

Blue MAC Spoofing: El Backdoor de Bluetooth

Castillo, Carlos., Gómez-Casseres, Jose Luis y Torres, Edgar.
{carlos-castillo,jl.gomez,edgar-torres}@javeriana.edu.co
Pontificia Universidad Javeriana, Bogotá D.C, Colombia

Resumen—El siguiente documento es un análisis del ataque de seguridad informática Blue MAC Spoofing en dispositivos móviles. Se inicia con una introducción de la tecnología de telefonía celular, incluyendo el protocolo de comunicación Bluetooth y sus mecanismos de seguridad.

Luego de esto se expone la realización del ataque, cómo se realiza y cuales de las políticas de seguridad del protocolo son vulneradas. Por último realizamos un análisis forense buscando rastros en el celular afectado para poder identificar el posible atacante y que acciones realizó en el dispositivo sin autorización. Al final del artículo se exponen las conclusiones y las consecuencias de este fallo de seguridad en los dispositivos móviles actuales.

Índice de Términos—Bluetooth, Spoofing, Seguridad, Forense.

I. INTRODUCCIÓN

Es muy común en estos días conectar nuestro celular con Bluetooth a un manos libres inalámbrico o a un dispositivo móvil para transferir un ringtone o alguna foto. Cada vez los dispositivos móviles traen funcionalidades avanzadas que los convierten en verdaderos gadgets en los cuales están contenidos, en un mismo aparato, servicios como cámara digital, reproductor mp3 entre otros.

Es por esto que se hace necesario un medio de comunicación con los siguientes objetivos:

“

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre estos.
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas y facilitar la sincronización de datos entre nuestros equipos personales.” [1]

Usualmente los usuarios mantienen estas conexiones abiertas debido a múltiples razones;

una de las principales es que la mayoría de las veces se desconoce el riesgo de sufrir un ataque en su dispositivo. El riesgo del que estamos hablando es que al dejar la conexión abierta, se está permitiendo el acceso a puertas traseras que admiten el ingreso y control del dispositivo.

“Puesto que los mecanismos de seguridad del Bluetooth son a nivel de usuario, llegar a suplantar la identidad de un dispositivo acoplado para acceder a un teléfono sin ser detectado es lo que se conoce como un ataque Blue MAC Spoofing” [2]

En el presente artículo se presentará una vulnerabilidad en la seguridad de la tecnología Bluetooth: el ataque Blue MAC Spoofing. Este hace uso de una falla en el estándar de la tecnología Bluetooth, la cual se explica mas adelante. “El ataque Blue MAC Spoofing permite suplantar la identidad de un dispositivo de confianza y/o enlazado para atacar un teléfono móvil y utilizar sus credenciales para acceder a perfiles que requieren autorización y/o autenticación.” [2]

II. FUNCIONAMIENTO DE LA TECNOLOGÍA BLUETOOTH [3]

El núcleo del sistema Bluetooth consiste en un transmisor de radio, una banda base y una pila de protocolos. El sistema permite la conexión entre dispositivos y el intercambio de distintos tipos de datos entre ellos. [3]

Esta tecnología funciona en la banda de 2.4 GHz libre para ISM (Industrial, Scientific and Medical). El sistema utiliza un transmisor de salto de frecuencia el cual sirve para reducir las interferencias y para disminuir la intensidad de la señal. Bluetooth admite una velocidad de transmisión de 1 Mbps en el modo de velocidad básica y una velocidad de transmisión aérea total

de 2 a 3 Mbps en el modo de transferencia mejorada (EDR).

El nombre que recibe una red Bluetooth es "piconet". Esta compuesta por un dispositivo maestro y varios dispositivos esclavos. Todos ellos comparten el canal físico y están sincronizados por un reloj y una secuencia de salto de frecuencia. Sin embargo, sólo el dispositivo maestro proporciona los valores de referencia.

La tecnología Bluetooth permite la transmisión bidireccional utilizando la técnica de acceso múltiple por división de tiempo (TTD). En el canal físico se encuentra una capa de enlaces y canales con sus respectivos protocolos de control. La jerarquía de abajo hacia arriba de los canales y enlaces es la siguiente: canal físico, enlace físico, comunicación lógica, enlace lógico y canal L2CAP.

- **Canal físico:** Se crea un enlace de este tipo entre un dispositivo esclavo y uno maestro, pero dos esclavos no pueden conectarse directamente de esta forma.

