

Implantació d’una plataforma de monitorització de la Seguretat Informàtica al CTTC

Memòria del treball de final de Màster

**David Company Estall**

Màster Universitari de Programari lliure

Administració de Xarxes i Sistemes Operatius

**Consultor UOC – Miguel Martín Mateo**

**Tutor CTTC – Jordi Escoda Ramon**

**Tutor UOC – Mario Gómez Molina**

Castelldefels - Desembre 2017

Aquesta obra està subjecta a una llicència de Reconeixement-NoComercial-CompartirIgual 3.0 Espanya de Creative Commons



**FITXA DEL TREBALL FINAL**

|  |  |
| --- | --- |
| **Títol del treball:** | *Implantació d’una plataforma de monitorització de la seguretat informàtica al CTTC* |
| **Nom de l’autor:** | *David Company Estall* |
| **Nom del consultor/a:** | *Miguel Martín Mateo* |
| **Nom del PRA:** | *Jordi Escoda Ramón* |
| **Data de lliurament:** | *12/2017* |
| **Titulació o programa:** | *Màster Universitari de Programari Lliure* |
| **Àrea del Treball Final:** | *TFM – Adm. de xarxes i sistemes operatius* |
| **Idioma del treball:** | *Català* |
| **Paraules clau** | *Seguretat informàtica, HIDS, NIDS* |
| **Resum del Treball (màxim 250 paraules):** *Amb la finalitat, context d’aplicació, metodologia, resultats i conclusions del treball* | |
| El present document descriu el procés d’implantació d’una plataforma de monitorització de la seguretat informàtica basada en el programari lliure Security Onion a l’empresa CTTC.  La plataforma de seguretat implantada permet:   * Capturar el tràfic de xarxa i detectar alertes de seguretat amb Snort * Monitoritzar la seguretat dels hosts amb OSSEC * Gestionar les alertes rebudes amb Sguil i Squert * Analitzar els possibles incidents de seguretat gràcies a l'eina de captura Bro i a l'eina de recepció i indexació de logs ELSA.   La implantació consisteix en la instal·lació del servidor Security Onion i d’un sensor en una màquina física, i en el desplegament de diversos agents OSSEC. També s’instal·la una consola d'analista per la gestió de les alertes. Previ a la fase d'implantació, es descriu una fase de desenvolupament on es comproven les funcionalitats de la plataforma i la idoneïtat de la solució en un entorn d'avaluació.  Un cop completada la implantació, es descriuen les tasques de posada a punt.  L'explotació de la plataforma de seguretat Security Onion ha permès al CTTC la detecció d'atacs, vulnerabilitats i errors de configuració en servidors. També, gràcies a la visibilitat aportada sobre el tràfic de la xarxa, permet una millora continua dels procediments i pràctiques de seguretat, i en general un augment del nivell de seguretat informàtica a l'empresa. | |
| **Abstract (in English, 250 words or less):** | |
| The document describes the setup of a Security Monitoring platform based on the Security Onion software in the CTTC company.    The deployed security platform allows:   * Capture network traffic and detect security alerts with software Snort * Monitor the hosts security with software OSSEC * Manage the security alerts with Sguil and Squert * Analyze the security incidents thanks to the Bro software and the ELSA logging and indexing tool.     The setup consists on the install of the Security Onion server and a sensor in a physical machine, and on the deployment of several OSSEC agents. An analyst console is also installed for alert management. Prior to the setup, a development phase is done in an evaluation environment to test the functionalities of the platform and the adequacy of the solution. Once the setup is completed, some tuning tasks are done and described.  The setup of the Security Onion platform in the CTTC has allowed the detection of attacks, vulnerabilities and configuration errors in the servers. Also, thanks to the provided visibility on the network traffic, it allows a continuous improvement of the procedures and security practices, and has increased the overall IT security level in the company. | |

**Í N D E X**

1. INTRODUCCIÓ 4

1.1 Context i justificació del Treball 4

1.2 Objectius del Treball 4

1.3 Enfocament i mètode seguit 5

1.4 Planificació del Treball 5

2. ANALISI DE REQUERIMENTS I ESTUDI DE VIABILITAT 8

2.1 Necessitats i requisits del client 8

2.2 Anàlisi de la situació actual 8

2.2.1 Xarxes disponibles 8

2.2.2 Sortides a Internet 9

2.2.3 Esquema de xarxa 9

2.2.4 Equipaments TI 10

2.2.5 Principals sistemes operatius utilitzats 11

2.2.6 Els principals serveis TI disponibles 11

2.2.7 Principals mesures de seguretat implementades 11

2.2.8 Problemàtiques de seguretat identificades 13

2.2.9 Principals amenaces i atacs de que pot ser víctima el CTTC 14

2.3 Definició dels requisits del sistema 17

2.3.1 Monitorització de xarxa 17

2.3.2 Monitorització de servidors 19

2.3.3 Eines d’anàlisi i visualització 20

2.4 Estudi d’alternatives de solució 20

2.4.1 HIDS 21

2.4.2 NIDS 22

2.4.3 Anàlisi i visualització 24

2.4.4 Solució integral 26

2.5 Valoració i elecció de les possibles solucions 27

2.5.1 Opció 1 - OSSEC + Snort + Elastic Stack 28

2.5.2 Opció 2 – Security Onion 28

2.5.3 Opció escollida i justificació 29

2.5.4 Valoració econòmica inicial 29

3. DISSENY DE LA SOLUCIÓ 31

3.1 Arquitectura de la solució 31

3.2 Definició dels nivells d’arquitectura 32

3.2.1 Captura de paquets 33

3.2.2 HIDS - OSSEC 33

3.2.3 NIDS - SNORT 38

3.2.4 NIDS - Bro 42

3.2.5 Sensors i servidor Security Onion 45

3.2.6 Altres eines dins Security Onion 46

3.3 Casos d’us real 47

4. DESENVOLUPAMENT 50

4.1 Instal·lació i avaluació de Security Onion 50

4.2 Valoracions inicials en la fase de desenvolupament 51

5. IMPLANTACIÓ 52

5.1 Descripció de la xarxa i col·locació del sensor 52

5.2 Descripció de la Instal·lació pas a pas 55

5.3 Consola de l’analista 55

5.4 Ajustaments i manteniment del sistema 56

5.4.1 Ajustaments inicials del sistema 56

5.4.2 Manteniment diari de Security Onion 56

5.4.3 Ajustaments de la plataforma i tractament dels falsos positius 57

5.5 Tasques pendents per després de la implantació 58

5.5.1 Equip i formació de l’equip de seguretat 58

5.5.2 Contingències i recuperació 58

5.5.3 Tasques de consolidació i explotació de la plataforma 58

6. RESULTATS I VALORACIONS 60

6.1 Deteccions i alertes rebudes 60

6.1.1 Detecció d’aplicacions vulnerables 60

6.1.2 Detecció d’errors de configuració 60

6.1.3 Polítiques de seguretat 61

6.1.4 Principals tipus d’atacs detectats i alertes 61

6.1.5 Estadístiques d’alertes 65

6.2 Millores en la seguretat informàtica al CTTC gràcies a la implantació de la plataforma 69

6.2.1 Enfortir la seguretat dels servidors web que reben més atacs 69

6.2.2 Actualització dels equips, serveis i plugins 70

6.2.3 Evitar que les contrasenyes viatgin en clar 70

6.3 Valoracions del programari Security Onion implantat 71

6.4 Evolucions futures 72

6.5 Valoració econòmica 72

7. CONCLUSIONS 74

8. GLOSSARI 76

9. BIBLIOGRAFIA 79

10. APENDIX I: Fase de desenvolupament - Instal·lació de Security Onion en mode avaluació 81

10.1 Passos a realitzar per la instal·lació 81

10.1.1 Setup Fase 1 81

10.1.2 Setup Fase 2 82

10.1.3 Verificació post-instal·lació 83

10.2 Instal·lació de l’agent OSSEC 84

10.2.1 Instal·lació de l’agent 84

10.2.2 Alta de l’agent en OSSEC Server 84

10.2.3 Configuració de l’agent OSSEC 85

10.3 Verificacions 85

10.3.1 Comprovació funcionament HIDS OSSEC 85

10.3.2 Verificació de funcionament NIDS Snort i Bro 91

10.3.3 Verificació de les eines de visualització 93

11. APENDIX II: Fase d‘implantació - Instal·lació de Security Onion en producció 97

11.1 Passos a realitzar per la instal·lació 97

11.2 Post instal·lació i proves 100

11.2.1 Estat dels serveis 100

11.2.2 Consola sguil 100

11.2.3 Testmyids 101

11.2.4 Port-mirroring en el switch 101

11.2.5 Comprovació de la carrega del sensor 103

11.2.6 Escaneig de ports 103

11.2.7 Escaneig de vulnerabilitats 104

11.2.8 Resultats de OSSEC 105

12. APENDIX III: Posada a punt del sistema Security Onion 108

12.1 Processos 108

12.2 Ajustaments a OSSEC 108

12.3 Ajustament de regles de Snort 109

12.3.1 Deshabilitar regles 109

12.3.2 Modificar regles 110

12.3.3 Crear noves regles 110

12.3.4 Autocategories 110

# INTRODUCCIÓ

## Context i justificació del Treball

Actualment la seguretat informàtica és un dels temes que està prenent més força dins del mon de les Tecnologies de la Informació. Ransomware, robatoris d’informació, atacs de denegació de servei, i tot un seguit de noves amenaces a les que les empreses han de poder fer front. Això fa que sigui necessari posar en funcionament eines informàtiques que permetin als tècnics i analistes tenir visibilitat de possibles esdeveniments de seguretat dins les seves xarxes i equips.

Dins del conjunt d’eines que poden ajudar amb la visibilitat d’incidents hi han els detectors d’intrusions de host (HIDS) i els detectors d’intrusions de xarxa (NIDS). Instal·lar aquestes eines dins la xarxa d’una empresa permet entre d’altres coses analitzar i monitoritzar el tràfic de xarxa, fer anàlisi de logs, verificar la integritat dels fitxers, detectar possibles errors de configuració referents a la seguretat, o rebre alertes quan es detectin certs patrons de tràfic o de comportament d’usuaris.

A banda, també cal una plataforma de gestió d’incidències que permeti als analistes de seguretat la visualització i l’anàlisi de les alertes.

