

Ciberseguridad

Grado en Ingeniería Informática

M4 - Detección y Prevención de Intrusiones

Oscar Delgado
oscar.delgado@uam.es

Álvaro Ortigosa (Coord.)
alvaro.ortigosa@uam.es

Sistema detector de intrusiones (IDS)

- Alarma contra intrusos de los sistemas informáticos



Detección de Intrusiones

- Proceso de monitorizar las redes y sistemas informáticos para detectar violaciones a la política de seguridad.



Un poco de historia

- El concepto de detección de intrusión fue introducido en el artículo que James Anderson escribió en 1980 para el NIST, llamado “Computer Security Threat Monitoring and Surveillance”.
- Ya en 1990, Heberlein diseñó y codificó el primer IDS de red.
- El “boom” comercial no llegaría hasta 1997, con el primer IDS de ISS.
- Para el año 2000, todas las grandes compañías de seguridad tenían una solución IDS.

Un poco de historia

- Tradicionalmente los hackers habían dirigido sus ataques contra los sistemas operativos, explotando sus vulnerabilidades.
- Los cortafuegos cambiaron radicalmente la situación.
- Actualmente la mayoría de los ataques exitosos se dirigen contra el servicio Web.
- Hoy en día, ¿qué empresa o institución tiene el puerto 80 cerrado?.

¿Qué es una intrusión?

- **Intrusión:** Una secuencia de acciones realizadas por un atacante que resulta en el compromiso de un sistema.
- **Detección de intrusión:** el proceso de identificación y respuesta a intentos de ataques. Como proceso que es, involucra *tecnología, personas y herramientas*.

¿Por qué es necesaria?

- ¿No ha sido suficiente ‘La situación actual’?
- Ok, los estudios demuestran que:
 - La inmensa mayoría de sistemas sufren vulnerabilidades.
 - El número de incidentes de seguridad se incrementa continuamente.
 - Usuarios y administradores son muy lentos aplicando parches a los sistemas vulnerables.
- En consecuencia, algunos expertos creen que los sistemas de computación nunca serán absolutamente seguros.

¿Por qué es necesaria?

- Los mecanismos de seguridad, como controles de acceso y autenticación, podrían desactivarse por un ataque o una mala configuración.
- Casi el 50% de los ataques con éxito *son llevados a cabo por usuarios internos del sistema*.
- Aunque un ataque no tenga éxito, es probable que “alerte” de su intento, lo que puede ser muy útil en el futuro.

¿Por qué es necesaria?

- En resumen, un sistema de detección de intrusión son **los ojos** de nuestra arquitectura de seguridad, que nos permite saber qué está pasando, dentro y fuera de nuestras redes.
- Cuando se detecta un intento de ataque se genera una **alerta**, que debería ser revisada de inmediato por un operador humano.



Términos sobre IDS

- **Falso positivo**

- Se produce cuando un IDS genera una alerta falsa sobre un ataque que no se ha producido.
- La elevada tasa de falsos positivos es uno de los grandes inconvenientes de los IDS.

- **Falso negativo**

- Se produce cuando un IDS no detecta un ataque y no genera la alerta correspondiente.
- Es muy peligroso, porque un único falso negativo puede provocar que una intrusión no sea detectada.

Sistema detector de intrusiones

- Tres componentes funcionales
 - Fuente de información que provee un flujo de registros de eventos
 - Un motor de análisis que encuentra síntomas de intrusiones
 - Un componente de respuesta que genera las reacciones basadas en la salida del motor de análisis

Firewall vs. Antivirus vs. IDS

- Comparten similitudes:
 - Por ejemplo, base de patrones de *mal comportamiento*.
- Difieren en:
 - Qué vigilan.
 - Cómo responden

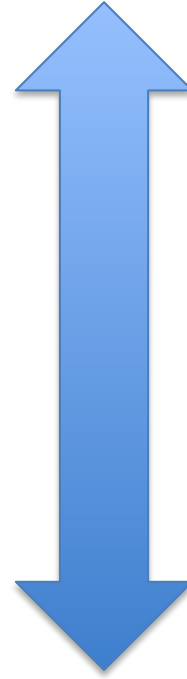
Diferencias: qué vigilan?

- **Firewalls**: previenen conexiones o transmisión de paquetes que violen las reglas de la política de seguridad.
- **Antivirus**: buscan la presencia de ficheros (o con parte de contenido) pre-definidos y la ejecución de comandos “problemáticos”
- **IDS**: buscan comportamientos anómalos del sistema/red, examinando los medios de comunicación y las llamadas al sistema, ya sea usando base de patrones predefinidos o técnicas de profiling.

Diferencias: cómo reaccionan?

