UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA

TESIS DE GRADO

"ARQUITECTURA DE SEGURIDAD PARA VOTO ELECTRONICO"

PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

Postulante: Diego Arthur Chindari Jimenez
Tutor: Lic. Efraín Silva Sanchez
Revisor: Lic. Juan Gonzalo Contreras Candia

LA PAZ – BOLIVIA
2009

DEDICATORIA

Este trabajo va dedicado a mis queridos padres Marcos y Remigia por darme la vida y creer siempre en mí.

A mis hermanos Franz, Blanca, Rossmery, Asunta y Kevin por su infinito amor, ternura y comprensión quien en momentos difíciles siempre estuvieron a mi lado brindándome su apoyo, consuelo y aliento.

AGRADECIMIENTOS

A Dios nuestro padre celestial por darme fuerzas a mirar siempre adelante.

A mi familia por haberles privado de muchos momentos juntos.

A Lic. Efrain Silva Sanchez por su acertada guía, consejos oportunos y su amistad en el desarrollo de la presente tesis como tutor.

Al Lic. Juan Contreras Candia por sus consejos, su paciencia, su tiempo, su admirable sencillez y su amistad en mi transcurso en la carrera.

A mis compañeros de carrera Hector, Alberto, Ruben y Edgar con los que pase muchos momentos alegres, divertidos y tristes.

A mis amigos del Centro de Estudiantes IPOD por ser un apoyo fundamental en mi vida y trabajar juntos por mí querida carrera.

A mis amigos de la Organización de Auxiliares de Docencia O.A.D. por compartir agradables momentos y trabajar por los auxiliares de la Universidad.

A mis compañeros de la Federación Universitaria Local F.U.L. por enseñarme a trabajar por los estudiantes y por nuestra querida Universidad, haciendo respetar el cogobierno y la autonomía universitaria.

Gracias...

RESUMEN

El presente trabajo muestra el diseño y el desarrollo de una arquitectura de seguridad para un sistema de voto electrónico presencial así como los resultados que se obtuvieron de su implementación. El uso de este tipo de sistemas ha ido en aumento y es necesario que generen altos niveles de confianza en los usuarios. El proyecto se enfoca en diseñar y construir una arquitectura que sirva como base para diseñar e implementar sistemas de voto electrónico seguros y auditables.

La contribución de este proyecto es la creación de una arquitectura que permita resolver el problema de la plataforma segura en los sistemas de voto electrónico, establecer una serie de requisitos que los sistemas deben cumplir para ser considerados seguros y auditables, proporcionar una manera de resolver los problemas más comunes de estos sistemas y determinar un conjunto de pruebas a las que se deben someter los sistemas para verificar su seguridad y generar un mayor nivel de confianza.

La utilización de protocolos criptográficos, así como programación en capas cuyo objetivo es separar la lógica de negocios como la lógica de diseño ya q se puede llevar a cabo en varios niveles, es así que en dicha tesis se añade una capa mas a la arquitectura 3 capas, un enfocada especialmente para la seguridad, mediante esto podemos controlar mejor y ser mas seguro nuestro sistema enfocado al voto electrónico, así también sirve como base para otros sistemas que necesitan seguridad en los datos de su sistema. La auditoria se enfoca en la creación de elementos auditables los cuáles permiten que los resultados de una elección puedan ser verificados en caso de que se sospeche de fraude y como consecuencia hacen que los usuarios tengan confianza en el sistema. La seguridad se refiere a la creación de protocolos criptográficos que cubran las distintas etapas de un sistema de votación electrónica y que garanticen que los resultados no puedan ser modificados o si lo han sido poder detectar esas modificaciones.

INDICE

LISTA DE FIGURAS

LISTA DE TABLAS

1. MARCO PRELIMINAR

1.1. INTRODUCCION

Los avances tecnológicos y de computación han influido notoriamente en la manera en que se realizan muchas de las actividades actuales substituyendo a las formas tradicionales. Durante años los sistemas de votación han sido llevados a cabo mediante métodos convencionales como las boletas de papel, llamadas telefónicas ó cartas y gran parte de los resultados se obtienen a través de un conteo manual. Con el paso del tiempo y el avance tecnológico surgieron los sistemas de voto electrónico, que comenzaron a ser utilizados en 1974 y eran conocidos como *DRE (Direct Recording Voting Systems)* o sistemas de almacenamiento directo del voto los cuáles originalmente eran referidos como "contadores electrónicos de votos".

La computación abre una puerta que puede simplificar en muchos sentidos la tarea de realizar elecciones, de ahí que la computadora esté sirviendo de base fundamental para la implementación de votación electrónica, la cual actualmente se divide en dos grandes grupos, presencial y remota. El voto electrónico es una realidad desde hace varios años en países como Francia, Alemania, Inglaterra, España, India, Brasil, Estados Unidos, Argentina y Venezuela entre otros, y se espera que dentro de poco este tipo de sistemas se aplique en Bolivia.

Sin embargo, dado el gran desconocimiento que existe sobre la tecnología con la que estos sistemas son construidos por una parte muy grande de la población, se introduce también una gran desconfianza en muchos sentidos.

Las características que estos sistemas deben cumplir para que los usuarios tengan confianza en ellos son: la opción registrada debe ser la misma que ellos quisieron emitir, los resultados finales deben mostrar realmente el sentir de todos aquellos que participaron en la elección, la opción que eligieron debe ser secreta, tener la seguridad de que los votos no pueden ser alterados y sobre todo deben ser sencillos de utilizar.

Para incrementar los niveles de confianza en éste tipo de sistemas, deben estar presentes desde su diseño conceptos como el de auditoria y auditabilidad, además deben contar con elementos de seguridad que los protejan no solo contra acciones impropias por parte de los votantes, sino que también de las que pudieran producirse por parte de los programadores, técnicos y administradores del sistema.

1.2. ANTECEDENTES

Desde que Internet se ha convertido en un fenómeno de masas, se viene estudiando en diversos países el efecto positivo que la red de redes podría ejercer en las relaciones Administración-ciudadano, facilitando la interacción entre ambos. En particular, el uso de las nuevas tecnologías conlleva grandes promesas en lo que se refiere a la mejora y modernización de los actuales procesos electorales y de participación tradicionales, abriendo también las puertas al desarrollo de portales de opinión y de sondeo.

Para poner en práctica los conceptos de la Democracia Electrónica, y para iniciar una necesaria experimentación de las diversas tecnologías de consulta electrónica, últimamente diversos países europeos han organizado varias pruebas piloto de voto electrónico donde solicitan a sus ciudadanos que opinen sobre diversos temas o que elijan a ciertos representantes. La correcta

estrategia de implantación de tecnología en los procesos electorales y de consulta implica una experimentación gradual. Esta estrategia permite depurar los sistemas tecnológicos y los procedimientos utilizados, al obtener datos fiables de pruebas reales con ciudadanos reales.

Por ejemplo, el Reino Unido ha realizado pilotos de voto electrónico durante las elecciones locales de 2002 y 2003, y tiene previsto hacer más cada año hasta las elecciones de 2008, cuando el uso de nuevas tecnologías para la emisión de votos vinculantes en las elecciones generales estará ya plenamente habilitado.

Experiencias Internacionales de voto electrónico

India

Las razones principales de la implementación tecnológica fueron dos:

- Reducir los altos índices de fraude electoral.
- Reducir los altos índices de violencia creados durante las votaciones tradicionales.

El sistema actualmente implementado consta de urnas electrónicas muy simples, con el software impreso sobre el chip central y no poseen electrónica que les permita conectarlas a alguna red, de manera que el hackeo es prácticamente imposible. No imprime un comprobante. La forma de selección en un tablet con botones especiales.

El proceso de recuento es básico ya que no se realiza en los lugares de elección sino que las urnas deben ser físicamente trasladadas hasta alguno de

los centros oficiales de consolidación (en los que participan veedores de los diversos partidos políticos). No se emite un acta de cierre y el sistema de recuento es automático.
Como las elecciones en muchas regiones del país suelen ser interrumpidas por ataques violentos de patotas partidarias, las urnas cuentan con un botón que puede ser operado en una emergencia por los funcionarios a cargo. Este botón "bloquea" e inutiliza la urna en caso de robo o vandalismo.[Misra, 2004]

Brasil

Brasil es posiblemente uno de los países más avanzados en la implantación de sistemas de voto electrónico. Las principales motivaciones que indujeron su implementación fueron las siguientes:

- Eliminar el fraude electoral.
- Reducir el tiempo de escrutinio.
- Facilitar el ejercicio de voto por parte de los analfabetos

El sistema que se ha estado utilizando es el de urna electrónica con teclado numérico para la emisión del voto. Tiene botones especiales de confirmación e impresión de acta inicial con activación por clave. El registro de los votos se realiza directamente en la memoria de la máquina de votación y en un diskette que luego será trasladado hasta la sede de la autoridad electoral. No se emite ningún comprobante físico de sufragio. El cierre se realiza mediante clave y se emite un acta. La transferencia para el recuento se realiza en forma tradicional con encriptación de datos y firma digital. Algunas máquinas poseen un modem interno para la transmisión de los datos. El escrutinio se efectúa por sumarización automática. La característica más destacable del sistema brasilero reside en que permite unificar el registro y verificación de la identidad del elector, la emisión y el escrutinio de voto en una misma máquina.

En los últimos comicios se obtuvieron resultados absolutamente confiables, y se notó una disminución del 50% de los votos nulos, con un aumento de la asistencia electoral y sin ninguna impugnación consistente.

En concreto y con un espíritu estadístico es importante señalar que utilizaron el voto electrónico más de 115 millones de personas. [TSE, 2003]

Bélgica

Bélgica decidió adoptar un sistema de votación electrónica para resolver los problemas ocasionados por la complejidad de su sistema electoral. Poseen de 1 a 5 elecciones simultáneas con listas de hasta 87 candidatos cada una y por elección, lo que deriva en totalizaciones de voto complejas y propensas a errores.
Utilizaron un sistema DRE, que se explica en la sección Tecnología existente de este anexo, con monitor touchscreen (sensible al tacto) y tarjeta magnética para habilitar la máquina y registrar el voto. Los ciudadanos pueden emitir su voto tocando sobre la pantalla con un lápiz óptico. El sufragio queda almacenado en la tarjeta que luego es leída por otro equipo electrónico que se encuentra separado, al estilo de urna. El sistema puede ser auditado mediante una comprobación efectuada sobre la tarjeta magnética.

Luego de que el elector se identifica, recibe la tarjeta magnética que le permite pasar al cuarto oscuro a utilizar la máquina de votación. Finalizada la elección, se efectúa el cierre de las urnas y se deben enviar los diskettes que poseen los totales de cada urna a un centro de cómputos donde se suman para obtener el resultado de los comicios.

Las tarjetas permanecen dentro de las urnas hasta que se conoce el resultado de la elección y sólo pueden ser retiradas para el proceso de auditoría. [SPF, 2004]

Costa Rica

Al igual que varios de los países que implementaron la votación electrónica, Costa Rica decidió modernizar su proceso electoral porque quería principalmente automatizar la totalización de resultados para agilizar su obtención.

Se realizó una prueba piloto, y las máquinas donde se emitían los votos consistían en PCs tradicionales, con teclado, monitor e impresora, ubicadas dentro de un contenedor. El programa de votación se cargaba mediante un CD.

Para poder seleccionar un candidato, la máquina despliega un listado con las opciones electorales. Mediante el teclado, el votante debe digitar el número correspondiente a la boleta que refleja su intención de voto. Luego de realizar una confirmación de la selección, el elector debe introducir en la impresora la boleta comprobante que le fue entregada por el presidente de mesa al ser identificado. En caso de ser necesario, para realizar esta operación, puede solicitar ayuda de un auxiliar técnico. Una vez que la impresora devuelve la boleta, se la debe doblar e introducir en una urna. Esta boleta posee una marca identificatoria y la firma de la autoridad competente.

Las máquinas poseen módems para enviar los totales calculados a la computadora central para consolidar los resultados de los comicios. [TSECR, 2002]

Venezuela

En las elecciones del año 2004, se utilizaron máquinas con pantalla touchscreen que imprimen un voto físico en un papel térmico, lo que permitiría auditar el proceso de votación. El elector selecciona, tocando la pantalla, la opción que desea. Para confirmar su sufragio oprime la opción "Votar". Al realizar esta acción la máquina emite un sonido indicando que el votante finalizó su proceso e imprime el voto en papel. Si luego de transcurrido un tiempo prudencial, la máquina no recibió la orden de sufragar, se desactiva automáticamente y emite un recibo indicando que el tiempo expiró.

El ticket impreso posee un código seguridad además de los principales datos del evento: tipo de elecciones, código que corresponde al centro de votación, mesa y tomo. El código de seguridad es esencial para evitar la falsificación del voto. Esta impresión es colocada por el elector en una urna. Una vez finalizado el día electoral el presidente de mesa cierra la misma y la maquina imprimirá el acta de dicha mesa.

Luego de esta impresión, la información final acumulada por cada máquina se transmite vía telefónica -en algunos casos satelital- en forma encriptada con clave pública y privada de 128 bits al centro de consolidación de datos. [Bracci, 2004]

Estados Unidos

Estados Unidos es una de las naciones con mayor experiencia en la regulación de las votaciones electrónicas y en la elaboración de la reglamentación apropiada para su efectiva implementación.

Se creó una Comisión Electoral Federal que establece el marco general que deben seguir los diferentes Estados en la elección de los instrumentos de votación electrónica.

Un detalle importante a considerar es que algunas empresas que disponen de sistemas de votación electrónicos se vieron sometidas a una serie de pruebas, descubriendo que el código fuente era vulnerable y que podían sufrir ataques de intrusión críticos. Se identificaron un total de 57 potenciales agujeros de seguridad en una supuesta votación electrónica, algunos tan graves como la posibilidad de que una persona no autorizada acceda a la base de datos donde se almacenan los resultados y cambiarlos. [Hernández,]

Inglaterra

En el año 2003 el gobierno británico hizo una apuesta al voto electrónico. Para llevar adelante los comicios se permitió utilizar Internet, televisión interactiva y SMS. Los resultados de estas experiencias mostraron que cuanto más fácil es la utilización de los sistemas implementados, los ciudadanos participan más a gusto y en mayor proporción en las votaciones; aunque persisten las preocupaciones vinculadas a cuestiones tales como la seguridad y vulnerabilidad del sistema.

España

Se organizaron una serie de experiencias piloto de voto electrónico. Éstas, planteadas como un acercamiento a la e-democracia, no tenían validez legal, si bien el procedimiento se articuló tal como se haría en elecciones reales. Así, la apertura y cierre de urnas se realizó en presencia de "juntas electorales

virtuales" formadas para tal efecto, cuyos miembros poseían claves que, sólo si estaban combinadas, permitían abrir o cerrar la votación.

Cada participante elegía una boleta que poseía preimpreso el candidato electoral y una banda magnética para el registro, recuento y totalización de votos. El sistema contó con una urna electrónica que poseía dos ranuras, una para validar el voto y otro para depositarlo. Ésta última era la que lo registró y lo contabilizó. Para cerrar la votación se volvían a combinar las claves y se emitía un acta. Luego mediante transporte tradicional o vía telefónica, se enviaban los datos a una central y se sumaban en forma automática. [DE, 2004]

Experiencias en Bolivia

En nuestro país todavía no se optaron por la implementación de esta tecnología, hasta el momento se continúa con la votación convencional, no se presentaron propuestas sobre la implementación de esta tecnología, pero se espera a futuro optar por el voto electrónico.

En todas ellas se persigue la consolidación de herramientas fiables, flexibles, seguras y eficaces de voto electrónico y participación ciudadana que entre otros beneficios reduzcan el abstencionismo electoral y aproximen el segmento de población joven a la política.

1.3. EL PROBLEMA DE LA INVESTIGACION

1.3.1. Planteamiento del Problema

Inicialmente se exponen los inconvenientes asociados a la aplicación de las tecnologías al acto de sufragio desde dos puntos de vista: el social y el tecnológico. Luego se describen algunos requisitos esenciales, que no son más que características innatas de los procesos de votación electrónica, con la finalidad de describir los problemas que trae aparejada su implementación. Otra importante fuente de problemas son las máquinas utilizadas para la emisión de los sufragios. Todo artefacto electrónico posee debilidades que pueden ser explotadas por terceros o límites acerca de las funcionalidades que ofrece. Aquí se puntualizan las flaquezas que poseen las máquinas.

Finalmente, se cierra citando diferentes documentos que avalan la problemática descripta acerca de los requisitos de la votación electrónica.

1.3.1.1. Inconvenientes del voto electrónico

1.3.1.1.1. Problemas sociales

Los problemas sociales son aquellos que surgen de las pautas culturales de cada país.

Existen varios inconvenientes que se detallan a continuación:

Secreto del voto

En Bolivia, todo lo relacionado con el secreto de la elección parece algo trascendente e intocable. El cuarto oscuro existe sólo en nuestro país. Por esto, cualquier implementación de la tecnología de voto electrónico debería considerar la existencia de este cuarto si pretende ser exitosa.

Solemnidad del acto

En nuestro país el episodio de votar es algo altamente solemne. Parte de esta concepción radica en que es obligatorio.

Elementos que rodean al sufragio

Los padrones, los sellos, los documentos, las boletas, etc. hacen a la solemnidad del acto.

1.3.1.1.2. Problemas tecnológicos

Los problemas tecnológicos son aquellos que se desprenden de las herramientas utilizadas para la concreción del sufrago. Se encuentran intrínsecos en las mismas.

Incertidumbre

Al administrarse una materia prima tan delicada y masiva como son los votos, todo sistema de administración genera un margen de error que en ciertas condiciones de competencia política torna casi indefinible la elección en base a los resultados procesados. Por supuesto, probablemente en ese punto, contar con medios electrónicos en parte de los procedimientos, permitiría bajar las tasas

de incertidumbre. Pero es imposible que éstos no presenten algunos problemas técnicos.

Verificabilidad

Un problema tecnológico a resolver es aquel relacionado con la existencia de mecanismos por los cuales se le permita al votante comprobar que su voto no ha sido modificado durante el proceso. Este es el objetivo de esta tesis como se detallará más adelante.

Posibilidad de fraude

En los comicios tradicionales, la existencia de fraude no sólo deja pistas concretas, sino que además involucra a mucha gente. Esto no ocurre con el voto electrónico, donde un grupo reducido de personas puede ser el causante de un fraude a gran escala sin dejar mayores huellas en los sistemas.

Intangibilidad de los procesos

Quienes observan las máquinas no pueden ver la forma en que éstas almacenan, manejan y cuentan los votos. Estos procesos son transparentes para cualquier persona que no haya estado en el equipo de desarrollo del software. A su vez, los votantes tampoco pueden observar por sí mismos el registro electrónico de su boleta. Semejante transparencia en la forma de operar de las máquinas produce desconfianza en los sistemas de voto electrónico.

1.3.1.2. Facilidad, velocidad, corrección y confianza

Estos conceptos son la base de interminables discusiones a cerca de la aplicabilidad de la tecnología a los sistemas electorales. Cada uno se

encuentra relacionado con el otro de una u otra manera. Los cuestionamientos que se realizan sobre las metodologías de votación son a causa de la falta de confianza del votante hacia el proceso de elección. Esa confianza se logra con un sistema que sea fácil, ágil y correcto. A continuación se analiza cada una de las características y se describe cómo se consigue que la gente crea en el método.

Facilidad

En cualquier democracia, todas las personas tienen el derecho de emitir un voto. Es por esto, que los sistemas de votación deben garantizar que cualquier individuo sea capaz votar, razón por la cual el método de sufragio debe ser entendible por cualquier persona. Si el objetivo es reemplazar los sistemas tradicionales de votación por máquinas electrónicas, el proceso a implementar debe ser lo más sencillo posible para que la gente no requiera de ayuda para emitir su voto. La necesidad de asistencia por parte de un tercero puede provocar rechazo en quien desee conservar su elección política en secreto absoluto. Si bien la tecnología podría contribuir a que personas discapacitadas, ciegos en particular, puedan votar sin soporte de nadie, también podría frustrar el intento de votación de gente mayor que no entiende cómo operar máquinas electrónicas.

Como suele ocurrir, lo que se gana por un lado se pierde por el otro. La manera de que gente reacia hacia los cambios o hacia la tecnología se logre adaptar a los sistemas de voto electrónico, es logrando que los mismos sean simples y sencillos. La facilidad de uso otorga mayor confianza hacia el usuario ya que la persona se siente capaz de emitir su voto sin ayuda. Esa sensación de competencia provoca aceptación y por tanto logra confianza en el sistema. Mediante la sencillez también se

consigue la corrección, ya que hay una mayor posibilidad de que el voto almacenado refleje la intención del elector. La base de la sencillez del proceso, radica en un alto porcentaje en las interfaces utilizadas. Cuando éstas son complejas dan lugar a confusiones y por tanto inducen a los sufragantes a cometer errores y votar por quienes no deseaban.

La persona que se presenta a votar, interactúa con la máquina a través de la interfaz, comúnmente usando un touch-screen. El objetivo de minimizar el error de votación puede incrementar la complejidad del software de la máquina. Por ejemplo, las máquinas DRE solicitan confirmación cuando el sufragante no selecciona ninguna opción (voto en blanco), pero para ello, requieren de un paso adicional para el votante. Además ocurre que cuando la interfaz no es simple y clara, la cantidad de usuarios que requieren asistencia es mayor. La cantidad de asistencia requerida juega un rol importante en la confianza en el proceso de votación porque la ayuda siempre viene de un trabajador partidario. Además los usuarios que solicitan ayuda arriesgan su anonimato y quienes la requieren podrían rehusarse a pedirla por ese motivo o por otros inconvenientes personales.

En definitiva se puede asegurar que la facilidad de uso y la simpleza del software de las máquinas de votación hacen a la corrección y al incremento de la confianza en el sistema.

Analizando un poco más profundamente, se puede ver que la corrección se refiere a que el voto almacenado refleje la intención del votante. En cambio, la confianza, radica en mantener en secreto la elección y que ningún paso del proceso de selección del candidato pueda originar una

duda en el elector acerca de si está operando bien la máquina o si cometió un error que podría modificar su voto.

Velocidad

Si bien la implementación de un sistema de voto electrónico podría perseguir objetivos tales como reducir los altos costos electorales causados por la emisión de boletas, el propósito principal es incrementar la velocidad con que se obtienen los resultados de los comicios. Cuando se decide implementar un sistema de voto electrónico, por lo general, se busca rapidez en el proceso de recuento de los votos.

La ventaja de los sistemas electrónicos, consiste en que cada máquina contabiliza los votos en pequeñas fracciones de tiempo y la única demora en la obtención del resultado es el envío de la información de cada mesa de votación a un centro de consolidación que se encargue de totalizar las sumas parciales de cada máquina. No importa cual fuere el medio elegido para transportar la información al centro de consolidación, los tiempos de recuento son varias veces inferiores a los manuales. Los ciudadanos asocian la tecnología con la velocidad. Si se implementa un sistema de voto electrónico y el tiempo en que arroja los resultados es el mismo que cuando se realizaba la votación por medio del sistema tradicional, enseguida las dudas sobre la conveniencia del cambio y la justificación de la inversión invadirían a la población.

Varios de los sistemas expuestos imprimen un comprobante en papel que es depositado en una urna de la máquina para que el elector compruebe que su voto refleja su intención. Este mecanismo se implementó como solución a la desconfianza generada por los sistemas

de registro directo, donde no existe prueba alguna de los votos emitidos y un error en las máquinas de votación puede cambiar los resultados de las elecciones. Sin embargo, se perdió de vista una de las características requeridas por los sistemas de votación electrónica: velocidad. Si bien la emisión de comprobantes en papel es únicamente para auditar el proceso, es inevitable efectuar el recuento manual, ya que el resultado verdadero de los comicios es aquel que éste arroja debido a que las boletas emitidas son las que reflejan la verdadera intención del elector. Si los totales arrojados por la máquina, difieren de los obtenidos mediante la suma de las boletas impresas, serían estas últimas las que tendrían valor. Por lo tanto, el hecho de implementar sistemas de votación electrónica, no exime de la necesidad de contar manualmente los votos, obteniendo los resultados oficiales en el mismo tiempo que los sistemas tradicionales.

La falta de velocidad en alcanzar las sumas de votos de cada partido, también genera desconfianza en la población. Se plantean interrogantes acerca de la eficiencia y corrección de las máquinas debido a la demora en la publicación de los resultados. Estas dudas derivan en otras que fueron mencionadas en párrafos anteriores, como ser si era necesario el cambio y si se justificaba la inversión. Todas estas incertidumbres llevan al fracaso la implementación de las tecnologías en los sistemas de votación.

Corrección

El requerimiento fundamental de cualquier sistema computacional es que sea capaz de guardar los datos que se le ingresan, para que estén disponibles cuando se los necesite, es decir, que sirva como soporte de información. Esta característica no es ajena a los sistemas de votación

electrónica. La corrección de los mismos radica en que el voto almacenado sea el verdadero reflejo de la intención del sufragante. Dada una interfaz en la cual les permita a los votantes ingresar su voto sin error, la confianza depende directamente del correcto almacenaje de las intenciones de voto de los sufragantes. La confianza de los ciudadanos en las máquinas depende de sus experiencias en el uso de las mismas tanto como en el entendimiento de su funcionamiento y el proceso que las rodea.

En cualquier sistema tradicional, los operadores pueden consultar los datos cargados y verificar su integridad. Sin embargo, en los sistemas de votación electrónica esa posibilidad no existe, motivo por el cual la confianza juega un rol muy importante. Una de las razones por las cuales los votantes confían en las máquinas DRE es debido a su semejanza con los cajeros automáticos.

