Ataque de Revelación de Identidades en un Sistema Anónimo de Correo Electrónico

Javier Portela García-Miguel[1], Delfín Rupérez Cañas[2], Ana Lucila Sandoval Orozco[2], Alejandra Guadalupe Silva Trujillo[2], Luis Javier García Villalba[2]

[1] Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Estadística e Investigación Operativa III
Escuela Universitaria de Estadística, Despacho 721, Universidad Complutense de Madrid (UCM)
Avenida Puerta de Hierro s/n, 28040 Madrid
E-mail: jportela@estad.ucm.es

[2] Grupo de Análisis, Seguridad y Sistemas (GASS), Departamento de Ingeniería del Software e Inteligencia Artificial
Facultad de Informática, Despacho 431, Universidad Complutense de Madrid (UCM)
Calle Profesor José García Santesmases s/n, Ciudad Universitaria, 28040 Madrid
Email: {delfinrc, asandoval, asilva, javiergv}@fdi.ucm.es

Resumen*—El objetivo de nuestro trabajo es desarrollar un ataque global de tipo SDA** ***(Statistical Disclosure Attack) **que permita identificar las relaciones entre usuarios de una red basada en un sistema anónimo de** ***mixes*****. Nuestros escenarios son más generales en relación a otros ataques SDA. Asimismo presentamos un nuevo esquema teórico de modelado basado en tablas de contingencia. Proporcionamos soluciones para todos los usuarios simultáneamente, debido a que la dependencia de los datos no posibilita centrarse en usuarios específicos sin tener en cuenta las posibilidades combinatorias. A diferencia de simulaciones desarrolladas sobre este mismo tema, este trabajo ha sido desarrollado con datos reales de una aplicación de correos electrónicos, tomando en consideración las propiedades especiales de las redes de comunicación establecidas entre usuarios reales.**

I. Introducción

En las redes de comunicación los *mixes* proporcionan protección contra potenciales observadores al ocultar la apariencia de mensajes, patrones, longitud y relación entre emisores y receptores. Chaum [1] propuso ocultar la correspondencia entre emisores y receptores cifrando mensajes y reordenándolos a través de un camino de *mixes* antes de enviarlos a su destino. Se han propuesto muchos otros diseños, incluidos Babel [2], Mixmaster [3] o Mixminion [4]. Las diferencias entre estos sistemas no serán abordadas en nuestro trabajo: la información que usamos sólo se relaciona a emisores y receptores que están activos en un período de tiempo y la manera con la cual se reordenan los mensajes no afecta al ataque. Otra clase de diseños de anonimato, como *Onion routing* [5] son de baja latencia y están orientados a *Web browsing* y otros servicios interactivos. Nuestro método no se enfoca en estos diseños, los cuales pueden ser tratados efectivamente con ataques con períodos cortos de tiempo o de conteo de paquetes [6]. Los ataques contra las redes de *mixes* pretenden reducir el anonimato al relacionar cada emisor y receptor con sus correspondientes mensajes enviados o recibidos, o bien relacionar emisores con receptores. Al observar la red los atacantes pueden deducir la frecuencia de las relaciones, comprometiendo los *mixes* o llaves, alterando o retrasando los mensajes. Pueden ser capaces de deducir el destino más probable de los mensajes a través de falsos mensajes enviados a la red, y utilizar esta técnica para aislar y conocer las propiedades de ciertos mensajes previamente definidos. En [7] se muestra un resumen de ataques basados en análisis de tráfico. En [8], [9], [10], [11] se trata el tema del k anonimato, situado en el contexto multidimensional. Agrawal y Kesdogan [12] presentaron el *disclosure attack*, un ataque centrado en un mix de lotes simple, cuyo objetivo es obtener información de un emisor particular Alicia. El ataque es global, en el sentido de que recaba información sobre el número de mensajes enviados por Alicia y recibidos por otros usuarios; y pasivo, ya que el atacante no puede alterar la red, por ejemplo, enviando falsos mensajes o retrasándolos. Se asume que Alicia tiene exactamente m receptores, que envía mensajes con la misma probabilidad a cada uno de sus receptores, y además, que envía un mensaje en cada lote de *b* mensajes. Se podrían identificar a los receptores de Alicia clasificados en conjuntos disjuntos a través de algoritmos numéricos. Danezis [13] presenta el *Statistical Disclosure Attack* (SDA), considerando las hipótesis de [12]. En el SDA los receptores se ordenan en términos de probabilidad. Alicia debe demostrar patrones de envío consistentes a largo plazo para obtener buenos resultados. En [14] se describe el SDA cuando se usa *threshold mix* o *pool mix*, considerando las hipótesis de artículos previos donde se conoce el número de receptores de Alicia, o se enfoca en un solo usuario de Alicia. El SDA de doble orientación [15] usa las posibilidades de réplicas entre usuarios. El *Perfect Matching Disclosure Attack* [16] pretende utilizar información simultánea de todos los usuarios para obtener mejores resultados en la revelación de los receptores de Alicia. Este trabajo se enfoca en el problema de obtener información de las relaciones o la comunicación entre usuarios de una red, donde se obtiene información parcial. El enfoque de modelado del algoritmo y esquema de solución

