FACTORES DE IDENTIFICACIÓN PARA USUARIOS Y DISPOSITIVOS MÓVILES

I. Buitrón Dámaso[1], G. B. Morales Luna[2], Departamento de Computación del Centro de Investigación y de Estudios Avanzados de Instituto Politécnico Nacional CINVESTAV-IPN, MEXICO
Recibido Noviembre 16, 2011 – Aceptado Marzo 15 de 2012
http://dx.doi.org/10.18566/puente.v6n1.a01

***Resumen*—Aquí se tratan algunos factores para determinar identificadores tanto de usuario como de dispositivos para aplicaciones móviles con el objetivo de identificación y autenticación. Los dispositivos pueden ser identificados por medio de los identificadores GSM y los usuarios por medio del número de suscriptor asignado por el proveedor de la red telefónica. También mostramos algunos pasos de implementación en la plataforma Android.**

***Palabras claves*— androide, identificación, imei, md5, móviles, sha-1.**

***Abstract*—We discuss some factor determining user and device identities over mobile applications in order to guarantee identity and authentication processes. Devices are determined through their GSM identifiers and users through their subscription numbers assigned by the network providers. We sketch some implementation steps in Android platforms.**

***Key words*— android, identification, imei, md5, mobiles, sha-1.**

I. INTRODUCCIÓN

Los servicios para los dispositivos móviles frecuentemente requieren de algún mecanismo para identificar a sus consumidores.

La identificación es el proceso para determinar a cada uno de los elementos de un conjunto y diferenciarlos entre sı. Desde el enfoque de servicios, el problema de identificación consiste en la asociación de dos conceptos: un identificador y un identificado, siendo este último un cliente o un consumidor del servicio.

El problema de identificación de usuarios de un servicio consiste en la implementación de identificadores que sean asociados al concepto de usuario.

El uso de identificadores propios de las infraestructuras de comunicaciones, así como de técnicas de verificación como el uso de las funciones hash proporcionan una forma para implementar mecanismos de identificación basados en estándares.

II. CONCEPTOS SOBRE SEGURIDAD

A. *Servicios de seguridad*

Para proveer seguridad en la comunicación de los dispositivos móviles se contemplan cuatro servicios de seguridad: autenticación, integridad, confidencialidad y no repudio.

1) Autenticación: Implica corroborar la identidad de una entidad (una persona, una computadora, etc.).
2) Confidencialidad: Este servicio garantiza que la información sólo está accesible para aquellos con autorización de acceso a la misma.
3) Integridad: Este servicio garantiza que la información no sea alterada por entidades no autorizadas.
4) No repudio: Este servicio tiene el objetivo de demostrar la entidad de procedencia de un mensaje.

B. *Problemática en la identificación*

Determinar un identificador para servicios que serán consumidos por dispositivos móviles implica cuestionarse: ¿Qué se desea identificar?

Dependiendo del sistema donde la identificación se desee implementar, un usuario puede ser representado por el dispositivo móvil u otro objeto relacionado al propio usuario.

Para ejemplificar el problema de identificación, se pueden plantear dos escenarios prácticos, en los cuales un dispositivo móvil realiza una solicitud a una computadora que le provee cierto servicio:

[1] I. Buitrón es estudiante de la Maestría en Ciencias en Computación en el Departamento de Computación del Centro de Investigación y de Estudios Avanzados de Instituto Politécnico Nacional. Teléfono +52 (55) 3927-4486. e-mail: ibuitron@computacion.cs.cinvestav.mx.

[2] G. Morales es profesor titular en el Departamento de Computación del Centro de Investigación y de Estudios Avanzados de Instituto Politécnico Nacional. Teléfono +52 (55) 5747-3759, Extensión: 6565. e-mail: gmorales@cs.cinvestav.mx.

1) Un dispositivo envía su identificador adjunto a un requerimiento emitido, estos datos son recibidos en el servidor y es posible identificar al usuario que emitió el requerimiento por medio de dicho identificador ya que existe registrada una relación de pertenencia entre el usuario y el dispositivo en el servidor.
2) Un usuario emite un requerimiento desde un dispositivo diferente al que se encuentra asociado en la relación de pertenencia. Este caso plantea la disyuntiva de decidir si el requerimiento es o no válido.

En estos dos casos surge la necesidad de considerar dos tipos de identificadores: los de usuarios y los de dispositivos.

III. IDENTIFICADORES

Dentro de la infraestructura de servicios de telefonía celular existen identificadores que los mismos proveedores usan y están definidos en los estándares y protocolos de comunicación.

A continuación se presentan algunos identificadores propios de las redes GSM (Global System for Mobile Communications).