- **Firewall**: negar conexión o eliminar un paquete
- **Antivirus**: poner en cuarentena el fichero sospecho y avisar al usuario.
- **IDS**: notificar al administrador del sistema de la sospecha de una intrusión

Más activo



Menos activo

Firewall vs. Antivirus vs. IDS

- Sin embargo la frontera no está tan clara:
 - Por ejemplo, un buen IDS debería detectar la acción de un virus informático.
- Normalmente tienen solapes de funcionalidad.
- Mayor nivel de seguridad si usamos los tres.
 - Defensa en profundidad.
 - Cada uno tiene sus propias limitaciones.

Conceptos de la D.I.

- Arquitectura.
- Estrategia de monitorización.
- Tipo de análisis.
- Temporización.

Arquitectura

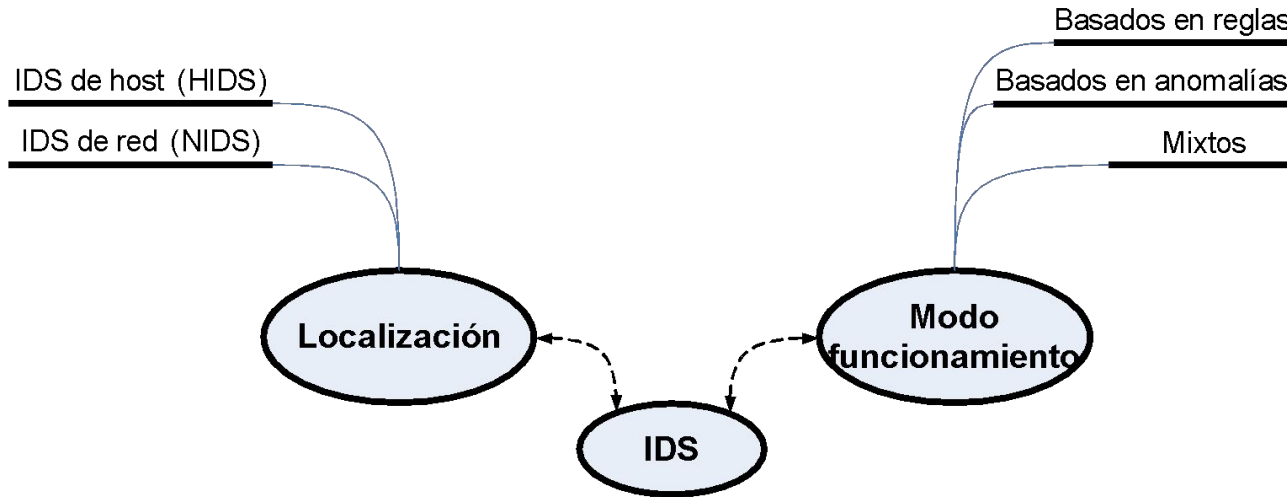
- La información para auditar debe ser almacenada y procesada fuera del sistema protegido.
 - Evitar que un intruso inhabilite el IDS borrando los registros de auditoría.
 - Evitar que un intruso modifique los resultados del detector de intrusiones para esconder su presencia.
 - Minimizar la pérdida de desempeño asociada con la ejecución del sistema detección de instrucciones.
- El **host** vigila al **objetivo**.

Estrategias de monitorización

- Clasificación de acuerdo a la **fuentes de información**:
 - Monitores basados en el host.
 - Monitores basados en la red.
 - Monitores basados en aplicaciones
 - Monitores basados en el objetivo

Tipos de IDS

- Los IDS pueden clasificarse, básicamente, utilizando dos criterios:



Host-based IDS (HIDS)

- Recogen información de los sistemas internos de un ordenador (normalmente a nivel del S.O.)
 - Logs (*audit trails*) del S.O y del sistema, por ejemplo.
- Dependen del éxito de los intrusos:
 - Asume que dejarán rastros al intentar adueñarse del equipo.
 - El HIDS intenta detectar esas modificaciones y hacer un informe de sus conclusiones

IDS de host

- Los IDS de host monitorizan, detectan y responden a la actividad del usuario, del sistema y ataques a un máquina dada.
- Esto lo hacen examinando parámetros del sistema operativo y del comportamiento del usuario a través de un *agente*:
 - Uso de CPU y memoria.
 - Monitorización de logs.
 - Intentos fallidos de login.
 - Monitorización del sistema de ficheros.
 - ...

IDS de host

- Son más adecuados para combatir amenazas internas:
 - Como hemos visto, la fuente de ataques más numerosa son los desempleados “deshonestos”.
 - Puertas traseras.
- Mucho más baratos:
 - 50€/PC para HIDS, 8.000€ para NIDS.
- Hay ataques que no son fácilmente detectados por los NIDS.



¿Dónde colocamos los agentes?

- Lo ideal sería en cada máquina de la organización, pero:
 - Los costes serían prohibitivos (estimación de 50-500€ por máquina).
 - La tasa de falsos positivos puede ser muy alta en máquinas que se reconfiguran frecuentemente.
- Las opciones más obvias son:
 - Servidores de “negocio” (perimetrales e internos).
 - Cortafuegos.
 - Servidores Web, DNS y de correo.