- **Enlace físico:** Se utiliza como medio de comunicación entre uno o dos enlaces lógicos que admiten tráfico síncrono, asíncrono e isócrono de unidifusión, y tráfico de difusión. El tráfico de los enlaces lógicos se multiplexa en el enlace físico ocupando las ranuras asignadas por el programador del gestor de recursos.

- **Enlaces lógicos:** Se utilizan para transportar el protocolo de control de la banda base y las capas físicas. Esto se conoce con el nombre de LMP: Protocolo de gestión de enlace. Esta comunicación es la que se establece cuando un dispositivo se une a una piconet.

- **L2CAP:** Esta capa se encuentra por encima de la banda base y su objetivo es ofrecer una abstracción de los canales de comunicación a los servicios y las aplicaciones. También es la encargada de la unificación de los datos mediante la multiplexación y desmultiplexación de varios canales.

La pila de protocolos que conforma el núcleo del sistema Bluetooth está compuesta por:

- Protocolo de radiofrecuencia (RF)
- Protocolo de control de enlace (LCP)
- Protocolo de gestión de enlace (LMP)
- Protocolo de adaptación y de control de enlace lógico (L2CAP)

Para más información sobre la especificación y la arquitectura del protocolo Bluetooth puede ingresar al blog de Gospel en donde se explica detalladamente este tema. [4]

III. VENTAJAS DE LA TECNOLOGÍA BLUETOOTH

"La tecnología inalámbrica Bluetooth es líder en el mercado y es la única de corto alcance de la que se distribuyen más de cinco millones de unidades todas las semanas, con una base fija de más de 500 millones de unidades a finales del año 2005." [5]

Es por esto que vale la pena exponer las diferentes ventajas que ofrece esta tecnología ya que, aunque es vulnerable, ofrece muchas preeminencias que no podemos dejar de aprovechar:

- Disponibilidad:** La especificación del estándar Bluetooth está disponible de forma gratuita para una gran variedad de empresas en todo el mundo. Una gran diversidad de fabricantes están implementando esta tecnología en numerosos dispositivos. Distintos sectores industriales están implantando esta tecnología en sus productos para reducir el número de cables y lograr conexiones sin interrupciones, transmitir sonido estéreo, transferir datos o establecer comunicaciones de voz.

- Tipos de dispositivos:** La tecnología Bluetooth se encuentra disponible en una gran variedad de tipos de dispositivos. Celulares, GPS, Impresoras, PDA, Laptops etc.. Debido a su bajo costo y su reducido tamaño, esta tecnología se puede implementar en dispositivos de tamaño muy reducido lo cual lo hace muy atractivo para las

compañías fabricantes de dispositivos móviles.

-Facilidad de uso: Bluetooth es una tecnología que no necesita infraestructura fija y es sencilla de instalar y configurar. La mayoría de drivers que existen para Windows por ejemplo son muy intuitivos para los usuarios y en materia de celulares, el envío de archivos es cuestión de unos pasos muy fáciles de realizar.

- **Conexiones seguras:** Bluetooth se ha diseñado, desde un principio, pensando en seguridad. La tecnología de salto adaptable (AFH) permite en cierto modo esta seguridad ya que la señal “salta” y limita las interferencias con otras señales, además de que cuenta con un cifrado de 128 bits y la autenticación mediante código PIN.

Vale la pena aclarar esta última “ventaja” que no es del todo cierta. En realidad este documento muestra como la seguridad de autenticación mediante el código PIN es vulnerada por el ataque Blue MAC Spoofing [2]. Con respecto al salto de frecuencia, anteriormente era un impedimento para poder sniffear una red que utiliza la tecnología Bluetooth pero esto ya no es del todo cierto porque ya existe un software que puede crear Sniffers Bluetooth caseros a bajo costo. [6]

IV. MECANISMOS DE SEGURIDAD EN BLUETOOTH

A continuación se exponen los mecanismos de seguridad de Bluetooth para poder entender de una mejor manera de qué vulnerabilidades se aprovecha nuestro ataque.

