Sistemas de detección de intrusos: un enfoque práctico

Antonio Villalón Huerta

avillalon@s2grupo.com

Abril, 2003

This is 100% Microsoft free

Contenidos

- Introducción.
- Detección en el cortafuegos.
- Detección en la red.
- Detección en la máquina.
- Conclusiones.
- Próximamente. . .

Introducción

Las definiciones habituales...

- **Intrusión:** Conjunto de acciones que intentan comprometer la integridad, confidencialidad o disponibilidad de un recurso.
  ⇒ No sólo penetraciones contra un sistema.
- **Detección de intrusos:** Análisis automático de parámetros que modelan la actividad de un entorno con el propósito de detectar e identificar intrusiones.
- **Sistema de detección de intrusos (IDS):** Mecanismo de seguridad que lleva a cabo la detección de intrusos.
  ⇒ No tiene por qué ser un programa o producto concreto.

Introducción

Un poco de historia...

- Primer trabajo sobre IDSes: 1980 (James P. Anderson).
- Década de los 80: diseño del primer sistema (IDES), que funcionaba en tiempo real (Dorothy Denning, Peter Neumann).
- Auge desde 1995 (crisis de los *firewalls*).
- Actualmente, uno de los campos con más investigación y avances.

Introducción

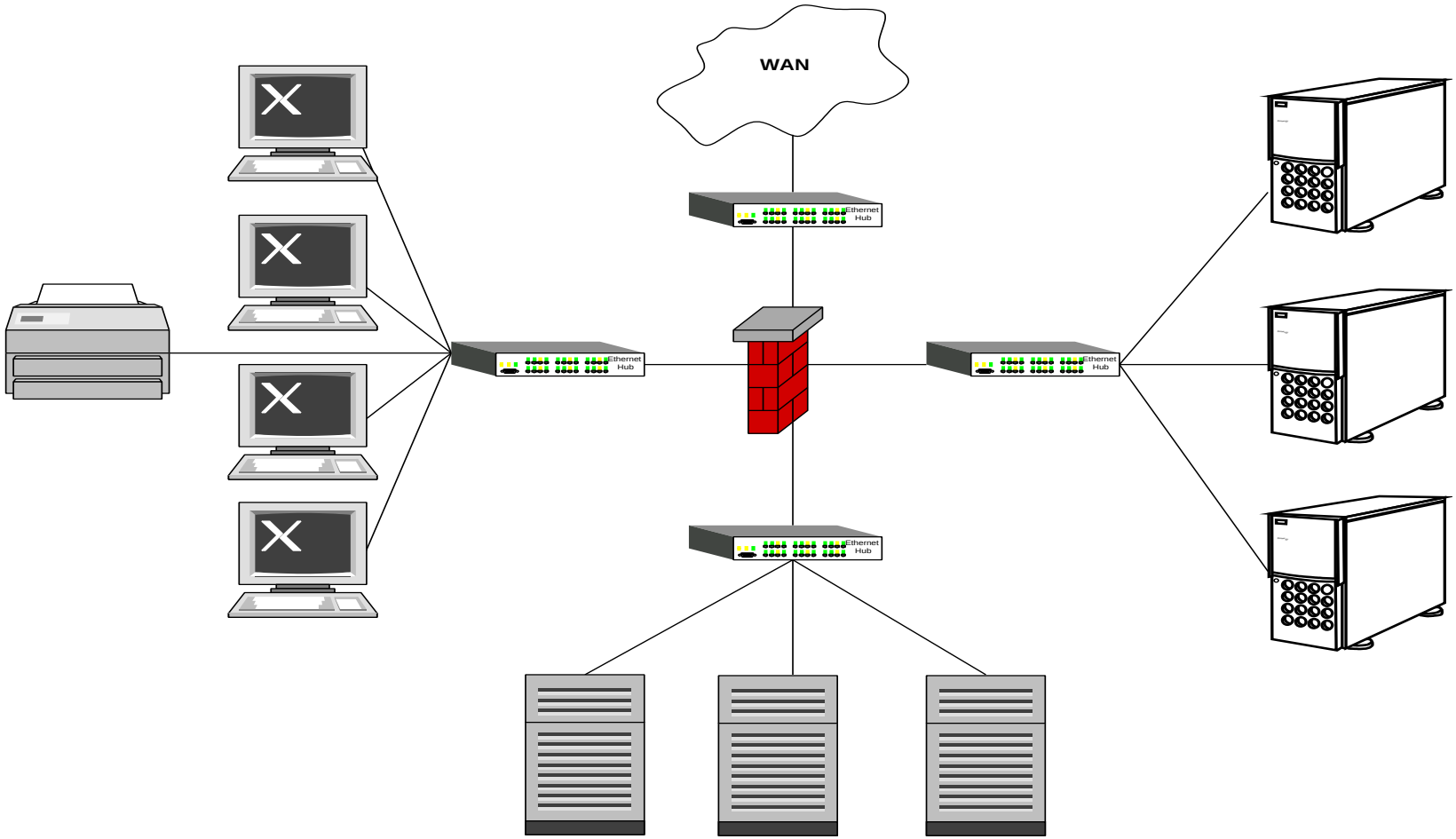
¿Por qué queremos detectar intrusiones?