Después de todo, si confiamos en una máquina electrónica que cuente dinero podremos confiar en que cuente correctamente los votos emitidos. La falacia de este argumento reside en que el cajero automático le otorga al usuario un comprobante de la operación. Si existiera una discrepancia el cliente puede notarlo y hacer el respectivo reclamo al banco interviniente. En las máquinas DRE no hay comprobante y tampoco hay forma de protesta posterior sobre los resultados almacenados.

En un principio, el problema parecería acotarse a la existencia de una vía por la cual el elector pueda verificar que su voto seleccionado es correcto. No obstante, el inconveniente es aún mayor. Las máquinas de votación efectúan varias transformaciones sobre los votos almacenados,

por lo que permitir comprobar la elección en una instancia del proceso, no implica que en otra etapa el sufragio pueda ser modificado.

Ésta es una de las razones por la cual varios de los sistemas no tuvieron éxito. La mayoría permite al elector constatar su elección antes de salir del cuarto oscuro o cabina, pero nunca después de abandonado el recinto de votación.

Un mecanismo más poderoso es aquel que permite al votante saber si su voto fue incluido correctamente en el recuento. Esto debe realizarse sin comprometer el anonimato del sufragante o sin proveer la posibilidad de que alguien infiera por quién votó otra persona.

Una forma simple pero insatisfactoria de hacer eso es proveer al votante de una única y aleatoria boleta numerada y luego publicar cada número de boleta con el voto asociado. Los sufragantes van a poder comprobar si en la publicación se encuentra su voto y si además fue registrado correctamente.

El problema reside en que nadie podría ser capaz de probarle a ninguna otra persona que su voto no fue incluido en la cuenta. Otro problema es que los votantes podrían ser forzados a revelar su voto, diciéndole a quien los obliga el número de boleta. Finalmente, no habría ninguna forma de probar que se hayan agregado votos de gente que no ha concurrido a votar.

Por estas razones se le debe otorgar al sufragante un comprobante que les permita verificar que su voto fue realmente incluido correctamente en el recuento o de lo contrario servir como prueba para que un fiscal

electoral pueda comprobar el error. Este comprobante no debe permitir de ninguna forma que un coerzor pueda conocer por quién votó una persona.

Implementando un sistema de votación electrónica que cumpla con la característica antes mencionada, su éxito dejaría de depender de la confianza de los ciudadanos en la fidelidad del proceso.

Garantizaría la corrección del sistema mediante la distribución a todos los votantes de la responsabilidad de verificar la exactitud del almacenado de votos. La confianza se incrementaría a medida que las personas comprueben que sus votos fueron contados conforme a sus deseos y que el secreto no fue violado. Este último punto es muy importante, ya que reduce los procedimientos que puedan implementarse para verificar los votos.

Confianza

En los puntos anteriores se han analizado 3 de las características de los sistemas de votación que se creen vitales para el éxito en la implementación de un sistema de voto electrónico.

La conjugación de la facilidad, velocidad y corrección en un sistema, generan en la población la confianza necesaria para que persista en el tiempo.

Sin embargo, las primeras dos cualidades pueden ser sacrificadas en cierto grado, aunque nunca eliminadas, con el único fin de garantizar la corrección en el almacenado y recuento de votos. Cualquier duda sobre

este aspecto, pondría en tela de juicio al procedimiento y terminaría por erradicarlo por completo.

De nada sirve tener el sistema de selección más simple, con la interfaz más amigable para el usuario, si la operación que realiza es incorrecta.

Lo mismo ocurre con la velocidad, el proceso más ágil no tiene valor si es erróneo. Pero también se puede apreciar que la corrección sin la velocidad o facilidad, tampoco subsistiría porque la población demandaría retornar a los sistemas de votación tradicionales ya que los tecnológicos no proveerían ninguna ventaja.

La forma de incrementar la confianza de los electores en los métodos de votación electrónica es garantizando la corrección de almacenado y recuento de sufragios, que la velocidad de acceso a los resultados sea considerablemente mayor a los sistemas tradicionales y que la complejidad de emitir un voto no sea considerable.

Evitando el doble registro de votos, manual y electrónico, y siendo las máquinas quienes efectúen el recuento, la velocidad está garantizada.
El problema radica en balancear la facilidad con el agregado de procesos que aseguren la corrección de los sistemas pero dificultan el uso de las máquinas de votación. La dificultad se acota a resolver hasta qué nivel se puede complicar la selección de los candidatos para implementar un sistema de verificación del voto, único recurso disponible para legitimar a los sistemas de votación electronicos.

1.3.2. Formulación del Problema

Para la elaboración de la formulación del problema se tomará como parámetro el punto anterior, la misma se presenta como una pregunta a continuación:

"Los ataques y vulnerabilidad a Sistemas de Registro o Grabación Electrónica Directa, no permiten proporcionar una adecuada arquitectura de servicio de seguridad a el proceso de votación, para su aplicación en sistemas de Voto Electrónico"

1.4. OBJETIVOS

1.4.1. Objetivo General

Implementar una Arquitectura de Seguridad que permita construir Sistemas de Voto Electrónico seguro, confiable y auditable, protegiendo la integridad de los datos en el proceso de votación evitando el fraude electoral.

1.4.2. Objetivos Específicos

Diseñar una Arquitectura de Seguridad para el sistema de voto electrónico, determinando su seguridad, confiabilidad y eficiencia mediante el proceso de pruebas paralelas

Determinar los ataques y las vulnerabilidades de los sistemas de voto electrónico presencial desde el punto de vista de la seguridad y la auditoria

Desarrollar una metodología para resolver de manera eficiente estas vulnerabilidades que afecta los sistemas de voto electrónico

Implementar y utilizar algoritmos avanzados de encriptación, desencriptacion y cifrado de los datos

Utilizar métodos biométricos para la autentificación de los votantes.

Diseñar un protocolo criptográfico óptimo para la arquitectura de seguridad de voto electrónico.

Aumentar la rapidez en cuanto a generación de resultados finales del proceso electoral.

1.5. LIMITES Y ALCANCES

Un sistema de voto electrónico es bastante complejo, éstos sistemas se encuentran formados por tres etapas principales, pre votación, votación y post votación.

La etapa de pre votación es la encargada de la generación de los archivos que utilizará el sistema para configurarse de acuerdo al número de elecciones, candidatos que participan en cada una y datos específicos sobre el lugar en donde estará ubicado el equipo.

En la etapa de votación se efectúa el proceso de recibir los votos y de generar resultados parciales que corresponden solamente al lugar en donde se encuentra ubicado el equipo.

Finalmente la etapa de post votación es la encargada de la recolección de los resultados parciales de los distintos equipos, de su análisis estadístico y de la obtención de los resultados finales.

Este proyecto se enfoca en la etapa de votación que a su vez puede dividirse en las tres mismas etapas, pero con la diferencia de que aquí la etapa de pre votación se refiere a la instalación de archivos y configuración del sistema.

La etapa de votación cuya entrada es la elección del votante, y su salida es su elección final considerando que se puede corregir el voto antes de confirmarlo, una vez que el votante ha completado el proceso de selección, no se permitirá gracias a la lógica del sistema poder realizar selecciones adicionales.

Finalmente la etapa de post votación cuya función es realizar la suma de los votos individuales, reportar el total de votos que obtuvo cada candidato y el número de votos nulos que se hayan obtenido cuando el votante haya elegido no votar por ningún candidato, cada una de las etapas cuenta con sus respectivos elementos de seguridad.

Un punto que no se cubre es el que se refiere a la forma de activación del equipo, haciendo solo notar que debe asegurarse que un usuario no pueda votar más de una ocasión. La figura 1 muestra los alcances del proyecto y sus relaciones con el resto de las etapas.

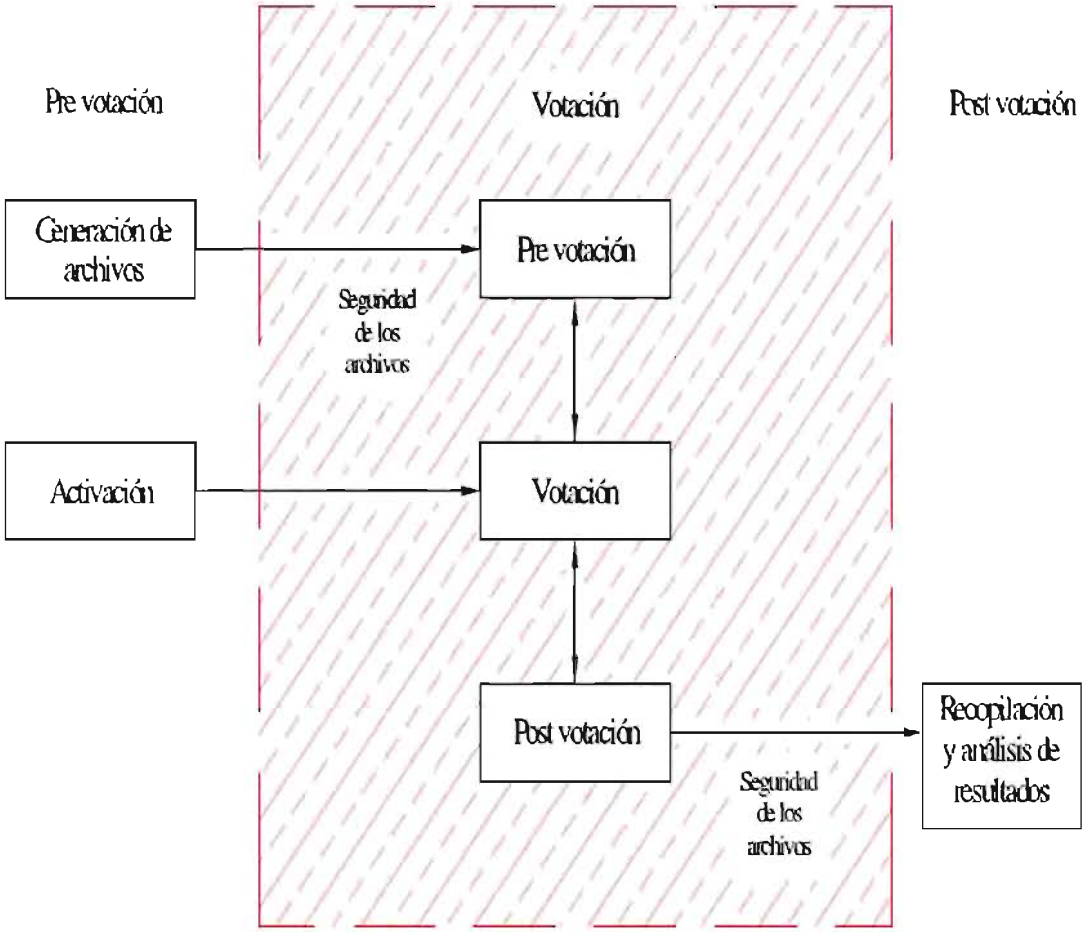


Figura 1. Alcances del proyecto.

1.6. FORMULACION DE LA HIPOTESIS

¿Es posible proteger la integridad de los datos contra posibles ataques al Registro Electrónico Directo mediante una arquitectura de seguridad apoyada en técnicas criptográficas, que proporcione al proceso de votación un adecuado servicio de seguridad para su aplicación en sistemas de voto electrónico?

1.7. JUSTIFICACION

1.7.1. Justificación Técnica

Uno de los principales problemas en los sistemas informáticos es el de la "plataforma segura" que es cuando no se cuenta con una arquitectura bien definida para diseñar un sistema seguro, éste problema también afecta a los sistemas de voto electrónico los cuales deben contar además con una arquitectura de auditoria.

Actualmente no se cuenta con una arquitectura bien definida que ayude a diseñar de manera eficiente estos sistemas, lo que se tiene es una gran cantidad de opiniones y recomendaciones acerca de sus puntos críticos y lo que se hace es ir agregando defensas una vez que el sistema ha sido vulnerado, cuando lo correcto es que éste sea seguro y confiable desde el momento de su creación.

La importancia de éste proyecto radica en identificar los aspectos vulnerables de los sistemas de voto electrónico desde el punto de vista de la seguridad y la auditoría, y de acuerdo con ello diseñar una arquitectura que permita resolverlos y construir sistemas de voto electrónico seguros y confiables.

1.7.2. Justificación Metodológica

Uno de los principales problemas en los sistemas informáticos es el de la "plataforma segura" que es cuando no se cuenta con una arquitectura bien definida para diseñar un sistema seguro, éste problema también afecta a

los sistemas de voto electrónico los cuales deben contar además con una arquitectura de auditoría.

Actualmente no se cuenta con una arquitectura bien definida que ayude a diseñar de manera eficiente estos sistemas, lo que se tiene es una gran cantidad de opiniones y recomendaciones acerca de sus puntos críticos y lo que se hace es ir agregando defensas una vez que el sistema ha sido vulnerado, cuando lo correcto es que éste sea seguro y confiable desde el momento de su creación.

La importancia de éste proyecto radica en identificar los aspectos vulnerables de los sistemas de voto electrónico desde el punto de vista de la seguridad y la auditoria, y de acuerdo con ello diseñar una arquitectura que permita resolverlos y construir sistemas de voto electrónico seguros y confiables.

2. MARCO TEORICO

2.1. VOTO ELECTRONICO

2.1.1. Definiciones del Voto Electrónico.

El voto electrónico puede definirse como [Prince, 2004]:

"Aplicación de dispositivos y sistemas de tecnología de la información y telecomunicaciones al acto del sufragio total o parcialmente, a todo el proceso electoral, o a algunas de las distintas actividades del sufragio, el registro y verificación de la identidad del elector. Incluye la emisión misma del voto en una urna electrónica (con o sin impresión inmediata de boleta en papel para control del ciudadano o de la autoridad); el recuento en la mesa o el global consolidado, la transmisión de resultados, u otras actividades".

Pero el voto electrónico no es simplemente un cambio de herramientas y materiales, no significa pasar de la urna de madera o de cartón al metal y al software, es mucho más por que las posibilidades que el nuevo sistema ofrece permiten rediseñar –corrigiendo- el sistema electoral completo [Prince, 2004].

2.1.2. Requisitos del Voto Electrónico.

El voto electrónico al igual que el tradicional debe cumplir con una serie de requisitos, los dos más importantes son [Prince, 2004]:

- La confianza del elector en el buen funcionamiento del sistema.

- La facilidad, comodidad y sencillez que presente el sistema de emisión del voto electrónico.

Otros requisitos que un sistema de voto electrónico debe poseer son:

- **Anónimo y privado.** Debe garantizar el anonimato y la privacidad al momento de emitir un voto. Los usuarios deben poder votar en total libertad y privacidad sin que su identidad pueda ser relacionada con su voto.

- **Elegible y auténtico.** Solo puedan votar los usuarios autorizados, también garantizando que puedan hacerlo sólo una vez.

- **Íntegro.** Los votos no sean cambiados o eliminados.

- **Exacto y verificable.** Procurar el correcto almacenamiento de los votos y de toda la información que registren, y todo el proceso deberá ser verificable.

- **Confiable.** Debe funcionar de manera robusta, sin pérdida de votos ni de datos o información. Cabe destacar que en el voto electrónico la confiabilidad se basa fundamentalmente en una cuestión de percepción por parte de los electores y no tanto en una razón técnica.

- **Fácil de utilizar.** Debe ser fácilmente utilizable por los electores para que no genere confusiones en el elector ni en las autoridades encargadas del escrutinio.

- **No coaccionable.** Los votantes no pueden demostrar a otros por quién votaron.

- **Verificable por el individuo.** El votante debe poder comprobar su voto. Un sistema de voto electrónico debe cumplir con los siguientes requisitos [Selker, 2003]:

  a) Asegurar el almacenamiento del voto emitido, esto se logra con una arquitectura que logre la detección y la corrección de errores.

  b) Prevención de alteraciones externas, especialmente las que involucren la modificación de los votos.

  c) Prevención de alteraciones internas que incluyen el desarrollo malicioso de sistemas de voto electrónico.

  d) Permitir la detección de falsificación de los contenidos de los archivos que el sistema utiliza o genera.

  e) Para [Saltman, 2003] los sistemas de voto electrónico deben cubrir tres aspectos fundamentales:

     - Debe ser sencillo para el votante emitir su voto.
     - El sistema debe procesar correctamente la opción elegida.

- Debe existir confianza del público en los resultados que arrojará el sistema.

2.1.3. Tipos de Voto Electrónico.

La votación electrónica puede ser dividida en dos grandes categorías la remota y la presencial como se muestra en la Figura 2.

La votación remota se puede realizar a través de Internet mediante una PC, teléfono celular u otro dispositivo desde cualquier locación geográfica cercana o lejana al lugar donde se realizan las elecciones. La votación presencial, por su parte, implica el uso de sistemas de captación electrónica del voto, con transmisión y escrutinio provisional a través de una "urna electrónica" ubicada en los lugares físicos donde se realiza la votación tradicional [Prince, 2004].

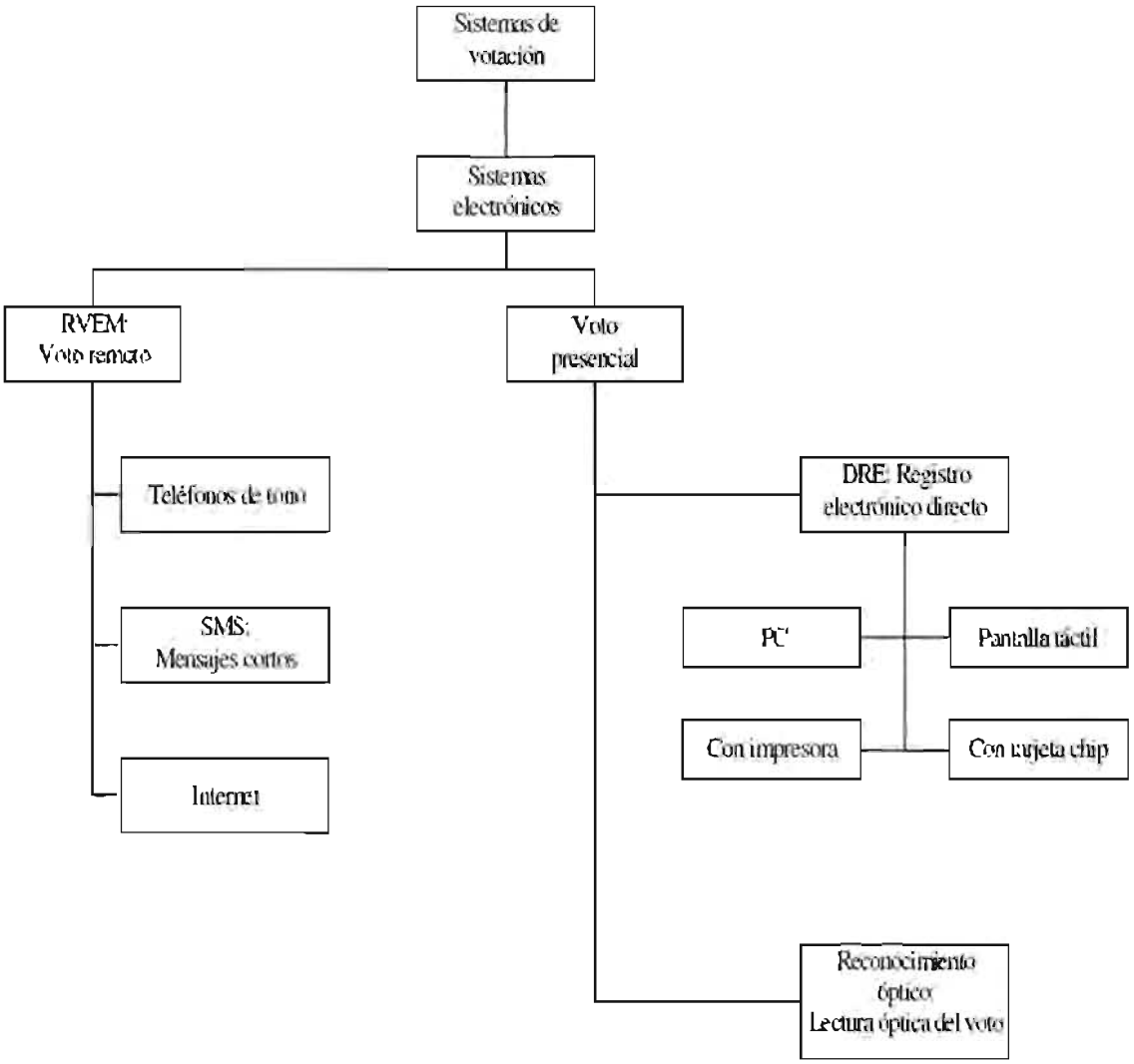


Figura 2. Tipos de voto electrónico.

Votación presencial.

La votación presencial es la que se utiliza principalmente y se pueden identificar dos grandes grupos:

- Registro Electrónico Directo (*DRE* por sus siglas en inglés).
- Lectura Óptica del Voto (*LOV*).

Estos dos se distinguen por la forma en que se emite un voto, ya sea de manera electrónica (*DRE*) o manual (*LOV*) y también por la forma en que se almacena el voto, ya sea directamente en una memoria o por digitalización óptica, pero ambos comparten una característica en común, automatizar el conteo de los sufragios y la obtención de resultados preliminares de manera casi inmediata, existen pequeñas variaciones entre ambas tecnologías las cuales pueden resumirse en la Tabla 1.

Sistema	Instrumento de votación	Registro del voto	Comprobante
Sistema LOV	Boleta por elección con código de reconocimiento	Dispositivo lector óptico que identifica la boleta y registra el voto	La boleta
	Boleta Múltiple y marca manual	Dispositivo con digitalizador que lee la boleta y registra el voto	La boleta
Sistema DRE	Urna electrónica con teclado numérico	Registro del voto en la memoria del dispositivo	No utiliza la boleta, ocasionalmente se cuenta con una impresora para emitir un comprobante
	Pantalla táctil, tarjeta magnética, puntero láser	Registro del voto en la tarjeta magnética y lectura en equipo por separado	No utiliza la boleta, se usa la banda magnética de la tarjeta, ocasionalmente cuentan con una impresora para emitir un comprobante
	Pantalla táctil, tarjeta con chip	Registro del voto en la memoria del dispositivo	No utiliza la boleta, ocasionalmente se cuenta con una impresora para emitir un comprobante

Tabla 1. Diferencias entre los sistemas de voto electrónico.

2.1.4. Sistemas de Almacenamiento Directo del Voto

Estos sistemas llamados DRE por sus siglas en inglés, *Direct-Recording Voting Systems*, son aquellos que ya no utilizan las boletas tradicionales. Existen varios tipos, aunque podemos mencionar tres generales: los de botones, los mini interruptores y los que utilizan una pantalla táctil como interfaz.

Los de botones fueron el reemplazo del sistema de palancas que se empleaba en Estados Unidos para la elección de representantes político, en el cuál se presionan botones que contienen los logotipos de los partidos o nombres de los candidatos para emitir el voto. Los que utilizan mini interruptores son sistemas que se activan mediante el toque de los votantes sobre una superficie flexible que se coloca sobre otra en donde se muestran las opciones disponibles para elegir. El sistema más novedoso involucra el despliegue de logotipos y nombre de los candidatos en un monitor con áreas sensibles al tacto; al presionar sobre el área donde se encuentra el nombre de un candidato o logotipo, el sistema responde de acuerdo a su programación.

2.1.4.1. Secuencia en la votación presencial con un sistema *DRE*.

Desde el punto de vista del votante, la votación por medio de sistemas electrónicos no es demasiado diferente a la metodología tradicional, el procedimiento es el siguiente:

1. El elector se identifica ante las autoridades de la mesa directiva con algún documento, éstas verifican su identidad en un padrón el cuál podría ser el tradicional (en papel) o digital.

2. El elector se dirige a la máquina de votación (PC adaptada, urna electrónica, etc.) y emite su voto, el equipo puede estar en un cuarto oscuro separado, o en una cabina, tras un biombo o sistema similar que asegure el secreto.

3. La emisión se concreta tocando sobre una pantalla sensible, o eligiendo la opción por medio de un teclado. La boleta digital puede incluir fotos de los candidatos y símbolos de los partidos. En el caso de ser un elector con alguna discapacidad, los sistemas proveen alternativas como sonido, plantillas con lenguaje braille, etc.

4. Una vez hecha la selección, el sistema le permite al elector verificar las opciones elegidas antes de emitir el voto. En este momento, si lo desea, puede cambiar un número limitado de ó cuantas veces quiera sus preferencias según la configuración del programa, cuando está seguro, elige la opción "Emitir" o "Votar" y el voto se almacena en los distintos medios de almacenamiento.

5. En ciertos casos (depende de la solución) la urna puede emitir un comprobante en papel que se deposita en una urna tradicional el cual sirve para llevar a cabo auditorias.

6. El elector recibe su documento con la constancia de haber votado. Si el padrón es digital y centralizado, se asienta la emisión y éste queda inhabilitado para –si quisiera o lo intentara- volver a votar en otro sitio.

Al finalizar el horario de votación, las autoridades de mesa realizan los procedimientos para que la urna realice los conteos, emita las actas necesarias y –en algunos casos- transmita los resultados a un centro de recopilación de datos. Esto debe hacerse del modo más seguro posible (datos cifrados, sistemas de clave pública – privada de alta seguridad, inalterabilidad de los mismos).

2.2. SEGURIDAD DEL VOTO ELECTRÓNICO.

2.2.1. Conceptos Relacionados.

A continuación se presentan algunos de los conceptos que se pueden encontrar cuando se revisa la seguridad de los sistemas de voto electrónico presencial.