IDS de host: ejemplos

- Soluciones sencillas (y baratas) para UNIX:
 - *TCPWrappers*
 - *Syslog*
 - *Swatch*
 - *Tripwire*
 - *OSSEC* (¡Interesante!)
- Soluciones comerciales para Windows:
 - *BlackICE* de ISS
 - *Host Intrusion Prevention* de McAfee
 - *Sentinel* de Enterasys

Resumen: IDS de host

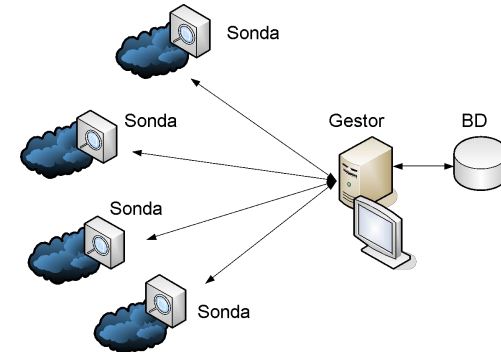
- Los agentes deben instalarse en cada máquina a monitorizar.
- Es una solución costosa de administrar:
 - La política de detección debe ajustarse para evitar una alta tasa de falsos positivos.
 - Esto puede ser un problema en grandes organizaciones, con sistemas muy heterogéneos.
- Requiere de recursos de sistema (CPU y memoria RAM).
- Los HIDS pueden detectar ataques invisibles para un NIDS.

IDS de red

- Los IDS de red (NIDS) capturan y analizan todo el tráfico de un segmento de red, en busca de actividad maliciosa.
- Normalmente usando dispositivos en modo promiscuo.
- Tradicionalmente, los NIDS han tenido problemas para trabajar en:
 - Entornos conmutados.
 - Entornos con comunicaciones cifradas.
 - Redes de alta velocidad (redes Gigabit o de más de 100 Mbps).
 - Ya existen algunos IDS capaces de trabajar en estas redes.

¿Cómo funciona?

- Un IDS de red se compone de:
 - **Sondas:** elementos recolectores del sistema. Capturan el tráfico y realizan un primer paso de procesamiento. Después, envían los eventos generados al sistema gestor.
 - **Sistema gestor:** encargado de almacenar y gestionar los eventos recibidos. Se compone de:
 - Base de datos.
 - Sistema de correlación de eventos.
 - **Consola de administración:** clasifica y muestra los eventos recibidos.



Implantación

- La implantación de un NIDS es un proceso complejo y delicado, que implica:
 - Elección del producto más adecuado (comercial o libre).
 - Elección del tipo de sondas.
 - Elección del número y ubicación de las mismas.
 - Ajuste de la política de cada sonda, para reducir el número de falsos positivos.
- Un error en cualquiera de estos pasos puede resultar muy caro (en todos los sentidos).

Tipos de sondas

- Las sondas pueden ser:
 - *Hardware dedicado*: muchos IDS comerciales utilizan esta fórmula.
 - **Ventajas**: en teoría, pueden alcanzar mayores tasas de captura.
 - **Inconvenientes**: son cajas negras, de las que se desconoce su funcionamiento.
 - *Un simple PC*: con sus tarjetas de red en modo promiscuo.
 - **Ventajas**: reduce costes y facilita el mantenimiento.
 - **Inconvenientes**: en teoría, podrían ser menos fiables.

Ubicación de las sondas

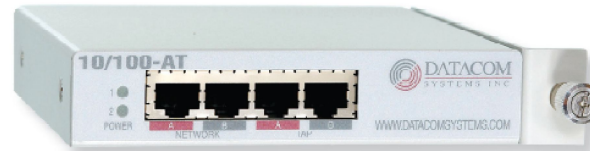
- La ubicación de las sondas es muy importante, pues determina qué tráfico puede monitorizarse.
- En principio, en entornos conmutados sería necesaria una sonda por cada segmento de red.
- Esto puede disparar los costes, así que existen un par de soluciones:
 - Puertos de spanning.
 - Uso de TAPs.

Spanning ports

- Algunos switches tienen un puerto de span (o de agregación), por el que se recibe todo el tráfico que pasa por el mismo.
- Si se conecta la sonda a ese puerto, ésta es capaz de ver todo el tráfico.
- Algunos problemas:
 - Si el tráfico es muy alto, el switch puede tener problemas de rendimiento y empezar a descartar paquetes.
 - Algunos switches sólo permiten un puerto de span por VLAN.

TAPs

- Un tap es un dispositivo hardware que se coloca en mitad de un cable de datos y envía una copia del tráfico que pasa a través de él a uno o más puntos.
- No tienen problemas de rendimiento ni necesitan configuración.
- Se debe elegir su ubicación con cuidado, pues puede sobrecargar fácilmente las sondas.