- ¿Es suficiente la prevención (cortafuegos, parches...)?
- Conocimiento del entorno.
- Detección rápida (¿inmediata?) de potenciales problemas...
- ...y respuesta ante los mismos.
- ...

OBJETIVO: Proteger nuestros recursos.

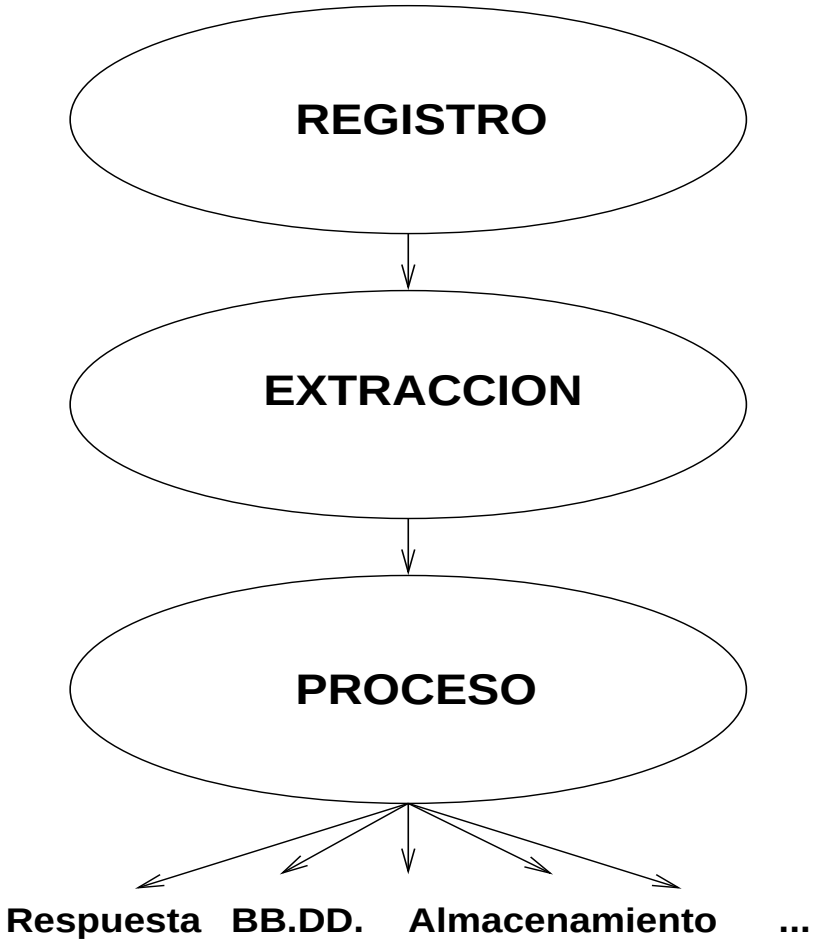
Introducción

Lo que queremos proteger...



