ISEC Lab #7

Seguridad en Terminales Smartphone

Miguel Ángel Domínguez Torres
mdominguez<arroba>isecauditors.com

1 Introducción

En esta ocasión, nuestro ISecLab tratará sobre una de las tecnologías que cada día está tomando más importancia tanto en el mundo empresarial como entre los ciudadanos de a pié. Hablo de la evolución que han sufrido los terminales de telefonía móvil, pasando de ser meros utensilios para conversar e intercambiar mensajes cortos (SMS), a convertirse en micro ordenadores que incorporan capacidades avanzadas de cómputo y que permiten a los usuarios realizar llamadas por videoconferencia, navegar a través de Internet, ejecutar aplicaciones de correo electrónico y procesadores de texto, jugar en red con otros usuarios y muchas otras funciones que están apareciendo día tras día. A este tipo de terminal se le ha denominado Smartphone o Teléfono Inteligente.

Hablaremos de las tecnologías que podemos encontrar en el interior de estos terminales. Tecnologías como los sistemas operativos Symbian, Windows Mobile y Blackberry OS, así como los problemas de seguridad que conlleva el hecho de abrir nuestros terminales móviles a Internet. Remarcaremos los problemas de seguridad que están acarreando las deficiencias en el desarrollo de estos sistemas operativos y de una tecnología de comunicación que se están volviendo más y más popular: BlueTooth

Por último señalaremos algunas recomendaciones que las empresas y los usuarios deben tener en cuenta para mitigar los peligros que conlleva la utilización de terminales smartphone.

2 Sistemas Operativos

Un sistema operativo, ya sea de ordenador personal (PC), supercomputador u otro tipo de hardware, tiene como uno de sus objetivos principales hacer de intermediario entre del hardware y las aplicaciones de usuario (navegador, cliente de correo, aplicación de contabilidad, etc.) que necesitan utilizar dicho hardware. En el caso de un smartphone, este objetivo sigue prevaleciendo y por tanto, los sistemas operativos que incorporan los teléfonos inteligentes (smartphone), deben proporcionar las interfaces necesarias para que los usuarios y aplicaciones puedan aprovechar todas las capacidades que el teléfono brinda. Cuantas más prestaciones proporcionan los terminal, la red de comunicación y los propios operadores de telefonía, más complejo es el sistema operativo que debe proporcionar las interfaces para aprovechar dichas prestaciones.

2.1 Symbian OS

Symbian OS es uno de los sistemas operativos que incorporan teléfonos smartphone. Los orígenes de este sistema operativo nacen de otro sistema operativo llamado EPOC que incorporaban las PDAs de la empresa Psion. A raíz de una iniciativa por parte de empresas de gran renombre dentro de la telefonía móvil (Nokia, Ericsson, Motorola, Samsung, Panasonic y Siemens), se crea una spin-off para el desarrollo de un nuevo sistema operativo orientado a una nueva gama de teléfonos con mayores capacidades de cómputo, los smartphone. Debido al cambio de orientación entre el sistema EPOC original y el nuevo sistema operativo, se decidió cambiar el nombre a éste y llamarlo Symbian OS.

Dentro de la gama de smartphones basados en Symbian encontramos los teléfonos Nokia con Interfaz gráfica series60 y los smartphones Sony-Ericsson basados en interfaz UIQ.

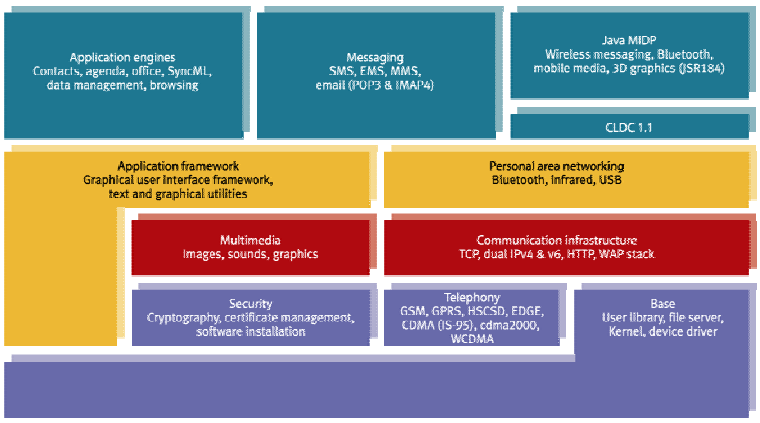

Ilustración 1 - Arquitectura Symbian OS

Dentro de la arquitectura del sistema operativo Symbian (Ilustración 1) nos encontramos con diferentes capas que van agrupando las funciones más básicas y proporcionando interfaces de carácter más general que facilitan el desarrollo de aplicaciones de usuario que encontramos en las capas más altas. Vemos como las capas inferiores están formadas por el propio kernel (núcleo) del sistema operativo, junto con un conjunto de librerías que dan soporta a la comunicación a través de enlaces GSM o GPRS. En las capas inferiores también encontramos las librerías básicas para implementar la seguridad en las demás capas de la arquitectura. Por encima encontramos diferentes pilas de comunicación (TCP/IP, WAP), Bluetooth, infrarojos, y los frameworks que permiten el desarrollo de aplicaciones de usuario.