Tabla I
EJEMPLO DE TABLA DE CONTINGENCIA

Receptores	Emisores			
	U3	U4	U5	Total enviados
U1	4	0	0	4
U2	0	1	0	1
U3	0	0	2	2
Total recibidos	4	1	2	7

Tabla II
EJEMPLO DE TABLA DE CONTINGENCIA CON INFORMACIÓN DE MARGINALES

Receptores	Emisores			
	U3	U4	U5	Total enviados
U1				4
U2				1
U3				2
Total recibidos	4	1	2	7

Tabla III
EJEMPLO DE TABLA CON COTAS OBTENIDAS

Receptores	Emisores			
	U3	U4	U5	Total enviados
U1	(1,4)	(0,1)	(0,2)	4
U2	(0,1)	(0,1)	(0,1)	1
U3	(0,2)	(0,1)	(0,2)	2
Total recibidos	4	1	2	7

son aplicados en datos de correos electrónicos. Como las soluciones individuales son interdependientes, nuestro ataque no se centra en un usuario en concreto, sino que pretende obtener la máxima información de todos los usuarios. La información utilizada es el número de mensajes enviado y recibido por cada usuario. Esta información es obtenida en rondas que pueden ser determinadas por intervalos de tiempo de una longitud determinada, o alternativamente en lotes de mensajes de igual tamaño. El marco base y supuestos necesarios para desarrollar nuestro algoritmo son los siguientes:

- El atacante conoce el número de mensajes enviados y recibidos por cada usuario en cada ronda.
- La ronda puede ser determinada por el sistema (lotes) o puede basarse en intervalos regulares de tiempo donde el atacante obtiene la información adicional de los mensajes enviados y recibidos. Hemos utilizado ambos métodos en nuestras aplicaciones obteniendo ligeramente mejores resultados al utilizar lotes (batches).
- El método está restringido, por el momento, a un sistema mix simple (sin considerar los *threshold mix* o *pool mix*).
- No se plantean restricciones sobre el número de amigos de cada usuario, ni sobre el número de mensajes a enviar. Ambos se consideran desconocidos de antemano.
- El atacante controla todos los usuarios del sistema. En nuestra aplicación nos centramos en los correos electrónicos que los usuarios de un dominio envían y reciben en este dominio.

Este artículo se compone de 5 secciones, siendo la primera la presente introducción. En la sección II plantea el problema, formulándolo con un nuevo enfoque a través de tablas de contingencia. Se presentan cotas y otras técnicas básicas de obtener información sobre el número de mensajes que envía cada usuario. La sección III explica el algoritmo propuesto, y detalla cómo puede obtenerse información relevante de las relaciones existentes (o no existentes) entre usuarios. La sección IV presenta la aplicación del algoritmo a datos reales. Finalmente, en la sección V se presentan las conclusiones sobre los resultados, y se plantean limitaciones y trabajos futuros a desarrollar sobre este ataque.

II. EL PROBLEMA

El atacante obtiene información de cuántos mensajes envía y recibe cada usuario en cada ronda. Normalmente el conjunto de emisores y receptores no es el mismo, aún cuando algunos usuarios puedan ser emisores y receptores en alguna ronda en particular. Además, el número total de usuarios en el sistema N no está presente en cada ronda, pues solo una fracción de ellos está recibiendo o enviando mensajes. En la Figura 1 se muestra una posible ronda, que por razones pedagógicas se compone de un mínimo de usuarios.

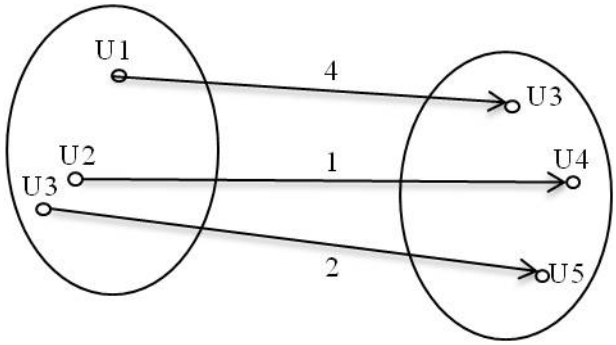

Figura 1. Relación entre emisores y receptores.

