenan en memorias extraíbles o bien se envían de un modo seguro a otro equipamiento para proceder a su recuento. Complementariamente, las máquinas DRE pueden producir un "recibo" impreso para que el votante pueda verificar que su voto ha sido correctamente recibido y contabilizado.

Figura 4: Dispositivo de voto DRE. *Fuente: Wikipedia.*

Existe, no obstante, una gran cantidad de variables y una enorme casuística a explorar en esta tipología de votación. Se trata de un tema que ha sido estudiado muy en profundidad y sobre el que existe bibliografía de primer nivel, por ejemplo la del Dr. Panizo. Para profundidad en el tema se recomienda la lectura de [4, 11, 56].

Voto (electrónico) remoto por e-mail

Ejercer el derecho de voto a través del correo electrónico con la papeleta como archivo adjunto es una modalidad muy limitada y residual, que no obstante se ofrece a no residentes y personal militar desplazado (votantes UOCAVA) en algunos estados de los USA.

En concreto se ofrece en 24 estados [57] si bien no podría considerarse voto democrático *stricto sensu* pues no cumple con el requisito imprescindible de ser secreto.

Se sortea este problema haciendo que el votante que usa esta tipología de voto firme un documento de exención de privacidad (*secrecy waiver*) en el que explícitamente renuncia a que su voto sea secreto.

Se trata de una solución de encaje legal delicado y en cualquier caso desaconsejada por la comunidad científica tal y como apunta David Jefferson, investigador del *Lawrence Livermore National Laboratory* de California así como Presidente del Consejo de la fundación *Verified Voting* [12].

Voto (electrónico) remoto por fax

Al igual que en caso anterior del voto remoto por email, se trata de una tipología muy limitada en su uso y que adolece de cualquier mínima garantía de privacidad.

Actualmente se ofrece en 7 estados de los Estados Unidos [57] para votantes incluidos en la UOCAVA y como en la tipología previa, requiere de una renuncia expresa del votante a la privacidad de su voto (*secrecy waiver*).

Voto (electrónico) remoto por teléfono

El voto por teléfono aporta la ventaja de que la implantación de la tecnología de voto es tan intensa que se podría llegar a un porcentaje muy alto de la población.

No obstante, de nuevo las enormes carencias en materia de privacidad hacen que su uso sea muy limitado.

Las experiencias más destacables han tenido lugar en Canadá, en las elecciones locales de la ciudad de Halifax y en las elecciones del candidato del partido Nueva Democracia de Saskatchewan y en Australia, en los estados de Victoria y Nueva Gales del Sur. En todas ellas fue una opción añadida al voto por internet, si bien su uso fue muy minoritario [58].

Voto remoto por delegación de poder

En esta modalidad de voto, el votante cede su derecho de voto en otra persona, que es la que introduce su papeleta en la mesa electoral correspondiente.

El representante elegido debe de ser de la suficiente confianza puesto que al ser secreto el voto, no hay forma de probar que la persona que ha recibido la delegación de voto haya votado de acuerdo a las indicaciones del votante ausente.

La delegación de voto debe ser asimismo individual. No son posibles las delegaciones del tipo "Los votantes A, B, C, D y E delegamos 5 votos en F".

El voto remoto por delegación de poder se permite entre otros en los siguientes países: Albania, Argelia, Canadá, China, Gabón, Guyana, India, Irak, Rusia, Estados Unidos y Reino Unido.

Voto remoto por correo postal

Se trata de una modalidad de voto a distancia en la que las papeletas se envían a los electores y éstos proceden a ejercer su derecho a voto haciendo su elección y enviándola por correo postal a la autoridad competente.

Se permite votar por correo bajo ciertas circunstancias en España, Filipinas, Reino Unido y Malasia entre otros si bien ha habido multitud de polémicas que han puesto en entredicho su uso.

Los Estados Unidos y Suiza son dos casos especiales del voto por diversas razones. En el caso de Suiza, se envían las papeletas por correo postal a todo el censo electoral y con posterioridad es el votante el que decide si enviar su voto por correo ordinario o depositarlo en las urnas en los días habilitados para ello.

En cuanto a los Estados Unidos, la ley electoral cambia sustancialmente de estado a estado, lo cual provoca que en algunos de ellos el voto por correo no esté permitido (Tejas, Luisiana, Arkansas, Tennessee, Arkansas, Indiana y Virginia Occidental) mientras que en otros únicamente dicha modalidad está autorizada (Oregon, Washington y Colorado).

Voto (electrónico) remoto por internet

El análisis pormenorizado y el desarrollo de una metodología de evaluación de las soluciones más relevantes presentadas hasta la fecha constituyen la principal motivación de esta disertación.

Las denominaciones alternativas de *e-voting* o *i-voting* suelen tener un significado variable dependiendo de la fuente que se consulte. De la misma manera, al haberse estandarizado su uso también entre los medios de comunicación generalistas, han ido adquiriendo multitud de matices que hacen muy difícil un uso riguroso y sistemático.

Se van a cubrir todos sus aspectos pormenorizadamente en los capítulos venideros. No obstante, se procederá a una somera introducción en las siguientes líneas.

De manera simplificada, el voto electrónico remoto por internet se define como:

"Aquel sistema de votación que se produce en un entorno remoto y no controlado, por medios electrónicos y en el que el voto es enviado total o parcialmente a través de una conexión a internet desde un ordenador personal o dispositivo móvil que no es una máquina construida al efecto de votar".

En esta tesis, la utilización del acrónimo VER es equivalente a la definición precedente.

2.1.2 Otras consideraciones

Una vez establecida una definición válida, se van a repasar brevemente una serie de cuestiones que, sin ser el objeto principal de estudio de esta tesis, conviene destacar puesto que contribuyen a la complejidad de un análisis pormenorizado de las experiencias previas a nivel internacional así como de las soluciones más destacadas hasta la fecha. Dichos aspectos son:

a) Estándares del VER
b) Modelos de elecciones por número y orden de candidatos
c) Obligatoriedad del voto
d) Adaptación de las propiedades insustituibles del voto democrático al VER.

a) Estándares del VER

Por extraño que parezca, hasta fechas muy recientes no había ningún estándar reconocido internacionalmente para las cuestiones que conciernen al VER. De hecho, aún hoy en día los acuerdos de cara a una estandarización son menores.

Como cabe esperar, dicha falta de estandarización ha contribuído decisivamente a la existencia de una amalgama de normas, pseudoestándares y definiciones que dificultan un avance organizado de las investigaciones en la materia.

El panorama ha mejorado con la reactivación del comité 1622 del IEEE en 2015. [23, 62]. El VSSC/1622 como es conocido, crea estándares y guías alrededor de un formato común para datos electorales. Su responsable es John Wack y entre sus miembros hay profesores e investigadores destacados en el campo del VER.

Es de esperar que en años venideros la renovada actividad del VSSC/1622, junto con los incipientes trabajos de evaluación de sistemas de VER [458, 461, 463] contribuyan a mejorar la tendencia hacia una mayor homogeneidad en los estandares internacionales del VER. Aún así, queda mucho camino por recorrer. De ahí la necesidad de la presente tesis.

b) Modelos de votación por número y orden de candidatos

Los votantes de un determinado territorio están acostumbrados a la tipología de voto autóctona. En realidad, la gran mayoría de ellas se pueden encuadran dentro una de las siguientes tipologías, organizadas de menor a mayor complejidad a la hora de trasladarlas a modelos matemáticos son:

- Sí/No: Elecciones de tipo referendo en las que únicamente cabe una respuesta afirmativa o negativa.
- 1 de *N*: El votante debe elegir únicamente un candidato/lista de candidatos entre *N* opciones.
- *M* de *N*: El votante debe escoger *M* opciones entre *N* posibles. ($M \geq 1$ y $M \leq N$).
- *M* de *N* ordenados: En este caso, el votante tiene que elegir M opciones de las *N* disponibles estableciendo también su orden ($M \geq 1$ y $M \leq N$).

Este punto posee una gran importancia en lo que se refiere al uso de un sistema de encriptación/decriptación u otro, pudiendo llegar a hacer inservible en la práctica una opción para el cuarto de los modelos presentados.

En el punto 2.2.4 y subapartados se aborda la exposición y análisis de los distintos esquemas del VER y su eficacia en función de la tipología de votación. En concreto en el punto 2.2.4.11 se extraen las principales conclusiones y usos recomendados para cada uno de los tres sistemas más utilizados (firma ciega, encriptación homomórfica y mix-nets).

c) Obligatoriedad del voto

De acuerdo al CIA World Fact Book, en el mundo hay 32 países en los que el voto es obligatorio por ley y 15 en los que se aplican activamente sanciones a las personas que no ejercen su derecho a voto (en 2 de ellos únicamente en determinadas regiones) [65].

Los países en los que el voto es obligatorio pero en la práctica no se sanciona a quien no cumple con la obligación son: Bélgica, Bolivia, Costa Rica, Congo, Egipto, Gabón, Grecia, Guatemala, Honduras, Líbano, Libia, Méjico, Panamá, Paraguay, República Dominicana, Tailandia y Turquía.

Los países en los que es obligatorio y sí se aplican sanciones a quien no cumple con su obligación son: Argentina (entre los 18 y los 70 años), Australia (mayores de 18 años. En Tasmania se multó con 26 dólares en 2010 a 6.000 personas que no ejercieron el derecho a voto), Brasil (no analfabetos de 18 a 70 años), Chipre, Ecuador (no analfabetos de 18 a 65 años), el estado de Guyarat en la India (desde junio de 2015), Liechtenstein, Luxemburgo (hasta los 70 años), Malasia, Corea del Norte, Nauru, Perú (entre los 18 y los 70 años), Singapur, Uruguay y el cantón de Schaffhausen de Suiza.

La obligatoriedad del voto no es un asunto baladí, puesto que los sistemas VER podrían por una parte contribuir a facilitar el cumplimiento de dicha obligación pero al mismo tiempo podría carecer de sentido en regiones donde las infraestructuras de telecomunicaciones o la penetración en el uso de ordenadores con conexión a internet sean bajas y otras opciones como centros de voto móviles pudieran tener una mayor utilidad.

En cualquier caso, la obligatoriedad o no del derecho a voto es un factor a tener en cuenta para la implantación de un sistema de VER en un determinado territorio.

d) Adaptación de las propiedades insustituibles del voto democrático al VER

Como se ha visto con anterioridad en este mismo apartado, el voto democrático tiene una serie de características inherentes e imprescindibles según establece la Constitución Española de 1978 en varios de sus artículos [66].

Por ejemplo, el punto uno del artículo 68 establece que "*El Congreso se compone de un mínimo de 300 y un máximo de 400 Diputados,* ***elegidos por sufragio universal, libre, igual, directo y secreto****, en los términos que establezca la ley".*

El reto por tanto es trasladar dichos requerimientos *sine qua non* al VER.

Los citados 5 requerimientos se organizan en dos grupos principales:

1. Voto universal, libre, igual y directo: La propiedad del VER que nos permite comprobar que el voto es universal, libre, igual y directo es la verificabilidad extremo a extremo (E2Ev). En determinadas publicaciones denominan integridad al conjunto de características que vienen probadas a través de la E2Ev [67].

   En el apartado 2.2 se desarrolla con mayor profundidad el concepto de E2Ev y se detallan varias de las definiciones más utilizadas [93, 94].

2. Voto secreto: privacidad: Tras satisfacer 4 propiedades (voto universal, libre, igual y directo) de una votación democrática a través de la verificabilidad extremo a extremo (E2Ev), quedaría por cumplir la quinta propiedad: voto secreto.

   El requerimiento de voto secreto se corresponde con la privacidad, la cual se complica notablemente en el caso del VER puesto que como se apuntó en su definición, la votación de produce en un entorno no controlado.

   Los trabajos de [21, 52, 53, 69, 70, 71] llevaron a una categorización en niveles dentro de la privacidad:

   - Privacidad del voto: El voto de un votante no es revelado a nadie.
   - Ausencia de recibo: Un votante no puede obtener información que pueda probar a un atacante/coercionador cómo votó.
   - Resistencia a la coerción: Un votante no puede colaborar con un atacante/coercionador para obtener información sobre cómo votó (incluso si el propio votante quisiera voluntariamente vender su voto a un atacante).

En fechas recientes ha habido varios intentos interesantes de satisfacer la resistencia a la coerción tales como Civitas y BeleniosRF [72, 73]. Ambas poseen un apartado dedicado en el capítulo 5 debido a su relevancia.

Cabe indicar que, pese a que están produciendo avances significativos, ninguna de las citadas soluciones ofrece las suficientes garantías en elecciones a gran escala por lo que su aplicación real llevará todavía algún tiempo.

En el apartado 2.2.3 se estudia en detalle la resistencia a la coerción y sus avances más significativos.

Por tanto y en lo referente a la adaptación de las propiedades insustituibles del voto democrático a soluciones de VER, se requiere que dicha solución sea verificable extremo a extremo (E2Ev) y resistente a la coerción. En el apartado 2.3 se desarrolla en profundidad la selección de cada requisito.

2.1.3 Esquema básico y actores principales del VER

Una vez establecida una definición válida para el VER y las características y exigencias básicas inherentes al mismo, se está en disposición de proponer un esquema básico de VER con sus actores necesarios.

El esquema propuesto en esta sección es genérico en la medida en que podría considerarse como un punto de partida sobre el que cada una de las soluciones propuestas en el capítulo 5 va a ahondar y tratar de perfeccionar de una manera propia.

Existe multitud de bibliografía sobre esquemas simplificados de sistemas de VER [12, 74, 75]. Para esta breve explicación se va utiliza una variante del esquema de Gjøsteen [74].

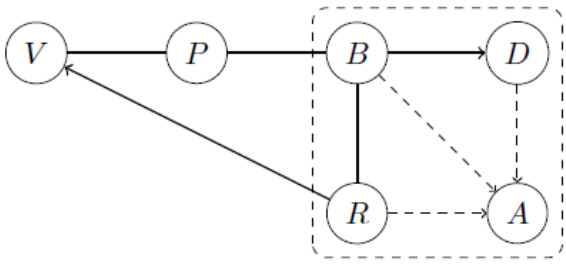


Figura 5: Protocolo simplificado del VER. Comunicaciones entre participantes [74]

En este esquema hay un votante V, un ordenador del votante P y una infraestructura (representada por un cuadrado dibujado con guiones) compuesta por una urna B, un generador de recibos R, un servicio de desencriptación D y un auditor A.

El votante V elige una serie de opciones (v_1,...v_k) de un conjunto $\mathcal{O} = \{1,2\ldots\}$; el ordenador rellena los votos con ceros hasta una longitud estándar k_{max}, lo encripta con un protocolo de encriptación y envía el voto encriptado a la urna B.

La urna B, junto con el generador de recibos R computa una serie de códigos de recibo que se envían directamente al votante. El votante tiene una tabla de correspondencia de opciones y códigos de recibo. Si los códigos enviados coinciden con las opciones seleccionadas, el votante sabe que su voto es correcto. Si no, ello indica que algo ha ido mal, lo notifica y se invalida el voto.

Cuando cierra la urna B, envía los votos encriptados al servicio de desencriptado D, el cual desencripta los votos y publica el resultado.

Los intercambios de mensajes entre las partes se detallan en la siguiente figura:

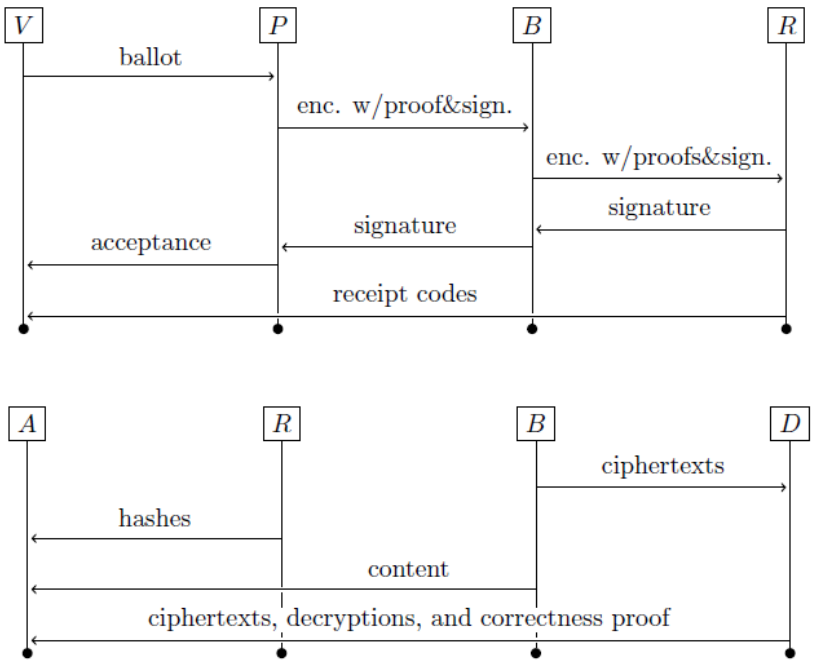


Figura 6: Protocolo simplificado del VER. Intercambio de mensajes entre las partes [74]

En la figura anterior el votante V comunica al equipo P el voto a enviar y le permite firmarlo. El votante espera por un mensaje de aceptación y por los códigos de recepción.

El ordenador P encripta el voto y firma el texto cifrado. Después, espera por la firma del generador de recibos R sobre un *hash* del voto encriptado antes de enviar el mensaje de aceptación al votante.
La urna B recibe los votos encriptados y firmados del ordenador P. Computa los cifertextos de los códigos de recibo y prueba al generador de recibos R que sus computaciones son correctas.

Cuando se cierra la urna, ésta decide los votos encriptados que deben ser desencriptados y los envía al servicio de desencriptado D y revela el contenido completo de la urna al auditor A.

El generador de recibos R verifica la firma del votante y cada prueba, genera los códigos de recibo y los envía directamente a cada votante. Además, firma un *hash* del voto y devuelve dicha firma a la urna. Sin esa firma, el ordenador P no informará al votante V de que el voto ha sido aceptado.

El servicio de desencriptado D desencripta los ciphertextos entrantes, baraja/mezcla las desencriptaciones y prueba al auditor que los ciphertextos de entrada contienen una mezcla de las desencriptaciones. Esto se realiza utilizando una mix-net cuya explicación formal se realiza en el punto 2.2.4.8.

El auditor A por su parte recibe el contenido de la urna B al completo y una lista de *hashes* de votos encriptados enviados por el generador de recibos R.

Posteriormente verifica que no hay votos agregados de manera incorrecta ni se han perdido votos por el camino. Una vez concluida esa tarea, computa él mismo otra lista de votos encriptados que son los que serán recontados, tras haberlos comparado con los ciphertextos *input* del servicio de desencriptación D.

En el siguiente apartado 2.2, se profundizará en las definiciones y las bases matemáticas y criptográficas más relevantes en el VER, así como en las principales técnicas y algoritmos más utilizados para tratar de satisfacer los requerimientos de E2Ev y privacidad. A su conjunto se le denomina *building blocks*.

2.2 Definiciones y *building blocks*

En el presente punto se van a pormenorizar las principales técnicas y conceptos criptográficos y matemáticos (*building blocks*) usados en el resto de la tesis.

Al tratarse de la base sobre la que se asientan los sistemas de VER a examinar con posterioridad, las explicaciones son extensas y en detalle, puesto que los *building blocks* aquí detallados van a ser referenciados en multitud de ocasiones en esta tesis.

Es interesante destacar que, como se verá con posterioridad, varias de las soluciones matemáticamente seguras y que sirven de base para una importante cantidad de soluciones de VER, demuestran ser en la práctica vulnerables a distintos tipos de ataques [87, 90, 91, 92], si bien los recursos necesarios para llevarlos a cabo son en general elevados y sólo al alcance de agencias estatales de los países que más presupuesto dedican a ciberseguridad.

2.2.1 Fases principales de un proceso electoral sobre una plataforma de VER

Con el objeto de establecer un marco común a los conceptos matemáticos y criptográficos posteriores, en primer lugar se definen las fases estándar de un proceso electoral sobre un sistema de VER de acuerdo con [23]:

1. **Preparación**

   Llevada a cabo por la(s) autoridad(es) electoral(es). Incluye la actualización del censo, el diseño de los votos, el envío de información previa/notificaciones a los votantes que lo requieran, la formación de las mesas electorales, la logística, el recuento de votos, la organización de los medios humanos y materiales etc.

2. **Registro/Autenticación**

   Dependiendo de la solución de VER, se utilizarán distintos mecanismos para el envío de credenciales a los votantes.

   Puede incluir desde una serie de códigos por correo postal [262] a un correo electrónico con instrucciones de acceso y códigos de votante asociados [1, 425] o incluso a combinaciones de seguridad con SMS y códigos QR asociados [249].

3. **Votación**

   Fase en la que el votante rellena su voto, normalmente a través de un cliente de software instalado o no en su propio ordenador y lo encripta (normalmente con su clave privada y la pública de la elección).

4. **Envío/depósito del voto**

   Las autoridades electorales reciben los votos. Al igual que en el punto 2, hay multitud de formas de envío de los votos, con distintas propiedades matemáticas y criptosistemas asociados. Los más relevantes se detallarán en este mismo apartado 2.2.

5. **Recuento**

   En este caso también existe variación entre las propuestas de VER, puesto que es imprescindible que el resultado sea el correcto (integridad) pero se debe a su vez garantizar resistencia a la coerción (privacidad reforzada). Sobre las distintas opciones se profundiza en los apartados 3.2 y 5. Principalmente se utilizan propiedades homomórficas (apartado 2.2.4.6), esquemas de mixnets (apartado 2.2.4.8) o esquemas mixtos.

6. **Auditoría**

   Como prueba de la integridad de los resultados y a disposición de quien lo pueda requerir, debe de existir una auditoría independiente al proceso de votación para garantizar las propiedades imprescindibles del VER. Es uno de los requisitos de la metodología de evaluación (apartado 2.3e para más detalles).

2.2.2 Verificabilidad extremo a extremo (E2Ev)

Los protocolos de VER comportan un conjunto de riesgos inherentes que pueden dar pie a ataques como se detalla en el apartado 2.4.

Dichos riesgos incluyen el propio dispositivo del votante, los nodos intermedios de comunicación entre el votante y el servidor que ejerce de urna, la propia implementación del sistema de VER etc. Todo ello con el condicionante de que la gran mayoría de los votantes no son expertos en seguridad informática (ni deberían serlo).

Por otra parte, el sistema de VER debe proteger simultáneamente la integridad y privacidad de las elecciones y sus votantes; en principio antagónicas según [23, 1, 260, 388].

Por lo que respecta a la integridad, la única forma de conseguirla es a través de una capacidad de verificación con una doble vertiente:

- El votante debe de ser capaz de comprobar que su voto ha sido correctamente recontado pero sin dar pistas de la opción elegida para evitar que pueda servir como prueba frente a un coercionador.

- Las autoridades también necesitan saber que el resultado de las elecciones es el correcto, por lo que demandan algún tipo de evidencia de que el recuento es válido. Por ello, es imprescindible que el proceso del VER sea E2Ev.

El concepto de E2Ev es fundamental para esta tesis: *"Un sistema de VER es E2Ev si cada voto es i) emitido como estaba previsto ii) guardado como se ha emitido y iii) contado como se ha guardado."* [93, 51, 77, 3, 359, 369].

Al tratarse de un estándar en el mundo del VER, se reproducen las tres condiciones previas en su versión original en inglés:

i) Cast as intended: Los votantes pueden tener acceso a una evidencia convincente de que su voto encriptado refleja fielmente su elección. Se requiere por tanto que el proceso de votación utilice canales seguros de comunicación para garantizar que ningún ataque cambie el voto y que todo ello sea verificable.

ii) Recorded as cast: Los votantes o los delegados pueden comprobar que sus votos han sido debidamente incluidos, siendo verificables los valores encriptados de los votos en una lista pública o *Bulletin Board* de votos encriptados. En este caso se suele facilitar al votante algún tipo de información a modo de "recibo" para posteriormente poder comprobar en el listado que el voto fue registrado correctamente.

iii) Counted as recorded: Cualquier parte puede verificar que todos los votos encriptados publicados son correctos y se han incluído en el recuento, si bien no hay forma de sabe cómo voto ningún votante. Para ello se utilizan mix-nets o cifrado homomórfico (apartados 2.2.4.8 y 2.2.4.6 para más detalles).

Para profundizar en cada característica, referirse a [93, 3, 51, 77, 93, 172].

De una manera más gráfica, tomada de la tesis de Sandra Guasch [369]:

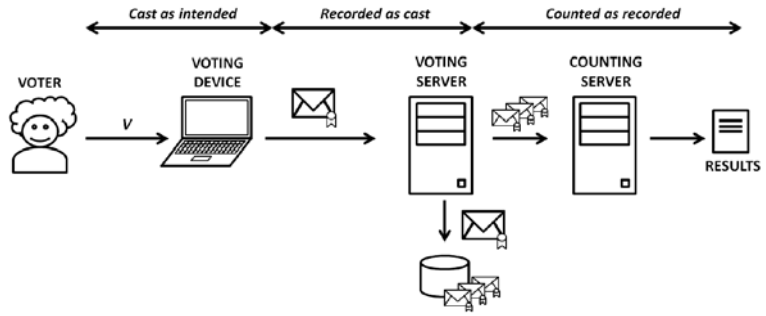


Figura 7.: Tipos de verificabilidad en un sistema de VER [369]

En la figura anterior, a las dos primeras fases (*"cast as intended"* y *"recorded as cast"*) se las suele denominar conjuntamente verificabilidad individual.

Paralelamente, cuando termina la fase de votación, comienza la parte denominada *"counted as recorded"*, por la que cualquier entidad puede verificar que el recuento representa el contenido de los votos enviados por los votantes (verificabilidad universal).

Dichas características, junto con la denominada *verificabilidad de eligibilidad* introducida por Kremer et al. [389] (los votos han sido enviados por votantes con derecho de voto) conforman la verificabilidad extremo a extremo o E2Ev.

Aún así, lo anteriormente dicho no es suficiente: los verificadores y cualquier votante tienen que ser capaces de comprobar esas tres condiciones independientemente del software utilizado en el sistema de VER. [93, 23]. Es la única manera de mantener la confianza en el sistema de VER aunque se sospechase que alguno de los actores no es honesto.

Si únicamente usuarios avanzados o con determinados conocimientos pueden verificar su resultado o el de las votaciones, entonces no se trata de un sistema verdaderamente E2Ev.

Últimos avances en la verificabilidad de sistemas de VER

Pese a su enorme importancia, tradicionalmente la privacidad ha atraído una mayor atención en lo que respecta a los requerimientos de un sistema de VER. Hasta fechas bastante recientes no se ha tratado de avanzar hacia un estándar sobre verificabilidad tal y como apuntan Jonker et al. [359]. Unicamente a partir de 2015 han ido produciéndose avances de la mano de Cortier et al. [362, 365, 366] y Kulyk et al. en [360].

La finalidad ha sido doble: por una parte tratar de llegar a una definición formal y universal de la verificabilidad y por otra intentar automatizar algún tipo de prueba para evaluarla.

Actualemente ninguno de los dos problemas está resuelto [362], en buena parte por unas suposiciones de seguridad en ocasiones difícilmente transladables a la realidad (el *Bulletin Board* es siempre honesto, así como los votantes y los dispositivos etc).

Aún así, una de las líneas de investigación más activa en fechas recientes es el esquema KTV [390], que se toma como base para una definición genérica de la verificabilidad:

Sea $\mathcal{P}$ un protocolo con un conjunto de agentes Σ.

Sea $\delta \in [0,1]$ la tolerancia, $\mathcal{J} \in \Sigma$ el juez y γ una meta.
En ese caso, se dice que el protocolo $\mathcal{P}$ es (γ, δ)-verificable por el juez $\mathcal{J}$ si para todos los adversarios π_A y $\pi = (\pi_P \parallel \pi_A)$ la probabilidad:

$$\Pr[\pi^{(l)} \mapsto \neg\gamma, (\mathcal{J}: accept)]$$

es δ-acotada como una función de l.

Para el lector que quiera profundizar en el esquema KTV, se le recomiendan las siguientes referencias bibliográficas: [390, 365, 366].

Si bien se están produciendo interesantes avances en la materia, basados en el citado esquema, todavía queda pendiente la automatización de las pruebas en modelos computacionales como apunta Cortier [362].

El principal motivo es que las herramientas actuales para el análisis simbólico de protocolos de seguridad tales como *ProVerif, Scyther* o *Avispa* no tienen buena aplicación para protocolos de VER. Entre las razones más importantes destaca el hecho de que los sistemas de VER utilizan primitivas criptográficas no estándar y que incluyen operadores asociativos y conmutativos, actualmente fuera del alcance de cualquier herramienta de análisis de protocolos de seguridad.

Sirva como ejemplo la definición del sistema de voto desplegado en Noruega en 2011, cuya representación en ecuaciones simbólicas es la siguiente [362]:

$$renc\left(enc\left(x_{plain}, x_{rand}, pk(x_{sk})\right), y_{sk}\right) = enc(x_{plain}, x_{rand}, pk(x_{sk} + y_{sk}))$$

$$dec\left(blind\left(enc\left(x_{plain}, x_{rand}, pk(x_{sk})\right), x_{blind}\right), x_{sk}\right) = blind(x_{plain}, x_{blind})$$

En las circunstancias actuales, dichas ecuaciones simbólicas están fuera del alcance de las herramientas de análisis de protocolos de seguridad.

Por ello, todavía falta algún tiempo antes de poder estandarizar y modelizar una definición universal de verificabilidad que pueda ser automatizada para su análisis en sistemas de VER. Hasta entonces, la evaluación de la E2Ev se debe realizar caso por caso dependiendo de la definición aplicada en cada sistema de VER.

Para una profundizar más en el concepto de E2Ev, su evolución temporal y variaciones se recomiendan las siguientes referencias: [50, 51, 3, 93, 77, 359, 360, 362, 365, 369, 390].

2.2.3 Resistencia a la coerción (RC)

Como se ha visto en el apartado anterior, la privacidad es una de las cualidades obligatorias en una votación democrática. En los sistemas de VER, el hecho de que la votación se lleve a cabo en un entorno remoto y no controlado contribuye a aumentar la dificultad de garantizar la privacidad del votante y del voto.

El VER no introduce el problema de la coerción y/o venta del voto (se encuentra presente en las otras tipologías de voto) pero facilita la labor a un potencial atacante. Por ejemplo, en el caso de que un votante quiera voluntariamente vender su voto, lo puede hacer sin ni siquiera tener que desplazarse.

Por ello, garantizar un suficiente nivel de privacidad reforzada es un factor indispensable a la hora de evaluar si una solución de VER cumple con los requisitos para ser desplegada en elecciones VAP reales.

Retomando lo dicho en el punto 2.1, las soluciones de VER están categorizadas en 3 niveles de menor a mayor exigencia:

- Privacidad del voto: El voto de un votante no es revelado a nadie
- Ausencia de recibo: Un votante no puede obtener información que pueda probar a un coaccionador cómo votó.
- Resistencia a la coerción (RC): Un votante no puede colaborar con un coaccionador para obtener información sobre cómo votó (incluso si el votante quiere voluntariamente vender su voto a un atacante).

El votante puede por supuesto transmitir al coaccionador su voto, pero éste no tiene forma de comprobar si la afirmación es verdad o no.

Otra definición ampliamente utilizada es: *"el votante no puede probar cómo voto o si votó, incluso en el caso de que pueda interactuar con el coercionador mientras vota [63]"*

Hasta mediados de los 2000 se tendía a considerar que la segunda propiedad (ausencia de recibo) era suficiente para un sistema de VER. Posteriormente, aparecieron varios esquemas ofreciendo ausencia de recibo a nivel teórico como [107] que posteriormente se demostraron insuficientes [108].

Fueron Juels, Catalano y Jakobsson quienes introdujeron formalmente el concepto de resistencia a la coerción en 2002 [70], depurándolo en 2005 [63] y 2010 [104].

Con ello pretendían solucionar el problema que se podía dar en sistemas de VER basados en mix-nets, en los que el coaccionador podía simplemente forzar a un votante a introducir un voto cifrado suministrado por él y luego comprobar que se encontraba dentro de la lista de votos cifrados en el Boletín de votos contabilizados o *Bulletin Board.*

Pese a que el esfuerzo fue notable, en su protocolo se parte de una serie de prerrequisitos que no son asumibles en unas elecciones reales [104]:

- Se asume que no hay corrupción posible en la fase de verificación del votante.
- Este recibe las credenciales por un canal incorruptible.
- Los generadores de recibos tampoco son corruptibles
- No se tienen en cuenta ataques DoS ni la verificación por parte del votante de que su voto ha sido tenido en cuenta.
- El coste computacional de recuento es cuadrático respecto al número de votantes por lo que su uso se limita a elecciones con un reducido número de participantes.

Otro de los problemas de los protocolos que intentan conseguir resistencia a la coerción es que toman como base la abstracción Dolev-Yao [109, 53], asumiendo pues que las primitivas criptográficas son perfectas. La práctica ha demostrado que esto no es cierto y ha dado lugar a varios ataques, incluso en fechas recientes [26, 27, 34, 464].

Otra línea de investigación muy relevante sobre la resistencia a la coerción la inició el gurú de la ciberseguridad Josh Benaloh [3]. Él mismo es el autor de un paper en 2013 en el que apunta que el avance de la tecnología paradójicamente juega en contra de la resistencia a la coercibilidad.

En concreto, reflexiona que cualquier votante que quiera vender su voto o que sea obligado a una opción en contra de su voluntad, puede ser forzado a llevar un dispositivo portátil de grabación para probar que ha cumplido con la coerción [106]. Nuevos *gadgets* como las Google Glass facilitarían aún más la labor del atacante.

En el año 2015, Achenbach et al. han presentado una propuesta [105] de re-votación negable (en el sentido de que no hay forma para un coaccionador de verificar si el votante ha vuelvo a votar para invalidad el voto bajo coacción. El votante puede negar haber re-votado y el atacante no tiene forma de saberlo) el cuál constituye un importante paso adelante respecto al trabajo original de Juels, Catalano y Jakobsson [104] en 2010.

En [104], se protegía la identidad del votante con votos duplicados, pero el votante no podía negar el hecho de haber re-votado (el voto bajo coerción sería sobreescrito y el coaccionador, al comprobar el código cifrado, se daría cuenta de que su voto no está presente, comprometiendo al votante).

Ello se debe a que para identificar los votos duplicados, se utiliza la tecnología de *Plaintext Equivalence Tests* (PET) [110], en la que el *output* es un único bit, que puede fácilmente revisar el coaccionador.

En [105] por el contrario, se mejora el protocolo de tal manera que el votante puede ocultar el hecho de haber re-votado para anular el voto realizado bajo coacción. Ello se consigue gracias a las propiedades homomórficas de la encriptación de las credenciales del votante para comprobar si hay más de un voto con la misma credencial.

Se usan para ello *Encrypted Plaintext Equivalency Texts* (EPET). A diferencia del PET, el *output* del EPET no es un bit, sino la encriptación de un bit, con su correrspondiente *padding* o relleno, por lo que el atacante no tiene forma de saber si el votante ha vuelto a votar o no (re-votación negable).

De una manera formal y siguiendo con la nomenclatura de [105]:

Un voto b_i es una tripla del tipo:

$$b_i = (Enc(\beta_i), Enc(pk_i), ts_i), NIZKPoK$$

Donde $Enc(\beta_i)$ es la encriptación de la opción válida β_i, $Enc(pk_i)$ es la encriptación de la clave pública del votante pk_i y ts_i es un *timestamp* o fecha y hora creada justo antes de que se envíe el voto y que no está encriptada.

A mayores, $NIZKPoK$ es una prueba de conocimiento cero sobre la firma de $(Enc(\beta_i), Enc(pk_i), ts_i)$ con respecto a la clave pública encriptada en el voto. De una manera más formal, el votante demuestra el conocimiento de una firma σ_i:

$$verify(\sigma_i, (Enc(\beta_i), Enc(pk_i), ts_i), pk'_i) = 1 \wedge pk'_i = pk_i$$

Por lo que respecta a la limpieza de los votos emitidos por un mismo votante, tras comprobar que todos los votos tienen una $NIZKPoK$ probando que están bien formados, se ordenan ascendentemente de acuerdo a su campo *timestamp* ts_i.

Para cada voto b_i, la autoridad de recuento verifica si la clave pública encriptada pk_i de b_i coincide con otra clave pk_j de otro voto b_j más reciente. Es decir,

$$\forall\, b_j\ (i < j < n)$$

la autoridad lleva a cabo la *EPET* distribuida para obtener

$$(d_{i,j}, \prod_{i,j}) := Epet(Enc(pk_i), Enc(pk_j))$$

y se publican $d_{i,j}$ y $\prod_{i,j}$ ($d_{i,j} = Enc(1)$ iff $pk_i = pk_j$)

Para la explicación completa del protocolo, referirse al paper original [105].

Aún tratándose de un nuevo paso adelante, [105] todavía parte de una serie de premisas (análogas a las de [104] además de otra referente al reloj del ordenador desde el cuál se vota) que son difícilmente aplicables a unas elecciones reales. Por ello, el sistema tal y como está explicado requiere de una mayor investigación y pruebas en entornos reales antes de que su uso pueda estandarizarse.

Por todo lo explicado en estas líneas y lo que implica sobre una propiedad imprescindible del voto como es la privacidad, la resistencia a la coerción se ha convertido en uno de los principales caballos de batalla en el desarrollo de soluciones para el VER. Como se ha visto, se están produciendo avances interesantes en los últimos tiempos, si bien todavía no hay ningún sistema que pueda garantizar su total cumplimiento.

De entre las soluciones analizadas en el capítulo 5, la comunidad científica suele destacar Civitas [67] y BeleniosRF [72] como dos de los mejores intentos académicos de conseguir resistencia a la coerción.

Finalmente, se adelanta que la RC es un requisito *sine qua non* para que una solución de VER sea considerada apta para su uso en elecciones reales vinculantes en el ámbito político junto con la E2Ev, como se explica en el apartado 2.3.

El motivo es que se trata de la característica que representa la propiedad del voto secreto recogida en el artículo 68 de la Constitución Española: *"El Congreso se compone de un mínimo de 300 y un máximo de 400 Diputados, elegidos por sufragio universal, libre, igual, directo y secreto"*.

2.2.4 *Building Blocks* Criptográficos

En el presente apartado se van a presentar los conceptos y protocolos criptográficos (*building blocks*) que van a ser el utilizadas a lo largo de la tesis.

En las explicaciones se le suponen al lector unos conocimientos previos de álgebra, cálculo y criptografía a nivel de ingeniero en informática. En cada uno de los conceptos se facilita bibliografía asociada para profundizar más allá de las explicaciones aquí ofrecidas y directamente relacionadas con el interés de esta tesis.

El apartado es extenso puesto que para desarrollar una metodología exhaustiva de evaluación de sistemas de VER, los criterios así como su selección tienen que estar fuertemente sustentados en las bases y conceptos criptográficos que se detallan.

2.2.4.1 Funciones *Hash* Criptográficas

La importancia de las funciones *hash* en el VER es relevante puesto que en la mayoría de las soluciones en las que se encripta el voto con un sistema de clave pública, en realidad lo que se está encriptando es un *hash* del voto.

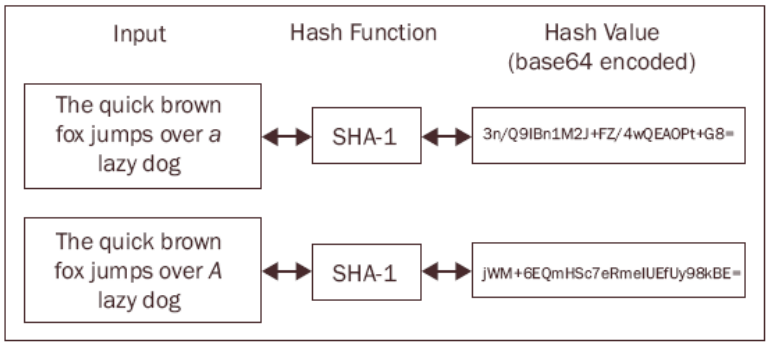


Figura 8: Ejemplo Función *hash* criptográfica. Fuente: Oxford and Manchest. Paradigm Project

En general, una función *hash* es aquella que permite verificar fácilmente que un determinado valor de entrada se corresponde con un valor de salida *hash*, pero la operación inversa

(conocido el valor *hash output* reconstruir el *input*) tiene una probabilidad descartable de ser realizada en tiempo computacional polinómico.

Cuando los valores del mensaje *input* son de gran longitud, o la misma es variable, se utilizan protocolos que incluyen el troceado del *input* en unidades menores, su posterior combinación y en caso de que sea necesario, su relleno o *padding*.

Las propiedades de las funciones *hash* criptográficas son:

- Determinismo: Un determinado *input* sobre el que se ejecuta la función *hash* produce siempre el mismo *output*.
- Uniformidad: Cada valor *hash output* debe ser generado aproximadamente con la misma probabilidad. De lo contrario aumentaría el riesgo de colisión de resultados (varios *inputs* asociados a un mismo valor *hash*), provocando que el sistema fuese más vulnerable desde un punto de vista criptográfico.
- Rango definido: Para multitud de aplicaciones y las operaciones entre ellas, es deseable que el *output* de la función *hash* sea de un tamaño fijo. Para ello, como se explicó anteriormente, se divide el *input* en bloques de un tamaño predeterminado. Generalmente, la longitud del bloque es mayor que el del propio valor *hash* de *output*, para evitar ataques de fuerza bruta. En el caso del protocolo SHA-1, el valor del *hash* es de 160 bits y cada bloque tiene un tamaño de 512 bits.
- Continuidad: Para otros usos de las funciones *hash*, como optimizaciones de búsquedas, el hecho de que dos *inputs* similares den como resultado dos *hashes* similares (continuidad) es una propiedad deseable.

  No obstante, en lo que respecta a la criptografía, la continuidad es una vulnerabilidad fatal que pone en riesgo la seguridad del sistema. Lo deseable es que ante cualquier pequeño cambio en el *input*, el *output* sea lo más diferente posible. Con ello se busca dificultar la labor de los atacantes a la vez que se facilita la detección de intrusiones.
- Unidireccionalidad: Dado un *input*, calcular el valor *hash* es una operación de bajo coste computacional pero la inversa (dado un *hash* computar el *input*) tiene una probabilidad despreciable de realizarse en tiempo polinomial.

Como consecuencia de las propiedades descritas, en el caso del VER las funciones *hash* criptográficas son muy útiles al permitir comprobar si el voto emitido por el votante ha sido modificado o no (a causa de un error o un ataque).

Su uso está también muy extendido para el cifrado de mensajes. Algunos de los ejemplos más relevantes son los algoritmos SHA-1. SHA-2, MD-4, MD-5, Whirlpool etc.

La gran mayoría de las funciones *hash* son del tipo iterativo, con la estructura Merkle - Damgård y I.V. (vector de inicialización) constante [99, 102].

En la práctica, se han reportado fallos, brechas de seguridad y ataques tanto en funciones *hash* [87] como en las alternativas basadas en el *Random Oracle* (apartado 2.2.4.2).

La principal causa de los fallos encontrados es que se realizan demostraciones basadas en modelos aleatorios perfectos. Por desgracia, reproducir dichas condiciones ideales en la práctica no es posible, produciéndose colisiones en las funciones *hash* que comprometen la seguridad y dan pie a ataques de suplantación [103].

Ello se debe a que las funciones "perfectamente aleatorias", no lo son en realidad porque tratar de comprimir información totalmente aleatoria en un *hash* compacto no es posible como nos demuestra la teoría de la información.

A modo de ejemplo, para el caso del protocolo SHA1, se aceptan mensajes longitud hasta 2^{64}. En consecuencia, la tabla de correspondencia de *input* y *hashes* sería de un tamaño tal que no sería ni remotamente manejable por ningún equipo de sobremesa.

Por ello, en la práctica tienen que funcionar de una forma no puramente aleatoria para ser utilizables. Se trata de compensar dichas deficiencias formales introduciendo *inputs*, *outputs* y claves suficientemente largas para que un atacante no pueda romperlas. Adicionalmente se utilizan "trucos" tales como calcular los valores según van siendo requeridos en vez de calcular la tabla completa a priori [112].

La profundización en los distintos protocolos y construcciones de funciones *hash* se aleja del propósito de esta tesis por lo que se recomienda al lector interesado referirse a las siguientes referencias: [87, 99, 100, 101].

2.2.4.2 Modelo del oráculo aleatorio. *(Random Oracle Model)*

Se trata de un modelo ideal de computación utilizado en multitud de protocolos criptográficos, originalmente desarrollado por Belare y Rogaway en 1993 [89].

Los oráculos aleatorios se desarrollaron como alternativa ideal a las funciones *hash* criptográficas. La idea subyacente es que a diferencia éste, en el R.O.M. todos los procesos de *hashing* que haya que realizar sobre el texto de *input* los realiza el oráculo mismo tras recibir el texto de la parte interesada a través de un canal seguro.

Dicha operativa tiene la ventaja de que se descarga de tareas computacionales intensas a las distintas partes del sistema de VER, transladándolas al oráculo, que dispone de una mayor capacidad.

En el R.O.M., las funciones *hash* se modelizan como funciones aleatorias usadas por ejemplo como prueba de congruencia en pruebas de conocimiento cero cuando se utiliza la heurística de Fiat-Shamir [20, 85] para convertirlas en no interactivas.

El problema radica en que ninguna función computable por un algoritmo finito puede implementar un oráculo totalmente aleatorio y al sustituirlo por una función computable en tiempo polinómico, aparecen fallos explotables por atacantes como sucede en la implementación del protocolo RSA [113], como se indica en [98].

Al estar basadas multitud de comunicaciones seguras a través de internet en variaciones del protocolo RSA, se añade una variable más de incertidumbre a la hora de implementar soluciones de VER.

Para profundizar en el modelo el oráculo aleatorio y sus implicaciones prácticas en la seguridad de los protocolos de internet se recomienda la siguiente bibliografía: [89, 97, 98, 111, 112, 115].

2.2.4.3 Modelo de secuencia común de referencia. *Common Reference String Model* (CRS)

Su origen se remonta a 1988, con la publicación del paper de M. BLum et. al [96] para la construcción de protocolos de pruebas de conocimiento cero no interactivas (NIZK) que se estudian en el punto 2.2.4.7. (Con anterioridad, las pruebas de conocimiento cero eran interactivas e implicaban un elevado número de iteraciones, por lo que su aplicación era limitada por su elevado coste computacional).

En el modelo CRS, existe una tercera parte totalmente confiable que genera una secuencia que envía a las dos partes implicadas (la parte que quiere probar que conoce algo o *"probador"* y el verificador de que ese "algo" es lo que tiene que ser).

En [111], R. Pass realiza un estudio sobre las propiedades ideales de los modelos de oráculo aleatorio y de secuencia común de referencia, llegando a la conclusión de que en el modelo CRS sólo se pueden alcanzar dichas propiedades si no se reutilizan las secuencias de referencia. En el caso del modelo de oráculo aleatorio, Pass sí que consigue construir un argumento de conocimiento cero denegable de 2 rondas que mantiene las propiedades ideales necesarias.

También en [89, 111, 115] se intenta sustituir el oráculo aleatorio por un tipo de funciones *hash* de acuerdo al modelo CRS si bien en [115] se muestran casos en los que dicha sustitución deriva en esquemas inseguros.

Por ello, existen implementaciones de NIZK seguras en el ROM pero no en el CRS. Aun así, se ha visto en el punto anterior que llevar a la práctica las condiciones ideales del ROM sigue suponiendo un desafío que puede llegar a comprometer la seguridad de los procedimientos que implementan el modelo.

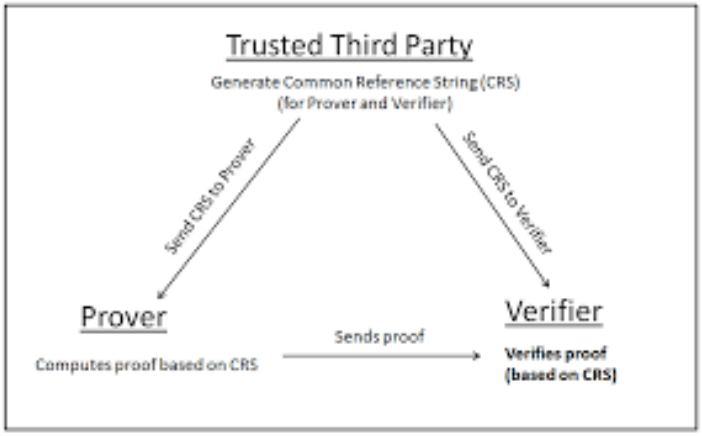


Figura 9: Esquema del modelo de secuencia común de referencia (CRS model)
Fuente: Norwegian University of Science and Technology

Para profundizar en el modelo de secuencia común de referencia o CRS, así como en su intercambiabilidad con el modelo de oráculo aleatorio y otras opciones, referirse a [96, 111, 114, 115].

2.2.4.4 Criptografía de clave pública *(Public Key Cryptography, PKC)*

También conocida como criptografía de clave asimétrica, en contraposición a la criptografía simétrica o de clave secreta, la cual se explica brevemente a continuación:

La criptografía simétrica es aquella que utiliza la misma clave para el encriptado y el desencriptado, por lo que el emisor y el receptor la deben intercambiar con anterioridad y de manera secreta.

Uno de los sistemas de criptografía simétrica más famosos es la máquina Enigma usada por el ejército alemán en la II Guerra Mundial. La ruptura de sus claves por parte del bando aliado fue una de las razones que decantaron la balanza de la guerra a su favor. Más recientemente, algunos ejemplos incluyen los algoritmos DES, 3DES, RC5, AES, Blowfish e IDEA.

Los sistemas de criptografía simétrica presentan un problema importante: la dificultad práctica de tener acceso a un canal de comunicación totalmente seguro para el intercambio de la clave a utilizar. Para un atacante es más fácil tratar de interceptar la clave que luego tratar de desencriptar los mensajes posteriores por fuerza bruta.

Otro problema destacado es el número de claves requeridas. Para *n* personas, serían necesarias $\frac{n\,(n-1)}{2}$ claves para todas las parejas de personas. Por tanto, para números *n* grandes, el sistema requiere demasiados recursos computacionales.

Por el contrario, una de las grandes ventajas de la criptografía asimétrica es que no requiere un canal seguro de comunicación para intercambiar las claves. Dicha característica, en un entorno de comunicaciones inseguro como es internet, supone una ventaja crucial.

En la criptografía de clave pública, cada participante tiene dos claves: Una pública p_k y otra privada s_k. Los datos encriptados con la clave pública sólo pueden ser desencriptados con su correspondiente clave privada.

Dicho par de claves se generan basándose en problemas matemáticos para los que no existe solución posible en tiempo polinómico, tales como: la factorización de enteros grandes, los logaritmos discretos y las relaciones de curva elíptica que se explican en apartados posteriores de este mismo punto 2.2.4.

Computacionalmente, es viable para el usuario típico generar el citado par de claves y utilizarlas para tareas de encriptado y desencriptado. La ventaja añadida es que es *"imposible"* para un atacante (en el sentido de inabarcable desde un punto de vista computacional) calcular la clave privada de un usuario a partir de su clave pública por lo que ésta última puede publicarse sin comprometer la seguridad.

El origen de los criptosistemas de clave pública se remonta a 1976, cuando Whitfield Diffie y Martin Hellman desarrollaron un protocolo de intercambio público de claves conocido como el *protocolo criptográfico Diffie-Hellman* [25], que se explica en el punto 2.2.4.10a del presente apartado.

Entre los problemas asociados a la criptografía de clave pública, se encuentra su complejidad computacional, por lo que en multitud de ocasiones se utiliza para transferir de manera encriptada la clave simétrica que se utilizará con posterioridad, debido a la menor complejidad de los logaritmos basados ésta última.

Por desgracia, la complejidad de generar primos lo suficientemente grandes para a su vez tener pares de claves lo suficientemente seguros ha llevado a prácticas arriesgadas que han puesto en peligro estándares de internet tales como RSA o TLS. Dos de los ejemplos más recientes de ataque (*FREAK* y logjam) se detallan en el apartado 2.4.2.3 [26, 32].

En cuanto a las aplicaciones prácticas de la PKC, las dos más conocidas son:

2.2.4.4.1 Encriptación de clave pública (PKE)

En este caso, el mensaje se firma con la clave pública del receptor. Con ello se garantiza que el mensaje puede únicamente abrirlo el propietario de la clave pública y privada asociada; es decir, el receptor. La encriptación pública se usa por tanto para garantizar la confidencialidad.

De una manera más formal, la encriptación de clave pública o PKE se compone de 3 algoritmos *(Gen, Enc, Dec)*.

El algoritmo *Gen* produce un par de claves pública y privada (p_k, s_k).

El algoritmo $(Enc, \cdot)$ aplicado a un texto sin codificar m devuelve un texto cifrado Ψ correspondiente a m. Por otra parte, el algoritmo (s_k, Ψ) devuelve el texto sin codificar correspondiente a Ψ.

El modelo de seguridad requiere que un adversario con acceso a la clave pública p_k no pueda distinguir entre dos textos cifrados que correspondan a distintos textos sin cifrar incluso si los textos cifrados son escogidos de manera no aleatoria. Son los llamados ataques CCA (*Chosen Ciphertext Attack*) [116, 117].

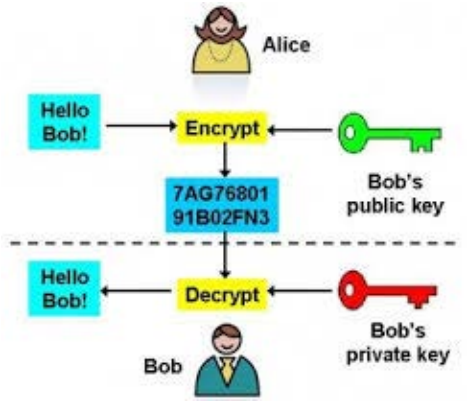


Figura 10: Esquema de la encriptación de clave pública
Fuente: Universidad de Adelaida (Australia)

Como se apuntó en 2.2.4.4, una de las vulnerabilidades de la encriptación de clave pública es su mayor coste computacional. Por ello, en la mayor parte de los casos se encripta un *hash* del texto original. (Según [118], el protocolo RSA es hasta 1000 veces más lento que DES en hardware y 100 veces más en software para modulo 512-bit)

Otra de las vulnerabilidades es la posibilidad de ataques *man in the middle* en los que el atacante sustituye la clave pública del receptor por la suya propia haciendo creer al emisor que está hablando con el receptor. Para más información, referirse a [123].

Para el lector interesado en profundizar en el tema, se recomiendan las siguientes referencias bibliográficas: [116, 117, 118, 119, 120, 121, 122, 123].

2.2.4.4.2 Firma digital

La otra gran aplicación de la criptografía de clave pública es la firma digital. En ésta, el mensaje se firma con la clave privada del emisor y lo puede desencriptar cualquier receptor que posea su clave pública (la del emisor). Ello garantiza la autoría del mensaje

así como su integridad. (Es como añadir un sello personal de lacre a un mensaje para certificar la autoría del mismo).

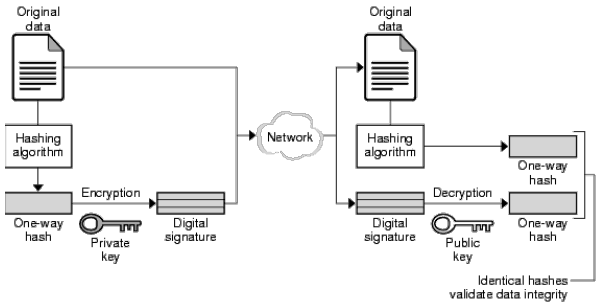


Figura 11: Esquema de la firma digital (incluyendo función *hash*). Fuente: Oracle Inc.

La gran debilidad de la firma digital reside en que si un adversario consigue hacerse con la clave privada de un usuario (como consecuencia de un ataque o un simple despiste), puede suplantarlo para cualquier mensaje a partir de ese momento.

En 1988, S. Goldwasser et. al en [124] definieron formalmente el protocolo de firma digital y desarrollaron una jerarquía de ataques que todavía se usa hoy en día. De mayor a menor criticidad son los siguientes:

1) Ruptura total o recuperación de la clave privada
2) Falsificación universal
3) Falsificación selectiva
4) Falsificación existencial

El sistema de firma digital está compuesto por 3 algoritmos:

- El algoritmo generador de claves *K*, el cuál con un *input* 1^k, produce un par de claves pública y privada (k_p, k_s). *k* es el parámetro de seguridad y *K* es probabilístico.
- El algoritmo de firma Σ. Dado un mensaje *m* y un par de claves (k_p, k_s), Σ produce una firma σ. El algoritmo de firma puede ser probabilístico.
- El algoritmo de verificación *V*, dada una firma σ, un mensaje *m* y una clave pública k_p, *V* comprueba si σ es una firma válida de *m* con respecto a k_p.

Para profundizar en la materia se recomiendan las siguientes referencias: [124, 119, 117].

Por concluir con el apartado de la PKE, reiterar su innegable aportación a la seguridad en las comunicaciones a través de internet al no requerir un canal de comunicación seguro como en el caso de la criptografía de clave secreta o simétrica.

Por otra parte, el hecho de que computacionalmente consuma muchos más recursos que la criptografía simétrica (son necesarias claves de mayor longitud) hace que en ocasiones

se usen criptosistemas híbridos, como en las elecciones de Nueva Gales del Sur [292] explicado en el apartado 3.2.5.

En ellos, la criptografía asimétrica se usa para transmitir una clave simétrica, que posteriormente es la que se utiliza en la transmisión de la información (criptografía simétrica, menos intensiva en el uso de recursos). Ejemplos de criptosistemas híbridos son el PGP y la familia de protocolos SSL/TLS.

Adicionalmente, el elevado coste computacional de la PKC ha llevado en ocasiones a reutilizar números primos en la generación de claves, dando pie a ataques masivos sobre protocolos tan utilizados como el RSA o TLS incluso en 2015 [26, 32] y que se detallan en el apartado 2.4.2.3.

Para concluir, las técnicas criptográficas más conocidas que utilizan PKC son: Diffie-Hellman, ElGamal, RSA, SSH, TLS, Paillier, Merkle-Hellman, *Internet Key Exchange* etc. Las más destacadas se explican en puntos sucesivos del presente apartado 2.2.4.

2.2.4.5 *Secret Sharing - Threshold System*

La idea de desarrollar un esquema de reparto de secretos (claves) comenzó como respuesta a la necesidad de guardar de un modo seguro las claves en los protocolos criptográficos. El esquema fue propuesto por Shamir [85] y Blakley [129] en 1979.

En este protocolo se pretende distribuir la clave s entre n participantes $(s_1,\ldots,s_n)$ de tal manera que t o más participantes ($t \leq n$) puedan reconstruir s.

Hay dos algoritmos asociados a este esquema: *(Gen, Rec). Gen* produce n partes de s $(s_1,\ldots,s_n)$ y *Rec* reconstruye s a partir de cualquier subgrupo de partes $(s_1,\ldots,s_n)$ de tamaño mayor o igual a t. Al valor t se le denomina *threshold* o umbral; de ahí la denominación.

Un sistema de reparto de secreto con umbral se considera perfecto si menos de t partes del secreto s no aportan ninguna información sobre él. A ese respecto, la implementación de Shamir es perfecta mientras que la de Blakley no lo es. Por ello, se profundiza a continuación en el sistema de reparto de secreto de Shamir [85]:

La implementación de Shamir está basada en la interpolación polinómica sobre un campo finito. Se necesitan pues k puntos para definir un polinomio de grado $k-1$.

En el algoritmo *Gen,* se tiene un secreto s de un campo Z_p siendo p primo mayor que el número de partes y mayor que cualquier $f(i)$.

El distribuidor selecciona un polinomio aleatorio de grado $k-1$

$$f(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_{k-1} x^{k-1}$$

De tal forma que

- los coeficientes $f_1,\ldots,f_{k-1}$ son elegidos aleatoriamente de Z_p
- $f_0 = s$ (f_0 es el secreto)

para $i \in \{1, \ldots, n\}$ el distribuidor envía la parte del secreto $s_i = (i, f(i)\ mod\ p)$ a la parte i.

En cuanto al algoritmo *Rec*, el secreto *s* puede ser reconstruido a partir de cualquier subgrupo de *k* elementos.

De acuerdo Lagrange, dados *k* puntos (x_i, y_i) para $i = 1, \ldots k$

$$f(x) = \sum_{i=1}^{k} y_i \prod_{j=1, j\neq i}^{k} \frac{x - x_j}{x_i - x_j}\ (mod\ p)$$

y por tanto:

$$s = f(0) = \sum_{i=1}^{k} y_i \prod_{j=1, j\neq i}^{k} \frac{-x_j}{x_i - x_j}\ (mod\ p)$$

Como se afirma en [130], existen varias propiedades que hacen muy interesante el protocolo Shamir:

- Se puede incrementar *n* (añadir nuevos participantes)
- Se pueden eliminar participantes
- Se pueden cambiar todas o algunas de las partes ($s_1 \ldots s_n$) de *s* sin cambiar el secreto *s*, simplemente seleccionando un nuevo polinomio $f'(x)$ y un nuevo grupo de partes (propiedad denominada seguridad proactiva).
- Existe una versión del esquema de Shamir con distribuidor o *dealer*.

Otra propiedad de gran utilidad del protocolo de Shamir es el homomorfismo. Sobre ella así como sobre el cifrado homomórfico se profundiza en 2.2.4.6.

Existen varios sistemas de VER que explotan el homomorfismo en esquemas de reparto de secreto [108, 131, 132].

Para otras implementaciones de sistemas de reparto de secreto con umbral, así como para el lector que quiera profundizar en la materia, se recomienda la lectura de: [85, 125, 31, 107, 128, 126, 127, 108, 131, 132, 129].

2.2.4.6 Homomorfismo y cifrado homomórfico

En álgebra abstracta, se denomina un homomorfismo a una función entre dos mismas estructuras algebraicas (grupos, anillos o espacios vectoriales p. ej.) que preserva las operaciones definidas en dichos objetos.

El homomorfismo tiene una gran importancia en los sistemas de VER, puesto que una vez que se encripta un voto con alguna de las técnicas vistas anteriormente, sería muy poco eficiente tener que desencriptar de nuevo cada uno de ellos individualmente para proceder a su recuento. Por ello, se recurre a las propiedades homomórficas para operar directamente sobre los votos encriptados.

En los sistemas de encriptado homomórfico, existe una operación algebraica $\oplus$ sobre el espacio de los textos originales sin cifrar que equivale a otra operación algebraica $\otimes$ (no necesariamente la misma) sobre el espacio de los mismos textos cifrados.

$$E(v_1 \oplus v_2) = E(v_1) \otimes E(v_2)$$

Ello es de un gran interés porque se pueden realizar operaciones sobre información cifrada (por tanto sin desvelar el contenido), sabiendo que se obtienen los mismos resultados que si se aplicaran las operaciones equivalentes sobre la información sin encriptar.

Así, se dice que un sistema de encriptado es homomórfico aditivo si:

$$E(v_1) \otimes E(v_2) = E(v_1 + v_2)$$

Paralelamente, un sistema de encriptado es homomórfico multiplicativo si:

$$E(v_1) \otimes E(v_2) = E(v_1 v_2)$$

Se conocen como sistemas de encriptado parcialmente homomórficos (PHE) a aquellos que lo son sobre un determinado conjunto de funciones (suma, multiplicación, funciones cuadráticas etc.). Hay multitud de sistemas PHE utilizados en *cloud computing*, *data mining*, transacciones bancarias etc.

A continuación se explican 2 de los más importantes para sistemas de Voto Électrónico Remoto: El Gamal y Paillier [64, 82].

2.2.4.6a Criptosistema ElGamal [64]

El criptosistema de ElGamal es homomórfico con la multiplicación por naturaleza.

Asumiendo en un grupo G de orden $|G| = q$

La clave pública es (G, q, g, h), donde g es un generador de G, $h = g^x$, siendo x la clave privada.

La encriptación de un voto es E (v) = (α, β) = (g^r, $v \cdot h^r$) para una r aleatoria o *random* r ϵ {0, 1, … , q - 1}

Para dos votos encriptados de la siguiente manera:

$$E\ (v_1) = (\alpha_1, \beta_1) = (g^{r1}, v_1 \cdot h^{r1})$$
$$E\ (v_2) = (\alpha_2, \beta_2) = (g^{r2}, v_2 \cdot h^{r2})$$

La propiedad homomórfica es por tanto:

$$E\ (v_1) \cdot E\ (v_2) = (\alpha_1, \beta_1) \cdot (\alpha_2, \beta_2) = (g^{r1}, v_1 \cdot h^{r1}) \cdot (g^{r2}, v_2 \cdot h^{r2})$$
$$= (g^{r1+r2}, (v_1 \cdot v_2)\ h^{r1\ +\ r2}) = E\ (v_1 \cdot v_2)$$

Para el caso del VER, lo que más interesa es la propiedad de la suma para el recuento de votos; por tanto un **sistema homomórfico aditivo**.

Para ello, se suele utilizar una versión modificada de ElGamal que acepta la suma:

La encriptación de un voto en este caso es E (v) = (α, β) = (g^r, $g^v \cdot h^r$) para una *r* aleatoria o *random* r ϵ {*0, 1, … , q - 1*}.

Para dos votos encriptados de la siguiente manera:

$$E\ (v_1) = (\alpha_1, \beta_1) = (g^{r1}, g^{v1} \cdot h^{r1})$$
$$E\ (v_2) = (\alpha_2, \beta_2) = (g^{r2}, g^{v2} \cdot h^{r2})$$

La propiedad homomórfica aditiva es:

$$E\ (v_1) \cdot E\ (v_2) = (\alpha_1, \beta_1) \cdot (\alpha_2, \beta_2) = (g^{r1}, g^{v1} \cdot h^{r1}) \cdot (g^{r2}, g^{v2} \cdot h^{r2})$$
$$= (g^{r1+r2}, g^{v1\ +\ v2} \cdot h^{r1\ +\ r2}) = E\ (v_1 + v_2)$$

Por tanto, si $v_1 + v_2$ no es muy grande, se puede resolver el algoritmo discreto $g^{v1\ +\ v2}$ y obtener la suma de los dos votos v_1 y v_2, permitiendo por tanto el recuento de votos sin tener que desencriptarlos.

2.2.4.6b Criptosistema Paillier [82, 152]

El criptosistema de Paillier es otra función de encriptación probabilística para sistemas de criptografía de clave pública, al igual que ElGamal.

La propiedad homomórfica de este sistema asume que la clave pública es módulo *m* y la base es *g*.

La encriptación de un voto se define como:

$$E(x) = g^x r^m\ mod\ m^2$$

para una *r* aleatoria o *random*

$$r \in \{0, 1, \dots, q-1\}$$

Para 2 votos encriptados de la siguiente manera:

$$E(v_1) = g^{v_1}\, r_1^m$$
$$E(v_2) = g^{v_2}\, r_2^m$$

La propiedad homomórfica es pues:

$$E(v_1) \cdot E(v_2) = (g^{v_1}\ r_1^m) \cdot (g^{v_2}\ r_2^m) = g^{v_1+v_2}\,(r_1 r_2)^m = E(v_1 + v_2\ mod\ m)$$

Aparte de ElGamal y Paillier, otros criptosistemas destacados son: RSA, Goldwasser – Micali, Benaloh, Naccache – Stern etc.

Se recomienda la lectura de [64, 82, 140, 141] profundizar en sistemas parcialmente homomórficos.

En contraposición a éstos, se conocen como sistemas de encriptado homomórfico completos (*Fully Homomorphic Encryption*) a aquellos que son homomórficos sobre cualquier función. Son por tanto mucho más versátiles y potentes puesto que permiten crear sistemas aplicables a cualquier funcionalidad.

El primer sistema de *Fully Homomorphic Encryption* (FHE) fue desarrollado por Craig Gentry en la Universidad de Stanford en 2009 y supuso un hito en criptografía [134]. El Dr. Gentry lo consigue haciendo el sistema homomórfico sobre el operador NAND. Como el lector sabrá, cualquier circuito lógico se puede construir utilizando únicamente funciones NAND.

La seguridad de su modelo está basada en la dificultad de problemas relacionados con retículos algebraicos o *lattices* (referirse al punto 2.2.4.10e para más detalles). En su primera versión, los textos cifrados eran de 128 MB y la encriptación de un solo bit llevaba más de 30 minutos de computación, haciéndolo inaplicable en la práctica.

Con posterioridad han surgido otras propuestas de sistemas FHE tales como [142, 143, 144] que mejoran la eficiencia de manera importante. No obstante, la enorme carga computacional de los sistemas FHE hacen que todavía no esté clara la posibilidad de una aplicación práctica a sistemas de VER al menos en un futuro próximo.

Volviendo al cifrado homomórfico en el VER, ya se ha comentado que la gran ventaja de los criptosistemas homomórficos es el hecho de que permite realizar las labores de recuento de votos mientras la información se mantiene encriptada.

En cuanto a las debilidades, destaca su coste computacional, así como su vulnerabilidad a ataques del tipo *RSA blinding attack* [145], debido a que estos sistemas son maleables por

definición (dado un texto cifrado C_i, es relativamente fácil crear otro texto cifrado C_j relacionado con C_i de tal manera que sus desencriptaciones estén también relacionadas. Referirse a [156] y al apartado 2.2.4.10i para más detalles).

Otra dificultad a la que se enfrentan los sistemas de VER que utilizan encriptación homomórfica es cómo asegurarse de que el voto encriptado es realmente la encriptación de un 1 o un 0 (opción seleccionada por el votante o no respectivamente). Para ello, en la práctica se usan pruebas de conocimiento cero no interactivas (NIZK) que se desarrollan en el apartado 2.2.4.7. La introducción de NIZK añade una mayor complejidad computacional al sistema, limitándolo para elecciones a gran escala.

Por tanto, se puede concluir que el papel de la criptografía homomórfica para los sistemas de VER es extremadamente importante y con los avances que es están produciendo en el campo es de esperar que vaya en aumento. No obstante, en la actualidad los sistemas que ofrecen una mayor seguridad (FHE) acarrean unos costes computacionales tan elevados que los imposibilita para ser llevados a la práctica.

El lector que quiera profundizar a mayores en la encriptación homomórfica, tanto parcial como total, así como consultar ejemplos aquí introducidos, puede referirse a: [23, 64, 80, 82, 130, 132, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145].

2.2.4.7 Pruebas de conocimiento cero (ZKP y NIZKP)

Una prueba o protocolo de conocimiento cero (*Zero Knowledge Proof* o ZKP) es un método por el que una parte denominada *"probador"* quiere demostrar a la otra parte *"verificador"* que un enunciado (normalmente matemático) es cierto. Existe por tanto una interacción entre probador y verificador.

Normalmente, el enunciado que recibe el probador por parte del verificador incluye algún tipo de prueba o desafío para cuya resolución se requiere algún tipo de conocimiento secreto. Es decir, el probador debe demostrar que tiene el conocimiento sobre un secreto, sin desvelarlo.

El hecho de que el probador responda correctamente al desafío da al verificador una cierta confianza de que el probador efectivamente posee el conocimiento requerido. La iteración n veces de dicha prueba y su correcta respuesta va paulatinamente reduciendo las posibilidades de que el probador acierte por casualidad y por tanto aumentando la probabilidad de que efectivamente conozca el secreto.

Para una aproximación intuitiva al concepto de prueba de conocimiento cero, referirse al reconocido ejemplo de Quisquater en su paper *"How to explain Zero-Knowledge Protocols to your children"* [153].

Las propiedades que se esperan de un protocolo de conocimiento cero son:

- Integridad: Si la prueba es verdadera, el verificador debe siempre aceptarla.
- Congruencia: Si la prueba es falsa, el verificador la rechazará con un nivel de probabilidad tan alto que la opción de aceptarla es insignificante, para una cantidad de iteraciones suficiente. En otras palabras, el probador no puede engañar y hacer creer al verificador que conoce el secreto.
- Conocimiento cero: El verificador no puede aprender nada más allá de la validez de la prueba de sus interacciones con el probador. Esta propiedad previene del caso en el que el verificador es suplantado por un elemento malicioso.

Las pruebas de conocimiento cero fueron introducidas por primera vez en 1985, en el paper *"The knowledge complexity of interactive proof systems"* de S. Goldwasser et al. [95]. De una manera más formal, se definen las propiedades anteriores basándose en el modelo computacional de la máquina de Turing.

Sean *P, V* y *S* máquinas de Turing.

Un protocolo de prueba interactivo es de conocimiento cero sobre un lenguaje *L* si

Para cualquier verificador *V'* de tiempo probabilístico polinomial, existe un simulador *S* de tiempo probabilístico polinomial, el cuál:

$$\forall x \in L, z \in \{0,1\}^*, View_{v'}\,[P(x) \leftrightarrow V'(x,z)] = S\,(x,z)$$

En este caso, z representa una cadena de conocimiento previo.

La definición anterior representa una *prueba de conocimiento cero perfecta*. En una prueba de conocimiento computacional en cambio, únicamente se pide que las vistas de *V'* y *S* sean computacionalmente indistinguibles.

En los sistemas de VER y por cuestiones de seguridad y de coste computacional, la variante más utilizada de las ZKP son las llamadas pruebas de conocimiento cero no interactivas, normalmente denominadas NIZKP.

Pruebas de conocimiento cero no interactivas (NIZKP)

Esta tipología de ZKP fue introducida por Blum et al. en 1988 [96] y se caracteriza por no requerir interacción entre el probador y el verificador.

En su lugar, los autores desarrollaron el modelo de secuencia común de referencia o *"CRS model",* detallado en el punto 2.2.4.3. En él, se demuestra que utilizando como *input* una secuencia común de referencia (CRS) *x* junto a otro parámetro aleatorio *r* generado por una tercera parte confiable, es suficiente que el probador compute una cadena π para que el verificador valide que $x \in L$, siendo *L* un lenguaje.

Al tratarse de un tipo concreto de prueba de conocimiento cero, debe cumplir con las tres propiedades enunciadas al inicio del presente punto. De una manera formal:

Sea *R* una relación binaria computable en tiempo polinomial.

En las duplas (*y, w*) ϵ *R: y* es la afirmación a probar y *w* el testigo.

L es el lenguaje *NP* (tiempo polinomial no determinístico) de las afirmaciones con los testigos en *R*.

Se supone que ambas partes, probador y verificador, están en posesión de una cadena de referencia σ obtenida de una distribución *D* a través de un tercer elemento confiable:

$$\sigma \leftarrow Setup\ (1^k).$$

Para probar que $y \in L$ con el testigo *w,* el probador computa

$$\pi \leftarrow Prueba\ (\sigma, y, w)$$

Y envía el *output* π al verificador

El verificador acepta si $Verificacion\ (\sigma, y, \pi) =$ y lo rechaza en otro caso.

- Integridad

$$\forall\ \sigma\ \in Setup\ (1^k), \forall\ (y, w) \in\ R_\sigma$$

La relación de testigo se puede generalizar como $(y, w) \in\ R_\sigma$ para tomar en consideración que σ puede tener influencia sobre las afirmaciones a probar.

$$\Pr[\pi\ \leftarrow Prueba\ (\sigma, y, w) : Verificación\ (\sigma, y, \pi) = aceptar = 1]$$

- Congruencia

La congruencia requiere que ningún probador pueda hacer a un verificador aceptar una afirmación falsa $y\ \notin\ L$ excepto para una probabilidad despreciable.

De una manera formal, para cada adversario *A*, existe una función *v* de probabilidad descartable de tal manera que:

$Pr[\sigma\ \leftarrow Setup(1^k), (y, \pi) \leftarrow A(\sigma) : = y \notin\ L\ \text{^}Verificación(\sigma, y, \pi) = aceptar = 1] = \text{v(k)} \approx 0$

- Conocimiento cero multiteorema (mejora por U. Feige et al. [154])

Un sistema de prueba no interactivo *(Setup, Prueba, Verificación)* es de conocimiento cero multiteorema si existe un simulador probabilístico en tiempo polinomial *S = (S1, S2)* de tal manera que para todos los adversarios *A* no uniformes de tiempo polinomial:

$$\Pr[\sigma \leftarrow Setup(1^k) : A^{Prueba\,(\sigma,..)}(\sigma) = 1] \equiv Pr[(\sigma,\tau) \leftarrow: S_1 : A^{S\,(\sigma,\tau,..)}(\sigma) = 1]$$

Siendo τ una *trapdoor*.

En el ejemplo, $S\ (\sigma,\tau,y,w)$ se obtiene como *output* $S_2\ (\sigma,\tau,y)$ para $(y,w) \in R_\sigma$

Es decir, sí es posible simular la prueba de una afirmación verdadera sin conocer el testigo.

Se pueden obtener también sistemas NIZKP en el modelo de oráculo aleatorio (referirse al punto 2.2.4.2) gracias a la heurística de Fiat-Shamir [20, 85].

En ella, se sustituye la respuesta del verificador por una función *H* modelizada como un oráculo aleatorio, que funciona como una caja negra que devuelve respuestas totalmente aleatorias para cada petición. En cualquier caso, es consistente en sus respuestas, devolviendo siempre la misma cadena para el mismo *input*.

La importancia de la heurística de Fiat-Shamir reside en permitir la conversión de ZKP (tales como los protocolos σ**,** ver punto 2.2.4.10h) en no interactivos (NIZKP), con una enorme aplicación en sistemas de VER.

No obstante, se han ido descubriendo debilidades en la heurística en lo que concierne al cumplimiento real de las propiedades de un sistema NIZK. En concreto, en el caso de la propiedad de congruencia y como demostraron Goldwasser et al. en [90] y Bitansky et al. [91], para el caso de funciones *hash* eficientes reales, hay determinados casos para los que la heurística de Fiar-Shamir no es congruente.

En el campo de los sistemas de VER, una de las soluciones que se consideran paradigmáticas es Helios, la cuál se describe y analiza en profundidad en el punto de 5.2 y subapartados. Pues bien, Helios utiliza una variante conocida como *weak Fiat-Shamir* que hace al sistema vulnerable, pudiendo darse el caso de que el proceso de recuento se itere indefinidamente como explican Bernhard et al. en [155].

La parte positiva es que en el mismo trabajo se detalla una nueva definición de Fiat-Shamir denominada *strong Fiat-Shamir* que solventa los problemas hallados en el *weak Fiat-Shamir*. No obstante, es necesaria una mayor aplicación práctica de la nueva propuesta para asegurarse que se trata de una solución aplicable en el VER.

Como bibliografía de la heurística Fiat-Shamir, referirse a [20, 85, 90, 91, 92, 155].

Para concluir, se va a comentar el uso práctico de ZKP y NIZKP en los sistemas de VER desde el punto de vista de los distintos actores:

- Votante

  Las ZKP y NIZP prueban que el voto encriptado contiene alguno de los valores válidos (por ejemplo 0 o 1) sin revelar cuál (el sentido del voto). Únicamente verifica que contiene una opción válida.

- Autoridades

Las ZKP y NIZP prueban que la desencriptación es correcta; es decir:

a. Se ha usado la clave privada correcta, correspondiente a la clave pública de la elección
b. El valor que las autoridades dan como resultado corresponde realmente al recuento de votos verificable en el tablón de recuento

Lo que no se revela es la clave privada de la elección.

- Mixnets (en el caso de que las haya. Referirse al punto 2.2.4.8)

  Prueba que la reencriptación y el mezclado/permutación ha sido realizado correctamente pero no la aleatoriedad utilizada ni la permutación aplicada a los textos cifrados.

Las referencias bibliográficas recomendadas para el lector interesado en profundizar en las ZKP son las sigiuentes: [94, 95, 96, 146, 147, 148, 150, 151, 152, 153, 154].

2.2.4.8 Mix – Networks o Mix-nets

Introducidas en 1981 por Chaum [31], son protocolos de enrutado que utilizan una sucesión de servidores proxy (denominados mixes). Cada uno de ellos recibe como *input* una colección de textos cifrados o ciphertexts (obtenido a través de protocolos de PKE explicados en 2.2.4.4.1), los re-encripta y mezcla (permuta) y los reenvía al siguiente destinatario (suele ser otro mix).

El principal propósito del uso de mix-nets en sistemas de VER es proveer anonimato e intrazabilidad a los votantes. Es decir, romper el vínculo entre voto y votante.

En el modelo de Chaum, k servidores mix $M_1,\ldots,M_k$ se preparan de manera secuencial. Cada servidor M_j genera un par de claves pública/privada, publicando su clave pública pk_j.

El usuario i-ésimo, para enviar de manera anónima su mensaje m_i, encripta el mensaje con todas las claves públicas de todos los servidores mix y publica el ciphertexto resultante $Enc_{pk_1}\left(Enc_{pk_2}(\ldots Enc_{pk_k}(m_i)\ldots)\right)$.

Sea L_0 la lista de todos los textos cifrados.

Para $j = 1, \ldots, k$

El *j*-ésimo servidor mix M_j toma como *input* L_{j-1}, quita la capa más exterior de encriptación usando su clave privada y permuta los ciphertextos para formar el *output* L_j. Una vez que el último servidor M_k ha desencriptado y mezclado la lista, se pueden publicar los mensajes sin cifrar.

La siguiente figura muestra el funcionamiento de un sistema mix-net según el modelo de Chaum [31], también llamado de desencriptación:

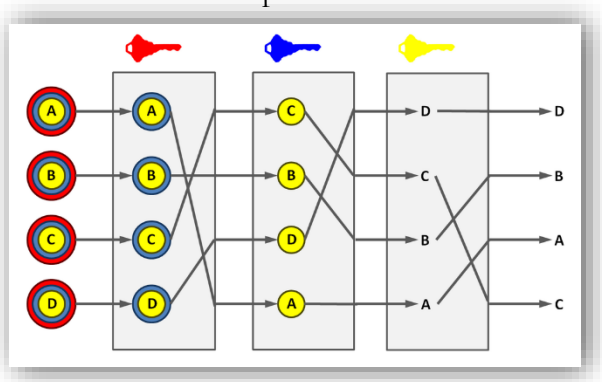


Figura 12: Mix-net de desencriptación. Fuente: prime pq for Wikipedia. CC License.

La solución de Chaum plantea dos problemas importantes: El primero que cualquier servidor mix puede reemplazar cualquier ciphertexto con uno de su elección. El segundo es que los ciphertextos crecen con cada servidor mix que se añade, aumentando la complejidad computacional.

En lo referente al segundo problema, Park et al. en [161] introdujeron las **mix-nets de re-encriptado** en las que los servidores mix explotan las propiedades homomórficas del criptosistema para re-aleatorizar los ciphertextos en lugar de desencriptarlos. De esta manera, únicamente un servidor mix honesto es necesario para garantizar el anonimato del voto. También se consigue descargar al votante de carga computacional puesto que cada servidor se encarga de su parte.

Algunos ejemplos de criptosistemas usados en mix-nets de re-encriptado son El-Gamal y Paillier. El problema como ya se ha visto en el punto 2.2.4.6, es que los criptosistemas basados en propiedades homomórficas, son maleables por definición y por tanto no completamente seguros frente a ataques del tipo IND-CCA2 [156].

La verificación de que cada servidor es honesto se suele realizar implementando ZKP (punto 2.2.4.7), por lo que cada servidor mix debe añadir una prueba de compromiso a la permutación que lleva a cabo. La desventaja es que la complejidad computacional aumenta de manera notable.

Es por ello que Sako y Killian primero [32] y con posterioridad Furukawa et al. [162], Neff [163] y Jakobsson et al. [164] fueron mejorando la eficiencia de las permutaciones de los

mix-nets aunque simultáneamente han ido apareciendo publicaciones encontrando vulnerabilidades en los nuevos sistemas [29, 165, 166, 167, 168].

En [170], Puiggalí et al. proponen un sistema de mix-net heurísticamente seguro que presentaron en la coferencia EVOTE 2010. Pese a suponer un avance con respecto a soluciones previas, en 2012, Khazaei et al. en [30] introdujeron una serie de ataques prácticos contra el sistema] que podrían llegar a comprometer su seguridad.

Los avances en las técnicas de mix-net son contínuos y cada vez más prometedores. No obstante, es todavía prematuro decir que se ha llegado a una solución totalmente satisfactoria aplicable a soluciones de VER como se puede comprobar en la bibliografía recomendada. Actualmente, una de las mixnets verificables más utilizadas a nivel práctico en el desarrollo de sistemas de VER es *Verificatum* desarrollada por Wikström [428].

Por lo que respecta a los sistemas de VER, los basados en mixnets presentan el problema de que el recuento se realiza voto a voto, comparado con los sistemas basados en homomorfismo, en los que se puede realizar sin desencriptar cada voto. Por otra parte, el recuento no puede comenzar hasta que todos los votantes hayan emitido su voto [176].

En general, se considera que para elecciones tipo referéndum donde sólo cabe la respuesta sí/no/en blanco, la opción de recuento con sistemas basados en homorfismo suele preferirse. Por el contrario, en elecciones donde hay multitud de opciones de voto, listas a ordenar etc., se prefieren los sistemas basados en mix-nets.

En la presente disertación, hay varias soluciones de VER que implementan mixnets, tales como Helios, Civitas, nVotes y Scytl que se detallan en sus apartados correspondientes.

La bibliografía en esta materia es muy amplia y en continua expansión. Las siguientes referencias constituyen una buena base sobre el tema: [30, 31, 32, 81, 82, 83, 130, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 176].

2.2.4.9 Esquemas de firma ciega

El esquema de firma ciega fue introducido en 1982 por D. Chaum [78]. En un principio, su uso estaba dirigido a los pagos telemáticos, si bien en 1992 Fujioka et al. [21] lo aplicaron a un sistema de voto electrónico.

En esta tipología de esquemas, la autoridad (o firmante), tras verificar la identidad del emisor, firma el mensaje que éste le envía sin conocer su contenido. Se suele usar el paralelismo del papel calco:

El emisor envuelve su mensaje en papel calco y lo envía a la autoridad para que lo firme sin ver el contenido. Si la identificación del emisor es correcta, la autoridad firma sobre el papel calco quedando por tanto también firmado el mensaje sin que su contenido haya sido desvelado.

Para confirmar que un mensaje es válido, éste deberá incluir la firma de la autoridad.

En la práctica, multitud de esquemas de firma ciega están presentes en diversos protocolos de clave pública como el RSA [173].

Pasando a una definición de un protocolo de firma ciega de una manera formal:

Sean (*n, e*) y (*n, d*) las claves pública y privada respectivamente de la autoridad o firmante. En este caso, *d* es la inversa de *e* mod ø (*n*).

Sea *r* un factor de ocultación aleatorio (*blinding factor*) de tal forma que $r \leftarrow \mathbb{Z}_n^*$ y m.c.d. *(r, n)* = 1. Es decir, *r* y *n* coprimos o primos relativos.

Suponiendo que el votante quiere enviar su voto *v*.

En primer lugar el votante calcula:

$$v' = v \cdot r^e \bmod n$$

Y lo envía a la autoridad. El valor *r* es usado para ocultar el voto *v* a la autoridad. Ésta tras la verificación, firma el voto ocultado o cegado de la siquiente manera:

$$S' = (v')^d = v^d \cdot (r^e)^d = v^d \cdot r \bmod n$$

y lo envía al votante.

El votante por su parte, cuando recibe *S'*, lo des-oculta para obtener el valor de *S*, con la firma válida de la autoridad, puesto que conoce *r*.

$$S = S' \cdot r^{-1} = v^d \cdot r \cdot r^{-1} = v^d \bmod n$$

Es importante destacar que, para evitar que un potencial adversario explote la maleabilidad del protocolo RSA (por sus propiedades homomórficas), es necesario que al voto *v* se le aplique con anterioridad una función *hash* (ver punto 2.2.4.1). De lo contrario, pueden darse ataques como el *RSA blinding attack* y variantes [174, 175].

De hecho, incluso tomando las precauciones oportunas, en los protocolos de firma ciega con mix-nets para romper el vínculo entre votante y voto, la E2Ev es muy difícil de probar, lo que supone un gran problema para su aplicación práctica a soluciones del VER.

En concreto, no se puede garantizar la verificabilidad universal, puesto que ninguna parte externa puede verificar que únicamente se han contado los votos (y todos los votos) de los votantes autorizados a ello.

Los esquemas de firma ciega no incluyen ningún protocolo para los votos de los votantes que se abstienen de votar, dejando vía libre a que un adversario (incluída una autoridad

corrupta) emita todos esos votos de la manera que más le interese y lo que es peor, nadie puede verificar desde fuera si algo así ha ocurrido o no.

Desde un punto de vista práctico, otro hándicap de los esquemas de firma ciega es que se requieren canales anónimos y totalmente seguros, cosa que en la práctica es muy difícil (por no decir virtualmente imposible) de conseguir.

En su haber, está el hecho de que son muy eficientes en su cometido, con un consumo contenido de recursos computacionales.

En conclusión, los esquemas de firma ciega, pese a ser los más eficientes, presentan algunas lagunas de seguridad y verificabilidad potencialmente peligrosas. Por todo ello, es necesario un salto cualitativo en las soluciones hasta la fecha para que su uso pueda equipararse al de los sistemas de VER basados en cifrado homomórfico o mix-nets.

Para profundizar en los esquemas de firma ciega, se recomiendan las siguientes referencias bibliográficas: [78, 88, 171, 21, 172, 173, 174, 175]

2.2.4.10 Otros conceptos y definiciones relevantes

En el presente sub-apartado se detallan una serie de conceptos y definiciones criptográficas relevantes de una manera más breve, puesto que se alejan del objeto principal de la tesis o bien se encuentran en estadios preliminares de investigación y no han sido implementadas en soluciones de VER hasta la fecha.

2.2.4.10a Protocolo de intercambio de claves Diffie-Hellman

El protocolo de intercambio de claves de Diffie-Hellman fue propuesto en 1976 por Whitfield Diffie y Martin Hellman [25]. Se denomina también protocolo de intercambio exponencial. En realidad, la patente ya expirada número *U.S. Patent 4.200.770* de 1977, incluye a Ralph Merkle como co-inventor.

El intercambio de claves Diffie-Hellman (DH) es utilizado en protocolos tan estandarizados como SSH, TLS o IPSec.

La siguiente figura es un resumen del protocolo DH:

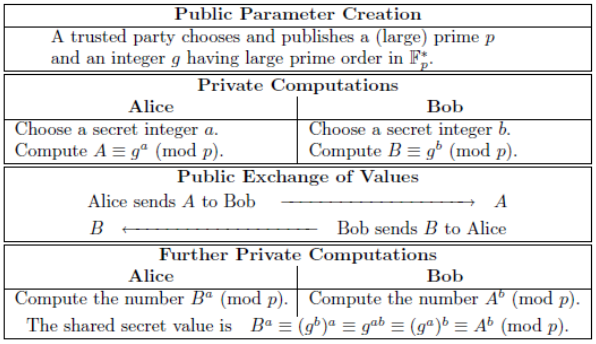


Figura 13: Protocolo de intercambio de claves Diffie-Hellman. Fuente: Universidad de Brown.

El objetivo del protocolo es que las dos partes implicadas (genéricamente Alice y Bob), puedan generar y compartir una clave a través de un canal inseguro de manera anónima.

En la figura previa, el módulo primo p y la base b son públicos (aunque deben modificarse en cada intercambio). g es una raíz primitiva modulo p.

Los valores secretos son a $1 \leq a < n$ (elegido por Alice) y b $1 \leq b < n$ escogido por Bob.

- Alice envía g^a y se lo envía a Bob, quién a su vez envía g^b a Alice.
- Alice computa $(g^b)^a$ y Bob a su vez computa $(g^a)^b$

En consecuencia, tanto Alice como Bob disponen de g^{ab}, que va a servir de clave secreta compartida.

La seguridad del protocolo se basa en el conocido como problema de Diffie-Hellman y que de forma informal se enuncia de la siguiente manera:

"Dado un elemento g y los valores g^a mod p y g^b mod p, determinar el valor de g^{ab}."

Siendo g el generador de un grupo, (normalmente el grupo multiplicativo sobre un cuerpo finito o un grupo de curva elíptica) y a y b enteros elegidos aleatoriamente.

Se asume que si un atacante puede resolver el problema del logaritmo discreto (subapartado 2.2.4.10c), entonces puede también computar los exponentes secretos a y b de Alice y Bob respectivamente y obtener por tanto la clave secreta g^{ab}.

En la actualidad, no existen algoritmos que resuelvan el problema del logaritmo discreto en tiempo computacional polinomial para primos p suficientemente grandes (se recomiendan 1024 o incluso 2048 bits de acuerdo con lo expuesto en [33]) y elementos g de orden primo aproximadamente $p/2$.

En el paper original [25], el protocolo de intercambio no introducía ningún tipo de autenticación de Alice y Bob y por tanto era vulnerable a un ataque del tipo *"man-in-the-*

middle". En él, un atacante, "*Eve*", puede establecer dos protocolos distintos de intercambio, uno con Alice y otro con Bob, haciendo creer a Alice que es Bob con quién se está comunicando y viceversa.

Eve por tanto se encuentra en medio entre las dos partes, recibiendo, descifrando y reenviando cada mensaje entre Alice y Bob, en ambos sentidos.

Para solucionar esta importante debilidad, versiones posteriores de DH han incluído la autenticación de Alice y Bob con herramientas tales como el protocolo STS [177].

Para concluir, indicar que es un algoritmo que introduce un método para compartir una clave secreta en un canal público pero no es un criptosistema de clave pública completo. Para ello, debería ser capaz de transmitir información específica y no únicamente cadenas aleatorias de datos.

En ese sentido, el primer criptosistema de clave pública completo fue el RSA publicado en 1978 por Rivest, Shamir y Adleman [178], todavía ampliamente usado. No obstante, el desarrollo natural sobre DH dio lugar al criptosistema de clave pública ElGamal [64], (punto 2.2.4.6) y ampliamente utilizado por diversas soluciones de VER.

Por todo lo arriba explicado, el protocolo Diffie-Hellman es de una vital importancia en la seguridad de muchas de las herramientas criptográficas más utilizadas en las soluciones de VER.

El lector que desee profundizar en el protocolo Diffie-Hellman así como en sus variantes y debilidades, puede dirigirse a [25, 33, 39, 43, 177, 178, 188].

2.2.4.10b Problema de la factorización de enteros (IFP)

La factorización de enteros es uno de los problemas aún sin solución más relevantes en seguridad informática y criptografía.

Consiste en descomponer un determinado número compuesto en divisores no triviales. Si dichos divisores se restringen a números primos, se habla entonces de factorización de primos.

Por el teorema fundamental de la aritmética, todo entero positivo puede representarse de forma única como producto de factores primos. Para el caso de números muy grandes, no existe ningún algoritmo de factorización hasta la fecha para resolver el problema en tiempo polinómico.

Dentro de ellos, los casos más complejos son los del producto de dos números primos de una longitud similar pero no igual, para evitar la resolución del problema con el método de factorización de Fermat.

Hasta la fecha, el mayor número factorizado ha sido de 232 dígitos (RSA-768) y conseguirlo llevó más de 2 años y medio de computación ininterrumpida utilizando simultáneamente cientos de CPUs de distintos centros de investigación internacionales [179].

Para ello, se hizo uso del algoritmo más eficiente de entre los existentes: "criba general del cuerpo de números" o "*general number field sieve*" o GNFS, cuya complejidad para factorizar un entero *n* con una longitud de $\lfloor log_2 n \rfloor + 1$ bits es del tipo:

$$L_n\left[\frac{1}{3}, \sqrt[3]{\frac{64}{9}}\right]$$

La importancia del problema matemático de la factorización de enteros reside en que multitud de protocolos en internet dependen de la supuesta "imposibilidad" en la solución del citado problema para números suficientemente grandes.

Un ejemplo de ello es el algortimo criptográfico RSA ampliamente utilizado en intercambios de información en internet para criptosistemas de clave pública.

Para profundizar en el IFP, los algoritmos más destacados y los avances en computación cuántica, se recomienda: [179, 180, 181, 182, 183].

2.2.4.10c Problema(s) del logaritmo discreto (DLP)

De una manera análoga al caso anterior, existen actualmente una serie de problemas matemáticos sin solución relacionados con el logaritmo discreto que se utilizan como funciones unidireccionales sobre las que recae la seguridad de varios de los protocolos de seguridad más relevantes de criptografía de clave pública (ElGamal y Diffie-Hellman como protocolo de intercambio de claves) que a su vez se usan en multitud de herramientas de VER.

En criptografía, la definición del problema del logaritmo discreto (DLP) en el grupo *G* del tipo $(\mathbb{Z}_p^*, \times)$ (o grupo multiplicativo cíclico finito de orden *p - 1*, siendo *p* un primo) y siendo *g* un generador de *G*, se define como:

Dado un $y \in G$, computar $x \in \mathbb{Z}_p^*$ de tal forma que $x = log_g y \ mod \ p$

Otras variantes relevantes del mismo problema (CDH más débil que el DLP y DDH más fuerte que el DLP) son:

- El problema de Diffie-Hellman computacional (CDH) [188]:

  Dados dos elementos del grupo *G*:

  a = *a* · g

b = $b \cdot$ g

computar c = $ab \cdot$ g y decidir si c = DH (a, b)

- El problema de Diffie-Hellman decisicional (DDH) [188]:

  Dados tres elementos del grupo *G*:

  a = $a \cdot$ g
  b = $b \cdot$ g
  c = $c \cdot$ g

  decidir si c = DH (a, b) o no

  Para prevenir ataques que puedan llegar a computar el DLP, se recomienda usar números primos de una longitud mínima de 1024 bits, si bien recientes ataques [26, 33] indican que en la actualidad, el umbral de seguridad se situaría en los 2048 bits para garantizar una protección a más largo plazo.

  Como bibliografía de referencia sobre el problema del logaritmo discreto, se recomienda: [119, 183, 184, 185, 186, 187, 188].

2.2.4.10d Criptografía de curva elíptica (ECC)

En 1985, Miller y Koblitz [191, 192] respectivamente fueron los pioneros en proponer el uso de curvas elípticas en el diseño de sistemas de PKC.

La ECC es un tipo de variante de la criptografía de clave pública que se basa en el uso de curvas elípticas para garantizar la seguridad del criptosistema.

Una curva elíptica *E* definida sobre un campo finito $\mathbb{F}_p$ es una ecuación del tipo:

$$E = y^2 = x^3 + ax + b \bmod p,$$

para

$$4a^3 + 27b^2 \neq 0$$

Siendo *p* el número primo que define el campo en el que opera la curva (todos los puntos se toman módulo *p*) y *a* y *b* dos coeficientes enteros que definen la curva.

El conjunto *G* de puntos que forma la curva junto con una operación aditiva forman un grupo abeliano con punto de identidad $\mathcal{O}$.

Tomando un punto *P* de $E(\mathbb{F}_p)$ y suponiendo que *P* tiene un orden primo *n*, entonces el subgrupo cíclico de $E(\mathbb{F}_p)$ generado por *P* es:

$$(P) = \{ \infty, P, 2P, 3P, \ldots, (n-1)P \}$$

El primo p, la ecuación de la curva elíptica E, el punto P y su orden n son públicos. La clave privada es un entero d que se selecciona de manera uniformemente aleatoria del intervalo $[1, n-1]$ y la correspondiente clave pública es $Q = dP$

El problema de determinar d dados los parámetros del dominio así como Q es lo que se conoce como el problema del algoritmo discreto de la curva elíptica o ECDLP.

Los algoritmos más rápidos y eficientes para resolver el ECDLP necesitan $\mathcal{O}\sqrt{n}$ pasos por lo que el tamaño del campo debería ser aproximadamente del doble del parámetro de seguridad. Ésta es precisamente una de las grandes ventajas de la ECC. ElGamal tradicional requiere claves de al menos 1024 bits para poder ser considerado seguro, mientras que ElGamal de curva elíptica obtiene el mismo nivel de seguridad con una clave de 160 bits.

En cuanto al uso efectivo de la ECC y el hecho de que no haya conseguido establecerse como el nuevo paradigma, heredero del RSA, existen varias razones:

Por una parte, varias patentes todavía se encuentran activas y protegiendo las implementaciones más destacadas (la empresa Certicom dispone de más de 130) por lo que muchos desarrolladores prefieren no arriesgarse a infringirlas.