Las aplicaciones de usuario están compuestas por contactos, agenda, navegación web, herramientas de oficina, mensajería (SMS, MMS, EMS, e-mail) y aplicaciones de terceras empresas desarrolladas en diferentes lenguajes de programación (C++ y Java son los principales lenguajes que se utilizan)

Todas estas características suponen puntos de entrada a nuestro terminal smartphone, incluyendo software de virus, troyanos, gusanos y ataques por parte de hackers.

Un concepto importante de seguridad que implementan los sistemas Symbian es la validación de código mediante la utilización de firmas digitales. Este concepto llamado **Symbian Signed** o **Java Verified** dependiendo del lenguaje de programación y del proceso de verificación de código que se realice, permite a los usuarios poder validar que las aplicaciones que descargan de Internet y que instalan en sus smartphones, provienen de fuentes fiables (no contienen virus, troyanos u otro tipo de malware).

Las versiones de Symbian OS que actualmente podemos encontrar en la mayoría de teléfonos smartphone varían entre v7.0, v7.0s y v8.0. La última versión de este sistema operativo es la v9, que incorporarán los smartphone de última generación y que dentro de los aspectos de seguridad diferenciadores respecto a las otras versiones, encontramos la nueva plataforma de seguridad PlatSec (Platform Security) basada en potenciar la detección de intentos de acceso no autorizados al hardware, software y datos del teléfono, así como prevenir que los programas actúen de forma no aceptable, ya sea por error o intencionadamente (malware).

Esto se consigue utilizando capabilities que protegen las APIs (acceso a datos confidenciales de usuario, acceso a la red, ...) para que sólo las aplicaciones de confianza puedan utilizar ciertas funcionalidades del sistema operativo.

Para utilizar una API protegida el programa deberá solicitar permiso (Autorización), y ésta autorización se consigue a través de Symbian Signed.

La firma de código según Symbian Signed se estructura en los siguientes pasos:

- El desarrollador/vendedor del software compra un ACS Publisher ID (certificado digital para firma de código) a Verisign, previa autenticación. Con este ID el vendedor firma el código.
- Las aplicaciones firmadas son enviadas a la empresa Symbian, quien pasa el código a un evaluador contratado por el propietario del código de entre la lista de evaluadores que Symbian acepta. El evaluador se encarga de testear el código siguiendo los criterios marcados por Symbian.
- La aplicación se evalúa contra el teléfono/s que especifique el vendedor.
- Si se pasa el test, El ACS Publisher ID se elimina y el código es firmado por Verisign con un certificado que está enlazado (certificate chain) con el certificado Symbian Root.

La otra opción que tienen los desarrolladores de software para integrar sus aplicaciones en Smartphones Symbian es utilizar la plataforma Java. Concretamente J2ME (Java 2 Micro Edition) en su configuración para

dispositivos móviles CDLC y (Connected Limited Device Configuration) con el conjunto de APIs (Interface, Red, etc..) que implementa el perfil MIDP (Mobile Information Device Profile). J2ME proporciona soporte a TLS/SSL y da acceso a librerías criptográficas (IAIK, BouncyCastle, etc..).

El modelo de seguridad de J2ME se divide en 2 niveles. Al nivel más bajo tenemos el bytecode (class file) verifier, encargado de controlar que las aplicaciones no puedan dañar el dispositivo.

A nivel de aplicación CLDL/MIDPv1, se basa en la idea de sandbox que Sun incorpora desde un inicio en su JVM. La diferencia es que en un dispositivo smartphone la memoria es un recurso escaso y por este motivo la implementación de la sandbox es limitada. Otro aspecto de J2ME es que No incorpora Security Manager. En su lugar se consigue eliminando funcionalidades del lenguaje Java (JNI, Reflection, etc.). En MIDP v2 se mejora el modelo de seguridad consiguiendo una mayor granularidad a través de **Dominios de Protección** que determinan el conjunto de permisos que se otorgan a un MIDlet (programas desarrollados con MIDP). Los MIDlets en MIDP v1 estaban siempre confinados en la Sandbox (Untrusted Domain) con acceso muy limitado a APIs y funciones sensibles del dispositivo. En cambio en MIDP v2 nos encontramos dos dominios de seguridad: **Untrusted** (sandbox – MIDP1.0) y **Trusted** (establecido mediante firmas digitales). Un dominio trusted se vincula a un dominio de protección en el momento de su instalación. Además cada dominio de protección tiene vinculada una política que controla como se asignan los permisos (**Interaction Modes**). Una aplicación solicitará los permisos que necesite mediante un fichero JAD (Java Application Descriptor).

