Modelo de Datos del Sistema de Voto Electrónico Presencial OTP-Vote

Silvia Bast[1], Pablo García[1], Germán Montejano[23]

[1] Departamento de Matemática - Facultad de Ciencias Exactas y Naturales - Universidad Nacional de La Pampa
Av. Uruguay 151- (6300) Santa Rosa - La Pampa - Argentina
{silviabast,pablogarcia}@exactas.unlpam.edu.ar
http:/www.exactas.unlpam.edu.ar
[2] Departamento de Informática - Universidad Nacional de San Luis
Ejército de los Andes 950 – (5700) San Luis – San Luis – Argentina
gmonte@unsl.edu.ar
http://www.unsl.edu.ar
[3] Departamento de Informtica – Facultad de Ingeniería - Universidad Nacional de La Pampa
Calle 9 esquina 10 – (6360) General Pico – La Pampa - Argentina
german.a.montejano@gmail.com
http://www.ing.unlpam.edu.ar

Resumen La incorporación del voto electrónico en el sistema electoral, se presenta actualmente como un tema con grandes controversias; se han producido profundos debates acerca de sus beneficios y desventajas.
Los sistemas de E-Voting deben presentar características tales como: anonimato, verificabilidad, elegibilidad, simplicidad, costo razonable, inviolabilidad, auditabilidad, no-coerción, robustez y escalabilidad. Al estudiar en detalle estos requerimientos se infiere que la seguridad de los datos que usan y producen tales sistemas representa un punto central para garantizar la confiabilidad de los mismos. La seguridad de los datos está representada por tres características: confidencialidad, integridad y disponibilidad.
En los sistemas objeto de estudio la característica de confidencialidad o anonimato del voto debe protegerse indefinidamente. No ocurre lo mismo con los datos de los votos, que sólo deben ser resguardados mientras dure el proceso eleccionario y luego de finalizado el mismo pasan a ser de conocimiento público.
El objetivo del presente trabajo es proponer un modelo que optimice los aspectos de confidencialidad e integridad de los datos, ofreciendo anonimato incondicional y seguridad computacional (que puede llevarse al nivel exigible) durante el proceso electoral.

Keywords: Sistemas de Voto Electrónico, Confidencialidad, Integridad, One Time Pad, Modelo de Almacenamiento Múltiples Canales Dato Único.

1. Introducción

1.1. Sistemas de Voto Electrónico

El voto electrónico puede definirse como un método en el cual los votos son emitidos y/o tabulados por medios electrónicos.
Según [1] "Un sistema de voto electrónico es un sistema cuyo principal elemento es un componente de software que mapea el procedimiento de voto electrónicamente". La definición del Observatorio del voto-E en América Latina afirma que "es una forma de votación basada en medios electrónicos que se diferencia del método tradicional por la utilización de tecnologías como hardware, software y procedimientos que permiten automatizar los procesos que comprenden unas elecciones". En [2] se los define como "los sistemas en que el registro, la emisión o el conteo de los votos en elecciones para cargos políticos y referendos involucra el uso de tecnologías de la información y las comunicaciones (TIC) que permiten automatizar los procesos que comprenden elecciones".
En [3], [4], [5] y [6] se proponen conjuntos de requerimientos específicos que estos sistemas deben presentar y que pueden resumirse en: secreto o anonimato, verificabilidad, elegibilidad (sólo pueden votar las personas que aparecen en el padrón), simplicidad, costo razonable, inviolabilidad, auditabilidad, no-coerción, robustez y escalabilidad.
McGaley y Gibson afirman en [7] que "Un sistema de votación es tan bueno como el público cree que es". Tal aseveración permite deducir que no sólo es fundamental construir sistemas sólidos, sino también demostrar tal solidez, por lo que los aportes que puedan realizarse en pos de definir parámetros, técnicas, modelos o estrategias que contribuyan a mejorar la seguridad de los datos de un proceso eleccionario revisten interés; este aspecto incluye los siguientes requerimientos: *Confidencialidad*, *Integridad* y *Disponibilidad*.