Capacidad de las sondas

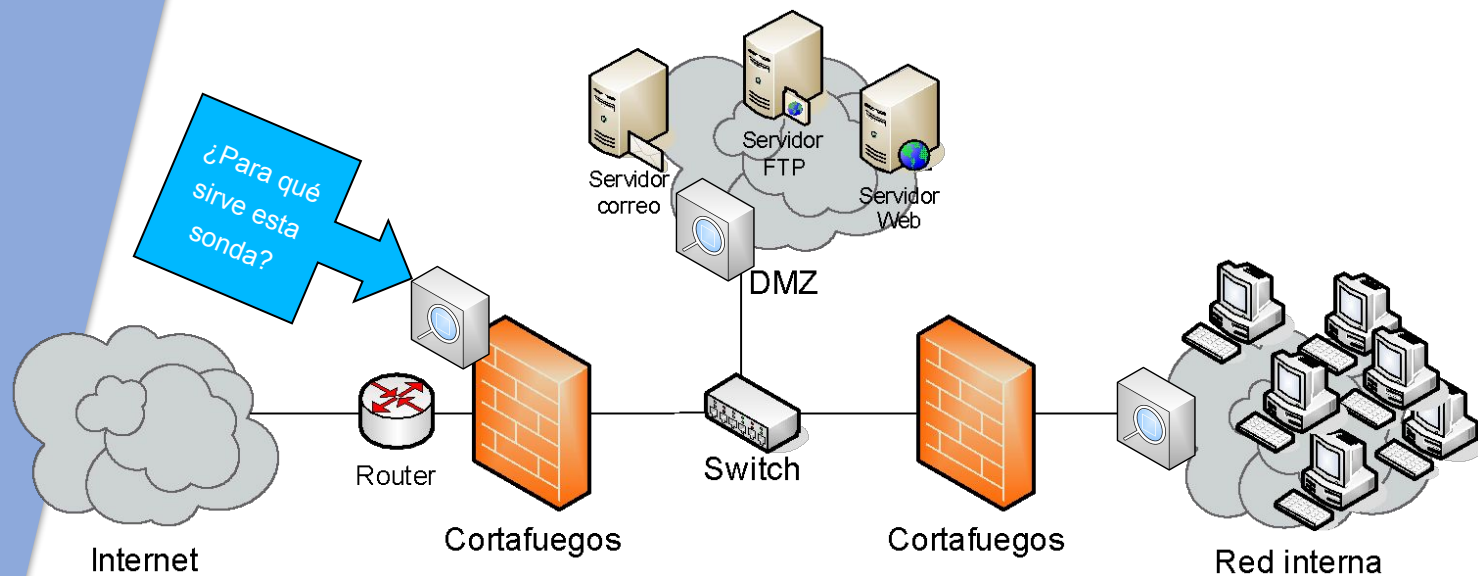
- Las sondas tienen una capacidad de procesamiento limitada: unos 80 Mbps a plena carga.
- Hay que repartir el tráfico entre las sondas en base a varios criterios:
 - La carga de la red determinará el número mínimo de sondas.
 - Distintas zonas de la red tendrán distintos tipos de tráfico:
 - No debe utilizarse una única sonda para todos ellos, pues esto impediría ajustar las políticas de cada sonda de forma individual.

ACTIVIDAD



Ejercicio de ubicación de sondas

¿Cuántas y dónde colocaríais las sondas de un NIDS en esta topología de red?



Otros dos

- Basado en aplicación: recogen información de aplicaciones en ejecución.
 - Por ejemplo logs de eventos y otras fuentes internas a la aplicación.
- Basados en el objetivo: estos monitores generan sus propios datos.
 - Usan funciones hash para controlar la modificación de objetos, y verificar si están de acuerdo a la política de seguridad.
 - En vez de actividades vigilan objetos.

IDS basados en firmas

Tipo de análisis

- El motor de análisis toma los datos de las fuentes y los examina buscando síntomas de ataques u otras violaciones a la política.
- La mayoría de los enfoques implican detección de malos usos (basados en firmas), detección de anomalías o combinación de ambos.

IDS basados en firmas

- Estos IDS funcionan buscando patrones previamente definidos en aquellos parámetros que monitorizan:
 - El tráfico de red para los NIDS.
 - El sistema de ficheros, por ejemplo, para los HIDS.
- Cuando se detecta una concordancia, se dispara la regla y se genera una alerta.
- Esta alerta se envía a una consola central, donde un operador humano debería analizarla y actuar en consecuencia.