Por otra parte, la carga matemática del RSA es más simple que la del ECC, (factorización vs logaritmos discretos sobre curvas elípticas) lo que hace que los desarrolladores se sientan más "cómodos" al entender mejor su funcionamiento.

Por último, las operaciones sobre la clave pública (tales como la verificación de la firma) sí son más veloces con RSA.

A pesar de las razones arriba expuestas, el NIST americano recomienda 15 curvas elípticas estándar, aunque en la práctica la mayoría de las implementaciones sólo aceptan 2 de ellas (P-256 y P-384) porque son las recomendadas por la NSA.

Otro caso que ha afectado a su aceptación mayoritaria tiene que ver con una serie de revelaciones del New York Times en 2013 que supuestamente derivarían de filtraciones de documentos internos de la NSA por parte de Edward Snowden en los que se indicaba que la NSA habría tratado de estandarizar el uso de la *Dual_EC_DRGB* porque contenía una *backdoor* introducida para ser utilizada como parte del programa de desencriptación *BULLRUN* de la misma agencia.

Tanto la NSA como el NIST negaron las acusaciones, y no se pudo probar nada, si bien la *Dual_EC_DRGB* desapareció de la guía de referencia del NIST.

Por lo que respecta al objeto principal de la tesis, en la actualidad no existe ningún sistema de VER que utilice ECC, si bien se van dando pasos interesantes como el trabajo realizado por M.A. Cerveró et al. [189].

En resumen, las implementaciones de ECC poseen un gran potencial aunque todavía no gozan de la suficiente implantación. Una mayor complejidad matemática así como la vigencia de numerosas patentes provocan que sea todavía necesario un tiempo antes de que esta nueva familia de algoritmos se vaya asentado como un nuevo estándar de PKC en sus múltiples aplicaciones, entre ellas el VER.

Para profundizar a mayores en el prometedor campo de la ECC se recomienda comenzar con las siguientes referencias bibliográficas: [46, 189, 190, 191, 192, 193, 194].

2.2.4.10e Criptografía basada en retículos (*Lattice-based cryptography*)

La criptografía basada en retículos (*lattice-based* según su denominación inglesa más extendida) parte del trabajo de Shor [196], en el que demostró que ataques exitosos con ordenadores cuánticos son posibles para sistemas criptográficos asimétricos basados tanto en el problema DLP como en el ECDLP.

Con ello quedó claro que los criptosistemas basados en la irresolubilidad de algoritmos discretos (DH y variantes, RSA, ElGamal etc.) no eran una opción segura a largo plazo y por tanto urgía encontrar una nueva línea de investigación criptográfica resistente a ataques de computación cuántica: las redes o retículos.

Un retículo *L* es un conjunto de puntos en un espacio de dimensión *n* con una estructura periódica.

De una manera más formal:

Dados *n* vectores linearmente independientes $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$, la red o retículo generada por ellos es el conjunto de vectores:

$$L(b_1, \ldots, b_n) = \left\{ \sum_{i=1}^{n} x_i b_i : x_i \in \mathbb{Z} \right\}$$

Donde $\{b_1, \ldots, b_n\}$ es una base de $\mathbb{R}^n$.

La aplicación de sistemas retículos en el campo de la criptografía se debe a un paper revolucionario de Atjai en 2004 [195]. El mismo autor había presentado un artículo en 1996 en el que demostraba que era posible utilizar reducciones aleatorias para establecer una conexión *worst-case* y *average-case* entre determinados problemas de retículos.

Ello fue lo que llamó fuertemente la atención de los criptógrafos, dado que la irresolubilidad en el *average-case* es una propiedad deseable para los criptosistemas.

Otro hito importante se produjo en 2009 cuando Craig Gentry propuso el primer esquema de encriptación totalmente homomórfico con retículos [134]. Posteriormente ha continuado investigando en IBM con resultados prometedores [135].

Ejemplos de problemas basados en retículos utilizados en criptosistemas son:

- *Shortest Vector Problem* o SVP (dada una base de un retículo, calcular el vector más corto de la misma).
- *Closest Vector Problem* o CVP (dada una base de un retículo y un vector, hallar el vector dentro del retículo a una menor distancia del primero).

Algunos de los algoritmos más avanzados que se han presentado en los 3 últimos años en este campo son: DLP [204], GLP [198] y GLISS [203].

Los algoritmos utilizados en la criptografía de retículos se dividen en dos tipos:

- De retículos generales, con unas bases que garantizan una seguridad sólida incluso en los peores escenarios. Su problema es que son algoritmos demasiado grandes y poco eficientes para su uso real.
- De retículos ideales, introducen estructuras algebraicas y son más eficientes. Por el momento no aportan una garantía de seguridad análoga a la tipología anterior.

Desde 2012 en adelante existe una muy interesante producción científica en la materia cada vez más cercana a casos reales de uso.

Por todo ello, se puede augurar que en los próximos años los avances irán haciendo posible la introducción paulatina de los sistemas de criptografía de retículo y su aplicación a todos los campos relacionados, incluídos los sistemas de Voto Electrónico Remoto, dando lugar a una nueva generación de soluciones con una seguridad reforzada, posiblemente a largo plazo, frente a nuevas tipologías de ataque.

Para profundizar en el campo de la criptografía de retículos, el autor recomienda la siguiente bibliografía: [134, 135, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204].

2.2.4.10f Cifrado por Bloques (*Block Cipher*) y Cifrado de flujo (*Stream Cipher*)

En el presente subapartado se van a explicar dos de las tipologías más famosas de algoritmos de criptografía simétrica: el cifrado por bloques y el cifrado de flujo.

El cifrado por bloques o *Block Cipher* es un tipo de algoritmo determinista de encriptación de clave simétrica que transforma bloques o conjuntos de *n* bits de datos en bloques de *n* bits de datos de texto cifrado (del mismo tamaño). La longitud fija *n* se denomina tamaño de bloque. La transformación tiene lugar usando una clave secreta proporcionada por el usuario.

El origen se remonta a 1949, con la introducción del concepto de cifrado de producto iterativo (según la traducción) por parte de Shannon en [205].

Entre las implementaciones más destacadas se encuentran la de Feistel, usada en el *Data Encryption Standard* o DES. No obstante, el cifrado por bloques tiene otras aplicaciones tales como la generación de números pseudoaleatorios o las funciones *hash* universales.

Normalmente las funciones que implementan cifrado por bloques suelen ser invertibles, por lo que dada la clave y el texto cifrado se puede obtener la información original.

El tamaño de bloque más habitual es de 64 bits (DES) o 128 bits (AES [207, 208]). Hasta el momento no se tiene noticia de que se haya producido ningún ataque exitoso al estándar AES-128.

En la práctica, para encriptar o desencriptar un mensaje de un determinado tamaño, en vez de usar directamente el cifrado de bloque, se le introduce en un modo de operación que utiliza el cifrado de bloques.

El modo de operación más simple es el conocido como *Electronic Code Book Mode* (ECB), el cuál divide el texto original en bloques, cada uno de los cuales es cifrado usando la clave secreta (Si el último bloque no tiene los bits suficientes, se rellena).

Su principal problema es que no aporta auténtica confidencialidad, pues dos bloques idénticos de entrada producen la misma secuencia cifrada.

Para evitar el problema de seguridad del ECB, la solución más ampliamente adoptada consiste en aleatorizar el texto de entrada sin encriptar utilizando un vector de inicialización (V.I.).

Un modo de utilización de un vector de inicialización es el siguiente:

- Se genera un V.I. aleatorio o pseudoaleatorio del tamaño establecido en el protocolo y se le añade al primer bloque con una operación lógica XOR.

- El texto obtenido se usa como V.I. para el siguiente bloque y así sucesivamente. El punto débil es la denominada propagación del error, puesto que un único fallo en uno de los pasos de cifrado se arrasta a todo el bloque y los subsiguientes.

Los modos de operación más usulaes son: *Cipher Block Chaining (CBC), Propagating Cipher Block Chaining (PCBC), Cipher Feedback (CFB), Output Feedback (OFB) y Counter (CTR).*

Para el lector interesado, el NIST tiene publicado un documento sobre recomendaciones para modos de operación [206].

Respecto a los protocolos de cifrado de bloque más destacados, aparte de *Lucifer/DES* y *Rijndael/AES*, destacan *IDEA* [210], *RC5* [209] y *Blowfish* [211].

En cuanto al cifrado de flujo o ***Stream Cipher*****,** es un algoritmo de cifrado de clave pública que encripta 1 bit o un byte cada vez. Para ello, utiliza como clave una secuencia o flujo de bits pseudoaleatorios; de ahí la denominación.

Para que la implementación de un cifrado de flujo sea segura, se le pide al generador del flujo de datos que sea impredecible y que no reutilice nunca una clave. En la práctica, los generadores se diseñan para que se parezcan al generador idealizado, que se denomina *One-Time-Pad.*

Para que la encriptación fuese segura (inmune a ataques de fuerza bruta), las claves deberían ser tan largas o más que el propio texto plano por lo que una película de 1 Gb. debería tener una clave aún mayor, haciéndolo inutilzable en la práctica excepto para archivos clasificados o de información reservada.

En la práctica, el cifrado de flujo aporta una seguridad fuerte pero no se puede considerar totalmente seguro. Aún así, los cifrados de flujo tienen la ventaja de ser lineales en tiempo y constantes en espacio y son por tanto veloces. También presentan una baja propagación de errores a diferencia del cifrado por bloques.

Entre sus desventajas, destaca el hecho de que tienen una baja difusión (toda la información de un símbolo de texto no encriptado se encuentra en un solo símbolo de texto cifrado) y además son más susceptibles a la inserción de porciones maliciosas por parte de un atacante activo en comparación con el cifrado por bloques.

Los cifrados de flujo se utilizan en general menos que los de bloques. No obstante, en los casos en los que la información es transmitida en tiempo real y en pequeños fragmentos (tales como la digitalización de una conversación telefónica o para algoritmos de seguridad sobre redes inalámbricas), la utilización de cifrado de bloques es muy poco eficiente porque la mayor parte de los bloques serían relleno y se ralentizaría la encriptación en una comunicación que por definición debe ser rápida.

Entre los cifrados de flujo más populares se encuentran: RC4 [213], PANAMA [215], SOSEMANUK [216] etc.

RC4 fue el más utilizado durante años, aunque los problemas de seguridad relacionados con su uso en el protocolo TLS han llevado a que la IETF haya prohibido su uso con Mozilla [212], con Microsoft emitiendo recomendaciones similares.

Para profundizar en el cifrado de flujo, referirse a: [212, 213, 214, 215, 216].

2.2.4.10g Infraestructura de clave pública (*Public Key Infrastructure*/PKI)

Una infraestructura de clave pública (o PKI por su denominación en inglés) se refiere al conjunto de mecanismos, procedimientos y políticas que conforman el marco sobre el que se van a implementar los elementos principales de seguridad como son: la autenticación, la integridad, la encriptación y el no repudio (la propiedad de evitar que un individuo u organización pueda negar haber realizado una acción relacionada a unos datos).

De una manera más simple, una PKI es un conjunto de procesos y estándares utilizados para asegurar que los intercambios de información necesarios pueden llevarse a cabo de una forma segura, así como para verificar la identidad de los usuarios de una manera fiable a través del uso de firmas digitales.

Los dos elementos fundamentales de la PKI son la criptografía de clave pública o PKC y la(s) autoridad(es) certificadora(s) o CA.

Respecto a la criptografía de clave pública, el lector puede dirigirse al apartado 2.2.4.4 de la presente tesis.

Por lo que respecta a la autoridad certificadora o CA; puede ser una única entidad o puede estar distribuida en varias, con el fin de aumentar la seguridad del sistema si la naturaleza de la información es especialmente delicada (registros médicos, transacciones financieras o votos). Otra denominación alternativa de la CA es TTP, *Trusted Third Party* o tercera parte confiable.

La(s) CA son pues, terceras partes confiables e independientes encargadas de verificar la identidad de la persona/organización y la posterior emisión de certificados digitales.

En ocasiones, la recepción de la petición de un certificado digital por parte de un usuario es llevada a cabo por una RA o autoridad de registro independiente para aumentar la seguridad de la PKI.

Una vez validada la identidad del usuario que solicita un certificado digital, la CA se encarga de emitir el certificado digital y también de revocarlo en el caso de que se produzca un problema o un error que comprometa la seguridad.

En la práctica, las CA son empresas tales como Symantec o Comodo. En concreto, las dos citadas controlan más del 70% del mercado mundial de emisión de certificados digitales, lo que en ocasiones ha sido motivo de procupación por parte de grupos de investigación por la excesiva acumulación de información crítica para la seguridad de las comunicaciones globales en unas pocas manos privadas.

En cuanto a los certificados digitales o DC, se usan para vincular de forma unívoca la clave pública de un determinado usuario a ese usuario de una manera criptográficamente segura. El DC por tanto incluye la clave pública de su propietario así como su identidad y determinados atributos necesarios tales como la fecha de caducidad del certificado.

Los certificados digitales se emiten de acuerdo a las recomendaciones técnicas del estándar x.509 publicado por *el International Telecommunication Union-Telecommunications Standardization Sector* o ITU-T.

En el caso del VER, la infraestructura de clave pública es utilizada por varias soluciones para la identificación de los votantes de una manera segura. En concreto, el sistema de votación de Estonia que se estudia en el punto 3.2.1 así como el sistema de voto Helios y algunas de sus variantes (apartado 5.2) son ejemplos del uso de la PKI en el campo del Voto Electrónico Remoto.

Para profundizar en la infraestructura de clave pública, su funcionamiento y sus aplicaciones se recomienda la siguiente bibliografía: [120, 217, 218, 219, 220].

2.2.4.10h Protocolos-σ. El protocolo Schnorr

Los protocolos-σ son pruebas de conocimiento cero de 3 movimientos que se utilizan para demostrar que un enunciado x pertenece al lenguaje $\mathcal{L}_{\mathcal{R}}$. Para ello se establece un protocolo interactivo entre el probador P y el verificador V:

P envía un mensaje de compromiso c a V. V responde con un reto r y finalmente P envía una contestación a a V. Después de esta interacción entre ambas partes, V decide si aceptar o rechazar la prueba basándose en la información intercambiada.

Dicho protocolo se dice que es sigma (σ) si satisface las propiedades de integridad, congruencia especial y de verificador honesto conocimiento cero especial tal y como se definen en [382].

Dentro de los protocolos σ, posiblemente el más utilizado en sistemas de VER sea el protocolo Schnorr que se explica a continuación:

El protocolo Schnorr [383]:

Dado un grupo cíclico $\mathbb{G}$ y un valor $g^x \in \mathbb{G}$, el protocolo Schnorr puede usarse para probar el conocimiento del exponente *x*:

- El probador computa $a = g^s$ siendo *s* un elemento aleatorio de $\mathbb{Z}_q$ y lo envía al verificador
- El verificador envía un reto aleatorio e al probador.
- El probador envía al verificador $z = s + xe$
- Por último, el verificador comprueba que $g^z = a \cdot (g^x)^e$

Este protocolo se puede simular de la siguiente manera:

El simulador toma como ejemplo un $z^* \in \mathbb{G}$, un $e^* \in \mathbb{Z}_q$ aleatorio y computa

$$a^* = g^{z^*} \cdot (g^x)^{-e^*}$$

Los valores resultantes (a^*, e^*, z^*) tienen la misma distribución que los originales.

2.2.4.10i IND, NM, CPA, CCA1 y CCA2

Dados un mensaje sin cifrar *m* y su correspondiente texto cifrado *c* resultante de su encriptación, se denomina indistinguibilidad, generalmente representada por el acrónimo IND a una interpretación fuerte de la privacidad por la que un atacante que obtiene dos mensajes y sus correspondientes textos cifrados no puede distinguir qué mensaje sin cifrar corresponde a cada texto cifrado. Su explicación formal se encuentra en [384].

En lo referente a la no-maleabilidad o NM, hace referencia a la resistencia de los textos cifrados frente a la manipulación. Dolev, Dwork y Naor formalizan en [156] la imposibilidad de un atacante de modificar *c* de tal manera que el texto sin encriptar *m'* resultante de la manipulación tenga una determinada relación con el texto original *m*.

Relacionadas con las propiedades IND y NM, se proponen una serie de modelos de ataque denominados CPA (*chosen-plaintext attack*), CCA1 (*non-adaptative chosen-ciphertext attack*) y CCA2 (*chosen-ciphertext attack*) de fuerza creciente.

En el caso del CPA, el adversario tiene acceso a la clave pública de encriptación en los esquemas de encriptación de clave pública (2.2.4.4.1 en la presente tesis).

En un esquema de encriptación de clave pública seguro IND-CPA, el adversario no puede averiguar qué texto en abierto se correponde a cada texto cifrado con una probabilidad de éxito mayor del 50%. Por otra parte, un esquema NM-CPA seguro se

considera también IND-CPA seguro. En el caso concreto del VER, el esquema de encriptación ElGamal (punto 2.2.4.6a) es IND-CPA seguro siempre que se mantengan las precondiciones para el *Decisional Diffie-Hellman* (2.2.4.10a) para el grupo cíclico subyacente $\mathbb{G}$. pero no es CCA2 seguro. Por otra parte, el sistema RSA básico no es seguro IND-CPA en el caso de que la encriptación sea una computación determinística.

En lo que respecta al modelo CCA1 [385] el adversario, además de poseer la clave pública, dispone de acceso a un oráculo de desencriptación al que puede consultar hasta inmediatamente antes de que el reto aleatorio (texto cifrado) es suministrado.

El tercer modelo de ataque es el más exigente: CCA2 [386]. En este caso, el atacante tiene, al igual que en CPA y CCA1 acceso a la clave pública y además a un oráculo de desencriptación al que puede consultar tanto antes como después de haber recibido el reto aleatorio (texto cifrado). La única limitación del ataque es que el atacante no puede preguntar al oráculo sobre el reto en sí. En el modelo CCA2, IND y NM son equivalentes. Como ejemplo de sistema CCA2 seguro destaca *RSA-OAEP*, como demostraron Fujisaki et al. en [387].

2.2.4.11 Conclusiones

Debido a la extensión del presente apartado 2.2.4 de bases matemáticas y criptográficas, antes de proceder al siguiente punto se van a extraer a continuación a modo de resumen las principales tecnologías de aplicación práctica a sistemas de VER estudiadas en el presente punto, debido a su utilización directa en numerosos apartados venideros de la tesis.

a) Firma ciega

Como se expuso en el punto 2.2.4.9, en los esquemas de firma ciega existe una autoridad encargada de firmar un voto encriptado sin necesidad de desencriptarlo. Una vez comprobada la elegibilidad del votante, la autoridad firma el voto encriptado y lo reenvía al votante como prueba de que su voto es válido.

Posteriormente, el votante puede enviar su voto a través de un esquema de mixnets para romper el vínculo entre votante y voto.

Ventajas

La principal ventaja de los sistemas de firma ciega es su eficiencia. Tanto en la fase de voto como en la fase de recuento, es comparativamente más eficiente que los otros dos esquemas principales (encriptación homomórfica y mixnets).

Inconvenientes

El principal inconveniente de los sistemas basados en firma ciega es que la verificabilidad universal está comprometida.

Ello se debe a que una autoridad podría añadir votos fraudulentamente haciéndose pasar por un votante que no haya votado y no habría forma de ser comprobado. Análogamente, una autoridad maliciosa podría proveer a determinados votantes corruptos más de una identificación para que pudiesen votar varias. En este caso tampoco podría verificarse que dicha acción hubiese ocurrido, lo cuál es incompatible con la necesidad de un sistema de VER de ser E2Ev.

El motivo subyacente es que en los sistemas de firma ciega no se aborda la cuestión de los votantes que no ejercen su derecho a voto.

Otro problema añadido es el hecho de que los sistemas de firma ciega requieren de canales de comunicación anónimos para enviar el voto. Como se sabe, en la práctica ello es muy difícil de conseguir.

Conclusión

Pese a tratarse del sistema más eficiente de los tres, el no poder garantizar la verificabilidad universal constituye un hándicap muy importante para su implantación.

Por ello, a nivel práctico el esquema de firma ciega es el que menos interés está suscitando de los tres explicados en este punto entre las soluciones del VER.

b) Encriptación homomórfica

En los sistemas de encriptación homomórfica, se explotan las propiedades homomórficas para poder realizar operaciones sobre votos encriptados sin tener que desencriptarlos individualmente con anterioridad. Para ello, se utilizan esquemas con propiedades homomórficas aditivas, tales como ElGamal Exp. [132] o Paillier [82].

Al poder operar sobre votos encriptados, éstos se pueden enviar a través de canales públicos, a diferencia del caso anterior de los esquemas de firma ciega en los que era necesario un canal anónimo.

En concreto, los votos son encriptados de tal manera que cada opción de voto tiene asignado un 1 o un 0, dependiendo de si ha sido seleccionada o no respectivamente. Cada valor se encripta individualmente y por tanto un voto está formado por tantos textos cifrados (ciphertexts en inglés) como opciones de voto tenga la elección.

Para el recuento de votos, simplemente se suman los textos cifrados (ciphertexts) de los votos sin necesidad de desencriptarlos utilizando las propiedades homomórficas de los esquemas arriba citados.

Como los votos no se desencriptan individualmente, la privacidad del votante no está comprometida. La contrapartida es que hay que realizar pruebas (Schnorr o NIZKP) para comprobar que los votos están formados correctamente. De no ser así, un votante malicioso podría introducir por ejemplo 15 como valor en su opción de voto y contar como 15 sufragios emitidos.

La necesidad de introducir dichas pruebas y de generar encriptaciones para todas las opciones de voto (y no solo la(s) seleccionada(s) por el votante) hacen que estos esquemas sean plausibles en elecciones con un número limitado de opciones.

Ventajas

El proceso de recuento es muy eficiente al requerir únicamente la multiplicación de los ciphertextos sin desencriptar. Además no hay que esperar al cierre de las urnas para comenzar el recuento.

Por último, se pueden implementar fácilmente técnicas de desencriptado de umbral (*threshold decryption*, explicadas en el punto 2.2.4.5) o distribución, muy útiles para aumentar la seguridad del sistema de VER.

Inconvenientes

Los votantes deben demostrar que sus votos encriptados contienen un voto válido, por lo que deben computar ZKP en sus equipos propios de sobremesa.

La práctica ha demostrado que sistemas suficientemente confiables de pruebas de conocimiento cero tienen un coste computacional difícilmente asumible por equipos estándar a nivel de usuario.

En lo que respecta a la autoridad, ésta podría desencriptar votos con anterioridad al recuento. Este problema se suele resolver utilizando esquemas distribuidos o de umbral, en los que las claves privadas se distribuyen entre varias partes para evitar precisamente que la privacidad pueda comprometerse.

No obstante, la implementación práctica de sistemas de distribución suficientemente seguros acarrea también un mayor coste de implantación comparado con los esquemas basados en mix-nets.

Otro punto conflictivo radica en que el coste de verificación tiene una fuerte correlación con el número de candidatos, por lo que los esquemas de encriptación homomórfica son más ineficientes cuantas más opciones de voto haya [221].

Por último, en esquemas homomórficos aditivos (Elgamal, Paillier), la distribución de claves es especialmente compleja porque utilizan la factorización como *trapdoor* [221] (la desencriptación de los votos con Elgamal exponencial es del tipo $g^{v_1+v_2+\cdots+v_n}$).

Conclusión

Los esquemas de voto que implementan esquemas de encriptación homomórfica son uno de los dos más utilizados (junto con los sistemas basados en mix-nets).

Aportan la gran ventaja de la eficiencia a la hora de recontar votos al no ser necesaria la desencriptación de los votos uno a uno antes de proceder al recuento.

Por otro lado, todavía persisten una serie de problemas: el alto consumo computacional de realizar ZKP en equipos no especializados del votante, la baja eficiencia en votaciones con multitud de candidatos y el alto coste de algunas implementaciones de homomorfismo aditivo.

A pesar de ello, la encriptación homomórfica supone una solución muy interesante para elecciones simples con pocas opciones y/o referéndums. Paralelamente, van apareciendo protocolos de ZKP y primitivas homomórficas más eficientes, resolviendo poco a poco las ineficiencias inherentes a esta tipología de esquemas.

En conclusión, actualmente la encriptación homomórfica es una de las dos principales "familias" de sistemas de VER junto con las mix-nets. La producción científica en la materia está en continua mejora y evolución por lo que es de esperar que sigan apareciendo soluciones del VER con implementaciones más eficientes de primitivas homomórficas. Por el momento, se erigen como la solución más interesante para votaciones con pocos candidatos y donde la eficiencia y rapidez en el recuento sean factores críticos.

Para un estudio más detallado sobre la comparación entre esquemas de cifrado homomórfico y de mix-nets, recientemente Kulyk et al. han presentado un estudio y un prototipo de herramienta de comparación de esquemas en [363], si bien todavía limitado en su alcance y características.

c) Mixnets

El tercer gran grupo de esquemas criptográficos de aplicación práctica a los sistemas de VER es el de las denominadas mix-networks o mix-nets.

El sistema de mix-nets presenta una secuencia de servidores donde cada uno de ellos recibe como *input* un conjunto de textos cifrados, los re-encripta y reordena o "baraja" aleatoriamente y envía el resultado como *input* al siguiente servidor.

De esa manera se rompe el vínculo entre el votante y el voto encriptado antes de la desencriptación del voto, de manera análoga a lo que sucede en unas votaciones tradicionales. Ello es crítico puesto que hoy en día en internet no es difícil rastrear una dirección IP, lo cuál podría poner en peligro el anonimato del votante.

De los dos tipos de mix-nets (de desencriptación y de re-encriptación), el que se está implementando en las soluciones de VER más recientes es el de re-encriptación.

El otro caballo de batalla en los esquemas de mix-nets es la verificabilidad del mezclado o barajado que realiza el servidor. Puesto que el *output* de cada servidor no puede ser relacionado con el *input* del mismo, hay que realizar una serie de pruebas que garanticen que cada mix no haya borrado o añadido votos irregularmente.

Desde 2005 han surgido una serie de técnicas para garantizar la verificabilidad [222, 223, 224, 225] que han sido utilizadas en algunas soluciones relevantes de VER que se estudian en el capítulo 5 como nVotes y Civitas entre otras.

Ventajas

Permiten romper de maner eficiente el vínculo votante – voto, son más versátiles que los esquemas basados en encriptación homomórfica, se pueden utilizar en elecciones con un mayor número de opciones de voto de una manera más eficiente, requieren poca capacidad de computación por parte del votante al no realizarse ZKP en su equipo y es fácil conseguir la propiedad de verificabilidad universal ya que los *outputs* de cada nodo de la mix-net son de acceso público.

Inconvenientes

La contrapartida a la menor complejidad computacional en la parte del votante implica una necesidad de recursos muy superior al caso de la encriptación homomórfica por parte de las autoridades de recuento y de los nodos de la mix-net.

Cada nodo debe computar multitud de encriptaciones y desencriptaciones y al mismo tiempo realizar las correspondientes ZKP para demostrar que se han mezclado los votos de una manera correcta.

Otra de las debilidades la constituye el hecho de que no se puede comenzar el recuento hasta que no hayan votado todos los votantes, lo que para elecciones a gran escala puede acarrear retrasos importantes.

Por último, el esquema de mix-nets es más vulnerable a ataques del tipo DoS al tener que estar todos los servidores de la mixnet disponibles para el recuento.

Conclusión

El esquema de mix-nets es, como se ha apuntado, uno de los dos principales en la implementación práctica de sistemas de VER.

Aporta una serie de ventajas que lo hacen más adecuado para elecciones con un gran número de opciones a votar o cuando no se quiere sobrecargar al votante con requerimientos computacionales.

Por otra parte, los mayores requerimientos en términos de capacidad de servidores mix-net, así como la vulnerabilidad a ataques de *Denial of Service* y el hecho de tener que esperar a que todos los votos se hayan recibido para comenzar el recuento constituyen una serie de problemas que todavía no están resueltos y que permiten un amplio margen de mejora.

En concreto, muchos de los últimos avances en el campo de las mix-nets se centran en el caso concreto de esquemas de encriptación maleables (las pruebas de conocimiento no aumentan con cada servidor por el que se pasa) los cuales, pese a suponer teóricamente un salto cualitativo, todavía no están desarrollados e implantados de manera práctica en ningún sistema de VER como apuntan Bernhard et al. en [226].

Entre los sistemas que se estudiarán en el apartado 5 y que utilizan el esquema de mix-nets se encuentran: Civitas, Scytl y Helios entre otros.

En el plano práctico, cabe destacar que los principales esquemas de implementación del VER (firma ciega, encriptación homomórfica y mix-nets) no son excluyentes entre sí y puede darse el caso de que se usen varios en un mismo sistema de VER [260, 292].

Para terminar con este último apartado sobre los *building blocks* criptográficos y matemáticos que sirven de base para el resto de la tesis, es conveniente recordar que actualmente se están llevando a cabo líneas de investigación de un gran interés en el campo de la criptografía y su aplicación al voto electrónico.

Si bien tanto la velocidad con la que se producen los avances como la calidad de la producción científica auguran una importante mejora en las técnicas criptográficas y la seguridad informática, todavía serán necesarios varios años de investigación para que sus beneficios sean de aplicación práctica en sistemas de VER.

2.3 Requerimientos del Voto Electrónico Remoto

Una vez analizados con suficiente detenimiento los conceptos, consideraciones, definiciones y *building blocks* criptográficos del Voto Electrónico Remoto en los apartados anteriores, se está en disposición de individuar y definir los requerimientos del mismo.

En el presente apartado 2.3 se abordan los requerimientos ideales que un sistema de VER debe cumplir. En esta tesis, se ha pretendido separar lo que son propiedades inherentes a un proceso democrático (corrección, justicia, elegibilidad) de las propias del sistema de VER. Por ello no se le pide al software que sea "correcto" "justo", o "elegible", puesto que son conceptos asociables al propio proceso electoral en sí amen de difícilmente cuantificables.

Además, una metodología cuyos criterios fuesen únicamente teóricos o basados en modelizaciones matemáticas, correría el riesgo de no tener una aplicación práctica y directa a sistemas de VER existentes, como apuntan P. Locher et al. en [235].

Es por ello que en el apartado 3 de la tesis se estudian en detalle las experiencias más relevantes hasta la fecha de VER a nivel internacional tanto en elecciones legislativas de carácter vinculante como en otro tipo de elecciones. Se detallan protocolos, algoritmos y metodologías de seguridad, pero sobre todo errores cometidos y ataques sufridos a lo largo de los últimos 15 años en más de 500 elecciones para un total de más de 6 millones de votos emitidos a través de sistemas de VER.

Ese conocimiento práctico permite enriquecer los requerimientos del presente punto 2.3 con otras necesidades descubiertas gracias a las experiencias realizadas sobre millones de votos emitidos sobre plataformas de VER. Esos requerimientos prácticos añadidos se definen y detallan en el capítulo 4 de la tesis, como paso previo a la definición completa de la metodología de evaluación.

Entrando en la definición de los requerimientos de un sistema de VER, la idea básica es que un sistema de VER debe proteger simultáneamente integridad y privacidad, al menos parcialmente antagónicas entre sí [23, 1, 260, 369, 388].

El artículo 68 de la Constitución Española establece que *"El Congreso se compone de un mínimo de 300 y un máximo de 400 Diputados, elegidos por sufragio universal, libre, igual, directo y secreto"*. Análogamente, el Consejo de Europa en [358], también afirma que: *"The five key principles of electoral law are: universal, equal, free, direct and secret suffrage and they are at the root of democracy"*.

Por tanto, una tarea fundamental a la hora de establecer los requerimientos de un Sistema de VER es definir las propiedades que se correponden con los cinco principios apuntados tanto por la Constitución Española como por el Consejo de Europa.

La cuestión de la definición de los requisitos de un sistema de VER es un sub-campo poco trabajado dentro de la e-democracia. Existe numerosa bibliografía sobre las bases criptográficas, los sistemas de VER en sí o incluso las experiencias previas de i-voting. No sucede así en el caso de la metodología de transcripción de los requisitos legales en requisitos técnicos, para cualquier ámbito de las TIC, incluido el VER.

En ese sentido, los trabajos más destacados en la materia son:

- El método KORA (*Concretization of Legal Requirements* en alemán) [461]
- *Common Criteria for Information Technology Security Evaluation SO/IEC 15408:2009* [460]
- ISO 27001/IT-Grundschutz [462]

Así como el sistema de evaluación holístico que combina las 3 anteriores, elaborado Simic-Draws et al. [457] el cuál, aunque algo incompleto y sin una aplicación práctica directa, supone una interesante base sobre la que construir.

Además, las siguientes fuentes abordan la cuestión en mayor o menor medida:

- Las recomendaciones sobre certificación de sistemas de (sic) e-voting del Directorado General de Democracia y Asuntos Políticos del Consejo de Europa [456]
- El trabajo de Volkamer sobre requisitos legales del voto [459]
- El trabajo de Bräunlich et al. sobre la transformación de criterios legales en TDGs (*Technical Design Goals*) [463]
- La tesis doctoral de Neumann [458], que se apoya en [457] y [463] para identificar 16 aspectos técnicos que debería cumplir un sistema de *i-voting*.

Todos ellos conforman un punto de partida sólido pero muy centrado únicamente en aspectos legales y su encaje en criterios técnicos predefinidos, en ocasiones alejados de requerimientos y necesidades reales y concretas. Además, no se profundiza más allá de una definición genérica de cada uno de ellos, por lo que su uso práctico es limitado.

En el anexo C de la presente tesis se detallan los *Technical Design Goals* (TDG) u objetivos técnicos de diseño de Bräunlich et al. [463] sobre los que se basa Neumann en su tesis doctoral de 2016 [458] para delimitar los requerimientos técnicos del voto electrónico:

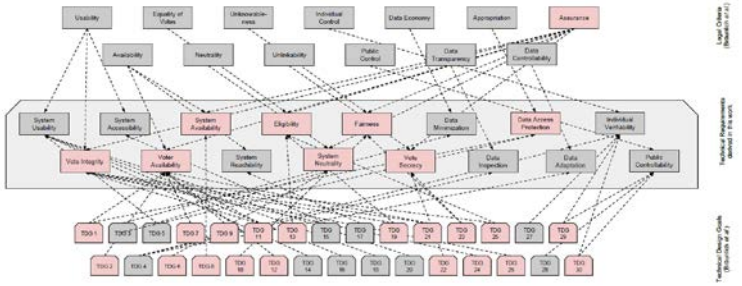


Figura 14: Definición de criterios técnicos del voto electrónico de Neumann [458] sobre la base de KORA [461] y Bräunlich et al. [463]

En la presente tesis se utilizan el conjunto de metodologías y fuentes descritas en la página anterior como punto de partida sobre el que articular una metodología de evaluación más práctica y de aplicación directa a los sistemas de VER más relevantes hasta la fecha.

En concreto, se ha trazado en un primer momento el paralelismo entre las 5 propiedades de cualquier votación democrática según la Constitución española y el Consejo de Europa:*" universal, libre, igual, directo y secreto"* y una categoría de requisitos novedosa que se plantea en esta tesis: los requisitos imprescindibles o *sine-qua-non* porque son ellos quienes salvaguardan las 5 características definidas en la ley.

Se trata de la verificabilidad (entendida como verificabilidad extremo a extremo [51, 93, 77, 3, 359, 369] y verificabilidad de la elegibilidad [389]), detallada en el apartado 2.2.2 de la tesis en lo que respecta a los aspectos de universal, libre, igual y directo y la privacidad (entendida en su acepción más exigente, la resistencia a la coerción [104]) como responsable de la propiedad de voto secreto.

El principal problema radica en que estas dos propiedades son al menos parcialmente antagónicas entre sí [23, 1, 260, 388, 438], por lo que no se puede conseguir una verificabilidad total sin sacrificar al menos en parte la privacidad y viceversa. La solución consiste en llegar a un compromiso o *"tradeoff"*: lo suficientemente verificable y privado para poder ser usado en elecciones reales. Por supuesto, dependiendo de la tipología de elección y el grado de uso del VER, dicho "umbral suficiente" de verificabilidad y privacidad variará.

Como primer paso, apoyándose en el método KORA [461], el *Common Criteria for Information Technology Security Evaluation SO/IEC 15408:2009* [460], la ISO 27001/IT-Grundschutz [462], el sistema que combina las 3 anteriores por Simic-Draws et al. [457] así como el trabajo de Bräunlich y Neumann [463, 458] se obteniene una radiografía que supone el punto de partida para la elección de los criterios tradicionales de evaluación.

Conviene aclarar que tal y como explican los desarrolladores de las metodologías citadas, los criterios y TDGs en ocasiones se solapan entre sí y su ámbito abarca más de una característica. Ello es la constatación de que los requisitos del VER son parcialmente contradictorios y en situaciones se sobreponen.

En la siguiente figura se puede apreciar también que los requisitos *sine-qua-non* no abarcan la totalidad de los criterios legales ni de los TDGs, apuntalando la necesidad de una serie de características adicionales que suponen el objetivo último del presente apartado 2.3:

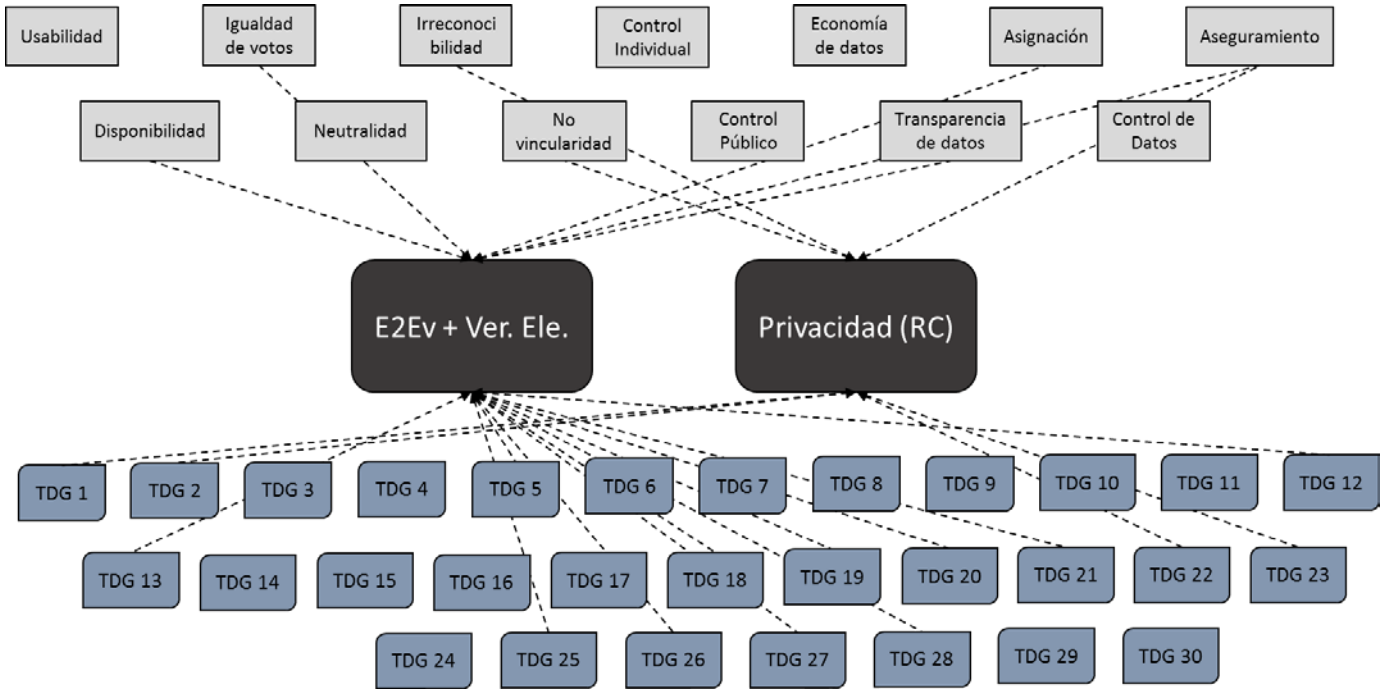


Figura 15: Relación entre los criterios *sine qua non* de un sistema de VER con la metodología KORA [461], Braunlich et al. [463] y Neumann [458]

El diagrama anterior posee un carácter abstracto, interesante como punto de partida pero alejado de lo que debe aportar una metodología práctica.

De hecho, en [458] se realiza una evaluación sistemática pero limitada de dos esquemas de VER (Estonia y Polyas). Se aborda la tarea desde un punto de vista probabilístico, centrándose únicamente en un componente matemático a alto nivel. Por ello, más que de análisis de sistemas de VER en su conjunto, se debe de hablar del estudio probabilístico de esquemas de VER, como el autor reconoce en el apartado de limitaciones de su tesis.

Por otra parte, en la figura previa quedan un número importante de requerimientos legales y TDGs sin asignación, por lo que es necesario ampliar el número de criterios para cubrirlos todos ellos y además dotar a la metodología de un enfoque práctico, dirigido al sistema de VER en su conjunto y no únicamente al esquema subyacentes.

A la hora de elegir los requerimientos tradicionales, se ha realizado un estudio tanto cualitativo como cuantitativo de la bibliografía más relevante de la materia, a cargo de Panizo, Popoveniuc, Benaloh, Rivest, Ryan, Volkamer, Zissis y Lekas, Puiggalí, Jonker o Simons, [4, 51, 68, 93, 172, 23, 1, 260, 359, 369, 388], añadiendo los siguientes:

Inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software y escalabilidad.

Una vez agregados y aplicando de nuevo la metodología KORA [461], Bräunlich [463] y Neumann [458], la figura queda de la siguiente manera:

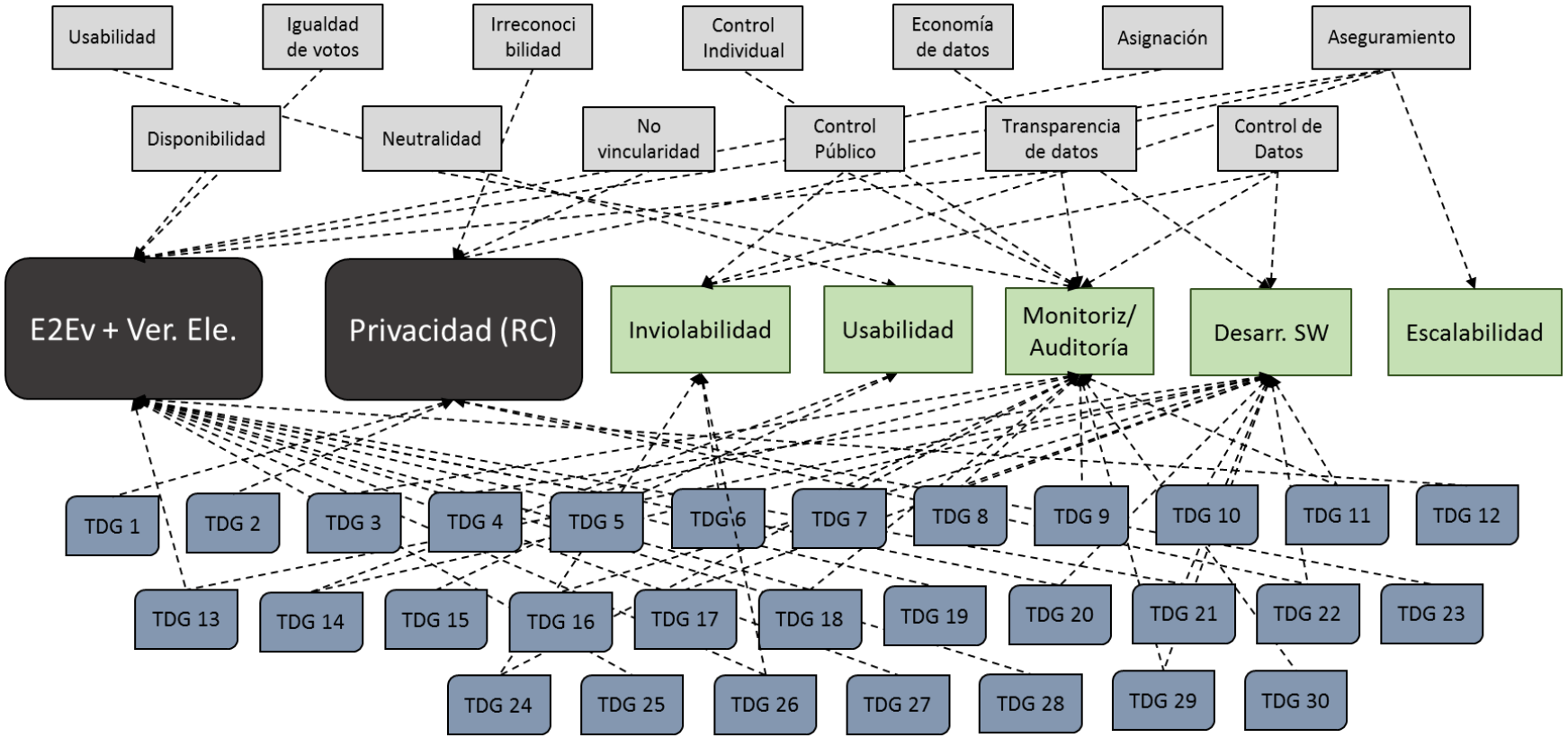


Figura. 16: Relación entre requisitos tradicionales del VER en la presente tesis, la metodología KORA [461], Braunlich et al. [463] y Neumann [458]

De una forma menos gráfica pero más fácil de interpretar, la relación entre los criterios legales de la metodología KORA [461] y Bräunlich [463], las *Technical Design Goals* de Neumann y Bräunlich y los criterios tradicionales del VER en la presente tesis es:

Requisito	**Criterio Legal [461, 463]**	***Technical Design Goals* [458, 463]**
E2Ev + Ver. Eleg.	Igualdad de votos, disponibilidad, asignación, aseguramiento, transparencia de datos	TDG 5, TDG 12, TDG 19, TDG 20, TDG 21, TDG 25, TDG 26, TDG 27, TDG 28
Privacidad (RC)	Irreconocibilidad, aseguramiento, no vincularidad	TDG 1, TDG 2, TDG 22, TDG 23
Inviolabilidad	Control público, control datos, aseguramiento	TDG 6, TDG 24, TDG 26
Usabilidad	Usabilidad	TDG 14, TDG 15
Monit./Audit.	Control de datos, Control Individual, Control público, neutralidad, transparencia de datos	TDG 3, TDG 5, TDG 7, TDG 8, TDG 9, TDG 11, TDG 18, TDG 24, TDG 29, TDG 30
Desarr. SW	Economía de datos, control de datos	TDG 4, TDG 8, TDG 10, TDG 11, TDG 13, TDG 14, TDG 16, TDG 17, TDG 20, TDG 21, TDG 22, TDG 29
Escalabilidad	Aseguración	

Tabla 1: Relación entre la metodología KORA [461], Bräunlich [463] y Neumann [458] con los requerimientos tradicionales del VER en la presente tesis.

Por otra parte, para facilitar la identificación y el posterior análisis, se asigna a cada uno de los puntos a valorar dentro de cada requisito un código compuesto por la inicial en mayúsculas del mismo junto a un número entero creciente empezando con el 1.

Los códigos de los requisitos de un sistema de Voto Electrónico Remoto son:

Requisito	*Formato de codificación*	*Ejemplo*
Inviolabilidad	I-*n*	I-4, I-6, etc.
Usabilidad	U-*n*	U-2, U-4, etc.
Monitorización/Auditoría	MA-*n*	MA-1, MA-7, etc.
Desarrollo Software	DSW-*n*	DSW-2, DSW-5, etc.
Escalabilidad	E-*n*	E-1, E-2, etc.

Tabla 2: Codificación requisitos VER

En los siguientes párrafos se define en detalle cada uno de los requerimientos:

2.3a Verificabilidad extremo a extremo ($E2E_v$) + Verificabilidad de la elegibilidad

Como se apuntó en el apartado 2.2.2, "Un sistema de VER es E2Ev si cada voto es i) emitido como estaba previsto ii) guardado como se ha emitido y iii) contado como se ha guardado." y los votos han sido emitidos por votantes con derecho a votar [51, 93, 77, 3, 359, 369].

O por su versión en inglés, más habitual en el campo:

i) Cast as intended: Los votantes pueden tener acceso a una evidencia de que su voto encriptado refleja fielmente su elección. Se requiere por tanto que el proceso de votación utilice canales seguros de comunicación para garantizar que ningún ataque cambie el voto y que todo ello sea verificable.

ii) Recorded as cast: Los votantes o los delegados pueden comprobar que sus votos han sido debidamente incluidos, siendo verificables los valores encriptados concretos de los votos en una lista pública o *Bulletin Board* de votos encriptados. En este caso se suele facilitar al votante algún tipo de información a modo de "recibo" para más tarde comprobar en el listado que su voto fue registrado correctamente.

iii) Counted as recorded: Cualquier persona puede verificar que todos los votos encriptados publicados son correctos y se han incluído en el recuento, si bien no hay forma de sabe cómo voto ningún votante. Para ello se utilizan mix-nets o cifrado homomórfico.

Dichas características, junto con la denominada *verificabilidad de eligibilidad* por Kremer et al. [389] (los votos han sido enviados por votantes con derecho de voto) conforman la **verificabilidad extremo a extremo o E2Ev**.

Aparte de lo ya mencionado, el votante y/o los observadores que quisieran verificar su voto, deben ser capaces de hacerlo independientemente del software usado y sin requerir conocimientos específicos de criptografía.

Tal y como se ha detallado en el apartado 2.2.2 dedicado en exclusiva a la verificabilidad extremo a extremo, se están produciendo interesantes avances de cara al establecimiento de una definición estándar, universalmente válida, modelizable y automatizable de la E2Ev, tal y como el esquema KTV [360, 381].

Aún así, todavía no se ha llegado a ese punto y por tanto la evaluación de la E2Ev se debe realizar caso por caso dependiendo de la definición aplicada en cada sistema de VER.

Para una profundizar a mayores en la E2Ev, su evolución temporal, variantes y definiciones, se recomiendan [50, 51, 3, 93, 77, 359, 360, 362, 365, 369, 390].

2.3b Privacidad/resistencia a la coerción (RC)

Como se ha comentado en el punto 2.2.3, la privacidad es, junto con la verificabilidad extremo a extremo, la cualidad imprescindible de los sistemas de VER puesto que la votación se produce en un entorno no controlado y sobre un equipo convencional, incrementa la complejidad y los problemas asociados a garantizarla para el votante y su voto.

La privacidad del VER está categorizada en 3 niveles de menor a mayor exigencia:

- Privacidad del voto: El voto de un votante no es revelado a nadie.
- Ausencia de recibo: Un votante no puede obtener información (p. ej un recibo) que pueda ser utilizado para probar a un atacante el sentido del voto.
- Resistencia a la coerción: Un votante no tiene manera de demostrar al coercionador cómo voto, incluso si ambos estuviesen colaborando activa y voluntariamente.

El caso de la venta de votos (incluso en plataformas como Amazon) se ha dado en numerosos casos y se trata de un riesgo real y documentado [227, 228].

En cuanto a cuál de los 3 niveles debería ser el mínimo necesario en unas elecciones, Hirt y Sako fueron los primeros en demostrar que ausencia de recibo es insuficiente en [108].

Por tanto, el nivel requerido para unas elecciones vinculantes en el ámbito político es el tercero y más exigente: la resistencia a la coerción.

Fueron Juels, Catalano y Jakobsson quienes introdujeron el concepto de resistencia a la coerción (RC en adelante) por primera vez en 2002 [70] y posteriormente lo depuraron en 2005 [63] y 2010 [104].

Pese a que el esfuerzo fue notable, especialmente en el caso de soluciones de VER basadas en mix-nets, su protocolo parte de unas premisas de difícil implementación real:

- Se asume que no hay corrupción posible en la fase de verificación del votante. Las credenciales se reciben por un canal incorruptible.
- Se estima que los generadores de recibos no son corruptibles.
- No se valoran ataques de DDos
- El coste computacional limita fuertemente su uso para elecciones con un elevado número de participantes.

Por ello, Juels et al. avanzaron hacia un notable acercamiento a un sistema de voto resistente a la coerción, sin llegar a conseguirlo completamente en un entorno real en la actualidad.

Otra línea de investigación relevante de RC se debe al experto en ciberseguridad Josh Benaloh [3], autor de un paper de 2013 en el que apunta al avance de la tecnología como freno al desarrollo de nuevas propuestas que cumplan con la propiedad [106].

Según Benaloh, el votante corrupto (bien sea obligado o por voluntad propia) puede llevar dispositivos portátiles de visualización, grabación o realidad aumentada para que el coercionador pueda comprobar que se vota de acuerdo a lo acordado como si estuviese votando él mismo incluyendo las *Google Glass* como ejemplo práctico.

En cuanto a los avances, Achenbach et al. han introducido en 2015 el concepto de revotación negable (un coaccionador no puede verificar si el votante ha vuelvo a votar para invalidad el voto bajo coacción) para facilitar la obtención práctica de sistemas resistentes a la coerción [105]. En el apartado 2.2.3 se realiza un análisis pormenorizado de su trabajo.

Ello contrasta con la línea de trabajo original de Juels et al. [104] sobre la RC, donde el votante no podía negar el hecho de que había votado varias veces. Ello se debía a que, para identificar los votos duplicados, se utilizaba la tecnología PET [110], en la que el *output* es un único bit, que puede fácilmente revisar el coaccionador.

En Achenbach et al. [105] por el contrario, se utilizan las propiedades homomórficas de la encriptación de las credenciales del votante para comprobar si hay más de un voto con la misma credencial, utilizando el sistema de *Encrypted Plaintext Equivalency Texts* (EPET). En él, el *output* es la encriptación de un bit con su correspondiente *padding* o relleno, por lo que el atacante no puede comprobar si el votante ha vuelto a votar o no.

Cabe puntualizar no obstante que, pese a constituir posiblemente la mejor solución hasta la fecha, Achenbach et al. también parten de una serie de premisas (análogas a las de [104] más alguna otra referente a reloj del ordenador desde el cuál se vota]) que no son totalmente aplicables a unas elecciones reales.

Por último y relacionado con las premisas de las que parten los sistemas que aspiran a ser resistentes a la coerción, tradicionalmente se han usado los supuestos de Dolev-Yao [53, 109], en los que se asume que las primitivas criptográficas son perfectas.

Ello es, como sabe el lector, en la práctica muy difícil de demostrar y ha dado lugar a multitud de ataques, explicados en el punto 2.4 [26, 27, 34].

En ese sentido y en cuanto a avances en la materia, en el año 2013, M.C. Carlos et al. desarrollaron un nuevo modelo de análisis de amenazas [229]. La novedad que aporta es que aborda los supuestos de los ataques y los atacantes de un modo dinámico (dependiendo del entorno), práctico y que tiene en cuenta las limitaciones de unas comunicaciones no perfectas y de los errores humanos. Se denomina en inglés *"Ceremony Analysis"*.

Con posterioridad, en el año 2015 se ha presentado su primera aplicación inicial a un sistema de Votación Electrónica Remota [230]: Helios (ver punto 5.2 para detalles).

Pese a que se están produciendo importantes avances, es necesaria una mayor investigación práctica antes de poder afirmar que se ha encontrado una solución completa a la cuestión de la RC.

En conclusión, la RC es la salvaguarda de la privacidad del votante y su voto, incluso aunque éste quisiera colaborar conscientemente con un atacante.

Por ello y de una manera análoga a la verificabilidad extremo a extremo (E2Ev), la resistencia a la coerción es un requisito *sine qua non* para que una solución de VER sea considerada apta para su uso en elecciones VAP.

Para más detalles sobre la RC, referirse al apartado 2.2.3 de la presente disertación.

2.3c Inviolabilidad (I-*n*)

Entendida como la idea de que tanto el software como los sistemas auxiliares deben estar adecuadamente protegidos por protocolos de autenticación suficientemente seguros (passwords, claves de acceso de un solo uso, tarjetas inteligentes, documentación electrónica etc.) y de probada confiabilidad, evitando accesos a través de terceras aplicaciones y/o servidores vulnerables. (I-1)

En la mayoría de los trabajos de referencia sobre los requerimientos del VER se incluye de una forma u otra la inviolabilidad [1, 4, 23].

Se valora también la existencia de protocolos detallados y validados a aplicar en caso de que la inviolabilidad fuese puesta en compromiso o incluso rota. (I-2)

Deben de existir herramientas para localizar y reconstruir el rastro del ataque, así como copias de seguridad de toda la información relevante de la votación separadas totalmente de los componentes conectados del sistema de VER. (I-3)

Para reforzar la seguridad, se valorará positivamente:

- Que el control de los nodos críticos se lleve a cabo de manera distribuída y nunca por un grupo de personas relacionadas o por los mismos equipos de personas. (I-4)
- La existencia de protocolos de *risk assessment* o evaluación de riesgo y de modelado de amenazas o *threat modeling.* (I-5)
- La implementación en la medida de lo posible principios de modularidad para que los potenciales errores o ataques queden lo más confinados posible. (I-6)
- La correcta actualización de las acciones y políticas arriba. (I-7)

A diferencia de las E2Ev y la RC, el grado de inviolabilidad de un sistema de VER puede implementarse en mayor o menor medida y por tanto tiene una ponderación numérica asignada, al igual que los siguientes criterios del presente apartado 2.3.

2.3d Usabilidad (U-*n*)

Uno de los mayores beneficios que puede aportar la introducción de soluciones de VER es la de hacer más accesible el voto a determinados colectivos de votantes para los que el proceso tradicional de votación les resulta una tarea especialmente ardua. Entre ellos se encuentran las personas de avanzada edad, movilidad reducida, con discapacidades, residentes en zonas remotas o en el extranjero etc.

En numerosas ocasiones, el votante no tiene conocimientos específicos en TIC, por lo que la usabilidad del sistema VER se antoja crucial, como ha quedado patente en numerosa bibliografía de referencia en la materia [4, 23, 51, 68, 93, 172].

Se valorará por tanto la simplicidad en los procesos de autenticación, voto y verificación (sin poner en compromiso la E2Ev y la privacidad). (U-1)

En cuanto a las personas con discapacidades físicas y los colectivos sin alfabetizar, (la ley para el caso de las personas con discapacidad mental es ambigüa y objeto de alguna polémica), es necesario que sean considerados como colectivos de interés especial (U-2).

Tanto la *"Convención de las Naciones Unidas sobre Derechos de Personas con Discapacidades"* [259] en su artículo 29 como el Consejo de Europa incluyen también entre sus recomendaciones varias menciones al acceso en igualdad de condiciones al derecho de voto. En el caso concreto del Consejo de Europa y *su "Legal, Operational and Technical Standards for e-voting"* de 2004, incluye detalles de usabilidad y accesibilidad en 10 estandares [54]. De ellos, el 3: *"Los sistemas de e-voting serán diseñados, en la medida de lo posible, para maximizar las oportunidades que dichos sistemas pueden proveer a personas con discapacidades"* y el 63: *"Se proporcionará a los usuarios, cuando sea demandado y posible, facilidades adicionales, tales como interfaces especiales u otros recursos equivalentes, tales como asistencia personal. Los servicios de usuario deberán estar en línea en la medida de lo posible con los principios establecidos en la Web Accesibility Iniciative (WAI)"* son los más explícitos en su demanda de acercar el VER a los colectivos con más limitaciones..

También C.Z. Acemyan et al. por una parte y K. Summers et al. por otra, han presentado sendos papers en los que se analiza la usabilidad y la accesibilidad en el VER [231, 232].

En concreto en [231], se analiza la usabilidad del sistema Helios y se descubre que un 38% de los participantes no pudieron votar correctamente y que sólo un 22% validó su voto.

Otros puntos a tener en cuenta son los siguientes:

1) El sistema de VER debe mostrar de una manera clara e inequívoca cuándo el voto ha sido contabilizado con éxito y el proceso de votación haya concluido. Existen precedentes de fallos destacados en este punto [235]. (U-3)
2) La privacidad e integridad del voto tienen preferencia sobre la usabilidad en caso de conflicto. (U-4)
3) La facilidad de los administradores de crear y gestionar unas elecciones, incluso sin conocimientos específicos en TIC (U-5)

2.3e Monitorización/auditoría (MA-*n*)

Se ha explicado en diversos puntos de la tesis la dificultad de replicar el proceso de votación estándar con sistemas de VER.

Ello deriva en la necesidad de prestar una especial atención al desarrollo e implementación de protocolos de monitorización y auditoría de todo el proceso electoral, para evitar que partes interesadas puedan explotar vulnerabilidades en beneficio propio.

La monitorización/auditoría debe velar por el control contínuo sobre el sistema y todas sus partes implicadas, cumpliendo tres requisitos principales: ser externa, independiente y distribuida para minimizar los riesgos de colusión entre las partes responsables. (MA-1)

Su inclusión entre los requisitos tradicionales del VER es una constante en la bibliografía de referencia [4, 23, 172, 260].

Se valorarán además los siguientes apartados:

1) Existencia de un protocolo de auditoría desde la fase de diseño. La opción de implementarlo con posterioridad no es recomendable. (MA-2)
2) Las herramientas a tal fin deberán también controlar que las estrategias de *risk assessment* y *threat modeling* se cumplan (si las hubiese), especialmente en el caso de que se produjese algún contratiempo o ataque. (MA-3)
3) Se generarán informes periódicos de actividad inalterables e imborrables que serán almacenados en áreas físicamente separadas del resto de instalaciones y custodiadas de manera distribuída por efectivos distintos a los encargados de la seguridad del resto de actividades de la votación (registro, votación, recuento etc.). (MA-4)
4) La monitorización/auditoría debe cubrir todas las fases de la elección, desde la obtención del censo electoral a su comprobación, envío de credenciales, votación, recuento, protocolo de riesgos y ataques, mantenimiento del sistema etc. (MA-5)
5) La información deber ser detallada, bien documentada, en el formato pertinente y respetar la privacidad del votante. (MA-6)
6) La existencia de "banco de pruebas" o test a disposición si fuese necesario (p. ej. En caso de sospecha de un ataque durante las elecciones, para comprobar que las salvaguardas están funcionando correctamente etc.). (MA-7)

7) La labor del auditor debe ser: independiente y distribuída de tal manera que ninguna de ellas tenga relación con el resto de autoridades/administradores de las elecciones ni con ninguno de los partidos o candidatos que concurren a los comicios. (MA-8)
8) En la eventualidad de un fallo grave, es de vital importancia auditar el ataque y la forma en la que se reaccionó. Debe por tanto existir un protocolo de auditoría de riesgos, errores y ataques y otro de auditoría del propio sistema de auditoría. (MA-9)
9) En caso de producirse un ataque exitoso en el que la aplicación de los protocolos de auditoría implique tener que renunciar a alguna propiedad del sistema de VER, se favorecerá la privacidad del votante y su voto incluso si se deben cancelar completamente las elecciones y tener que postponerse o repetirse. (MA-10)

Las referencias destacadas sobre la auditoría de sistemas de VER son: [233, 234].

2.3f Desarrollo software (DSW-*n*)

Aparte de los requerimientos habituales de diseño, implementación y documentación de ingeniería del software para cualquier tipología de programa (DSW-1), existen una serie de puntos de especial interés en el caso del Voto Electrónico Remoto:

1) Se aplicará un enfoque distribuido, en especial en las tareas de *setup* de las elecciones y en las modificaciones críticas del sistema. El objetivo es que no exista ninguna autoridad que pueda realizar cambios críticos de manera unilateral. (DSW-2)
2) Pese a tratarse de un software de gran complejidad, debe de ser simple de usar y disponer de una guía de usuario y administrador debidamente documentada y disponible con suficiente antelación. El objetivo es evitar problemas de usabilidad ya comentados con anterioridad [231]. (DSW-3)
3) Idealmente, debería existir un sitio web seguro con una sección clara e intuitiva de FAQ. (DSW-4)
4) La información sobre las distintas opciones políticas se debe presentar de una manera totalmente objetiva e imparcial. (DSW-5)
5) El sistema de votación no debe suministrar al votante ninguna prueba de su voto que pudiese servir para deducir la opción elegida. (DSW-6)
6) El sistema debe de garantizar la privacidad del voto en todos los pasos del sistema de VER, no permitiendo reconstruir el vínculo entre voto y votante. (DSW-7)
7) El proceso de votación tiene que poder cancelarse en cualquier momento sin conservar ningún tipo de información sobre la(s) opción(es) que se hubiesen seleccionado. (DSW-8)
8) El software debe ser testado en plataformas, sistemas operativos y navegadores que representen más de un 1% de la cuota de mercado. (DSW-9)
9) El software no debe permitir el acceso a través de ningún programa ajeno, p. ej. de *social media,* ni incluir links a direcciones u otros programas gestionados por servidores que no estén debidamente controlados. Las elecciones en Nueva Gales del Sur en

2015 son un buen ejemplo de cuán crítico es este punto. (Referirse a 3.2.5 para más detalles). (DSW-10)

10) En lo referente a las primitivas criptográficas (sobre las que se asienta buena parte de la privacidad, integridad y seguridad del sistema de VER), debe estar claro su diseño e implementación y se testará con anterioridad en condiciones más exigentes que las propias elecciones. (DSW-11)

    Además, suele ser la parte del código que más recursos consumen, por tanto se debe verificar con el máximo rigor (y utilizando equipos análogos a los que se van a utilizar en las elecciones) que las tareas se realizan en un tiempo razonable y compatible con el normal desarrollo de unas elecciones.

11) Es recomendable que equipos de investigadores independientes tengan acceso al código fuente para revisarlo y comprobar la ausencia de *bugs* que pudiesen comprometer la seguridad del sistema de VER. (DSW-12)

    El punto de vista de las empresas desarrolladoras de software respecto al código fuente como un activo de gran valor es comprensible. No obstante, existen opciones aceptables como permitir el acceso al código a grupos independientes de investigadores previa firma de un NDA (*non-disclosure agreement*), para asegurar la propiedad intelectual de la empresa a la vez que se realiza una revisión imparcial del software.

12) Se implementarán, siempre que sea posible, modelos protocolizados o estandarizados abiertos, para facilitar la interoperabilidad. (DSW-13)

13) Deberá mantenerse el sistema totalmente actualizado, sobre todo contra los ataques más habituales en sistemas de VER (*man-in-the-middle*, *eavesdropping*, *Denial of Service*, FREAK, Logjam). (DSW-14)

2.3g Escalabilidad. (E-*n*)

En numerosos puntos de la presente tesis se ha podido comprobar que uno de los mayores retos que afrontan los sistemas de VER es cubrir el *gap* entre lo que la teoría indica y lo que sucede en las elecciones reales [105, 145, 26, 27, 33, 175].

La idea fundamental pues, es comprobar la escalabilidad del sistema con pruebas reales en entornos lo más similares posible a las elecciones en las que va a ser usado. (E-1)

Se deberá prestar especial atención a las operaciones más críticas tales como la autenticación de usuarios, la encriptación y desencriptación de votos, recuento de votos, primitivas de mix-nets etc. (E-2)

El sistema debería ser testado en condiciones incluso más duras que la suposición más exigente de la votación que se maneje. El día que comienza el período de VER no puede ser la primera ocasión en la que el sistema se expone a ese nivel de exigencia (E-3).

Esta sugerencia podría parecer trivial, pero en ocasiones las constricciones de tiempo y recursos junto a una falta de planificación hacen que se llegue a esa situación no deseada.

En este requerimiento se valora pues hasta qué punto el conjunto de protocolos del sistema prevén, calculan, detallan y prueban los límites y capacidades del esquema de VER.

Se deberá indicar también claramente el tamaño o complejidad máxima de comicios que el esquema puede manejar, desde las vertiente software (según la modalidad de elecciones así como las capacidades matemáticas y criptográficas) y ex_software (infraestructura, accesos, costes, logística por un segundo canal, recursos humanos etc). (E-4)

Por último, la escalabilidad se entiende también en el sentido de la capacidad de manejar elecciones públicas vinculantes en el ámbito político, las más complejas y exigentes. (E-5)

Con el presente punto sobre la escalabilidad se concluye el apartado dedicado a los requisitos tradicionales de un sistema de VER.

Para concluir con el apartado 2.3 de requerimientos de un sistema de Voto Electónico Remoto, recordar que existen dos tipos básicos de requisitos:

- Los que por las propiedades que representan son condición *sine qua non* para que un sistema de VER sea considerado apto para su uso en elecciones vinculantes de carácter político a nivel nacional: verificabilidad extremo a extremo o E2Ev (representa y protege la integridad de sistema) y la resistencia a la coerción o RC (representa y protege la privacidad). Son los encargados de proteger las propiedades inherentes de las votaciones democráticas (universal, libre, igual, directo y secreto) y su evaluación es en términos de "cumple" o "no cumple".
- Requisitos igualmente necesarios para un sistema de VER pero que por su naturaleza se puede evaluar su grado implementación de una manera más gradual: inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software y escalabilidad.

A continuación, en el capítulo 3 se estudian en detalle las experiencias reales de uso de sistemas de VER en los distintos países (el estado del arte). De ellas se extraen una serie de conclusiones y nuevos requisitos que se añaden a los del presente punto 2.3.

Posteriormente, en el capítulo 4 se integran todos los criterios obtenidos con el doble análisis, se le asigna una ponderación a cada uno de ellos y se desarrolla el núcleo del sistema de evaluación de la presente tesis.

Para concluir con el presente apartado, se incorpora una figura final que recoge en una forma más gráfica el conjunto de requerimientos tradicionales de un sistema de VER. Para obtener el desglose completo de cada uno de ellos, referirse al anexo A.

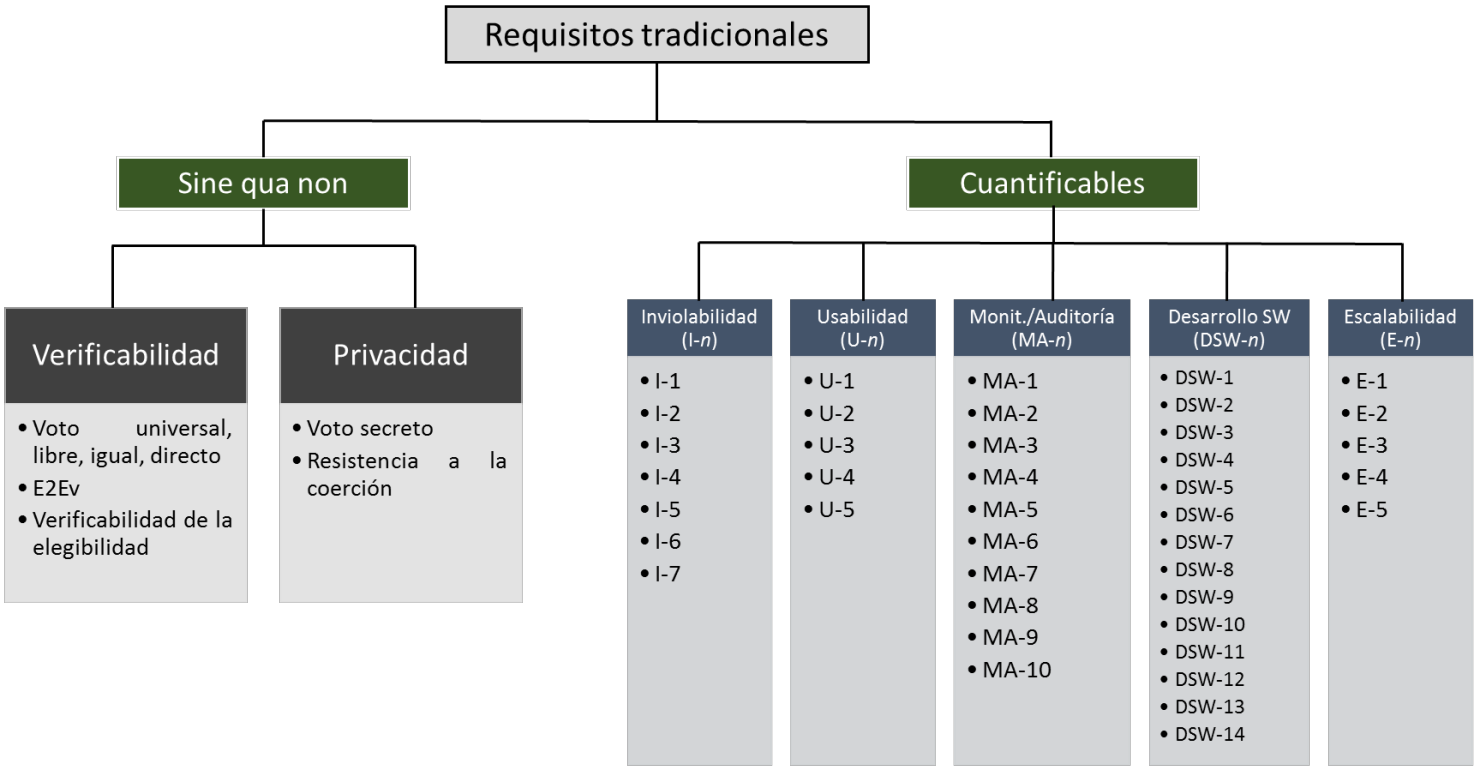


Figura 17.: Requisitos tradicionales del Voto Electrónico Remoto (VER)

2.4 Seguridad del Voto Electrónico Remoto. Definición y clasificación de ataques

Como se ha apuntado en apartados previos, el simple hecho de entrar al colegio electoral, elegir una papeleta, introducirla en un sobre y depositarla en la urna podría parecer un proceso fácilmente replicable utilizando TIC. Un análisis más pormenorizado, así como la gran cantidad de ataques exitosos que se han producido han demostrado que conseguir los niveles requeridos de seguridad y privacidad es una tarea extremadamente ardua y su estudio constituye la principal motivación de la presente tesis.

Si a ello se añade el hecho de que las elecciones son un evento que se celebra en un período de tiempo reducido, disminuye por tanto muy notablemente la capacidad de reacción y respuesta ante ataques a gran escala.

Se trata por tanto de un reto de proporciones nada desdeñables cuya resolución requiere de cantidades considerables de tiempo y recursos, tanto tecnológicos como humanos.

En el presente apartado se quiere llamar la atención sobre cuestiones referentes a la seguridad del VER yexplicar las principales tipologías de ataques, detallando algunos de ellos que por su repercusión e impacto destacan sobre los demás.

2.4.1 Seguridad del Voto Electrónico Remoto

En el voto tradicional, al ser todos los sobres y papeletas indistinguibles entre sí, desde el momento en que el votante deposita su papeleta en la urna, el vínculo entre votante y voto se rompe, por lo que es imposible asociar un voto a su votante. Se protege pues el anonimato del voto y de esa manera la privacidad del votante.

Además, las urnas son transparentes normalmente y además de los responsables de mesa designados al azar, hay una serie de observadores de distintos partidos y también independientes vigilando el cumplimiento de la normativa así como efectivos de las Fuerzas y Cuerpos de Seguridad del Estado.

Por otra parte, el proceso de recuento es abierto y transparente; en otras palabras, verificable. Hay diferentes testigos de distintos ámbitos y partidos políticos que garantizan la fiabilidad del recuento. De hecho, el proceso de recuento en sí mismo es claro e inteligible por todos los presentes.

En cambio, en el caso del VER, existe una autoridad o un conjunto de autoridades distribuidas que son las encargadas de velar para que el recuento sea correcto. Para replicar las propiedades de verificabilidad y privacidad se recurre a conceptos matemáticos y criptográficos desconocidos para la gran mayoría de los votantes.

Comparando el VER con la banca on-line, en ésta última cada usuario tiene acceso a su extracto con los movimientos producidos, fecha, cantidad, ordenante, beneficiario etc. Existe por tanto una trazabilidad de las operaciones. En el caso de que no se tuviese acceso al citado extracto; ¿cómo sabría el usuario que el banco está siendo honesto con él?

En el caso del VER, dicho extracto unificado de todos los usuarios se suele resolver como un *Bulletin Board* o Tablón de Anuncios donde aparecen votos y votantes encriptados para salvaguardar su privacidad.

El problema radica en que existe numerosa bibliografía detallando ataques exitosos contra sistemas de votación verificables extremo a extremo (E2Ev) que usan tablones de anuncios codificados, incluyendo Helios [18].

A mayores de lo arriba mencionado sobre los ataques producidos a sistemas de VER E2Ev, la rectificación de los errores en el caso de la banca on-line [17] es mucho más simple. Las operaciones fraudulentas se pueden cancelar o revertir con posterioridad, incluso abonando las necesarias multas/intereses. Tienen pues, *"vuelta atrás"*.

En el caso de unas votaciones de carácter vinculante, ¿cuál sería la solución?. ¿Quizás repetirlas? Como se ha indicado en el capítulo 1.2, el coste de unas elecciones en España supera los 120 millones de euros, sin incluir gastos de precampaña. Ello sin contar con los potenciales perjuicios derivados de la inestabilidad política (prima de riesgo país, aumento de costes de financiación del sector privado, paralización de inversiones etc.).

Por otra parte, ¿Qué se haría con las leyes aprobadas desde la formación del gobierno hasta el momento de descubrirse el ataque a las elecciones?

Pueden haber pasado meses o incluso años hasta que se descubra el ataque y sus consecuencias, habiéndose podido promulgar en ese *impasse* leyes que afectan a sectores críticos para un país (defensa, estructura del Estado, comunicaciones, tributación, economía etc.).

Por todo ello, es de una importancia capital tener las garantías necesarias antes de ofrecer la posibilidad del VER en unas elecciones VAP y además introducirlo de una manera gradual y ligada a pruebas objetivas evaluadas por empresas y expertos independientes, como en el caso de Suiza [279, 281, 283, 430].

2.4.2 Clasificación de ataques

En general, el sistema de VER ideal debería ser lo más distribuido posible para tratar de minimizar los nodos críticos. Un nodo crítico de un sistema es aquel cuyo fallo puede poner en peligro el correcto funcionamiento del mismo.

En el caso de un sistema de VER, depende de cada implementación, pero ejemplos de nodo crítico serían: un único servidor donde se almacena todo el recuento electoral y/o todos los datos de votantes, una única autoridad verificadora o incluso el uso de IPs estáticas o rangos fijos de direcciones.

Todo ello complica la programación de las soluciones propuestas y aumenta los requerimientos y la complejidad de los sistemas de VER, como se explica en el apartado 5.2 y siguientes.

Existe multitud de bibliografía interesante sobre la clasificación de los ataques en los sistemas de VER [11, 22, 23, 24]. En esta tesis, se dividen los ataques en: *backdoors*, ataques al cliente/votante del VER, ataques a la red, ataques por corrupción/confabulación entre partes y ataques de ingeniería social (incluída la coerción).

2.4.2.1 *Backdoors* o puertas traseras

Las puertas traseras son métodos que permiten evitar los controles habituales de autenticación en un sistema informático, algoritmo o criptosistema. Puede tratarse de una parte oculta dentro del código o un programa separado.

En criptografía, son muchos los expertos como el Dr. Matthew Green de la Johns Hopkins University que se han posicionado en contra de su introducción, puesto que suponen una brecha de seguridad que puede ser explotada por sus conocedores o por otros sujetos si se filtra su existencia.

Existen numerosos casos conocidos de problemas de seguridad derivados de *backdoors* tales como *Back Orifice*, los descubiertos en Plug-ins de *WordPress*, los gusanos *Sobig* y *Mydoom* o el de la familia *Galaxy* de *Samsung Android*, descubierto en enero de 2014.

En el caso concreto de los sistemas de VER, en el año 2015 salió a la luz un ataque denominado FREAK (*Factoring Attack on RSA-EXPORT Keys*) que explotaba las debilidades de una *backdoor* que había quedado latente en los protocolos SSL/TLS desde los años 90 que permitía rebajar la calidad del protocolo RSA al inferior *EXPORT-RSA*, el cuál sí es atacable con las capacidades de computación actuales en tiempos lo suficientemente cortos para que el usuario no se diese cuenta [26, 27].

El Dr. Halderman y su equipo calcularon que un 36,7% de los más de 14 millones de servidores proveedores de certificados seguros aceptaban *Export-RSA* y por tanto eran

vulnerables al ataque. Entre los sitios de internet vulnerables se encontraban la Casa Blanca, el FBI, IBM, Facebook, Symantec etc.

2.4.2.2 Ataques del lado del cliente/votante

Hay un dato que conviene tener presente antes de entrar en este tipo de ataques: Multitud de estudios de expertos y empresas de seguridad informática cifran el número de ordenadores personales infectados con virus/malware en el entorno del 30-40% del total [23].

Los ataques que incluyen la infección del cliente de software con virus/malware presentan una amplia casuística: los que se producen en la fase de autenticación, los que aprovechan vulnerabilidades del sistema de VER para modificar el voto sin que ni el votante ni las autoridades se den cuenta, o los que envían una copia del voto a una tercera parte (como prueba de que se ha votado de acuerdo a una coerción previa).

Otra variedad, quizás la más simple de todas, es un malware que deniega el servicio (DoS o *Denial of Service*) en el que simplemente no se cuenta el voto del votante, bien sea mostrando un mensaje de error o bien haciendo creer al usuario que su voto se ha recontado sin problemas cuando en realidad no ha sido así.

Por último, la proliferación de otros dispositivos móviles con conexión a internet como tablets y teléfonos móviles, así como su creciente papel en los experimentos de VER como sistemas complementarios de verificación han contribuido a aumentar la inseguridad de los procesos de votación remotos [264].

La razón es que las medidas de seguridad de los citados dispositivos son menores que las de los equipos convencionales, llegando al punto de no disponer ni siquiera de antivirus contra potenciales amenazas. La inmensa variedad de plataformas, versiones y fabricantes no hace sino contribuir a que se conviertan en una nueva vía para los atacantes.

En el punto correspondiente a la experiencia del VER en Estonia (3.2.1) se profundiza en la experiencia de un grupo de hackers quienes mostraron cómo atacar el sistema de voto a partir de un fallo de seguridad en un dispositivo móvil. Se da la paradoja que se había incluído el uso de un segundo canal (el móvil) precisamente para incrementarla.

El hecho de que al menos un 30-40% de los equipos personales conectados a internet estén infectados de malware no hace sino reafirmar la premisa de que queda todavía mucho camino por recorrer en la mejora de la seguridad de los sistemas de VER.

2.4.2.3 Ataques a la red. FREAK y Logjam

En el apartado anterior se explicaron los ataques en la parte del cliente del sistema de VER, pero existe otra parte fundamental de la infraestructura que puede ser víctima de ataques: la red y sus servidores.

En los sistemas de VER, existen una serie de protocolos cliente-servidor que son ejecutados en distintas capas (TCP/IP, DHCP, NTP, DNS etc.) amén de otros usados en comunicaciones inalámbricas y móviles. Todos ellos son potenciales objetivos de ataques durante unas elecciones, como de hecho ya ha sucedido [36, 236, 287, 339, 464].

En sistemas E2Ev, se previene al menos parcialmente la introducción de votos fraudulentos o su modificación sin detección, pero nada previene contra un ataque que simplemente deje de enviar una serie de votos, permitiendo que se "pierdan".

Aunque se llegase a detectar que una serie de votos no han sido recepcionados, no existe ningún protocolo que evite este tipo de ataques, poniendo pues en serio peligro unas potenciales elecciones a gran escala.

Un tipo de ataque bien documentado a un servidor o una serie de servidores es el llamado DDoS o *Distributed Denial of Service*.

En ellos, el atacante inunda el servidor/servidores objetivo con una cantidad tan grande de tráfico que éste o se colapsa, o se ralentiza tanto que no puede realizar su función con una velocidad mínimamente aceptable.

Una estructura gráfica de un ataque DDoS se aprecia en la siguiente figura:

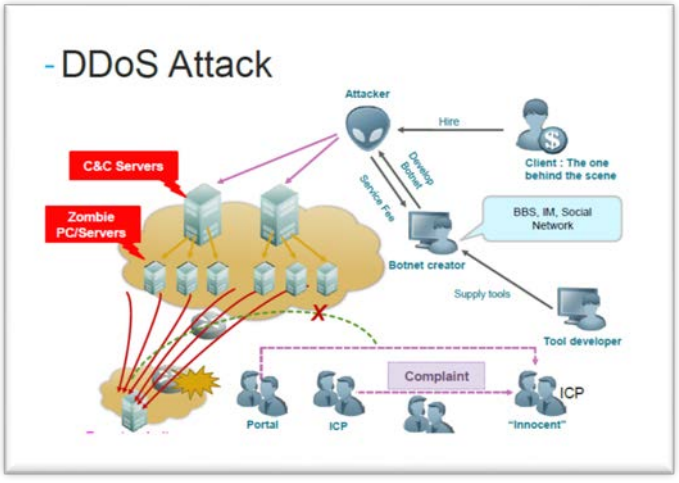


Figura 18: Ataque DDoS.
Fuente: NSFocus Information Technology. RSA Conference Asia/Pacific 2013.

De acuerdo a [23], hay al menos 4 ataques DDoS que se han producido en elecciones reales: las primarias del Partido Democrático de Arizona en 2000, en las elecciones del Nuevo Partido Democrático de Ontario en 2003, las elecciones del Pueblo en Hong Kong en 2012 y las elecciones del Nuevo Partido Democrático en Canadá en 2012. En los Estados Unidos, ha habido un claro aumento en el tráfico de Internet desde China, Irán y Rusia en fechas electorales, como se detalla en el punto 3.2.4.

El problema de los ataques DDoS es que son muy fáciles de organizar, existiendo kits e incluso particulares/organizaciones en la *dark web* ofreciendo todo lo necesario para perpetrar dichas acciones a precios asequibles.

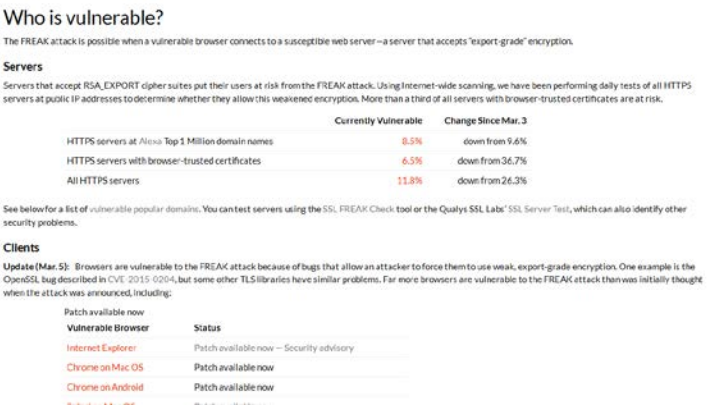
Como se ha explicado con anterioridad, actualmente no existe una protección procedimental contra ataques de este tipo y por tanto, pese a que existen herramientas y soluciones a nivel de arquitectura de software para tratar de minimizar los efectos (*Cloudflare* [436] o *Fail2Ban* [437] entre otros), el riesgo de un ataque DDoS a gran escala está siempre presente en elecciones con VER, especialmente en las que hacen uso de mix-nets.

Ataques FREAK y Logjam

En el año 2015 aparecieron dos nuevos ataques que explotaban deficiencias muy notables en dos de los protocolos más comunes de la red, comprometiendo seriamente la seguridad de la misma. Se denominaron FREAK [26] (*Factoring RSA Export Keys*) y Logjam [32].

2.4.2.3a Ataque FREAK

El 3 de marzo de 2015 salió a la luz un informe llevado a cabo por INRIA, Microsoft Research, IMDEA et al. [26] en el que presentaban un fallo crítico en los protocolos TLS/SSL que afectaban a los navegadores más importantes: *Internet Explorer*, *Google Chrome, Safari, Opera, Android Browser* y *Blackberry Browser*.



Who is vulnerable?

The FREAK attack is possible when a vulnerable browser connects to a susceptible web server—a server that accepts "export-grade" encryption.

Servers

Servers that accept RSA_EXPORT cipher suites put their users at risk from the FREAK attack. Using Internet-wide scanning, we have been performing daily tests of all HTTPS servers at public IP addresses to determine whether they allow this weakened encryption. More than a third of all servers with browser-trusted certificates are at risk.

	Currently Vulnerable	Change Since Mar. 3
HTTPS servers at Alexa Top 1 Million domain names	8.5%	down from 9.6%
HTTPS servers with browser-trusted certificates	6.5%	down from 36.7%
All HTTPS servers	11.8%	down from 26.3%

See below for a list of vulnerable popular domains. You can test servers using the SSL FREAK Check tool or the Qualys SSL Labs' SSL Server Test, which can also identify other security problems.

Clients

Update (Mar. 5): Browsers are vulnerable to the FREAK attack because of bugs that allow an attacker to force them to use weak, export-grade encryption. One example is the OpenSSL bug described in CVE-2015-0204, but some other TLS libraries have similar problems. Far more browsers are vulnerable to the FREAK attack than was initially thought when the attack was announced, including:

Patch available now

Vulnerable Browser	Status
Internet Explorer	Patch available now – Security advisory
Chrome on Mac OS	Patch available now
Chrome on Android	Patch available now

Figura 19: Ataque FREAK. Fuente: Elaboración propia a partir de [26]

De acuerdo a la vulnerabilidad descubierta, un atacante podía interceptar conexiones HTTPS entre clientes y servidores vulnerables, obligándoles a utilizar protocolos de encriptación debilitados. En consecuencia, el atacante puede romper el protocolo criptográfico, accediendo o modificando información privada de la víctima.

En este ataque en concreto, se explota una vulnerabilidad del tipo *backdoor*; introducida originariamente por el gobierno de los Estados Unidos:

Hasta el año 1992, el gobierno norteamericano consideraba la criptografía legalmente como "equipamiento militar auxiliar", aplicando estrictas restricciones a su exportación.

Cuando a principios de los años 90 *Netscape Corporation* desarrolló el protocolo SSL, tuvo que crear dos versiones; una "fuerte" para los Estados Unidos con claves RSA de criptografía asimétrica de 1024 bits y más, así como otra más "débil", que sería la que tendría licencia para ser exportada y cuyas claves RSA no excederían los 512 bits.

Dicho tamaño pretendía mantener un equilibrio entre un nivel aceptable de seguridad, a la vez que se trataba de un tamaño hackeable por una agencia con los recursos de la NSA americana. En otras palabras, el gobierno de los Estados Unidos quería reservarse la posibilidad de acceder a comunicaciones encriptadas extranjeras que considerase relevantes gracias a su ingente capacidad computacional, dejando al resto de agencias de inteligencia mundiales con menores recursos sin esa opción.

Desde entonces, las limitaciones estadounidenses a la exportación de criptografía se han suavizado (si bien no han desaparecido completamente) y con el tiempo se desarrollaron protocolos para negociar siempre con la clave más fuerte disponible.

Ello nos llevaría a pensar que ya no deberían darse conexiones cliente-servidor con claves RSA débiles de 512 bits.

Es en este punto donde comienza el trabajo de investigación de [26] con el desarrollo de una herramienta de análisis con la que descubrieron un *bug* que afecta a multitud de clientes TLS y que les lleva a aceptar claves RSA exportables de 512 bits incluso cuando el cliente no las pide.

Ello implica que los ataques "*man in the middle*" son plausibles en estos protocolos, reduciendo la clave a *"export"* aunque no lo haya solicitado el cliente sino el *"man in the middle"*.

El equipo de investigadores descubrió también que este *bug* en mucho más frecuente de lo que cabría pesar, afectando a un 36.7% de los más de 14 millones de servidores emisores de certificados de confianza de navegador. Y por si esto fuera poco, se comprobó que entre los sitios vulnerables se encontraban los de la misma NSA, la Casa Blanca o el FBI.

No obstante, como se ha indicado, una clave RSA de 512 bits sigue siendo relativamente difícil de romper en un tiempo reducido, por lo que se podría pensar que, pese al *bug*, la mayoría de las comunicaciones seguirían siendo seguras.

En la práctica, existe un último factor que contribuye a comprometer la seguridad: la generación de claves RSA por parte de un servidor es un proceso que consume muchos recursos, por lo que muchos de ellos únicamente generan una clave y la siguen utilizando hasta que se reinicia el servidor, lo cuál puede suceder meses después.

Por ello un atacante únicamente tendría que factorizar la clave de 512 bits una vez y a partir de ahí usarla con todas las conexiones de ese servidor.

En cuanto al coste real, ¿cuánto cuesta romper una clave RSA de 512 bits? La respuesta es preocupante: únicamente 104 USD en servicios de *cloud computing* y 7.5 horas. A partir de ese momento un atacante, tendría la clave y accedería a todas las comunicaciones en tiempo real y en texto sin encriptar, pudiendo realizar cuantos ataques deseara hasta que el servidor se reiniciase.

Desde la publicación del ataque FREAK se han implementado parches para los principales navegadores y se mantiene un listado actualizado de sitios vulnerables, así como una herramienta para comprobar el equipo del usuario [26].

Además, En [47, 48], el profesor Halderman y la Dra. Teague demostraron cómo el sistema desarrollado para las elecciones de 2015 en New South Wales era vulnerable a FREAK. Para profundizar más en este ataque, referirse a [26, 27, 34].

2.4.2.3b Ataque Logjam

Sólo dos meses después de desvelarse el ataque FREAK, en mayo de 2015 un grupo internacional de expertos en ciberseguridad encontró otro fallo crítico en los protocolos TLS/SSL derivado de errores en el protocolo de intercambio *Diffie-Hellman* [27, 33].

La forma más común de atacar una implementación de DH consiste en capturar el valor de g^a y calcular a [25]. Es el conocido como el problema del logaritmo discreto, el cual se considera matemáticamente irresoluble en tiempo polinómico siempre y cuando se utilicen primos de una longitud suficiente (en la actualidad 2048 bits).

En el caso del ataque Logjam, se explota una debilidad en el protocolo TLS que permite a un atacante *"man-in-the-middle"* rebajar la calidad criptográfica de las conexiones mediante el uso de Diffie-Hellman Exportable (DHE), con claves de un máximo de 512 bits de longitud y por tanto hackeables incluso con medios modestos.

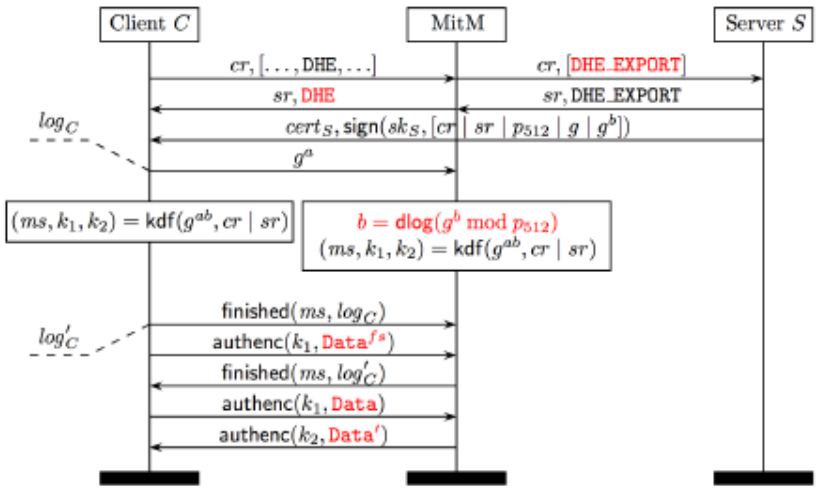


Figura 20: Ataque Logjam [33]

La clave está en la manera en la que el protocolo TLS compone DH y DHE. Cuando un servidor detecta un DHE (de 512 bits y no suficientemente robusto) en un *handshake* con el cliente, produce un mensaje *ServerKeyExchange* firmado el cual contiene un p_{512} de 512 bits. El problema reside en que la estructura del mensaje es la misma en los dos casos y no incluye información sobre qué *ciphersuite* ha escogido el servidor.

Por ello, aunque el cliente ofrezca un DH; el atacante *man in the middle* puede cambiarlo por un DHE vulnerable de 512 bits y enviarlo al servidor sin que éste note el cambio.

En ese punto, el atacante reescribe el *ServerHello*, cambiando la *ciphersuite* DHE vulnerable por otra no exportable y envia el mensaje *ServerKeyExchage* desde el servidor al cliente tal cual (el cliente no comprueba nada y en realidad está aceptando una *ciphersuite* vulnerable).

El cliente interpretará la tripla de nivel de exportación 512 bits (p_{512}, g, g^b) como válida y procederá con el *handshake*. En este momento, el cliente y el servidor tienen diferentes transcripciones de *handshake* pero un atacante que pueda computar b en tiempo real puede obtener las claves y suplantar al servidor.

Para que el ataque sea exitoso, habría que poder computar logaritmos discretos en tiempo casi-real y por otra parte retrasar la finalización del *handshake* del protocolo TLS hasta que la computación del logaritmo discreto de 512 bits hubiese terminado.

Podría parecer que el ataque no debería tener éxito puesto que la computación de un único logaritmo discreto de 512 bits lleva varios años de computación en tiempo de operación de procesador o alrededor de una semana con unos cuantos miles de procesadores.

En realidad, la computación de un logaritmo discreto se puede dividir en dos fases claramente diferenciadas, una de las cuales (la más intensiva en computación) puede ser pre-computada para un determinado primo *p*.

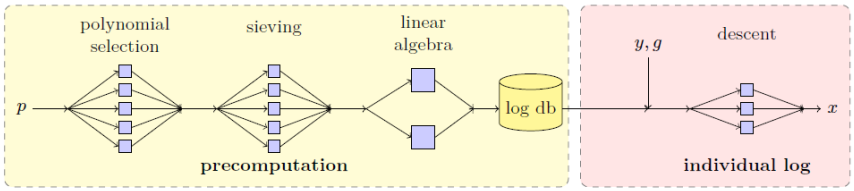


Figura 21. Ataque Logjam, precomputación [23]

En la figura se puede apreciar como para un determinado primo *p*, hay una gran parte del algoritmo susceptible de pre-computación genérica, dejando la última parte para computar el *log* individual.

Si para cada sesión en el protocolo TLS se generase un nuevo primo *p*, entonces el ataque sería prácticamente imposible de llevar a cabo.

El problema de nuevo reside en que la generación de primos muy grandes es una operación muy costosa en términos de computación, reutilizándose los primos generados.

En el caso concreto de los dominios incluidos en el *Top 1 Million HTTPS Domains* de Alexa que aceptan DHE (un 8.4% del total), un 92.3% usan uno de los dos primos más habituales. Ello quiere decir que pre-computando esos dos primos conocidos, se podría atacar en tiempo real ese porcentaje de las conexiones cliente servidor bajo el protocolo TLS en dominios HTTPS que aceptan protocolo de intercambio DHE.

En cuanto al coste de la pre-computación, en el caso de claves de 512 bits, se utilizó un cluster de 36 nodos y dos Intel Xeon E5-2650 de 8 núcleos por nodo conectados a través de *Infiniband FDR*. Con esos medios, cada primo les llevó aproximadamente una semana.

Una vez terminada la pre-computación de un determinado primo, se pudieron romper 3500 intercambios de claves con una mediana de tiempo de 70 segundos. El equipo utilizado incluyó dos CPUs Intel Xeon E5-2699 de 18 núcleos y 128 GB de RAM.

Se podría pensar que 70 segundos es un tiempo demasiado prolongado para un *handshake* bajo TLS. En realidad y según los autores del ataque, se podría optimizar el código de la parte de los *logs* individuales hasta rebajar el tiempo por debajo del minuto. Paralelamente, TLS permite "trucos" para mantener una sesión activa como puede el envío de *warning alerts* que resetean el timer del protocolo de *handshake*.

La investigación fue un paso más allá y realizó estudios sobre *Diffie-Hellman* con claves de longitud 768 y 1024 bits, basándose en estudios previos de coste computacional [38, 39, 42] de W. Geiselmann, T Kleinjung et al.

Para DH-768, calcularon que la parte de pre-computación llevaría un total de 36.000 años de coste de computación para un primo determinado mientras que la fase de *log* individual se realizaría en un día de coste computacional. Los dos valores están dentro de lo que los autores denominan "capacidad académica" o los medios de los que disponen numerosos centros de investigación/universidades en el mundo.

Por lo que respecta a *Diffie-Hellman* con claves de 1024 bits, y basándose en bibliografía previa [39, 43], se ha hecho una aproximación de pre-computación de 45 millones de años de coste de computación. La parte de *log* individual por su parte se podría completar aproximadamente en 30 días de coste de computación.

Los autores del ataque calcularon que para poder concluir los 45 millones de años de coste de computación en un año natural y los 30 días de log individual en tiempo real, se requeriría una inversión en equipamiento en la magnitud de los cientos de millones de USD.