La información de esta ronda se puede representar en una tabla de contingencia (vea la Tabla I), donde el elemento (i, j) representa el número de mensajes enviados del usuario i al usuario j. El atacante solamente ve la información presente en las marginales agregadas donde, las filas representan el número de mensajes enviados por cada usuario, y las columnas, el número de mensajes recibido por cada usuario, según aparece en la Tabla II. Por medio de los valores marginales es posible obtener información importante. Las cotas de los elementos pueden ser útiles, ya que nos pueden proporcionar relaciones directas entre usuarios. Las cotas de Fréchet sobre tablas de contingencia son muy conocidas en estudios de revelación [19]. Denotando con n_{ij} el contenido del elemento (*i*,*j*), n_{i+} el valor marginal de la fila *i*, n_{+j} el valor marginal de la columna *j* y n el total. Las cotas de Fréchet se establecen como se muestra en la ecuación 1. Por ejemplo, partiendo de la Tabla II, se obtienen las cotas presentadas en la Tabla III.

$$max(n_{i+} + n_{+j} - n, 0) \leq n_{ij} \leq \text{mín}(n_{i+}, n_{+j}) \quad (1)$$

III. EL ALGORITMO

El objetivo principal del algoritmo que proponemos es extraer información relevante sobre las relaciones (o no relaciones) entre cada par de usuarios. Esta información puede ser obtenida en forma de reglas (0=relación, 1=no relación) o como probabilidades estimadas de relación, Otras fuentes de información obtenidas que pueden ser utilizadas

son la distribución estimada de mensajes del usuario *i* al *j* por unidad de tiempo, y la media estimada de mensajes de *i* a *j* por unidad de tiempo. La información obtenida por el atacante son las sumas marginales, por fila y columna, de cada una de las rondas $1, ..., T$ donde T es el número total de rondas. Hay que observar que en cada ronda la dimensión de la tabla es diferente, pues no tomamos en cuenta a usuarios que no envían (marginal de la fila=0) ni reciben (marginal de la columna=0) mensajes. Decimos que un elemento (*i*, *j*) está "presente" en una ronda si las marginales correspondientes no son cero. Esto significa que el usuario *i* está presente como emisor y el *j* como receptor. Se puede construir una tabla final A resumiendo todas las rondas y obteniendo una tabla con todos los mensajes enviados y recibidos por cada usuario en el intervalo de tiempo total considerado para el ataque. Cada elemento (*i*, *j*) de esta tabla final representaría el número total de mensajes enviados de *i* a *j*. Aunque la información obtenida en cada ronda es más precisa y relevante, un estimado exacto de la tabla A sería el principal objetivo ya que por ejemplo, un cero en la celda (*i*, *j*) y en la celda (*j*, *i*) significaría no relación entre los usuarios *i* y *j*. Mientras que un valor positivo indicaría que algún mensaje ha sido enviado en alguna ronda. Se presenta un algoritmo para generar tablas factibles (tablas cuyas sumas marginales en cada fila y columna coinciden con los valores marginales conocidos por el atacante).

Algoritmo 1

1. Comenzar con la columna 1, fila 1: generar n_{11} de una distribución uniforme entera en las cotas de la ecuación 1 donde $i = 1, j = 1$.
2. Para cada elemento n_{k1} en esta columna, si los elementos del renglón hasta $k-1$ se han obtenido, se calculan nuevas cotas para n_{k1} a partir de la ecuación 2.

$$\max\big((0, (n_{+1} - \sum_{i=1}^{k-1} n_{i1}) - \sum_{f=k+1}^{r} n_{i+}\big) \leq$$
$$n_{ij} \leq \min(n_{k+}, n_{+j} - \sum_{f=1}^{k-1} n_{i1}) \qquad (2)$$

   El elemento n_{k1} se genera entonces según un entero uniforme.
3. EL último elemento de la fila se rellena automáticamente dado que las cotas superior e inferior coinciden, haciendo $n_{(k+1)+} = 0$ por conveniencia.
4. Una vez que una columna está rellena, las marginales por fila n_{i+} y el valor N se actualizan por substracción de los elementos ya calculados, y el resto de la tabla se trata como una tabla nueva con una columna menos.
   El algoritmo calcula columna a columna hasta tener toda la tabla llena.