- **Autenticación:** “Es el proceso por el cual un dispositivo Bluetooth verifica su identidad en otro dispositivo para poder acceder a los servicios que ofrece.” [7]. Este proceso consiste en crear una clave de enlace común de forma segura.
- **Autorización:** “Es el procedimiento que determina los derechos que tiene un dispositivo Bluetooth para acceder a los servicios que ofrece un sistema.”[8] Este procedimiento se lleva a cabo a través de

niveles de confianza. Esta funcionalidad se maneja desde una lista de dispositivos de confianza, los cuales no necesitan autorización futura para utilizar servicios en el dispositivo acoplado.

V. TECNOLOGÍA GSM

Uno de los campos de aplicación de la tecnología Bluetooth es la telefonía móvil GSM, que significa Global System for Mobile Communications, (Sistema Global para Comunicaciones Móviles). Como su nombre lo indica, esta tecnología trae consigo el objetivo de poder brindarle a las personas comunicaciones a nivel mundial.

Lo que se pretende lograr con la tecnología GSM es una especie de roaming internacional, algo más global, que no sólo se refiera a un país o ciertas zonas específicas del mismo. Es como tener el mismo número para más de 150 países ya que es una tecnología satelital. A pesar de que empezó a desarrollarse desde hace más de 10 años, hasta ahora es que está empezando a ser utilizada en todo el mundo.

La ventaja de la tecnología GSM no sólo esta en que se pueden mandar mensajes de texto sino que también se pueden enviar pequeños archivos, como fotos, mensajes de voz y timbres o sonidos. Todo esto es con equipo inalámbrico apoyado en operaciones satelitales.

Los teléfonos GSM también son conocidos como teléfonos de Tercera Generación, aunque esto no es totalmente cierto. Los teléfonos de Tercera Generación están basados en la tecnología GSM pero son más avanzados aún. Éstos ofrecen transmisión de video en línea, acceso a Internet de alta velocidad, y en general la calidad y capacidad es mucho mayor.

Otra de las ventajas que presenta tener un teléfono GSM es que utiliza tecnología confiable y segura, que ha sido desarrollada por expertos a nivel mundial. Todo esto es con el fin de evadir riesgos por el uso de celulares, para evitar más controversia acerca de las emisiones y si dañan o no al cuerpo humano o si producen enfermedades.

VI. SIM CARD

“Una tarjeta SIM (Subscriber Identity Module, ‘Módulo de Identificación del Suscriptor’) es una tarjeta inteligente desmontable usada en teléfonos móviles que almacena de forma segura la clave de servicio del suscriptor usada para identificarse ante la red, de forma que sea posible cambiar la línea de un terminal a otro simplemente cambiando la tarjeta.” [9] El uso de la tarjeta SIM es obligatorio en las redes GSM.

La tarjeta SIM GSM tiene poca memoria, alrededor de 2 a 3 KB y éste reducido espacio es utilizado directamente por el teléfono. Sin embargo, existen SIM con aplicaciones adicionales que amplían la capacidad de éstas, llegando a ser mayor a 512 KB. Finalmente debemos anotar que las tarjetas SIM menores, de 32 KB y 16 KB son las predominantes en zonas de redes GSM menos desarrolladas.

“Las tarjetas SIM almacenan información específica de la red usada para autenticar e identificar a los suscriptores en ella, siendo la más importante el ICC-ID, el IMSI, la clave de autenticación (Ki) y la identificación de área local (LAI). La tarjeta SIM también almacena otros datos específicos del operador como el número del SMSC (centro de servicio de mensajes cortos), el nombre del proveedor de servicio (SPN), los números de servicio de marcado (SDN) y las aplicaciones de servicios de valor añadido (VAS).” [9]

“La clave de autenticación (Ki, Authentication key) es un valor de 16 bytes usado para autenticar las tarjetas SIM en la red móvil. Cada tarjeta SIM tiene una Ki única asignada por el operador durante el proceso de personalización. La Ki también se almacena en una base de datos (conocida como HLR, acrónimo de Home Location Register) de la red del operador.” [9]

Las tarjetas SIM son en sí, un medio de almacenamiento de información del celular y por tanto es importante revisarla ya que a veces se encuentran datos valiosos sobre la información, configuración y autenticación del teléfono móvil.

VII. EJECUCIÓN DEL ATAQUE

El ataque fue realizado utilizando dos teléfonos celulares Nokia 6131 y un portátil HP Pavilion dv6120LA con Mandriva Linux Spring 2007 instalado y un dispositivo Bluetooth V2.0 Encore compatible con la pila de protocolo para Bluetooth de Linux BlueZ [21]

A. Fase de emparejamiento

La primera fase del ataque consiste en el emparejamiento de dos dispositivos móviles. En este suceso se crea entre los dos aparatos una clave de enlace común de una forma segura intercambiando un código de seguridad Bluetooth el cual se conoce comúnmente como pin.

