IDS/IPS - Suricata

Suricata (https://oisf.net/) es un IDS/IPS de código abierto.

Características Principales

Procesamiento Multi Hilo

Una de la características más importantes de Suricata que permite la ejecución de varios procesos / subprocesos de forma simultánea. Podemos entonces asignar el número de subprocesos por CPU / Cores y qué subprocesos. De esta forma es capaz, entre otras cosas, de procesar una gran cantidad de paquetes de forma simultánea aumentando así el rendimiento.

Tenemos 4 módulos de subproceso:

- captura de paquetes
- decodificador. Seguimiento del flujo de secuencias y conexiones para luego reensamblar de forma correcta la secuencia original. Inspección de la capa de aplicación.
- detección / comparación de firmas
- procesamiento de eventos y salida de alertas

Podemos configurar suricata de forma que cada CPU pueda dedicarse a un subproceso o módulo.

Detección de Protocolos

A parte de los protocolos IP, TCP, UDP e ICMP, Suricata tiene palabras claves para otros protocolos como FTP, HTTP, TLS, SMB. De esa forma podemos escribir reglas independientemente del puerto que un protocolo use, ya sea por defecto o no ya que éste es automáticamente detectado.

Estadísticas de Rendimiento

Estadísticas y análisis de rendimiento. Estas estadísticas se vuelcan el en archivo /var/log/suricata/stats.log.

Módulo de Logs HTTP

Suricata, independientemente de las alertas, vuelca todas las peticiones HTTP (tanto desde HOME_NET > EXTERNAl_NET como en el sentido contrario) en un archivo /var/log/http.log.

El registro de estas peticiones se almacenan en formato de Log Apache.

Opciones Comunes

- -c para indicar ruta y archivo de configuración
- -i para indicar la interfaz de red
- -r leer fichero .pcap
- -s indicamos ruta y archivo de reglas
- -l ruta de logs de alertas
- -D para ejecutar como un servicio
- –pidfile para indicar PID en modo servicio
- –user / –group usuario y grupo para ejecutar suricata

Configuración

El archivo de configuración se encuentra en /etc/suricata/surucata.yaml.

max-pending-packets

Número de paquetes que es capaz de procesar de forma simultánea. Su configuración dependerá de las características de memoria, CPU/Cores etc. Si tenemos un buen hardware, podemos elevar el rendimiento elevando el número de paquetes a procesar, que usará más memoria y recursos. Por defecto se establece en 50.

action-order

Una regla suricata (compatible con Snort) está compuesta de 3 partes fundamentales:

- Action: que puede ser pass, drop, reject y alert.
- Header: cabecera de la regla
- Rule Options: opciones de la regla

Con action-order establecemos el orden de qué ocurre cuando se establece un coincidencia con una regla establecida. Es decir, que independientemente de cómo suricata cargue los archivos de reglas, estas se procesarán en el orden establecido en esta directiva. Por defecto se establece en: pass, drop, reject, alert

Salida de Eventos (logs)

Aquí establecemos la configuración de la salida de los eventos de alertas.

Si no se establece en línea de comandos, existe una directiva para indicar la carpeta donde se almacenarán las alertas:

default-log-dir: /var/log/suricata/

La forma más básica de archivo de alertas sería:

- fast:
enabled: yes
filename: fast.log
limit: 32

Con fast establecemos una alerta por línea. Habilitamos con enabled, y damos un nombre al archivo con filename. Con append (yes) le decimos que cuando reinicie suricata no se sobreescriba el archivo.

Establecimiento de log para su uso con barnyard:

- unified-log:
enabled: no
filename: unified.log

Podemos establecer un tipo de alerta Unified (para usar con barnyard):

- unified-alert:
enabled: no
filename: unified.alert
limit: 32

O en formato Unified2:

- unified2-alert:
enabled: yes
filename: snort.unified2
limit: 32

Con limit 32 establecemos un límite en MB. para los archivos.

En el archivo http.log se vuelcan todos los Request HTTP. Este tipo de log, que NO son alertas, se establece de la siguiente manera:

- http-log:
enabled: yes
filename: http.log

Tendremos todos los requests en formato Apache-Log.

Stats

Son las estadísticas que ya hemos comentado. Para activarlas y configurarlas:

- stats:
enabled: yes
filename: stats.log
interval: 5
append: yes

Con interval indicamos el tiempo de refresco en segundos para la generación de las estadísticas.

