Sistemes de detecció d'intrusos en xarxa

Joaquín García Alfaro

PID_00191680

Universitat Oberta de Catalunya

Índex

Introducció

L'objectiu d'aquest mòdul és aprofundir els nostres coneixements en l'àrea dels sistemes de detecció d'intrusos (IDS). Com veurem al llarg d'aquest mòdul, aquests sistemes abasten en realitat una varietat enorme de conceptes, tècniques i eines, les quals són sovint difícils de definir de manera aïllada. De fet, fins i tot la utilització del terme *intrús* és de vegades mal interpretada en la literatura. El concepte d'*intrús* és utilitzat en general per a caracteritzar accions comeses per usuaris no autoritzats que violen la política de seguretat i aconsegueixen introduir-se de forma il·lícita en un sistema. Tot i així, els IDS no es limiten a descobrir únicament la fase final d'una entrada no autoritzada en un sistema, sinó que en general abasten també mecanismes capaços de tractar les fases prèvies a una intrusió, i també moltes altres accions hostils contra un sistema i, fins i tot, poden ocupar-se de qualsevol altra tasca que pugui ser considerada potencialment perillosa o amb un comportament fora del que és habitual (respecte al funcionament normal del sistema).

Encara que existeixen, en general, dues grans categories d'IDS en el mercat, segons si el seu procés de detecció se centra directament en el tractament del flux de dades d'una xarxa (coneguts com a NIDS), o si se centra en les tasques executades en el nivell d'equip (coneguts com a HIDS), en aquest mòdul ens centrarem únicament en els IDS en el nivell de xarxa. Veurem també que el procés de detecció es pot resumir en dues modalitats principals. D'una banda, diferenciarem els mecanismes de detecció que efectuen un procés d'anàlisi de dades per tal de detectar usos indeguts (de l'anglès, *misuse-based intrusion detection*), coneguts *a priori*. Aquesta primera tècnica (coneguda en anglès com a *signature-based intrusion detection*) es basa a obtenir un conjunt de patrons o signatures definides explícitament en forma de regles de detecció. La segona categoria (definida en anglès com a *anomaly-based intrusion detection*) es basa en la detecció d'accions considerades com a anormals o fora del que és habitual. Els IDS que utilitzen aquest segon tipus de detecció, es basen en la utilització de processos heurístics i models probabilístics.

Tot seguit, veurem un exemple concret de NIDS mitjançant l'eina de programari lliure Snort. Snort pot ser utilitzat de manera directa tant per operaris de xarxa com per terceres eines que basen la seva captura i registre de paquets en xarxes TCP/IP a través del motor de recollida de Snort. De fet, Snort es pot utilitzar tant per a garantir una vigilància continuada a la recerca d'intents d'intrusió o d'usos indeguts en una simple xarxa local, com per a implementar sistemes de detecció distribuïts. En aquest mòdul, repassarem breument els orígens de l'eina i n'estudiarem de manera general l'arquitectura i la manera de treballar, a més de les possibilitats d'ampliar-ne les característiques de generació d'informes per mitjà d'eines gràfiques addicionals.

Finalment, tractarem la necessitat de funcionalitats addicionals per a completar el procés de detecció d'un NIDS com Snort. Repassarem l'ús de nous elements encarregats de gestionar sistemes i tècniques de detecció heterogènies, amb l'objectiu final de poder controlar i reaccionar apropiadament davant d'escenaris d'intrusió més complexos. Aquests nous elements, generalitzats amb el nom de SIEM, tenen com a objectiu facilitar la gestió de grans volums d'informació generats per eines de detecció, no únicament IDS, sinó també esdeveniments generats a partir de sistemes de tallafocs, escàners de vulnerabilitats, o fins i tot sistemes antivirus. Veurem que aquests elements proporcionaran tècniques necessàries per a normalitzar els esdeveniments rebuts i finalment realitzar tasques de fusió d'informació i correlació d'alertes.

Objectius

Els objectius bàsics que assolireu amb l'estudi d'aquest mòdul són els següents:

1. Comprendre les diferents tècniques de detecció que poden utilitzar els sistemes de detecció d'intrusos i aprendre a classificar aquests sistemes segons diversos criteris, com ara el lloc on es produeix el procés d'anàlisi de les dades o el mecanisme de detecció concret.
2. Veure un exemple concret d'IDS en xarxa mitjançant l'eina de programari lliure Snort.
3. Entendre les limitacions lligades als processos tradicionals d'un sistema de detecció d'intrusos, incloent com a problemàtica la possibilitat de falsos positius i falsos negatius, i la necessitat de complements addicionals per a consolidar el tractament d'incidents detectats.
4. Conèixer l'existència d'eines finals de gestió, com ara gestors d'esdeveniments capaços de normalitzar, fusionar i posar en correspondència alertes recollides de manera distribuïda per sondes de detecció heterogènies.

1. Detecció d'intrusos en xarxa

La instal·lació d'un sistema per a la detecció d'intrusos* pretén millorar la seguretat d'un sistema per mitjà de la incorporació d'eines o dispositius capaços d'avisar els operadors en el moment que es produeixin atacs coneguts contra la seguretat del sistema o desviacions respecte al comportament habitual dels seus usuaris o equips.

*** En anglès, *intrusion detection system* (IDS).**

En realitat, la detecció d'intrusos es basa en la idea implícita de violació de la política de seguretat d'un sistema, fet que comporta un atac parcial o total l'objectiu final del qual és obtenir un accés amb privilegis d'administrador al sistema.

Els mecanismes per a la detecció d'intrusos miren de trobar i reportar activitat maliciosa en el trànsit de xarxa o en l'àmbit del sistema i les aplicacions, de manera que puguin arribar a reaccionar adequadament abans que es produeixi l'objectiu final de l'atac.

En la majoria dels casos és desitjable poder identificar l'atac exacte que s'està produint, de manera que es pugui aturar l'atac i recuperar-se'n. En altres situacions, només serà possible detectar i informar de l'activitat sospitosa que s'ha trobat, davant la impossibilitat de conèixer què ha succeït realment.

Generalment, el procés de detecció treballarà amb la premissa que ens trobem en la pitjor de les situacions, és a dir: que l'atacant ha obtingut un accés al sistema i que és capaç d'utilitzar-ne o modificar-ne els recursos.

A continuació introduirem dues definicions bàsiques en el camp de la detecció d'intrusos, amb l'objectiu d'aclarir els termes comuns que s'utilitzaran més endavant.

Una **intrusió** és una seqüència d'accions realitzades per un usuari o un procés deshonest, amb l'objectiu final de provocar un accés no autoritzat en un equip o un sistema complet.

La intrusió consistirà en la seqüència de passos realitzats per l'atacant que viola una determinada política de seguretat. L'existència d'una política de segure-

tat, en la qual es contemplen una sèrie d'accions deshonestes que cal prevenir, és un requisit clau per a la intrusió. És a dir, la violació només es podrà detectar quan les accions observades es puguin comparar amb el conjunt de regles definides en la política de seguretat.

La **detecció d'intrusos** és el procés d'identificació i resposta davant de les activitats il·lícites observades contra un o diversos recursos d'una xarxa.

Aquesta última definició introdueix la noció de *procés de detecció d'intrusos*, que involucra tot un seguit de tecnologies, usuaris i eines necessàries per a arribar a bon terme.

A l'hora de classificar aquest tipus de sistemes trobem bàsicament dues categories principals, segons la localització des de la qual s'obtenen les dades. Així, parlarem d'IDS en el nivell de xarxa o NIDS* i d'IDS en el nivell d'equip o HIDS**. Els NIDS basen el seu procés de detecció a partir de, per exemple, el flux de dades capturat per mitjà de la interfície de xarxa associada a l'IDS. Els HIDS analitzaran informació d'esdeveniments en el nivell del sistema operatiu (com ara intents de connexió i crides al sistema).

*** De l'anglès,** ***network-based intrusion detection systems.***
**** De l'anglès,** ***host-based intrusion detection systems.***

Necessitat d'instal·lar un NIDS

La millor manera d'entendre la necessitat d'incorporar aquests elements podria ser la comparació entre la seguretat d'una xarxa informàtica i la seguretat d'un edifici: les portes d'entrada exerceixen un primer nivell de control d'accés, però normalment no ens quedem aquí; instal·larem detectors de moviment o càmeres de vigilància en punts clau de l'edifici per tal de detectar l'existència de persones no autoritzades o que fan un mal ús dels recursos i posen en perill la seguretat. A més, hi haurà vigilants de seguretat, llibres de registre en què s'apuntarà tot el personal que accedeix a un determinat departament que considerem crític, etcètera. Tota aquesta informació es processa des d'una oficina de control de seguretat, on se supervisa el registre de les càmeres i es porten els llibres de registre. Tots aquests elements, projectats en el món digital, configuren el que es coneix en l'àmbit de la seguretat de xarxes informàtiques com a *mecanismes de detecció*.

En el nivell de xarxa, un NIDS ha de ser capaç de poder detectar els atacs següents:

- Escàners de vulnerabilitats maliciosos. Un NIDS haurà de ser capaç de detectar activitats malicioses realitzades per eines associades a una eina d'intrusió (*rootkit*), una xarxa de zombis (*botnet*), etc., i que comportin, per exemple, un escaneig il·lícit de ports, cerca de versions antigues de serveis o interrogacions que comporten una denegació de servei.

- Propagació de virus, cucs o temptativa d'instal·lació dels components d'una eina d'intrusió o d'aplicacions troianes.

- Explotació (interna o externa) de vulnerabilitats conegudes en protocols com ara DNS, FTP, HTTP, ICMP, SMTP, POP3, RPC.

- Utilització abusiva de serveis de xarxa, tant per part d'usuaris interns com externs a la xarxa.

- Atacs d'enginyeria social contra aplicacions de missatgeria instantània, xats, P2P, etc. La detecció d'atacs basats en tècniques de pesca (*phishing*) podria ser un exemple dins d'aquesta categoria.

Exemple

Un exemple d'utilització abusiva de serveis de xarxa és la instal·lació il·lícita de servidors de música o de servidors de transferència de fitxers mitjançant tècniques de P2P.

De manera indirecta, un NIDS també podria detectar i reportar incidents associats al mal funcionament de la xarxa, no solament a causa d'atacs o accions malintencionades, sinó simplement a causa d'errors de configuració o errades en els equips informàtics del sistema. Alguns NIDS especialitzats en aquest darrer tipus de detecció han evolucionat cap a sistemes complets per a la detecció de vulnerabilitats en xarxa, coneguts com a VDS*. Els VDS solen incorporar funcions avançades per a l'execució de processos interns d'auditoria d'una xarxa, amb l'objectiu d'identificar i aïllar els components locals (interns) de la xarxa que hagin provocat l'incident.

*** De l'anglès, *vulnerability detection system.***

El tipus de detecció proporcionat per un NIDS es pot classificar en dues categories principals: detecció basada en usos indeguts i detecció basada en anomalies. Presentarem, a continuació, alguns dels detalls més rellevants de les dues categories.

Detecció d'intrusos en xarxes sense fil

A diferència dels sistemes de detecció d'intrusos que supervisen el trànsit de xarxes tradicionals mitjançant signatures d'intrusions, els sistemes de detecció d'intrusos en xarxes sense fil solen oferir tècniques addicionals per a supervisar l'espectre radioelèctric d'aquestes xarxes a la recerca d'accessos no autoritzats o accions anòmales. La utilització d'aquests sistemes és important, ja que pot prevenir els administradors del sistema sobre forats de seguretat provocats per errors de configuració o per una mala interpretació de les polítiques de seguretat de l'organització. Els esdeveniments recollits pels detectors poden evitar accessos indeguts a la xarxa molt abans que es produeixin els incidents. Amb aquests components també es pretén detectar atacs de tipus *man-in-the-middle* i la suplantació d'adreces MAC físiques o indicis de possibles atacs de denegació de servei (DoS) a causa de la saturació del mitjà sense fil utilitzat.

1.1. Detecció basada en usos indeguts

La detecció basada en el model d'usos indeguts compta amb el coneixement *a priori* de seqüències i activitats deshonestes. Els IDS que implementen aquest esquema analitzen els esdeveniments a la recerca de patrons d'atac coneguts o activitats que ataquin vulnerabilitats típiques dels sistemes supervisats.

Aquestes seqüències o patrons es coneixen amb el nom de *signatures d'atacs* i podríem comparar-les amb les signatures víriques que utilitzen els productes actuals de detecció de virus.

Així doncs, els IDS basats en el model d'usos indeguts compararan els esdeveniments recollits amb les signatures d'atac que conserven emmagatzemades en les seves bases de coneixement.