IDS basados en firmas

- Para un HIDS una firma podría ser algo como:

```
IF número_intentos_login_último_minuto >= 10  
AND mismo_usuario THEN ALERT  
("Ataque de fuerza bruta contra cuenta de usuario")
```

- Por otro lado, para un NIDS:

```
IF misma_dir_IP_origen AND diferentes_puerto_destino  
AND num_conexiones >= 10 THEN ALERT  
("Escaneo de puertos")
```

IDS basados en firmas

- Las firmas reales no son tan sencillas. Esta es una firma de Snort para detectar un tipo de ataque contra dispositivos Cisco:

```
alert tcp any any -> $HOME_NET 23 (msg: "BLEEDING-EDGE  
EXPLOIT Cisco Telnet Buffer Overflow"; flow:  
to_server,established; content:"|3f 3f 3f 3f 3f 3f 3f 3f 3f 3f 3f 3f  
3f 3f 3f 3f 61 7e 20 25 25 25 25 25 58 58|"; threshold: type limit,  
track by_src, count 1, seconds 120;  
reference:url,www.cisco.com/warp/public/707/cisco-sn-20040326-  
exploits.shtml; classtype: attempted-dos; sid: 2000005; rev:4; )
```

IDS basados en firmas: implantación

- Todos los IDS basados en firmas necesitan de una configuración inicial cuidadosa.
- Esta configuración consiste, principalmente, en el refinado de la base de reglas.
- Este refinado es necesario para adaptar el conjunto de reglas por defecto del IDS al tipo de tráfico más habitual de la red que se pretende monitorizar.
- Si no se lleva a cabo este paso, tendremos un sistema prácticamente inservible, que generará multitud de falsos positivos -> **efecto “que viene el lobo”**.

Refinado base de reglas

- Cuanto más general sea una regla, más posibilidades tendrá de detectar un ataque en particular.
- Aunque también, más posibilidades de que cualquier otro paquete, que no tenga nada que ver, concuerde con sus criterios y provoque un falso positivo.
- Pero... si hacemos la regla demasiado específica, podemos generar un falso negativo, al no comprobar alguna característica importante del tráfico malicioso.
- Se trata, por tanto, de encontrar un equilibrio entre ambos criterios, que adapten la base reglas a nuestro tráfico sin provocar nunca falsos negativos.

Refinado base de reglas: un ejemplo

- Una regla que en Snort suele generar muchos falsos positivos es SHELLCODE x86 NOOP.
- Su función es tratar de detectar ataques de desbordamiento de búfer.
- Su firma es la siguiente:

```
SHELLCODE x86 NOOP content:"|90 90 90 90 90 90 90 90 90 90  
90 90 90 90|"; depth:128
```

- Es decir, cualquier paquete que contenga una cadena de bytes con valor 0x90 repetidos un número concreto de veces (¿un e-mail? ¿un archivo de sonido o video?) provocará un disparo de esta regla.

Snort

- Sin duda, el mejor IDS libre y uno de los mejores de todo el mercado.
 - Muy extendido.
 - Fiable.
 - Extensísima documentación.
- Desarrollado y utilizado por una comunidad muy activa, lo que proporciona:
 - Bases de reglas exactas y rápidamente actualizadas.
- La base de reglas se almacena en ficheros de texto plano:
 - Control total sobre los mismos.
 - No tan obvio: muchos IDS comerciales son “cajas negras”.
 - No se conoce ni se tiene control sobre su base de reglas.
 - Por tanto, no se puede reducir la tasa de falsos positivos manualmente.

Ejemplos de IDS

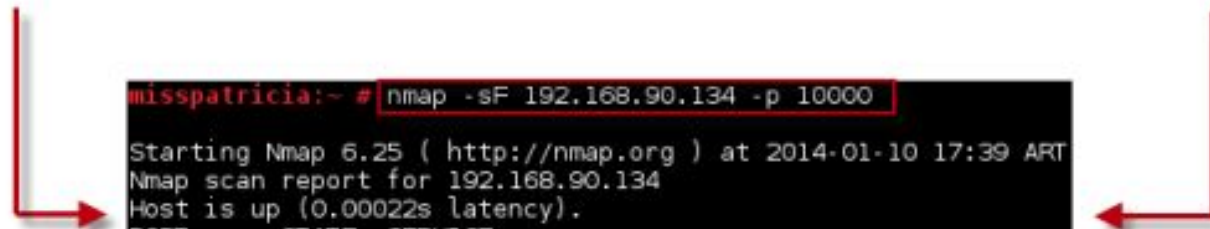
- Snort
 - Pertenece a la categoría de NIDS.

Una vez que *snort* ya se está ejecutando, podrá visualizarse en la consola todas aquellas actividades que ocurran a nivel de red. En la siguiente imagen puede visualizarse un simple **ping al host**, donde se indica la dirección IP de origen y destino.

```
01/10-13:52:32.440936 192.168.90.1 -> 192.168.90.134
ICMP TTL:64 TOS:0x0 ID:0 IpLen:20 DgmLen:84 DF
Type:8 Code:0 ID:976 Seq:1 ECHO
=====
01/10-13:52:32.441002 192.168.90.134 -> 192.168.90.1
ICMP TTL:64 TOS:0x0 ID:38854 IpLen:20 DgmLen:84
Type:0 Code:0 ID:976 Seq:1 ECHO REPLY
=====
```