2.2.1.1. Seguridad.

Algunos métodos típicos de implementar la seguridad en el voto electrónico se enfocan en (1) aislar el proceso para que nadie pueda ver o modificar un voto y (2) construir el sistema bajo un esquema basado en el aislamiento como medida de seguridad, esto es conocido como "seguridad a través de la oscuridad" basada en que si nadie sabe cómo funciona el código nadie puede alterarlo, sin embargo los temas más recientes sobre seguridad hablan del valor de la revisión por parte de expertos, sobre la redundancia y sobre los modelos de código abierto.

2.2.1.2. Criptografía.

Para [Fischer, 2003] el uso de la criptografía en estos sistemas proporciona un nivel más elevado de algunas de las propiedades que se cubren con la auditoria, especialmente en el aspecto de privacidad ya que además de almacenar el voto de manera aleatoria este se encuentra cifrado, lo que impide conocer su contenido y dificulta el poder modificarlo. Aunque la criptografía es sólo una pequeña parte de la seguridad de un sistema también se considera como una parte crítica que permite que algunos tengan acceso a la información y otros no [Boneh, 1999]. La criptografía no es un problema, existen una gran cantidad de algoritmos criptográficos que han sido probados satisfactoriamente, el problema principal es la arquitectura de seguridad que este tipo de sistemas deben tener, ya que según el principio de Kerckhoffs la seguridad de un sistema depende solo de la secrecía de la llave y no de la de los algoritmos [Bonhe, 1999]. Para [Selker, 2003], la tecnología existente es capaz de producir sistemas de voto electrónico seguros, confiables y auditables, estos sistemas tienen como base una arquitectura basada en la redundancia en cada una de las etapas del proceso de votación lo que los hace resistentes contra los ataques externos y contra la inserción de código malicioso.

2.2.2. Amenazas para un sistema de voto electrónico.

Existen una amplia variedad de ataques sobre los sistemas de voto electrónico, desde los individuos involucrados en la creación, la distribución y el uso de estos, así como atacantes externos. Las siguientes amenazas deben tenerse en cuenta al momento de desarrollar un sistema de voto electrónico [Selker, 2003].

Desarrollo Malicioso. Una organización, el autor del código de un sistema de votación o ambos pueden insertar código malicioso, este código puede alterar los votos, desechar algunos, o producir resultados incorrectos, además de hacer que el funcionamiento del sistema se vaya degradando.

Ataques Externos. Hasta la fecha, los atacantes externos no han tenido mucho tiempo y acceso a los sistemas de voto para poder alterarlos, principalmente debido a que en el caso de los sistemas de voto electrónico presencial no se cuenta con un elemento que permita que el sistema pueda ser alterado, por ejemplo un teclado, ya que la interacción entre el equipo y el votante se limita a presionar botones o presionar sobre una pantalla táctil.

Votantes Maliciosos. Un votante que obtenga un acceso indebido al sistema puede tratar de votar en más de una ocasión, votar por alguna otra persona o tratar de robar los votos de otros usuarios.

2.2.3. Arquitectura de Seguridad

Diseñar sistemas seguros requiere atención en muchos niveles, el enfoque de [Selker, 2003] comienza asegurando que no hay un solo punto para una posible falla después de que el votante ha emitido su voto.

El principio de redundancia es central, habilita al sistema para continuar trabajando incluso si ha ocurrido una falla en algún momento. Tener múltiples programas para procesar cada etapa del voto mejora la confiabilidad, sin importar quien los escribió ó cómo los escribió.

La arquitectura que él propone está compuesta de cuatro capas principales: Una interfaz con el usuario que captura los votos, el registro para asegurar que el usuario es válido, un testigo para crear registros seguros y auditables y un acumulador para producir una salida, existen otras capas que le proporcionan al votante pruebas de su voto fue registrado.

La interfaz de usuario. Quizá el componente más importante de cualquier arquitectura de votación es la interfaz de usuario, la arquitectura propuesta por [Selker, 2003] permite que los módulos de la interfaz de usuario sean desarrollados de manera independiente del resto de la arquitectura. La interfaz de usuario toma dos entradas: la definición de la interfaz y la boleta en blanco, la definición de interfaz describe la manera en que se recibe un voto, la interfaz de usuario recolecta los votos de los usuarios así como los datos de registro, después se cifra la boleta, la información de registro es añadida a los votos cifrados y los paquetes resultantes se transmiten al sistema de registro.

El sistema de registro. Es el centro de esta arquitectura de votación, el servidor tiene acceso a todos los votantes permitidos, cuando el sistema recibe un paquete que contiene información de registro y una boleta cifrada verifica si el votante es válido y posteriormente realiza una modificación al archivo de votantes para deshabilitar al votante e impedir que pueda votar nuevamente.

Cada módulo extrae la boleta cifrada, la firma y la envía al módulo testigo para obtener otras firmas, una vez que el testigo regresa las firmas estas pueden ser añadidas al dato cifrado, entonces el paquete completo (sin ningún tipo de información individual) es enviado hacia el módulo de acumulación.

El módulo testigo. Es el más sencillo de todos ya que simplemente toma una boleta cifrada y produce una firma, la boleta es firmada y se produce un valor *Hash*, el cual combinado con la llave privada del módulo testigo produce un número que hasta donde se sabe solo puede ser producido por el que tiene la llave privada.

El módulo acumulador. Este módulo toma el paquete que contiene la boleta cifrada y una serie de firmas producidas por el sistema de registro y el módulo testigo. El acumulador separa las firmas y usa la llave pública del testigo para verificarlas, luego utiliza otro conjunto de firmas para descifrar la boleta. Una vez que la boleta se encuentra en texto simple, las selecciones son almacenadas y la boleta es almacenada tanto en texto simple como cifrado.

La redundancia contra el comprobante impreso.

Una gran controversia que ha surgido últimamente es si se debe añadir un comprobante impreso en los sistemas de votación, la respuesta según [Selker, 2003] es que no debe existir, ya que es fácil de perder a comparación de un comprobante almacenado en la computadora.

Añadir comprobantes a un sistema de votación electrónica, subestima la confianza del público en el sistema. Para [Selker, 2003] en lugar de invertir dinero en crear comprobantes impresos se debería invertir en crear sistemas realmente seguros y confiables.

2.2.4. Seguridad Criptográfica

Este sistema propuesto por [Selker, 2003] utiliza el siguiente esquema criptográfico para alcanzar sus objetivos de seguridad.

- Todos los módulos tienen sus propias llaves privadas.

- Los módulos firman digitalmente todo lo que transmiten, así que los datos se encuentran protegidos contra ataques intermedios al momento de la transmisión.

- Todas las transmisiones son realizadas con *SSL* (*Secure Socket Layer*), el método más confiable para transmisión que se tiene actualmente.

Además de estas medidas, se debe asegurar que el votante sea válido, el sistema de registro no debe tener conocimiento de cómo fue emitido el voto, la boleta debe estar separada del acceso al votante y contar con un cifrado que evita que el voto pueda ser observado por otros en el sistema de registro.

2.2.5. El Problema de la Plataforma Segura

Para [Rivest, 2001], existe un problema fundamental que se debe solucionar cuando se diseñan sistemas de voto electrónico, el "problema de la plataforma segura.", la criptografía no es el problema, existen muchas técnicas que han sido probadas eficientemente, el problema es la interacción de la criptografía con los votantes.

Muchos protocolos asumen que el votante tiene una plataforma computacional segura que ejecutará correctamente una parte del protocolo. El problema de la plataforma segura se presenta sobre todo en las votaciones a través de Internet, donde el sistema esta al alcance de muchas personas en la red, es por eso que debe diseñarse para evitar ataques externos.

2.2.6. Seguridad de los Votos

Para [Cranor & Cytron, 1997] los votos son el punto fundamental de los sistemas de voto electrónico, y deben cumplir con las siguientes propiedades fundamentales:

Correctos. Un sistema es correcto si no es posible alterar un voto, si no es posible para un voto válido ser eliminado del conteo final y si no es posible contabilizar un voto no válido. En la mayoría de los sistemas con esta propiedad el resultado final será adecuado debido a que no se generan errores o si se generan pudieron ser corregidos.

Invulnerabilidad. Un sistema es invulnerable si permite que solo voten electores autorizados y que estos lo realicen solo una vez.

Privacidad. Un sistema es privado si ningún tipo de autoridad o nadie más puede hacer una relación de un voto con la persona que lo emitió y que ningún votante puede probar que votó por alguien en particular.

2.2.7. Seguridad de datos críticos

[Kohno et al, 2004] además de los votos considera que existen otros datos que deben ser protegidos en los sistemas de voto electrónico, otros aspectos fundamentales en cuanto a la seguridad de un sistema de voto electrónico son:

Seguridad de los datos críticos. En los sistemas de votación, proteger la integridad y la privacidad de los datos críticos es de suma importancia, estos archivos se pueden cargar en el sistema de varias maneras, como

son el introduciendo un medio de almacenamiento secundario o descargando la información de alguna red. Es importante que estos archivos tengan una forma de comprobar que provienen de donde deben y que no han sido modificados en el trayecto, para esto es importante el manejo de las firmas digitales.

Seguridad del código fuente. Al crear un sistema seguro, realizar el diseño de manera correcta es sólo un parte, ya que el diseño debe ser implementado de manera segura posteriormente [Kohno et al., 2004]. Si se tiene una implementación que se ha seguido con buenas prácticas de programación pero está algo incompleta se puede pensar que en un futuro versiones más completas del código tendrán al menos la misma calidad, aunque también puede presentarse lo opuesto, ya que es muy complicado construir un sistema seguro sobre bases inseguras. Es de vital importancia evitar las definiciones *hardcoded* en especial si se trata de información relacionada con la seguridad, por ejemplo definir dentro del código contraseñas que se utilicen para cifrar o descifrar alguno de los distintos elementos que se generen dentro del sistema.

2.3. SEGURIDAD Y ARQUITECTURA DE SEGURIDAD.

2.3.1. Conceptos

La seguridad de un sistema es una mezcla de prevención, detección y respuesta. La prevención es hacer un blanco difícil o poco atractivo de atacar, la detección involucra identificar si se realizó o se está realizando un ataque y finalmente la respuesta que permite reaccionar al ataque detectado de manera decisiva para prevenir o disminuir sus efectos [García et al., 2004].

Criptografía. Criptografía es el estudio de técnicas matemáticas relacionadas con los aspectos de la seguridad de la información tales como la confidencialidad, la integridad, y la autenticación de entidades [Menezes, 1996]. Actualmente se emplean dos tipos de criptografía, la simétrica y la asimétrica.

Criptografía Simétrica. En este tipo de criptografía se utiliza la misma contraseña o llave para cifrar y descifrar la información. Entre algunos métodos de criptografía simétrica se pueden mencionar *Blowfish*, *IDEA* (*International Data Encryption Algorithm*), *FEAL* (*Fast Data Encipherment Algorithm*), *DES* (*Data Encryption Standard*) y los más comunes que son el *3-DES*, y el *Rijndael-AES*. El usar la misma llave para cifrar y para descifrar es un problema a la hora de enviar datos, ya que el remitente debe enviar previamente la llave al destinatario para que éste pueda descifrar la información, y debe hacerlo por un canal seguro.

Por lo tanto la criptografía simétrica se emplea especialmente para almacenamiento seguro de datos (solamente una persona necesita la llave). Para envío de datos es preferible la criptografía asimétrica.

Criptografía Asimétrica. Aquí se utilizan dos contraseñas o llaves, una llamada llave pública y una llamada llave privada, la información se cifra con la llave pública y se descifra con la llave privada, no presenta el problema de transmisión de la llave que tiene la criptografía simétrica ya que la llave pública no sirve para descifrar la información.

Los sistemas de criptografía asimétrica incluyen el *DH* (*Diffie & Hellman*), *ElGamal*, *DSA* (*Digital Signature Algorithm*), *Merkle-Hellman*, *Chor-Rivest*, *LUC*, *McEliece*, y finalmente el *RSA* que es el más ampliamente utilizado.

La criptografía asimétrica ofrece varias ventajas sobre la simétrica, como son: [García et al., 2004]

- No se requiere compartir llaves.
- Da origen al concepto de firmas digitales.
- Permite el establecimiento de identidad.

2.3.2. Arquitectura de Seguridad

Una arquitectura de seguridad es el proceso de seleccionar elementos y principios de diseño que cumplan con las necesidades de seguridad del sistema [Graff, 2003.]

2.3.2.1. Arquitectura de Seguridad Multicapa.

[Probst, 2002] menciona que existen varios niveles de mecanismos de seguridad como se muestra en la figura 3.

Componentes que pueden ser reutilizados para el desarrollo de aplicaciones seguras están disponibles especialmente para los niveles más bajos, llamados criptografía y comunicación segura, en los niveles más altos, como los modelos de autorización, controles de acceso, autenticación y auditoría, adecuar o realizar componentes requiere una arquitectura más específica y normalmente no se pueden realizar de manera que satisfagan los requisitos de la aplicación.

Figura 3. Niveles de un mecanismo de seguridad.

2.3.2.2. Marco de seguridad genérico.

Las aplicaciones modernas son realizadas utilizando una arquitectura multicapa, el software se divide en diversos niveles o capas de acuerdo a su funcionalidad y cada capa es capaz de comunicarse con la capa inferior o superior a ella de acuerdo a su funcionalidad a través de una interfaz bien definida.

La figura 4 ilustra una arquitectura de capas creada para proveer mecanismos de seguridad de alto nivel en un ambiente multi-nivel.

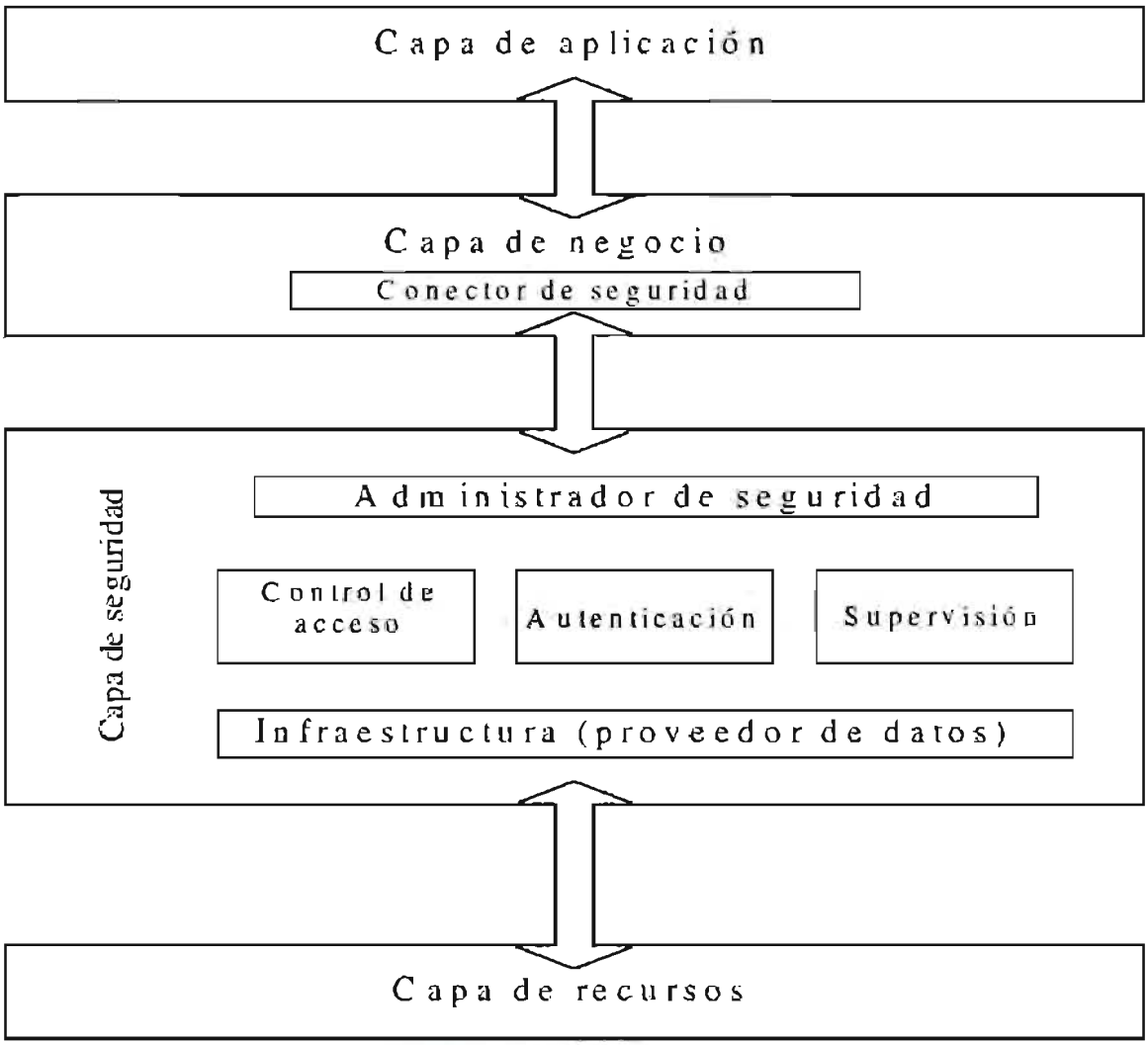


Figura 4. Arquitectura de seguridad multicapa

En particular una capa de seguridad se establece entre la capa de negocio y las capas de recursos, ésta capa tiene una interfaz distinta a la de recursos intercambiando datos seguros con los sistemas de recursos, además tiene un componente utilizado como punto de entrada para la capa de negocio (aplicación) a la capa de seguridad, finalmente la capa de seguridad contiene componentes de seguridad de alto nivel para la coordinación en general, para proveer mecanismos de seguridad

(autenticación, control de acceso, supervisión) y para la infraestructura requerida para reforzar los mecanismos de seguridad.

La capa de seguridad consiste de un conjunto de clases con un objetivo en común, en este caso la seguridad y puede ser llamada marco o "*framework*".

2.3.2.3. Componentes del *framework.*

Para proveer mecanismos de seguridad de alto nivel el *framework* ofrece un coordinador central y tres componentes específicos que corresponden a la autenticación, el control de acceso y la auditoria.

Administrador de seguridad. Controla al resto de los componentes del *framework*.

Autenticación. Responsable de asegurar la correcta autenticación en la que los futuros controles de acceso estarán basados, éste verifica la identidad del sujeto basado en un identificador.

El *framework* no requiere un tipo especial de identificador, por lo que el desarrollador del software puede utilizar varios mecanismos de autenticación, es tarea del componente de autenticación validar al identificador de acuerdo con el método que se utilice. El *framework* puede ofrecer un método basado en contraseñas.

Controlador de acceso. Es responsable de controlar el acceso a los objetos de acuerdo con un modelo de control de acceso particular y basado en una autenticación válida.

Supervisión. Debe ser capaz de registrar actividades de seguridad relevantes, el *framework* provee un método flexible de auditoria que recolecta mensajes del resto de los componentes y opcionalmente los puede almacenar en diversos medios.

El sistema de auditoria permite filtrar los mensajes para cada medio de salida lo cual puede ser utilizado para imprimir mensajes críticos directo en la pantalla y otro tipo de información en una base de datos.

2.3.2.4. Componentes del control de acceso.

A continuación se describen los componentes relacionados con el control de acceso.

Sujeto. Se refiere a los actores o entidades del sistema, como personas, procesos o entidades.

Objeto seguro. Es la base para todas las clases de objetos que necesitan ser protegidos dentro del *framework*. Para asegurar esto es necesario que el cliente no obtenga una referencia directa al objeto.

Autorizaciones. Contiene el tipo de derechos de acceso a un recurso. El *framework* ofrece componentes especializado para una autorización positiva (permisos) o una autorización negativa (prohibiciones).

Restricciones. Permite autorización más restringida dentro del sistema en un modo más flexible, ejemplos de estas restricciones incluyen la localización (por ejemplo accesos desde una dirección IP específica) o

restricciones de tiempo (accesos permitidos solo entre determinadas horas).

Modelo del control de acceso. Este componente recolecta sujetos, objetos, autorizaciones y restricciones de los componentes proveedores de datos y los transfiere a la base de autorización.

Cuando el controlador de acceso contacta a éste modelo para manejar una petición se genera una búsqueda que contiene al sujeto que realizó la petición y al objeto que está siendo requerido, después manda este patrón al modelo de autorización donde las reglas son buscadas y analizadas, dependiendo de las reglas uno de los siguientes resultados es regresado al controlador de acceso:

Verdadero: Si una regla es encontrada y el acceso es concedido, por ejemplo el sujeto tiene un permiso para acceder al objeto.

Falso: Si una regla es encontrada y el acceso es denegado, por ejemplo una prohibición niega el acceso del sujeto al objeto.

2.3.2.5. Mecanismo de control de acceso canónico.

Para proveer controles de acceso genéricos se debe encontrar una manera flexible de reforzar los controles de acceso. La premisa básica es que los controles de acceso pueden ser establecidos en términos de los sujetos que acceden a los objetos. La Figura 5 ilustra los pasos particulares para un control de acceso canónico.

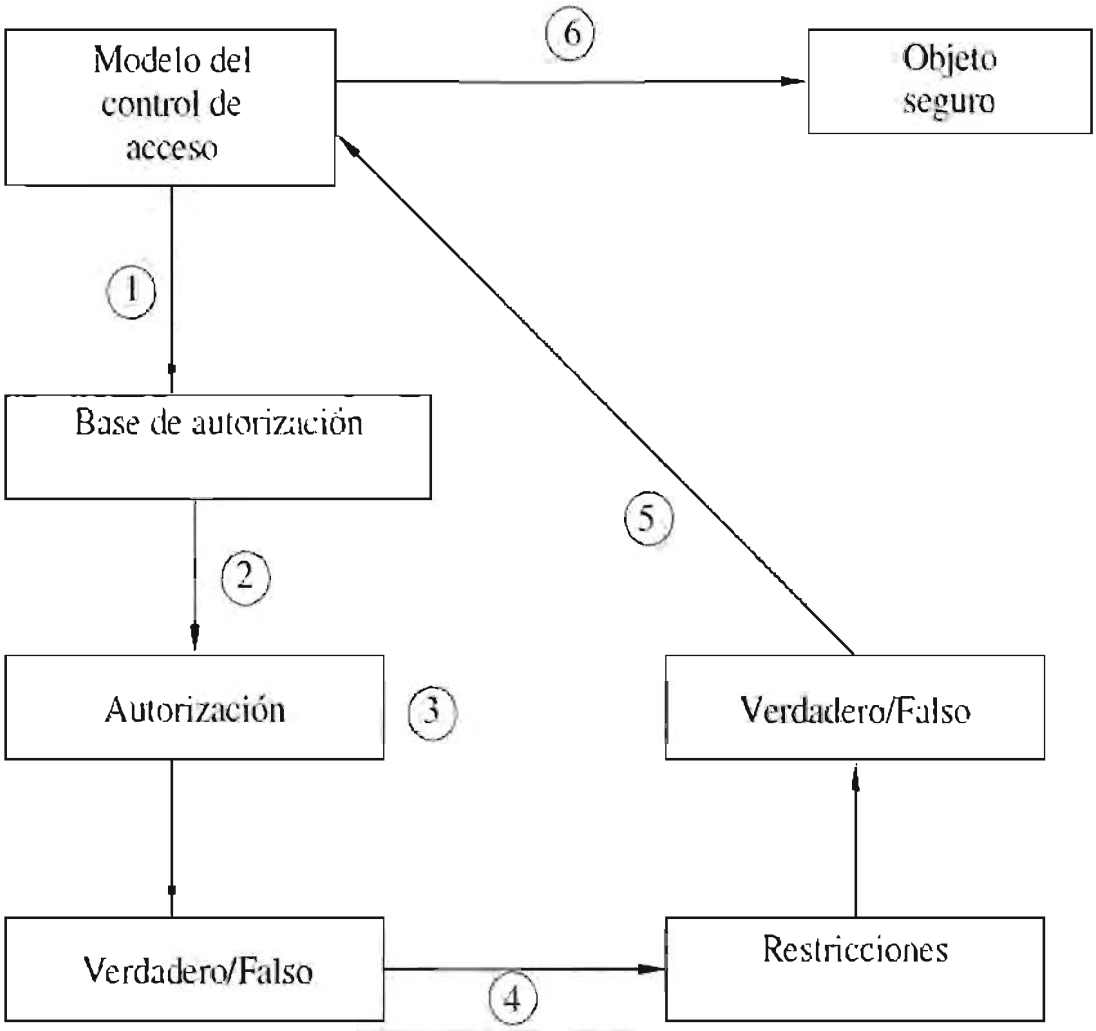


Figura 5. Pasos en un control de acceso canónico.

En general, un sujeto desea acceder a un objeto protegido de alguna manera, la operación solicitada en el objeto define las autorizaciones que son necesarias para realizar esta tarea. La manera en que se decide si el acceso es aprobado o no es determinada por el modelo de control de acceso, las autorizaciones definidas por ese modelo, y las restricciones asignadas a esas autorizaciones.

Primero el controlador de acceso recibe una petición de un sujeto ya identificado para cierta operación sobre un objeto protegido, cada modelo de control de acceso busca una combinación sujeto/objeto en la base de autorización (1) que concuerde con la petición. El proceso de búsqueda regresa una lista de reglas de autorizaciones que concuerdan con la combinación sujeto/objeto.

Cada autorización es explícitamente verificada invocando un método de verificación (2). Sin embargo, existe la posibilidad de definir restricciones adicionales que restrinjan un poco más un acceso a un objeto protegido, nuevamente se verifican estas restricciones adicionales (3/4). Cuando ambas, la autorización y la restricción conceden el acceso, el modelo de control de acceso reporta un resultado positivo (acceso concedido) o negativo (acceso denegado) dependiendo del modelo (5). Este resultado es entonces regresado al controlador de seguridad quien finalmente da o niega el acceso al objeto solicitado (6).