Dichos guarismos son perfectamente asumibles teniendo en cuenta el presupuesto anual de la de la *National Science Foundation* (7.000 millones de USD anuales) o del *U.S. Consolidated Cryptologic Program* en 2012 (10.500 millones de USD). Únicamente en el año 2013, el presupuesto de la NSA en *"Management, Facilities and Support"* fue de 5.200 millones de USD, el de *"Data Collection Expenses"* de 2.500 millones y el *"Equipamiento en Sistemas de IT"* de 1.600 millones de USD [44].

El hecho de que su coste esté dentro de las posibilidades de una agencia estatal con un gran presupuesto llevó a los autores a preguntarse si efectivamente la NSA americana estaba rompiendo las comunicaciones DH de 1024 bits.

No existen certezas que prueben que la NSA esté realizando dicho tipo de actividades de pre-computación de los primos más habituales del DH-1024, si bien hay alguna evidencia de que la NSA posee dicha capacidad de desencriptación de acuerdo a una serie de documentos de Edward Snowden publicados por *Der Spiegel* [41] en diciembre de 2014.

Por último, en [33] se enumeran una serie de recomendaciones para tratar de minimizar el impacto del ataque *logjam* entre las que se destacan:

- Realizar una transición del protocolo *Diffie-Hellman* al intercambio de claves *Diffie-Hellman* de curva elíptica (ECDH) [45, 46]
- Incrementar el tamaño de los primos a usar hasta un mínimo de 2048 bits a la vez que se desactiva el DHE. Pre-computar un primo de 2048 bits se calcula que requiere 10^9 veces más computación que uno de 1024 bits por lo que en la actualidad sería suficientemente seguro
- Evitar los primos de 1024 bits con una mayor probabilidad de haber sido o estar siendo pre-computados

A modo de conclusión de este apartado 2.4.3 sobre ataques a la red, es relevante destacar que los citados FREAK y *logjam* son ejemplos de agujeros en la seguridad masivos en los protocolos más utilizados en la actualidad en las comunicaciones por internet y que afectaban a un porcentaje de los servidores y sitios web muy importante (36.7% y 8.4%). Conviene tenerlo muy presente a la hora de introducir sistemas de VER con garantías.

2.4.2.4 Ataques por corrupción/confabulación entre partes

Los apartados explicados hasta la fecha incluyen un atacante ajeno a los actores incluidos en unas elecciones. No obstante, puede darse también que sean una o varias de las partes implicadas en unas elecciones las que decidan atacar el sistema.

Puede tratarse de un votante que decide voluntariamente vender su voto a una parte interesada, o de los responsables de velar por la seguridad del recuento final, o de transportar físicamente los datos de un servidor a otro (realizar la recepción de datos en un servidor y el recuento en otro equipo distinto desprovisto de cualquier tipo de conexión es una práctica común para mejorar la seguridad de los sistemas de VER) o incluso de los propietarios de las claves privadas finales necesarias para realizar el recuento.

El hecho de delegar una serie de tareas críticas como puede ser el acceso a los votos del electorado (aunque sean encriptados), el recuento o el transporte de los mismos en un solo individuo (en el peor de los casos) o en un grupo de ellos (cuando se implementan filosofías distribuidas) constituye una debilidad estructural de los sistemas de VER.

Un adversario de los arriba mencionados tendría la capacidad de instalar un malware o una versión con un modelo de recuento distinto, o simplemente descargar una distribución de datos fraudulenta haciendo extremadamente difícil descubrir el ataque.

Pese a que lo arriba detallado pudiera considerarse un exceso de celo o bien unos estándares de seguridad excesivamente exigentes, la realidad es que en casos como en Estonia (el país con la tradición más prolongada de uso de VER en elecciones VAP) los análisis independientes revelaron que la tipología de ataques por corrupción/confabulación entre partes hubiesen podido darse [236].

Todos los sistemas y primitivas criptográficas parten de una serie de precondiciones más o menos optimistas como se verá a lo largo de la tesis. También en el sistema *open source* Helios [1] se ha demostrado que la confabulación entre un tablón deshonesto y el verificador de credenciales podría comprometer el resultado de unas elecciones [371, 377].

Por contraposición, en unas elecciones convencionales existe una cantidad muy elevada de mesas electorales, cada una de ellas realizando un recuento público en presencia de multitud de partes con intereses contrapuestos, por lo que el riesgo de comprometer las elecciones "desde dentro" es mucho menor y en el remoto caso de que sucediese, el problema estaría mucho más aislado por la estructura de mesas electorales independientes.

2.4.2.5 Ataques de ingeniería social

Esta tipología de ataque tiene la dificultad añadida de no poder ser contrarrestada con herramientas hardware o software convencionales.

En este contexto, ingeniería social se refiere a la manipulación psicológica de uno o varios usuarios para que realicen una serie de acciones o revelen una información privada.

Según el prestigioso *hacker* luego convertido en consultor de seguridad Kevin Mitnick, es mucho más fácil convencer/embaucar a una persona para que desvele su *password* que crackear el sistema con el mismo fin [28].

Según el mismo autor, las tácticas de ingeniería social triunfan porque las personas actuan de acuerdo a los siguientes principios:

- Todos quieren ayudar
- El primer impulso es de confianza hacia otro
- No gusta decir que no
- A las personas les gusta que las alaben

Un ataque de ingeniería social que tenga presentes los citados cuatro principios tiene muchas posibilidades de ser exitoso al menos en parte.

Su tipología es variada, incluyendo ataques de pretexto/excusa, *phishing*, señuelo, *quid pro quo* (aquí se incluyen los falsos servicios técnicos), redes sociales…

Es difícil establecer una única cifra de éxito en los ataques con ingeniería social, puesto que dependen del sector, de la segmentación del grupo de potenciales víctimas y de la técnica usada, pero normalemente se acepta un abanico entre 0.001% y un 30%.

Extrapolando a las elecciones españolas, en 2011 los ciudadanos con derecho a voto ascendieron a casi 36 millones, de los que efectivamente votaron alrededor de 24,5 millones. Con el abanico de tasa de éxito presentado, el número de potenciales víctimas que caerían en fraudes de ingeniería social sería de al menos 24.500 votantes.

Incluso si se tomase el caso más realista de una introducción del VER inicial de un 10% a modo de prueba y una tasa de participación de un 70% (similar a la del voto tradicional), se estaría hablando potencialmente de más de 1700 personas.

De hecho, se podría refinar fácilmente el ataque seleccionando determinadas circunscripciones sin un claro sesgo electoral hacia un determinado partido para personalizar el ataque y conseguir el objetivo manipulando únicamente unos pocos votos. Otra posibilidad de aumentar la eficiencia sería dirigir el ataque a segmentos de población con una menor cultura en TIC y por tanto más vulnerables.

El caso arriba descrito por supuesto no tiene porqué darse, pero debería servir de aviso a la hora de implementar una estrategia de seguridad y comunicación eficiente y robusta.

Una vez expuestas las principales tipologías de ataques, sólo nos queda reiterar la opinión de que pese a los importantes avances que se están produciendo en las nuevas soluciones de VER, persisten todavía una serie de vulnerabilidades que podrían comprometer los resultados de unas elecciones vinculantes en el ámbito político a gran escala.

Hasta que se lleguen a superar dichas vulnerabilidades, es necesario continuar perseverando en la investigación y desarrollo de nuevas soluciones, en la mejora de las ya existentes y mantener o incluso incrementar pruebas piloto en distintos ámbitos y circunstancias, exigiendo el mayor grado posible de transparencia para facilitar el análisis exhaustivo por parte de expertos independientes así como de la comunidad científica.

Todo ello, unido a una introducción paulatina del VER, constituye el camino más razonable para avanzar en la materia sin renunciar a proteger el carácter inviolable del voto, manteniendo la integridad y privacidad de nuestros procesos democráticos.

Parte II

Análisis práctico y definición formal de la metodología

Divide et impera
Divide y domina
-Filipo II de Macedonia

Capítulo 3

ANTECEDENTES, EXPERENCIAS PREVIAS Y ESTADO DEL ARTE

十人十色

Diez personas, diez colores. Para gustos los colores

-Proverbio japonés

3.1 Breve historia y antecedentes de los procesos democráticos y del Voto Electrónico Remoto

Establecer el origen concreto de los primeros procesos electorales vinculantes en la Historia de la Humanidad no es tarea sencilla, puesto que se entremezclan realidad y leyenda.

De acuerdo al reconocido historiador y profesor de Harvard Thorkild Jacobsen, el rey sumerio Gilgamesh fue el primero en delegar parte de su poder en un consejo de ancianos en el siglo XXVII A.C.; es lo que Jacobsen denomina "democracia primitiva".

Democracia

No obstante, se reconoce universalmente como cuna de la democracia moderna a la Antigua Grecia del siglo VI A.C. con Solón primero (594 A.C.) y Clístenes (508 A.C.) después. Ellos fueron quienes introdujeron reformas para terminar con la tiranía como sistema de gobierno, adoptando la democracia en su lugar.

Por supuesto no se trataba de democracias universales ya que únicamente podían votar y ser elegidos ciudadanos no extranjeros y libres que poseyeran más de una cierta cantidad de bienes. No obstante, el hecho de pasar de un sistema basado en el linaje a otro fundamentado en el individuo y sus propiedades fue un innegable avance.

Voto secreto

Originariamente, las votaciones eran asamblearias y a mano alzada o cantando el voto; por lo que no existía voto secreto salvo en el caso del *"voto de ostracismo":*

Fue Clístenes quien introdujo el uso del ostracismo como salvaguarda a la democracia por primera vez en el 487 A.C. . Todos los años, en la época de la Asamblea, a la que tenían derecho más de 42.000 ciudadanos atenienses con más de 2 años de servicio militar, se

sometía a votación si había voluntad de un voto de ostracismo (enviar al exilio a un ciudadano/mandatario que supusiese "un peligro para la democracia").

Si más de 6.000 ciudadanos votaban afirmativamente, se producía la votación secreta dos meses después. En ella, los ciudadanos con derecho a voto que quisiesen ejercer su derecho, escribían el nombre de la persona que querían exiliar por 10 años en un trozo de cerámica o una concha de ostra (en griego *ostrakon*, de ahí la denominación de ostracismo). No existía lista previa de "candidatos" y se votaba una única vez. De nuevo más de 6000 ostrakon con el mismo nombre suponían el destierro del "elegido".

Avanzando hasta el Medievo, en 1188 bajo el reinado de Alfonso IX de León tuvieron lugar las Cortes de León, reconocidas por la UNESCO como el "más antiguo sistema parlamentario europeo" y que sirvieron como base a la Magna Carta inglesa de 1215 e incluso fueron estudiadas por los Padres Fundadores de los Estados Unidos de América como modelo para la Constitución Americana.

Hay que continuar hasta año 1795 para encontrar la primera referencia a la necesidad de que el voto sea secreto. Es en Francia, más en concreto en el artículo 31 de la Constitución de ese año, donde por primera vez se establece que "Todas las elecciones serán celebradas por voto secreto". Dicha condición se mantuvo en la posterior Constitución de 1848.

La siguiente región en celebrar unas elecciones con voto secreto fue Tasmania en 1856, seguida de Nueva Gales del Sur en 1858.

En cuanto a los Estados Unidos, el "voto australiano" como fue denominado, se introdujo en 1888 y el primer presidente en ser elegido mediante voto secreto fue Grover Cleveland en 1892.

No fue hasta 1925 cuando se introdujo la prohibición expresa de pagar a cambio del voto y hasta el año 1950 Carolina del Sur no introdujo papeletas emitidas por el Estado.

Voto remoto

En cuanto al voto remoto, una de las primeras experiencias documentadas se remonta al año 1864 en los Estados Unidos. En plena Guerra Civil, con numerosos soldados desplazados al frente, se adoptó la posibilidad de voto a los soldados ausentes de su jurisdicción.

Cabe aclarar que la motivación para adoptar el voto remoto no era otra que facilitar en lo posible la reelección del presidente Lincoln. Fue la labor del lobby republicano la que propició su introducción y nunca se consideró como una opción permanente sino como una medida excepcional para un período bélico.

Sufragio femenino y sufragio universal

Por lo que respecta al sufragio femenino, Nueva Zelanda fue el primer país en permitirlo en el año 1893. En Europa, los pioneros fueron el Gran Ducado de Finlandia en 1907 y Noruega en 1913. España por su parte lo adoptó en 1931, antes que Francia (1944), Japón (1945), Italia (1946), Andorra (1970) o Suiza (1991).

Por último, el sufragio universal es un logro muy reciente en numerosos países, en contra de la creencia más extendida. De nuevo Nueva Zelanda fue la nación pionera en el 1893, seguida de Finlandia y Noruega. España lo introdujo también en 1931.

Cabe destacar que en los Estados Unidos seguían existiendo restricciones en el voto a los ciudadanos afroamericanos hasta 1964, cuando se aprobó la Vigesimocuarta Enmienda que prohibía condicionar el voto en las elecciones federales. Si bien en 1965 se aprobó la Ley de Derecho de Voto, la citada Vigesimocuarta Enmienda no ha sido oficialmente ratificada en Arizona, Arkansas, Georgia, Luisiana, Mississippi, Oklahoma, Carolina del Sur ni Wyoming.

De igual manera, Australia mantuvo restricciones de voto a los aborígenes de origen hasta 1962 y en Inglaterra no se instauró el sufragio universal hasta 1968.

Ello nos da una buena perspectiva histórica de lo tardío del acceso al derecho al voto y de lo crítico del mantenimiento de sus atributos para garantizar el correcto funcionamiento de los sistemas democráticos.

En la actualidad existen 125 democracias [6], de las cuales 89 se pueden considerar democracias libres. En total suponen 2.900 millones de personas (un 40% de la población mundial).

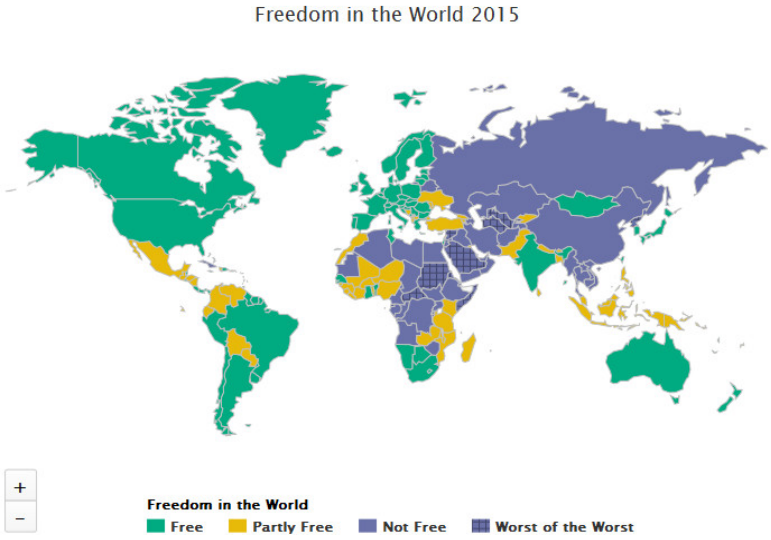


Figura 22: Democracias libres en el mundo [6]

Voto electrónico

Las primeras experiencias datan del año 1964 en los Estados Unidos con la introducción de la tecnología de tarjetas perforadas en dos condados de Georgia [7]. El 17 de agosto de 1965, se concede la patente americana *US Patent 3,201,038* al Dr. J. P. Harris del sistema de voto de tarjeta perforada *Votomatic*.

Otro hito en el voto electrónico fue el desarrollo y obtención de patente de la primera DRE (*Direct Recording Electronic*), conocida como *Video Voter*.

La patente fue concedida el 19 de febrero de 1974 a McKay, Ziebold y Kirby (*US Patent 3,793,505*). Ésta fue la base sobre la que McKay, Smith y Deutsch obtuvieron a su vez la *US Patent 4,025,757*, versión comercial de *Video Voter,* con terminales controlados por un centro de datos. Su uso comenzó en el mismo año en los condados de Streamwood y Woodstock en Illinois, a los que prosiguieron otros en el mismo estado en las elecciones de los años 1976 y 1980.

En 1975 Roy Saltman elabora el primer informe para evaluar las "Tecnologías de Voto Computerizado" [8], iniciando el programa gubernamental de *Voting Systems Standards*.

En 1988, Roy Saltman alerta por primera vez en su informe *"Accuracy, Integrity and Security in Computerized Vote-Tallying"* de los riesgos del voto con tarjetas perforadas, sugiriendo su inmediata cancelación como método de votación.

Pese a ello, siguieron en uso hasta que en las elecciones presidenciales del año 2000 hubo una serie de irregularidades en el recuento de votos de tarjetas perforadas en Florida [9, 10] que propiciaron su prohibición en el *Help America Vote Act* de 2002.

Voto Electrónico Remoto

Dentro del Voto Electrónico Remoto, existen otros dos sistemas aparte del que se realiza a través de internet: el VER por fax y el VER por correo electrónico.

En ambos casos, su uso se limita básicamente a los Estados Unidos y a ciudadanos residentes en el extranjero o a militares desplazados. Su utilización está permitida en 24 estados en el caso del fax y en 7 en el caso del voto por correo electrónico.

Hay práctica unanimidad entre los expertos en seguridad informática de la insuficiente seguridad y nula privacidad que ambos brindan y de la necesidad de su eliminación como opción de voto [12].

Por lo que respecta al VER a través de Internet, éste constituye el principal objeto de estudio de la presente Tesis. Sus inicios se remontan a 1990 cuando la *Federal Election Commission* (FEC) americana publicó su primera edición de VSS (*Voting Systems Standards*).

En agosto del año 1996 se produjo la primera elección gubernamental a través de internet, cuando el *Reform Party* eligió a su candidato a la Presidencia de los EEUU ofreciendo entre otras, la opción de voto por internet a los miembros del partido que acudiesen al congreso.

Dos años después en 1998, Alemania comenzó con un período de prueba del voto remoto a través de internet que se prolongó hasta las elecciones del año 2005 al *Bundestag*, cuando una denuncia de dos ciudadanos alemanes al Tribunal Constitucional inició un procedimiento que culminó en 2008 con la prohibición del VER [14].

Con posterioridad, en el año 2000, los estados de Alaska, Arizona, California, Florida, Utah, Carolina del Sur, Tejas y Maryland iniciaron experiencias piloto de VER. La más destacada fue la de Arizona, en la que se emitieron un total de 39,942 votos para elegir al candidato demócrata en las primarias del citado estado americano. [15]

En 2002 les siguió el Reino Unido y en 2005 tuvieron lugar en Estonia las primeras elecciones vinculantes en el ámbito político que ofrecieron el VER.

En el siguiente capítulo 3.2 y sub-apartados se detallan y analizan las experiencias internacionales más relevantes de VER en elecciones vinculantes en el ámbito político.

3.2 Experiencias previas de Voto Electrónico Remoto en elecciones públicas vinculantes en el ámbito político

En el presente apartado, se va a realizar un exhaustivo repaso a las principales experiencias reales de Voto Electrónico Remoto en elecciones públicas vinculantes en el ámbito político en los países donde se ha apostado más firmemente por su implantación.

Es importante destacar la singularidad de cada uno de los casos estudiados, puesto que la legislación electoral y la idiosincrasia de cada país varían muy notablemente. Ello, unido al hetereogéneo conjunto de requisitos de cada sistema de VER en función del país, hace que se deba abordar cada caso como único y no extrapolable directamente.

La experiencia acumulada de millones de votos emitidos en diversas plataformas, así como las incidencias y ataques ocurridos, conforman la base sobre la que extraer los criterios adicionales a añadir a la metodología de evaluación explicada en el capítulo 4.

3.2.1 Estonia

Las primeras experiencias piloto de VER vinculantes a nivel mundial se remontan a finales de los años noventa y primeros años del siglo XXI. Desde entonces, en un plazo inferior a 5 años, Estonia incorporó el Voto Electrónico Remoto a nivel nacional para sus elecciones locales del año 2005.

Dicha transición fue posible por la voluntad política de sus dirigentes que realizaron en el año 2002 los cambios legislativos necesarios para permitir la adopción de un sistema de tarjetas de identidad electrónicas o *ID Cards* como medio de articular en VER.

Desde 2005, Estonia se ha erigido como el país que más firmemente ha apostado por el VER, si bien su periplo no ha estado exento de incidencias [236]. Con todo, su apuesta por el VER es firme, a diferencia de Alemania, Holanda, Reino Unido o Noruega puesto que se ha seguido apoyando y fomentando su uso a pesar de las críticas.

Según las propias autoridades estonias, el aumento en la utilización del sistema de VER se debe a que más del 98% de los estonios en edad de votar dispone de una *ID Card* (que se utiliza junto con el software cliente de votación para que el votante se autentique) [248].

El sistema de VER de Estonia se ha utilizado en las siguientes 8 EVAP:

- Elecciones locales de octubre de 2005, octubre de 2009 y octubre de 2013.
- Elecciones al Parlamento de marzo de 2007, marzo de 2011 y marzo de 2015.
- Elecciones al Parlamento Europeo de junio de 2009 y mayo de 2014.

En el sistema de VER estonio, los votantes que dispongan de *ID Card* pueden votar por internet durante 7 días: desde el décimo día al cuarto día antes de las elecciones. A partir de ese momento, únicamente se puede votar de la manera tradicional.

En realidad, se habla de un único sistema de VER cuando son tres las posibilidades:

1) Con la *ID Card* y sus códigos PIN asociados, un ordenador con conexión a internet y un lector de *ID Cards*. Supone casi un 90% de los votos aunque su tendencia es ligeramente a la baja.

2) Con el *Digital ID*. Es otro tipo de documento que permite identificar a una persona en un entorno electrónico así como firmar documentos. El procedimiento de votación es análogo al anterior. En la práctica su porcentaje de uso no supera el 1,5% y se mantiene estable en ese guarismo.

3) Con una *Mobile ID SIM Card* (un tipo especial de tarjeta SIM que permite códigos PIN y certificados de seguridad), un ordenador con acceso a internet y un teléfono móvil. En cualquier caso, el voto se sigue realizando a través del ordenador por lo que no se puede hablar propiamente de *m-voting* o voto a través de dispositivo móvil.

   Actualmente un poco más del 10% de los votos electrónicos se realizan a través de este método, con una ligera tendencia al alza.

En cuanto a la penetración del sistema, el porcentaje de Voto Electrónico Remoto sobre el total de votos emitidos ha fluctuado entre un 1,9% en las elecciones locales de 2005 (primer uso) y un 31,3% en las elecciones al parlamento europeo de 2014, con un crecimiento sostenido casi sin excepción.

A continuación se presenta la tabla completa de las elecciones arriba comentadas:

	Local 2005	*Parliamentary 2007*	*European 2009*	*Local 2009*	*Parliamentary 2011*	*Local 2013*	*European 2014*	*Parliamentary 2015*
Eligible voters	1.059.292	897.243	909.628	1.094.317	913.346	1.086 935	902.873	899.793
Participating voters	502.504	555.463	399.181	662.813	580.264	630.050	329.766	577.910
Voter turnout	47.4%	61.9%	43.9%	60.6%	63.5%	58.0%	36.5%	64.2%
I-voters	9.317	30.275	58.669	104.413	140.846	133.808	103.151	176.491
I-votes counted	9.287	30.243	58.614	104.313	140.764	133.662	103.105	176.329
I-votes cancelled	30	32	55	100	82	146	46	162
Multiple I-votes	364	789	910	2.373	4.384	3.045	2.019	4.593
I-voters among participating voters	1.9%	5.5%	14.7%	15.8%	24.3%	21.2%	31.3%	30.5%

I-votes cast abroad among I-votes	n.a.	2% 51 states	3% 66 states	2.8% 82 states	3.9% 105 states	4.2% 105 states	4.69% 98 states	5.71% 116 states
I-voting period	3 days	3 days	7 days	7 days	7 days	7 days	7 days	7 days
I-voters using mobile-ID	n.a.	n.a.	n.a.	n.a.	2.690	11.753	11.609	22.084
I-voters using mobile-ID among I-voters	n.a.	n.a.	n.a.	n.a.	1.9%	8.6%	11,00%	12.2%
Share of I-votes verified by the voter	n.a.	n.a.	n.a.	n.a.	n.a.	3.4%	4,00%	4.3%

Tabla 3: Elecciones en Estonia con VER.
Fuente: Propia basado en Gobierno esloveno, Comité electoral nacional de Estonia.

Actualmente 3 de cada 10 votantes estonios ejercen su derecho a través del VER. Dicho porcentaje de utilización, unido a la cantidad total de votos manejados, hace de Estonia un caso práctico de utilización de sistemas de VER de una enorme importancia a la hora de estudiar las fortalezas, debilidades, mejoras posibles y ataques.

Un hito que contribuyó a mejorar la transparencia del sistema de VER estonio fue la publicación de buena parte del código fuente del mismo (si bien no su totalidad) en 2013.

En cuanto al sistema en sí, se pueden distinguir 2 períodos en el uso del VER en Estonia:

1) Primer período: De 2005 a 2011:

Se trata de un esquema muy sencillo que replicaba el modelo de voto postal de doble sobre [243].

La autoridad central de voto genera un par de claves RSA y hace pública la parte pública s_{pub}. El votante v se autentica para votar utilizando su *ID Card* y recibe la lista de candidatos. Realiza su elección c_v y la encripta con s_{pub} (para la encriptación se usa el protocolo *RSA-OAEP* y se genera un factor aleatorio r).

Con ello se tiene en voto encriptado o sobre interior que se define como:

$$b_{anon=Enc\ s_{pub}(c_v,r)}$$

Posteriormente, el sobre externo se construye firmando el voto encriptado con la *ID Card* del votante $b = Sig_v\ (b_{anon})$ y se envía b al servidor de votación.

Como protección contra la coerción, el sistema permite re-votar, teniéndose en cuenta el último voto emitido. Dicha política anti-coerción se ha demostrado vulnerable a varios tipos de ataques [236].

Una vez concluido el período habilitado para el VER, el votante tiene siempre la posibilidad de votar en papel, anulando el voto precedente.

El sistema activo hasta 2011 presentaba varias debilidades La más famosa y grave de ellas la hizo pública un estudiante, al descubrir que el sistema de VER no ofrecía un *feedback* fiable al votante sobre si su voto había sido recibido por el servidor o no.

En marzo 2011, el mismo estudiante pidió que se invalidaran los resultados tratando de demostrar de un modo práctico que la brecha de seguridad era real y podía ocurrir.

Finalmente, el Tribunal Supremo decidió no anular los resultados de las elecciones puesto que, aunque dicho ataque podría potencialmente haberse producido, no existió prueba alguna de que efectivamente hubiese tenido lugar.

Aún así, el hecho de hacer público un fallo de seguridad tan importante llevó a que se revisase el sistema de VER y se mejorase de cara a futuras elecciones, dándo lugar a una nueva versión que es la que se ha venido usando desde 2013 y que permite la verificación por parte del votante de que su voto ha sido contado correctamente.

Dicho avance ha hecho que surjan otra serie de dudas sobre el nuevo sistema de VER, derivadas del hecho de que suministrar al votante un "recibo de voto", arremete contra el segundo nivel de privacidad explicado en el apartado 2.2.3 y 2.3b, "ausencia de recibo" o *"receipt-freeness",* tal y como establece el Consejo de Europa en la recomendación 51 de su publicación sobre recomendaciones de estándards para e-voting [54]: *"A remote e-voting system shall not enable the voter to be in possession of a proof of the content of the vote cast"* .

2) Segundo período: 2013 – actualidad:

En el anterior esquema, la ausencia de capacidad de verificación por parte del votante sobre si su voto había sido recibido o no, llevó a una revisión del protocolo por parte de la misma compañía encargada de desarrollar la primera versión, Cybernetica AS.

Para aumentar la seguridad, decidieron que la verificación se realizase a través de otro canal. Tras sopesar la solución adoptada en el caso noruego (postal y SMS, referirse a 3.2.2 para más detalles), decidieron que dicha solución era demasiado costosa e implicaba una serie de problemas que no justificaban su elevado precio.

La decisión adoptada fue la de utilizar dispositivos móviles (smartphones y tablets) como canal alternativo de verificación.

Utilizando la notación del apartado anterior, una vez que el votante envía al servidor $b = Sig_v\ (b_{anon})$, se producen los siguientes pasos:

- El servidor manda al votante una referencia única y aleatoria de su voto o *vr*.
- El votante transfiere *r* y *vr* a su dispositivo móvil (canal alternativo para aumentar la seguridad) a través de un código QR.

- El dispositivo móvil se conecta con el servidor a través de una conexión HTTPS segura y envía *vr*.
- El dispositivo móvil del votante descarga el voto b_{anon} correspondiente junto con el listado L de todos los candidatos.
- El dispositivo móvil computa $Enc\ s_{pub}(c, r)$para todos los $c \in L$.

Si para algún c' se cumple que $Enc\ s_{pub}(c', r) = b_{anon}$, se muestra c' al usuario.

Si $c_v = c'$ el votante acepta que el voto ha sido contado como se ha enviado.

De una manera más gráfica:

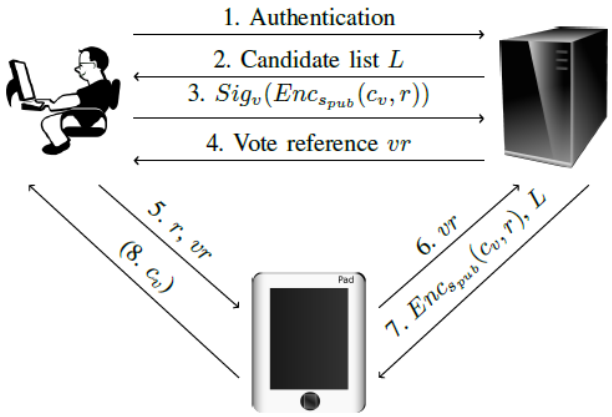


Figura 23: Sistema de VER de Estonia con verificabilidad del voto. Fuente: Cybernetica AS [243]

El dispositivo móvil únicamente interactúa con valores aleatorios y desconoce por tanto la identidad del usuario.

Para reforzar la seguridad, el votante dispone de 60 minutos para completar las acciones 4 a la 7 y únicamente puede descargarse b_{anon} 3 veces.

Infraestructura de servidores del sistema

En lo que respecta a la infraestructura de servidores del sistema, una parte de su código fuente se hace público 2 o 3 semanas antes de las elecciones. Existen 4 servidores:

1) Servidor direccionador de votos/*Vote Forwarding Server* (VFS/HES)

Es el único públicamente accesible a través de una conexión HTTPS por parte del software de cliente. Es el encargado de verificar su eligibilidad y realiza también la función de intermediario con el servidor repositorio de los votos (VSS/HTS), no accesible a través de internet.

2) Servidor repositorio de votos o *Vote Storage Server* (VSS o HTS en estonio)

Almacena los votos firmados y encriptados durante el período de votación on-line. Cuando recibe un voto del VFS, confirma que está formado correctamente y verifica

la firma digital del votante utilizando un servidor OCSP (*On-line Certificate Status Protocol*) externo.

3) Servidor de registro o *Log Server*

Realiza funciones de registro y monitorización, recolectando sucesos y estadísticas de los dos servidores previos (VFS y VSS). Es accesible únicamente por parte de las autoridades en modo remoto. Su código fuente no ha sido hecho público.

4) Servidor de recuento de votos o *Vote Counting Server* (VCS)

Se utiliza únicamente en la parte final de la elección. Nunca está conectado a ninguna red. Las autoridades electorales utilizan un DVD para copiar los votos encriptados desde del VSS y lo introducen en el VCS.

El servidor de recuento dispone de un módulo hardware de seguridad denominado HSM que es quien contiene la clave privada de la elección. Posteriormente, se usa éste para desencriptar los votos, contarlos y generar el resultado final de la elección.

Para mayor claridad, la siguiente figura resume el sistema de VER de Estonia:

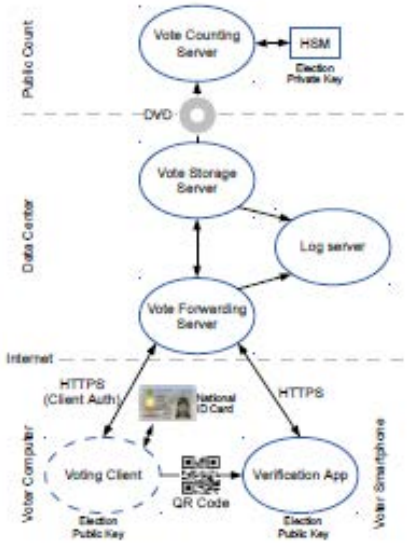


Figura 24: Sistema de VER de Estonia. Componentes y flujos principales de información. [236]

Vulnerabilidades y ataques al sistema de VER de Estonia

Al sistema de VER de Estonia se le achacan unas premisas muy optimistas de honestidad de las partes implicadas y de robustez a ataques y malware por parte del PC del usuario y de su dispositivo móvil.

Por citar dos ejemplos: el votante no tiene forma de saber que el código QR que se le presenta es el correspondiente a su voto (podría tratarse de un código ya repetido de otro votante que eligió al mismo candidato) y su voto podría no haber sido tenido en cuenta.

Por otra parte, si el PC o el dispositivo móvil del usuario está infectado con un malware que permite espiar los códigos PIN asociados a su *ID Card*, en el caso de que el votante vuelva a insertar su *ID Card*, incluso para otra actividad totalmente distinta, el malware podría suplantar su voto y el votante no se daría cuenta.

En noviembre de 2014, Springall et al. [236] presentaron un paper con un exhaustivo repaso a su labor como observadores independientes de las elecciones locales estonias de 2013 (utilizando ya la segunda versión mejorada del sistema de VER).

Su publicación fue polémica puesto que incluía una larga lista de errores procedimentales y de diseño, concluyendo con la recomendación de la cancelación del VER en Estonia inmediatamente.

El listado de errores y brechas de seguridad incluían:

- Ignorar mensajes de error de los servidores, prohibir grabar en la sala de servidores, una mala gestión de las copias de backup, el uso de conexiones HTTP inseguras, el uso de dispositivos USB personales con datos previos para el transporte de información crítica y una insuficiente transparencia general.
- Errores en el código publicado que hacían al sistema vulnerable a ataques DDos.
- Ataques tanto sobre la parte del cliente software como sobre la parte de servidores.
- Error de diseño por el cuál el servidor de recuento tiene acceso a la asociación entre los votos encriptados y los votos desencriptados. Por ello, un atacante interno podría potencialmente suplantar dicha información relacionada (y grabada físicamente en un DVD) por otra de su interés, comprometiendo el resultado de las elecciones.

Por todo ello, los autores del paper concluyen que un atacante a nivel estatal (la conocida como "*cyberwarfare"*), un "atacante sofisticado" o un *"insider"* deshonesto podrían romper el sistema de VER tanto desde un punto de vista procedimental como técnico. Por todo ello concluyeron solicitando a Estonia que paralizase el VER.

Las autoridades estonias por su parte acusaron a los autores de presentar su trabajo y convocar una rueda de prensa sin avisar y únicamente 3 días antes del comienzo del período de votación. De hecho, uno de los autores, Halderman, ha sido acusado de una cuestión similar en las elecciones estatales de Nueva Gales del Sur [288, 292].

En el caso de las elecciones europeas de 2014, se identificaron errores en la implementación de la verificación con iOS.

Además, un total de 1131 sesiones de voto fallaron y devolvieron un mensaje de error en el que se decía que el certificado usado para firmar el voto no era todavía válido. El mensaje de error se debía a un *bug* por el que no se tenía en cuenta la zona horaria cuando se verificaba la validez del certificado. Ello afectó a los votantes que habían actualizado el certificado el mismo día en que intentaron votar [249].

Con posterioridad a todos estos eventos, en diciembre del 2015, S. Heiberg et al. presentaron un exhaustivo trabajo [249] sobre el análisis de los registros de las votaciones de 2013, 2014 y 2015.

Se trata de un paper interesante debido a que incluye el análisis de las elecciones parlamentarias de 2015 y aporta también una mención clara y directa al trabajo de Springall et al. [236] que dejaba en evidencia una larga lista de carencias del sistema de VER estonio.

Además, sus autores indican que no se ha producido ningún tipo de ataque a gran escala en las elecciones de 2013, 2014 y 2015.

Para contrastar dicha afirmación, se apoyan en un componente software en Python que se ha desarrollado *"ex-profeso"* para realizar la labor de registro y monitorización de los votos. (El volumen de registros generados no permite una revisión manual. En las elecciones de 2015 se generaron más de 5,5 millones de mensajes en el registro).

El uso del sistema de VER estonio en la votación de 2015 fue el siguiente:

Session kind	Sessions	Voters	Voters (u)
All sessions	211,215	–	–
Voting	201,811	179,262	2,771
Successful	181,084	176,491	0
ID card	159,000	155,267	0
Mobile-ID	22,084	21,307	0
Unsuccessful	20,727	15,007	2,771
ID card	14,328	11,226	2,366
Mobile-ID	6,399	3,864	422
Verification	9,404	7,563	41
Successful	8,439	7,522	0
Unsuccessful	965	120	41

Tabla 4: VER en las elecciones parlamentarias de 2015 de Estonia [249]

Las 20.727 sesiones de voto no exitosas se desglosan de la siguiente manera:

Reason for failure	Sessions	Voters	Voters (u)
Unsuccessful voting sessions	20,727	15,007	2,771
Explicit error	5,513	3,405	826
Common error	1,509	1,289	404
Maintenance	1	1	0
Under-aged voter	30	30	27
Ineligible voter	507	307	294
Voting ended	2	2	1
No new voters	87	77	54
Session expired	882	877	31
Certificate issue	641	298	271
ID card	572	298	271
Mobile-ID	69	–	–
Pre-2011 Mobile-ID user	366	249	89
Bad Mobile-ID number	974	–	–
DigiDocService failure	0	0	0
Mobile-ID failures	1,956	1,553	70
Incident	67	34	3
Other reason	15,214	12,072	2,009
Discontinued (Mobile-ID)	1,454	1,039	68
Authentication	1,008	731	51
Signing	446	415	20
Abnormal	0	0	0
Vote not submitted	13,760	11,103	1,947
ID card	12,283	9,779	1,744
Mobile-ID	1,477	1,353	206

Tabla 5: Resumen de las sesiones de voto no exitosas de VER en las elecciones parlamentarias de 2015 de Estonia [249]

Las sesiones no exitosas suponen alrededor de un 10% del total de accesos.

Por su parte, 4.034 votantes decidieron repetir su voto, siendo el segmento de población que más utilizó en VER el de 30-40 años. En cuanto al género, un 52,6% de los votantes fueron mujeres y un 47,4% hombres.

Con posterioridad a las elecciones se hicieron públicos una serie de eventos relacionados con la seguridad del VER: la incorrecta encriptación de un voto (reclamada por un activista del Partido Pirata estonio), la acusación de compra de votos en una residencia de la tercera edad en Voru o las dudas vertidas por una televisión pública sobre una distribución sospechosa de la edad de los votantes que usaron el VER (hubo más votantes de 90 años que de 18 años que usaron el VER).

En el capítulo 6 de [249] se detallan los ataques en sus distintas variantes si bien concluyen que no ha existido ningún evento que pudiese definirse como *"ataque"* contra el sistema de VER sino *"anomalía"*.

Por último, apuntan a la incapacidad de investigar a fondo las causas de dichas *"anomalías"* por razones técnicas y de protección de la privacidad por parte de las autoridades estonias.

Heiberg et al. concluyen que no se ha producido ningún ataque a gran escala al sistema de VER estonio y que las *"anomalías"* que no se han podido explicar constituyen un buen punto de partida para posteriores trabajos de investigación.

Su labor es minuciosa y de un gran valor pero cabe puntualizar que los autores no son totalmente imparciales, puesto que al menos dos de ellos pertenecen a la empresa desarrolladora del sistema de VER estonio (Cybernetica) o a un Centro de Excelencia vinculado a ella (Smartmatic – *Cybernetica Centre of Excellence for Internet Voting*).

La última noticia que se ha producido con respecto al voto remoto en Estonia data de finales de enero de 2016. El Tribunal Europeo de Derechos Humanos (ECHR) ha decidido aceptar la apelación recibida por parte de la asociación *MTU Ausad Valimised* con respecto a la sanción que les fue impuesta por el Comité de Protección al Consumidor como consecuencia de la campaña llevada a cabo entre el 18 de marzo y el 7 de abril de 2013 en la que advertían sobre los riesgos del voto electrónico [254].

Conclusión

El caso de Estonia es de un enorme valor para el avance de los sistemas de VER por la decidida apuesta y la experiencia real acumulada durante más de 10 años.

En cuanto al sistema en sí, no se tiene evidencia de que hayan ocurrido ataques a gran escala. No obstante, las debilidades descubiertas [236], unidas al hecho de no cumplir con

la condición de ausencia de recibo y la existencia de *"anomalías",* indican que actualmente presenta una serie carencias potencialmente peligrosas y explotables por parte de un atacante con suficientes medios. Estonia ejemplifica perfectamente la necesidad de añadir criterios adicionales a la metodología de evaluación de sistemas de VER. Como se ha visto en [236], la gran mayoría de errores detectados no hubiesen sido tenidos en cuenta dentro de los requerimientos tradicionales detallados en el apartado 2.3.

En cuanto al sistema de VER en sí, las carencias observadas deberían solventarse para poder afirmar que es suficientemente seguro para su uso en elecciones VAP. Una mejor predisposición de las autoridades estonias de cara a la revisión del sistema por parte de la comunidad científica redundaría en una mayor seguridad y confianza en el esquema.

Para profundizar en el VER en Estonia, referirse a las siguientes referencias bibliográficas: [56, 236, 243, 244, 245, 246, 247, 248, 249, 254].

3.2.2 Noruega

Otra experiencia paradigmática en el uso de sistemas de VER para elecciones vinculantes en el ámbito político es la que tuvo lugar de 2008 a 2014 en Noruega.

Pese a que en 2014 se canceló el programa y no hay visos de que se recupere (al menos en el corto o medio plazo), desde varios puntos de vista el caso del VER en Noruega es un ejemplo a seguir por países interesados en implantarlo:

- En primer lugar, las autoridades noruegas se tomaron el tiempo necesario para organizar, evaluar, planificar y aprobar el proyecto del VER.
- En segundo lugar, al tratarse de una cuestión tan seria como el voto en unas elecciones políticas vinculantes, se huyó de partidismos y se incluyó a expertos e investigadores independientes desde el primer momento.
- A mayores, todo el proceso fue gradual: únicamente cuando hubo un informe positivo del comité de expertos se decidió seguir adelante con la aprobación parlamentaria del proyecto piloto. Desde ese momento se tardaron 3 años más en realizar las primeras pruebas de VER en 10 referendos locales.

Los puntos comentados podrían parecer obvios y su aplicación de sentido común, pero la práctica indica que en muchas ocasiones ésto no ha sido así.

Cronología

Henrik Nore, *Project Manager* desde 2008 a 2014 del *Election Management Body* (EMB) del Ministerio de Gobierno Local y Modernización de Noruega, presenta en [253] la cronología del sistema de VER noruego:

- **Mayo de 2004**: El Ministerio de Gobierno Local y Desarrollo Regional organiza un grupo de trabajo para evaluar la introducción del *e-voting* (sic) en el país.
- **Febrero de 2006**: El grupo de trabajo presenta su informe *"Electronic Voting, Challenges and Opportunities"*. En él, apuestan por una gradual introducción del "voto electrónico" por su impacto positivo sobre la accesibilidad al voto, una mayor rapidez y precisión en el recuento y un ahorro en costes en el largo plazo.
- **2007**: Aprobación parlamentaria no unánime del proyecto de VER en Noruega
- **2008**: Inicio del proyecto.
- **2009 y 2010**: Proceso abierto de concurso para elegir el proveedor del sistema de VER y prueba del mismo en 10 referéndums locales.
- **Septiembre 2011**: Primera utilización del sistema de VER en unas elecciones VAP (elecciones locales y regionales).
- **Septiembre 2013**: Utilización en las elecciones parlamentarias noruegas.
- **Junio 2014**: El Ministerio de Gobierno Local y Modernización decide poner fin al proyecto de VER en Noruega.

Por tanto, pasaron 7 años desde la configuración del grupo de trabajo hasta la utilización en elecciones reales, pasando por una serie de pruebas en referendos consultivos. Únicamente la elaboración del informe de viabilidad llevó casi 2 años. Además, se incluyó a expertos independientes desde un principio, incrementando la transparecia y fiabilidad.

Concurso público y transparencia

Una vez lanzado oficialmente el proyecto, el proceso de *"tender"* o concurso fue totalmente público y transparente, llevando prácticamente todo el año 2009. Se inició con un "diálogo competitivo" donde se ponían a prueba las capacidades técnicas de las empresas que optaban al contrato. A mayores, todas las reuniones e intercambios de información se realizaron en ingles, fueron grabados y hechos accesibles públicamente.

Como consecuencia del citado "diálogo competitivo", el número de empresas candidatas pasó de 11 a 3 (ErgoGroup, Computas e Indra), que fueron las que presentaron formalmente oferta. Finalmente, el consorcio formado por ErgoGroup y Scytl fue el vencedor.

La transparencia del proceso fue tal que actualmente (noviembre de 2016) todavía está disponible el link oficial con toda la información pública sobre el concurso, incluyendo los archivos de las ofertas enviadas por las distintas empresas candidatas en 2009 [250].

El exquisito compromiso de las autoridades noruegas por preservar totalmente la transparecia durante todo el proceso de elección de la empresa concesionaria es sin duda un ejemplo sobre cómo abordar la potencial introducción de un sistema de VER para elecciones VAP en un país.

Código fuente abierto (OSS)

La decisión sobre exigir código abierto no fue un requerimiento político. La motivación, en palabras de H. Nore en [253] fue el deseo de una absoluta transparencia en la administración de las elecciones.

No obstante, la apuesta no estuvo exenta de riesgo puesto que los responsables del proyecto no querían "ahuyentar" a potenciales empresas desarrolladoras con la suficiente capacidad para elaborar un sistema de VER de alta calidad (y que en principio suelen ser reticentes a desvelar completamente su código fuente).

Por ello, omitieron a propósito el requerimiento de códido fuente abierto en los prerrequisitos. Posteriormente, según avanzaba el proceso de diálogo competitivo, los responsables noruegos del proyecto sondearon a varias empresas y descubrieron para su alivio que al menos dos no se oponían totalmente (al menos de palabra).

Cuando finalmente decidieron añadir el requisito de código abierto, fue un momento de tensión y riesgo porque si se hubiesen retirado todas las empresas candidatas, el proyecto del VER hubiese quedado seriamente debilitado.

Para su fortuna, la propuesta salió bien y el sistema vencedor fue OSS. Según los responsables noruegos, ello abarató las tareas de revisión y corrección de errores y permitió encontrar *bugs* que en algunos casos comprometieron la seguridad de los votos.

Sorprendentemente, ninguna de las partes interesadas o *stakeholders* revisaron el código y finalmente tuvieron que contratar a una consultora especializada para ello. Se encontraron [256] vulnerabilidades en la generación de valores aleatorios inseguros y en el uso de claves no suficientemente largas.

En cualquier caso, para futuras experiencias de VER en el ámbito político sería interesante tener en cuenta la baja participación de los *stakeholders* en su revisión en el caso de Noruega pese a su demanda previa de que el sistema fuese OSS.

Estructura del sistema de VER noruego

Desde un punto de vista criptográfico y como se ha visto en el capítulo 2.2, existen 2 principales tipos de esquemas de VER para el recuento de votos: el basado en propiedades homomórficas (punto 2.2.4.6) y el basado en mix-nets (punto 2.2.4.8).

En el caso de las elecciones noruegas, debido a su naturaleza compleja en la que hay que seleccionar no solo una lista sino elegir, ordenar o incluso añadir candidatos de otras listas, la opción del recuento homomórfico pierde efectividad y no es recomendable.

Los desarrolladores decidieron encriptar los votos con ElGamal y posteriormente utilizar un sistema de mix-nets para romper el vínculo entre voto y votante.

Se optó también por el uso de códigos de retorno a través de un canal alternativo (SMS) para evitar en lo posible el riesgo de ataque si el ordenador estuviese infectado.

La siguiente figura resume el protocolo del sistema de VER noruego:

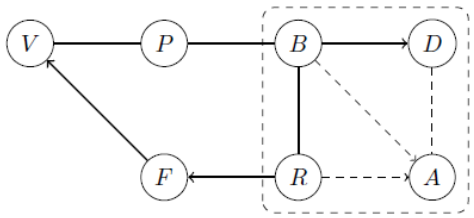


Figura 25: Resumen del protocolo de VER noruego con sus distintos actores [261]

- El votante V introduce su voto en el ordenador P el cuál lo encripta y envía a la urna B.

- La urna B y el generador de códigos de retorno R cooperan para computar una secuencia de códigos de retorno asociados al voto.
- Dichos códigos se envían por SMS al teléfono móvil F del votante.
- Posteriormente, el votante verifica el código recibido con la lista de opciones precomputadas que tiene en su poder desde antes de las elecciones y que le ha sido enviada por correo ordinario.
- Una ver concluye el período de votación con VER se cierran las urnas.
- Únicamente entonces, los votos encriptados o textos cifrados incluidos en B son desencriptados por el desencriptador D.
- El auditor A por su parte supervisa el proceso completo.

Cabe destacar que el protocolo arriba explicado ha sido similar en las elecciones de 2011 y de 2013 pero para 2013 se introdujeron dos avances para hacerlo más eficiente: la variante multi-ElGamal y se mejoraron las NIZK.

El resto de mejoras fueron:

- Implementación del cliente de voto en Javascript
- Mejora de la verificabilidad individual, permitiendo a los votantes comprobar que sus votos estaban en el tablón (*recorded-as-cast*)
- Optimización del sistema de *threshold* para la distribución de claves de desencriptación y recuento
- Implementación de una nueva versión de mixnet verificable que suministraba pruebas de detección de manipulación en vez de análisis heurísticos [42].

Por otra parte, el modelo de VER noruego parte de una serie de premisas sobre canales autenticados, confidenciales e identificados entre las distintas partes de la infraestructura:

- En lo que respecta a la comunicación con el teléfono F del votante V, se considera que los mensajes son confidenciales y que el adversario no puede interferir en su integridad.
- Se usan modelos idealizados de la PKI y de las funciones *hash*.

La siguiente figura representa de una manera más práctica el sistema noruego:

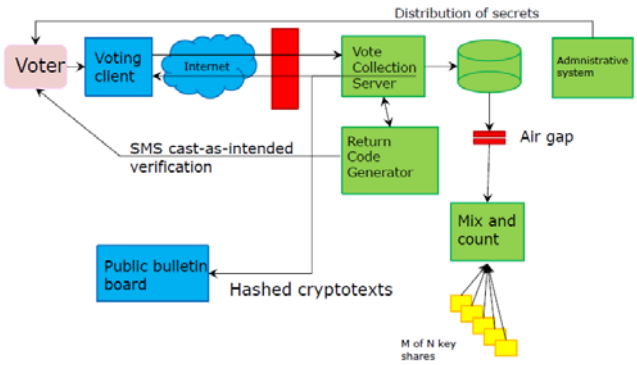


Figura 26: Modelo detallado del sistema de VER noruego [262]

El votante podía autenticarse a través de un portal de internet sobre una conexión segura de 4 maneras:

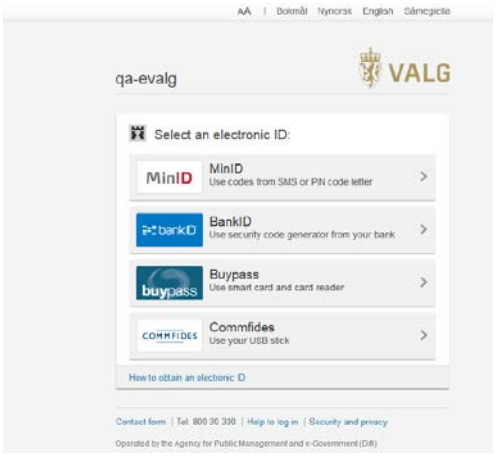


Figura 27: Autenticación en el sistema de VER noruego [262]

Una vez autenticado, procede a votar:

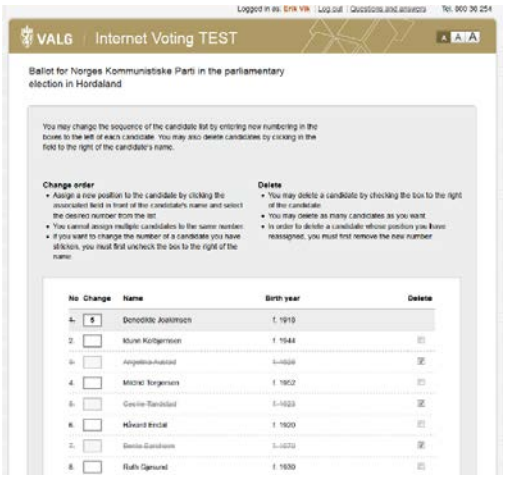


Figura 28: Votación en el sistema de VER noruego [262]

Y una vez enviado el voto, se envía al dispositivo móvil del votante la prueba de verificación.

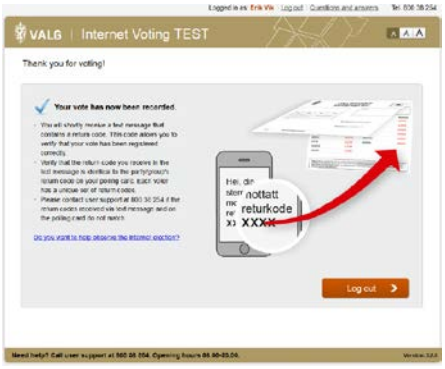


Figura 29: Envío de SMS de verificación en sistema de VER noruego [262]

El votante recibe un SMS confirmando varios detalles de su voto, así como el código asignado al partido que ha votado en su tarjeta de votación electoral única y personal que le fue enviada con antelación a la apertura del período de votación con el sistema de VER.

Cada tarjeta electoral tiene unos códigos de 4 números aleatorios asignados a cada partido por lo que no hay dos tarjetas iguales:

Figura 30: Envío de SMS de verificación de opción votada y tarjeta electoral personal [262]

Por último, se ofrece al votante la opción de comprobar que su voto se encuentra efectivamente en la urna digital. Para ello, éste se puede descargar el *hash* de su voto del tipo SHA-256, acceder a la página Git-Hub donde se actualiza la urna digital cada hora y comprobar que su(s) *hash* de voto(s) está(n) presente(s):

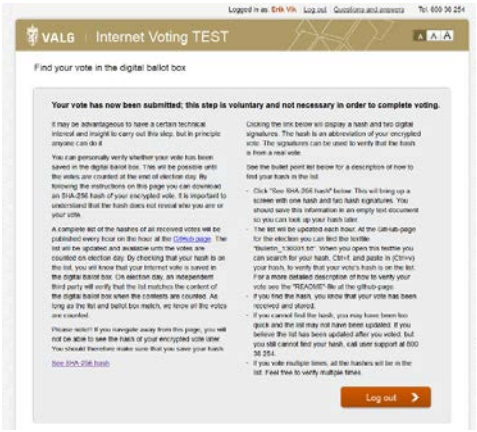


Figura 31.: Opción de comprobación de *hash* de voto en urna digital [262]

Aparte del procedimiento de votación y el modelo detallado el sistema, es también interesante referirse a la separación de funciones en el modelo noruego. Tal y como apuntan Barrat et al. [258] y Belleboni [56], se han realizado esfuerzos notables para evitar la acumulación de roles y capacidades, optando por distribuirlos para aumentar la seguridad:

Los servidores de voto son controlados por dos ministerios distintos (uno de ellos por el Ministerio de Justicia y el otro por el Ministerio de Industria y Comercio) y todos los procesos de mantenimiento de los servidores encargados del cribado de votos, mezclado,

desencriptación y recuento se realizan de manera transparente y bajo el control de distintos *stakeholders*.

Aun así, podrían darse ataques por colusión entre las partes tal y como se explica en [56]:

- Una colusión entre el servicio de autenticación, el generador de las tarjetas electorales y el generador de códigos de retorno permitiría añadir votos fraudulentos. Dicha colusión no es usual ni cuenta con unas grandes probabilidades de ocurrir pero existe.
- El sistema de generación de recolección de votos con el generador de recibos podría llegar a eliminar votos.
- El auditor en colusión con el sistema de mix-nets podría llegar a eliminar votos.

Por lo que respecta a la protección contra la coerción, la principal medida es la posibilidad de re-votar. Pese a tratarse de una idea sólida e interesante, la realidad ha puesto de manifiesto que no es perfecta. Por ejemplo, la tasa de votantes mayores de 90 años en Noruega creció de una manera que indica que quizás otros miembros de su familia/lugar de residencia pudieron haber votado en su lugar. Ello no quiere decir que haya habido coerción, porque puede ser que su "representante" votase de acuerdo a la voluntad del anciano. Aún así, la privacidad no se hubiese respetado.

En caso de que hubiese algún problema con el VER, de manera contemporánea hay habilitados una serie de colegios electorales donde se puede votar en papel. El voto en papel siempre prevalece sobre el VER.

A mayores, el votante puede volver a votar en papel el día de las elecciones, lo que anularía cualquier voto anterior fuese en el formato que fuese.

Como se apuntó en [260], el problema que conllevó el hecho de permitir votar más de una vez es que había que mantener los votos relacionados con el votante hasta que se cerrase el período de VER, aumentando el riesgo de que en caso de un ataque exitoso, se pusiese en riesgo la privacidad de los votantes.

Elecciones locales de 2011

Las elecciones locales de supusieron el primer piloto real de utilización de VER vinculante en el ámbito político en Noruega.

Participaron 10 de los 429 distritos del país y tuvieron lugar en los días 11 y 12 de septiembre de 2011. El período de tiempo asignado para el VER fue del 10 de agosto al 9 de septiembre de 2011.

En la siguiente tabla se detallan las principales estadísticas de participación:

Municipality	Eligible Voters	Turnout	Turnout (%)	Internet Votes	Internet Votes (as % of total)	Cleansed Votes	Cleansed as % of Internet Votes
Ålesund	34,535	20,580	59.59%	5,434	26.40%	245	4.31%
Bodø	36,635	23,936	65.34%	6,957	29.07%	269	3.72%
Bremanger	2,955	1,938	65.58%	407	21.00%	38	8.54%
Hammerfest	7,752	4,349	56.10%	1,126	25.89%	64	5.38%
Mandal	11,764	7,354	62.51%	1,457	19.81%	66	4.33%
Radøy	3,687	2,459	66.69%	768	31.23%	42	5.19%
Re	6,870	4,384	63.81%	981	22.38%	61	5.85%
Sandnes	48,689	30,358	62.35%	8,193	26.99%	325	3.82%
Tynset	4,163	2,855	68.58%	903	31.63%	56	5.84%
Vefsn	10,456	6,161	58.92%	1,328	21.55%	58	4.18%
TOTAL	**167,506**	**104,374**	**62.31%**	**27,554**	**26.40%**	**1,224**	**4.25%**

Tabla 6: Participación y utilización del VER en las elecciones locales noruegas de sept. 2011 [257]

El total de votantes llamados a las urnas en los 10 distritos participantes fue de 167.506, de los cuales votaron 104.374 o bien un 62.31%.

El número de votos emitidos a través del sistema de VER fue de 27.554 o bien un 26,4% sobre el total de votos. La columna *"Cleansed Votes"* se refiere a los votos que fueron eliminados de la urna debido a que el votante decidió volver a votar, ya sea:

- De nuevo a través del sistema de VER
- En papel durante el período habilitado para el VER
- En papel durante las jornadas electorales tradicionales

Pese a publicarse el código fuente para su revisión por parte de la comunidad científica y otros *stakeholders*, no hubo ningún *feedback*. En consecuencia, las autoridades noruegas contrataron a la empresa Computas AS (finalistas del tender que ganó Scytl) para realizarla.

El hecho de que sea el mismo gobierno quién pague la auditoría podría levantar sospechas sobre la objetividad del informe. No obstante, la alternativa, que era que no se auditase el sistema, era peor.

Pese a que no se encontraron errores generales o masivos, Computas sí que localizó entre 54 y 57 *hashes* de votos que se encontraban en el Generador de Códigos de Retorno y de los que en cambio no había rastro en Servidor Recolector de Votos.

Adicionalmente se produjeron irregularidades en una pequeña parte de las tarjetas de votación con códigos aleatorios para cada votante. En [252], Gebhardt et al. repasan los problemas derivados del uso de dos centros de impresión físicamente separados: la aleatorización manual de las tarjetas por una cara, su transporte al segundo centro de impresión, la necesidad de cambiar a un papel de gramaje menor al haberse agotado el primer tipo utilizado, un proceso de impresión más lento de lo calculado etc.

A consecuencia de ello, se detectaron 74 casos en los que los códigos impresos no coincidían con los que el votante tenía a su disposición en la herramienta de VER. A mayores, las autoridades recibieron otras 35 llamadas de incidencias relacionadas con los códigos. Sobre un total de más de 27.000 votos emitidos por internet, no suponen un elevando porcentaje pero aún así conviene ser mencionado.

En cuanto a los protocolos de gestión y separación de funciones, pese a que metodológicamente estaban bien desarrollados, los plazos previstos se demostraron demasiado ajustado a medida que iban surgiendo imprevistos.

Respecto a las premisas de seguridad, se supone que el *Vote Collector Server* o VCS y el *Return Code Generator* o RCG no cooperan para romper la privacidad del votante.

Otra posibilidad de ataque es que un grupo de conspiradores reporten de manera maliciosa que han recibido códigos de retorno incorrectos. Las autoridades no tienen forma de saber si dichas afirmaciones son ciertas sin romper la privacidad del votante por lo que sería un problema de difícil solución.

Los ataques arriba detallados, aun siendo potencialemente posibles, difícilmente se podrían producir a gran escala. Además, la correcta monitorización del sistema ha hecho que se haya detectado incluso un número muy reducido de incidencias.

En [257], dos de las principales conclusiones del proceso de votación de 2011 fueron:

- La esperada mejora en el tiempo de recuento no se produjo, en buena medida por el uso de mix-nets y la imposibilidad de comenzar el recuento hasta que no se cerraron las urnas y se transfirió su contenido
- El uso de un sistema de VER no redujo la confianza de los votantes en las elecciones o en las propias administraciones públicas.

Por todo ello, el primer experimiento piloto en unas elecciones vinculantes en el ámbito político en Noruega fue un razonable éxito, con algunos contratiempos, pero ninguno de ellos críticos o a gran escala.

Elecciones Parlamentarias de 2013

Tras el moderado éxito del uso del VER en las elecciones locales de 2011, las autoridades noruegas decidieron ampliar la siguiente prueba piloto a 12 distritos para un total de 250.159 votantes (aproximadamente un 7% del total de 3.600.000 noruegos llamados a las urnas).

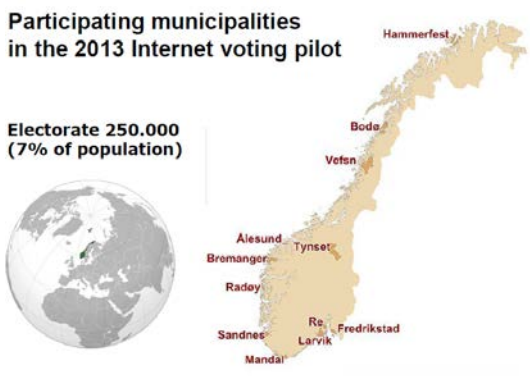


Figura 32: Distritos participantes en el piloto de VER noruego en las elecciones parlamentarias de 2013 [262]

El período de votación anticipado a través del sistema de VER y en papel en determinados colegios electorales abarcó del 12 de agosto al 6 de septiembre de 2013.

Se emitieron un total de 72.969 votos a través del sistema de VER, que resultaron en 70.090 después del proceso de eliminación de votos, representando un 36.4% del total de votos emitidos.

En cuanto al sistema de VER, la base es la misma que en las elecciones locales de 2011 y los responsables del proyecto y las empresas desarrolladoras tampoco cambiaron.

Lo que sí se produjo fue una mejora del mismo basándose en los análisis tanto internos como externos durante el anterior piloto:

- Se mejoró el sistema de VER desde un punto de vista criptográfico (en concreto el criptosistema de ElGamal y la NIZKP) como ya se apuntó [261]
- Se optimizó el sistema de distribución de claves necesarias para la desencriptación y el recuento entre los 9 miembros del IEC (*Internet Election Commitee*): hacían falta al menos 6 de ellos para crear la clave de desencriptación de los votos
- La impresión de las tarjetas de códigos se centralizó en un único centro para minimizar el riesgo de que volviesen a suceder las incidencias que tuvieron lugar en 2011.
- Se introdujo un nuevo sistema de administración central electrónica mejorada denominado EVA (*Elektronisk Valgadministrativt System*)
- El cliente de voto se implementó totalmente en Javascript
- Se mejoró la comprobación de la corrección del voto
- Se integró una mix-net verificable (*Verificatum* [428])
- Se mejoró la verificabilidad individual, permitiendo a los votantes comprobar que sus votos estaban en el tablón (*recorded-as-cast*)

Con ocasión de estas elecciones se contrató a la empresa Quality AS para la auditoría y monitorización de todo el sistema VER y sus protocolos asociados.

Uno de los errores encontrados por ellos involucraba un fallo en ElGamal que hacía que se generasen números fijos, comprometiendo la encriptación de los votos y por tanto su privacidad (aunque no su integridad porque no se podrían modificar al estar firmados digitalmente). Para cuando se descubrió el 3 de septiembre, una importante cantidad de votos (29.000) podían estar potencialmente afectados.

El error se subsanó inmediatamente y se mejoró el criptosistema. Dos días más tarde, el Ministerio emitió un comunicado en el que explicaba el problema e incidía en su correcta solución y mejora global de la seguridad. Las autoridades decidieron continuar con las elecciones sin cambios. También en [264], profesores e investigadores de la Universidad de Berna hallaron una serie de debilidades mostrando que todavía quedaban vulnerabilidades que debían ser solucionadas.

Finalmente y pese a que no se produjeron ataques conocidos al sistema de VER en las elecciones al parlamento de 2013, el Ministerio de Gobierno Local y Modernización decidió el 25 de junio de 2014 cancelar el proyecto de VER en Noruega debido a que "no hay un deseo político amplio de introducir internet voting (sic)" y a que "los votantes tienen un conocimiento limitado sobre los mecanismos de seguridad del sistema" [251].

Conclusión

Pese a su cancelación, la experiencia del VER en las elecciones vinculantes en el ámbito político en Noruega puede calificarse de éxito. No ha estado exento de eventualidades de código y procedimentales, aunque no a gran escala.

Dos de las grandes aportaciones de Noruega al campo del VER han sido:

- El reseñable esfuerzo de transparencia en todo el proyecto, desde el *tender* al seguimiento y la contratación de expertos idependientes.
- El excelente proceso de diseño e implantación de un sistema de VER, asignando suficientes medios y tiempo, permitiendo e incluso fomentando activamente la participación de expertos e investigadores desde un primer momento.

Quedó como asignatura pendiente cómo atraer e incrementar la labor de supervisión por parte de investigadores y demás *stakeholders*.

En cuanto a la cancelación del proyecto, no hay unanimidad sobre las causas:

Los participantes en el proyecto [262, 263] arguyen que el fin del proyecto se debe a motivos únicamente políticos. Afirman que en el año 2014 cambió el gobierno noruego a un

partido que tradicionalmente se había opuesto al VER (el Partido Conservador) y que por ello decidieron concluir el piloto.

Por otra parte, Halderman et al. defienden en [23] que el sistema de VER noruego no es E2Ev y que por tanto no se debería de usar en elecciones, menos todavía en unas vinculantes en el ámbito político.

En el momento de escribir estas líneas (noviembre de 2016) el proyecto continúa cancelado aunque no se debe descartar una posible reactivación del VER en Noruega, quizás ligada a un nuevo cambio en el partido gobernante.

El lector que quiera investigar más detalladamente sobre el sistema de VER noruego puede referirse a las siguientes referencias como bibliografía recomendada: [56, 251, 252, 253, 255, 256, 257, 258, 260, 261, 262, 263, 264, 369, 430].

3.2.3 Canadá

Los casos de Estonia y Noruega suponen dos de los mejores ejemplos de introducción de sistemas de VER en elecciones vinculantes en el ámbito político. En ambos países ha existido un plan coordinado a nivel estatal para su introducción gradual.

En el resto de experiencias estudiadas en el presente capítulo 3.2, la adopción de sistemas de VER no se produce como consecuencia de una planificación a nivel nacional sino desde entidades locales que deciden, en virtud a la autonomía que les otorga la legislación electoral de su país, introducir modalidades de voto alternativas.

Por tanto, cada municipalidad o condado decide introducir el VER (para aumentar la participación de sus votantes, por tratarse de una zona especialmente remota etc.) de manera autónoma, independiente y no coordinada.

Dentro de la categoría de sistemas de VER independientes en el ámbito local, Canadá es uno de los países que mayor número de experiencias acumula. Al tratarse de elecciones no relacionadas entre sí, sin un hilo conductor común y con procedimientos más locales y atomizados, resulta mucho más difícil tener acceso a información y datos unificados.

La Dra. Goodman, en sus papers *"The Patchwork of Internet Voting in Canada"* [266] e *"Internet Voting in Ontario: Time for Overarching Standards"* [269] de 2014 y 2015 respectivamente, aborda en detalle el problema de la falta de coordinación y estándares en el campo del VER en Canadá.

En primer lugar, establece la jerarquía de los distintos Cuerpos de Gestión Electorales (*Electoral Management Bodies* o EMBs) canadienses: Gobierno Federal, Provincias, Municipalidades y Primeras Naciones o *"First Nations"*.

Goodman explica que cada nivel administrativo goza de una gran independencia y que la comunicación y coordinación entre ellos es en general mejorable.

En la actualidad, ni el EMB federal ni ninguno de los EMBs provinciales tiene un plan concreto de sistemas de VER coordinados al menos hasta 2019. Por tanto, es muy difícil encontrar algún tipo de estandarización entre las experiencias de VER para establecer alguna comparación entre ellas.

Sí que es cierto que en algunas como British Columbia han preparado en fechas relativamente recientes (febrero de 2014) estudios de viabilidad como un primer paso previo a una potencial introducción del VER [268].

En lo que respecta al nivel municipal, se han producido experiencias relevantes en las provincias de Ontario y Nova Scotia desde 2003 y 2008 respectivamente.

Conviene también aclarar el caso de las *"First Nations"* o Primeras Naciones: Se trata de 634 comunidades compuestas por los pobladores originales de los territorios correspondientes al Canadá actual. Disponen de un estatus legal especial y cada una de ellas lleva a cabo una serie de elecciones y referendos para elegir a sus líderes y para otra serie de bandos y cuestiones propias de la comunidad.

No existe uniformidad sobre la ley estatal de aplicación para las elecciones, referendos y bandos de cada comunidad:

- Actualmente, existen 238 comunidades que se rigen por el *Indian Act* y el *Indian Act Election Regulations*. Dicha legislación no prevé la utilización de sistemas de VER por lo que las comunidades regidas por ella no pueden implementarlos.

- Las restantes 396 comunidades se rigen por acuerdos propios o códigos de elección personalizados. Por ello, en su caso sí que podrían introducir sistemas de VER para las votaciones comunales.

Existen algunos precedentes de VER en las siguientes comunidades de Primeras Naciones: *Huu-ay-aht, Tahltan* y *Squamish* en la Columbia Británica entre 2011 y 2014, *Nipissing* en Ontario en 2013 y 2014 así como la organización *"Unión de Indios de Ontario"* en 2014.

En lo que se refiere a otras experiencias de VER en Canadá, se encuentran las elecciones dentro de agrupaciones, sindicatos, colegios profesionales, partidos políticos etc. las cuales tuvieron lugar por primera vez en 2003 con la elección del candidato del *New Democratic Party* a nivel nacional, si bien su implantación sostenida tuvo lugar a partir de 2009.

Principales experiencias de VER en elecciones públicas vinculantes en el ámbito político en Canadá

Tal y como se ha indicado anteriormente, la falta de legislación y estandarización del VER en Canadá a nivel estatal hace que las experiencias sean aisladas, atomizadas y difícilmente comparables entre ellas.

No obstante, existen regiones que han mostrado un mayor compromiso el VER, tratando de empujar *"desde abajo"* para regular y unificar las elecciones que se van produciendo. En ese sentido, Ontario y Nueva Escocia son las provincias que mayor experiencia han ido acumulando desde 2003 y 2008 respectivamente.

Ontario

En la provincia de Ontario vienen produciéndose elecciones vinculantes a nivel local desde el año 2003 con uso del VER, destacando el caso de la ciudad de Markham.

Inicialmente, la motivación principal para su introducción fue la de revertir la decreciente participación en los comicios locales. Dicho objetivo no se consiguió, pero se concluyó que el VER podía suponer una forma más eficiente de acercar las elecciones al electorado y al menos frenar el ritmo de caída en la participación.

En consecuencia, se produjeron nuevos pilotos en 2006, 2010 y 2014 cada vez con más localidades adscritas, si bien la mayoría de ellas (un 70%) de menos de 10.000 habitantes

Se ha destacado el caso de Markham porque la ciudad y sus condados adyacentes tienen una población aproximada de 300.000 habitantes, con una penetración de acceso a internet de alta velocidad en los hogares de más del 80%, un 56% de población con titulación universitaria y más de 900 empresas tecnológicas instaladas, siendo el mayor municipio en haber implementado el VER en Ontario.

En cuanto al proceso de votación, se produce en dos fases:

- Todos los votantes reciben una notificación por correo postal para inscribirse on-line denominada VIP (*Voter Information Package*). El registro on-line incluye la introducción de un código PIN aleatorio que aparece en cada VIP, la fecha de nacimiento así como la selección de un código personal de 7 dígitos.
- El votante recibe un segundo correo postal con otro código PIN aleatorio para introducir junto con el password seleccionado para poder votar.

Se introduce también el uso de *captchas* para evitar accesos indeseados si bien el no poder votar varias veces o el no ser compatible el VER con la votación en papel el día de las elecciones induce a pensar que el sistema no cumple con las condiciones de E2Ev y RC.

El proceso de selección de la empresa proveedora de los servicios tampoco ha tenido los niveles de publicidad ni transparencia deseables.

No obstante, la participación ha sido notable como se puede apreciar en la siguiente tabla:

	2003	2006	2010
Population	~230,000	~260,000	~300,000
Eligible voters	158,000	164,000	164,000
Overall turnout (#)	42,198	61,948	65,927
Overall turnout (%)	28.0%	37.9%	35.5%
Internet voting registration (#)	11,708	16,251	17,231
Internet voting turnout (#)	7,210	10,639	10,597
Internet voting as % of eligible voters	4.5%	6.5%	5.7%
Internet voting as % of votes cast	17.1%	17.2%	16.1%
When offered	5 days during advance voting period; 24h/d	6 days during advance voting period; 24h/d	
Vendor	ES&S	ES&S	ES&S and Intelivote

Tabla 7: VER en Markham y alfoz (Ontario) en las elecciones locales de 2003, 2006 y 2010 [268]

Añadiendo el resto de condados de Ontario que participaron en las elecciones, el total de votantes llamados a las urnas en 2006 fue de 397.500 votantes y de 800.000 en 2010. En cuanto al coste, en Markham 2010, el presupuesto fue de 1.2 millones de CAD.

En las elecciones del año 2014, cada una de las 97 municipalidades que implementaron el VER eligió un proveedor de tecnología según sus necesidades/presupuesto. *Intelivote Systems inc* fue el proveedor seleccionado en 48 de ellas para un total de 909.000 votos, según su propia afirmación [272]. Por otra parte, en Markham, la empresa seleccionada fue la española Scytl [273] y según la web del Ayuntamiento de la ciudad, el total de votos emitidos a través del sistema de VER fue de 11.002 [274].

Nueva Escocia

En Nueva Escocia, el primer piloto de VER fue en el año 2008 comenzando con cuatro municipalidades. Ya en el año 2012 su número creció hasta 14 si bien únicamente la capital Halifax (participante en las dos ocasiones) posee un número de votantes relevante.

En todas las municipalidades menos en Halifax, el VER se permitía incluso en el día de las elecciones y no únicamente durante el período de votación anticipada.

En cuanto al proceso de votación, en 2008 (y en las elecciones menores de 2009 al consejo de la ciudad) se enviaba al votante un código PIN el cuál, junto con su fecha de nacimiento, servía para autenticarse y poder votar. En 2012 se mejoró el sistema y se adoptó una solución análoga a la de Markham explicada anteriormente: Se añadió un password elegido por el votante como tercera credencial de autenticación.

Aún así, la escasa información disponible, así como la precariedad del sistema de autenticación hacen prever que el sistema utilizado en Nueva Escocia (al menos hasta 2009) carecía de las suficientes garantías para ofrecer un sistema de VER suficientemente seguro.

Se permitía asimismo el voto remoto por teléfono, el cuál tampoco garantiza los niveles mínimos de E2Ev y RC.

En cuanto a la participación, la siguiente tabla recoge los principales guarismos:

	2008	2009 (district by-election)	2012
Population	~385,500	~385,500	~390,000
Eligible voters	279,326	12,476	298,209
Overall turnout (#)	101,116	4,391	110,114
Overall turnout (%)	36.2%	35.2%	36.9%
Internet voting registration (#)	N/A	N/A	N/A
Internet voting turnout (#)	~25,000	3,258	66,272*
Internet voting as % of eligible voters	9.0%	26.1%	22.2%
Internet voting as % of votes cast	24.7%	74.2%	60.2%
When offered	Three days during advance voting period; 24h/d	Five days from beginning of advance voting period to end of general voting day; 24h/d	13 days during advance voting period; 24h/d
Vendor	Intelivote		Scytl

Tabla 8: VER en Halifax (Nueva Escocia) en las elecciones locales de 2008, 2009 y 2012 [268]

Es destacable el porcentaje de votos en 2009 y 2012 (superior al 60%). La razón más plausible posiblemente sea la escasez de opciones de voto presencial.

Finalmente, dicha relevancia alcanzada por el VER sobre el total de votos emitidos ha llevado a la decisión de continuar con su uso para las elecciones locales de Halifax de octubre de 2016 [275].

Con el estudio del caso de Halifax y Nueva Escocia concluye el subapartado de experiencias de VER en Canadá implementadas y que siguen en activo.

Por lo que respecta a proyectos de VER que se evaluaron y finalmente se abandonaron, destacan dos: Kitchener en la provincia de Ontario en 2011 y Edmonton en la provincia de Alberta en 2012.

En el caso de Kitchener, en junio del 2011 se encargó la realización de un informe para evaluar la viabilidad del VER para las elecciones locales de 2014. En noviembre de 2012 se concluyó el informe dirigido por el secretario del ayuntamiento (*clerk*) y el 10 de diciembre de 2012 el consejo aprobó a propuesta del secretario la no introducción del VER.

Sus razones fueron:

- No se ha probado que afecte de manera relevante a la participación en unos comicios
- Su coste, añadido al de mantener el sistema de voto tradicional no era desdeñable
- Los sistemas de VER no son fáciles de auditar
- No existe en Canadá un estándar para la evaluación de sistemas de VER

Por lo que respecta a Edmonton, en la provincia de Alberta, en febrero de 2012 las capitales de los municipios de Edmonton, St. Albert y Strathcona enviaron una propuesta

conjunta al Ministerio de Asuntos Municipales para realizan un proyecto piloto de VER en las elecciones municipales de 2013 en Alberta.

Como respuesta, la ciudad de Edmonton realizó un simulacro de elección con VER y tecnología de Scytl el 2 de noviembre de 2012. Para evitar que tuviese sesgo político, los votantes tenían que elegir su color preferido de gominola. De ahí el nombre por el que se conoce el experimento: *"Jellybean Internet Voting Election"* [276].

Pese a no demandar requisitos especiales para tomar parte (únicamente completar un formulario on-line y subir una copia de un documento de identidad), únicamente 497 personas participaron en el simulacro.

Las autoridades de Edmonton también contrataron a una tercera empresa para comprometer el sistema de VER. Se realizaron 5 intentos de ataque por parte de la empresa contratada y otros 8 tuvieron lugar pese a no provenir de dicha empresa. Las autoridades comunicaron que los 13 ataques fueron repelidos.

Una vez concluido el piloto y obtenida toda la información relevante, el secretario del ayuntamiento de Edmonton escribió un informe positivo, recomendando la adopción del VER para las siguientes elecciones locales de 2013.

No obstante, en la reunión del Consejo de la ciudad del 6 de febrero de 2013, la propuesta fue rechazada, aduciendo que tenían reticencias respecto al coste y la seguridad en la introducción del VER.

En cuanto al nivel provincial y estatal, tal y como apunta la Dra. Goodman en [266], en la actualidad el responsable de todos los EMBs, conocido como CEO (*Chief Electoral Officer*) ha postpuesto el horizonte para el primer piloto coordinada a nivel estatal para 2019.

Para completar este apartado, el autor de esta tesis se puso en contacto con la Dra. Goodman y mantuvo una teleconferencia en junio de 2016 para actualizar datos y conocer de primera mano las últimas experiencias y noticias del VER en Canadá.

En la conversación, la Dra. Goodman incidió en las elecciones municipales de octubre de 2016 en Nueva Escocia en las que se vuelve a ofrecer el VER desarrollado por Scytl e Intelivote.

Por lo que respecta a Ontario, después de las elecciones municipales de 2014 en las que tomaron parte 97 de las 444 municipalidades de la provincia (414 de las cuales organizan elecciones), para los siguientes comicios de 2018 se espera la participación de entre 200 – 250 municipalidades para un censo total de 7 millones de potenciales votantes.

Además, la Dra. Goodman estima que hay un 85-90% de posibilidades de que la provincia de Alberta se sume al VER en las próximas elecciones (tienen ya el precedente del piloto no vinculante *"Jellybean"* [276]).

Para concluir y en lo que respecta al nivel federal, la llegada al poder del Partido Liberal en sustitución del Partido Conservador puede suponer un espaldarazo al VER en Canadá puesto que éste último ha sido tradicionalmente más reacio a su implantación. Aún así, la Dra. Goodman no estima probable la posibilidad de que se produzca algún avance significativo antes de 2019.

Conclusión

El caso de Canadá dentro del conjunto de países con experiencia en VER ofrece una curiosa dicotomía: se han emitido más de 2 millones de votos a través de sistemas de VER, lo que supone una de las cifras más altas a nivel mundial pero por otro lado, la falta de regulación estatal y provincial así como de estádares aplicables hace que las experiencias sean altamente atomizadas y sin coordinación entre ellas. De hecho, esta circunstancia es extrapolable al conjunto del ecosistema del VER a nivel global.

Además, la mejorable transparencia de los procesos de adjudicación unida a la escasa presencia de profesionales independientes e investigadores en las primeras fases, hace que el VER no alcance todo su potencial en Canadá pese a utilizarse en elecciones municipales con una cierta regularidad.

Por todo ello, y en línea con la Dra. Goodman, es necesario desarrollar una serie de estándares y regulaciones para canalizar todos los esfuerzos que se están realizando.

Además de abordar la cuestión de manera "*upside-down*", la solución más razonable para Canadá es el modelo "*bottom-up*" de tal manera que la creciente utilización local y poco coordinada de sistemas de VER presionen a niveles legislativos superiores para que asuman la necesidad de estandarizar y protocolizar el sector.

Con todo, la situación actual parece indicar que no va a desarrollarse en un futuro próximo (2-4 años) la legislación que permita esa introducción del VER protocolizada, gradual y con los requerimientos deseables de participación de expertos y otros *stakeholders*.

Para el lector interesado en profundizar en las experiencias previas de VER en Canadá, se seleccionan las siguientes referencias bibliográficas: [265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276].

3.2.4 Estados Unidos de América

La implantación y desarrollo del Voto Electrónico Remoto en Estados Unidos muestra similitudes con el caso de Canadá, en el sentido de que no existe un desarrollo legislativo a nivel nacional o estatal que sirva como base de aplicación a los pilotos que tienen lugar.

En el caso de pruebas piloto dentro del alcance del presente apartado 3.2 de experiencias de VER en elecciones VAP, su ámbito se cirscuncribe a los colectivos reconocidos en el UOCAVA (*Uniformed and Overseas Citizens Absentee Voting Act*) [294].

La escasa implantación de soluciones de VER en los Estados Unidos no quiere decir que no existan empresas o grupos de investigación de primer nivel dedicados a la materia. De hecho, varias de las soluciones más destacadas en el campo tienen sello americano: Helios, Remotegrity o la compañía Everyone Counts son ejemplos de actores relevantes de origen americano en el campo del VER.

Historia del VER en los Estados Unidos de América

El germen del Voto Electrónico Remoto en los Estados Unidos estuvo íntimamente relacionado con la necesidad de mejorar los procedimientos de voto tradicional para los habitantes que se encontraban fuera del país en el momento de las elecciones.

El primer paso que se dio en esa dirección fue el ya citado UOCAVA que entró en vigor en 1986. En él, se designaba al Secretario de Defensa como responsable de diseñar y administrar la forma en la que se implementaba un sistema que permitiese a las categorías de ciudadanos aludidos inscribirse para poder ejercer su derecho a voto desde el extranjero. Éste, a su vez, encargó al FVAP (*Federal Voting Assistance Program*) [302] el desarrollo y puesta en marcha del sistema.

No obstante, desde un principio y de una manera creciente a lo largo de la década de los 90, fueron apareciendo problemas derivados de la idiosincrasia en los distintos servicios de correos del mundo. Los votos debían recorrer numerosos territorios, cada uno con protocolos y ordenanzas propias antes de llegar al votante que los había solicitado y posteriormente realizar el trayecto inverso hasta llegar de nuevo a la autoridad estadounidense.

El hecho de que buena parte de los votos proviniesen de zonas donde estaban desplegados los militares norteamericanos y por tanto en conflicto, no hacía sino acentuar las dificultades logísticas, incrementando los retrasos y extravíos.

Por todo ello, el FVAP fue a lo largo de la década trabajando con distintos estados y condados con el objetivo de encontrar soluciones a los problemas logísticos descritos con anterioridad. Contemporáneamente, tuvo lugar la denominada *"burbuja punto com"* que en

el caso del VER cristalizó en el primer proyecto piloto en elecciones VAP en los EEUU: el *VOI Project* (Con anterioridad en el año 1996 el *Reform Party* de Ross Perot ya había utilizado el voto remoto).

VOI Project (*Voting Over Internet*) [305]

Se trata del primer proyecto que tuvo lugar para evaluar la posibilidad de implantar un sistema de voto remoto sobre internet para votantes UOCAVA. La entidad responsable de diseñarlo y probarlo fue el FVAP y se testó en las elecciones generales americanas del año 2000 como prueba de concepto.

Su alcance fue muy limitado, participando un total de 84 personas que emitieron 83 votos desde los estados y condados que se ofrecieron voluntarios para la prueba piloto: Carolina del Sur, los condados de Okaloosa y Orange (Florida), el condado de Dallas (Tejas) y el condado de Weber (Utah).

Pese a lo limitado de su alcance, el informe de evaluación del FVAP sobre la prueba de concepto indica que *"El proyecto piloto VOI es un estudio de viabilidad que ha demostrado que un sistema independiente para registro y voto remoto sobre internet (sic) puede ser una alternativa segura y viable al proceso por correo ordinario para entornos estrechamente controlados y a pequeña escala"* [305].

Y se concluye con las siguientes 4 recomendaciones:

1. *"Implementar un proyecto piloto a mayor escala para registro remoto y verificación de estado que esté integrado electrónicamente con sistemas existentes de registro de votantes para las elecciones generales de 2004 con un alcance de 1 a 3 estados.*
2. *Continuar con la participación en el desarrollo de estándares de registro y votación a través de internet.*
3. *Apoyar iniciativas legislativas estatales que permitan el registro y votación por internet.*
4. *Continuar investigando para identificar soluciones a cuestiones pendientes para permitir la eventual implementación y uso operacional de un sistema de registro y votación remoto".*

El balance positivo del VOI y las recomendaciones finales fueron la base del mandato del Congreso incluido en el *National Defense Authorization Act* (NDAA) de 2002 requiriendo al FVAP la realización de un proyecto de demostración de *"voto electrónico remoto".*

De esa manera, el FVAP inició la segunda experiencia piloto de VER en 2002 que se denominó proyecto SERVE (*Secure Electronic Registration and Voting Experiment*).

SERVE Project (Secure Electronic Registration and Voting Experiment)

La sección 1604 del NDAA para el año fiscal 2002 encargó al Secretario de Defensa llevar a cabo una *"demostración expandida"* del VOI en un proyecto denominado *SERVE* (*Secure Electronic Registration and Voting Experiment*).

El FVAP invitó a todos los estados interesados a participar. Finalmente 55 condados de los estados de Arkansas, Florida, Hawaii, Carolina del Norte, Carolina del Sur, Utah y Washington se adhirieron al proyecto.

Al igual que en el caso del *VOI, SERVE* era una prueba piloto dirigida a los ciudadanos UOCAVA de los condados participantes.

Desde un principio se trabajó con varios de los expertos que participaron en el proyecto VOI e invitaron a empresas privadas tales como Accenture y Verisign.

Se puso también especial énfasis en los protocolos de seguridad tanto a nivel físico como de medios materiales y personal: el servidor central se encontraba en un edificio propiedad de Accenture en Reston, Virginia con medidas excepcionales de seguridad. Además, los accesos estaban controlados de manera electrónica y hacían falta por lo menos dos personas autorizadas para leer los datos encriptados. A mayores, el FVAP envió a cada jurisdicción participante ordenadores dedicados exclusivamente al proyecto SERVE que únicamente podían desencriptar los datos de su demarcación etc.

El sistema se diseñó para manejar hasta 100.000 votos y podía ser expandido hasta un total de 6.000.000 de votos, cubriendo de esa forma la totalidad de ciudadanos UOCAVA con suficiente margen.

Para finales de 2003, SERVE estaba listo para su implantación en una votación real que debía producirse en el año 2004 (primarias y elecciones presidenciales de 2004).

No obstante, para asegurarse de que el sistema cumplía con los requerimientos en matería de tecnología y seguridad, se conformó un grupo de revisión denominado *Security Peer Review Group* (SPRG) formado por 10 miembros relevantes de los ámbitos de investigación e industria en segurdad informática y criptografía, algunos de ellos elegidos a propósito por sus posiciones contrarias a la implantación de sistemas de voto por internet.

Con ello se pretendía hacer un ejercicio de transparencia y someter a *SERVE* a un examen lo más riguroso posible.

El resultado de la revisión del SPRG fue que cuatro de los miembros (incluida la doctora Simons, expresidenta de la ACM) consideraron que *SERVE* no cumplía con los requisitos necesarios para garantizar la privacidad y la integridad de las elecciones y recomendaban su no puesta en marcha en un informe firmado por ellos [306]. Éste alcanzó una notable relevancia y llegó a la opinión pública a través del New York Times.

La consecuencia fue que dos semanas después, el día 6 de febrero de 2004, el Secretario Adjunto de Defensa Paul. D. Wolfowitz ordenó a David Chu, Subsecretario de Defensa que procediese a cancelar el proyecto *SERVE* antes incluso de haberse utilizado.

El FVAP comentó que la cancelación no se debió al desarrollo del *SERVE* en sí, sino a:

- La ausencia de estándares de evaluación y certificación
- La complejidad en la integración de los sistemas de gestión de elecciones (EMS) de cada estado
- La necesidad de más tiempo de estudio y preparación para poner de acuerdo a todas las partes involucradas en un proyecto de VER a escala nacional

llamando una vez más la atención sobre las principales motivaciones para la realización de la presente tesis [302].

Cabe destacar no obstante que la cancelación del proyecto *SERVE* no implicó que se aparcase el voto remoto en los EEUU. De hecho, el FVAP ha seguido teniendo asignada la tarea original de 2002 de implementar un proyecto de voto electrónico.

Desde 2004 hasta 2015 se ha seguido trabajando en distintas líneas (tanto en el voto desde ordenadores personales como desde quioscos de votación) hasta que finalmente en 2015 el NDAA retiró el requerimiento de desarrollar el proyecto de demostración al FVAP, poniendo punto y final al mandato de desarrollar un sistema de VER a nivel estatal [302].

Terminada la iniciativa a nivel nacional, el FVAP ha puesto a disposición de los estados o condados que quieran desarrollar (dentro de sus atribuciones legales) sus propias experiencias piloto de VER las recomendaciones y *expertise* adquiridas desde finales de los 90.

De entre las experiencias a nivel local o estatal encuadradas dentro del UOCAVA, destacan 2: Virginia Occidental y Washington D.C., ambas en 2010.

Piloto de Voto Electrónico Remoto en Virginia Occidental 2010 para UOCAVA

El UOCAVA de 1986 fue la primera iniciativa legislativa americana en la que se trataba de dar solución al voto de no residentes y militares desplegados en el exterior.

Con posterioridad, en 2002 y a consecuencia de los problemas e irregularidades detectados en el estado de Florida en las elecciones presidenciales del año 2000, se promulgó el *Help America Vote Act* o HAVA [307].

Entre sus objetivos principales, se buscaba mejorar las tecnologías de votación, eliminando los sistemas de tarjetas perforadas y palancas para sustituirlos por equipos electrónicos de votación tales como máquinas DRE (referirse al trabajo del Dr. Panizo en [4]).

Pese a ello, todavía hasta el año 2009 la mitad de los votos emitidos desde el extranjero no llegaban a tiempo de ser recontados [309]. Como consecuencia, en ese mismo año se promulgó el *Military and Overseas Voter Empowerment* (MOVE) Act. Desde entonces, se requiere que los estados remitan los votos a su censo UOCAVA con no menos de 45 días de antelación. Desde su entrada en vigor, el porcentaje de votos que no llegan a tiempo se ha reducido desde la mitad a un tercio.

Además, la implementación de MOVE ha sido desigual entre los distintos estados norteamericanos con 18 de ellos que únicamente permiten enviar los votos a sus UOCAVA por correo ordinario: Arkansas, Connecticut, Georgia, Illinois, Kentucky, Maryland, Michigan, Minnesota, New Hampshire, New York, Ohio, Pennsylvania, South Dakota, Tennessee, Vermont, Virginia, Wisconsin y Wyoming

En el caso de Virginia Occidental en 2010, 5 condados se adhirieron al programa piloto de VER para las primarias de ese año y un total de 8 para las elecciones generales. A mayores, otros 55 condados permitían a sus votantes imprimir el voto que recibían electrónicamente, elegir su opción y enviarlo por correo tradicional, fax o email (estas dos últimas opciones como ya se ha explicado en el apartado 2.1.1 no cumplen con los requisitos de privacidad para el votante).

Las empresas adjudicatarias fueron Scytl y Everyone Counts. En cuanto al procedimiento de votación, comienza con el votante solicitando la papeleta a través de la *Federal Post Card Application* (FPCA) o la *West Virginia Electronic Absentee Ballot Application* (WVEABA). Las autoridades del condado recibían las peticiones de los votantes que lo habían solicitado y las pasaban a las empresas adjudicatarias.

Posteriormente, las empresas enviaban a los votantes que lo habían solicitado un email con un identificador de usuario único y una URL. El votante accedía al link y debía introducir tanto el identificador único de usuario como información personal (fecha de nacimiento etc.).

Una vez elegida la opción, el votante obtenía un código con el que puede comprobar que su voto se ha recibido y contado correctamente (sin mostrar la opción que votó).

Los votos se almacenan encriptados con sistemas de 2048 bits y un acceso SSL a la aplicación en servidores redundantes para garantizar su funcionamiento. En la noche de las elecciones, se transladaba la información a un equipo aislado físicamente y sin conexión de ningún tipo para proceder al desencriptado de datos, en el que se rompía en vínculo votante-voto.

Para proceder a la disociación del vínculo, desencriptación y recuento era necesaria la colaboración de los Comisarios de Condado, cada uno de ellos depositario de una parte de la clave privada de las elecciones.

Debido a la muy reducida participación en el experimento piloto (ver tabla siguiente), no se publicó el reparto de votos por condado (en el caso del condado de Mason únicamente una persona votó a través de internet, por lo que publicarlo hubiese equivalido a revelar el voto del votante).

Los condados de Mason, Monroe y Putnam participaron únicamente en las elecciones generales del 2 de noviembre de 2010 mientras que los 5 restantes tomaron parte tanto en las primarias del 11 de mayo de 2010 como en las generales de noviembre.

La cifra total de votos emitidos en las primarias fue de 63 mientras que en las elecciones generales fueron 125.

Condado	*Votos emitidos*	*Compañía*
Jackson	10	Scytl
Kanawha	35	Everyone Counts
Marshall	9	Scytl
Mason	1	Scytl
Monongalia	22	Everyone Counts
Monroe	3	Everyone Counts
Putnam	15	Everyone Counts
Wood	30	Everyone Counts

Tabla 9: Votos emitidos en la experiencia piloto de VER en Virginia Occidental para las elecciones generales de 2010. Fuente: Elaboración propia a partir de [258] y [308].

En cuanto al código de las soluciones software utilizadas, no fue hecho público y tampoco se enviaron los procesos para revisión por parte del NIST o el *Cryptograpic Algorithm Validation Program* (CAVP), si bien no existe ningún requerimiento que lo demande.

Por tanto, se trató de un piloto muy limitado en su alcance (63 y 125 votos) y cuyo proceso de diseño, implementación y análisis se podría haber abordado de una forma más abierta y transparente para involucrar en lo posible a la comunidad científica y académica. En cuanto al piloto en sí, discurrió sin incidentes ni ataques conocidos y no hubo ninguna queja formal sobre el uso de sistemas de VER.

Por su parte, el NIST en su *Interagency Report 7770*, *"Security Considerations for Remote Electronic UOCAVA Voting"* de febrero de 2011 llamaba la atención sobre las debilidades de los sistemas de VER (malware en el ordenador del votante, redes de comunicación poco seguras, errores del votante). Posteriormente, en su *report* de mayo del 2012 recomendaba una mayor investigación antes de poder introducir el VER a nivel nacional [308].

En su conjunto, el piloto de Voto Electrónico Remoto en Virginia Occidental 2010 para UOCAVA se puede considerar una experiencia de razonable éxito a su escala.

Proyecto Piloto de VER en Washington, D.C. *"D.C. Digital Vote-by-Mail Service"* (DVBM) en septiembre de 2010

El segundo piloto destacado en el ámbito del VER para EVAP en los EEUU se produjo en Washington D.C. en 2010 y se dirigió nuevamente al colectivo de votantes UOCAVA.

En ese mismo año, el *"Washington D.C. Board of Election and Ethics"* (BOEE) inició un piloto con financiación estatal para desarrollar un sistema de VER para las elecciones generales de noviembre de 2010 denominado *"D.C. Digital Vote-by-Mail Service"* (DVBM).

A diferencia de la experiencia en Virginia Occidental, en el caso del DVBM se adoptó desde un primer momento un enfoque de código abierto, fomentando la transparencia y la participación de expertos en el campo. El socio tecnológico seleccionado por el BOEE para desarrollar el sistema fue la fundación sin ánimo de lucro *"Open Source Digital Voting"* (OSDV), hoy conocida como OSET (*Open Source Elections Technology*).

La OSDV incluyó la solución para Washington D.C. dentro de un proyecto que denominó *TrustTheVote Project*, desarrollado con el framework *Ruby on Rails*, almacenado en un *web server* Apache COTS y sobre la base de datos relacional MySQL.

La arquitectura de red se resume en la siguiente imagen:

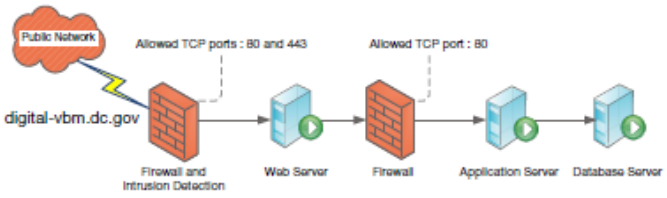


Figura 33: Arquitectura de red de la solución DVBM para Washington D.C. [36]

Las peticiones web HTTPS se interpretan por el servidor web a través del puerto TCP 443. Como el servidor web y el servidor de aplicación se ejecutan en equipos distintos, incluso si se atacase al servidor de aplicación, el atacante no tendría acceso a la clave privada HTTPS.