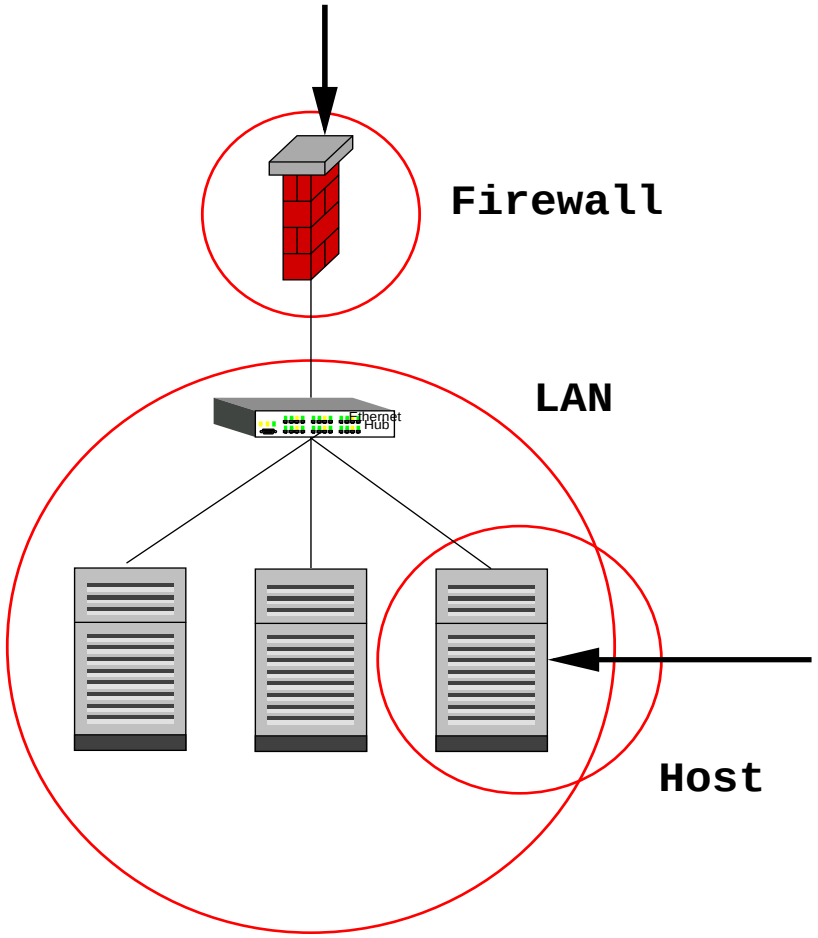
Introducción

¿Cómo protegerlo?



Introducción

¿Dónde protegerlo?



Detección en el cortafuegos

¿Por qué detectar en nuestro *firewall*?

- Elemento crítico en cualquier plataforma.
- Nexo de unión con el mundo exterior.
- Punto por el que pasa todo el tráfico (el 'bueno' y el 'malo').
- Mecanismos simples (no implican modificar la arquitectura, ni probablemente incorporar nuevo *software*)...
- ...y efectivos (baja probabilidad de falsos positivos).

Detección en el cortafuegos

¿Qué información maneja mejor un cortafuegos?

La incluida en las diferentes cabeceras de una trama:

- Direcciones origen y destino.
- Puertos origen y destino (servicios).
- Protocolos.
- *Flags*.
- ...

Detección en el cortafuegos

¿Qué ataques detecto mejor en un cortafuegos?

Los que 'vienen' en dichas cabeceras:

- Todo tipo de barridos de puertos.
- *IP–Spoofing*.
- Ciertos troyanos.
- Tráfico anómalo.
- ...

Detección en el cortafuegos

Registro en un cortafuegos

- `ipchains (-l)`
- `iptables (-j LOG)`
- `ipfilter (log)`
- *Firewall–1* (`Track log`)
- ...

Detección en el cortafuegos

Extracción de eventos significativos

- Acceso a puertos 'anómalos':

```
block in log quick on elxl0 from any to any port = 79
block in log quick on elxl0 from any to any port = 31337
```

- Orígenes falseados (p.e. dirección interna que proviene de red externa):

```
block in log quick on elxl0 from 127.0.0.0/8 to any
block in log quick on elxl0 from 192.168.0.0/16 to any
```

- Violaciones de protocolo:

```
block in log quick proto tcp from any to any flags SF/SF
block in log quick proto tcp from any to any flags SR/SR
```

- ...

Detección en la red

¿Por qué detectar en la red?

- ¿Y por qué no?
- Un cortafuegos no es la panacea: ni detecta todos los ataques ni mucho menos es capaz de detenerlos.
- Facilidad de detección en una red:
  - Mucho *software* gratuito.
  - Pocos recursos *hardware* (y baratos).
  - Escasas modificaciones de la arquitectura.
  - ...

Detección en la red

Requisitos mínimos

- *Hardware.*
  - PC de gama media + 2 tarjetas de red (TAP + Gestión).
  - *Switch* con soporte para *port mirroring.*
- *Software.*
  - Sistema Unix (Linux, FreeBSD, Solaris...).
  - *Software* de detección:
    - ISS RealSecure.
    - Dragon IDS.
    - ...
    - **SNORT.** $\Leftarrow$

Detección en la red

¿Por qué SNORT?

- Sistema probado y fiable.
- Correcto soporte y actualización.
- Funciona sobre diferentes operativos y arquitecturas.
- Productos de terceros (AR, BBDD...).
- Barato (¡Libre!)

Detección en la red

Registro de eventos

- SNORT guarda registros en `/var/log/snort/` (‘`-l`’).
- Ficheros binarios y de texto plano.
- Ejemplo:

```
[**] [1:886:8] WEB-CGI phf access [**]
[Classification: sid] [Priority: 2]
{TCP} 192.168.1.13:1885 -> 192.168.2.2:80
```

- Otras opciones:
  - `snort -s` junto a `syslog`.
  - Módulos de salida: `alert-syslog`, `alert-fast`, XML, CSV...
  - ...
- Por defecto, nivel de registro demasiado alto.