1.2. Motivación

El presente trabajo se basa en la siguiente premisa. En los sistemas de Voto electrónico es necesario proteger:

- Indefinidamente la privacidad del votante: aún después de finalizada la elección, dado que en caso de que algún intruso obtenga una copia digital de registros que permitan relacionar el votante con su voto contaría con todo el tiempo para intentar descifrarlo. Todas las personas desean mantener su privacidad asegurada indefinidamente y sería de suma gravedad que se conociera por quién votó un elector en particular. Por ejemplo, conocer la trayectoria como votante de un candidato actual podría influir en el electorado.
- La seguridad de los datos de los votos mientras dure el proceso electoral: la protección de la información de los sufragios emitidos sólo debe soportar el lapso de tiempo que corresponda al proceso de votación, dado que el modelo propuesto registra únicamente los datos de los sufragios y no de los electores y luego del tiempo previsto la información resultante del conteo, es decir los resultados de la elección, son públicamente conocidos.

Específicamente, el modelo propuesto ofrece anonimato de forma indefinida y seguridad computacional, que puede llevarse al nivel exigible, durante el proceso electoral.

2. Modelo Propuesto

Como se afirmó anteriormente la seguridad de los datos en cualquier sistema está dada por el cumplimiento de las características de confidencialidad, integridad y disponibilidad. Específicamente en el caso de los sistemas de voto electrónico, las mismas deben aplicarse a los datos que surjan o se modifiquen en los momentos de:

- Configuración de la elección.
- Desarrollo de la elección: esto incluye la autenticación de los electores y la emisión del voto.
- Cierre de la elección y recuento de votos.

2.1. Datos del Modelo Propuesto

Existirán en tales sistemas objetos que almacenen información de Padrón Electoral, los Cargos que están en juego en la elección, los Candidatos que se postulan para tales cargos y finalmente el Registro de los Votos emitidos por los electores. En el Cuadro 1 se muestran las características de seguridad que deben presentar cada uno de los datos mencionados.

	Confidencialidad	Integridad	Disponibilidad
Padrón Electoral	No	Si	Si
Votos	Si	Si	Si
Candidatos	No	Si	Si
Cargos	No	Si	Si

Cuadro 1: Datos de los Sistemas de E-Voting - Características de Seguridad

El proceso eleccionario consiste de tres grandes etapas:

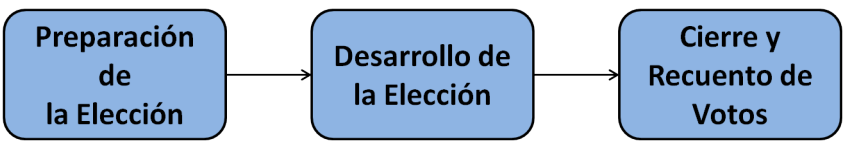

Figura 1: Etapas del Proceso Eleccionario

En la siguiente Figura pueden observarse las etapas con sus datos de entrada y salida.

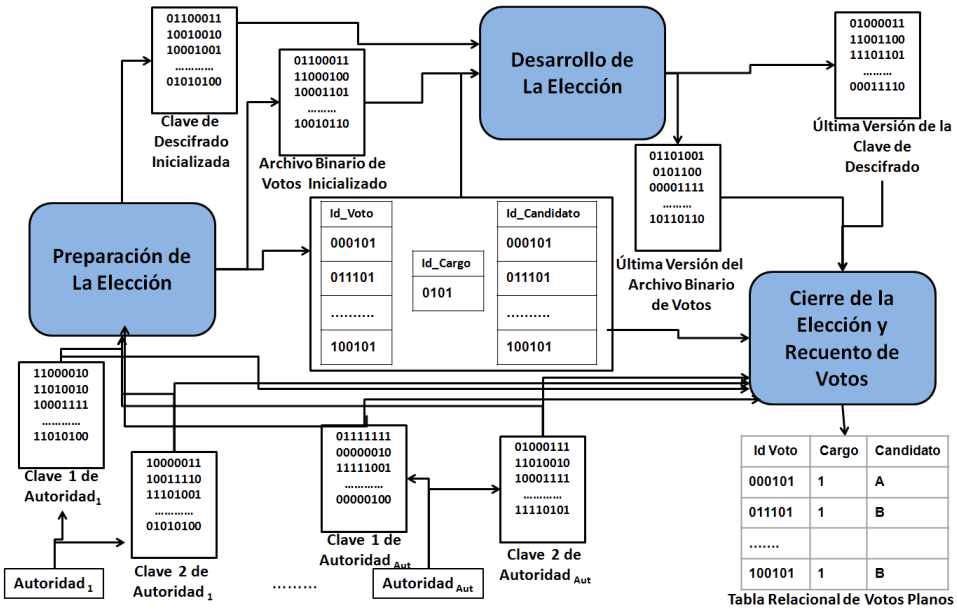

Figura 2: Etapas del Proceso Eleccionario con Datos de Entrada y Salida

Los elementos de datos que aparecen en el Modelo son:

- Claves.
- Archivos de datos que almacenan bits.
- Tablas relacionales.