En cuanto a las papeletas electrónicas, en el DVBM eran archivos pdf que el votante podía rellenar con un *pdf reader* y que posteriormente eran almacenados en el servidor de datos. Éste encriptaba los votos completos con una clave pública cuya clave privada correspondiente estaba en posesión de los responsables de la votación y mantenida *offline*. Los votantes no podían votar más de una vez, comprometiendo pues la resistencia a la coerción.

El votante, para identificarse, recibía sus credenciales en los meses anteriores a la elección por correo postal: un código de identificación o *ID number*, el nombre completo del votante, el ZIP o código postal de residencia y un PIN de 16 caracteres hexadecimales.

En línea con lo comentado sobre el enfoque abierto, las autoridades planificaron un simulacro de elecciones días antes para probar el sistema en un entorno real, dando la oportunidad a quién quisiera de tratar de atacarlo [311], (Requisito PIA-7 de la metodología).

Entre los grupos que recogieron el guante, se encontraba el profesor Halderman y su equipo de la Universidad de Michigan.

En un paper publicado en febrero de 2012, se detallan las vulnerabilidades y cómo se atacó el sistema *TrustTheVote* [36]. En realidad, tuvieron lugar dos ataques distintos: uno sobre la aplicación web y otro sobre la infraestructura de red:

- En lo que respecta al ataque a la aplicación web, los investigadores encontraron un error en el código que procesaba las papeletas subidas al servidor de almacenamiento:

  Cuando un votante subía su voto, el servidor lo almacenaba utilizando un archivo temporal y encriptándolo mediante la ejecución del comando *gpg* con el nombre del archivo temporal como parámetro:

  gpg "/tmp/voto123.pdf"

  Los programadores habían utilizado dobles comillas en lugar de comillas simples, lo que permitía ataques del tipo *"Shell Injection"* a partir de la extensión del archivo subido (es decir, sustituyendo el *.pdf* del ejemplo por "." y un comando a ejecutar).

  El equipo de la Universidad de Michigan utilizó la vulnerabilidad para realizar ataques con los que consiguió la clave pública de encriptación de votos (permitiendo la sustitución de votos reales por votos arbitrarios), recuperar los votos reales (violando por tanto la privacidad de los votantes) e incluso obtener acceso a un archivo pdf de que contenía las credenciales de acceso de los votantes.

  Además y a modo de prueba de que habían hackeado el sistema, incluyeron la canción *"The Victors"* de la Universidad de Michigan en la página de confirmación de la recepción del voto. Aún así, las autoridades responsables de la votación tardaron dos días en enterarse del ataque y lo hicieron porque un votante les contactó preguntando cuál era la canción que sonaba en la pantalla de confirmación del voto.

- En cuanto al ataque a la infraestructura de red, lograron acceder a uno de los servidores con el password por defecto de administrador que encontraron en un manual del modelo en internet. Revisando los *logs* de acceso, descubrieron intentos de acceso fraudulento desde IPs de Irán, China, India y Nueva Jersey; recordando una vez más que el entorno de comunicaciones sobre internet es hostil en multitud de ocasiones.

Como consecuencia de los ataques, el BOEE decidió el 6 de octubre de 2010 cancelar tanto el simulacro de elecciones como la posibilidad de usar el DVBM en las elecciones de noviembre de 2010 [312]. Por tanto, al igual que en el caso del proyecto *SERVE*, el DVBM fue cancelado antes de ser utilizado en unas elecciones reales.

En la actualidad únicamente 5 estados permiten el uso de sistemas de VER en elecciones vinculantes en el ámbito político: Alabama, Alaska (subiendo el archivo en pdf y mandándolo por el sistema de VER), Arizona, Misuri y Dakota del Norte.

De todos ellos, únicamente Alaska lo ofrece a todos sus ciudadanos con derecho de sufragio, en contraposición a los otros cuatro estados, que lo ofrecen exclusivamente a sus ciudadanos UOCAVA (y en el caso de Misuri, únicamente a militares destacados en zonas de "fuego hostil").

En Arizona, se han hecho pruebas piloto al menos en 2008 y 2010, si bien no se ha encontrado información sobre el número de participantes o el sistema utilizado [301].

En Alaska, al solicitar la autorización para votar por internet, se le informa al votante de que está cediendo su derecho a la privacidad de su voto, por lo que no parece un sistema seguro [313]. Además, el votante tiene que subir su voto en formato pdf, tiff o jpg al servidor de la aplicación. Ello añade una serie de vulnerabilidades derivadas de la utilización de formatos de archivo que han sido objeto de ataques con anterioridad.

Por lo que respecta a Alabama, han implementado el sistema de Everyone Counts pero únicamente para los militares UOCAVA y se ha utilizado por primera vez en las Primarias de marzo de 2016. La repercusión mediática ha sido escasa, la participación de grupos de expertos e investigadores muy limitada o nula y todavía no hay disponible ningún tipo de informe sobre su desempeño.

En cuanto a Dakota del Norte y Misuri, no hay datos sobre los pilotos que se ha realizado. Únicamente se detalla en sus sitios web que existe la opción, si bien no existe ningún tipo de informe o instrucción adicional [314, 315].

Por todo ello, la implantación de sistemas de VER para elecciones VAP en los Estados Unidos es muy minoritaria y en las pocas iniciativas que se producen, la información, el proceso de desarrollo y su seguimiento son escasos.

Un aspecto adicional que dificulta la implantación del VER es la diversidad de legislaciones incluso en lo que respecta a los requisitos de identificación de los votantes, hasta el punto de no requerir ningún tipo de documento identificativo para votar:

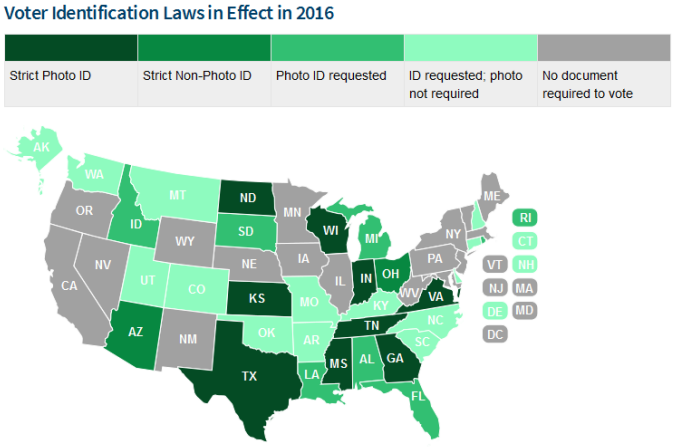


Figura 34: Requisitos de identificación de los votantes en los distintos estados de USA [309]

Para concluir con el presente capítulo es relevante destacar que, aparte de las tres experiencias presentadas en el presente punto, ha habido otras aplicaciones del VER en los Estados Unidos, si bien no en elecciones en el ámbito político sino para elegir al representante de un partido en las primarias de un determinado estado.

Tal es el caso del *Reform Party* en 1996, el Partido Demócrata en el año 2000 o el Partido Republicano en Utah en 2016. Todos estos casos, junto con otros como la iniciativa *"Empower LA"* se detallan en el apartado 3.3 de la presente tesis "Experiencias de Voto Electrónico Remoto en otros ámbitos".

Conclusión

Al igual que en Canadá, en los Estados Unidos existe una importante descentralización administrativa que otorga a cada estado o condado bastante libertad sobre los métodos de votación a emplear en elecciones dentro de su ámbito.

Ambos países comparten el hecho de que en la actualidad no existe ni una legislación a nivel nacional sobre el VER ni indicios de que se vaya a desarrollar en los próximos años.

En el caso de los EEUU, existen además grupos de presión relevantes tanto a favor como en contra de la implantación de una política nacional para el VER, si bien varios de los investigadores y expertos más relevantes en la materia se decantan por un enfoque más conservador [23, 301], ralentizando el avance del mismo.

En ese sentido, los precedentes de los proyectos piloto *VOI* y *SERVE*, así como los problemas detectados en la experiencia de Washington D.C. y la retirada de atribuciones al FVAP dibujan un panorama que no invita al optimismo en cuanto a un ritmo fluido de avance en la implantación del Voto Remoto Electrónico en los Estados Unidos.

El lector interesado en profundizar en el VER en los Estados Unidos puede referirse a la siguiente bibliografía: [23, 36, 258, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313].

3.2.5 Australia

Historia del VER en Australia

La experiencia de Australia en Voto Electrónico se remonta al año 2001, si bien no ha sido tan intensamente utilizado como en Canadá, Suiza o Estonia.

Nótese que en el párrafo anterior se ha hecho mención al Voto Electrónico pero no al Voto Electrónico Remoto. Ello se debe a que la mayor parte de las experiencias en Australia han sido en el ámbito del Voto Electrónico en entornos controlados y a través de máquinas o quioscos de voto. Para profundizar en dicha variante del Voto Electrónico, referirse al trabajo del Dr. Luis Panizo [4].

En lo que respecta al VER tal y como está definido esta tesis, las experiencias se reducen a un piloto muy reducido de militares en el extranjero en 2007, las elecciones estatales de Nueva Gales del Sur en marzo de 2011 y los mismos comicios en marzo de 2015.

En cuanto al primer piloto de noviembre de 2007, se seleccionó un conjunto de representantes de las Fuerzas Armadas Australianas (*Australian Defence Force)* desplegados en Afganistán, Irak, Timor Oriental y las Islas Salomón. El proyecto lo desarrollaron conjuntamente la Comisión Electoral Australiana y el Departamento de Defensa. La infraestructura utilizada fue la red militar *Defence Restricted Network* (DRN) y no la *World Wide Web*.

Despues de reunir todos los votos, se encriptaron y se mandaron de un servidor *Citrix* a la base de datos REV. En total había 2.012 efectivos registrados y finalmente se emitieron 1.511 votos con un coste de 521 dólares por voto. El proceso incluía imprimir los votos enviados electrónicamente para su recuento por las autoridades pertinentes.

Con posterioridad, en el año 2009 el *Comité Conjunto para Asuntos Electorales* recomendó no continuar con el VER para efectivos de defensa en el exterior debido a su elevado coste tanto económico como de recursos humanos. En consecuencia, se decidió volver al sistema de voto postal para las misiones exteriores australianas.

La segunda experiencia fue mucho más ambiciosa ya que involucró un número mucho mayor de votantes así como la utilización de redes de comunicación convencionales. La ocasión fueron las Elecciones Estatales de Nueva Gales del Sur de marzo de 2011.

Un año antes, el 16 de marzo de 2010, el Presidente de Nueva Gales del Sur Richard Torbay, anunció que el Comisario Electoral investigaría el "voto por internet" (sic) para mejorar el acceso de las personas con discapacidad visual de la provincia.

Se introdujo una enmienda al *Parliamentary Electories and Elections Act 1912* y se presentó el informe de viabilidad el 23 de julio de 2010. El 2 de septiembre fue aprobado y se reservaron fondos para el proyecto. Fue bautizado como *iVote*.

Desde un principio no se introdujo un límite de duración al proyecto, por lo que más que de un piloto, el VER en Nueva Gales del Sur constituye una forma más de votación aceptada de manera indefinida mientras no sea revocada por el Comisario Electoral.

En lo que respecta a los votantes con derecho a usar *iVote*, en un principio se pensó como un sistema únicamente para personas con discapacidad visual, si bien finalmente se decidió ampliar su uso a los siguientes casos [291]:

1. El votante se encuentra a más de 20 kilómetros del colegio electoral más cercano.
2. El votante tiene una discapacidad visual tal (sic), o una incapacidad física tal (sic) o es tan analfabeto (sic) que no puede votar sin asistencia.
3. El votante tiene una discapacidad (de acuerdo al *Anti-Discrimination Act 1977*) que le dificulta votar en un colegio electoral o sin asistencia.
4. El votante no va a estar en Nueva Gales del Sur durante las horas de votación en el día de las elecciones.

Los cuatro puntos precedentes se han mantenido para las elecciones estatales de marzo de 2015. (En Australia votar es obligatorio y no ejercer el derecho conlleva una sanción económica).

El sistema *iVote* no incluye únicamente el VER sino también el voto por teléfono, estudiado en el apartado 2.1.1 de la presente tesis y desaconsejado por su insuficiente nivel de verificabilidad y privacidad.

En cuanto a la opción de VER de *iVote*, en 2011 fue desarrollada por la empresa americana Everyone Counts. En cambio, para 2015 la empresa adjudicataria fue la española Scytl, partiendo de la base del sistema previo para desarrollar su versión mejorada de *iVote*.

El sistema originalmente diseñado para 2011 requería de un preregistro por parte del votante con derecho a usar el VER por pertenecer a alguno de los 4 criterios de utilización detallados previamente.

El votante podía preregistrarse por teléfono o bien on-line. En ambos casos, seleccionaba un código de 6 dígitos y posteriormente recibía en la dirección postal indicada un código ID de votante de 8 dígitos.

Para votar, el votante introducía los dos códigos (el de 6 dígitos seleccionado por él mismo y el de 8 dígitos enviado a su domicilio) y, tras haber votado, recibía un número a modo de recibo para comprobar que su voto había sido incluído en el recuento (pero sin poder verificar la opción votada).

El período de votación abarcó los 12 días anteriores a la votación, con un total de 51.103 votantes preregistrados. De ellos, 44.605 enviaron su voto a través del VER y 2.259 por teléfono.

En cuanto al código fuente (mayormente cerrado), la propiedad intelectual pertenece a Everyone Counts aunque algunos sistemas accesorios son propiedad del *New South Wales Electoral Commitee*.

En total se produjeron 10 quejas de entre los 46.864 votantes que utilizaron *iVote* en cualquiera de sus dos opciones (VER o teléfono).

Elecciones Generales Estatales de Nueva Gales del Sur de marzo de 2015

En marzo de 2015 tuvieron lugar las Elecciones Generales Estatales de Nueva Gales del Sur, en las que se elegían los 93 escaños de la Cámara Baja así como 21 de los 42 escaños de la Cámara Alta del estado.

Al igual que en 2011, se ofreció a los mismos 4 colectivos la posibilidad de usar el sistema *iVote* para ejercer su derecho a voto en cualquiera de las dos variantes: Voto Electrónico Remoto o voto telefónico.

En estas elecciones, el NSWEC decidió cambiar de proveedor del sistema de VER, pasando a ser la compañía Scytl.

La base del sistema *iVote* así como su funcionamiento debía ser análogo pero se trató de mejorar el mismo, pasando a realizarse la encriptación del voto en el terminal del votante para garantizar los requisitos *cast-as-intended* y *recorded-as-cast* (punto 2.2.2 para más detalles).

El sistema *iVote* 2015

Existen 3 modos de votar: por teléfono (marcación por tonos), VER y voto electrónico en un entorno controlado por la NSWEC.

En cuanto al sistema desarrollado, se adapta a un pliego de condiciones del NSWEC *(New South Wales Electoral Commitee)* en el que se considera que el riesgo de coerción se considera bajo y por tanto la resistencia a la coerción NO es una propiedad indispensable [292].

La autoridad australiana NSWEC y Scytl publicaron un paper en el que se detalla el funcionamiento de *iVote* [288].

Las fases de votación son 4: registro, emisión verificación de propiedades de E2Ev y verificación de voto desencriptado:

1. Registro del votante: De manera análoga al caso de 2011, el votante se puede registrar online o por teléfono, seleccionando un código PIN de su elección. Posteriormente, recibe su número de *iVote* por otro canal (email, SMS o llamada telefónica). El PIN elegido por el votante y el número de *iVote* son sus credenciales para votar. El período de registro comprendió desde el 12 de febrero de 2015 hasta el 28 de marzo de 2015.

2. Emisión del voto: El votante se identifica con su PIN personal y su número de *iVote* a través con un ordenador personal, smartphone o teléfono con capacidad de marcación por tonos. Cuando se emite el voto, éste se encripta junto con un número de recibo aleatorio único de 10 dígitos en el equipo del votante o bien en los servidores de voto por teléfono del NSWEC. Tras haber sido emitido, se le envía al votante el número de recibo de 10 dígitos para realizar las verificaciones posteriores.
3. Verificación de que el voto ha sido emitido como estaba previsto y se guardado como se ha emitido. En inglés: *cast-as-intended* y *recorded-as-cast*. La verificación se articula llamando al Servidor de Verificación: se le solicita al votante su PIN, número de *iVote* y de recibo aleatorio único. Si la identificación es correcta, un sistema de *text-to-speech* comunica el voto.

   Este sistema de verificación no cumple con la resistencia a la coerción, pero como se comentó anteriormente, dicho requisito no es obligatorio para las autoridades de Nueva Gales del Sur.
4. Verificación de voto desencriptado: Una vez concluidas las elecciones, todos los números de recibo aleatorios son subidos a un sitio denominado *"Receipt Number Website"* donde el votante puede comprobar que su número está incluido..

De una manera más gráfica, las 4 fases del proceso de votación con *iVote* se reproducen de la siguiente manera:

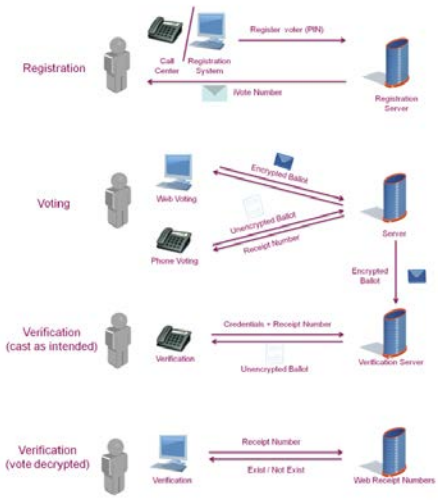


Figura 35: Fases de VER con *iVote* [288]

El criptosistema utilizado es la variante *multi-ciphertext* de ElGamal IND-CPA (2.2.4.6a).

Las Elecciones Estatales Generales de Nueva Gales del Sur de marzo de 2015 suponen los comicios vinculantes en el ámbito político con VER que mayor número de votos han gestionado con un total de **283.669**. De ellos, un 1.7% fueron verificados.

Un 91% de los usuarios de *iVote* (257.730 votantes) utilizó la herramienta debido a que se encontraba fuera de Nueva Gales del Sur en el momento de las elecciones, un 2% (4.818) eran votantes con discapacidad visual, un 4% tenía otra discapacidad (12.714) y un 3% (8.407 votantes) no tenía ningún colegio electoral a menos de 20 kilómetros [291].

En cuanto a la modalidad de *iVote*, 280.573 utilizaron el sistema de VER (98.9%) y 3.096 (1.1%) se decantaron por el sistema telefónico.

Aunque en las elecciones locales de Ontario de 2014 el número total de votos VER fue superior, su uso en Canadá no está coordinado por ninguna provincia ni por el gobierno a nivel estatal. Se trata de la unión de numerosas municipalidades que deciden implantar el VER para sus comicios locales de manera individual, con sus propios requisitos y criterios de selección. Por ello, no se puede considerar el conjunto de votos emitidos en las elecciones de Ontario como un todo homogéneo directamente comparable con las elecciones estatales de Nueva Gales del Sur

Respecto al coste, el Gobierno de Nueva Gales del Sur tuvo de provisionar 2,6 millones de dólares australianos extra debido a que el presupuesto original preveía únicamente 100.000 votos emitidos a través de *iVote*. Como finalmente fueron más de 283.000 y debido a la modalidad de pago de una tarifa fija por voto emitido, se tuvo que provisionar la citada cantidad para cubrir la diferencia [293].

A mayores, se asignaron 1,7 millones de dólares más para mejoras en el *"software de soporte"*.

Una vez explicado el sistema *iVote* y su impacto en las elecciones de marzo de 2015, conviene repasar una serie de características y eventos relacionados con el mismo:

iVote 2015 ha sido desarrollado partiendo de *iVote* 2011 con encriptación del tipo simétrico autenticado. Ello se debe a que, por la tipología de voto de Nueva Gales del Sur, en ocasiones se tienen que elegir y ordenar centenares de candidatos en un solo voto. En consecuencia, un sistema de encriptación asimétrico puro sería muy poco eficiente y no podría realizarse en el dispositivo del votante en un tiempo razonable.

Por tanto, no se pueden re-aleatorizar los votos encriptados de la manera más habitual en las o mix-nets. En consecuencia, la mix-net utilizada no es verificable e *iVote* es un sistema de VER que no cumple con la condición de verificabilidad universal. Para el recuento se utilizó AES [429] + ElGamal [64].

En lo referente a los contratiempos durante las elecciones, el profesor Halderman (observador de las elecciones estonias de 2013 y co-firmante de un paper sobre las debilidades de su sistema de VER [236]) y V. Teague publicaron un trabajo unos pocos días antes de las elecciones, apuntado a una vulnerabilidad de *iVote* 2015. En concreto, demuestran que *iVote* es vulnerable al ataque FREAK [26] (punto 2.4.2.3a de la tesis).

El *bug* hallado deriva del uso en la página de *iVote* de un Javascript de una herramienta de análisis llamada Piwik cuya configuración SSL era deficiente. Llegado el caso, se podría llegar a montar un ataque del tipo *man-in-the-middle* y modificar votos. Para cuando se subsanó la vulnerabilidad en el sistema, se había emitido más de 66.000 votos con *iVote*.

Los desarrolladores de *iVote* defienden que el sistema es seguro y que la vulnerabilidad del javascript de Piwik venía heredada del sistema *iVote* 2011 desarrollado por otra compañía. Ponen en duda también el objetivo último del paper puesto que, pese a conocer los autores la vulnerabilidad con anterioridad, no la comunicaron a las autoridades australianas hasta las 2 de la tarde del 20 de marzo y después de previamente ponerlo a disposición de los medios de comunicación.

Por último, argumentan que se introdujo el sistema de verificación telefónico para luchar contra posibles ataques *man-in-the-middle*. Además, el hecho de que no hubiera incidencias ni quejas por parte de los usuarios unido a que el patrón de votación fue muy similar al de 2011, lleva a pensar que no se produjo ningún ataque efectivo.

Conclusión

Las Elecciones Generales Estatales de Nueva Gales del Sur del 28 de marzo de 2015 han supuesto los comicios con un mayor número de votos emitidos a través de un sistema de VER (280.000).

El sistema actual deriva del *iVote* original desarrollado en 2011 por Everyone Counts. Actualmente la empresa desarrolladora es Scytl y al desarrollar la versión de 2015 a partir de la de 2011, se han arrastrado algunas debilidades en el diseño que dieron posteriormente pie a vulnerabilidades [287].

En cuanto a la coerción, las autoridades de Nueva Gales del Sur decidieron no incluirla como una amenaza real entre los requisitos, por lo que el sistema no es resistente a la coerción.

Al tratarse de EVAP, sería prudente que las autoridades valorasen aumentar el nivel de exigencia actual en cuanto a verificabilidad universal y privacidad.

Se recomienda al lector interesado en profundizar en el VER en Australia la siguiente bibliografía: [258, 286, 287, 288, 289, 290, 291, 292, 293, 369, 430].

3.2.6 Suiza

Suiza es un país con una relevante tradición democrática, caracterizado por un notable nivel de descentralización y una serie de particularidades que lo hacen único.

Entre ellas, posiblemente la más conocida es la implementación de prácticas de democracia directa de forma paralela a la democracia representativa. Ésta se articula a través de referendos (obligatorios u optativos) en los tres niveles estatales (federal, cantonal y comunal) e iniciativas populares que reúnan un mínimo de firmas. En la práctica, ello implica que los ciudadanos suizos participan en una media de 3 o 4 procesos de votación anualmente.

En lo que respecta al VER, hasta noviembre de 2016 se calcula que se han realizado más de 200 votaciones y referendos. Cabe también resaltar que no todas ellas han sido grandes proyectos. En muchas han participado únicamente unos cuantos cientos de votantes.

La gran diferencia de Suiza con respecto a otros países con un elevado número de elecciones con VER como Canadá, es que en el caso suizo sí que existe una coordinación a nivel federal (nacional) encargada de velar por una uniformidad en todo el proceso de introducción gradual del VER.

En Suiza, por la propia frecuencia de las elecciones, así como por la orografía y la distribución de la población, el voto por correo tradicionalmente ha representado más del 95% del total de los sufragios [283]. A pesar de ello, hasta 1992 los suizos residentes en el extranjero tenían que volver al país para votar. Desde ese año, se les envían las papeletas por correo postal una semana antes que al resto de los ciudadanos.

Sirva como ejemplo de la descentralización de Suiza el hecho de que entre el primer cantón en adoptar el voto por correo (Basilea en 1978) y los últimos (Valais y Ticino en 2005) han pasado más de 25 años. En la actualidad 14 cantones están experimentando con el VER y es posible que su adopción por parte de todos ellos lleve todavía varios años.

Historia del VER en Suiza

Sus orígenes se remontan al año 2000, cuando el gobierno federal invitó a los cantones interesados a desarrollar un sistema de VER con su apoyo. Los 3 cantones que se presentaron voluntariamente fueron Ginebra, Neuchâtel y Zurich, procediendo a diseñar e implementar cada uno su propio sistema con ayuda financiera del gobierno federal. A cambio, se comprometían a compartirlo con el resto del país cuando fuese finalmente aprobado por el gobierno. Se convirtieron así en los cantones piloto del VER en Suiza.

Las soluciones de Ginebra y Zurich son muy similares en su funcionamiento: el envío previo de una información única y aleatoria por correo postal tipo código PIN que sirve

al votante para identificarse, junto con alguna clave o información secreta elegida libremente por él y que permiten votar una única vez. Una importante diferencia reside en que, en el caso de Ginebra, el propietario del sistema de VER es el propio cantón mientras que en el segundo caso se subcontrató a una tercera empresa (Unisys).

El caso de Neuchâtel es diverso puesto que se ha desarrollado un portal más completo de e-gobierno en el que una de las acciones que se puede realizar es votar. La empresa responsable en este caso es Scytl.

La primera experiencia de VER en Suiza tuvo lugar en el municipio de Anières (cantón de Ginebra) del 7 a 18 de enero de 2003. De 740 personas que votaron en el referendo comunal, el 44% o 326 lo hicieron a través de internet [284].

Posteriormente, en 2004 se utilizó por primera vez un sistema de VER a nivel federal también en Ginebra, uniéndose un año después Zurich y Neuchâtel.

La utilización del VER no ha estado, en cualquier caso, exenta de vicisitudes: de 2005 a 2008 su utilización fue interrumpida en Ginebra y desde 2011 hasta la actualidad, Zurich ha decidido suspender su uso por cuestiones técnicas y de seguridad.

En cuanto al perfil de votante al que iban dirigidos los pilotos, hasta 2007 se restringía únicamente a residentes en Suiza (mayoritariamente ciudadanos suizos, si bien en los referendos comunales de Ginebra se permite votar a extranjeros).

En 2008, Neuchâtel fue el primer cantón en ofrecer el VER a sus ciudadanos expatriados en otros países. Se trató de una decisión lógica puesto que en ocasiones se producían problemas en el envío y manejo de las papeletas al extranjero y el votante suizo no residente no podía ejercer su derecho. En 2009, se sumó el cantón de Ginebra y en 2010 Zurich, poco antes de cancelar su programa en 2011.

También en 2009 se adhirió el primer cantón no piloto al uso del VER para sus no residentes: Basilea (BS). En los siguientes 2 años, se sumaron los siquientes cantones también para sus no residentes: Friburgo (FR), Grisones (GR), St. Gallen (SG), Soleura (SO), Argovia (AG), Lucerna (LU), Escafusa (SH), Turgovia (TG) y Berna (BE).

Por motivos de eficiencia económica, en vez de desarrollar cada uno un nuevo sistema de VER propio, se adhirieron a alguno ya existente. Basilea, Lucerna y Berna decidieron asumir el sistema de Ginebra y los restantes 7 se decantaron por el modelo de Zurich.

En 2011, cuando Zurich decidió cancelar su proyecto, el cantón de Argovia tomó el relevo como coordinador, posición que mantiene actualmente.

Hasta la fecha, ningún otro cantón ha decidido adherirse al modelo de Neuchâtel, probablemente debido a su distinto enfoque, más ambicioso en el sentido de la introducción de una estructura de e-gobierno completa.

A modo de resumen, en la siguiente tabla se muestran las votaciones federales en las que han participado los expatriados suizos desde 2008 a 2014, especificando cada cantón así como el sistema de VER utilizado:

	NE	GE	ZH	BS	FR	GR	SG	SO	AG	LU	SH	TG	BE
01.06.2008	✔												
30.11.2008	✔												
08.02.2009	✔												
17.05.2009	✔												
27.09.2009	✔	✔											
29.11.2009	✔	✔		✔									
07.03.2010	✔	✔		✔									
26.09.2010	✔	✔	✔	✔	(✔)	(✔)	✔	(✔)					
28.11.2010	✔	✔	✔	✔	(✔)	(✔)	✔	✔	✔	✔	✔	✔	
13.02.2011	✔	✔	✔	✔	(✔)	(✔)	✔	✔	✔	✔	✔	✔	
23.10.2011				✔		✔	✔		✔				
11.03.2012	✔	✔		✔	(✔)	✔	✔	✔	✔	✔	✔	✔	(✔)
17.06.2012	✔	✔		✔	(✔)	✔	✔	✔	✔	✔	✔	✔	✔
23.09.2012	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
25.11.2012	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
03.03.2013	✔	✔		✔	✔	✔	✔		✔	✔	✔	✔	✔
09.06.2013	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	(✔)
22.09.2013	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔
24.11.2013	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	(✔)
09.02.2014	✔	✔		✔	✔		✔	✔	✔	✔	✔	✔	✔
18.05.2014	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔

Tabla 10: VER para expatriados suizos 2008 – 2014 [281]

Con posterioridad, el cantón de Glaris también se ha añadido al consorcio del sistema de Zurich, gestionado por Argovia.

En cuanto al volumen total de sufragios, de 2004 a 2012 se gestionaron más de 258.000 votos con sistemas de VER en 115 votaciones y referendos tanto a nivel federal como cantonal y local. Posteriormente, entre los años 2013 y 2014 se gestionaron más de 100.000 votos en más de 70 nuevas experiencias [278].

Una dificultad propia del VER en Suiza proviene de la descentralización administrativa del país y concierne a la disponibilidad y uniformidad de los datos disponibles. Cada cantón tiene un grado elevado de libertad sobre la categorización en las experiencias de VER, por lo que obtener información detallada y desagregada en ocasiones no es posible [281].

Reforma de diciembre de 2013: establecimiento de un sistema coordinado, progresivo y sujeto a pruebas objetivas

En diciembre de 2013, tras el aumento de los cantones que ofrecían el VER, el gobierno federal aprueba una revisión de la ley que regula los derechos políticos (VPR, SR 161.11) y que entra en vigor el 15 de enero de 2014.

Se articula sobre 4 principios [283]:

1. La seguridad primero: expansión gradual.
2. Orientado al pacto: nivel confederal y nivel cantonal trabajan juntos.
3. Requerimientos reforzados de seguridad: verificabilidad y auditoría.
4. Autoridades aprobadoras: Consejo Federal y Cancillería Federal.

En cuanto a los requerimientos y su implicación sobre el porcentaje de VER permitido en función de su cumplimiento, Barbara Perriard, Jefa de la Sección de Derechos Políticos de la Cancillería Federal Suiza, presenta la siguiente tabla en [283]:

Conformity with new security requirements per canton	Limits	
	Cantonal electorate	Federal electorate
First generation systems | uncertified second generation systems	30%	10%
Individual verifiability and certification	50%	30%
Complete verifiability (individual plus universal verifiability) and certification	100%	100%

Tabla 11: Requerimientos de implantación del VER en Suiza y porcentaje máximo permitido [283]

A los sistemas de VER anteriores a la reforma de diciembre de 2013 todavía en uso, se les ha otorgado una licencia por 2 años a partir del 14 de diciembre de 2015. Además, la Cancillería Federal evalúa cada sistema de nuevo para cada votación.

Se basan para la evaluación en el *Common Criteria Protection Profile* para *Online Voting Products REF-BSI* y en el estándar *Common Criteria OWASP Top 10*. Además, los sistemas de VER deben incluir un *risk assessment* que explique cómo se implementan los requerimientos.

Adicionalmente, en los niveles superiores (2 y 3) ni el dispositivo del votante ni el canal de transmisión se presupone fiable. En el nivel intermedio (nivel 2), el servidor encargado de proveer una prueba de contenido del voto (como los códigos de retorno en el modelo noruego) es confiable mientras que en el nivel 3 el servidor no es fiable ni para la prueba de contenido ni para almacenar el voto. Por último, en los casos 2 y 3, las entidades evaluadoras son autoridades acreditadas mientras que en el nivel 1 es la propia Cancillería.

La estrategia de introducción y desarrollo del VER como un proceso coordinado, gradual y sujeto a pruebas objetivas evaluadas por terceras partes especializadas como en Suiza [283], supone posiblemente la iniciativa más interesante, prudente y realista de implantación segura del voto por internet en opinión del autor de la presente tesis.

Elecciones Parlamentarias Federales de octubre de 2015

El último ejemplo relevante de VER en Suiza han sido las Elecciones Parlamentarias que tuvieron lugar el 18 de octubre de 2015.

En ellas, 13 cantones solicitaron permiso para utilizar sus sistemas de VER (Berna no lo solicitó). De ellos, a 9 no les fue concedida la autorización por "*fallos de seguridad que afectaban al secreto del voto*".

Los 4 que pudieron proceder fueron Ginebra, Basilea, Lucerna y Neuchâtel. De ellos, los 3 primeros pertenecen al sistema de Ginebra y Neuchâtel pertenece al de Scytl.

En total, unos 34.000 ciudadanos suizos no residentes así como 96.000 residentes en los cantones de Ginebra y Neuchâtel pudieron ejercer su voto a través del sistema de VER entre los días 28 de septiembre hasta el 17 de octubre (en el caso de Neuchâtel desde el 21 de septiembre). En concreto, de los 132.134 votantes con derecho a utilizar el VER, 13.370 decidieron emitir su voto de esa manera.

Recientemente, la OSCE ha emitido un informe sobre las elecciones parlamentarias federales de Suiza del 18 de octubre de 2015 en las que alerta sobre algunas carencias de los sistemas de VER en lo que respecta a su resistencia a la coerción (RC) [285] y a la no creación de las claves criptográficas en un entorno público (contraviniendo recomendaciones previas de la misma OSCE).

En el mismo informe, se evalúa positivamente el control de acceso a infraestructuras críticas y no se reportan intentos de ataques DDos o malware. También se valora satisfactoriamente la transparencia en las operaciones de desencriptación y recuento.

En cuanto a la publicación del código fuente, no es un requisito federal pero aún así el sistema de Ginebra permite su revisión en dependencias gubernamentales a cualquier ciudadano que lo solicite. En el caso de Neuchâtel, el código no ha sido hecho público.

En líneas generales, el informe de la OSCE valora positivamente los dos sistemas de VER desplegados en Suiza (el de Ginebra y el de Neuchâtel), si bien aboga por una mayor transparencia, auditoría e interoperatibidad entre sistemas.

Por lo que respecta a los 9 cantones a los que les fue denegado el permiso, todos ellos pertenen al sistema de VER vinculado originalmente al cantón de Zurich hasta su baja y desde entonces gestionado por Argovia.

Dicha solución de VER es propiedad y está gestionada por la empresa americana Unisys. Algunos políticos suizos tales como el parlamentario socialista Jean-Christophe Schwaab han expresado su preocupación al considerar que *"Es bien conocido que las empresas americanas instalan backdoors en su software de tal manera que la NSA (National Security Agency) y otras agencias gubernamentales pueden tener acceso a los datos. El secreto del voto no debería ponerse a merced de agencias de inteligencia extranjeras"* [277].

Actualemente, el cantón de Zurich (y los 8 adicionales), ha perdido su autorización y por tnto no puede utilizar el VER. Por su parte, Neuchâtel y Ginebra han sido certificados y están preparando la calificación al nivel 2. También el cantón de Friburgo ha obtenido la certificación de nivel 1 con un nuevo sistema desarrollado por Swiss Post y Scytl. A futuro, Swiss Post ha declarado que va a tratar de obtener el nivel 2 en 2017 y Ginebra se ha marcado como objetivo el nivel más ambicioso (nivel 3) para finales de 2018 [430].

Conclusión

Suiza, por su descentralización e idiosincrasia, supone un caso de estudio muy valioso del VER. Sus ciudadanos participan de media en un total de 3-4 votaciones al año.

El VER en Suiza data del 2000, cuando se dieron los primeros pasos de cara a evaluar su utilización. Desde entonces, se ha mantenido el enfoque de coordinación rigurosa por parte del gobierno federal en los distintos pilotos que han ido teniendo lugar.

En la actualidad, de los 26 cantones suizos, 14 de ellos ofrecen uno de los tres sistemas de VER, correspondientes a los tres primeros cantones que decidieron participar en las primeras pruebas piloto de VER a principios de siglo (Ginebra, Neuchâtel y Zurich). En 2011, Zurich abandonó el proyecto de VER y su papel coordinador en su *"consorcio"* lo asumió Argovia.

A finales de 2013, el Gobierno Federal introdujo una serie de modificaciones legislativas encaminadas a reforzar las medidas de seguridad. De hecho, la idea principal es *"Security First"* por lo que existen una serie de niveles máximos de uso de VER ligados a unos criterios objetivos de desempeño del sistema.

En ese sentido, en las últimas elecciones parlamentarias federales, el sistema de VER de Zurich no fue autorizado a ser utilizado por lo que los electores de los 9 cantones asociados no votaron de esa manera.

En total, se han producido en Suiza más de 200 votaciones vinculantes para un total de más de 260.000 votos emitidos utilizando sistemas de VER. Ello conforma uno de los mejores bancos de pruebas y una fuente de información muy valiosa en entornos reales.

En el presente 2016, Neuchâtel, Ginebra y Friburgo han sido certificados. Éste último lo consiguió tras presentar un nuevo sistema de VER desarrollado por Swiss Post y Scytl.

En resumen, la estrategia suiza de establecer una serie de requisitos de seguridad exigentes y unas pruebas de acreditación objetivas en función de las cuáles se decide el nivel de utilización de al que está limitado el sistema de VER supone la mejor de las posibilidades entre las analizadas en la presente disertación. El sistema suizo es, por filosofía, diseño, rigurosidad y progresividad, una referencia dentro del conjunto de políticas e implementaciones del VER a nivel mundial.

Se invita al lector interesado en profundizar en las experiencias de Voto Electrónico Remoto en Suiza comenzar con las siguientes referencias: [258, 277, 278, 279, 280, 281, 282, 283, 284, 285, 369, 430].

3.2.7 Otras experiencias destacadas (Francia, Finlandia y Nueva Zelanda)

En los subapartados 3.2.1 – 3.2.6 se han repasado los casos más relevantes de VER por su enfoque, tamaño, compromiso, duración o aportación al campo. Para complementarlo, en el presente punto se repasan someramente las experiencias en otros países que han realizado estudios pormenorizados de implentación o incluso han llegado a realizar algún tipo de prueba piloto.

Francia

Las primeras experiencias de Francia con el voto electrónico se remontan a 2001 y 2002 en la modalidad de puestos o kioscos de voto. El primer piloto en unos comicios reales tuvo lugar en 2003 en las elecciones para la Asamblea de Ciudadanos Franceses residentes en el Extranjero (AFE). En concreto los nacionales franceses residentes en los Estados Unidos tuvieron la oportunidad de votar a sus representantes a través de internet. En la actualidad, la utilización del VER en Francia se centra todavía mayoritariamente en los ciudadanos de dicha Asamblea.

Posteriormente hubo experiencias similares en 2006, 2009 y 2012. El total de votantes susceptibles de tomar parte en los pilotos fue de 50.000 en 2003 (residentes en los EEUU y Canadá), 525.000 en 2006 (residentes en Europa, Asia y Oriente Medio), 340.000 en 2009 (residentes en África, Norteamérica y Sudamérica) y 1.5 millones en 2012 (todos los franceses residentes en el extranjero).

En los tres primeros casos, no se conocen detalles técnicos de los sitemas de VER. Únicamente que debían cumplir con las recomendaciones de la CNIL (*Commission Nationale Informatique et Liberté*). Ya en las elecciones de 2012, la solución implementada fue analizada por la Agencia de Seguridad Francesa y obtuvo la homologación RGS necesaria para poder gestionar los comicios.

Por lo que respecta a las compañías desarrolladoras del sistema, en 2003 fue Election Europe, anteriormente conocida como Election.com. En 2006 fue EDAS y en 2009 Scytl (si bien Athos Origin gestionó las elecciones), que repitió en 2012.

De hecho, el Ministerio de Asuntos Exteriores francés decidió adquirir una "licencia permanente de voto por internet" a la compañía Scytl en 2009, tras comprobar los buenos resultados de participación y del piloto realizado en ese mismo año. Por ello, es de esperar que los siguientes procesos electorales gestionados por el citado Ministerio sigan utilizando la tecnología de Scytl.

En cuanto al número de votos emitidos a través del sistema de VER, en 2003 fueron 4.384, en 2006 10.201, en 2009 6.091 y en 2012 240.000, representando el 55% del total

del voto desde el extranjero (la primera vez que el VER suponía la mayoría de sufragios enviados) [320].

En todos los casos, el código fuente no era abierto, si bien en 2006 y 2009 se exigía el acceso por parte de un experto independiente para verificar la confidencialidad, seguridad y precisión del sistema, como apuntan Barrat et al. [258]. Ese experto debía redactar un informe con sus conclusiones que debía ser enviado al Ministerio de Asuntos Exteriores y a la Comisión Electoral. Dicho informe no ha sido hecho público total ni parcialmente.

En cuanto a las elecciones parlamentarias de 2012, pese a que no hubo constancia de ningún ataque a gran escala, sí que se produjeron quejas de falta de transparencia y seguridad por parte del Partido Pirata [316] así como fallos de seguridad en lo que respecta a la versión de Java soportada, lo que llevó a que Firefox lo bloqueara [319].

En los años 2013 y 2014, se volvió a dar la posibilidad a los votantes para la Asamblea de los Franceses en el Extranjero (AFE) de emitir su voto a través del sistema desarrollado por la misma compañía, mejorando el porcentaje del año 2012 del 55% (65% en 2013 y 73% en 2014), consolidando la opción del VER en esas elecciones.

Adicionalmente, en las elecciones locales de París de 2013, se pudo votar a través de VER, si bien no existe bibliografía técnica para poder analizar la solución planteada.

Además, hubo una serie de polémicas debidas a que un periodista pudo emitir hasta 5 votos, en uno de ellos indentificándose como el ex-presidente Nicolas Sarkozy [317, 318]. Se detectó que uno de los fallos del sistema consistía en que para identificarse bastaba con realizar un pago de 3 € con una tarjeta de crédito e identificarse con el nombre y la dirección de un residente en París que estuviese en el censo. Por todo ello, el ex-primer ministro François Fillon, que perdió las elecciones por un estrecho margen, acusó al vencedor Jean-François Copé de un "fraude a escala industrial".

En resumen, Francia ha dado una serie de pasos para introducir el VER desde principios de la década pasada, utilizándose en la actualidad de manera periódica un sistema desarrollado por Scytl para las elecciones a la AFE. A mayores se han implementado otras soluciones en elecciones puntuales de carácter local sin coordinación entre ellas.

Si bien no se ha reportado ningún ataque a gran escala, es cierto que una mayor transparencia en todo el proceso de diseño e implementación de los sistemas de VER desplegados contribuirían a una mayor implantación y una mejora en la percepción popular del VER en un país con una tradición tan destacada en prácticas democráticas como Francia.

Un papel más activo del gobierno central como líder y aglutinador; estableciendo un marco legal, un acceso transparente a la información y unas pautas comunes supondrían un espaldarazo muy importante al Voto Electrónico Remoto en el país galo.

Para el lector interesado en profundizar en el VER en Francia, se recomiendan las siguientes referencias: [258, 316, 317, 318, 319, 320].

Finlandia

El inicio oficial del único piloto hasta la fecha en Finlandia tuvo lugar en el año 2005, si bien ya entre 2000 y 2004 el Ministerio de Justicia estableció unas pautas previas en 3 memorandos todavía accesibles (en finlandés) [323].

En marzo de 2006 se procedió a modificar la ley electoral con el objeto de permitir el uso de dispositivos electrónicos en las elecciones [323] y la prueba piloto tuvo finalmente lugar en las elecciones municipales del año 2008.

En concreto, 3 municipios participaron en el piloto (Karkkila, Kauniainen y Vihti) que fue voto electrónico no remoto por lo que no entra en el objeto de estudio de la tesis.

Únicamente comentar que el coste total del proyecto fue de 1.630.550 EUR [258] para un total de algo más de 12.000 votos electrónicos emitidos.

En el capítulo de incidencias, 232 votos no fueron contabilizados debido a un error de usabilidad [325] por el que los votantes no confirmaron su opción (paso necesario) antes de extraer la tarjeta electrónica, abandonando el recinto sin ser conscientes de que su voto no había sido tenido en cuenta.

Como consecuencia, varios de los votantes en la citada situación presentaron quejas al amparo de la Ley Electoral. En primera instancia, las Cortes de Helsinki fallaron a favor de la validez de las elecciones, si bien con posterioridad, el Tribunal Supremo decidió anularlas en las tres municipalidades que tomaron parte en el proyecto piloto, teniendo lugar unas nuevas elecciones en las citadas municipalidades en septiembre de 2009.

Como consecuencia, el Consejo de Ministros decidio el 13 de enero de 2010 que la implantación del voto electrónico presencial no iba a continuar [326].

Posteriormente, en noviembre de 2013 el Ministerio de Justicia conformó un grupo de trabajo para volver a evaluar la posibilidad de utilizar un sistema de voto electrónico, en esta ocasión remoto, para elecciones y referendos de carácter consultivo.

En abril de 2015 se presentó el informe final [324] en el que se proponía la organización de una prueba piloto en el ámbito del VER para referendos consultivos a nivel municipal de una manera gradual y lo más transparente posible. El objetivo primordial sería familiarizar a la ciudadanía con el voto electrónico remoto y conseguir que aumente su aceptación y confianza en él.

De la misma manera, los ayuntamientos dispondrían de una herramienta efectiva y económica (el Ministerio les facilita el servicio libre de coste) para llevar a cabo referendos

consultivos sobre cuestiones relevantes y aumentar la implicación y participación de los ciudadanos en las decisiones gubernamentales.

Desde la presente tesis se opina que el modelo de implementación finlandés aporta un enfoque muy interesante por lo gradual de la medida y el planteamiento de crear grupos de trabajo con expertos independientes y tiempo suficiente para el correcto desarrollo del proyecto.

Por otra parte, en el informe se recomienda la no adopción de sistemas de VER para elecciones generales (municipales, europeas, parlamentarias, presidenciales etc) en un primer estadio. Lo justifican por la mayor complejidad y presupuesto que demandarían.

En lo que respecta al horizonte temporal planteado, se propone el período 2016-2020 y un presupuesto asociado de 2.4 millones de EUR. Se dejaría asimismo a cada municipalidad participante libertad para decidir si permitiría la votación adelantada y/o la opción paralela del voto por correo [324].

En resumen, tras una experiencia previa en 2008 de voto electrónico en entorno controlado con algunos contratiempos relevantes, Finlandia ha decidido en 2013 realizar un estudio de viabilidad a fondo sobre el VER. En su informe final de 2015 se aboga por realizar una prueba piloto para referendos consultivos en el ámbito local en el período 2016-2020.

Desde la presente tesis se considera un enfoque prudente y muy a tener en cuenta como alternativa realista y viable de implementación del VER para los países interesados en ello.

Las referencias bibliográficas recomendadas para el lector interesado son las siguientes: [258, 321, 322, 323, 324, 325, 326].

Nueva Zelanda

Los orígenes del voto electrónico en Nueva Zelanda se remontan al año 2001, cuando en el *"Local Electoral Act"* se estableció el marco legal aún vigente de aplicación en las elecciones locales. En él, se indicaba que "método de votación" incluía "cualquier tipo de voto electrónico". No se desarrollaba más el concepto, pero dejaba abierta la puerta a su adopción.

Posteriormente, en el año 2007 la Comisión Electoral publicó un borrador sobre la estrategia de voto a largo plazo en el que se mencionaba el "voto online", valorando los pros y contras de realizar un piloto a pequeña escala [327].

En los años 2010 y 2011, sendos informes del Comité Electoral y de Justicia y de la SOLGM (*New Zealand Society of Local Government Managers*) volvían a recomendar el desarrollo de un sistema de VER como alternativa a los métodos tradicionales.

La creciente demanda, unida a una baja y menguante participación ciudadana en los comicios locales (menos de un 40%), hizo que el *Department of Internal Affairs* en septiembre de ese mismo año conformara un panel de expertos informáticos, miembros de distintos departamentos del gobierno neozelandés y autoridades locales para realizar un estudio de viabilidad al que denominaron *Online Voting Working Party* (OVWP) [328].

A partir de ese momento, el gobierno estableció una política de total transparencia, haciendo públicos todos los informes, enmiendas, actas de reuniones, requerimientos etc. Dicha información se encuentra disponible en la página del Ministerio de Interior [328].

Se debe valorar muy positivamente el enfoque adoptado por el gobierno neozelandés respecto al acceso a toda la información relativa a la preparación de la prueba piloto, tratando de acercar a la ciudadanía el VER con total transparencia, con el objetivo de aumentar la participación y confianza en el mismo. 11 meses después de la creación del OVWP, el 4 de agosto de 2014 se publicó el informe final, en el que se consideraba plausible la organización de una prueba piloto de VER para las elecciones locales de 2016 [327].

El 9 de diciembre de 2014, la ministra adjunta de gobierno local, Louis Upston, anunció que el gobierno, tras haber analizado el informe final, había decidido continuar con los pasos necesarios para permitir a un grupo reducido de municipalidades llevar a cabo una prueba piloto de VER durante las elecciones locales de 2016.

El 12 de mayo de 2015, el gobierno publicó un informe con un total de 96 requerimientos para el piloto de VER [330], seguido de una actualización de los mismos en noviembre del mismo año [331]. Finalmente, las 8 municipalidades elegidas para tomar parte del piloto fueron: Whanganui, Rotorua, Matamata Piako, Selwyn, Masterton, Porirua, Palmerston North y Wellington.

En febrero de 2016, los territorios seleccionados enviaron una serie de informes detallando el grado de cumplimiento de los requerimientos especificados en [331]. En él, algunos de ellos informaban de que no estaban en disposición de garantizar los siguientes puntos: una revisión independiente del código fuente, un test de penetración en el conjunto del sistema, una aseguración independiente del piloto y el desarrollo de una estrategia detallada de coordinación de comunicaciones a nivel nacional.

Por todo ello, la responsable Sra. Upston decidía cancelar el piloto de VER para las elecciones locales de 2016 [332]. El principal motivo aducido son las restricciones de tiempo para implementar el piloto, causando preocupación respecto a "la seguridad y la integridad del voto". También especifica que el Gobierno es receptivo a nuevas propuestas de organización de pilotos en el ámbito del VER como parte de un programa gradual de introducción en las elecciones locales.

Para el lector interesado en profundizar en el VER en Nueza Zelanda, se recomiendan las siguientes referencias: [327, 328, 329, 330, 331, 332].

3.3 Experiencias de Voto Electrónico Remoto en otros ámbitos.

En el anterior apartado 3.2 se ha hecho un pormenorizado repaso de las experiencias más destacadas de Voto Electrónico Remoto en elecciones públicas de carácter vinculante en el ámbito político.

Dicha tipología de comicios es la más exigente en sus requisitos, puesto que implican la cesión de la porción de soberanía del votante en sus representantes, base de la democracia y de las leyes que nos gobiernan. Por esa misma razón, constituyen la variedad que atrae en mayor medida a potenciales atacantes. Existen pues otros tipos de elecciones que por su naturaleza y alcance no son tan críticas: de carácter consultivo, privadas, vinculantes pero acotadas a asociaciones, sindicatos, universidades etc.

En todas ellas, al ser menor el beneficio a obtener en una potencial manipulación del proceso electoral, se puede también esperar con razonable certeza que la inversión en medios que realizarán los potenciales atacantes será también inferior.

Después de todo, no parece muy probable que un grupo de atacantes invierta millones de euros para llevar a cabo un ataque DDoS al sistema de VER de las elecciones de los representantes sindicales de los profesores de una provincia o en una consulta popular sobre propuestas de nombres para un nuevo parque, por poner dos ejemplos concretos.

Ello no implica en la mayoría de los casos que los sistemas de VER en esta tipología de elecciones sean menos seguros. De hecho, suelen ser los mismos o al menos muy similares a los utilizados en elecciones vinculantes en el ámbito político.

Por tanto, en numerosas ocasiones la utilización del VER en votaciones consultivas o en elecciones vinculantes fuera del ámbito político supone una excelente vía de acceso escalonado, minimizando riesgos y potenciales consecuencias derivadas de fallos en el diseño o la implementación.

El objetivo es que tanto los organizadores como los votantes e incluso los desarrolladores vayan adquiriendo experiencia y mejorando su desempeño:

- En el caso del votante, familiarizándose con el VER, aceptándolo y mejorando su percepción del mismo
- En el caso de los organizadores, para acercar de una manera más eficaz y económica procesos electorales y participativos a sus votantes
- En el caso de las empresas desarrolladoras, para ir mejorando el producto y sus procesos asociados y gradualmente ir aumentando la complejidad y requerimientos del sistema a elecciones cada vez más relevantes hasta llegar a las VAP incluso para todos

los ciudadanos en comicios generales (situación que en la actualidad únicamente se da en Estonia).

En las siguientes páginas se van a repasar los casos más relevantes de países que han realizado experiencias piloto de VER para elecciones en otros ámbitos distintos de las elecciones vinculantes de carácter político.

Estados Unidos

Las iniciativas de VER en los Estados Unidos tienen una larga trayectoria: ya en agosto de 1996 el *Reform Party* eligió a su candidato a la presidencia ofreciendo por primera vez la opción de votar por internet. En total, más de 2.000 votos fueron emitidos de dicha manera [15]. Posteriormente, en enero del 2000 el Partido Republicano de Alaska realizo una encuesta entre sus miembros en la que 35 de ellos votaron por internet.

En marzo de ese mismo año, tuvieron lugar en Arizona las primarias del Partido Demócrata para elegir a su candidato a las elecciones presidenciales. En diciembre del año anterior decidieron encargar a la empresa election.com la organización del evento, incluyendo la opción de voto por internet. A finales de febrero, se enviaron más de 843.000 certificados de voto con un PIN asociado para poder votar a través de un ordenador personal con conexión a internet

En total, durante los 4 días que estuvo disponible el VER (del 7 al 10 de marzo de 2000), 39.942 votos fueron emitidos (un 41% del total de 86.907). En términos de participación, ésta se triplicó con respecto a 1996.

En cuanto a incidencias, no se tuvo constancia de ningún ataque al sistema, si bien hubo un número indeterminado de afiliados que no pudieron votar debido a problemas en la recepción de los códigos PIN, la versión del navegador o el sistema operativo [333].

A mayores, la Organización No Gubernamental *"Voting Integrity Project"* interpuso una demanda al entender que el voto por internet era una manera de discriminar a las minorías del país, en general con menos recursos y por tanto acceso más limitado a internet.

Por todo ello y aunque no se produjeron ataques, no se consideró escalable el proyecto y se concluyó que era necesaria una mayor investigación sobre el VER antes de poder utilizarse de una manera más habitual.

Aún así, la experiencia mayoritariamente positiva sirvió también de impulso para que el gobierno central pusiera en marcha los proyectos de VER *VOI* y *SERVE*, explicados en el apartado 3.2.4 de la tesis.

Posteriormente, en 2004 el Partido Demócrata en Michigan eligió a su candidato en unas primarias donde 46.543 personas votaron por internet [335, 336].

Ya en el año 2008 vio la luz Helios [1], uno de los sistemas de VER más relevantes y que ha servido de base para el desarrollo numerosas soluciones a partir de entonces. Su enfoque *open source* y su gratuidad supusieron desde un primer momento un importante atractivo para su utilización. El capítulo 5.2 está dedicado al análisis en detalle este sistema.

En total, se han emitido más de 100.000 votos con Helios en elecciones tales como las de rector de la Universidad de Lovaina en 2009 [49], las de representantes de alumnos y referendos en la Universidad de Princeton así como en la *International Association for Cryptology Research* (IACR) desde 2010 [338].

En lo que respecta a las experiencias más recientes, Nueva Jersey tiene un piloto en marcha del que todavía no se conocen detalles sobre su primera prueba real [300]. Además, el Partido Republicano de Utah eligió a su candidato de las primarias ofreciendo también una opción de VER en marzo de 2016. El sistema está desarrollado por Smartmatic y la participación fue del 90%, si bien no se detalla el número total de votos emitidos [295, 310, 334].

Una de las principales razones para su introducción fue el hecho de que la mayoría de los votantes son miembros de la Iglesia de los Santos de Jesús de los Últimos Días, conocidos comúnmente como mormones que se encontraban realizando labores de misionero por el mundo. El Presidente del Partido Republicano en Utah ha expresado su intención de continuar utilizado el VER en futuras primarias del partido.

Por último, cabe destacar la iniciativa del *"Los Angeles Department of Neighborhood Empowerment"*, el cual ha contratado con Everyone Counts 50 elecciones con voto VER y telefónico por un total de 552.000 USD. El objetivo es aumentar la participación ciudadana, muy escasa hasta la fecha con una media de 264 votos en cada Consejo de Vecinos [298] pese a las importantes voces que se han alzado en contra del VER en California [299].

Canadá

Como ya se ha comentado en el apartado 3.2, uno de los pilotos no vinculantes más importantes tuvo lugar el 2 de noviembre de 2012 en la ciudad de Edmonton, provincia de Alberta. Es la conocida como *"Jellybean Internet Voting Election"* [276] en la que se emitieron un total de 497 votos [344].

El experimento tenía como objetivo evaluar la implantación de un sistema de VER para las elecciones generales de 2013. El jurado popular votó a favor en enero de ese mismo año pero un mes después el ayuntamiento finalmente decidió no autorizar su uso.

En opinión de la Dra. Goodman, consultada en junio de 2016, existe un 85-90% de posibilidades de que Alberta introduzca finlamente el VER en las siguientes elecciones.

Por lo que respecta a la última experiencia de VER en elecciones fuera del ámbito político, en febrero del 2016 el Nuevo Partido Democrático de Nueva Escocia eligió a su líder utilizando el VER. Se emitieron más de 2.700 votos, ninguno de ellos en papel [343].

Francia

Las principales experiencias francesas han sido en el sector público, en concreto de tres de sus ministerios: Asuntos Exteriores, Interior y Educación.

El caso del Ministerio de Asuntos Exteriores, ya he ha estudiado en profundidad en el punto 3.2.7 de la presente tesis.

Por lo que respecta al Ministerio de Interior, se utilizó en diciembre de 2014 un sistema de VER para la elección de los representantes de los trabajadores para las agrupaciones PRIF y CAPN [345].

En cuanto al Ministerio de Educación, a finales de 2014 se eligieron a los representantes del sindicato de profesores utilizando únicamente un sistema de VER. Se gestionaron más de 1000 mesas electorales para un total de 1,76 millones de votos emitidos [346].

Para concluir, conviene destacar que en noviembre de 2014, 268.000 simpatizantes del partido UMP pudieron elegir a su presidente a través de internet y en diciembre de 2015 los estudiantes del Instituto Nacional de Lenguas Orientales y Civilizaciones (INALCO) utilizaron un sistema de VER para elegir a sus representantes [347].

Grecia

En Grecia, el desarrollo del sistema de VER Zeus [339] en 2012 supuso el inicio de una serie de pilotos vinculantes fuera del ámbito político, sobre todo en los años 2012 y 2013. En concreto, la plataforma se ha utilizado en más de 120 elecciones para un total de 22.000 votos emitidos [340].

En cuanto al sistema en sí, está desarrollado tomando como base Helios Voting [1], con el que comparte al menos la mitad del código fuente según sus propios autores [339], siendo también en código abierto. Está desarrollado en Python y debido a la tipología de los votos utilizados en las universidades griegas, hubo que abandonar la idea de usar Helios Voting tal cual con recuento homomórfico de los votos y pasar a una implementación de mix-nets. Es además un criptosistema de 2048 bits con un orden de 2047 bits.

Su ámbito de uso más idestacado ha sido el universitario, en elecciones a rector. Destaca su utilización en las siguientes instituciones: la Universidad del Egeo, la Universidad Agrícola de Atenas, la Universidad de Patras y la Universidad de Atenas.

Su utilización no ha estado exenta de incidentes, tanto ataques informáticos como físicos (sentadas, sabotajes etc.) [339], incluyendo el descubrimiento de un *bug* en el código de las

elecciones de la Universidad de Tracia dos días antes del inicio de las mismas, el 22 de octubre de 2012. Como consecuencia, se corrigió el problema y las elecciones tuvieron lugar 5 días después de lo previsto.

A modo de ejemplo de la utilización de Zeus en otro ámbito distinto del universitario, el sistema fue utilizado por el partido de nuevo cuño "*Recreate Greece*" en sus elecciones internas y congresos [339].

En el momento de escribir estas líneas (noviembre de 2016), el proyecto parece haber entrado de una fase de *impasse* puesto que lleva un tiempo sin actualizarse y sin utilizarse en nuevos procesos electorales.

El lector interesado en profundizar en el sistema Zeus, incluso en analizar su código fuente, puede referirse a las siguientes referencias: [339, 340, 341, 342].

España

El caso de España presenta una curiosa dicotomía: su uso no está ampliamente extendido pero el país cuenta con un tejido empresarial de primera categoría mundial, que exporta sistemas de Voto Electrónico a todo el mundo. Sirva como ejemplo el hecho de que las experiencias de VER en Suiza, Noruega, Canadá, Australia, Estados Unidos, y Francia entre otras han utilizado soluciones desarrolladas en España.

Por lo que se refiere a los procesos electorales vinculantes fuera del ámbito político, o bien no vinculantes, las primeras experiencias tuvieron lugar en el Pais Vasco a finales de la década de los 90 con el desarrollo del sistema de voto electrónico (no remoto) denominado Demotek [348].

Posteriormente, en noviembre de 2003 se realizó otra prueba piloto con motivo de las elecciones del Parlamento Catalán por el que se permitía a los residentes catalanes en Argentina, Bélgica, Chile, Estados Unidos y Méjico votar desde cualquier ordenador conectado a internet. El sistema de VER fue desarrollado por Scytl y se emitieron un total de 730 votos electrónicos. Paralelamente, se realizaron otros dos pilotos de Voto Electrónico presencial desarrollados por Indra y Demotek [351].

En el ámbito del gobierno corporativo empresarial, la Junta de Accionistas de Unión Fenosa en 2003 fue pionera en la utilización del VER [353].

En marzo de 2004 se produjo una prueba piloto en la localidad de Jun (Granada) [349] y en junio del mismo año, el ayuntamiento de Madrid organizó una consulta popular denominada "Madrid Participa". Se ofrecía la opción de votar por internet, teléfono móvil que permitiese la ejecución de java y por sms. Se emitieron 882 votos, de los cuales un 11% fueron VER [350].

A nivel nacional, destaca el piloto desarrollado con motivo del referendo no vinculante sobre la Constitución Europea de 2005, en un proyecto denominado PVI (Prueba de Voto por Internet) en el que participó un municipio menor de 100.000 habitantes de cada provincia. El sistema de VER fue desarrollado por la compañía Indra y fue objeto de críticas por parte del Observatorio de Voto Electrónico por una serie de irregularidades encontradas tanto en su diseño como en la implementación [352].

Más recientemente, otra iniciativa relevante tuvo lugar en junio de 2010 en Barcelona. El Ayuntamiento, junto con las compañías Indra y Scytl organizó una consulta ciudadana sobre el futuro de la Avenida Diagonal.

El censo para la votación era el conjunto de residentes en Barcelona mayores de 16 años (por tanto distinto al oficial, al incluír a ciudadanos de entre 16 y 18 años y a ciudadanos extranjeros residentes en el municipio), totalizando 1.414.783 votantes.

Se ofrecía tanto voto electrónico en entorno controlado como VER durante un período de 5 días sin otra modalidad de participación. Existían 3 modos de autenticación:

- Certificados digitales oficiales utilizados en otros procedimientos electrónicos con la Administración
- Un password de uso único mandado al teléfono móvil del votante (previo envío de una serie de datos personales)
- A través de los portales web de los organismos asociados a la consulta y para los que el votante tenía ya acceso previo (la opción de voto estaba embebida en los citados portales de las entidades participantes).

Finalmente, 172.161 personas emitieron su voto (un 12.17% del censo), de los cuales un 48.3% lo hicieron a través de una máquina *ad hoc* en un entorno controlado y el restante 51.7% por medio del sistema de VER [258].

El sistema no preveía ningún tipo de verificación para el votante y el código era cerrado aunque accesible previa firma de un NDA, opción que fue utilizada por la Universidad Politécnica de Cataluña. En el apartado de los incidentes, se produjo un caso de suplantación que fue denunciado, si bien el acusado fue finalmente absuelto [354].

Por lo que respecta al avance del VER en fechas recientes en España, cabe destacar una mayor implantación en procesos internos de elección de los partidos políticos españoles, plasmado en las consultas vinculantes de los partidos PSOE, Izquierda Unida y Podemos entre 2014 y la primera mitad de 2016.

También dentro de ellas se pueden establecer diferencias en cuanto a las compañías encargadas del desarrollo de los sistemas. En el caso del grupo Podemos, sus votaciones han sido desarrolladas por la empresa Agora Voting/nVotes [355]. En el subapartado 5.4 se detalla y analiza su sistema de voto.

Por lo que respecta a las iniciativas de VER en distintas consultas a militantes y simpatizantes que han tenido lugar en 2016, el PSOE la implementa a través de su iniciativa miPSOE, que incluye una aplicación para iOS y Android.

En líneas generales, parece que el uso del VER en España en entornos no vinculantes en el ámbito político se encuentra en una fase expansiva, como demuestran iniciativas de democracia ciudadana tales como Open Seneca [453], participada por Telefónica y Scytl.

Austria

Aunque el origen del VER en Austria se remonta al año 1998 con la modificación del *Students' Union Act* para permitir el voto remoto (tanto postal como electrónico), no fue hasta 2007 cuando el entonces Ministro Federal de Ciencia e Investigación Dr. Johannes Hahn, anunció que el voto electrónico remoto sería permitido por primera vez en las elecciones del Sindicato de Estudiantes Austriaco del año 2009.

El desarrollo software junto con los servicios de consultoría fueron adjudicados a Scytl. La coordinación del proyecto era responsabilidad del Ministerio Federal de Ciencia e Investigación con la colaboración de diversas universidades y grupos de expertos. Su presentación oficial se produjo en diciembre de 2008.

El 18 de marzo de 2009 se realizó un simulacro en las universidades de economía y negocio de Viena y en la de Leoben. El 8 de mayo de 2009 hubo unas jornadas de trabajo para que comisionados y observadores analizasen el código fuente previa firma de un NDA. Los críticos adujeron que no tuvieron acceso más que a una parte de las más de 183.000 líneas de código.

La votación tuvo lugar entre los días 18 y 22 de mayo de 2009 (las fechas reservadas para el voto tradicional eran el 26, 27 y 28 del mismo mes). Sobre un censo total de 230.528 estudiantes, 2.161 decidieron ejercer su derecho a voto por medio de la herramienta de VER, lo que supone un 0.94% del total. Por su parte, los votos en papel fueron un total de 57.231; un 24.83% del censo.

El votante se autenticaba con su tarjeta de estudiante, una tarjeta activada de ciudadano y la introducción de un códido de 4 dígitos y otro de 6. Una vez realizada la selección y antes de enviar el voto, el votante debía enviar también una declaración jurada de que su voto se había producido de manera secreta, no observado ni influenciado. Posteriormente lo firmaba digitalmente con otro código de 6 dígitos y recibía un código de confirmación y otro de verificación.

A continuación, para el recuento se volcaban todos los datos encriptados a un CD-ROM y desde ahí se introducían a un equipo que nunca había sido conectado a internet y cuya única función era el recuento de los votos, a través de la clave de la clave privada de la comisión, que a su vez había sido dividida en 4 partes.

Con posterioridad a las elecciones, todos los discos duros, CD-ROMs y equipos fueron destruidos por la empresa Reisswolf.

Se produjeron también una serie de incidentes relacionados con retrasos en el recuento de votos y adicionalmente, en dos casos no se pudo desencriptar los votos. Ello, unido a que la participación no aumentó sustancialmente (pese a un desembolso total de al menos 900.000 EUR) y a una serie de quejas por parte de asociaciones de estudiantes y universidades, propició que el 2 de abril la Ministra de Ciencia B. Kärl anunciase que el VER no volvería a utilizarse en las elecciones al Sindicato de Estudiantes de 2011.

Finalmente, el 13 de diciembre de 2011, la Corte Constitucional falló anulando las elecciones al Sindicato de Estudiantes de 2009, poniendo fin al VER en Austria hasta la actualidad.

Para conocer más sobre el VER en Austria, referirse a [356, 357].

3.4 Conclusiones de las experiencias hasta la fecha

En los apartados 3.2 y 3.3 se han repasado las experiencias más destacadas de Voto Electrónico Remoto tanto en elecciones vinculantes en el ámbito político (VAP en adelante) como en otros tipos de elecciones y referendos.

En total se han estudiado los casos de 12 países y más de 500 elecciones que conjuntamente suman más de 6 millones de votos emitidos.

La variedad de enfoques, ritmos de implantación, incidentes y respuestas ante los mismos suponen una fuente muy valiosa de información para integrar en la metodología de evaluación de sistemas de VER, objetivo último de la tesis.

Algunos países han apostado desde un primer momento muy decididamente por el VER y se han mantenido fieles a su opción elegida (Estonia), mientras otros se han limitado a realizar pruebas en elecciones no vinculantes (Nueva Zelanda) o han decidido cancelar sus programas (Noruega, Reino Unido, Alemania etc.).

Existe otro gran bloque de países que han implementado un modelo gradual de utilización del VER (únicamente para votantes no residentes, o bien solamente para elecciones locales etc.) como pueden ser Canadá, Suiza o Australia.

Entre las prácticas destacadas e interesantes de cara a una potencial armonización en protocolos y legislación, al menos a nivel europeo destacan el sistema gradual de porcentaje de VER permitido en función del desempeño del sistema de Suiza y el excelente esfuerzo de transparencia por parte de Noruega y Finlandia. Esas tres experiencias constituyen una base firme sobre la que desarrollar el VER en Europa en años venideros.

Los criterios adicionales obtenidos del presente capítulo, así como su introducción en la metodología y su ponderación asociada se desarrollan en el capítulo 4 de la presente tesis.

Antes de proceder a presentar la metodología, se introduce la siguiente tabla a modo de resumen del presente capítulo, con las experiencias previas más relevantes de VER en todos sus ámbitos, con sus principales guarismos y situación actual:

País	Período	N. elecciones	N. votos	Tipología	Estatus Actual	Apuntes
Estonia	2003 -	8	756.277	VER VAP	Activo para todo tipo de elecciones	Único país que ha apostado por la introducción universal del VER en VAP (todos los comicios y disponible para todo el censo).
Noruega	2008 – 2014	2	97.644	VER VAP elecc. locales	Cancelado en 2014.	El partido actualmente en el gobierno (Partido Conservador) no es partidario del VER. No es descartable que se retome si vuelve al poder el partido que lo impulsó (Laborista).
Canadá	2003 -	> 200	> 2 mill.	VER VAP elecc. locales	Activo en las provincias de Ontario y Nueva Escocia.	Dentro de las provincias donde está activo, los municipios son libres de adherirse al VER. No se espera legislación a nivel estatal hasta al menos 2019.
USA	VAP: 2000 – 2015 Otros: 1996 -	3 > 20	272 >100.000	VER VAP para votantes UOCAVA en 5 estados	Activo en Alabama, Alaska, Arizona, Misuri y Dakota del Norte para UOCAVA.	A nivel nacional, en 2015 el NDAA canceló el mandato de organizar un piloto para evaluar la implantación del VER. A nivel local, activo para UOCAVA en 5 estados. Poca información del sistema y participación. En otros ámbitos, activo (primarias de partidos políticos).
Suiza	2003 -	> 200	> 260.000	VER VAP y referendos	Activo en 13 cantones. *On-hold* en cantones de FR, GR, SG, SO, AG, SH, TG	Desde 2014 el gobierno central ha implementado un sistema de porcentajes permitidos de VER en función de una serie de requerimientos a cumplir. Objetivo: implementación

						gradual asociada al nivel de desempeño de cada solución.
Australia	2007 -	3	326.689	VER VAP en elecciones regionales	Activo en Nueva Gales del Sur	Las autoridades de Nueva Gales del Sur consideran bajo el riesgo de coerción por lo que el sistema no está diseñado para ser RC. Se permite incluso el voto por teléfono, no suficientemente seguro.
Francia	2003 -	VAP: 5 Otras:> 4	> 262.676 > 2 mill.	VER VAP para franceses no residentes y también VER en otros ámbitos	Activo para elección de AFE, elecciones de ministerios y sindicatos	Desde 2009, M. Asuntos Exteriores ha contratado una licencia permanente con una empresa proveedora para implementar el VER en las elecciones a la AFE. También uso esporádico en elecc. locales.
Grecia	2012 - ??	> 120	> 22.000	VER en otros ámbitos	Proyecto Zeus poco actualizado	VER utilizado en otros ámbitos distintos del VAP. Especialmente en elecciones universitarias, aunque también de manera interna el partido político *Recreate Greece*.
Finlandia	2006 – 2008 2013 - ??	1	12.234	VER en elecc. locales	En estudio una propuesta de VER para referendos no vinculantes.	Única experiencia de en 2008, terminó invalidándose en los tribunales. Desde 2013 existe un grupo de trabajo que en 2015 presentó informe con propuesta para implementar gradualmente el VER empezando por referendos consultivos. Propuesta de primer piloto para período 2016-2020
N. Zelanda	2013 - 2016	--	--	VER en elecciones locales	Cancelado en 2016	Se estuvo estudiando su implantación para las elecciones locales de 2016 pero finalmente las autoridades consideraron que no

						había tiempo suficiente para su desarrollo con todas las garantías. No descartable una implantación futura.
Austria	2008 - 2011	1	2.161	VER en otro ámbitos	Invalidado en 2011 por el tribunal supremo	Primera y única experiencia en las elecciones del sindicato de estudiantes de 2009. Hubo una serie de fallos y polémicas que llevaron a su anulación y la cancelación del programa.
España	1998 -	> 10	> 200.000	Pilotos de VER en locales y europeas. También en referendos y consultas.	No hay legislación desarrollada para el VER	El mayor piloto tuvo lugar en Barcelona en 2010. Consulta popular con más de 172.000 votos emitidos. Además, algunos partidos políticos los usan para consultar a la militancia (2014-). Iniciativa Open Seneca.

Tabla 12: Resumen de las principales experiencias de Voto Electrónico Remoto tanto en elecciones vinculantes en ámbito político como en otros ámbitos

Capítulo 4

CRITERIOS ADICIONALES DE EVALUACIÓN Y METODOLOGÍA COMPLETA

Guárdate del hombre de un solo libro
-B. Disraeli

El presente capítulo 4 constituye la culminación de combinar el punto 2.3, *"Requerimientos de un sistema de Voto Electrónico Remoto"* con el aprendizaje práctico extraido de las experiencias reales recogidas en los puntos 3.2 *"Experiencias previas de VER en elecciones vinculantes en al ámbito político"* y 3.3 *"Experiencias de VER en otros ámbitos"*.

En el punto primer punto 4.1 se definen y detallan el conjunto de criterios adicionales de la metodología de evaluación, que derivan del estudio cualitativo y cuantitativo de las experiencias en elecciones en todos los ámbitos, como se ha recogido en el capítulo 3.

Con posterioridad, en el apartado 4.2 se recogen la totalidad de criterios seleccionados en los puntos 2.3 y 4.1 y se les asigna una ponderación basada en los resultados de la encuesta técnica del anexo B enviada a 31 expertos internacionales del VER tanto en su vertiente académica como empresarial. De ellos, 21 ofrecieron su feedback, añadiendo un elemento extra de solidez y objetividad a la metodología. Ya en el subsiguiente capítulo 5 *"Análisis y comparativa de las soluciones más relevantes de VER"*, se aplica la metodología completa a los sistemas de VER más importantes hasta la fecha.

4.1. Criterios adicionales de evaluación de sistemas de VER

En el apartado 2.3 de la presente tesis se definieron una serie de requisitos basándose en los trabajos más relevantes de los principales autores en la materia: [51, 68, 93, 172, 23, 1, 260, 359, 369, 388] así como en las siguientes metodologías y estándares:

- Método KORA (*Concretization of Legal Requirements* en alemán) [461]
- *Common Criteria for Information Technology Security Evaluation SO/IEC 15408:2009* [460]
- ISO 27001/IT-Grundschutz [462]
- Sistema Simic-Draws et al. [457que combina las 3 anteriores.
- Las recomendaciones sobre certificación de sistemas de (sic) e-voting del Directorado General de Democracia y Asuntos Políticos del Consejo de Europa [456]
- El trabajo de Volkamer sobre requisitos legales del voto [459]
- El trabajo de Bräunlich et al. sobre la transformación de criterios legales en TDGs (*Technical Design Goals*) [463]

- La tesis doctoral de Neumann [458], que se apoya en [457] y [463] para identificar 16 aspectos técnicos que debería cumplir un sistema de *i-voting*.

Dichos requisitos son:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software y escalabilidad y suman un total de 43 subapartados para la evaluación de sistemas de VER (referirse al Anexo A para el listado completo).

Entre ellos hay dos que son condición *sine-qua-non* para que el sistema sea susceptible de ser utilizado en elecciones vinculantes en el ámbito político, por representar las características de una votación democrática según la Constitución Española y el Consejo de Europa [66, 358]: **la verificabilidad extremo a extremo (E2Ev) y la resistencia a la coerción (RC)** (puntos 2.2.2, 2.2.3 y 2.3 de la presente tesis).

Los 5 restantes: **inviolabilidad, usabilidad, monitorización/auditoría, operacional software y escalabilidad**, pueden hallarse implementados en distintas intensidades según cada sistema de VER.

A continuación se van a incorporar los criterios adicionales de cariz marcadamente práctico, seleccionados tras el exhaustivo proceso de estudio tanto cualitativo como cuantitativo de todas las experiencias de VER detalladas en el capítulo 3 y que totalizan más de 500 elecciones y 6 millones de votos emitidos.

En cada uno de ellos se explica la experiencia o experiencias que han motivado su inclusión en la metodología al no encontrarse entre los requerimientos tradicionales.

Los criterios adicionales son: **desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.**

De manera análoga al apartado 2.3 de requisitos tradicionales, se adjuntan las codificaciones de los criterios adicionales para facilitar su lectura y posterior aplicación a las soluciones más relevantes en el capítulo 5:

Requisito	*Formato de codificación*	*Ejemplo*
Desarrollo ex_software	DESW-*n*	DESW-4, DESW-7, etc.
Protocolo contra inc. y ataques	PA-*n*	PA-1, PA-4, etc.
Versatilidad	V-*n*	V-1, V-2, etc.
Coste	C-*n*	C-2, C-3, etc.
Mantenimiento	M-*n*	M-1, M-2, etc.

Tabla 13: Codificación criterios adicionales VER

4.1a Desarrollo ex_software (DESW-*n*)

En el apartado 2.3f se detallaban los requerimientos del desarrollo software del sistema de VER. En la práctica, se ha comprobado cómo los protocolos y actividades del sistema no relacionadas con el software (de ahí el nombre ex_software) tienen una importancia capital en un correcto funcionamiento del mismo.

Los casos de Estonia, Noruega, Estados Unidos, Australia y Suiza [236, 237, 249, 260, 264, 277, 285, 308, 36] entre otros han demostrado que una gestión inadecuada del desarrollo ex_software puede conllevar consecuencias muy negativas que socaven la seguridad y la confianza en el VER. Prácticas reales como las siguientes son ejemplo de ello:

- Dejar contraseñas de las redes inalámbricas y credenciales de acceso de autoridades a la vista (en realidad, la propia utilización de redes inalámbricas en dichas instalaciones requiere de un estudio previo de viabilidad y seguridad),
- Un deficiente control de acceso a las instalaciones críticas de una manera no distribuida ni protocolizada
- La transmisión del recuento final de la votación en un CD o una memoria USB sin la correcta supervisión de una tarea con una importancia tan crítica.

Por ello, el desarrollo de protocolos de actuación ex_software debe diseñarse, implementarse y revisarse paralelamente al software de VER. (DESW-1). Únicamente planteando las actuaciones ex_software como parte inseparable al desarrollo software se estará en disposición de obtener un sistema de VER completo, integrado y suficientemente seguro.

Los apartados concretos que debe incluir el desarrollo ex_software son:

- Protocolo de distribución de credenciales, permisos y responsabilidades: con especial énfasis en evitar colusiones entre autoridades. Para ello no se debe permitir la acumulación de atribuciones en una única persona, implementando una política de distribución de responsabilidades, especialmente en los nodos críticos del sistema (DESW-2).
- Protocolo automatizado de control de accesos y vigilancia de las instalaciones e infraestructuras del sistema de VER, con uno o varios niveles reforzados de acceso en función de la criticidad del sub-sistema (DESW-3).
- Protocolo de auditoría y observadores independientes. Además de la normativa aplicable a las autoridades, debe existir un protocolo detallado para auditores y observadores independientes del proceso electoral. Su labor deberá disponer de libertad de movimientos y supervisión siempre que no interfieran con la seguridad y/o la privacidad de la votación. (DESW-4).
- Protocolo de backup. En la práctica no es posible conseguir un sistema 100% seguro, aunque se diseñe e implemente con dicha finalidad (DESW-5).

En ocasiones se descubren fallos y vulnerabilidades una vez han concluido las elecciones, incluso años después. Por ello, es crítico desarrollar un protocolo de *backup*.

Idealmente, deberían realizarse una serie de copias periódicas de seguridad de manera automátizada de acuerdo a un procedimiento distribuido entre autoridades. Con posterioridad se procedería a almacenarlas en una localización no conectada al sistema de VER y aislada del resto de las instalaciones. Este punto enlaza con el criterio de mantenimiento del apartado 4.1e.

Además, se tendrán en cuenta los siguientes puntos:

1) El conjunto de protocolos y acciones debe estar regido por el principio de la distribución de atribuciones y responsabilidades, evitando la concentración de las mismas en una única persona o grupo de personas relacionadas. El riesgo de colusión entre las partes debe estar siempre presente. (DESW-6).
2) La existencia de sistemas de votación complementarios al VER (DESW-7).
3) Se deberá informar a los votantes con suficiente antelación y medios el modo concreto en el que se va a articular en VER, poniendo a su disposición canales para resolver las dudas y recibir *feedbacks*(DESW-8).
4) Se informará con suficiente antelación de las opciones de re-votar en caso de que existan, especificando el procedimiento y la primacía del voto en papel sobre el voto electrónico en caso de conflicto (DESW-9).
5) Organización de encuestas de opinión y técnicas sobre cohortes seleccionadas de perfiles de votante para obtener *feedbacks* lo más fiables posible sobre tendencias, usabilidad, fallos y mejoras futuras (DESW-10).
6) El envío de credenciales de autenticación se producirá a través de dos canales distintos, para reforzar la seguridad (DESW-11).
7) Debe existir un protocolo de inicialización que incluirá una prueba final del correcto funcionamiento de las infraestructuras y medios de la votación inmediatamente anterior al comienzo del período de VER (DESW-12).
8) Se implementarán, siempre que sea posible, sistemas protocolozados o estandarizados, para facilitar la interoperabilidad entre sí (DESW-13).
9) Se valorará la existencia de un servicio telefónico gratuito de asistencia previo al inicio de las elecciones y durante las mismas (DESW-14).
10) Se tendrán en cuenta todas las actividades realizadas para publicitar el sistema de VER así como las jornadas abiertas presenciales y *webinars* para formar a la población en el manejo de la misma (DESW-15).

4.1b Protocolo contra incidencias y ataques (PIA-*n*)

La experiencia ha demostrado que los procesos electorales vinculantes que utilizan sistemas de VER son regularmente objeto de ataques [23, 36, 236, 287, 339]. Entre los países

que han sido víctima destacan: Estados Unidos, Australia y Ucrania [238]. El lector puede referirse al apartado 3.2 y subsecciones para profundizar en dichos ataques. Algunos han tenido lugar en fechas tan recientes como 2015, explotando las vulnerabilidades del ataque FREAK (punto 2.4.2.3a de la presente tesis), poniendo en peligro hasta 66.000 votos.

En el sostenido avance y complejidad de los ataques, se ha llegado al punto en el que existen empresas y hackers especializados en vender herramientas listas para atacar vulnerabilidades del navegador, sistema operativo o programa que se elija a modo de un "ataque a la carta" temporizado para una fecha en concreto. [239, 240].

Por ello, es fundamental el VER contemple los ataques producidos hasta la fecha y diseñe e implemente unos protocolos de seguridad actualizados hasta el momento de la elección. Ello no garantiza que el sistema sea inmune, pero representa la exigencia mínima en un entorno con antecedentes reales de intentos de agresión.

Los puntos a valorar en este apartado son los siguientes:

1) Idealmente existirán los siguientes documentos desarrollados: *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP) y Control Validation Audit (CVA).* (PIA-1).
2) La existencia de protocolos específicos contra ataques y la política de prevención en función del esquema del sistema de VER (p. ej. protocolos específicos contra ataques de tipo DDos en soluciones que utilizan mix-nets). (PIA-2).
3) Se recomienda que toda la información e infraestructuras esté dentro del país donde de las elecciones, a excepción de la porción estrictamente necesaria relacionada con el voto de los ciudadanos residentes en el extranjero. (PIA-3).
4) Se implementarán protocolos que garanticen la no pérdida permanente de información en caso de ataque, con redundancia de equipos y sistemas. (PIA-4).
5) El enfoque distribuido del protocolo contra ataques, de tal manera que para inutilizar el sistema de VER no sea suficiente atacar un único recurso/localización. (PIA-5).
6) Acciones de concienciación ciudadana. Es crítico que se forme a los votantes y sean conscientes de los ataques en los que ellos son el vector (*phishing*, ingeniería social etc.) (PIA-6).
7) La contratación de hackers y expertos independientes para poner a prueba el sistema antes de desplegarlo en elecciones reales (PIA-7).

4.1c Versatilidad (V-*n*)

Se puede abordar desde una doble vertiente:

1) Como se ha apuntado en el apartado 2.1.2 de *"otras consideraciones"*, existen multitud de elecciones según el número y orden de los candidatos. Además, cada país posee un sistema de elección e idiosincrasia propias. Por ejemplo, en Noruega(3.2.2) y Australia (3.2.5), las listas suelen incluir multitud de candidatos y en ocasiones se deben

incluso ordenar o completar con nombres de otras listas. Por el contrario, en Suiza (3.2.6) es habitual votar en referendos del tipo sí/no.

Desde el punto de vista del desarrollo de un sistema de VER, ello supone una enorme diferencia como se ha visto en el apartado 2.2 de *building blocks* criptográficos. Cada tipología de VER tiene asociado un esquema más idóneo para un funcionamiento eficiente y más seguro frente a ataques. Por ello, un sistema que disponga de varias versiones adaptadas según la tipología de votación tendrá una mayor versatilidad y puntuará mejor en el presente apartado (V-1).

2) La versatilidad de un sistema de VER también va ligada al punto 2.3d de usabilidad. Una de las principales motivaciones de apostar por el Voto Electrónico Remoto es la de permitir a sectores de población con algún tipo de limitación con el sistema de voto tradicional (discapacidad visual, auditiva, problemas de movilidad, edad avanzada etc.) ejercer su derecho de una forma más sencilla y adaptada.

   Se valorará por tanto que el sistema de Voto Electrónico Remoto ofrezca soluciones destinadas específicamente para dichos colectivos (V-2).

Otros aspectos a tener en cuenta son los siguientes:

1) El votante debe de poder votar utilizando su equipo personal, con una conexión de internet convencional y sin instalar software adicional (V-3).
2) El sistema debe desarrollarse y probarse en navegadores y dispositivos que tengan una implantación superior al 1% del censo de votantes (V-4).
3) El *interface* debería cumplir con los requisitos AA del WCAG 2.0 (V-5).

4.1d Coste (C-*n*)

El aspecto del coste de un sistema de VER está relativamente poco documentado pese a su indudable importancia a la hora de decantarse por una solución u otra. Entre los requisitos tradicionales del VER no se suele incluirse debido a que en un primer estadio de estudio criptográfico/metodológico, no es primordial.

Aún asi, como todo proyecto, tiene asignado un presupuesto y en ese sentido unas elecciones en cualquier ámbito no son una excepción. En el apartado 1.2 se ofrecen algunas nociones sobre los costes asociados a procesos electorales en varios países.

En lo que respecta a los sistemas de VER utilizados en votaciones reales, en ocasiones existe un cierto hermetismo en lo que concierne a las ofertas presentadas por las empresas desarrolladoras de software en elecciones VAP.

Los casos de Noruega, Australia o Finlandia han sido los más transparentes hasta la fecha en cuanto a coste y procedimientos de *tender* o subasta pública. En concreto, algunos datos concretos sobre costes son los siguientes:

- 1.2 mill. de CAD el piloto de Markham (Canadá) en 2010 [268, 270, 271]
- 1.6 millones de EUR el piloto de Turku (Finlandia) en 2008 [258]
- Más de 900.000 EUR el piloto de Austria en 2009 [356, 357]
- 6.7 millones de NOK el piloto de Noruega en las elecciones locales de 2011 [418]
- Al menos 4.3 millones de AUD en las elecciones estatales de Nueva Gales del Sur en 2015 [291, 293]
- 2.4 millones de EUR de presupuesto para 2016-2020 en Finlandia [324]
- De 5 a 7 millones de NZD para un piloto en Nueva Zelanda con entre 25.000 y 75.000 de VER y entre 8-10 millones de NZD para un piloto a nivel nacional con unos 125.000 VER en total. [328]

En pilotos menores, tales como procesos únicamente locales o incluso simplemente consultivos es todavía más difícil (cuando no imposible) tener acceso a los costes. Aún así, por ejemplo, la iniciativa *Empower L.A. 2016* para consultas ciudadanas tiene un presupuesto asignado de 552.000 USD para un máximo de 50 elecciones [298].

Lo que también se conoce es que las mayores compañías del sector ofrecen distintos niveles de seguridad de software dependiendo de la tipología de elección y el presupuesto disponible. A mayor seguridad y complejidad, mayor es el precio del producto. Por ello, es de gran importancia que se destinen partidas suficientes para poder abordar la introducción de tecnologías de VER sin comprometer la seguridad de las elecciones.

En este criterio se valora por tanto la transparencia y la claridad en la presentación del coste de implantación (C-1) del sistema de VER (bien sea en términos totales o en coste por votante) así como el precio en sí (C-2). A igualdad de nivel de seguridad, lógicamente un sistema más económico puntuará mejor en este apartado.

4.1e Mantenimiento (M-*n*)

En ocasiones obviado en los artículos más académicos/científicos si bien en proyectos reales es de una importancia vital como se pudo comprobar en el caso de Australia entre otros [287]. El mantenimiento se entiende desde una doble vertiente:

1) En relación con el propio sistema de VER (tanto software como ex_software). Es decir, cuán actualizado está el sistema y por tanto cómo de seguro se está frente a los últimos ataques que se van descubriendo [47].

   Cuanto más frecuentes y rigurosas sean las actualizaciones del sistema, mejor se puntuará en el presente apartado. Se recomienda la existencia de un *log* no modificable y accesible únicamente por parte de los agentes autorizados para conocer en cada momento el estado de actualización de cada componente del sistema de VER. (M-1)

   Este aspecto es muy relevante, puesto que en ocasiones se abandonan las labores de mantenimiento en sistemas más académicos incluso durante años, como en el caso

de Helios Voting [1]. En el apartado 5.2 se analiza en profundidad, junto con sus variantes. En su caso, ha pasado por paréntesis de más de 2 años en los que no se realizó ninguna labor de mantenimiento.

2) La existencia de un protocolo de conservación de los datos y *back up* de las elecciones bajo unas condiciones de seguridad suficientemente sólidas que garanticen la privacidad de los votos y los votantes incluso en un futuro, cuando mejoren las capacidades de ataque de los adversarios que quieren comprometer el sistema. (M-2) en lo que se denomina *"everlasting privacy"* [235, 241, 242].

   El requisito del mantenimiento en el Voto Electrónico Remoto toma una especial relevancia porque puede darse la situación de que un ataque se descubra una vez que hayan concluido las elecciones. Ello implica necesariamente mantener los resultados de las elecciones y los votos (aunque sean encriptados y privados del vínculo con el votante) a diferencia de las elecciones tradicionales, en las que pasado un período de tiempo, se destruyen las papeletas.

   Finalmente, en este apartado se valorará también el coste asociado a dicho mantenimiento de los datos (M-3).

Con este último subapartado han quedado definidos los criterios adicionales de evaluación para sistemas de VER derivados de la experiencia concreta en elecciones vinculantes en todos los ámbitos: **desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.**

En la siguiente figura se ofrece un resumen de una forma más gráfica de los criterios adicionales detallados en este punto 4.

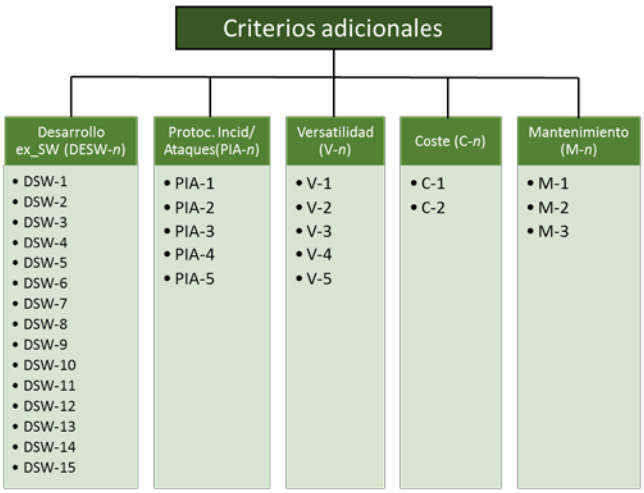


Figura 36: Criterios adicionales del VER

4.2 Metodología de evaluación para sistemas de VER

Como se ha ido desgranando a lo largo de la tesis, los criterios que conforman la metodología provienen de dos fuentes:

- En el apartado 2.3 *"Requerimientos de un sistema de Voto Electrónico Remoto"*, se definen 7 requisitos exigibles a todo sistema de VER: **Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software y escalabilidad.**
- En el apartado 4.1 *"Criterios adicionales de evaluación de sistemas de VER"*, se incorporan 5 factores adicionales para la metología: **desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.**

El hecho de añadir los criterios del apartado 4.1 responde a la necesidad de tener en cuenta las experiencias reales de utilización de sistemas de VER en elecciones vinculantes tanto en el ámbito político como en otros ámbitos y que se repasan en el capítulo 3 de la presente tesis *"Antecedentes, experiencias previas y estado del arte"*.

Una metodología de evaluación que únicamente tuviera en cuenta requerimientos teóricos o *"ex - ante"* detallados en el apartado 2.3 iría en la línea de los trabajos de Neumann [458] o la metodología KORA [461]. Ello implicaría que se centraría más en los esquemas criptográficos, alejándose de la naturaleza práctica del voto y los sistemas de VER.

Por ello, los **12 criterios** que conforman la metodología de evaluación para sistema de voto electrónico son los siguientes:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

De ellos, 2 se consideran *sine-qua-non* puesto que representan las propiedades que un proceso electoral democrático debe respetar según la Constitución Española y el Consejo de Europa [66, 358]. Por tanto, su evaluación es del tipo *"cumple"* o *"no cumple"*: son la verificabilidad extremo a extremo (E2Ev) y la resistencia a la coerción (RC).

Los 10 restantes: inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento pueden encontrarse implementados en diverso grado. Cada uno de ellos se subdivide en un número de puntos concretos de evaluación, sumando en total 73. En el anexo A se ofrece el listado completo y la definición asociada.

La siguiente figura ofrece un resumen gráfico de la metodología de evaluación:

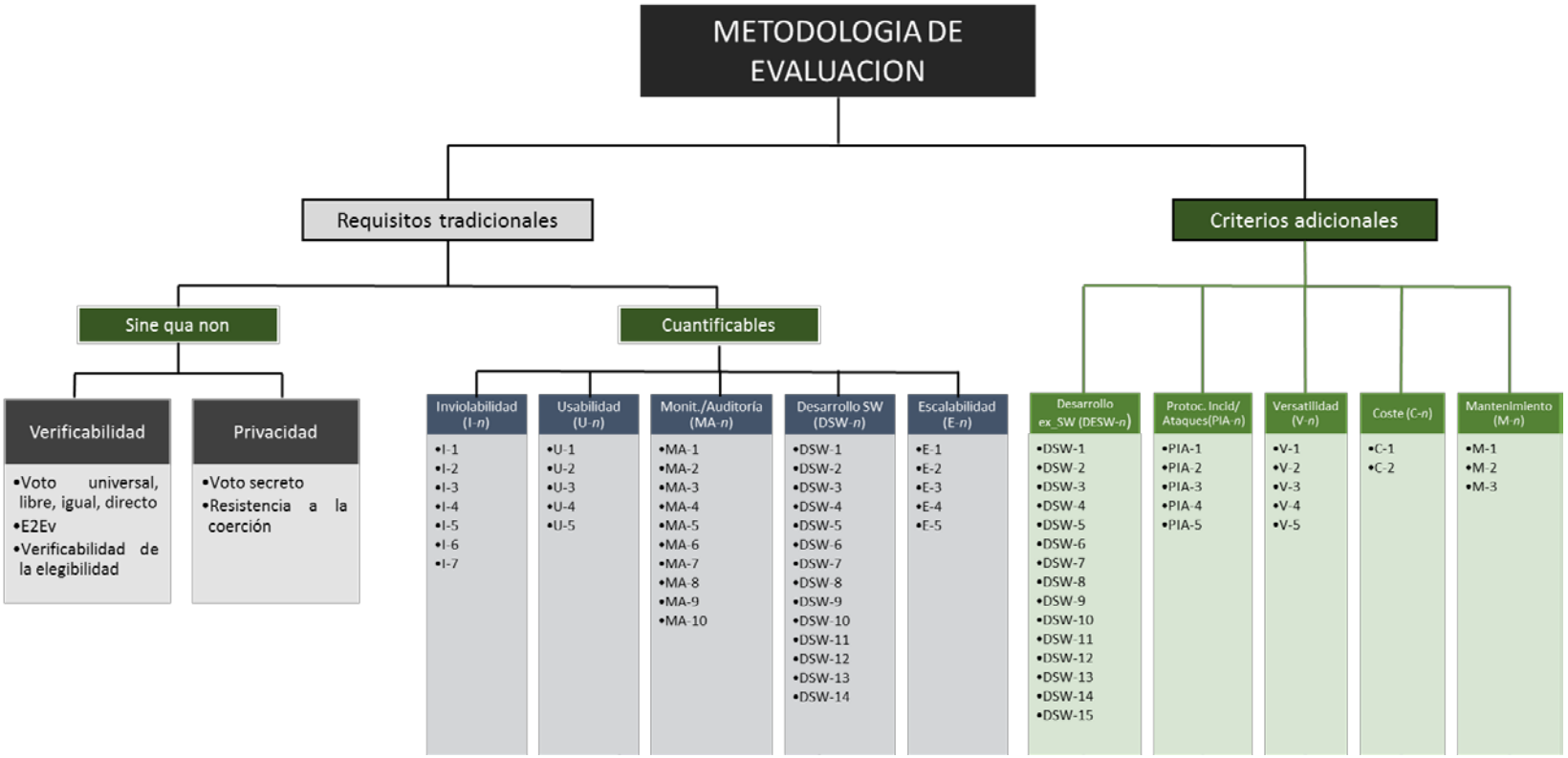


Figura 37: Estructura de la metodología de evaluación de sistemas de VER

Como mejora adicional de la metodología, se ha introducido un factor de diferenciación entre los distintos requisitos. Si bien todos cubren una parcela imprescindible del VER, algunos presentan una mayor criticidad. Por ello, se ha incorporado un coeficiente de ponderación individual, para destacar su importancia relativa dentro del conjunto.

Para decidir el coeficiente de cada uno, se preparó un formulario técnico para recabar la opinión de renombrados expertos en VER tanto a nivel nacional e internacional.