Como hemos comentado, el dominio trusted se consigue mediante la utilización de firmas digitales. El proceso para firmar estos MIDlets se llama Java Verified y lo gestiona el UTI (Unified Testing Initiative) formado por Sun, Motorola, Nokia, Siemens y Sony Ericsson.

Según el modelo de seguridad de MIDP v2, las aplicaciones no firmadas o con una firma no verificable se ejecutan como untrusted (todas las MIDP v1). En cambio las aplicaciones que prueban al terminal que son de confianza mediante un certificado y una firma digital se ejecutan como trusted, pudiendo acceder a funciones y APIs privilegiadas. El proceso de firma de código es similar a Symbian Signed: El código es revisado por un evaluador y firmado en este caso con un certificado que cuelga del certificado raíz del UTI que está incluido en el smartphone.

En febrero de 2005 apareció una noticia que explicaba el fallo de la empresa Nokia al no incluir el certificado raíz para verificar los códigos firmados con Java Verified, en algunos de sus teléfonos smartphone (Nokia 6600, 6230 y 3220). Esto significa que las aplicaciones firmadas con Java Verified no pueden ser verificadas en estos teléfonos y por tanto no pueden instalarse.

2.2 Windows Mobile Smartphone

Microsoft, como no podía ser de otra forma, también proporciona su variante de sistema operativo Windows para smartphone basado en el sistema Windows CE. Actualmente la versión de Windows Mobile 2003 Second Edition (Windows CE 4.2) es la más común que encontramos en los smartphone basados en este sistema operativo. Aunque los nuevos teléfonos podrán incorporar la nueva versión Windows Mobile 2005 (Windows CE 5.0).

Dentro de los fabricantes más conocidos que proporcionan smartphones con Windows Mobile, encontramos teléfonos de Motorola y los TSM520 distribuidos por Movistar.

La arquitectura de Windows Mobile es similar a cualquier otro sistema operativo Windows, teniendo en mente la restricciones de memoria y capacidad de cómputo de los terminales sobre los que se ejecuta. Por tanto, podemos encontrarnos conceptos como el registro de Windows, librerías MFC y comunicaciones NetBios. O aplicaciones de usuario, en su versión reducida, para Internet Explorer (Pocket Explorer), Pocket Outlook, Word Mobile y Excel Mobile.

El modelo de seguridad de Windows Mobile se basa en tres preguntas:

- ¿Qué aplicaciones pueden ejecutarse y qué puede hacer la aplicación?
- ¿Quién puede acceder a determinados aspectos de configuración?
- ¿Qué pueden hacer las aplicaciones de escritorio (ActiveSync y RAPI – Remote API)?

La herramientas para poder responder estas preguntas nos las proporcionan librerías de control de acceso basadas en roles y políticas de seguridad. Librerías criptográficas (CryptoAPI), protocolos de autenticación y VPN (TLS/SSL, WTLS, PAP, CHAP, PPTP, IPSec) y una infraestructura de clave pública (PKI).

La PKI está formada por una serie de repositorios de certificados digitales. Algunos de estos repositorios de certificados se utilizan en el establecimiento de comunicaciones SSL, otros contienen certificados de CA y certificados de usuario, y existe un tipo de repositorio que se utiliza para la validación de código firmado digitalmente. Al igual que Symbian, Microsoft también incorpora su idea de firma digital de código que permite a terceras partes desarrollar aplicaciones para los smartphones que incorporan Windows Mobile.

El proceso de firmar código se rige por el programa **M2M (Mobile2Market)**, que permite certificar aplicaciones como diseñadas para Windows Mobile y que proporciona una autoridad de certificación (CA – Certification Authority) y servicios de firma digital.

Las políticas de seguridad se utilizan para configurar los parámetros de seguridad que serán aplicados, mientras que los roles, identifican niveles de acceso y junto con los certificados digitales (forman parte de la PKI) permiten aplicar las políticas de seguridad.

Windows Mobile soporta dos modelos de seguridad: **One-Tier y Two-Tier**.