Aquest projecte Final de Màster té com a objectiu la implantació dins d’una empresa d’una plataforma de seguretat integrada per programari de HIDS, NIDS i de les corresponents eines d’anàlisi i visualització per permetre la recollida i el tractament de les alertes de seguretat. L’empresa on s’implantarà és el Centre Tecnològic de Telecomunicacions de Catalunya (CTTC). El CTTC és una fundació sense ànim de lucre dedicada a la recerca en tecnologies i sistemes de comunicacions, focalitzat en les capes física, data-link, i de xarxa. El CTTC està situat a Castelldefels i disposa de les següents infraestructures TI: varis centres de dades repartits en dos edificis veïns, servidors, xarxa ethernet i xarxa wireless, i diversos serveis accessibles des d’ Internet.

## Objectius del Treball

Els principals objectius del projecte són:

* Disposar d’una eina per a la detecció i gestió diària de les alertes de seguretat, amb el fi de millorar la seguretat informàtica de l’empresa.
* Poder detectar atacs informàtics, i poder configurar respostes actives per minimitzar-ne els efectes.
* Poder detectar errors de configuració en els equips que puguin suposar un risc de seguretat
* Disposar d’eines per la valoració de l’estat de la seguretat, amb quadres de comandament que permetin veure’n la seva evolució.

A nivell personal, els objectius són:

* Dur a terme una implantació real i basada en programari lliure d’una plataforma de seguretat destinada a ser posada en producció.
* Consolidar els meus coneixements en matèria de programari lliure i seguretat informàtica

## Enfocament i mètode seguit

Hi han dos estratègies possibles per dur a terme el projecte.

La primera consistiria en desplegar els agents HIDS i el servidor, els sensors i el servidor NIDS, tot per separat. Després instal·lar les eines d’anàlisi i gestió de les alertes i integrar-ho tot conformant així la plataforma de seguretat.

La segona estratègia seria desplegar una solució integral que incorpori i integri totes les eines HIDS, NIDS més les eines d’anàlisi i gestió. Existeixen distribucions de programari lliure que porten integrades eines de detecció d’intrusions, de monitorització de xarxa i de gestió d’alertes, tot en un.

Les dues estratègies són apropiades per aconseguir els objectius, però la segona presenta els següents avantatges:

* Ens abstreu de les complexitats d’integració i de les dependències entre les diferents eines (HIDS, NIDS i eines de gestió).
* Redueix el temps dedicat a la implantació de la solució

## Planificació del Treball

Els recursos necessaris per realitzar el projecte són:

***Personal:***

* Un persona fent al mateix temps de cap de projecte i tècnic per implementar la solució
* Dedicació: 4 hores al dia durant 3 mesos - 240 hores

***Programari escollit:***

* Distribució **Security Onion** (https://securityonion.net/): es tracta d’una distribució Linux que incorpora eines de detecció d’intrusions, monitorització de seguretat de xarxa i gestió de logs i anàlisi forense da xarxa.
* Inclou les eines
  + Snort (https://www.snort.org/): Detecció d’intrusions en xarxa (IDS)
  + Ossec (http://www.ossec.net/) : Detecció d’intrusions en host (HIDS)
  + Bro (https://www.bro.org/): Monitorització, captura i anàlisi de xarxa
  + Sguil (https://bammv.github.io): Consola d’analista de seguretat de xarxa, amb una GUI que dona accés a les alertes i captures en temps real.
  + Squert (http://www.squertproject.org/) : aplicació web per fer consultes i visualitzar dades d’alertes IDS
  + ELSA (https://github.com/Security-Onion-Solutions/security-onion/wiki/ELSA): Enterprise Log Search and Archive (ELSA), és una eina que permet rebre, arxivar i indexar logs. Disposa d’un frontend web i motors molt eficients per a la recerca de logs.

***Maquinari:***

* Equip virtual dins d’un clúster de virtualització, amb dos interfícies de xarxa i 3Gb de RAM, on instal·larem Security Onion en versió d’avaluació.
* Switch configurable amb capacitat de fer port-mirroring
* Sensor de xarxa: equip físic amb un disc de 2Tb, processador quadcore, 16 Gb de RAM, i tres interfícies de xarxa a 1 Gbps (Intel) per la gestió i la captura de tràfic.

***Desplegament:***

Es contempla un desplegament en dos fases

***Primera fase:*** Avaluació de la solució

* Es farà un desplegament “standalone” en mode “evaluation mode” de la plataforma Security Onion.
* Permetrà familiaritzar-nos amb els diferents elements de la solució, fent simulacions d’atacs i veient com es visualitzen
* S’avaluarà el producte, i es podrà decidir l’arquitectura més apropiada per la solució.
* S’obtindran uns coneixements de base que permetran abordar la fase de producció.

***Segona fase:***Implantació de la solució i posada en producció

* Es desplegarà un servidor màster que farà de receptor dels esdeveniments i alertes, i un equip sensor, amb diferents targetes de xarxa, que permetran fer captures de tràfic en els punts crítics de la xarxa que volem monitoritzar.
* Es configurarà un switch amb port-mirroring per enviar el tràfic de xarxa cap al sensor.
* S’instal·laran agents OSSEC a diferents servidors que enviaran les alertes al servidor màster.
* Es configurarà una consola d’analista, amb accés a les diferents eines d’anàlisi i visualització d’alertes.
* S’anirà ajustant progressivament el sistema per tal de que monitoritzi correctament la xarxa i només ens avisi en cas d’alertes rellevants.
* S’aniran analitzant les alertes rebudes, classificant-les. Al mateix temps s’aniran definint procediments d’operació del sistema

# ANALISI DE REQUERIMENTS I ESTUDI DE VIABILITAT

## Necessitats i requisits del client

El CTTC és un centre dedicat a la recerca en tecnologies i sistemes de comunicacions. Hi treballen més de 100 usuaris, principalment investigadors, estudiants de doctorat, visitants i personal de suport. Les seves instal·lacions es troben repartides en dos edificis veïns situats al Parc Mediterrani de la Tecnologia de Castelldefels.

El centre disposa dels serveis TI habituals en un centre de recerca. Per un banda es disposa de serveis de connectivitat com l’accés a Internet, telefonia IP, xarxa WiFi, VPN (accés remot) i videoconferència. Després hi han altres serveis TI com són el correu electrònic, servidors de fitxers, pàgines web, fitxers en cloud, certificats digitals, eines d’ofimàtica i gestió, repositoris de control de versions i Intranet. Finalment hi ha maquinari i programari per fer simulacions.

Conscients de la necessitat cada cop creixent de portar una gestió eficient i proactiva de la seguretat informàtica, el CTTC vol disposar d’una plataforma de monitorització de la seguretat. Necessita dotar-se d’un seguit d’eines informàtiques que permetin fer una monitorització de xarxes i equips per poder conèixer el que està passant, i poder fer front a possibles amenaces que apareguin.

La seguretat es veu com un procés, i una plataforma de monitorització de seguretat es veu com un element indispensable per obtenir un coneixement més a fons dels riscos als que estan sotmesos la xarxa i els servidors

L’objectiu final és poder estar més ben preparats per fer front a les possibles amenaces i incidents i poder posar en marxa elements de prevenció més eficaços.

## Anàlisi de la situació actual

A continuació es descriuen els diferents elements que formen els sistemes TI del CTTC. També quines són les mesures referents a la seguretat implementades fins ara i quines es pensen que són les principals amenaces.

### Xarxes disponibles

La xarxa del CTTC està segmentada mitjançant VLANs i adreçaments IP separats. Les xarxes principals són:

* **Xarxa DMZ** - En aquesta xarxa trobem servidors heterogenis que tenen serveis publicats a Internet, com webs de projectes, repositoris de control de versions i serveis HTTP de cara al públic. Tenen restringit els accessos cap a xarxes internes.
* **Xarxa de servidors** - En aquesta xarxa trobem servidors que donen serveis a la xarxa interna. També alguns servidors donen serveis accessibles des d’ Internet: servidors de Email, servidors web, etc.
* **Xarxa interna** - Aquí trobem els equips d’usuari, impressores, etc. Aquesta xarxa està segmentada en diferents subxarxes i VLANs. Aquests equips no són accessibles directament des d’Internet.
* **Xarxes dels laboratoris** - Tenen equips heterogenis. Tenen alguns equips accessibles des d’Internet. Aquesta xarxa també està segmentada en subxarxes (VLANs).
* **Xarxes WiFi** - Són diferents xarxes WiFi amb propòsits diferents: WiFi per convidats, WiFi per usuaris interns, WiFi Eduroam (wireless acadèmic)

### Sortides a Internet

El centre disposa de:

* Sortida a Internet principal, amb connexió a la xarxa acadèmica. Adreçaments IP públics 84.88.61.0/24 i 84.88.62.0/23. IPv6 2001:40b0:7c22::/48
* Sortida a Internet secundària per fibra de Movistar. Serveix com a sortida a Internet de Backup i per encaminar el tràfic de les xarxes WiFi de convidats.

### Esquema de xarxa

A continuació (figura 2.1) podem veure l’esquema lògic de la xarxa CTTC.

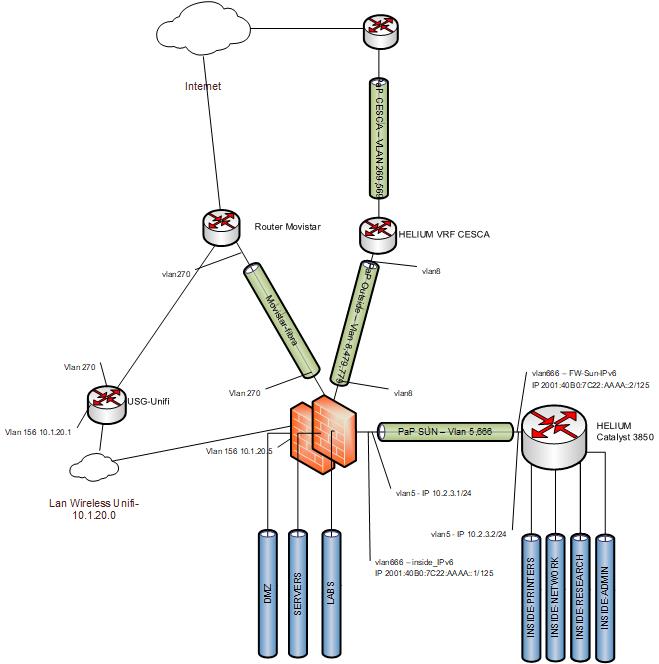


Figura 2.1: Esquema de xarxa

### Equipaments TI

Els diferents equipaments TI que podem trobar al CTTC són:

* PCs i portàtils propietat del CTTC - poden connectar-se a la xarxa interna, tant amb cable Ethernet com amb WiFi.
* Portàtils externs de visitants i convidats - només poden connectar-se a les xarxes WiFi de convidats
* Mòbils i tabletes - només poden connectar-se a les xarxes WiFi de convidats
* Servidors físics: per simulacions, magatzem de dades per projectes, per experiments
* Clúster de virtualització (VMware) amb servidors virtuals
* Punts d’accés Wireless - Solució Unifi, de l’empresa Ubiquiti.
* Switchos i routers de xarxa i altres equipaments com tallafocs
* Impressores, sensors, raspberries, equipaments de seguretat física com càmeres, detectors i demés. Segons el propòsit de l’equipament es disposa d’una subxarxa i VLAN dedicada.
* Telèfons IP

### Principals sistemes operatius utilitzats

Els sistemes operatius que podem trobar al centre són:

* Ubuntu i altres distribucions de Linux (Debian, Fedora, LinuxMint). La majoria de servidors estan basats en Linux (Ubuntu).
* Windows 7, Windows 8, Windows 10, Windows server 2008, 2012 i 2016
* MacOS

### Els principals serveis TI disponibles

Els serveis TI del CTTC són:

* Correu electrònic (postfix i imap) i llistes de distribució (mailman)
* Servei de fitxers (SMB/CIFS)
* Servei de fitxers en cloud propietari (NextCloud i OwnCloud)
* Programari d’ofimàtica (MsOffice, Adobe Acrobat)
* Programari científic (Matlab, Ansys, Mathematica, Keysight, Envi, Mentor graphics, etc.)
* VPN: accés remot a recursos TI interns (Cisco Anyconnect)
* Webs: web corporativa, Intranet, webs de projectes, webs de serveis administratius i de gestió, sistemes de peticions d’assistència (tecnologies Joomla i Wordpress principalment)
* Sistemes de monitorització (Nagios) i de backup (Veeam)

### Principals mesures de seguretat implementades

Les principals mesures de seguretat implementades al CTTC són:

Tallafocs de xarxa- Cisco ASA 5510:

* Filtratge de ports i IPs
* Filtratge a nivell de sessions
* Control de l’activitat de xarxa i del tràfic de les aplicacions
* Configuració de dos equips en redundància
* Servidor VPN per connexions remotes
* 25 interfícies definides, amb IPv4 i IPv6: servers, DMZ, laboratoris, xarxes wireless, i altres.

Tallafocs de Host en els equips servidors:

* Basat en el programari Iptables en el cas de servidors Linux
* S’estableix un segon nivell de protecció darrera el tallafocs perimetral

Tallafocs d’aplicacions:

* Es disposa de l’eina RSFirewall ( https://www.rsjoomla.com/joomla-extensions/joomla-security.html ) en les pàgines HTTP web amb Joomla. Es una extensió de seguretat que s’activa a la web i afegeix un nivell de seguretat extra ajudant a controlar la integritat dels fitxers i els permisos.

Sistemes de prevenció d’intrusions:

* Sistema fail2ban, mod-security o mod-evasive per prevenció d’intrusions en el email i pàgines web.

Segmentació de xarxa en VLANs:

* Segmentació en VLANs, algunes definides com a interfície del tallafocs, altres a nivell de switch de nivell 3
* Reducció dels dominis de broadcast

Antivirus amb consola centralitzada:

* En els equips es disposa d’antivirus Panda
* Panda treballa en cloud de manera a compartir els arxius de firma de manera més eficient entre els equips vigilats
* Consola de control centralitzada que permet administrar la solució i veure l’estat d’actualització dels equips i la recepció de les alertes de virus.

Plataforma de neteja de correu:

* Basada en Ironport de Cisco
* anti-spam
* Antivirus
* filtratge de correus
* Llistes de reputació i llistes blanques i negres
* Us de tècniques d’heurística per establir la reputació dels correus.

Backups:

* Backup de les màquines virtuals amb la solució Veeam Backup
* Backup en disc de les dades amb desenvolupament propi basat en bash script i la utilitat rsync.

Servidor de VPNs:

* El tallafocs Cisco ASA 5510 fa també de concentrador de VPN. Els usuaris poden accedir via VPN als serveis corporatius (servei de fitxers, serveis de gestió, testbeds i equips de simulació, etc.)

### Problemàtiques de seguretat identificades

A continuació enumerem les principals problemàtiques de seguretat que s’han pogut identificar al CTTC.

#### Equips heterodoxes

Al CTTC hi ha molta diversitat d’equips, degut a que és un centre d’investigació i doncs amb requeriments canviants i sempre diferents. Es disposen, tal com s’ha vist en l’apartat anterior, d’equips sovint amb diferents configuracions i sistemes operatius. Tabletes, laptops, sensors, equips de simulació i mesura, servidors, i tot un ecosistema de dispositius TI que dificulta l’aplicació de polítiques de seguretat uniformes.

#### Espais on hi han equips que poden escapar al control del serveis de suport informàtic

Al CTTC no tots els equips i servidors TI estan sota la supervisió de l’equip de suport informàtic. Existeixen laboratoris i testbeds on els investigadors instal·len i configuren els seus propis equips. En alguns casos aquests equips han de ser accessibles des d’ Internet. Malgrat que els equips estan ubicats en xarxes protegides mitjançant tallafocs centralitzat, sempre pot existir un risc de que un equip mal configurat pugui suposar un risc de seguretat.

#### Usuaris amb drets d’administrador

Els investigadors necessiten tenir drets d’administrador en els seus equips, ja que han de poder instal·lar el programari necessari per la seva feina. Això els dona molta flexibilitat, però al mateix temps incrementa el risc de que els equips puguin ser compromesos a nivell de seguretat.

#### Usuaris descuidats amb equips desactualitzats

Enllaçant amb l’apartat anterior, si els usuaris tenen drets d’administrador i no són curosos amb els programes que instal·len i els llocs que visiten, poden acabar amb l’equip infectat per malware. També la seguretat pot quedar compromesa si no tenen els equips actualitzats a nivell de sistema operatiu o de programes, o si no reporten als serveis de suport informàtic problemes de seguretat que hagin tingut.

#### Servidors desactualitzats i amb vulnerabilitats

Tot i que es parli de servidors sota la supervisió dels serveis de suport informàtic, no sempre és fàcil o possible tenir els equips al 100% d’actualitzacions i pegats de seguretat. Algunes vegades les actualitzacions són disruptives i provoquen tallades de servei que s’han de planificar prèviament. De vegades també les actualitzacions poden suposar un risc per l’estabilitat de sistemes i serveis crítics, i cal un temps d’anàlisi i prova abans de poder-les aplicar, el que suposa un risc de seguretat.

#### Equips desinventariats o desatesos

En una xarxa gran, amb molts usuaris i molts equips, de vegades és difícil tenir al dia l’inventari d’equips en funcionament, i poden existir equips desatesos sense supervisió. Aquests equips estaran desactualitzats i suposaran un problema de seguretat en cas de ser compromesos, podent servir de plataforma per atacs posteriors cap a altres equips crítics.

#### Portàtils que surten del centre i poden ser perduts o robats

Els portàtils dels usuaris surten del les instal·lacions del CTTC i poden contenir informació sensible i confidencial que pot quedar exposada en cas de pèrdua o robatori de l’equip.

### Principals amenaces i atacs de que pot ser víctima el CTTC

#### Hacking d’un servidor vulnerable

Al CTTC es publiquen alguns serveis a Internet.

* La majoria són servidors de pàgines web corporatives o la Intranet. També webs d’esdeveniments, congressos i projectes que es realitzen al centre. Algunes tenen plataformes de pagament integrades, però mai es guarden les dades de les targetes de crèdit. Altres tenen mòduls d’autenticació d’usuaris.
* En el marc d’alguns projectes, alguns servidors publiquen informació a Internet per compartir progressos i resultats amb els clients.
* El correu electrònic
* Servei de fitxers en cloud
* Eventualment es pot obrir algun servidor FTP de manera temporal

Aquests serveis publicats podrien ser atacats mitjançant:

* Explotació remota d’alguna vulnerabilitat
* Atac per força bruta per tractar d’endevinar una contrasenya dèbil

El principals riscos d’un atac en aquests sistemes serien:

* Defacing de pàgines web
* Accés no autoritzat a algun servidor, des d’on es podria pivotar per atacar un altre servidor.
* Control remot d’un servidor per usar-lo com a plataforma per realitzar activitats d’infecció amb malware o enviament de Spam.
* Robatori d’informació sensible.

#### Atac amb Ransomware i Malware

Els atacs per ransomware són cada vegada més nombrosos. El ransomware és un programari que segresta les dades d’un usuari, normalment encriptant-les, i només li torna el control d’aquestes després del pagament d’una suma de diners. Per exemple, al 2017, l’atac per el ransomware wannacry va paralitzar l’activitat de nombrosos comerços, empreses i inclús hospitals.

Els principals factors de risc que faciliten aquest tipus d’atacs són una combinació de:

* Usuaris desinformats que poden ser enganyats via phising a executar arxius o clicar en enllaços maliciosos.
* Tenir activada l’execució automàtica de macros a programes com MsOffice
* Equips amb programari des actualitzat, principalment el sistema operatiu (Windows la majoria de vegades), el navegador, adobe flash o Java.
* Accés lliure sense restriccions a Internet.

Les principals amenaces d’un atac d’aquest estil són:

* Pèrdua de dades crítiques per l’activitat de l’empresa, que poden inclús obligar-la a tancar. Tot i pagar el rescat, no es pot estar segur de recuperar les dades.

#### Usuari maliciós dins el sistema que fa mal us de les dades

CTTC és un centre de recerca. Les dades referents als projectes realitzats, investigacions en curs, resultats, dissenys, programes desenvolupats, són un actiu valuós susceptible de ser robat o difós de manera inadequada.

Les dades estan a disposició dels usuaris mitjançant:

* Serveis de fitxers, amb els corresponents ACL i permisos d’accés.
* Emails
* Serveis de fitxers en el cloud que també estan en local en els equips.
* Pàgines web

Els possibles atacs són:

* Usuaris que recopilen dades i les transfereixen a altres empreses, les roben per un futur ús, o en fan un ús il·lícit.
* Usuaris que espien activats d’altres grups.
* Un ordinador es perd o és robat i dades que poden valuoses queden exposades.

I els riscos associats a un atac d’aquest tipus:

* Robatori i difusió inapropiada de dades sensibles: dades de projectes d’ investigació realitzats, dissenys, prototips, etc.
* Sancions econòmiques per divulgació d’informacions, ja que algunes dades estant subjectes a drets d’autor
* Pèrdua d’avantatge competitiva enfront d’altres empreses competidores que s’aprofitin d’aquestes dades

#### Atac de denegació de servei

Els atacs distribuïts de denegació de servei (DDoS) normalment apunten a organitzacions grans, i el CTTC no entraria dins d’aquesta categoria. No obstant, els serveis web, d’accés a internet i de comunicacions podrien ser afectats si un atac d’aquest tipus es produís.