Las metodologías utilizadas para decidir la muestra de expertos a contactar fueron *snow-ball sampling* [454] y *judgement sample* [455], extendiéndose la recogida de datos 12 meses.

De las 31 personas contactadas, finalmente 21 ofrecieron su *feedback* y en consecuencia, el factor asignado a cada uno de los criterios refleja su contribución. También se ha tenido en cuenta la experiencia de los dos directores de la presente tesis en sus más de 15 años de implicación directa en experiencias de VER.

La explicación en detalle y la justificación de cada criterio se encuentra en los apartados 2.3 y 4.1, si bien en esta sección se repasarán de manera muy somera, precediendo a la ponderación asignada.

En el apéndice D se encuentra la tabla con el conjunto de las respuestas emitidas sobre las ponderaciones, si bien se ha anonimizado a los expertos a petición propia.

Para facilitar el cálculo y la lectura del análisis de los sitemas de VER, los factores de ponderación recibidos se ajustaron proporcionalmente para que su suma total fuese de 10 puntos.

Verificabilidad Extremo a Extremo ($E2E_v$)

Todo sistema de VER debe cumplir simultáneamente dos propiedades imprescindibles y en principio antagónicas entre sí: integridad y privacidad [23, 1, 260, 388, 438].

Como establece el Consejo de Europa [358]: "Los cinco principios clave del voto en una ley electoral son: universal, igual, libre, directo y secreto y ellos son la raíz de la democracia". Por tanto, cualquier sistema de VER debe respetarlos.

La propiedad encargada de la salvaguarda de los cuatro primeros (universal, igual, libre y directo) es la E2Ev. Definida como: "Un sistema de VER es E2Ev si cada voto es i) emitido como estaba previsto ii) guardado como se ha emitido iii) contado como se ha guardado y cumple con la propiedad de verificabilidad de la elegibilidad.", para más detalles referirse a los apartados 2.2.2 y 2.3a y [51, 77, 3, 389].

Ponderación

La integridad es una propiedad inherente e irrenunciable a un sistema de VER por la misma definición formal de la ley electoral. En consecuencia, la E2Ev no tiene asignada ni ponderación ni valoración numérica.

Su evaluación es en términos de "cumple" ○ o "no cumple" ×. Un caso especial lo constituyen los sistemas de VER que cumplen bajo ciertas premisas más o menos difíciles de reproducir en entornos reales.

En esas circunstancias, puede ser injusto incluir en el mismo grupo a un sistema que no cumple de ninguna manera con la propiedad junto con otro que ha realizado un esfuerzo adicional y está más cerca de llegar al nivel óptimo. Por ello, existe también el símbolo Δ que representa "cumple bajo ciertas premisas".

Privacidad/resistencia a la coerción (RC)

Siguiendo con lo establecido en el Consejo de Europa [358], la quinta propiedad de una ley electoral democrática es el voto secreto.

Proteger la privacidad del votante y del voto es una característica de cualquier tipología de votación. En el VER, el hecho de votar en un entorno remoto y no controlado implica que las dificultades para asegurar la privacidad aumentan notablemente.

Retomando lo dicho en el punto 2.2.3, la protección de la privacidad en las soluciones de VER está categorizadas en 3 niveles de menor a mayor exigencia:

- Privacidad del voto: El voto no es revelado a nadie
- Ausencia de recibo: Un votante no puede obtener información que pueda probar a un coaccionador cómo votó.
- Resistencia a la coerción: Un votante no puede demostrar el sentido de su voto ni siquiera colaborando con un coaccionador.

En el año 2000, Hirt y Sako fueron los primeros en demostrar que la segunda condición no era suficiente para garantizar la privacidad del votante [108].

En 2002, Juels et al. introdujeron de manera formal el concepto de resistencia a la coerción en 2002 [70] y lo depuraron en 2005 [63] y 2010 [104].

Por ello y teniendo en cuenta la enorme responsabilidad que conllevan las elecciones vinculantes en el ámbito político, el nivel a exigir es el tercero: la resistencia a la coerción.

Ponderación

Análogamente al caso anterior de la E2Ev, la privacidad es una propiedad inherente e imprescindible para un sistema de VER. Por tanto, la RC se evalúa también en términos de "cumple" ○ o "no cumple" ×.

Existe también el símbolo Δ que representa "cumple bajo ciertas premisas" y que sirve para destacar los sistemas han realizado los mayores avances para cumplir con la RC.

Inviolabilidad (I-*n*. Apartado 2.3c)

La inviolabilidad hace referencia al concepto de seguridad del sistema, entendiendo por sistema no solo el software desarrollado sino el conjunto de aplicaciones y protocolos que integran la solución de VER.

Se debe demostrar rigurosidad en la política de seguridad del sistema, a través de protocolos de seguridad diseñados, implementados y mejorados continuamente (I-7) con el *feedback* (I-1) (I-2) de las vulnerabilidades que irán apareciendo.

Se valoran de forma especial: la existencia de una filosofía distribuida de implementación en las políticas de seguridad, sobre todo en los nodos críticos (I-4), tratando de minimizar la posibilidad de colusión entre partes y también la implementación protocolos de *risk assessment* y de *threat modeling* (I-5).

Para una descripción completa de los factores a tener en cuenta en el apartado de inviolabilidad, referirse al punto 2.3c y al anexo A.

Ponderación

El consenso de los expertos consultados fue de asignar a la inviolabilidad el mayor factor de peso, **1.2** debido a su papel insustituible dentro del sistema VER y su influencia sobre el resto de requisitos.

Usabilidad (U-*n*. Apartado 2.3d)

En el diseño y desarrollo de una solución de VER, además de las cualidades imprescindibles relacionadas con la integridad, la privacidad, la seguridad etc., se debe tener presente el objetivo último de acercar y facilitar el voto al votante tanto como se pueda (U-1).

En ese sentido, los colectivos que más se pueden beneficiar de los sistemas de VER son posiblemente las personas de edad avanzada así como los grupos con limitaciones físicas (tanto de movilidad como visuales, auditivas etc.) para ejercer su derecho (U-2).

Es pertinente recordar en este punto el *"Legal, Operational and Technical Standards for e-voting"* del Consejo de Europa [54] así como sus dos recomendaciones más claras sobre usabilidad y accesibilidad en los sistemas de VER:

- Recomendación 3: *"Los sistemas de e-voting serán diseñados, …, para maximizar las oportunidades que duchos sistemas pueden proveer a personas con discapacidades"*
- Recomendación 63: *"Se proporcionará a los usuarios, cuando sea demandado y posible, facilidades adicionales, tales como interfaces especiales u otros recursos equivalentes, tales como asistencia personal. Los servicios de usuario deberán estar en línea en la medida de lo posible con los principios establecidos en la Web Accesibility Iniciative (WAI)".*

Por ello, el diseño y ejecución del sistema de VER no puede olvidar el objetivo de facilitar la usabilidad a todos los segmentos de votantes, en especial a los más vulnerables.

Acemyan et al. por una parte y Summers et al. por otra [231, 232], han presentado sendos papers en los que demuestran que queda mucho camino por recorrer en usabilidad de sistemas de VER, incluso entre las soluciones más populares.

Para una descripción completa de los factores a tener en cuenta en el apartado de usabilidad, referirse al punto 2.3d y al anexo A.

Ponderación

Siendo la usabilidad de un sistema de VER una característica imprescindible, dentro del conjunto de criterios a evaluar, existen otros con una mayor prioridad y así quedó de manifiesto entre los expertos consultados, quienes le asignaron un coeficiente de **0.8**.

Monitorización/auditoría (MA-*n*. Apartado 2.3e)

En un proceso de votación, se produce la cesión de la parte proporcional de poder del votante al vencedor. Dependiendo de la naturaleza de la elección, dicha cesión será más o menos duradera y de una intensidad mayor o menor. El caso de mayor relevancia es el de las elecciones públicas vinculantes en el ámbito político.

En unas elecciones existen partes interesadas que podrían estar tentadas de influir de manera irregular en el proceso electoral. Éstas pueden ser desde miembros de partidos políticos hasta lobbies, asociaciones, autoridades de diversa índole o incluso el propio desarrollador del sistema de VER.

Además, es necesario asegurarse de que los protocolos en materia de seguridad y el resto de requisitos se cumplan de acuerdo a lo previsto.

Es evidente pues, la necesidad de un control externo e independiente de manera contínua (MA-1) sobre el conjunto del sistema y todos sus componentes, incluyendo el software, los accesos, la documentación, la gestión de errores, el proceso de auditoría, etc.

También la monitorización y la auditoría deberían abordarse de una manera distribuída, evitando acumular responsabilidades y capacidades en un único individuo (MA-8).

Se valorará por tanto la existencia de protocolos de auditoría y observadores externos; su grado de distribución, su diseño, implementación y actualización, la generación de informes periódicos de actividad (no editables ni transformables, MA-4), su correcto transporte y almacenaje, la existencia de un *"banco de pruebas de diagnóstico"* para utilizar en el caso de cualquier eventualidad (MA-7) y la independencia del sistema de auditoría.

Para una descripción completa de los factores a tener en cuenta en el apartado de monitorización/auditoría, referirse al punto 2.3e y al anexo A.

Ponderación

La monitorización y la auditoría constituyen la salvaguarda imprescindible que garantiza que el sistema se comporta de acuerdo a lo previsto y en caso contrario aporta la infomación necesaria para subsanar el error y sus consecuencias. Es como el cuerpo de policía encargado de velar por la seguridad.

Por ello la existencia de una auditoría independiente, bien diseñada e implementada es de una importancia capital para cualquier sistema de VER.

En esa línea, el consenso de los expertos junto con la opinión del autor y los directores de la presente tesis ha sido asignar la ponderación máxima de **1.2**.

Desarrollo software (DSW-*n.* Apartado 2.3f)

Un sistema de VER, se trata de un programa de software de una elevada complejidad con una serie de protocolos de actuación específicos asociados.

Por ello, la transcripción en código fuente de todos los requerimientos y políticas de integridad, seguridad, distribución, privacidad y demás atributos de un sistema de VER es de una importancia capital.

Se valorará pues positivamente una correcta ingeniería del software, con un diseño, implementación y documentación debidamente desarrollados (DSW-1).

También se tienen en consideración entre otros: la compatibilidad de sistema con las distintas plataformas de uso (DSW-9), el control de accesos desde programas externos (DSW-10), la distribución en los permisos asociados a los cambios críticos del programa

(DSW-2), una correcta implementación de las primitivas criptográficas (DSW-11) y la disponibilidad del código fuente para la revisión por parte de la comunidad científica (incluso previa firma de NDAs. DSW-12).

También una correcta política de actualizaciones, con la introducción de módulos específicos contra los ataques conocidos hasta la fecha (DSW-14) contribuye a obtener un mejor resultado en este apartado.

Para una descripción completa de los factores a tener en cuenta en el apartado de desarrollo software, referirse al punto 2.3f y al anexo A.

Ponderación

El desarrollo del software es la forma de plasmar *"negro sobre blanco"* y de una manera verificable y auditable buena parte de los requisitos de un sistema de VER.

De su desempeño depende en gran medida el éxito o fracaso de las elecciones y por tanto su cociente de ponderación es, por consenso de los expertos consultados, el máximo que se otorga en la presente tesis: **1.2**.

Escalabilidad (E-*n*. Apartado 2.3g)

A lo largo de la tesis ha quedado patente que la transición de un modelo teórico a condiciones reales de uso ha sido origen de numerosos fallos y ataques a sistemas de VER [145, 105, 175, 87, 90, 91, 92, 26, 33, 464].

Por ello, confiar cuestiones importantes de seguridad a una teórica escalabilidad del sistema de VER es una praxis poco segura y a evitar siempre que sea posible.

En ese sentido, se valora positivamente que el sistema de VER (tanto en su vertiente software como ex_software) haya sido testado en condiciones más exigentes que la votación que va a gestionar (E-1).

No se debe permitir que las elecciones constituyan el primer momento en que el sistema funciona bajo unas condiciones iguales o análogas (E-3), sobre todo en lo referente a nodos críticos y primitivas criptográficas (E-2). Se valora también la escalabilidad como capacidad de gestionar EVAP, las más complejas y exigentes (E-5).

Para una descripción completa de los factores a tener en cuenta en el apartado de escalabilidad, referirse al punto 2.3g y al anexo A.

Ponderación

La escalabilidad es un factor fundamental a evaluar en un sistema de VER, si bien el factor de ponderación asignado dentro de la metodología de evaluación por el conjunto de expertos, el autor y los directores de la presente tesis es de **0.8**.

Desarrollo ex_software (DESW-*n*. Apartado 4.1a)

Como complemento inseparable del requisito de desarrollo software aparece la necesidad de un correcto diseño e implementación de todos los componentes del sistema que no sean propiamente el software; de ahí la denominación ex_software.

De las experiencias reales de VER en Estonia, Noruega, Australia, EEUU y Suiza entre otros, se ha podido verificar cómo los protocolos y actividades ex_software tienen una importancia capital para un correcto funcionamiento del mismo [23, 36, 236, 249, 260, 264, 277, 285, 308].

Un deficiente control de accesos, la existencia de protocolos opacos de recuento y transporte de votos así como la presencia de credenciales (*logins* y *passwords*) de autoridades a la vista de cualquier asistente son ejemplos reales de brechas de seguridad inaceptables ex_software en unas elecciones con implementación de sistemas de VER.

Es por ello que el desarrollo de protocolos de actuación ex_software debe diseñarse, implementarse y revisarse de manera paralela al software del VER. Únicamente planteando las actuaciones ex_software como parte intrínsecamente ligada al desarrollo software se estará en disposición de obtener un sistema de VER completo, coordinado y suficientemente seguro (DESW-1).

Se valorará la existencia de: protocolos seguros de distribución de credenciales, permisos y responsabilidades (DESW-2), de control de accesos y vigilancia (DESW-3), de auditoría y observadores independientes (DESW-4) y de *backup* (DESW-5).

Para una descripción completa de los factores a tener en cuenta en el apartado de desarrollo ex_software, referirse al punto 4.1a y al anexo A.

Ponderación

Debido a su influencia sobre otros criterios de la metodología y al tratarse de la otra cara de la moneda inseparable del desarrollo software, el coeficiente de ponderación asignado al desarrollo ex_software por los encuestados es de **1.2**.

Protocolo contra incidencias y ataques (PIA-*n*. Apartado 4.1b)

El segundo factor que se ha extraído de la experiencia en el uso del VER en elecciones vinculantes surge de la elevada probabilidad de que se produzcan incidencias o incluso ataques [23, 36, 236, 287, 339].

En los casos más graves dentro de la denominada *"cyberwarfare"*, los objetivos de dichos ataques han sido elecciones públicas vinculantes en el ámbito político como sucedió en Ucrania [238], Estados Unidos y Australia.

En los apartados 2.4 y 3.2 y se puede profundizar sobre dichos ataques y su tipología. Algunos de ellos han llegado a poner en peligro hasta 66.000 votos explotando las vulnerabilidades del ataque FREAK (apartado 2.4.2.3a de la presente tesis).

La tipología y complejidad de los ataques se ha vuelto tan sofisticada que existen hasta foros y sitios web en los que se ofertan ataques del tipo *"zero-day exploits"* adaptados a cada navegador, sistema operativo o programa [239, 240].
En un entorno con amenazas tan reales, se valorará la existencia de protocolos específicos de *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP) y Control Validation Audit (CVA)* (PIA-1). También la prevención que se introduzca en función de la tipología de votación y del sistema de VER (PIA-2) así como el enfoque distribuido del protocolo (PIA-5), las actividades de formación ciudadana en ciberseguridad (PIA-6) y la contratación de hackers y expertos independientes para tratar de comprometer el sistema (PIA-7).

Para una descripción completa de los factores a tener en cuenta en el apartado de protocolo contra incidencias y ataques, referirse al punto 4.1b y al anexo A.

Ponderación

Ningún sistema puede considerarse totalmente seguro. De hecho, cuanto más esté en juego en unas elecciones, mayor es la probabilidad de que se produzca una incidencia o un ataque durante el proceso electoral.

Por tanto, es de una vital importancia que el sistema de VER diseñe, implemente y actualice continuamente un protocolo contra incidencias y ataques. Ello no conseguirá que el sistema sea 100% seguro pero es el mínimo exigible a una solución para que pueda considerarse como apta para una votación vinculante en cualquier ámbito.

Todo ello quedó plasmado en la ponderación asignada por parte del plantel de expertos al presente criterio: **1.2**, siendo el quinto y último requisito en alcanzar tal relevancia.

Versatilidad (V-*n*. Apartado 4.1c)

Entendida desde una doble vertiente:

1) Un sistema que disponga de distintas versiones adaptadas según la tipología de votación (V-1), como se detalla en el apartado 2.1.2.
2) Un sistema de VER que ofrezca implementaciones específicas para ayudar a los colectivos más susceptibles de beneficiarse del mismo (votantes de edad avanzada, con discapacidades visuales, auditivas o motoras) obtendrá una mejor puntuación (V-2).

Poder votar desde el equipo personal, sin instalar software adicional es también un factor a tener en cuenta (V-3), junto al desarrollo para las distintas plataformas con más de un 1% de cuota de mercado (V-4) y la adhesión al estándar de diseño WCAG 2.0 (V-5).

Para una descripción completa de los factores a tener en cuenta en el apartado de versatilidad, referirse al punto 4.1c y al anexo A.

Ponderación

El coeficiente de consenso asignado en este caso fue de **0.6**.

Coste (C-*n*. Apartado 4.1d)

Al igual que cualquier otro proyecto, un sistema de VER tiene asociado un presupuesto. Ello repercute en la disponibilidad de recursos que pueden destinarse a su desarrollo.

Pese a su indudable importancia, la documentación sobre el coste real de desarrollo de un sistema de VER es en ocasiones escasa y poco contrastable. Las compañías desarrolladoras llegan en ocasiones a realizar ofertas a pérdidas para asegurarse el cliente y obtener la rentablidad en sucesivos contratos o con el mantenimiento del mismo.

Análogamente, es habitual que se ofrezcan distintas versiones de VER con precios diferentes en función de la calidad y el conjunto de garantías que ofrece el sistema.

En ese sentido, se valorará positivamente la claridad y disponibilidad de información en cuanto a la política de costes de la empresa desarrolladora (C-1) así como el coste en sí de los comicios (C-2).

No obstante, en el caso de no disponer de suficiente presupuesto para adoptar una solución de VER con suficientes garantías de seguridad, es preferible postponer su implantación hasta dichos recursos estén disponibles.

La experiencia ha demostrado reiteradamente que los riesgos son elevados incluso con los medios adecuados [23, 36, 236, 287, 339], como para plantearse introducir un sistema de VER sin los niveles necesarios de seguridad por escasez de recursos.

Para una descripción completa de los factores a tener en cuenta en el apartado de coste, referirse al punto 4.1d y al anexo A.

Ponderación

El coeficiente de consenso asignado es de **1.0**, reflejando su relevancia pero quedándose un escalón por debajo. La integridad, privacidad y sus factores relacionados tienen una importancia superior al mero coste del proyecto. Como ya se ha comentado, si no hay

presupuesto suficiente, es preferible esperar antes que poner en riesgo unos comicios democráticos.

Mantenimiento (M-*n*. Apartado 4.1e)

El mantenimiento se aborda desde una doble perspectiva:

1) La actualización constante del propio sistema de VER (tanto software como ex_software). Cuanto más frecuentes y rigurosas sean las actualizaciones del sistema, mejor puntuación se obtendrá (M-1).
2) El mantenimiento como *"everlasting privacy"* [235, 241, 242]. El sistema de VER tiene que garantizar que los datos relacionados con la votación se conservan de un modo seguro, garantizando el anonimato del votante y su voto en el largo plazo (M-2).

   Con ello se persigue que si se descubre un ataque con posterioridad a que hayan concluido las elecciones, se dispone de un *backup* seguro de los datos. De manera ideal, la protección de los datos debería garantizarse incluso teniendo en cuenta los avances en capacidad computacional previstos en los años venideros.

Por último, se tendrá también en cuenta el coste asociado al mantenimiento en la doble vertiente explicada anteriormente (M-3).

Para una descripción completa de los factores a tener en cuenta en el apartado de mantenimiento, referirse al punto 4.1e y al anexo A.

Ponderación

La elevada cantidad de incidencias y ataques que se producen en procesos de votación con sistemas de VER justifican la importancia de un correcto protocolo de mantenimiento en el doble sentido del concepto (del propio sistema y de la *everlasting privacy*).

Nadie está en disposición de predecir el futuro ni las capacidades computacionales de dentro de 10 o 20 años (y menos todavía teniendo en cuenta los recientes avances en computación cuántica). Por ello, el coeficiente promedio de los expertos consultados ha sido de **0.8**, no tan elevado como el de los criterios *"core"*.

CONCLUSIÓN

Todos los capítulos y apartados de la presente tesis hasta este punto han tenido un objetivo principal: realizar un estudio exhaustivo, cualitativo y cuantitativo desde todos los puntos de vista posibles de cara a desarrollar una metodología actualizada y rigurosa de evaluación de sistemas de VER.

Por ello se ha comenzado repasando los conceptos, consideraciones, definiciones y *building blocks* asociados a un sistema de VER desde un punto de vista criptográfico y matemático. (Capítulo 2 y subapartados).

Después de conformar una base matemática y criptográfica sólida, se han definido los 7 requerimientos de todo sistema de VER en el apartado 2.3 en función de la bibliografía más destacada sobre la materia.

A continuación, se han estudiado en profundidad la seguridad y los ataques relacionados con el VER en el apartado 2.4.

En el capítulo 3 se han repasado exhaustivamente las experiencias reales de VER en los países más destacados tanto en el ámbito político como en otros ámbitos. En total se han estudiado más de 500 experiencias y 6 millones de votos a través de sistemas de VER.

De toda esa labor de investigación, se han obtenido 5 criterios adicionales de evaluación de sistemas de VER que se han detallado en el apartado 4.1 *"Criterios adicionales de evaluación de sistemas de VER"*.

Finalmente, en el presente apartado 4.2 *"Metodología de evaluación para sistemas de VER"* se han unificado los 12 criterios obtenidos y se han dividido en dos categorías: 2 requisitos *sine qua non* (E2Ev y RC) y 10 que se puntúan de 0 a 10 con un decimal de precisión: **inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento** y que se subdividen en 73 puntos concretos de evaluación para sistemas de VER. El listado completo se encuentra en el anexo A de la tesis.

A cada uno de los 10 últimos se le ha asignado un factor de ponderación que refleje su importancia dentro de la metodología de evaluación.

La asignación del factor se ha realizado de acuerdo a todos los proyectos de VER evaluados y a la profunda experiencia de más de 15 años en el campo de los directores de la presente tesis, junto con las respuestas de 21 expertos a nivel internacional a un formulario técnico sobre la metodología y el peso que cada factor debería tener. La selección de los expertos se ha realizado utilizando las tecnologías de sampleado *snowball* [454] y *judgement* [455]. En el anexo B de la presente tesis se adjunta el formulario remitido.

La metodología completa queda conformada pues de la siguiente manera para cada uno de los sistemas de VER a evaluar:

$$\sum \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Siendo f_i el factor o requerimiento i, w_i el peso o ponderación del requerimiento i, n el número total de requisitos de aplicación al VER que se esté evaluando y t el total de requerimientos de la metodología.

En cuanto a la codificación y la ponderación de los criterios, queda de la siguiente manera tras haber realizado por motivos de facilidad de lectura y evaluación un ajuste proporcional para que la suma total de factores fuesen 10 puntos:

Criterio	**Codificación**	**Ponderación**
Verificabilidad extremo a extremo	E2Ev	N.A.
Privacidad/resistencia a la coerción	RC	N.A.
Inviolabilidad	(I-*n*)	**1.2**
Usabilidad	(U-*n*)	**0.8**
Monitorización/Auditoría	(MA-*n*)	**1.2**
Desarrollo software	(DSW-*n*)	**1.2**
Escalabilidad	(E-*n*)	**0.8**
Desarrollo ex_software	(DESW-*n*)	**1.2**
Protocolo contra incidencias y ataques	(PA-*n*)	**1.2**
Versatilidad	(V-*n*)	**0.6**
Coste	(C-*n*)	**1.0**
Mantenimiento	(M-*n*)	**0.8**

Tabla 14: Criterios, codificación y ponderación de la metodología de VER

En el anexo A se ofrece una tabla resumen de los 75 puntos que conforman la metodología de evaluación de una forma resumida y más manejable.

Finalmente, una vez diseñada y definida la metodología de evaluación, el siguiente paso es aplicarla de una manera práctica a las soluciones de VER más destacadas hasta la actualidad. Ello se realiza en el siguiente capítulo 5 "Análisis y comparativa de los sistemas más relevantes de Voto Electrónico Remoto".

Para concluir el presente capítulo se presenta a continuación un esquema resumen de la metodología completa de evaluación para sistemas de Voto Electrónico Remoto:

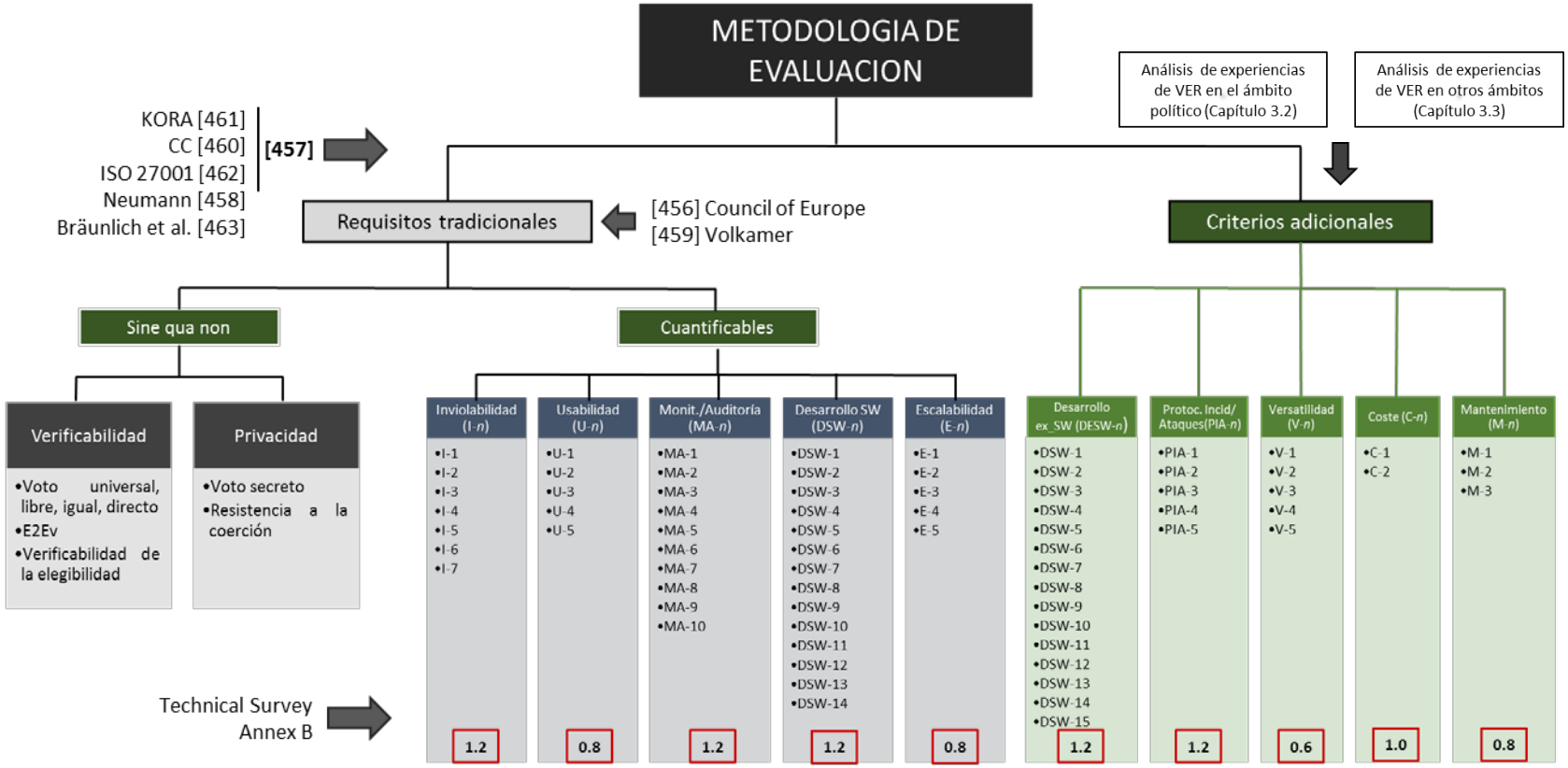


Figura 38: Metodología de evaluación de sistemas de Voto Electrónico Remoto completa

Parte III

Aplicación de la metodología a los sistemas VER más relevantes

Caesar caesaris Deus dei
Al César lo que es del César, a Dios lo que es de Dios
-Proverbio latino

Capítulo 5

ANÁLISIS Y COMPARATIVA DE LOS SISTEMAS DE VOTO ELETRÓNICO REMOTO MÁS RELEVANTES

Post tenebras, lux
Después de la oscuridad, la luz
-Job 17:12

5.1 Introducción

Recapitulando hasta este punto, en el capítulo 2 se han detallado las primitivas matemáticas y criptográficas de aplicación a los sistemas de VER, los requisitos previos y la tipología de ataques más relevantes.

A continuación, en el capítulo 3, se han pormenorizado los antecedentes, las principales experiencias a nivel internacional del VER tanto en elecciones en el ámbito político vinculante como en otros ámbitos y se ha repasado el estado actual de arte.

En el capítulo 4, se han seleccionado los criterios adicionales para un sistema de VER derivados de la experiencia real en más de 500 elecciones y 6.000.000 de votos emitidos a través de internet. Posteriormente y apoyándose en el *feedback* de 21 expertos de la comunidad científica y empresarial del Voto Electrónico Remoto, se ha asignado un coeficiente a cada factor, conformando la metodología de evaluación.

Como colofón, en el presente capítulo 5 se aplica la metodología a los principales sistemas de VER implementados hasta la fecha, evaluando sus fortalezas y debilidades, para aportar una radiografía concreta de cada uno. Con ello se introduce un componente práctico adicional a una metodología que ya de por sí tiene en cuenta tanto las propiedades más académicas como las experiencias reales para dotarla de un cariz eminentemente funcional.

Para cada una de las soluciones analizadas se introducen el esquema y sus características para posteriormente analizar el desempeño concreto de cada uno de los criterios, utilizando la codificación propia detallada en el apartado 4.2.

Este capítulo se divide en dos grandes apartados:

- Apartado I centrado en sistemas de VER que se han implementado en la práctica y han sido utilizados en comicios reales.

- Apartado II sobre esquemas de VER relevantes por sus características pero que no han sido desarrollados de un modo práctico hasta un uso real.

Apartado I

5.2 Helios Voting

5.2.1 Introducción

Helios Voting [1] es un sistema de VER de origen académico en código abierto desarrollado por Ben Adida, basándose en los modelos de Cramer y Benaloh [132,3] así como en el protocolo Sako-Kilian/Benaloh [32] cuyos *inputs* son textos cifrados con ElGamal [64].

Helios ha sido utilizado en numerosas ocasiones en elecciones reales en el ámbito no político, como por ejemplo en las elecciones a rector de la Universidad de Lovaina [49] o en las elecciones al Consejo de Administración de la *International Association for Cryptographic Research* (IACR) [393]. En total, se han gestionado más de 100.000 votos en sus distintas versiones [396].

El gran paso adelante que introdujo Helios fue su enfoque abierto, lo que permitió a la comunidad científica analizarlo, ponerlo a prueba y proponer mejoras. En el siguiente apartado se estudian las más relevantes hasta la fecha.

Actualmente, se puede acceder al código con todas sus actualizaciones [394], a la documentación [395] y al sitio para organizar unas elecciones [396].

5.2.2 Características

Helios Voting es un sistema de VER en código abierto, públicamente auditable, concebido para entornos con bajo riesgo de coerción. Fue introducido por su autor Ben Adida en [1] y en varios sentidos fue una herramienta pionera como primer sistemade VER accesible, públicamente auditable, funcional y gratuito.

En ese sentido, Helios no aporta ninguna novedad desde el punto de vista criptográfico. Su aportación es la de combinar distintas técnicas criptográficas para conformar un sistema auditable y *open-source* de Voto Electrónico Remoto.

De las dos principales características que debe de cubrir un sistema de VER, Helios aclara desde el principio que la privacidad, entendida como resistencia a la coerción no es una de ellas y que debe ser usado en elecciones con un riesgo bajo de coerción tales como asociaciones, universidades etc. En el paper original lo deja patente: *"la privacidad está garantizada si se confía en Helios"*. La integridad en cambio, sí que se convierte en una prioridad,

permitiendo la verificación tanto individual por parte del votante como universal puesto que existe un tablón en el que aparecen los votos cifrados.

El proceso de voto es como sigue:

1. El/los usuario(s) o responsable(s) crea(n) una nueva elección introduciendo los parámetros y las lista de votantes autorizados.
2. El BPS (*Ballot Preparation System*) genera la papeleta de voto y la clave pública y privada p_k y s_k respectivamente.
3. Cada votante recibe un e-mail que contiene su nombre de usuario, password así como la URL de la elección.
4. Una vez clica, la aplicación Javascript comienza y descarga los parámetros.
5. El votante selecciona su(s) opción(es) y el BPS crea un voto que es encriptado con la clave pública p_k. El voto contiene también una prueba de conocimiento cero para asegurar que el voto está bien formado (evitando que en vez de un 1, un posible votante malioso haya introducido por ejemplo 10 para que su voto cuente 10 veces) puesto que los votos no se desencriptan individualmente sino a través de las propiedades homomórficas de ElGamal exponencial.
6. El cliente software muestra al votante un *hash* del voto encriptado. El votante tiene dos opciones: auditar el voto o bien enviarlo.
7. Si decide auditarlo, el sistema muestra al votante el valor aleatorio utilizado para encriptar su voto. El votante puede entonces tomar dicho valor para verificar que su voto contiene la(s) opción(es) seleccionadas. La verificación la puede escribir el propio votante o bien utilizar la herramienta BEV (*Ballot Encryption Verification*) en Python suministrada por Helios.

   El voto auditado no obstante, ya no es válido y el votante tiene que reiniciar el proceso de votación. Por ello, Helios se encuadra dentro de los sistemas de protocolo *cast-or-audit* para garantizar la verificabilidad individual. El votante puede repetir tantas veces como quiera la verificación hasta que se convenza de que Helios es fiable.
8. Si por el contrario el votante decide "sellar" su voto, en ese momento el BPS le pedirá que se identifique con sus credenciales (ID de usuario y password).
9. El votante envía el ID de usuario, password, voto encriptado y la ZKP al servidor, que comprueba que toda la información es correcta.
10. Una vez concluido el período de votación, el servidor de Helios publica en el tablón (*bulletin board* en la versión original) los votos encriptados, junto con el nombre del votante (en versiones posteriores se utilizan alias [49] para proteger la privacidad del votante o bien no se publica ninguna referencia del votante).
11. El/los administrador(es) de la elección pulsa el botón *"shuffle"* con lo que se activa el proceso de re-encriptación y permutación. Sucesivamente, *"shuffle proof"* inicializa las pruebas del proceso de mezclado (puesto que se usa una mixnet verificable).

12. El/los administrador(es) proceden a desencriptar la multiplicación de los votos encriptados, finalmente obteniendo el recuento de las elecciones por las propiedades homomórficas de ElGamal aditivo o exponencial cuando activa el botón *"tally"*.

De una manera más formal y teniendo en cuenta la versión más reciente en la que se utilizan técnicas de desencriptación de umbral o *threshold* (punto 2.2.4.5) que permiten que con *k* de *n* autoridades se pueda desencriptar:

Considerando por cuestiones de simplicidad unas elecciones del tipo sí/no o bien 1 o 0 respectivamente:

Utilizando la notación tradicional para los votantes de Alice, Bob y Charlie; si Alice quiere votar la opción v_a, encripta su voto con la clave pública p_k dando lugar a $\{v_a\}_{pk}$.

También se adjunta al voto una prueba de conocimiento cero ZKP_a que compruebe que el voto es válido, es decir, $v_a = 0$ o $v_a = 1$. Si no se hiciese esta comprobación, un votante malicioso podría enviar $v_a = i$, siendo i un entero positivo o negativo, cambiando el resultado de la votación como se verá más adelante.

Posteriormente, Alice envía $\{v_a\}_{pk}$, ZKP_a a la urna que al ser pública, permite a Alice comprobar que su voto está presente:

Urna electoral	
Votante	**Voto**
Alice	$\{v_a\}_{pk}, ZKP_a$
Charlie	$\{v_c\}_{pk}, ZKP_c$
Bob	$\{v_b\}_{pk}, ZKP_b$

Tabla 15: Urna electoral Helios Voting

Para el recuento se usa la propiedad homomórfica de ElGamal [64]: la multiplicación de la encriptación de los votos corresponde a la encriptación de la suma de los votos.

$$\prod_{i=1}^{n}\{v_i\}_{pk} = \left\{\sum_{i=1}^{n} v_i\right\}_{pk}$$

Dicha operación la puede llevar a cabo cualquier parte y únicamente restaría desencriptar $\{\sum_{i=1}^{n} v_i\}_{pk}$ por parte de las autoridades.

Como se ha apuntado en el apartado anterior, la naturaleza *open-source* de Helios ha hecho que la comunidad científica haya ido contribuyendo a su mejora, encontrando debilidades y poniendo solución a muchas de ellas. A continuación se detallan las más relevantes:

5.2.2.1 Helios 2.0 y el ataque de Estehghari et al. [372, 374]

En la versión 2.0, Helios introducía herramientas de *threshold* para distribuir los permisos y aumentar la seguridad en el desencriptado. Poco después de su publicación, Estehghari et al. [374] demostraron que se podía atacar el cliente software utilizando como vector de ataque el pdf que se permitía enviar a los candidatos con una declaración suya adjunta.

En concreto, el atacante prepara un archivo pdf que contiene además de la declaración del candidato una función maliciosa en JavaScript que se activa cuando se abre el archivo. A continuación, se usa ingeniería social a través del Rootkit u ocultador del navegador (en este caso Mozilla Firefox) para instalar la extensión maliciosa y suprimir el aviso del gestor de extensiones de Firefox.

El único evento no habitual durante el ataque es el reinicio momentáneo del navegador puesto que Firefox necesita realizar esta acción para cargar los cambios realizados. Una vez instalada la extensión que altera los votos, se consigue que el votante acepte un voto incorrecto sin darse cuenta del cambio.

El ataque tiene una serie de premisas de partida como son el hecho de que únicamente funciona sobre el sistema operativo Windows XP, el navegador Firefox, las versiones del Adobe Acrobat/Reader 7.0.0, la 8.1.2 o la 9.0.0 y el hecho de que el votante tiene derecho de escritura sobre el archivo de instalación de Firefox. Ninguna de las citadas premisas es especialmente exigente o infrecuente.

De hecho según los autores, en torno a un 85% de los equipos personales utilizan versiones vulnerables del Acrobat, un 30% del total de navegadores instalados es Mozilla Firefox y el 80% de ellos tienen instalado el plugin the Adobe Acrobat/Reader. Todo ello implica que, en el hipotético caso de unas elecciones nacionales, habría posiblemente millones de equipos vulnerables al ataque.

Si bien éste era limitado en su alcance, el hecho de que fuese desarrollado en 2 semanas y conllevase únicamente 950 líneas de código, (el 10% de ellas específicas para Helios) fue una importante llamada de atención sobre las vulnerabilidades del equipo del votante en el VER.

Como consecuencia, Helios 2.0 se actualizó a su versión 3.0 en la que los votantes podían postear el voto auditado al servidor Helios. Como comentan los autores del ataque, ello implica que el votante puede comprobar, además de que el *hash* ha sido correctamente computado, la información del voto (el factor aleatorio, el voto y el *hash* pueden ser publicado en un sitio público). Y todo esto se puede realizar antes de enviarlo. Por ello, el votante puede apuntar o fotografiar el voto y el *hash* y comprobarlo en otro equipo para asegurarse de que no ha sido víctima del ataque.

A mayores, los autores explican un ataque sobre la privacidad del votante que se implementaría de una manera análoga a lo detallado: en este caso la extensión se diseñaría para capturar el nombre del candidato elegido antes de la encriptación del voto, así como la dirección email del votante durante la fase de autenticación. Utilizando el servicio SMTP, se podría enviar la información a una dirección email del atacante o través de una conexión HTTPS de tal forma que no se puede obtener ninguna información de la comunicación.

Según los autores, Helios 3.0 es vulnerable a este tipo de ataque sobre la privacidad. De hecho, esa es la línea que siguieron Cortier et al. [371], sobre las debilidades del secreto del voto en Helios y cómo solucionarlas.

Conviene también hacer mención del trabajo de Küsters et al. en [18] en el que explican una tipología de ataques que ellos denominan *"clash attacks"* o ataques de colisión. Éstos están basados en poner en entredicho la asunción de Helios de que las autoridades, el navegador del votante y el tablón son honestos. De esa manera, el navegador podría emitir el mismo voto a distintos votantes con la misma opción, reutilizando la misma secuencia de factores aleatorios r_1, r_2, etc.

Dicho ataque es válido para las variantes de Helios con alias [49] o con nombres separados (se eliminan los nombres al lado de los *hash* de votos en el tablón para evitar que se pueda relacionar un voto con el votante). Para evitarlo, proponen que el votante participe en la generación de las cadenas aleatorias a utilizar en la encriptación del voto o bien que los votantes puedan comprobar que no existen dos votos iguales.

5.2.2.2 Ataques sobre la privacidad de Helios [371]

Una de las primeras vulnerabilidades de Helios fue introducida por Cortier y Smyth en 2011 y posteriormente ampliada [371] en lo que se conoce como la familia de ataques contra la privacidad del voto. En su modo más simple, indica que nada impide a un atacante copiar un voto del tablón y reenviarlo como su propio voto.

Siguiendo con la nomenclatura del presente apartado, en el caso de que Charlie fuese un votante deshonesto, la urna quedaría de la siguiente manera:

Urna electoral	
Votante	**Voto**
Alice	$\{v_a\}_{pk}, ZKP_a$
Charlie	$\{v_a\}_{pk}, ZKP_a$
Bob	$\{v_b\}_{pk}, ZKP_b$

Tabla 16: Urna electoral Helios Voting con votante deshonesto

De una manera detallada [371], para una papeleta con el siguiente formato:

Input: Los parámetros criptográficos (p, q, g), la clave pública h, la lista de candidatos $t = (t_1 \dots t_l) \cup \{\epsilon\}$ y el voto v.

Output: El voto encriptado $(a_1, b_1), \dots, (a_l, b_l)$, las firmas de conocimiento $(\bar{a}_1, \bar{b}_1, \bar{c}_1, \bar{s}_1, \bar{a}'_1, \bar{b}'_1, \bar{c}'_1, \bar{s}'_1), \dots (\bar{a}_l, \bar{b}_l, \bar{c}_l, \bar{s}_l, \bar{a}'_l, \bar{b}'_l, \bar{c}'_l, \bar{s}'_l)$ y la firma de conocimiento $(\bar{a}, \bar{b}, \bar{c}, \bar{s}, \bar{a}', \bar{b}', \bar{c}', \bar{s}')$.

Pasos para la construcción del voto:

1. Si $v \notin t$ el *script* termina.
2. Codificar el voto v como un secuencia de bits. $\forall\, i$ tal que $1 \leq i \leq l$ sea $m_i = 1$ si $v = t_i$ y 0 en otro caso
3. Se encripta la secuencia de bits que representa el voto: $\forall\, i$ tal que $1 \leq i \leq l$,

$$(a_i, b_i) = (g^{r_i} \bmod p, g^{m_i} \cdot h^{r_i} \bmod p)$$

   Donde $r_i \in_R \mathbb{Z}_q^*$.

4. $\forall\, i$ tal que $1 \leq i \leq l$, sea $(\bar{a}_i, \bar{b}_i, \bar{c}_i, \bar{s}_i, \bar{a}'_i, \bar{b}'_i, \bar{c}'_i, \bar{s}'_i)$ la prueba de conocimiento que demuestra que (a_i, b_i) contiene un 0 o un 1, es decir, que cada candidato recibe como máximo un voto (evitar el voto de un entero que suponga varias papeletas a un mismo candidato).
5. Sea $(\bar{a}, \bar{b}, \bar{c}, \bar{s}, \bar{a}', \bar{b}', \bar{c}', \bar{s}')$ la prueba de conocimiento para demostrar que el texto cifrado $(a_1 \cdot \dots \cdot a_l), (b_1 \cdot \dots \cdot b_l)$ contiene un 0 o un 1. Es decir, que como máximo un candidato recibe un voto.

La descripción del ataque es la siguiente:

Sea un elección cuyos candidatos son $t_1 \dots t_l$ y hay tres votantes con derecho de voto denominados id_1, id_2, id_3 siendo el último de ellos deshonesto.

Habiendo los votantes honestos enviado sus votos, el tablón sería de la siguiente manera:

$$id_1, ciph_1, spk_1, spk'_1$$
$$id_2, ciph_2, spk_2, spk'_2$$

En los que para $i \in \{1, 2\}$

$ciph_i = (a_{i,1}, b_{i,1}), \dots, (a_{i,l}, b_{i,l})$
$spk_i = (\bar{a}_{i,1}, \bar{b}_{i,1}, \bar{c}_{i,1}, \bar{s}_{i,1}, \bar{a}'_{i,1}, \bar{b}'_{i,1}, \bar{c}'_{i,1}, \bar{s}'_{i,1}), \dots (\bar{a}_{i,l}, \bar{b}_{i,l}, \bar{c}_{i,l}, \bar{s}_{i,l}, \bar{a}'_{i,l}, \bar{b}'_{i,l}, \bar{c}'_{i,l}, \bar{s}'_{i,l})$
$spk'_i = (\bar{a}_i, \bar{b}_i, \bar{c}_i, \bar{s}_i, \bar{a}'_i, \bar{b}'_i, \bar{c}'_i, \bar{s}'_i)$

$ciph_i$ es el i-esimo voto encriptado, spk_i demuestra que los textos cifrados $(a_{i,1}, b_{i,1}), \dots, (a_{i,l}, b_{i,l})$contienen un 0 o un 1 y spk'_i demuestra que $(a_{i,1} \cdot \dots \cdot a_{i,l}), (b_{i,1} \cdot \dots \cdot b_{i,l})$ contiene un 0 o un 1.

En primer lugar, se puede atacar la ausencia de independencia de voto:

El adversario puede observar el tablón y seleccionar un voto que ya esté presente:

$$id_k, ciph_k, spk_k, spk'_k$$

Siendo id_k el votante cuya privacidad se va a poner en compromiso, $\forall\, k \in \{1, 2\}$.

El adversario envía $ciph_k, spk_k, spk'_k$ y el tablón quedaría:

$$id_1, ciph_1, spk_1, spk'_1$$
$$id_2, ciph_2, spk_2, spk'_2$$
$$id_3, ciph_k, spk_k, spk'_k$$

Todos ellos representan votos válidos puesto que $spk_1, spk'_1, spk_2, spk'_2, spk_k, spk'_k$ todos contienen firmas válidas de conocimiento.

Una vez atacada la ausencia de independencia del voto, se puede violar la privacidad:

La adición homomórfica de los votos revela el recuentro encriptado $(a_{1,1} \cdot a_{2,1} \cdot a_{k,1}, b_{1,1} \cdot b_{2,1} \cdot b_{k,1}), \dots, (a_{1,l} \cdot a_{2,l} \cdot a_{k,l}, b_{1,l} \cdot b_{2,l} \cdot b_{k,l})$

Y teniendo en cuenta las desencriptaciones parciales, los textos cifrados pueden ser desencriptados para desvelar el número de votos de cada candidato.

Como habrá al menos dos votos para el candidato del votante id_k, queda también comprometida la privacidad del votante honesto que eligió la otra opción. En el caso expuesto, la existencia de dos votos idénticos en el tablón podría provocar que el ataque fuese detectado por algunas de las partes implicadas.

Por ello, en el mismo paper, se ofrece otra alternativa de ataque basada en la explotación de la maleabilidad de los votos (ver apartado 2.2.4.10i):

Dado un voto válido V1

$$(a_1, b_1), \dots, (a_l, b_l), (\bar{a}_1, \bar{b}_1, \bar{c}_1, \bar{s}_1, \bar{a}'_1, \bar{b}'_1, \bar{c}'_1, \bar{s}'_1), \dots (\bar{a}_l, \bar{b}_l, \bar{c}_l, \bar{s}_l, \bar{a}'_l, \bar{b}'_l, \bar{c}'_l, \bar{s}'_l), (\bar{a}, \bar{b}, \bar{c}, \bar{s}, \bar{a}', \bar{b}', \bar{c}', \bar{s}')$$

Los siguientes votos también lo son:

V2:

$$(a_1, b_1), \dots, (a_l, b_l) \quad , \quad (\bar{a}_1, \bar{b}_1, \bar{c}_1, \bar{s}_1 + q, \bar{a}'_1, \bar{b}'_1, \bar{c}'_1, \bar{s}'_1 + q), \dots, (\bar{a}_l, \bar{b}_l, \bar{c}_l, \bar{s}_l + q, \bar{a}'_l, \bar{b}'_l, \bar{c}'_l, \bar{s}'_l + q) \quad , \quad (\bar{a}, \bar{b}, \bar{c}, \bar{s}, \bar{a}', \bar{b}', \bar{c}', \bar{s}' + q)$$

y V3:

$$(a_{\pi(1)}, b_{\pi(1)}), \dots, (a_{\pi(l)}, b_{\pi(l)}),$$

$(\bar{a}_{\pi(1)}, \bar{b}_{\pi(1)}, \bar{c}_{\pi(1)}, \bar{s}_{\pi(1)}, \bar{a}'_{\pi(1)}, \bar{b}'_{\pi(1)}, \bar{c}'_{\pi(1)}, \bar{s}'_{\pi(1)})$, …, $(\bar{a}_{\pi(l)}, \bar{b}_{\pi(l)}, \bar{c}_{\pi(l)}, \bar{s}_{\pi(l)}, \bar{a}'_{\pi(l)}, \bar{b}'_{\pi(l)}, \bar{c}'_{\pi(l)}, \bar{s}'_{\pi(l)})$, $(\bar{a}, \bar{b}, \bar{c}, \bar{s}, \bar{a}', \bar{b}', \bar{c}', \bar{s}')$

Donde π es una permutación sobre $\{1, \dots, l\}$.

En el voto V2 se cambia la papeleta (por tanto, también el *hash* del voto en el tablón) pero no se modifica la elección del candidato.

En cuanto a V3, permite emitir a propósito un voto distinto a alguien de quien el atacante conozca la intención de su sufragio.

Ambas tipologías de ataque son especialmente poderosas cuando en el tablón aparecen *hashes* del voto en lugar de los votos completos, puesto que los *hashes* serán distintos.

Ese es el caso de Helios 3.0, por lo que [371] consiguió demostrar fallos de seguridad de la versión más actual del sistema VER en el momento del ataque.

Las soluciones que aportaron iban en la línea de linkar unívocamente cada voto con su votante. En concreto a través de una credencial privada del votante, como puede ser añadiendo su identificador a las pruebas de conocimiento para garantizar que los votos están bien formados.

En esa línea, el protocolo de voto Civitas [67, 73] basado en el trabajo de Juels et al. [63] requiere ese vínculo entre el voto y las credenciales privadas del votante en lo que se ha dado a conocer como *"eligibility verifiability"* o verificabilidad de la elegibilidad [367]: cualquiera puede verificar que cada voto publicado en el tablón fue enviado por un votante autorizado y como máximo un voto por votante es contabilizado.

Como se apunta en [377], la versión oficial más reciente de Helios Voting sigue presentando vulnerabilidades en la privacidad del voto, por lo que han ido surgiendo variantes no desarrolladas por su autor original Ben Adida, tratando de solucionar desde distintas perspectivas las debilidades de Helios en la materia.

Ninguna de ellas se corresponde con la versión funcional de Helios Voting en su sitio oficial, si bien es perfectamente plausible que sus avances e investigaciones se tengan en cuenta en futuras implementaciones o como "banco de pruebas" para futuros sistemas de VER. A continuación se repasan las más relevantes:

5.2.2.3 Helios con recuento a través de Mix-nets [375]

Tal y como explican los autores Bulens et al., la versión original de Helios así como sus sucesivas actualizaciones presentan una serie de limitaciones tales como la complejidad en el cálculo de la validez de los votos cuando aumenta el número de candidatos (realizado en el equipo del votante) o el hecho de que las pruebas de validación deben modificarse para cada tipo de regla de relleno de votos.

En consecuencia, proponen una variante de Helios en la que el recuento de votos se realiza utilizando una modificación del esquema T2H2 de Shoup y Gennaro [397] que preserva la seguridad en la emisión y contiene embebido un cifertexto homomórfico ElGamal, manteniendo la eficiencia del esquema original.

Para mejorar la seguridad del sistema, en lugar de responsabilizar a los mismos administradores (*trustees*) del mezclado y la desencriptación, los autores separan ambos roles, asignando unos distintos a cada una de las acciones.

Se parte de las premisas de que los administradores de desencriptación no violan la privacidad del votante (podrían hacerlo si colaborasen entre ellos para desencriptar los votos antes de ser mezclados) y los administradores de mezclado mantienen en secreto sus permutaciones.

Además, en su sistema no se eligen los administradores de mezclado previamente a fin de mejorar la seguridad. Con ellos se aprovecha el hecho de que no tienen ninguna clave que generar antes del comienzo de las elecciones (a diferencia de los administradores de desencriptación, que sí deben generar la clave privada conjuntamente).

Por lo que respecta al criptosistema elegido, los autores realizaron pruebas sobre varios, decantándose al final por una solución personalizada basada en el esquema de umbral TDH2 [397], el cuál es seguro contra ataques CCA2 (referirse al punto 2.2.4.10i) si se acepta que DDH (punto 2.2.4.10c) es "irresoluble" en un tiempo razonable dentro del modelo del oráculo aleatorio.

La necesidad de personalizar el esquema TDH2 se explica puesto que su implementación original está basada en *Hash*-ElGamal, el cuál no es homomórfico. En concreto, los textos cifrados presentan el formato $(g^r, m \oplus H(h^r))$ donde la función *H* está modelizada como un oráculo aleatorio.

La modificación que se introduce para la presente variante consiste en aplicar el esquema de encriptación estándar de ElGamal, el cuál sí presenta propiedades homomórficas.

En lo que respecta al mezclado de votos, utilizan la prueba de Terelius y Wilkström [398]. Los autores explican que su implementación consistió en unas 200 líneas de código en Python, utilizando el *Python Cryptography Toolkit* para la generación de aleatoriedades. En términos prácticos, dicha implementación permite el mezclado de 25 votos por segundo utilizando un *"portátil estándar"* y un módulo *p* primo de 2048 bits.

Experiencias reales de Helios basado en mix-nets: Los autores realizaron dos pruebas reales con su versión de Helios, una de ellas con la implementación finalmente elegida y otra con el criptosistema de Cramer-Shoup [399]. Ambas tuvieron lugar en el ámbito universitario, en concreto para que los alumnos eligiesen a sus representantes en el consejo general de estudiantes, así como en el respectivo consejo de estudiantes de cada facultad.

Se ofreció asimismo la posibilidad de votar en papel durante los 2 días siguientes a los 4 habilitados para el VER, incluso a aquello estudiantes que quisiesen votar en ambos formatos. En ese caso, el último voto (en papel) prevalecería.

El formato de voto en ambos casos era especialmente grande, con los candidatos organizados en listas de hasta 127 y hasta 259 candidatos en una única papeleta.

El número total de votos emitidos a través de Helios fue de 2.564 y 3.016 respectivamente. En el paper no se detallan más estadísticas como tiempo total de recuento de votos, desencriptación o mezclado, si bien se extraen dos conclusiones:

- Los autores se decantan por el criptosistema HDTH2 [397] frente a Cramer-Shoup [399] por la mayor sencillez del primero en tareas de auditoría y recuento. En concreto, HDTH2 permite verificar la independencia y validez de los votos inmediatamente sin esperar a que se haga pública la clave privada sk^A, permitiendo abordar los problemas en el momento en el que surgen y no en la fase de recuento. Además, simplifica el rol de los administradores al abrir y publicar sus porciones de sk^A.
- En lo que respecta a la comparativa de recuento homomórfico o con mixnets (la variante propuesta), los autores reconocen que su variante entraña una complejidad *"remarcadamente mayor"* respecto a la versión estándar de Helios. Por complejidad entienden no únicamente la carga computacional sino también cuestiones organizacionales: puesta en marcha de las máquinas de las autoridades, transferencia de datos de una manera segura, trazabilidad de la confidencialidad de las claves etc. En cada una de las dos pruebas, el recuento duró *"varias horas"* (no se especifica más), tardando bastante más de los que los votantes esperaban.

En conclusión, se trata de una interesante experiencia pese a que los autores recomiendan la utilización del recuento homomórfico cuando sea posible. El hecho de se hayan realizado dos pruebas reales sin duda aumenta el valor del trabajo.

El artículo concluía con una serie de acciones futuras tales como integrar la solución propuesta para que en un futuro el organizador pudiese elegir la metodología de Helios a utilizar o el desarrollo de una interfaz para convertir el formato de datos de los autores a otro que pudiese ser manejado por *Verificatum*. Desafortunadamente, en el momento de escribir estas líneas ninguna de las acciones arriba enumeradas ha tenido continuidad.

5.2.2.4 Helios con credenciales o Helios-C [377]

La variante Helios-C es una aplicación práctica del trabajo de Cortier et al. en [377] sobre la transformación de sistemas de voto de verificabilidad débil o *"weak verifiability"* en sistemas de verificabilidad fuerte o *"strong verifiability"*.

En concreto, los autores entienden como verificabilidad fuerte:*"verificabilidad individual y universal contra tablones y autoridades de registro no simultáneamente deshonestos"*. Como definiciones de verificabilidad individual y universal toman las propuestas por Juels et al. en [104].

La necesidad de Helios-C viene por el hecho de que Helios en su última versión oficial, no puede garantizar la verificabilidad debido al *"ballot stuffing"* o relleno de votos. En [155, 379] se realiza un pormenorizado análisis de los ataques a Helios basados en la debilidad de la heurística Fiat-Shamir como prueba de conocimiento y en los problemas de maleabilidad derivados de la misma, los cuales impiden probar la verificabilidad de Helios. Otro factor que contribuye a ello es la ausencia de pruebas de verificabilidad en un modelo computacional de una forma automatizada, como se ha comentado anteriormente en el presente punto 5.2.

Como primer paso para solucionar dicho problema, los autores plantean una construcción general que transforma un sistema de VER en el que se requiere simultáneamente a las autoridades de registro y al tablón ser honestos (verificabilidad débil) por otro en el que únicamente se requiere que no sean deshonestos simultáneamente (verificabilidad fuerte). A dicha construcción menos exigente para ser verificable se le denomina *"verificabilidad de elecciones con menores precondiciones de confianza"*.

Para ello, crean una nueva autoridad denominada *"autoridad de registro"*, encargada de suministrar a cada votante una credencial o clave de firma, que a su vez tiene una parte pública o clave de verificación. Con ello, se reduce el poder del tablón, que tradicionalmente era el único con capacidad de controlar quién tiene derecho de votar. Como medida adicional, proponen borrar los registros una vez haya cumplido con su función.

Posteriormente y basándose en el nuevo esquema, se presenta Helios-C o bien Helios con credenciales. Por motivos de simplicidad, en su implementación proponen una votación del tipo sí/no o 1/0 así como un canal autenticado entre el votante y el tablón.

Las primitivas criptográficas utilizadas por Helios-C son:

- El criptosistema IND-CPA ElGamal [64] $D = (KeyGen, Enc, Dec)$ en un grupo $\mathbb{G}$ en el que se cumple DDH [25, 188],
- El esquema de firma Schnorr (2.2.4.10h y [383]) $S = (SKeyGen, Sign, SVerify)$ sobre el grupo $\mathbb{G}$,
- La NIZKP de Chaum-Pedersen [400] $DisjProof_H(g, pk, R, S)$ para demostrar que (R, S) encripta g^0 o g^1
- El sistema de NIZKP [401] $EqDl_G(g, R, vk, c)$ para probar que $log_g vk = log_R c$ para $g, R, vk, c \in \mathbb{G}$.
- H y G son funciones *hash* en $\mathbb{Z}_q$.

Helios-C consiste en 8 algoritmos:

$\mathcal{V}^{heliosc} = (Setup, Credential, Vote, Validate, VerifyVote, Box, Tally, Verify)$, que se definen en detalle en [377].

Por lo que respecta a su implementación práctica, se encuentra disponible en [402], si bien su última actualización data de 2013.

Para la generación y envío de las credenciales, los autores usaron una tercera parte y las enviaron por correo ordinario (en el paper no se especifica más). Para evitar que los votantes tuviesen que copiar claves excesivamente largas, se les hacía llegar el factor aleatorio de generación, de entre 12 a 15 caracteres alfanuméricos de longitud.

Helios-C requiere a los votantes firmar sus votos, introduciendo un retardo añadido respecto a la versión estándar de Helios. En la siguiente tabla se cuantifican dichos retardos según la acción concreta y el número de candidatos para un ordenador con un procesador Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz sobre Firefox 18:

Num. Candidatos	*2*	*5*	*10*	*20*	*30*	*50*
T. formación voto	600	1197	2138	4059	6061	9617
Firma voto	196	215	248	301	358	484
Verificación firma	< 10	< 10	< 10	< 10	< 10	< 10
Verificación voto	110	210	390	720	1070	1730

Tabla 17: Retardo añadido de Helios-C respecto a Helios en milisegundos.
Fuente: Elaboración propia a partir de [377]

Las dos primeras filas se refieren a acciones que tienen lugar en el ordenador del votante. Se aprecia que a medida que aumentan los candidatos, los retardos empiezan a ser relevantes.

Los autores especifican que la prueba se realizó sobre una simulación de elección con 30 votos aproximadamente e indican que el siguiente paso sería realizar experimentos similares con un mayor número de opciones de voto, con varios administradores o incluso con técnicas de umbral o *threshold*.

También introducen la posibilidad de que sea el propio votante quien genere sus credenciales para eliminar el riesgo de que la autoridad de registro sea corrupta. Argumentan también que por cuestiones prácticas la opción elegida es la preferible, al no requerir al votante realizar una acción crítica para la seguridad de su voto careciendo de los conocimientos necesarios.

Por último, avanzan como áreas de investigación futuras la posibilidad de concebir una definición de verificabilidad en esquemas que no admiten recuentos parciales así como el diseño de un sistema de VER sin premisas de partida en cuanto a la honestidad de las partes implicadas.

Por desgracia, la variante de Helios con credenciales supuso un muy interesante piloto puntual sin continuidad, puesto que su última actualización data de 2013, cuando tuvo lugar su desarrollo original.

No obstante, Helios-C aporta un enfoque interesante y que se repite en ocasiones en Helios y en otros sistemas de VER más académicos: líneas de investigación con mayor o menor continuidad pero que contribuyen a explorar mejoras en entornos menos exigentes que unas elecciones reales.

Dichas líneas de investigación en ocasiones terminan cristalizando en sistemas de VER propios y en otras no pasan de un esfuerzo puntual; pero sin duda constituyen una suerte de "banco de pruebas" de un indudable valor en la mejora del voto electrónico.

5.2.2.5 KTV-Helios [360, 361, 381]

Como se ha explicado anteriormente en el apartado y en [371, 377], uno de los problemas sin resolver de la versión oficial de Helios es su vulnerabilidad respecto a la privacidad de participación (la información disponible públicamente no debería revelar si un votante ha votado o no). Es más, no existe una definición formal de privacidad de participación.

Su interpretación también depende del país como se apunta en [360] y en el apartado 2.1 de la tesis: en algunos países como en Australia el voto es obligatorio mientras que en Alemania o Suiza entre otros el hecho de votar o no es secreto. Por ello, la relevancia de la privacidad de participación es aún más importante si cabe en ciertos países.

La solución propuesta por Kulyk, Teague y Volkamer (de ahí el nombre KTV-Helios) gira sobre la idea de que cualquier votante puede añadir votos nulos, los cuales deben cumplir con dos propiedades: no deben ser tenidos en cuenta en el recuento y deben ser indistinguibles de los votos válidos en el tablón.

Para ello, los votos nulos son la encriptación de 1 y se utiliza una infraestructura de clave pública o PKI en lugar de un mecanismo basado en credenciales dedicadas como Civitas [67, 73]. También se presupone un canal anónimo.

La mayor diferencia con respecto al protocolo clásico de Helios es que se permiten varios votos referenciados a un mismo votante (aunque únicamente uno de ellos válido):

Para un conjunto de candidatos $\{c_1, \dots, c_L\}$ y un par de claves de ElGamal generadas y publicadas distribuidamente (g, h):

Un voto $c \in \{c_1, \dots, c_L\}$ se encripta de la siguiente manera:

$$Enc(c) = (g^r, c \cdot h^r)$$

Y se envía como voto final: $(Enc(c), pk, P)$siendo pk la clave pública del votante y P la prueba de elegibilidad.

El tablón valida P y todos los votos con P validada aparecen publicados.

Los votos de relleno pueden ser enviados por cualquier votante. Su formato es: $(Enc(1), pk, P)$.

El tablón por tanto quedaría de la siguiente manera:

Votante	**Votos**
pk_1	$v_{1,1}, \dots, v_{1,m_1}$
…	…
pk_n	$v_{n,1}, \dots, v_{n,m_n}$

Tabla 18: Tablón antes del recuento en KTV-Helios [361]

Para cada votante pk_i, se computa la siguiente multiplicación para determinar el cifertexto definitivo de su voto:

$$v_i = \prod_{j=1}^{m_i} v_{i,j}$$

Puesto que los votos nulos son la encriptación de un 1, en la multiplicación son un elemento neutro y por tanto no modifican el sentido del voto.

Después de la multiplicación, el tablón queda de la siguiente manera:

Votante	**Votos**	**Voto final encriptado**
pk_1	$v_{1,1}, \dots, v_{1,m_1}$	V_1
…	…	…
pk_n	$v_{n,1}, \dots, v_{n,m_n}$	V_n

Tabla 19: Tablón después de la eliminación de votos nulos en KTV-Helios [361]

Posteriormente se mezclan los votos con una mixnet verificable para romper el vínculo votante-voto y obteniendo $V_1', \dots, V_n'$, listos para ser desencriptados.

Antes de ello, se realiza una última comprobación con PET (*Plain Equivalence Tests* [110]) para eliminar los votos nulos o inválidos:

Si dos cifertextos encriptan el mismo texto original, el PET devuelve un 1. Es decir; se encripta la lista de opciones de voto usando el factor aleatorio *r* y se obtiene:

$$E = \{e_1 = Enc(c_1), \dots, e_L = Enc(c_L)\}$$

$$\forall\, i = 1, \dots n \text{ se computa } PET\left(V_i', e_j\right), j = 1, \dots, L$$

Si $\exists\, j$ tal que $PET\left(V_i', e_j\right) = 1$, se añade c_j al recuento. En caso contrario, se descarta V_i' como un voto nulo o inválido.

Con ello, los autores afirman que han implementado un nuevo método para conseguir privacidad de participación así como la verificabilidad de elegibilidad si bien también apuntan las debilidades del mismo:

- Para la verificabilidad de elegibilidad las presunciones de seguridad son: la autoridad de registro es honesta (el listado de votantes con derecho de voto es correcta), la autoridad de certificación es honesta (la lista de pk's es correcta) y las claves de firma de los votantes no son filtradas al adversario (ni por el votante mismo ni por un equipo infectado que envía votos fraudulentos).
- En cuanto a la privacidad de participación, los *proxys* de votos de relleno son honestos y el equipo del votante no está infectado y es fiable.

A mayores recomiendan: la implementación de un algoritmo formal de inserción de votos de relleno de tal manera que asegure las propiedades sin sobrecargar el sistema así como el desarrollo de una interfaz que mejore la usabilidad y fácil comprensión del protocolo, bastante complejo para los no expertos.

Para completar el presente apartado, conviene destacar que en 2016 se ha producido un ulterior avance en la línea del KTV-Helios por parte de Bernhard et al. en [381]: en el paper original del KTV-Helios, se enumeran las propiedades que cumple (verificabilidad de la elegibilidad y privacidad de participación) pero no se demuestran.

En ese sentido, [381] trata de dar una definición formal de la privacidad de participación probabilística como paso previo a la demostración de la misma en KTV-Helios. Por otra parte, prueban que la citada versión de Helios proporciona verificabilidad contra ataques por parte de un tablón deshonesto de acuerdo a la definición de verificabilidad en [377].

En lo que respecta a la verificabilidad, los autores presuponen que: el registro de votantes con derecho de voto y la PKI (apartado 2.2.4.10g) son honestos. Además, las claves secretas de los votantes honestos no son filtradas a ningún atacante. En unas elecciones reales, podría argumentarse que dichas precondiciones son algo voluntaristas y no tan fáciles de garantizar en la práctica.

Aún así, el esfuerzo realizado es considerable y de gran utilidad, llegando a la definición siguiente sobre verificabilidad para un esquema de voto S (referirse a [381 para la demostración completa):

Un esquema de voto S es verificable si la probabilidad de éxito $Pr\left[Exp_{A,S}^{ver-b} = 1\right]$ *es insignificante para cualquier adversario en tiempo polinomial probabilístico o PPT.*

Por lo que respecta a la privacidad del voto, se apoyan en la definición de [403] basada en juegos (*game-based*). Como explica Cortier en [362], es difícil de obtener una única definición universal y actualemente co-existen varias. La obtención de una única válida para todos los casos es todavía un reto en el VER.

En el caso de KTV-Helios, la definición que se prueba de privacidad de voto es la siguiente:

Un esquema de voto S satisface la privacidad de voto si existe una función de simulación de PPT $SimProof(BB, R)$ *de tal forma que para cualquier adversario PPT la cantidad*

$$Adv_{A,S}^{bpriv} \coloneqq \left| Pr\left[Exp_{A,S}^{bpriv,0} = 1\right] - Pr\left[Exp_{A,S}^{bpriv,1} = 1\right] \right|$$

es insignificante (referirse al paper original para la demostración completa).

Por último, en lo que respecta a la privacidad de participación, la definición que prueban los autores para KTV-Helios es la siguiente:

El esquema de voto S en la elección con parámetros (N, n_h, L, p) *consigue privacidad de participación-δ dado un subgrupo* C_s *de aversarios PPT, si para cualquier adversario* $A \in C_s$ *y un votante honesto* id^A *se cumple que:*

$$Pr\left[Exp_{A,S,id^A}^{ppriv,0} = 0\right] - Pr\left[Exp_{A,S,id^A}^{ppriv,1} = 0\right] - \delta$$

es insignificante en el parámetro de seguridad (referirse a [381] para la demostración completa).

Por tanto, con las tres definiciones aquí reproducidas, Bernhard et al. [381] prueban que KTV-Helios satisface la privacidad de voto y la verificabilidad contra ataques contra tablones deshonestos según las definiciones de [403, 377] así como una definición probabilística de la privacidad de participación.

Como trabajos futuros, los autores apuntan a avanzar en el campo de la resistencia a la coerción (apartado 2.2.3 de la presente tesis), tratando de formalizar una definición para KTV-Helios así como a la implementación de un sistema con sus componentes críticos distribuidos para aumentar la seguridad del esquema.

En resumen, KTV-Helios conforma una línea de investigación reciente que trata de poner solución a varios de los problemas de la versión actual de Helios (privacidad de voto y participación así como la verificabilidad [377]).

En ese sentido, aporta sin duda avances interesantes desde el punto de vista de las definiciones probabilísticas, si bien es necesaria una implementación práctica de las principales conclusiones dentro de un sistema de VER totalmente operativo para poder afirmar que realmente cumple con todo lo que promete.

5.2.2.6 Helios distribuido o *Distributed-Helios* [370]

En este último apartado dentro de las versiones modificadas del Helios original se aborda un problema mucho más práctico con respecto a los anteriores análisis del Helios-C o el KTV-Helios, de un perfil más teórico/académico.

Chung el al. en [370] plantean el caso muy real de la posibilidad de un ataque DoS (*Denial of Service*) a Helios y cómo reforzar su implementación para hacerla resistente a dicha tipología de ataques (apartado 2.4 de la presente tesis).

Como los autores indican, la inmensa mayoría de los sistemas de VER funcionales cuentan únicamente con un servidor de recogida de votos a modo de urna, haciéndolos vulnerables con el riesgo de que colapsen en el transcurso de las elecciones (como ya sucedió en uno de los casos analizados [339, 340]).

Para solucionar dicha vulnerabilidad, añadieron capacidad multiservidor a Helios; en concreto 3, 5 y 7 que se comunican utilizando el protocolo de Paxos desarrollado originariamente por Lamport [404].

Paxos es un protocolo de consenso de origen teórico pero que en la actualidad tiene uso en multitud de sistemas *Chubby*, desarrollado por Google y utilizado entre otros por *Google File System* y *Bigtable* [405].

En él, los miembros o servidores se comunican entre ellos a través de mensajes que esperan un tiempo aleatorio antes de ser enviados o que pueden incluso caerse por el camino. Los miembros deben por tanto ponerse de acuerdo en determinados valores bajo los cuales el protocolo Paxos permite al sistema continuar funcionando incluso con un determinado número de fallos [370, 404, 405].

En el caso de Helios distribuido, el objetivo es conseguir que el sistema de VER siga funcionando pese a sufrir un ataque DoS. Para ello, los autores partieron de la última versión disponible de Helios Voting [396], sobre la que modificaron la opción de acceso a través de Google para su prueba piloto.

También tuvieron que modificar la función *save()* de Django para salvar la base de datos en todos los servidores de las distintas pruebas del piloto:

```
def actualsave(classtype, obj):
      for currentserver in servers:
        super(classtype,obj).save(
                using=currentserver[0])
```

Figura 39: Modificación de la función *save()* en *Distributed-Helios* [370]

A mayores, para facilitar la comunicación entre servidores, los autores utilizan *Twisted*, un recurso *open-source* basado en eventos para Python que además de ser robusto y flexible es asíncrono, por lo que la espera de mensajes no bloqueará código.

En cuanto al piloto en sí, se realizaron pruebas del *Distributed-Helios* con 3, 5 y 7 servidores así como con la versión estándar de Helios. La elección del número de servidores responde al número real que utiliza el protocolo *Chubby*, mencionado previamente.

En primer lugar, los autores tuvieron que modificar de nuevo el protocolo para introducir una latencia en la ejecución de ciertas operaciones de carga de páginas para evitar que éstas cargasen antes de que Paxos hubiese terminado de ejecutarse.

Una vez solventada esa situación, se realizaron simulaciones de elecciones en las cuatro configuraciones para cuantificar el retardo que conlleva convertir Helios en resistente a ataques de DoS, al menos parcialmente.

Los resultados fueron los siguientes:

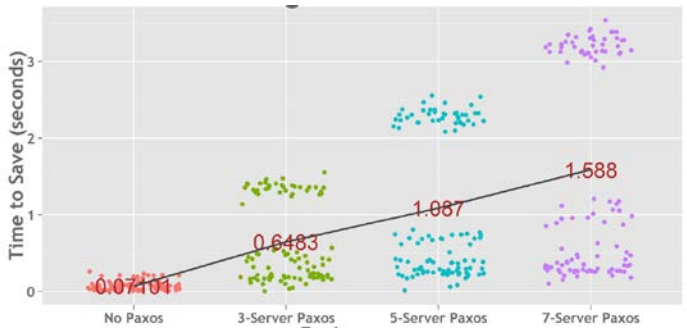


Figura 40: Tiempos de salvado para distintas configuraciones de Helios [370]

En el paper original se detalla en profundidad cada tiempo en función de la operación concreta, pero a modo de resumen baste decir que existe una progresión lineal de unos 0.23 segundos de retardo por cada servidor que se introduzca y se gestione con Paxos.

En porcentaje, 3 servidores conllevan un incremento del 912.97% en el tiempo de ejecución mientras que para el caso de 5 y 7 servidores los aumentos son del 1530.9% y 2236.36% respectivamente. Utilizando la función *lm ()* de modelo linear, se obtiene la recta de regresión definida por la ecuación:

$$y = 0.2352x - 0.06885$$

donde *x* es el número de servidores e *y* es la mediana del incremento de tiempo.

En cuanto a las limitaciones, *Distributed-Helios* no es resistente al problema de los generales bizantinos [406] (aquellos provocados por servidores corruptos o mensajes corruptos). En otras palabras, el protocolo presupone que los servidores y los votantes son honestos.

Pese a ello, se trata de un ejercicio de mejora de Helios con una indudable aplicación práctica que busca resolver una debilidad concreta del esquema original.

Los propios autores apuntan que se trata únicamente de un primer paso en la protección de sistemas de VER frente a ataques DoS, pero su importancia es sin duda notable.

En cuanto a las recomendaciones finales, no abogan por utilizar *Distributed-Helios* en elecciones a pequeña escala con bajo riesgo de ataque, puesto que los retrasos (de hasta 3 segundos por operación en el caso de 7 servidores) pueden hacer el sistema poco práctico. En caso de prevalecer la seguridad, D-Helios es la mejor opción en palabras de los autores.

Para concluir, las líneas futuras de investigación incluyen ampliar el modelo de ataques para incluir los fallos bizantinos y mejorar la eficiencia introduciendo variantes más eficientes de Paxos como *Fast Paxos* [407] o *The Raft* [408].

Se debe tener presente no obstante que incluso las pruebas realizadas por Google para conseguir un protocolo eficiente contra ataques DoS han llevado miles de líneas de código sin haberlo conseguido, pese a ocupar muchos de ellos solo una hoja de pseudocódigo.

Como en otros muchos casos, encontrar un equilibrio adecuado entre seguridad y recursos, adaptado a cada situación es la solución más recomendable.

Una vez explicado en profundidad el sistema de VER con sus principales actualizaciones y variantes, se está en disposición de aplicar la metodología de evaluación definida en el capítulo 4 para analizarlo en detalle.

5.2.3 Análisis

Para el análisis del sistema de VER Helios con la metodología de evaluación definida y explicada en los capítulos anteriores, cabe destacar previamente que se va a tener en cuenta la última versión oficial disponible en [394, 396].

Ello es debido a que es ésta la que ha sido utilizada en un mayor número de elecciones y ha sido desarrollada por su autor original. Incluye sucesivas mejoras para subsanar las vulnerabilidades descubiertas por Estehghari et al. en [372, 374].

En cuanto a las mejoras y actualizaciones explicadas en el apartado anterior como Helios-C, KTV-Helios, *Distributed Helios* o Helios con recuento por mix-nets; o bien no han sidos llevadas a la práctica (KTV-Helios) o bien la experiencia ha sido limitada (Helios-C, *Distributed Helios* y Helios con recuento por mix-nets) y su continuidad es una incógnita.

Una vez hecha la aclaración, se recuerda que los 12 criterios que conforman la metodología de evaluación para sistema de voto electrónico son los siguientes:

Verificabilidad Extremo a Extremo ($E2E_v$), resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

1. Verificabilidad extremo a extremo

Como ya se apuntó en el apartado 2.2.2, *"Un sistema de VER es E2Ev si cada voto es i) emitido como estaba previsto ii) guardado como se ha emitido y iii) contado como se ha guardado."* y los votos han sido emitidos por votantes con derecho a votar (verificabilidad de elegibilidad) [51, 93, 77, 3, 359, 369, 389]. O en inglés, más habitual en el campo: *i) Cast as intended, ii) Recorded as cast, iii) Counted as recorded* junto con *elegibility verifiability.*

En el caso de Helios, existe división de opiniones. Por una parte, en el paper original [1] y Jonker et al. en [359] justifican que es verificable extremo a extremo.

En la postura contraria se sitúan Cortier et al. en [377] cuando afirman que Helios no puede considerarse totalmente verificable debido a que un tablón deshonesto podría añadir votos sin que se pudiese evitar o comprobar.

En realidad, lo que proponen los autores de [377] es introducir Helios-C como alternativa que permite la verificabilidad con unos requerimientos menos exigentes puesto que parten del supuesto de que el tablón y la autoridad de registro no son deshonestos simultáneamente, pero en cualquier caso su esquema también tiene una serie de precondiciones que pueden cumplirse o no en la práctica.

En esta misma línea se pronuncian Smyth et al. [409], quienes aducen que incluso Helios-C no es verificable de acuerdo a la definición que ellos aportan.

En [241], Arapinis et at. proponen una modificación de las herramientas de verificación automática de propiedades de equivalencia aKiSs [391] y ProVerif [392] para evaluar diferentes versiones simplificadas de Helios [1]. La modelización que han realizado del sistema de VER es muy simple, no incluyendo ZKP y restando por tanto validez práctica al experimento. Aún así, la línea abierta es interesante y va en concordancia con lo explicado en el apartado 2.2.2 de la modelización de la definición de verificabilidad con el fin último de automatizar su evaluación (de hecho, hay un autor común en [241] y [365]).

En resumen, dependiendo de la definición que se tome de verificabilidad (más o menos restrictiva pero en ningún caso universalmente formal o correcta como argumentó Cortier en [362]), Helios puede considerarse E2Ev o no.

Si se tiene en cuenta la concepción original de Helios y lo que se establece en su página web oficial, se trata de un sistema ideado para elecciones en entornos con bajo riesgo de

coerción tales como clubs, comunidades online, o asociaciones de estudiantes, no siendo suficientemente seguro para elecciones vinculantes en el ámbito político.

Teniendo en cuenta el último párrafo y siempre que se limite el uso a ese ámbito, Helios se puede considerar E2Ev con condiciones, al no ser estrictamente necesario ceñirse a la definición de verificabilidad más exigente [377, 409].

Conclusión: **Δ,** cumple bajo ciertas premisas

2. Privacidad/resistencia a la coerción

Por lo que respecta a la privacidad entendida como resistencia a la coerción (RC) y definida por Juels et al. en [104], ésta implica que un votante no puede colaborar con un coaccionador para obtener información de cómo votó, incluso en el caso de que el votante quiera voluntariamente corromper su voto.

En el caso de Helios, la privacidad ha sido uno de los principales caballos de batalla como se ha explicado en detalle en el punto 5.2.2 [360, 371, 379, 381].

En el punto 5.2.2.2 se ha detallado la familia de ataques contra la privacidad del voto [371], que en su modalidad más simple permite a un atacante copiar un voto del tablón y reenviarlo como su propio voto. También en [381] se llama la atención sobre la ausencia de privacidad de elección (no se puede ocultar quién ha participado en los comicios).

Además, en [379], Salamonsen detalla 5 tipos de ataques relacionados con la privacidad del voto en Helios, proponiendo en 2014 una solución que, siendo en teoría correcta, en la práctica implica que la encriptación de un único voto lleve entre 2 y 5 horas con 24 CPUs *"state of the art"*. Por ello, su implementación real en unas elecciones no resulta viable.

De hecho, es el propio Adida quien apunta que Helios debe ser utilizado únicamente en entornos con bajo riesgo de coerción, porque no garantiza resistencia frente a ella.

Las iniciativas que mejoran en esa línea como KTV-Helios [361], no están implementadas en la práctica y son los propios autores los que establecen que únicamente ofrecen un "grado de resistencia a la coerción", si bien es una línea en la que deben seguir trabajando.

Por todas las razones expuestas, no se puede considerar a Helios como RC.

Conclusión: **×,** no cumple

3. Inviolabilidad (I-*n*)

Los ejes sobre los que se sustenta la inviolabilidad en esta tesis se detallan en el apartado 2.3c y tienen asignados los códigos I-*n*. Un resumen de ellos son:

- Un control de accesos robusto. (I-1) (I-2)

- La existencia de protocolos específicos de *risk assessment* y *threat modeling*. (I-5)
- La disponibilidad de copias de seguridad aisladas y *offline*. (I-3)
- Una sólida política de distribución de permisos y responsabilidades, con especial énfasis en los nodos críticos. (I-4)
- La modularidad con el fin de confinar en lo posible los errores/ataques. (I-6)
- La correcta actualización periódica de los puntos previos. (I-7)

En cuanto al control de accesos, Helios permite software de terceros para autenticarse (Facebook, Google), prácticas poco recomendadas puesto que se pierde control sobre una parte fundamental de cualquier sistema de VER. (I-1)

Respecto a los protocolos de seguridad, *risk assessment, threat modeling* y las copias de seguridad (I-3) (I-5), no se hace mención específica en la documentación oficial [395].

La distribución de permisos y responsabilidades (I-4) se implantó a partir de la versión 2.0, especialmente en la fase de desencriptado con la existencia de varios *trustees* si el organizador de la elección así lo selecciona. Una debilidad se localiza en el servidor que recibe lo votos puesto que sus funciones no están distribuidas y es vulnerable a un ataque del tipo DoS (lo que propició el desarrollo de *Distributed-Helios* [370] y 5.2.2.6).

Finalmente, las últimas actualizaciones oficiales de la documentación datan del 29 de agosto de 2012 aunque existen actualizaciones posteriores del código. Además, el autor original siempre ha mostrado una destacable disposición a colaborar con los papers y artículos que tratan de mejorar el sistema (I-7).

En conclusión, Helios presenta una política de inviolabilidad que se podría catalogar de intermedia puesto que implementa prácticas parciales de distribución de tareas y su filosofía *open-source* hace que sea la propia comunidad científica quien lo evalua en profundidad proponiendo mejoras; las últimas de ellas en este mismo 2016.

Conclusión: 5/10 puntos +1 punto de bonus porque desde un principio Adida establece que Helios es un sistema para elecciones menores con escaso riesgo de coerción o de ser atacadas, delimitando pues su ámbito de utilización.

Total: 6/10 puntos

4. Usabilidad (U-*n*)

Para este criterio, los conceptos clave son: simplicidad y claridad en el proceso de votación (U-1), la existencia de versiones/adaptaciones para colectivos con discapacidad (U-2) (Consejo de Europa [54] y *"Convención de las Naciones Unidas sobre Derechos de Personas con Discapacidad"* [259]), así como la facilidad para un votante sin conocimientos de criptografía y seguridad de emitir correctamente su voto y que sea contabilizado (U-3) (U-4).

Los trabajos más destacados en materia de usabilidad en Helios son [231, 373, 380, 410]. Aunque cada uno de ellos desde su óptica, dos comentarios destacan sobre el resto: 1) la utilización de terminología demasiado técnica para el usuario (*"fingerprint", "encrypt", "audit", "verify", "check credentials"* etc.) y 2) el proceso de votación y verificación es poco claro, sin una guía clara de los pasos restantes ni sobre las diferencias en las etapas en comparación con el voto tradicional, haciendo que un 38% de los votantes en [231] y la mitad en [380] no fueran capaces de completar su proceso de votación correctamente.

En este último caso, cabe destacar que el botón *"help"* que aparece durante el proceso de votación únicamente abre una nueva ventana para enviar un correo electrónico, no aportando una solución ágil que permita resolver la cuestión durante la sesión de votación. Además, no existen versiones de Helios en otros idiomas mayoritarios aparte del inglés (español, francés, alemán etc.) pese a que presumiblemente contribuiría a una mayor expansión y alcance del mismo.

En el caso de los administradores, el hecho de que deban guardar la clave privada de ElGamal puede entrañar dificultad para alguien sin conocimientos técnicos previos (U-1) (U-5). También la circunstancia de que durante el recuento no aparezca ninguna pista sobre el avance del mismo puede provocar que algún administrador crea que no se está produciendo correctamente y aborte el proceso como se apunta en [380].

Entre los aspectos positivos, destaca que una mayoría de los votantes se sintieron muy cómodos utilizando Helios, valorando especialmente su conveniencia y comodidad, concluyendo que las ventajas compensan las dificultades de uso (la cuál consideraron que se reduciría con una formación previa).

Además, varios de los equívocos podrían evitarse mejorando la interfaz de usuario y el lenguaje utilizado, sin tener que entrar en la estructura de Helios.

Por todo ello, en el apartado de usabilidad se considera que, si bien no existe ningún problema insalvable o incluso de difícil solución, queda una importante labor a realizar puesto que no es aceptable que más de un tercio de los votantes no pudiesen completar el proceso de voto con éxito o que no exista una herramienta de ayuda más desarrollada.

Total: 4/10 puntos

5. Monitorización/auditoría (MA-*n*)

Como se explica en detalle en el punto 2.3e, en monitorización/auditoría se valora la existencia de un protocolo específico para dichas tareas (MA-2), así como de herramientas independientes generadoras de informes periódicos (*logs*) que no puedan ser borrados ni alterados (MA-4). Su almacenamiento deberá encontrarse físicamente separado del resto de servidores de las elecciones.

Debe existir también la figura de un auditor independiente a ser posible con atribuciones distribuidas para reducir riesgos de colusión (MA-1) (MA-8).

En el caso de Helios, el título del paper original es: *"Helios: Web-based Open-Audit Voting"*, enfatizando desde un primer momento la palabra auditoría. En realidad, el concepto está enfocado más hacia el hecho de que cualquier persona (votante o no votante) puede comprobar o verificar que los votos son correctos y que han sido contabilizados. Por ello, no existe una implementación concreta de una política de auditoría. En consecuencia, muchos de los códigos (MA-*n*) no son de aplicación.

El problema surge cuando, repasando las vulnerabilidades de Helios [18, 155, 371, 377] se constata que muchas de ellas están basadas en la capacidad que tiene el tablón de insertar votos fraudulentamente (*ballot stuffing*). Dicho de otra manera: el correcto funcionamiento de Helios depende de que ni el tablón ni la autoridad verificadora sean deshonestas. De ahí la gran importancia de una monitorización/auditoría de dichos elementos por parte de una autoridad externa.

La citada figura independiente de auditoría (MA-1) (MA-8) no está recogida en la implementación *oficial* de Helios, si bien sí se ha introducido en alguna de las elecciones más importantes en las que se ha utilizado [49], donde votaron más de 8.000 estudiantes con la versión 2.0. De hecho, se invitó a distintos expertos independientes a formar parte de la comisión de control del proceso electoral (MA-4) (MA-8).

Además, se habilitó un día entero para que cualquier participante pudiese auditar el tablón y formalizar cualquier de comentario o queja que tuviese (MA-6).

Adicionalmente, los organizadores encargaron a una empresa externa otra versión del código de auditoría en Python, mejorando su independencia y fiabilidad (MA-1) (MA-8).

Por último, se habilitó un *Service Desk* a disposición de los votantes para resolver cualquier duda que surgiese.

En resumen, [49] fue una "puesta de largo" real de Helios y por ello los organizadores (incluyendo al autor principal, Dr. Ben Adida), junto con las autoridades de la Universidad de Lovaina implementaron una serie de medidas extra que no forman parte del protocolo original al alcance de cualquier potencial usuario. El motivo fue reforzar la seguridad y auditoría para reducir en lo posible el riesgo de ataques y eventos inesperados.

En esa línea cabe hacer dos puntualizaciones:

- Los propios autores [49] apuntan que unas elecciones con VER no se limitan simplemente a instalar un software y esperar que todo vaya sobre ruedas. Cada proyecto es único y conlleva una serie de costes e implementaciones, implicando una necesidad de recursos específicos. La realidad es que esos aspectos (incluída una política de

auditoría/monitorización) no forman parte integral de la versión oficial de Helios a descargarse para organizar unas elecciones.

- Helios ha sido concebido desde un principio (y así lo ha aclarado el autor) como un sistema de VER para elecciones en entornos con bajo riesgo de coerción y en ámbitos menores. En consecuencia, no era tampoco de esperar un despliegue de medios para garantizar una auditoría independiente de primer nivel.

Por todo ello, el apartado de monitorización/auditoria de Helios no presenta un nivel de desempeño destacado. Su política se basa en permitir "auditar" la verificabilidad individual y universal, persistiendo desafortunadamente la posibilidad de que un tablón deshonesto y/o una autoridad de verificación puedan introducir votos fraudulentamente.

En consecuencia, se mantiene la necesidad de desarrollar una política de monitorización/auditaría independiente completa dentro del esquema original de Helios.

Conclusión: 3/10 puntos +1 punto extra porque Helios es un sistema para elecciones menores en ámbitos de bajo riesgo de coerción, como su autor ha siempre puntualizado.

Total: 4/10 puntos

6. Desarrollo software (DSW-*n*)

En el presente apartado, además de las condiciones habituales de diseño, implementación y documentación de ingeniería del software (DSW-1) se valoran una serie de aspectos propios del VER explicados en detalle en el punto 2.3f de la tesis. Algunos de ellos son:

El enfoque distribuido del software (DSW-2), la simplicidad de uso (DSW-3), la seguridad de acceso (evitando terceros programas) (DSW-10), la imparcialidad en las opciones mostradas (DSW-5), la compatibilidad del software (DSW-9), la correcta implementación de las primitivas criptográficas (DSW-11), la revisión del código por parte de expertos independientes (DSW-12) y las actualizaciones frecuentes (DSW-14).

Helios, con su enfoque totalmente *open-source*, un autor de referencia en el campo y un consejo de asesores de primer nivel, ha ejercido un fuerte efecto de atracción por parte de la comunidad científica.

Ello ha derivado en numerosos de artículos de investigación y capítulos dedicados detallando errores y/o ataques [18, 155, 371] a la vez que se proponen soluciones y mejoras que han redundado en un contínuo progreso de la herramienta [370, 360, 377, 381].

En ese sentido, Helios presenta un elevado nivel de cumplimiento en: diseño, implementación y documentación del software (DSW-1), imparcialidad (DSW-5), compatibilidad del software (DSW-9), sitio web seguro con sección de FAQ (DSW-4), cancelación del voto (DSW-8), uso de estándares abiertos (DSW-13), implementación de las primitivas

criptográficas (DSW-11) y especialmente en lo que se refiere a la revisión del código por parte de expertos independientes (DSW-12), fundamental para facilitar la búsqueda de errores y propiciar su mejora.

En cuanto a los apartados con margen de mejora, son los siguientes:

- Helios permite la identificación a través de terceros programas (Facebook y Google) lo cuál es muy poco recomendable al perderse control sobre una de las tareas primordiales de un sistema de VER (DSW-10).
- La usabilidad, como se ha explicado en este mismo punto 5.2.3 (DSW-3).
- El enfoque distribuido no está suficientemente pulido [411], lo que podría llegar a bloquear el recuento de las elecciones si alguno de los responsables no aparece o si sucede algún imprevisto (DSW-2).
- En los últimos 3 años, el ritmo de las actualizaciones se ha ido ralentizando y las mejoras han sido menos significativas (DSW-14).

En resumen, Helios presenta un desarrollo software sólido, favorecido por un equipo desarrollador de primer nivel junto con un enfoque *open source* que favorece revisiones exhaustivas por parte de la comunidad científica. Todo ello redunda en un software bien documentado, testado y revisado por numerosos expertos. Quedan no obstante una serie de campos a mejorar que, si bien no empañan sus fortalezas (más teniendo en cuenta el ámbito de uso de Helios), impiden que se le pueda otorgar una puntuación mayor.

Total: 7,5/10 puntos

7. Escalabilidad (E-*n*)

La idea fundamental consiste en que, en el momento en que se utiliza el sistema de VER, éste haya sido previamente testado tanto en capacidades software como ex_software (la infraestructura, los equipos, los recursos humanos, la logística y los costes) (E-1).

Idealmente, las pruebas deberían ser más exigentes que las elecciones que se vayan a realizar (E-3). Se debe evitar que se produzca un fallo o incluso interrupción en el proceso electoral por falta de planificación y pruebas.

Se deberá indicar también el tamaño o complejidad máxima de comicios que el esquema puede manejar (E-4) desde las dos vertientes: software (según la modalidad de elecciones así como las capacidades matemáticas y criptográficas) y ex_software (infraestructura, accesos, costes, logística, recursos humanos etc).

En el caso de Helios, uno de sus puntos fuertes más destacados es la claridad con la que expone su alcance y capacidades, no intentando abarcar más de lo que puede (E-4). Se centra en votaciones en entornos con poco riesgo y de pequeño tamaño. Su escalabilidad viene por tanto delimitada por esas dos componentes.

Las elecciones más grandes en las que se ha utilizado Helios en conocimiento del autor son las de la Universidad de Lovaina [49], con un total de casi 8.000 votos enviados. Ellas conformaron un proyecto de notable envergadura, con numerosos medios materiales y logísticos al alcance de organizadores y votantes. Además, se creó códido adicional adaptado como por ejemplo el encargado del procedimiento de recuento.

Finalmente, la experiencia fue un éxito y definió un nuevo rango de elecciones que Helios puede abarcar, mostrando de paso una interesante escalabilidad. Es cierto que ésta vino de unos medios especialmente generosos que no suelen estar al alcance de alguien sin los conocimientos técnicos necesarios queriendo organizar unas elecciones menores.

Por ello, la utilización directa de Helios entendida como la descarga del código fuente [394] y su posterior organización de unas elecciones sin llevar a cabo modificaciones presenta limitaciones de escalabilidad.

Su rango de utilización recomendable lo definen los procesos electorales que se llevan a cabo regularmente a día de hoy a través de Helios: las elecciones de la IACR [412] y las de representantes de estudiantes de la Universidad de Princeton [413], éstas últimas alojadas en la misma web oficial de Helios y utilizadas en varias ocasiones en cada curso académico.

En cuanto al tamaño y la complejidad, ambas presentan un total de votos emitidos que va de varios cientos hasta algo más de 1.000 votos en total, en un formato de elecciones simple, sin excesivos candidatos ni necesidad de decidir un orden entre ellos (lo cuál dispararía su coste computacional).

Por tanto, la escalabilidad inmediata probada de Helios llega hasta algo más de 1.000 votos. A partir de ahí, con los suficientes medios se ha llegado hasta censos de 25.000 votantes en elecciones poco complejas desde el punto de vista criptográfico (pocos candidatos sin ordenar).

En el caso de la escalabilidad hacia elecciones con más complejidad (número elevado de candidatos, con o sin orden asociado), se ha estudiado en 5.2.2 el caso de Helios con recuento a través de mixnets [375].

En ellas, se llevaron a cabo dos experimentos con 3.000 votos emitidos, pero los propios autores del paper no aportaron todos los datos completos de tiempos requeridos para cada operación, limitándose a recomendar el uso de la opción de recuento con propiedades homomórficas por su menor complejidad. Además, tras esas experiencias, la línea de investigación de Helios con recuento a través de mixnets ha quedado en punto muerto.

Por ello, la escalabilidad en Helios presenta dos limitaciones: de tamaño (hasta 2.000 votos sin ninguna modificación ni necesidad adicional) y además de tipología de elecciones, derivada de la modalidad de recuento homomórfico.

En resumen, si el organizador se ciñe a las directrices sobre la capacidad de Helios y las experiencias hasta la fecha, la escalabilidad no debería ser un problema. Ello implica una serie de limitaciones. Además, aparte de las recomendaciones de uso, no existe ninguna herramienta para probar el sistema antes de lanzar las elecciones (E-3).

En consecuencia, su desempeño en escalabilidad es medio-bajo debido a que sus pruebas han sido limitadas, con pocas expectativas de ampliarse a corto-medio plazo y con un rango de uso que se queda algo limitado con respecto a su planteamiento original.

Total: 4/10 puntos

8. Desarrollo ex_software (DESW-*n*)

Dentro de este apartado se valoran el conjunto de protocolos y procesos del sistema de VER aparte del software propiamente dicho, como se detalla en el apartado 4.1a:

- Distribución de credenciales, permisos y responsabilidades (DESW-2)
- Control de accesos y vigilancia (DESW-3)
- Protocolo de *back-up* (DESW-5)
- Organización de información relativa a las elecciones, con suficiente antelación y alternativas de voto en caso de fallar el VER (DESW-8)
- Protocolo de inicialización de los comicios (DESW-12)
- Envío de credenciales a través de un segundo canal (DESW-11)

La implementación a fondo de muchos de los puntos anteriores implica unas necesidades de recursos de todo tipo (monetarios, de recursos humanos, de organización etc.) únicamente al alcance de presupuestos en elecciones vinculantes en el ámbito político como las explicadas en el apartado 3.2.

Por lo que respecta a Helios, ya se ha comentado que no pretende abarcar más de lo que puede y por tanto en su documentación oficial se especifica que es un esquema apto para ámbitos de bajo riesgo de coerción y con un tamaño de censo limitado. Ello justifica que su definición e implementación de políticas de desarrollo ex_software sea escasa.