One-Tier está implementado mayoritariamente en Pocket PC y su principal característica es que no hace distinción entre aplicaciones privilegiadas y no privilegiadas. Todas las aplicaciones se ejecutan como Trusted (acceso total).

Si una aplicación está firmada se ejecuta como trusted y si no está firmada se comprueba la política de seguridad. Si una aplicación no firmada es aceptada se ejecutará también como trusted.

El modelo de seguridad two-tier, implementado en la mayoría de Windows Mobile Smartphone, distingue entre aplicaciones privilegiadas y No privilegiadas. La aplicaciones pueden ser firmadas privilegiadas, firmadas No privilegiadas y No firmadas.

De forma similar al modelo de seguridad para aplicaciones en Symbian, las firmas de código permiten determinar que aplicaciones tienen privilegios y los permisos asignados.

Las aplicaciones privilegiadas se ejecutan en modo **Trusted** (acceso total), las No privilegiadas se ejecutan en modo **Normal o Untrusted** (acceso limitado), y para las aplicaciones No firmadas se comprueba la política de seguridad para determinar que hacer (si se acepta la aplicación se ejecuta en modo No privilegiado).

Como ya se ha comentado, las políticas de seguridad (Security Policies) son parámetros que indican como se comportará el dispositivo en determinadas situaciones. Algunas preguntas que responde una política de seguridad son:

- ¿Se pueden ejecutar aplicaciones no firmadas?
- ¿Se debe preguntar (prompt security policy) al usuario para confirmar si se ejecuta?
- ¿El dispositivo es one-tier o two-tier?
- ¿Qué derechos tienen las peticiones remotas (RAPI) al dispositivo?

Existen muchas combinaciones de respuestas a estas preguntas, no todas son coherentes. Las configuraciones que combinan políticas de seguridad consistentes son:

- **Sin seguridad (Security Off - Unrestricted)**: Todas las aplicaciones de cualquier fuente pueden instalarse y ejecutarse con acceso a todos los recursos del dispositivo. No son necesarias las firmas de código. Por supuesto no es una práctica recomendada.

- **Consultar (Prompt – Standard, Two-tier y One-tier)**: El usuario debe decidir que hacer cuando la fuente no es confiable. Tampoco requiere que las aplicaciones estén firmadas, pero ahora se consulta al usuario. Debido a la existencia de numerosas aplicaciones no firmadas es la opción más utilizada.

- **Firmados M2M**: Las aplicaciones tienen que estar firmadas para ejecutarse. No se permiten aplicaciones anónimas.

- **Bloqueado (Locked – Restricted)**: Sólo pueden acceder fabricantes u operadores. No permite la instalación de aplicaciones desarrolladas por terceros. Poco viable comercialmente en dispositivos de usuario final.

2.3 BlackBerry OS

El tercer Sistema Operativo que vamos a estudiar es Blackberry OS. Este sistema operativo desarrollado por la empresa RIM (Research In Motion), comparte muchas de la características de seguridad incorporadas en Symbian OS y Windows Mobile (criptografía, PKI, comunicaciones TLS/SSL, WTLS, etc.). No obstante, tenemos que ver Blackberry OS desde un punto de vista diferente.

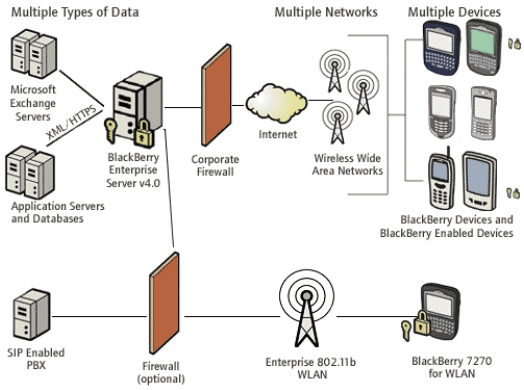

Ilustración 2 - Plataforma BlackBerry

Los elementos que forman una plataforma basada en BlackBerry son los Smartphones (u otros terminales Wireless), el software de escritorio y el servidor BlackBerry Enterprise Server (BES) que es el nodo central de la plataforma.

Este sistema operativo ha sido diseñado principalmente para entornos empresariales, permitiendo extender los servicios de información de la empresa a los empleados móviles:

- **Acceso al correo**: integración con tres de los sistemas de mensajería más utilizados en el sector empresarial: Microsoft Exchange, Lotus Domino y Novell GroupWise. Utilizando el servidor BES como pasarela, los dispositivos BlackBerry pueden consultar y enviar correo corporativo.