“A partir de este código PIN Bluetooth, la dirección BD_ADDR de cada dispositivo y varios números aleatorios de 128 bits se obtiene la clave de enlace común a ambos dispositivos a través de los algoritmos E22 y E21”[2].

El proceso varía dependiendo de la marca y modelo del teléfono celular ya que, junto con esto, también varía el firmware o sistema operativo del dispositivo. En nuestro caso, con el Nokia 6131, vamos a Ir a -> Menú Bluetooth -> Dispositivos Acoplados -> Opciones -> Acoplar nuevo dispositivo. En ese momento se intercambian los PIN y se crea el enlace comentado en el párrafo anterior.

Una vez los dispositivos se encuentran emparejados, ya los dos tienen su respectiva clave de enlace y pueden autenticarse automáticamente para futuras sesiones, es decir, no necesitan autorización para realizar la conexión.

B. Fase de descubrimiento de dispositivos

En esta fase nos disponemos a realizar un escaneo de los dispositivos móviles que se encuentran dentro de nuestro alcance. Para esto utilizamos la herramienta BluezScanner [10] la cual utiliza la pila de protocolos de Bluetooth para GNU/Linux Bluez [11].

```
[carlos@localhost BlueZScanner]$ ./bluezscanner -c
+ BlueZScanner, por Gospel <gospel.endorasoft.es>

Detectando dispositivos ...

Dispositivo (1) encontrado:
MAC: 00:18:0F:16:2F:70      Nombre: Nokia
Fabricante del Chip Bluetooth:
- Desconocido
Class: 0x520204 [01010010000001000000100]
- Servicios soportados (Service Classes):
- Telephony (Cordless telephony, Modem, Headset service, ...)
- Object Transfer (v-Inbox, v-Folder, ...)
- Networking (LAN, Ad hoc, ...)
- Tipo de dispositivo (Device Class):
- Phone > Cellular

Dispositivo (2) encontrado:
MAC: 00:16:20:FB:F6:EA      Nombre: Alejo Cel
Fabricante del Chip Bluetooth:
- Sony Ericsson Mobile Communications AB
Class: 0x520204 [01010010000001000000100]
- Servicios soportados (Service Classes):
- Telephony (Cordless telephony, Modem, Headset service, ...)
- Object Transfer (v-Inbox, v-Folder, ...)
- Networking (LAN, Ad hoc, ...)
- Tipo de dispositivo (Device Class):
- Phone > Cellular
```

Figura 1. Descubrimiento de dispositivos con BlueZScanner [10]

Esta aplicación nos permite, además de detectar los dispositivos, mostrar características de estos tales como la MAC, el nombre de fabricante, el tipo de dispositivo y los servicios Bluetooth disponibles. Con esto logramos detectar los dos dispositivos que realizaron el emparejamiento dándonos la posibilidad de saber las BD_ADDR's para utilizarlas en la siguiente fase del ataque.

Luego que ya sabemos los dispositivos que están al alcance, tratamos de enviar un archivo utilizando el protocolo OBEX para la transferencia de archivos por Bluetooth, lo cual no nos permite ya que no somos un dispositivo de confianza:



Figura 2. Intento de transferencia de archivo desde el Laptop

Para realizar la transferencia del archivo desde el laptop sin necesidad de autenticación necesitamos pasar a la siguiente fase del ataque.

C. Fase de suplantación de identidad de un dispositivo de confianza.

Conociendo la dirección BD_ADDR a suplantar, procedemos a cambiar la MAC de nuestro

adaptador Bluetooth USB con la del dispositivo de confianza. Para esto utilizamos la herramienta bdaddr [22] a la cual se le envían como parámetros la nueva BD_ADDR y el puerto de Bluetooth a utilizar que usualmente es el hci0.

```
[root@localhost bdaddr]# ./bdaddr -i hci0 00:16:20:FB:F6:EA
Manufacturer: Cambridge Silicon Radio (10)
Device address: 00:18:00:16:2F:70
New BD address: 00:16:20:FB:F6:EA

Address changed - Reset device now
[root@localhost bdaddr]#
```

Figura 3. Cambio de la MAC del adaptador USB por la MAC del dispositivo de confianza acoplado.