Visualización de eventos

Podemos visualizar los eventos directamente en consola, en fichero o mediante syslog:

- console:
enabled: yes

- file:
enabled: yes
filename: /var/log/suricata.log

- syslog:
enabled: no
facility: local5
format: "[%i] - "

Variables

Tenemos dos tipos: address-groups y port-groups.

address-groups:

HOME_NET: any
EXTERNAL_NET: any
HTTP_SERVERS: "$HOME_NET"
SMTP_SERVERS: "$HOME_NET"
SQL_SERVERS: "$HOME_NET"
DNS_SERVERS: "$HOME_NET"
TELNET_SERVERS: "$HOME_NET"
AIM_SERVERS: any

port-groups:

HTTP_PORTS: "80"

SHELLCODE_PORTS: “!80”
ORACLE_PORTS: 1521
SSH_PORTS: 22

Reglas

Su ubicación se define por la directiva default-rule-path, y luego se escriben los nombres de archivo de las reglas que queremos habilitar, quedando algo así:

default-rule-path: /etc/suricata/rules/

rule-files:
- local.rules
- backdoor.rules
- bad-traffic.rules
- chat.rules
- ddos.rules
- nmap.rules

Además, configuramos también:

classification-file: /etc/suricata/classification.config

reference-config-file: /etc/suricata/reference.config

Formato de Reglas

El lenguaje usado por Suricata (que es el mismo utilizado por Snort) es flexible y potente, basado en una serie de normas que serán las que nos sirvan de guía para la escritura de las reglas.

Dentro de estas normas tenemos:

- la descripción de cada regla
- cabecera
- opciones
- uso de preprocesadores

Las reglas deben ser escritas en una sola línea, de lo contrario habrá que usar el carácter de escape (\):

alert tcp any 110 -> (content: “filename=\”TOMOFONICA.TXT.vbs\””;\

nocase; msg: “Virus tomofonica”;)

La reglas las podemos dividir en dos secciones lógicas, a saber: cabecera de la regla y

opciones:

- La cabecera contiene la acción de la regla en sí, protocolo, IPs, máscaras de red, puertos origen y destino y destino del paquete o dirección de la operación.
- La sección opciones contiene los mensajes y la información necesaria para la decisión a tomar por parte de la alerta en forma de opciones.

Resumiendo lo visto hasta ahora, las reglas las dividiremos de la siguiente manera:

Cabecera

- Acción
- Protocolos involucrados
- Direcciones IP
- Números de puerto
- Dirección de la operación

Opciones

- Mensaje
- Opciones de decisión

Ejemplo 1

Veamos ahora un ejemplo de regla para alertar un escaneo nmap del tipo TCP ping:

alert tcp $EXTERNAL_NET any -> $HOME_NET any / (msg:"Escaneo ping con nmap";flags:A;ack:0; / reference:arachnids,28;classtype:attempted-recon; sid:628;/ rev:1;)

Analicemos esta alerta:

Cabecera

- Acción de la regla: alert
- Protocolo: tcp
- Direccion IP origen: $EXTERNAL_NET
- Puerto IP origen: any (cualquiera)
- Direccion IP destino: $HOME_NET
- Puerto IP destino: any (cualquiera)
- Dirección de la operación: -> (puede ser ->, <-)

Opciones

- Mensaje: msg
- Opciones: flags:A;ack:0; reference:arachnids..(1)

Un poco de teoría:

- flags:A Establece el contenido de los flags o banderas TCP, en este caso ACK
- ack:0 Caso particular para valor ACK=0, es el valor que pone nmap para TCP ping scan.
- reference:arachnids,28 Referencia un a un Advisory, alerta tipo Bugtrack, etc.
- classtype:attempted-recon Categoría de la alerta según unos niveles predefinidos y prioridades
- sid:628 Identificación única para esta regla según unos tramos determinados.
- rev:1 Identificación de la revisión o versión de la regla.

Ejemplo 2

alert tcp any any -> 192.168.1.0/24 111 (content:"|00 01 86 a5|"; msg: "mountd access";)

Acceso al demonio de administración mountd de UNIX, el cual tiene diversos problemas de desbordamiento de memoria intermedia, que permiten a un atacante remoto obtener privilegios de administrador en los sistemas vulnerables.

Dos consideraciones:

1. El orden de la sección opciones. Primero las opciones y después el mensaje. El orden es indiferente.
2. La opción 'content'. Es una de las opciones más importantes ya que nos permite la búsqueda de contenidos dentro del campo datos del paquete IP. Se puede añadir 'nocase' para que la búsqueda de los datos no sea sensible a las mayúsculas. Estos datos pueden estar en formato hexadecimal, texto plano o binario.

Cabecera de las reglas: Acciones

Hemos vistos que la primera parte de la cabecera es la acción de la regla. La acción de la regla indica que se debe hacer cuando detecte un paquete que coincida con el criterio de la regla. Las acciones pueden ser:

- alert genera una alerta usando el método de alerta seleccionado y almacena el log
- log archiva el log del paquete
- pass ignora el paquete

- activate activa la alerta y llama a una regla dinámica
- dynamic cuando es llamada por una regla activate se pone en funcionamiento

Algunos ejemplos:

pass tcp any any -> $HOME_NET any (msg:"all traffic";)

pass ip 10.x.x.0/22 any -> any any (content: "Open Port * 80"; msg: \ "Open Port 80.";)

alert tcp any 110 -> (content: "filename=\"TOMOFONICA.TXT.vbs\""; \ nocase; msg: "Virus tomofonica";)

Autor: Fabian Martinez Portantier

Fuentes:

- https://seguridadyredes.wordpress.com