- **Información corporativa**: Acceso a la información por HTTP, HTTPS o WTLS utilizando el servicio MDS (Mobile Data Service) de BES

Para proteger la información de mensajería y datos corporativos es necesario que las comunicaciones vaya protegidas ya que el canal de transmisión es inseguro.

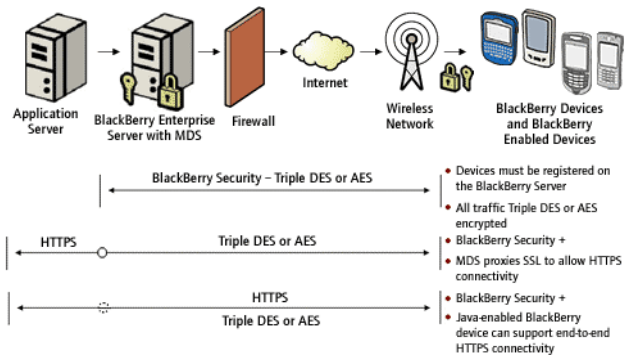

Ilustración 3 - Seguridad en Comunicaciones BlackBerry

Para conseguir esto, el dispositivo BlackBerry y el servidor BES cifran la comunicación utilizando un sistema de clave simétrica 3DES o AES. Estas claves son establecidas durante el proceso de activación del dispositivo (enrollment).

En el caso de comunicaciones con servidores de Aplicaciones o con Internet, es posible que éstas vayan protegidas con HTTPS entre servidor BES y Servidor de aplicaciones (modo proxy), e incluso utilizar HTTPS de extremo a extremo, obteniendo una mayor seguridad, ya que el BES no actúa como punto débil en los datos transmitidos.

Uno de los aspectos importantes al tratarse de un entorno empresarial, es que se pueda realizar una gestión centralizada de los dispositivos distribuidos a los empleados. Esta característica proporcionada por BlackBerry permite a los Administradores poder establecer los parámetros de seguridad de forma centralizada para todos los usuarios (o grupos de usuarios), utilizando el servidor BES. Además un administrador puede decidir eliminar todos los datos de un terminal BlackBerry si determina que éste ha sido robado o perdido. De esta forma la privacidad de la información que podría tener un directivo de una empresa en su dispositivo no se ve comprometida.

RIM hizo especial hincapié en que los sistemas de seguridad incorporados fuesen validados y certificados por normativas de seguridad Americanas (FIPS140-2, DoD), así como analizados por empresas del sector de la seguridad especializadas en hacking ético. Este hecho demuestra la implicación de RIM en el diseño y desarrollo de un sistema operativo con altas garantías de seguridad. De todas maneras, esto no es suficiente para que una plataforma con smartphones basados en Blacberry OS sea segura. Se debe tener especial cuidado en realizar una configuración correcta y segura de la plataforma, además de un mantenimiento periódico.

3 INSeguridad

Hemos descrito algunas de las capacidades de seguridad más importantes que proporcionan los sistemas operativos Symbian, Windows Mobile y Blackberry.

Todas estas características generan nuevas oportunidades y mejoran los procesos de negocio, pero a la vez introducen amenazas que ponen en peligro tanto el terminal Smartphone como los recursos de información de la compañía, ya que aparecen nuevos puntos de entrada para ataques por parte de hackers y todo tipo de código malicioso. Podemos englobar los problemas de seguridad en varios aspectos:

- Errores de desarrollo del software por parte de fabricantes, operadores y terceros que introducen agujeros de seguridad.
- Errores u omisiones en la configuración de las plataformas de seguridad por parte de administradores que abren puertas a atacantes externos o internos.
- Desconocimiento de los peligros de seguridad que pueden conllevar ciertas acciones (p.e. comerciales con Smartphone acceden al correo corporativo y descargan aplicaciones de Internet sin firmar y que contienen virus y troyanos).

Estos riesgos pueden plasmarse en pérdidas económicas para la empresa por causa de fraudes (envío de SMS/MMS, llamadas,...) o robo/alteración de información sensible y/o personal (Email, Agenda, etc.) que repercute en la imagen de la compañía, pérdida de competitividad o incluso conllevan sanciones legales.

El primero de estos aspectos se ha visto reflejado en la implementación de uno de los protocolos de comunicación más utilizados actualmente en todo tipo de dispositivos; me refiero a Bluetooth. La implementación incorrecta de la pila Bluetooth en terminales de algunos frabricantes, entre los que encontramos Nokia y Sony Ericsson, ha provocado que aparezcan un gran número de ataques contra estos dispositivos.

La siguiente tabla muestra algunos de los ataques a Bluetooth.