Luego de esto desconectamos el adaptador para que los cambios sean aceptados y finalmente, cuando lo volvemos a conectar, colocamos el comando hciconfig verificando que en realidad se cambio la MAC de nuestro adaptador, realizando exitosamente la suplantación de identidad.

```
[root@localhost bdaddr]# hciconfig
hci0: Type: USB
BD Address: 00:16:20:FB:F6:EA ACL MTU: 384:8 SCO MTU: 64:8
UP RUNNING PSCAN INQUIRY
RX bytes:495 acl:0 sco:0 events:25 errors:0
TX bytes:331 acl:0 sco:0 commands:20 errors:0

[root@localhost bdaddr]#
```

Figura 4. Confirmación de que la MAC ha sido cambiada.

Luego de tener la MAC cambiada, ya hemos suplantado la identidad del dispositivo enlazado y podemos proceder a enviar archivos, acceder a la información del otro teléfono entre muchas otras opciones más.

D. Fase transferencia de archivo

A partir de este momento el atacante tiene la posibilidad de enviar archivos al equipo de la víctima sin pedir previa autorización. Esto se puede lograr desde la línea de comandos con el comando "obex_push" al cual se le envían como parámetros el canal, la MAC destino y el archivo o también se puede instalar el paquete GNOME o KDE de Linux que soporte Bluetooth para enviarlo por medio de interfaz gráfica.

Nombre	Tamaño	Tipo de archivo	Modificado	Permisos	Propietario
+ Archivos de música	0 B	Carpeta		dr-x-----	carlos
+ Grabaciones	0 B	Carpeta		dr-x-----	carlos
+ Gráficos	0 B	Carpeta		dr-x-----	carlos
+ Imágenes	0 B	Carpeta		dr-x-----	carlos
+ NO NAME	0 B	Carpeta		dr-x-----	carlos
+ Temas	0 B	Carpeta		dr-x-----	carlos
+ Tonos	0 B	Carpeta		dr-x-----	carlos
+ Videoclips	0 B	Carpeta		dr-x-----	carlos
owned.jpg	46,7 KB	Imagen JPEG	19-11-07 16:28	-rw-----	carlos

Figura 5. Transferencia de archivos al celular sin necesidad de confirmación.

VIII. HERRAMIENTAS

A. BlueZ

Esta aplicación es considerada la más completa y potente implementación libre de Bluetooth y desarrollada para GNU/Linux. Su funcionamiento esta basado en el reconocimiento de los diferentes protocolos manejados por Bluetooth:

1. HCID (Host Controller Interface Daemon):

El Demonio de la Interface Controladora de la Máquina “Esta interfaz proporciona una capa de acceso homogénea para todos los dispositivos bluetooth de banda base” [1], permite un acceso al estado del hardware y a los registros del control de los dispositivos. La capa HCI es la encargada de intercambiar datos con el firmware con la capa HCI del dispositivo Bluetooth que se este utilizando. Este protocolo es implementado por el driver de la capa de transporte, es decir, por el driver controlador del bus físico.

2. SDPD (Service Discovery Protocol Daemon):

El Protocolo de Descubrimiento de Servicios es el encargado de buscar todos los servicios disponibles en los diferentes servidores de aplicación cercanos, estos servicios son publicados por estos servidores junto con los atributos y la descripción de los servicios que ofrecen; la descripción de estos servicios incluye la forma de uso y los atributos representan los parámetros de entrada que necesitan.

3. Rfcomm:

Este protocolo realiza una simulación de los puertos serie a través del protocolo L2CAP. Soporta hasta 9 puertos serie RS-232 y permite hasta 60 conexiones simultaneas ente dos dispositivos Bluetooth.

“Para los propósitos de RFCOMM, un camino de comunicación involucra siempre a dos aplicaciones

que se ejecutan en dos dispositivos distintos (los extremos de la comunicación). Entre ellos existe un segmento que los comunica. RFCOMM pretende cubrir aquellas aplicaciones que utilizan los puertos serie de las máquinas donde se ejecutan. El segmento de comunicación es un enlace Bluetooth desde un dispositivo al otro (conexión directa).”[1]

4. HCIdump

Lee la información pura de las tramas enviadas y recibidas por el protocolo HCI hacia y desde un dispositivo Bluetooth. Es una librería propia de BlueZ y permite especificar diferentes filtros para la selección de diferentes tipos de paquetes como L3CAP, rfcomm, sdp, obex, etc.

