Seguridad en los Sistemas de Información

Dr. Manuel Sánchez Rubio

IDS / IPS

Índice

- Introducción
- Usos de un IDS
- Tipos de IDS
- Ataques contra un IDS
- Ejemplo: Snort sobre FC6

- ¿Qué tarea realiza un IDS?
  - Monitorización de un recurso en busca de violaciones de la política de seguridad establecida
  - ¿Recurso?
    - Dentro de una máquina
      - ficheros
    - Tráfico de una red
      - todo tipo de paquetes

Introducción

¿Por qué necesitamos un IDS?

- Los cortafuegos no son infalibles
  - Confiar únicamente en un cortafuegos es arriesgado, en el mejor de los casos
- Protegen contra ataques que no detectan los cortafuegos
  - USB con troyano □¿qué pasa?
  - Ataques intranet□intranet
- Seguridad en profundidad
  - Nos protege frente a fallos de otros sistemas de seguridad (no confiar sólo en un sistema de seguridad)
- Automatización □ Facilitan la gestión de la seguridad
  - Avisar al gestor de alarmas
  - Posibilidad de actuación programada

Introducción

¿Qué ofrece un IDS? :

- Mejora la seguridad interna
  - Cortafuegos protege el perímetro
  - Gran parte de los ataques se producen desde el interior (consciente o inconscientemente)
- Ofrece información sobre el uso de un sistema
  - Sniffer continuo de la red estadísticas de uso
- Permite realizar el seguimiento de un ataque en curso
  - El ataque mejor pararlo cuanto antes, pero si estamos interesados en seguir las fases del ataque el IDS nos puede ayudar (descubrir al culpable)

Tipos de IDS

- Arquitectura general de un IDS :
  - Recogida de información
  - Análisis de información
  - Emisión de respuesta

Tipos de IDS

- Básicamente, dos tipos de IDS + uno
  - HIDS (Host IDS)
    - Protegen únicamente el equipo en el que están instalados
  - NIDS (Network IDS)
    - Instalación independiente (un solo equipo). Suele ser conveniente equipo dedicado o elemento de construcción de red (swicth), si es posible
    - Protegen una red o parte de ella (depende del tráfico que pasa por ellos)
  - Modelado de comportamiento
    - Modela un equipo y alerta si el comportamiento no es normal

HIDS

- Recogida de información
  - Logs del Sistema: acceso, errores, etc.
  - Propia estructura de ficheros del SO: tamaño, fecha de modificación, acceso, etc.
- Análisis de la información
  - Búsqueda de patrones extraños
    - Ej: usuarios no autorizados
  - Búsqueda de modificaciones en el SO (un binario con un tamaño diferente, por ejemplo)
    - Intrusos intentan cubrir sus huellas
- Respuesta
  - Emisión de alarmas al administrador
  - Ejecución de comandos
    - Ej: matar procesos

HIDS

Ventajas:

- Detectan troyanos y modificaciones del sistema con facilidad: tienen acceso al sistema
- Pueden tratar con tráfico cifrado (observan antes del cifrado/después del descifrado)
  - Ej: troyano que se comunica con C&C de forma cifrada
- Pueden detectar ataques que no detecta un NIDS
  - Ej: accesos locales no autorizados (logs del sistema)

HIDS

- Desventajas :
  - No detectan ataques de red (para eso están los NIDS)
  - Son vulnerables a ataques a las fuentes de información Logs
  - El HIDS, al residir en el equipo, puede ser atacado (ej: intruso intenta desactivarlo)
  - Es necesario poner uno en cada equipo del sistema
  - Es complicado configurarlo de forma correcta y eficiente
    - Depende de tus habilidades … (meter horas)

HIDS

Ej: Tripwire (comercial, pero existe versión de código libre)

- HIDS basado en recogida de información del sistema de ficheros
- Puesta en marcha: generación de base de datos de hashes de los ficheros; cifrado de la misma
- Se guarda dicha BD en otro equipo
  - Intruso no la puede tocar …
- Activación: Creación de la BD del equipo en el estado actual, y comparación con la original
- Detecta modificaciones de los ficheros indicados
  - No es viable supervisar todos los ficheros