Ataque	Descripción
Descubrimiento de Dispositivos	Si un dispositivo está en modo visible, es muy fácil identificarlo y atacarlo. Los dispositivos no visibles aún pueden ser atacados si conocemos su MAC (p.e: la obtuvimos en una comunicación anterior cuando el dispositivo estaba visible). Existen herramientas como RedFang que permiten realizar un ataque de fuerza bruta para descubrir dispositivos.
Emparejamiento de Dispositivos (Pairing y Bonding)	Podemos encontrar dispositivos donde la autenticación no está activada y por tanto tenemos libertad para conectarnos a ellos. Por fortuna, para utilizar ciertos servicios (p.e: Transferencia de ficheros), será

	necesario que se establezca primero un emparejamiento entre dispositivos, consistente en una clave secreta compartida. Existen problemas de implementación en los procesos de autenticación y autorización que permiten el robo de información, realizar llamadas o enviar SMS/MMS. Este tipo de vulnerabilidades se deben en gran medida a una falta de tests de seguridad en el proceso de fabricación de los dispositivos (ya sea por falta de interés, reducción de costes o por presión competitiva que obliga a sacar al mercado los dispositivos en un tiempo record). Esto hace esperar que en un futuro aparezcan problemas de seguridad incluso más graves.
BlueBug	Permite la descarga no autorizada de información, envío y lectura de SMS, establecer llamadas, redirección de llamadas, etc. Mediante el establecimiento de una conexión serie (serial profile) y explotando los comandos AT, sin pasar por el protocolo OBEX.
BlueJacking	Permite el envío anónimo de mensajes utilizando Bluetooth. No implica la eliminación o alteración de datos en el dispositivo. Para ello se activa el bluetooth y se crea un nuevo contacto en la agenda de direcciones. En el campo de nombre se escribe el mensaje y eso es todo. Esta técnica que se está popularizando entre los usuarios (gente joven principalmente que incluso llega a montar sistemas de chat) se basa en abusar del protocolo de emparejamiento (pairing protocol). El problema de seguridad surge cuando un atacante (bluejacker) utilizando esta técnica consigue que la victima finalice el proceso de emparejamiento, y por tanto tener acceso a la información del dispositivo. También llamado un **OBEX Push Attack**, ya que OBEX permitiría cargar objetos de forma anónima en un dispositivo.
Escaneo PSM	Port Scanning para bluetooth No todos los puertos PSM (Protocol/Service Mulplexer) se registran con un SDP (Service Discovery Protocol) local. De manera que si hacemos un bypass de la base de datos SDP e intentamos conectar de forma secuencial con diferentes puertos PSM, podríamos localizar funcionalidades ocultas. Podría ser una técnica que se utilizase para ocultar una puerta trasera en dispositivos bluetooth.
BlueSnarfing	**OBEX (Object Exchange):** Es un protocolo de sesión (binary HTTP Protocol), que se utiliza para el

	intercambio de todo tipo de objetos. **BlueSnarfing:** - Se conecta al dispositivo sin alertar al propietario de la petición y consigue acceso a partes de los datos almacenados (agenda, imágenes, calendario, tarjetas, ...) - Acceso al IMEI (International Mobile Equipment Identity), que es un identificador único del teléfono y se utiliza en la clonación de teléfonos ilegales - Normalmente sólo es posible si el teléfono está en modo visible, aunque existen herramientas que permiten evadir esto como por ejemplo **btscanner**.
Malware	Bluetooth puede ser utilizado para instalar virus o troyanos que permitan el control del dispositivo. También es utilizado como medio de distribución en el caso de gusanos como Cabir.
DoS	**BlueSmack**: ataque de DoS afecta a muchas iPAQs y es el equivalente al "Ping of Death" en la pila Bluetooth. **Nokia 6310i**: puede ser apagado enviando mensajes OBEX inválidos.
Ingeniería Social	La falta de concienciación en seguridad y el desconocimiento de la tecnología hacen posible este tipo de ataques.

Tabla 1 - Vulnerabilidades Bluetooth

Muchos dispositivos Nokia y Sony Ericsson son vulnerables a ataques Bluetooth. Un par de ejemplos los encontramos en:

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0143

Multiple vulnerabilities in Nokia 6310(i) Mobile phones allow remote attackers to cause a denial of service (reset) via malformed Bluetooth OBject EXchange (OBEX) messages, probably triggering buffer overflows.

http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2005-0681

Nokia Symbian 60 allows remote attackers to cause a denial of service (phone restart) via a Bluetooth nickname.

Por supuesto la naturaleza insegura del propio protocolo también se ha puesto de manifiesto durante el último mes a través de un estudio realizado por unos investigadores Israelitas que han demostrado la viabilidad de atacar el sistema de cifrado de Bluetooth de forma práctica.