5. L2ping

Envía una “echo request” y recibe un paquete del mismo tipo, funciona igual que un ping en los ordenadores normales.

6. Bluepin

Es usado por hcid para preguntar al usuario por un código PIN cuando se esta intentando realizar el emparejamiento de los dispositivos bluetooth.

7. Bluemon

“Bloqueo de las X en función de la distancia de un dispositivo Bluetooth (hace L2ping hasta que llega a un limite predefinido). Puede ser usado para bloquear servicios en función de la distancia de una persona, PDAs para empresarios, cuerpos de seguridad, etc.” [13].

8. P3nfsd

Permite realizar un acoplamiento de el sistema de archivos del dispositivo Bluetooth que se este utilizando, al de los sistemas operativos GNU/Linux, haciendo posible copiar o editar cualquier tipo de archivo que se encuentre en el dispositivo con la herramienta preferida de la maquina UNIX.

9. BtBrowser

Permite el descubrimiento de los dispositivos Bluetooth junto con los servicios que ofrecen y los canales para cada servicio y saber si el dispositivo tiene encriptación.

10. BlueMoto

Este programa cambia el profile del dispositivo para poder pasar por un manos libres y recibir comandos desde un móvil, enviando shellscripts para cada acción.

11. OBEX (Object Exchange Protocol)

OBEX es un protocolo de intercambio de datos que funciona sobre cualquier capa de transporte (Bluetooth, USB, TCP, etc). Se implementa un Obexftp que pretende acceder a los sistemas de archivos de los dispositivos compatibles con Obex. [14]

La gran ventaja de BlueZ es que es una implementación libre desarrollada para Linux que permite manipular fácilmente los diferentes comandos de Bluetooth y acomodarlos a las necesidades del usuario. Es por esta razón que los ataques con esta nueva tecnología se desarrollan todos desde este sistema operativo, dada su versatilidad, aunque requiere de un cierto conocimiento.

Un ejemplo de las posibilidades que brinda esta herramienta es sustituir al dispositivo que recibe la conexión (spoofing), el stack de BlueZ registra en rfcomm como "closed" la conexión que acabamos de romper. Así podemos saber en que canal estaba conectado.

B. TULP2G

Una vez comprendido el estándar de comunicación de Bluetooth, se hace el uso de TULP2G una herramienta para extraer información de tipo forense de un teléfono celular.

A pesar de ser una herramienta muy completa para la extracción de evidencia forense, solo personal experto sabe que buscar específicamente o como hacer uso de la información encontrada de acuerdo al ataque que se realice. El almacenamiento de datos es realizado en XML, pero para la extracción y presentación de estos tiene diferentes plugins que

permiten especificar los tipos de datos buscados y diferentes tipos de presentación en HTML. [4]

IX. RECOLECCIÓN DE EVIDENCIA

Para recolectar evidencia del ataque realizado, es necesario saber cómo funciona y cuál es el "ruido" que hace el atacante cuando lo ejecuta; como ya se ha visto la ejecución de este en una sección previa de este artículo, retomaremos los pasos presentados allí para recopilar la evidencia necesaria.

Como el objetivo del ataque es suplantar la dirección BDADDR de un dispositivo de confianza que tenga un celular o una PDA, como lo puede ser un dispositivo de manos libres u otro celular, los primeros indicios que tendríamos de que se está realizando el ataque es examinar una transferencia de archivos desde un dispositivo que no conocemos sin pedir ningún tipo de autorización o si el dispositivo está apagado. La aparición de archivos extraños en el dispositivo podría ser un indicio.

Existen tres tipos de datos que se pueden obtener de un teléfono móvil: [16]

1. Información local
2. Información de cuenta incluyendo call logs
3. Información almacenada localmente

Las primeras dos se pueden obtener del proveedor de telefonía celular local (Comcel, Movistar) pero la tercera se encuentra almacenada en el dispositivo.

"The way the data are stored on the handset will depend to a large extent on the make and model of the phone. Simplistically speaking, the newer the model the grater the sophistication and the larger the amount of data stored." [16]. Por ejemplo, los Nokias antiguos sólo podían guardar la lista de contactos en la tarjeta SIM pero los nuevos modelos proporcionan la opción de elegir si guardarlos en el teléfono o en la SIM.