Uso de Snort

```
alert tcp any any -> 192.168.90.0/24 any (msg:"Paquete FIN detectado dentro de la red";  
flags: F; sid: 100000;)
```



```
misspatricia:~ # nmap -sF 192.168.90.134 -p 10000  
Starting Nmap 6.25 ( http://nmap.org ) at 2014-01-10 17:39 ART  
Nmap scan report for 192.168.90.134  
Host is up (0.00022s latency).  
PORT      STATE SERVICE  
10000/tcp  closed snet-sensor-mgmt  
MAC Address: 00:0C:29:6B:16:39 (VMware)  
Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
```

```
[**] [1:100000:0] Paquete FIN detectado dentro de la red [**]  
[Priority: 0]  
01/10-15:39:16.023723 192.168.90.1:39316 -> 192.168.90.134:10000  
TCP TTL:54 TOS:0x0 ID:56146 IpLen:20 DgmLen:40  
*****F Seq: 0x4D46CC45 Ack: 0x0 Win: 0x400 TcpLen: 20
```

IDS basados en firmas

- Ventajas
 - Son conceptualmente sencillos.
 - Flexibles:
 - Posibilidad de escribir nuevas firmas cuando se descubren nuevos ataques (¡muchos IDS comerciales no lo permiten!).

IDS basados en firmas

- Inconvenientes
 - No detectan nuevos ataques que no hayan sido definidos previamente en su base de reglas.
 - Necesitan de un continuo ajuste de su base de reglas. Si no se realiza:
 - La tasa de falsos positivos puede incrementarse con el tiempo.
 - El sistema pierde toda su eficacia, pues es incapaz de detectar las amenazas más recientes que son, claramente, las que más se utilizan.

IDS basados en anomalías

IDS basados en anomalías

- Este tipo de IDS funciona generando un modelo del comportamiento “normal” o habitual del sistema o del usuario.
- Después, monitoriza la actividad del sistema o de la red, clasificándola como **normal** o **anómala**: todo lo que no sea normal, es anómalo.
- El punto clave es que esta clasificación se lleva a cabo utilizando **heurísticas**, en vez de patrones o firmas.

IDS basados en anomalías

- Para generar el modelo, se han utilizado todo tipo de técnicas de inteligencia artificial:
 - Redes neuronales.
 - Sistemas clasificadores difusos.
 - Mapas autoorganizados o de Kohonen.
- Pero ... no funcionan.
- Los resultados no son los esperados:
 - El tráfico de una red en producción tiene muy poco de “normal”.

Resumen: IDS basados en anomalías

- Su ventaja principal es que podrían detectar un tipo de ataque desconocido (siempre que sea suficientemente “anómalo”).
- Los IDS basados en anomalías son la eterna promesa del campo de la detección de intrusión:
 - Reducirían la tasa de falsos positivos.
 - Detectarían ataques desconocidos.
- Después de cientos de artículos en la literatura (incluidos unos cuantos de un profesor de esta asignatura), seguimos en, casi, el mismo punto.
 - Es un campo de investigación abierto. ¿Alguien se apunta?

Combinación de técnicas

- Significativas ventajas:
 - El análisis de anomalías protege contra ataques nuevos o desconocidos.
 - El análisis de malos usos previene que un adversario con mucha paciencia pueda gradualmente convertir un comportamiento raro en algo normal.

Temporización

- Modo batch (basado en intervalos):
 - Los datos se envían al motor de análisis en un fichero, abarcando eventos de un período determinado.
 - Los resultados son obtenidos **después** que la intrusión ha tenido lugar.
 - Modelo apropiado cuando el ancho de banda / capacidad de procesamiento no es suficiente para un análisis en tiempo real.

Temporización (ii)

- En tiempo real:
 - Los datos son enviados al motor de análisis a medida que los eventos ocurren (o con un pequeño retardo) y son procesados inmediatamente.
 - El proceso es suficientemente rápido para permitir que los resultados del análisis **afecten** el progreso o resultado final de cualquier intrusión que detecta.
 - Permite, llegado el caso, **respuesta automática**.

Objetivos

- Atribución (*accountability*):
 - Capacidad de atribuir responsabilidad de una actividad o evento a quien corresponda.
 - Normalmente para pedir compensación / responsabilidades -> ayuda que sea una persona (y no una máquina).
 - Más útil aún sería obtener direcciones físicas u otros enlaces al mundo físico.

Objetivos (ii)

- Respuesta:
 - Una respuesta ocurre cuando el análisis produce un resultado accionable.
 - No limitado a tomar represalias contra el atacante.
- Ejemplos:
 - Registrar resultados de análisis (posterior informe).
 - Disparar alarmas de una variedad predefinida (mensaje en consola, SMSs, mails, etc).
 - Modificar la configuración del objetivo (x ej. firewall). □ **IPS** (prevention)
 - Contra atacar.