Claves: el Modelo hace uso de **Claves One Time Pad (OTP)**.
OTP es un algoritmo criptográfico que puede crear un texto cifrado del que nadie puede obtener el texto plano y que no puede quebrarse aún con potencia de cálculo infinito e ilimitada cantidad de tiempo. Está basado en el Secreto perfecto de Shannon [8]. Las mencionadas claves presentan las siguientes características:

- Son aleatorias.
- Son tan largas como el mensaje mismo.
- A partir del mismo texto cifrado, aplicando una clave diferente se produce un texto plano distinto.

Archivos de Datos que Almacenan Bits: son elementos fundamentales en el Modelo propuesto y se van modificando durante el proceso eleccionario:

- Clave de Descifrado.
- Archivo Binario de Votos.

La Clave de Descifrado: se genera a partir de operaciones **XOR ($\oplus$)** de claves OTP. La operación XOR presenta las siguientes propiedades:

- Es conmutativa: Es decir que $A \oplus B = B \oplus A$
- Asociativa:$(A \oplus B) \oplus C = A \oplus (B \oplus C)$
- Autoinversa:$(A \oplus B) \oplus B = A$

El Archivo Binario de Votos: se genera en base al modelo de almacenamiento **Múltiples Canales Dato Único (MCDU)** propuesto en [9].
El esquema deriva del modelo de almacenamiento que propone Non Interactive Dining Cryptographers (NIDC) [10] que consiste en un único canal de slots en el que se almacenan los datos en posiciones aleatorias. Gráficamente se muestra en la Figura 3.

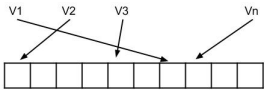

Figura 3: Propuesta de Almacenamiento de NIDC

Con este esquema es posible que dos o más datos elijan aleatoriamente un mismo slot para depositar una información. Esta coincidencia recibe el nombre de *colisión*. Una colisión implica la pérdida de todos los datos que se almacenan en el mismo slot. El objetivo es entonces, asegurar que la proporción de datos perdidos se mantenga por debajo de un valor deseado, con una probabilidad específica.
Para profundizar el concepto de probabilidad de colisiones, se tiene en cuenta la Paradoja del Cumpleaños (Birthday Paradox [11]), que afirma que en un grupo de 23 personas, la probabilidad de que haya al menos 2 que cumplan años el mismo día (de los 365 posibles) es muy cercana a $\frac{1}{2}$. Esta afirmación implicaría que el tamaño del vector donde se guardará la información deberá ser muy grande con respecto a la cantidad de datos que se almacenarán en el mismo y, a pesar de ello, la probabilidad de colisiones se mantendrá lejos de niveles que pudieran ser aceptables, si se piensa por ejemplo, en un sistema de E-Voting.
Basándose en que es posible separar un único arreglo de slots, en varios arreglos o canales, tales que la suma de los slots de cada uno de los canales sea igual a la cantidad de slots contenidos en el arreglo único, [9] propone varios modelos:

- **Único Canal - Dato Único (UCDU)**: almacenar el dato en una posición seleccionada aleatoriamente de un canal (esquema original). Vale para este caso lo expuesto sobre Birthday Paradox.

- **Único Canal - Dato Múltiple (UCDM)**: almacenar el dato en varias posiciones seleccionadas aleatoriamente en el mismo canal. Se demostró mediante simulaciones que los resultados no son los esperados, dado que aumentan significativamente las colisiones y con ellas, la pérdida de datos.
- **Múltiples Canales - Dato Único (MCDU)**: consiste en almacenar el dato en posiciones aleatorias (potencialmente distintas) de cada uno de los múltiples canales. El tamaño total T de slots del vector, se divide en Q canales que funcionan en forma paralela, el voto se registra una vez en cada canal en posiciones potencialmente diferentes, los resultados mejoran, dado que la probabilidad de que un voto se pierda simultáneamente en todos los canales, corresponde a la probabilidad de eventos independientes y por lo tanto es el producto de las probabilidades. Es aconsejable que la cantidad de slots de cada canal sea razonablemente mayor que N. La idea detrás de esta propuesta es lograr una mayor utilización de los slots vacíos que se presentan en el modelo original.
- **Múltiples Canales - Dato Múltiple (MCDM)**: almacenar el dato en varias posiciones aleatorias (potencialmente distintas) de cada uno de los múltiples canales. Se realizaron las correspondientes simulaciones y los resultados no fueron auspiciosos dado que aumentan significativamente las colisiones lo que provoca pérdidas de datos