Para todas las marcas y los modelos la información que debe ser recuperada en una investigación es: [16]

1. Fecha y hora de llamadas y mensajes de

texto.

2. *Llamadas realizadas, recibidas y perdidas.*
3. *Mensajes de texto recibidos y enviados.*
4. *Mensajes borrados.*
5. *Número del teléfono.*
6. *La lista de contactos.*

Además de buscar en el dispositivo en si, existe información valiosa que se encuentra almacenada en la SIM de la cual hablamos anteriormente. La siguiente es una lista de información útil en un análisis forense a un dispositivo móvil:

1. *Identificador de Área Local: Identificador de donde esta el teléfono actualmente situado.*
2. *Número Serial: Identifica la SIM*
3. *Número del Cliente: Identifica al cliente ante el proveedor de servicio de telefonía celular.*
4. *Número del Celular*
5. *Mensajes de Texto: Usualmente la SIM alcanza a guardar al menos 12 mensajes de texto.*
6. *Mensajes Borrados*
7. *Números marcados*
8. *Último número marcado*

Como se observa, toda esta información que pueden recuperar ciertas herramientas especiales sirve como evidencia para encontrar pruebas sobre un posible acto delictivo y no para saber sobre posibles ataques en nuestro dispositivo. Luego de realizar el ataque al celular, lo que se buscaba eran rastros en este dispositivo para poder identificar alguna pista que nos llevara al posible atacante y también para poder saber que acciones malintencionadas e in-autorizadas se le realizaron a la victima.

Pero luego de buscar exhaustivamente en Internet y mirando una variedad de artículos sobre el tema [19], se concluyó que la variedad de herramientas que existen actualmente en el mercado sirven para recolectar información del tipo que mencionamos anteriormente y no algún tipo de log de registro de acciones en el teléfono celular ya que se piensa que

la memoria y el tamaño del dispositivos son muy limitados y el sistema operativo no se dedica a guardar este tipo de datos.

Otra alternativa que se analizó fue construir un sniffer de bajo costo para Bluetooth flasheando el adaptador USB convencional con el firmware de un sniffer comercial. El nombre de esta herramienta es BTSniff [20] y esta disponible solo para Linux. “El firmware del sniffer comercial se ha obtenido mediante ingeniería inversa a partir de una versión trial que ofrecía un fabricante de estos dispositivos, lo cual es ilegal en algunos países.” [6]

X. CONCLUSIONES

El ataque Blue MAC Spoofing es una vulnerabilidad que explota el estándar mismo y por eso es tan preocupante que se pueda realizar ya que corregirlo llevaría a modificar el estándar completo cambiando el firmware de los dispositivos que usen esta tecnología para que no dejaran cambiar la BD_ADDR.

Este fallo de seguridad prácticamente es un Backdoor por el cual se pueden enviar diferentes archivos, desde una inocente imagen hasta un troyano sin autorización. Un ejemplo de esto es una aplicación desarrollada para tomar el control del celular de una persona [21]. Este software permite el acceso transparente de un celular a otro, tomando el control por completo. El único inconveniente es que para conectarse, la victima necesita dar la autorización a menos que el celular atacante este en la lista de dispositivos confiables de la victima.

Es en este momento en donde entra en acción el ataque que estamos analizando en este documento, el cual logra conectarse y transferir archivos sin autorización previa del usuario, suplantando la identidad de un dispositivo de confianza. Es por esto que este ataque mas la aplicación de control del dispositivo formarían una nueva manera de intrusión en donde no sería necesario conocer la victima ni pedirle autorización para poder controlar el teléfono y descargar información de éste.

Finalmente en la parte de análisis forense concluimos que es muy complicado reunir los rastros suficientes para poder encontrar el posible atacante y las acciones que realizó en el teléfono. La única aproximación posible es descargar los datos explicados anteriormente de la SIM y del teléfono para ver que acciones no realizó el usuario. Sin embargo es difícil detectar quien fue el posible atacando y mucho más reunir las posibles evidencias que lo incriminen.

Por otro lado la mejor forma de prevención es mantener el Bluetooth encendido solamente cuando se vayan a transferir archivos y no generar acoplamientos ya que si no existen dispositivos confiables, no hay objetivo para suplantar la identidad del dispositivo que pueda tener acceso libre al otro.