El tiempo empleado depende de la complejidad del problema (número de elementos, número promedio de mensajes). Para tablas grandes, toma menos de 3 minutos obtener un millón de tablas factibles, es decir aún cuando el numero de datos de correos electrónicos sea alto, no representa problema alguno. Repetir el algoritmo como está escrito para cada tabla no proporciona soluciones uniformes, porque algunas tablas son más probables que otras debido al orden utilizado al rellenar filas y columnas. Como debemos considerar a priori todas las soluciones igualmente posibles para una ronda determinada, se realizan dos modificaciones adicionales: i) Antes de generar soluciones se reordenan aleatoriamente las filas y columnas de la tabla; ii) Una vez que se generan todas las tablas deseadas, solo se conservan aquellas que son diferentes entre sí. Estas dos modificaciones han significado una mejora muy importante en los resultados de nuestro ataque. Decidir el número de tablas a generar plantea un problema interesante. Calcular el número de tablas factibles distintas en una tabla de contingencia con marginales fijos es todavía un problema abierto, que ha sido abordado a través de: métodos algebraicos, que son poco prácticos incluso para dimensiones moderadas, y por aproximaciones normales, que dan malos resultados con matrices dispersas, con muchos ceros y valores bajos, que es justo el tipo de matriz en nuestras aplicaciones. Hasta ahora la mejor aproximación para estimar el número de tablas factibles es utilizar las tablas generadas. Un estimado del número de tablas puede ser obtenido al promediar sobre las tablas generadas el valor $\frac{1}{q(T)}$ [20]. El número de tablas factibles va desde valores moderados que son fácilmente abordados como 100,000 obteniendo todas las tablas por simulación, hasta números tan altos como 10^{13}. Generar todas las tablas posibles para este último caso llevaría, con el ordenador que hemos usado, al menos 51 días. La razón principal por la que se complica llevar a cabo un ataque determinístico de intersección es la cantidad de tablas factibles, aún cuando las dimensiones de usuarios sean bajas o moderadas. Además, los ataques estadísticos centrados en un único usuario sin tener en cuenta las relaciones entre todos los usuarios son muy optimistas, pues la dimensión de las posibilidades es tan grande que llevaría años de comportamiento consistente de un usuario en particular para alcanzar convergencia débil. La información obtenida finalmente consiste en un número fijo de tablas factibles generadas para cada ronda. Considerando la información obtenida sobre todas las rondas, la media de cada elemento sobre todas las tablas para todas las rondas es un estimado del valor real de este elemento. La media obtenida en cada elemento y ronda se agrega sobre todas las rondas para obtener un estimado de la tabla agregada, $\widehat{A}$. Regularmente, los elementos en $\widehat{A}$ son estrictamente positivos excepto para casos triviales, debido a que es muy probable que para cada elemento exista una ronda al menos en la que el estimado sea positivo, generando con ello una media final positiva. Además, los elementos finales generados no son buenos estimados debido a que son valores medios obtenidos a partir de cotas. Es posible reescalar la matriz $\widehat{A}$ para resumir el numero total de mensajes pero los estimados siguen sin ser precisos. Por otro lado, hemos encontrado que existe una relación lineal entre los elementos estimados y los valores reales.

Para obtener información relevante sobre las relaciones es necesario fijar los elementos cero más probables. Para cada

elemento, se estima la probabilidad de cero. Esto se hace calculando el porcentaje de tablas con ese elemento cero para cada ronda que el elemento está presente, y multiplicando las probabilidades obtenidas para todas esas rondas (el elemento será cero en la tabla final si es cero en todas las rondas). Se utilizan las siguientes expresiones si calculamos las probabilidades para el elemento (*i*, *j*) y se generan M tablas por ronda:

$$log\Big(p\Big(el\big(i,j\big)=0\Big)\Big) = -N_p \log(M) + \sum_{t=1,(i,j)p}^{T} \log(n_t^{(i,j)})$$

$n_t^{(i,j)}$= N^o de tablas con elemento *(i, j)* = 0 en la ronda *t*.

N_p= N^o de rondas con elemento *(i, j)* presente.

Los elementos de la tabla final se ordenan por su probabilidad de cero, a excepción de los elementos que ya son ceros triviales (elementos que representan pares de usuarios que nunca han coincidido en ninguna ronda). Los elementos cero menos probables son considerados candidatos a "relación existente". El objetivo principal del método es detectar con precisión: 1. Celdas que son cero con alta probabilidad (no relación $i \rightarrow j$). 2. Celdas que son positivas con alta probabilidad (relación $i \rightarrow j$).

Nuestro método de clasificación consiste en seleccionar la probabilidad de un punto de corte p (valores cercanos a 0,85 han dado buenos resultados en nuestras aplicaciones) y considerar clasificadas como "celdas cero" aquellas con probabilidad de cero $> p$, en tanto se considerarán clasificadas como "celdas positivas" aquellas con probabilidad de cero $< 1 - p$. El resto de celdas se considerarán como "no clasificadas". Este es un enfoque conservador del problema de clasificación, que se utiliza cuando es importante detectar elementos que pertenecen a ciertas clases con alta probabilidad, aunque el método conlleve a elementos no clasificados. Nuestro método es simétrico debido a que el intervalo de rechazo es determinado por un único valor p (puede también ser asimétrico, si el investigador lo desea, fijando diferentes puntos de corte en cada extremo). Por lo regular, en nuestras aplicaciones el porcentaje de celdas no clasificadas es menor del 15 %.