2.3.3. Aspectos de Seguridad en un Sistema Distribuido.

[Bidan, 1997] realiza un análisis de los elementos que se pueden encontrar en una arquitectura de seguridad.

Para él, la seguridad significa protección en contra de accesos no autorizados a la información, está relacionada con la confidencialidad (la información sólo es accesible a los usuarios autorizados a ella), la integridad (la información puede ser sólo modificada por usuarios que tengan el permiso para hacerlo) y la disponibilidad (el uso del sistema no puede ser negado de una manera maliciosa).

La seguridad es reforzada utilizando funcionalidades de seguridad como el cifrado, la autenticación y el control de acceso.

El cifrado consiste en hacer la información ilegible (para asegurar la confidencialidad), inalterable (firma digital para asegurar la integridad) o ambas, el cifrado se utiliza para proteger información almacenada o para el intercambio de información contra lectura o modificación.

La autenticación permite asegurar la identidad de la entidad, esto es, verificar que la entidad (un usuario o proceso) es quien asegura ser. Más específicamente, el protocolo de autenticidad permite asociar cada operación del sistema con un usuario único, permitiendo verificar si la operación está permitida o no, un ejemplo de un protocolo de autenticidad es cuando se solicita un nombre de usuario y una contraseña.

El control de acceso rige las operaciones en las entidades del sistema. Por ejemplo, el control de acceso a un sistema de archivos consiste en verificar cuáles usuarios están autorizados a acceder a los archivos, el control de acceso también verifica la interacción entre entidades, además trata con el flujo de información entre entidades. El control de acceso depende de la autenticación, la identidad de la entidad debe ser única e inolvidable.

2.3.3.1. Especificaciones del cifrado.

Los datos son calificados como texto simple (*plaintext*) cuando el acceso a estos permite conocer la información que contienen, el proceso de transformar esos datos de una manera que esconda la información contenida es cifrar la información, el dato resultado es conocido como

texto cifrado (*ciphertext*), el proceso de convertir un texto cifrado en texto simple es conocido como descifrado.

Un algoritmo de cifrado está compuesto por una función de cifrado y su correspondiente función de descifrado, la función de cifrado debe asegurar ya sea la confidencialidad o la integridad.

En general, las funciones de cifrado y descifrado no son secretas ya que la seguridad del algoritmo de cifrado está basado en el uso de las llaves. La función de cifrado toma como entrada el texto simple y una llave de cifrado para calcular el texto cifrado, de manera inversa, dado un texto cifrado, el texto simple es calculado utilizando la correspondientes llave y función de descifrado como se muestra en la Figura 6.

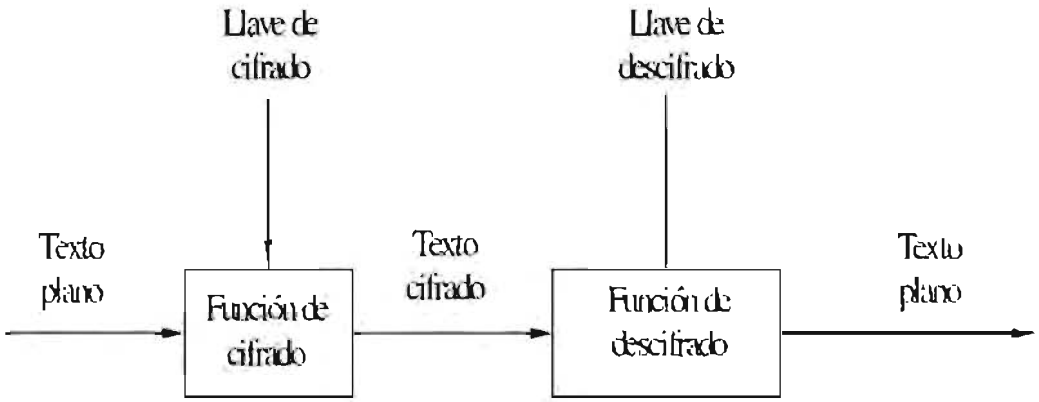


Figura 6. Esquema de cifrado y descifrado.

2.3.3.2. Especificación de los requerimientos de cifrado.

Se basa en especificar el algoritmo que se va a utilizar, así como los parámetros que describen su comportamiento. Los parámetros deben

incluir: uso del algoritmo, esto es, si se utiliza para cifrarlos datos (asegurar confidencialidad), firmarlos (asegurar la autenticidad), realizar una función *Hash* (para detectar manipulación) y el manejo de las llaves que incluye el tamaño, forma en que se almacenan y durante cuanto tiempo serán útiles.

2.3.3.3. Especificaciones de la autenticación.

Las entidades de autenticación permiten verificar que las entidades sean quienes aseguran ser. Un protocolo de autenticación especifica el proceso de autenticación, esto es, las entidades que participan, el intercambio de mensajes entre estas entidades y el formato de estos mensajes (texto simple o cifrado), sin embargo, el principal objetivo de estos es la autenticación de las entidades, algunos de ellos deben además tratar con propiedades adicionales de seguridad.

2.3.3.4. Especificaciones del control de acceso.

Una *ACP* (*Access Policy Control*) define el conjunto de reglas llamadas *ACR* (*Access Control Rules*) que se especifican para una pareja de entidades (e1,e2) que pueden ser usuarios, procesos, archivos, etc., un ejemplo es una regla que especifique si la entidad e1 tiene permitido el acceso a e2 o no.

2.3.4. Protocolos Criptográficos.

Para [Menezes, 1996] los diferentes elementos que se encuentran presentes y que hay que considerar cuando se trabaja con elementos criptográficos son:

Primitivas criptográficas. Son las herramientas básicas que se utilizan para proveer seguridad, estas incluyen los algoritmos del cifrado simétrico, los del cifrado asimétrico y algunas que no requieren de llaves como las funciones *Hash*, o los elementos aleatorios.

Protocolo criptográfico. Es un algoritmo distribuido definido por una secuencia de pasos que especifican las acciones requeridas por dos o más entidades para lograr un objetivo específico de seguridad.

Mecanismo criptográfico. Término más general que abarca protocolos, algoritmos y técnicas no criptográficas para lograr sus objetivos de seguridad.

Falla de protocolo. Ocurre cuando un mecanismo falla en alcanzar los objetivos para los que fue creado, de una manera en que un adversario gana ventaja, no rompiendo alguna primitiva como un algoritmo de cifrado directamente, pero sí manipulando el protocolo.

Algunas causas de la falla de un protocolo son:

- Debilidad en alguna de las primitivas que pueda ser amplificada por un mecanismo o un protocolo.
- Medidas de seguridad que son subestimadas o no son entendidas claramente.
- Sobrestimar algunos principios de seguridad aplicables a una amplia clase de primitivas como el cifrado.

2.3.5. Ataques a los esquemas de cifrado.

El objetivo de este tipo de ataques es recuperar el texto simple a partir del texto cifrado, o peor aún deducir la llave de cifrado, algunos de los tipos de ataques que se pueden encontrar son:

Ataque a texto cifrado. Es cuando un adversario trata de de deducir la llave de cifrado o el texto simple observando el texto cifrado, cualquier esquema de seguridad vulnerable a este tipo de ataque es considerado absolutamente inseguro.

Ataque a texto simple elegido. Es cuando se elige un texto simple y se obtiene su correspondiente texto cifrado, posteriormente el adversario usa esta información para deducir el texto simple que corresponde al texto cifrado que contiene la información que desea conocer.

Ataque adaptativo. Se realiza a un texto simple elegido en donde la elección del texto simple dependerá del texto cifrado recibido.

Ataque a un texto cifrado elegido. Es cuando el adversario selecciona el texto cifrado y luego obtiene el correspondiente texto simple.

2.3.6. Ataques a los protocolos.

Al paso de los años, se han identificado diferentes tipos de ataques a protocolos criptográficos, los ataques que se pueden realizar se clasifican en ataques pasivos y activos. Un ataque pasivo es en el que el adversario solo se desea conocer la información enviada por el canal de comunicación, un atacante pasivo solo se enfoca en la confidencialidad de los datos. Un

ataque activo es cuando el adversario trata de borrar, añadir o alterar la información transmitida, un atacante activo se enfoca en la integridad, la autenticación y la confidencialidad de los datos. Para [Menezes, 1996] algunos de los ataques que se llevan a cabo sobre los protocolos son:

1. Ataque de llaves conocidas, es cuando el adversario tiene acceso a llaves utilizadas anteriormente y con ellas puede determinar llaves nuevas.

2. Ataque de repetición, es cuando el adversario registra una conversación y la repite posteriormente, ya sea entera o solo una porción.
3. Personificación, es cuando el adversario asume la identidad de uno de los participantes legítimos.

4. Ataque de diccionario, se realiza comúnmente contra las contraseñas, típicamente una contraseña se almacena como una imagen de una función *Hash*, aquí el adversario tiene una lista de posibles contraseñas y trata de comparar los *Hash* de estas con el de las contraseñas almacenadas.

5. Búsqueda, es un ataque similar al de diccionario pero aquí lo que se trata de descifrar es el mensaje.

2.3.7. Modelos para evaluar la seguridad.

La seguridad de los protocolos y de las primitivas criptográficas puede ser evaluada bajo diferentes modelos. Los más prácticos son el computacional, el probable y el *ad hoc*.

Seguridad probable. Un método criptográfico es *probablemente seguro* si la dificultad para vencerlo puede mostrarse como igual de difícil que resolver un problema que ya se sabe que es complicado (típicamente teoría de números), como la factorización de enteros o el cálculo de logaritmos discretos.

Seguridad computacional. Esta mide la cantidad de esfuerzo computacional requerido utilizando los mejores métodos conocidos actualmente para vulnerar un sistema, se debe asumir que los sistemas deben haber sido bien estudiados para determinar qué ataques son relevantes. Una técnica se dice que es computacional mente segura si el nivel computacional requerido para vulnerarla supera por un margen considerable los recursos computacionales del adversario.

Seguridad *ad hoc*. Esta aproximación consiste de una variedad de argumentos que son mayores que los que posee el adversario. Las primitivas criptográficas y los protocolos que pasan a este análisis se dice que tienen una seguridad heurística.

Diseño de una arquitectura. Para realizar el diseño de una arquitectura de software es importante considerar el ciclo de vida del desarrollo de software, que comienza con la identificación de una necesidad y termina

con la verificación formal del software desarrollado en contra de esta misma [IPL, 1997].

Para el desarrollo de una arquitectura se tienen varios métodos, entre los que se encuentran el secuencial o de cascada y el progresivo o iterativo. El proceso secuencial se basa en los siguientes pasos: recopilación de requisitos, desarrollo, pruebas y entrega final.

El método progresivo consta de las siguientes partes: requisitos, diseño, implementación y pruebas y revisión, pero a diferencia del secuencial, aquí se tiene un ciclo iterativo que permite ir creando versiones del sistema que si bien al principio no cumple con los requisitos establecidos si cuenta con una funcionalidad para realizar distintas pruebas y con base en ello modificar el diseño o la forma de implementación, una vez que las pruebas son satisfactorias esa versión sirve como base para aplicar nuevamente los pasos del proceso iterativo y así hasta que el sistema finalmente cumple con los requisitos deseados.

2.3.8. Diseño de una Arquitectura.

Para realizar el diseño de una arquitectura de software es importante considerar el ciclo de vida del desarrollo de software, que comienza con la identificación de una necesidad y termina con la verificación formal del software desarrollado en contra de esta misma.

Para el desarrollo de una arquitectura se tienen varios métodos, entre los que se encuentran el secuencial o de cascada y el progresivo o iterativo. El proceso secuencial se basa en los siguientes pasos: recopilación de requisitos, desarrollo, pruebas y entrega final.

El método progresivo consta de las siguientes partes: requisitos, diseño, implementación y pruebas y revisión, pero a diferencia del secuencial, aquí se tiene un ciclo iterativo que permite ir creando versiones del sistema que si bien al principio no cumple con los requisitos establecidos si cuenta con una funcionalidad para realizar distintas pruebas y con base en ello modificar el diseño o la forma de implementación, una vez que las pruebas son satisfactorias esa versión sirve como base para aplicar nuevamente los pasos del proceso iterativo y así hasta que el sistema finalmente cumple con los requisitos deseados.

3. DESARROLLO

3.1. MODELO DE VOTACION

El modelo de votación que se siguió para el desarrollo de los elementos de auditoría y de seguridad está basado en tres etapas cómo se ha revisado en el capítulo de trabajos relacionados. La primera es la de pre votación que inicia con el encendido del sistema, siguiendo la verificación de los componentes del equipo, posteriormente aparece la parte de verificación de la integridad y autenticidad de los datos de configuración, a continuación el equipo se configura y se termina con la impresión de los distintos documentos generados. La etapa de votación inicia con la habilitación del equipo para que un usuario pueda emitir su voto, éste tiene la opción de corregir o confirmar su elección y una vez que termina de participar en todas las elecciones el sistema se deshabilita. La etapa de post votación comienza con la habilitación del sistema por un administrador, aquí se realiza el conteo total de los votos, la generación del archivo de resultados y la impresión de las actas con los resultados finales.

3.2. DESARROLLO DE LOS ELEMENTOS DE AUDITORIA

Para realizar un sistema auditable se tomó como elemento básico la redundancia, por lo que se tenían varias maneras de presentar y almacenar la misma información, los elementos que se consideran que deben estar presentes en cada una de las etapas del proceso de votación para crear un sistema auditable son los siguientes:

3.2.1. Etapa de Pre Votación

Los elementos que se desarrollaron para esta etapa fueron:

Comprobante del funcionamiento de componentes. Los datos del procesador se obtuvieron del archivo *administrador de dispositivos* que se genera cada que inicia el sistema operativo *Windows*, los datos que se obtenían eran el tipo de procesador, la frecuencia a la que trabaja y la cantidad de memoria del equipo que también se genera cada que arranca el sistema operativo, para la prueba de impresión primero se configuraba el la impresora y se enviaba una cadena para que fuera impresa pidiéndole al usuario que confirmara si se había impreso de manera correcta, para el sonido se pedía al usuario que se colocara los audífonos y escuchara un sonido que se reproducía haciendo uso de las funciones Windows para manejo de archivos de sonido (en formato *WAV*) pidiéndole que confirmara si escuchaba el sonido, para los medios de almacenamiento, solo se creaba un archivo en cada uno de los medios removibles.

Datos de configuración. Se solicita confirmar si los datos de configuración que se mostraban en pantalla eran los correctos, si lo eran se procedía al siguiente paso, de lo contrario se reportaba el error en la bitácora y el sistema se apagaba.

Actas iníciales. Aquí se generan las actas de apertura con los datos específicos del lugar donde se encuentra el equipo, posteriormente se generan las actas con los datos de cada elección y el número de votos de cada contendiente, al ser las actas iníciales, estas deben tener cero votos para cada participante, la cantidad de votos se muestra en número y letra.

3.2.2. Etapa de Votación

Para esta etapa se desarrollaron los siguientes elementos:

Comprobante impreso. Formado por el nombre de la elección y la opción elegida.

Comprobante electrónico. Formado por los números de elección y opción elegida separados por el carácter "-".

Reporte directo hacia el votante. Se muestra en la pantalla de bienvenida el número de votos que se ha registrado en cada elección, al finalizar la participación del usuario, en la pantalla de agradecimiento se muestra nuevamente el número de votos que debió haberse incrementado en uno.

Auditoría durante la elección. Cada determinado número de votos se genera un documento con los resultados hasta ese momento, una vez que se tiene más de uno de estos archivos, se verifica el más reciente con el inmediato anterior, revisando que ningún candidato tenga menos votos en el archivo más reciente que los que tenía en el conteo anterior, si esto ocurre se registra el error, éste procedimiento también se realiza para el total de votos registrados en cada elección.

Almacenamiento en diversos medios. Para que el sistema sea considerado robusto, en todo momento los votos registrados deben ser los mismos en cada uno de los medios de almacenamiento, que son una memoria USB, una memoria Compact Flash (CF) y el disco duro del equipo. Se debe almacenar en un medio una vez que se aseguró que ya se ha registrado el voto en otro de los medios.

3.2.3. Etapa de Post Votación

Para la etapa de generación de resultados, los elementos generados fueron los siguientes:

Actas finales. Una vez que termina la elección, se generan las actas finales con los datos de la elección y los resultados, éstas incluyen el nombre de cada contendiente y el total de votos en número y letra que han recibido, éste documento también se obtiene de forma impresa.

Archivo de resultados finales. Este archivo contiene los resultados finales en un formato específico que facilite su análisis posterior, se incluyen todos los tipos de elecciones y participantes de cada elección, incluyendo los votos nulos así como un número que indica el total de votos registrados en cada elección.

Pantalla de resultados finales. Se muestran los resultados finales de cada elección para poder realizar una comparación inmediata con la información contenida en las actas.

Acta de cierre de la votación. Este documento es similar al acta de apertura, pero en este caso contiene la hora en que se da por terminado el proceso de votación.

De ésta manera se cuenta con tres elementos para un recuento en caso de duda sobre los resultados arrojados por las actas finales, se cuenta con los comprobantes impresos, los comprobantes electrónicos (*EBI*) y el archivo de votos almacenado en diversos medios, y para verificar el total de votos

en cada una de las elecciones se cuenta además con la información contenida en la bitácora.

El archivo de registros o bitácora. Es un elemento especial de la auditoría ya que se encuentra presente en todas las etapas del proceso de votación, se forma con la hora y la clave del evento que ocurrió, la Tabla 2 muestra los eventos y las claves con las que se registran.

Evento ocurrido	Clave del evento
Fecha de encendido incorrecta	100
Apagado	101
Fecha de encendido correcta	102
Encendido	103
Verificación de componentes	104
Audio correcto	105
Audio incorrecto	106
Configuración correcta	107
Impresión correcta de documentos iníciales	108
Impresión incorrecta de documentos iníciales	109
Configuración incorrecta	110
Configuración del equipo	111
Habilitación válida	112
Habilitación inválida	113
Finalizó el voto correctamente	114
Impresión correcta del voto	115
Impresión incorrecta del voto	116
Entrada al modulo de administración	117
Finaliza la jornada electoral	118
Impresión incorrecta de documentos finales	119

Impresión incorrecta de documentos finales	120
Deshabilitar equipo	121
Memoria USB correcta	122
Memoria USB incorrecta	123
Memoria Flash correcta	124
Memoria Flash incorrecta	125
Falló impresora	126
Auditoría intermedia correcta	127
Auditoría intermedia incorrecta	128

Tabla 2. Eventos registrados en la bitácora

La emisión de un voto también se registraba, pero solamente la elección en la que se había participado con el fin de no violar la propiedad de confidencialidad. El registro era de la siguiente manera:

NUMERO DE LA ELECCIÓN * 10 + 1 (cuando se corrige la elección) ó + 0 (cuando se confirma)

La Tabla 3 contiene los elementos generados, la etapa en la que aparecen y el formato en el que se presentan.

Elemento	Etapa	Formato
Funcionamiento de los componentes	Pre votación	Impreso / pantalla / archivo
Datos de configuración	Pre votación	Impreso / pantalla / archivo
Actas iniciales	Pre votación	Impreso / archivo
Comprobante impreso	Votación	Impreso
Comprobante electrónico (EBI)	Votación	Archivo

Reporte directo hacia el votante	Votación	Pantalla
Resultados de auditorías intermedias	Votación	Archivo
Almacenamiento en diversos medios	Votación	Archivo
Actas finales	Post votación	Impreso / archivo
Archivo de resultados finales	Post votación	Archivo
Acta de cierre de la votación.	Post votación	Impreso / archivo
Pantalla de resultados finales	Post votación	Pantalla

Tabla 3. Elementos auditables generados en las distintas etapas.

3.3. DESARROLLO DE LOS ELEMENTOS DE SEGURIDAD

Para el manejo de la seguridad se consideraron las tres etapas que forman el proceso de votación completo, generación de medios, votación y análisis de resultados. Para realizar un protocolo criptográfico lo primero que se debe revisar es qué datos se desean proteger y qué tan importante es su confidencialidad, asegurar que nadie conozca su contenido, y su integridad, protegerlos de una posible modificación o si han sido modificados poder detectarlo. Antes de analizar el desarrollo del protocolo criptográfico es importante conocer la siguiente nomenclatura:

Generador de medios.

Llave pública cifrada	$\hat{e}_{GM}$
Llave privada	d_{GM}
Llave simétrica	k_{GM}
Llave especial	k_{ESP}

Urna electrónica.

Llave pública	e_U

Llave privada	d_U
Llave simétrica	k_U
Llave privada para firmar	d_{Uf}
Llave pública para verificar la firma	e_{Uf}

Análisis de resultados.

Llave pública	e_R
Llave privada	d_R
Llave simétrica	k_R
Llave especial	k_{ESP}

Generales.

Datos	a
Función Hash	H
Archivo cifrado	c
Firma digital	s
Firma digital de varios archivos	s_i
Password	p
Resultados	r
Resultados tecleados	r_T
Dato modificado	*

3.3.1. Manejo Inicial de las Llaves.

El manejo inicial de las llaves pública y privada es fundamental para el buen funcionamiento del protocolo, la generación de éstas debe realizarse en diferentes equipos, uno que genere llaves para el equipo generador de medios, otro para la urna electrónica y uno más para el equipo que analiza

los resultados, comunicándose entre ellos para intercambiar las llaves necesarias, con esto se evita tener todas las llaves en un solo equipo. El instalar las llaves con anterioridad en los equipos permite defenderse del ataque más obvio que es el de suplantación, que consiste en crear un juego propio de llaves y que el sistema al cargarlos de algún dispositivo las tome como válidas.

La Tabla 4 muestra la ubicación de las distintas llaves en los diferentes equipos para poder implementar este protocolo. Las llaves simétricas se generan en el momento que se requieren y son las que se transportan por un canal de comunicación que se considera inseguro, de estas llaves no se requiere que esté instalada ninguna en alguno de los equipos.

Generador de medios	Urna electrónica	Equipo de resultados
Llave publica de la urna electrónica (e_U) Llave privada del generador de medios (d_{GM})	Llave privada de la urna electrónica (d_U) Llave pública del generador de medios cifrada ($\hat{}e_{GM}$) Llave publica de resultados (e_R)	Llave privada del equipo que analiza los resultados (d_R) Llave pública de la urna electrónica (e_R)

Tabla 4. Ubicación inicial de las llaves públicas y privadas.

3.3.2. Etapa de Generación de Medios

Una vez que se han generado los archivos de configuración, se debe asegurar que éstos estén protegidos contra el ataque de suplantación. El protocolo para el manejo de la seguridad en esta etapa es:

$s = d_{GM}(a, H(a))$

$k_{ESP} = s_1 + s_2 + s_3 + \ldots + s_n$

$\hat{}e_{GM} = k_{ESP}(e_{GM})$

$c = k_{GM}(a)$

$p = e_U(k_{GM})$

El funcionamiento es el siguiente: Primero se obtiene una firma digital (s) de los archivos que se van a manejar con el objetivo de poder asegurar su autenticidad y su integridad. Posteriormente tomando partes de esas (s_1, s_2, …, s_n) firmas se crea la llave especial (k_{ESP}) y con ella se cifra la llave pública (e_{GM}) que sirve para verificar la integridad y autenticidad de los datos. El siguiente

paso es cifrar los archivos de configuración (a) utilizando la llave simétrica del equipo generador de medios (k_{GM}) que es generada al momento de ser requerida y que se protege cifrándola con la llave pública de la urna electrónica (e_U) lo que garantiza que solo ésta podrá descifrarla. Una vez finalizados estos procedimientos los archivos que se transportan por un canal de comunicaciones inseguro son los datos cifrados (c), la firma digital (s) y la llave simétrica cifrada o password (p).

3.3.3. Etapa de Votación

Aquí se trabajó con las etapas de pre votación que es la verificación de la seguridad de los archivos de configuración, votación encargada de la seguridad de los votos y archivos que se generan durante la elección y post votación que maneja la seguridad de los archivos de resultados.

3.3.3.1. Etapa de pre votación.

Lo primero que hay que evitar es que el sistema se encienda en una fecha u hora incorrecta, para esto hay que obtener la hora del sistema y se compararla con una hora de encendido determinada y si la hora del sistema es menor, se reporta que aún no es momento de encender el equipo y éste se apaga. Si la fecha y la hora de encendido son correctas, se procede a la verificación de la integridad de los datos, el protocolo que realiza la verificación de la integridad y autenticidad de los datos es el siguiente:

$k_{GM} = d_U (p)$
$a = k_{GM}^{-1} (c)$
$k_{ESP} = s_1+s_2+\ldots+s_n$
$e_{GM} = k_{ESP}^{-1}(\hat{}e_{GM})$
$e_{GM} (H(a),s)$

Primero se descifra la llave simétrica cifrada (p) utilizando la llave privada de la urna electrónica (d_U) y se obtiene la llave simétrica (k_{GM}) que cifró los datos de configuración. Una vez que se han descifrado éstos datos (c) y se han obtenido los archivos originales (a) se procede a crear, con las firmas de los datos recibidos, la llave especial (k_{ESP}) que descifra la llave pública del generador de medios (e_{GM}) utilizada para verificar la autenticidad de los datos, si las firmas no han sido modificadas la llave creada corresponde a la utilizada para cifrar la llave pública y esta se descifra de manera adecuada lo que permite realizar la verificación de la integridad de los datos.

Una vez que los datos han sido verificados se procede a la configuración del equipo, en esta etapa se tienen los siguientes archivos:

- Datos de configuración.
- Comprobantes de componentes.
- Comprobante de validez de los datos.
- Actas iníciales.
- Actas de apertura.