Luego de este análisis se concluye que la utilización de canales paralelos de slots genera una utilización más eficiente del espacio asignado para el almacenamiento de los datos cuando se utiliza un modelo MCDU (Figura 4).

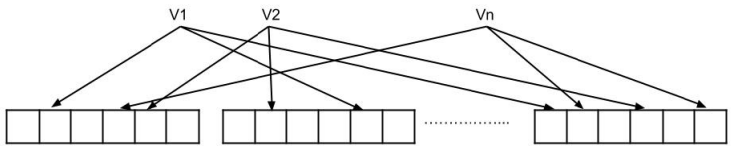

Figura 4: Propuesta de Almacenamiento de MCDU

Cada canal implementado agrega un factor al producto que determina la probabilidad de que el dato se pierda simultáneamente en todos los canales, único caso en el que el mismo será efectivamente perdido. Se desprenden del mencionado trabajo conclusiones tales como:

- Para un número fijo de votantes N, el número mínimo aconsejable de slots (S) para cada canal paralelo está dado por la fórmula:

$$S = \left| \frac{N}{ln2} \right| + 1 \tag{1}$$

- Para valores dados de T (total de slots a implementar) y N (número de votantes), existe un número óptimo de canales paralelos, expresado por la fórmula:

$$Q_{ot} = \ln 2 \frac{T}{N} \tag{2}$$

que debe llevarse al entero siguiente resultando entonces

$$Q_{ot} = |Q_{ot}| + 1 \tag{3}$$

- El valor esperado para la variable PVP (Probabilidad de Votos Perdidos) se obtiene mediante la aplicación de la ecuación:

$$|PVP| = \left(1 - e^{-\frac{N}{S}}\right)^{Q} \tag{4}$$

donde Q indica número de canales.

- Se obtiene una cota inferior apropiada para la probabilidad de que no se pierdan votos, que fue corregida en [8] resultando la ecuación:

$$[PR(X) > 1 - \left(\left(\frac{1}{S}(N-1)\right)^{Q}\right)^{N} \tag{5}$$

donde $X =$ *"No se pierde ningún voto"*

- Se enuncia una técnica concreta para definir valores para S y Q de forma que la probabilidad de que no se pierda ningún voto sea mayor que un valor dado.

Las fortalezas de este modelo de almacenamiento son:

- La aleatoriedad que propone para las posiciones donde se almacena la información, que es un aporte fundamental a la característica de anonimato.
- El uso eficiente del almacenamiento.
- La disminución de de la probabilidad de colisiones, que puede llevarse a cualquier nivel exigible a través del uso de las fórmulas para la configuración de parámetros involucrados.

Tablas Relacionales: se usan para almacenar los datos de los Cargos, los Candidatos, los Identificadores de Votos y también de los Votos Emitidos una vez finalizado el proceso eleccionario.

3. Etapas Modelo de Datos Propuesto

A continuación se detallan las actividades que se llevan a cabo en cada una de las etapas del modelo.

3.1. Etapa de Preparación de la Elección

Las actividades que deben llevarse a cabo en esta etapa son:

- Especificar la semántica de la tupla en la que se almacenarán los votos.
- Definir las dimensiones del Archivo Binario de Votos y Clave de Descifrado.
- Generar las tablas: Cargos, Candidatos e Identificadores de Votos.
- En el momento previo al comienzo del acto eleccionario se inicializan el Archivo Binario de Votos y la Clave de Descifrado.

Definición de Semántica de la Tupla: debe especificarse:

- Cantidad total de bits de cada slot (*Tbslot*).
- Cantidad de bits del identificador del voto (*Tbid*) y sus ubicaciones.
- Cantidad de bits de cada cargo que se vota (*Tbcargo*) y sus ubicaciones.
- Cantidad de bits del código del candidato votado (*TbCandidato*) para cada cargo de la elección y sus ubicaciones.
- Cantidad de bits de control adicionales y sus ubicaciones.