NIDS

- Recogida de información
  - Tráfico de la red: igual que sniffers, pero analiza
- Análisis de la información
  - Búsqueda de patrones
    - Ataques tienen estructura bien conocida: ej DoS con TCP SYN (saturar server)
    - Tráfico de troyanos en propagación: patrones dentro de los paquetes
  - Búsqueda de “tráfico extraño”
    - Podríamos entrenar al sniffer con tráfico “normal”
- Respuesta
  - Emisión de alarmas
  - Respuesta coordinada con un cortafuegos
    - Ej: ataque DoS desde una IP determinada. IDS indica al contafuegos que haga bloqueo
    - Válido para ataques que pasen por el cortafuegos

NIDS

Ventajas:

- Un solo NIDS puede proteger a varios equipos
  - Supervisa tráfico con origen y destino a muchos equipos
- Dispositivo pasivo (no consume recursos de otros equipos, y puede hacerse invisible)
- Son capaces de detectar ataques en tiempo real (respuesta coordinada)
  - HIDS avisa cuando ya está en intruso
  - NIDS supervisión de tráfico en tiempo real respuesta en tiempo real

NIDS

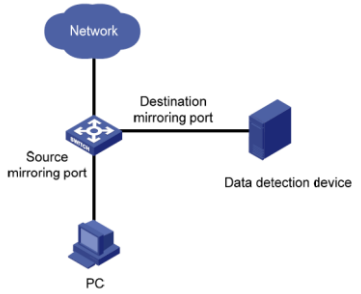
- Análisis por patrones o firmas:
  - Detectan el ataque comparándolo con su base de datos de firmas
    - Paquete payload se compara con patrones previamente descargados (webs los publican)
  - Alto rendimiento, e identifican el ataque realizado
    - La firma identifica el ataque
  - No sirven contra ataques innovadores, o que no estén en la BD
    - Muy importante actualizar patrones !!!

NIDS

Análisis heurístico

- "No sabemos si es peligroso, pero no es normal"
- Detecta la existencia de "tráfico anómalo"
  - Es necesario dejarlos un tiempo "aprendiendo" el tráfico de la red
  - Utilizan "inteligencia": redes neuronales
  - Modelan la "normalidad" y avisan si van algo "anormal"
- Protegen contra cualquier ataque
  - "Si no es normal, es ataque"
- Muy complicados de configurar (generan muchos falsos positivos)

NIDS



Desventajas:

- Problemas con redes conmutadas (se comporta como un “sniffer”)
  - Solución: switchs con puertos de análisis (mirroring ports)
- Pueden perder paquetes si trabajan en redes muy saturadas (>100Mbps, problemas)
- No entienden el tráfico cifrado
- No saben detectar el éxito de un ataque, solo que se ha producido
  - Ej: ve tráfico de DoS pero, ¿han conseguido colgar el servidor?

NIDS

Ej: Snort

- NIDS basado en el análisis de red
- Funciones adicionales para detección de portscanning, ataques sobre HTTP, etc...
- Análisis por patrones o por heurística
- Conexión a BD para almacenar incidencias, así como para establecer respuestas

SNORT

- Snort sigue la filosofía Unix de configuración
  - La configuración es en texto plano
  - Potente y compleja
- La configuración de Snort consiste en:
  - Configuración global (snort.conf)
  - Ficheros de reglas (*.rules)

SNORT

- ¿Cómo conseguir los ficheros de reglas?
  - Descarga de www.snort.org
    - Suscriptores: actualización inmediata
    - Registrados: 30 días después
    - No registrados: Cada vez que actualizan snort
  - Descarga de otros usuario (www.bleedingsnort.com)
  - Escritura propia

- snort.conf (/etc/snort.conf)

var HOME_NET 192.168.3.0/24

var EXTERNAL_NET !$HOME_NET

var DNS_SERVERS [192.168.3.1,192.168.3.10]

var HTTP_SERVERS [192.168.3.1,192.168.3.2,192.168.3.88]

var HTTP_PORTS 80

var RULE_PATH /usr/local/snortrules

[un montón de opciones de configuración de snort]

include $RULE_PATH/local.rules

include $RULE_PATH/bad-traffic.rules

include $RULE_PATH/attack-responses.rules

include $RULE_PATH/bleeding-all.rules

SNORT

- Aspecto de la regla más básica

alert tcp any any → any any (msg:”Hola mundo”; sid: 1)