Antes de poder establecer una comunicación, dos dispositivos Bluetooth han de intercambiar una clave secreta generada a partir de que en ambos se teclee idéntico PIN...

El sistema engaña al dispositivo víctima haciéndole creer que ha perdido la clave, forzando así al comienzo de una nueva comunicación siempre que se desee.

Otro elemento importante dentro de sistemas operativos Symbian, en algunos dispositivos Windows Mobile (p.e: Motorola MPx220) y Terminales Blackberry Java Enabled, es el soporte a la ejecución de aplicaciones Java mediante la plataforma J2ME. Ya hemos comentado, al hablar de Symbian, el modelo de seguridad que implementan CDLC y MIDP. J2ME no está exenta de problemas de seguridad: En octubre de 2004, se publicó una alerta comunicando múltiples vulnerabilidades en J2ME. Concretamente relacionadas con errores de implementación en el verificador de clases de la máquina virtual java (KVM) que incorporan múltiples dispositivos smartphone.

Estos problemas permitirían a un MIDlet malicioso evadir las medidas de seguridad del dispositivo y realizar acciones no autorizadas, pudiendo acceder a datos y a servicios del terminal, repercutiendo económicamente o vulnerando la privacidad e integridad de la información. Los fallos en la implementación de J2ME de nuevo ponen de manifiesto las deficiencias en el control de calidad durante el desarrollo, que acaban introduciendo vulnerabilidades de seguridad en plataformas y aplicaciones.

Además de los problemas en el desarrollo de plataformas y aplicaciones, también debemos tener en cuenta que muchos de los usuarios de dispositivos smartphone tienen escasos conocimientos de la tecnología y son víctimas a diario de ataques de Ingeniería Social por parte de hackers, Virus y Troyanos.

En Junio de 2004 aparece el primer código malicioso que ataca teléfonos móviles con Symbian OS. Se llama Cabir y es un gusano que utiliza Bluetooth para propagarse. Llega en forma de fichero con nombre “caribe.sis”, a través de un mensaje MMS. Cuando el usuario lo ejecuta y se instala en el dispositivo el gusano es activado. Actualmente existen múltiples mutaciones de este gusano que han aumentado su virulencia y métodos de propagación. Dado el gran incremento a nivel mundial que está teniendo la telefonía móvil, aumenta en gran medida la posibilidad de crear un gusano que se propague a través de todos los sistemas vulnerables causando un gran daño antes de poder responder al ataque (Warhol Worm).

```
"In the future, everybody will have 15 minutes of fame"
                    -Andy Warhol
```

Un intento para crear este gusano ha sido ComWar (Commwarrior) que se propagaba utilizando MMS y Bluetooth, reenviándose por MMS a todos los contactos de la agenda.

Windows Mobile no queda fuera de peligro. El virus WinCE4.Duts es un poc (proof of concept) desarrollado para demostrar la vulnerabilidad de Windows Mobile. Este virus ejecuta un Keylogger, permite el control remoto del dispositivo y la ocultación de procesos entre otras cosas. También han aparecido vulnerabilidades en Pocket IE que permiten la ofuscación de URLs

, el acceso a ficheros locales y recientemente se ha descubierto un DoS contruyendo una página HTML en un formato específico.

Pocket IE está más limitado, en funcionalidad, que su versión para PC, y por tanto el riesgo se reduce. Sin embargo el crecimiento de funcionalidades es de esperar a medida que los dispositivos incorporan más memoria y capacidad de procesamiento. Y por tanto los riesgos también aumentarán.

El esquema de la plataforma Blackberry que hemos visto nos muestra que el servidor BES proporciona un punto único de fallo en la arquitectura. Es primordial que además de proteger los dispositivos también se proteja el servidor BES utilizando un sistema de cortafuegos y permitiendo únicamente el acceso a los dispositivos autorizados. Actualmente existen algunas vulnerabilidades relacionadas con la plataforma BlackBerry, aunque todas ellas relacionadas con ataques de DoS. Al igual que cualquier otro servidor que nuestra empresa deja accesible desde Internet, BES debe ser securizado, protegido y mantenido debidamente para evitar el acceso no autorizado a sistemas de mensajería y datos corporativos.

En conclusión, podemos ver como a pesar de los esfuerzos, en muchos casos insuficientes, puestos por fabricantes, operadores y terceras empresas, en incorporar medidas de seguridad en los terminales smartphone, al igual que ha sucedido con los PCs, los errores en el desarrollo, las deficiencias de configuración y el desconocimiento general de los usuario, proporcionan un amplio conjunto de vulnerabilidades que explotan hackers y todo tipo de malware para acceder de forma no autorizada a los dispositivos smartphone.