En el moment de detectar la concordança d'algun esdeveniment o seqüència d'esdeveniments amb alguna signatura d'atac, el component llançarà una alarma.

La majoria de NIDS comercials basen la seva detecció en el model d'usos indeguts. Per això, la seva configuració s'acostuma a realitzar mitjançant l'ús de signatures que caracteritzen atacs coneguts i que es posen a l'abast dels operaris d'aquests sistemes de manera automàtica mitjançant una subscripció en línia als proveïdors dels productes. Cada signatura defineix un conjunt d'esdeveniments i condicions que representen l'atac en qüestió. El reconeixement es basa, doncs, en algorismes de tipus "reconeixement de patrons" o *pattern-matching*, amb l'objectiu d'identificar l'empremta de l'atac en el trànsit de xarxa que el NIDS en qüestió supervisa. A continuació, i segons la configuració del NIDS, es podran generar i emmagatzemar un fitxer de registre (*log*) o una alarma amb un nivell semàntic més gran associat a l'atac, per a la seva anàlisi posterior per part de l'operari de la xarxa.

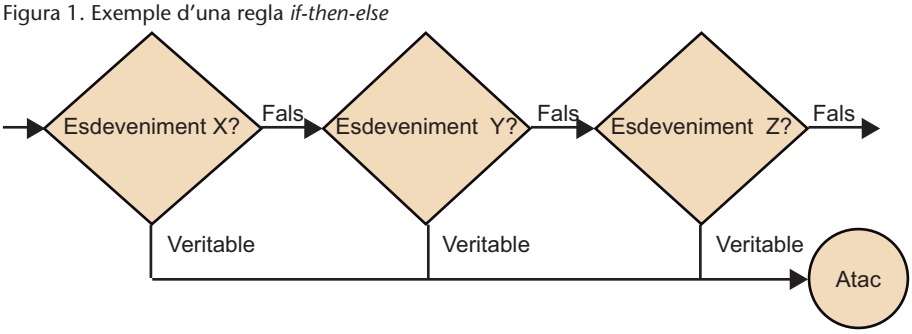
A l'hora d'implementar un esquema de detecció basat en usos indeguts, dues de les tècniques més utilitzades són el reconeixement de patrons i la transicions d'estats:

- **Detecció basada en reconeixement de patrons.** Mitjançant la utilització de regles del tipus *if-then-else* per a examinar les dades, un IDS basat en reconeixement de patrons processarà la informació per mitjà de funcions internes al sistema, de manera completament transparent per a l'usuari. La figura 1 mostra l'esquema d'aquesta mena de regles.

Lectura recomanada

S. Northcutt; J. Novak. (2002). *Network Intrusion Detection* (3a edició). New Riders Publishing.

Figura 1. Exemple d'una regla *if-then-else*

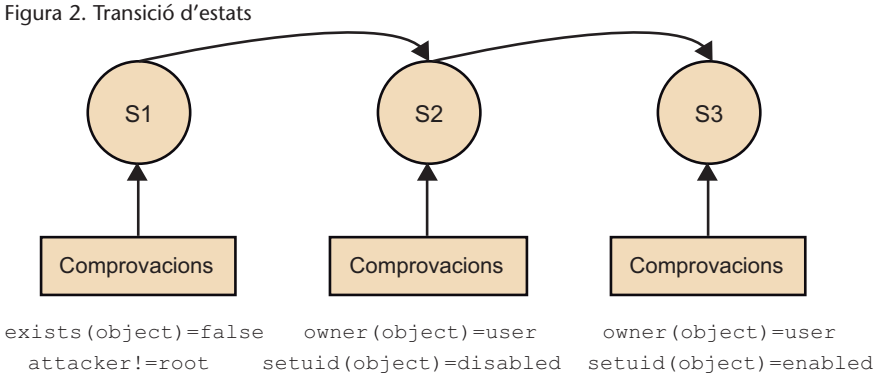


Encara que aquest model permet detectar una intrusió a partir de patrons coneguts *a priori*, el seu inconvenient principal és que els patrons no defineixen un ordre seqüencial de les accions.

Detectar, mitjançant aquest model, atacs formats per una seqüència d'esdeveniments pot arribar a comportar grans dificultats. D'altra banda, el manteniment i l'actualització de la base de dades de patrons són uns altres punts crítics del model.

- **Detecció basada en transició d'estats.** Aquest model fa servir autòmats finits per a representar els atacs, en què els nodes representen els estats, i les fletxes (arcs) les transicions. La figura 2 en mostra un exemple.

Figura 2. Transició d'estats



La utilització de diagrames de transició facilita l'associació entre els estats i els diferents passos que executa un atacant des del moment en què entra en un sistema, amb privilegis limitats, fins que n'obté el control.

Com a avantatges principals d'aquest model podem destacar que els diagrames de transició permeten obtenir una representació a alt nivell d'escenaris d'intrusió i ofereixen així una manera d'identificar una sèrie de seqüències que componen l'atac.

D'altra banda, aquests diagrames defineixen de forma molt senzilla els atacs que s'han de detectar. El motor de detecció d'un IDS basat en aquest model podria arribar a utilitzar diferents variants del mateix diagrama per a identificar atacs similars.

En canvi, els diagrames de transició i, per tant, els diferents passos de la seqüència, s'han de crear mitjançant llenguatges específics que sovint solen ser molt limitats i insuficients per a recrear atacs complexos.

Aquesta limitació impedeix que aquest model pugui detectar alguns dels atacs més comuns, de manera que cal fer servir tècniques mixtes que complementin regles del tipus *if-then-else* i tècniques de transició d'estats.

1.2. Detecció basada en anomalies

Els processadors d'esdeveniments que basen la seva detecció en un esquema d'anomalies intenten identificar activitats sospitoses comparant el comportament d'un usuari, procés o servei, amb el comportament de perfil classificat com a normal.

Un perfil serveix com a mètrica (mesura d'un conjunt de variables) de comportaments normals. Qualsevol desviació que superi un cert llindar respecte al perfil emmagatzemat es tractarà com a evidència d'atac o intrusió.

Un dels requisits d'aquest model és la necessitat d'inicialització d'un perfil per defecte, que s'anirà adaptant progressivament al comportament d'un usuari, procés o servei no sospitós. Cal, per tant, l'ús de processos heurístics i descriptors estadístics que ajudin a modelitzar correctament canvis en el comportament tan bon punt es produeixin. Altres propostes tracten d'incorporar tècniques d'intel·ligència artificial per a executar aquestes tasques (com, per exemple, l'ús de xarxes neuronals o d'algorismes genètics).

La detecció basada en anomalies ofereix uns avantatges ben clars respecte a la detecció basada en usos indeguts. L'avantatge més destacable és la possibilitat de detectar atacs desconeguts. Això és possible perquè, independentment de com l'atacant hagi aconseguit la intrusió en un sistema, tan bon punt les seves activitats es comencin a desviar del comportament d'un usuari normal, el processador d'esdeveniments llançarà una alarma que avisarà sobre una possible intrusió.

Tot i així, l'esquema de detecció basat en anomalies presenta força inconvenients. El primer que hem de destacar és la manca de garantia en el procés de detecció: un intrús podria executar les seves accions de mica en mica per provocar canvis en el perfil d'usuari del processador d'esdeveniments amb la finalitat que la seva presència en el sistema passés desapercebuda.

Inconvenients

Els inconvenients de l'esquema de detecció basats en anomalies fan que la majoria dels sistemes de detecció comercials disponibles en l'actualitat implementin, en general, esquemes basats en el model d'usos indeguts.

Com a segon inconvenient podem destacar la dificultat que apareix a l'hora de classificar i descriure amb precisió els atacs detectats mitjançant analitzadors basats en anomalies. Generalment, un IDS no solament ha de llançar una alarma sinó que ha d'especificar d'on procedeix l'atac, quins canvis ha sofert el sistema, etc.

A més, la taxa de falsos positius i negatius que es pot donar utilitzant aquest esquema de detecció és un gran inconvenient, ja que no sempre una desviació respecte al perfil esperat coincidirà amb un atac o intent d'intrusió. En el cas d'un NIDS, és possible que el nombre d'alarmes llançades (en una xarxa de

mida mitjana) superi fàcilment el centenar. Això provoca que els administradors de la xarxa sovint acabin ignorant les alarmes llançades pel sistema de detecció o, fins i tot, desactivant completament el sistema.

Falsos positius i falsos negatius

Un fals positiu succeeix en aquelles situacions en què el NIDS caracteritza com a maliciós trànsit legítim, que no forma part de cap atac; és, per tant, un esdeveniment detectat per equivocació. Per contra, un fals negatiu es produeix en aquelles situacions en què trànsit maliciós és descartat i se'l considera trànsit legítim per equivocació; és, per tant, un esdeveniment que hauria de ser detectat, però que escapa al procés de detecció.

2. Detecció d'intrusos en xarxa amb Snort

Snort és una eina de seguretat molt completa basada en codi obert per a la creació de sistemes de detecció d'intrusos en entorns de xarxa. És molt popular entre la comunitat d'administradors de xarxes i serveis. Gràcies a la seva capacitat per a la captura i registre de paquets en xarxes TCP/IP, Snort es pot utilitzar per a implementar des d'un simple detector (*sniffer*) de paquets per al monitoratge del trànsit d'una petita xarxa fins a un sistema complet de detecció d'intrusos en temps real.

Vegeu també

En l'apartat 3 d'aquest mòdul podeu veure alguns exemples pràctics sobre Snort.

Com a monitor de xarxa, Snort es comporta com una autèntica aspiradora (d'aquí el seu nom) de datagrames IP, que ofereix diferents possibilitats quant al seu tractament. Des d'actuar com un simple monitor de xarxa passiu que s'encarrega de detectar el trànsit maligne que circula per la xarxa fins a la possibilitat d'enviar a servidors de fitxers de registre o servidors de base de dades tot el trànsit capturat.

Però, a banda d'unes característiques excel·lents com a detector de paquets i generador d'alertes, Snort té moltes altres característiques que han permès que es converteixi en una de les solucions de programari més completes per a la construcció de sistemes de detecció en entorns de xarxa basats en reconeixement de patrons. Snort s'autodefineix com un NIDS lleuger*. Aquest qualificatiu de *lleuger* significa que, com a IDS, el seu disseny i implementació li permeten poder funcionar sota diferents sistemes operatius i que les seves funcions com a mecanisme de detecció podran formar part de diferents productes de seguretat (fins i tot comercials).

*** De l'anglès, *lightweight network intrusion detection system*.**

La popularitat de Snort s'ha incrementat aquests darrers anys en paral·lel a l'increment de popularitat de sistemes operatius de codi obert, com pot ser la família de sistemes GNU/Linux i BSD (NetBSD, OpenBSD, FreeBSD i Mac OS X).

Ara bé, la seva naturalesa com a producte de codi obert no el limita a estar disponible únicament per a aquest tipus de sistemes operatius. Snort pot funcionar sota solucions comercials com, per exemple, Microsoft Windows.

Des del punt de vista del seu motor de detecció, Snort formaria part de la categoria de detecció basada en usos indeguts. Mitjançant un reconeixement de signatures, Snort contrastarà tot el trànsit capturat en les seves regles de detecció.

Una regla de detecció de Snort no és res més que un conjunt de requisits que li permetran activar una alarma, si es compleixen. Un exemple seria una regla de

Snort que permetria verificar l'ús d'aplicacions P2P per a l'intercanvi de fitxers a través d'Internet, tot verificant l'ús de la cadena `GET` en serveis diferents al port tradicional del protocol HTTP. Si un paquet capturat per Snort coincideix amb aquesta senzilla regla, el seu sistema de notificació llançarà una alerta per a indicar els fets. Una vegada llançada l'alerta, es pot emmagatzemar de diferents maneres i amb diferents formats, com ara un únic fitxer de registre del sistema, una entrada en una base de dades d'alertes, un esdeveniment SNMP, etc.

A continuació veurem els orígens de Snort i una anàlisi de la seva arquitectura i algunes de les seves característiques més destacables.

Versió comercial de Snort

Tot i que Snort està disponible sota llicència GPL (GNU Public License), hi ha productes comercials basats directament en Snort i distribuïts per l'empresa Sourcefire, fundada pel creador de Snort, Marty Roesch. L'anàlisi d'aquestes versions comercials queda fora del propòsit d'aquest mòdul didàctic. Per a més informació, podeu visitar http://sourcefire.com.

2.1. Origen de Snort

De manera molt resumida, podem definir Snort com un detector (*sniffer*) de paquets amb funcionalitats addicionals per al registre de paquets i la generació d'alertes i un motor de detecció basat en usos indeguts.