Para definir las dimensiones de los atributos, debe evaluarse la probabilidad de que algún intruso pueda obtener un dato válido de entre todos los posibles. A través del aumento de la redundancia en la cantidad de bits usados para el almacenamiento de cada uno de los atributos, la probabilidad de obtener una tupla válida de entre todas las combinaciones de valores posibles puede llevarse a cualquier valor deseado.
Sea *IdV = "Se obtiene un valor válido de Identificador de Voto"*
La probabilidad de que este hecho ocurra es:

$$P(IdV) = N/2^{TbId}$$

Donde N es la cantidad de electores y $TbId$ es la cantidad de bits del identificador del voto.
Sea *CargoV = "Se obtiene un valor válido de Cargo"*
Para cada cargo la probabilidad de que este hecho ocurra es:

$$P(CargoV) = 1/2^{TbCargo}$$

Donde $TbCargo$ es la cantidad de bits de cada cargo que se vota.
Sea *CandidatoV = "Se obtiene un valor válido de Candidato"*
La probabilidad de que este hecho ocurra es:

$$P(CandidatoV) = CantidaddeCandidatosparaelcargo/2^{TbCandidato}$$

Donde $TbCandidato$ es la cantidad de bits asignada para almacenar el candidato.
Para que sea posible obtener todos los valores de una tupla, se deben generar valores válidos para todos los atributos.
Se define el evento:
TodosVálidos = "Todos los atributos de la tupla toman valores válidos"
La probabilidad de que este hecho ocurra es:

$$P(TodosValidos) = P(IdV)P(CargoV)P(CandidatoV)$$

Por ejemplo para 240 electores, 10 candidatos y un cargo con TBId=256, TBCargo=64 y TBCandidato=256, se tiene:

$$P(TodosValidos) = (240/2^{256})(1/2^{64})(10/2^{256}) = 9,7036E-171$$

El valor obtenido para $P(TodosValidos)$ es extremadamente bajo para los parámetros seleccionados. Además a través del aumento de la redundancia en el almacenamiento de los atributos puede llevarse a cualquier valor deseado.

Dimensiones del Archivo Binario de Votos y Clave de Descifrado: se definen las dimensiones del Archivo Binario de Votos y Clave de Descifrado. Para N electores:

- Cantidad total de tuplas o filas del Archivo Binario de Votos T.
- Cantidad de slots o cantidad de tuplas por canal S.
- Cantidad de canales paralelos Q.

En esta etapa se hace uso de las fórmulas propuestas 1, 2, 3, 4 y 5.

Generación de Tablas Relacionales: se generan las tablas: Cargos, Candidatos e Identificadores de Votos.
La tabla de Cargos, estará configurada de la siguiente forma:

Id Voto	Descripción
Código Binario	Descripción textual

La tabla de Candidatos, se define como sigue:

Id Candidato	Descripción
Código Binario	Descripción textual

La tabla de Identificadores de Votos queda definida:

Id Voto	Usado
Código Binario	Si/No (Indica si el identificador ya fue asignado a un voto)

Debe tenerse en cuenta que los datos se almacenan en el Archivo Binario de Votos a través de una operación XOR.
Los Id de voto deben presentar las siguientes características:

- Deben ser distintos a los que ya están almacenados en la tabla.

$$Id_i \neq Id_j \, \forall j, 1 \leq j \leq i-1$$

- El XOR del Id_i con cada uno de los Identificadores que ya residen en la tabla, no debe dar como resultado alguno de los Id existentes. Este requerimiento impide que se presenten casos como el que se muestra en la Figura 5.

$$Id_i \oplus Id_j \neq Id_k \, \forall j \neq i \, 1 \leq j \leq i-1 \, \forall k \neq i \, 1 \leq k \leq i-1$$

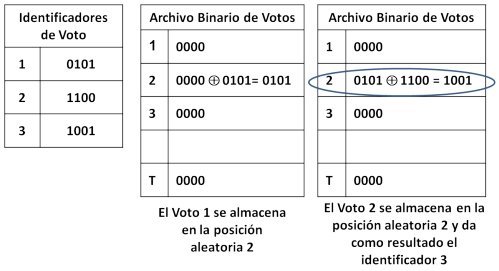

Figura 5: Ejemplo de Colisión que produce un Id Válido

- El XOR Id_i con cada una de las combinaciones de grupos de los identificadores que ya residen en la tabla, no debe dar como resultado alguno de los Id existentes.

$$Id_i \oplus C\oplus_N^j : 1 \leq j \leq i-1$$

  Donde $C\oplus_N^j$ indica el $\oplus$ de los identificadores resultantes de la combinatoria de N elementos tomados de a j.