A. IMEI

Cada dispositivo tiene relacionado un número llamado IMEI (International Mobile Equipment Identity) para identificar teléfonos GSM, WCDMA (Wideband Code Division Multiple Access), iDEN (Integrated Digital Enhanced Network) y satelitales. En las redes GSM, el IMEI es usado para identificar los dispositivos válidos y puede ser usado para denegar el servicio por medio de listas negras que cada proveedor de servicios telefónicos implemente en caso de recibir un reporte de robo generalmente por parte del suscriptor, es usado sólo para identificación del dispositivo y no tiene relación permanente con el suscriptor de la línea, consta de 15 dígitos decimales donde se incluye información del modelo, fabricante y número de serie del dispositivo [1] [2].

Consta de 4 partes:

- TAC (Type Allocation Code) [3], indica en los primeros dos dígitos el RBI (Reporting Body Identifier).
- FAC (Final Assembly Code), indica el fabricante del equipo.
- Número de serie del teléfono.
- Digito verificador.

Se tiene que cada IMEI puede expresarse mediante 50 bits. Resulta importante mencionar que el IMEI no identifica de manera unívoca a los dispositivos, ya que existe la posibilidad de alterar el valor de este identificador y como consecuencia la posibilidad de clonación de IMEIs [4], [5], [6] [13]. En esta propuesta se requiere de un identificador de dispositivo y utiliza al IMEI para formar un identificador, sin embargo, el uso del IMEI puede ser sustituido por otro identificador de dispositivo y por lo tanto no es esencial en la propuesta.

B. Número del suscriptor

Un identificador de usuario puede ser el número de suscriptor con el proveedor de telefonía celular (SusID). Esto implica que el concepto de identidad de usuario se vincula al número del suscriptor. Así, la identificación del usuario se vincula a la tarjeta SIM, en el caso de pérdida de dicha tarjeta en las redes GSM, los proveedores de telefonía celular tienen la capacidad de proporcionar una tarjeta nueva con los datos de la tarjeta anterior y como consecuencia los datos de identificación del usuario permanecen sin cambios y no dependen del dispositivo.

IV. DISEÑO FORMAL E IMPLEMENTACIÓN

A. Diseño Formal

Sea U un espacio de identificadores de usuario, D un espacio de identificadores de dispositivo y $P(t)$ una relación temporal de pertenencia, es decir $P(t) \subset U \times D$, que cumple con las siguientes características en todo tiempo t :

- Todo usuario tiene un dispositivo:
  $\forall u \in U \exists d \in D : (d,u) \in P(t)$
- Un dispositivo sólo puede ser usado por un usuario:
  $$\forall d \in D \forall u,v \in U :$$
  $$\left[(d,u) \in P(t) \wedge (d,u) \in P(t)\right] \Rightarrow u = v$$

Estas dos propiedades quedan resumidas planteando que $P(t)$ es una función $D \rightarrow U$ que es suprayectiva.

Consideremos por el momento un conjunto E de etiquetas. Una función de etiquetado es del tipo $\varphi : U \times D \rightarrow E$. Decimos que φ distingue

- *usuarios si* $\forall (u_0, d_0), (u_1, d_1) \in U \times D :$
  $(u_0, d_0) = (u_1, d_1) \Rightarrow u_0 = u_1$

- *dispositivos si* $\forall(u_0,d_0),(u_1,d_1)\in U\times D$:
  $(u_0,d_0)=(u_1,d_1)\Rightarrow d_0=d_1$
- *parejas de usuarios-dispositivos si*
  $\forall(u_0,d_0),(u_1,d_1)\in U\times D$:
  $(u_0,d_0)=(u_1,d_1)\Rightarrow (d_0=d_1)\wedge(u_0=u_1)$

Sea ahora B = Bytes el alfabeto que coincide propiamente con el ASCII, sea B^* el conjunto de palabras de bytes y $B^+=B^*-\{nil\}$. Diremos que para $E=B^+$ una función de etiquetado $\varphi: U\times D\to B+$ es una de identificación. Por otra parte, sean K_d, K_e dos espacios de claves públicas y claves privadas, respectivamente. En todo esquema de cifrado de clave pública existe un conjunto $K\subset K_d\times K_e$ consistente de parejas de claves (pública–privada). Para $E=K$, una función de registro es una de etiquetado $\psi: U\times D\to K$. La noción de identificación puede ser usada como sinónimo de autenticación de una entidad, o alternativamente, también es usada para referirse al proceso de asignación de una identidad, sin que esto conlleve un medio de comprobación de la entidad identificada [7]. Convendremos aquí en llamar identificación a esta segunda noción, en tanto que autenticación involucra un proceso de verificación de identidad.