Snort va ser desenvolupat en 1998 amb el nom d'APE. El seu desenvolupador, Marty Roesch, intentava implementar un detector de paquets multiplataforma (encara que el desenvolupament inicial es va fer per al sistema operatiu GNU/Linux) que comptés amb diferents opcions de classificació i visualització dels paquets capturats. Marty Roesch va implementar Snort com una aplicació basada en la biblioteca `libcap` (per al desenvolupament de la captura de paquets) la qual cosa garantia una gran portabilitat tant en la captura com en el format del trànsit recollit.

Una mica més que un detector

L'autor de Snort mirava d'indicar amb el nom Snort que la seva aplicació era alguna cosa més que un detector. En anglès, el terme *snort* significa una acció d'inhalar o d'esnifar de forma més obsessiva i violenta. A més, Marty va dir en el seu moment que ja tenia massa aplicacions anomenades `a.out` i que tots els noms populars per a detectors, anomenats TCP-something, ja estaven ocupats.

Snort es va començar a distribuir per mitjà del lloc web *Packet Storm** el 22 de desembre de 1998, només amb mil sis-centes línies de codi i un total de dos fitxers font. En aquella època, l'ús principal que li va donar el seu autor era d'analitzador de les seves connexions de xarxa a través d'un cable mòdem i com a depurador de les aplicacions de xarxa que estava implementant.

*** http://www.packetstormsecurity.com**

El primer analitzador de signatures desenvolupat per a Snort (també conegut com a analitzador de regles per la comunitat de desenvolupament de Snort) es va afegir com una nova funcionalitat de l'aplicació el gener de 1999. Aquesta nova funcionalitat va permetre que Snort comencés a utilitzar-se com a detector d'intrusions.

El desembre de 1999 va aparèixer la versió 1.5 de Snort. En aquesta versió, el seu autor va decidir una nova arquitectura basada en connectors (*plug-ins*) que encara es conserva en les versions actuals. Després d'aquesta versió, Marty Roesch va abandonar l'empresa on treballava i va començar a dedicar-se a temps complet a la tasca d'afegir noves funcionalitats que milloressin les capacitats de configuració i facilitessin l'ús de Snort en entorns més professionals. Gràcies a la gran acceptació que obtenia el seu NIDS entre la comunitat d'administradors, Marty va pensar que era un bon moment per a oferir el seu producte amb un suport per a empreses i va obtenir el finançament necessari per a fundar Sourcefire.

Enllaç d'interès

En el lloc web http://sourcefire.com trobareu més informació sobre *Sourcefire*.

No obstant això, Snort continua sent codi lliure i promet seguir sent-ho per sempre. L'última versió disponible* de Snort es presenta amb més de 75.000 línies de codi i una reestructuració total pel que fa al disseny original de la seva arquitectura inicial.

*** Versió 2.9 en el moment d'escriure aquest mòdul didàctic.**

Tot i que el suport i desenvolupament actual de Snort es fa des Sourcefire de forma comercial, hi ha la versió lliure sota llicència GNU. Aquesta versió es pot descarregar lliurement des de la web d'Snort** i permet que qualsevol usuari pugui disposar de suport per a les últimes versions disponibles i les últimes actualitzacions dels fitxers de regles per a aquestes versions.

**** http://www.snort.org**

Actualment, Snort disposa d'un gran repertori d'accessoris que permeten reportar notificacions en diferents gestors de bases de dades (com ara MySQL i PostgreSQL) i un gran nombre de preprocessadors de trànsit que permet analitzar crides RPC i escaneig de ports abans que aquests es constrastin amb el conjunt de regles associat a la recerca de nous incidents.

Els conjunts de regles de Snort també han anat evolucionant a mesura que l'aplicació creixia. La grandària dels conjunts de regles per a l'última versió Snort disponibles per a descàrrega s'incrementa de manera similar a la velocitat d'aparició de nous *exploits*. Aquests fitxers de regles es troben actualment classificats en diferents categories com, per exemple, P2P, atacs de denegació de servei, atacs contra serveis web, virus, trànsit pornogràfic, etc.

Cadascuna d'aquestes regles està associada a un identificador únic (sensor ID, SID) que permet reconèixer i trobar informació sobre l'atac o mal ús detectat. Per exemple, el SID per a l'atac `SSH banner attack` és el `1838`. A més, gràcies a l'ús majoritari de Snort entre la comunitat d'administradors de xarxa, altres NIDS han adoptat el format de les regles de Snort i també la codificació utilitzada per als bolcats dels paquets capturats (basada en el `libcap`).

El suport d'aquests fitxers de regles augmenta cada dia. D'aquesta manera, qualsevol usuari de Snort, o de qualsevol altre NIDS amb un format de regles compatible, podria crear les seves regles pròpies a mesura que anessin apareixent nous atacs i col·laborar amb la comunitat de desenvolupament de Snort per a mantenir perfectament actualitzada la seva base de signatures.

2.2. Arquitectura de Snort

Snort proporciona un conjunt de característiques que en fan una eina de seguretat molt potent, entre les quals destaquen la captura del trànsit de xarxa, l'anàlisi i el registre dels paquets capturats i la detecció de trànsit maliciós o deshonest. Abans de veure amb més detall les característiques de Snort, és important conèixer-ne i comprendre'n l'arquitectura.

Snort està format per un conjunt de components, la majoria desenvolupats com a connectors que permeten la personalització de Snort. Entre aquests components destaquen els preprocessadors, que permeten que Snort manipuli de manera més eficient el contingut dels paquets abans de passar-lo a l'element de detecció, i el seu sistema de notificacions, que permeten que la informació reportada es pugui enviar i emmagatzemar en diferents formats i seguint diferents mètodes.

Connectors de Snort

El terme *connector* (*plug-in*) fa referència a mòduls de programari d'una aplicació desenvolupats de forma independent del nucli general de l'aplicació. L'objectiu és afegir funcionalitats addicionals sense necessitat d'afectar el codi font del nucli o la resta de components. Per a això, l'aplicació ha de proporcionar una interfície de programació o API (*application programming interface*) que permeti el desenvolupament i la compilació d'aquesta mena de complements. Així doncs, un connector Snort és un component desenvolupat conforme a l'API de connectors de Snort, que s'utilitzarà al costat del nucli del codi de Snort, però separat, de manera que un canvi en el codi del component no n'afecti el nucli o els altres components.

L'arquitectura central de Snort es basa en els quatre components següents:

- Detector de paquets
- Preprocessador
- Motor de detecció
- Sistema de notificacions

D'acord amb aquesta estructura, Snort permetrà la captura i el preprocessament del trànsit de la xarxa mitjançant els dos primers components (detector de paquets i preprocessador) i posteriorment els comprovarà per mitjà del motor de detecció (segons el conjunt de regles activades) i generarà, per part de l'últim dels components, les notificacions pertinents.

La figura 3 mostra l'arquitectura bàsica de Snort que acabem de comentar. Observant la figura podem fer un símil entre Snort i una màquina mecànica per a l'ordenació automàtica de monedes:

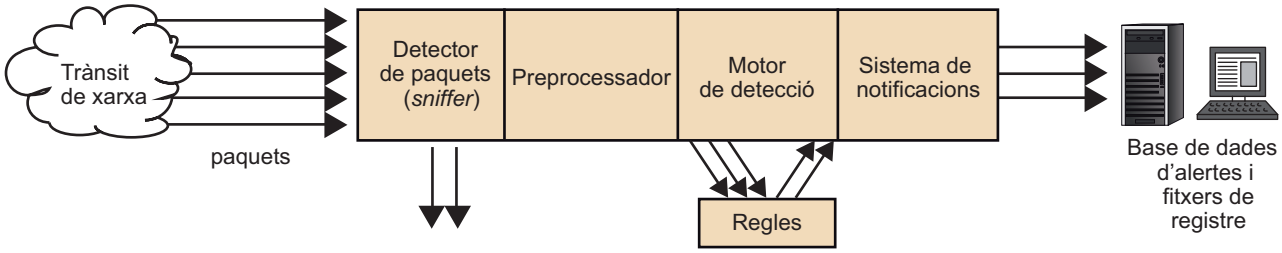
1) Pren totes les monedes (paquets de la xarxa recollits pel detector de paquets).

2) Cada moneda es deixarà caure per una rampa per a determinar a quin grup de monedes pertany (preprocessador de paquets).

3) Ordena les monedes segons cada tipus de moneda i les empaqueta en forma de canons segons la categoria (motor de detecció).

4) Finalment, l'administrador decidirà què fer amb cada un dels canons de monedes ordenades (sistema de notificacions).

Figura 3. Arquitectura bàsica de Snort



Tant el preprocessador com el motor de detecció i el sistema de notificacions de Snort són també implementats en forma de components independents. Tot seguit examinarem amb més detall cada un dels components bàsics de Snort que acabem de veure.

2.3. Detector de paquets

Un detector de paquets (*sniffer*) és un dispositiu (de *programari* o de *maquinari*) que es fa servir per a capturar els paquets que viatgen per la xarxa a la qual està associat.

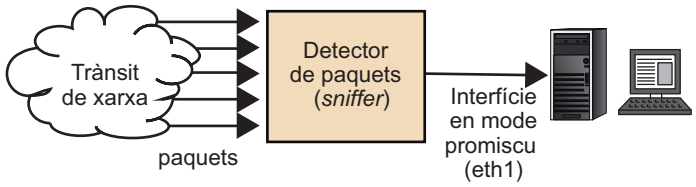
En el cas de xarxes TCP/IP, aquest trànsit acostuma a ser trànsit de datagrames IP, tot i que també és possible l'existència de trànsit de diferents tipus, com ara trànsit IPX o trànsit AppleTalk. A més, atès que el trànsit IP consisteix també en diferents tipus de protocols, com TCP, UDP, ICMP, protocols d'encaminament, IPSec, etc., molts detectors necessitaran conèixer *a priori* el tipus de trànsit per a poder interpretar més endavant els paquets que es van recollint i poder mostrar-los en un llenguatge comprensible per un administrador de xarxa.

Com moltes altres eines relacionades amb la seguretat en xarxes, els detectors es poden fer servir amb objectius més o menys deshonestos. Entre els diferents usos que es poden donar a un detector, podem pensar en anàlisis de trànsit per a la solució de congestions i problemes de xarxa, millora i estudi del rendiment dels recursos, captura passiva d'informació sensible (contrasenyes, noms d'usuari, etc.).

D'aquesta manera, igual que la resta de detectors tradicionals, el descodificador de paquets de Snort serà l'element encarregat de recollir els paquets, que

la resta de components examinaran i classificaran més endavant. Per a això, el detector de paquets ha de ser capaç de capturar tot el trànsit que pugui, per passar-lo després al següent component (el preprocessador), que s'encarregarà de detectar quin tipus de trànsit s'ha recollit. La figura 4 mostra un esquema del funcionament del detector de paquets de Snort.

Figura 4. Funcionament del detector de paquets de Snort



Interfície de xarxa en mode promiscu

El mode promiscu és el mode en què la targeta de xarxa d'un equip connectat a una xarxa (sia xarxa cablejada o xarxa sense fil) permetrà la captura de tot el trànsit que circuli per aquesta interfície de xarxa.

2.4. Preprocessador

A mesura que el detector de paquets de Snort va recollint el trànsit que passa per la xarxa, l'anirà lliurant a l'element de preprocessament. Aquest element anirà adaptant els paquets capturats i els lliurarà al motor de detecció.

Així doncs, el preprocessador obtindrà paquets sense tractar (*raw paquets*) i els verificarà mitjançant un conjunt de connectors. Per exemple, utilitzarà un connector per al tractament de paquets relacionats amb trànsit de tipus RPC* o un connector per al tractament de paquets relacionats amb escanejos de ports. Aquests connectors verificaran els paquets a la recerca de determinats comportaments que permetin que Snort en determini el tipus. Un cop determinat el tipus, el paquet s'enviarà cap al motor de detecció.

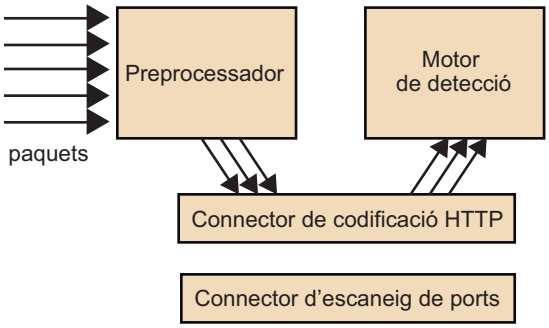
* Del inglés, *remote procedure calls.*

Aquesta característica de preprocessament és realment important per a una eina de detecció, ja que és possible la utilització de terceres aplicacions que es poden activar i desactivar segons les necessitats del nivell de preprocessament. Per exemple, si a un administrador de xarxa no el preocupa el trànsit RPC que entra i surt de la seva xarxa (i no necessita, per tant, analitzar-lo), pel motiu que sigui, en tindrà prou a desactivar el connector de RPC i seguir utilitzant la resta.