Para estos archivos la confidencialidad no es tan importante porque la información que contienen es de conocimiento público, lo que se debe proteger es su integridad, los que más riesgo tienen son los datos de configuración ya que son transportados de un lugar a otro mientras que los comprobantes y las actas al generarse dentro del sistema, corren menor riesgo de que alguien tenga acceso a ellos. De todas maneras estos archivos cuentan con cierto nivel de seguridad ya que son cifrados de manera simétrica con una llave de 128 bits de longitud que se genera en el momento en que es requerida, el cifrado se realiza utilizando el algoritmo.

3.3.3.2. Etapa de votación.

En esta etapa el principal elemento a proteger es el voto almacenado, asegurándose que se cumplan las propiedades de privacidad y correctez que establecen que un voto no debe ser modificado y que no se puede hacer una relación voto - votante, si bien se cuenta con un almacenamiento aleatorio que evita el relacionar un voto con el usuario que lo emitió, una medida para proteger al voto de ser modificado y

reforzar su privacidad es el cifrado, cada voto se cifra con su propia llave simétrica de 128 bits utilizando el algoritmo.

En esta etapa además se cifran también los siguientes elementos:

- Comprobante electrónico.
- Resultados de auditorías intermedias.

Las llaves utilizadas para cifrar los distintos documentos se crean en el momento en que se necesitan y posteriormente se cifran de manera asimétrica utilizando la llave pública del equipo que analiza los resultados, esto como una medida más de seguridad para prevenir que en algún momento pudieran ser extraídos del equipo y modificados.

3.3.3.3. Etapa de post votación.

Una vez que finaliza la jornada electoral, como primera medida de seguridad se cifra la llave simétrica utilizada para el cifrado de las EBI con la llave pública del equipo que analiza los resultados, lo que garantiza que en caso de que quieran ser revisados solo podrán serlo en el lugar donde se encuentre el equipo de análisis de resultados. Posteriormente se genera la seguridad necesaria para el archivo de resultados que será enviado a través de un canal de comunicaciones inseguro. Al igual que en los datos de configuración, proteger la confidencialidad no es tan importante ya que los resultados se conocen e incluso quedan registrados en las actas impresas.

El objetivo del protocolo es depender lo menos posible de la seguridad de las llaves, en especial de las privadas, para la etapa de envío de los resultados se tienen dos opciones.

Opción 1. Capturando los resultados de un acta impresa.

El protocolo para esta opción incluye como última medida de seguridad que los resultados se capturen de un acta que los contenga y en base a esta información poder asegurar que el archivo recibido no ha sido modificado. El protocolo es el siguiente:

d_{Uf}, e_{Uf}
$s = d_{Uf}(r, H(r))$
$c = k_U(r)$
$\hat{}d_{Uf} = k_U(d_{Uf})$
$p = e_R(k_U)$
$k_{ESP} = s(r)$
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$

Lo primero es crear un juego de llaves asimétricas (d_{Uf} y e_{Uf}) que serán utilizadas para firmar archivo de resultados (r) y obtener su firma digital (s). Posteriormente se cifran el archivo de resultados y la llave privada (d_{Uf}) que se utiliza para generar la firma de los resultados con la llave simétrica (k_U) y ésta se cifra con la llave pública (e_R) del equipo que verifica los resultados. Después se crea una llave especial (k_{ESP}) formada por la firma de los resultados y con ella se cifra la llave pública (e_{Uf}) que verifica la integridad de los datos. Así se obtiene un nuevo conjunto de datos a enviar formado por el archivo de resultados cifrado (c), las llaves pública y privada utilizadas para verificar su integridad y

autenticidad cifradas ($\hat{}e_{Uf}$, $\hat{}d_{Uf}$) y la llave simétrica de los resultados cifrada (p).

Opción 2. Envío de la firma a través de un canal de comunicaciones seguro.

En esta opción, se debe enviar la firma digital de los resultados por medio de un canal de comunicaciones considerado seguro. El protocolo es el siguiente:

d_{Uf}, e_{Uf}
$s = d_{Uf}(r, H(r))$
$c = k_U(r)$
$p = e_R(k_U)$
$k_{ESP} = s(r)$
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$

Primero se deben generar las llaves pública y privada (d_{Uf} y e_{Uf}) para verificar la integridad de los datos. Con la llave privada se obtiene la firma digital (s) del archivo de resultados (r), posteriormente éste se cifra con la llave simétrica (k_U) y ésta se cifra utilizando la llave pública del equipo de análisis de resultados (e_R), luego se crea la llave especial (k_{ESP}) a partir de la firma digital de los resultados y con ésta se cifra la llave pública (e_{Uf}) que verifica la integridad de los archivos.

Así se obtiene un nuevo conjunto de datos a enviar formado por la firma de los resultados (s), el archivo de resultados cifrado (c), la llave pública que verifica la integridad y autenticidad cifrada ($\hat{}e_{Uf}$) y la llave simétrica cifrada (p). Aquí el archivo que contiene la firma digital debe ser enviado

a través de un canal de comunicaciones seguro junto con algún identificador del equipo que lo generó.

3.3.4. Etapa de Análisis de Resultados.

En esta etapa el protocolo se encarga de descifrar los archivos recibidos y de verificar su integridad y autenticidad, asegurando detectar si se ha realizado algún cambio en su contenido, al finalizar los datos quedan listos para su análisis. Se cuenta un protocolo por cada una de las opciones de envío generadas en la etapa anterior.

Opción 1. Capturando los resultados del acta final.

El protocolo para esta opción es el siguiente:

$k_U = d_R$ (p)
$r = k_U^{-1}$ (c)
$d_{Uf} = k_U^{-1}$ (^d_{Uf})
$s = d_{Uf}(r_T)$
$k_{ESP} = s$
$e_{Uf} = k_{ESP}$(^e_{Uf})
e_{Uf} (H(r), s)

Al llegar el conjunto de archivos formado por {c, ^e_{Uf}, ^d_{Uf}, p} al equipo que recopila y analiza los resultados, lo primero es descifrar la llave asimétrica (k_U) utilizando la llave privada del equipo de análisis de resultados (d_R), después se descifran los resultados (r) y la llave privada que genera la firma digital de los resultados (d_{Uf}). Posteriormente se teclean los resultados contenidos en el acta (r_T) y se crea la firma digital (s) de éstos, con esta

firma se forma la llave especial (k_{ESP}) que descifra la llave pública (e_{Uf}) y ésta verifica la integridad de los datos. Una vez que la llave se descifra solo resta verificar la autenticidad del archivo de resultados utilizando la firma que se calculó de los resultados tecleados, éstos al ser iguales a los contenidos en el archivo darán como resultado una verificación válida y se procede a su análisis.

Ventajas.

- No se depende de la seguridad de ninguna llave pública o privada.
- Se pueden enviar los datos a través de un canal de comunicaciones inseguro.
- Es complejo modificar el acta con resultados.

Desventajas.

- Se debe esperar a que llegue el acta al lugar donde se analizan los resultados para poder procesarlos.
- Se debe tener cuidado al momento de teclear los resultados contenidos en el acta para que la firma se genere de manera adecuada, pudiendo resolver este problema con la generación de un código de barras que contenga los resultados.

Opción 2. Envío de la firma a través de un canal de comunicaciones seguro.

El protocolo para esta opción es el siguiente:

$k_U = d_R$ (p)

$r = k_U^{-1}(c)$

$k_{ESP} = s$

$e_{Uf} = k_{ESP}(\hat{}e_{Uf})$

$e_{Uf}(H(r), s)$

Al llegar el conjunto de archivos formado por {c, $\hat{}e_{Uf}$, p} al equipo que recopila y analiza los resultados y una vez que se tenga la firma digital {s}, el procedimiento es el siguiente:

Lo primero es descifrar la llave simétrica (k_U) utilizando la llave privada del equipo de análisis de resultados (d_R), y con ella se descifran los resultados (r), con la firma digital (s) recibida se crea la llave especial (k_{ESP}) que descifra la llave pública (e_{Uf}) que verifica integridad del archivo de resultados.

Ventajas.

- Se envían menos datos.
- No se tiene que esperar a que el acta llegue y que los datos se escriban de manera correcta.

Desventajas.

- Se debe tener un canal de comunicaciones seguro.
- La seguridad depende de que nadie obtenga la firma y la substituya por una propia. Las Figuras 7 y 8 muestran respectivamente el funcionamiento del protocolo en las tres etapas y considerando las dos diferentes opciones para el envío de resultados al equipo analizador de resultados.

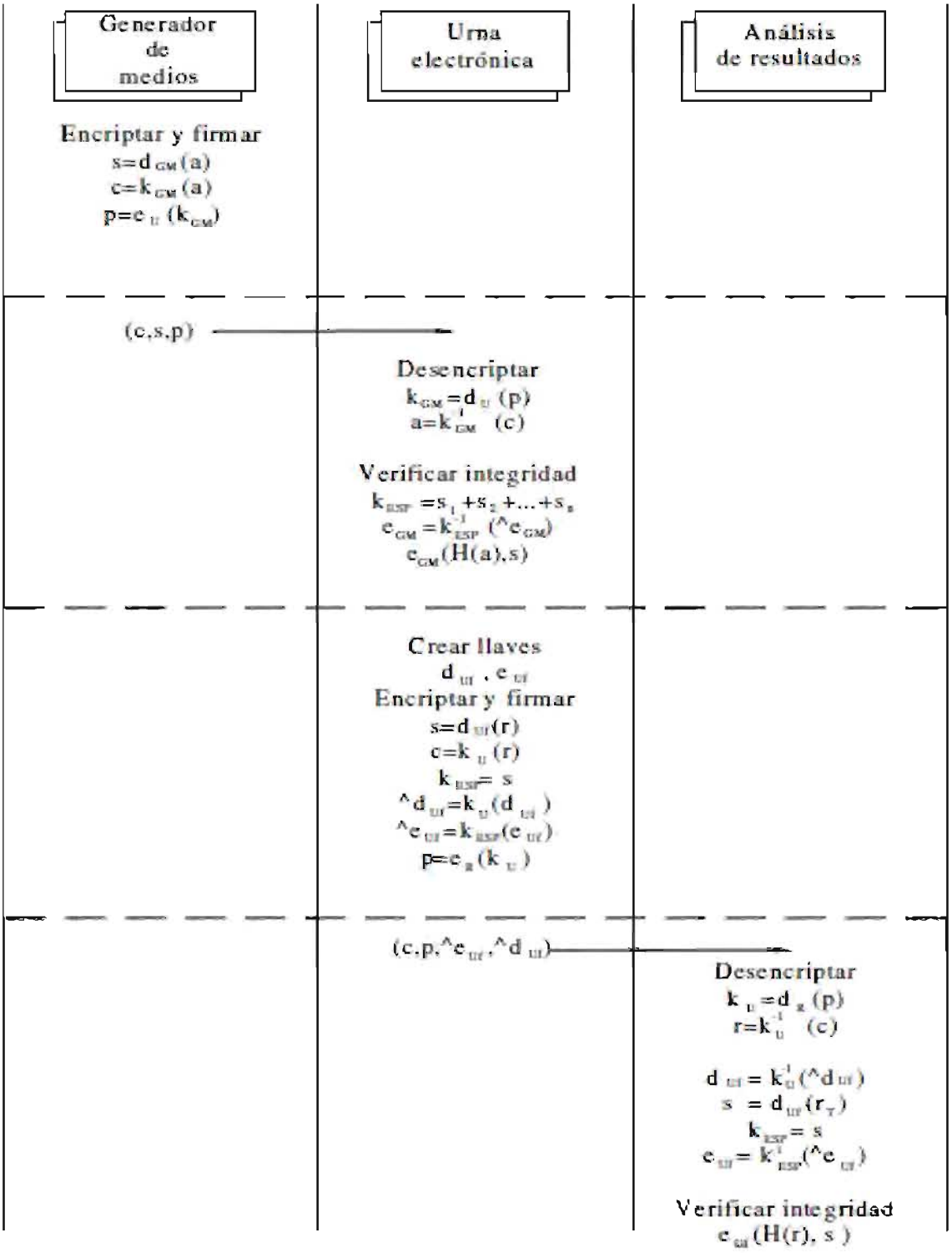


Figura 7. Funcionamiento del protocolo criptográfico para la Opción 1.

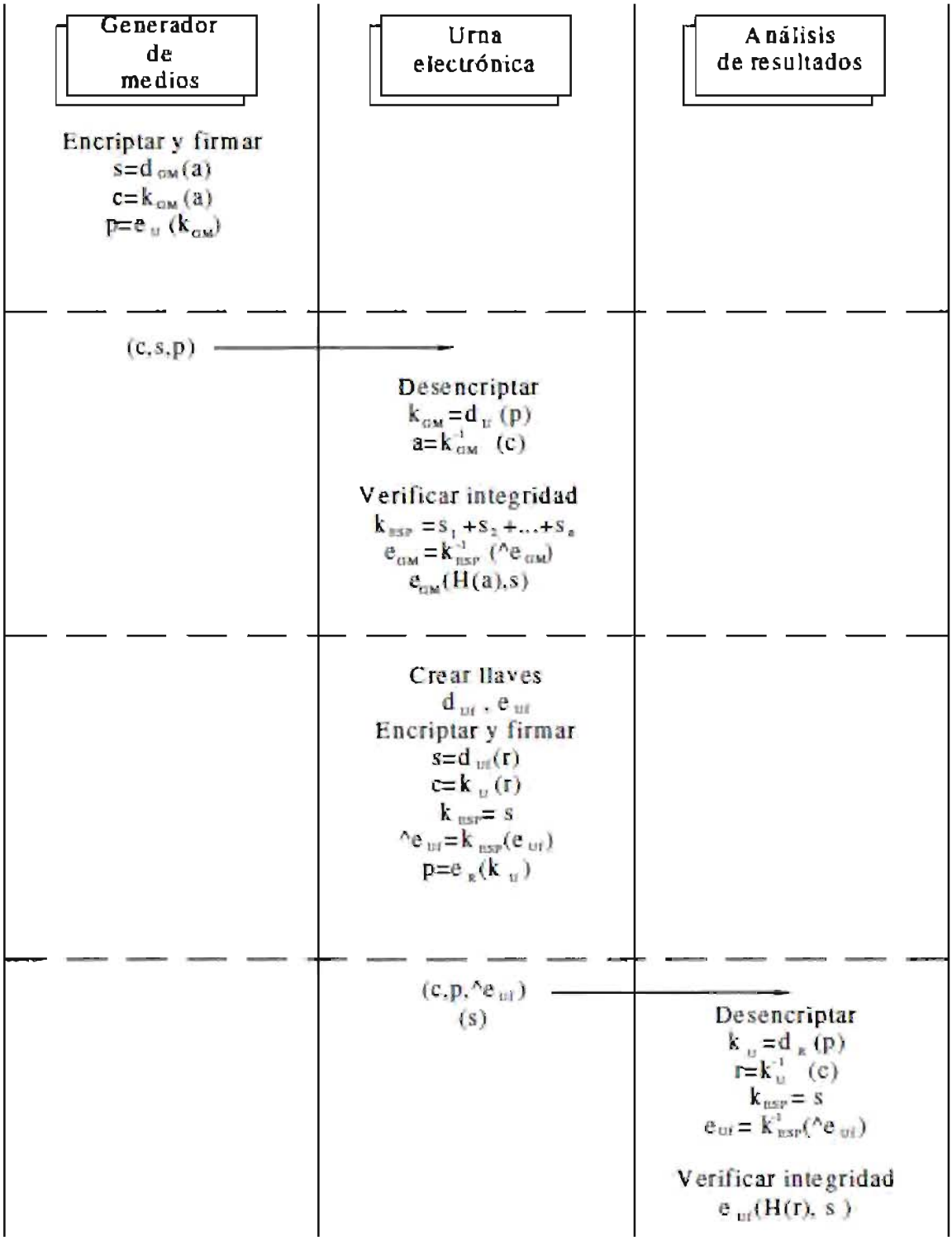


Figura 8. Funcionamiento del protocolo criptográfico para la Opción 2.

3.4. DESARROLLO DE LA ARQUITECTURA

Con base en lo revisado en los antecedentes acerca de la creación de arquitecturas de seguridad, el desarrollo de la arquitectura de seguridad se realizó de la siguiente manera.

Se tomaron los distintos niveles que menciona Probst [Probst, 2002] y se adaptaron a las necesidades del sistema, éste al ser un sistema de voto presencial no requería de la parte de comunicación segura por lo que se eliminó, y se cambio el orden de las capas de Autenticación y Control de acceso, así que los elementos que se consideraron en la arquitectura de seguridad fueron:

- Autenticación.
- Control de Acceso.
    - Sujeto.
    - Objeto seguro.
    - Autorización.
    - Restricciones.
- Criptografía.

[Probst, 2002] menciona a la auditoría como un elemento más de esta arquitectura, pero se trabajó de manera diferente al ser el objetivo del proyecto tener una arquitectura de seguridad y de auditoría, por lo que la auditoría se encuentra presente en cada una de las capas. Para la construcción de la arquitectura se trabajó con las distintas etapas que forman el proceso de votación en el sistema, diseñando una arquitectura para cada una de ellas.

3.4.1. Arquitectura de la etapa de pre votación.

Aquí no se incluyó la parte de autenticación, sólo el control de acceso, la criptografía y la auditoría.

El control de acceso está formado por los siguientes elementos:

Sujeto: El usuario que enciende el equipo.
Objeto seguro: La interfaz de configuración.
Autorización: El permitir el acceso a la etapa de configuración.
Restricciones: La única restricción es la fecha y hora de encendido que debe ser posterior a la que el sistema tiene registrada para comenzar su funcionamiento.

La criptografía está encargada del cifrado de los distintos archivos que se generen. La Figura 9 muestra las capas de esta arquitectura así como los elementos tanto de seguridad como de auditoría que se encuentran en cada una de ellas.

Seguridad	Auditoria
Control de acceso. (Fecha y hora de encendido.)	(Bitácora.)
Criptografía. (Validación de la autenticidad de los datos de configuración. Cifrado de los elementos generados en la parte de auditoría.)	Bitácora. (Comprobantes generados : Funcionamiento de los componentes. Comprobante de validez de los datos. Acta de apertura. Actas iniciales.)

Figura 9. Arquitectura de seguridad y auditoría de la etapa de pre votación.

Una vez que se tienen los elementos que conforman cada capa de la arquitectura es importante conocer cómo interactúan entre ellos durante el proceso que se lleva cabo en esa etapa, esto se muestra en la Figura 10.

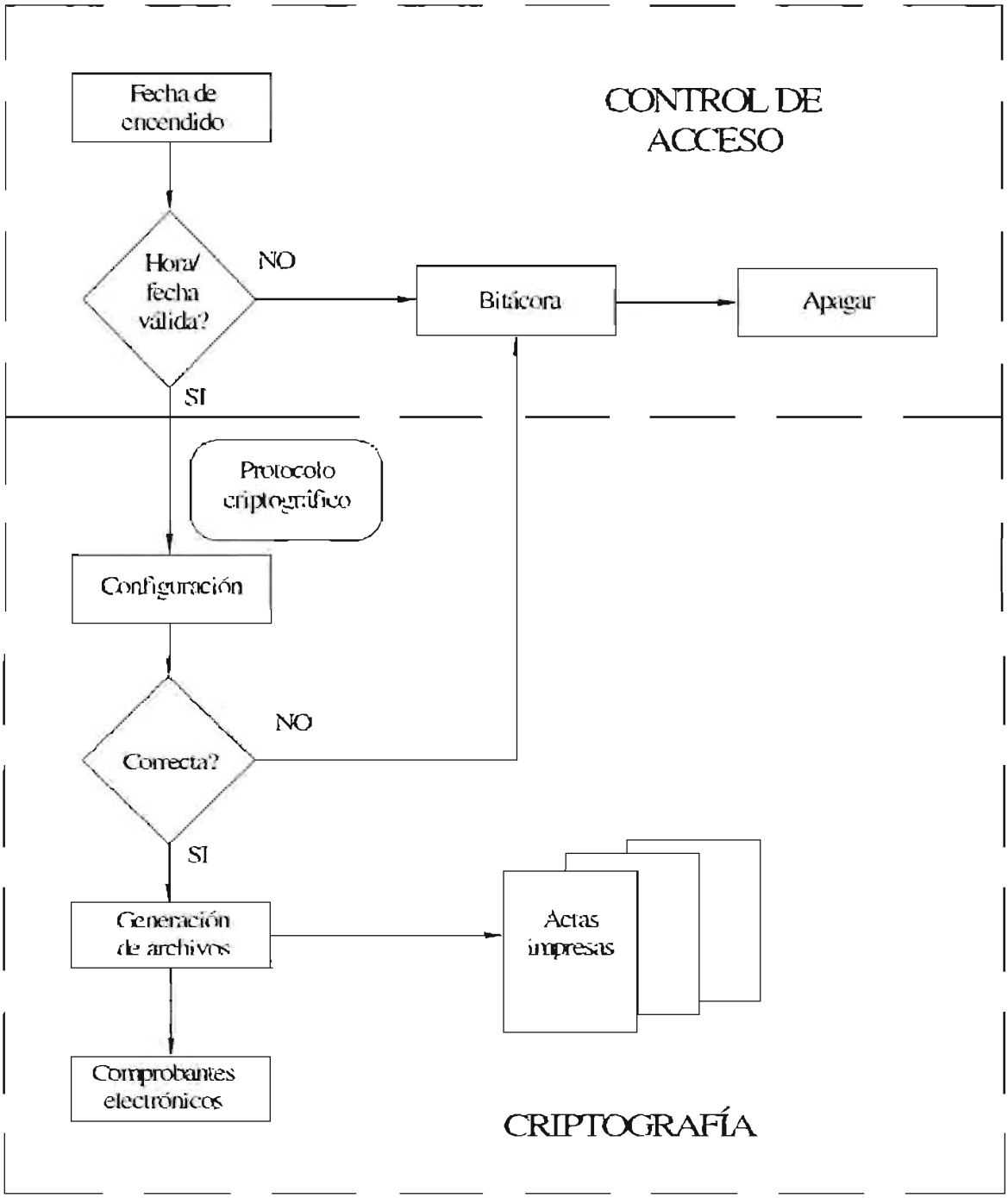


Figura 10. Interacción de los elementos de la arquitectura en la etapa de pre votación.

3.4.2. Arquitectura De La Etapa De Votación.

En esta etapa el método de autenticación consiste en comparar código de barras esperado contra el introducido, ya que ésta es la forma en que se activa el sistema en donde se realizan las pruebas, si coinciden entonces comienza la tarea del control de acceso.

El control de acceso se basa en decidir qué tipo de usuario se encuentra presente, si es un usuario normal o un administrador, al ser la etapa de votación el usuario será un votante normal, los componentes del control de acceso son:

Sujeto: Usuario (Votante).
Objeto seguro: Interfaz de votación que permite el acceso al archivo de votos.
Autorización: Acceso al archivo de votos para poder añadir información.
Restricciones: La restricción es que el votante solo puede participar una vez, ya que terminó sus elecciones el sistema no deberá permitir que éste vote nuevamente. Otras restricciones que podrían existir son relativas al tiempo, por ejemplo que el sistema no permita que el votante vote si se ha tardado más de un determinado número de minutos en realizar su elección o que el sistema no permita más votos después de cierta hora. La Figura 11 muestra las capas de esta arquitectura y sus elementos.

Seguridad	Auditoría
Autenticación. Comparación entre el código esperado y el leído.	(Bitácora.)
Control de acceso. Votante. Administrador.	Bitácora. Deshabilitación del equipo una vez que el usuario ha terminado de votar.
Criptografía. Cifrado de los votos individuales. Descifrado de los votos individuales para conteos parciales. Votos almacenados de manera aleatoria. Cifrado de los elementos producidos en la auditoría.	Almacenamiento en diversos medios. Comprobantes generados : Comprobante impreso. Comprobante electrónico. Archivos de resultados de las auditorías intermedias.

Figura 11. Arquitectura de seguridad y auditoría de la etapa de votación.

La manera en la que interactúan las distintas capas y sus elementos durante el proceso de votación se muestra en la Figura 12.