Recordamos que un esquema de cifrado se construye como sigue: sean M el espacio de mensajes y C el espacio de textos cifrados, ambos definidos sobre el alfabeto B. Una función de cifrado E de textos es del tipo $E: M\times K_e\to C$, con una correspondiente función de descifrado $D: C\times K_d\to M$, tales que $\forall(k_e,k_d)\in K$ se cumple:

- $\forall m\in M: D(E(m,k_e),k_d)=m$
- $\forall c\in C: E(D(c,k_d),k_e)=c$

Recordamos también que un esquema de firma se construye como sigue: sea S un espacio de firmas y sea f una función de firmado $M\times K_d\to S,(m,k_d)\to s$. Si $s=f(m,k_d)\in S$ se dice que s es la firma del mensaje $m\in M$, usando la clave privada k_d. Es convencional que tanto el espacio de mensajes M como el de firmas S coincidan con el de palabras en ASCII B^*. En el esquema de firma, también se tiene una función de verificación de firma:

$v: M\times S\times K_e\to\{0,1\}$

de manera que $\forall(m,s,k_e)\in M\times S\times K_e$:

$v(m,s,k_e)=1\Leftrightarrow s=f(m,k_d)$

Donde k_d es la clave privada y k_e su pública correspondiente.

B. Políticas de implementación

1) Funciones *hash*: Una función *hash* es un procedimiento determinístico que toma como parámetro de entrada una cadena de longitud variable o mensaje y produce como salida una cadena de longitud constante o valor *hash*, es pues del tipo $h:\{0,1\}\to\{0,1\}^n$, donde n es un parámetro fijo de longitud. Una función *hash* debe cumplir idealmente con las siguientes características:

- Es fácil computar el valor *hash* dado un mensaje: Computable(m) ⇒ Computable($h(m)$)
- Es inviable computar el mensaje dado su valor *hash*. Computable($h(m)$) =⇒ Computable(m) (12)
- Es inviable modificar un mensaje sin alterar su valor *hash*. $m=m\Rightarrow h(m)=h(m)$ (13)
- Es computacionalmente complejo encontrar dos mensajes con un mismo valor *hash*. $h(m_1)=h(m_2)\Leftrightarrow m_1=m_2$ (14) Cuando dos mensajes diferentes que generen un mismo valor *hash*, se conoce como una colisión de la función.

Las características anteriores de las funciones *hash* hacen que sean ampliamente usadas en aplicaciones de seguridad informática, tales como firmas digitales, MACs (*Message Authentication Codes*) [11] [12] y otras formas de autenticación.

Se considera <u>*quebrantada*</u> a una función *hash*, cuando se ha encontrado alguna violación a dichas características como el caso de las colisiones.

2) Identificadores para implementación: Según se puede observar en la sección IV-A, el problema de identificación reside principalmente en definir la función de identificación φ a implementar. La identificación puede ser realizada evaluando o la unión de los dos identificadores o éstos por separado.

En el caso específico de la plataforma Android, el uso del número de suscriptor como identificador de usuario y el uso del IMEI como identificador de dispositivo, no están restringidos, sin embargo existe la posibilidad de que en otras plataformas pueda existir restricción en la transmisión de dichos datos.

TABLA I:
LONGITUD DE SALIDA DE LAS FUNCIONES *HASH*

Función	***Longitud (bits)***
MD5	128
SHA-1	160
SHA-256	256
SHA-512	512

C. Uso de funciones hash

Se propone el uso de funciones *hash* con el objetivo de evitar la transmisión de dichos números pero sin perder la posibilidad de realizar verificación con ellos. Las funciones *hash* con mayor difusión son MD5 [8], [9] (*Message-Digest algorithm 5*) y el grupo de funciones SHA [10] (*Secure Hash Algorithm*). Se sugiere el uso de ellas con el propósito de verificar estos identificadores, debido a que han sido implementadas nativamente en una gran cantidad de dispositivos y eso facilita la integración con muchos sistemas de cómputo. La longitud del resultado depende de la función que se aplique, en la Tabla I se puede observar una comparativa de 4 funciones comunes documento.

Se sugiere el uso de la función MD5 o SHA-1 [14] por su longitud corta. A pesar de que éstas se consideran funciones quebrantadas en la actualidad, la aplicación de cualquiera de ellas no requiere que ésa ofrezca una seguridad similar a las funciones que no han sido quebrantadas, ya que el espacio de valores de una función *hash* es considerablemente más grande que el espacio de identificadores esperados. Por otra parte el diseño que en este artículo se propone no se limita al uso de MD5 o SHA-1, de modo que puede usarse cualquier otra función *hash* (por ejemplo, el grupo de funciones SHA-2 [10]), considerando sus características particulares en el momento de la implementación. En las siguientes dos secciones bosquejamos la implementación de funciones de etiquetado que distinguen dispositivos, usuarios o parejas de dispositivos y usuarios.