La figura 5 mostra un esquema on el preprocessador de Snort utilitza dos dels seus connectors per a verificar el tipus de paquets que rep i decidir si cal passar-los al motor de detecció o no.

Els preprocessadors de Snort ofereixen una gran flexibilitat per a la implementació de diferents algorismes de tractament de trànsit a Snort.

Figura 5. Preprocessador de Snort que combina dos connectors de tractament de dades



2.5. Regles de detecció

Com ja avançàvem, Snort basa la seva detecció en el model d'usos indeguts. Per aquesta raó, Snort s'ha de configurar mitjançant un conjunt de regles que utilitzarà el mòdul de detecció per a efectuar el reconeixement d'atacs i signatures d'intrusió. Les regles de Snort s'agrupen en general en conjunts de signatures que categoritzen els incidents. Així, trobarem conjunts de regles associades a la detecció de troians, a la detecció d'atacs de desbordament de memòria intermèdia (*buffer overflow*), etc.

Cada regla es pot dividir en dues parts. En primer lloc, tenim la capçalera de la regla, en què indiquem l'acció associada a aquesta regla en cas de complir-se (generació d'un fitxer de registre o generació d'una alerta), el tipus de paquet (TCP, UDP, ICMP, etc.), l'adreça d'origen i destinació del paquet, etc. En segon lloc, tenim el camp `option` de la regla, on trobarem la informació que ha de contenir el paquet (a la part de dades, per exemple) perquè s'activi l'acció associada a la regla.

Per a indicar el contingut d'aquestes dues parts, Snort té una sintaxi pròpia que permet especificar fins al més mínim detall les condicions que s'han de complir perquè un paquet sigui associat a les accions indicades per cada una de les regles. A tall d'exemple, el format general d'una regla Snort és el següent (l'ús dels símbols '[' i ']' indica que els atributs en la regla són opcionals):

```
<Accio> <Protocol>
<IP_origen> <Port_origen> -> <IP_destinacio> <Port_destinacio>
[(<opcio_1>; ...; <opcio_k>)]
```

Exemple

La regla següent indicaria a Snort que generés alertes del tipus *FTP root access attempt* en veure trànsit de tipus TCP amb la cadena "USER root", procedent d'una adreça IP amb màscara de xarxa 10.0.0.1/24, amb qualsevol port d'origen i amb destinació a alguna adreça IP dins de la xarxa 192.168.1.0/24:

```
alert tcp
any any -> 192.168.1.0/24 21
(content: 'USER root'; msg: 'FTP root user access attempt';)
```