La principal connexió a Internet del centre es una connexió a la xarxa acadèmica proveïda per el CSUC, de 100Mbps simètrics. Un atac DDoS amb capacitat suficient podria saturar l’ample de banda i deixar el centre incomunicat.

Els riscos serien:

* una paralització important de l’activitat i de les comunicacions externes mentre durés l’atac.
* Tenint en compte que no proveïm serveis crítics cap a l’exterior, l’impacte econòmic no seria crític, donat que els atacs de DDoS no solen durar molt de temps.

## Definició dels requisits del sistema

A partir de les problemàtiques identificades, s’estableixen com a requisits el disposar de:

* Plataforma per la monitorització de la seguretat de xarxa i de servidors (NIDS/HIDS)
* Eines de visualització i gestió de les alertes i anàlisi de l’estat de la xarxa

El disposar d’aquestes eines permetria mitigar les següents amenaces abans citades (Bezborodov, 2016) (Conde, 2017):

* Detectar intents de hacking de servidors
* Detectar servidors compromesos o que tenen una activitat sospitosa
* Detectar vulnerabilitats en servidors, serveis i equips
* Detectar errors de configuració que puguin comprometre la seguretat de servidors
* Detectar fluxos de dades sospitosos o poc habituals
* Tenir evidències i alertes en el cas d’accessos no autoritzats o robatoris de dades per part d’usuaris maliciosos.
* Impedir o mitigar els efectes dels atacs de ransomware i malware
* Detectar i parar intents d’atac per força bruta
* En cas d’atac amb èxit, tenir eines d’anàlisi per determinar que ha passat i l’abast del problema.

A continuació passem a detallar quines són les característiques principals d’aquestes eines de seguretat.

### Monitorització de xarxa

Els sistemes de detecció d’intrusions de xarxa (Network Instrusion Detection Systems - NIDS) permeten dur a terme la monitorització de la xarxa.

Les seves funcionalitats principals serien:

* Captura de tràfic de xarxa en temps real
* Anàlisi del tràfic de xarxa contra conjunts de regles o patrons predefinits o definits per l’usuari. Aquests patrons poden correspondre amb tràfic maliciós, i llençar avisos en cas de que es detecti una coincidència.
* Registre de paquets de xarxa per anàlisi posterior
* Recerca d’anomalies en el tràfic, basant-se en un anàlisi estadístic. Per exemple una màquina que sobtadament generi molt més tràfic de l’habitual
* Anàlisi a nivell de funcionament dels protocols. Per exemple enviament d’ordres mal formades o fer passar un tipus de tràfic per un altre per sortejar el tallafocs.
* Possibilitat de definir i customitzar la solució, afegint regles pròpies

Els principals sistemes de monitorització de xarxa basats en programari lliure són:

* Snort (www.snort.org)
* Suricata (suricata-ids.org)

L’estructura del NIDS acostuma a ser distribuïda i consta dels següents elements: les sondes, el màster server i la xarxa de gestió

* Les sondes tenen com a finalitat capturar i analitzar el tràfic de la xarxa on es connecten. Les sondes tenen normalment dos interfícies de xarxa, una en mode promiscu per monitoritzar el tràfic, i una altre per la seva gestió i enviament de les alertes.
* El màster server o consola és l’equip que rebrà les alertes de les diferents sondes i en permet el seu anàlisi.
* La xarxa de gestió pot ser una xarxa dedicada que connecta entre sí les sondes i el màster server.

Els avantatges d’aquesta estructura distribuïda són:

* Es fàcilment escalable, ja que es poden afegir més sondes.
* La utilització d’una xarxa de gestió dedicada permet dificultar els atacs sobre els NIDS
* Evitar que unes sondes analitzin el tràfic d’altres sondes.

Els NIDS tenen les següents limitacions

* No tot és observable, i no totes les xarxes poden ser monitoritzades. No sempre es pot disposar de sondes en totes les xarxes, o de personal per analitzar tots els incidents.
* Les sondes poden descartar paquets si el tràfic excedeix la seva capacitat d’anàlisi.
* Es necessari anar configurant i ajustant el sistema per reduir al màxim el nombre de falsos positius (deteccions falses).

### Monitorització de servidors

Els Host Intrusion Detection Systems (HIDS) permetran monitoritzar l’activitat dels servidors que es volen protegir. Les funcions que realitzen són:

* **Integritat de Fitxers** (detecció de canvis en els sistemes de fitxers): mitjançant la creació d’una base de dades de referència es fa una instantània del sistema de fitxers. A partir d’aquesta instantània, qualsevol canvi pot ser detectat comparant amb la situació inicial.
* **Anàlisi de registres**: es fa un processat automàtic dels logs del sistema i de les aplicacions. Això permet descobrir activitats tals com:
  + Connexions d’usuaris locals o remotes
  + Canvis d’usuari
  + Arrencada i parada de màquina
  + Instal·lació de paquets
  + Intents d’accés fallits
  + A partir dels logs d’aplicacions, es poden detectar quan es produeixen molts intents fallits des d’una mateixa IP, o també intents d’intrusió.
* **Detecció de rootkits**: Permeten detectar les empremtes que queden quan un atac ha tingut èxit, i llençar l’avís corresponent.
* **Regles a mida:** Permeten fer regles pròpies per adaptar el sistema a l’entorn supervisat.
* **Resposta activa:** Permeten activar regles de resposta automàtica a intents d’intrusió

Els HIDS consten de:

* Un servidor que s’encarrega de la recepció de les alertes, esdeveniments, logs, i del seu anàlisi
* Agents instal·lats en els diferents equips i servidors encarregats de recollir la informació
* Els HIDS consten d’agents per plataformes Unix, Linux, Windows, VmWare, etc.
* Tenen també un mode de funcionament sense agent (“agentless”) per sistemes on no sigui possible instal·lar un agent. En aquest cas només s’envien els logs de l’equip al servidor.

Els principals sistemes de monitorització de host basats en programari lliure són:

* OSSEC (https://ossec.github.io/)
* Tripwire (https://www.tripwire.com/)

### Eines d’anàlisi i visualització

Els HIDS i NIDS generen molta informació en cru. Els HIDS generen logs i alertes quan es produeixen atacs o informació important en el sistema. Els NIDS generen encara més informació ja que poden realitzar captures de tràfic de xarxa en temps real. Al final, la combinació de les informacions detectades per les diferents fonts és la que permet a l’analista de seguretat saber si l’atac s’ha produït i ha tingut èxit, i que ha passat abans i després. Es necessari disposar d’eines que permetin rebre, tractar, relacionar, ordenar i analitzar aquestes dades.

Es requereixen eines que permetin:

* Visualització d’alertes NIDS i HIDS
* Recepció centralitzada de logs amb capacitat per normalització de logs (posar-los en un format tractable).
* Fer consultes sobre esdeveniments i dades IDS relacionades
* Accedir a les captures de dades, dades de sessió, i esdeveniments en temps real
* Fer recerques avançades de logs capturats per els sistemes de detecció.
* Fer correlació d’esdeveniments i dades de diferents orígens en temps real

## Estudi d’alternatives de solució

A continuació s’apunten quines són les alternatives per implementar la solució. Apuntem solucions basades en programari lliure. Les alternatives es comenten basant-nos en el estudi de la documentació disponible, i no en base a proves que s’hagin realitzat amb els programaris enumerats.

### HIDS

#### OSSEC



OSSEC és un programari de codi obert HIDS que incorpora les funcionalitats de:

* Monitorització de integritat de fitxers
* Detecció de rootkits
* Monitorització de processos

Els avisos poden ser via logs o via email, i es poden exportar cap a sistemes de recepció i anàlisi d’esdeveniments de seguretat.

Funciona amb agents instal·lats en els servidors, i suporta varietat de plataformes tal com Linux, Solaris, AIX, HP-UX, BSD, Windows, Mac i VMware ESX.

Està mantingut per una amplia comunitat de desenvolupadors, i integrat amb varis sistemes de seguretat comercials com AlienVault o Atomicorp. El seu us està molt estès en ISPs, universitats, entitats governamentals i grans empreses. (Cid, OSSEC Pàgina web del programari, 2017)

#### Tripwire



Tripwire és un HIDS Opensource, tot i que té una versió *enterprise*. Està focalitzat sobretot en el revisió de la integritat dels fitxers. Quan es detecten canvis en el sistema de fitxers, es generen alertes, que són configurables. Per exemple depenent de la gravetat podem configurar per que la alerta s’enviï a un grup determinat d’usuaris.

La versió de codi obert només està disponible per Linux. Per poder-la instal·lar s’ha de compilar primer, i demana un coneixement avançat d’administració Linux per fer-ho. La comunitat de desenvolupadors és activa, i la última versió disponible és del Octubre 2017.

La versió Opensource està recomanada per petits entorns o per vigilar un petit nombre de servidors, però per entorns grans és recomana la versió enterprise.

#### Logwatch

Logwatch és un sistema de gestió de logs customitzable. (Logwatch web site, 2017). Permet també:

* Parsejar logs dels sistemes per extreure informació analitzable
* Crear reports amb informació dels diferents serveis que corren en un sistema (usuaris registrats, activitat, etc.)
* Crear scripts d’anàlisi de logs customitzables

### NIDS

#### SNORT



Snort és un NIDS lliure i gratuït. Té 3 modes de funcionament diferents segons la utilització que se li vulgui donar:

* Sniffer de paquets: amb la interfície de xarxa en mode promiscu, és capaç de capturar els paquets i treure’ls per consola
* Packet logger: captura paquets i els emmagatzema en un fitxer de logs o una base de dades.
* IDS - Sistema de detecció d’intrusions: recol·lecta el tràfic en temps real, l’analitza i el compara amb regles o patrons per detectar intrusions i atacs. A partir de les deteccions, pot enviar alertes o realitzar accions preventives per mitigar l’atac, per exemple modificar el tallafocs per bloquejar una adreça IP.
* També pot detectar atacs en base a la inspecció de anomalies en el tràfic i en la utilització dels protocols de xarxa.

Les regles per detectar intrusions són molt flexibles i modificables, per tal que s’adaptin al nostre entorn de xarxa. Existeixen molts grups de regles pre-configurades llestes per ser usades. Es poden usar expressions regulars en la seva construcció.