Es cierto que existe una documentación en la página oficial [395] y que desde la versión 2.0 se ha implementado una filosofía de permisos distribuidos (DESW-2). Aún así, se podría mejorar y actualizar la información disponible para facilitar su uso a administradores sin conocimientos específicos de ciberseguridad.

Con ocasión del mayor piloto con Helios [49], se implementaron medidas *ad-hoc* como la contratación de una empresa especializada para auditar el recuento final (DESW-4), la

instalación de diversos puntos de información especializados en el campus, la organización de demostraciones (DESW-9) (DESW-15), una prueba de uso etc. Todas ellas son buenas prácticas ex_software pero no están disponibles en la versión estándar de Helios.

Curiosamente, son los mismos autores de [49] los que apuntan en las conclusiones: *"each election is a significant project on its own. One cannot simply install a piece of software and expect an election to run smoothly"*, validando por tanto la necesidad de una política de desarrollo ex_software actualmente inexistente.

El desempeño de Helios en la materia puede definirse como mejorable, incluso circunscribiéndose al ámbito de uso recomendado por los autores. En otras palabras, si una persona sin conocimientos tratase de organizar unas elecciones con Helios en el ámbito recomendado, muy probablemente debería referirse a un experto en VER e incluso también a los desarrolladores de Helios.

Total: 3/10 puntos

9. Protocolo contra incidencias y ataques (PIA-*n*)

Cuanto mayor es la relevancia de los comicios, mayor es la acumulación de poder en los vencedores y mayor es por tanto la tentación de querer influir legal (o ilegalmente) en ellos. En ese sentido, el VER supone un vector de ataque muy atractivo a explotar de múltiples maneras, algunas de ellas explicadas en esta tesis [26, 32, 238, 277, 36, 236].

Es fundamental que el esquema de VER presente una documentación actualizada y pormenorizada sobre los ataques más relevantes y las actualizaciones que protegen frente a ellos. Otras recomendaciones incluyen:

El enfoque distribuido de la seguridad (PIA-5), la modularidad para confinar en lo posible los ataques en caso de que se produzcan, las medidas específicas en función de las características propias del sistema (ataques DoS, de ingeniería social etc.) (PIA-2), mantener toda la información posible en servidores del país donde tienen lugar las elecciones (PA-3), políticas de *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP)* y *Control Validation Audit (CVA)* (PIA-1) etc. (referirse a 4.1b para más información).

En el caso de Helios, tanto por sus recursos limitados como por el ámbito de uso (elecciones menores en entornos de menor riesgo de ataque), no cabe esperar un protocolo contra incidencias y ataques totalmente detallado y actualizado como si se tratase de unas elecciones vinculantes en el ámbito político.

En la página oficial del esquema sí que existe un apartado denominado *"Attacks and Defenses"* cuya última actualización se remonta a octubre de 2011.

Desde ese momento, Helios se ha venido utilizando en elecciones menores sin contratiempos pero aún así, sería muy recomendable mantener una documentación sobre ataques más actualizada y a poder ser acompañada de un protocolo detallado para usuarios y administradores. El hecho de que sea un proyecto *open-source* debería hacer más sencillo mantener actualizada dicha documentación, reduciendo el potencial riesgo de ataques (aún siendo limitado por su ámbito y características).

Total: **3/10 puntos** + **1 punto** por filosofía *open source* + rango de uso limitado. **4 puntos**

10. Versatilidad (V-*n*)

Tal y como se detalla en el apartado 4.1c, la versatilidad se aborda desde la doble vertiente de la capacidad del sistema de manejar distintas tipologías de elecciones (V-1) y de la existencia de versiones adaptadas para usuarios con necesidades especiales (V-2).

A mayores, se valora la no necesidad de instalar programas adicionales, así como la existencia de versiones para los navegadores y equipos con una cuota de mercado de más de un 1%. Por último, se tiene en cuenta el desempeño de la página oficial del sistema de VER con respecto al estándar WCAG 2.0 [414].

En lo que respecta a la existencia de distintas versiones adaptadas a diferentes tipologías de votación, en [375] se abordó la necesidad de implementar el recuento por mix-nets como alternativa a la propiedad homomórfica aditiva de ElGamal exponencial (2.2.4.6, 2.2.4.8) para elecciones con un gran número de candidatos u orden requerido.

Desafortunadamente, la iniciativa no tuvo continuidad y actualmente la única versión operativa es la basada en propiedades homomórficas: más simple y menos costosa de manejar pero limitada a elecciones de tipo simple, con pocos candidatos y listas reducidas.

En cuanto a la existencia de versiones para votantes con necesidades especiales, al autor de la tesis no le consta que existan (V-2).

Dos aspectos en los que Helios presenta un comportamiento satisfactorio son:

- La existencia de versiones para los principales navegadores del mercado (V-4)
- La no necesidad de instalar un software específico o que requiera conocimientos técnicos para votar (V-3)

Por último, en cuanto al grado de cumplimiento del estándar WCAG 2.0 (V-5), el autor ha realizado tres pruebas con tres herramientas distintas:

- Con Tawdis [415] se encontraron un total de 12 problemas y 79 advertencias aunque ninguno de ellos es crítico:

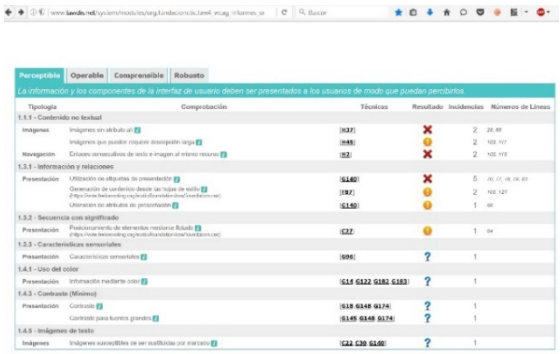


Figura 41: Helios con respecto al estándar WCAG 2.0 según Tawdis

- Con WAVE (*Web Accesibility Evaluation Tool*) [416], aparecieron 11 errores y 13 alertas. La gran mayoría son los mismos que en el caso anterior: una imagen sin texto alternativo y enlaces adyacentes que llevan a la misma página. Según esta herramienta, ninguno de ellos es crítico (aunque sí fácilmente subsanables):

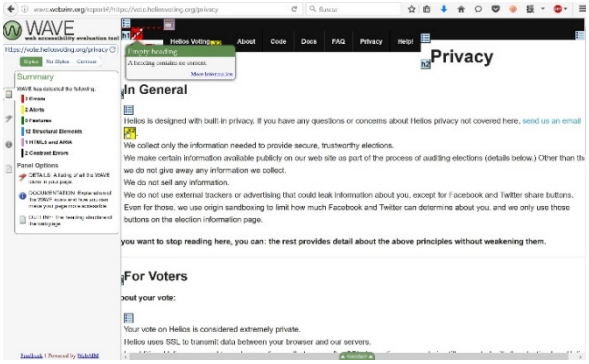


Figura 42: Helios con respecto al estándar WCAG 2.0 según WAVE

- La tercera herramienta, *Access Monitor* [417], da un paso más y asigna una nota numérica así como una letra en función del desempeño de Helios [396]: 5.4 puntos y letra A (por debajo de AA y AAA):

Figura 43: Helios con respecto al estándar WCAG 2.0 según *Access Monitor*

En conclusión, Helios no presenta versatilidad en lo que respecta a modalidades adaptadas a usuarios con necesidades especiales (V-2). Por otra parte, sí existe una versión de recuento de mix-nets sin actualizar (V-2). En parte se ha realizado con el sistema de voto Belenios [451].

Puntúa positivamente en versatilidad de navegadores (V-4) y ausencia de instalación de software (V-3) y no llega al nivel exigido en cuanto al estándar WCAG 2.0 si bien obtiene una valoración "A" y los errores encontrados no son críticos (V-5).

Aún así, la ausencia de versión adaptada a usuarios con necesidades especiales y la limitación en las modalidades de elecciones que soporta pesan más que su desempeño en cuanto a navegadores y ausencia de instalación.

Total: 4/10 puntos

11. Coste C (C-*n*)

En grandes pilotos, con cientos de miles de votos emitidos y organizados a nivel nacional, los datos suelen ser más abundantes. En el caso de votaciones menores, , la información es muy difícil de conseguir.

En el apartado 4.1d se detallan varios costes reales de elecciones en varios ámbitos, si bien ninguno de los pilotos costó menos de 552.000 USD en el caso de la iniciativa no vinculante *Empower LA* [298].

Dicho nivel de gasto supone una barrera a la utilización de sistemas de VER para multitud de colectivos y asociaciones sin acceso a tales recursos. Por ello, el papel que cumple Helios, pese a tener margen de mejora, es de una importancia notable.

Cabe destacar que, en contra de lo comúnmente aceptado, la implementación de unas elecciones con Helios no suele ser totalmente gratis. Para casos muy simples, en entornos poco problemáticos y con pocos cientos de votos podría llegar a ser así (asumiendo la existencia de voluntarios y un mínimo de recursos por parte de la Universidad o entidad que lleva a cabo las elecciones).

No obstante, a poco que aumente la complejidad, es muy recomendable disponer de recursos para programar módulos específicos, contratar a empresas especializadas para formar a los voluntarios/trabajadores y llevar a cabo auditorías, preparar una infraestructura suficiente que soporte los flujos de datos y computaciones etc. Ese fue el caso en [49] sobre el que no se detallaron los costes finales del piloto.

Helios facilita, por tanto, un acceso gratis o ***quasi- gratis*** (C-2) a una herramienta de VER que cumple con su cometido dentro de los entornos para la que ha sido desarrollada.

Total: 9/10 puntos

12. Mantenimiento (M-*n*)

Como se explica en el punto 4.1e, se refiere tanto al mantenimiento del sistema (M-1) como en el sentido de la *"everlasting privacy"* (M-2) entendido como el mantenimiento a largo plazo garantizando el seguridad de los datos [235, 241, 242].

En el caso de Helios, la *"Everlasting Privacy"* ha sido estudiada en [241] si bien todavía no se ha implementado para la versión oficial.

En cuanto al mantenimiento software y ex_software, desde finales del año 2012 no se han producido actualizaciones de calado en la página oficial [395], si bien la comunidad científica ha tomado Helios como la base sobre la que probar nuevas líneas de investigación y mejoras muy prometedoras en el campo del VER como se ha estudiado en el presente punto 5.2 [370, 375, 377, 381, 371].

En resumen, Helios está vivo, si bien en los últimos años las investigaciones y mejoras vienen de la mano de la comunidad científica sobre la base de la versión estándar (por el hecho de ser *"open source"* y bien documentada) con el apoyo de su autor principal Ben Adida (suele aparecer en los agradecimientos de los nuevos papers).

El punto que se echa en falta es transladar toda esa actividad a la versión original de Helios para que el proyecto crezca y continue siendo el gran referente del VER en código abierto.

Total: 5/10

5.2.4 Conclusiones y valoración final

La aparición de Helios en 2008 [1] supuso un espaldarazo fundamental al Voto Electrónico Remoto al tratarse del primer esquema auditable, gratuito y *open source* plenamente operativo y al alcance de cualquiera que quisiese organizar unas elecciones.

Desde un punto de vista técnico, Helios está basado en el protocolo *Simple Verifiable Voting* de Benaloh [3] y la característica que lo convirtió en revolucionario (aparte de ser públicamente auditable, *open source* y gratuito) fue la separación de la preparación del voto y su envío (favoreciendo la verificabilidad).

En cuanto al ámbito de uso, el propio autor Dr. Ben Adida afirma en la web oficial [396]: *"Helios no debería utilizarse para elecciones en el ámbito político. Las elecciones on-line son apropiadas cuando no se esperan grandes intentos de fraude o coerción a los votantes"…"Ello no tiene que ver con Helios en sí, sino que no nos fiamos de que los equipos de los votantes sean suficientemente seguros".*

Helios aborda la cuestión de la verificabilidad pero no la coerción, asumiendo que se van a seguir sus recomendaciones de uso y los entornos de utilización van a ser asociaciones, universidades, clubes etc.

En total se han emitido más de 100.000 votos a través de Helios, 8.000 de ellos en el piloto más grande en la Universidad de Lovaina en 2009 [49] que contó con medidas especiales que supusieron un coste no especificado por los responsables.

Desde un principio, Helios ha suscitado una gran atención en la comunidad científica, siendo objeto de numerosos ataques y estudios, detallados en el punto 5.2.2:

Entre los ataques destacan: Estehghari et al. [372, 374], usando como vector el cliente software así como aquellos sobre la privacidad, explotando las debilidades en cuanto a maleabilidad de la heurística de Fiat Shamir [155, 371, 379] usada en Helios.

En [377], Cortier et al. apuntan a que Helios sigue presentando vulnerabilidades en cuanto a la privacidad del voto y a la verificabilidad de la elegibilidad.

De hecho, éste último paper es un excelente ejemplo de mejoras en el VER que han venido propiciadas gracias a Helios. En él, los autores, partiendo de una debilidad en el esquema original, proponen una nueva versión denominada Helios con credenciales o Helios-C con una autoridad de registro que mejora la privacidad del Helios original, consiguiendo que los requisitos para un correcto funcionamiento sean menos exigentes.

Otros ejemplos de mejoras sobre el esquema original que podrían tener aplicación en otros sistemas de VER son el Helios con recuento a través de mix-nets [375], Helios con votos de relleno o Helios-KTV [360, 381] o Helios distribuido [370].

En cuanto a su uso práctico, Helios se sigue utilizando actualmente en la elección de los miembros de la dirección de la IACR [412] y de los representantes de alumnos de la Universidad de Princeton [413] todos los años.

En resumen, Helios se ha erigido desde sus inicios en una valiosísima herramienta en una doble vertiente:

- Como sistema de VER plenamente operativo, *open source* y auditable en elecciones en entornos de bajo riesgo a coste *quasi-cero*.
- Como punto de partida o "banco de pruebas" para la comunidad científica. Ha dado pie a numerosas iniciativas y proyectos, muchos de los cuales tienen aplicación práctica en el campo del Voto Electrónico Remoto. De hecho, Helios es la base de otras herramientas de VER muy similares tales como Zeus [341, 342] y BeleniosRF [72].

Es deseable que el sistema se actualice más regularmente y que se desarrollase una versión operativa de recuento con mix-nets para poder ampliar su rango a elecciones de mayor complejidad (numerosos candidatos/listas ordenadas).

Por lo que respecta a una potencial utilización de Helios en elecciones VAP, no es una teoría plausible en el corto o medio plazo puesto que ni la estructura ni los medios (logísticos, humanos y financieron) son los suficientes actualemente.

Con todo, Helios sigue representando hoy en día posiblemente el mejor esquema académico plenamente funcional, *open source*, gratuito y verificable, lo que por sí es un gran hito, prestando un enorme servicio a la comunidad científica y al VER en su conjunto.

Para concluir el presente apartado 5.2 dedicado a Helios, se aplica la fórmula de la metodología para calcular su valoración final y se presenta también un análisis radial para apreciar de una manera más gráfica sus fortalezas y debilidades dentro de la metodología:

$$\sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Criterio	Ponderación	Helios
Verificabilidad extremo a extremo	N.A.	**Δ**
Privacidad/resistencia a la coerción	N.A.	**X**
Inviolabilidad	**1.2**	6 * 1,2 = **7.2**
Usabilidad	**0.8**	4 * 0.8 = **3.2**
Monitorización/Auditoría	**1.2**	4 * 1,2 = **4.8**
Desarrollo software	**1.2**	7.5 * 1.2 = **9**
Escalabilidad	**0.8**	4 * 0.8 = **3.2**
Desarrollo ex_software	**1.2**	3 * 1.2 = **3.6**
Protocolo contra incidencias y ataques	**1.2**	4 * 1.2 = **4.8**
Versatilidad	**0.6**	4 * 0.6 = **2.4**
Coste	**1.0**	9 * 1.0 = **9**
Mantenimiento	**0.8**	5 * 0.8 = **4**
TOTAL	**10**	**51.2**

Tabla 20: Metodología aplicada a Helios Voting

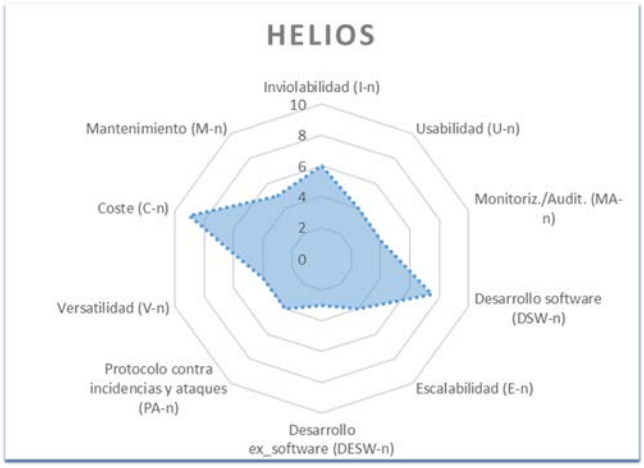


Figura 44: Análisis radial de Helios

5.3 Scytl

5.3.1 Introducción

En este apartado, se entiende por sistema Scytl al conjunto de esquemas implementados por la compañía Scytl en los distintos proyectos que ha desarrollado.

En la presente tesis se han repasado numerosas experiencias de VER, varias de ellas implementadas por Scytl, incluyendo algunas de las más relevantes: (Elecciones noruegas en 2011 y 2013, elecciones de la Asamblea de Franceses en el Extranjero a partir de 2009, elecciones de Nueva Gales del Sur en 2015, elecciones del cantón de Neuchâtel etc.).

Scytl nació en Barcelona en 2001 como una *spin-off* de un grupo pionero en Voto Electrónico que llevaba activo desde 1994. De hecho, las dos primeras tesis en Europa sobre seguridad en el voto electrónico salieron de dicho grupo.

En la actualidad es la mayor compañía del mundo dedicada al diseño y desarrollo de soluciones de VER y a una serie de actividades relacionadas: auditoría, consultoría, formación etc. Posee además alianzas con Microsoft, Swiss Post y HP.

Son titulares de más de 40 patentes en la materia. En 8 de ellas es co-autor el tristemente desaparecido fundador de la empresa, Dr. Andreu Riera. Además, suman más de 250 trabajadores repartidos por todo el mundo.

Por lo que respecta a su producción científica, Scytl tiene una división dedicada en exclusiva al I+D liderada por Jordi Puiggalí, CSO y SVP de Scytl [419]. En la actualidad, los campos en los que están investigado más intensamente son la *everlasting privacy* [241], la corrección del votante en mixnets y la verificabilidad individual (en este último ámbito acaban de presentar una tesis doctoral [369] realizada por la Dra. Sandra Guasch).

En cuanto a la tipología de elecciones que ha gestionado Scytl, abarca desde las más exigentes (elecciones vinculantes en el ámbito político a nivel nacional como en Noruega) hasta otras de menor riesgo de ataque como pueden ser elecciones vinculantes en otros ámbitos (sindicatos, asociaciones etc) o referendos consultivos sin implicaciones legales.

5.3.2 Características

En primer lugar, cabe destacar que no existe un único sistema de VER Scytl. Como se ha ido desgranando a lo largo de la presente tesis, especialmente en el capítulo 3 "Antecedentes, experiencias previas y estado del arte", cada país establece una serie de requisitos propios al VER de acuerdo a su propia legislación. Por tanto, cada solución de VER se adapta a la idiosincrasia de cada territorio.

En el caso de Scytl, las tres experiencias más destacadas hasta la fecha son: las elecciones parlamentarias de Noruega de 2013 (apartado 3.2.2), las elecciones de Nueva Gales del Sur de 2015 (apartado 3.2.5) y el proyecto de VER del cantón de Neuchâtel en Suiza (apartado 3.2.6). Todas ellas son elecciones públicas vinculantes en el ámbito político, las de mayor complejidad y exigencia.

Cada una de ellas está explicada en detalle en el apartado indicado. No obstante, se van a destacar de manera resumida las principales caracteríticas de cada una:

Esquema noruego de 2013

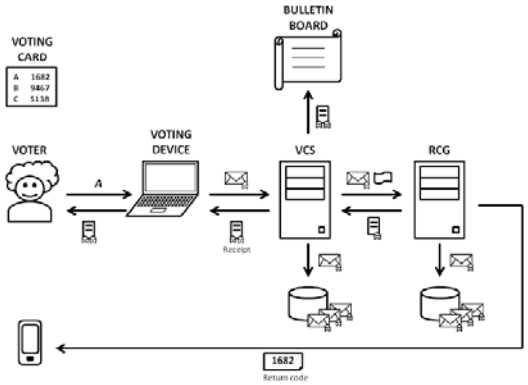


Figura 45: Esquema noruego 2013 [369]

Como se ha apuntado, cada país establece unos requisitos específicos al VER y en el caso de Noruega se requería la opción de votar en más de una ocasión (y que al final prevaleciese el último voto electrónico o bien el voto tradicional si se acudía a las urnas). Las autoridades también solicitaron la publicación del código fuente, hecho por el que accedieron a pagar un extra a la empresa.

Además, Scytl propuso un sistema con códigos de retorno enviados por un segundo canal (en este caso un SMS al teléfono móvil del votante). Adicionalmente, el votante había recibido con antelación un correo postal con una tarjeta personal que contenía unos códigos aleatorios personales asociados a cada opción de voto para poder comprobar que la opción votada había sido recibida correctamente (verificabilidad individual).

Para la encriptación se vuelve a utilizar ElGamal [64] y el recuento se realiza con mixnets, descifrando los votos individualmente, obteniendo un cifertexto por cada voto.

A mayores, se puso el *bulletin board* a disposición de los votantes en un repositorio Github que actualizaba cada hora con los votos almacenados por el VCS. Éstos podían utilizar el recibo que obtenían al final del proceso (un *hash* de su voto) firmado por los dos servidores para comprobar que su sufragio había sido recibido.

En la versión de 2013, se introdujo una mejora en este punto, ya que el RCG también almacenaba los votos en una urna local, permitiendo comprobar si uno de los dos era deshonesto. Por otra parte, se partía de una serie de premisas de seguridad [56]:

- El servicio de autenticación, el generador de tarjetas electorales y el generador de códigos de retorno no son simultáneamente deshonestos
- El auditor y el sistema de mix-nets no son simultáneamente deshonestos
- El VCS y el RCG no son simultáneamente deshonestos

El resto de mejoras en la versión de 2013 con respecto a la de 2011 fueron [430]:

- Implementación del cliente de voto en Javascript
- Mejora de ElGamal y la ZKP [261]
- Mejora de la verificabilidad individual, permitiendo a los votantes comprobar que sus votos estaban en el tablón (*recorded-as-cast*)
- Mejora de los controles de comprobación de la corrección de voto
- Optimización del sistema de *threshold* para la distribución de claves de desencriptación y recuento
- Implementación de una nueva versión de mixnet verificable que suministraba pruebas de detección de manipulación en vez de análisis heurísticos [42].
- Introducción de un nuevo sistema de administración central electrónica denominado EVA.

Finalmente, de 250.159 potenciales votantes, 70.090 de ellos emitieron su voto a través de la herramienta de VER, para un total de 72.969 votos (algunos re-votaron por internet).

Esquema *iVote* 2015

En Nueva Gales del Sur (NGS), el objetivo era construir un sistema de VER sobre la base que había desarrollado la empresa responsable de la solución de 2011 (Everyone Counts).

La motivación inicial de su desarrollo fue facilitar una opción de voto a personas con discapacidad visual, aunque finalmente se amplió su uso a otros colectivos: el votante que está a más de 20 kilómetros del colegio electoral más cercano o que no va a estar en NGS el día de los comicios. Como el lector recordará, Australia es uno de los países donde votar es obligatorio por ley y existen sanciones en caso de no ejercer el derecho.

Otra característica relevante del VER en NGS es que las autoridades consideran el riesgo de coerción como bajo y no lo incluyen como uno de los requisitos que debe de cumplir la herramienta. Por ello, se permite votar por teléfono y verificar el voto por el mismo medio (contraviniendo las recomendaciones sobre privacidad).

Scytl mejoró el esquema original y consiguió que cumpliese los requisitos de *cast as intended* y *recorded as cast* [369]. De todos modos, en la nueva configuración, el software del VS (*Verification Server*) fue desarrollado por otra entidad contratada por las autoridades.

El esquema es el siguiente:

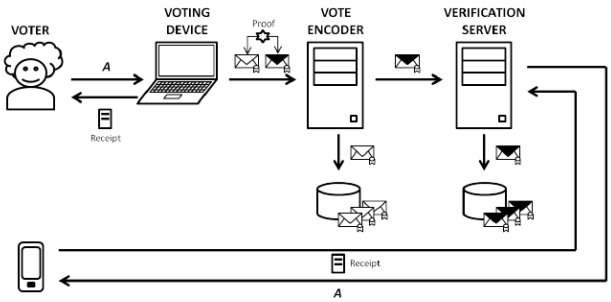


Figura 46: Esquema *iVote* 2015 [369]

Para votar, el votante se autentica con un PIN de su elección y un número de *iVote* que recibe por un canal diferente (se podía también votar por teléfono y en un entorno controlado pero esos dos casos están fuera del ámbito de la presente tesis).

Posteriormente, el voto se encripta junto con un valor aleatorio y se envía. A continuación, el votante recibe un número de resguardo único que puede usar para verificar su voto. (tras introducir el número *iVote*, el PIN y el número de resguardo).

Finalmente, una vez que se ha concluido el recuento, se suben los números de recibo de los votos desencriptados a un sitio web donde los votantes pueden comprobar que sus votos fueron incluidos en el mismo.

El criptosistema es ElGamal IND-CPA en el que se cumple DDH [25, 188]. Para más información sobre el proceso de voto, referirse a 3.2.5 y [288, 369].

Con respecto al esquema de Noruega o a Helios, la solución de *iVote* 2015 es más simple. Al no existir el requisito de privacidad/resistencia a la coerción, se almacenan dos sobres distintos conteniendo el voto en el tablón (*Vote Encoder*) y en el *Verification Server*. El sobre en el tablón puede ser únicamente abierto por los administradores y el del *Verification Server* por el votante una vez identificado (número *iVote*, PIN y número de resguardo).

En el caso del sistema noruego y de Helios, la prueba de verificabilidad individual (*cast-as-intended*) no puede realizarse hasta después de haberse cerrado la urna, de lo contrario se comprometería la privacidad.

Durante el transcurso de las elecciones hubo un contratiempo con respecto a una aplicación de análisis de datos de un desarrollador externo [287]. En el punto de inviolabilidad del análisis se detalla la vulnerabilidad, si bien no ha habido ninguna prueba de que el sistema fuese efectivamente atacado.

Finalmente, más de 280.000 votantes emitieron su voto a través de *iVote*, suponiendo la mayor elección en términos de VER dentro del ámbito político vinculante.

Cabe destacar como se dice en [288], que el sistema no es verificable universalmente puesto que la implementación original utilizaba una encriptación autenticada simétrica que no permite la *re-randomización* de los votos encriptados, lo que a su vez dificulta la utilización de mix-nets verificables como Wikström [222], base de los sistemas E2Ev.

Para el recuento se utilizó AES [429] + ElGamal [64].

Esquema Neuchâtel 2015

La tercera experiencia destacada ha tenido lugar en marzo de 2015 en el cantón suizo de Neuchâtel. Al igual que en los dos casos anteriores, el sistema de VER se adapta a la legislación propia tanto cantonal como federal suiza. Una de las primeras diferencias evidentes es que en este caso, no se permite votar más de una vez (con las implicaciones que conlleva en cuanto a resistencia a la coerción), a diferencia de Noruega.

En [369], la Dra. Guasch hace un pormenorizado análisis del sistema de Neuchâtel. En la presente tesis, al tratarse de un punto dentro de un apartado de una metodología, se resaltarán únicamente los aspectos originales del esquema, haciendo especial énfasis en las diferencias con los dos anteriores (esquema noruego 2013 y *iVote* 2015). El lector interesado en todos los detalles, puede referirse a [369, 420, 430].

Para la encriptación, se utiliza la variante *Signed El Gamal* [421], el esquema de firma ESA-FDH y la mix-net verificable de Bayer y Groth [224].

Algunas de las características propias del sistema de Neuchâtel van en la línea de presentar un sistema de VER con verificación de la condición de *cast-as-intended*. Ello es requisito para ser autorizado en elecciones de Suiza hasta un máximo del 50% del electorado, de acuerdo con la nueva legislación federal que entró en vigor en enero de 2014 y que se detalla en el apartado 3.2.6 y [283].

En el esquema noruego, la generación de los códigos de retorno se dividía en dos servidores (*ballot box server* y *code generation server*), los cuales no cooperaban para desencriptar los votos y por tanto romper la privacidad del sistema [261].

La contrapartida fue que la complejidad de instalar e implementar dos estructuras distintas e independientes conllevó un alto gasto en recursos organizativos y monetarios. La escasez de empresas proveedoras de los servicios necesarios también aumentaba el riesgo de no poder garantizar una total independencia entre las dos infraestructuras [369].

El esquema de Neuchâtel implementa una estructura con un único servidor, basándose en una primera propuesta del esquema noruego, mejorada con los avances que se han producido en los últimos años en la tecnología de cliente, sobre todo en JavaScript.

Para evitar que un único ente genere los códigos, la solución que se adopta es separar la creación de los códigos en dos fases, computándose la primera en el equipo del votante.

Como se explica en [369], el esquema consta de 4 fases:

- Configuración: Se generan las claves pública y privada de la elección así como un conjunto de códigos de retorno para cada votante, que se imprimen en una tarjeta. Cada votante recibe la suya por correo postal.
- Votación: Tras seleccionar sus opciones, el votante encripta su voto (con *Signed El-Gamal* [421] y la clave pública de la elección). Además, en el equipo del votante se realiza la primera parte de la computación de los códigos de retorno (denominados "códigos parciales de retorno") y unas ZKP para demostrar que los votos encriptados y los códigos parciales de retorno se corresponden a las mismas opciones votadas.

  El votante envía el voto encriptado, los códigos parciales de retorno y las ZKP al servidor de voto (*Voting Server*). Éste verifica las ZKP, guarda los votos encriptados y usa los códigos parciales de retorno para generar los códigos finales de retorno, que se envían de vuelta al votante.
- Confirmación: La introducción de esta fase se debe a que la legislación suiza no permite al votante emitir más de un voto. Para mantener el requisito de *cast-as-intended* sin volver a votar, las fases hasta la votación inclusive se consideran preliminares al envío efectivo del voto. El envío real se produce cuando el votante, tras haber recibido los códigos de retorno finales, los coteja con los de su tarjeta personal y manda una confirmación de que el voto es correcto.
- Recuento: Se pasan los votos encriptados y confirmados a través de una mixnet verificable (en este caso Bayer y Groth [224]) para romper el vínculo voto-votante y posteriormente se desencriptan y recuentan.

De una manera formal, el esquema implementa los siguientes algoritmos: en el esquema de encriptación $(Gen_e, Enc, Dec, EncVerify)$, en el esquema de firma $(Gen_s, Sign, SignVerify)$ y en los dos esquemas NIZKP $(ProveEq, VerifyEq, SimEq)$ y $(ProveDec, VerifyDec, SimDec)$.

A mayores, los siguientes algoritmos: $Setup(1^\lambda)$, $Register(1^\lambda, id, sk_a, sk_c)$, $CreateVote(id, \{v_{j1}, \dots, v_{jt}\}, P_v^{id}, S_v^{id})$,$ProcessBallot(BB, id, b)$, $CreateBallotProof(b, sk_a, P_a^{id})$,$AuditBallotProof(\{v_{j1}, \dots, v_{jt}\}, \sigma, S_a^{id})$, $ProcessConfirm(BB, id, C_b)$, $Tally(BB, sk)$, $VerifyTally(BB, r, \Pi)$. Para el detalle de cada uno de ellos y su funcionamiento, se remite a [369, 420].

La siguiente figura es un esquema a modo de resumen del protocolo de Neuchâtel 2015:

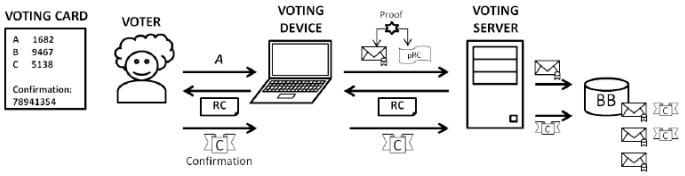


Figura 47: Esquema Neuchâtel 2015 [369]

Finalmente, el esquema propuesto fue desplegado por primera vez en el cantón de Neuchâtel en el referéndum federal celebrado el 8 de marzo de 2015. En total, 5.132 votantes ejercieron su derecho a través del sistema de VER suponiendo un 21.45% de todos los que se habían registrado para votar por internet. La votación tuvo lugar sin incidentes.

5.3.3 Análisis

Para la realización del análisis del sistema de Scytl, aparte de las referencias bibliográficas que se han nombrado y las que aparecerán a continuación, se ha contado con la inestimable ayuda de Jordi Puiggalí, CSO y SVP de Investigación y Seguridad, quién se ha mostrado siempre dispuesto y diligente a resolver cualquier cuestión que pudiese surgir.

En este apartado, se va a tener en cuenta en cada requisito la solución de las tres analizadas que mejor desempeño ha demostrado puesto que se entiende que la capacidad de Scytl llega al punto más alto que se haya alcanzado entre esos tres esquemas.

Se recuendan los 12 criterios que conforman la metodología de evaluación:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

1. Verificabilidad extremo a extremo

Como ya se apuntó en 2.2.2 los requisitos son: *i) Cast as intended, ii) Recorded as cast, iii) Counted as recorded,* junto con *elegibility verifiability.* [51, 93, 77, 3, 359, 369, 389].

Al igual que en el caso de Helios, no hay unanimidad entre los autores. En el caso de Nueva Gales del Sur *iVote* 2015, es la propia empresa Scytl la que dice que el sistema no es verificable extremo a extremo, por la idiosincrasia de las elecciones que obliga a usar un esquema de encriptación simétrica [288].

En cuanto al esquema noruego, hay diversidad de opiniones. En [262] H. Nore, *Project Manager* por parte del gobierno noruego, afirma que se ha conseguido que el sistema sea E2Ev porque se puede verificar todo el proceso gracias a los códigos de retorno y las ZKP:

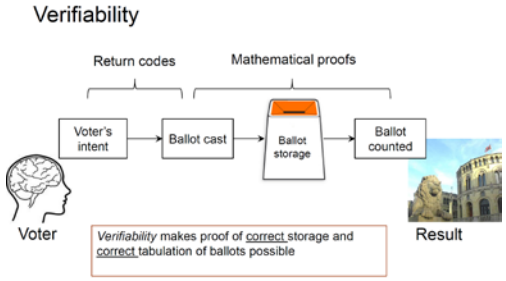


Figura 48: Verificabilidad en el esquema de Noruega 2013 [262]

Además, el organismo especializado invitado como observador a las elecciones noruegas "*The Carter Center*", escribe en su informe final [260] que pese a que las autoridades no requerían que el sistema fuese E2Ev en su acepción tradicional (*Cast as intended, Recorded as cast, Counted as recorded*), el esquema planteado a través de verificabilidad individual y de proxy junto con las ZKP y el tablón accesible en GitHub conformaban un esquema E2Ev.

Por otra parte y tal y como se ha estudiado en 2.2.2 y 2.2.4.7, no existe una definición universalmente válida de la E2Ev y de hecho en los últimos años han surgido líneas de investigación que están aportando nuevos puntos de vista a la cuestión [360, 362, 366] y que no se han podido aplicar a los sistemas anteriores. En esa línea se encuadra también el nuevo concepto de "*universal cast-as-intended*" de la Dra. Guasch en [369] que es de prever se irá implementando en nuevas soluciones de VER de Scytl.

Por ello, en función de la definición que se tome de la E2Ev y las precondiciones más o menos duras de componentes honestos, el sistema de Noruega y Neuchâtel podría considerarse E2Ev o no. Una colusión entre el generador de códigos, el servicio de autenticación y el generador de tarjetas electorales podría comprometer la integridad de la votación [56]. Dicha hipótesis es altamente improbable por la distribución de roles y los servicios independientes de auditoría y observadores internacionales invitados.

Conclusión: **Δ,** cumple bajo ciertas premisas (si se aceptan los requerimientos de seguridad y comportamiento del protocolo noruego [260, 261]).

2. Privacidad/resistencia a la coerción

La privacidad como resistencia a la coerción (RC) y definida por Juels et al. en [104], implica que un votante no puede colaborar con un coercionador para obtener información de cómo votó, incluso en el caso de que quiera voluntariamente vender su voto.

En el caso de los 3 esquemas estudiados, en *iVote* 2015 las autoridades no consideran relevante el riesgo de coerción y por tanto se permiten sistemas de verificación de voto por teléfono, que no protegen la privacidad del votante.

En lo que respecta a Neuchâtel 2015 y Noruega 2013, el mecanismo de códigos de retorno (en el caso suizo únicamente por pantalla) comporta una serie de ventajas indudables en lo que se refiere a verificabilidad, pero por otra parte suponen una prueba que puede utilizar el votante deshonesto para probar la forma en la que ha votado a un coercionador. Los esquemas cumplen por tanto el primer escalón denominado privacidad del voto.

Conclusión: **×,** no cumple

3. Inviolabilidad (I-*n*)

Entendida como la protección del acceso al software y a los sistemas auxiliares a través de protocolos de autenticación seguros, evitando accesos a través de terceras aplicaciones y/o servidores vulnerables. Se valora por tanto:

- Un control seguro de accesos al programa (I-1)
- Protocolos específicos de seguridad *risk assessment* y *threat modeling* (I-5)
- La existencia de copias de seguridad aisladas y *offline* (I-3)
- Una sólida política de distribución de permisos y responsabilidades, reforzada en los nodos críticos para proteger de colusiones entre las partes (I-4)
- La modularidad del sistema para aislar los errores/ataques (I-6)
- La periódica actualización de los puntos previos (I-7)

Las tres experiencias que se han revisado pertenecen a la tipología de elecciones VAP; las de mayor el nivel de exigencia. En ese sentido, todos los puntos arriba presentados se cumplen en mayor o menor medida.

No obstante, ha habido vulnerabilidades que, si bien no terminaron derivando en ataques consumados, sí que supusieron un toque de atención sobre la dificultad de obtener sistemas de VER seguros: en las elecciones de NGS con el sistema *iVote*, en [47] se reportó un ataque aprovechando la presencia de la herramienta de análisis Piwik alojada en un servidor cuya configuración SSL era vulnerable al ataque FREAK [26]. Pese a que Piwik no es una herramienta de Scytl y su presencia se debía a que había sido introducida por el desarrollador de la versión de 2011, se debería haber comprobado su vulnerabilidad (I-1).

Es también cierto que los descubridores de la vulnerabilidad [47] decidieron darla a conocer antes a los medios de comunicación que a los desarrolladores de *iVote* y en cualquier caso varios días después de descubrirla, sembrando dudas sobre si su intención fue la de mejorar el sistema o simplemente tratar de dañar su reputación.

Desde Scytl apuntaron que precisamente introdujeron el sistema de verificación por teléfono para evitar ataques del tipo *man-in-the-middle* como el FREAK y que en cualquier caso hacían falta conocimientos avanzados de computación para potencialmente explotar la vulnerabilidad. Además, tanto el patrón de votación como la ausencia de quejas e incidencias refuerzan la tesis de que no se produjo un ataque efectivo.

Por lo que respecta a las elecciones de Noruega, en el 2013 la consultora contratada como auditora descubrió un error en ElGamal. Se subsanó el error inmediatamente y se comenzó una investigación para comprobar el alcance del mismo (I-6).

Las autoridades reaccionaron con total transparencia durante todo el incidente y concluyeron que las elecciones podían continuar. Es interesante destacar que nadie descubrió el error en el código pese a ser público y estar disponible desde varios meses antes. En esa línea, H. Nore en [262] hace una reflexión sobre cuánto se involucra realmente la comunidad científica y el público en revisar el código fuente tras conseguir que se publique.

Por último, en el caso de Neuchâtel 2015, las elecciones discurrieron sin contratiempos, indicando también una mejora en la inviolabilidad del sistema Scytl.

Conclusión: 7/10 puntos +1 punto de bonus debido a que el perfil de elecciones que se han analizado son las más exigentes, atrayendo la atención tanto de la comunidad científica como de hackers éticos y maliciosos.

Total: 8/10 puntos.

4. Usabilidad (U-*n*)

En este caso, los puntos a valorar son:

- La simplicidad y claridad en el proceso de votación (U-1)
- La existencia de versiones/adaptaciones para colectivos con discapacidad (según el Consejo de Europa [54] y la *"Convención de las Naciones Unidas sobre Derechos de Personas con Discapacidad"* [259]) (U-2)
- La tasa de éxito de un votante sin conocimientos de criptografía ni seguridad para emitir correctamente su voto y que sea contabilizado. (U-3)

En el caso de Australia, la motivación original de introducir el voto por internet fue precisamente dar una opción adicional a las personas con discapacidad. En el año 2015, en una encuesta de uso de *iVote* 2015, más de 95% de los usuarios dijeron estar satisfechos o muy satisfechos con el sistema. La misma proporción se mantuvo en cuanto a la satisfacción en el sistema de asistencia.

En cuanto a aspectos a mejorar, la herramienta actualmente está disponible únicamente en inglés pese a que hay otros 23 idiomas reconocidos en el estado. La Comisión Electoral ya ha mostrado su intención de incluirlos en comicios venideros [292].

En Noruega, las autoridades incluyeron ya desde 2009 un apartado en el pliego de condiciones para concurrir al desarrollo del sistema de VER de *"Accessibility and Usability Requirements part of the tender"* en el que se especificaba que el sitio web debía cumplir con el nivel AA del WCAG 2.0 [258]. Además, existía un centro de atención al votante para resolver las dudas que pudiesen surgir durante el proceso.

En el caso de Neuchâtel, al igual que en el de Noruega, se introdujo una capa de usabilidad para reducir la longitud de los códigos a guardar por el usuario, rebajándolos de 410 y 52 caracteres a 7 y 4. El problema que surgió entonces es que se podría tratar de atacar dichos códigos por fuerza bruta. Para evitar dicha situación, se limitó el número de llamadas que puede realizar un cliente de voto al servidor.

En los tres casos, la introducción del VER aumentó la participación por lo que al menos se puede concluir que la usabilidad no perjudicó la participación.

También es cierto, como apunta Guasch en [369], que la usabilidad es uno de los campos con mayor margen de mejora. Al final, se le está pidiendo al votante conservar una tarjeta de códigos que no puede mostrar a nadie, recibir un código por otra vía distinta a la que

votó y confirmar que su voto se recibió correctamente (Noruega), proceder a votar realmente (Neuchâtel) o verificar el voto a través de la herramienta telefónica (*iVote* 2015). En todos esos casos, el proceso es más complejo que el tradicional de votación. (U-1)

Por todo ello, el desempeño en usabilidad es satisfactorio (con versiones para personas con discapacidad (U-2), adherencia a los estándares y uso de capas intermedias de usabilidad) pero sigue existiendo un potencial de mejora.

Total: 7.5/10 puntos.

5. Monitorización/auditoría (MA-*n*)

Se valora la existencia de un protocolo específico para dichas tareas durante todo el ciclo de vida del proyecto (MA-1) (MA-2), así como de herramientas generadoras de informes periódicos (*logs*) que no puedan ser borrados ni alterados. Su almacenamiento deberá encontrarse físicamente separado del resto de servidores de las elecciones (MA-4).

Es también altamente recomendable la figura de un auditor independiente con atribuciones distribuidas para reducir riesgos de colusión (punto 2.3e) (MA-8).

En el caso de un contratiempo, ya sea un error o un ataque, deberá auditarse y actuar de acuerdo a lo establecido en el protocolo, prevaleciendo la privacidad del votante en caso de compromiso entre propiedades (MA-9) (MA-10).

En lo que respecta a Scytl, los 3 sistemas evaluados han sido desplegados en elecciones VAP en 3 países con una larga tradición democrática y cuyos habitantes mantienen un alto nivel de confianza en sus instituciones (Noruega, Suiza y Australia).

Por ello, todo el proceso de *"tender"* o subasta pública ha sido de una gran transparencia en todos los casos (más si cabe todavía en Noruega), con multitud de partes implicadas y la contratación de auditores independientes para todo el proyecto (MA-1) (MA-2).

En el caso noruego, además de la transparencia de todo el proceso, las autoridades contrataron tanto en 2011 como en 2013 a auditores externos (Quality AS [260] y *mnemonic*) para comprobar la seguridad y calidad del código fuente y elaborar un informe de *risk analysis* y *risk assessment*. En total revisaron más de 211.000 líneas de código encontrándose *bugs* cuya solución hizo el sistema de VER más seguro [256] (MA-5) (MA-6) (MA-8).

A mayores, el KRD (la autoridad electoral) realizó un llamamiento público para participar en el proceso de auditoría de la fase de desencriptación y recuento. Al no responder ningún *stakeholder*, se decidió contratar a un experto independiente para llevar a cabo dichas actividades, seleccionando a la empresa que había quedado finalista en el *tender* (Computas) para implementar las herramientas de auditoría. Finalmente, la desencriptación y el recuento se retransmitieron en *live streaming* y contaron con la presencia de observadores y expertos internacionales.

En Australia, la *New South Wales Electoral Commission* también contrató a un auditor independiente especializado y en Neuchâtel, las autoridades invitaron a expertos independientes a auditar todo el ciclo de vida del proyecto.

En resumen, en los casos estudiados la actividad de monitorización/auditoría ha sido satisfactoria porque se unen varios factores:

- Legislación adaptada y moderna
- Transparencia en todo el ciclo de vida del proyecto
- Democracias maduras y con alto grado de confianza de sus ciudadanos
- Elecciones VAP: se toman muchas más precauciones
- Contratación de auditores expertos e independientes
- Suficientes medios a disposición de las elecciones

En cuanto a los aspectos a mejorar, destaca la detección precoz de fallos. En algunos casos ha dado la sensación que los plazos iban un poco justos (Noruega). Aún así se ha logrado un alto nivel de protección contra potenciales colusiones gracias a las medidas de auditoría implementadas.

Total: 8.5/10 puntos.

6. Desarrollo software (DSW-*n*)

Aparte de las condiciones habituales de diseño, implementación y documentación de ingeniería del software (DSW-1) se valoran una serie de aspectos detallados en el apartado 2.3f de la presente tesis. Los más relevantes son:

- Enfoque distribuido del software (DSW-2)
- Simplicidad de uso (DSW-3)
- Seguridad de acceso (evitando terceros programas) (DSW-10)
- Imparcialidad en las opciones de voto mostradas (DSW 5)
- Compatibilidad del software (DSW-9)
- Correcta implementación de las primitivas criptográficas (DSW-11)
- Posibilidad de cancelar el proceso de voto en cualquier momento (DSW-8)
- Revisión del código por parte de expertos independientes (DSW-12)
- Utilización de estándares abiertos cuando sea posible (DSW-13)
- Actualizaciones frecuentes, en especial para proteger contra los nuevos ataques y vulnerabilidades descubiertas (DSW-14)

Scytl presenta un notable desempeño en el desarrollo software. Una trayectoria de más de 15 años diseñando e implementando sistemas de VER, unido a un equipo experimentado de programadores hace que cumplan las recomendaciones en materia de:

- Enfoque distribuido del software (la inicialización y recuento de las elecciones se hace de manera distribuida usando el protocolo Shamir [430, 260]) (DSW-1) (DSW-2)
- Servicio web seguro e intuitivo (DSW-4)
- Rotura del vínculo voto-votante (uso de mix-nets verificables [369]) (DSW-7)
- Compatibilidad con las principales plataformas (DSW-9)
- Sistema actualizado (DSW-14)
- Ausencia de terceros programas (salvo en *iVote* que se subsanó) (DSW-10)
- Presentación de las opciones de voto de manera objetiva e imparcial (DSW-5)
- Acceso al código fuente (totalmente libre en el caso de Noruega [253, 256] y disponible a expertos e investigadores previa firma de un NDA) (DSW-12)

En cuanto a los aspectos a mejorar, la usabilidad (DSW-3), el desarrollo de un sistema que no suministre una prueba de la opción elegida de voto (si la legislación del país lo permite) (DSW-6) y tratar que los plazos de desarrollo tengan un poco más de margen para evitar prisas de última hora (no depende únicamente de Scytl) son apartados con potencial.

Total: 8.5/10 puntos.

7. Escalabilidad (E-*n*)

Entendida como la capacidad que tiene el sistema de VER en su conjunto (software, infraestructuras, dispositivos, recursos humanos, logística, costes) de manejar correctamente las necesidades desde el primer voto hasta su capacidad máxima sin perder calidad.

Para ello, se deberá probar el esquema en condiciones más exigentes de las reales, sin incurrir en cálculos o proyecciones teóricas (I-1) (I-3). Ello no es trivial ya que la casuística a testar es muy extensa, como explica Nore, *Project Manager* en Noruega 2011 y 2013 [262].

En el caso de Scytl, su capacidad para la escalabilidad ha quedado probada con la gestión de los comicios estatales de Nueva Gales del Sur en el que se emitieron 283.669 votos (280.573 con VER) a través de su sistema *iVote* [288, 291, 293] en lo que ha sido la mayor votación de VER hasta la fecha en la tipología de comicios vinculantes en el ámbito político, los más exigentes desde el punto de vista de coordinación y recursos necesarios.

Por tanto, su escalabilidad no es únicamente en cuanto al número de votos con las necesidades software y ex_software que conlleva, sino también en cuanto a la tipología, convirtiendo al sistema de VER de Scytl en uno de los muy pocos a nivel mundial escalable y con suficiente experiencia en comicios públicos vinculantes en el ámbito político.

Total: 9,5/10 puntos.

8. Desarrollo ex_software (DESW-2)

Entendido como el conjunto de protocolos y procesos del esquema VER aparte del software propiamente dicho (referirse al apartado 4.1a para más detalles):

- Política distribuida de credenciales, accesos, permisos y responsabilidades (DESW-2) (DESW-3)
- Protocolo de auditoría y observadores (DESW-4)
- Almacenamiento y distribución de *back-up offline* (DESW-5)
- Alternativas de voto en caso de fallar el VER (DESW-7)
- Actividades de promoción y formación en VER (*webinars*, jornadas de puertas abiertas etc.) (DESW-15)
- Protocolo de inicialización de los comicios (DESW-12)
- Envío de credenciales a través de un segundo canal (DESW-11)
- Servicio de atención al ciudadano durante el proyecto, reforzado durante el período de votación, incluyendo asistencia telefónica (DESW-14)
- Encuestas a cohortes seleccionadas de votantes para investigar tendencias, fallos y futuras mejoras. (DESW-10)

Todos los puntos anteriores requieren una elevada cantidad de recursos económicos, humanos y de infraestructuras. En el caso anterior de Helios, al tratarse de un sistema *"open-source"* de carácter más académico y enfocado a elecciones en ámbitos de menor complejidad, no se dispone de los medios necesarios para implementar una política ex_SW sólida.

Por el contrario, en el caso del sistema Scytl utilizado en EVAP, los organizadores son los propios gobiernos, quienes disponen de una gran cantidad de recursos disponibles en las líneas que se ha indicado (económicos, humanos y de infraestructuras) para minimizar en lo posible el riesgo de un fallo o ataque.

En los 3 casos estudiados, los países han aportado suficientes medios, garantizando fondos incluso para medidas tales como el envío de credenciales por un canal y de códigos por otro como en el caso de Noruega [260, 261]. También en NGS, el Gobierno provisionó partidas extra de gasto para mejorar el sistema de VER y cubrir el mayor número de votos electrónicos emitidos respecto a las expectativas iniciales [292].

La realidad es que ni siquiera poniendo todos los medios suficientes a disposición de una política robusta ex_software se puede garantizar que las elecciones vayan a transcurrir sin contratiempos. Lo que sí es cierto es que sin ellos es muy probable que algo salga mal.

Scytl presenta un desempeño satisfactorio en:

- El desarrollo paralelo al desarrollo software (DESW-1)
- Distribución de credenciales, accesos, permisos y responsabilidades (DESW-2) (DESW-3)
- Protocolo de auditoría y observadores (DESW-4)
- Almacenamiento y distribución de copias de seguridad aisladas (en Noruega contaron con 2 *datacenters* separados cientos de kilómetros entre sí) (DESW-5)
- Filosofía distribuida (DESW-6)

- Alternativas de voto en caso de fallar el VER (DESW-7)
- Protocolo de inicialización de los comicios (incluyendo *zero count protocols*) (DSW-12)
- Envío de credenciales a través de un segundo canal (DESW-11)
- Servicios de atención al ciudadano durante el ciclo de vida del proyecto, reforzado durante el período de votación [258, 262,430]. (DESW-8) (DESW-9) (DESW-14)

Por otra parte, su desempeño podría mejorarse aumentando el número de actividades de difusión y formación en VER (DESW-15), así como incluyendo encuestas post-electorales para obtener *feedbacks* directos de cara a una mejora contínua del sistema (DESW-10).

Total: 9/10 puntos.

9. Protocolo contra incidencias y ataques (PIA-*n*)

No existe ningún sistema 100% seguro. Este es un axiona por todos conocido. En algún momento hay que hacer concesiones a la usabilidad o a la limitación de recursos. Lo mismo sucede en el VER. El objetivo último es que el sistema sea lo suficientemente seguro para que a ningún atacante le valga la pena el esfuerzo o recursos a invertir por la recompensa a obtener.

En el caso de los sistemas de Scytl, la tipología de elecciones hace que susciten una mayor atención tanto por parte de atacantes como de expertos para tratar de comprometerlos. Por ello, es fundamental atender a las recomendaciones del protocolo de ataque explicadas en el apartado 4.1b de la tesis. Algunas de las más destacadas son:

- Existencia de un protocolo específico contra ataques, permanentemente actualizado (PIA-2)
- Un enfoque distribuido del protocolo de tal forma que atacar un nodo/infraestructura no sea suficiente para inutilizarlo (PIA-5)
- Modularidad del sistema para confinar en lo posible los ataques
- Medidas específicas en función de las características propias del sistema (vulnerabilidad contra ataques DoS, de ingeniería social etc.) (PIA-2)
- Mantener el máximo posible de información en servidores del país donde tienen lugar las elecciones (PIA-3)
- Políticas de *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP)* y *Control Validation Audit (CVA)* (PIA-1)
- La contratación de hackers/expertos independientes para poner a prueba el sistema previamente a su utilización real (PIA-7)
- Realizar acciones de concienciación ciudadana para que los votantes sean conscientes de su responsabilidad y adquieran formación para detectar ataques en los que ellos son el vector (*phishing*, ingeniería social etc.) (PIA-6)

En el caso de Scytl, hubo vulnerabilidades en el código en los casos de Noruega y Nueva Gales del Sur [260, 292] que podrían haber potencialmente facilitado un ataque, si bien los requerimientos para llevarlos a cabo eran bastante elevados. En el caso de Neuchâtel, las elecciones transcurrieron sin contratiempos.

Es importante también tratar de no apurar los plazos de implementación, puesto que las constricciones de tiempo han demostrado en el caso de Noruega que pueden afectar la calidad del código y el necesario tiempo de revisión y prueba [262].

El desempeño de Scytl es en general muy satisfactorio, contanto con protocolos claros y detallados contra ataques, que denominan internamente *"Business Continuity Plans"*. Se podría mejorar en lo referente a formación del votante de cara a ayudarle a evitar ataques en los que él es la víctima (*fishing*, ingeniería social etc.) (PA-6) así como la contratación de hackers y expertos independientes para tratar de comprometer el sistema previamente a su puesta en funcionamiento real (PA-7).

Total: **8/10 puntos.**

10. Versatilidad (V-*n*)

Desde la doble vertiente de la capacidad del esquema de manejar distintas tipologías de elecciones (V-1) y de la existencia de versiones adaptadas para usuarios con necesidades especiales (V-2).

También se tiene en cuenta la no necesidad de instalar programas adicionales para usar la herramienta (V-3), la existencia de versiones para navegadores y equipos con una cuota de mercado superior al 1% (V-4) y el cumplimiento del estándar WCAG 2.0 [414] (V-5).

En cuatro aspectos Scytl presenta un comportamiento satisfactorio:

- La existencia de versiones adaptadas a usuarios con necesidades especiales (Australia [292], Noruega [430]) e Instituto Municipal de Personal con Discapacidad de Barcelona (2016). (V-2)
- La existencia de distintas versiones según la tipología de elecciones (recuento homomórfico, mix-nets etc.) (V-1)
- La existencia de versiones para los principales navegadores del mercado (V-4)
- La no necesidad de instalar software específico o de conocimientos técnicos (V-3).

En cuanto al grado de cumplimiento del estándar WCAG 2.0, se realizaron al igual que en el caso de Helios tres pruebas con *iVote* y la página de Neuchâtel con tres herramientas distintas si bien con WAVE [416] se producía un error en una de ellas:

- Con Tawdis [415]

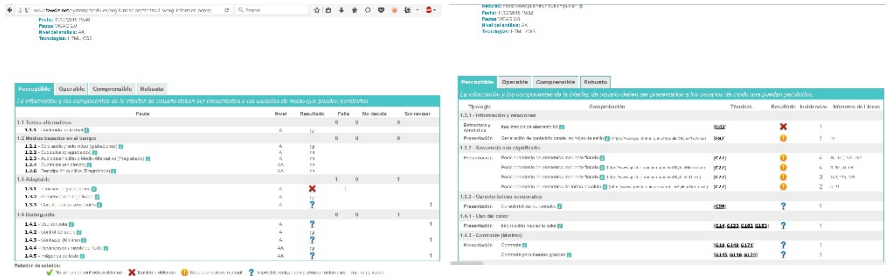


Figura 49: Scytl con respecto al estándar WCAG 2.0 según Tawdi

- Con WAVE (*Web Accesibility Evaluation Tool*) [416]

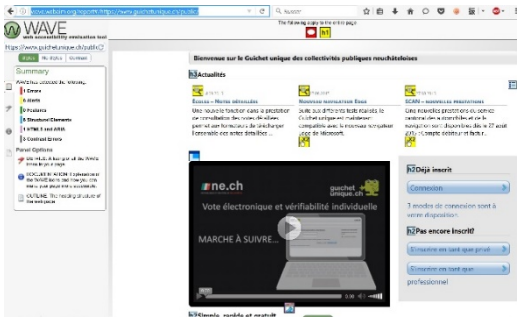


Figura 50: Scytl con respecto al estándar WCAG 2.0 según WAVE

- Con *Access Monitor* [417]

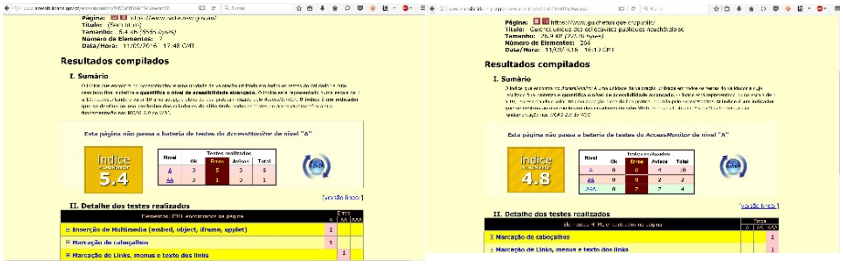


Figura 51: Scytl con respecto al estándar WCAG 2.0 según *Access Monitor*

Por tanto y al igual que Helios, no llega al nivel AA del estándar WCAG 2.0 si bien obtiene una valoración A en una de las herramientas y los errores/advertencias encontrados no son críticos. En resumen, en cuanto a versatilidad, el desempeño es altamente satisfactorio, si bien se podría mejorar en la adherencia a los estándares de diseño (V-5).

Total: 8/10 puntos.

11. Coste (C-*n*)

Este es un aspecto generalmente menos estudiado en el VER, especialmente cuando se aborda desde un punto de vista más académico o de investigación. No obstante, el objeto

último del Voto Electrónico es implantarse como una opción real que facilite el voto y lo haga más asequible y seguro.

Por ello, forma parte de un proyecto con un presupuesto asignado. Especialmente en el caso de las votaciones vinculantes en el ámbito político, suele haber suficientes fondos puesto que la prioridad absoluta es obtener un sistema suficientemente seguro, pudiéndose afrontar gastos imprevistos si surgiesen y fuesen justificados.

Las referencias que hay en cuanto a costes se han detallado en el punto 4.1d de la tesis y pertenecen en su mayoría a datos facilitados por los gobiernos dentro de su política de transparencia. En concreto, los costes vienen siendo de un mínimo de entorno a 500.000 USD [298] para un año de consultas no vinculantes y de cerca de un millón de euros en elecciones vinculantes [356, 418].

Son cantidades más propias de organizaciones gubernamentales y fuera del alcance de organizaciones profesionales y asociaciones. Por otra parte, los servicios prestados son de primer nivel, con una experiencia probada y adaptados a cada circunstancia, lo que muy pocas compañías pueden ofrecer.

En cualquier caso, una política de precios un poco más clara (C-1) contribuiría a dar a conocer el VER de una manera más atractiva y directa, potencialmente atrayendo un mayor número de clientes.

Total: 7/10 puntos.

12. Mantenimiento (M-*n*)

Entendido en una doble vertiente: el mantenimiento del sistema en sí (software y ex_software) (M-1) y la salvaguarda de datos a largo ontando con los avances en capacidad computacional, también denominada *"everlasting privacy"* (M-2).

Por lo que respecta al mantenimiento del sistema, el volumen de proyectos que maneja Scytl es muy elevado, con al menos 7 nuevos clientes en este año 2016 [422]. Ello, unido a los proyectos multianuales que gestionan, tales como los de Nueva Gales del Sur, el Ministerio de Asuntos Exteriores Francés o el cantón de Neuchâtel permiten mantener plenamente actualizados los sistemas de Scytl.

Por lo que respecta a la *everlasting privacy* [235, 241, 242], en palabras de J. Puiggalí es uno de los principales ámbitos de investigación en Scytl y se esperan avances en un plazo razonable de tiempo.

Total: 8,5/10

5.3.4 Conclusiones y valoración final

Scytl ha sido uno de los grupos pioneros en el Voto Electrónico. Comenzaron su andadura como empresa en 2001, si bien ya desde 1994 su núcleo fundador investigaba en la Universidad Autónoma de Barcelona, publicando las dos primeras tesis europeas sobre seguridad en el voto electrónico.

En la actualidad cuentan con una plantilla de más de 250 profesionales, 40 patentes y abarcan todas las disciplinas en el campo: consultoría, auditoría, formación y elecciones. En el año 2014 cerraron una ronda de financiación de más de 100 millones de USD y mantienen colaboraciones con organizaciones punteras como Swiss Post o HP. Cuentan también con una división dedicada a la I+D en el campo dirigida por Jordi Puiggalí con numerosas publicaciones relevantes, incluyendo la tesis doctoral de 2016 de la Dra. Guasch [369].

En cuanto a los proyectos realizados, su rasgo distintivo es la experiencia probada en la tipología de elecciones más exigente: las vinculantes en el ámbito político. Se han encargado del sistema de VER en varias de las elecciones más representativas en los últimos años incluyendo las de Noruega [261, 262, 263], Australia [288, 292] y el cantón suizo de Neuchâtel [369].

Destaca por un desempeño excelente en:

- Desarrollo software
- Desarrollo ex_software
- Escalabilidad
- Versatilidad
- Mantenimiento
- Auditoría/Monitorización

También ha habido incidencias: se encontraron *bugs* en el código en Noruega [263, 430] así como un programa de una tercera parte vulnerable al ataque FREAK que, en manos expertas, podría haber puesto potencialmente en peligro algunos votos [47, 48]. Los errores se subsanaron rápidamente y las últimas elecciones transcurrieron sin problemas.

Además, los puntos en los que Scytl tiene margen de mejora son:

- La usabilidad, aunque cada sistema debe adaptarse a la legislación del país. Además, la integridad o la privacidad tiene preferencia sobre la usabilidad.
- Una política de precios más transparente, con un escalón de acceso claro y económico para elecciones menores.

En resumen, Scytl ha gestionado más de 100.000 comicios en 20 países, mostrando en la actualidad un historial exitoso de elecciones gestionadas a nivel internacional de primera categoría mundial en su tipología más exigente: las vinculantes en el ámbito político.

Por todo ello, Scytl es un líder mundial de VER y una de las muy pocas empresas a nivel global con la capacidad, los recursos y el bagaje necesarios para abordar los proyectos de VER de mayor complejidad.

Desde aquí, agradecerles su total transparencia y disposición a contestar a las preguntas que se les realizaron, tanto a nivel técnico como de producto o marketing.

Para concluir con este apartado, se adjunta la fórmula de la metodología, su aplicación al sistema de Scytl y el análisis radial correspondiente para poder apreciar de una manera más gráfica las fortalezas y debilidades:

$$\sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Criterio	Ponderación	Scytl
Verificabilidad extremo a extremo	N.A.	**Δ**
Privacidad/resistencia a la coerción	N.A.	**X**
Inviolabilidad	**1.2**	8 * 1,2 = **9.6**
Usabilidad	**0.8**	7.5 * 0.8 = **6**
Monitorización/Auditoría	**1.2**	8.5 * 1,2 = **10.2**
Desarrollo software	**1.2**	8.5 * 1.2 = **10.2**
Escalabilidad	**0.8**	9.5 * 0.8 = **7.6**
Desarrollo ex_software	**1.2**	9 * 1.2 = **10.8**
Protocolo contra incidencias y ataques	**1.2**	8 * 1.2 = **9.6**
Versatilidad	**0.6**	8 * 0.6 = **4.8**
Coste	**1.0**	7 * 1.0 = **7**
Mantenimiento	**0.8**	8.5 * 0.8 = **6.8**
TOTAL	**10**	**82.6**

Tabla 21.: Metodología aplicada a Scytl

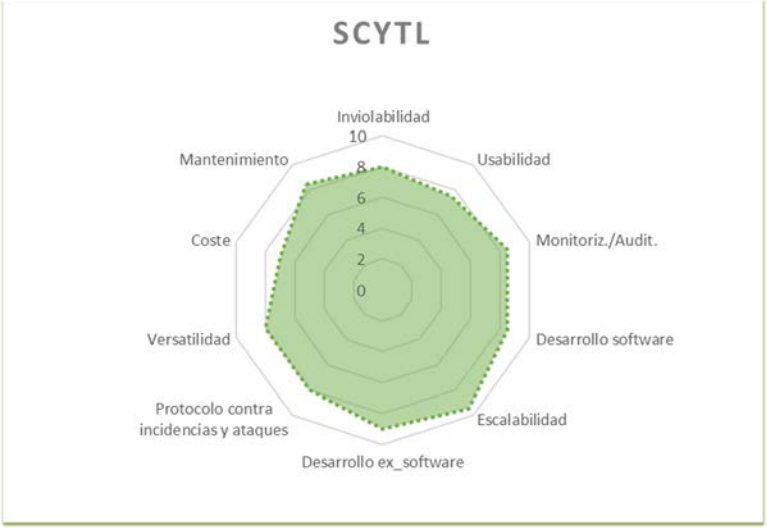


Figura 52: Análisis radial de Scytl

5.4 Agora voting/*n*Votes

5.4.1 Introducción

Agora Voting es una empresa nacida en el año 2014 cuyo germen se remonta al año 2009 en el seno del Partido de Internet [423], aunque en la actualidad son políticamente independientes. Sus fundadores son Lucas Cervera como CEO, Eduardo Robles en calidad de CTO y David Ruescas ejerciendo de ingeniero de I+D. En el año 2015 fueron una de las 18 empresas a nivel europeo en ser aceptadas en el proyecto *Impact Accelerator*, con una dotación de 100.000 euros [435].

Hasta el momento, su sistema ha sido utilizado por los partidos políticos Podemos, Ahora Madrid y Barcelona en Comú para referendos y elecciones primarias internas. Otros de sus clientes incluyen: el Ayuntamiento de Lugo, el sindicato Unión Policía Municipal Madrid, la UNED, la fundación Eurochild y la asociación ASTIC [424]. En total, su sistema se ha utilizado para emitir más de 1 millón de votos en más de 50 procesos electorales. En recientes fechas, han cambiado su nombre a *n*Votes.

5.4.2 Características

A la hora de presentar las características de *n*Votes y realizar su análisis, la información disponible es sensiblemente menor en comparación con Helios Voting y Scytl. En esos dos casos, existe una amplia bibliografía:

- Respecto a Helios, especialmente de congresos y conferencias de la comunidad científica, muchos de ellos de investigadores de primer nivel internacional.
- Por lo que respecta a Scytl, a los artículos científicos se unen las memorias técnicas y los resúmenes elaborados por los gobiernos para los que han trabajado, aportando una importante fuente extra de información.

Para complementar la información disponible públicamente, el autor se ha puesto en contacto con Eduardo Robles y David Ruescas, quienes se han mostrado siempre dispuestos a resolver las cuestiones que se les planteaba. Desde aquí agradecerles su disponibilidad.

Además, han puesto a disposición del autor un documento denominado *"Technical Overview"* [425] y otro denominado *"Protocolo de actuación - cliente"* [426] que han servido de base para el análisis del sistema.

Según lo reflejado en su *"Technical Overview"* y complementado con las conversaciones con el equipo de nVotes, la secuencia de voto es la siguiente:

1. Generación de la clave pública de la elección de forma distribuida por parte de las autoridades

2. El votante accede a la página de registro y facilita la información personal solicitada, incluyendo un código que se le ha enviado por SMS.
3. El sistema de registro comparará la información recibida con el censo y si coincide, se redireccionará a la página de VER
4. El votante cumplimenta su voto, lo encripta y lo envía. Alternativamente, en vez de enviarlo, puede auditarlo, al estido de Helios [1], pero en ese caso su voto ya no es tenido en cuenta, debiendo proceder a votar de nuevo (método *cast-or-audit*)
5. Cuando concluye el período de votación, las autoridades proceden con el mezclado y la desencriptación de los votos de manera conjunta
6. Se recuentan los votos desencriptados
7. Se publican los resultados de las elecciones, incluyendo los resultados del recuento, los cifertextos de los votos y las ZKP del mezclado y la desencriptación.
8. Los votantes y terceras partes pueden descargar y ejecutar el verificador de la elección

Respecto a los componentes del esquema, son los siguientes:

1. Registro: aplicación Python, base de datos del registro, plataforma SMS API de *Es-endex*, certificado para servidor con soporte TSL, *Fail2ban* [437] y *Cloudflare* [436] para protección contra ataques DDoS y redundancia hardware 1+1
2. Mesa electoral virtual (Polling Station): validación del servidor TLS, javascript de voto *cast-or-audit*, librerías de clientes de encriptación javascript, *Random Number Generator*, cliente de autentificación HMAC, mánager de elecciones Scala REST API, base de datos *Postgresql* y *Failban* y *Clodfare* para protección contra ataques DoS
3. Autoridad de la elección: *HTTP distributed queue*, validación de cliente y servidor TLS, librería mixnet *Verificatum* [428] y librería de tabulación *OpenSTV*
4. Verificador de elección: aplicación Python/Java

Por lo que respecta a las primitivas criptográficas, son las siguientes:

- Esquema de encriptación homomórfica ElGamal [64]
- Esquema de encriptación de umbral de Pedersen [84]
- Mixnet verificable *Verificatum* [428]
- Heurística Fiat-Shamir para transformar ZKP en NIZKP [20]
- Prueba Schnorr ZKP sin encriptar [383] para convertir el esquema en seguro IND-CCA2 (punto 2.2.4.10i).

5.4.3 Análisis

Se recuendan los 12 criterios que conforman la metodología de evaluación para sistema de voto electrónico:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

1. Verificabilidad extremo a extremo

Como se explica en el apartado 2.2.2, los componentes son: *i) Cast as intended, ii) Recorded as cast, iii) Counted as recorded,* junto con *elegibility verifiability.* [51, 93, 77, 3, 359, 369, 389].

El equipo de *n*Votes en su *"Technical Overview"* argumenta que cumple con la verificabilidad individual y la verificabilidad universal. Basándonos en el trabajo de Benaloh et al. [93], ello podría equivaler a cumplir con las tres condiciones del principio del punto.

Lo que sucede, como en el resto de los casos estudiados, es que existen una serie de precondiciones de integridad que afectan a la verificabilidad:

- El mecanismo *cast-or-audit* es utilizado por un número suficiente de votantes para que una alternación de votos no pase inadvertida.
- Los administradores de la urna electrónica son honestos
- Un ataque que controle totalmente el registro/urna electrónica es detectado

Acemyan en [231], Summers en [232] y Nueva Gales del Sur [291] han demostrado que el porcentaje de votantes que verifican su voto no es elevado y que por tanto no se debería delegar en ellos una parte relevante de la seguridad.

Respecto a las otras dos precondiciones, al igual que en el caso de Helios y Scytl, existen una serie de avances en la definición formal de verificabilidad [360, 362, 366, 369] que no han sido aplicados a soluciones precedentes por lo que *n*Votes puede considerarse E2Ev o no dependiendo de si se aceptan las precondiciones.

Conclusión: **Δ,** cumple bajo ciertas premisas (si se aceptan las precondiciones de integridad de [425] para la definición tradicional de E2Ev).

2. Privacidad/resistencia a la coerción

La privacidad entendida como resistencia a la coerción (RC, apartado 2.2.3 de la presente tesis) según la definición de Juels et al. [104] implica que un votante no puede revelar el sentido de su voto a un coercionador, incluso queriendo colaborar con él.

En el caso de *n*Votes, tras emitir el voto, el votante recibe un código de verificación por lo que el escalón previo a la RC, la ausencia de recibo se cumple parcialmente (si se aceptan las precondiciones de seguridad y honestidad del dispositivo del votante). Por tanto, el nivel superior de resistencia a la coerción no se alcanza.

Además, el administrador de las elecciones puede comprobar si un determinado votante ha votado o no, lo cual va en contra del requisito de privacidad en varios países.

Conclusión: **×,** no cumple

3. Inviolabilidad (I-*n*)

Entendida como la protección del acceso al software y a los demás sistemas mediante protocolos de autenticación seguros, evitando accesos de terceras aplicaciones y el uso de servidores vulnerables (I-1). Se valora además:

- La existencia de protocolos específicos de *risk assessment* y *threat modeling* (I-5)
- La existencia de copias de seguridad *offline* (I-1)
- Una sólida política de distribución de permisos y responsabilidades, intensificada en los nodos críticos para dificultar potenciales colusiones entre las partes (I-4)
- La modularidad del sistema para aislar errores/ataques (I-6)
- La actualización periódica de los puntos previos (I-7)

Dentro de la *"Technical Overview"* [425], existen apartados dedicados a lo que los autores denominan "incoercibilidad", incluyendo análisis sobre los riesgos a la integridad, la privacidad y la disponibilidad. En ellos, reconocen la potencial posibilidad de fraudes en el registro, *ballot stuffing* [360, 362, 366] (si los administradores son corruptos), DoS (aunque cuentan con *Cloudflare* [436] y *Fail2Ban* [437] para proteger contra ellos) y de intrusión que podrían comprometer la disponibilidad.

En esa línea, los medios se han hecho eco de una polémica surgida en torno a la integridad del censo utilizado en elecciones primarias y referendos consultivos en [431, 432, 433 y 434] y que incluía la posibilidad de que un voto no correspondiese con un votante real. Pese a ser cierto que el censo no era gestionado por nVotes, su sistema es el encargado de gestionar los comicios y cualquier polémica o duda de fraude afecta a su reputación.

En cuanto a la distribución de permisos, responsabilidades, en la actualidad la generación de claves y la encriptación (I-4) se produce de manera descentralizada utilizando el esquema de encriptación de umbral de Pedersen [84].

En lo que no hay una separación total de funciones es en el caso del recuento, puesto que la autoridad encargada del mismo puede coincidir con el propietario del censo, pudiendo dar lugar a colusiones (I-4).

Por ello, la política de inviolabilidad actualmente tiene una serie de vulnerabilidades que, en el caso de unas elecciones privadas de una asociación o grupo político, siendo preocupantes, son decisión última de los administradores de las mismas. En el caso de unas elecciones públicas vinculantes en el ámbito político, no son aceptables.

Total: 4/10 puntos

4. Usabilidad (U-*n*)

Las ideas clave en este requisito son:

- Simplicidad y claridad en el proceso de votación (U-1)
- Existencia de versiones/adaptaciones para colectivos con necesidades especiales según el Consejo de Europa [54] y la *"Convención de las Naciones Unidas sobre Derechos de Personas con Discapacidad"* [259] (U-2)
- Tasa de éxito de emisión de un voto entre votantes sin conocimientos previos de criptografía y/o seguridad informática. (U-3)

Por lo que respecta a los administradores, se valora la capacidad de crear y gestionar una elección sin conocimientos técnicos específicos. (U-5)

De todos los factores que se evalúan, Agora Voting/*n*Votes presenta un correcto desempeño en cuando a la simplicidad y claridad del proceso de votación (U-1), así como en la tasa de éxito de emisión del voto debido a su lenguaje cercano y fácil (U-3). También en lo que concierne a la labor del administrador, los menús son intuitivos y no se requieren conocimientos técnicos para crear una elección (solamente conocer la tipología de comicio y el número de candidatos a elegir) (U-5).

En cuanto a los aspectos a mejorar:

- No existe en la actualidad ninguna versión ni adaptación para colectivos con necesidades especiales (U-2)
- La autenticación por SMS puede resultar complicada para votantes de mayor edad, constituyendo un impedimento para ellos. (U-1)
- No se han realizados estudios de satisfacción para mejorar la usabilidad
- No existe una capa intermedia de usabilidad en los códigos de verificación, por lo que son largos para un votante sin conocimientos técnicos (U-1)

Por ello, su desempeño se considera bueno, pero con margen de mejora.

Total: 6/10 puntos

5. Monitorización/Auditoría (MA-*n*)

Es fundamental la existencia de un protocolo sólido de auditoría durante todo el ciclo de vida del proyecto, cumpliendo con tres propiedades principales: externo, independiente y distribuido para disminuir el riesgo de colusión entre las partes (MA-1).

En 2.3e se detallan todos los aspectos. Algunos de los más relevantes son:

- Herramientas de control de los protocolos de *risk assessment* (MA-3)
- Generación periódica de *logs* inmodificables sobre accesos y cambios (MA-4)

- Existencia de un banco de pruebas previo a las elecciones, auditoría del protocolo de ataques y su grado de cumplimiento en caso de que se produjesen (MA-7) (MA-9)
- Las actividades de monitorización/auditoría no deben poner en peligro la privacidad del votante ni el sentido de su voto. (MA-10)

En el caso de nVotes, es fundamental una monitorización continua porque en la *technical overview* [425] se comenta la potencial posibilidad de fraudes en el registro y elegibilidad, *ballot stuffing* [360, 362, 366] (si los administradores son corruptos), e intrusión.

En ese sentido y por la tipología de comicios en los que se ha utilizado su sistema (no en el ámbito político o simplemente consultivo), el protocolo de monitorización/auditoría se basa en la formación a los administradores. En palabras de los responsables al autor de la tesis, actualmente se está implemetando un documento completo que engloba todas las actividades de auditoría.

Por tanto, su desempeño en este apartado no es satisfactorio, debiendo publicar el protocolo completo sobre el que están trabajando en cuanto sea posible.

Total: 3/10 puntos

6. Desarrollo Software (DSW-*n*)

Además de las características habituales de diseño, implementación y documentación de ingeniería del software (DSW-1) se valoran los siguientes puntos (2.3f para los detalles):

- Enfoque distribuido del software (DSW-2)
- Imparcialidad en las opciones de voto mostradas (DSW-5)
- Compatibilidad del software (DSW-9)
- Correcta implementación de las primitivas criptográficas (DSW-11)
- Posibilidad de cancelación del voto en cualquier momento (DSW-8)
- Revisión del código por parte de expertos independientes (DSW-12)
- No acceso a través de terceros programas/*social media* (DSW-10)
- Implementación de estándares abiertos cuando sea posible (DSW-13)
- Actualizaciones periódicas (DSW-14)

De ellos, cumple con la imparcialidad (DW-5), la compatibilidad (DSW-9), la implementación de las primitivas (DSW-11), el uso de estándares abiertos (DSW-13), el no acceso a través de terceros programas (DSW-10), la posibilidad de cancelación (DWS-8) y la política de actualizaciones (DSW-14).

Por lo que respecta a la revisión por parte de expertos (DSW-12), el código es abierto y por tanto está a disposición de la comunidad científica, pero quizás Ágora podría alentar más revisiones por parte de expertos independientes. El caso noruego [260, 261] nos ha mostrado que los *stakeholders* se involucran mucho menos de lo previsto en la revisión del

código, hasta el punto que los organizadores tuvieron que contratar a una empresa especializada para realizarlo.

En cuanto al enfoque distribuido (DSW-2), se encuentra parcialmente implementado: aplicándose en la generación de claves y en la encriptación pero no en la separación de funciones entre el censo y el tablón, por lo que el administrador puede conocer si un votante ha votado o no (aparece en la tabla de los votantes). Si el administrador es además el responsable del censo (como puede ocurrir en elecciones privadas), se estaría ante un caso de colusión.

Por último, las primitivas criptográficas (DSW-11) están bien implementadas si bien la heurística de Fiat-Shamir [20] ha sido estudiada en profundidad y se han encontrado una serie de vulnerabilidades que podrían potencialmente poner en riesgo las elecciones, como se explica en el punto 2.2.4.7 y [90, 91, 155].

Total: 6.5/10 puntos

7. Escalabilidad (E-*n*)

Representa la capacidad del esquema VER en su conjunto (software, infraestructuras, dispositivos, recursos humanos, logística, etc) de manejar correctamente las necesidades de los comicios sin perder calidad en los servicios (E-1).

Para ello, se deberá testar el esquema en condiciones más exigentes de las reales, sin incurrir en cálculos o proyecciones teóricas (E-3). Además, se valora la capacidad del esquema de manejar elecciones vinculantes en el ámbito político (E-5).

nVotes presenta un desempeño a medio camino entre Helios y Scytl: en el primero, al tratarse de un esquema más cercano al ámbito académico y científico, las elecciones que se han realizado no han pasado de los 4.000 votos emitidos ese caso en concreto se realizó un despliegue especial en términos de infraestructura y desarrollo ex_software [49].

En cuanto a Scytl, han coordinado elecciones públicas vinculantes en el ámbito político como las de Nueva Gales del Sur en las que se han emitido más de 280.000 votos o las de la de la Asamblea de Franceses Residentes en el Extranjero 2012 con más de 1.5 millones de votos a través de su esquema de VER, erigiéndose en referencia dentro del sector.

En el caso de Ágora, han gestionado votaciones de hasta 112.000 sufragios en consultas populares de un partido político. Además, no se encargaron de la organización completa de la misma, sino que muchas de las tareas ex_software fueron controladas por el mismo partido político, con algunas carencias respecto a la distribución de responsabilidades.

En cuanto a elecciones públicas vinculantes en el ámbito político, en el momento de escribir estas líneas (noviembre de 2016) *n*Votes no tiene experiencia en ese tipo de comicios. (U-5)

Lo que sí permite su herramienta es realizar un simulacro de la votación por lo que cumplen con (U-3), al menos parcialmente.

Total: 5.5/10 puntos

8. Desarrollo ex_software

Entendido como el conjunto de protocolos y procesos del sistema VER aparte del software propiamente dicho (referirse al punto 4.1a para más detalles):

- Distribución de credenciales, accesos, permisos y responsabilidades (DESW-6)
- Protocolo de auditoría y observadores (DESW-4)
- Almacenamiento y distribución de *back-up offline* (DESW-5)
- Alternativas de voto en caso de fallar el VER (DESW-7)
- Actividades de promoción y formación en VER (webinars, jornadas de puertas abiertas etc.) (DESW-9) (DESW-15)
- Protocolo de inicialización de los comicios (DESW-12)
- Envío de credenciales a través de un segundo canal etc. (DESW-11)
- Servicios de atención al ciudadano durante el proyecto, reforzado durante el período de votación, incluyendo asistencia telefónica (DESW-14)
- Encuestas a votantes para investigar tendencias, fallos y futuras mejoras (DESW-10).

Por lo que respecta al desarrollo ex_software, cumplen con el protocolo de inicialización de comicios (DESW-12), el envío de credenciales a través de SMS y la atención al votante (DESW-11).

En este último punto, ofrecen dos niveles de servicio:

- El primero y más directo si así se contrata, en el que es nVotes mismo quien se encarga de responder en caso de cualquier duda.
- El segundo, incluye una formación al personal encargado de gestionar las elecciones, quienes reciben en primer lugar las dudas/errores. En caso de que ellos no puedan resolverlas, se contacta con el personal de nVotes que esté de *"guardia técnica"*.

Respecto a los aspectos no completamente implementados destacan:

- La distribución de credenciales, accesos, permisos y responsabilidades (DESW-6)
- El protocolo de *backup*, ya que dependiendo del cliente, nVotes se encarga de ello o se delega en los responsables de la elección (no supone una solución óptima puesto que en caso de producirse un problema con las copias de *backup*, aunque nVotes no fuese el responsable de ellas, su reputación puede verse dañada) (DESW-5)

Por último, en cuanto a los aspectos no implementados destacan:

- El protocolo de auditoría y observadores externos (DESW-4)
- Las actividades de promoción y formación en VER (DESW-15)
- La realización de encuestas dirigidas a obtener el *feedback* de los usuarios para mejorar el sistema (DESW-10)

Debido a la criticidad de algunos de los aspectos no implementados o insuficientemente desarrollados (auditoria, observadores, distribución y *backup*), el presente apartado necesita mejorarse y completarse para ser apto en elecciones vinculantes en el ámbito político.

Total: 4/10 puntos

9. Protocolo contra incidencias y ataques (PIA-*n*)

Cuanta mayor es la relevancia de las elecciones, mayor poder es transferido y por tanto mayor atención suscitan, también entre potenciales atacantes.

Por ello, la existencia de un protocolo de ataques es fundamental en una doble función: prevención y en caso de consumarse, como guía para abordar la situación de la manera más segura posible y aislando en la medida de lo posible los efectos negativos.

Los principales puntos que se examinan son los siguientes:

- Existencia de un protocolo específico contra ataques, actualizado contra los últimos ataques conocidos y adaptado a las características del sistema (PIA-2)
- El enfoque distribuido del PIA de tal forma que atacar un nodo/infraestructura no sea suficiente para inutilizarlo (PIA-5)
- Modularidad del sistema para confinar en lo posible los ataques (PIA-4)
- Mantener el máximo posible de información en servidores del país donde tienen lugar las elecciones (PIA-3)
- Políticas de *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP)* y *Control Validation Audit (CVA)* (PIA-1)
- La contratación de hackers y expertos independientes para poner a prueba el sistema antes de su despliegue real (PIA-7)
- Acciones de concienciación ciudadana para que los votantes adquieran formación en detectar ataques donde ellos son el vector (*phishing*, ingeniería social etc.) (PIA-6)

En nVotes, en parte debido al ámbito de uso del sistema hasta la fecha, no disponen de un protocolo contra incidencias y ataques. De los puntos anteriores, cumplen con el almacenamiento de la mayor cantidad de información en España (PIA-3) y parcialmente el enfoque distribuido del sistema y la modularidad del mismo (PIA-5) (PIA-4).

Además, han reforzado las medidas de seguridad contra ataques del tipo DoS introduciendo las herramientas *Cloudflare* [436] y *Fail2ban* [437] (PIA-2).

En resumen, *n*Votes debe desarrollar e implementar un protocolo completo contra ataques incluyendo los puntos destacados anteriormente. Por ahora, están poniendo en práctica una serie de acciones positivas más o menos coordinadas en esa dirección. En cualquier caso, no son suficientes para unas elecciones de tipo VAP.

Total: 4/10 puntos

10. Versatilidad (V-*n*)

Se valoran los siguientes aspectos concretos:

- Versiones adaptadas a usuarios con necesidades especiales (V-2)
- Distintas versiones según la modalidad de elecciones (sí/no, 1 de *n*, *k* de *n*, *k* de *n* ordenados etc.) (V-1)
- Versiones para los principales navegadores del mercado (V-4)
- La no necesidad de instalar un software específico ni de conocimientos técnicos para votar. (V-3)
- El cumplimiento del estándar WCAG 2.0 [414] (V-5)

Agora Voting/*n*Votes actualmente no dispone de versiones adaptadas a usuarios con necesidades especiales (V-2). Respecto al apartado de distintas versiones habilitadas para las diferentes modalidades de elecciones (V-1), el equipo desarrollador ha decidido aplicar el modelo de desencriptado y recuento con mix-nets, en principio más adaptado a comicios con un elevado número de candidatos.

Cabe destacar que según los datos aportados por Wikström en su *"Verificatum"* [428], su mix-net ofrece un rendimiento (incluso en elecciones con pocos candidatos o del estilo referéndum sí/no) lo suficientemente bueno como para plantearse la no necesidad de desarrollar otra versión basada en recuento homomórfico para elecciones menores [439]. La elección alternativa de disponer únicamente de versión homomórfica sin la opción de mix-nets es menos deseable.

Respecto al grado de cumplimiento del estándar WCAG 2.0 (V-5), como en el caso de Helios y Scytl se realizaron una serie pruebas con las mismas 3 herramientas:

- Con Tawdis [415]

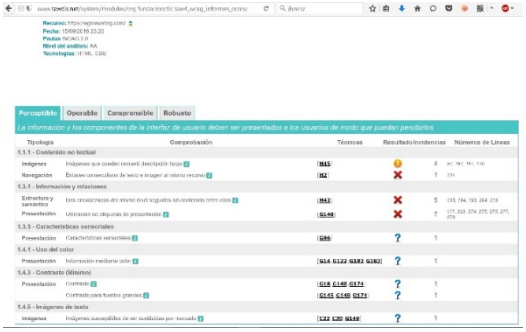


Figura 53: nVotes con respecto al estándar WCAG 2.0 según Tawdis

- Con WAVE (*Web Accesibility Evaluation Tool*) [416]

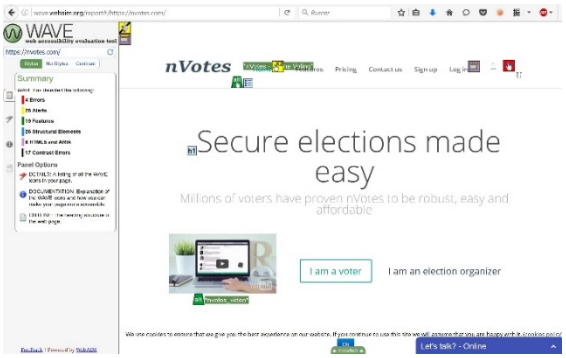


Figura 54: nVotes con respecto al estándar WCAG 2.0 según WAVE

- Con *Access Monitor* [417]

Figura 55: nVotes con respecto al estándar WCAG 2.0 según *Access Monitor*

Por tanto, al igual que Helios y Scytl, no llega al nivel AA del estándar WCAG 2.0 si bien obtiene una valoración A en una de las herramientas y una nota ligeramente superior a los otros dos sistemas estudiados. Además, los errores encontrados se refieren en su mayoría a cuestiones no críticas tales como la no existencia de textos alternativos en las imágenes.

En resumen, el desempeño es satisfactorio en cuanto a la existencia de versiones para los principales navegadores (V-4) y la no necesidad de instalar un software específico ni de conocimientos técnicos para votar (V-3).

Por otro lado, es incompleto en cuanto al estándar WCAG 2.0 (V-5) y la existencia de una versión adaptada a las distintas modalidades y no existe ninguna versión adaptada a votantes con necesidades especiales (V-2).

Total: 4/10 puntos

11. Coste (C-*n*)

Dentro del actual nicho de mercado de Agora Voting/*n*Votes (elecciones vinculantes no en el ámbito político o elecciones no vinculantes/consultivas), suele haber muy poca información concreta sobre los costes asociados.

Por ello es destacable la claridad con la que *n*Votes presenta sus costes en su página web [424, estructurando las opciones en tres planes: *Basic, Hosted* y *Managed* (C-1).

En la siguiente figura se ofrece una captura de pantalla a modo de guía:

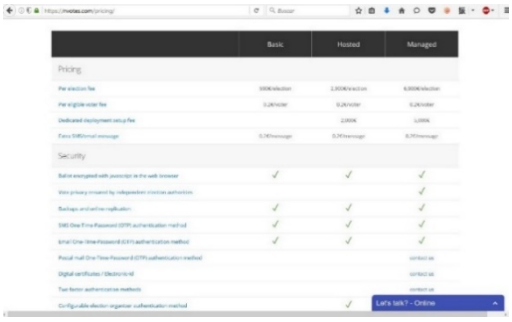


Figura 56: Coste *n*Votes [424]

Como se puede apreciar, nVotes cobra una catidad fija por lección más un coste fijo por votante (0.2€) y por email/SMS extra (0.2€). Existen 3 tipos de producto, en función de los servicios aportados por nVotes: *Basic, Hosted* y *Managed*. Además de la captura de pantalla, hay otros 40 items incluídos o no según la modalidad elegida.

En su versión más simple, se puede organizar una elección de hasta 1.000 votantes por poco más de 1.000 euros (C-2).

En ese sentido, se trata de una opción asequible y con unos costes claros para asociaciones y grupos con recursos limitados, si bien se deben tener en cuenta las limitaciones del sistema para comicios vinculantes en el ámbito político.

Total: 8.5/10 puntos

12. Mantenimiento (M-*n*)

En el doble sentido de mantenimiento del sistema software y ex_software (M-1) así como en cuanto a la seguridad de los datos a largo plazo o *"everlasting privacy"* (M-2).

Dada su filosofía *open source*, los cambios en el software son accesibles públicamente y se puede comprobar que se producen actualizaciones y mejoras contínuas. Por lo que respecta a la vertiente ex_software, queda todavía bastante labor por desarrollar aunque el equipo de *n*Votes indica que están trabajando activamente en ello (distribución, separación de roles, auditoría, protocolo contra ataques etc).

Por último, en cuanto a *"everlasting privacy",* el sistema es vulnerable a computación cuántica a 20-30 años vista si bien están investigando en esta línea.

Total: 8/10 puntos

5.4.4 Conclusiones y valoración final

Agora Voting/*n*Votes es una start-up española formada en 2014 cuyo germen se remonta al Partido de Internet en 2009. Han gestionado más 1 millón de votos emitidos en 50 elecciones, todas ellas vinculantes en otros ámbitos (no públicas vinculantes en el ámbito político) o bien consultivas/referendos.

Es de agradecer la disponibilidad de dos de los fundadores del sistema, Eduardo Robles y David Ruescas para resolver todas las preguntas que fueron surgiendo.

En relación con estos otros dos sistemas de VER, *n*Votes ocupa un espacio intermedio entre los ellos. Es *open source* como Helios, pero en lugar de ofrecer un sistema quasi-gratuito aunque difícil de implementar en unas elecciones mínimamente relevantes, ofrece una política de precios claros y asequibles aunque con algunas limitaciones en cuanto a separación de roles y riesgo de colusión.

Por otra parte, no tiene la capacidad de manejar elecciones más complejas que requieran de una infraestructura fuerte, un desarrollo ex_software sólido y testado o una serie de protocolos de auditoría o contra ataques testados en condiciones reales.

Por tanto, su ámbito se ciñe necesariamente a elecciones sin entrar en la categoría de comicios vinculantes en el ámbito político.

nVotes presenta un rendimiento satisfactorio en:

- Filosofía *open source*, facilitando la revisión por parte de la comunidad científica
- Un lenguaje simple y cercano que facilita el proceso de voto y de creación de unas elecciones por parte del administrador
- Política de precios clara
- Compatibilidad
- Estándares abiertos
- Servicio de atención al votante durante las elecciones

Por lo que respecta a los puntos con un desempeño insuficiente cabe enumerar:

- La ausencia de protocolos de auditoria/monitorización, incidencias/ataques y *backup*
- Política de distribución de credenciales, accesos, permisos y responsabilidades
- La ausencia de versiones para votantes con necesidades especiales

Además, el sistema tal y como está implementado ahora, no controla la potencial colusión entre el responsable del censo y el administrador de las elecciones, pudiendo coincidir.

Por último, el administrador de las elecciones puede saber si un votante ha votado o no, lo cuál va en contra de la condición de privacidad del voto. Ello dio lugar a una serie de

artículos en prensa haciéndose eco de la posibilidad de autenticarse con un DNI y un teléfono móvil aunque no correspondiesen a un votante real [431, 432, 433, 434].

Por todo ello, el esquema de *n*Votes no se encuentra actualmente en disposición de ser utilizado en elecciones vinculantes políticas hasta que no solucione las carencias en auditoria, monitorización, distribución, *backup* y potenciales colusiones. Su ámbito es el de comicios o referendos en asociaciones, agrupaciones profesionales, sindicatos o partidos.

Para concluir con este apartado, se adjunta la metodología, su aplicación al sistema de nVotes y un análisis radial para apreciar de una manera gráfica las fortalezas y debilidades:

$$\sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Criterio	Ponderación	nVotes
Verificabilidad extremo a extremo	N.A.	**Δ**
Privacidad/resistencia a la coerción	N.A.	**X**
Inviolabilidad	**1.2**	4 * 1,2 = **4.8**
Usabilidad	**0.8**	6 * 0.8 = **4.8**
Monitorización/Auditoría	**1.2**	3 * 1,2 = **3.6**
Desarrollo software	**1.2**	6.5 * 1.2 = **7.8**
Escalabilidad	**0.8**	5.5 * 0.8 = **4.4**
Desarrollo ex_software	**1.2**	4 * 1.2 = **4.8**
Protocolo contra incidencias y ataques	**1.2**	4 * 1.2 = **4.8**
Versatilidad	**0.6**	4 * 0.6 = **2.4**
Coste	**1.0**	8.5 * 1.0 = **8.5**
Mantenimiento	**0.8**	8 * 0.8 = **6.4**
TOTAL	**10**	**52.3**

Tabla 22: Metodología aplicada a *n*Votes

Figura 57: Análisis radial de *n*Votes

Apartado II

5.5 Civitas

5.5.1 Introducción

Civitas [67] es el primero de los esquemas del punto 5 dedicado a sistemas de VER relevantes que no han llegado a desarrollarse totalmente ni a implementarse en ningún tipo de elecciones reales. Por tanto y respecto a Helios, Scytl y nVotes, el análisis será más breve puesto que al no haberse utilizado en la práctica, la mayoría de los criterios no son de aplicación.

Civitas [67] apareció en 2008 como la implementación concreta del protocolo JCJ (Juels, Catalano y Jakobsson) [63], el cuál introdujo el concepto de resistencia a la coerción (RC), descrito en el apartado 2.2.3 de la presente tesis y que constituye una de las propiedades *sine qua non* para un sistema de VER.

El paper original ofrecía una implementación con un razonable grado de detalle, si bien quedaron algunos flecos pendientes en materia de precondiciones y manejo de credenciales que hacían difícil un uso práctico de Civitas, como se verá en el apartado 5.5.2 de características.

Con posterioridad, en 2014 apareció otro paper [441] que mejoraba algunos de los aspectos prácticos del sistema original, incluyendo la eficiencia del algoritmo a la hora de eliminar los votos duplicados, introduciendo además el uso de tarjeas inteligentes o *Smart cards*. Con todo ello, los autores llegaron a ofrecer una aproximación concreta del coste de registro y votación de cada votante.

De todos modos y pese al avance, Civitas no ha sido utilizado en elecciones reales (a diferencia de Helios, Scytl y nVotes).

5.5.2 Características

Civitas [67] fue el desarrollo concreto del esquema JCJ, pionero en la introducción del concepto de Resistencia a la Coerción: *"el votante no puede probar cómo voto o si votó, incluso en el caso de que pueda interactuar con el coercionador mientras vota"* [63].