SNORT

alert tcp any any -> any any (msg:"Hola mundo"; sid: 1)

- Campos de la cabecera
  - Acción
    - alert: genera alerta y log paquete
    - log: log del paquete (binario)
    - pass: ignora el paquete
    - activate: alerta y activa una regla dinámica
    - dynamic: inactiva hasta que se activa
    - drop: hace que iptables tire el paquete
    - reject: igual que drop pero avisa

SNORT

alert tcp any any -> any any (msg:"Hola mundo"; sid: 1)

- Protocolos (TCP, UDP, ICMP, IP)
- Direcciones IP (src, dst) pueden ser
  - Variables ($HOME_NET)
  - Direcciones IP individuales (1.1.1.1)
  - Bloques CIDR (192.168.0.0/24)
- Puertos pueden ser
  - Puertos individuales
  - Rangos de puertos ("80:85",":1024","1025:")
- Operador director
  - ->, <>

SNORT

alert tcp any any -> any any (msg:”Hola mundo”; sid: 1)

- Cuerpo de la regla
  - La parte más compleja
  - Entre paréntesis
  - Concatenación de opciones (palabras clave con parámetros opcionales) separados por ;
  - Opciones:
    - Generales: informan sobre la regla
    - Payload: buscan información en el payload
    - Non-payload: información no relacionada con el payload (como conexión establecida o no)

SNORT

- Opciones generales
  - msg: especifica el mensaje de la alerta
  - reference: incluye una URL para mas info
  - classtype y priority: nos dan una idea sobre el tipo de ataque y la gravedad del mismo
  - sid y rev: identifican de forma única la regla

SNORT

- Ejemplo

alert tcp $EXTERNAL_NET any -> 192.168.3.0/24 80 (msg:"Sample alert"; classtype: web-application-activity; reference:url,http://www.vorant.com/advisories/20060405.html; sid:2000123; rev:1;)

- El uso de classtype implica una prioridad por defecto
- Se puede usar priority para cambiarla
- Cada sid debe ser único (>4000000)

SNORT

- Payload
  - Busca dentro del paquete (no en las cabeceras)
  - Existen muchas opciones de búsqueda
    - “content” busca cadenas de caracteres o binarias

```
alert tcp any any -> any 139 (content:"|5c 00|P|00|I|00|P|00|E|00 5c|";)
```

    - “nocase” busca patrón pero ignorando el case

```
alert tcp any any -> any 21 (msg:"FTP ROOT"; content:"USER root"; nocase;)
```

    - “offset” desplaza un cierto número de caracteres antes de buscar

```
alert tcp any any -> any 80 (content: "cgi-bin/phf"; offset:4; depth:20;)
```

SNORT

- Ejemplo

alert tcp $EXTERNAL_NET any -> 192.168.3.0/24 80

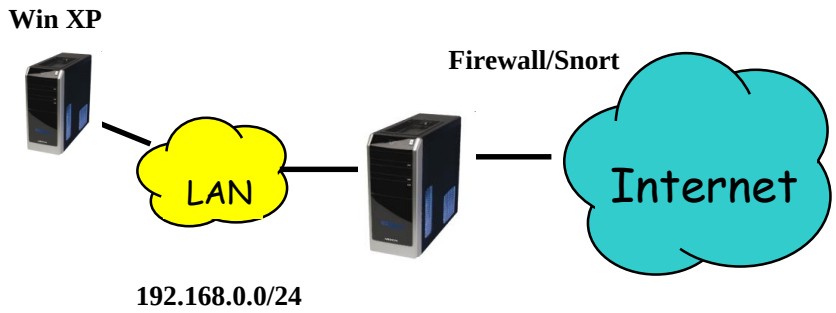
(msg:”Sample alert”; content:”http|3a|//www.vorant.com/test.cgi?id=pwn3d”; nocase; offset:12; classtype: web-application-activity;

reference:url,http://www.vorant.com/advisories/20060405.html;

sid:2000123; rev:1;)