Snort com a programari s’ha convertit en un estàndard. Es molt utilitzat per tot tipus d’empreses i institucions, i és un projecte molt viu. Està mantingut i desenvolupat per l’empresa Sourcefire. (Snort pàgina del programari, 2017)

#### Suricata



Suricata és un NIDS que té capacitats de:

* Detecció i prevenció d’intrusions, monitorització de seguretat de xarxa i processat offline de fitxers pcap (fitxers de paquets de dades capturats).
* Captura i emmagatzematge de paquets de xarxa amb pcap logger
* Igual que Snort, permet inspeccionar el tràfic i comparar-lo amb grups de regles i patrons.
* Té formats d’entrada i sortida estàndard, com YAML i JSON, que faciliten la seva integració amb SIEM (Security Information and Event Monitoring) existents, i amb altres eines de recepció/visualització de logs i d’anàlisi.
* Processat avançat dels fluxos HTTP a alt nivell (d’aplicació), que el fa molt eficient per detectar malware.
* Arquitectura multi-thread que millora el rendiment del motor

Open Information Security Foundation (OISF) són propietaris i recolzen el projecte Suricata, que darrerament té molta projecció i és cada cop més usat. (Suricata Pàgina del programari, 2017)

#### Bro



Bro és un IDS de codi lliure i gratuït, però amb característiques diferents que Snort o Suricata. Bro no es basa tant en firmes o patrons, si no que registra l’activitat de xarxa, i gràcies als seus analitzadors de protocols permet estudiar els fluxos de dades i oferir a l’analista una visió a més alt nivell del que està passant o del que ha passat. Així no només permet capturar paquets, si no que incorpora analitzadors de capa d’aplicació, entre d’altres, per els protocols HTTP, FTP, SMTP i DNS.

Es una eina d’anàlisi de tràfic de xarxa, i també permet fer anàlisi forense.

Per defecte Bro només registra la informació d’esdeveniments en fitxers, però també es pot configurar per que enviï alertes, Emails o que executi comandes.

Disposa de interfícies amb altres aplicacions per intercanvi d’informació a temps real, per exemple pot revisar checksums MD5 de descarregues HTTP contra el repositori de registres de Hash de malware Team Cymru.

Bro està desenvolupat i mantingut per el *International Computer Science Institute* de la universitat de Berkeley. (Bro pàgina web, 2014)

### Anàlisi i visualització

#### Sguil



Sguil consta d’una GUI intuïtiva que dona accés a les alertes de seguretat, dades de sessió o captures de paquets. Es una eina per facilitar la monitorització i l’anàlisi dels esdeveniments de seguretat de xarxa en temps real.

Permet treballar amb dades de diferents fonts:

* alertes NIDS de Snort i Suricata
* Alertes HIDS de OSSEC
* Logs HTTP de Bro

Permet generar informes, i les dades es guarden en una base de dades MySql. També està connectat amb altres programes i permet bascular amb un click cap a programes de captura de paquets com Wireshark o NetworkMiner, o programes de obtenció de informació com ELSA. (Sguil - Pàgina web del programari, 2017)

#### Squert



Squert és una aplicació web que permet fer consultes i visualitzar les dades emmagatzemades a la base de dades de Sguil. Donada una alerta de seguretat, ens permet obtenir informació de context amb l’ús de metadades, representacions en el temps o agrupant els resultats de manera lògica. (Burks, Squert pàgina web, 2017)

#### ELSA



Enterprise Log Search and Archive (ELSA) és una eina per la recepció, emmagatzematge i processament de logs de sistema. Permet indexar els logs, normalitzar-los i té una interfície web per fer recerques avançades i molt eficients en temps. Està optimitzada per recerques de logs de NIDS, Bro, OSSEC o syslog.

Permet configurar alertes per email, tasques programades, i també extreure gràfiques i panells de manera dinàmica. (Burks, ELSA Security Onion, 2017)

#### Elasticsearch, logstash i Kibana - “Elastic Stack”

##### Elasticsearch



Elasticsearch és una eina de codi lliure, que té una interfície web i consta d’un motor de recerca de text. Permet indexar i analitzar en temps real dades de logs. Permet també l’accés a aquests logs mitjançant recerques avançades. Està desenvolupat en línia amb el motor logstash per recol·lecció de dades i parseig de logs, i també amb la plataforma d’anàlisi i visualització kibana. (Elasticsearch Site, 2017)

##### Logstash



Logstatsh és un programari de codi obert que recopila dades de logs de diferents fonts, les processa i transforma i les envia a elasticsearch o a una altre eina que permeti la seva explotació.

##### Kibana



Kibana permet visualitzar i explorar les dades de Elasticsearch. Proveu histogrames i gràfics circular o lineals.

### Solució integral

#### SecurityOnion



Security Onion és una distribució Linux basada en Ubuntu que incorpora nombroses eines de seguretat especifiques que inclouen Snort, Bro, Suricata, Sguil, Squert, Snorby, Xplico, Networkminer i altres.

Security Onion incorpora les funcions de:

* Captura de paquets
* Detecció d’intrusions de Host i de Xarxa (HIDS i NIDS)
* Eines d’anàlisi i visualització

Les diferents funcions són realitzades per:

* Captura de paquets i emmagatzematge de paquets - netsniff-ng
* NIDS basat en regles - Snort o Suricata
* NIDs basat en anàlisi de protocols - Bro
* HIDS - OSSEC
* Eines de visualització i anàlisi - Sguil/Squert i ELSA.

Security onion presenta tres escenaris de desplegament:

* Standalone: els components del server i del sensor s’instal·len en una mateixa màquina (física o virtual)
* Server-sensor: servidor i sensor s’instal·len en màquines separades
* Hybrid: una instal·lació standalone a la que s’afegeixen més sensors.

Security Onion integra totes aquestes eines en una sola solució. Es de codi obert i gratuït. (Solutions, 2017)

## Valoració i elecció de les possibles solucions

Dins de l’etapa d’alternatives de solució no hem valorat programari comercial o solucions amb maquinari propietari de IDS/IPS (appliances). Ens hem centrat en alternatives de codi obert i gratuïtes.

Els motius són:

* **Econòmics:** l’objectiu del projecte és l’implantació dins d’una empresa, i el pressupost és limitat. Inclou la compra i utilització del maquinari necessari per fer una implantació operativa, però no dona marge per l’adquisició de llicències de programari comercial.
* **D’oportunitat:** En el camp que abasta el projecte, existeixen nombroses alternatives de codi lliure i gratuïtes de gran qualitat i que són estàndards dins la industria. La majoria de les solucions comercials inclús integren aquestes eines de codi lliure de base, i el preu a pagar es justifica per els serveis de valor afegit que incorporen.

### Opció 1 - OSSEC + Snort + Elastic Stack

La primera opció és instal·lar per una banda OSSEC com HIDS, Snort com NIDS, més l’eina de recepció i anàlisi de logs, esdeveniments i dades ElasticStack (Elasticsearch, Logstash i Kibana).

L’arquitectura seria:

* Un servidor màster per OSSEC i Snort, amb les eines de gestió de logs i anàlisi
* Agents OSSEC instal·lats en els servidors a vigilar.
* Almenys una sonda Snort

Els avantatges d’aquesta solució són:

* Entorn NIDS/HIDS complert, amb eines reconegudes com OSSEC i Snort
* Disposa d’eines per gestió d’esdeveniments de seguretat, cerca avançada de logs, generació d’alarmes, anàlisi de xarxa i generació d’informes.

Els desavantatges d’aquesta segona alternativa són:

* Les diferents eines de monitorització i anàlisi no estan integrades en el moment de la instal·lació. Cal fer la integració manualment i és una tasca que pot ser complexa.

### Opció 2 – Security Onion

La opció 2 consistiria en instal·lar la plataforma integrada de seguretat Security Onion.

Els principals avantatges de Security Onion són:

* Incorpora un HIDS reconegut i de qualitat com OSSEC
* Incorpora, a escollir, dos dels NIDS més utilitzats i reconeguts: Snort i Suricata. En el moment de la instal·lació escollim si activem l’un o l’altre
* Incorpora Bro com NIDS d’anàlisi de protocols. Bro és un bon complement als NIDS bastats en regles, ja que pot aportar major profunditat a l’anàlisi
* Incorpora potents eines d’anàlisi i de visualització de paquets, i de recerca avançada de logs: Sguil, Squert i ELSA, Wireshark, NetworkMiner
* Tots els programes que integren la solució estan pre-instalats i pre-configurats des del moment de la posada en marxa. Estan interconnectats, es complementen, i es pot pivotar entre ells, el que facilita i agilitza la fase d’ implantació.
* Es un entorn flexible i customitzable

Possibilitats de desenvolupament i escalada

* Es poden afegir sensors si es necessita augmentar l’abast de la solució per cobrir més segments de xarxa
* Es pot integrar també amb elastic stack (elasticsearch, logstatsh i kibana), segons siguin les necessitats en l’explotació de logs i de les dades recollides.

Possibles desavantatges

* No treballa com IPS per defecte. Si volem que treballi com IPS ho hem de fer a mà, i no està massa documentat
* En tenir molts programes integrats, l’aprenentatge requereix un temps.
* El propi sistema no disposa de cap eina per fer backup de la configuració. Cal usar Interprogramari extern.

### Opció escollida i justificació

Dins del nostre estudi de viabilitat, la solució que hem trobat que ens sembla més completa i que millor s’adapta als requeriments expressats és la opció 2 amb Security Onion.

Les dues opcions presenten característiques similars, ja que es basen principalment en el mateix programari per NIDS/HIDS. Però els motius per els que preferim Security Onion (opció 2) enfront de la instal·lació individual de OSSEC + Snort + elastic stack són:

* Security Onion incorpora el NIDs basat en l’anàlisi de xarxa Bro, com a complement per Snort. També porta incorporades per defecte altres eines de seguretat que poden servir de complement i ajuda, com Snorby, Xplico, o Networkminer.
* Security onion, en el moment de la instal·lació, porta tots els programes pre-instal·lats i integrats. Això ens pot fer estalviar molt de temps en el moment de la posada en marxa inicial i implantació de la solució.

### Valoració econòmica inicial

La valoració econòmica inicial de la opció escollida és la següent:

* **Programari:** En ser un projecte fet amb programari lliure, no s’ha de pagar per les llicències del programari.
* **Maquinari:** per implementar el projecte, i basant-nos en els requeriments de maquinari de la solució, s’ha comprat un PC que faci les tasques de sensor per la captura de tràfic i dades. Les característiques del maquinari són (processador i7, 3 targetes de xarxa a Giga, 16G de Ram, disc dur de 3TB). El preu és de 800€.