El algoritmo lleva a un test de clasificación binaria para los elementos diagnosticados, donde 0 en un elemento (*i*, *j*) significa no relación emisor-receptor de *i* a *j*, y 1 significa relación positiva emisor-receptor de *i* a *j*. Algunas métricas características para los tests de clasificación binaria son la sensibilidad, especificidad, valor predictivo positivo y valor predictivo negativo. Consideramos TP a los verdaderos positivos, FP a los falsos positivos, TN a los verdaderos negativos y FN a los falsos negativos:

Sensibilidad = $\frac{TN}{TN+FP}$ mide la capacidad del test para reconocer valores negativos verdaderos.

Especificidad = $\frac{TP}{TP+FN}$ mide la capacidad del test para reconocer valores positivos verdaderos.

Valor predictivo positivo = $\frac{TP}{TP+FP}$ mide la precisión del test en predecir valores positivos.

Valor predictivo negativo = $\frac{TN}{TN+FN}$ mide la precisión del test en predecir valores negativos.

No hay una manera perfecta de describir esta información con solo número. Para nuestro caso, donde el tamaño de las clases difiere, debido a que la tasa de negativos (valores 0) es muy alta comparada con la de positivos, se puede utilizar el coeficiente de correlación de Matthews para evaluar el desempeño del test, MCC, que se define así:

$$\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \quad (3)$$

Este coeficiente regresa un valor entre -1 y 1. Un coeficiente de $+1$ representa una predicción perfecta, 0 una predicción aleatoria y -1 una predicción inversa.

IV. Aplicación a datos de correo electrónico

Se realizaron un gran número de simulaciones a medida que el método era desarrollado, pero las especiales singularidades de los datos de correos electrónicos eran más apropiadas par medir la confiabilidad del método. Se utilizaron como base datos proporcionados por el Centro de Computación de la Universidad Complutense de Madrid, a fin de evaluar el funcionamiento del ataque. Se obtuvo el tiempo del envío, emisores y receptores (anónimos) para cada mensaje enviado durante un lapso de 12 meses, en un dominio restringido a una Facultad. Se determinaron la longitud de las rondas y tamaño de los lotes para evaluar el método. Fueron eliminados aquellos mensajes enviados de manera evidente a listas, mensajes institucionales y mensajes que provenían o eran enviados fuera del dominio.

Como un primer ejemplo, la Tabla IV presenta los resultados obtenidos para 4 meses. Consta de un total 97 emisores y 103 receptores. La complejidad de las rondas (el número de usuarios o dimensión de las tablas en las rondas) crece a medida que crece la longitud del intervalo de ronda o tamaño del lote. La tabla final agregada A tiene 9909 elementos. El punto de corte utilizado fue $p = 0,85$.

Tabla IV
Resultados de la Simulación

Tamaño del lote	% Falsos Negativos	% Falsos Positivos	Sensibilidad	Especificidad	MCC	% No clasificado
7	5	12	0,45	0,96	0,67	21
10	10	20	0,34	0,95	0,58	22
20	4	30	0,34	0,98	0,43	14
40	4	59	0,34	0,98	0,44	13
60	3	63	0,33	0,98	0,42	13
80	3	65	0,33	0,98	0,41	13

Los resultados empeoran cuando crece el tamaño del lote, y el atacante tendrá que obtener más información (rondas) para disminuir este efecto. Mientras que, los resultados son muy buenos para tamaños de lotes pequeños y permiten revelar algunas relaciones positivas así como un gran número de relaciones no existentes con un MCC = 0,67. Está claro que aumentar el tamaño del lote obliga al investigador a aumentar el punto de corte p, para evitar un alto porcentaje de falsos positivos, que es intolerable más allá de 50 %. Un punto

de corte más alto significa como consecuencia un mayor porcentaje de celdas no clasificadas. Cuando el tamaño del lote es grande se puede utilizar un punto de corte asimétrico.

Las siguientes figuras son presentadas en orden para estudiar la sensibilidad del método a variaciones en el punto de corte p, tamaño del lote y horizonte de datos recogidos. Las Figuras 2, 3, 4 se realizan para un horizonte de 4 meses, p = 0,20 y tamaño de lote 20. El número de tablas generadas por ronda afecta la precisión del método, pero menos de lo que intuitivamente se podría sospechar. Aún con tamaños grandes de lote, que dan lugar usualmente a espacios grandes de soluciones factibles, el generar más de 50000 tablas no mejora significativamente el método (Figuras 2 y 3). Las Figuras 5, 6 y 7 se realizan para p = 0,20 y tamaño de lote 20. A medida que la información obtenida crece en número de meses, la precisión del método mejora (Figura 5). La Figura 8 se realiza para un horizonte de 4 meses, p = 0,20. El tamaño del lote afecta de manera significativa la precisión del método. Cuanto más alto sea el tamaño, los resultados son peores pues la dimensión de las tablas es mayor y por lo tanto la complejidad del problema crece (Figura 8). El método de clasificación con opción de rechazo presentado es simétrico respecto a $\alpha = 1 - p$, y por lo tanto una curva ROC no es apropiada: la sensibilidad y especificidad crecen a medida que α decrece. Pero a la vez el porcentaje de celdas no clasificadas también se incrementa. El investigador debe decidir un punto de corte adecuado que no derive en un número alto de celdas no clasificadas. La Figura 9 muestra que trazando una línea vertical en $\alpha = 0,20$ $(p = 0,80)$ nos arroja un razonable 20 % de celdas no clasificadas, con sensibilidad de 0,45 y especificidad cerca de 0,98. La Figura 9 se realiza para un horizonte de 4 meses, tamaño de lote 20.