Un exemple més complet de regla Snort és el següent:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP EXPLOIT STAT *
dos attempt"; flow:to_server,established; content:"STAT "; nocase;
content:"*"; reference:bugtraq,4482; classtype:attempted-dos;
sid:1777; rev:1;)
```

L'acció definida en una regla de Snort es pot escollir entre cinc accions bàsiques:

- **Alert**. Generació d'una alerta que a més conté la informació de registre corresponent al paquet que activi la regla.

- **Log**. Generació únicament de la informació de registre associada al contingut del paquet associat a l'activació de la regla.

- **Pass**. Deixa passar el paquet associat a la regla sense necessitat de reportar (*log*) o alertar sobre el fet.

- **Activate**. Generació d'una alerta juntament amb l'activació d'una regla dinàmica (vegeu l'acció *Dynamic*).

- **Dynamic**. Regla que roman inactiva i que s'activa per mitjà de l'acció *activate* per tal de, per exemple, activar la generació de registres associats als paquets associats a la regla inicial amb l'objectiu d'obtenir informació addicional respecte al trànsit posterior al paquet que va activar la regla (i amb una durada de temps determinada).

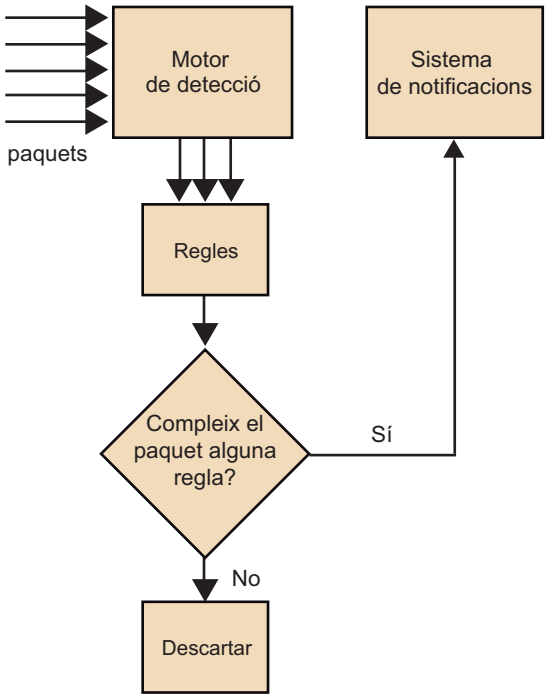
Accions bàsiques

Versions ampliades de Snort poden proporcionar accions addicionals, com ara accions reactives per a bloquejar el trànsit detectat. Vegeu, per exemple, les accions addicionals de Snort Inline, una versió reactiva de Snort disponible en la adreça d'internet següent: http://snortinline.sourceforge.net. El projecte no es manté actualment, però és possible accedir a les últimes versions del projecte que van ser alliberades fins a mitjan 2008 i que oferien les accions addicionals (*drop*, *sdrop* i *reject*). Aquestes accions permeten transformar Snort en un IDS reactiu (IPS, *intrusion prevention system*, sistema de prevenció d'intrusions).

2.6. Motor de detecció

El motor de detecció conté els algorismes de tractament necessaris per a concloure el procés de detecció. A partir de la informació proporcionada pel preprocessador i els seus connectors associats, el motor de detecció contrastarà aquestes dades amb la base de regles definida per l'operador. Si alguna de les regles coincideix amb la informació obtinguda, el motor de detecció s'encarregarà d'avisar el sistema de notificacions, indicant la regla que ha saltat. La figura 6 mostra un esquema senzill sobre el comportament general del motor de detecció de Snort.

Figura 6. Motor de detecció de Snort



De tots els elements que hem vist, el motor de detecció i la sintaxi utilitzada per les regles de detecció són les parts més complicades i que costa més comprendre a l'hora d'estudiar el comportament de Snort. Tot i així, una vegada ens posem a treballar amb Snort i haguem après mínimament la sintaxi utilitzada, és bastant senzill arribar a personalitzar i ajustar el comportament de la funcionalitat de detecció de Snort. A més a més, els conjunts de regles es poden activar o desactivar amb facilitat, de manera que es pot definir el comportament de detecció desitjat segons el tipus de xarxa en què es configurarà Snort.

Nota

Al final d'aquest mòdul trobareu un conjunt d'activitats per a aprofundir els vostres coneixements sobre la configuració d'Snort.

2.7. Sistema de notificacions

Una vegada que la informació capturada pel descodificador de paquets de Snort és analitzada pel motor de detecció, els resultats s'han de reportar d'alguna manera. Mitjançant aquest component serà possible realitzar aquesta funció i els resultats es podran generar en diferents formats i cap a diferents equips.

Quan el motor de detecció llança una alerta, el resultat pot implicar la generació d'un fitxer de registre, l'enviament de l'alerta a través de la xarxa mitjançant un missatge SNMP o fins i tot l'emmagatzematge de les informacions associades a l'alarma de forma estructurada per algun sistema gestor de base de dades com ara MySQL o PostgreSQL.

A més, és possible emprar eines alternatives (desenvolupades per terceres parts) que facilitin la visualització i el tractament de la informació reportada per

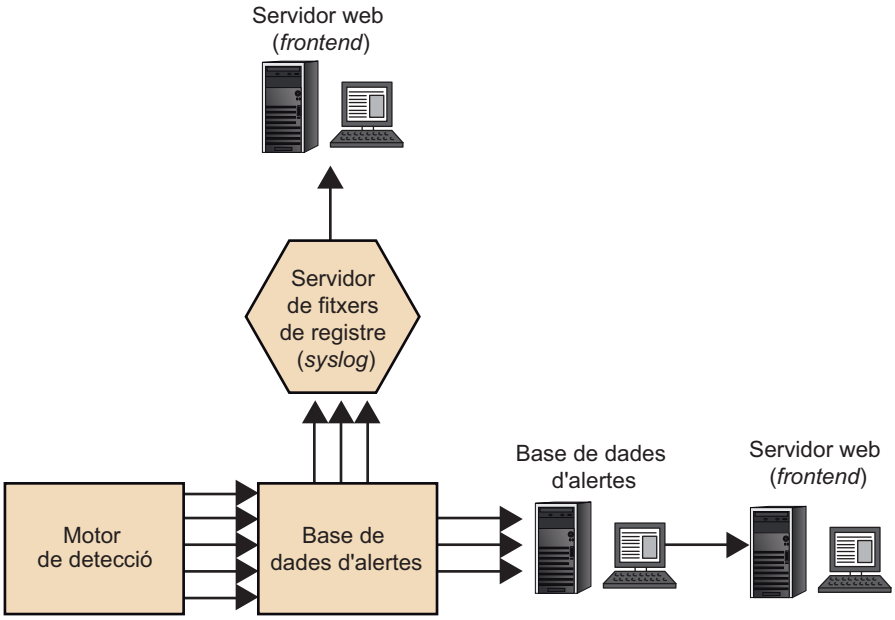
Snort. Moltes d'aquestes eines estan disponibles i es poden descarregar lliurament d'Internet.

Com en el cas del motor de detecció i el preprocessador, el sistema de notificacions de Snort també utilitza un esquema de connectors per al tractament de la informació. La figura 7 mostra un esquema del funcionament d'aquest sistema de notificacions mitjançant connectors.

Alertes de Snort

Si intenteu posar en marxa Snort en una xarxa congestionada i amb un nombre raonable de signatures de detecció activades, la quantitat d'alertes llançades per Snort pot arribar a un centenar en poc temps. Aquest volum enorme d'informació no serà fàcil de tractar amb la utilització d'un simple navegador de fitxers de registre.

Figura 7. Sistema de notificacions de Snort, mitjançant connectors



El propòsit d'instal·lar Snort en una xarxa no és solament obtenir informació (intents d'intrusió), sinó analitzar aquesta informació i poder prendre les accions necessàries en funció de les dades obtingudes. Si el nombre de regles activades és alt i el trànsit de la xarxa augmenta amb facilitat, no serà gaire senzill analitzar la informació reportada per Snort a menys que utilitzem eines de visualització adequades.

D'altra banda, l'aspecte interessant d'un sistema de detecció com Snort no és simplement registrar esdeveniments, sinó ser capaç de reaccionar als intents d'intrusió en un període de temps raonablement curt. Així doncs, caldrà utilitzar segones aplicacions que ajudin a consolidar i analitzar la informació reportada per Snort amb l'objectiu d'alertar els administradors de la xarxa dels intents d'intrusió analitzats.

Actualment hi ha un gran nombre d'utilitats per a treballar amb les informacions generades per Snort. Algunes d'aquestes eines són mantingudes per la pròpia comunitat de desenvolupadors de Snort, encara que també podem trobar aplicacions desenvolupades per terceres parts o aplicacions comercials.

Vegeu també

Algunes de les solucions existents per a reportar la informació recollida per Snort de manera gràfica s'estudien en el subapartat 3.6. d'aquest mòdul.

3. Gestió d'esdeveniments, alertes i incidents

Els sistemes de detecció d'intrusos tradicionals com ara Snort permeten la detecció d'accions elementals que solen correspondre a etapes prèvies a una intrusió o accions aïllades que formen part d'atacs més complexos. En realitat, per a poder anticipar-se i aturar una intrusió, caldrà utilitzar elements addicionals per tal de completar i concloure el procés de detecció realitzat pels IDS tradicionals.

En aquest apartat tractarem aquests complements, que es poden resumir amb el concepte de sistema SIEM*. Igual que els IDS, els sistemes SIEM són components imprescindibles per tal de garantir la seguretat d'una xarxa informàtica. Tot i que existeixen des de fa més de dues dècades, ha estat l'expansió d'Internet a corporacions i institucions allò que ha propiciat les evolucions i millores recents que resumim a continuació:

* De l'anglès, *security information and event management system.*

Atacs distribuïts

Atacs que no es poden identificar buscant patrons de forma aïllada, sinó que s'han de detectar a partir de la combinació de múltiples indicis trobats en diferents equips de la xarxa.

- **Recollida i normalització d'esdeveniments**. Un SIEM ha de ser capaç de gestionar la recollida dels esdeveniments provinents de fonts extremadament heterogènies. Per tant, cal dur a terme un procés de normalització de dades, no solament a nivell sintàctic, sinó també a nivell semàntic. A nivell sintàctic, l'existència de formats i estàndards, com el format IDMEF** o el format IODEF***, pot ajudar en les tasques de normalització sintàctica d'alertes provinents de sondes de detecció heterogènies. Quant a la normalització semàntica, la representació ontològica de les alertes lidera la majoria de treballs actuals en aquesta matèria.

- **Consolidació de les funcions de supervisió d'eines de detecció**. La segona funció vital atribuïda a un SIEM és la consolidació d'alertes de baix nivell produïdes pels components de seguretat d'una xarxa, entre els quals destaquen sistemes tallafoc, IDS, antivirus i sistemes de detecció de vulnerabilitats. Com hem vist per al cas dels IDS, les eines de seguretat en xarxa són vulnerables a problemes lligats amb falsos positius i falsos negatius. Per això, allò que s'espera de la utilització d'un SIEM és que endegui processos de fusió, agregació i correlació d'alertes procedents dels equips anteriors, i reduir la taxa de falsos positius, a més de millorar el diagnòstic per tal de reduir també els falsos negatius.

- **Activació de la reacció**. En general, la majoria d'IDS solen ser configurats com a mecanismes purament passius. No obstant això, la majoria de solucions existents avui en dia proporcionen la tecnologia necessària per a convertir-los en solucions actives o semiactives (a l'espera de confirmació d'un operador abans d'activar el procés de reacció), amb l'objectiu de

** De l'anglès, *intrusion detection message exchange format.*
*** De l'anglès, *incident object description and exchange format.*

poder reaccionar i neutralitzar les activitats o accions detectades. Aquesta funcionalitat, de vegades desconeguda, s'ha d'analitzar i activar amb precaució, ja que el procés de detecció, tot i que millorat gràcies a les tasques de normalització i correlació d'un SIEM, no és infal·lible. Per tant, una configuració automàtica per a reaccionar davant del procés de detecció en el cas de falsos positius, podria dur a situacions indesitjades com ara el bloqueig d'usuaris legítims d'un sistema o la desactivació de serveis per error. La tendència actual és deixar l'activació d'aquesta funcionalitat a partir de la interfície d'un SIEM, atesa la problemàtica de falsos positius durant el procés de detecció.

Tot seguit detallarem algunes de les tasques necessàries per a poder realitzar les funcionalitats que s'esperen d'un SIEM.

3.1. Configuració de les sondes de detecció

A banda d'informació provinent d'un IDS, les dades recollides per un SIEM poden provenir de qualsevol altre component amb capacitat de creació de fitxers de registre, com ara:

- **Sistemes tallafoc** (o qualsevol altre tipus de component per al filtratge de paquets per mitjà de llistes de control d'accés en l'àmbit de xarxa). Encara que sovint hem vist aquests components com a sistemes de protecció l'objectiu del qual és bloquejar trànsit considerat perillós per al sistema, aquests components poden ser configurats per a reportar la informació observada a través de les seves interfícies de xarxa. Els seus fitxers d'auditoria són, per tant, de gran interès per a complementar el procés d'agregació i correlació d'esdeveniments del SIEM.

- **Servidors de correu electrònic** (basats en protocols com ara SMTP, POP o IMAP). Novament, els fitxers d'auditoria generats pels servidors oferiran una gran quantitat d'informació per a caracteritzar i descobrir activitats malicioses, com ara la propagació de cucs, atacs de tipus dia zero *zero-day* o trànsit de control associat a activitats de xarxes de zombis.

- **Servidors de gestió de trànsit** (basats, per exemple, en el protocol SNMP). La informació reportada pels components contindrà també informació associada a accions relacionades amb la violació de polítiques de seguretat interna de l'organització on estiguin instal·lats.