# DISSENY DE LA SOLUCIÓ

En el primer apartat d’aquest capítol identificarem i explicarem els diferents components que formen la plataforma de seguretat Security Onion. Destacarem les funcions principals dels mòduls HIDS i NIDS, així com el seu funcionament i la seva configuració. Explicarem també com s’integren aquests mòduls i les eines d’anàlisi entre ells.

En el segon apartat exposarem casos d’us real i com la plataforma de seguretat ens haurà de permetre donar resposta als requisits plantejats en la fase d’anàlisi.

En el tercer apartat explicarem les proves que realitzarem per provar la plataforma de seguretat, tant en la fase de desenvolupament com en la fase de implantació.

Finalment en l’últim apartat acabarem exposant les tasques a realitzar per tenir la plataforma de seguretat funcionant i operativa.

## Arquitectura de la solució

La plataforma de seguretat HIDS/NIDS (figura 3.1) té una estructura distribuïda que consta dels següents elements:

* Servidor
* Sensor NIDS i agents OSSEC
* Xarxa de gestió del HIDS/NIDS
* Consola de l’analista

El **servidor** centralitza la recepció d’esdeveniments i logs dels sensors i agents. Permet també gestionar els sensors.

La **sonda o sensor** s’encarrega de la captura i l’anàlisi del tràfic de xarxa. La sonda pot disposar de varies targetes ethernet i pot capturar el tràfic de varies xarxes i VLANs. També disposa d’una interfície per la seva gestió, que s’utilitza també per l’enviament d’alertes cap al servidor.

Els **agents OSSEC** s’instal·len en els equips a vigilar. Extreuen informació a partir dels logs, canvis en fitxers o deteccions, i l’envien al servidor.

La **xarxa de gestió** és la xarxa que connecta el servidor amb les interfícies de gestió de les sondes NIDS.

La **consola de l’analista** és un PC o màquina virtual que té instal·lades les eines de gestió i anàlisi, i que s’usa per connectar-se remotament al servidor per estudiar les alertes recollides.

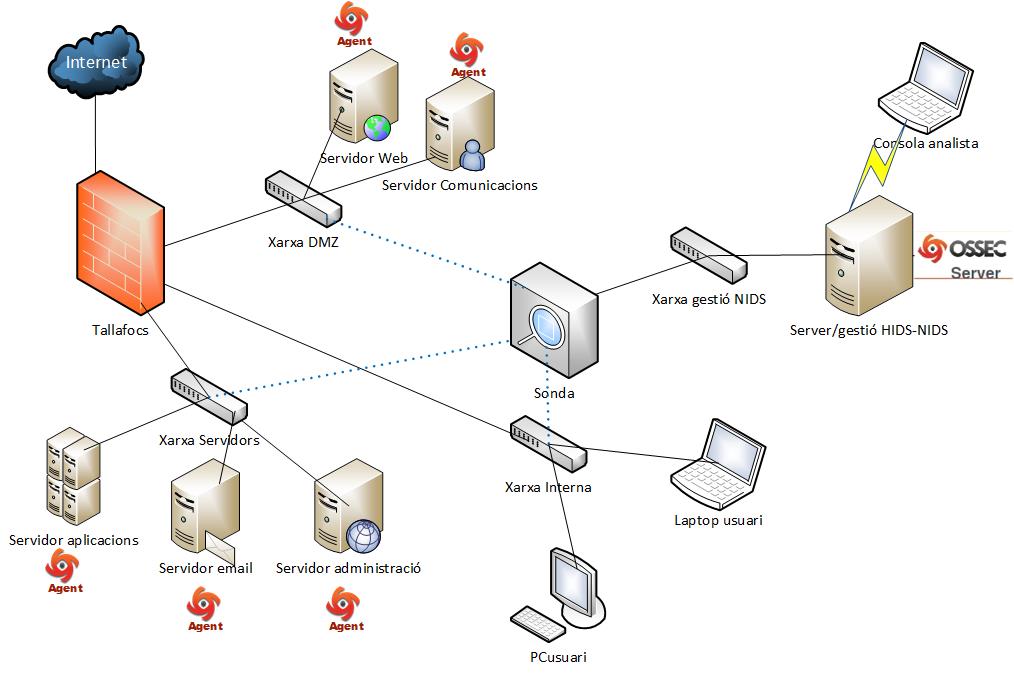


Figura 3.1 - Esquema lògic amb els elements de la solució

El sistema es pot escalar afegint noves sondes per ampliar el nombre de xarxes a monitoritzar.

## Definició dels nivells d’arquitectura

En la fase d’estudi de requeriments i anàlisi de viabilitat prèvia s’ha escollit la plataforma Security Onion com la més adequada per satisfer els requisits del projecte.

Security Onion realitza 3 funcions principals:

* Captura de paquets
* NIDS i HIDS
* Ofereix potents eines d’anàlisi

Per realitzar aquestes funcions consta de dos components:

* **sensors /agents** que capturen dades
* **servidor màster** on es tracten aquestes dades

### Captura de paquets

La funcionalitat de captura de paquets es realitza amb l’eina **netsniff-ng** a nivell del sensor. Netsniff-ng fa una captura de paquets d’alt rendiment, que es guarden en el format **pcap**. Pcap és un format compatible amb d’altres eines de captura com són tcpdump o Wireshark. Els arxius pcap es poden reproduir, amb eines com tcpreplay, per analitzar el tràfic capturat a posteriori. (Burks, Wiki Security Onion, 2017)

El sensor de Security Onion rep i emmagatzema el tràfic capturat per netsniff-ng en el directori:

***/nsm/sensor\_data/HOSTNAME-INTERFACE/dailylogs/YYY-MM-DD***

El servidor Security Onion mitjançant el procés sguild permet guardar les dades pcap capturades, i implementa un mecanisme d’esborrat de dades antigues per fer lloc a les noves.

En el següent esquema (figura 3.2) es pot veure l’arquitectura de la funció de captura de paquets de Security Onion

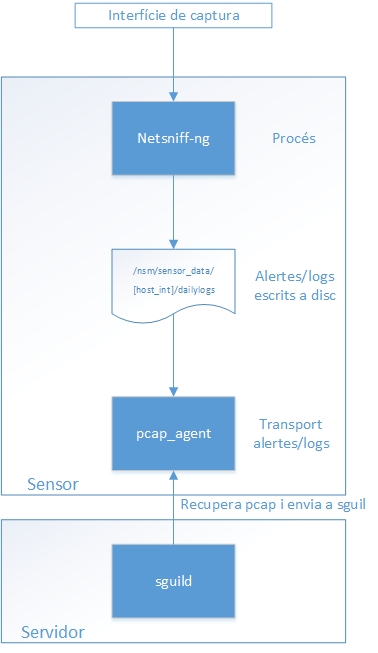


Figura 3.2 - Captura de paquets amb Security Onion

### HIDS - OSSEC

Per realitzar la funció de HIDS, Security Onion compta amb l’eina OSSEC (Cid, Log analysis using OSSEC, 2007).

OSSEC permet:

* Anàlisi de logs
* Control de la integritat de fitxers
* Detecció de rootkits
* Avisos en temps real i resposta activa

#### Funcionament de OSSEC

OSSEC treballa en un model agent/servidor (també pot treballar en local en un únic equip a vigilar), on es té:

* Agents OSSEC instal·lats en els servidors a vigilar
* Servidor OSSEC que centralitza la recepció de les alertes

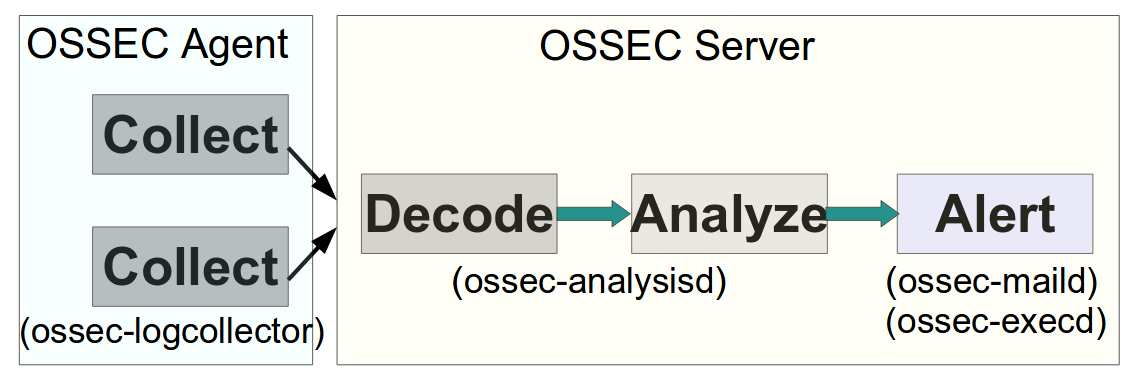


Figura 3.3 - Esquema descriptiu de l’anàlisi del flux dels logs en arquitectura client/servidor

Els processos implicats són (figura 3.3):

* **ossec-logcollector**: Captura els logs
* **ossec-analysisd**: Anàlisi i decodificació dels logs
* **ossec-maild**: Enviament d’alertes
* **ossec-execd**: Respostes actives (per exemple bloquejant adreces IP en el tallafocs)

El procés principal per anàlisi de logs és **ossec-analysisd**, que realitza 3 funcions principals (figura 3.4):

* **Pre-decodificació de logs**: Tractament dels logs recollits del sistema i de les aplicacions, per deixar-los en un format adequat per ser descodificats. Llegeix la capçalera del log i n’extreu els camps genèrics, com la data, el hostname, el programa i altres.
* **Decodificació de logs**: Identifica la informació clau dins els logs, a partir d’expressions (decoders). Els decoders més importants corresponents a les aplicacions més comunes estan pre-instal·lats en el servidor OSSEC (apache, iptables, pam, ftp…). També es poden definir decoders per part dels usuaris. Els decoders permeten descodificar els logs i extreure’n la informació rellevant (adreça IP, protocol, usuari, acció...) per ser usada després en les regles.
* **Anàlisi basat en regles**: Partint de la informació descodificada, es fan comparacions amb les regles definides per trobar coincidències.