D. Método AND

El método AND tiene como objetivo considerar a la unión de los dos identificadores para obtener un resultado y este valor será el objeto de verificación, haciendo uso del diseño especificado en la sección IV-A para este método la función de etiquetado φ ha de distinguir parejas de dispositivos y usuarios, y por lo tanto ha de cumplir con la implicación (6). Se propone aplicar una función *hash* a la suma de los dos identificadores:

UID = Hash(SusID + IMEI)

La ventaja principal de este método es que se obtiene un identificador (UID) más corto. La desventaja principal es que la ausencia o alteración de cualquiera de los dos identificadores tiene como consecuencia que el UID no verifique.

En teoría eso es un punto favorable, pero en la práctica implica que los requerimientos sólo pueden ser emitidos por el usuario registrado, con su dispositivo registrado, para ser válidas. Ver Fig. 1.

Algoritmo 1: Validación AND

```
Entrada: UID' = Hash(SusID) XOR Hash(IMEI)
{UID a
validar}

if UID = UID' then
  return true
else
  return false
end if
```

Fig.1 Algoritmo de validación AND

A pesar que en el diseño se hace referencia a un modelo ideal, donde un usuario siempre usa su propio dispositivo, en la práctica es común que un usuario haga uso de un dispositivo diferente al suyo, por ejemplo, cuando un usuario intenta hacer una llamada usando su tarjeta SIM desde otro dispositivo. Este caso puede ser común y es válido. De modo que se considera hacer una validación de los identificadores por separado.

E. Método OR

Este método se encuentra contemplado en el diseño de la sección IV-A, donde la función de identificación φ cumple con 4 la implicación (4) o con la (5) para realizar la verificación de cada identificador, dejando la posibilidad a que sólo un identificador sea válido y delegando la decisión de considerar como válida la alerta a las políticas de operación del proyecto.

Para este caso se propone que el UID se construya mediante la concatenación de los resultados de aplicar las funciones *hash* a los identificadores:

UID = Hash (SusID)||Hash(IMEI)

El costo de aplicar este método es la longitud del UID (será el doble de la longitud de la salida de la función *hash*), pero a cambio se obtiene la posibilidad de verificar cada identificador por separado. Ver Fig. 2.

Algoritmo 2: Validación OR

```
Entrada: UID' = Hash(SusID') XOR
Hash(IMEI') {UID a
 validar}

if Hash(SusID) = Hash(SusID') then
  return true  {Usuario válido}
else if Hash(IMEI) = Hash(IMEI') then
  return true{Dispositivo válido}
else
  return false
end if
```

Fig. 2. Algoritmo de validación OR

V. CONCLUSIONES

Los dispositivos y los usuarios de éstos no necesariamente pueden ser representados por un identificador, de modo que surge la necesidad de considerar identificadores de usuario e identificadores de dispositivo.

Se propone usar el IMEI como identificador de dispositivo y el número de suscriptor como identificador de usuario, pero la existencia de políticas que prohíban la transmisión de estos datos restringe su uso y lleva a utilizar un mecanismo para poder hacer verificación de estos dos datos sin la necesidad de transmitirlos. Las funciones *hash* pueden ser usadas para verificar el identificador mediante su valor *hash*.

El factor de decisión de la validez de una alerta está en función de la verificación tanto del identificador de usuario como del identificador de dispositivo. Se propone un método que crea un identificador en función de la suma de los dos identificadores que resulta de longitud corta pero no permite validar los identificadores por separado. Se propone un segundo método en el que se crea un identificador que es el resultado de la concatenación de los dos identificadores y como consecuencia permite la verificación de los identificadores por separado y delega la decisión de validez a las políticas de implementación del proyecto.

AGRADECIMIENTOS

El presente trabajo ha sido apoyado parcialmente por la Corporación Universitaria para el Desarrollo de Internet, A.C., y el Consejo Nacional de Ciencia y Tecnología (no. de ref. C430/039/10), ambas instituciones establecidas en la República Mexicana.