En la siguiente figura se muestra la arquitectura de Civitas:

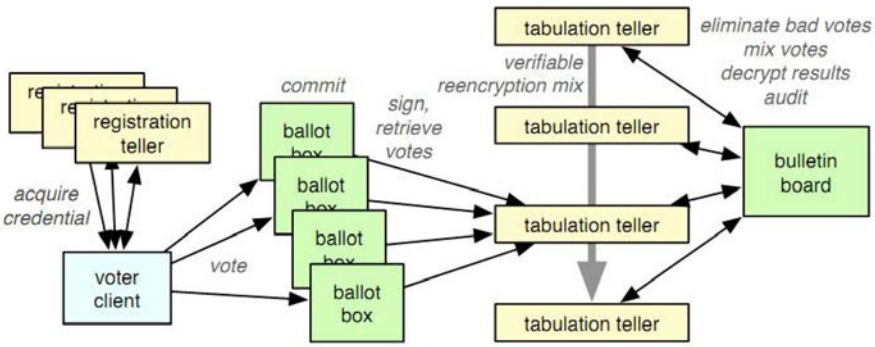


Figura 58: Arquitectura Civitas [67]

Los componentes del esquema son:

- Supervisor: El administrador de la elección. Diseña el voto (papeleta), inicia y finaliza el proceso y nombra a los responsables (*tellers*).
- *Registrar*: Autoriza a los votantes
- *Registration tellers*: Generan las credenciales que usan los votantes para emitir sus votos
- *Tabulation tellers*: Recuentan los votos

Todos estos agentes utilizan un servicio de *logs* de sólo escritura (no modificación ni borrado), de acceso público y protegidos por firma electrónica. Se utilizan varias instancias de *logs* durante una elección. Una de ellas es el *bulletin board* o tablón, mientras que el resto se denominan *ballot boxes* y son usadas por los votantes para emitir sus votos.

El protocolo está dividido en 4 fases: setup, votación, recuento y verificación:

Setup:

1. El administrador crea la elección y designa a los responsables publicando sus claves públicas individuales.
2. Los responsables de tabulación generan colectivamente un par de claves $(pk(sk_T), sk_T)$ (respectivamente $(pk(sk_R), sk_R)$) para un esquema de encriptación y se publica la clave pública.

Registro:

3. Los responsables de registro generan credenciales de votante. Cada credencial está asociada con un único votante. Para ello, se crea un *nonce* o número aleatorio de un solo uso, que se envía al votante y servirá como clave privada *d*. La parte pública se obtiene computando la encriptación de la credencial privada *d* con la clave pública

$pk(sk_T)$: $penc(pk(sk_R), m'', d)$. Esto se hace de una manera distribuida, de tal forma que ningún responsable de registro conoce el valor de ninguna credencial privada *d*.

4. Los responsables de registro publican las credenciales públicas de votante firmadas y anuncian el listado de candidatos $t = (t_1, \dots, t_l)$

Votación:

5. Cada votante elige un candidato $s \in t$ y computa dos ciphertextos $M = penc(pk(sk_T), m, s)$ y $M' = penc(pk(sk_T), m', d)$ siendo *m* y *m' nonces*. *M* contiene el voto y *M'* la credencial. A mayores, el votante contruye una NIZKP para demostrar la correcta construcción de *M* y *M'* y la validez del candidato ($s \in t$). La ZKP ofrece resistencia contra la abstención obligada por parte del coercionador. El voto es pues una tripla con los dos ciphertextos y la ZKP que se publica en el *bulletin board*.

La clave de la resistencia a la coerción de Civitas reside en que el votante puede generar credenciales falsas y usarlas en caso de que el coercionador le obligue a votar a una opción en contra de su voluntad o bien a facilitarle su credencial de voto (le entregaría una falsa).

Las credenciales falsas se crean ejecutando un algoritmo en modo local que genera partes de credencial privada falsas que para el adversario son indistinguibles. El algoritmo pide la clave privada del votante para poder ejecutarse.

Tras el período de votación, comienza la siguiente fase:

Recuento:

6. Los responsables de tabulación analizan los votos presentes en el *bulletin board* (las triplas de dos ciphertextos y la NIZKP) y descartan aquellos con ZKP inválidas.
7. Posteriormente se eliminan los votos duplicados de un mismo votante, quedando únicamente un voto por credencial. Se utilizan *Plaintext Equality Tests* [110] para comprobar si dos ciphertextos tienen el mismo texto plano sin revelarlo.
8. A continuación, se ejecuta una mix-net de re-encriptación sobre los votos para romper el vínculo voto-votante, permitiendo pues la propiedad de la resistencia a la coerción (aunque sí mantiene el vínculo dentro del voto entre el ciphertexto del voto y el de la credencial). También se ejecuta la mix-net sobre la lista de credenciales públicas publicadas por los responsables de registro.
9. Se eliminan los votos no válidos (contienen una credencial aleatoria para evitar la coerción) comparando las credenciales de los votos anonimizados con las credenciales autorizadas anonimizadas. Aquellos votos cuayas credenciales no coincidan son eliminados. La depuración de votos no válidos con PET es verificable.
10. Finalmente, los votos remanentes (pero no las credenciales) son desencriptados por parte de los responsables de tabulación utilizando *Distributed ElGamal* y publican los resultados junto con una prueba de integridad de la desencriptación.

Verificación:

11. Un verificador verifica que todas las pruebas incluidas en los votos, así como las utilizadas durante el recuento son correctas.

Premisas de seguridad

1. El adversario no puede suplantar a un votante durante la fase de registro
2. Cada votante se fía de al menos un responsable de registro
3. El canal entre ellos es inviolable
4. Los votantes se fían de sus clientes de voto
5. Los canales por los que los votantes envían sus votos con anónimos
6. Al menos una de las urnas al las que un votante envía sus votos es honesta
7. Existe al menos un responsable de tabulación honesto
8. Las premisas de DDH (2.2.4.10a) y RSA [113] se cumplen y SHA-256 implementa un Oráculo Aleatorio (2.2.4.2). (La necesidad de que se cumpla DDH proviene de que las pruebas de seguridad usadas son una reducción de dicho esquema).

La premisa 1 supone que el votante no quiere vender o ceder o no es obligado a dar su credencial al adversario. Además, en la 3 se supone que el equipo no está infectado, ni controlado por un *insider*. Lo mismo sucede con la red de conexión del equipo.

En cuanto a la cuarta, los autores fían su cumplimiento a redes de anonimización como Tor [443]. A mayores, ya se ha visto en esta tesis que Diffie Hellman tiene una serie de vulnerabilidades potencialmente peligrosas cuando se lleva a la práctica [33, 145] análogamente a lo que sucede con el protocolo RSA [145, 174, 175].

Inconvenientes de la versión inicial de Civitas

- Uno de los problemas asociados a Civitas son unas premisas de seguridad demasiado optimistas, sobre todo en lo que concierne a clientes de voto y comunicaciones.
- La eliminación de votos duplicados y credenciales no válidas lleva un tiempo cuadrático puesto que no se pueden desencriptar éstas últimas (si no ya no sería RC).
- La mejor manera de cumplir con la premisa 1 es realizar el registro en persona y en un entorno controlado, de forma que sea mucho más difícil para un coercionador tener acceso a las credenciales. El problema es que se trata de una solución poco *"Electrónica"* o *"Remota",* y en contra de la filosofía del VER. En otras palabras, el manejo de credenciales no está desarrollado suficientemente.
- No contempla medidas contra ataques del tipo DDoS aunque el sistema es vulnerable, al tener que realizar multitud de comprobaciones y filtrado de votos. Por ello, es comparativamente más fácil de colapsar que otros esquemas.
- No abordan el problema de recuperar credenciales perdidas, puesto que aportar una manera sin poner en compromiso la resistencia a la coerción es un problema abierto.

- Proyección de coste elevado (*worst case scenario* de 12USD por votante, en las elecciones presidenciales, la ciudad de Nueva York costaría más de 27.5 millones de USD).

Mejoras al Civitas original y Neumann/Volkamer proposal (NV14) [441]

En años posteriores, aparecieron mejoras a varias de las debilidades de Civitas:

- Varias propuestas han reducido la complejidad de recuento de cuadrática a lineal [444, 445].
- La vulnerabilidad contra ataques *overflooding* puesto que Civitas está basado en canales anónimos y se podría enviar un enorme número de votos hasta ralentizar o incluso colapsar el sistema. Haenni et al. plantean una solución basada en un número fijo de credenciales falsas [446].
- Respecto a la robustez de Civitas, Shirazi et al. encuentran una vulnerabilidad y plantean soluciones a ella [447].
- El problema del manejo de credenciales se aborda en [442, 448] utilizando tarjetas inteligentes o *smart cards*.

Los mismos autores de [448] junto con C. Feier y R. E. Koenig presentaron en 2014 una posterior evolución [441] basada también en *Smart Cards* para suavizar algunas de las premisas de seguridad del Civitas original y que se ha denominado NV14 en esta tesis.

Una de las principales modificaciones con respecto al esquema inicial es la fase de registro. En NV14, se distinguen dos partes: *offline* y *online*. En la fase *offline,* el votante consulta a la autoridad de registro supervisado (*SRA*) y ésta comprueba que no se encuentra bajo coerción. Posteriortmente, éste inserta su *Smart Card* en el lector y crea un código PIN.

Posteriormente, en la fase *online*, el votante se conecta de manera remota al sitio de la elección, selecciona sus responsables de registro preferidos, éstos son transferidos a su *Smart Card*, se crean las conexiones seguras entre el votante y su selección de responsables y concluye la fase de registro.

Por tanto, se añade un paso *offline* en un entorno controlado para aumentar la seguridad si bien ello choca con el concepto de VER (una parte del proceso no es remota).

En este esquema, el formato del voto es:

$$\langle \{c\}_{pk_{EK}}, \{vote\}_{pk_{EK}}, \sigma, \phi \rangle$$

Siendo $\{c\}_{pk_{EK}}$ y $\{vote\}_{pk_{EK}}$ la credencial privada y el voto del votante respectivamente, ambos encriptados con la clave pública de la elección. σ es una prueba de que $\{vote\}_{pk_{EK}}$ contiene una elección válida y ϕ es una ZKP que demuestra que el emisor conoce *c* y *vote*.

En cuanto a las premisas de seguridad, son las siguientes:

- Cada votante confía en la *SRA* y en al menos la mitad de los responsables de registro remoto
- El adversario no puede corromper ni la *Smart Card* ni el lector de *Smart Cards* ni más de *k* de *n* responsables de tabulación
- El adversario no puede tomar control del cliente de voto del votante
- El adversario no puede corromper todos los *ballot boxes*
- El adversario está restringido a computaciones en tiempo polinomial probabilístico y las primitivas criptográficas funcionan correctamente.
- El adversario no puede controlar todos los nodos de la mix-net
- En algún momento de la fase de voto, el votante no está bajo control consciente del adversario
- El votante no se olvida ni confunde su código PIN (incluir una herramienta de reenvío del PIN olvidado implica un riesgo muy importante de quebrantar la RC)

Por tanto, en NV14 siguen existiendo una serie de premisas muy exigentes, tales como la no corrupción de las *Smart Cards*, los lectores de las mismas ni el cliente de voto.

Posteriormente en el paper se realizan una serie de comprobaciones sobre el desempeño de las tarjetas en las dos fases en las que toman parte (registro y votación) dado que son el eslabón débil desde el punto de vista de la capacidad computacional.

Las tarjetas utilizadas son las *Java Card NXP JCOP J20A80G*. Con ellas y aplicando las fórmulas para registro y voto siguientes (referirse a [441] para la explicación completa):

$$t_{registration} = \left(1 \cdot t_{mul} + 2 \cdot t_{exp} + t_{RSAenc}\right) \cdot |TRT| + t_{mul} \cdot |TRT| + t_{div}$$

$$t_{voting} = 2 \cdot t_{mul} + 4 \cdot t_{exp} + 2 \cdot t_{mul} + 2 \cdot t_{exp} + 2 \cdot t_{sub} + t_{mul}$$

Para el caso de 5 responsables de registro, se obtienen unos tiempos de 15.234 seg. para el registro y 5.329 seg. para la votación.

Con ello, los autores concluyen que el uso de su modelo de tarjetas en NV14 es plausible en elecciones reales (pese a unas premisas de seguridad todavía poco realistas).

Dejan como desarrollos futuros pendientes:

- El perfeccionamiento de las premisas de seguridad
- La implementación y uso en pilotos de votación reales
- La mejora de la usabilidad del sistema
- Civitas con Criptografía de Curva Elíptica (2.2.4.10d en la tesis) para hacer las *Smart Cards* más eficientes, al necesitar una menor longitud en el tamaño de las claves.

Por tanto, Civitas es un proyecto sobre el que se están produciendo avances interesantes. En el medio plazo es probable que se implemente totalmente y se utilice en pilotos reales.

5.5.3 Análisis

Se recuendan los **12 criterios** que conforman la metodología de evaluación para sistema de voto electrónico:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

En el caso de Civitas y como sucede con el otro esquema del apartado II de este capítulo (BeleniosRF), al no haber sido utilizado en elecciones reales en ningún ámbito, muchos de los criterios de evaluación no tienen aplicación.

En concreto inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento no pueden evaluarse sin utilización real.

Por ello, se analizarán los requisitos restantes, que además son los *sine-qua-non* para cualquier sistema de VER:

1. Verificabilidad extremo a extremo

Como ya se ha indicado en el apartado 2.2.2 y en el resto de sistemas analizados, los requisitos son: *i) Cast as intended, ii) Recorded as cast, iii) Counted as recorded,* junto con *elegibility verifiability.* [51, 93, 77, 3, 359, 369, 389].

D. Zissis y D. Lekkas, en su libro *"Design, Development and Use of Secure Electronic Voting Systems"* [172] realizan un análisis de algunos sistemas de VER. En concreto, en el capítulo 4, S. Neumann y M. Volkamer analizan las vulnerabilidades de Civitas.

Es importante la puntualización puesto que son los mismos Neumann y Volkamer quienes desarrollan la última mejora de Civitas [441], denominada NV14 en esta tesis y analizada en el apartado anterior. Son por tanto expertos en Civitas.

En su análisis sobre Civitas, establecen que el sistema cumple con las propiedades de *ii) Recorded as cast, iii) Counted as recorded* pero no la de *i) Cast as intended* puesto que no es verificable (en el *Bulletin Board* salen todas las opciones encriptadas). También en [441] establecen que una de las condiciones para que el sistema sea E2Ev es que no se le pueden poner restricciones a la capacidad del adversario, lo que va en contra de varias de las premisas de NV14, entre ellas *"El adversario está restringido a computaciones en tiempo polinomial probabilístico y las primitivas criptográficas funcionan correctamente".*

Por lo que respecta a la verificabilidad de la elegibilidad o *elegibility verifiability*, el creador de Civitas, M. R. Clarkson ha co-publicado un paper sobre la materia [409]. En él, establecen

que Civitas no cumple con la propiedad porque un adversario que conozca la clave privada del tabulador podría enviar votos no autorizados.

Posteriormente proponen un concepto de *"weak elegibility verifiability"* en la que añaden la premisa de que el adversario no conoce la clave privada. En ese caso Civitas sí cumple con la propiedad.

En resumen, Civitas cumple dos de las tres condiciones tradicionales de la E2Ev: *ii) Recorded as cast, iii) Counted as recorded* y una versión debilitada de la *eligibility verifiability*. Existe por tanto un esfuerzo patente de cumplir con la propiedad y se están produciendo avances interesantes ([409] es de 2016) si bien actualmente no se puede afirmar que Civitas sea E2Ev, debido a unas premisas de seguridad demasiado restrictivas y difícilmente aplicables a unas elecciones reales.

Conclusión: ×, no cumple

2. Privacidad/resistencia a la coerción (RC)

Entendida como la propiedad por la que *"el votante no puede probar cómo voto o si votó, incluso en el caso de que pueda interactuar con el coercionador mientras vota [63]"* o bien como *"la propiedad por la que el votante no puede colaborar con un coeccionador para obtener información de cómo votó, incluso en el caso de que el votante quiera voluntariamente vender su voto"*. El apartado 2.2.3 está dedicado a la RC.

La gran aportación de Civitas ha sido precisamente en este aspecto. Está basado en el protocolo JCJ (Juels, Catalano y Jakobsson) [63], el cuál introdujo el concepto de resistencia a la coerción.

Civitas es el primer sistema de VER que consigue un elevado nivel de resistencia a la coerción utilizando credenciales falsas y Plain Equivalence Tests [110], desarrollados por 2 de los 3 autores del protocolo JCJ, Jakobsson y Juels. La explicación completa del protocolo se encuentra en 5.5.2.

No obstante, los propios autores de Civitas afirman que su sistema proporciona un mayor nivel de seguridad a la privacidad que ningún sistema desarrollado anteriormente "*under carefully articulated trust assumptions*". Es decir, bajo unas premisas de seguridad/confianza articuladas cuidadosamente.

En la práctica, ello implica un procedimiento de registro que incluye una parte en persona (por tanto no remota) como solución ideal [442]. Además, asume que el cliente de voto es incorruptible y la gestión de las credenciales aún no está suficientemente desarrollada [67, 441]. Por último, un problema sin resolver es cómo actuar en caso de que el votante pierda su credencial y solicite recuperarla.

Por todo ello, el sistema es pionero en la implemetanción de la propiedad de Resistencia a la Coerción pero todavía depende de una serie de premisas que impiden afirmar que Civitas es RC en cualquier caso.

Conclusión: **Δ,** cumple bajo ciertas premisas

5.5.4 Conclusiones y valoración final

El protocolo JCJ [63] y su implementación concreta Civitas [67] han supuesto el intento más serio de conseguir el nivel más elevado de privacidad en un sistema de VER, la resistencia a la coerción.

Ello ha sido posible gracias al uso de credenciales falsas por parte del votante para emitir un voto no válido indistinguible para el adversario en caso de tener que votar bajo su influencia. Además, el uso de la tecnología de *Plaintext Equivalence Test* [110] ha sido fundamental para conseguir su objetivo.

La versión original partía de una serie de premisas muy exigentes:

- La inviolabilidad del cliente de voto
- La existencia de canales anónimos
- La perfección de las primitivas criptográficas, incluyendo DDH [25, 33, 145] y RSA [145, 174, 175], las cuales han sido atacadas con éxito
- El procedimiento original de recuento tenía una complejidad cuadrática, si bien en [444, 445] se han presentado propuestas para reducirla a lineal
- La gestión de credenciales estaba poco detallada

Posteriormente, en [441, 442] se introdujeron mejoras con el uso de *Smart Cards* que permitieron suavizar algunas premisas de seguridad: se suponen las tarjetas y sus lectores incorruptibles, además una parte del registro es *offline* y en algún momento del periodo de votación el votante no se encuentra bajo el control del adversario.

Con esas premisas, el sistema es resistente a la coerción y además los tiempos de registro y votación con las *Smart cards* son razonables (15 y 5 seg. respectivamente).

Quedan problemas abiertos como se expone en [105]:

- Los sistemas que usan *fake credentials* como Civitas son contradictorios sobre sobre si el voto ha sido enviado con éxito. Por tanto, no se cumple la parte de la E2Ev referida a *"cast as intended"*
- Cómo resolver el caso de que el votante pierda sus credenciales

A pesar de las premisas exigentes y los problemas abiertos, Civitas y su versión más evolucionada NV14 [441] son una de las líneas más relevantes de investigación en privacidad reforzada en sistemas de VER.

Recientemente se han producido avances en cuanto a la RC con el desarrollo de una nueva herramienta criptográfica denominada *Encrypted Plaintext Equivalency Texts* (EPET) explicada en 2.2.3 y [105] con un gran potencial.

Otro aspecto positivo radica en que los propios autores de Civitas y NV14 reconocen que sus sistemas van en la buena dirección pero todavía no están listos para una utilización en elecciones reales. Con ello se evita el despliegue real de un esquema antes de estar suficientemente maduro.

En resumen, Civitas es una excelente iniciativa de VER con un enfoque sobre la privacidad reforzada (resistencia a la coerción) muy necesario en el sector. En la presente tesis, es el único esquema que obtiene una calificación de ***Δ,*** *cumple bajo ciertas premisas* en el apartado de la RC.

Los recientes avances en la línea de Civitas [105] ayudarán en el medio plazo a que surjan nuevas formas de privacidad más sólida en los sistemas de VER, aspecto clave para un mayor desarrollo y difusión del VER en todos los ámbitos.

En cuanto a la metodología de evaluación y su aplicación al sistema de Civitas, la mayoría de los criterios no son de aplicación, quedando de la siguiente manera:

$$\sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Criterio	**Ponderación**	**Civitas**
Verificabilidad extremo a extremo	N.A.	**X**
Privacidad/resistencia a la coerción	N.A.	**Δ**
Inviolabilidad	**1.2**	
Usabilidad	**0.8**	
Monitorización/Auditoría	**1.2**	
Desarrollo software	**1.2**	
Escalabilidad	**0.8**	
Desarrollo ex_software	**1.2**	
Protocolo contra incidencias y ataques	**1.2**	
Versatilidad	**0.6**	
Coste	**1.0**	
Mantenimiento	**0.8**	
TOTAL	**10**	

Tabla 23: Metodología aplicada a Civitas

5.6 BeleniosRF

5.6.1 Introducción

BeleniosRF es un esquema de VER desarrollado por Cortier et al. [72] con el objetivo de cumplir simultáneamente con la E2Ev y el mayor nivel posible de privacidad (en este caso un nuevo concepto denominado *strong receipt-freeness*), las dos propiedades *sine-qua-non* según el planteamiento de la presente tesis.

Los principales componentes en los que se basa BeleniosRF son 2:

- El concepto de firmas sobre ciphertextos aleatorizables [449] por el cual el *bulletin board* puede re-aleatorizar el voto sin modificarlo, de una forma verificable.
- En el esquema Belenios [450, 451] a su vez muy influenciado por Helios [1], en concreto por su versión con credenciales o Helios-C [377]. De hecho, el autor es el mismo en los dos casos y el nombre Belenios viene de Belenos, dios celta del sol, equivalente de Helios, dios griego del sol. Belenos + Helios = Belenios.

En palabras de sus autores, BeleniosRF es una "versión *receipt-free* de Helios". La parte *receipt-free* viene del hecho de que el votante no tiene control ni conocimiento sobre la aleatoriedad utilizada para formar el voto final almacenado en la urna o *ballot box*.

5.6.2 Características

La base sobre la que se configura BeleniosRF es Helios. Más en concreto Helios con credenciales o Helios-C [377], explicado en el punto 5.2.2.4 de la presente tesis. El motivo se debe a que su implementación, protege contra una de las mayores debilidades contra la integridad de Helios: el *ballot stuffing*.

Ello es posible por la transformación que consiguen Cortier et al. en [377] de sistemas de VER que presentan *"weak verifiability"* (necesidad de que la autoridad de registro y el *bulletin board* sean honestos para ser verificable) en sistemas con *"strong verifiability"* (verificables si la autoridad de registro y el *bulletin board* no son simultáneamente deshonestos) con unas premisas de seguridad menos exigentes y por tanto más cercanas a la realidad.

Lo consiguen retirando al *bulletin board* la capacidad de controlar el derecho a votar y añadiendo una nueva autoridad denominada autoridad de registro o *registrar authority*.

La *registrar authority* puede entonces emitir a cada votante sus credenciales de voto, que de hecho son claves de firma cuya parte pública puede incluso hacerse accesible para reforzar la verificabilidad. También puede elegirse que la generación de claves sea *offline* y se envíen al votante por correo ordinario para minimizar el riesgo de ataque on-line a la autoridad de registro.

De esa introducción de credenciales de voto para evitar el problema del *ballot stuffing* proviene el nombre *Helios with credentials* o Helios-C del que deriva Belenios [450, 451], origen de BeleniosRF.

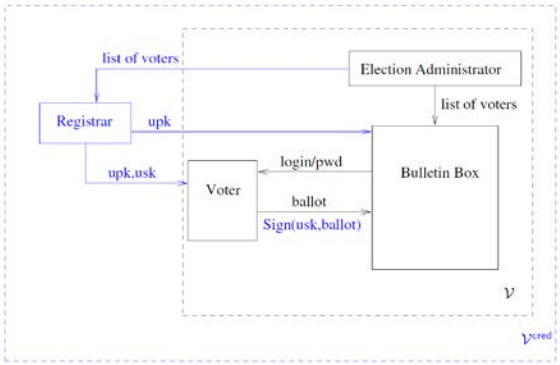


Figura 59: Construcción de la verificabilidad fuerte (anti-*ballot-stuffing*) [377]

Sobre la base de Helios-C, Glondu [450, 451] implementó el sistema de voto Belenios, cuyo proceso de voto es muy similar al Helios original [1], por tanto ni *receipt-free* ni resistente a la coerción (referirse a 5.2.3 para la los detalles).

En el transcurso de la preparación del presente apartado, el autor de la tesis se puso en contacto con la Dra. Cortier, quién muy amablemente respondió a las preguntas planteadas y ofreció la información más actualizada sobre Belenios y BeleniosRF. Desde aquí agradecer a a Dra. Cortier por su impagable colaboración.

En el transcurso de las comunicaciones, la Dra. Cortier explicó que en este momento existe una primera versión operativa de Belenios [451] que ha sido utilizada en 20 pilotos con entre 200 y 800 votos en algo menos de la mitad de ellas y unos 20 en el resto.

También indicó que la verificabilidad de Belenios se refuerza con el enfoque *audit-or-cast* de Helios en el caso de que no se asuma que el ordenador del votante es honesto, lo que a su vez compromete la resistencia a la coerción.

Por tanto, sobre la base de Belenios, se desarrolla la versión sin recibo, para reforzar la privacidad. A tal fin, se introduce la segunda tecnología que forma la base de BeleniosRF: las firmas sobre ciphertextos aleatorizables [449], que permite alcanzar la segunda gran propiedad del esquema de voto, la *"strong receipt freeness"* (ausencia de recibo fuerte).

El concepto, introducido por primera vez en BeleniosRF, se basa en la definición de privacidad de Bernhard, Cortier et. al [403] y suministra al adversario un oráculo adicional denominado $\mathcal{O}receiptLR$ que le permite emitir sus votos en lugar de cualquier votante (honesto o deshonesto).

Con ello se cumplen los siguientes escenarios:

- El votante que quiere convencer al comprador del voto cómo ha votado puede preparar su voto de una manera aletoria para construir un recibo convincente.
- Un votante que haya sido corrompido antes de la fase de envío del voto puede simplemente seguir las instrucciones del adversario
- Un votante puede grabar y también falsificar su interacción con la urna

Por tanto y según [72], la definición de la ausencia de recibo reforzada o *strong Receipt Freeness* (sRF) es la siguiente:

Dado un protocolo de voto:

$$\mathcal{V} = (Setup, Register, Vote, Valid, Append, VerifiVote, Publish, Tally, Verity)$$

Para un conjunto de identidades de votantes ID y una función resultado ρ, el esquema es sRF si existe un algoritmo $SimProof$ tal que ningún adversario eficiente puede distinguir entre los juegos $Exp_{\mathcal{B},\mathcal{V}}^{srf,0}(\lambda)$ y $Exp_{\mathcal{B},\mathcal{V}}^{srf,1}(\lambda)$.

O bien, para cualquier algoritmo eficiente $\mathcal{A}$:

$$\left|Pr\left[Exp_{\mathcal{B},\mathcal{V}}^{srf,0}(\lambda) = 1\right] - Pr\left[Exp_{\mathcal{B},\mathcal{V}}^{srf,1}(\lambda) = 1\right]\right|$$

es insignificante en λ

La única condición que piden los autores es que las acciones del votante durante la emisión del voto no sean visibles para el adversario (aunque posteriormente el votante puede aportar pruebas (manipuladas) de las acciones realizadas).

En cuanto al concepto de firmas sobre cifertextos aleatorizables, la idea es: dada una firma sobre un cifertexto, cualquiera (sin conocer la clave de firma ni el mensaje encriptado) puede aleatorizar el cifertexto y adaptar la firma a la nueva encriptación. Por tanto, la pareja de un cifertexto y una firma sobre él puede ser aleatorizada simultáneamente y consistentemente [449].

Como adaptar una firma de un cifertexto a una firma de otro cifertexto va en contra de la asunción de infalseabilidad de las firmas, los autores definen una asunción más débil: "*la infalseabilidad de firmas sobre cifertextos aleatorizables implica que la única cosa que un adversario puede hacer es producir firmas sobre encriptaciones de mensajes de los que ya conoce la firma o una encriptación; pero no sobre la encriptación de un mensaje nuevo*".

Por último, los autores extendieron su primitiva a firmas extractables sobre cifertextos aleatorizables: dada la clave de desencriptación de la firma sobre un cifertexto, se puede extraer la firma del texto sin encriptar. Ello permite al usuario de un esquema de firma ciega recuperar la firma de un mensaje después de que el firmante haya firmado su encriptación.

La siguiente figura representa de una manera gráfica el esquema de firmas extractables sobre cifertextos aleatorizables:

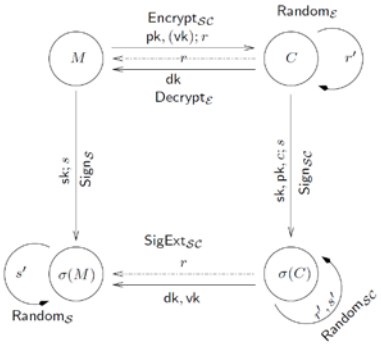


Figura 60: Firmas extractables en ciphertextos aleatorizables [449]

En él, un mensaje *M* puede ser encriptado utilizando la moneda aleatoria *r* ($Encrypt_{SC}$). El firmante puede firmar el cifertexto *C* ($Sign_{SC}$) y cualquiera puede aleatorizar el par ($Random_{SC}$).

Por otra parte, la firma del texto sin encriptar se puede obtener usando *dk* (para $SigExt_{SC}$) o las monedas *r* (si $\sigma(C)$ no ha sido aleatorizada). El resultado es el mismo que una firma de M por el firmante ($Sign_S$).

En el caso del VER, la parte del esquema que se utiliza es la siguiente:

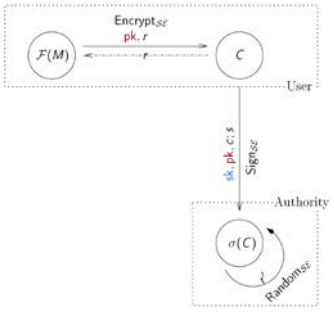


Figura 61: Firmas sobre ciphertextos aleatorizables aplicadas al VER [449]

La principal característica es un algoritmo $Random_{SE}$ que toma como *input*: *sk*,*pk*, un cifertexto *c* bajo *pk* y una firma σ sobre c válida bajo *sk* y como *output* se obtiene la realeatorización *c'* de *c* junto con una firma σ' válida sobre *c'*.

El esquema de firma está compuesto por los siguientes algoritmos: $(setup, EKeyGen, SKeyGen, Encrypt^+, Decrypt^+, Sign^+, Verify^+, Random^+)$. Para la definición detallada de cada uno, referirse a [72].

Una vez explicados los conceptos de *strong verifiability* y su relación con el *ballot stuffing* [377], la ausencia de recibo reforzada [403, 72] y firmas extractables en cifertextos aleatorizables [449], se está en disposición de definir el sistema BeleniosRF:

Sus autores lo definen como *"un protocolo de votación con ausencia de recibo reforzada que se construye a partir de [449] y [377]."* (firmas sobre cifertextos aleatorizables y Helios-C).

El proceso comienza cuando $EKeyGen$ genera el par de claves pública y privada de la elección (pk, sk) y $SKeyGen$ a su vez crea (upk, usk), par de claves de usuario.

El votante envía su voto encriptándolo con $Encrypt^+$ y las claves públicas pk y upk. Posteriormente lo firma con $Sign^+$con respecto a su clave privada usk.

Cuando la urna recibe un voto válido, lo *randomiza* con $Random^+$ y publica el resultado (el par cifertexto/firma resultante) en el *bulletin board* público *PBB*. Los votantes pueden comprobar que su voto está presente porque pueden verificar la adaptación de su firma, que es válida en sus cifertextos *randomizados*.

El recuento se hace de la forma "estándar" de los sistemas de VER: los votos encriptados se *re-randomizan* y mezclan con el algoritmo $Shuffle^+$ y se genera una prueba de corrección en la ejecución. Posteriormente se desencriptan los votos (con otra prueba de que se ha hecho correctamente) y se publican los resultados. El recuento no esta basado en propiedades homomórficas sino en mix-nets para reducir la complejidad.

BeleniosRF, $\mathcal{V}^{BeleniosRF}$ se compone de los siguientes algoritmos: $(Setup, Register, Vote, Valid, Append, VerifyVote, Publish, Tally, Verify)$ [72].

Allí, se demuestra cómo el protocolo Belenios cumple con la propiedad de ausencia de recibo reforzada y es segura RCCA [452] (un nivel inferior a CCA2 explicado en 2.2.4.10i).

Por ello, en opinión de los autores, BeleniosRF es el primer esquema de VER *strong-verifiable* y *strong-receipt-free* aunque no ha sido todavía implementado en código ni puesto a prueba en elecciones reales.

5.6.3 Análisis

Se recuendan los **12 criterios** que conforman la metodología de evaluación para sistema de voto electrónico:

Verificabilidad Extremo a Extremo ($E2E_v$), privacidad/resistencia a la coerción (RC), inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento.

BeleniosRF, al igual que Civitas, no ha sido utilizado en ninguna elección o referendo real o piloto y por tanto la inviolabilidad, usabilidad, monitorización/auditoría, desarrollo software, escalabilidad, desarrollo ex_software, protocolo contra incidencias y ataques, versatilidad, coste y mantenimiento no pueden evaluarse.

En cuanto los requisitos restantes, son la E2Ev y la privacidad/resistencia a la coerción, los *sine-qua-non* de cualquier sistema de VER:

1. Verificabilidad extremo a extremo

Como ya se ha indicado en el apartado 2.2.2 y en el resto de sistemas analizados, los requisitos son: *i) Cast as intended, ii) Recorded as cast, iii) Counted as recorded,* junto con *elegibility verifiability.* [51, 93, 77, 3, 359, 369, 389].

Los autores demuestran que cumple con la propiedad de *"strong verifiability"*, que implica que *"un esquema de VER es verificable individualmente y universalmente asumiendo que el bulletin board y la autoridad de registro no son simultáneamente deshonestas"* [377].

Por tanto, se deben de aceptar dichas premisas. Además y tal y como la Dra. Cortier, (coautora de BeleniosRF) indicó en sus comunicaciones con el autor de la tesis, otra condición para que sea E2Ev es que se asuma que el ordenador del votante es honesto. De nuevo, esto añade condicionalidad a la propiedad.

También en el propio paper original [72], se menciona que no se profundiza en la verificabilidad, al entender que la propiedad es heredada del esquema de Helios-C [377].

Con todo, Cortier afirma que han avanzado en la prueba de la verificabilidad de BeleniosRF si bien en el momento de escribir estas líneas dicho trabajo no está publicado.

Hasta que dicho paper esté disponible y se pueda estudiar, no se puede considerar a BeleniosRF como E2Ev.

Conclusión: ×*, no cumple pero representa un paso adelante interesante en sistemas de VER y podría ser que en futuras implementaciones mejorase su desempeño en E2Ev.

2. Privacidad/resistencia a la coerción (RC)

Entendida como la propiedad por la que *"el votante no puede probar cómo voto o si votó, incluso en el caso de que pueda interactuar con el coercionador mientras vota [63]"* o bien como *"la propiedad por la que el votante no puede colaborar con un coeccionador para obtener información de cómo votó, incluso en el caso de que el votante quiera voluntariamente vender su voto"*. El apartado 2.2.3 está dedicado en detalle a la RC.

En el caso de BeleniosRF, se introduce un nuevo concepto: la "ausencia de recibo fuerte", por la que un votante malicioso no puede producir un recibo probando cómo voto, sin importar si decidió actuar maliciosamente antes, durante o después de votar. Existe la precondición de que el adversario no puede ver las acciones del votante mientras está votando.

El nuevo concepto introduce una premisa que parece indicar que se sitúa en un punto medio entre la ausencia de recibo y la resistencia a la coerción. No osbstante, no existe más bibliografía de la ausencia de recibo reforzada aparte de [72], por lo que no está totalmente desarrollada/implementada. Sería además necesario un estudio de comparabilidad entre ésta y la resistencia a la coerción.

Mientras no exista esa información adicional a evaluar, no se puede afirmar que BeleniosRF sea resistente a la coerción.

Conclusión: ×*, no cumple pero representa un paso adelante interesante en sistemas de VER y podría ser que en futuras implementaciones mejorase su desempeño en RC

5.6.4 Conclusión y valoración final

Una buena parte del equipo detrás de Helios-C [377] (la versión del sistema clásico de VER que trata de evitar el *ballot-stuffing*) y Belenios [450] decidió tratar de mejorar el sistema y reforzar su protección de la privacidad. De esa manera surgió BeleniosRF [72].

Es remarcable el esfuerzo realizado, porque en muchas ocasiones los nuevos sistemas y líneas de investigación se centran en una de las dos propiedades *sine-qua-non* antagónicas entre sí, integridad o privacidad. Por el contrario, BeleniosRF desde un principio se diseña tratando de alcanzar el máximo nivel posible de ambas.

Para ello, se apoyan en:

- Helios-C/Belenios para obtener *"strong-verifiability"* (aunque no la prueban formalmente en BeleniosRF)
- El concepto criptográfico de firmas sobre cifertextos aleatorizables [449] para obtener el nuevo concepto de *"strong receipt-freeness"* o ausencia de recibo reforzada.

El uso de ambos conceptos constituye el acercamiento de BeleniosRF al desarrollo de un esquema E2Ev (*strong-verifiability*) y RC (*strong-receipt-freeness*).

En el momento de escribir estas líneas, por lo que respecta a la *strong-verifiability*, pese a que se supone que BeleniosRF la hereda de Helios-C, no está todavía probado formalmente, si bien la Dra. Cortier he comentado que está listo para publicarse. A mayores, para que sea E2Ev, se debe aceptar la premisa de que el ordenador del votante es honesto, lo cuál resulta arriesgado en opinión de una parte de los expertos en ciberseguridad.

Por lo que respecta a la *strong-receipt-freeness*, por el modo en el que está definida [72] con respecto a las premisas para que se cumpla, parece que se encuentra entre *receipt-freeness* y el nivel deseado de la resistencia a la coerción. Con todo, sería interesante que los autores ofreciesen algún tipo de estudio comparativo entre su *strong-receipt-freeness* y la RC.

Por último, el hecho de que todavía no haya sido implementado ni puesto a prueba en ningún tipo de elección o piloto hace que todavía queden incógnitas respecto al desempeño concreto en los 10 criterios de evaluación de la metodología desarrollada en la presente tesis que no han podido se aplicados.

En resumen, BeleniosRF representa una propuesta de esquema de VER que trata de dar solución a la dicotomía integridad-privacidad. Aunque actualmente no cumple con la E2Ev ni la RC, presenta unas importantes potencialidades de las que habrá que estar expectante en próximos papers y experiencias.

Como en el resto de esquemas de VER, se concluye el capítulo dedicado a BeleniosRF aplicando la fórmula y la tabla de la metodolgía:

$$\sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{n} \cdot \frac{n}{t} = \sum_{i=1}^{n} \frac{f_1 \cdot w_1 + \cdots + f_n \cdot w_n}{t}$$

Criterio	**Ponderación**	**BeleniosRF**
Verificabilidad extremo a extremo	N.A.	**X***
Privacidad/resistencia a la coerción	N.A.	**X***
Inviolabilidad	**1.2**	
Usabilidad	**0.8**	
Monitorización/Auditoría	**1.2**	
Desarrollo software	**1.2**	
Escalabilidad	**0.8**	
Desarrollo ex_software	**1.2**	
Protocolo contra incidencias y ataques	**1.2**	
Versatilidad	**0.6**	
Coste	**1.0**	
Mantenimiento	**0.8**	
TOTAL	**10**	

Tabla 24: Metodología aplicada a BeleniosRF

5.7 Votescript y los pioneros del Voto Electrónico en España

En el presente apartado se va a realizar un somero repaso al que probablemente fue el primer proyecto de Voto Electrónico *quasi-remoto* en España: Votescript.

Su concepción original y desarrollo se produjo entre los años 2.000 y 2.004 con el Profesor Doctor Justo Carracedo, exdirector de la Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación de la Universidad Politécnica de Madrid liderando el proyecto.

El sistema Votescript no entra dentro de la definición de VER de la presente tesis porque requiere de un dispositivo específico para votar (tarjeta de votación o TV) y la votación tiene lugar en cabinas instaladas en determinados emplazamientos públicos. Por tanto, su filosofía es más cercana a la del Voto Electronico tal y como se ha desarrollado en la tesis del Profesor Doctor Luis Panizo [4].

Aún así, Votescript fue pionero en la introducción de una serie de conceptos e ideas que a día de hoy son considerados estándares ineludibles en la materia como la verificabilidad individual y la verificabilidad global (restringida a los interventores, pero muy cercana al concepto actual de verificabilidad universal).

La arquitectura de Votescript es la siguiente:

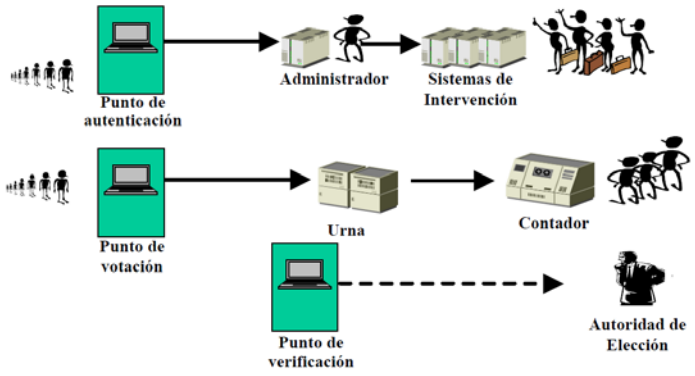


Figura 62: Arquitectura del sistema Votescript [465]

Entre las aportaciones del sistema al Voto Electrónico destacan:

- Su enfoque multidisciplinar
- Un listado inicial de requerimientos a cumplir del VER
- Los conceptos de verificabilidad individual y global
- Un sistema robusto contra la compra-venta de votos/extorsión
- La introducción de sistemas de intervención
- Un tratamiento inicial del problema de la *everlasting privacy* [235, 241, 242]

Posteriormente, a partir de los conceptos de Votescript, el grupo de investigación encargado desarrolló junto con la Subdirección General de Política Interior y Procesos Electorales y la Fábrica Nacional de Moneda y Timbre - Real Casa de la Moneda (FNMT-RCM) un proyecto conjunto para estudiar la viabilidad de la implantación del voto electrónico (o telemático en la acepción más correcta acuñada por los investigadores) para los españoles residentes en el extranjero. El proyecto se denominó sistema VERA (Votación Electrónica para los Residentes Ausentes).

Se eligió el municipio de El Hoyo de Pinares (Ávila), con un censo electoral de 1.786 personas. El objeto de la votación era decidir el día en el que celebrar la romería de la Virgen. Por tanto, una cuestión sin implicaciones políticas. Se estableció asimismo que los resultados de las elecciones serían vinculantes.

Finalmente votaron un 58% de las personas que recogieron su Tarjeta de Votación y el piloto fue un éxito desde el punto de vista técnico. Por otra parte, sí se produjeron deficiencias en cuanto al nivel de formación en TIC de los votantes, la usabilidad de la interfaz o la presencia de asistentes de voto en los centros de votación, que podrían haber comprometido el secreto del voto [465].

La experiencia VERA por tanto, refuerza lo expresado en la presente tesis: una metodología de evaluación que tenga únicamente en cuenta los requisitos tradicionales del VER es incompleta. Se deben añadir criterios adicionales prácticos derivados de la experiencia en pilotos reales para abarcar la totalidad de elementos integrantes del sistema de VER.

En 2013, la Profesora Doctora Emilia Pérez Belleboni (una de las personas implicadas en el desarrollo de Votescript) presentó su tesis doctoral sobre la aplicación del mismo sistema para el desarrollo de un esquema de voto telemático auditable y verificable a escala paneuropea [56].

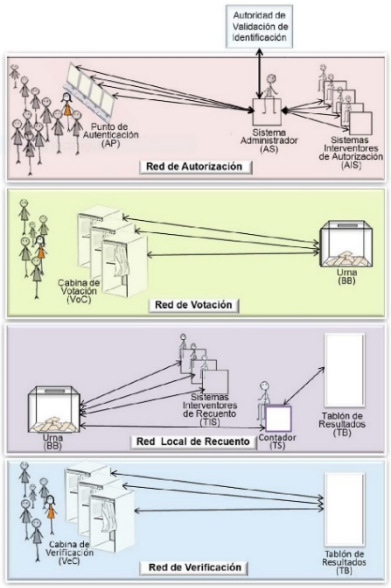


Figura 63: Esquema básico del sistema Votescript [56]

La tesis de la doctora Pérez Belleboni aborda la cuestión del voto telemático desde un prisma compartido por la presente tesis: la necesidad de enfocar los desarrollos desde una base común de aplicación (al menos) a los distintos países de la Unión Europea.

Algunas de las aportaciones más relevantes, detalladas en el capítulo 8 de su tesis son:

- Separación de la identidad del votante y la del voto (por el uso de 2 redes independientes y 2 tarjetas inteligentes)
- Garantiza que el voto emitido no podrá ser conocido en el futuro
- Identificación robusta de votantes basada en tarjeta de identidad electrónica y estudio de viabilidad de uso de *European Citizen Card* en votaciones paneuropeas
- Permite ser empleado en votaciones paneuropeas
- El sistema es resistente a ataque del tipo *eavesdropping* y *man-in-the-middle*
- Propuesta de elementos y criterios de evaluación para sistemas de voto telemático

Para el conjunto de contribuciones, la explicación en detalle del sistema y su aplicación al entorno paneuropeo, referirse al trabajo completo de la Dra. Pérez Belleboni [56].

La última gran aportación a esta tesis se resume en la siguiente figura del Dr. Carracedo:

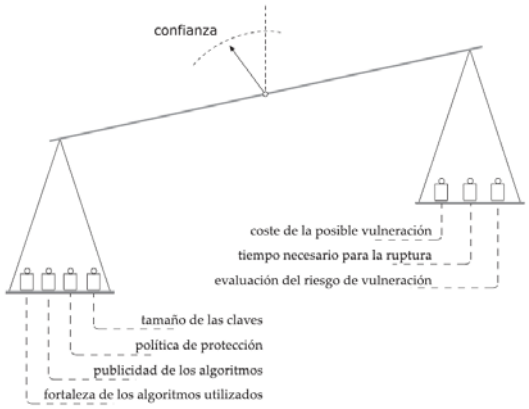


Figura 64: Confianza en los sistemas que interesan a los ciudadanos [467]

En su libro "Seguridad en Redes Telemáticas" [467], plantea una idea clave sobre el nivel de seguridad que deben tener las comunicaciones seguras para conseguir la confianza de los ciudadanos.

El Dr. Carracedo afirma que la confianza surge cuando las medidad de protección introducidas superan el peso de los riesgos de vulneración. Todo ello se resume en su muy acertada cita: *"El escudo protector debe ser de tal grado que el coste y el esfuerzo necesario para romperlo sea superior al beneficio que pueda obtener el atacante".*

Para profundizar en el sistema Votescript, referirse a [56, 465, 466, 467].

5.8 Conclusión

Siendo ésta una tesis de marcado cariz práctico, el presente capítulo 5 dedicado al análisis y comparativa de los sistemas de VER más relevantes, ha supuesto la culminación del proceso de desarrollo de una metodología holística de evaluación.

Tras estudiar los principales *building blocks* criptográficos y la seguridad del VER, definir los requerimientos tradicionales, evaluar las experiencias más relevantes de VER y finalmente establecer la metodología y añadir los factores de ponderación, se seleccionaron 5 soluciones para su evaluación en profundidad: Helios, Scytl, nVotes, Civitas y BeleniosRF.

La elección de los sistemas se ha llevado a cabo tratando de incluir aquellos que:

- Han sido más ampliamente utilizados
- Mayor atención han atraído dentro de los círculos académicos e investigadores
- Aportan algún tipo de avance relevante a la materia
- No entran en ninguno de los puntos anteriores, para tratar que la muestra sea lo más amplia posible

Posteriormente se han agrupado en dos categorías: Aquellos que han sido utilizados en elecciones/referendos reales en cualquier ámbito (Helios, Scytl y nVote) y los que todavía no, pero que aún así realizan alguna aportación relevante al VER (Civitas y BeleniosRF).

De cada uno de ellos se ha obtenido un perfil detallado, enumerando cuáles de los 75 puntos que conforman la metodología se cumplen y cuales no. A partir de ahí, se les ha asignado un valor numérico que, si bien es un resumen interesante, no supone la totalidad de la investigación realizada. Por tanto, se recomienda al lector referirse a los análisis en profundidad, incluyendo las fortalezas, debilidades y comentarios, no parándose únicamente en el valor numérico final.

Además, se ha estudiado el esquema de voto telemático Votescript y sus principales aportaciones al VER.

Cada sistema tiene su apartado correspondiente al que referirse, pero a modo de conclusión del presente capítulo se va a repasar muy someramente cada uno de ellos.

Helios

Sistema auditable, gratuito y *open source* plenamente operativo desarrollado por Ben Adida en 2008 [1], basándose en el protocolo *Simple Verifiable Voting* de Benaloh [3]. Fue pionero en agrupar las características arriba referidas y supuso un espaldarazo muy importante al VER en un momento en el que se estaban cancelando proyectos piloto en varios países.

El propio Adida limita su ámbito de utilización a elecciones con bajo riesgo de coerción, puesto que Helios no trata de dar solución a ese problema.

Otra de sus más relevantes aportaciones ha sido la de servir de base para multitud de nuevos sistemas y mejoras sobre el propio Helios (Helios-C, Helios con Mix-nets, KTV Helios, Belenios, Zeus, *Distributed Helios* etc.) que han redundado en un contínuo desarrollo tanto de sí mismo como del VER en su conjunto [360, 371, 372, 375, 377].

En cuanto a las debilidades, destaca sobre todo la falta de recursos, al tratarse de un sistema académico. Ello redunda negativamente en el ritmo de las actualizaciones así como en el tipo de elecciones que puede abarcar. Además, ha habido una serie de ataques sobre la privacidad y la integridad [360, 372, 377] que, si bien han servido de base para nuevas versiones "no oficiales", no han sido implementadas en la herramienta original [394].

En cualquier caso, ése es quizás el rol a jugar por Helios en el universo VER: una excelente base *"open source"* que atrae a varias de las mejores mentes de la comunidad científica y que sirve para testar, mejorar y desarrollar nuevas herramientas y avances.

En el ámbito de su uso real, para el caso de elecciones de bajo riesgo (universitarias, asociaciones, clubes etc.) supone una excelente opción casi gratis (siempre hay alguna necesidad mínima de recursos humanos y materiales), para un número de votantes hasta un máximo de unos 1.000-2.000. En guarismos mayores, empieza a ser necesaria una infrestructura más completa que encarece los comicios [49].

Para el análisis completo de Helios Voting, referirse a 5.2 y subapartados.

Scytl

Bajo el título genérico de Scytl se incluyen una serie de sistemas desarrollados por la compañía homónima. Fundada en el año 2001 como una *spin-off* de la Universidad Autónoma de Barcelona, su núcleo fundador investigaba sobre el Voto Electrónico desde el año 1994, llegando a publicar las dos primeras tesis doctorales de la materia en Europa.

En la actualidad, son una de las muy pocas compañías con alcance global en el mundo del Voto Electrónico Remoto. Cuentan con más de 250 trabajadores, 40 patentes y colaboraciones con empresas y organizaciones como Swiss Post, Microsoft o HP.

Otro rasgo destacado es su probada experiencia en elecciones públicas vinculantes en el ámbito político en numerosos países, incluyendo los casos de Noruega [261, 262, 263], Australia [288, 292] y Neuchâtel [369] en Suiza. El hecho de haber participado en varias experiencias dentro de la tipología más exigente (EVAP), ha atraído intentos de ataque [47, 260], si bien no se ha podido probar que hayan influido en el resultado de las mismas.

Por experiencia y recursos, presentan una excelente capacidad en: desarrollo software y ex_software, escalabilidad, versatilidad, mantenimiento y auditoría/monitorización.

Por otro lado, los puntos en lo que tienen margen de mejora son la usabilidad (sin poner en riesgo la verificabilidad o la privacidad) y una política de precios más simple y transparente especialmente para las elecciones más simples.

En total han desarrollado más de 100.000 elecciones en total en más de 20 países, erigiéndose como una de las muy pocas compañías líderes a nivel global con capacidad de implementar proyectos completos en todo tipo de comicios.

Por todo ello, de todos los sistemas analizados, Scytl es el que obtiene una mejor valoración y el único capacitado para manejar elecciones vinculantes en el ámbito político.

Por último, agradecer a Jordi Puiggalí, Vicepresidente de Investigación y Seguridad de Scytl y a su equipo por su disposición a responder a todas las cuestiones que les planteó el autor de la tesis en el desarrollo de la misma.

El lector puede encontrar en el apartado 5.3 el análisis completo del sistema Scytl.

Agora Voting/nVotes

La empresa nVotes (anteriormente Agora Voting) es una start-up española formada en 2014. Sus orígenes se remontan al año 2009 y al partido de Internet, si bien en la actualidad no tienen afiliación política [423].

En cuanto a su sistema, pese a tratarse de una empresa, implementan un enfoque *open source* con la conocida *mix-net* verificable *"Verificatum"* de Wikström [428]. Hasta el momento, no ha sido utilizado en elecciones vinculantes en el ámbito político.

Su ámbito se ha circunscrito al de referendos y votaciones internas de partidos políticos (Podemos, Barcelona en Comú, En Marea, Ahora Madrid, Sosialistisk Venstreparti), administraciones (Ayuntamiento de Lugo) y sindicatos/fundaciones/universidades (Unión de Policia Municipal de Madrid, Asociación de Ingenieros de Barcelona, UNED, Eurochild y ASTIC) para un total de 1 millón de votos gestionados.

En cuanto a las fortalezas de su sistema: la filosofía *open source* contribuye a que la comunidad científica pueda mejorarlo (aunque el caso de Noruega ha demostrado que se debe fomentar activamente para conseguirlo), es intuitivo y con un lenguaje cercano, posee una política clara de precios y presenta una buena compatibilidad.

Por lo que respecta a los aspectos a mejorar, el ámbito de elecciones en los que ha sido usado hace que no disponga de protocolos específicos de auditoría/monitorización, incidencias/ataques ni *backup*. La política de distribución de credenciales, accesos, permisos y responsabilidades podría refinarse y no existen versiones para votantes con necesidades especiales.

Además, su configuración actual permite que el responsable del censo y el administrador pueda ser la misma persona, lo cuál supondría un caso de colusión no aceptable. Por

último, el administrador puede saber si el votante ha votado o no, lo cuál compromete la privacidad del votante (por ejemplo, un votante coaccionado no puede no votar y decir al coercionador que sí lo hizo).

Unos incidentes con el control del censo tuvieron repercusión mediática [431, 432, 434]. Aunque éste no estaba bajo control de nVotes, al ser utilizado este sistema, sería recomendable mejorar el sistema para evitar pérdidas reputacionales.

Por todo lo dicho anteriormente, el sistema nVotes en la actualidad presenta una serie de debilidades que no permiten su utilización en elecciones vinculantes en el ámbito político. Su uso debería ceñirse a elecciones y referendos en asociaciones, universidades etc. e independientemente se deberían solventar los problemas de secreto del censo y colusión. Al tratarse de una compañía todavía joven, es de esperar que en un futuro vayan apareciendo versiones mejoradas de su sistema de VER.

Desde aquí agradecer a Eduardo Robles y David Ruescas su amabilidad y predisposición para resolver todas las dudas que le surgieron al sutor sobre su sistema.

Referirse al apartado 5.4 para el análisis concreto de Agora Voting/Civitas

Civitas

Se trata del primer sistema de VER analizado que no ha sido utilizado en elecciones reales. Su selección viene motivada porque se trata del más relevante intento de implementar la propiedad de la resistencia a la coerción en una solución de VER. Tomando como base el trabajo de Juels et. al [63], se consigue desarrollar un protocolo que posiblemente sea el que más cerca se ha quedado de satisfacerla.

Ello se consigue permitiendo al votante generar credenciales falsas indistinguibles para el coercionador, de tal forma que el voto emitido con esa credencial no válida será descartado durante la fase de recuento. Posteriormente, el votante puede re-votar su opción cuando ya no esté bajo el control del coercionador.

En el paper original, los prerrequisitos de seguridad eran muy exigentes y poco realistas en un caso práctico (sobre todo en lo referente a cliente de voto, canal de comunicación y gestión de credenciales), por lo que en [441] Neumann y Volkamer implementaron una versión con *Smart Cards* más cercana a la realidad en la que conseguían tiempos de 15 segundos para un registro y 5 para una votación.

Pese al gran esfuerzo realizado, se sigue partiendo se unas premisas optimistas tales como: la inviolabilidad del cliente de voto, *Smart Cards* y lectores, o la posibilidad del votante de estar a solas en algún momento de la votación. Además, quedan problemas abiertos como el cumplimiento del *"cast as intended"* de la E2Ev o el manejo de credenciales perdidas.

En resumen, Civitas constituye un acercamiento muy prometedor a la cuestión de la resistencia a la coerción pero actualmente queda todavía una importante labor por realizar antes de poder afimar que su cumplimiento está solventado.

El lector interesado puede referirse al apartado 5.5 para una descripción completa del esquema Civitas y variantes.

Belenios RF

Es un esquema desarrollado por Cortier et al. [72] que nace con el objetivo (ambicioso) de satisfacer simultáneamente con la E2Ev y la RC, siendo el primero que establece ambos como meta, no priorizando ninguno.

Los 2 principales componentes sobre los que se apoya Belenios RF son:

- El esquema Belenios (prácticamente igual a Helios-C, explicado en 5.2.2.4)
- Las firmas sobre ciphertextos aleatorizables [449], que permiten al *Bulletin Board* re-aleatorizar un voto sin modificarlo.

Se eligió Belenios como base porque, al igual que Helios-C (desarrollado también por Cortier), protege contra el *ballot-stuffing* [377].

En lo que respecta a las presunciones de seguridad, se supone que el terminal del votante es honesto y que el coercionador no puede ver las acciones del votante mientras vota.

Los autores han decidido iniciar las pruebas reales sobre Belenios (no RF) por lo que habrá que esperar un tiempo antes de poder ver algún tipo de implementación práctica.

Además, sería interesante que se presentase algún tipo de trabajo demostrando la equivalencia entre la E2Ev y la RC con los dos conceptos en teoría equivalentes que han introducido: *strong verifiability* y ausencia de recibo reforzada respectivamente. En ese sentido, la Dra. Cortier adelantó que tienen ya listo para publicar un paper en el que demuestran que BeleniosRF cumple con la *strong verifiability*.

Por tanto, BeleniosRF se trata de un esquema incipiente muy prometedor al ser el primero que aborda las dos principales características *sine-qua-non* del VER: E2Ev y RC.

Todavía falta algún tiempo para satisfacer ambas, pero desde luego habrá que estar expectantes ante una propuesta tan prometedora.

Desde aquí el autor quiere agradecer a la Dra. Cortier por su disponibilidad para resolver las cuestiones que fueron surgiendo dutrante el estudio de BeleniosRF.

Los lectores interesados disponen en el apartado 5.6 del estudio completo de BeleniosRF.

Votescript

Originariamente desarrollado entre los años 2.000 y 2.004 por el Profesor Doctor Justo Carracedo y su equipo, las aportaciones de Votescript en el ámbito del Voto Electrónico (o telemático como lo denominan sus autores) son muy relevantes, más si cabe considerando los años en los que se desarrolló el proyecto.

Teniendo en cuenta tanto el período original 2.000-2.004 como la tesis de la Dra. Pérez Belleboni [56] en 2013, las principales son:

- El enfoque multidisciplinar
- La definición de un listado inicial de requerimientos a cumplir del VER
- Los conceptos de verificabilidad individual y global
- Un sistema robusto contra la compra-venta de votos/extorsión
- La introducción de sistemas de intervención
- Un tratamiento inicial del problema de la *everlasting privacy* [235, 241, 242]
- La separación de la identidad del votante y del voto (a través de 2 redes independientes y 2 tarjetas)
- La identificación robusta de votantes basada en *ID Card* electrónica y estudio de viabilidad de uso de *European Citizen Card* en votaciones paneuropeas
- Permite ser empleado en votaciones paneuropeas
- La resistencia a ataques *eavesdropping* y *man-in-the-middle*

Además, el Dr. Carracedo en su libro "Seguridad en Redes Telemáticas" [467], plantea que la confianza en los sistemas de VER surge cuando las medidas de protección introducidas superan el peso de los riesgos de vulneración (ver figura 64); totalmente en la línea de lo expresado por el autor de la presente tesis en las conclusiones.

Por todas las aportaciones detalladas, se ha querido introducir el apartado sobre Votescript en esta disertación pese a no ser estrictamente un sistema de VER de acuerdo a la definición utilizada (se utiliza una tarjeta específica para votar y la acción de voto tiene lugar en un entorno controlado).

La visión y conceptos introducidos por el equipo desarrollador de Votescript se han convertido en estándares hoy en día. Su trabajo sin duda allanó el camino a todos los proyectos que vinieron después. Para los lectores interesados en profundizar en el sistema, referirse a [56, 465, 466, 467].

Parte IV

Conclusiones con respecto a los objetivos, mejoras propuestas y líneas futuras de trabajo

El futuro tiene muchos nombres:
para los débiles es lo inalcanzable,
para los temerosos lo desconocido,
para los valientes, es la oportunidad
-Víctor Hugo

C a p í t u l o 6

CONCLUSIONES Y APORTACIONES CON RELACIÓN A LOS OBJETIVOS. MEJORAS PROPUESTAS Y LÍNEAS FUTURAS DE TRABAJO

七転び八起き

Caerse siete veces, levantarse ocho

-Proverbio japonés

6.1 Conclusiones y aportaciones con respecto a los objetivos

Avanzando sobre la tesis del Dr. Panizo centrada en el análisis y clasificación de sistemas de voto electrónico en entornos controlados, la presente disertación tiene como objetivo último tratar de responder a las siguientes dos cuestiones:

¿Existe en la actualidad algún sistema/tecnología de Voto Electrónico Remoto lista para ser implantada en procesos electorales?

y de ser así,

¿Bajo qué condiciones y hasta qué punto en términos de nivel de uso, tecnología y tipología de elecciones sería suficientemente segura su introducción?

En el capítulo 1 se ha explicado cómo la aplicación de Tecnologías de la Información a los procesos electorales implica una serie de dificultades añadidas con respecto a otros sectores como el de banca on-line o el del comercio electrónico, a saber:

- La necesidad de asegurar simultáneamente la integridad y la privacidad reforzadas, antagónicas entre sí, al menos en parte [1, 23, 260, 388]
- Conseguirlo en el largo plazo (ataques descubiertos tras concluir los comicios)
- Lo que está en juego es sumamente valioso y difícilmente revertible en caso de fraude. Como se dice en [23]: *Election Security is National Security*
- La existencia de un sistema tradicional fácil, intuitivo y verificable que funciona razonablemente bien

De una manera más gráfica, si se produce algún error en una transferencia bancaria, existe un registro de operaciones que se puede verificar, comprobando dónde se ha producido el fallo y tomando las medidas pertinentes para su resolución. En el caso del VER, dicho registro no existe o no debería existir, puesto que estaría quebrantando el secreto de voto.

A ello se unen tres potenciales vectores de ataque que son: el equipo del votante, la red y el propio sistema de VER:

- En el caso del dispositivo del votante, se estima que entre un 30 y un 40% de los equipos personales están infectados por algún tipo de *malware* [23]. Además, éste se encuentra en un entorno no controlado, dificultando la protección de la privacidad.
- En lo que respecta a la red, existe una notable variedad de ataques que han tenido como objetivo los protocolos criptográficos asociados [26, 33, 156, 170, 339, 371].
- A mayores y en cuanto al propio sistema de VER, se han identificado varias vulnerabilidades que podrían poner en peligro unas elecciones [87, 90, 105, 145, 155, 377].

Por último, existen precedentes de intentos reales de ataques en elecciones vinculantes en el ámbito político con implementación del VER, en Australia, Estados Unidos y Ucrania entre otros [26, 33, 236, 238, 287, 464].

En resumen, a los peligros habituales de cualquier actividad on-line se suman unos requerimientos más exigentes de seguridad y un fuerte efecto llamada para potenciales atacantes por la trascendencia de lo que está en juego.

Por otra parte, los beneficios potenciales de la introducción del VER son enormes:

- Mejor acceso general al voto, especialmente para colectivos como: residentes en el extranjero, personas con discapacidad motora, visual, temporalmente ausentes etc.
- Ahorro de costes con respecto al procedimiento tradicional
- Mayor implicación y participación ciudadana en la vida pública
- Si se desarrolla correctamente, podría incluso incrementar la seguridad y transparencia (en caso contrario supone una excelente plataforma para manipular las elecciones)

En cualquier caso, para poder tomar una determinación sobre el VER es necesario hacer un análisis ordenado, protocolizado y exhaustivo del mismo. Eso es precisamente lo que se ha tratado de realizar en esta tesis.

Para ello, se ha implementado un proceso que, comenzando con el análisis de las principales primitivas criptográficas, ha culminado con la aplicación de la metodología desarrollada a los sistemas de VER más destacados hasta la fecha.

Los pasos seguidos, tal y como se ha explicado en el apartado 1.3 de objetivos han sido:

1. Estudio y armonización del estado del arte y la seguridad del VER

En el capítulo 2, se han repasado los principales conceptos y primitivas matemáticas y criptográficas de aplicación al VER, constatando que quedan problemas abiertos sobre todo a la hora de trasladar el funcionamiento matemáticamente perfecto de las mismas a la realidad. También se han introducido los últimos avances en la materia: Homomorfismo total y criptografía de curva elíptica y de retículo, si bien todavía no se han implementado en ningún esquema concreto.

Por último, se ha realizado un análisis de la seguridad del VER junto con un estudio sobre la tipología y los ataques más relevantes hasta la fecha.

Aportación 1

Establecimiento de una base matemática y criptográfica formal unificada sobre la que construir una metodología de aplicación a los distintos sistemas de VER, así como a las diversas idiosincrasias de cada país.

2. Definición de requisitos tradicionales homogéneos del VER

El artículo 68 de la Constitución Española establece que *"El Congreso se compone de un mínimo de 300 y un máximo de 400 Diputados, elegidos por sufragio universal, libre, igual, directo y secreto"*. Análogamente, el Consejo de Europa en [358], también afirma que: *"The five key principles of electoral law are: universal, equal, free, direct and secret suffrage and they are at the root of democracy"*.

Por tanto, los requisitos *sine-qua-non* son aquellos que salvaguardan esas 5 propiedades: La verificabilidad extremo a extremo en lo que respecta al sufragio universal, libre, igual y directo y la resistencia a la coerción, como garante del voto secreto.

Pese a que existe poca bibliografía respecto a la equivalencia de requerimientos legales en condiciones técnicas, el autor se apoyó en los siguientes trabajos para definir el resto de requerimientos tradicionales:

- KORA (*Concretization of Legal Requirements* en alemán) [461]
- *Common Criteria for IT Security Evaluation SO/IEC 15408:2009* [460]
- ISO 27001/IT-Grundschutz [462]
- Sistema combinado de los 3 anteriores por Simic-Draws et al. [457]
- "*Recomendaciones sobre certificación de sistemas de (sic) e-voting*", Directorado General de Democracia y Asuntos Políticos del Consejo de Europa [456]
- El trabajo de Volkamer sobre requisitos legales del voto [459]
- El trabajo de Bräunlich et al. sobre la transformación de criterios legales en TDGs (*Technical Design Goals*) [463]
- La tesis doctoral de Neumann [458], que se apoya en [457] y [463] para identificar 16 aspectos técnicos que debería cumplir un sistema de *i-voting*

Aportación 2

Definición de los 5 requisitos (41 aspectos) de todo sistema de VER de acuerdo a las metodologías arriba enumeradas. Resumido en las figuras 15 y 16 y en la tabla 1.

3. Estudio cualitativo y cuantitativo de las principales experiencias de Voto Electrónico Remoto

En total se han investigado más de 500 elecciones y 6 millones de así como los ataque se incidencias ocurridos en todos los ámbitos en un total de 12 países.

Aportación 3

Estudio cualitativo y cuantitativo de las experiencias más destacadas de VER cuyo resumen se recoge en la tabla 12.

4. Definición de criterios adicionales para la metodología no cubiertos por los requisitos tradicionales.

Aportación 4

Sobre la base de la aportación 3, se definen el resto de criterios que van a formar parte de la metodología de evaluación, junto con sus partes integrantes hasta totalizar 32 requerimientos técnicos. Todo ello se recoge en la figura 36.

Posteriormente, en la figura 37 se reproduce en conjunto de criterios de la metodología, con sus subdivisiones y las relaciones entre ellos.

5. Consulta a expertos tanto de la industria como de la comunidad científica para revisar la metodología y añadir factores de ponderación a los criterios

Al no tener todos los criterios la misma importancia, se realiza una encuesta técnica entre 31 expertos internacionales (seleccionados utilizando las metodologías de sampleado *snowball* [454] y *judgement* [455]) en el campo del VER, obteniendo *feedback* de 21 de ellos con el fin de recoger su *input* sobre los factores de ponderación a añadir y ofrecer una mejora extra a la metodología.

Aportación 5

En el anexo B se reproduce la encuesta enviada en sus versiones en español e inglés que ha servido para introducir los factores de ponderación a la metodología.

6. Definición formal de la metodología

Compuesta finalmente por 2 requisitos *sine-qua-non* (garantes de las 5 propiedades intrínsecas a una votación democrática [66, 358]) y otros 10 cuantificables, que suman un total de 73 puntos de evaluación con su correspondiente codificación y ponderación asociadas, detallados en el anexo A.

Aportación 6

Integrando las aportaciones 2, 3, 4 y 5, se define la metodología completa de evaluación de sistemas de Voto Electrónico Remoto.

El esquema íntegro de la metodología se presenta en la figura 38 y en la tabla 14.

7. Aplicación de la metodología a los 5 esquemas de VER más relevantes:

Una vez desarrollada la metodología, se ha aplicado a los 5 sistemas de VER más importantes: Helios, Scytl, Agora/nVotes, Civitas, BeleniosRF.

Para cada uno de ellos, se evalúan de los 73 puntos cuantificables de la metodología más los 2 *sine-qua-non*, ofreciendo un resumen final de evaluación con sus fortalezas, debilidades y recomendaciones junto con una tabla de valoraciones numéricas finales y una figura radial para un análisis gráfico más intuitivo.

Aportación 7

Evaluación completa de los sistemas Helios, Scytl, nVotes, Civitas y BeleniosRF.

Aportación 8

Repaso del sistema de voto telemático Votescript. Aportaciones al VER en los ámbitos de verificabilidad, niveles de protección y requisitos a cumplir entre otros.

Para concluir, se presenta la comparativa final y tabla resumen a continuación:

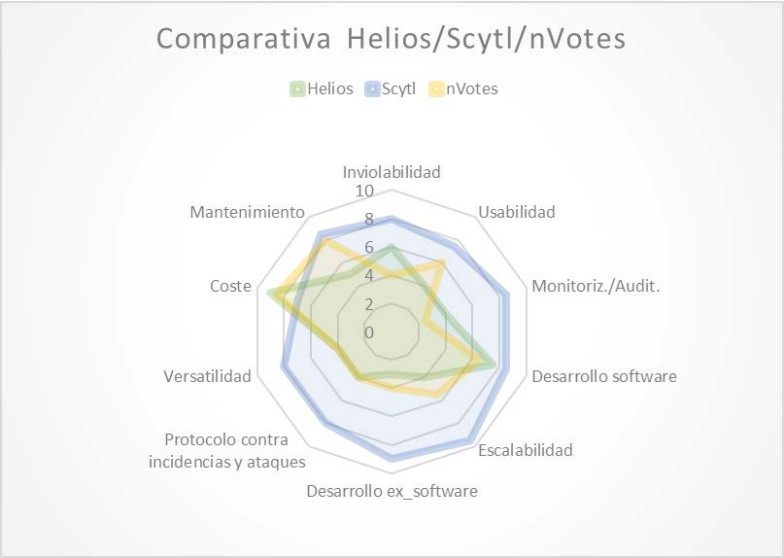


Figura 65: Comparativa Helios/Scytl/nVotes

Criterio	**Codificación**	**Ponderación**	**Helios**	**Scytl**	**nVotes**	**Civitas**	**BeleniosRF**
Verificabilidad extremo a extremo	E2Ev	N.A.	**Δ**	**Δ**	**Δ**	**X**	**X***
Privacidad/resistencia a la coerción	RC	N.A.	**X**	**X**	**X**	**Δ**	**X***
Inviolabilidad	(I-*n*)	**1.2**	6 * 1.2 = **7.2**	8 * 1.2 = **9.6**	4 * 1.2 = **4.8**		
Usabilidad	(U-*n*)	**0.8**	4 * 0.8 = **3.2**	7.5 * 0.8 = **6**	6 * 0.8 = **4.8**		
Monitorización/Auditoría	(MA-*n*)	**1.2**	4 * 1.2 = **4.8**	8.5 * 1.2 = **10.2**	3 * 1.2 = **3.6**		
Desarrollo software	(DSW-*n*)	**1.2**	7.5 * 1.2 = **9**	8.5 * 1.2 = **10.2**	6.5 * 1.2 = **7.8**		
Escalabilidad	(E-*n*)	**0.8**	4 * 0.8 = **3.2**	9.5 * 0.8 = **7.6**	5.5 * 0.8 = **4.4**		
Desarrollo ex_software	(DESW-*n*)	**1.2**	3 * 1.2 = **3.6**	9 * 1.2 = **10.8**	4 * 1.2 = **4.8**		
Protocolo contra incidencias y ataques	(PA-*n*)	**1.2**	4 * 1.2 = **4.8**	8 * 1.2 = **9.6**	4 * 1.2 = **4.8**		
Versatilidad	(V-*n*)	**0.6**	4 * 0.6 = **2.4**	8 * 0.6 = **4.8**	4 * 0.6 = **2.4**		
Coste	(C-*n*)	**1.0**	9 * 1.0 = **9**	7 * 1.0 = **7**	8 * 1.0 = **8.5**		
Mantenimiento	(M-*n*)	**0.8**	5 * 0.8 = **4**	8.5 * 0.8 = **6.8**	8 * 0.8 = **6.4**		
TOTAL		**10**	**51.2**	**82.6**	**52.3**		

Tabla 25: Aplicación de la metodología de evaluación a las soluciones de VER más destacas

Una vez presentadas todas las aportaciones, finalmente se está en disposición de responder a las dos grandes preguntas que dan sentido a la presente tesis:

¿Existe en la actualidad algún sistema/tecnología de voto electrónico remoto lista para ser implantada en procesos electorales?

Si se entiende como implantación en procesos electorales la utilización en elecciones públicas vinculantes en el ámbito político para la totalidad de la población, la respuesta es no. Actualmente no existe ningún sistema que desde la presente tesis se juzgue como suficientemente seguro para ser utilizado como sustituto del voto tradicional en unas elecciones legislativas de carácter nacional.

Ello se debe a que ninguno de los esquemas evaluados ha conseguido satisfacer las dos propiedades *sine-qua-non* que protegen los 5 principios del voto democrático recogidos en la Constitución y el Consejo de Europa [66, 358]: la E2Ev y la RC.

En cambio, si como implantación en procesos electorales se entiende un desarrollo limitado, acotado y progresivo, tanto en tipología de elecciones como en porcentaje de la población, entonces la respuesta es que posiblemente sí y se puede pasar a la siguiente pregunta:

¿Bajo qué condiciones y hasta qué punto en términos de nivel de uso y tipología de elecciones sería suficientemente segura su introducción?

La propia naturaleza del voto implica un equilibrio o *"trade-off"* entre la integridad (con su verificabilidad asociada) y la privacidad. Una privacidad absoluta implica una verificabilidad incompleta y viceversa.

Asumiendo que un sistema 100% seguro no existe (y menos todavía en un entorno como internet), el objetivo es ofrecer **una solución de voto electrónico remoto lo suficientemente segura como para que al atacante no le valga la pena la inversión a realizar en comparación con el beneficio a obtener**. Ello guarda una estrecha relación con la figura 64 del apartado 5.7, ideada por el profesor Carracedo, pionero del VER en España.

Por tanto, en las elecciones con más en juego (las estatales vinculantes en el ámbito político), el potencial beneficio de un atacante modificando el resultado es tan importante que se puede esperar una gran inversión por su parte. Por ello, el VER está menos recomendado para esta tipología de comicios.

Si aún así, se quiere introducir (el VER) de todos modos, se deberían seguir las siguientes recomendaciones:

- Acotar su implantación a segmentos específicos de la población con necesidades especiales como votantes con limitaciones físicas o quizás residentes en el extranjero.

- Siempre de la mano de un socio tecnológico de probada capacidad (requisito E-5 en la metodología). En el caso de los sistemas analizados en la presente tesis, únicamente Scytl ofrece un nivel suficiente de *expertise*.

- Incluso contando con un *partner* experimentado, se deberán extremar las precauciones, suministrando suficientes medios tanto humanos como materiales y financieros al equipo encargado, que deberá estar formado por expertos en la materia y sin estar sometido a presiones externas de carácter político o mediático.

  En ese sentido, los ejemplos de Noruega [262, 430], Finlandia [324], Suiza [283] o Nueva Zelanda [331, 332] son los referentes que establecen el nivel mínimo deseable de transparencia y profesionalidad a lo largo de todo el ciclo de vida del proyecto.

  También el caso de los Estados Unidos en 2010, abriendo un período de prueba en el que se invitaba a expertos y hackers a evaluar e incluso atacar el sistema [311], aporta una práctica altamente recomendable en opinión del autor (requisito PIA-7).

  Adicionalmente, un esquema al estilo del suizo [283] que incluye una serie de "exámenes" a los distintos sistemas de VER, en función de los cuales se les concede una licencia temporal renovable para poder presentarse a las licitaciones de VER hasta un cierto porcentaje de votos supone, en opinión del autor otra de las iniciativas a implementar por lo acertado de su planteamiento.

- Por último, el Voto Electrónico Remoto debe de ser una opción adicional de votación, no la única. Tras el período habilitado el VER, todo votante debe poder votar de la manera tradicional (aunque haya votado también previamente a través del VER), prevaleciendo siempre el voto tradicional sobre el electrónico. Esta es una de las medidas más recomendables contra la coerción (DESW-7 y DESW-9).

- En el indeseable caso de producirse un ataque o imprevisto, el principio absoluto a proteger es la privacidad del votante y el voto, llegando a la cancelación completa de los comicios si así fuese necesario (MA-10 en la metodología).

En lo que respecta a elecciones vinculantes en otros ámbitos distintos al político (compañías, universidad, asociaciones, clubs, sindicatos, partidos políticos etc.) o bien a referendos no vinculantes, se puede aplicar el mismo principio de ofrecer **una solución de voto electrónico remoto lo suficientemente segura como para que al atacante no le valga la pena la inversión a realizar en comparación con el beneficio a obtener.**

En los casos arriba citados, la ganancia a obtener por un potencial atacante es menor, como también lo suelen ser los medios a su alcance. Por lo tanto, es más plausible introducir el VER en estos ámbitos, teniendo siempre en cuenta que ningún sistema es 100% seguro y que se recomienda mantener el voto tradicional al final del proceso.

Después de todo, siguen existiendo una serie de problemas sin resolver en cuanto a la verificación extremo a extremo y la resistencia a la coerción. Aunque están apareciendo primitivas y esquemas que parecen avanzar en la buena dirección, su implementación completa llevara todavía algunos años.

Una vez expuestas las conclusiones, resta únicamente apuntar que el asunto del Voto Electrónico Remoto no deja indiferente a nadie. Hay defensores acérrimos como el caso de Estonia, así como asociaciones [12] e investigadores firmemente en contra como Rubin, Halderman o Teague. También hay quien manifiesta posiciones intermedias como Benaloh, o el Dr. Justo Carracedo, favorables al voto electrónico en entornos controlados o bien a un uso únicamente en elecciones fuera del ámbito político como Adida [1].

Huyendo de los extremos, el autor de la tesis comparte la idea de que el VER tiene un papel que cumplir en los procesos electorales futuros, como sistema adicional pero no sustituto del tradicional. Después de todo, el sufragio universal es un derecho muy reciente en la historia de la humanidad y merece ser protegido con el mayor de los celos.

Lo importante es que su introducción sea transparente, con recursos suficientes, paulatina tanto en porcentajes como en tipologías, opcional y sin presiones externas a criterios estrictamente técnicos y profesionales. Por suerte, hay suficientes espejos en los que mirarse para realizar un buen trabajo.

En ese sentido, con la presente tesis el autor ha tratado de contribuir a la creación de una necesaria base común en forma de metodología práctica de evaluación de sistemas de Voto Electrónico Remoto. Ésta supone un punto de partida sobre el que construir y adaptar a la idiosincrasia concreta de cada país o situación particular, pero sobre unos principios firmes y universales que deben regir los sufragios a cualquier nivel.

6.2 Mejoras propuestas y líneas futuras de trabajo

A lo largo de la tesis ha quedado patente que *"Election security is national security"* [23] y que el VER conlleva unas dificultades extra respecto a otros ámbitos de utilización de las TIC.

En el momento actual, persisten una serie de retos que conforman líneas futuras de trabajo destacadas. De su desempeño futuro dependerá en buena medida el papel que el VER desempeñará en años venideros:

- Los avances criptográficos, con el encriptado homomófico completo, las criptografías de curva elíptica y de retículos y la computación cuántica como exponentes de nuevas herramientas dirigidas a reforzar la seguridad del VER (capítulo 2 de la tesis).
- La usabilidad, puesto que en la actualidad persisten tasas elevadas de votantes que no pueden emitir su voto correctamente con sistemas de VER actualmente en uso [231].
- El marco legal no está suficientemente desarrollado. Lo mismo sucede con su translación a criterios técnicos. La presente tesis trata de hacer aportes en esta materia en los apartados 2.2 y 2.3 pero queda mucha labor pendiente desde el punto de vista legislativo.
- La cuestión de la E2Ev y la RC como requisitos imprescindibles del VER. Actualmente existen esquemas implementando simultáneamente ambas (Civitas, BeleniosRF). Conviene, no obstante, prestar especial atención a las definiciones alternativas que en ellos se proponen de los dos conceptos. En numerosas ocasiones se "*estira"* la definición, partiendo de premisas de seguridad excesivamente optimistas.
- La necesidad de conseguir un sistema de VER inviolable aunque el equipo del votante esté infectado con *malware*.

En cuanto a la tesis en sí, las principales mejoras y líneas futuras van en una doble línea:

1. Actualización contínua de la metodología en base a:
   - La inclusión de los avances criptográficos, tecnológicos y legales que se produzcan, como por ejemplo las novedades en computación post-cuántica [468, 469].
   - El estudio de las nuevas experiencias del VER en países destacados como Suiza, Australia, Canadá, Estonia, España o Finlandia
   - La revisión por parte de expertos multidisciplinares en el campo
2. La elaboración, (posiblemente en colaboración con un partner tecnológico u otras instituciones) de un software de evaluación de sistemas de VER basado en la metodología desarrollada en la presente disertación

BIBLIOGRAFIA

[1] B. Adida. “Helios: Web-based open-audit voting.” *En Proceedings of the 17th USENIX Security Symposium (Security08)*, San Jose, CA, 2008.

[2] B.B. Bederson, B. Lee, R.M. Sherman, P.S. Herrnson, and R.G. Niemi. “Electronic voting system usability issues.” En CHI ’03: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 145–152, New York, NY, USA, 2003.

[3] J. Benaloh. “Simple verifiable elections”. En *Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop*, pages 5–5. USENIX Association Berkeley, CA, USA, 2006.

[4] L. Panizo. “Desarrollo de una metodología para el análisis y la clasificación de los sistemas de voto electrónico”, *Departamento de Ingeniería Eléctrica y de Sistemas y Automática, Universidad de León*, 2014.

[5] “A Global Survey on the Cost of Registration and elections”, *United Nations Development Program, Bureau for Development Policy*, New York USA, 2005.

[6] “Freedom in the World 2015”, *Freedom House*, Washington DC, 2015.

[7] M. Bellis. “The History of Voting Machines” *http://www.inventors.about.com*, 2000.

[8] R. G. Saltman. “Effective Use of Computing Technology in Vote-Tallying”. *National Bureau of Standards (NBS) Special Publication* 500-30, 1975.

[9] J. Wand, K. Shotts, J. Sekhon, W. Mebane Jr, M. Herron, H. Brady. “The Butterfly Did It, The Aberrant Vote for Buchanan in Palm Beach County, Florida”, *American Political Science Review*, *Vol.95, No. 4*, 2001.

[10] G. Adams, “Voting Irregularities in Palm Beach, Florida” *http://www.amstat.org/misc/VotingIrregularitiesArticle.pdf* , 2001.

[11] V. M. Morales Rocha. “Seguridad en los Procesos de Voto Electrónico Remoto: Registro, Votación, Consolidación de Resultados y Auditoría.” *Departamento de Telemática, Universidad Politécnica de Cataluña*, 2009.

[12] Verified Voting Foundation. *https://www.verifiedvoting.org/resources/internet-voting/email-fax/,* 2012.

[13] Federal Election Commission. Voting Systems Standards. 1990. *http://votingmachines.procon.org/sourcefiles/fec1990.pdf* , 1990.

[14] J. Monteiro, P. Swatman, L. Tavares. “Towards the Knowledge Society”. *Springer Science Business Media*, 2013.

[15] R. Rojas. “Encyclopedia of Computers and Computer History”. *Routledge. ISBN: 978-1579582357*, 2001.

[16] H. Jonker, J. Pang. “Bulletin Boards in Voting Systems: Modelling and Measuring Privacy”. *Computer Science & Communication Department. University of Luxembourg. Availability, Reliability and Security (ARES) 2011 Sixth International Conference. ISBN: 978-1-4577-0979-1*, 2011.

[17] “Internet Security Threat Report 2014. Volume 19” *Symantec Corporation. 350 Ellis Street, Mountain View, CA 94043 USA*. *http://www.symantec.com/content/en/us/enterprise/other_resources/b-istr_main_report_v19_21291018.en-us.pdf* , 2014.

[18] R. Küsters, T. Truderung, A. Vogt. “Clash Attacks on the Verifiability of E-Voting Systems”. *University of Trier, Germany,* 2012.

[19] H.A. von Spakovsky. “The Dangers of Internet Voting”. *The Heritage Foundation. 214 Massachussetts Avenue, NE Washington, DC 20002*. 2015

[20] A. Fiat, A. Shamir. “How to prove yourself: Practical solutions to identification and signature problems.” *CRYPTO'86, 6th Annual International Cryptology Conference on Advances in Cryptology*, *pp. 186-194. Santa Barbara, USA*. 1986.

[21] A. Fujioka, T. Okamoto, K. Ohta. “A practical secret voting scheme for large scale elections” *ASIACRYPT'92, Workshop on the Theory and Application of Cryptographic Techniques*, *pp. 244-251. LNCS 718, Gold Coast, Australia*. 1992.

[22] S. Mudana. “Security Flaws in Internet Voting System” *Computer Science Department. University of Auckland. New Zealand,* 2005

[23] “The Future of Voting” *U.S. Vote Foundation, 4325 Old Glebe Road, Arlington, USA*, 2015.

[24] R. Sekibuule. “Security Analysis of Remote E-Voting”. *Advances in Systems Modelling and ICT Application*, 2007.

[25] W. Diffie, M. Hellman. “New Directions in Cryptography”. *IEEE Transactions on Information Theory. 22 (6), pp. 644-654,* 1976.

[26] FREAK Attack: *https://freakattack.com/* , 2015

[27] B. Beurdouche et. al. “A Messy State of the Union: Taming the Composite State Machines of TLS”. 2015.

[28] K. Mitnick. “CSEPS Course Workbook”. *Mitnick Security Publishing*, pp.4, 2004.

[29] B. Pfitzmann, A. Pfitzmann. “How to break the direct RSA-Implementation of mixes”. *Institut für Rechnerentwurf und Fehlertoleranz, Universität Karlsruhe*. 1990.

[30] S. Khazaei, B. Terelius, D. Wikström. “Cryptanalysis of a Universally Verifiable Efficient Re-Encryption Mixnet”. *KTH Royal Institute of Technology, Sweden*. 2012.

[31] D. Chaum. “Untraceable electronic mail, return addresses and digital pseudonyms”. *Commum. ACM, 24(2), pp. 84-88,* 1981.

[32] K. Sako, J. Kilian. “Receipt-free mix-type voting scheme- a practical solution to the implementation of a voting booth” *EUROCRYPT'95*, *pp. 393-403*, 1995.

[33] D. Adrian, K. Bhargavan, Z. Durumeric, P. Gaudry, M. Green, J.A. Halderman et al. “Imperfect Forward Secrecy: How Diffie-Hellman Fails in Practice”. *INRIA Paris Rocquencourt, INRIA Nancy-Grand Est, CNRS and Université de Lorraine, Microsoft Research, University of Pennsylvania, Johns Hopkins, University of Michigan*, 2015.

[34] M. Green. “A Few Thoughts on Cryptographic Engineering. Attack of the week: FREAK (of factoring the NSA for fun and profit)”. *http://blog.cryptographyengineering.com/2015/03/attack-of-week-freak-or-factoring-nsa.html* , 2015.

[35] M. Green. “A Few Thoughts on Cryptographic Engineering. Attack of the week: Logjam” *https://blog.cryptographyengineering.com/2015/05/22/attack-of-week-logjam/* . 2015

[36] S. Wolchok, E. Wustrow, D. Isabel, J.A. Halderman. “Attacking the Washington, D.C. Internet Voting System”. The University of Michigan, Ann Arbor. *16th Conference on Financial Cryptography & Data Security*, 2012.

[37] A. Aiken, D.J. Farber, D. Dill, E. Felten, A. Rubin, B. Simons, et al. “Computer Technologists' Statement on Internet Voting”. *https://www.verifiedvoting.org/projects/internet-voting-statement/* , 2008.

[38] W. Geiselmann, H. Kopfer, R. Steinwandt, E. Tromer. “Improved routing-based linear algebra for the number field sieve.*” Information Technology: Coding and Computing*, 2005.

[39] W. Geiselmann and R. Steinwandt. “No wafer-scale sieving hardware for the NFS: Another attempt to cope with 1024-bit”. In *Eurocrypt*, 2007.

[40] Microsoft Security Bulletin MS15-055. “Vulnerability in Schannel could allow information disclosure”. 2015.

[41] Spiegel Staff. “Prying eyes. Inside the NSA’s war on Internet security.”*Der Spiegel*. http://www.spiegel.de/international/germany/inside-the-nsa-s-war-on-internet-security-a-1010361.html , 2014

[42] T. Kleinjung, K. Aoki, J. Franke, A.K. Lenstra, E. Thomé, J.W. Bos, P. Gaudry, A. Kruppa, P.L. Montgomery, D.A. Osvik, H. te Riele, A. Timofeev, P. Zimmermann. “Factorization of 768-bit RSA modulus” In *CRYPTO*, 2010.

[43] T. Kleinjung. “Cofactorisation strategies for the number field sieve and an estimate for the sieving step for factoring 1024 bit integers,” http://www.hyperelliptic.org/tanja/SHARCS/talks06/thorsten.pdf 2006.

[44] The Washington Post staff. “The Black Budget” *The Washington Post* http://www.washingtonpost.com/wp-srv/special/national/black-budget/ , 2013.

[45] NSA Suite B Cryptography, Suite B Implementers' Guide to NIST SP 800-56A, 2009.

[46] Certicom Research, Standards for efficient cryptography, SEC 1: Elliptic Curve Cryptography, 2009.

[47] J.A. Halderman, V. Teague. “The New South Wales iVote System: Security Failures and Verification Flaws in a Live Online Election”. *University of Michigan and University of Melbourne*. 2015

[48] V. Teague, J.A. Halderman. “Security flaws in New South Wales puts thousands of on-line votes at risk”. *Freedom to Tinker blog*, https://freedom-to-tinker.com/blog/teaguehalderman/ivote-vulnerability/ 2015.

[49] B. Adida, O. de Marneffe, O. Pereira, J.-J. Quisquater. “Electing a University President using Open-Audit Voting: Analysis of real-world use of Helios”. 2009.

[50] A. Kiayias, T. Zacharias, B. Zhang. “End-to-end verifiable elections in the standard model”. In *Elisabeth Oswald and Marc Fischlin, editors, EUROCRYPT 2015, part II, volumen 9057 de LNCS*, pp. 468-498. Springer, 2015.

[51] S. Popoveniuc, J. Kelsey, A. Regenscheid, P. Voral. “Performance requirements for end-to-end verifiable elections” en *EVT/WOTE,* 2010.

[52] S. Kremer, M. Ryan. “Analysis of an Electronic Voting Protocol in the Applied Pi Calculus”. ESOP ’05: vol. 6345 de *LNCS*, pp. 186-200. Springer, 2005.

[53] S. Delaune, S. Kremer, M. Ryan. “Coercion-Resistance and Receipt- Freeness in Electronic Voting”. *CSFW’06: 19th Computer Security Foundations Workshop*, pp. 28-42, 2006.

[54] Council of Europe. Committee of Ministers. “Legal, Operational and Technical Standards for e-voting, rec (2004)”. *Bruselas: Council of Europe, Committee of Ministers*, 2004.

[55] Directorate of Democratic Institutions. “E-Voting Handbook” *Strasbourg: Council of Europe Publising. ISBN 978-92-871-6948-8*, 2010.

[56] E. Pérez Belleboni. “Aplicación de documentos de identificación electrónica a un esquema de voto telemático a escala paneuropea, seguro, auditable y verificable”. *Departamento de Ingeniería y Arquitecturas Telemáticas, Universidad Politécnica de Madrid*, 2013.

[57] J. Puiggalí, V. Morales-Rocha. “Remote Voting Schemes: A comparative Analysis”. *Scytl Secure Electronic Voting. En VOTE-ID 2007, Bochum, Germany,* 2007.

[58] http://www.scytl.com/en/products/election-day/scytl-phone-voting/

[59] Council of Europe. “Electronic Democracy. (e-democracy)” *Council of Europe Publishing. F-67075 Strasbourg Cedex*. *ISBN: 978-92-871-6647-0*, 2009.

[60] S. Coleman, D. Norris. “A new agenda for e-democracy”. *Oxford Internet Institute*, 2005.

[61] “E-Democracy 2015 Citizen Rights in the world of the new computing paradigms”. *6th International Conference on e-democracy*, http://www.edemocracy2015.eu

[62] IEEE Voting Systems Standard Committee (VSSC). *IEEE VSSC/1622: Common Data Format for Election Equipment.* http://grouper.ieee.org/groups/1622/

[63] A. Juels, D. Catalano, M. Jakobsson. “Coercion-resistant electronic-elections”. En *Proceedings of the 2005 ACM workshop on Privacy in the electronic society, pp. 61-70*, 2005.

[64] T. ElGamal. “A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms”. In *Advances in Cryptology: Proceedings of CRYPTO 84, volumen 196 of Lecture Notes in Computer Science,* pp. 10-18, Santa Barbara, California, USA, Springer-Verlag, 1985.

[65] “World Factbook: Suffrage”. *Central Intelligence Agency, CIA*. https://www.cia.gov/library/publications/the-world-factbook/fields/2123.html

[66] La Constitución Española de 1978.http://www.congreso.es/consti/constitucion/indice/index.htm

[67] M. Clarkson, S. Chong, A. Myers. “Civitas: Toward a Secure Voting System”. *Department of Computer Science, Cornell University*, 2007.

[68] H. Li, A.R. Kankanala, X. Zou. “Taxonomy and Comparison of Remote Voting Schemes”. *Purdue University*, 2014.

[69] T. Okamoto. “Receipt-Free Electronic Voting Schemes for Large Scale Elections”. *En SP’97: 5th International Workshop on Security Protocols*, *LNCS, vol 1361 pp. 25-35, Springer, 1998*.

[70] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant electronic-elections”. *Cryptology ePrint Archive, Report 2002/165,* 2002.

[71] M. Backes, C. Hritcu, M. Maffei. “Automated Verification of Remote Electronic Voting Protocols in the Applied Pi-Calculus”. *En CSF’08: 21st Computer Security Foundations Symposium. IEEE Computer Society, 2008*, *pp. 195-209,* 2008.

[72] V. Cortier, G. Fuchsbauer, D. Galindo. “Belenios RF: A Strongly Receipt-Free Electronic Voting Scheme”. *CNRS/LORIA, France, IST Austria, SCYTL Secure Electronic Voting*, *Spain*. 2015.

[73] M.R. Clarkson, S. Chong, A.C. Myers. “Civitas: Toward a secure voting system”. *En 2008 IEEE Symposium on Security and Privacy*, *pp. 354-368, IEEE Comp. Society Press,* 2008.

[74] K. Gjøsteen. “Analysis of an internet voting protocol”, 2010.

[75] D. Bernhard, V. Cortier, O. Pereira, B. Smyth, B. Warinschi. “Adapting Helios for probable ballot privacy”. *University of Bristol, LORIA – CNRS, Univ. Catholique de Louvain*, 2011.

[76] Ministerio de Interior. http://elecciones.mir.es/resultadoslocales2015/ , 2015

[77] D. Chaum. “Secret-Ballot Receipts: True Voter-Verifiable Elections”. *IEEE Security and Privacy, vol 2, no. 1, pp. 38-47*, 2004.

[78] D. Chaum. “Blind signatures for untraceable payments”. *Advances in Cryptology – Crypto’82, Springer-Verlag, pp.199 – 203*, 1982.

[79] D. Chaum. “Security without Identification: Transaction System to make Big Brother Obsolete”. *Communications of the ACM, v. 28, no. 10, pp. 1030 – 1044*, 1985.

[80] I. Damgård, M. Jurik. “A Generalisation, and a Simplification and some Applications of Paillier’s Probabilistic Public-Key System”. *PKC’01, pp. 119 – 136*, 2001.

[81] M. Jacobsson, A. Juels, R. Rivest. “Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking”. *Proceedings of the 11th USENIX Security Symposium, pp. 339 – 353*, 2002.

[82] P. Paillier. “Public-Key Cryptosystems based on Composite-Degree Residuosity Classes”. *J. Stern Editor, EUROCRYPT’99, pp. 223 – 238, Springer – Verlag, LNCS 1592*, 1999.

[83] K. Peng, C. Boyd, E. Dawson, K. Viswanathan. “A correct, private and efficient mix network”. *Proceedings of PKC’04, LNCS 2947, Pringer – Verlag, pp. 439 – 454*, 2000.

[84] T. Pedersen. “A Threshold Cryptosystem without a Trusted Party”. *Advances in Cryptology – EUROCRYPT’91. D. Davies editor. Springer – Verlag LNCS series*, 1991.

[85] A. Shamir. “How to share a secret”. *Communications of the ACM 22, 11*. *pp. 612 – 613*, 1979.

[86] K. Sako, J. Killian. “Secure Voting Using Partially Compatible Homomorphisms”. *Advances in Cryptology – CRYPTO’94, LNCS 839, pp. 411 – 424*, 1994.

[87] X. Wang, H. Yu. “How to Break MD5 and Other Hash Functions”. *Advances in Cryptology – EUROCRYPT’05, vol. 3494 LNCS, Springer, pp. 1 – 18*, 2005.

[88] Z. Xia, S. Schneider. “A new recepit-free e-voting scheme based on blind signature (abstract)”. *Proc. of Workshop on Trustworthy Elections (WOTE 2006), pp. 19–35*, 2005.

[89] M. Bellare, P. Rogaway. “Random Oracles are Practical: A Paradigm for designing Efficient Protocols”. *En 1st ACM Conference on Computer and Communications Security, pp. 62 – 73*, 1993.

[90] S. Goldwasser, Y. T. Kalai. “On the (in)security of the Fiat-Shamir paradigm”. *En 44th Annual Symposium on Found. of Computer Science, pp. 102 – 115*, *IEEE Computer Society Press*, 2003.

[91] N. Bitansky, D. Dachman-Soled, S. Garg, A. Jain, Y. T. Kalai, A. López-Alt, D. Wichs. “Why Fiat-Shamir for proofs lacks a proof” *En TCC*, 2013.

[92] D. Dachman-Soled, A. Jain, Y. T. Kalai, A. López-Alt. “On the (in)security of the Fiat-Shamir paradigm, revisited”. *Cryptology ePrint Archive, Report 2012/706*, 2012.