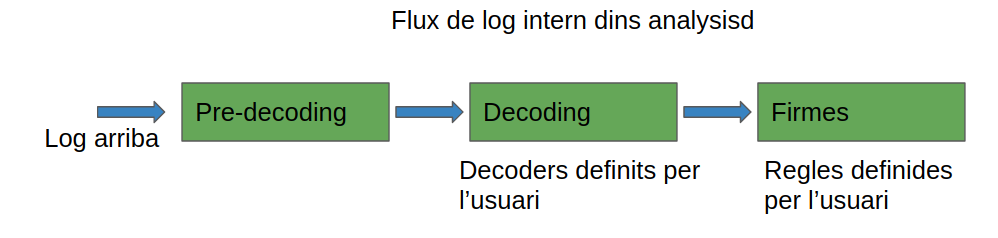


Figura 3.4 - Esquema del flux dels logs dins del procés analysisd

#### Decoder

Els decoders de OSSEC s’escriuen en format XML i es guarden en el fitxer:

***etc/decoders.xml***

El seu format és:

* **Nom del decoder**: Ha de tenir sentit, per poder fer-ne referència més endavant.
* **prematch o program\_name:** Camp que permet fer referència a un decoder previ.
* **regex:** Expressió regular que caracteritza la sintaxi del log i d’on es poden extreure els camps destacats
* **ordre**: S’especifica que significa cada camp destacat (id, srcip, dstip, srcport, dstport, url, action, status, user, location, etc)
* **Estructura en arbre:** Els decoders poden tenir una estructura en forma d’arbre, amb decoders fills que fan referència a decoders pare, per permetre la interpretació dels missatges de log.

***Exemple de decoder:***

A partir de la línia de log d’apache:

192.168.2.190 - - [18/Oct/2027:13:10:06 -0500] "GET /xxx.html HTTP/1.1" 200 1732

El següent decoder extreu els camps srcip (ip origen), url i id

<decoder name="web-accesslog">  
 <type>web-log</type>  
 <prematch>^\d+.\d+.\d+.\d+ </prematch>  
 <regex>^(\d+.\d+.\d+.\d+) \S+ \S+ [\S+ \S\d+] </regex>  
 <regex>"\w+ (\S+) HTTP\S+ (\d+) </regex>  
 <order>srcip,url,id</order>  
 </decoder>

#### Regles

Les **regles** de OSSEC tenen les següents característiques:

* Tenen una estructura en forma d’arbre
* Tenen format XML. Moltes estan definides per defecte, i també poden ser definides o modificades per l’usuari.
* Busquen coincidències a partir de la informació descodificada en la fase del decoder.
* Hi han dos tipus de regles: atòmiques (basades en un sol esdeveniment), o compostes (basades en patrons entre diferents logs)

El format de la regla és:

* **id** de la regla
* **level**: nivell de la regla, de 0 (més baix) a 15 (més alt)
* **patró**: expressió regular, o camp tipus srcip, id, usuari, etc.

***Exemple de regla composta:***

Està composta de dos regles.

* La primera genera un esdeveniment si el log és del procés sshd
* La segona s’activa si la primera regla es compleix i el password introduït és erroni.

<rule id = "111" level = "5">  
<decoded\_as>sshd</decoded\_as>  
<description>Logging every decoded sshd message</description>  
</rule>

<rule id=”122” level=”7”>  
<if\_sid>111</if\_sid>  
<match>^Failed password</match>  
<description>Failed password attempt</description>  
</rule>

#### Funció de comprovació d’integritat dels fitxers

OSSEC comprova mitjançant el procés **syscheck** la integritat dels fitxers del host on està instal·lat. Això significa que verifica de manera periòdica si els fitxers importants (o registres en el cas de Windows) han patit canvis, per buscar traces de possibles atacs.

Per comprovar l’integritat, syscheck calcula un checksum MD5/SHA1 de cada fitxer clau del sistema, i en monitoritza els canvis. Cada 6 hores (per defecte, ja que és un valor configurable), l’agent OSSEC escaneja el sistema i envia els checksums al servidor, que els guarda i compara amb els de referència (que reflexa l’estat original del fitxer) per veure si hi ha hagut modificacions. En el cas de que no coincideixin, salta una alerta.

La configuració de la funcionalitat està en el fitxer **ossec.conf**, on poden:

* Afegir o ignorar fitxers/directoris a monitoritzar
* Ajustar la severitat dels avisos segons el fitxer/directori
* Per evitar falsos positius, quan un fitxer ha canviat 3 vegades, els canvis posteriors són ignorats

El procés de syscheck també permet recollir informació de:

* Quins canvis s’han realitzat en fitxers de text
* Si hi han canvis en els permisos, en el propietari o en el mida d’un fitxer

#### Integració de OSSEC dins l’arquitectura Security Onion

A continuació veiem com s’integra OSSEC dins la plataforma de seguretat Security Onion (figura 3.5). (Lambert, s.f.)

1. L’agent OSSEC envia les alertes detectades, esdeveniments i checksums MD5/SHA1 dels fitxers vigilats cap al procés OSSEC-server que es troba en el sensor de Security Onion.
2. El procés OSSEC-server guarda les alertes rebudes en el fitxer

***/var/ossec/logs/alerts/alerts.log i /var/ossec/logs/archives/archives.log***

1. Aquesta informació es tractada pel procés **sguild** per guardar-la en la base de dades del servidor Security Onion. Posteriorment aquesta informació pot ser tractada amb les eines d’anàlisi i gestió de logs Sguil i ELSA.

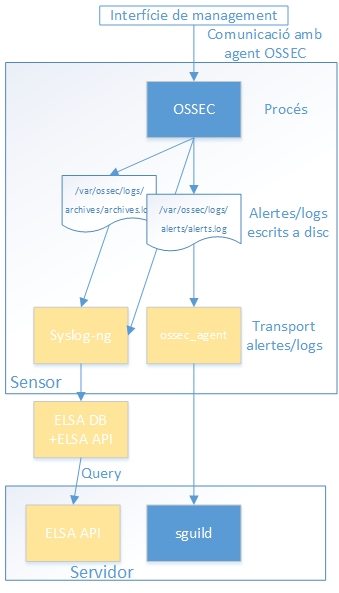


Figura 3.5 - Integració de OSSEC dins de l’arquitectura de Security Onion

### NIDS - SNORT

Per realitzar la funció de NIDS, Security Onion dona a escollir entre Snort o Suricata. Per la nostre implementació escollim Snort, ja que és considerat un estàndard de facto, disposa d’una comunitat més gran de desenvolupadors i també consumeix menys recursos de programari. (Aanval, 2016)

SNORT treballarà en mode NIDS (pot treballar en altres modes), el que vol dir que el programa monitoritzarà el tràfic de xarxa i si apareixen esdeveniments de seguretat, els registrarà i activarà una alerta. Snort disposa de varis conjunts de regles activades, que es van actualitzant contínuament, com els de VRT-Talos (per subscripció i pagament) o els de Emerging Threats.

#### Funcionament de Snort

Pel seu funcionament, Snort disposa dels següents mòduls (figura 3.6) (Snort Users Manual 2.9.11, 2017):

* **Sniffer**: S’encarrega de la captura de paquets. Dins de Security Onion, Snort es compila amb el mòdul pf\_ring, que en una màquina amb múltiples CPU cores permet accelerar la captura i processar més tràfic.
* **Pre-processadors**: Processen els paquets de dades rebuts, previ a l’aplicació de les regles. Es normalitza i prepara el flux de bytes per la següent fase de detecció, realitzant un parseig de les capçaleres dels paquets, detectant anomalies o errors, i restablint l’ordre dels fragments. Snort porta incorporats diversos plug-ins específics dissenyats per processar diferents tipus de tràfic, com per exemple SMTP preprocessor (per tràfic de Email) , o HTTP Decode (normalització del tràfic HTTP).
* **Motor de detecció**: Es el conjunt de regles amb les que comparar i buscar coincidències per detectar problemes de seguretat. Moltes venen ja incorporades, i també poden ser definides per l’usuari.
* **Alertes/registre**: Aquest mòdul gestiona la sortida (output) de les dades després de la fase d’anàlisi i detecció. Permet generar alertes, registrar les dades capturades en un format tipus XML o CSV, enviar informació a un servidor syslog o guardar-la en una base de dades.

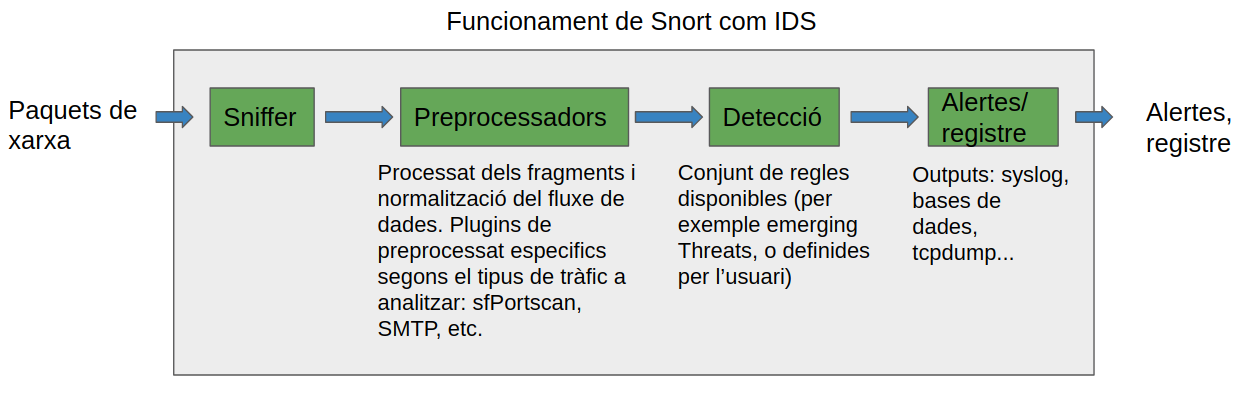


Figura 3.6 - Esquema del funcionament de Snort com a IDS

#### Configuració bàsica de Snort

Snort es configura mitjançant l’arxiu

***/etc/nsm/[nom\_sensor\_intf]/snort.conf***

Amb el fitxer snort.conf es configuren els següents paràmetres (Snort Users Manual 2.9.11, 2017):

1. **Variables de xarxa:** xarxes internes i externes
2. **Decoder:** determina el protocol usat a partir dels paquets capturats
3. **Motor:** Configurar el motor de detecció base
4. **Llibreries:** Configurar les llibreries dinàmiques: són les llibreries usades després per els pre-processadors
5. **Pre-processadors:** Podem escollir quins pre-processadors volem activar.
6. **Output plugins:** normalment és unified2
7. **Categories de regles:** decidim quines categories de regles carreguem.
8. **Customitzacions:** Customització del pre-processador, del joc de regles del decoder i del joc de regles d’objectes compartits

Després de fer qualsevol canvi en la configuració de Snort cal reiniciar el sensor.