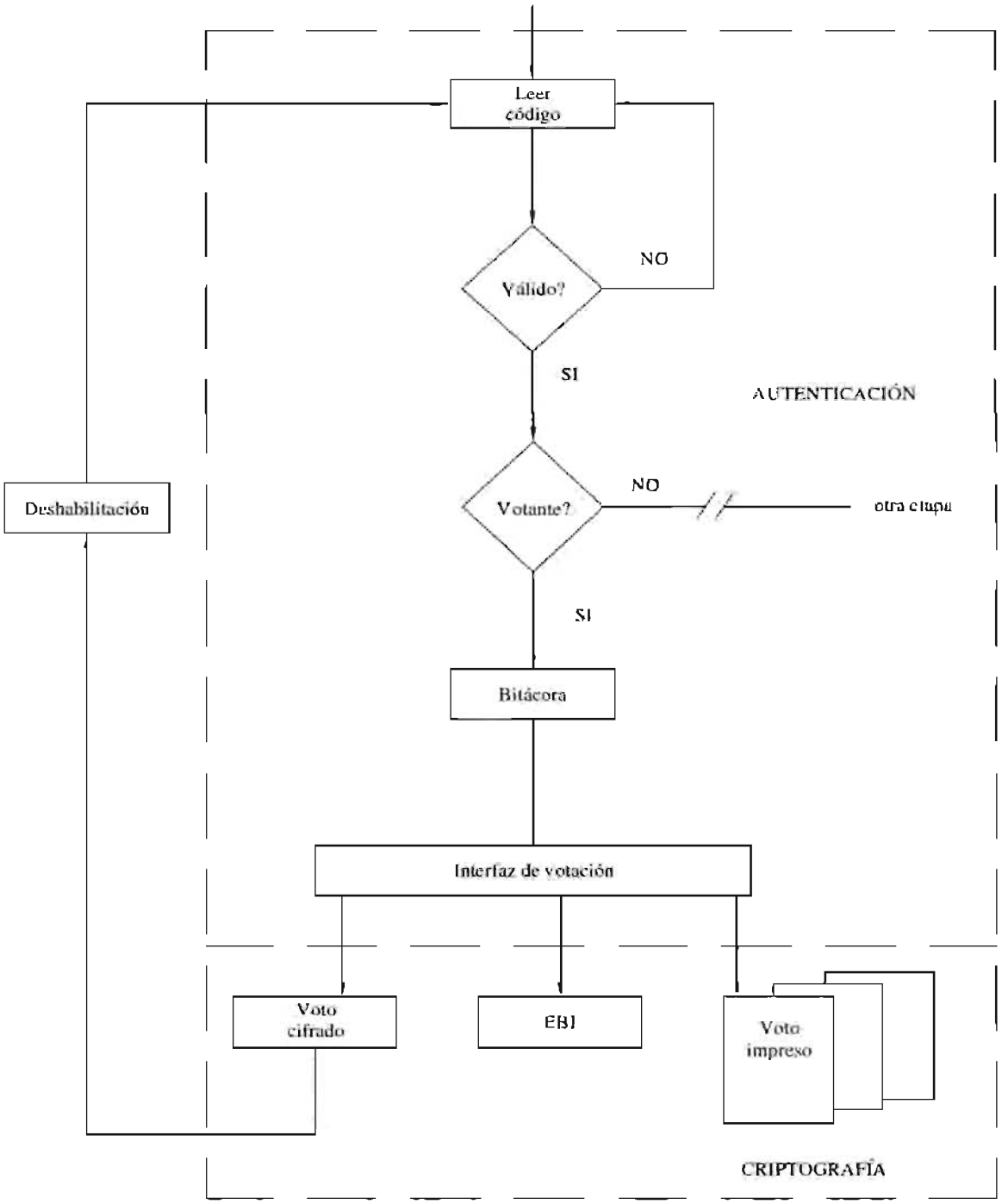


Figura 12. Interacción de los elementos de la arquitectura en la etapa de votación.

3.4.3. Arquitectura de la etapa de post votación.

El método de autenticación es el mismo que en la etapa anterior sólo que esta vez el usuario sería un administrador y no un votante.

El control de acceso se basa en decidir qué tipo de usuario se encuentra presente, si es un usuario normal o un administrador, al ser la etapa de post votación el usuario tendría que ser un administrador, los componentes del control de acceso son:

Sujeto: Usuario (Administrador).
Objeto seguro: Interfaz de administración que permite el acceso al archivo con los votos.
Autorización: Acceso al archivo de votos para poder leer la información y realizar el conteo final.
Restricciones: No se tiene alguna restricción en especial, pudiendo existir una restricción relacionada con el tiempo, por ejemplo que el sistema no permita entrar al módulo de administración sino hasta después de determinada hora.

La criptografía está encargada de cifrar los archivos que se generen incluyendo el que contiene los resultados finales en un formato específico. La Figura 13 muestra la arquitectura para esta etapa y los elementos que se encuentran en cada capa.

Seguridad	Auditoría
Autenticación. (Comparación entre el código esperado y el leído.)	(Bitácora.)
Control de acceso. (Votante. Administrador.)	(Bitácora. Apagado del equipo.)
Criptografía (Descifrado de los votos individuales para el conteo. Cifrado de los elementos producidos en la auditoria. Protocolo criptográfico.)	(Archivo de resultados finales. Comprobantes generados : Actas finales. Acta de cierre de votación. Archivo de resultados.)

Figura 13. Arquitectura de seguridad y auditoría de la etapa de post votación

La manera en que se relacionan los distintos elementos de cada capa en esta arquitectura se muestra en la Figura 14.

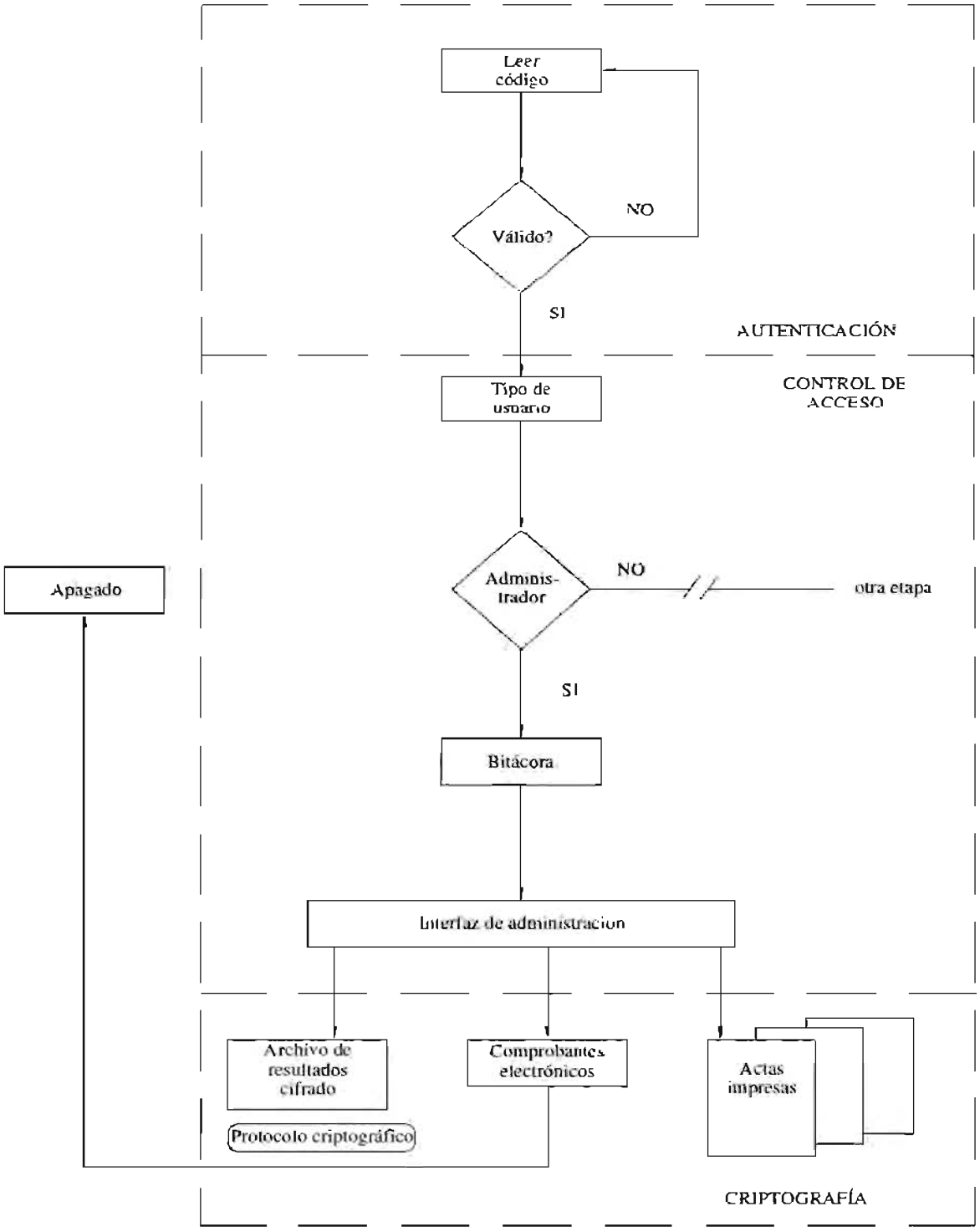


Figura 14. Interacción de los elementos de la arquitectura en la etapa de post votación.

4. PRUEBAS Y RESULTADOS

4.1. PRUEBAS DE LA SEGURIDAD

La primera prueba fue al protocolo criptográfico que se encuentra entre el equipo generador de medios y la urna electrónica. Los algoritmos de cifrado son considerados muy seguros ya que no se pueden alterar los datos tratando de romper la llave con la que fueron cifrados esto de acuerdo a la longitud de llaves mostrada en la Tabla 5 [García et al, 2004].

Nivel de seguridad	Longitud de llave simétrica	Longitud de llaves públicas.
Aceptable (de 5 a 10 años)	80 bits	1024 bits
Buena (posiblemente para siempre)	128 bits	2048 bits
Extrema	192 bits	4096 bits
Muy extrema	256 bits	8192 bits

Tabla 5. Seguridad y longitud de llaves.

Las pruebas consistieron en crear un nuevo conjunto de datos de configuración y de resultados, generar nuevos juegos de llaves y probar los distintos casos de posesión de las llaves originales por parte de los atacantes. La nomenclatura utilizada en las pruebas es la misma que en el desarrollo pudiéndose consultar en la página 46 y 47.

4.1.1. Generador de Medios a Urna Electrónica.

La Tabla 6 muestra los elementos originales, los modificados y los resultados que se obtuvieron en esta etapa.

Caso	Datos originales	Datos modificados	Resultado de la verificación
1	---	d_{GM}, e_U, s, a	ERROR
2	e_U	d_{GM}, s, a	ERROR
3	e_U, s	d_{GM}, a	ERROR
4	e_U, d_{GM}	s, a	ERROR
5	e_U, d_{GM}, s	a	ERROR
6	e_U, d_{GM}, s, a	---	ÉXITO

Tabla 6. Resultados de la prueba para la etapa de generador de medios a la urna.

Caso 1. Suplantación de todos archivos y las llaves.

Datos originales: ---

Datos modificados: d_{GM}, e_U, s, a.

Generación:

$*s = *d_{GM}$ (*a)	Se firman los datos modificados.
$*c = k_{GM}$ (*a)	Se cifran los datos.
$*p = *e_U$ ($*k_{GM}$)	Se cifra la llave simétrica.

Transporte (*s, *c, *p)

Verificación:

$*k_{GM} \neq d_U$ (*p)	**ERROR**: No coincide la llave pública creada por el atacante con la llave privada original.

Caso 2. Se conoce la llave pública original, el resto de los datos son modificados

Datos originales: e_U

Datos modificados: $d_{GM;}$ s, a.

Generación:

$*s = *d_{GM}$ (*a)	Se firman los datos modificados.
$*c = k_{GM}$ (*a)	Se cifran los datos
$*p = e_U$ (*kGM)	Se cifra la llave simétrica.
Transporte (*s, *c, *p)	
Verificación.	
$*k_{GM} = d_U$ (*p)	Coincide la llave pública con la llave privada.
$*a = *k_{GM}$-1 (*c)	Los datos se descifran correctamente.
$e_{GM} \neq *k_{ESP}^{-1}(\hat{}e_{GM})$	**ERROR:** La llave pública no se descifra de manera correcta por que la firma modificada crea una llave especial diferente a la esperada

Caso 3. Se conocen la llave pública y la firma original, el resto de los datos se modifican.

Datos originales: e_U, s.

Datos modificados: $d_{GM;}$ a.

Generación:

$*s = *d_{GM}$ (*a)	Se firman los datos modificados.
$*c = k_{GM}$ (*a)	Se cifran los datos.
$*p = e_U$ ($*k_{GM}$)	Se cifra la llave simétrica.
Transporte (s, *c, *p)	
Verificación:	
$*k_{GM} = d_U$ (*p)	Coincide la llave pública con la llave privada.
$*a = *k_{GM}^{-1}$ (*c)	Los datos se descifran correctamente.

$e_{GM} = k_{ESP}^{-1}(\hat{}e_{GM})$	La llave pública se descifra de manera correcta.
$e_{GM}(*a, s)$	**ERROR:** La verificación no es válida por que los datos modificados no corresponden con la firma original.

Caso 4. Se conocen las llaves pública y privada.

Datos originales: e_U, d_{GM}.

Datos modificados: s, a.

Generación:

$*s = d_{GM}(*a)$	Se firman los datos modificados.
$*c = *k_{GM}(*a)$	Se cifran los datos.
$*p = e_U(*k_{GM})$	Se cifra la llave simétrica.

Transporte (*s, *c, *p)

Verificación:

$k_{GM} = d_U(*p)$	Coincide la llave pública con la llave privada.
$*a = k_{GM}^{-1}(*c)$	Los datos se descifran correctamente.
$e_{GM} \neq k_{ESP}^{-1}(\hat{}e_{GM})$	**ERROR:** La llave pública no se descifra de manera correcta por que la firma modificada no genera la llave especial correcta.

Caso 5. Solo se modifican los archivos de configuración, el resto de los datos y firmas son los originales.

Datos originales: e_U, d_{GM}, s.

Datos modificados: a.

Generación:

$*s = d_{GM} (*a)$	Se firman los datos modificados.
$*c = *k_{GM} (*a)$	Se cifran los datos.
$*p = e_U (*k_{GM})$	Se cifra la llave simétrica.
Transporte (s, *c, *p)	
Verificación:	
$k_{GM} = d_U (*p)$	Coincide la llave pública con la llave privada.
$*a = k_{GM}^{-1} (*c)$	Los datos se descifran correctamente.
$e_{GM} = k_{ESP}^{-1}(\hat{}e_{GM})$	La llave pública se descifra de manera correcta por que la firma original genera la llave correcta.
$e_{GM}(*a, s)$	**ERROR:** La verificación no es válida por que los datos modificados no coinciden con la firma original.

Caso 6. Caso correcto donde no se modifica ningún dato y se usan las llaves originales.

Datos originales: e_U, d_{GM}, s, a.
Datos modificados:---
Generación:

$s = d_{GM} (a)$	Se firman los datos originales.
$c = k_{GM} (a)$	Se cifran los datos.
$p = e_U (k_{GM})$	Se la llave simétrica.
Transporte (s, c, p)	
Verificación:	
$k_{GM} = d_U (p)$	Coincide la llave pública con la llave privada.
$a = k_{GM}^{-1} (c)$	Los datos se descifran correctamente.

$e_{GM} = k_{ESP}^{-1}(\hat{e}_{GM})$	La llave pública se descifra de manera correcta por que la firma original genera la llave correcta.
$e_{GM}(a, s)$	**ÉXITO:** La verificación es válida por que los datos coinciden con la firma original.

4.1.2. Urna Electrónica.

La urna electrónica incluye una parte de las pruebas anteriores, la verificación de la autenticidad y de la integridad de los datos se realiza dentro de la misma, en la etapa de votación sólo se consideraron las propiedades de los algoritmos de cifrado ya que es la parte que menos riesgos corre al no tener algún modo de acceder a la información que contiene, la seguridad incluía el cifrado de los diferentes documentos y de los votos además del almacenamiento aleatorio que no permitía una relación voto-votante, una vez que termina el proceso, los elementos que se generan de acuerdo al protocolo se prueban en otro equipo que es lo que se tendría en un caso real.

4.1.3. Urna Electrónica a Equipo de Analizador de Resultados

Aquí se revisa el funcionamiento del protocolo para las dos opciones que se tienen para enviar los resultados al equipo que los analizará.

Opción 1. Capturando los resultados del acta final.

Los resultados de las pruebas para el envío de datos de la urna electrónica al equipo analizador de resultados utilizando la opción de capturar los resultados contenidos en un acta final se muestran en la Tabla 7.

Caso	Datos originales	Datos modificados	Resultado de la verificación
1	---	e_{Uf}, e_R, d_{Uf}, s, r	ERROR
2	e_{Uf}, e_R	s, r, d_{Uf}	ERROR
3	e_{Uf}, e_R, d_{Uf}	s, r	ERROR
4	e_{Uf}, e_R, d_{Uf}, s	r	ERROR
5	e_{Uf}, e_R, d_{Uf}, s, r	---	ÉXITO

Tabla 7. Resultados para la opción 1 de la urna al equipo analizador de resultados.

Caso 1. Suplantación de todas las llaves y los datos.

Datos originales: ---

Datos modificados: s, r, d_{Uf}, e_{Uf}, e_R.

Generación:

$*s = *d_{Uf}(*r)$	Se firman los resultados modificados.
$*c = k_U(*r)$	Se cifran los resultados.
$\hat{}d_{Uf} = k_U(*d_{Uf})$	Se cifra la llave privada.
$k_{ESP} = *s$	Se crea la llave especial.
$\hat{}e_{Uf} = k_{ESP}(*e_{Uf})$	Se cifra la llave pública.
$p = *e_R(k_U)$	Se cifra la llave simétrica.

Transporte ($*c$, $\hat{}d_{Uf}$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$kU \neq dR$ (p).	**ERROR:** La llave pública no corresponde con la privada.

Caso 2.Uso de las llaves públicas originales, el resto de los datos se modifica.

Datos originales: e_{Uf}, e_R

Datos modificados: s, r, d_{Uf}.

Generación:

$*s = *d_{Uf}(*r)$	Se firman los resultados modificados.
$*c = k_U(*r)$	Se cifran los resultados.
$\hat{}d_{Uf} = k_U(*d_{Uf})$	Se cifra la llave privada.
$k_{ESP} = *s$	Se crea la llave especial.
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública.
$p = e_R(k_U)$	Se cifra la llave simétrica.

Transporte ($*c$, $\hat{}d_{Uf}$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U = d_R(p)$	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
$d_{Uf} = k_U^{-1}(\hat{}d_{Uf})$	La llave privada se descifra de manera correcta.
$s = d_{Uf}(r_T)$	Se leen los resultados del acta y se firman.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} \neq k_{ESP}(\hat{}e_{Uf})$	**ERROR:** La llave especial no es la esperada por que los datos han sido modificados y no puede descifrar a la llave pública.

Caso 3. Uso de las llaves públicas y privadas originales.

Datos originales: e_{Uf}, e_R, d_{Uf}.

Datos modificados: s, r.

Generación:

$*s = d_{Uf}(*r)$	Se firman los resultados modificados
$*c = k_U(*r)$	Se cifran los resultados.
$\hat{}d_{Uf} = k_U(d_{Uf})$	Se cifra la llave privada.
$k_{ESP} = *s$	Se crea la llave especial.
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública.
$p = e_R(k_U)$	Se cifra la llave simétrica.

Transporte ($*c$, $\hat{}d_{Uf}$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U = d_R(p)$	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
$d_{Uf} = k_U^{-1}(\hat{}d_{Uf})$	La llave privada se descifra de manera correcta.
$s = d_{Uf}(r_T)$	Se capturan los resultados del acta y se firman.
$k_{ESP} = s$	Se crea la llave especial.
$eUf \neq k_{ESP}(\hat{}e_{Uf})$	**ERROR:** La llave especial no es la esperada por que los datos han sido modificados y no puede descifrar a la llave pública.

Caso 4. Uso de las llaves pública, privada y firma originales, solo se modifican los resultados.

Datos originales: e_{Uf}, e_R, d_{Uf}, s.

Datos modificados: r.

Generación:

$*s = d_{Uf}(*r)$	Se firman los resultados modificados.
$*c = k_U(*r)$	Se cifran los resultados.
$\hat{}d_{Uf} = k(d_{Uf})$	Se cifra la llave privada.
$k_{ESP} = s$	Se crea la llave especial.

$^e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública.
$p = e_R(k_U)$	Se cifra la llave simétrica.

Transporte (*c, $^d_{Uf}$, $^e_{Uf}$, *p)

Verificación:

$k_U = d_R$ (p)	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
$d_{Uf} = k_U^{-1}(^d_{Uf})$	La llave privada se descifra de manera correcta.
$s = d_{Uf}$ (r_T)	Se leen los resultados del acta y se firman.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} = k_{ESP}(^e_{Uf})$	La llave especial es la esperada y descifra a la llave pública.
e_{Uf}(*r, s)	**ERROR:** La verificación falla por que la firma no corresponde con los resultados modificados.

Caso 5. No se modifican los datos y se usan las llaves originales.

Datos originales: s, r, e_{Uf}, e_R, d_{Uf}.

Datos modificados:---

Generación:

$s = d_{Uf}$ (r)	Se firman los resultados.
$c = k_U(r)$	Se cifran los resultados.
$^d_{Uf} = k(d_{Uf})$	Se cifra la llave privada.
$k_{ESP} = s$	Se crea la llave especial.
$^e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública.
$p = e_R(k_U)$	Se cifra la llave simétrica.

Transporte (c, $^d_{Uf}$, $^e_{Uf}$, p)

Verificación:

$k_U = d_R$ (p)	La llave pública corresponde con la privada.

$r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
$d_{Uf} = k_U^{-1}(\hat{}d_{Uf})$	La llave privada se descifra de manera correcta.
$s = d_{Uf}(r_T)$	Se leen los resultados del acta y se firman.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} = k_{ESP}(\hat{}e_{Uf})$	La llave especial es la esperada y descifra la llave pública.
$e_{Uf}(r, s)$	**ÉXITO:** La verificación es correcta.

Opción 2. Envío de la firma a través de un canal de comunicaciones seguro.

Los resultados de las pruebas para el envío de datos de la urna electrónica al equipo analizador de resultados utilizando la opción de enviar la firma digital a través de un canal de comunicaciones seguro se muestran en la Tabla 8.

Caso	Datos originales	Datos modificados	Resultado de la verificación
1	---	eUf, eR, dUf, s, r	ERROR
2	eUf, eR	s, r, dUf	ERROR
3	eUf, eR, dUf	s, r	ERROR
4	eUf, eR, dUf, s	r	ERROR
5	eUf, eR, dUf, s, r	a	ÉXITO

Tabla 8. Resultados para la opción 2 de la urna al equipo analizador de resultados.

Caso 1. Suplantación de todas las llaves y datos.

Datos originales: ---

Datos modificados: s, r, d_{Uf}, e_{Uf}, e_R.

Generación:

$*s = *d_{Uf}(*r)$	El atacante crea sus propios resultados y los firma con su llave privada.
$*c = k_U(*r)$	Se crea la llave simétrica y se cifran los resultados.
$k_{ESP} = *s$	Se crea la llave especial con la firma modificada.
$\hat{}e_{Uf} = k_{ESP}(*e_{Uf})$	Se cifra la llave pública creada por el atacante con la llave especial.
$p = *e_R(k_U)$	Se cifra la llave simétrica de resultados con la llave pública creada por el atacante.

Transporte ($*c$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U \neq d_R(p)$.	**ERROR:** La llave pública no corresponde con la privada.

Caso 2.Uso de las llaves públicas originales, el resto de los datos se modifica.

Datos originales: e_{Uf}, e_R

Datos modificados: s, r, d_{Uf}.

Generación:

$*s = *d_{Uf}(*r)$	El atacante crea sus propios resultados y los firma con su llave privada.

$*c = k_U(*r)$	Se crea la llave simétrica y se cifran los resultados.
$k_{ESP} = *s$	Se crea la llave especial con la firma modificada.
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública original con la llave especial.
$p = e_R(k_U)$	Se cifra la llave simétrica de resultados con la llave pública original.

Transporte ($*c$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U = d_R(p)$	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
s	Se recupera la firma del canal de comunicaciones seguro.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} \neq k_{ESP}(\hat{}e_{Uf})$	**ERROR:** La llave especial no es la esperada por que la firma original no es igual a la firma modificada y no puede descifrar a la llave pública.

Caso 3. Uso de las llaves públicas y privadas originales, el resto de los datos se modifica.

Datos originales: e_{Uf}, e_R, d_{Uf}.

Datos modificados: s, r.

Generación:

$*s = d_{Uf}(*r)$	El atacante crea sus propios resultados y los firma con la llave privada original.

$*c = k_U(*r)$	Se crea la llave simétrica y se cifran los resultados.
$k_{ESP} = *s$	Se crea la llave especial con la firma modificada.
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública original con la llave especial.
$p = e_R(k_U)$	Se cifra la llave simétrica de resultados con la llave pública original.

Transporte ($*c$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U = d_R\ (p)$	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
s	Se recupera la firma del canal de comunicaciones seguro.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} \neq k_{ESP}(\hat{}e_{Uf})$	**ERROR:** La llave especial no es la esperada por que la firma original no es igual a la firma modificada y no puede descifrar a la llave pública.

Caso 4. Uso de las llaves pública, privadas y de la firma originales, solo se modifican los resultados.

Datos originales: e_{Uf}, e_R, d_{Uf}, s

Datos modificados: r.

Generación:

$*s = d_{Uf}(*r)$	El atacante crea sus propios resultados y los firma con la llave privada original.

$*c = k_U(*r)$	Se crea la llave simétrica y se cifran los resultados.
$k_{ESP} = s$	Se crea la llave especial con la firma original.
$\hat{}e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública original con la llave especial.
$p = e_R(k_U)$	Se cifra la llave simétrica de resultados con la llave pública original.

Transporte ($*c$, $\hat{}e_{Uf}$, $*p$)

Verificación:

$k_U = d_R\ (p)$	La llave pública corresponde con la privada.
$*r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
s	Se recupera la firma del canal de comunicaciones seguro.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} = k_{ESP}(\hat{}e_{Uf})$	La llave especial es la esperada y descifra a la llave pública.
$e_{Uf}(*r, s)$	**ERROR:** La verificación falla por que la firma original no corresponde con los datos modificados

Caso 5. No se modifican los resultados y se utilizan las llaves originales.

Datos originales: s, r, e_{Uf}, e_R, d_{Uf}.

Datos modificados:---

Generación:

$s = d_{Uf}\ (r)$	Se generan los resultados y se firman con la llave privada original.

$c = k_U(r)$	Se crea la llave simétrica y se cifran los resultados.
$k_{ESP} = s$	Se crea la llave especial con la firma original.
$^e_{Uf} = k_{ESP}(e_{Uf})$	Se cifra la llave pública original con la llave especial.
$p = e_R(k_U)$	Se cifra la llave simétrica de resultados con la llave pública original.