4 Conclusiones

Ninguno de los sistemas operativos que hemos descrito está libre de problemas de seguridad ya sea por una u otra causa. Las empresas y usuarios deben ser conscientes de que para poder aprovechar las ventajas que proporcionan los dispositivos Smartphone, es necesario establecer medidas de seguridad que salvaguarden la información contenida en los dispositivos y que no supongan una amenaza económica o de privacidad para los usuarios o las empresas que proporcionan estos dispositivos a sus empleados.

Las mismas recomendaciones que sirven para los sistemas tradicionales, se pueden aplicar a los sistemas móviles:

- Utilizar VPN para protección del canal de comunicación extremo-a-extremo
- Instalar firewalls personales
- Utilizar software antivirus: Un usuario puede instalar un código malicioso en casa y luego en el trabajo conectarlo a la red de la empresa, propagando la infección entre los servidores de la empresa. De igual forma el usuario puede cargar información de la empresa en el teléfono que luego sea distribuida a Internet desde casa. Algunos programas antivirus son F-Secure, Kaspersky, TrendMicro, Symantec, etc. Aunque la mayoría están
- aún en fases beta.
- Mecanismos de cifrado: Utilizar herramientas de cifrado para proteger las comunicaciones y la información sensible de forma que en caso de pérdida o robo, la información no quede comprometida.

Estos softwares permiten reducir algunos de los riesgos de seguridad que se han comentado (virus, troyanos, privacidad e integridad de la información). Pero, en el caso de las empresas, ¿Qué deben hacer éstas para adaptarse a la tecnología de los terminales Smartphone de forma que no se ponga en peligro la continuidad del negocio?

- Adaptar la política y procedimientos de seguridad introduciendo el nuevo canal de comunicación, determinando las requerimientos de seguridad y las medidas que deben implementarse para salvaguardar los activos de la compañía.
- Implantar estos mecanismos de seguridad para combatir los nuevos peligros y extender las protecciones actuales para abarcar los puntos de entrada que suponen una amenaza a la disponibilidad de los recursos de información y en definitiva a la continuidad del negocio.
- Revisar y Mantener la plataforma de forma periódica y en concordancia con los objetivos del negocio.
- Formar y concienciar a los empleados (seminarios, mesas de trabajo, avisos, etc.) sobre la correcta utilización de la plataforma de terminales Smartphone.

La política de utilización y gestión de los terminales Smartphone, así como el sentido común de los usuarios finales, deberían considerar aspectos relacionados con:

- No instalar software sin verificar antes que está libre de virus (código firmado).
- No abrir mensajes de e-mail o MMS en los que no se confía o que provienen de fuentes desconocidas.
- No habilitar Bluetooth si no es imprescindible y tenerlo en modo visible el tiempo mínimo necesario.
- Sólo aceptar comunicaciones Bluetooth de terminales en los que se confía.
- Aplicar todas las actualizaciones disponibles para solventar deficiencias en el software base incorporado por el fabricante y el operador.
- No permitir la instalación de aplicaciones que no estén homologadas por la empresa (aplicaciones que hayan sido revisadas y aprobadas por el departamento de TI e incluso por dirección).
- No permitir o bloquear la descarga de aplicaciones desde Internet.
- Utilización de software de administración para controlar los dispositivos Smartphone de forma centralizada y en concordancia con la política de seguridad corporativa.

5 Referencias

Referencias
Symbian Signed http://www.symbiansigned.com
Java Verified, UTI (Unified Testing Initiative) http://javaverified.com/index.jsp
M2M (Mobile2Market) http://msdn.microsoft.com/mobility/windowsmobile/partners/mobile2market/default.aspx
J2ME (Java 2 Micro Edition) http://www.microsoft.com/windowsmobile/smartphone/default.mspx
Bluetooth http://www.bluetooth.com http://www.kjhole.com/Standards/BT/BTdownloads.html
Ataques a la pila Bluetooth y Vulnerabilidades en Terminales Nokia y SonyEricsson http://www.thebunker.net/security/bluetooth.htm http://trifinite.org/ http://www.securityfocus.com/bid/13854/info http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0143 http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2005-0681 http://www.securityfocus.com/bid/13782/info
Vulnerabilidades En Dispositivos J2ME http://www.securityfocus.com/bid/11461/info http://packetstormsecurity.com/hitb04/hitb04-adam-gowdiak.pdf
Virus Cabir, WinCE4.Duts http://www.vsantivirus.com/dust-a.htm