## Trabajos Prácticos – Seguridad de la información – dc.uba.ar – 2do cuat. 2013

**Importante: En la última semana de clases deben sí o sí presentar lo que tengan hecho hasta el momento. De no ser aprobado el TP en dicha instancia, deben recuperarlo a más tardar el 20 de diciembre de 2013, que es la fecha definitiva de entrega del TP.**

## Entregables Tp de Implementación

Resumen de 1 o 2 carillas que incluya actividades realizadas, herramientas utilizadas, librerías, material consultado, a entregar en la clase del 23/10 o por mail hasta el 25/10.
Una presentación con transparencias y demo en vivo a dar la última semana de clases.
Un informe al final de cuatrimestre (fecha limite 20/12) de por lo menos 12 carillas, en letra arial 10, espaciado simple, contando lo que hicieron y las decisiones que tomaron (no incluye código fuente).
El código fuente de la aplicación desarrollada.
De ser posible, entregar un DVD con una máquina virtual en vmware o virtualbox con las aplicaciones funcionando.

## TP 1 - Malito – Desarrollo de malware para android.

Se deben desarrollar por lo menos dos aplicaciones maliciosas para android. Se asume que la aplicación la instala el dueño del dispositivo, y acepta los permisos que la aplicación le solicita.

La primer aplicación maliciosa, debe poder, con los mínimos permisos, obtener una copia de todas las imágenes que se encuentran en la memoria SD del dispositivo, y enviarlas a un sitio web controlado por el atacante. Mejoras posibles: Enviar solo algunas imágenes, por ejemplo usando la técnica "Pornographic Images Jacking Algorithm" presentada en la Ekoparty 2013, o hacerlo solo cuando el dispositivo está conectado por wifi.

La segunda aplicación maliciosa debe implementar la mayor cantidad posible de las siguientes características (pueden agregar las que uds consideren):

- Integrarse (bindearse) a una aplicación "benigna" preexistente, como por ejemplo algún juego popular que no sea gratuito, de ser posible con un mecanismo automático. De esta manera, al instalar el supuesto juego, se instala además el código malicioso.
- Comportarse como un remote access tool (rat), es decir, poder ser administrado en forma remota por el atacante, opcionalmente ofuscando la comunicación, y poder realizar algunas de las siguientes acciones (se puede buscar el código fuente androrat para tener algunos ejemplos, pero no vale copiar el código)
    - Obtener contactos (y toda su información)
    - Obtener el registro de llamadas
    - Obtener los mensajes
    - Ubicación del GPS/RED

    - Tomar una foto con la cámara
    - Capturar el audio con el micrófono
    - Enviar mensajes de texto
    - Abrir una URL en el browser
    - Hacer que el teléfono vibre.
    - Descargar un binario dinámicamente y ejecutarlo.
- Desarrollar dos aplicaciones, firmadas por el mismo par de claves, que por si solas no pidan permisos pero que hablando entre si, puedan sumar sus capacidades y realizar tareas maliciosas.
- Actuar como ransomware: cifrar documentos personales del usuario, y pedir un "rescate" para descifrarlos.
- Obtener privilegios de administrador usando algún bug conocido para alguna versión en particular de android, para dificultar la detección y el borrado del malware por parte del usuario.

Libro de referencia:

Xuxian Jiang, Yajin Zhou, Android Malware, SPRINGER BRIEFS IN COMPUTER SCIENCE

## TP2 - GARAN-TITO

El Timestamp es un mecanismo que sirve para demostrar que un dato existe y no fue alterado a partir de un momento dado en el tiempo. Un Timestamp es emitido por una Autoridad de Timestamping que actúa como tercero confiable testificando la existencia de dichos datos electrónicos en una fecha y hora concretos.

Implementar una autoridad de time-stamping (TSA) siguiendo los lineamientos del RFC3161. Se debe incluir un sitio web que permita interactuar con la TSA (también es recomendable implementar algún otro mecanismo de interacción), y un documento que explicite la política de uso.

A partir de dicha autoridad, implementar dos servicios:

1- Servicio de “dar fe de que el contenido de una pagina web existía y estaba publicado en un momento dado”. Para esto, se debe generar un PDF de la página, que incluya un sello de tiempo emitido por la TSA, usando el formato PAdES.

2- Servicio web del Sistema Nacional publico de licitaciones (sinapuli).

Se deben publicar los llamados a licitación mediante el servicio indicado en el punto anterior. Se debe permitir a los oferentes registrarse, y dar un mecanismo por el cual suben primero un hash, y eso les devuelve un sello de tiempo, para demostrar que presentaron en ese momento. Luego de cerrado el periodo de presentación de ofertas, tienen 48 horas para presentar un documento que tenga el hash antedicho. Pensar e implementar mecanismos para que ninguna de las partes pueda hacer trampa (presentar más de una vez, hacerlo fuera de término, decir que no se recibió la propuesta, presentar una propuesta distinta).

Además, se deben analizar (no implementar) los mecanismos existentes para disminuir la posibilidad de fraude por parte de la TSA.