V. Conclusiones y Trabajo Futuro

Este trabajo presenta un método para detectar relaciones (o no relaciones) entre usuarios en un entorno de comunicaciones, donde la información obtenida es incompleta. Es el primer enfoque práctico al problema de revelación de datos de correos electrónicos, y, en nuestro conocimiento, es el primer trabajo en el cual se utilizan datos reales no simulados, para evaluar el rendimiento de ataque de revelación. Los resultados son alentadores pues se obtiene una alta especificidad y una moderada o alta sensibilidad, con un rango de celdas no diagnosticadas relativamente bajo. El método puede ser aplicarse a otras escenarios, como *pool mixes*, o situaciones donde se puede utilizar información adicional. Se ha utilizado también computación paralela con buenos resultados para acelerar el método. El ataque también puede ser utilizado en otros entornos de comunicaciones como redes sociales o protocolos *peer to peer*, y a problemas reales de de-anonimización que no tienen por qué ser del dominio de las comunicaciones, como revelar tablas públicas o investigación forense. Se necesita profundizar en la investigación, en los aspectos de la selección de puntos de corte p, el número óptimo de tablas a generar o incluir mejoras en la solución final, quizá rellenando celdas iterativamente y ciclando el algoritmo.

Agradecimientos

Los autores agradecen la financiación que les brinda el Subprograma AVANZA COMPETITIVIDAD I+D+I del Ministerio de Industria, Turismo y Comercio (MITyC) a través del Proyecto TSI-020100-2011-165. Asimismo, los autores agradecen la financiación que les brinda el Programa de Cooperación Interuniversitaria de la Agencia Española de Cooperación Internacional para el Desarrollo (AECID), Programa PCI-AECID, a través de la Acción Integrada MAEC-AECID MEDITERRÁNEO A1/037528/11.

Referencias

[1] D. Chaum, "Untraceable Electronic Mail, Return Addresses, and Digital Pseudonyms" *Communications of ACM*, Vol. 24 No. 2, pp. 84-88, 1981.

[2] C. Gulcu and G. Tsudik, "Mixing E-mail with Babel", *in Proceedings of the Network and Distributed Security Symposium (NDSS 96)*, pp. 2-16, February 1996.

[3] U. Moller, L. Cottrell, P. Palfrader, and L. Sassaman, "Mixmaster Protocol - version 2", *IETF Internet Draft*, July 2003, http://www.abditum.com/mixmasterspec.txt.

[4] G. Danezis, R. Dingledine, and N. Mathewson, "Mixminion: Design of a Type III Anonymous Remailer Protocol", *in Proceedings of the IEEE Symposium on Security and Privacy*, pp. 2-15, May 2003.

[5] TorStatus *Tor Network Status*, 2010. http://torstatus.cyberphunk.org.

[6] A. Serjantov and P. Sewell, "Passive Attack Analysis for Connection Based Anonymity Systems", *Computer Security - ESORICS 2003*, Springer-Verlag, LNCS 2808, pp. 116-131, October 2003.

[7] J. F. Raymond, "Traffic Analysis: Protocols, Attacks, Design Issues, and Open Problems", *in Proceedings of the International Workshop on Design Issues in Anonymity and Unobservability*, Springer-Verlag New York, Inc. pp. 10-29, 2001.

[8] M. Ercan, C. Clifton and E. Nergiz, "Multirelational *k-Anonymity", IEEE Transaction on Knowledge and Data Engineering*, Vol. 21, No. 8, pp. 1417-1421, 2009.

[9] D. Sacharidis, K. Mouratidis and D. Papadias, "*k*-Anonymity in Presence of External Databases", *IEEE Transaction on Knowledge and Data Engineering*, Vol. 22, No. 3, pp. 392-403, 2010.

[10] S. Kisilevich, L. Rokach, Y. Elovici and B. Shapira, "Efficient Multidimensional Suppression for K-Anonymity", *IEEE Transaction on Knowledge and Data Engineering*, Vol. 22, No. 3, pp. 334-347, 2010.