Esta última característica apunta a prevenir que una o varias colisiones en un slot, den como resultado identificadores válidos que podrían corresponder a valores válidos de otro voto, ya que en MCDU, el almacenamiento de los votos se lleva a cabo mediante una operación XOR. La verificación de esta condición es costosa. Para subsanar este problema, se recurre a las heurísticas, realizando sucesivas pruebas con un simulador de colisiones propuesto en [12], que fue modificado para el presente trabajo con el fin de obtener el grado de multiplicidad de las colisiones. Se realizaron simulaciones con parámetros para 120, 240 y 480 votantes. Para valores de Q: 1, 5, 10 y el Q_{op}. A medida que aumentan los canales también aumentan las colisiones, pero al dividir la cantidad de slots por el Q_{op}, la probabilidad de pérdidas de votos disminuye. Para una elección de 240 votantes, lo que representaría una mesa electoral nacional, las colisiones de 7 votos en todos los casos permanecen en cero, entonces se reduciría severamente la probabilidad de pérdidas por colisiones séptuples. Se trabaja actualmente en la formalización matemática de las probabilidades de colisiones múltiples.

$$Id_i \oplus C\oplus_N^j : 1 \leq j \leq 6$$

Por lo que el tiempo y complejidad del algoritmo se reducen notablemente.Se trabaja actualmente en la formalización matemática de las probabilidades de colisiones múltiples.
Deben definirse también con similares características en sus claves, las tablas de Cargos y Candidatos.

Inicialización de Archivo Binario de Votos y Clave de Descifrado: se inicializan el Archivo Binario de Votos y la Clave de Descifrado, mediante el aporte de Claves OTP de las Autoridades Electorales. Las mismas se usarán en:

- Preparación de la Elección.
- Cierre de la Elección.

En el transcurso del proceso debe garantizarse que las mismas se encuentren seguras y aisladas. Antes del comienzo de la elección, cada una de las autoridades que conforman la Junta Electoral aportará dos claves One Time Pad (OTP).

- Las primeras claves aportadas por cada una de las autoridades servirán para dar valores al Archivo Binario donde residirán los votos. Inicialmente el mencionado archivo tendrá asignado cero en cada una de sus posiciones. Luego se aplicará el XOR de cada una de las claves 1 aportadas por las autoridades de la Junta electoral.

$$ArchivoBinario = ArchivoBinarioInicial \oplus Clave1_1$$
$$ArchivoBinario = ArchivoBinario \oplus Clave1_{Aut}$$
$$\forall\, Aut\, 2 \leqslant Aut \leqslant CantidaddeAutoridades$$

- Las segundas claves de las autoridades inicializarán la Clave de Descifrado, que en un principio se encuentra con todos sus elementos en cero.

$$ClaveDescifrado = ClaveDescifradoInicial \oplus Clave2_1$$
$$ClaveDescifrado = ClaveDescifrado \oplus Clave2_{aut}$$
$$\forall\, Aut\, 2 \leqslant Aut \leqslant CantidaddeAutoridades$$

3.2. Etapa de Emisión de Votos

Durante el proceso de Desarrollo de la Elección, tienen lugar dos actividades fundamentales:

- Autenticación del elector.
- Emisión del voto.