REFERENCIAS

[1] G. Association, "IMEI allocation and approval guidelines, version 5.0," GSM Association, Tech. Rep., september 2010, document explaining IMEI in detail from GSM Association. [Online]. Available: http://www.gsmworld.com/documents/DG06 v5.pdf

[2] D. M. A. Allen and P. Gosden, "A uniform resource name namespace for the GSM Association (GSMA) and the international mobile station equipment identity (IMEI)," Research in Motion (RIM) and GSM Association, Tech. Rep., january 2011. [Online]. Available: http://tools.ietf.org/html/draft-montemurro-gsma-imei-urn-06

[3] 3GPP, "Release 99/4 CRs to 22.016 on Type Approval Code," 3GPP, Tech. Rep., june 2002, 3GPP Change Request re Type Allocation Code. [Online]. Available: http://www.3gpp.org/ftp/tsg sa/TSG SA/TSGS 16/Docs/PDF/SP-020237.pdf

[4] V. S. Rawat, "Industry wants ban on handsets with cloned IMEI number," Business Standard, June 2011. [Online]. Available: http://www.business-standard.com/india/news/industry-wants-banhandsetscloned-imei-number/372543/

[5] S. López, "Google corregirá dispositivos de Android infectados por malware remotamente," BSecure, March 2011. [Online]. Available: http://www.bsecure.com.mx/featured/dispositivos-infectados-seran-solucionados-de-manera-remota-google/

[6] ——, "Tras la tendencia, cibercriminales se mudan de la PC al dispositivo móvil," BSecure, March 2011. [Online]. Available: http://www.bsecure.com.mx/featured/cibecriminales-siguen-la-tendencia-se-mudan-de-la-pc-al-dispositivo-movil/

[7] H. C. van Tilborg, Encyclopedia of Cryptography and Security, 2nd ed. 233 Spring Street, New York, NY 10013, USA: Springer Science+Bussiness Media Inc., 2005.

[8] R. Rivest, "The MD5 Message-Digest Algorithm," MIT Laboratory for Computer Science and RSA Data Security, Inc., Tech. Rep., April 1992. [Online]. Available: http://tools.ietf.org/html/rfc1321

[9] S. Turner and L. Chen, "Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms," IECA and NIST, Tech. Rep., March 2011. [Online]. Available: http://tools.ietf.org/html/rfc6151

[10] N. I. of Standards and Technology, "Secure hash standard," Federal Information Processing Standards Publication, Tech. Rep., October 2008, this standard specifies five secure hash algorithms - SHA-1, SHA-224, SHA-256, SHA-384, and SHA-512 - for computing a condensed representation of electronic data. [Online]. Available: http://csrc.nist.gov/publications/fips/fips180-3/fips180-3_final.pdf

[11] N. I. of Standards and Technology, "Computer data authentication," Federal Information Processing Standards Publication, Tech. Rep., may 1985. [Online]. Available: http://www.itl.nist.gov/fipspubs/fip113.htm

[12] H. Krawczyk and M. Bellare and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", IBM and UCSD, Tech. Rep., February 1997. [Online]. Available: http://tools.ietf.org/html/rfc2104

[13] Cameron McDonald, Philip Hawkes and Josef Pieprzyk, "SHA-1 collisions now 252", Macquarie University and Qualcomm, Tech. Rep. [Online]. Available: http://eurocrypt2009rump.cr.yp.to/837a0a8086fa6ca714249409ddfae43d.pdf

[14] D. Eastlake and P. Jones, “US Secure Hash Algorithm 1 (SHA1)” Motorola and Cisco Systems, September 2001. [Online]. Available: http://tools.ietf.org/html/rfc3174

BIOGRAFÍA

Israel Buitrón Dámaso Es estudiante de la Maestría en Ciencias en Computación en el Departamento de Computación del CINVESTAV-IPN, es Ingeniero en Sistemas Computacionales por la ESCOM-IPN. Sus áreas de interés son la Criptografía y la Seguridad en los Sistemas de Información. Es mexicano por nacimiento.

Guillermo Benito Morales Luna Es Investigador Titular en el Departamento de Computación del CINVESTAV-IPN, es Licenciado en Física y Matemáticas por la ESFM-IPN, Maestro en Ciencias con especialidad en Matemáticas por el CINVESTAV-IPN, y Doctor en Ciencias Matemáticas por el Instituto de Matemáticas de la Academia Polaca de Ciencias, en Varsovia, Polonia. Sus áreas de interés son los Fundamentos Matemáticos de Computación, Lógica y Deducción Automática, Criptografía y Teoría de la Complejidad. Ha sido profesor en el IPN y en la Benemérita Universidad Autónoma de Puebla. Ha realizado dos estancias sabáticas en el Instituto Mexicano del Petróleo. Es mexicano por nacimiento y le fue concedida la ciudadanía polaca.