[93] J. Benaloh, R. Rivest, P. Ryan, P. Stark, V. Teague, P. Vora. “End to End Verifiability”, 2014.

[94] U. Feige, A. Fiat, A. Shamir. “Zero Knowledge Proofs of Identity”. *En Alfred V. Aho, editor, STOC, pp. 210 – 217*, ACM 1987.

[95] S. Goldwasser, S. Micali, C. Rackoff. “The knowledge complexity of interactive proof systems (extended abstract)”. *En Robert Sedgewick, editor, STOC, pp. 291 – 304, ACM*. 1985.

[96] M. Blum, P. Feldman, S. Micali. “Non-interactive zero-knowledge and its applications”. *En STOC 88, pp. 103–112*, 1988.

[97] J.-S. Coron, J. Patarin, Y. Seurin. “The Random Oracle Model and the Ideal Cipher Model are Equivalent”. *U. of Luxembourg, U. of Versailles, Orange Labs*, 2008.

[98] G. Leurent, P. Nguyen. “How Risky is the Random-Oracle Model?”. *Advances in Cryptology-CRYPTO’09*, 2009.

[99] I. Damgård. “A Design Principle for Hash Functions”. *En CRYPTO, volume 435 of Lecture Notes in Computer Science, pp. 416–427*, Springer, 1989.

[100] S. Matyas, C. Meyer, and J. Oseas. “Generating strong one-way functions with cryptographic algorithms.” *In IBM Technical Disclosure Bulletin 27(10a), pp. 5658–5659,* 1985.

[101] B. Preneel, R. Govaerts, J. Vandewalle. “Hash Functions Based on Block Ciphers: A Synthetic Approach*”. En CRYPTO vol. 773 de LNCS, pp. 368–378, Springer,* 1993.

[102] R.C. Merkle. “A Certified Digital Signature”. *En Advances in Cryptology - CRYPTO '89 Proceedings, vol. 435 de LNCS, G. Brassard, ed, Springer-Verlag, pp. 218-238*, 1989.

[103] J.-S. Coron, Y. Dodis, C. Malinaud, P. Puniya. “Merkle – Damgård Revisited: how to Construct a Hash Function”. *Univ of Luxembourg, New - York University, Gemplus Card International*, 2006.

[104] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant Electronic-Elections”. *RSA Laboratories, CNRS-Ecole Normale Supérieure, Indiana Univ., School of Informatics*, 2010.

[105] D. Achenbach, C, Kempka, B. Löwe, J. Müller-Quade. “Improved Coercion-Resistant Electronic Elections through Deniable Re-Voting”. *En JETS, The Usenix Journal of Election Tech. and Systems,* 2015.

[106] J. Benaloh. “Rethinking Voting Coercion: The Realities Imposed by Technology”. *Microsoft Research the Usenix Journal of Election Technology and Systems*, Aug. 2013.

[107] J. Benaloh, D. Tuinstra. “Receipt-free secret ballot elections”. *En 26th ACM STOC, pp. 544 – 553*, 1994.

[108] M. Hirt, K. Sako. “Efficient Receipt-Free voting based on homomorphic encryption*” In B. Preneel, editor, EUROCRYPT’00, vol. 1807 de LNCS, pp. 539 – 556*, 2000.

[109] D. Dolev, A.C. Yao, A. C. (1983), "On the security of public key protocols". *IEEE Trans. on Information Theory, IT-29, pp. 198–208, 1983*

[110] M. Jakobsson, A. Juels, “Mix and match: Secure function evaluation via ciphertexts.*” En Advances in Cryptology (Asiacrypt’00), LNCS 1976, pp. 162-177*, 2000.

[111] R. Pass. “On Deniability in the Common Reference String and Random Oracle Model”. *Royal Institute of Technology, Stockholm*, 2003.

[112] M. Green. “What is the Random Oracle Model and why should you care, parts I, II, III and IV”. *http://blog.cryptographyengineering.com/2011/09/what-is-random-oracle-model-and-why.html* , 2011.

[113] RSA Laboratories. “PKCS #1 v2.1: RSA Cryptography Standard”, 2002.

[114] R. Canetti, R. Pass, A. Shelat. “Cryptography from sunspots: How to use an Imperfect Reference String”. *IBM, Cornell, U. Virginia,* 2006.

[115] R. Canetti, O. Goldreich, S. Halevi. “The Random Oracle Metodology, Revisited”. *En 30th STOC, pp. 209-218,* 1998.

[116] A. Herzberg, M. Jakobsson, S. Jarecki, H. Kraczyk, M. Yung. “Proactive Public Keys and Signature Systems”. *Conf. on Computer and Communications Security 1997, pp. 100-110*, 1997.

[117] M. Blumenthal. “Encryption: Strengths and Weaknesses of Public-Key Cryptography”. *Department of Computing Sciences. Villanova University,* 2000.

[118] S. Vadhal, A. Rosen. “Public-Key Encryption in Practice” *Harvard University*, 2006

[119] D. Pointcheval. “Practical Security in Public-Ket Cryptography”. LIENS – CNRS, École Normale Supérieure. *In Proceedings of the 4th Conference on Information Security and Cryptology ’01. LNCS 2288, pp. 1 – 17, Springer – Verlag*, 2002.

[120] CGI. “White Paper: Public Key Encryption and Digital Signature: How do they work”. *CGI Group*, 2004.

[121] W. Diffie. “The first ten years of public-key cryptography*”. En Proceedings of the IEEE, vol 76, Nº 5*, 1988.

[122] M. Hellman. “An overview of Public-Key Cryptography”. *IEEE Comm. Society Magazine*, 1978.

[123] S. Halevi, H. Krawczyk. “Public-Key cryptography and password protocols”. *ACM Transactions on Information and System Security, pp. 230 – 268*, 1999.

[124] S. Goldwasser, S. Micali, R. Rivest "A digital signature scheme secure against adaptive chosen-message attacks". *SIAM Journal on Computing, 17(2), pp. 281–308*, 1988.

[125] J. Benaloh, “Verifiable Secret-ballot Elections”, *PhD Thesis, Yale University*, 1987.

[126] S. Iftene. “General Secret Sharing based on the Chinese Remainder Theorem with Applications in E-voting”. *Electronic Notes in Theoretical Computer Science vol. 186, pp. 67–84*, 2007.

[127] D. G. Nair, V. P. Vinu, G. S. Kumar. “An improved E-voting scheme using secret- sharing based secure multi-party computation”. *Cochin University of Science and Technology*, ICCN-2014.

[128] B. Schoenmakers. “A simple publicly-verifiable secret-sharing scheme and its applications to electronic voting*”. In Advances in Cryptology—CRYPTO ’99, Vol. 1666 of LNCS, Springer-Verlag, pp. 148-164. Eindhoven University of Technology*, 1999.

[129] G. R. Blakley, (1979). "Safeguarding cryptographic keys". *Proceedings of the National Computer Conference 48, pp. 313–317,* 1979.

[130] L. Ronquillo. “Securing e-voting systems”. *DemTech, Democratic Technology,* 2015.

[131] B. Lee, K. Kim. “Receipt-free electronic voting through collaboration of voter and honest verifier”. *En Proc. of JW-ISC2000, pp. 101–108*, 2000.

[132] R. Cramer, R. Gennaro, B. Schoenmakers. “A secure and optimally efficient multi-authority election scheme”. *En Proc. of the 16th anual Int. Conf. on Theory and App. Of cryptographic techniques. EUROCRYPT’97, pp. 103–118,* 1997.

[133] B. Hemenway, R. Ostrovsky. “On homomorphic Encryption and Chosen-Ciphertext Security”. *University of Michigan and University of UCLA. En Proceedings of PKC,* 2012.

[134] C. Gentry. “Fully Homomorphic Encryption Using Ideal Lattices”. *Stanford University and IBM Watson,* 2009.

[135] C. Gentry, S. Halevi. “Implementing Gentry’s Fully-Homomorphic Encryption Scheme”. *IBM Research, 2011.*

[136] M. K. Ibrahem, N.M. Kiatan. “Homomorphic Encryption Protocol for Secure Electronic Voting System”. *Al Nahrain University*, 2011.

[137] H. Lipmaa, “Secure Electronic Voting Protocols”. *Cybernetica AS and University of Tartu,* 2005.

[138] L. Morris. “Analysis of Partially and Fully Homomorphic Encryption”. *Rochester Institute of Technology*, 2013.

[139] S. Bailey, J’D. Bush, A. Conner, L. Michel. “Homomorphic Encryption”. *Venderbilt University*, 2012.

[140] S. Goldwasser, S. Micali *(1982*). "Probabilistic encryption and how to play mental poker keeping secret all partial information". *Proc. 14th Symposium on Theory of Computing: 365–377,* 1982.

[141] J. Benaloh. “Dense Probabilistic Encryption”. *Clarkson University,* 1994.

[142] M. van Dijk, C. Gentry, S. Halevi, V. Vinod. “Fully Homomorphic Encryption over the Integers”. *EUROCRYPT’10, Springer*, 2010.

[143] C. Gentry, S. Halevi, N.P. Smart. “Homomorphic Evaluation of the AES Circuit”. *En CRYPTO’12. Springer, 2012.*

[144] L. Ducas, D. Micciancio. “FHE Bootstrapping in less tan a second”. *Cryptology eprint archive,* 2015.

[145] P. Kocher. “Timing Attacks on Implementations of Diffie-Hellman, RSA, DDS, and Other Systems”. *Cryptography Research Inc,* 1996.

[146] R. Gennaro, D. Micciancio, T. Rabin. “An Efficient Non-Interactive Statistical Zero-Knowledge Proof System for Quasi-Safe Prime Products”. *M.I.T. Laboratory for Computer Science, I.B.M. T.J. Watson Research Center*, 1998.

[147] S. Canard, D. Pointcheval, O. Sanders. “Efficient Delegation of Zero-Knowledge Proofs of Knowledge in a Pairing-Friendly Setting”. *Orange Labs, Applied Crypto Group, École normale supérieure, CNRS & INRIA. In the Proceedings of the 17th International Cnference on Practice and Theory in Public-Key Cryptography (PKC’14). Hugo Krawczyk Ed., Springer-Verlag, LNCS 8383, pp. 167–184*, 2014.

[148] M. Green. “Zero-Knowledge Proofs: An illustrated premier”. http://blog.cryptographyengineering.com/2014/11/zero-knowledge-proofs-illustrated-primer.html , 2014.

[149] J. Groth. “Short Non-Interactive Zero-Knowledge Proofs”. *Univ. College London*. 2010.

[150] K. Haralambiev. “Efficient Cryptographic Primitives for Non-Interactive Zero-Knowledge Proofs and Applications”. *New York University*. 2011.

[151] J. Groth, R. Ostrovsky, A. Sahai. “New Techniques for Non-Interactive Zero-Knowledge”, 2011.

[152] I. Damgård, M. Jurik, J.B. Nielsen. “A Generalization of Paillier’s Public-Key System with Applications to Electronic Voting”. *Aarus University*, 2002.

[153] J-J. Quisquater L. Guillou, T. Berson. "How to Explain Zero-Knowledge Protocols to Your Children". *Advances in Cryptology – CRYPTO '89: Proceedings 435, 628–631,* 1990.

[154] U. Feige, D. Lapidot, A. Shamir. “Multiple Non-Interactive Zero Knowledge Proofs Under General Assumptions”. *SIAM J. Comput. 29(1), pp. 1–28*, 1999.

[155] D. Bernhard, O. Pereira, B. Warinschi. “How not to prove yourself: Pitfalls of the Fiat-Shamir Heuristic and Applications to Helios”. *University of Bristol, Université Catolique de Louvain*, 2012.

[156] D. Dolev, C. Dwork, M. Naor. ”Non–malleable Cryptography”. *Proceedings of the 23rd annual ACM symposium on Theory of computing, pp. 542–552*, 1991.

[157] D. Wikström. “Mixnets for Voting”. *School of Computer Science and Communication. KTH Royal Institute of Technology. Sweden*, 2010.

[158] P. Bibiloni, A. Escala, P. Morillo. “Vote Validability in Mix-Net-Based eVoting”. *University of Ballearic Islands, University Politécnica at Barcelona, Spain and Scytl*, 2015.

[159] J. Esch. “Prolog to a survey on mix networks and their secure applications”. *Proceedings of the IEEE, 94 (12), pp. 2139–2141*, 2006.

[160] K. Sampigethaya, R. Poovedran. “A survey on mix networks and their secure applications”. *Proceedings of the IEEE, 94 (12), pp. 2142–2181,* 2006.

[161] C. Park, K. Itoh, K. Kurosawa. “Efficient anonymous channel and all / nothing election scheme”. *In Eurocrypt, pp. 248–259*, 1993.

[162] J. Furukawa, K. Sako. “An efficient scheme for proving a shuffle”. *In J. Kilian, editor, CRYPTO, volumen 2139 of LNCS, pp. 368–387, Springer,* 2001.

[163] C.A. Neff. “A verifiable secret shuffle and its application to e-voting.” *In CCS’01: Proc. of the 8th ACM Conference on Computer and Communications Security, pp. 116–125, New York*, 2001.

[164] M. Jakobsson, A. Juels, R. L. Rivest. “Making Mix-nets robust for electronic voting by randomized partial checking.” *En D. Boneh, editor, USENIX Security Symposium pp. 339–353*, 2002.

[165] B. Pfitzmann. “Breaking Efficient Anonymous Channel”. *En Eurocrypt, pp. 332–340,* 1994.

[166] Y. Desmedt, K. Kurosawa. “How to break a practical mix and design a new one”. *En B. Preneel, editor, EUROCRYPT, volumen 1807 of LNCS, pp. 557–572. Springer,* 2000.

[167] M. Mitomo, K. Korosawa. “Attack for flash mix”. *In T. Okamoto, editor, ASIACRYPT, volumen 1976 of LNCS, pp. 192–204, Springer*, 2000.

[168] D. Wikström. “Five practical attacks for *optimistic mixing for exit polls*”. *En M. Matsui and R. J. Zuccherato, editors, Selected Areas in Cryptography, vol. 3006 of LNCS, pp. 160–175.* 2004.

[169] S. Khazaei, D. Wikström. “Randomized partial checking re-visited” *Cryptology ePrint Archive. Report 2012/063,* http://eprint.iacr.org/ , 2012.

[170] J. Puiggalí Allepuz, S. Guasch Castelló. “Universally Verifiable efficient re-encryption mixnet”. *In R. Krimmer and R. Grimm, editors, Electronic Voting, vol. 167 of LNI, pp. 241–254. GI,* 2010.

[171] M. Ohkubo, F. Miura, M. Abe. “An improvement on a practical secret sharing voting scheme”. *En Proc. of the 2nd International Workshop on Information Security, ISW’99, pp. 225-234*, 1999.

[172] D. Zissis, D. Lekkas.” Design, Development and Use of Secure Electronic Voting Systems”. *IGI Global. ISBN 978-1-4666-5823-3,* 2014.

[173] RSA Laboratories. “What is a blind signature scheme?” *EMC Corporation (RSA Laboratories es una división de EMC).* http://japan.emc.com/emc-plus/rsa-labs/standards-initiatives/what-is-a-blind-signature-scheme.htm , 2015.

[174] I. K. Salah, A. Darwish, S. Oqeili. “Mathematical attacks on RSA Cryptosystem”. *University of Jordan, Royal Scientific Society, Jordan*, 2006.

[175] M. Witteman, J. van Woudenberg, F. Menarini. “Defeating RSA multiply-always and message blinding countermeasures”. *Riscure BV. The Netherlands*, 2007.

[176] D. Schliebner. “Electronic Remote Voting”. *Humboldt - University of Berlin*, 2011.

[177] W. Diffie, P.C. van Oorschot, M.J Wiener. "Authentication and Authenticated Key Exchanges", *Designs, Codes and Cryptography (Kluwer Academic Publishers) (2), pp. 107–125,* 1992.

[178] R. Rivest, A. Shamir, L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems" *Communications of the ACM 21 (2), pp. 120–126,* 1978.

[179] Kleinjung, Aoki, Franke, Lenstra, Thomé, Bos, Gaudry, Kruppa, Montgomery, Osvik te Riele, Zimmermann. "Factorization of a 768-bit RSA modulus" *International Association for Cryptologic Research*, 2010.

[180] E. Landquist. “The Quadratic Sieve Factoring Algorithm”. *MATH 488: Cryptographic Algorithms,* 2001.

[181] C. Pomerance. “Smooth Numbers and the Quadratic Sieve”. *Algorithmic Number Theory. MSRI Publications. Volume 44,* 2008.

[182] C. Pittet. “Mathematical Aspects of Shor’s Algorithm”. *CNRS, France*, 2014.

[183] P. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. *SIAM J. Sci. Statist. Comput., 26*, 1997.

[184] F. Verkauteren. “Discrete Algorithms in Cryptography”. *ESAT/COSIC, K.U. Leuven, ECRYPT Summer School,* 2008.

[185] L. Maurits. “Public Key Cryptography using Discrete Logarithms in Finite Fields”. *University of Adelaide*, Australia.

[186] K. Kwangjo. “Public Key Cryptography”. *4th International Workshop on Practice and Theory in Public Key Cryptosystems, PKC 2001, Springer*. 2001.

[187] R. Oyono. “The Discrete Logarith Problem and its application in Cryptography”. *Lectures in Cryptography for Master Class. Universidad Autónoma de Madrid,* 2009.

[188] B. Blanchet, D. Pointcheval. “The Computational and Decisional Diffie-Hellman assumptions in CryptoVerif”. *CNRS École Normale Supérieure, INRIA, Paris*, 2010.

[189] M.A. Cerveró, V. Mateu, J.M. Miret, F. Sebé, J. Valera. “An Elliptic Curve Based Homomorphic Remote Voting System*”. University of Lleida, Spain. RECSI 2014, Alicante*, 2014.

[190] D. Hankerson, A. Menezes, S. Vanstone. “Guide to Elliptic Curve Cryptography”. *Springer, ISBN 0-387-95273-X*, 2004.

[191] V. Miller. "Use of elliptic curves in cryptography". *CRYPTO. Lecture Notes in Computer Science 85, pp. 417–426. doi: 10.1007/3-540-39799-X_31. ISBN 978-3-540-16463-0.* 1985.

[192] N. Koblitz. "Elliptic curve cryptosystems". *Mathematics of Computation 48 (177), pp. 203–209, doi:10.2307/2007884. JSTOR 2007884,* 1987.

[193] *"Fact Sheet NSA Suite B Cryptography". U.S. National Security Agency,* 2005.

[194] J.H. Silverman. “An Introduction to the Theory of Elliptic Curves”. *Brown University y NTRU Cryptosystems Inc.,* 2006.

[195] M. Ajtai. “Generating hard instances of lattice problems” *In Complexity of computations and proofs, volumen 13, pp. 1 – 32. Seconda University Napoli, Caserta*, 2004.

[196] P. Shor. “Algorithms for quantum computation: discrete logarithms and factoring.” *En Proc. of the 35th Annual Symposium on Foundations of Computer Science, pp. 124–134, IEEE,* 1994.

[197] S. Bai, A. Langlois, T. Lepoint, D. Stehlé, R. Steinfeld. “Improved Security Proofs in Lattice-based Cryptography: using the Rényi divergence rather tan the statistical distance”. *U. Lyon, CNRS, ENSL, INRIA, UCBL, EPFL, CryptoExperts, Monash University,* 2015.

[198] T. Güneysu, V. Lyubashevsky, T. Pöppelmann. “Practical Lattice-based Cryptography: A Signature Scheme for Embedded Systems”. *Ruhr-University Bochum, INRIA,* 2013.

[199] D. Micciancio, O. Regev. “Lattice-based Cryptography”. 2008.

[200] C. Peikert. “Lattice Cryptography for the internet”. *Georgia Institute of Technology*, 2014.

[201] J. H. van de Pol. “Lattice-based cryptography”. *Eindhoven Univ. of Technology*, 2011.

[202] J. Howe, T. Pöppelmann, M. O’Neill, E. O’Sullivan, T. Güneysu. “Practical Lattice-based Digital Signature Schemes”. *ACM Transactions on Embedded Computing Systems (TECS) - Special Issue on Embedded Platforms for Crypto and Regular Papers, vol. 14, issue 3*, 2015.

[203] L. Ducas, A. Durmus, T. Lepoint, V. Lyubashevsky. “Lattice Signatures and Bimodal Gaussians”. *CRYPTO 2013,* 2013.

[204] L. Ducas, V. Lyubashevsky, T. Prest. “Efficient Identity-Based Encryption over NTRU Lattices*.” En ASIACRYPT, pp. 22–41*, 2014.

[205] C. Shannon. “Communication Theory of Secrecy Systems”. *Bell System Technical Journal*, *vol. 28, pp. 656–715,* 1949.

[206] M. Dworkin. “Recommendation for Block Cipher Modes of Operation. Methods and Techniques”. *National Institute of Standards and Technology, NIST,* 2001.

[207] “Advanced Encryption Standard”. *Federal Information Processing Standards Public. 197*, 2001.

[208] J. Daemen, V. Rijmen. “The Block Cipher Rijndael”. *Smart Card Research and Applications. LNCS 1820, Springer-Verlag, pp. 288-296*, 2000.

[209] Rivest, R. L. (1994). "The RC5 Encryption Algorithm". *Proceedings of the Second International Workshop on Fast Software Encryption (FSE), pp. 86–96,* 1994.

[210] X. Lai, J. Massey, “A Proposal for a New Block Encryption Standard” *EUROCRYPT 1990*, *pp. 389–404,* 1990.

[211] B. Schneier. "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)". *Cambridge Security Workshop Proceedings (*Springer-Verlag*), 191–204,* 1993.

[212] A. Popov. "Prohibiting RC4 Cipher Suites". RFC 7465. *https://tools.ietf.org/html/rfc7465* , 2015

[213] R. Rivest, Ron; J. Schuldt, *"Spritz – a spongy RC4-like stream cipher and hash function"* MIT, 2014.

[214] M.J.B. Robshaw. "Stream Ciphers". *RSA Laboratories Technical Report TR-701, Ver 2.0,* 1995.

[215] J. Daemen, C. Clapp "Fast Hashing and Stream Encryption with PANAMA" *Fast Software Encryption (FSE) Conference,* 1998.

[216] C. Berbain, O. Billet, A. Canteaut, N. Courtois, H. Gilbert, L. Goubin, A. Gouget, L. Granboulan, C. Lauradoux, M. Minier, T. Pornin, H. Sibert. "SOSEMANUK, a fast software-oriented stream-cipher". *France Télécom, INRIA, Axalto, PRiSM, École Normale Supérieur, Cryptolog*, 2005.

[217] C. Adams, S. Lloyd. "Understanding PKI. Concepts, Standards and Deployment Considerations". *Addison – Wesley, Pearson Education Inc. ISBN 0 - 672 – 32391 – 5*, 2003.

[218] L. Langer, A. Schmidt, J. Buchmann. "Secure Online Elections in Practice". *Technische Universität Darmstadt,* 2008.

[219] V. Karatsiolis, L. Langer, A. Schmidt, E. Tews, A. Wiesmaier. "Cryptographic Application Scenarios". *Technische Universität Darmstadt,* 2010.

[220] D. Bruschi, G. Poletti, E. Rosti. "E-Vote and PKI's: A need, a bliss or a curse?". *Università degli Studi di Milano*, 2003.

[221] K. Peng, F. Bao. "Efficient multiplicative homomorphic e-voting". *Information Security, LNCS, vol. 6531, Springer. DOI 10.1007/978-3-642-18178-8_32, pp. 381-393*, 2011.

[222] D. Wikström. "A Sender Verifiable Mix-net and a New Proof of a Shuffle". *School of Computer Science and Communication. KTH Royal Institute of Technology. Sweden*, 2005.

[223] H. Lipmaa, B. Zhang. "A More Efficient Computationally Sound Non-Interactive Zero-Knowledge Shuffle Argument". *IACR Cryptology ePrint Archive 394,* 2011.

[224] S. Bayer, J. Groth. "Efficient Zero-Knowledge Argument for Correctness of a Shuffle." *En Eurocrypt 2012, pp. 263–280*, 2012.

[225] M. Chase, M. Kohlweiss, A Lysyanskaya, S. Meiklejohn. "Malleable Proof Systems and Its Applications*". En Eurocrypt 2012. Vol. LNCS 7237, Springer, pp. 281–300,* 2012.

[226] D. Bernhard, S. Neumann, M. Volkamer. "Towards a Practical Cryptographic Voting Scheme based on Malleable Proofs". *University of Bristol, CASED / TU Darmstadt. Vote-ID 2013,* 2013.

[227] S. Tribune. "University of Minnesota student who offered his vote on eBay gets community service." *Star Tribune. http://www.startribune.com/politics/state/26063069.html,* 2008.

[228] Local 6. "Man accused of trying to sell vote", 2004.

[229] M.C. Carlos, J.E. Martina, G. Price, R.F. Custódio. "An Updated Threat Model for Security Ceremonies". *28th ACM Symposium on Applied Computing*, 2013.

[230] T. Martimiano, E. dos Santos, M. Olembo, J.E. Martina, R.A. Reinaldo de Moraes. "Ceremony Analisys Meets Verifiable Voting: Individual Verifiability in Helios". *En 9th International conference on Emerging Security Information, Systems and Technologies, SECURWARE 2015,* 2015.

[231] C.Z. Acemyan, P. Kortum, M.D. Byrne, D.S. Wallach. "From Error to Error: Why Voters Could not Cast a Ballot and Verify Their Vote with Helios, Pret a Voter and Scantegrity II". *Rice University. En The Usenix Journal of Election Technology and Systems*, 2015.

[232] K. Summers, D. Chisnell, D. Davies, N. Alton, M. Mckeever. “Making Voting Accesible: Designing Digital Ballot Marking for People with Low Literacy and Mild Cognitive Disabilities”. *En JETS, The Usenix Journal of Election Technology and Systems*, 2014.

[233] J. Puiggalí, J. Chóliz, S. Guasch. “Best Practices in Internet Voting”. *Scytl,* 2009.

[234] L. J. Aceto, M. M. Shafer, E. B. Smith III, C. J. Walker. “Internet Voting System Security Auditing from System Development through Implementation: Best Practices from Electronic Voting Deployments”. *RedPhone Corporation, Data Defenders LLC, Dominion Voting Systems,* 2012.

[235] P. Locher, R. Haenni. “Verifiable Internet Elections with Everlasting Privacy and Minimal Trust”. *En VoteID 2015,* 2015.

[236] D. Springall, T. Finkenauer, Z. Durumeric, J. Kitcat, H. Hursti, M. MacAlpine, J. A. Halderman. “Security Analysis of the Estonian Internet Voting System”. *University of Michigan, Open Righs Group, U.K. In CCS 2014. ACM 978-1-4503-2957-6/14/11*, 2014.

[237] Internet Voting Solution, 2013. Cybernetica AS. http://cyber.ee/uploads/2013/03/cyber_ivoting_NEW2_A4_web.pdf, 2013.

[238] M. Clayton. “Ukraine election narrowly avoided “wanton destruction”from hackers”. Christian Science Monitor. http://www.csmonitor.com/World/Passcode/2014/0617/Ukraine-election-narrowly-avoided-wanton-destruction-from-hackers-video, 2014

[239] J. A. Green. “Cyber warfare. A multidisciplinary Analysis”. *Routledge. ISBN: 978-1-138-79307-1,* 2015.

[240] A. Greenberg. “Shopping for zero-days: A Price List for Hacker’s Secret Software Exploits”. *Forbes Magazine.* http://www.forbes.com/sites/andygreenberg/2012/03/23/shopping-for-zero-days-an-price-list-for-hackers-secret-software-exploits/#6f4504256033 , 2012.

[241] M. Arapinis, V. Cortier, S. Kremer, M. Ryan. “Practical Everlasting Privacy”. *U. of Birmingham, LORIA, CNRS, France*, 2013.

[242] T. Moran, M. Naor. “Receipt-Free Universally-Verifiable Voting with Everlasting Privacy”. *Weizmann Institute of Science, Israel*, 2006.

[243] S. Heiberg, J. Willemson. “Verifiable Internet Voting in Estonia”. *Cybernetica, Estonia*, 2014.

[244] S. Heiberg, A. Parsovs, J. Willemson. “Log Analysis of Estonian Internet Voting 2013 - 2014”. *Smartmatic – Cybernetica Centre of Excellence for Internet Voting, Software Technology and Applications Competence Centre, Tartu University,* 2015.

[245] Cybernetica AS. “Internet Voting Solution”. http://cyber.ee/uploads/2013/03/cyber_ivoting_NEW2_A4_web.pdf

[246] A. Sinak, S. Özkan, H. Yildirim, M. S. Kiraz. “End-2-End Verifiable Internet Voting Protocol Based in Homomorphic Encription”. *METU, TUBITAK BILGEM UEKAE, Necmettin Erbakan University*, 2014.

[247] H. Lipmaa. “A Simple Cast-As-Intended E-Voting Protocol by Using Secure Smart Cards”. *University of Tartu*, 2012.

[248] Gobierno de Estonia. “INTERNET VOTING in Estonia”. http://issuu.com/vabariigi_valimiskomisjon/docs/internet_voting_in_estonia, 2016.

[249] S. Heiberg, A. Parsovs, J. Willemson. “Log Analysis of Estonian Internet Voting 2013 - 2015”. *Smartmatic – Cybernetica Centre of Excellence for Internet Voting, Software Technology and Applications Competence Centre, Tartu University,* 2015.

[250] Ministry of Local Government and Regional Development of Norway. “Specification, tenders, evaluation and contract of the Internet Voting Pilot”.https://www.regjeringen.no/en/historical-archive/Stoltenbergs-2nd-Government/Ministry-of-Local-Government-and-Regiona/tema-og-redaksjonelt-

innhold/kampanjesider/e-vote-trial/System-documentation/specification-tenders-evaluation-and-con/id612121/ , 2010.

[251] Ministry of Local Government and Modernisation. “Internet voting pilot to be discontinued”. https://www.regjeringen.no/en/aktuelt/Internet-voting-pilot-to-be-discontinued/id764300/ , 2014.

[252] I. S. Gebhardt Stenerud, C. Bull. “When Reality Comes Knocking Norwegian Experiences with Verifiable Electronic Voting”. *Norwegian Ministry of Local Government and Regional Dev.* 2012.

[253] H. Nore. “Can we trust internet voting? Internet voting in Norway”. *The Ministry of Local Government and modernization*, Nov. 2013.

[254] The Baltic Course. “European Human Rights Court accepts appeal of Estonian e-voting critics”http://www.baltic-course.com/eng/baltic_states/?doc=115942 2016.

[255] F. Khaki. “Implementing End-to-End Verifiable Online Voting for Secure, Transparent and Tamper-Proof Elections*”. International Data Corporation, sponsored by Scytl*, 2014.

[256] T. E. Bjørstad. “Technical Report. Source Code Audit of Norwegian Electronic Voting System. Ministry of Local Government and Regional Develop.”. *mnemonic AS,* 2013.

[257] J. Barrat i Esteve, B. Goldsmith, J. Turner. “Speed and Efficiency of the Vote Counting Process. Norwegian E-Vote Project”. *International Foundation for Electoral Systems (IFES)*, 2012.

[258] J. Barrat i Esteve, B. Goldsmith, J. Turner. “International Experience with E-Voting. Norwegian E-Vote Project”. *International Foundation for Electoral Systems (IFES)*, 2012.

[259] United Nations. “Conventions on the Rights of Persons with Disabilities”. http://www.un.org/disabilities/convention/conventionfull.shtml, 2006.

[260] The Carter Center. “Expert Study Mission Report. Internet Voting Pilot. Norway’s 2013 Parliamentary Elections”, 2014.

[261] K. Gjøsteen. “The Norwegian Internet Voting protocol”, 2013.

[262] I

[263] Tor E. Bjørstad. “The Rise and Fall of Internet Voting in Norway. (and the spiders from Mars)”. In *31th Chaos Communication Congress,* 2014.

[264] R. E. Koenig, P. Locher, R. Haenni. “A Security Flaw in the Verification Code Mechanism of the Norwegian Internet Voting System”. *Bern University of Applied Sciences,* 2013.

[265] N. J. Goodman. “Internet Voting in a Local Election in Canada”. *ISBN: 978-3-319-04351-7*. , 2014.

[266] N. J. Goodman, J. H. Pammett. “The Patchwork of Internet Voting in Canada”, 2014.

[267] Verified Voting. “Canada: British Columbia to pursue Internet voting at municipal elections”. http://thevotingnews.com/british-columbia-to-pursue-internet-voting-at-municipal-elections-vancouver-sun/ , 2015.

[268] Independent Panel on Internet Voting. “Recommendations Report to the Legislative Assembly of British Columbia”, 2014.

[269] N. J. Goodman, N. Wellsbury. “Internet Voting in Ontario: Time for Overarching Standards”. *University of Toronto, Town of Ajax*, 2015.

[270] S. Huycke, T. Tecsa. “Markham Votes 2014 – Internet Voting Program”, 2012.

[271] K. Kitteringham, A. Brouwer. “Markham’s Online Voting Experience”, 2010.

[272] Intelivote Systems inc. “Intelivote releases trends in Electronic Voting for 48 Ontario Municipalities”. http://www.intelivote.com/news/2014/12/19/intelivote-releases ,2014.

[273] Scytl online voting. “Markham selects Scytl online voting to improve accesibility and convenience”. https://www.scytl.com/en/markham-selects-scytl-online-voting-to-improve-accessibility-and-convenience/ , 2014.

[274] City of Markham. “City-wide Results Summary”. https://www.markham.ca/wps/wcm/connect/markhampublic/2f142444-8111-4021-8e0f-44a87382302a/City+Wide+Results+Markham+2014+%28By+Voting+Subdivision%29.pdf?MOD=AJPERES&CACHEID=2f142444-8111-4021-8e0f-44a87382302a , 2015.

[275] Halifax Regional Council. “2016 Municipal and School Board Election Council Report”. http://www.halifax.ca/council/agendasc/documents/151201ca1421.pdf , 2015.

[276] The City of Edmonton. “2012 Jellybean Internet Voting Election Public Involvement Campaign”. http://www.edmonton.ca/city_government/municipal_elections/2012-jellybean-internet-voting-election-public-involvement.aspx , 2013.

[277] Swiss Info. “Hacking fears jeopardize evoting rollout”. http://www.swissinfo.ch/eng/voting-with-a-click_hacking-fears-jeopardise-e-voting-rollout/41635672 2 sept. 2015.

[278] Federal Chancellery of the Swiss Confederation. “New Provisions for Internet Voting”. https://www.bk.admin.ch/themen/pore/evoting/index.html?lang=en , 2014.

[279] D. Galindo, S. Guasch, J. Puiggalí. “2015 Neuchâtel’s Cast-as-Intended Verification Mechanism”. *Scytl Secure Electronic Voting, Spain. En VoteID 2015, Berna,* 2015.

[280] Richard Hill. “Challenging an e-voting system in court”. *Hill & Associates, Ginebra, Suiza. En VoteID 2015,* 2015.

[281] M. Germann, U. Serdült. “Internet Voting for Expatriates: The Swiss Case”. *Journal of eDemocracy 6(2): 197 - 215. ISSN: 2075-9517*. 2014.

[282] J. Gerlach, U. Gasser. “Three Case Studies from Switzerland”. *Berkman Center Research Publication No. 2009-03.1,* 2009.

[283] B. Perriard. “Vote électronique: the long path towards the digitalization of political rights”. *Swiss Federal Chancellery,* 2015.

[284] M. Chevallier, M. Warynski and A. Sandoz. “Success factors of Geneva’s e-voting system”. *Cancillería Federal y Cantón de Ginebra,* 2005.

[285] Office for Democratic Institutions and Human Rights. “Swiss Confederation: Federal Assembly Elections. 18 October 2015. Final Report”. *Varsovia*, 2016.

[286] Electoral Council of Australia. “Internet Voting in Australian election systems”, 2013.

[287] J. A. Halderman, V. Teague. “The New South Wales iVote System: Security Failures and Verification Flaws in a Live Online Election”. *U. of Michigan, U. of Melbourne. In VoteID 2015*, 2015.

[288] I. Brightwell, J. Cucurull, D. Galindo, S, Guasch. “An overview of the iVote 2015 Voting System”. *New South Wales Electoral Commission, Scytl Secure Electronic Voting,* 2015.

[289] C. Burton, C. Culnane, S. Schneider. “Secure and Verifiable Electronic Voting in Practice: The use of vVote in the Victorian State Election”. *Victorian Electoral Commission, University of Surrey*, 2015.

[290] N. Heninger. “Factoring as a service. Crypto 2013 rump session”. http://crypto.2013.rump.cr.yp.to/981774ce07e51813fd4466612a78601b.pdf , 2013.

[291] Electoral Commission New South Wales Homepage. http://www.elections.nsw.gov.au/voting/ivote.

[292] R. Smith. “Internet Voting and Voting Interference. A report for the New South Wales Electoral Commission”. https://www.elections.nsw.gov.au/__data/assets/pdf_file/0003/118380/NSWEC_2013_Report_V2.0.pdf , 2013.

[293] Electoral Commission NSW. "Report on the Conduct of the 2015 State General Election". http://www.office.elections.nsw.gov.au/__data/assets/pdf_file/0011/205688/2015_NSWEC_Report_on_the_Conduct_of_the_2015_State_General_Election_AC.pdf , 2015.

[294] N. Hastings, R. Peralta, S. Popoveniuc, A. Regenscheid. "Security Considerations for Remote Electronic UOCAVA Voting" *National Institute of Standards and Technology (NIST), U.S. Department of Commerce., NISTIR 7770,* 2011.

[295] "iVote Advisory Committee, Final Report". *iVote Advisory Committee, Utah, USA*, 2015.

[296] F. Hao, P. Y. A. Ryan, J. A. Halderman. "Real World Electronic Voting: Design, Analysis and Deployment. Chapter 7: Practical Attacks on Real-world E-voting". *No editado a la hora de escribir estas líneas. ISBN: 978-1498714693*, 2016.

[297] W. J. Kelleher. "Internet Voting in the USA: History and Prospects; or, How NIST has Misled Congress and the American People about Internet Voting Insecurity". *The Internet Voting Research and Education Fund*. 2013.

[298] Department of Neighborhood Empowerment, Los Angeles, USA. "2016 Neighborhood Council Online Election Voter Registration Portal". http://empowerla.org/onlinevoting/ , 2016.

[299] D. Jefferson, B. Simons. "California's Internet Voting Initiatives". http://cacm.acm.org/blogs/blog-cacm/198792-californias-internet-voting-initiatives/fulltext Communications of the ACM, 2016.

[300] T. Hester Jr. "New Jersey: Bill to Permit Overseas & Military Voters to Vote Using Internet Advances". PolitickerNJ. http://politickernj.com/2015/12/bill-to-permit-overseas-military-voters-to-vote-using-internet-advances/ , 2015.

[301] Verified Voting Foundation. Verified Voting, Internet Voting Resources. https://www.verifiedvoting.org/resources/internet-voting/ .

[302] Federal Voting Assistance Program (FVAP). "Review of FVAP's Work Related to Remote Electronic Voting for the UOCAVA Population". https://www.fvap.gov/uploads/FVAP/Reports/FVAP_EVDP_20151229_final.pdf 29 Dic. 2015.

[303] Federal Voting Assistance Program (FVAP). "UOCAVA" https://www.fvap.gov/info/laws/uocava.

[304] US Vote Foundation. https://www.usvotefoundation.org/.

[305] Federal Voting Assistance Program (FVAP). "Voting Over the Internet Pilot Project". https://www.fvap.gov/uploads/FVAP/Reports/voi.pdf , 2001.

[306] D. Jefferson, A. Rubin, B. Simons, D. Wagner. "A Security Analysis of the Secure Electronic Registration and Voting Experiment (SERVE)". http://www.servesecurityreport.org , 2004.

[307] The United States Department of Justice. "The Help America Vote Act of 2002". https://www.justice.gov/crt/help-america-vote-act-2002 , 2002.

[308] B. Westfall. "Internet Voting". *West Virginia Secretary of State's Office*. http://bowencenterforpublicaffairs.org/wp-content/uploads/2014/06/Internet-Voting-West-Virginia.pdf , 2014.

[309] National Conference of State Legislatures. "The Canvass, States and Election Reforms*". Issue 37.* http://www.ncsl.org/Portals/1/Documents/legismgt/elect/Canvass_Feb_2013_no_37.pdf , 2013.

[310] I. Lapowsky. "Utah's Online Caucus gives Security Experts Heart Attacks". *Wired Magazine, March 2016 Issue.* http://www.wired.com/2016/03/security-experts-arent-going-like-utahs-online-primary/ 2016.

[311] A. McLaughlin. Media Release: "Board Announces Public Test of Digital Vote by Mail Service". https://www.dcboee.org/pdf_files/nr_588.pdf , 2010.

[312] A. McLaughlin. Media Release: "Board Releases Statement on Hacking of Digital Vote by Mail System". https://www.dcboee.org/pdf_files/nr_595.pdf 6 de octubre de 2010.

[313] State of Alaska. División of Elections: Absentee Voting by Electronic Transmission. http://www.elections.alaska.gov/vi_bb_by_fax.php .

[314] Missouri Secretary of State Jason Kander. Military and Overseas Voting Access Portal. https://www.momilitaryvote.com/ .

[315] North Dakota Secretary of State. Elections Portal: https://vip.sos.nd.gov/absentee/Default.aspx .

[316] EDRi: I-voting problems in France. https://edri.org/edrigramnumber10-13e-voting-france-problems-2012/ 4 de julio de 2012.

[317] The Indepedent: "Fake Votes mar France's first electronic election". http://www.independent.co.uk/news/world/europe/fake-votes-mar-france-s-first-electronic-election-8641345.html 3 de junio 2013.

[318] Verified Voting. E-Voting system used in France is flawed. http://thevotingnews.com/e-voting-system-used-in-french-election-is-flawed-help-net-security/ 5 de junio 2013.

[319] Jaxenter. Out of date Java Mozilla Firefox Plugin. https://jaxenter.com/france-e-voting-blunder-as-portal-requires-out-of-date-java-mozilla-firefox-plugin-104597.html 8 junio 2012.

[320] Scytl Report. "French Ministry of Foreign Affairs. French Expats vote online in 2012 legislative elections". https://www.parliament.uk/documents/speaker/digital-democracy/FR_Successcase.pdf , 2012.

[321] A. Driza Mauer, J. Barrat. "E-voting Case Law: A Comparative Analysis". *Ashgate Publishing Limited. ISBN: 978-1-4724-4675-6*, 2015.

[322] Ministry of Justice of Finland. "Memorandum on the e-voting experiment". http://www.vaalit.fi/en/index/currentissues/electronicvoting.html (en finlandés), 2009.

[323] "The electronic voting Project and legislation". http://www.vaalit.fi/en/index/currentissues/electronicvoting/theelectronicvotingprojectandlegislation.html (memorándums en finlandés).

[324] Ministry of Justice of Finland. "Working group proposes: Four-year experiment with internet voting in municipal referenda". http://www.oikeusministerio.fi/en/index/currentissues/tiedotteet/2015/04/tyoryhmaehdottaanelivuotinenkokeilunettiaanestyksestakunnallissakansanaanestyksissa.html 22 de abril 2015.

[325] A. Vähä-Sipilä. "A Report on the Finnish E-Voting Pilot". Electronic Frontier Finland – Effi. https://www.verifiedvoting.org/wp-content/uploads/2014/09/Finland-2008-EFFI-Report.pdf , 2009.

[326] Ministry of Justice of Finland. "Electronic voting will not be developed further on the current basis". http://oikeusministerio.fi/en/index/currentissues/tiedotteet/2010/01/sahkoisenaanestyksenkehittamis.html , 2010.

[327] The Online Voting Party. "Online Voting in New Zealand". http://www.nbr.co.nz/sites/default/files/images/Online-Voting-in-New-Zealand-report_0.pdf , 2014.

[328] The Department of Internal Affairs of New Zealand. "Online Voting". http://www.dia.govt.nz/online-voting , 2016.

[329] Radio NZ. "Dissapointment online-voting pin pulled". http://www.radionz.co.nz/news/political/301906/disappointment-online-voting-pin-pulled , 2016.

[330] The Department of Internal Affairs of New Zealand. "Requirements for a trial of online voting in local elections. A framework to guide Local Government". https://www.dia.govt.nz/vwluResources/Online-voting-requirementsPDF/$file/Online-voting-requirements-2015-05-12.pdf , 2016.

[331] The Department of Internal Affairs of New Zealand. "Requirements for a trial of online voting in local elections. A framework to guide Local Government". Updated Version.

https://www.dia.govt.nz/vwluResources/Reqs-for-trial-online-voting-in-local-elections-Nov15-pdf/$file/Reqs-for-trial-online-voting-in-local-elections-Nov15-pdf.pdf , 2015.

[332] Office of the Associate Minister of Local Government of New Zealand. "Councils' progress on plans to trial online voting in the 2016 local elections". https://www.dia.govt.nz/diawebsite.nsf/Files/Online-voting-cabinet-papers-2016/$file/Cabinet-paper-Councils-progress-on-plans-to-trial-online-voting-in-the-2016-local-elections.pdf 2016.

[333] J. Kantor. "Arizonians vote in their pajamas". http://www.slate.com/articles/news_and_politics/net_election/2000/03/arizonans_vote_in_their_pajamas.html , 2000.

[334] Smartmatic. "Utah – Republican Presidential Caucus". http://www.smartmatic.com/case-studies/article/utah-republican-presidential-caucus-2016/ 2016.

[335] R. M. Alvarez, T. E. Hall. "Electronic Elections. The Perils and Promises of Digital Democracy". *Princeton University Press. ISBN: 978-0-691-12517-6*. 2008.

[336] C. M. Smith. "Convenience Voting and Technology. The Case of Military and Overseas Voters". *PALGRAVE MACMILLAN. ISBN: 978-1-137-39858-1*, 2014.

[337] J. Eversen. "Utah GOP wants to keep online voting, despite worries". http://www.deseret-news.com/article/865651611/Utah-GOP-wants-to-keep-online-voting-despite-worries.html?pg=all , 2016.

[338] S. Haber, J. Benaloh, S. Halevi. "The Helios e-voting Demo for the IACR". https://www.iacr.org/elections/eVoting/heliosDemo.pdf *IACR*, 2010.

[339] G. Tsoukalas, K. Papadimitriou, P. Louridas, P. Tsanakas. "From Helios to Zeus". *USENIX Journal of Election Technology and Systems (JETS). Vol 1, Number 1,* 2013.

[340] P. Louridas, G. Tsoukalas, K. Papadimitriou, P. Tsanakas. "Zeus: Bringing Internet Voting to Greece", 2013.

[341] Zeus Electronic Voting.Source Code: https://github.com/grnet/zeus .

[342] Zeus Electronic Voting. https://zeus.grnet.gr/zeus/ .

[343] The Signal. "Nova Scotia NDP breaks record with Electronic Voting". http://signalhfx.ca/nova-scotia-ndp-breaks-record-with-electronic-voting/ , 2016.

[344] The City of Edmonton. "Internet voting", 2016.

[345] Scytl. "Ministry of Interior. France". https://www.scytl.com/en/customer/ministry-of-interior/ , 2015.

[346] Scytl. "Ministry of Education. France". https://www.scytl.com/en/customer/ministry-of-education/ , 2014.

[347] P. Nessmann. "Could the future of Democracy be Digital?". *French National Center for Scientific Research (CNRS)*. https://news.cnrs.fr/articles/could-the-future-of-democracy-be-digital 2015.

[348] Departamento de Seguridad del Gobierno Vasco. "Voto Electrónico. Demotek." http://www.euskadi.eus/botoelek/euskadi/eusk_demotek_exp_c.htm.

[349] J. Barrat i Esteve. "Retos sociales y jurídicos de las votaciones electrónicas. Informe sobre las pruebas desarrolladas en Jun (Granada)". *Universidad de León*. http://www.votobit.org/archivos/jun_argentina.pdf , 2004.

[350] J. Barrat, J. M. Reniu. "Electronic Democracy and Citizen Participation. Madrid Participa". Observatorio del Voto Electrónico. http://www.votobit.org/archivos/participaingles.pdf , 2004.

[351] J. Barrat, J.M. Reniu. "Informe de las experiencias de voto electrónico empleadas en las elecciones catalanas de noviembre de 2003". *Universidad de León y Universidat de Barcelona*. http://www.votobit.org/archivos/informe_203.pdf 2004.

[352] Observatorio Voto Electrónico. “Informe 2M6”. http://www.votobit.org/archivos/PruebaVotoInternet2005.pdf , 2005.

[353] E. Montalbán Calderón. “E-voto en el ámbito de las elecciones locales”. Institut de Ciències Polítiques i Socials. https://ddd.uab.cat/pub/worpap/2015/hdl_2072_253825/ICPSWP338.pdf , 2015.

[354] La voz de Barcelona. “Absuelven al periodista acusado de suplantar la identidad de Fernández Díaz en la votación de la Diagonal”. http://www.vozbcn.com/2013/02/23/138403/absuelven-periodista-identidad-fernandezdiaz/ 23 de febrero de 2013.

[355] Agora Voting/nVotes. http://agoravoting.org/#index .

[356] Austrian Computer Society. “Austria e-government”. http://www.ocg.at/ak/edemocracy/wiki2/en/doku.php?id=projects:austria:e-government_and_e-voting_in_austria#austrian_students_union_election_2009 2011.

[357] Verified Voting Foundation. “E-Voting Pilot in Austria Cancelled by Constitutional Court”. http://thevotingnews.com/e-voting-pilot-in-austria-cancelled-by-constitutional-court-wu-ac-at/ , 2012.

[358] Council of Europe. Venice Commission. “European Standards of Electoral Law in Contemporary Constitutionalism”. *Council of Europe Publishing, ISBN: 92-871-5909-2* , 2005.

[359] H. Jonker, S. Mauw, J. Pang. “Privacy and Verifiability in Voting Systems”. *University of Luxembourg*, 2013.

[360] O. Kulyk, V. Teague, M. Volkamer. “Extending Helios Towards Private Elegibility Verifiability”. *Technische Universität Darmstadt, University of Melbourne, Karlstad Univ*, 2015.

[361] O. Kulyk, V. Teague, M. Volkamer. “Extending Helios Towards Private Elegibility Verifiability”. *Vote ID 2015,* 2015.

[362] V. Cortier. “Formal Verification of E-voting: solutions and challenges”. *ACM SIGLOG News Vol. 2, No. 1,* 2015.

[363] O. Kulyk, M. Volkamer. “Efficiency Comparison of Various Approaches in E-Voting Protocols”. *Technische Universität Darmstadt, Karlstad University*, 2015.

[364] M. de Vries, W. Bokslag. “Evaluating e-voting: theory and practice”. *Technical University of Eindhoven*, 2016.

[365] V. Cortier, D. Galindo, R. Küsters, J. Müller, T. Truderung. “Verifiabilty Notions for E-Voting Protocols”, 2015.

[366] V. Cortier, D. Galindo, R. Küsters, J. Müller, T. Truderung. “SoK: Verifiability Notions for E-Voting Protocols”, 2015.

[367] S. Kremer, M. Ryan, B. Smyth. “Election verifiability in electronic voting protocols”. *CNRS & INRIA, University of Birmingham*, 2008.

[368] P. Y. A. Ryan. “Pretty Good Democracy”. *University of Luxembourg.* 2009.

[369] S. Guasch Castelló. “Individual Verifiability in Electronic Voting”. *Universidad Politécnica de Cataluña,* 2016.

[370] D. Chung, M. Bishop, S. Peisert. “Distributed Helios – Mitigating Denial of Service Attacks in Online Voting”. *University of California Davis*, 2016.

[371] V. Cortier, B. Smyth. “Attacking and fixing Helios: An analysis of ballot secrecy”. *CNRS & INRIA*, 2013.

[372] S. Estehghari, Y. Desmedt. “Exploiting the Client Vulnerabilities in Internet E-voting Systems: Hacking Helios 2.0 as an Example”. *University College London, UCL y RCIS*, 2010.

[373] C.Z. Acemyan, P. Kortum, M.D. Byrne, D.S. Wallach. "Usability of Voter Verifiable, End-to-end Voting Systems: Baseline Data for Helios, Pret a Voter and Scantegrity II". Rice University. *En JETS, The Usenix Journal of Election Technology and Systems*. *ISBN 978-1-931971-14-0*, 2014.

[374] S. Estehghari, Y. Desmedt. "Presentation: Exploiting the Client Vulnerabilities in Internet E-voting Systems: Hacking Helios 2.0 as an Example". *University College London, UCL y RCIS, AIST Japan*, 2010.

[375] P. Bulens, D. Giry, O. Pereira. "Running mixnet-based elections with Helios". *BlueKrypt, Université Catholique de Louvain,* 2011.

[376] Y. Desmedt, P. Chaidos. "Applying Divertibility to Blind Ballot Copying in the Helios Internet Voting System". *University College London,* 2012.

[377] V. Cortier, D. Galindo, S. Glondu, M. Izabachène. "Election Verifiability for Helios under Weaker Trust Assumptions". *LORIA – CNRS, INRIA, École Polytechnique Féminine,* 2014.

[378] P. Y. A. Ryan, P. B. Rønne, V. Iovino. *"Selene: Voting with Transparent Verifiability and Coercion-Mitigation"*, 2015.

[379] K. Salamonsen. "A Security Analysis of the Helios Voting Protocol and Application to the Norwegian County Election". *Norwegian University of Science and Technology*, 2014.

[380] J.-L. Weber, U. Hengartner. "Usability Study of the Open Audit Voting System Helios". *University of Waterloo*, 2009.

[381] D. Bernhard, O. Kulyk, M. Volkamer. "Security Proofs for Participation Privacy and Stronger Verifiability for Helios". *University of Bristol, Technische Universität Darmstadt, Karlstad University,* 2016.

[382] I. Damgård.: "On σ-protocols". Disponible en: http://www.cs.au.dk/~ivan/Sigma.pdf , 2010.

[383] C.- P. Schnorr.: "Efficient Identification and Signatures for Smart Cards". *En Gilles Brassard, editor, Advances in Cryptology - CRYPTO '89, vol. 435 de Lecture Notes in Computer Science, pp. 239–252. Springer,* 1989.

[384] S. Goldwasser, S. Micali: "Probabilistic encryption". *Journal of Computer and System Sciences 28(2), pp. 270-299*. 1984.

[385] M. Naor, M. Yung: "Public-key cryptosystems probably secure against chosen ciphertext attacks". *En Proceedings of the 22nd annual ACM symposium on Theory of Computing, pp. 427-437, ACM,* 1990.

[386] C. Rackoff, D. Simon: "Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack". *En J. Feigenbaum (ed.) Advances in Cryptology – CRYPTO'91, LNCS, vol. 576, pp. 433-444. Springer Berlin Heidelberg,* 1992.

[387] E. Fujisaki, T. Okamoto, D. Pointcheval, J. Stern: "RSA-OAEP is secure under the RSA assumption". *En Journal of Cryptology, vol 17, pp.: 81-104*, 2004.

[388] B. Chevallier-Mames, P. Fouque, J. Stern, D. Pointcheval, and J. Traore. "On some incompatible properties of voting schemes". *En Proc. IAVoSS Workshop On Trustworthy Elections,* 2006.

[389] S. Kremer, M. Ryan, B. Smyth: "Election verifiability in electronic voting protocols". *En Proc. 15th ESORICS, vol. LNCS 6345, pp. 389, 404,* 2010.

[390] R. Küsters, T. Truderung, A. Vogt: "Acountability: Definition and Relationship to Verifiability". *En Proc. 17th ACM Conference on Computer and Communications Security, pp. 526–535, 2010.*

[391] R. Chadha, Ş. Ciobâcă, S. Kremer: "Automated verification of equivalence properties of cryptographic protocols". *En 21st ESOP'12, vol 7211 of LNCS, Springer*, 2012.

[392] B. Blanchet, M. Abadi, C. Fournet: "Automated verification of selected equivalences for security protocols". *Journal of Logic and Algebraic Programming, 75(1),* 2008.

[393] International Association for Cryptologic Research. Elections page: http://www.iacr.org/elections/.

[394] Código fuente actualizado de Helios Voting. https://github.com/benadida/helios-server.

[395] Documentación Helios Voting: http://documentation.heliosvoting.org/ .

[396] Página principal de Helios Voting: https://vote.heliosvoting.org/ .

[397] V. Shoup, R. Gennaro: "Securing threshold cryptosystems against chosen ciphertext attack". *Journal of Cryptology 15, 2, pp. 75-96*, 2002.

[398] B. Terelius, D. Wikström: "Proofs of restricted shuffles". En *Progress in Cryptology, AFRICACRYPT 2010, vol. 6055 de LNCS, pp. 100-113*, 2010.

[399] R. Cramer, V. Shoup: "A practical public key cryptosystem provably secure against adaptative chosen ciphertext attack". En *CRYPTO'98. Vol. 1462 LNCS, pp.: 13-25,* 1998.

[400] D. Chaum, T.P. Pedersen: "Wallet databases with observers". *CRYPTO'92, vol. 740 de LNCS, pp.89–105, Springer*, 1992.

[401] R. Cramer, I. Damgård, B. Schoenmakers: "Proofs of partial knowledge and simplified design of witness hiding protocols". *CRYPTO'94, vol. 839 de LNCS, pp. 174–187, Springer,* 1994.

[402] S. Glondu: "Helios with credentials: Proof of concept and mock election results". http://stephane.glondu.net/helios 2013.

[403] D. Bernhard, V. Cortier, D. Galindo, O. Pereira, B. Warinschi: "SoK: A comprehensive analysis of game-based ballot privacy definitions". *En 2015 IEEE Symposium on Security and Privacy, pp. 499 – 516, IEEE,* 2015.

[404] L. Lamport: "The part-time parliament". *ACM Transactions on Computer Systems, 16(2), pp.: 133 – 169,* 1998.

[405] T. D. Chandra, R. Griesemer, J. Redstone: "Paxos made live: An engineering perspective". En *Proceedings of the 26th Annual ACM Symposium on Principles of Distributed Computing, pp. 398–407,* 2007.

[406] L. Lamport, R. Shostak, M. Pease: "The byzantine generals problem". *ACM Transactions on Programming Languages and Systems, 4(3), pp. 382–401*, 1982.

[407] L. Lamport: "Fast Paxos". *Distributed Computing, 19(2), pp. 79–103*, 2006.

[408] D. Ongaro, J. Ousterhout: "In search of an understandable consensus algorithm". *In Proceedings of the 2014 USENIX Annual Technical Conference, pp. 305–319,* 2014.

[409] B. Smyth, S. Frink, M. R. Clarkson: "Election Verifiability: Cryptographic Definitions and an Analysis of Helios and JCJ (Technical report)". *IACR*, 2016.

[410] F. Karayumak, M. M. Olembo, M. Kauer, M. Volkamer: "Usability Analysis of Helios – An Open Source Verifiable Remote Electronic Voting System". *CASED, Technische Uni,* 2010.

[411] IACR: "Final Report of IACR Electronic Voting Committee". https://www.iacr.org/elections/eVoting/finalReportHelios_2010-09-27.html , 2010.

[412] International Association for Cryptologic Research. https://www.iacr.org/ .

[413] Princeton University, Undergraduate Elections. https://princeton.heliosvoting.org/ .

[414] Web Content Accesibility Guidelines. https://www.w3.org/TR/WCAG20/ .

[415] Tawdis, CTIC Centro Tecnológico. http://www.tawdis.net/ .

[416] WAVE: Web Accessibility Evaluation Tool. http://wave.webaim.org/ .

[417] Access Monitor. Fundación para la ciencia y tecnología. Portugal. http://www.acessibilidade.gov.pt/accessmonitor/ .

[418] Gobierno de Noruega. https://www.regjeringen.no/globalassets/upload/krd/kampanjer/valgportal/e-valg/e_valg_systemlosning/tilbud_ergogroup/ssa-u_appendix_7_total_price_and_pricing_provisions.pdf .

[419] Scytl, R&D Department. Articles and Publications. https://www.scytl.com/en/articles-and-publications/ .

[420] S. Guasch, D. Galindo, J. Puiggalí: "2015 Neuchâtel's Cast-as-Intended Verification Mechanism". *VoteID 2015: The 5th International Conference on e-Voting and Identity,* 2015.

[421] Y. Tsiounis, M. Yung: "On the Security of ElGamal Based Encryption". *PKS'98, LNCS 1431, pp. 117-134*, 1998.

[422] Scytl, proyectos y clientes. https://www.scytl.com/es/category/proyectos-y-clientes/ .

[423] Nace la Start-up Agora Voting. https://blog.agoravoting.org/index.php/2014/09/25/nace-la-startup-agora-voting/ , 2014.

[424] nVotes homepage. https://nvotes.com/ .

[425] D. Ruescas, E. Robles: "Agora Voting: Technical Overview", 2015.

[426] Agora Voting: "Protocolo de actuación - #CLIENTE#", 2015.

[427] Agora Voting: "Guía de uso (administrador)", 2015.

[428] D. Wikström: "Verificatum". http://www.verificatum.com/.

[429] NIST: "Federal Information Processing Standards Publication 197: Advanced Encryption Standard (AES)". http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf , 2001.

[430] C. Bull, H. Nore, S. Guasch, J. Puiggalí: "Internet Voting in Europe: Norway and Switzerland case studies". *NVTS AS, Scytl Secure Electronic Voting*. 2016.

[431] 20 minutos: http://www.20minutos.es/noticia/2419700/0/podemos-defiende-fiabilidad/sistema-votacion-acusaciones/primarias/ 2015.

[432] El Confidencial Digital: http://www.elconfidencialdigital.com/politica/Podemos-evitar-consulta-llene-trolls_0_2691930793.html 2016.

[433] El Español. http://www.elespanol.com/espana/20160511/123987880_0.html , 2016.

[434] El Confidencial. http://www.elconfidencial.com/espana/2014-11-01/el-sistema-de-afiliacion-a-podemos-se-puede-manipular-y-existe-la-posibilidad-de-fraude_433634/ 2014.

[435] Impact Accelerator. http://www.impact-accelerator.com/portfolio/.

[436] CloudFlare: https://www.cloudflare.com/.

[437] Fail2ban: http://www.fail2ban.org/wiki/index.php/Main_Page.

[438] M. Maffei, E. Tuosto: "Trustworthy Global Computing: 9th International Symposium, TGC 2014". *LNCS 8902*, 2014

[439] Verificatum: "Complexity Analysis of the Verificatum Mix-net". Versión 3.0.2. 2015.

[440] S. Taha, J. Murray: "An Overview of End-to-End Verifiable Voting Systems". *National University of Sciences and Technology, Pakistan, Newcastle University*, *U.K. "Real-World Electronic Voting: Design, Analysis and Deployment". Ed.: F. Hao y Peter Y.A. Ryan, CRC Press,* 2016.

[441] S. Neumann, C. Feier, M. Volkamer, R. E. Koenig. "Towards a Practical JCJ/Civitas Implementation". *Lecture Notes in Informatics, vol. P-220,* 2013.

[442] J. M. Barros da Silva Mendes: "Trusted civitas: Client Trust in CIVITAS Electronic Voting Protocol". *Trabajo de Máster en Ingeniería Informática y de Computadores. Instituto Superior Técnico, Universidade Técnica de Lisboa,* 2011.

[443] R. Dingledine, N. Mathewson, P. F. Syverson: "Tor: The second-generation onion router". *En Proc. of USENIX Security Symposium, pp. 303-320*, 2004.

[444] R. Araujo, S. Foulle, J. Traor: "A practical and secure coercion-resistant scheme for internet voting". En *Towards Trustworthy Elections, vol. 6000 LNCS, pp. 330-342*, 2010.

[445] O. Spycher, R. E. Koenig, R. Haenni, M. Schläpfer: "A new approach towards coercion-resistant remote e-voting in linear time". En *Financial Cryptography, vol. 7305 LNCS, pp.182-189,* 2011.

[446] R. E. Koenig, R. Haenni, S. Fischli. "Preventing board flooding attacks in coercion-resistant electronic voting-schemes". En *SEC, pp. 116-127*, 2011.

[447] F. Shirazi, S. Neumann, I. Ciolacu, M. Volkamer: “Robust electronic voting: Introducing robustness in Civitas”. En *International Workshop on Requirements Engineering for Electronic Voting Systems (REVOTE), IEEE Computer Society, pp. 47-55,* 2011.

[448] S. Neumann, M. Volkamer: “Civitas and the real world: Problems and solutions from a practical point of view”. En *ARES 2012, IEEE Computer Society, pp. 180-185*, 2012.

[449] O. Blazy, G. Fuchsbauer, D. Poincheval, D. Vergnaud: “Signatures on Randomizable Ciphertexts”. En *International Conference on Theory and Practice in Public Key Cryptography PKC 2011, LNCS 6571, pp. 403–422,* 2011.

[450] S. Glondu: “Belenios specification”. Version 0.1. *http://www.belenios.org/specification.pdf* 2013.

[451] Belenios official website. *http://www.belenios.org/index.php* .

[452] R. Canetti, H. Krawczyk, J. B. Nielsen. “Relaxing Chosen-Ciphertext Security”, 2003.

[453] Open Seneca. Official Website. *http://www.openseneca.com/*.

[454] L.A. Goodman. “Snowball sampling”. *Annals of Mathematical Statistics. 32 (1), pp. 148–170*. 1961.

[455] Deming, W. Edwards. “Sample Design in business research”. *John Wiley and Sons. p. 31. ISBN: 0-471-52370-4,* 1990.

[456] Directorate General of Democracy and Political Affairs. “Certification of e-voting systems”. *Council of Europe*, 2011.

[457] D. Simic-Draws, S. Neumann, A. Kahlert, P. Richter, R. Grimm, M. Volkamer, A. Rossnagel. “Holistic and Law Compatible IT Security Evaluation: Integration of Common Criteria, ISO 27001/IT-Grundschutz and KORA”. *International Journal of Information Security and Privacy, 7(3), pp. 16-35,* 2013.

[458] S. Neumann. “Evaluation and Improvement of Internet Voting Schemes Based on Legally-Founded Security Requirements”. *Ph. D Thesis. TU Darmstadt*, 2016.

[459] M. Volkamer, D. Hutter. “From Legal Principles to a Internet Voting System”. *German Research Center for Artificial Intelligence, DFKI Saarbrücken,* 2002.

[460] Common Criteria for Information Technology Security Evaluation. SO/IEC 15408:2009. *https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R3.pdf* .

[461] KORA (Konkretisierung Rechtlicher Inforderungen, Concretization of Legal Requirements). V. Hammer, U. Pordesch, A. Rossnagel. “Betriebliche Telefon- und ISDN-Anlagen rechtsgemäss gestaltet”. Spinger, 1993.

[462] ISO 27001 y German Federal Office for Information Security (BSI). “Zuordnungstabelle ISO 27001 sowie ISO 27002 und IT-Grundschutz”. *https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Hilfsmittel/Doku/Vergleich_ISO27001_GS.pdf?__blob=publicationFile* , 2013.

[463] K. Bräunlich, R. Grimm, P. Richter, A. Rossnagel. “Sichere Internetwahlen: Ein rechtswissenschaftlich-informatisches Modell”. *Nomos*, 2013.

[464] The Washington Post. “Arizona: Russian hackers targeted Arizona election system” *https://www.washingtonpost.com/world/national-security/fbi-is-investigating-foreign-hacks-of-state-election-systems/2016/08/29/6e758ff4-6e00-11e6-8365-b19e428a975e_story.html* , 29 agosto 2016.

[465] J. Carracedo, A. Gómez, J. D. Carracedo. “Sistema VOTESCRIPT: Una propuesta innovadora para resolver los problemas clásicos de la votación electrónica”. *U. Politécnica de Madrid, Univesidad Complutense de Madrid.* 2003.

[466] A. Gómez, J. Moreno, E. Pérez. “Del voto electrónico al telemático”. *Boletín Red IRIS.* 2004.

[467] J. Carracedo. “Seguridad en Redes Telemáticas”. *McGraw Hill, ISBN: 84-481-4157-1*. 2004.

[468] Ducas L. and Micciancio D. FHEW: Bootstrapping homomorphic encryption in less than a second. In: Eurocrypt 2015, April 26-30 Sofia, Bulgaria, 2015, 617-640, 2015.

[469] Kim M., Lee HT., Ling S., and Wang H. On the efficiency of FHE-based private queries. IEEE Transactions on Dependable and Secure Computing, 2016

Anexo A. Criterios de evaluación de la metodología.

Criterio	Código	Explicación
Verificabilidad extremo a extremo	**E2Ev**	Referirse al apartado 2.2.2
Privacidad/Resistencia a la coerción	**RC**	Referirse al apartado 2.2.3
Inviolabilidad	**I-*n***	
	I-1	Protección del software y sistemas auxiliares con sistemas de autenticación suficientemente seguros. Acceso desde terceras aplicaciones/web o servidores vulnerables no permitidos.
	I-2	Existencia de protocolos de actuación en caso de inviolabilidad comprometida.
	I-3	Herramientas de rastreo y copias de seguridad *offline* disponibles.
	I-4	Control distribuido de nodos críticos con reparto de funciones para minimizar riesgos de colusión.
	I-5	Existencia de protocolos de *Risk Assessment* y de *Threat Modelling*.
	I-6	Implementación de principios de modularidad para confinar en lo posible los *bugs* o ataques.
	I-7	Correcta actualización de los puntos I-1…I-6
Usabilidad	**U-*n***	
	U-1	Simplicidad en los procesos de autenticación, voto y verificación.
	U-2	Atención especial a colectivos con discapacidades/sin alfabetizar de acuerdo a lo previsto en el Consejo de Europa [54] y las Naciones Unidas [259].
	U-3	Claridad en el modo de mostrar cuándo ha concluido el proceso de votación y el voto ha sido tenido en cuenta. Tasa de éxito de votación entre votantes sin formación
	U-4	Preferencia de la integridad y la privacidad sobre la usabilidad.
	U-5	Interfaz de administrador intuitivo y sencillo para crear y gestionar unos comicios.
Monitorización/Auditoría	**MA-*n***	
	MA-1	Externa, independiente y distribuida.
	MA-2	Existencia del protocolo desde la fase de diseño, para velar por el correcto desarrollo de todo el ciclo del proyecto.
	MA-3	Control específico sobre las estrategias de *Risk Assessment* y *Thread Modelling*.
	MA-4	Generación de informes periódicos de actividad inalterables e imborrables almacenados *offline*, en instalaciones aisladas y custodiados por personal distinto al de otra áreas susceptibles de colusión.

	MA-5	Implementación práctica desde la obtención del censo electoral hasta el mantenimiento post-electoral.
	MA-6	Información detallada, bien documentada y en el formato pertinente.
	MA-7	Existencia de un "banco de pruebas" o test a modo de exámen de que el sistema funciona correctamente (ante la previsión de un ataque, antes de comenzar las elecciones, en mitad de las mismas, etc.).
	MA-8	Miembros del equipo de monitorización/auditoría independientes del resto de autoridades/administradores de las elecciones.
	MA-9	Existencia de un protocolo de auditoría de ataques producidos y otro de auditoría del propio sistema de monitorización/auditoría.
	MA-10	En caso de ataque exitoso, el sistema prima la privacidad del votante y su voto aún a costa de tener que cancelar los comicios.
Desarrollo Software	**DSW-*n***	
	DSW-1	Requisitos habituales de diseño, implementación y documentación de ingeniería del software.
	DSW-2	Enfoque distribuido del software, en especial para las operaciones críticas del mismo. Ninguna autoridad debe poder modificar atributos críticos unilateralmente.
	DSW-3	Dentro de las posibilidades, facilidad de uso y existencia de guía de usuario y administrador debidamente documentada y disponible con la suficiente antelación.
	DSW-4	Existencia de un sitio web suficientemente seguro, accesible y con una sección clara de FAQ.
	DSW-5	La forma de mostrar las distintas opciones de voto debe ser totalmente objetiva e imparcial, sin ningún tipo de preferencia por ningún candidato.
	DSW-6	El sistema de votación no debería suministrar al votante información suficiente como para poder deducir la opción elegida.
	DSW-7	El sistema debe garantizar la privacidad del votante durante todos los pasos de la votación, impidiendo reconstruir el vínculo voto-votante.
	DSW-8	El proceso de votación debe de poder cancelarse en cualquier momento sin que el sistema guarde ningún tipo de información que pudiese poner en peligro la privacidad del votante.
	DSW-9	El software debe ser testado en las distintas plataformas, sistemas operativos y navegadores que representen más de un 1% de la cuota de mercado.
	DSW-10	El software no debe permitir el acceso a través de ningún programa ajeno (incluida la *social media*) ni incluir links a webs o programas gestionados por servidores ajenos al sistema de VER.
	DSW-11	Las primitivas criptográficas se testarán con suficiente antelación en condiciones más exigentes que las de las propias elecciones para evitar colapsos y anticipar posibles necesidades extra de recursos.

	DSW-12	Acceso al código fuente por parte de expertos/invstigadores independientes para comprobar su correcto funcionamiento y buscar posibles *bugs*. La empresa desarrolladora podrá exigir la firma de un acuerdo de confidencialidad para no poner en riesgo su propiedad intelectual.
	DSW-13	Implementación en la medida de lo posible de sistemas protocolozados y estándares abiertos para facilitar la interoperabilidad.
	DSW-14	Correcta actualización del sistema, sobre todo en lo que respecta a los ataques más frecuentes en esquemas de VER.
Escalabilidad	**E-*n***	
	E-1	Prueba de la capacidad máxima del sistema tanto en su vertiente software como ex_software en entornos tan exigentes como las elecciones que van a gestionar.
	E-2	Existencia de tests específicos para las operaciones más críticas del sistema (autenticación, encriptación y desencriptación, recuento, primitivas criptográficas etc.).
	E-3	Existencia de bancos de pruebas más exigentes que las elecciones a gestionar
	E-4	Existencia de indicaciones y métricas claras del tamaño o complejidad máxima de elecciones testadas por el sistema tanto software (capacidades matemáticas y criptográficas, número de votantes etc.) como ex_software (infraestructura, costes, logística, segundos canales, recursos humanos etc.).
	E-5	Escalabilidad entendida como tipología de comicios para los que se tiene capacidad y experiencia (desde consultas no vinculantes hasta EVAP).
Desarrollo ex_software	**DESW-*n***	
	DESW-1	Diseño, desarrollo, implementación y revisión de protocolos ex_software paralelamente a la vertiente software para que conformen un todo uniforme más robusto y menos vulnerable.
	DESW-2	Existencia de un protocolo seguro de distribución de credenciales, permisos y responsabilidades.
	DESW-3	Existencia de un protocolo automatizado de control de accesos y vigilancia de las infraestructuras del sistema de VER.
	DESW-4	Existencia de un protocolo de auditoría y observadores independientes.
	DESW-5	Existencia de un protocolo de *back-up* distribuido.
	DESW-6	Filosofía de distribución de atribuciones y responsabilidades en todo el desarrollo ex_software para disminuir en lo posible el riesgo de colusión.
	DESW-7	Existencia de sistemas de votación complementarios al VER.
	DESW-8	Se informará a los votantes con suficiente antelación del proceso concreto en que se va a articular el VER poniendo a su disposición distintos canales de comunicación.

	DESW-9	En caso de existir la opción de re-votación, se reforzará la información sobre el tema, aclarando la primacía del voto tradicional en papel en caso de duda.
	DESW-10	Organización de encuestas de opinión sobre cohortes preseleccionadas para obtener *feedbacks* fidedignos sobre usabilidad, fallos, mejoras, tendencias etc.
	DESW-11	Envío de credenciales de autenticación a través de canales alternativos.
	DESW-12	Existencia de un protocolo maestro de inicialización a ejecutar inmediatamente antes del comienzo de los comicios para verificar que todos los sistemas se encuentran operativos y en disposición.
	DESW-13	Implementación en la medida de lo posible de sistemas protocolizados y estandarizados para facilitar la interoperabilidad entre ellos.
	DESW-14	Existencia de un servicio telefónico gratuito de asistencia previo a las elecciones y durante las mismas.
	DESW-15	Existencia de una estrategia de comunicación completa para publicitar y dar formación sobre el VER incluyendo jornadas abiertas presenciales y *webinars*.
Protocolo contra incidencias y ataques	**PIA-*n***	
	PIA-1	Comprobar existencia de *Risk Assessment (RA), Privacy Impact Assessment (PIAS), Penetration Testing (PT), Statement of Applicability (SoA), Control Validation Plan (CVP) y Control Validation Audit (CVA).*
	PIA-2	Existencia de protocolos específicos contra ataques y prevención reforzada en función del esquema criptográfico de VER elegido.
	PIA-3	Información e infraestructuras se mantienen en la medida de lo posible dentro del territorio donde se celebran las elecciones.
	PIA-4	Implementación de protocolos y actuaciones de refuerzo dirigidas a minimizar el riesgo de pérdidas permanentes de información.
	PIA-5	Se valora positivamente el enfoque distribuido del protocolo contra ataques de tal forma que no haya nodos críticos únicos cuyo ataque comprometa la viabilidad del sistema de vER.
	PIA-6	Existencia de actividades de formación y concienciación ciudadana dirigidas a minimizar el riesgo de ataques en los que el votante es el vector (*phishing*, ingeniería social etc.)
	PIA-7	Contratación de hackers/expertos independientes para tratar de comprometer el sistema antes de su utilización en comicios reales.
Versatilidad	**V-*n***	
	V-1	Existencia de versiones adaptadas a las distintas tipologías de elecciones existentes (referirse al apartado 2.1.2 para más detalles)
	V-2	Existencia de soluciones específicas para los colectivos más vulnerables y con más posibilidad de beneficiarse de la introducción del VER.

	V-3	El votante idealmente debe poder votar utilizando su equipo personal a través de una conexión estándar a internet y sin instalar software adicional.
	V-4	El sistema debe testarse en navegadores y dispositivos con una cuota de mercado mayor al 1%.
	V-5	El interface cumple con el nivel AA del WCAG 2.0 [414].
Coste	**C-*n***	
	C-1	Transparencia y claridad en la presentación de los costes del sistema en todas sus vertientes
	C-2	El coste del sistema con respecto a su calidad, desempeño y comparación con opciones alternativas.
Mantenimiento	**M-*n***	
	M-1	En relación con el propio sistema de VER y sus protocolos asociados. Cubre tanto la vertiente software como la ex_software. Se valora la frecuencia, rigurosidad y existecia de *logs* para comprobar el estado actual y el histórico de actualizaciones y mantenimiento.
	M-2	Mantenimiento como *everlasting privacy* [235, 241, 242]
	M-3	Coste asociado al mantenimiento.

Tabla a: Desglose completo de los criterios de la metodología de evaluación de sistemas de Voto Electrónico Remoto

Anexo B. Encuesta para la introducción de factores de ponderación a los criterios de evaluación de la metodología.

Versión en español

ENCUESTA SOBRE LA PONDERACIÓN ASIGNADA A LOS CRITERIOS DE LA METODOLOGÍA DE EVALUACIÓN DE SITEMAS DE VOTO ELECTRÓNICO REMOTO

En la metodología se incluyen dos tipos de criterios:

- Los que se denominan "sine-qua-non" por su importancia y que son: **Verificabilidad extremo a extremo** o E2Ev (definida e investigada en [2, 3, 4, 5, 6, 16]) y privacidad entendida en su definición más exigente de **resistencia a la coerción** (referirse a [7, 8, 9, 10, 11, 12] para su definición y estudios relevantes].
- Los que pueden encontrarse implementados en distintas intensidades. Para su elección se han tomado como referencia los principales trabajos en la materia: [1, 2, 4, 13, 14, 15] junto con las metodologías y sistemas siguientes:
  - Método KORA (*Concretization of Legal Requirements* en alemán) [22]
  - *Common Criteria for IT Security Evaluation SO/IEC 15408:2009* [21]
  - ISO 27001/IT-Grundschutz [23]
  - Sistema Simic-Draws et al. [18] que combina las 3 anteriores.
  - Las recomendaciones sobre certificación de sistemas de (sic) e-voting del Directorado General de Democracia y Asuntos Políticos del Consejo de Europa [17]
  - El trabajo de Volkamer sobre requisitos legales del voto [20]
  - El trabajo de Bräunlich et al. sobre la transformación de criterios legales en TDGs (*Technical Design Goals*) [24]
  - La tesis doctoral de Neumann [19], que se apoya en [18] y [24] para identificar 16 aspectos técnicos que debería cumplir un sistema de *i-voting*.

  Son los siguientes: **inviolabilidad, usabilidad, monitorización/auditoría, operacional software (el software en sí, con su diseño, implementación y revisión) y escalabilidad**. Los 43 sub-apartados concretos de cada uno de ellos se pormenorizan en la tesis doctoral.

Para complementar los criterios más "académicos", se han estudiado más de 500 elecciones realizadas sobre plataformas de VER en elecciones tanto vinculantes como no vinculantes en todo tipo de ámbitos totalizando más de 6 millones de votos en más de 10 países

y se han extraído los principales ataques y carencias a cubrir no incluidos en los requerimientos tradicionales.

Los criterios adicionales son: **desarrollo ex_software (el desarrollo de todo lo no incluido en el SW: credenciales, protocolos de control de acceso, de back-up, de auditoría, filosofía distribuida de seguridad etc.), protocolo contra ataques, versatilidad, coste y mantenimiento** y suman 31 características adicionales.

Por último y una vez definida la totalidad de factores y con el fin de dar un paso más en la metodología, se asigna a cada criterio un coeficiente de ponderación entre 0.6 y 1.2 con el fin de que los criterios más relevantes tengan un peso mayor, sin superar el doble respecto a los menos relevantes, para evitar sobre-representaciones e infra-representaciones.

Es en este punto en el que, para realizar un trabajo más sistemático y exhaustivo, se crea el siguiente formulario para recoger la opinión sobre los factores más relevantes del Voto Electrónico Remoto de una serie de expertos en el campo entre los que se encuentra usted. Le rogamos incluya su visión sobre el peso de los distintos factores, asignando un valor entre 0.6 y 1.2 (un decimal de precisión) sin que la suma total exceda los 10 puntos en total.

Le agradezco de antemano su valiosísima contribución.

Criterio	Ponderación (0,6 – 1,2)
Inviolabilidad	
Usabilidad	
Monitorización/auditoría	
Operacional SW	
Escalabilidad	
Desarrollo ex-SW	
Protocolo contra ataques	
Versatilidad	
Coste	
Mantenimiento	

Tabla b: Ponderación de criterios de evaluación

Bibliografía

[1] L. Panizo. "Desarrollo de una metodología para el análisis y la clasificación de los sistemas de voto electrónico", *Departamento de Ingeniería Eléctrica y de Sistemas y Automática, Universidad de León*, Dic. 2014.

[2] J. Benaloh, R. Rivest, P. Ryan, P. Stark, V. Teague, P. Vora. “End to End Verifiability” Feb. 2014.

[3] A. Kiayias, T. Zacharias, B. Zhang. “End-to-end verifiable elections in the standard model”. In *Elisabeth Oswald and Marc Fischlin, editors, EUROCRYPT 2015, part II, volumen 9057 de LNCS*, pp. 468-498. Springer, Abril 2015.

[4] S. Popoveniuc, J. Kelsey, A. Regenscheid, P. Voral. “Performance requirements for end-to-end verifiable elections” in *EVT/WOTE* 2010.

[5] J. Benaloh. “Simple verifiable elections”. In *Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop*, pages 5–5. USENIX Association Berkeley, CA, USA, 2006.

[6] D. Chaum. “Secret-Ballot Receipts: True Voter-Verifiable Elections”. *IEEE Security and Privacy, vol 2, no. 1, pp: 38-47*, 2004.

[7] A. Fujioka, T. Okamoto, K. Ohta. “A practical secret voting scheme for large scale elections” *ASIACRYPT’92, Workshop on the Theory and Application of Cryptographic Techniques*. Pp. 244-251. LNCS 718, Gold Coast, Australia. 1992.

[8] S. Delaune, S. Kremer, M. Ryan. “Coercion-Resistance and Receipt- Freeness in Electronic Voting”. *CSFW’06: 19th Computer Security Foundations Workshop*, pp. 28-42. IEEE Computer Society, 2006.

[9] T. Okamoto. “Receipt-Free Electronic Voting Schemes for Large Scale Elections”. In *SP’97: 5th International Workshop on Security Protocols*, ser. LNCS, vol 1361 pp. 25-35. Springer 1998.

[10] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant electronic-elections”. *Cryptology ePrint Archive, Report 2002/165,* 2002.

[11] M. Backes, C. Hritcu, M. Maffei. “Automated Verification of Remote Electronic Voting Protocols in the Applied Pi-Calculus”. In *CSF’08: 21st Computer Security Foundations Symposium. IEEE Computer Society, 2008*, pp. 195-209. 2008.

[12] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant Electronic-Elections”. *RSA Laboratories, CNRS-Ecole Normale Supérieure, Indiana University, School of Informatics*. 2010.

[13] “The Future of Voting” U.S. Vote Foundation, 4325 Old Glebe Road, Arlington VA 22207, USA. 2015.

[14] H. Li, A.R. Kankanala, X. Zou. “Taxonomy and Comparison of Remote Voting Schemes”. *Purdue University*. 2014.

[15] D. Zissis, D. Lekkas.”Design, Development and Use of Secure Electronic Voting Systems”. IGI Global. ISBN 978-1-4666-5823-3. 2014.

[16] O. Kulyk, V. Teague, M. Volkamer. “Extending Helios Towards Private Elegibility Verifiability”. Vote ID 2015. 2015.

[17] Directorate General of Democracy and Political Affairs. “Certification of e-voting systems”. Council of Europe. 2011.

[18] D. Simic-Draws, S. Neumann, A. Kahlert, P. Richter, R. Grimm, M. Volkamer, A. Rossnagel. “Holistic and Law Compatible IT Security Evaluation: Integration of Common Criteria, ISO 27001/IT-Grundschutz and KORA”. *International Journal of Information Security and Privacy, 7(3), pp.: 16-35, July-September.* 2013.

[19] S. Neumann. “Evaluation and Improvement of Internet Voting Schemes Based on Legally-Founded Security Requirements”. *Ph. D Thesis. TU Darmstadt*. 2016.

[20] M. Volkamer, D. Hutter. “From Legal Principles to a Internet Voting System”. German Research Center for Artificial Intelligence, DFKI Saarbrücken. 2002.

[21] *Common Criteria for Information Technology Security Evaluation. SO/IEC 15408:2009.* https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R3.pdf

[22] KORA (Konkretisierung Rechtlicher Inforderungen, Concretization of Legal Requirements). V. Hammer, U. Pordesch, A. Rossnagel. "Betriebliche Telefon- und ISDN-Anlagen rechtsgemäss gestaltet". Spinger, 1993.

[23] ISO 27001 and German Federal Office for Information Security (BSI). "Zuordnungstabelle ISO 27001 sowie ISO 27002 und IT-Grundschutz". https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Hilfsmittel/Doku/Vergleich_ISO27001_GS.pdf?__blob=publicationFile 2013.

[24] K. Bräunlich, R. Grimm, P. Richter, A. Rossnagel. "Sichere Internetwahlen: Ein rechtswissenschaftlich-informatisches Modell". Nomos, 2013.

Versión en inglés

SURVEY ON WEIGHTING COEFFICIENTS ASSIGNED TO THE CRITERIA FOR THE EVALUATION METHODOLOGY FOR I-VOTING SYSTEMS

For the evaluation of i-voting schemes, there are two main types of requirements:

- The "*Sine-qua-non*" ones, because of the properties they embody and preserve: Verifiability (individual, universal and eligibility verifiability) [2, 3, 4, 5, 6, 16] and privacy as Coercion Resistance according to the definition by Juels et al. [10, 12].
- The ones which can be implemented to different degrees for every voting system. For its selection, the following references have been taken into account [1, 2, 4, 13, 14, 15] together with the following methodologies and studies:

    - KORA method (*Concretization of Legal Requirements* in German) [22]
    - *Common Criteria for IT Security Evaluation SO/IEC 15408:2009* [21]
    - ISO 27001/IT-Grundschutz [23]
    - Simic-Draws et al. [18] methodology, combining the previous 3 items.
    - The reccomendations on e-voting systems certification by the Directorate General for Democracy and Political Affairs of the Council of Europe [17]
    - The study on legal e-voting legal requirements by Volkamer et al. [20]
    - The work by Bräunlich et al. on Legal Criteria translation into TDGs (*Techdnical Design Goals*) [24]
    - Dr. Neumann's PhD thesis [19].

    They are: **inviolability, usability, monitoring/auditing, Software Development and scalability**. Each one of them is sub-divided in concrete, measurable items, totaling 41 (the complete explanation, with the definition of each requirement and item is included in the Ph. D thesis hopefully to be presented in the current academic year).

In order to complement the traditional requirements, the author has undergone a mixed qualitative and quantitative case study of the Switzerland, Canada, Estonia, Austria, Spain,

France, Australia, New Zealand, Norway, Finland, Germany, the UK and Holland experiences in both politically binding and non-binding environments, adding up to more than 600 elections and 6 million votes cast.

The additional criteria are: **ex_software development (including: credentials, Access-control protocols, back-up, auditing, distributed approach etc.), events and attacks protocol, versatility, cost and maintenance.** Similarly to the traditional requirements, each additional criteria is divided into concrete, measureable items totalling 32.

Therefore, the evaluation methodology is comprised of 75 items: 2 of them unconditional (verifiability and privacy) and 73 measurable, organized in 10 requirements. As a last step towards a robust methodology, weighting factors are assigned to the measurable requirements, with the invaluable feedback from both industry and academia i-voting experts. The weighting coefficient ranges from 0.6 to 1.2 so the most relevant ones are worth at most double of the least critical ones, avoiding over and under-representations.

The following form is a very simple survey in order to gather the opinion of relevant researchers and industry leaders in the field of i-voting. It would be greatly appreciated if you could share your opinion on the weighting factors. We honestly believe that it will contribute to a more robust, systematic and fair evaluation methodology.
Thank you very much in advance for your understanding and cooperation.

Requirement	Weighting (0,6 – 1,2)
Inviolability	
Usability	
Monitoring/auditing	
SW Development	
Scalability	
ex-SW Development	
Events and attacks protocol	
Versatility	
Cost	
Maintenance	

Tabla c: Ponderación de criterios de evaluación (inglés)

Bibliography

[1] L. Panizo. "Desarrollo de una metodología para el análisis y la clasificación de los sistemas de voto electrónico", *Departamento de Ingeniería Eléctrica y de Sistemas y Automática, Universidad de León*, Dic. 2014.

[2] J. Benaloh, R. Rivest, P. Ryan, P. Stark, V. Teague, P. Vora. “End to End Verifiability” Feb. 2014.

[3] A. Kiayias, T. Zacharias, B. Zhang. “End-to-end verifiable elections in the standard model”. In *Elisabeth Oswald and Marc Fischlin, editors, EUROCRYPT 2015, part II, volumen 9057 de LNCS*, pp. 468-498. Springer, Abril 2015.

[4] S. Popoveniuc, J. Kelsey, A. Regenscheid, P. Voral. “Performance requirements for end-to-end verifiable elections” in *EVT/WOTE* 2010.

[5] J. Benaloh. “Simple verifiable elections”. In *Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop*, pages 5–5. USENIX Association Berkeley, CA, USA, 2006.

[6] D. Chaum. “Secret-Ballot Receipts: True Voter-Verifiable Elections”. *IEEE Security and Privacy, vol 2, no. 1, pp: 38-47*, 2004.

[7] A. Fujioka, T. Okamoto, K. Ohta. “A practical secret voting scheme for large scale elections” *ASIACRYPT’92, Workshop on the Theory and Application of Cryptographic Techniques*. Pp. 244-251. LNCS 718, Gold Coast, Australia. 1992.

[8] S. Delaune, S. Kremer, M. Ryan. “Coercion-Resistance and Receipt- Freeness in Electronic Voting”. *CSFW’06: 19th Computer Security Foundations Workshop*, pp. 28-42. IEEE Computer Society, 2006.

[9] T. Okamoto. “Receipt-Free Electronic Voting Schemes for Large Scale Elections”. In *SP’97: 5th International Workshop on Security Protocols*, ser. LNCS, vol 1361 pp. 25-35. Springer 1998.

[10] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant electronic-elections”. *Cryptology ePrint Archive, Report 2002/165,* 2002.

[11] M. Backes, C. Hritcu, M. Maffei. “Automated Verification of Remote Electronic Voting Protocols in the Applied Pi-Calculus”. In *CSF’08: 21st Computer Security Foundations Symposium. IEEE Computer Society, 2008*, pp. 195-209. 2008.

[12] A. Juels, D. Catalano, M. Jakobsson. “Coercion - Resistant Electronic-Elections”. *RSA Laboratories, CNRS-Ecole Normale Supérieure, Indiana University, School of Informatics*. 2010.

[13] “The Future of Voting” U.S. Vote Foundation, 4325 Old Glebe Road, Arlington VA 22207, USA. 2015.

[14] H. Li, A.R. Kankanala, X. Zou. “Taxonomy and Comparison of Remote Voting Schemes”. *Purdue University*. 2014.

[15] D. Zissis, D. Lekkas.”Design, Development and Use of Secure Electronic Voting Systems”. IGI Global. ISBN 978-1-4666-5823-3. 2014.

[16] O. Kulyk, V. Teague, M. Volkamer. “Extending Helios Towards Private Elegibility Verifiability”. Vote ID 2015. 2015.

[17] Directorate General of Democracy and Political Affairs. “Certification of e-voting systems”. Council of Europe. 2011.

[18] D. Simic-Draws, S. Neumann, A. Kahlert, P. Richter, R. Grimm, M. Volkamer, A. Rossnagel. “Holistic and Law Compatible IT Security Evaluation: Integration of Common Criteria, ISO 27001/IT-Grundschutz and KORA”. *International Journal of Information Security and Privacy, 7(3), pp.: 16-35, July-September.* 2013.

[19] S. Neumann. “Evaluation and Improvement of Internet Voting Schemes Based on Legally-Founded Security Requirements”. *Ph. D Thesis. TU Darmstadt*. 2016.

[20] M. Volkamer, D. Hutter. “From Legal Principles to a Internet Voting System”. German Research Center for Artificial Intelligence, DFKI Saarbrücken. 2002.

[21] Common Criteria for Information Technology Security Evaluation. SO/IEC 15408:2009. https://www.commoncriteriaportal.org/files/ccfiles/CCPART1V3.1R3.pdf

[22] KORA (Konkretisierung Rechtlicher Inforderungen, Concretization of Legal Requirements). V. Hammer, U. Pordesch, A. Rossnagel. “Betriebliche Telefon- und ISDN-Anlagen rechtsgemäss gestaltet”. Spinger, 1993.

[23] ISO 27001 and German Federal Office for Information Security (BSI). “Zuordnungstabelle ISO 27001 sowie ISO 27002 und IT-Grundschutz”. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Hilfsmittel/Doku/Vergleich_ISO27001_GS.pdf?__blob=publicationFile 2013.

[24] K. Bräunlich, R. Grimm, P. Richter, A. Rossnagel. “Sichere Internetwahlen: Ein rechtswissenschaftlich-informatisches Modell”. Nomos, 2013.

Anexo C. Listado de Technical Design Goals de Bräunlich et al. [463].

Bräunlich et al. [BGRR13] han derivado los siguientes objetivos técnicos de diseño (TDG) como resultado de su investigación multidisciplinar, que sirvieron de base a la tesis doctoral de Neumann [458] entre otros:

TDG 1: *Unauthorized parties must not have the possibility to view voter data.*
TDG 2: *Unauthorized parties must not have the possibility to manipulate voter data.*
TDG 3: *Only data required shall be stored.*
TDG 4: *Any voter must have the possibility to view and influence both extent and purpose of her personal data.*
TDG 5: *The ballot must be neutral.*
TDG 6: *Unauthorized parties must not have the possibility to change the ballot data.*
TDG 7: *The election committee must start the election at the predetermined time.*
TDG 8: *After a system failure, it must be possible to resume the election.*
TDG 9: *The election committee must stop the election at the predetermined time.*
TDG 10: *The calculation of intermediate results must not be possible.*
TDG 11: *The calculation of the election result must start after the voting phase by members of the election committee.*
TDG 12: *Only eligible voters may access successfully the Internet Voting system.*
TDG 13: *Eligible voters may cast only one binding vote.*
TDG 14: *The essential steps of the vote casting process must be understandable to any voter.*
TDG 15: *Any voter must be able to conduct the vote casting process.*
TDG 16: *All voters must obtain the same result with equal usage.*
TDG 17: *Eligible voters must have the possibility to cast votes at any time of the voting phase.*
TDG 18: *The vote may only be cast and stored after a con_rmation by the voter.*
TDG 19: *It must be ensured that the vote is correctly transmitted.*
TDG 20: *Any voter must receive a message regarding the (non-)success of her voting process.*
TDG 21: *A voting note must only be taken after a binding vote has been cast.*
TDG 22: *Third parties must not be capable of linking a vote to the voter who cast the respective vote.*
TDG 23: *The voter must not be capable of proving her vote to any third party.*
TDG 24: *It must not be possible to manipulate the stored binding votes.*
TDG 25: *The system must compute the correct result.*
TDG 26: *It must not be possible to manipulate the election result.*
TDG 27: *Any voter must be able to verify that her vote has been included in the election result.*
TDG 28: *The public must be able to verify that the election result has been derived correctly.*
TDG 29: *The election must be protocolled.*
TDG 30: *The election data must be archived in a traceable and evidence-proven manner.*

Anexo D. Respuestas de los expertos nacionales e internacionales a la encuesta técnica sobre las ponderaciones de los criterios de la metodología de evaluación

	I-n	**U-n**	**MA-n**	**DSW-n**	**E-n**	**DESW-n**	**PIA-n**	**V-n**	**C-n**	**M-n**
Experto 1	1,2	0,7	1,2	1	0,6	1	1,1	0,6	0,9	0,7
Experto 2	1	0,6	1	1,2	0,9	1,2	1,2	0,7	0,9	0,6
Experto 3	1,1	0,7	1,1	1,1	0,8	1,2	1	0,6	1,1	0,7
Experto 4	1,2	1	1,2	1	1,1	1	1,1	0,8	0,8	1
Experto 5	1	0,8	1	1,2	0,7	1,2	1,1	0,6	1,1	0,6
Experto 6	1	0,6	1,1	1,2	0,8	1	0,9	0,6	1	0,8
Experto 7	1,2	1	1,2	0,9	0,6	0,9	1,1	0,8	0,6	0,7
Experto 8	1,2	0,7	1,1	1,2	0,8	1,1	1	0,6	0,9	0,9
Experto 9	1,1	0,6	1	1,2	0,6	1,2	1,2	0,6	1	1
Experto 10	1,2	0,8	1,1	1,1	0,7	1,2	1,2	0,6	1	0,7
Experto 11	1,2	0,9	1	1	0,8	1	1,1	0,7	0,6	0,6
Experto 12	1	0,7	1,1	1,2	0,7	1,2	1	0,6	0,9	0,7
Experto 13	1,2	0,8	1,1	1,2	0,8	1,1	1	0,6	0,8	0,8
Experto 14	1,2	1	1,2	1	0,9	1	1,2	0,8	0,6	0,9
Experto 15	1,1	0,6	1,1	1,1	0,6	1	1,2	0,7	1	0,6
Experto 16	1	0,6	1,2	1,1	0,6	1,1	1,2	0,6	1,1	0,8
Experto 17	1,2	0,6	1	1,2	0,7	1,2	1,2	0,6	1	0,7
Experto 18	1	0,8	1,2	1,1	0,6	1,2	1,2	0,6	1,1	0,7
Experto 19	1	0,8	1,2	1,2	0,8	1,2	1,1	0,6	0,9	0,7
Experto 20	1,2	0,7	1,1	1,1	0,7	1,1	1,2	0,6	0,9	0,8
Experto 21	1,2	0,6	1,2	1,2	0,8	1,2	1,2	0,6	1,1	0,7
MEDIA	**1,12**	**0,74**	**1,11**	**1,12**	**0,74**	**1,11**	**1,12**	**0,6**	**0,92**	**0,75**
MEDIA 10	**1,2**	**0,8**	**1,2**	**1,2**	**0,8**	**1,2**	**1,2**	**0,6**	**1**	**0,8**

Tabla d: Respuestas completas de los expertos a la encuesta sobre ponderaciones de los criterios de evaluación de la metodología