Capítulo 4
Resultados

"Deja las cosas un poco mejor de lo que las encontraste"

H. Jackson Brown.

En este Capítulo se presentan los resultados obtenidos en el presente trabajo, el estado actual de los sistemas desarrollados, los beneficios probados en una red local y en la red UDLA.

Capítulo 4. Resultados

Todas las herramientas y sistemas que conforman a ASIS se probaron en una red privada conformada por los siguientes equipos:

- Ultra 10 (bell.pue.udlap.mx)
- Ultra 5 (bell2.pue.udlap.mx)
- Ultra 5 (laboper.pue.udlap.mx)
- Macintosh 6500

y actualmente algunos de ellos ya están funcionando sobre la red UDLA.

4.1 Crack_UDLA

El sistema Crack_UDLA (Capítulo 3.1.1) se instaló en *laboper.pue.udlap.mx* y para mostrar su salida se probó sobre la tabla de passwords de la UDLA vigente al 15 de marzo de 2001. El proceso para "adivinar" passwords estuvo vivo durante 10 dias y en ese lapso se adivinaron alrededor de cincuenta passwords (de un total de trece mil usuarios, aproximadamente). Para mantener la privacidad de esos usuarios no se publican esos resultados. Por otro lado, con esta nueva versión se establece que algunos passwords actuales ya no son tan seguros como se pensaba, pues fueron adivinados, así surge una nueva política para la generación de passwords para el centro de cómputo: El password generado debe contener al menos un signo de puntuación en medio del mismo, independientemente si lo tiene al principio y/o al final. También debe incluir al menos un dígito independientemente si lo tiene al principio y/o al final.

4.2 Nessus_UDLA

Para el caso de Nessus_UDLA, la herramienta se tiene funcionando en *laboper.pue.udlap.mx* y se ejecutó sobre las sig. máquinas:

- acadaplic.pue.udlap.mx
- adminaplic.pue.udlap.mx
- develop.pue.udlap.mx
- mailweb.pue.udlap.mx
- lara.pue.udlap.mx
- pcmoonray.pue.udlap.mx
- ssray1.pue.udlap.mx

- ssray2.pue.udlap.mx
- cindy.pue.udlap.mx
- bell.pue.udlap.mx
- laboper.pue.udlap.mx

Para cada una de ellas se tiene su reporte de estado, generado con versión para web, las cuales se pueden consultar a detalle en el lugar indicado en el Apéndice A, según la política propuesta en esta tesis.

4.3 Ataca

El sistema **Ataca**, como se mostró en el capítulo 3.1.3 se desarrolló y ejecutó sobre la máquina lara.pue.udlap.mx e informa a través de su interfaz web sobre las vulnerabilidades más comunes sobre el sistema operativo Unix en su versión Solaris 2.6 Este fue la primer propuesta de herramienta para chequeo de vulnerabilidades y sobre la cual se basó la idea de NESSUS_UDLA.

Con estas herramientas se cumple la tarea propuesta de la creación de un *Hacker's work bench* para "auto atacar" y detectar los hoyos de seguridad conocidos hasta hoy antes que cualquier persona ajena al centro de cómputo de la UDLA.

4.4 Kerberos.

El instalador automáticos de kerberos realizado en este trabajo de tesis ya se utiliza para las versiones de Solaris 2.6, 2.7 y 2.8 y permite configurar a una máquina como cliente kerberos ejecutando sólo un procedimiento. Anterior a este trabajo se tomaba en promedio 20 minutos instalar un cliente. Con este procedimiento se tienen dos ventajas:

- Instalación promedio de un minuto
- No es necesario conocer las peculiaridades de la versión del sistema operativo de la máquina

4.5 Secure Shell.

Secure Shell está instalado en los servidores principales de la red UDLA y otras máquinas. En pruebas realizadas, se observa que los servicios típicos que lo utilizan

(telnet y ftp) hacen un 15% aproximadamente más lenta la comunicacíon. Se acepta como un buen parámetro considerando que toda la plática entre máquinas es encriptada.

4.6 Wrappers.

La actualización de la versión de wrappers a la 7.6 permite cerrar hoyos de seguridad que se tenían con las versiones anteriores. La instalación se hace de forma automática a través de un pequeño sistema que reduce su instalación de veinticinco a dos minutos.

4.7 Programas de monitoreo.

La contrucción de programas de monitoreo que se realizó en este trabajo permite obtener en lapsos de tiempo muy cortos información sobre actividad sospechosa en la red UDLA. Anteriormente se recolectaba este tipo de información pero se presentaban los siguientes problemas:

- Los programas de monitoreo tardaban aproximadamente dos horas en su ejecución.
- Ocupaban casi el 100% de recurso de CPU
- Nadie analizaba las salidas.

Ahora los sistemas de detección ocupan sólo unos minutos y no son una carga significativa para la máquina. Además se hace un análisis de la información obtenida y se reportan los casos anómalos.

Con estas herramientas se cumple la propuesta del *Administrator 's work bench* para prevenir, autocorregir e informar sobre las actividades consideradas como anormales en la red UDLA.

La conjunción del *Hacker 's work bench* y del *Administrator 's work bench* presentan el esquema integral de seguridad para la red UDLA propuesto al inicio de este trabajo de tesis. Asímismo, se presenta una documentación detallada sobre todos los sistemas instalados, para que los miembros del centro de cómputo tengan la habilidad de hacer modificaciones al esquema según las nuevas necesidades de la red.