SNORT

- Ejemplo de funcionamiento



Ataques contra un IDS

NIDS

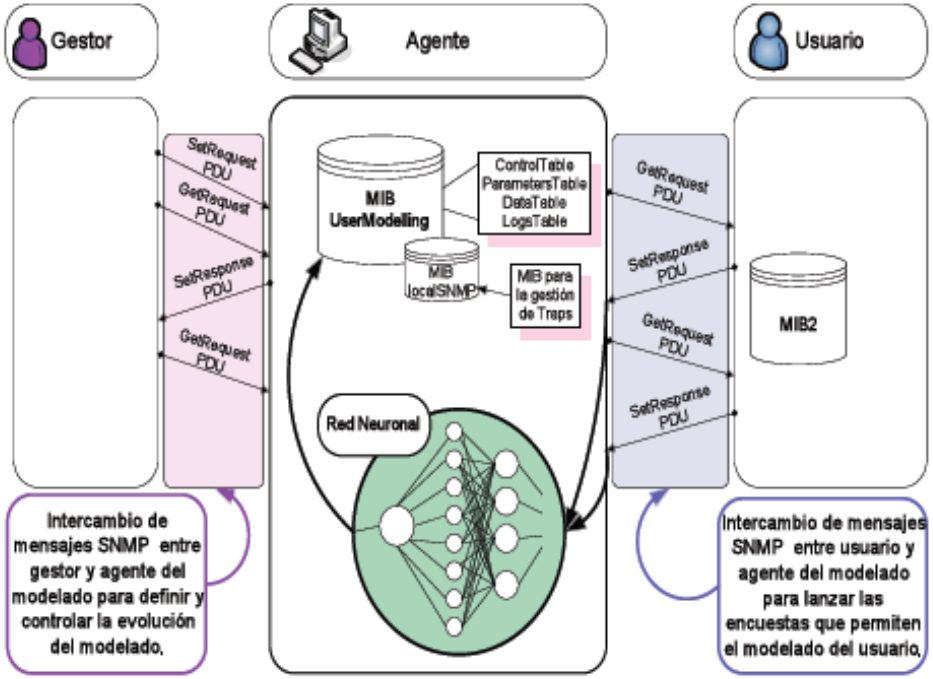
- Saturación de la red
  - “algún paquete se le escapará”
- Detección del NIDS y ataque por DoS
  - Acceso a red susceptible a ataques de red
- Fragmentación de los paquetes IP
  - Persigue saturar el buffer del NIDS: buffer limitado
- Ataques “lentos”
  - Objetivo: expandir el patrón del ataque para que no sea reconocible
  - Ej: port scaner “inteligente”: aleatorizar sondeo de puertos y esperar tiempo entre envío de paquetes

Modelado de comportamiento

- ¿Cómo se comportan nuestros usuarios de manera normal?
  - Uso de red y horario
  - Tipo de tráfico
  - Uso de CPU
  - Uso de memoria
  - Etc.
- Red neuronal que recoge parámetros y modela el comportamiento normal
  - Comportamiento anormal red neuronal avisa

Modelado de comportamiento

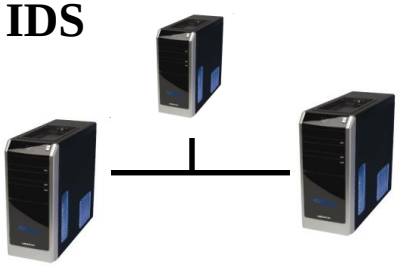
- Modelado de comportamiento
  - Propuesta basada en SNMP



IPS

- IPS: Intrusion Prevention Systems
- Similares a un IDS, pero se ponen entre dos elementos