Transporte (c, $^d_{Uf}$, $^e_{Uf}$, p)

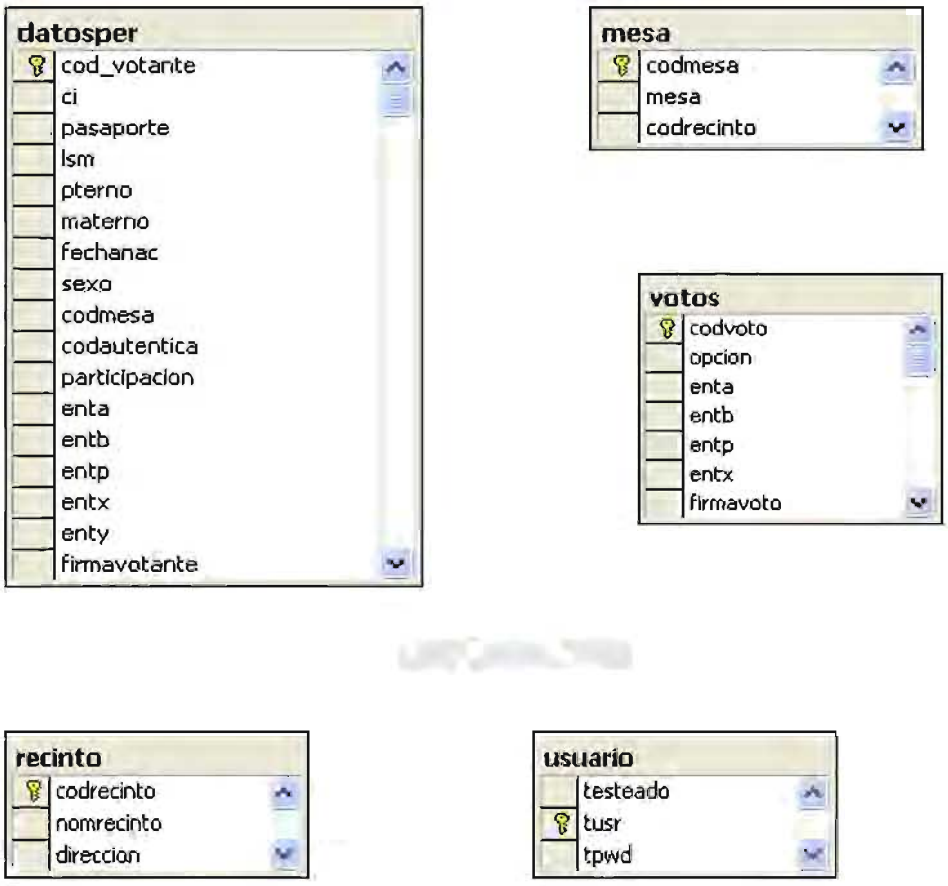
Verificación:

$k_U = d_R(p)$	La llave pública corresponde con la privada.
$r = k_U^{-1}(c)$	Los resultados se descifran de manera correcta.
s	Se recupera la firma del canal de comunicaciones seguro.
$k_{ESP} = s$	Se crea la llave especial.
$e_{Uf} = k_{ESP}(^e_{Uf})$	La llave especial es la esperada y descifra la llave pública.
$e_{Uf}(r, s)$	**ÉXITO:** La verificación es correcta.

Los protocolos diseñados son muy eficientes ya que detectan si los archivos de configuración o de resultados han sido modificados sin importar que el atacante tenga acceso a las llaves públicas o privadas. El crear una llave especial formada de la firma de los datos y el manejo que se da de esta misma es una muy buena medida de seguridad que garantiza que al ser modificados la llave no será correcta y la verificación fallará además el hecho de que ésta llave no se almacene en ningún equipo ni se transporte por ningún canal de comunicaciones inseguro hace que sea imposible el ataque de suplantación.

En el caso de los procedimientos realizados en la urna electrónica, al detectarse un error se muestra la ventana que indica que la validación ha sido incorrecta, se reporta el error en la bitácora y el sistema se apaga. En el equipo que analiza los resultados, solo se cuentan con los programas para descifrar y firmar de manera independiente por lo que solo se muestra el aviso de "La verificación no es correcta".

4.2. BASE DE DATOS



4.3. CAPTURA DE PANTALLAS

Figura 14. Pantalla inicial. *Muestra el acceso al sistema por parte de los Administradores como los usuarios votantes*

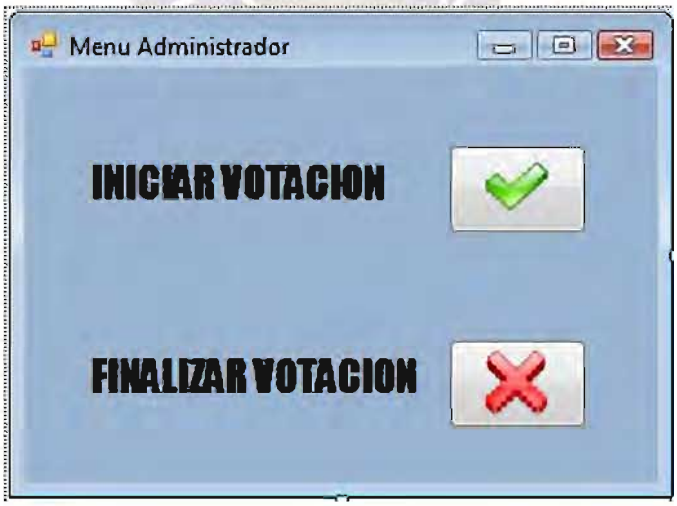


Figura 15. Menu de Administrador. *Muestra el inicio y el cierre de la votación.*

Figura 15. Menu de Administrador. *Muestra la información especifica la cual debe de ser correcta para proceder a la etava de votación*

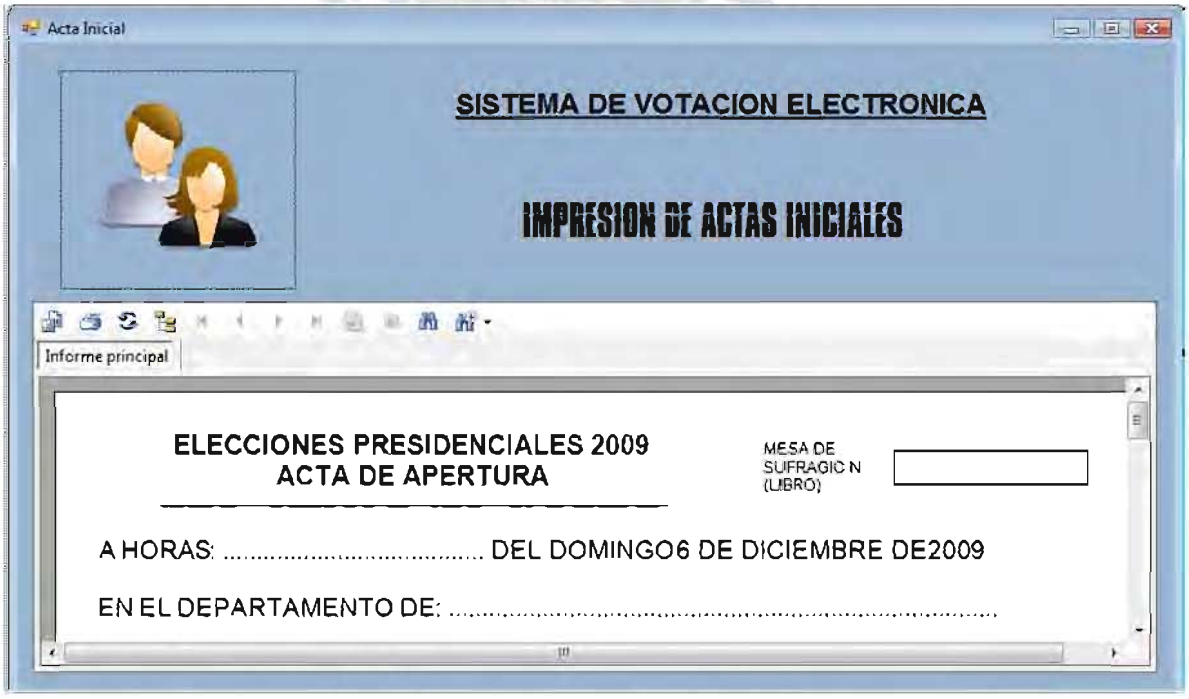


Figura 16. Impresión del acta inicial. *Pantalla para la impresión de actas iniciales, de apertura y de comprobante de componentes.*

Figura 17. Bienvenida al usuario. *Pantalla que el usuario observa, procede a entrar al modulo de votacion*

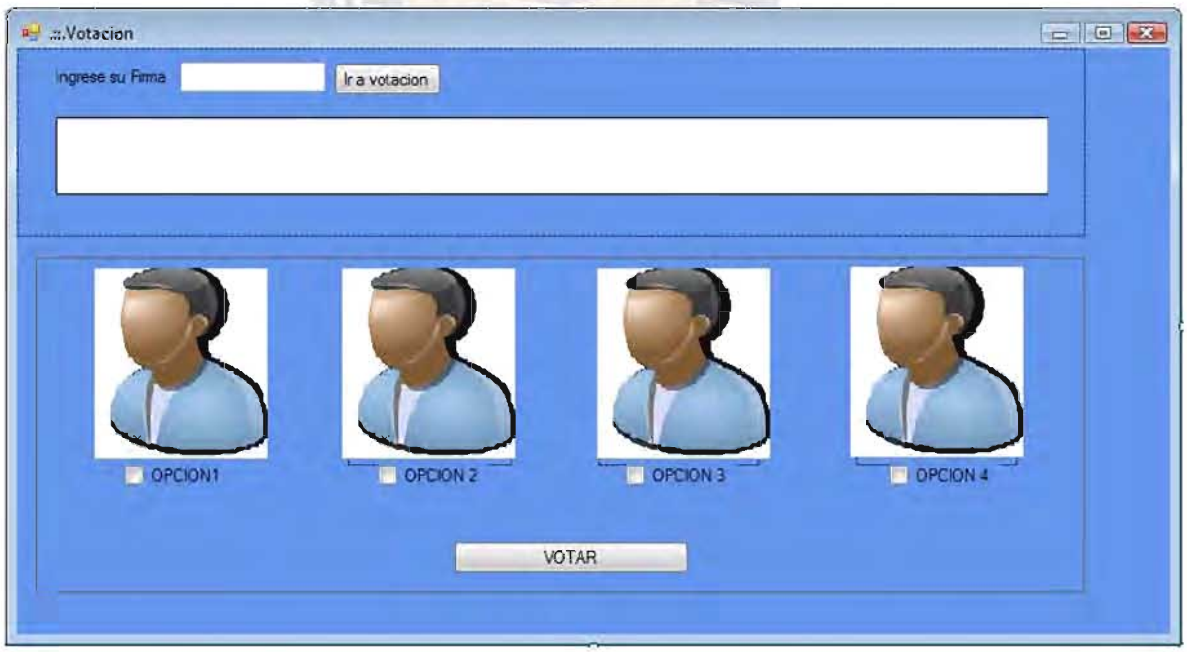


Figura 18. Pantalla de votación del usuario. *Aquí el usuario puede elegir una opción para poder votar*

Figura 19. Confirmar o Corregir el voto. *El usuario puede corregir o confirmar su voto antes de imprimir su voleta de sufragio*

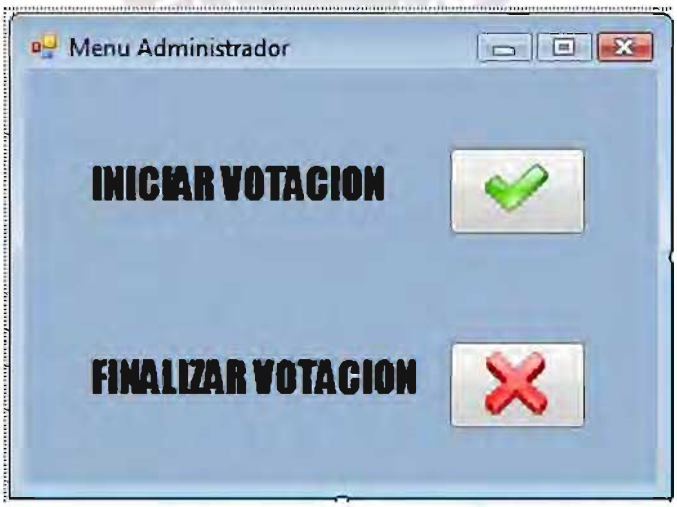


Figura 20. Menú del administrador. *Pantalla que utilizan los funcionarios para la finalización de la jornada electoral, tiene la opción de regresar a la etapa de votación.*

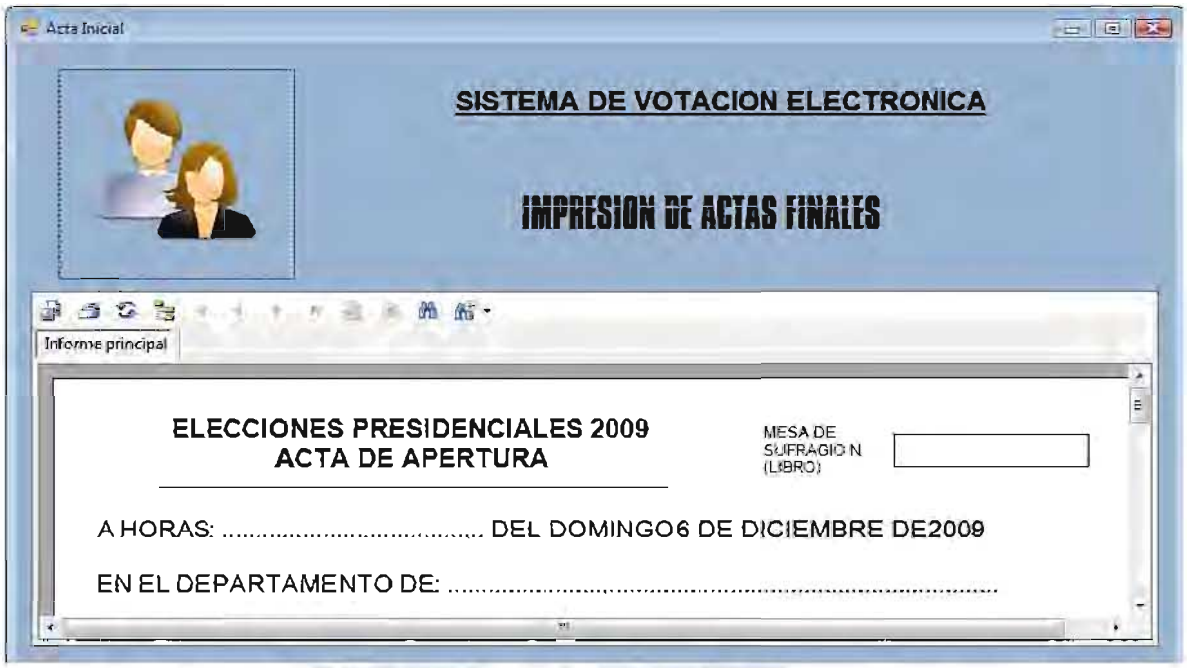


Figura 21. Impresión de actas finales. *Pantalla para imprimir las actas finales con determinado número de copias.*

Figura 22. Llave privada.

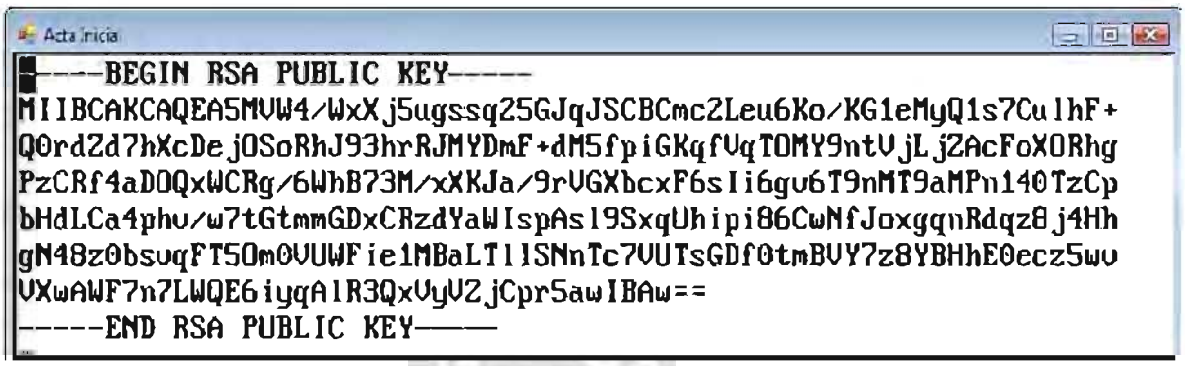


Figura 23. Llave pública en formato PEM

Figura 24 Pantalla Ejemplo de Encriptar y Desencriptar Datos.

Figura 25 Pantalla de Encriptar Datos.

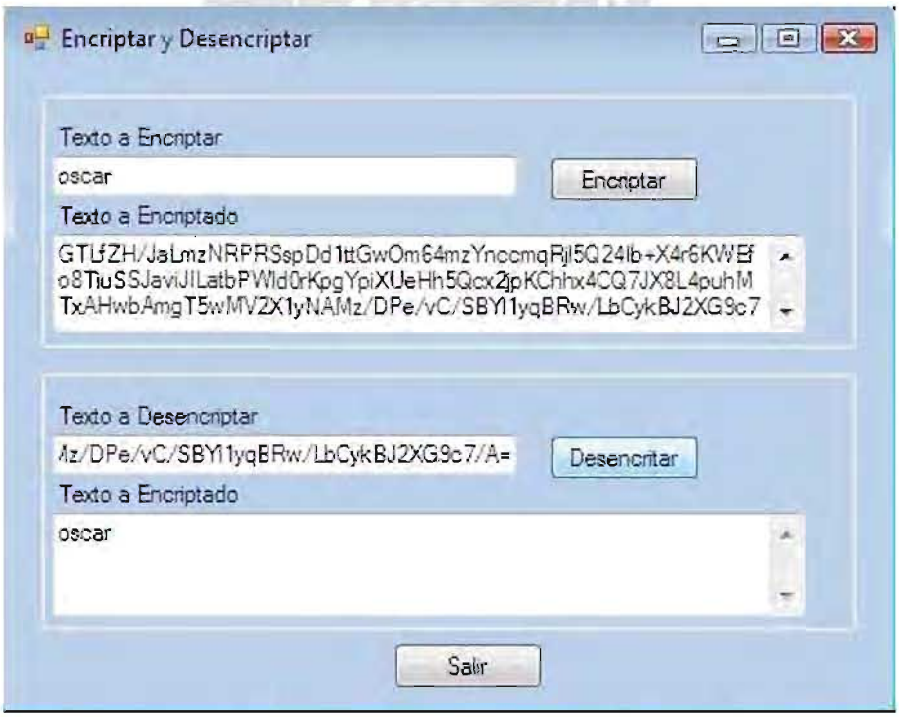


Figura 26 Pantalla de Desencriptar Datos.

5. CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

Seguridad.

Si bien los votos son el elemento fundamental de estos sistemas, existen otros elementos que se deben proteger que son los archivos críticos que incluyen a los archivos de configuración y de resultados que en este tipo de sistemas son los que corren mayor riesgo al ser transportados por canales de comunicación considerados inseguros. En cuanto al manejo de las llaves, el generar las simétricas en el momento que se requieren utilizando métodos que tienen un buen nivel de entropía que se traduce en una alta aleatoriedad y con un adecuado tamaño de llave para ambos tipos de cifrado garantiza que el sistema será seguro por mucho tiempo. Los protocolos criptográficos que se diseñaron demostraron ser eficientes al no depender de la seguridad de ninguna de las llaves privadas utilizadas, el tomar como última medida de verificación la integridad de los datos creados originalmente permíte detectar una modificación a los datos sin importar que el atacante cree su propio conjunto de llaves y archivos o que llegue a tener acceso a los originales. El instalar con anterioridad las llaves en los distintos equipos las protege contra el ataque de suplantación, ésta instalación debe seguir ciertas normas para realizarse como puede ser el que todos los involucrados en la elección estén presentes para evitar que las llaves sean alteradas al momento de su instalación.

Sobre el desarrollo, utilizar Microsoft *Visual Studio* permite generar primitivas criptográficas adecuadas ya que cuenta con algoritmos que han sido probados por expertos y que son considerados muy seguros, teniendo eso como base la construcción de los protocolos que es dónde se presentan la mayor cantidad

de ataques se realizó sobre elementos seguros lo que permitió el desarrollo de los protocolos sin la necesidad de preocuparse por las primitivas. Gracias al tipo de implementación que se tiene con Microsoft *Visual Studio* basada en el uso de librerías y llamadas a funciones que sólo deben llenarse con los datos requeridos, hace que el realizar cambios de algoritmo de cifrado sea muy sencillo teniendo en la mayoría de las ocasiones que cambiar solamente el nombre del algoritmo o la función que se utiliza.

Arquitectura.

La arquitectura diseñada cumple satisfactoriamente con el objetivo principal para el que fue creada, que era resolver el problema de la plataforma segura ya que cubre los puntos vulnerables de éste tipo de sistemas desde el punto de vista de la seguridad y de la auditoría especificando claramente los elementos que se deben encontrar en cada una de las diferentes capas que la conforman y la interacción que debe existir entre ellos, lo que permite realizar el diseño e implementación de sistemas de voto electrónico que generen altos niveles de confianza en el público que los utiliza. La arquitectura cubre completamente los puntos vulnerables tanto de seguridad como de auditoría y los que se cubren parcialmente son debido a que no fueron tratados durante el desarrollo del proyecto.

Auditoría.

El trabajar con el concepto de redundancia como base garantiza que se tengan diversas versiones de un mismo elemento, ésto hace al sistema más confiable además de permitir detectar si se han cometido alteraciones durante la votación gracias a elementos como las auditorías intermedias y los comprobantes almacenados en diversos formatos y medios. Los elementos a incluir deben ser de acuerdo a las necesidades en cada etapa, de acuerdo a esto, la arquitectura de auditoría diseñada es eficiente ya que permite que en la

etapa de pre votación se tengan elementos que permiten comprobar el correcto funcionamiento del equipo y de la configuración del sistema, para la etapa de votación tener elementos que incrementen la confianza de los votantes al momento en que están utilizando el sistema además de contar con medios para verificar que todo el proceso se está llevando a cabo de manera adecuada, finalmente para la etapa de post votación se tienen suficientes elementos que contienen los resultados finales en distintos formatos lo que permite poder realizar una comparación de los resultados registrados en cada uno de ellos, además de permitir un recuento de los votos ya sea de manera manual utilizando los *VVPAT* o en otro equipo usando los *EBI*. En el caso de la bitácora, éste documento contiene información sobre todos los eventos que ocurren en el equipo, la versión que se desarrolló además permite realizar un recuento sobre el total de votos registrados en cada elección.

Trabajo con el sistema.

Si bien se puede pensar que las tres etapas de un sistema de voto electrónico están muy relacionadas entre si, separarlas permite trabajar de una manera modular lo que facilita el manejo de un sistema que en conjunto podría resultar muy complejo, la único que se debe considerar en la relación entre las etapas son los archivos que entrega una de ellas como salida y que recibe otra como entrada así como las restricciones que se deben tener y las condiciones que se deben cumplir para poder pasar de una etapa a la otra.

5.2. RECOMENDACIONES

Se puede utilizar la arquitectura diseñada como base para la creación de una que cubra las etapas de generación de archivos y de análisis de resultados. Si bien el protocolo que se diseñó cubre la seguridad y la integridad de los datos que se obtienen de la etapa de generación de archivos no se trabajan los

lementos de seguridad y auditoría que ésta debe poseer al momento de su creación, de manera similar con la etapa de análisis de resultados, la arquitectura diseñada maneja la seguridad de los datos que recibe, pero se debe crear una arquitectura más específica para cuando se está realizando el análisis de los mismos.

Dentro de lo que fue la etapa de votación no se trató lo referente a la activación del equipo, la identificación y la validación de usuarios, éste es otro punto sobre el que se puede trabajar, revisando como se activa el equipo, si existe otro sistema que lo active, cómo tener una comunicación segura entre ellos de manera que no haya otro equipo que pueda activarlo, el manejo que se le debe dar a la base de datos que contenga los usuarios válidos, qué información contenida en ésta misma debe estar protegida y evitar que sea modificada.

Como trabajos futuros para el proyecto desarrollado se tiene el incremento de la calidad del código fuente, que consiste en reducir el número de advertencias que se generan al analizarlo, desarrollar programas y configurar la impresora para permitir la generación de códigos de barra, para el control de la hora de encendido del equipo se debe eliminar la dependencia del reloj del sistema teniendo como opción tomar la hora de algún dispositivo externo como un reloj o un GPS, hacer que el sistema sea totalmente configurable a partir de información contenida en archivos de texto y trabajar con lo que es la seguridad propia del equipo donde se realiza la votación, que incluye seguridad física, del sistema operativo y de la aplicación.

Finalmente como aportes a sistemas que no son de voto electrónico, es conveniente que los conceptos que se manejan se apliquen a otro tipo de aplicaciones para que éstas cuenten con elementos que permitan verificar que su funcionamiento y los resultados que arrojan son adecuados, por ejemplo en

una aplicación bancaria se debe tener una manera de demostrar que las operaciones fueron realizadas de manera adecuada o en algún programa de análisis o procesamiento de datos se debe poder verificar que los resultados son congruentes con lo esperado, esto además de incluir las recomendaciones sobre la calidad del código fuente.