Autenticación del Elector: la autenticación consiste en verificar que el elector figure dentro del padrón electoral, es decir que sea un votante válido. En el presente trabajo se propone proceder de la forma habitual, registrándose el usuario en el lugar de la elección, con la presencia de una autoridad de la Junta Electoral, mediante la presentación del documento de identidad del elector, o si se cuenta con recursos suficientes, podría emplearse algún método de autenticación a través de datos biométricos, tales como (huellas dactilares, voz, retina).
Si bien el modelo propuesto es teórico, se sugiere que luego de la autenticación, y aplicando una analogía con el proceso que se sigue en el sistema de voto tradicional cuando el presidente de mesa entrega al votante el sobre habilitándolo a emitir el sufragio, sea la autoridad de mesa, quien habilite mediante un mecanismo, la emisión de un único voto en la mesa correspondiente. El uso de mecanismos de autenticación y votación totalmente separados es un aporte fundamental para no permitir el registro de la relación entre el voto y el votante que lo emite, lo que se agrega valor a la característica de anonimato del voto.
Emisión del Voto: para el almacenamiento de los Votos se sigue el esquema MCDU. Para cada voto:

- El sistema genera una Clave OTP $ClaveVoto_v$ de dimensión $TBslot$ bits que:
  - Se almacenará por medio de operaciones XOR aportando a la Clave de Descifrado Final de los votos.
  - Cifrar la información del voto.
- El elector genera su voto que combinado con la clave produce la Contribución Final de Voto. El aporte de la clave de voto a la clave de descifrado se lleva a cabo a través de la siguiente operación:

$$ClaveDescifrado = ClaveDescifrado \oplus ClaveVoto_v \forall\, v\, 1 \leqslant v \leqslant N$$

- Para cada Voto se genera $CadenadeVoto_v$ formada por: Id de Voto (asignado aleatoriamente), Id de Cargo y además el Id del Candidato seleccionado.
- El sistema genera una cadena de TBslot bits $ContribucionInicial_v$, con todos sus elementos en cero.
- El sistema produce un conjunto de números aleatorios $CjtoQ = \{q_i\}$ para cada uno de los Q canales, donde q_i representa el lugar donde se almacenará el voto en el canal i-ésimo.
- Se realiza el XOR de la $CadenaDeVoto_v$ con los slots que corresponden a los q_i de la $ContribucionInicial_v$. Esto es:

$$Contribucion_{vi} = CadenaDeVoto_v \oplus ContribucionInicial_{vi}\ \forall i\, \epsilon CjtoQ$$

En la Figura 6 se muestra un ejemplo de la aplicación de esta fórmula.

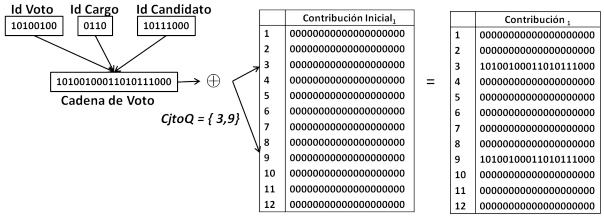

Figura 6: Ejemplo de Generación de Contribución Inicial

Finalmente se aplica:

$$ContribucionFinal_v = Contribucion_v \oplus ClaveVoto_v$$
$$ArchivoBinariodeVotos = ArchivoBinariodeVotos \oplus ContribucionFinal_v$$

3.3. Etapa de Cierre de la Elección y Recuento de Votos

Al momento de cierre de la elección se requiere la intervención de las Autoridades de la Junta Electoral. El proceso de descifrado de los votos consta de tres sub-procesos:

- Aplicación de las Claves 1 de las Autoridades a la última versión del Archivo Binario de Votos.
- Aplicación de las Claves 2 de las Autoridades a la última versión de la Clave de Descifrado.
- XOR entre el Archivo Binario de Votos y la Clave de Descifrado resultantes de los pasos anteriores.

Debe aclararse que las claves de las autoridades son las mismas que se usaron en la etapa inicial, y deben ser protegidas mientras dure el proceso de emisión de votos. Cada una de las autoridades aporta nuevamente las claves iniciales:

$$ArchivoBinariodeVotos = ArchivoBinariodeVotos \oplus clave1_{aut}$$
$$\forall\, Aut\ 1 \leqslant Aut \leqslant CantidaddeAutoridades$$

$$ClaveDescifrado = ClaveDescifrado \oplus clave2_{aut}$$
$$\forall\, Aut\ 1 \leqslant Aut \leqslant CantidaddeAutoridades$$

Para obtener el Archivo Binario de Votos Descifrado se aplica

$$ArchivoBinariodeVotosDescifrado = ArchivoBinariodeVotos \oplus ClaveDescifrado$$

Luego se procede de la siguiente manera:

- Se eliminan las tuplas que corresponde a votos vacíos (secuencias de 0).
- Se eliminan las tuplas con votos que no corresponden con la información de ninguno de los Id de voto, dado que estos votos se fueron generando por colisiones.
- Se recorre el Archivo Binario de Votos Descifrado, obteniendo cada tupla y se inserta cada registro en la TabladeVotos.
- Luego para cada Id de Voto de la tabla de IDs se encuentran todas las ocurrencias (a lo sumo Q, una por cada canal).
- Posteriormente se agrupa cada Identificador de voto con su candidato seleccionado y se procede a insertar estos datos en la tabla relacional resultante denominada VotosPlanos sobre la que se llevará a cabo el conteo general. Para cada Id se almacena un único voto.
- Luego se produce el conteo de los votos mediante una consulta SQL.

4. Conclusiones y Problemas Abiertos

Se arriba a las siguientes conclusiones:

- El Modelo mantiene el anonimato indefinidamente, esta característica es aportada por:
  - La aleatoriedad en el almacenamiento de datos del modelo subyacente, esto es, Modelo Múltiples Canales Dato Único.
  - Las claves One Time Pad que se usan para el cifrado.
  - La separación total de los procesos de acreditación y emisión de voto.

- En cuanto a la seguridad durante el proceso electoral es aportada por:
  - La aleatoriedad en el almacenamiento de datos del Modelo Múltiples Canales Dato Único.
  - Las claves One Time Pad que se usan para el cifrado.
  - El uso de la redundancia suficiente en los atributos de las tuplas y en la configuración de los parámetros de la elección.

En relación a las futuras extensiones del trabajo pueden mencionarse:

- Establecer un protocolo que garantice que la Clave del Voto se mantiene inalterable:
  - Para su aplicación a la Clave de Descifrado.
  - Como aporte a la Contribución Final del Voto que modificará finalmente el Archivo Binario de Votos.
- Profundizar el análisis sobre la información intermedia que habría que disponer para dar transparencia al proceso a la vista de terceros o auditores.
- Refinar la semántica de las tuplas.
- Analizar algoritmos alternativos para optimizar el recuento.
- Desarrollar técnicas de recuperación de colisiones.
- Desarrollar modificaciones al modelo para que soporte Verificación End to End.

Referencias

1. **Ondrisek, B.** "E-Voting System Security Optimization System Sciences" - HICSS09. 42 Hawaii International Conference on, 2009.
2. **Idea Internacional** "Consideraciones Esenciales para la Democracia y la Asistencia Electoral" - I. I. Una introducción al Voto Electrónico - 2011.
3. **Epstein, J.** "Electronic Voting Computer" - 2007, 40, 92-95.
4. **Kazi, M.; Rokibul, A. y Tamura, S.** "Electronic Voting - Scopes and Limitations" - International Conference on Informatics, Electronics & Vision (ICIEV) - May 2012.
5. **Prince, A.** "Consideraciones, Aportes y Experiencias para el Voto Electrónico en Argentina"- 2005.
6. **van de Graaf, J. and Henrich, C. and Muller-Quade, J.** "Requirements for Secure Voting" - 2011.
7. **McGaley, M. y Gibson, J. P.** "A Critical Analysis of the Council of Europe Recommendations on E-Voting" - Proceedings of the USENIX/Accurate Electronic Voting Technology Workshop on Electronic Voting Technology Workshop - USENIX Association - 2006.
8. **Shannon, C. E.** "Communication Theory of Secrecy Systems"- Bell System Tecnical Journal - Journal 28 (1949) 656–715.
9. **García, P.** "Una Optimización para el Protocolo Non Interactive Dining Cryptographers" - ISBN-13: 978-3-639-85270-7. ISBN-10: 3639852702. EAN: 9783639852707. Editorial Académica Española (https://www.eae-publishing.com/) - 2017.

10. **van de Graaf, J.** "Anonymous One Time Broadcast Using Non Interactive Dining Cryptographer Nets with Applications to Voting Towards Trustworthy Elections" - Springer - Verlag Berlin - Heidelberg -2010 - 231-241
11. **Flajolet, P.; Thimonier, L. y Gardy, D.** "Birthday paradox, Coupon Collectors, Caching Algorithms and Self-organizing Search Discrete Applied Mathematics" 1992, 39, 207-223.
12. **García, P.** "Optimización de un Protocolo Dining Cryptographers Asíncrono" - Msc Thesis - Universidad Nacional de San Luis (Argentina) - 2013.