Detección en la red

Extracción de eventos significativos

- ¿Qué son eventos significativos?
  - Ponderación de los diferentes ataques.
  - Recursos críticos.
  - Ataques repetitivos (fuente, tipo...).
  - ...
- ¿Cómo los extraigo?
  - *Parsing* de ficheros de registro.
  - Registro sólo de los eventos significativos.
  - Registro diferenciado de eventos.
  - ...
- Consideraciones: tiempo real.

Detección en la máquina

¿Por qué detectar en la propia máquina?

- Ataques no detectables fácilmente en la red.
- Usuarios internos del sistema.
- Decepción de intrusos.
- ...

Detección en la máquina

Problemas a los que nos enfrentamos...

- Gran número de sistemas.
- Entornos heterogéneos.
- Responsabilidades difusas.
- ...

⇓

Ejemplo: UJI

Detección en la máquina

Tipos de detección

- Verificadores de integridad (SIV).
- Monitores de registros (LFM).
- Sistemas de decepción (*Honeypots*).

Detección en la máquina

Verificadores de integridad

- IDEA: Queremos ser capaces de detectar cualquier modificación de ciertos ficheros del sistema.
- ¿Cómo? Generalmente mediante funciones resumen (p.e. MD5).
- Problemas:
  - Resumen inicial. ¿En quién confio?
  - Nivel de privilegio.
  - ...
- Ejemplos: Solaris ASET, Tripwire, md5sum...

Detección en la máquina

Verificación de integridad: ejemplos

```
# find /sbin -type f -exec md5sum {} \; >/etc/firmas.orig
# find /sbin -type f -exec md5sum {} \; >/etc/firmas.now
# diff /etc/firmas.orig /etc/firmas.now \
  2>/dev/null >/dev/null \
  || echo "Cambios en /sbin" | mailx -s "PROBLEMAS" root
#
```

```
# find /dev -type f -perm -4000 && \
  echo "Setuidados en /dev" | mailx -s "PROBLEMAS" root
#
```

Detección en la máquina

Monitores de registros

- IDEA: Queremos monitorizar los *logs* para detectar patrones que puedan denotar una intrusión.
- ¿Cómo? Programas específicos: desde productos comerciales hasta simples *shellscripts*.
- Problemas:
  - Conocimiento del entorno.
  - Nivel de privilegio.
  - ...
- Ejemplos: Swatch, LogSentry...

Detección en la máquina

Monitores de registros: ejemplos

```
# tail -100 /var/log/messages | grep -w su
Apr  9 18:03:55 bruja su[2389]: - pts/2 toni-root
Apr  9 18:04:02 bruja su[2390]: + pts/2 toni-root
Apr  9 18:04:50 bruja su[2408]: + pts/4 toni-root
#
```

```
# grep -w phf /var/log/apache/access_log
192.168.0.1 - - [11/Dec/2002:02:50:09 +0100] \
   "GET /cgi-bin/phf" 404 -
192.168.0.3 - - [12/Dec/2002:15:24:38 +0100] \
   "GET /cgi-bin/phf" 404 -
#
```

Detección en la máquina

Sistemas de decepción

- IDEA: Queremos proporcionar conocimiento negativo en un atacante para poder estudiar sus intenciones, sus métodos, para ganar tiempo y pruebas tras una denuncia...
- ¿Cómo? Simulando errores de configuración o vulnerabilidades que realmente no lo son.
- Problemas:
  - Conocimiento del entorno.
  - Vulnerabilidades asociadas al sistema de decepción.
  - ...
- Ejemplos: PortSentry, FakeBO...

Detección en la máquina

Sistemas de decepción: ejemplos

```
# grep uucp /etc/services
uucp    540/tcp    uucpd    # uucp daemon
# echo "Sonrie a la camara oculta" | nc -l -p 540 &
[3] 2597
#
```

```
# cat /tmp/get_root
#!/bin/sh
(whoami && date ) >>/tmp/.registro
while :; do
    /bin/echo -n "# "
    read orden && echo $orden >>/tmp/.registro
done
# touch /tmp/.registro && chmod 222 /tmp/.registro
# chmod +x /tmp/get_root
#
```

Conclusiones

- Ningún mecanismo de seguridad trabaja **en lugar de** otros, sino **junto a** ellos.
- La detección de intrusos es un mecanismo más que nos ayuda a proteger nuestros activos.
- Es factible (económica, política, técnicamente...).
- Detección multinivel (cortafuegos, red y máquina) con un objetivo común.
- Aún así, ¿podemos garantizar que detectamos todas las intrusiones?
- ...

¿Qué nos falta?

Pero...

- Sensores **independientes** que no 'se ven' entre sí. Objetivo: detección distribuida (DIDS).
- Proceso de eventos: centralización, correlación...
- Respuesta ante incidentes.
- Respuesta automática.
- ...

¡Muchas gracias!

¡¡Muchas gracias a todos!!