BIBLIOGRAFIA

- [Arango & Sánchez, 2004]
  ARANGO, R. SANCHEZ, F. *Voto electrónico,* Todo Linux No 30, 2004.
- [Bidan, 1997]
  BIDAN, C. ISSARNY, V. *Security benefits from Software Architecture,* 1997.
- [Bolin, 2005]
  BOLIN, R. KATZ, E. *Electronic voting machines and the standards-setting process,* 2005.
- [Boneh, 1999]
  BONEH, D. *Twenty Years of Attacks Against the RSA Crypto-system,* Notices of the American Mathematics Society, 5(2), 1999.
- [Cranor & Cytron, 1997]
  CRANOR, L. CYTRON, R. *Secure Voting Systems,* Proceedings of the Hawai`s International Conference on System Sciences Hawaii U.S.A, 1997.
- [FEPADE, 2004]
  FEPADE, *La urna electrónica: Avances y Perspectivas,* Simposium sobre Urnas Electrónicas organizado por el IEDF realizado en México D.F, 2004.
- [Fischer, 2003]
  FISCHER, E. *Election reform and electronic voting systems (DREs): Analysis of Security Issues*, CRS Report for Congress, 2003.
- [García et al., 2004]
  GARCIA, L. MORALES, G. GONZALEZ, S. *Implementación del algoritmo RSA para su uso en el voto electrónico*, Simposium sobre Urnas Electrónicas organizado por el IEDF realizado en México D.F, 2004.
- [Graff, 2003]
  GRAFF, M. VAN W, K, *Secure Coding*, 2003.
- [IPL, 1997]

IPL. *Software testing and software development lifecycles*, 1997.

- [Jones, 2004A]
  JONES, D. *Auditing Elections*, Communications of the ACM Vol 47. No 10, pp 46-50, 2004.
- [Jones, 2004B]
  JONES, D. *Parallel Testing: A menu of options*, 2004.
- [Kohno et al., 2004]
  KOHNO, T. STUBBLEFIELD, A. RUBIN, A. WALLACE, D. *Analysis of an Electronic Voting System*, IEEE Symposium on Security and Privacy. 2004.
- [Menezes, 1996]
  MENEZES, A. OORSCHOT, P. VANSTONE, S. *Handbook of Applied Cryptography*. 1996.
- [Mercuri, 2002]
  MERCURI, R. *A better ballot box*, 2002.
- [Prince, 2004]
  PRINCE, A. *Consideraciones, aportes y experiencias para el voto electrónico en Argentina*, Buenos Aires. 2004.
- [Probst, 2002]
  PROBST, S. ESSMAYR, W. WEIOOL, E. *Reusable Components for eveloping Security-Aware Applications*, 2002.
- [Rivest, 2001]
  RIVEST, L. *Electronic Voting*, 2001.
- [Saltman, 2001]
  SALTMAN, R. *Auditability and Voter Confidence in Direct Recording (DRE) Voting Systems*, 2001.
- [Saltman, 2003]
  SALTMAN, R. *Auditability of non-ballot, poll-site voting systems*, 2003.
- [Selker, 2003]
  SELKER, T. GOLLER, J. *The SAVE System: Secure Architecture for Voting*

Electronically, BT Technology Journal Vol 22 No 44, pp 89-95, 2003.

SITIOS WEB DE INTERÉS.

OpenSSL.
http://www.openssl.org/
Flawfinder.
http://www.dwheeler.com/flawfinder/
RATS.
http://www.securesoftware.com/resources/download_rats.html
QTDesigner.
http://www.trolltech.com/download/opensource.html
QTEmbedded.
ftp://ftp.trolltech.com/qt/source/qt-embedded-free-3.1.2.tar.bz2
Compilador cruzado *arm-linux-gcc*
http://www.applieddata.net/developers/linux/files/tools/arm-linux-gcc-3.3.2.tar.bz2
Doxygen.
http://www.stack.nl/~dimitri/doxygen/index.html
Descarga de paquetes de *Linux Debian* para distintas arquitecturas.
http://www.debian.org/distrib/packages
Página de los fabricantes de la *BitsyX*.
http://www.applieddata.net/

ANEXOS

ESTANDARES PARA LOS SISTEMAS DE VOTO ELECTRÓNICO

Los estándares permiten certificar la exactitud, la legalidad y la confianbilidad de los distintos sistemas de votación que pudieran adquirir, contratar o desarrollar los estados de la Unión. El documento ha sido sometido al análisis público y a la opinión de empresas, funcionarios, académicos, expertos técnicos y grupos de defensa de los intereses públicos, con su consiguiente impacto sobre el documento final.

Los estándares se dividen en dos volúmenes: I) Estándares sobre la capacidad funcional de sistema. II) Documentación que se exige para obtener la certificación. La certificación sera expedida por una autoridad independiente. A continuación ofrecemos el índice de los estándares y la introducción (en inglés) del documento original.

1. INTRODUCTION

1.1 Objectives and Usage of the Voting System Standards

State and local officials today are confronted with increasingly complex voting system technology and an increased risk of voting system failure. Responding to calls for assistance from the states, the United States Congress authorized the Federal Election Commission (FEC) to develop voluntary national voting systems standards for computer-based systems. The resulting FEC Voting System Standards ("the Standards") seek to aid state and local election officials in ensuring that new voting systems are designed to function accurately and reliably, thus ensuring the system's integrity. States are free to adopt the Standards in whole or in part. States may also choose to enact stricter performance requirements for systems used in their jurisdictions.

The Standards specify minimum functional requirements, performance characteristics, documentation requirements, and test evaluation criteria. For the most part, the Standards address what a voting system should reliably do, not how system components should be configured to meet these requirements. It is not the intent of the Standards to impede the design and development of new, innovative equipment by vendors. Furthermore, the Standards balance risk and cost by requiring voting systems to have essential , but not excessive, capabilities.

The Standards are not intended to define appropriate election administration practices. However, the total integrity of the election process can only be ensured if implementation of the Standards is coupled with effective election administration practices.

The Standards are intended for use by multiple audiences to support their respective roles in the development, testing, and acquisition of voting systems:

- Authorities responsible for the analysis and testing of such systems in support of qualification and/or certification of systems for purchase within a designated jurisdiction;
- State and local agencies evaluating voting systems to be procured within their jurisdictions; and
- Designers and manufacturers of voting systems.

1.2 Development History for Initial Standards

1.2 DEVELOPMENT HISTORY FOR INITIAL STANDARDS

Much of the groundwork for the Standards' development was laid by a national study conducted in 1975 by the National Bureau of Standards, now known as the National Institute of Standards and Technology (NIST). This study was requested by the FEC's Office of Election Administrator's predecessor, the Office of Federal Elections of the General Accounting Office. The report, "Effective Use of Computing Technology in Vote-Tallying," made a number of recommendations bearing directly on the Standards project. After analyzing computer-related election problems encountered in the past, the report concluded that one of the basic causes for these difficulties was the lack of appropriate technical skill at the state and local level for developing or implementing sophisticated and complex standards against which voting system hardware and software could be tested.

Following the release of this report, Congress mandated that the FEC, with the cooperation and assistance of the National Bureau of Standards, study and report on the feasibility of developing "voluntary engineering and procedural performance standards for voting systems used in the United States." (2 U.S.C. §431 Note) The resulting 1983 study cited a substantial number of technical and managerial problems that affected the integrity of the vote counting process. It also asserted the need for a federal agency to develop national performance standards that could be used as a tool by state and local election officials in the testing, certification, and procurement of computer-based voting systems. In 1984, Congress approved initial funding for the Standards.

The FEC held a series of public hearings in developing the initial Standards. State and local election officials, election system vendors, technical consultants, and others reviewed drafts of the proposed criteria. The FEC considered their many comments and made appropriate revisions. Before final issuance, the FEC publicly announced the availability of the latest draft of the Standards in the Federal Register and requested that all interested parties submit final comments. The FEC meticulously reviewed all responses to the notice and incorporated corrections and suitable suggestions.

Ultimately, the final product was the result of considerable deliberation, close consultation with election officials, and careful consideration of comments from all interested parties.

In January 1990, the FEC Issued the performance standards and testing procedures for punchcard, marksense, and direct recording electronic (DRE) voting systems. The Standards did not cover paper ballot and mechanical lever systems because paper ballots are sufficiently self-explanatory not to require technical standards and mechanical lever systems are no longer manufactured or sold In the United States. The FEC also did not incorporate requirements for mainframe computer hardware because it was reasonable to assume that sufficient engineering and performance criteria already governed the operation of mainframe computers. However, vote tally software installed on mainframes is covered by the Standards.

1.3 UPDATE OF THE STANDARDS

Today, over two-thirds of the States have adopted the Standards in whole or In part. As a result, the voting systems marketed today are dramatically improved. Election officials are better assured that the voting systems they procure will work accurately and reliably. Voting system failures are declining, and now primarily involve pre-Standard equipment, untested equipment configurations, or the mismanagement of tested equipment. Overall, systems integrity and the election processes have Improved markedly.

However, advances in voting technology, legislative changes, and the proliferation of electronic voting systems make an update of the Standards necessary. The Industry has been marked by widespread integration of personal computer technology and non-mainframe servers Into DRE voting systems.

In addition, voting systems need to be responsive to the Americans with Disabilities Act (ADA) of 1990 and guidelines developed to assist in implementing the ADA.

1.4 ACCESSIBILITY FOR INDIVIDUALS WITH DISABILITIES

Voters and election officials who use voting systems represent a broad spectrum of the population, and include individuals with disabilities who may have difficulty using traditional voting systems. In developing accessibility provisions for the Standards, the FEC requested assistance from the Access Board, the federal agency in the forefront of promulgating accessibility provisions. The Access Board submitted technical standards

designed to meet the diverse needs of voters with a broad range of disabilities. The FEC has adopted the entirety of the Access Board's recommendations and incorporated them into the Standards. These recommendations comprise the bulk of the accessibility provisions found in Section 2.2.7. Implementing these provisions, however, will not entirely eliminate the need to accommodate the needs of some disabled voters by human interface.

The FEC anticipates that during the lifetime of this version of the Standards increased obligations will be placed upon election officials at every jurisdictional level to provide voting equipment tailored to meet the needs of voters with disabilities. To facilitate jurisdictions in meeting accessibility needs, the Standards mandate that every voting system incorporate some accessible voting capabilities. The Standards also mandate that systems incorporating a DRE component meet specific technological requirements. To do so, it is anticipated that a vendor will have to either configure all of the system's voting stations to meet the accessibility specifications or will have to design a unique station that conforms to the accessibility requirements and is part of the overall voting system configuration.

Under no circumstances should compliance with requirements for accessibility be viewed as mutually exclusive from compliance with any other provision of the Standards. If a voting system contains a machine uniquely designed to meet the accessibility requirements, such a machine will be tested for compliance with the accessibility requirements, as well as for compliance with all of the DRE standards, in order to ensure that an accessible machine does not unintentionally abrogate the mandates of the Standards.

1.5 DEFINITIONS

The Standards contain terms describing function, design, documentation, and testing attributes of equipment and computer programs. Unless otherwise specified, the intended sense of technical terms is that which is commonly used by the information technology industry. In some cases terminology is specific to elections or voting systems, and a glossary of those terms is contained in Appendix A. Non-technical terms not listed in Appendix A shall be interpreted according to their standard dictionary definitions.

Additionally, the following terms are defined below:

- Voting system;
- Paper-based voting system;
- Direct record electronic (DRE) voting system;
- Public network direct record electronic (DRE) voting

systems;

- Precinct count voting system; and
- Central count voting system.

1.5.1 Voting System

A voting system is a combination of mechanical, electromechanical, or electronic equipment. It includes the software required to program, control, and support the equipment that is used to define ballots; to cast and count votes; to report and/or display election results; and to maintain and produce all audit trail information. A voting system may also include the transmission of results over telecommunication networks.

Additionally, a voting system includes the associated documentation used to operate the system, maintain the system, identify system components and their versions, test the system during its development and maintenance, maintain records of system errors and defects, and determine specific changes made after system qualification. By definition, this includes all documentation required in Section 9.4.

Traditionally, a voting system has been defined by the mechanism the system uses to cast votes and further categorized by the location where the system tabulates ballots. However, the Standards recognize that as the industry develops unique solutions to various challenges and as voting systems become more responsive to the needs of election officials and voters, the rigid dichotomies between voting system types may be blurred. Innovations that use a fluid understanding of system types can greatly improve the voting system industry, but only if controls are in place to monitor and control integrity through the proper evaluation of the system brought for qualification.

As such, vendors that submit a system that integrates components from more than one traditional system type or a system that includes components not addressed in this Standard shall submit the results of all beta tests of the new system. Vendors also shall submit a proposed test plan to the appropriate independent test authority recognized by the National Association of State Election Directors (NASED) to conduct national qualification testing of voting systems. The Standards permit vendors to produce or utilize interoperable components of a voting system that are tested within the full voting system configuration.

1.5.2 Paper-Based Voting System

A Paper-Based Voting System, (referred to in the initial Standards as a Punchcard and

Marksense [P&M] Voting System) records votes, counts votes, and produces a tabulation of the vote count from votes cast on paper cards or sheets. A punchcard voting system allows a voter to record votes by means of holes punched in designated voting response locations. A marksense voting system allows a voter to record votes by means of marks made by the voter directly on the ballot, usually in voting response locations. Additionally, a paper based system may record votes using other approaches whereby the voter's selections are indicated by marks made on a paper ballot by an electronic input device, as long as such an input device does not independently record, store, or tabulate the voters selections.

1.5.3 Direct Record Electronic (DRE) Voting System

A Direct Record Electronic (DRE) Voting System records votes by means of a ballot display provided with mechanical or electro-optical components that can be activated by the voter; that processes data by means of a computer program; and that records voting data and ballot images in memory components. It produces a tabulation of the voting data stored in a removable memory component and as printed copy. The system may also provide a means for transmitting individual ballots or vote totals to a central location for consolidating and reporting results from precincts at the central location.

1.5.4 Public Network Direct Record Electronic (DRE) Voting System

A Public Network Direct Record Electronic (DRE) Voting System is an election system that uses electronic ballots and transmits vote data from the polling place to another location over a public network as defined in Section 5.1.2. Vote data may be transmitted as individual ballots as they are cast, periodically as batches of ballots throughout the election day, or as one batch at the close of voting. For purposes of the Standards, Public Network DRE Voting Systems are considered a form of DRE Voting System and are subject to the standards applicable to DRE Voting Systems. However, because transmitting vote data over public networks relies on equipment beyond the control of the election authority, the system is subject to additional threats to system integrity and availability. Therefore, additional requirements discussed in Section 5 and 6 apply.

The use of public networks for transmitting vote data must provide the same level of integrity as other forms of voting systems, and must be accomplished in a manner that precludes three risks to the election process: automated casting of fraudulent votes, automated manipulation of vote counts, and disruption of the voting process such that the system is unavailable to voters during the time period authorized for system use.

1.5.5 Precinct Count Voting System

A Precinct Count Voting System is a voting system that tabulates ballots at the polling place. These systems typically tabulate ballots as they are cast, and print the results after the close of polling. For DREs, and for some paper-based systems, these systems provide electronic storage of the vote count and may transmit results to a central location over public telecommunication networks.

1.5.6 Central Count Voting System

A Central Count Voting System is a voting system that tabulates ballots from multiple precincts at a central location. Voted ballots are typically placed into secure storage at the polling place. Stored ballots are transported or transmitted to a central counting place. The systems produce a printed report of the vote count, and may produce a report stored on electronic media.

1.6 APPLICATION OF THE SATNDARDS AND TEST SPECIFICATIONS

The Standards apply to all system hardware, software, telecommunications, and documentation intended for use to:

- Prepare the voting system for use in an election;
- Produce the appropriate ballot formats;
- Test that the voting system and ballot materials have been properly prepared and are ready for use;
- Record and count votes;
- Consolidate and report results;
- Display results on-site or remotely; and
- Maintain and produce all audit trail information.

In general, the Standards define functional requirements and performance characteristics that can be assessed by a series of defined tests . Standards are mandatory requirements and are designated by use of the term "shall." Some voting systems use one or more readily available commercial off-the-shelf (COTS) devices (such as card readers, printers, or personal computers) or software products (such as operating systems, programming language compilers, or database management systems). COTS devices and software are exempted from certain portions of the qualification testing process as defined herein, as long as such products are not modified for use in a voting system.

Generally, voting systems are subject to the following three testing phases prior to being purchased or leased:

- Qualification tests;
- State certification tests; and
- State and/or local acceptance tests.

1.6.1 Qualification Tests

Qualification tests validate that a voting system meets the requirements of the Standards and performs according to the vendor's specifications for the system. Such tests encompass the examination of software; the inspection and evaluation of system documentation; tests of hardware under conditions simulating the intended storage, operation, transportation, and maintenance environments; operational tests to validate system performance and function under normal and abnormal conditions; and examination of the vendor's system development, testing, quality assurance, and configuration management practices. Qualification tests address individual system components or elements, as well as the integrated system as a whole.

Since 1994, qualification tests for voting systems have been performed by Independent Test Authorities (ITAs) certified by the National Association of State Election Directors (NASED). NASED has certified an ITA for either the full scope of qualification testing or a distinct subset of the total scope of testing. To date, ITAs have been certified only for distinct subsets of testing. Upon the successful completion of testing by an ITA, the ITA issues a Qualification Test Report to the vendor and NASED. The qualification test report remains valid for as long as the voting system remains unchanged.

Upon receipt of test reports that address the full scope of testing, NASED issues a Qualification Number that indicates the system has been tested by certified ITAs for compliance with the Standards and qualifies for the certification process of states that have adopted the Standards. The Qualification Number applies to the system as a whole, and does not apply to individual system components or untested configurations.

After a system has completed qualification testing, further examination of a system is required if modifications are made to hardware, software, or telecommunications, including the installation of software on different hardware. Vendors request review of modifications by the appropriate ITA based on the nature and scope of changes made and the scope of the ITA's role in NASED qualification. The ITA will determine the extent to which the modified system should be resubmitted for qualification testing and the extent of testing to be conducted.

Generally, a voting system remains qualified under the standards against which it was tested, as long as no modifications not approved by an ITA are made to the system. However, if a new threat to a particular voting system is discovered, it is the prerogative of NASED to determine which qualified voting systems are vulnerable,

whether those systems need to be retested, and the specific tests to be conducted. In addition, when new standards supersede the standards under which the system was qualified, it is the prerogative of NASED to determine when systems that were qualified under the earlier standards will lose their qualification, unless they are tested to meet current standards.

Among other things, qualification testing complements and evaluates the vendor's developmental testing and beta testing. The ITA is expected to evaluate the completeness of the vendor's developmental test program, including the sufficiency of vendor tests conducted to demonstrate compliance with the Standards as well as the system's performance specifications. The ITA undertakes sample testing of the vendor's test modules and also designs independent system-level tests to supplement and check those designed by the vendor. Although some of the qualification tests are based on those prescribed in the Military Standards, in most cases the test conditions are less stringent, reflecting commercial, rather than military, practice.

1.6.2 Certification Tests

Certification tests are performed by individual states, with or without the assistance of outside consultants, to:

- Confirm that the voting system presented is the same as the one qualified through the Standards;
- Test for the proper implementation of state-specific requirements;
- Establish a baseline for future evaluations or tests of the system, such as acceptance testing or state review after modifications have been made; and
- Define acceptance tests.

Precise certification test scripts are not included in the Standards, as they must be defined by the state, with its laws, election practices, and needs in mind. However, it is recommended that they not duplicate qualification tests, but instead focus on functional tests and qualitative assessment to ensure that the system operates in a manner that is acceptable under state law. If a voting system is modified after state certification, it is recommended that States reevaluate the system to determine if further certification testing is warranted.

Certification tests performed by individual states typically rely on information contained in documentation provided by the vendor for system design, installation, operations, required facilities and supplies, personnel support and other aspects of the voting system. States and jurisdictions may define information and documentation requirements additional to those defined in the Standards. By design, the Standards,

and qualification testing of voting systems for compliance with the Standards, do not address these additional requirements. However, qualification testing addresses all capabilities of a voting system stated by the vendor in the system documentation submitted to an ITA, including additional capabilities that are not required by the Standards.

1.6.3 Acceptance Tests

Acceptance tests are performed at the state or local jurisdiction level upon system delivery by the vendor to:

- Confirm that the system delivered is the specific system qualified by NASED and, when applicable, certified by the state;
- Evaluate the degree to which delivered units conform to both the system characteristics specified in the procurement documentation, and those demonstrated in the qualification and certification tests; and
- Establish a baseline for any future required audits of the system.
- Define acceptance tests.

Some of the operational tests conducted during qualification may be repeated during acceptance testing.

1.7 OUTLINE OF CONTENTS

The organization of the Standards has been simplified to facilitate its use. Volume I, Voting System Performance Standards, is intended for use by the broadest audience, including voting system developers, equipment manufacturers and suppliers, independent test authorities, local agencies that purchase and deploy voting systems, state organizations that certify a system prior to procurement by a local jurisdiction, and public interest organizations that have an interest in voting systems and voting systems standards.

- Section 2 describes the functional capabilities required of voting systems.
- Sections 3 through 6 describe specific performance standards for election system hardware, software, telecommunications and security, respectively.
- Sections 7 and 8 describe practices for quality assurance and configuration management,

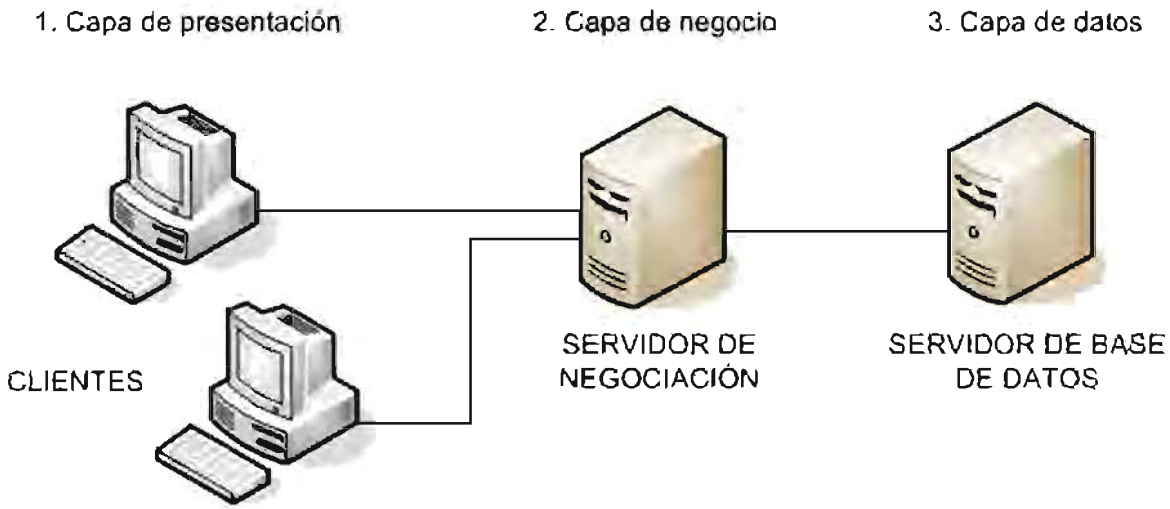
respectively, to be used by vendors, and required information about vendor practices that will be reviewed in concert with system qualification and certification test processes and system purchase decisions.

- Section 9 provides an overview of the test and measurement process used by test authorities for qualification and re-qualification of voting systems.
- Appendix A provides a glossary of important terms used in Volume I.
- Appendix B lists the publications that were used for guidance in the preparation of the Standards. These publications contain Information that is useful in interpreting and complying with the requirements of the Standards.
- Appendix C addresses issues of usability of voting systems, commonly referred to as "human factors." This appendix does not represent mandates that voting systems will be tested against, but rather contain recommendations and best practices on usability issues designed to provide vendors and election officials with guidance on designing and procuring systems that are easy and intuitive to use by voters.

Volume II, Voting System Qualification Testing Standards describes the standards for the technical Information submitted by the vendor to support testing; the development of test plans by the ITA for initial system testing and testing of system modifications; the conduct of system qualification tests by the ITA; and the test reports generated by the ITA. This volume complements the content of Volume I and it is intended primarily for use by ITAs, state organizations that certify a system, and vendors.

PROGRAMACIÓN POR CAPAS

La **programación por capas** es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.



La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería el modelo de interconexión de sistemas abiertos.

Además, permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas. En dichas arquitecturas a cada nivel se le

confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El diseño más utilizado actualmente es el diseño en tres niveles (o en tres capas).

Capas y niveles

1.- Capa de presentación: es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

2.- Capa de negocio: es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos para almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

3.- Capa de datos: es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador, si bien lo más usual es que haya una multitud de ordenadores en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor). Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las

necesidades lo aconseja se pueden separar en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en que resida la capa de negocio.

Si, por el contrario, fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o más ordenadores que realizarían solicitudes a una única base de datos. En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de negocio, y otra serie de ordenadores sobre los cuales corre la base de datos.

En una arquitectura de tres niveles, los términos "capas" y "niveles" no significan lo mismo ni son similares.

El término "capa" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico:

Presentación/ Lógica de Negocio/ Datos.

En cambio, el término "nivel" corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo:

- Una solución de tres capas (presentación, lógica del negocio, datos) que residen en un solo ordenador (Presentación+lógica+datos). Se dice que la arquitectura de la solución es de tres capas y *un nivel*.

- Una solución de tres capas (presentación, lógica del negocio, datos) que residen en dos ordenadores (presentación+lógica, lógica+datos). Se dice que la arquitectura de la solución es de tres capas y *dos niveles*.

- Una solución de tres capas (presentación, lógica del negocio, datos) que residen en tres ordenadores (presentación, lógica, datos). La arquitectura que la define es: solución de tres capas y *tres niveles*.