[11] G. Ghinita, P. Kalnis and Y. Tao, "Anonymous Publication of Sensitive Transactional Data", *IEEE Transaction on Knowledge and Data Engineering*, Vol. 23, No. 2, pp. 161-174, 2011.

[12] D. Agrawal and D. Kesdogan, "Measuring Anonymity: the Disclosure Attack". *IEEE Security & Privacy*, Vol. 1, No. 6, pp. 27-34, Nov.-Dec. 2003.

[13] G. Danezis, "Statistical Disclosure Attacks: Traffic Confirmation in Open Environments", *in Proceedings of the Security and Privacy in the Age of Uncertainty, (SEC2003)*, Kluwer, pp. 421-426, 2003.

[14] G. Danezis and A. Serjantov, "Statistical Disclosure or Intersection Attacks on Anonymity Systems", *Lecture Notes in Computer Science 3200*, pp. 293-308, 2005.

[15] G. Danezis, C. Diaz and C. Troncoso, "Two-sided Statistical Disclosure Attack", *in Proceedings of the 7th International Conference on Privacy Enhancing Technologies (PET' 07) LNCS 4776*, pp. 30-44, 2007.

[16] C. Troncoso, B. Gierlichs, B. Preneel and I. Verbauwhede, "Perfect Matching Disclosure Attacks", *in Proceedings of the 8th International Symposium on Privacy Enhancing Technologies (PET' 08), LNCS 5134*, pp. 223, 2008.

[17] A. Pfitzmann and M. Kihntopp, "Anonymity, Unobservability, and Pseudonymity - A Proposal for Terminology", *LNCS 2009*, pp. 1-9, 2001.

[18] L. Willenborg and T. Waal, "Elements of Statistical Disclosure Control", *Lecture Notes in Statistics*, Vol. 155, No. 15, pp. 261, 2001.

[19] A. Dobra and S. E. Fienberg, "Bounds for Element Entries in Contingency Tables Given Marginal Totals and Decomposable Graphs", *in Proceedings of the National Academy of Sciences of the United States of America*, Vol. 97 No. 22, pp. 11885-11892, 2000.

[20] Y. Chen, P. Diaconis, S. P. Holmes and J. S. Liu, "Sequential Monte Carlo Methods for Statistical Analysis of Tables", *Journal of the American Statistical Association*, Vol. 100, pp. 109-120, 2005.

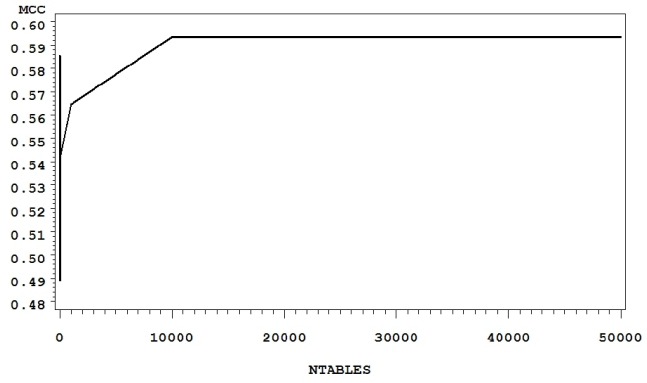

Figura 2. Coeficiente MCC vs N^o Tablas / Ronda.

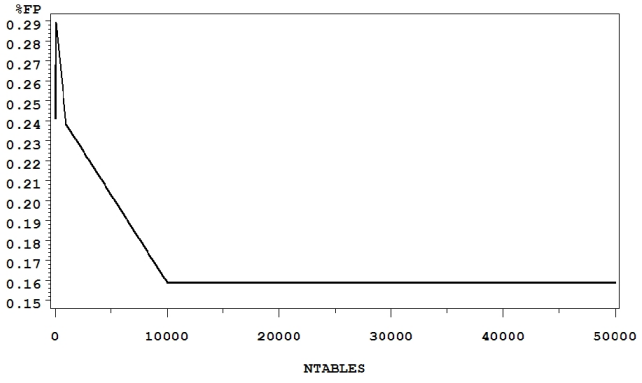

Figura 3. Tasa de Falsos Positivos vs número de tablas / Ronda.

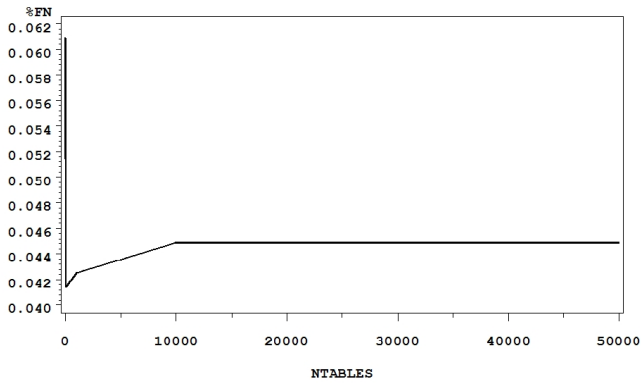

Figura 4. Tasa de Falsos Negativos vs N^o Tablas / Ronda.