IDS



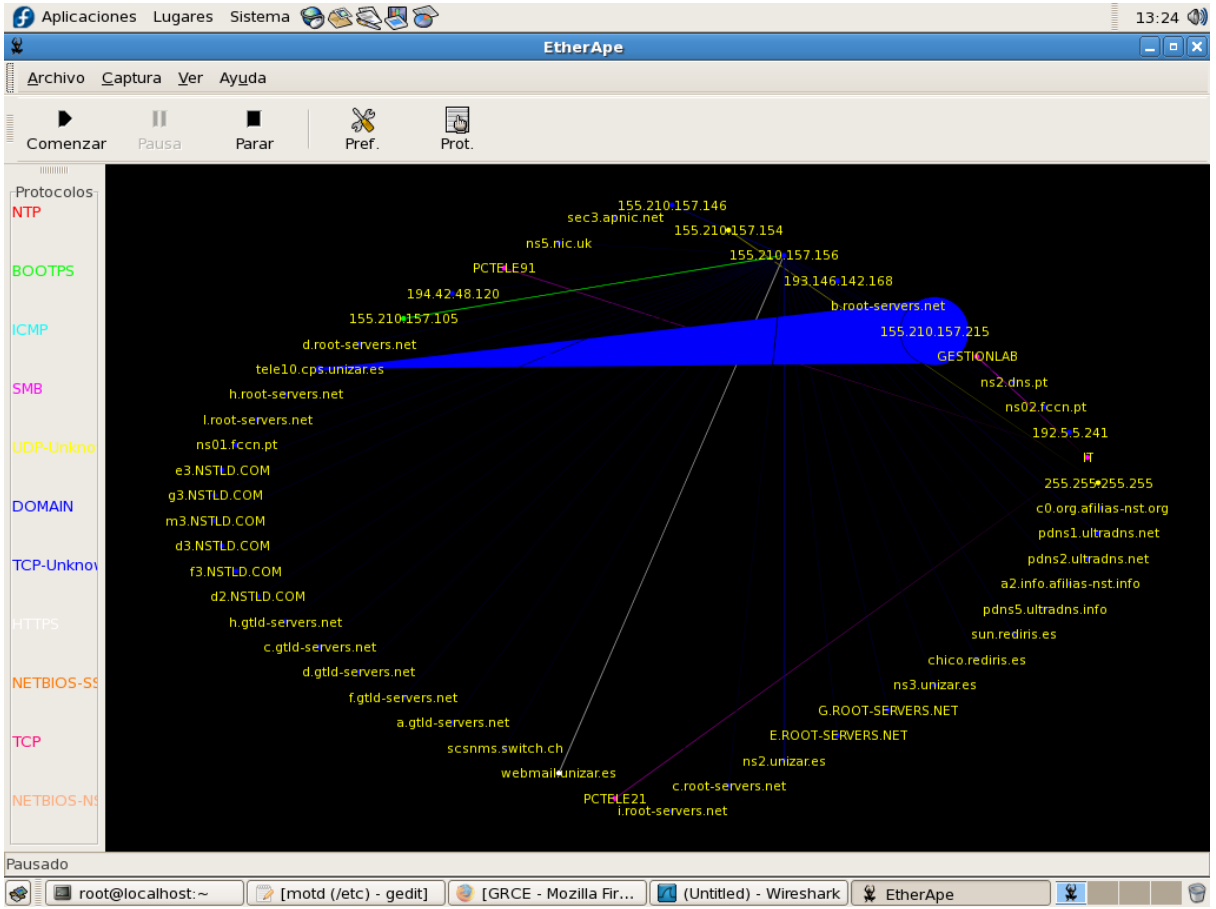
IPS

- Pueden ser HIPS o NIPS
- Son elementos activos (protegen ante ataques)

IPS

- Ventajas:
  - No solo avisan Protegen
- Desventajas:
  - Análisis de tráfico Cuello de botella
    - Hasta que no termina el análisis no transmiten el tráfico (reducción de BW efectivo de la red)

Etherape & tcpreplay



Conclusiones

- IDS = Elemento de seguridad a considerar dentro de una estrategia de seguridad
- Diferentes estrategias emplear la mejor en cada caso (son compatibles)
- Recomendables después de un cortafuegos y una correcta securización de los equipos

Preguntas.

UNIVERSIDAD INTERNACIONAL DE LA RIOJA

unir

www.unir.net