Una altra informació que cal tenir present durant la configuració de les sondes de detecció d'un SIEM seran les dades, els esdeveniments i les alarmes proporcionats per un IPS*. Un IPS es basa principalment en la investigació de vulnerabilitats dels equips i sistemes. La majoria de solucions de tipus IPS combinen en realitat tècniques de detecció d'intrusos amb mecanismes de control d'accés tradicionals. La frontera entre IDS i IPS és actualment difícil de definir i es

*** De l'anglès, *intrusion prevention system*.**

Vegeu també

Els mecanismes de control d'accés tradicionals s'estudien en el mòdul "Sistemes de tallafocs" d'aquesta assignatura.

pot complementar amb moltes altres solucions preventives, com ara sistemes de detecció de vulnerabilitats** i sistemes antivirus.

** En anglès, *vulnerability detection systems* o VDS.

Els VDS tracten d'analitzar la configuració de sistemes desplegats en xarxa, amb l'objectiu de descobrir parts mal configurades que potencialment presenten vulnerabilitats que podrien ser objectiu d'atacs. Aquests sistemes inclouen també detecció de defectes programari (és a dir, errors de programació o *bugs*), errors de concepció en la configuració topològica d'una xarxa, errors de maquinari, etc. D'altra banda, els sistemes antivirus estan dissenyats per a protegir estacions de treball i servidors contra programari maliciós (*malware*) conegut. La majoria d'aquests sistemes utilitzaran una base de dades de signatures d'antivirus que identifica el programari maliciós conegut. Des del punt de vista de prevenció és d'esperar que tant un VDS com un antivirus sigui capaç de corregir les vulnerabilitats i de desinfectar els equips víctimes del programari malición. Així doncs, la instal·lació i combinació d'IDS amb IPS, VDS i sistemes antivirus té com a objectiu final detectar i reaccionar davant del concepte general d'intrusió.

3.2. Polítiques de recollida d'informació

Atès que l'objectiu d'un SIEM és poder oferir als operaris del sistema capacitats de gestió centralitzada, és habitual que la configuració dels components de recollida associats al SIEM (IDS i els components d'exemple anteriors) es faci per mitjà de la interfície d'usuari del SIEM. És per això que se sol utilitzar una política de recollida d'informació global, administrada pel SIEM, i que posteriorment es podrà refirnar per a la configuració local de cada un dels equips associats al SIEM (IDS, tallafoc, servidors de correu, IPS, VD, etc.).

Aquesta política, o les seves subpolítiques associades, se sol definir a partir de l'ús de les regles basades en l'ús d'expressions regulars, cerca de patrons en trànsit, senyalització de protocols, etc. També és possible que els components incorporin la possibilitat de tractar una recollida d'esdeveniments mitjançant configuracions i basada en el reconeixement d'activitats anòmales. En conseqüència, la política ha d'oferir la sintaxi i la semàntica necessàries per a poder tractar l'ús de models estadístics, recollida per mitjà de mineria de dades, sistemes experts, xarxes bayesianes, etc. Finalment, la política de recollida d'esdeveniments i informació de registre del SIEM haurà de permetre també un conjunt d'accions, bé individuals (per exemple, una regla per al reconeixement d'un esdeveniment per part d'un component específic), bé accions generals (per exemple, una regla per al reconeixement d'accions que poden afectar diversos components de detecció, sense especificar el component en concret que haurà de realitzar la tasca de recollida).

Alguns altres dels criteris tècnics que caldrà considerar en la política de recollida d'esdeveniments d'un SIEM serà determinar amb precisió la localització dels components de recollida que s'han de controlar en el sistema. Com veu-

rem més endavant, aquest aspecte determinarà la cobertura i els mòduls de visió global que el sistema incorporarà per tal de garantir que el procés posterior d'agregació i correlació d'informació garanteixi una gestió de qualitat. Per això també caldrà especificar dins de les polítiques de recollida la topologia física i la descripció lògica del sistema que el SIEM ha de supervisar. És important poder tractar i estructurar aquest sistema en termes de subxarxes, de manera que es puguin identificar les parts del sistema que tenen una necessitat més gran de vigilància i així les funcions de correlació puguin reconèixer les tasques associades a incidents específics per cada subxarxa del sistema. Per exemple, si el sistema conté zones de tipus DMZ (zones desmilitaritzades), ha de ser possible crear regles específiques per a vigilar de més a prop les accions que es realitzin en aquesta part de la xarxa. L'ús de màscares de subxarxa (que s'especifica en termes de CIDR* o encaminament interdomini sense classes, per exemple) sol ser habitual en les polítiques de recollida de fitxers de registre de la majoria de SIEM.

*** Sigla de *classless inter-domain routing.***

3.3. Normalització de la informació recollida

Un cop recollida la informació, i abans de passar als processos subsegüents per a la detecció d'escenaris d'atac, el SIEM ha de garantir que és possible posar en correspondència totes les dades que provinguin de la detecció d'un mateix esdeveniment (*a priori*, esdeveniments maliciosos, sospitosos o anòmals). Així doncs, haurà de disposar d'un conjunt de processos de normalització per a preprocessar les dades recollides i anticipar-se als problemes que podrien entorpir la posada en correspondència de dades relatives a esdeveniments derivats d'un mateix flux de trànsit de xarxa (com ara l'origen del trànsit, destinació, ports, etc.) i que corresponen a les tasques de supervisió de diferents sondes de monitorització que van ser instal·lades en diferents llocs del sistema (ja siguin sondes desplegades en un mateix domini o en diferents dominis).

El procés de normalització ha de garantir, per exemple, que dades associades amb el trànsit de xarxa tinguin el mateix format pel que fa a la classificació de l'origen del trànsit i també metadades associades a l'instant de temps en què es va detectar l'esdeveniment, protocols i serveis associats, adreces d'origen i de destinació, contingut dels paquets associats al trànsit, etc. Com que la naturalesa de les sondes és potencialment heterogènia, el procés de normalització ha de garantir, bé amb formats de propietat del SIEM, bé amb esforços provinents d'estàndards existents, que no hi hagi problemes d'interoperabilitat que limitin l'expressivitat de les informacions que el mòdul de correlació del SIEM haurà d'agregar.

Exemples d'esforços

Alguns exemples d'esforços quant al problema de normalització, són els següents:

- CIDF (*common intrusion detection framework*)
- IDWG (*intrusion detection working group*)
- IDMEF (*intrusion detection message exchange format*)
- INCH (*extended incident handling*)
- FINE (*format for incident report exchange*)

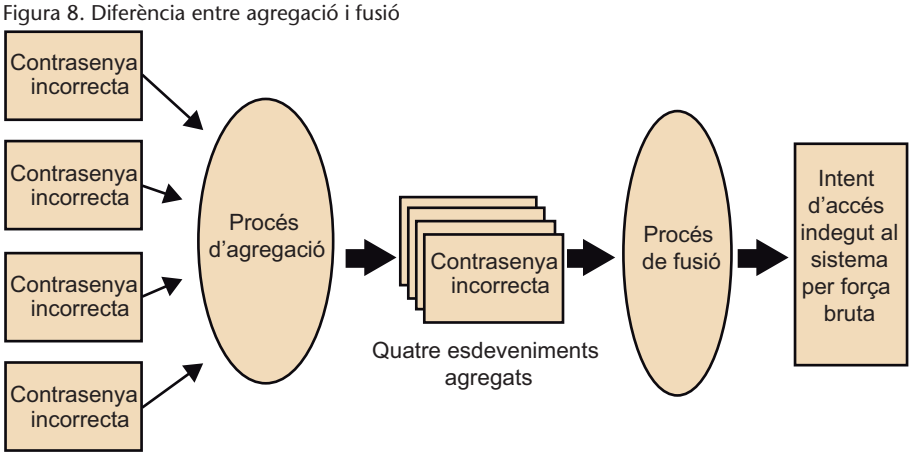
- IDIP (*intrusion detection and isolation protocol*)
- OSEC (*open security evaluation criteria*)
- IODEF (*incident object description and exchange format*)

Cada un dels exemples anteriors ha intentat definir llenguatges comuns per a especificar la descripció dels esdeveniments i les activitats associades als esdeveniments que s'han d'intercanviar els components de seguretat. A part de llenguatges (tant sintàctics com semàntics) per a garantir homogeneïtat en l'intercanvi de dades, és també prioritari permetre la descripció d'un procés comú que especifiqui el protocol precís per a l'intercanvi de dades entre les diferents sondes configurades en un SIEM. Dels exemples enumerats anteriorment, dos dels formats i procediments que compten amb el suport de grups de treball del Internet Engineering Task Force (IETF) son l'IDMEF i l'IODEF. L'IODEF és el format més recent i proporciona compatibilitat amb formats anteriors (per exemple, compatibilitat amb informacions normalitzades en el format IDMEF); a més, s'espera que en el futur sigui el format utilitzat per SIEM i components de seguretat en general per a l'intercanvi d'informació d'incidències.

3.4. Agregació i fusió de la informació

Les funcions d'agregació i fusió de dades són utilitzades per a reduir de forma intel·ligent grans volums de dades que, probablement, poden contenir esdeveniments redundants (repeticions i informacions congruents). Totes dues funcions han de ser aplicades abans de posar en correspondència esdeveniments detectats per diferents components del sistema. En primer lloc, el procés d'agregació s'haurà d'encarregar d'agrupar les dades que s'hagin produït a partir de la detecció d'un mateix esdeveniment, reportat per una mateixa sonda o per sondes diferents. Un cop produïda l'agrupació d'aquestes informacions, es produirà el procés de fusió de la informació, amb l'objectiu de resumir i oferir un única dada que caracteritzi l'esdeveniment detectat. La figura 8 mostra amb un exemple senzill la diferència entre les funcions d'agregació i de fusió.

Figura 8. Diferència entre agregació i fusió



Durant el procés d'agregació, les informacions corresponents a esdeveniments detectats s'agruparan en sessions i s'intentarà unificar les dades d'un mateix esdeveniment que puguin ser utilitzades més endavant durant el procés de fusió, com ara l'adreça d'origen, l'adreça de destinació, els ports, els protocols, etc. D'aquesta manera, les diferents dades associades a un atac cap a un element específic del sistema s'agruparan en una sola sessió i un únic identificador. La resta de dades reportades per altres sondes del sistema s'enllaçaran amb la mateixa referència, de manera que durant el procés final de fusió sigui possible la generació d'una alerta per identificador. Aquesta alerta contindrà tota la informació observada per les diferents sondes configurades pel SIEM. Les alertes generades a partir del procés de fusió contindran, per tant, una síntesi de tot el coneixement del SIEM sobre cada un dels atacs bàsics observats pel sistema de supervisió. Com a resultat, es redueix la quantitat de dades necessària que cal emmagatzemar en el sistema sense que es produeixi cap pèrdua informació. Un cop fusionada tota la informació reportada pel sistema, les alertes es comunicaran a l'últim component del SIEM, encarregat de gestionar i posar en correspondència (correlacionar) el flux d'alertes.

3.5. Correlació d'alertes

De manera general, podem definir el procés de correlació d'alertes com la interpretació conceptual de múltiples alertes amb l'objectiu de proporcionar una millora semàntica i de reduir la quantitat global d'alarmes en un sistema de detecció d'intrusos.

Lectura recomanada

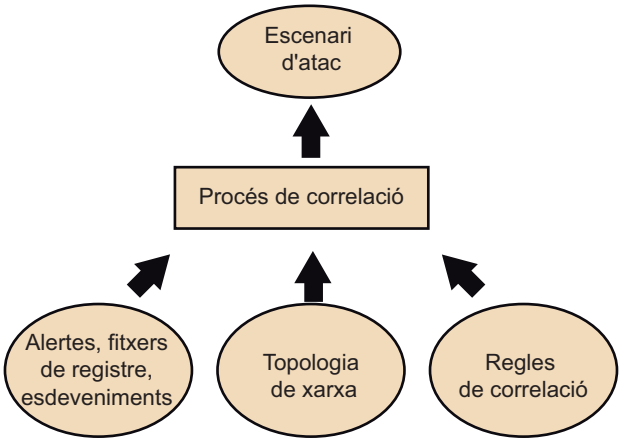
C. Kruegel; F. Valeur; G. Vigna (2004). *Intrusion Detection and Correlation: Challenges and Solutions* (1a ed.). Springer.

La correlació d'alertes és considerada, doncs, una de les claus en l'evolució dels sistemes de detecció d'intrusos, ja que tracta de solucionar els inconvenients més rellevants d'aquests sistemes (és a dir, l'excés d'alertes), millorar-los semànticament i disminuir els falsos positius i negatius. Els treballs relacionats amb la correlació d'alertes en el camp de la detecció d'intrusos són relativament recents; la majoria tracten sobre observacions i experiments realitzats en sistemes actuals. La part teòrica en aquest camp encara està, doncs, en procés de consolidació.