IPS = IDS 2.0

Sistemas de prevención de
intrusión

Sistemas de prevención de intrusión

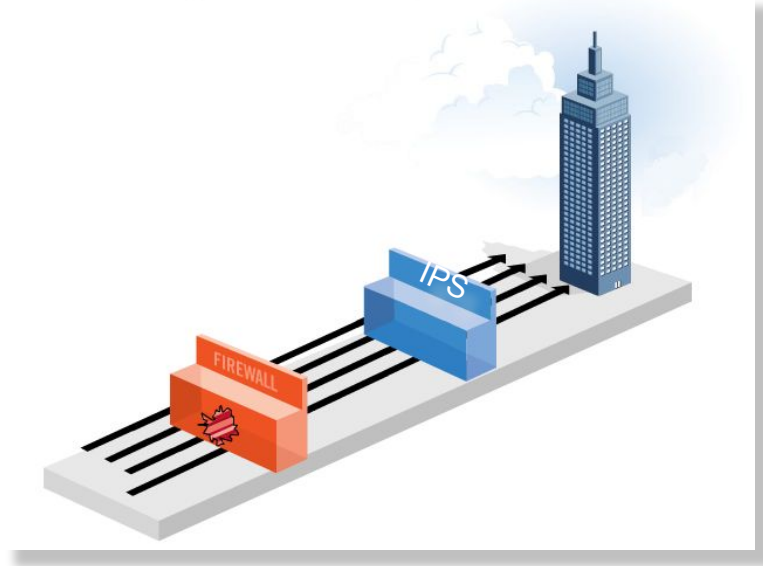
- Los IDS son elementos completamente pasivos: sólo detectan los ataques.
- Las acciones correspondientes tendrán que llevarlas a cabo personal humano.
- Con un volumen alto de eventos o mucha complejidad de los mismos, el tiempo de reacción (en el caso mejor) puede ser de mucho minutos.
- Para entonces, el atacante ya está de vuelta en casa.

Sistemas de prevención de intrusión

- El paso lógico es, entonces, dotar de capacidad de acción a los IDS.
- Los sistemas de prevención de intrusión (IPS) pueden, por tanto, reaccionar a los ataques, llevando a cabo alguna acción previamente programada.
- Sin embargo...
 - Los IPS no tienen ninguna “inteligencia”.
 - Toda la reacción de la que son capaces es cortar conexiones, para evitar que el ataque tenga éxito.

Sistemas de prevención de intrusión

- Inconvenientes:
 - Para funcionar correctamente, necesitan ser conectados “en serie”.
 - Son un cuello de botella y un punto único de fallo.
 - Para mitigar este problema funcionan en modo *fail-over*: si fallan, dejan pasar el tráfico sin analizar.



Sistemas de prevención de intrusión

- Más inconvenientes:
 - Un falso positivo en un IDS no tiene ninguna consecuencia. En todo caso, tiempo perdido para el analista.
 - Sin embargo, un falso positivo en un IPS significa una conexión cortada.
 - Imaginad un banco online. Sus activos más importantes son su reputación y la disponibilidad constante de su infraestructura.
 - ¿Quién es el valiente que se arriesga a conectar un IPS a su red a las 9 de la mañana?.

Determinación de estrategia

- La estrategia óptima dependerá de factores como:
 - Nivel de criticidad o sensibilidad del sistema protegido.
 - La naturaleza del sistema (x ej, complejidad del hardware y plataformas de software).
 - La naturaleza de la política de seguridad de la organización.
 - El nivel de amenaza en el entorno donde el sistema es operado.

Engañando a un IDS

¡Nos atacan!

- En general, existen dos formas de “atacar” a un IDS:
 - Decimos “atacar” porque es un ataque lógico, no físico, porque el IDS no debería ser alcanzable por red.
 - Provocar que el IDS genere muchísimas alertas, con la intención de camuflar las alertas reales entre la multitud.
 - Conociendo las firmas que utiliza el IDS, utilizar técnicas de evasión para que no se disparen las reglas correspondientes.

IDS Killers

- Su funcionamiento es sencillo:
 - Toman como argumento una base de reglas, que para Snort son públicas, y generan paquetes que coinciden con las firmas de dichas reglas.
 - Si ahora se inunda la red monitorizada con este tráfico, el IDS comenzará a generar alertas a máxima velocidad.
 - De esta forma, no es difícil saturar el IDS, o sus bases de datos, provocando su parada.
 - En cualquier caso, existen formas de protegerse de este tipo de ataques.
 - *Stick* y *Snot* son dos herramientas públicas que implementan este ataque.

Técnicas de evasión

- El método anterior es algo “ruidoso”.
- Otro enfoque es utilizar el conocimiento sobre la base de reglas para no ser detectados, utilizando *técnicas de evasión*.
- Algunas de estas técnicas incluyen:
 - **Escaneos “lentos”**: alargar un escaneo de puertos durante varios días, para no levantar sospechas.
 - **Ataques de fragmentación**: se provoca la fragmentación artificial de un paquete en muchos trozos, para tratar de que el IDS no detecte su contenido completo.
 - **Ataques de diccionario inverso**: ataques de fuerza bruta en los que se fija la contraseña, en vez del usuario.