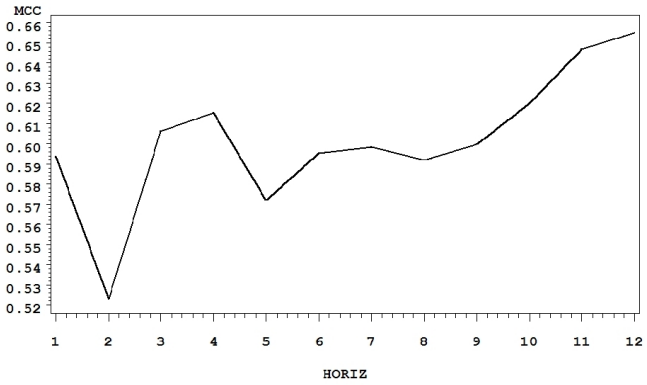

Figura 5. Coeficiente MCC vs Horizonte del Ataque.

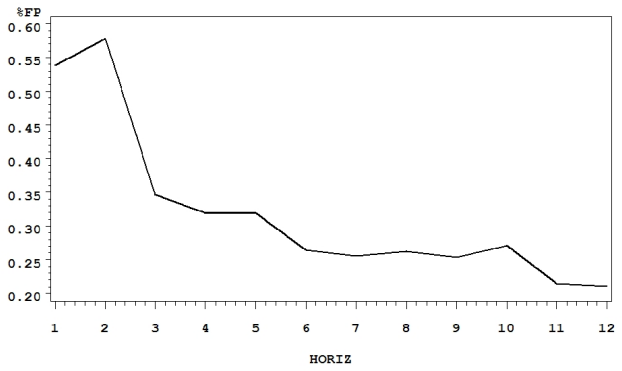

Figura 6. Tasa de Falsos Positivos vs Horizonte del Ataque.

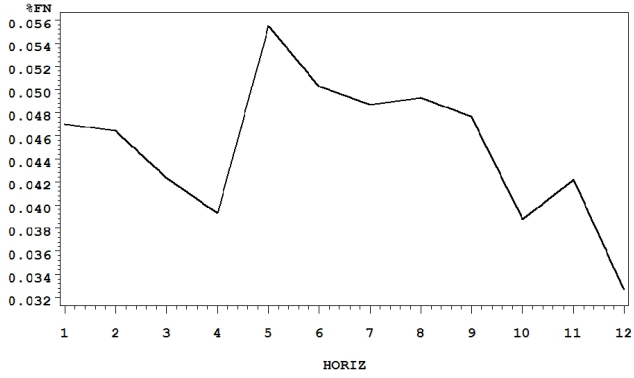

Figura 7. Tasa de Falsos Negativos vs Horizonte del Ataque.

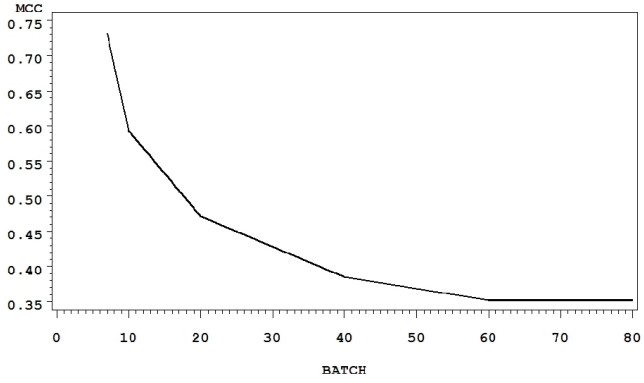

Figura 8. Coeficiente MCC Coefficient vs Tamaño de Lote.

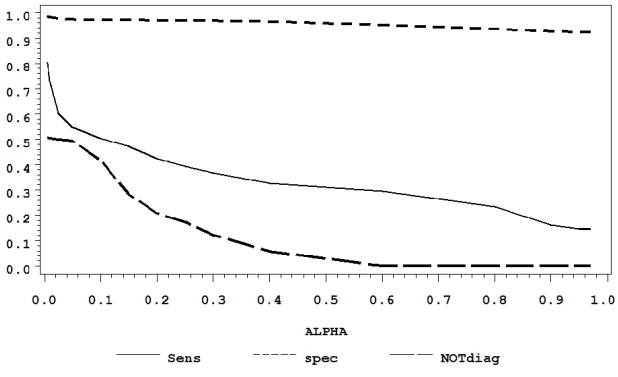

Figura 9. Sensibilidad, Especificidad y Celdas no Clasificadas vs $\alpha = 1 - p$.