REFERENCIAS

- [1] Wikipedia: La Enciclopedia Libre, Bluetooth. (Agosto, 2007) Disponible: <http://es.wikipedia.org/wiki/Bluetooth>
- [2] A. Moreno, Blue MAC Spoofing. (2007) . Disponible: <http://gospel.endorasoft.es/bluetooth/seguridad-bluetooth/blue-mac-spoofing.html>
- [3] InFotecnología, Funcionamiento de la Tecnología Bluetooth. Disponible: <http://www.info-tecnologia.com.ar/redes/Funcionamiento-de-la-tecnologia-Bluetooth.php>
- [4] A. Moreno, Arquitectura de Protocolos Bluetooth, (2007) Disponible: <http://gospel.endorasoft.es/bluetooth/especificacion-bluetooth/arquitectura-de-protocolo/index.html>
- [5] InfoTecnología, Ventajas de la Tecnología Bluetooth. Disponible: <http://www.info-tecnologia.com.ar/redes/Ventajas-de-la-tecnologia-Bluetooth.php>
- [6] A. Moreno, Avances en Sniffing Bluetooth , (2007). Disponible: <http://elblogdegospel.blogspot.com/2007/08/avances-en-sniffing-bluetooth.html>
- [7] SIG Bluetooth, Bluetooth Security, Disponible: <http://www.bluetooth.com/Bluetooth/Learn/Security/>
- [8] A. Moreno, elementos de Seguridad en Bluetooth, (2007) <http://gospel.endorasoft.es/bluetooth/seguridad-bluetooth/elementos-de-seguridad.html>
- [9] Wikipedia: La Enciclopedia Libre, Tarjeta SIM , (2007) Disponible: http://es.wikipedia.org/wiki/Tarjeta_SIM
- [10]A. Moreno, BlueZScanner: el escaner Bluetooth basado en BlueZ (2007) Disponible: <http://elblogdegospel.blogspot.com/2006/04/bluezscanner-el-escaner-bluetooth.html>
- [11]BlueZ Project, Official Linux Bluetooth protocol stack, (2007) Disponible: <http://www.bluez.org/>
- [12]P. Lucistnick. Manual de FreeBSD. Capitulo 27: Networking Avanzado. Disponible: <http://www.freebsd.org/doc/es.ISO8859-1/books/handbook/network-bluetooth.html>
- [13]Pancake. Bluetooth Hacking. (2006) . Disponible http://www.kaslab.net/downloads/Telematicas_2006/Telematicas_2006-Bluetooth_Hacking-pancake.pdf
- [14]G. Ortega . Obex (2007, Agosto) Disponible: <http://coffeliusiwiki.bloggear.net/obex>
- [15]J J. van den Bos & R. van der Knijff. TULP2G – An Open Source Forensic Software Framework for Acquiring and Decoding Data Stored in Electronic Devices. Netherlands Forensic Institute. (2005). Volumen 4. Disponible: <http://www.utica.edu/academic/institutes/ecii/publications/articles/B4A85456-BE75-87F8-E45EB9C6082FDF4E.pdf>.
- [16]B. Mellars, Forensic Examination of Mobile Phones. (2004) Disponible: <http://faculty.colostate-pueblo.edu/dawn.spencer/Cis462/Homework/Ch4/Forensic%20examination%20of%20mobile%20phones.pdf>
- [17]M. Musters, Cell Phone Forensics. (2007) Disponible: <http://www.computerforensics.ca/images/Cell%20Phone%20Forensics.pdf>
- [18]E-evidence info, The electronic evidence Information. Disponible: <http://www.e-evidence.info/cellarticles.html>.
- [19]Darkircop. Security. <http://darkircop.org/security/>
- [20]La comunidad Dragon. Super Bluetooth Hack, Controla Teléfonos con Bluetooth Activado. Disponible: <http://www.dragonjar.us/super-bluetooth-hack-controla-telefonos-con-bluetooth-activado.shtml>
- [21]M Holtmann. "Bluetooth adapters and Bluetooth enabled products". 2005. Disponible en: <http://www.holtmann.org/linux/bluetooth/features.html>
- [22]M Holtman "bdaddr.c". Disponible en: <http://csourcesearch.net/data/package/bluez-utils/bluez-utils-2.17/test/bdaddr.c>