Sistemas señuelo: Honeypots

Sistemas señuelo

- **Problema:** ¿Cómo podemos defendernos de un atacante, cuando no sabemos quién es ni cómo actúa?
- En pocas palabras un honeypot es un trampa:
 - *A nivel de máquina:* corren servicios reales en una máquina “sacrificada” o se simulan estos servicios en un entorno controlado.
 - *A nivel de red:* se simulan máquinas y redes completas, servidores que no existen para hacer creer a un atacante que ha encontrado una organización vulnerable.

Y esto, ¿para qué?

- Básicamente, para aprender de los atacantes.
- Imaginad esta situación:
 - Un grupo hacker ha descubierto una nueva vulnerabilidad en el servicio IMAP, que corre en el puerto 143 TCP.
 - El cortafuegos de cualquier organización debería detener las conexiones entrantes a ese puerto.
 - Pero entonces ... el ataque no podrá ser llevado a cabo y no tendremos ningún detalle sobre el mismo.
 - Si colocamos un honeypot en la red, podremos obtener información sobre qué están tratando de hacer los atacantes.

Tipos de honeypots

- En función de lo real que sea la simulación los honeypots se consideran de:
 - **Baja interacción**
 - ✓ Simulan servicios, aplicaciones y sistemas operativos.
 - ✓ Suponen un riesgo bajo y son fáciles de implantar y mantener.
 - Son fácilmente detectables por atacantes con experiencia.
 - Capturan una cantidad de información limitada.
 - **Alta interacción**
 - ✓ Servicios, aplicaciones y SO's reales.
 - ✓ Capturan mucha información
 - Suponen un alto riesgo (pueden ser utilizadas como plataformas para nuevos ataques) y son difíciles de mantener.

Y esto, ¿cómo se hace?

- Se establece una red muy controlada, donde cada paquete que entra o sale es monitorizado, capturado y analizado.
- Existen buenas herramientas libres para esta tarea:
 - **Sebek**: para sistemas de alta interacción.
 - **Honeyd**: para sistemas de baja interacción.

El futuro

El futuro

- Retos para los IDS actuales:
 - *Reducir la tasa de falsos positivos*
 - Es el gran problema de los IDS actuales. Tiene difícil solución. ¿Quizás la correlación de eventos?
 - *Correlación*
 - Hace referencia al cruce de información de diversas fuentes para obtener mejores conclusiones: “Si sé que un ataque ha sido dirigido contra un servicio que no existe en la máquina objetivo, ¿debería generar una alerta?”

El futuro

- Más retos para los IDS:
 - *Ser capaces de operar a altas velocidades*
 - Las redes cada vez son más rápidas. La mayoría de los IDS actuales no pueden trabajar a velocidades de Gigabit. (Los que lo hacen, lo consiguen a duras penas).
 - *Cifrado*
 - El uso de cifrado, afortunadamente, es cada vez más general. Un IDS no puede, en general, inspeccionar tráfico cifrado.
 - *Integración con otros dispositivos (routers, cortafuegos, etc...)*
 - Un cortafuegos podría ajustar sus reglas, o un router sus rutas en base al tráfico o ataques que el IDS detecta.

El futuro

- Las redes inalámbricas son cada más utilizadas en todos los ámbitos.
- Son, además, especialmente vulnerables:
 - Ataques al protocolo WEP.
 - Inyección de tráfico.
 - Suplantación de usuarios.
 - Ataques de denegación de servicio.
- La detección de intrusión en este entorno es una necesidad natural:
 - Ya existen proyectos de IDS y honeypots para redes inalámbricas.

Últimas palabras...

Resumen

- Los IDS y los honeypots son como una mascota:
 - Al principio hacen mucha ilusión ...
 - ... luego descubres que necesitan mucho “cariño”.
- Ambas medidas son inútiles si no se está dispuesto a dedicarles mucho tiempo y recursos para mantenerlos operativos.
- Sin embargo, un IDS es imprescindible en cualquier infraestructura de seguridad seria.
- Quizás la solución más equilibrada sea:
 - Utilizar HIDS en las máquinas más críticas, como servidores de producción, pasarelas VPN, etc...
 - Combinar con NIDS en el resto de la red.

Resumen

- Por otro lado, los honeypots son una tecnología relativamente compleja.
- No son adecuados si no se tiene interés real en mantenerlos en buenas condiciones. Esto implica:
 - *Estudiar la información recolectada*: este puede ser un paso que necesite de mucho tiempo y esfuerzos.
 - *Compartir los análisis con la comunidad de seguridad*: sólo de esta forma se puede avisar de forma temprana de nuevas amenazas.
- ¡Un honeypot descuidado es la mejor forma de buscarse problemas!
 - Pues, con seguridad, será utilizado como plataforma para otros ataques.

so does anyone have
any questions?



©hugh

Malware

Introducción al análisis forense (digital)