La majoria de SIEM comercials no tenen encara autèntiques funcionalitats per a la realització d'una correlació d'alertes completa. Encara que la major part de solucions existents parlen de serveis de correlació, la gran majoria es limita a emmagatzemar enllaços lògics entre alertes emmagatzemades en una mateixa base de dades relacional que posteriorment pot ser consultada per l'administrador responsable mitjançant una consola de control. En canvi, dins del camp d'investigació acadèmica sobre sistemes de detecció d'intrusos hi ha un gran nombre de propostes per a la inclusió de tècniques de correlació d'alertes en sistemes de nova generació. La major part d'aquestes propostes exploten essencialment la informació continguda en l'interior de les alertes i, addicional-

ment, tracten de fer referències explícites a coneixements annexos, necessaris per al procés final de la posada en correspondència de tots els esdeveniments observats en un mateix sistema. Per tant, la correlació en aquests sistemes no es limita únicament a la informació continguda en l'interior d'alertes generades per les sondes del SIEM, sinó que també fan ús d'un coneixement *a priori* de l'estat de vigilància dels sistemes, sobre els atacs que es poden realitzar, i fins i tot sobre la topologia del sistema i les regles de correlació generades pels operaris i interpretades per sistemes experts. La figura 9 resumeix de manera general aquest tipus de sistemes de correlació.

Figura 9. Funció de correlació



Tal com veiem en la figura 9, la primera entrada en el procés de correlació correspon als esdeveniments ja agregats i fusionats pels mòduls inferiors del SIEM. L'objectiu del procés de correlació és precisament posar en correspondència aquestes agrupacions d'esdeveniments i aconseguir reconstruir escenaris d'atac als quals podrien pertànyer les accions observades. Per a això, caldrà combinar les alertes rebudes amb informacions físiques i lògiques del sistema (per exemple, la topologia del sistema i l'estructuració de l'adreçament IP dels equips) i el coneixement predefinit d'activitats malicioses. Aquest últim es representa en la figura 9 en forma de regles de correlació.

Les propietats físiques i lògiques de la xarxa han de ser especificades en una base de dades topològica, bé de forma manual (pels administradors del sistema), bé mitjançant l'ús de serveis automàtics de descobriment i creació assistida de topologies de xarxa. Aquests últims s'hauran d'encarregar de la creació de mapes topològics amb les configuracions dels dispositius i les polítiques de seguretat associades al sistema d'informació protegit.

La posada en coneixement de les activitats malicioses s'ha de configurar en el sistema mitjançant regles de correlació. Cada regla de correlació es definirà per caracteritzar conjunts d'accions que corresponen a un mateix escenari d'intrusió, i especificarà les condicions necessàries per a dur a terme una acció i les conseqüències en el sistema després de l'execució de cada acció. Tal com passa amb les regles de detecció d'un IDS basat en usos indeguts, aquestes

regles de correlació seran l'element bàsic per a garantir la detecció d'escenaris d'atac formats per les accions detectades en el sistema. Així doncs, per a garantir l'eficàcia del procés de correlació, tindrà una importància cabdal el fet de garantir una configuració correcta de les regles de correlació pertinents per a cada sistema, igual que la seva correcta actualització i posada en funcionament. L'eliminació o corrupció d'una sola regla que contingui informació sobre múltiples incidents podrà ocasionar la pèrdua de detecció d'un gran nombre d'escenaris. La configuració d'aquesta part del SIEM serà, per tant, molt propensa a errors i requereix un coneixement profund per part dels operadors encarregats de configurar el sistema.

La detecció d'un incident o escenari d'intrusió es deriva durant el procés de correlació a partir de les sèries d'esdeveniments indicats mitjançant el conjunt de regles de correlació preconfigurades en el sistema. A partir d'un petit conjunt de regles de correlació, és possible la definició de multitud d'escenaris d'intrusió. Així doncs, l'objectiu del procés de correlació és reduir l'excés d'informació que haurà de ser gestionada per l'operador del sistema. En lloc de sollicitar a l'administrador l'anàlisi de milers d'esdeveniments, el procés de correlació proporciona la generació d'informes d'incidents. Cada informe contindrà la representació dels escenaris que, amb una probabilitat alta, es podrien haver desenvolupat en el sistema a partir dels esdeveniments (és a dir, accions primitives) detectats per les sondes del SIEM.

Amb el mateix objectiu, el procés de correlació es pot configurar per a reduir també el nombre de falses alarmes que s'analitzen (falsos positius). Per a això, el sistema es pot configurar per a executar accions de verificació internes desencadenades després de la generació de cada escenari d'intrusió i verificar la certesa que aquest incident hagi tingut lloc en el sistema o no. D'aquesta manera, els incidents que es puguin descartar amb una probabilitat alta, s'eliminaran de l'informe final que l'operador haurà d'analitzar. Aquesta verificació interna es pot limitar, per exemple, a l'execució d'una anàlisi de vulnerabilitats del sistema, que serveixi per a decidir si un incident concret es pot descartar, si les vulnerabilitats associades no són presents en el sistema representat en la base de dades topològica del sistema; és a dir, un incident que comporti l'explotació de vulnerabilitats o serveis no desplegats en el sistema es podrà descartar amb una alta probabilitat i evitar, així, sobrecarregar les capacitats d'anàlisi de l'operador del sistema. Addicionalment, aquests incidents poden ser reportats com a falsos positius en lloc d'alarmes o escenaris d'intrusió. Tot això facilitarà l'ordenació dels mecanismes de resposta en el sistema i també l'optimització de les contramesures que caldrà desplegar en el sistema sobre els incidents reals que hagin estat detectats.

3.6. Generació d'informes i interfície gràfica de control

L'element més característic d'un SIEM és el component gràfic que s'ofereix per a gestionar la generació d'informes i proporcionar el control d'ordres a l'usua-

ri final. En aquest aspecte, en l'actualitat hi ha moltes solucions, tant gratuïtes com comercials. En aquest subapartat comentarem algunes eines d'exemple, la majoria de les quals encarregades d'oferir una interfície gràfica per a solucions de seguretat tradicionals, com és el cas del NIDS Snort.

Vegeu també

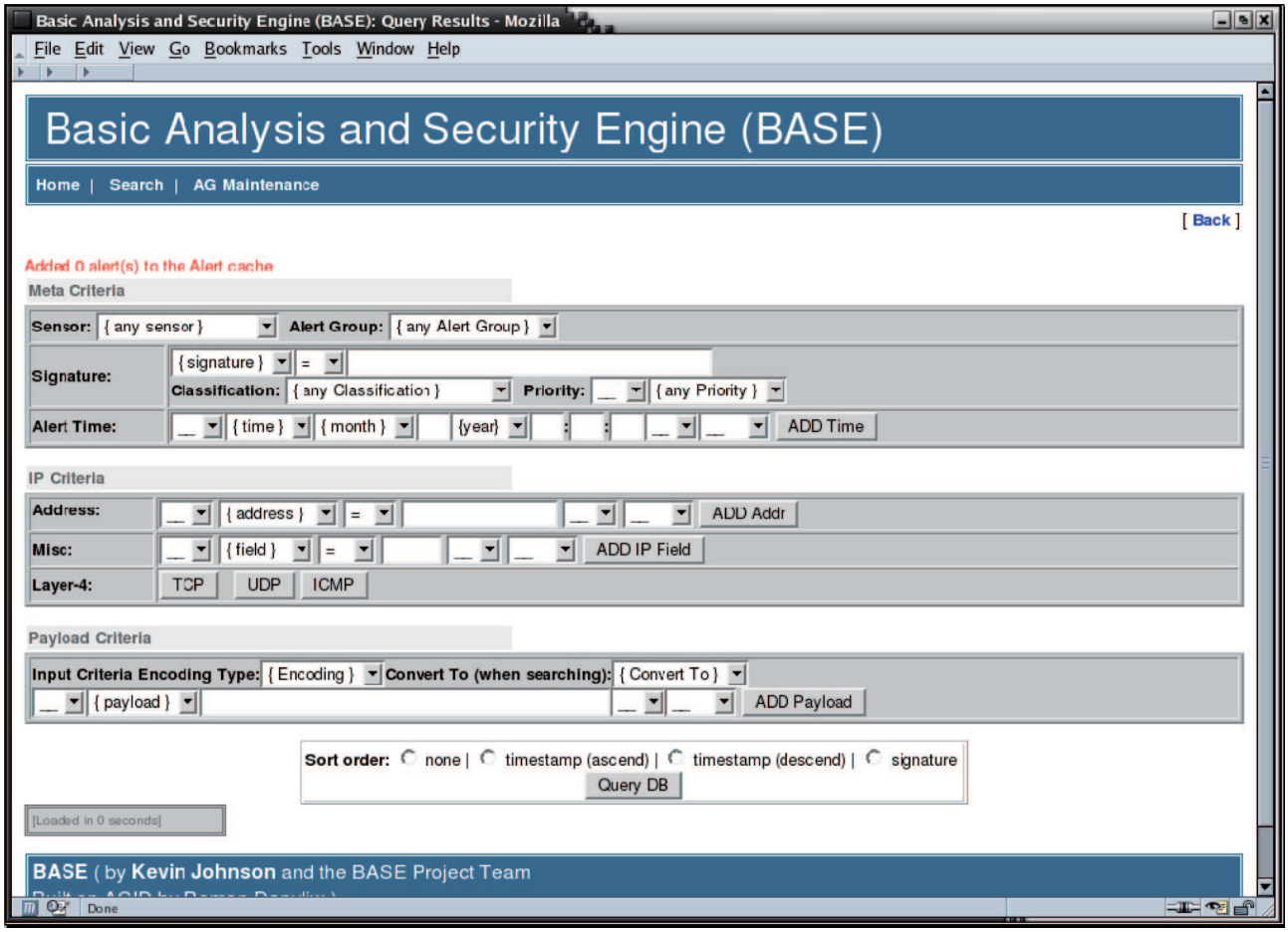
Snort s'estudia en l'apartat 2 d'aquest mòdul.

3.6.1. BASE

Basic Analysis and Security Engine (BASE)* proporciona una interfície web completa per a gestionar alertes i informes d'activitat maliciosa basada en sondes Snort. Es tracta d'un projecte gratuït i de codi obert, que al seu torn es basa en un projecte anterior (ACID, Analysis Console for Intrusion Database) desenvolupat dins del projecte AIRCERT del centre de coordinació CERT de la Carnegie Mellon. BASE es pot utilitzar també per a configurar de manera gràfica diferents sondes de detecció basades en Snort, sobre una única xarxa o diferents xarxes. La figura 10 mostra una captura de pantalla de la interfície principal de BASE per a la recerca d'alertes generades mitjançant les diferents sondes.

* http://base.secureideas.net

Figura 10. Interfície per a la recerca d'alertes mitjançant BASE



El conjunt d'eines associades amb BASE proporciona també tot un conjunt de *scripts* escrits en llenguatge PHP per a gestionar la base de dades on les sondes emmagatzemaran les alertes. La interfície i la base de dades de BASE també es poden utilitzar per a emmagatzemar informació independent respecte a les dades reportades per sondes Snort. Per exemple, és possible combinar en la mateixa base de dades informació reportada per mitjà de tallafocs basats en Netfilter o missatges de control d'accés generats per productes de seguretat Cisco. Altres característiques que podem destacar són les següents:

- Descodificació i visualització de paquets TCP/IP associats (reportats) a les alertes o informes reportats per les sondes de detecció.

- Creació de diagrames i estadístiques basades en dates, hores, signatures, protocols, etc.

- Interfície per a la consolidació de cerques i creació de vistes amb múltiples consultes. Els resultats d'aquestes accions es retornen de forma estructurada amb informació per a facilitar la comprensió de les alertes llançades per les sondes configurades gràcies a BASE. En aquesta informació es ressaltaran, entre altres informacions, les adreces d'origen/destinació, els ports d'origen/destinació, l'estat de les marques TCP/IP (*TCP/IP flags*), etc., associades amb els paquets detectats per les sondes.

- Gestió d'incidents. Es proporciona la possibilitat de crear grups d'alertes lògiques on emmagatzemar la informació dels incidents que es vulguin destacar. També hi ha opcions de maneig de múltiples incidents que permeten descartar les alertes considerades com a falsos positius i l'exportació final a través de correu electrònic o emmagatzematge de les alertes trobades en la base de dades.

Altres productes equivalents a BASE que proporcionen una manera similar de representar gràficament les alertes i els incidents reportats per eines de tipus Snort són RazorBack*, Snorby** i Engage Security IDScenter***.

* http://www.intersectalliance.com/projects/RazorBack

** http://www.snorby.org

*** http://www.engagesecurity.com/products/idscenter

3.6.2. Sourcefire 3D System

El sistema Sourcefire 3D és una família de productes comercials que es poden combinar per a configurar, gestionar i visualitzar el sistema de notificacions de la versió comercial de l'IDS Snort (Sourcefire). Altres funcionalitats són la gestió de signatures criptogràfiques, necessàries per a la construcció i el monitoratge de xarxes privades virtuals (VPN) per mitjà de Sourcefire. Entre els elements de Sourcefire 3D, destaquem els següents:

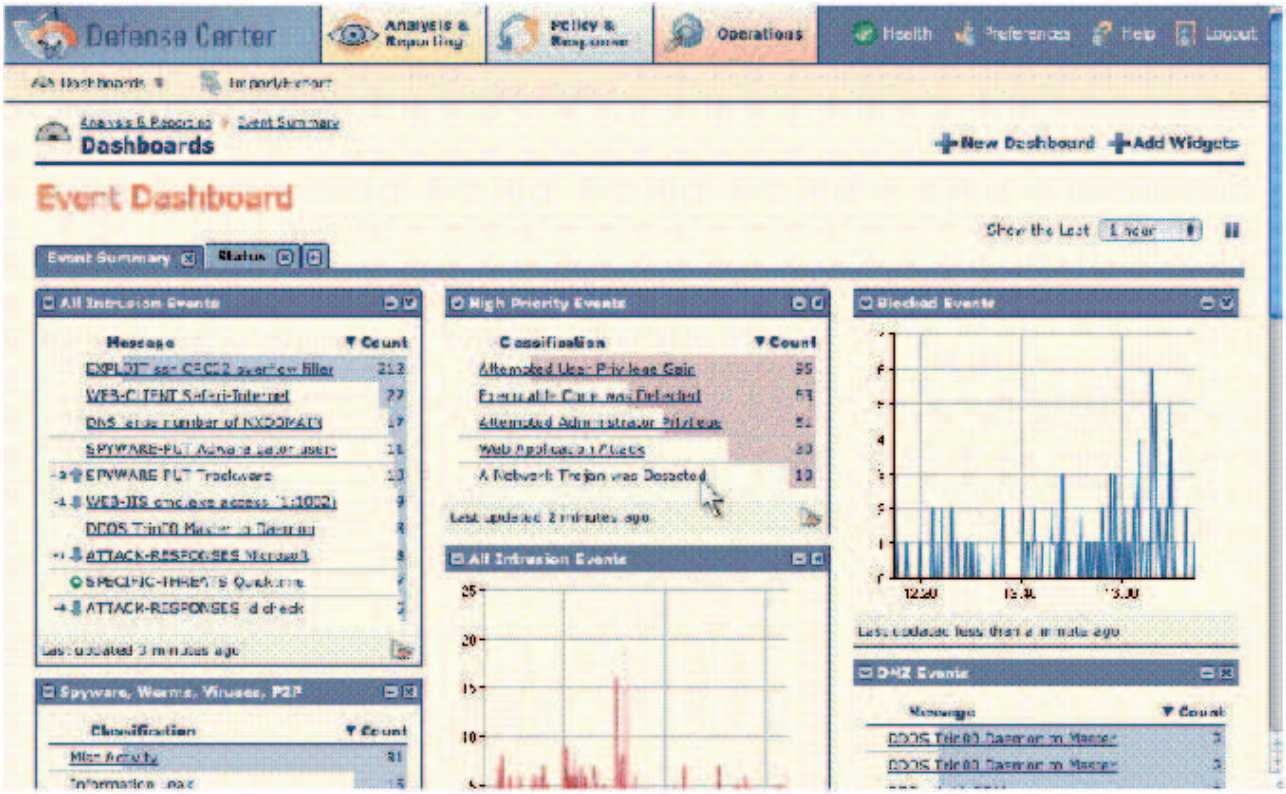
- Sensors 3D IPS, que proporcionen capacitats reactives (tipus IPS) per a configurar regles de configuració que permetin contrarestar els atacs detectats a partir de les sondes de Sourcefire.

- Centre de defensa (DC, *defense center*), que proporciona una utilitat gràfica per a gestionar les funcions d'agregació d'alertes, gestió d'informes i tractament de polítiques de seguretat distribuïdes. La figura 11 mostra una captura de pantalla del tauler de control associat al DC de Sourcefire.

- Real-time Network Awareness (RNA), que ofereix un control de la supervisió passiva de Sourcefire i també una gestió d'inventari en temps real dels components associats.

- Real-time User Awareness (RUA), que ofereix gestió per a les funcions de correlació de Sourcefire, com ara classificació per mitjà d'adreces IP, tipus de trànsit, protocols, etc.

Figura 11. Tauler de control de la família de productes Sourcefire 3D



El sistema Sourcefire 3D DC proporciona mesures de reacció, de manera que un procés actiu o semiactiu (a l'espera de confirmació d'un operador) de reacció es pot inicialitzar com a resposta a les activitats detectades per les sondes de detecció del sistema. El sistema de resposta es pot inicialitzar a partir de components de tipus IPS configurats en el sistema, i també a partir d'esdeveniments del susbistema RNA. Les respostes es poden configurar de manera que

poden ser des d'una simple generació de sessions SNMP fins a una activació de *scripts* i tasques programades a partir de l'API proporcionada pels fabricants de Sourcefire 3D. Aquesta API conté una sèrie de mòduls reactius que es poden combinar mitjançant terceres eines, incloent-hi la possibilitat de reacció basada en canvi de taules d'encaminament mitjançant encaminadors Cisco i la creació de regles de sistemes de tallafocs (Cisco, Check Point, etc.) o escàners de vulnerabilitats tipus Nmap i Nessus.

3.6.3. OSSIM

Per acabar, finalitzarem aquest subapartat amb OSSIM*, un complet SIEM de codi obert. A més de poder-se combinar amb Snort, OSSIM ofereix una gran quantitat de funcionalitats per emmagatzemar i proporcionar informació gràfica per a una gran família d'aplicacions de seguretat en xarxa, com ara:

* De l'anglès, *Open Source Security Information Management Tool.*

Enllaç d'interès

OSSIM está disponible en el lloc web http://www.alienvault.com/.

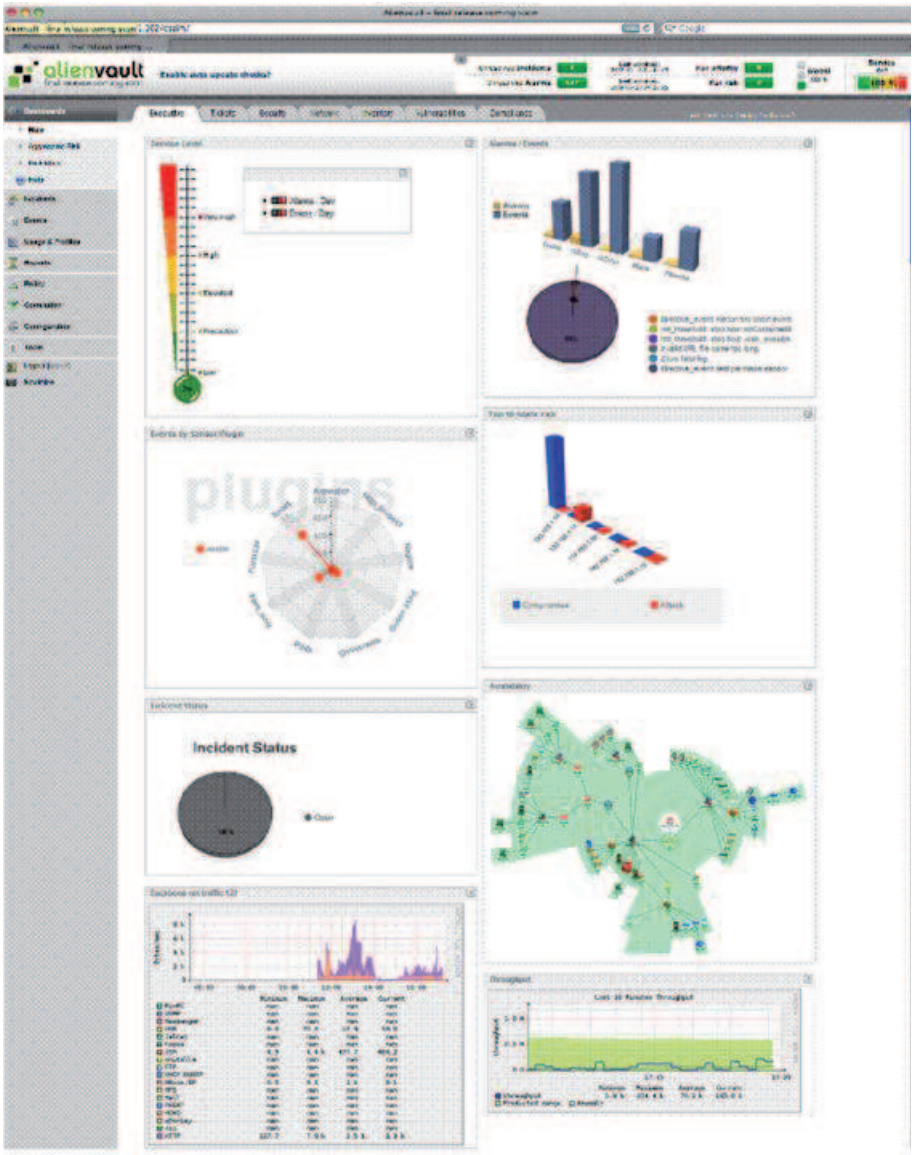
- **Nessus:** escaneig de vulnerabilitats.
- **P0f:** detecció remota de sistemes operatius.
- **Nagios:** monitoratge de xarxes.
- **Pads:** sistema de detecció basat en anomalies.
- **Osiris:** sistema de detecció d'intrusos basat en equip.
- **OSSEC:** sistema per a la detecció d'atacs d'integritat en l'àmbit del sistema operatiu, el sistema de fitxers, la instal·lació d'eines d'intrusió, etc.

La majoria de les aplicacions anteriors s'integren en OSSIM en forma de sondes de detecció o gestors d'informació independents (combinant, per exemple, les seves bases de dades amb les d'OSSIM). Encara que en alguns casos és possible adaptar les seves interfícies d'usuari a la consola d'administració general d'OSSIM, la majoria s'integren en OSSIM a partir de les seves funcions d'exportació de dades, o mitjançant la seva API. En alguns casos, pot caldre modificar o adaptar part del codi font per a incorporar funcionalitats addicionals i permetre les tasques d'agregació o correlació d'alertes. Un cop realitzades les modificacions, OSSIM es pot utilitzar com a interfície global i totes les aplicacions anteriors seran gestionades com a subaplicacions d'OSSIM (figura 12).

Tal com passa amb Snort i Sourcefire, la versió bàsica d'OSSIM també es pot complementar amb una versió comercial més completa i amb suport tècnic amb el nom AlienVault Professional SIEM*. Totes dues versions contenen funcionalitat suficient per a supervisar i gestionar la seguretat de grans xarxes mitjançant l'ampli conjunt d'eines de detecció i prevenció integrables en el producte final.

* Disponible en http://alienvault.com.

Figura 12. Interfície gràfica d'OSSIM



Resum

Les xarxes d'ordinadors es troben exposades a atacs informàtics amb tanta freqüència que cal imposar una gran quantitat de requisits de seguretat per a la protecció dels seus recursos.

Encara que les deficiències d'aquests sistemes es poden comprovar mitjançant eines convencionals, no sempre són corregides. En general, aquestes debilitats poden provocar un forat en la seguretat de la xarxa i facilitar entrades il·legals en el sistema.

De la mateixa manera que per a garantir la seguretat física d'un edifici se solen instal·lar detectors de moviment, càmeres de vigilància, llibres de registre, etc., una xarxa informàtica necessita components equivalents en el món digital per a recollir i generar escenaris d'intrusió, processar alertes i prevenir activitats intrusives.

En aquest mòdul didàctic s'han presentat els sistemes de detecció d'intrusos (IDS), l'objectiu dels quals és precisament proporcionar aquests elements complementaris als mecanismes de seguretat tradicionals i poder oferir capacitats addicionals per a avisar i guiar els administradors de la xarxa en el moment en què es produeixin atacs i intrusions informàtiques.

També hem fet una primera aproximació a Snort, un sistema de detecció d'intrusos en xarxa, basada en codi obert i ofert a la comunitat d'administradors de xarxa com a eina de programari lliure. L'objectiu principal de Snort és ajudar els administradors d'una xarxa a realitzar una vigilància continuada del trànsit de la xarxa a la recerca d'intents d'intrusió o usos indeguts.

Finalment, hem conclòs el mòdul amb una presentació general de funcionalitats addicionals per a facilitar la normalització, la consolidació i la posada en correspondència dels esdeveniments recollits per sondes de detecció heterogènies. S'ha presentat en detall alguna d'aquestes funcionalitats i s'han repassat exemples de productes que proporcionen una interfície gràfica addicional per a configurar i ampliar les capacitats ofertes per sistemes de detecció d'intrusos en xarxa com Snort, combinats amb sistemes preventius, antivirus i sistemes de detecció de vulnerabilitats.

Activitats

1. Intenteu instal·lar l'última versió disponible de Snort i féu una llista de les diferents categories en les quals es troben recollides les signatures de detecció en la distribució general de Snort.
Pista: consulteu els enllaços http://www.snort.org i http://www.snort.org/snort-rules/.

2. Busqueu cinc dels preprocessadors utilitzats per Snort i estudieu quin és el flux d'execució d'aquests components.
Pista: consulteu les informacions relacionades amb Frag3, Stream5, RPC Decode, SSL/TLS, DNS, etc.

3. Cerqueu i estudieu quin és l'algorisme utilitzat per Snort per a realitzar l'operació de *multiple-string matching*. Intenteu aïllar i analitzar en quina part del codi de Snort està implementat aquest algorisme.
Pista: inicieu la cerca per Aho-Corasick.

4. Tracteu d'explicar amb exemples situacions en les quals un NIDS com Snort serà propens a la generació de: 1) falsos positius; 2) falsos negatius.
Pistes:

- Fals positiu: situació en què les signatures de configuració de Snort són massa generals (poc explícites) i trànsit normal, que no correspon a cap atac, és alertat per Snort.
- Fals negatiu: Regles que són tan precises (poc generals) que una modificació mínima en el trànsit associat a un atac aconsegueix evadir la detecció de Snort.

5. Enumereu alguns exemples d'atac que no poden ser caracteritzats (o descrits) mitjançant regles de Snort. Expliqueu per què no és possible aquesta caracterització.
Pista: penseu en atacs que requereixin detecció basada en caracterització d'anomalies.

6. Descriviu dues o tres tècniques d'evasió per a escapar a la detecció de Snort.
Pista: consulteu els articles de Handley i Paxson i de Blunden i estudieu tècniques com la fragmentació de trànsit i el polimorfisme.

Referències bibliogràfiques

M. Handley; V. Paxson (2001, agost). "Network Intrusion Detection: Evasion, Traffic Normalization and End-to-End Protocol Semantics". A: *USENIX Security Symposium*.
B. Blunden (2009). *The Rootkit Arsenal: Escape and Evasion in the Dark Corners of the System*. Jones & Barlett.

Glossari

amenaça *f* Violació potencial de la seguretat, que existeix sobre la base d'unes circumstàncies, capacitats, accions o esdeveniments que puguin arribar a causar una infracció de la seguretat o causar algun dany en el sistema.

atac *m* Agressió a la seguretat d'un sistema fruit d'un acte intencionat i deliberat que en viola la política de seguretat.

bug *m* Vegeu **error**.

cavall de Troia *m* Programa, aparentment inofensiu, que conté en el seu interior un atac contra una vulnerabilitat no corregida.
sin. **troià**

CERT *m* Vegeu ***computer emergency response team***.

common vulnerabilities and exposures *m* Estàndard públic per a la identificació de vulnerabilitats. Associa un identificador únic a cada vulnerabilitat diferent.
sigla **CVE**

common vulnerability scoring system *m* Marc comú per a l'avaluació de la criticitat de vulnerabilitats.
sigla **CVSS**

computer emergency response team *m* Equip de respostes a emergències informàtiques; una de les seves principals tasques és la gestió de vulnerabilitats.
sigla **CERT**

computer security incident response team *m* Equip de resposta a incidents de seguretat informàtica; una de les seves principals tasques és la gestió de vulnerabilitats.
sigla **CSIRT**

CSIRT *m* Vegeu ***computer security incident response team***.

CVE *m* Vegeu ***common vulnerabilities and exposures***.

CVSS *m* Vegeu ***common vulnerability scoring system***.

DDoS *f* Vegeu **denegació de servei distribuïda**.

denegació de servei *f* Atac que intenta saturar recursos de la víctima, com ara la memòria o la capacitat de càlcul i processament.
sigla **DoS**
en **denial of service**

denegació de servei distribuïda *f* Denegació de servei que es produeix des de diversos punts de connexió.
sigla **DDoS**
en **distributed denial of service**

denial of service *m* Vegeu **denegació de servei**.

detector *m* Aplicació que intercepta tota la informació que passa per la interfície de xarxa a la qual està associada.
en **sniffer**

distributed denial of service *m* Vegeu **denegació de servei distribuïda**.

DoS *f* Vegeu **denegació de servei**.

eina d'intrusió *f* Programa que permet l'accés privilegiat a un ordinador i aconsegueix ocultar la seva presència a l'administrador. Sol usar diverses vulnerabilitats per a instal·lar-se i aconseguir el seu propòsit.
en **rootkit**

error *m* Defecte de programació que pot desencadenar una deficiència de seguretat.
en **bug**

escàner de vulnerabilitats *m* Aplicació que permet comprovar si un sistema és vulnerable a un conjunt de deficiències de seguretat.

exploit *m* Programa o *script* que permet explotar una o diverses vulnerabilitats; és a dir, un programa que permet realitzar un atac aprofitant la vulnerabilitat.

exploració de ports *f* Tècnica utilitzada per a identificar els serveis que ofereix un sistema o un equip en particular.

malware *m* Vegeu **programari maliciós**.

política de seguretat *f* Conjunt de regles i pràctiques que defineixen i regulen els serveis de seguretat d'una organització o sistema, amb el propòsit de protegir-ne els recursos crítics i sensibles. En altres paraules, és la declaració d'allò que està permès i d'allò que no ho està.

programari maliciós *m* Programa amb finalitats malintencionades.
en **malware**

risc *m* Expectativa de pèrdua expressada com la probabilitat que una amenaça particular exploti una vulnerabilitat concreta amb resultats especialment perjudicials.

rootkit *f* Vegeu **eina d'intrusió**.

sniffer *m* Vegeu **detector**.

troià *m*
sin. **cavall de Troia**.

vulnerabilitat de dia zero *f* Vulnerabilitat que, en el moment de ser explotada, no se'n té coneixement previ de l'existència.
en **zero-day vulnerability**

vulnerabilitat de seguretat Fallada o debilitat en el disseny, la implementació, l'operació o la gestió d'un sistema, que pot ser explotada per tal de violar la seva política de seguretat.

zero-day vulnerability *f* Vegeu **vulnerabilitat de dia zero**.

Bibliografia

Beale, J.; Foster, J. C.; Posluns J.; Caswell, B. (2003). *Snort 2.0 Intrusion Detection*. Syngress Publishing

Garcia-Alfaro, J. (2004). *Mecanismos para la detección de ataques e intrusiones*. A: Herrera, J.; Garcia-Alfaro, J.; Perramon, X. *Seguridad en redes de computadores*. Barcelona: Fundació Universitat Oberta de Catalunya

Garcia-Alfaro, J. (2007). *Detección de ataques en red con Snort*. A: Herrera, J.; Garcia-Alfaro, J.; Perramon, X. *Aspectos avanzados de seguridad en redes*. Barcelona: Fundació Universitat Oberta de Catalunya

Koziol, J. (2003). *Intrusion Detection with Snort*. Sams Publishing

Kruegel, C.; Valeur, F.; Vigna, G (2004). *Intrusion Detection and Correlation: Challenges and Solutions*. Springer-Verlag

Northcutt, S.; Novak, J. (2002). *Network Intrusion Detection (3a ed.)*. New Riders

Miller, D. R; Harris, S.; Harper, A. A.; Vandyke, S.;Blask C. (2011). *Security Information and Event Management (SIEM) Implementation*. Mc Graw Hill

Proctor, P. E. (2001). *The practical intrusion detection handbook*. Prentice-Hall

Rehman, R. (2003). *Intrusion Detection Systems with Snort. Advanced IDS Techniques Using Snort, Apache, MySQL, PHP, and ACID*. Prentice Hall PTR