#### Construcció de regles amb Snort

Les regles per la detecció d’intrusions dins de Snort es troben en el directori (Snort Users Manual 2.9.11, 2017)

***/etc/nsm/rules/***

Les regles estan formades per una capçalera més unes opcions

**[action][protocol][sourceIP][sourceport] -> [destIP][destport] ( [Rule options] )**

La capçalera de la regla té els següents camps:

* Acció de la regla **[action]**: El que Snort ha de fer quan troba un paquet que coincideix amb el criteri de la regla. El més habitual és **alert**
* Tipus de tràfic **[protocol]**: Els 4 tipus de protocol possibles són **TCP, UDP, ICMP, IP**
* Adreça origen **[sourceip]**: Pot ser una adreça literal, o la variable **$EXTERNAL\_NET**
* Port origen **[sourceport]**: Literal, o una variable, per exemple **$HTTP\_PORTS**
* Operador de direcció “**->**” indica la orientació del tràfic al que s’ha d’aplicar la regla
* Adreça destí **[destip]**: Pot ser una adreça literal, o la variable **$HOME\_NET**
* Port destinació **[destport]**: Literal, o una variable, per exemple **any**

Les opcions de la regla són:

* Generals
  + Message: missatge, en forma de cadena de text, per donar sentit a l’esdeveniment que la regla està detectant.
  + Flow: Especifica la direcció del tràfic de xarxa per que la regla s’apliqui. Es pot usar amb d’altres paràmetres que identifiquen la sessió, com per exemple “established” per el cas TCP.
  + Reference: la regla pot incloure referències a fonts d’informació externes, com per exemple CVEs (identificadors de vulnerabilitats conegudes).
  + Classtype: paraula clau que usa Snort per classificar el tipus d’atac
  + sid/rev: identificador de snort per la regla. S’utilitza junt amb rev (revisió) i permet als plugins de sortida identificar les regles
* Detecció
  + Content: regles que cerquen contingut específic dins el paquet de dades, i activen accions de resposta basades en les dades contingudes.
    - distance/offset: paraules clau que, en escriure la regla, permeten especificar on començar la cerca en relació a l’inici del camp de dades del paquet o d’una coincidència.
    - within/depth: paraules clau que permeten especificar quant de lluny cercar en relació al final d’una coincidència prèvia
  + PCRE: paraula clau que permet escriure expressions regulars en perl
  + Byte test: permet a una regla comparar bytes contra un valor binari específic

**Exemple (figura 3.7)**:

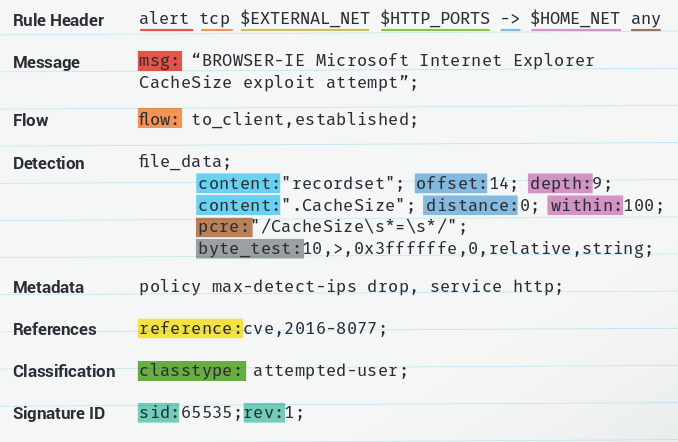


Figura 3.7 - Exemple de regla amb Snort

#### Integració de Snort dins de Security Onion

A continuació veiem com s’integra SNORT dins la plataforma de seguretat Security Onion (figura 3.8). (Lambert, s.f.)

1. Les dades capturades en la interfície de xarxa del sensor són enviades al procés SNORT per el seu tractament.
2. El procés SNORT processa les dades, genera alertes i les registra dins **/nsm/sensor\_data/[hotsname\_interface]**
3. El programa Barnyard2 permet interpretar de manera eficient els fitxers de sortida binaris generats per Snort. Barnyard2 dona format a aquests esdeveniments en format Unified2, i els fa arribar al procés sguild que els registra en la base de dades del servidor Security Onion.
4. També s’envia la informació dels logs capturats cap al procés syslog-ng, que els processa i guarda el resultat en la base de dades ELSA\_DB. Aquestes dades podran ser consultades des de l’eina ELSA

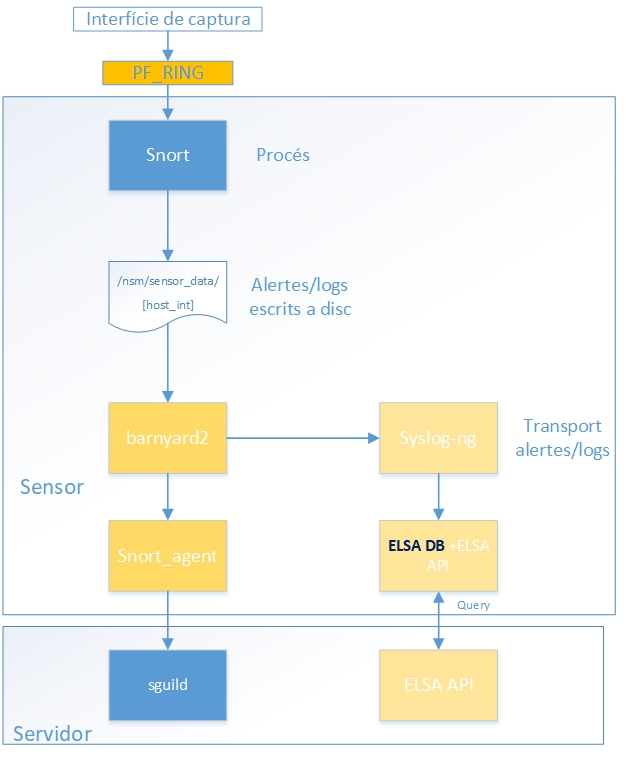


Figura 3.8 - Esquema del funcionament de Snort dins de l’arquitectura de Security Onion

### NIDS - Bro

Bro monitoritza el tràfic de la xarxa i genera logs d’informació de:

* Connexions TCP/UDP/ICMP - **conn.log**
* Activitat DNS - **dns.log**
* activitat FTP - **ftp.log**
* activitat HTTP - **http.log**
* activitat SSL - **ssl.log**

Tota aquesta informació recollida es pot consultar mitjançant l’eina ELSA integrada dins Security Onion. Amb aquesta informació també es generen alertes de seguretat.

Bro és altament adaptable, ja que disposa d’un llenguatge de scripting que permet fer polítiques de monitorització específiques per el nostre entorn.

#### Funcionament de Bro

L’arquitectura de Bro consta dels següents elements (figura 3.9) (Slagell, s.f.):

* **Motor d’esdeveniments:** rep els paquets capturats a la interfície de xarxa, i descodifica els diferents protocols
* **Mòdul de scripts:** permet interpretar les polítiques de monitorització i a partir dels esdeveniments rebuts, realitzar un anàlisi lògic. La sortida d’aquest mòdul són logs i alertes.

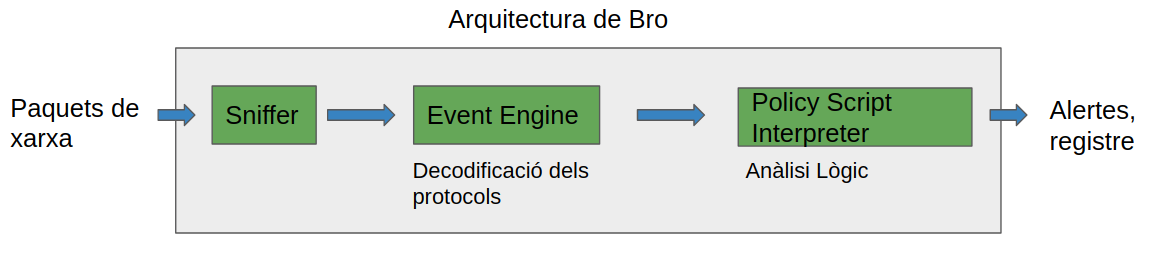


Figura 3.9 - Arquitectura de Bro

#### Configuració bàsica de Bro

Els principals arxius de configuració del programa Bro són:

* **/opt/bro/etc/broctl.cfg** - paràmetres globals de bro. Entre d’altres el email on s’ha d’enviar els missatges, la gestió dels logs i el script de polítiques locals.
* **/opt/bro/etc/networks.cfg**- En aquest arxiu s’afegeixen les xarxes que es consideren locals dins l’entorn monitoritzat.
* **/opt/bro/etc/node.cfg** - s’especifica els nodes bro en funcionament i la seva configuració, que pot ser standalone (un sol node) o bé en clúster. També la interfície de xarxa on realitzem la captura de paquets. Normalment estarà en mode standalone.

Bro disposa de l’eina **BroControl**, per permetre el control i la operació de Bro, ja sigui per comandes o mitjançant una shell interactiva. En treballar amb Bro de manera integrada amb Security Onion, podem obviar relativament totes les comandes necessàries per la instal·lació i desplegament de Bro.

#### Funcionalitats de Bro

Dins de Security Onion, podem trobar els logs de Bro dins

***/nsm/bro/logs/***

L’arxiu de logs va rotant cada hora per defecte, i els logs més antics es van arxivant.

Els logs d’informació recopilats per Bro són una font d’informació molt complerta que ens ajuda a comprendre millor la nostre xarxa. A partir de la informació de les connexions establertes, el tràfic http, ssl, syslog, dhcp, fitxers i software, podem:

* Estudiar el comportament de la nostre xarxa: tipus de fitxers que viatgen, clients browser més utilitzats, màquines que generen més tràfic, etc.
* Analitzar que ha passat donat un problema de seguretat: màquines implicades, urls visitades, fitxers descarregats, etc.
* Generar alertes a partir d’esdeveniments sospitosos: logins des d’un país no esperat, logins amb un hostname no usual, accessos a dominis sospitosos, comprovar si arxius descarregats són malware, etc.
* Incorpora scripts que permeten detectar violacions de polítiques locals: intents d’autenticació fallits, escanejos de ports (comptant intents de connexió per adreça font), etc.

#### Integració de Bro dins Security Onion

A continuació veiem com s’integra Bro dins la plataforma de seguretat Security Onion (figura 3.10). (Lambert, s.f.)

1. Les dades capturades en la interfície de xarxa del sensor són enviades al procés Bro pel seu tractament.
2. El procés Bro guarda la informació extreta dins **/nsm/bro/logs**
3. El procés syslog-ng es nodreix d’aquests logs, els tracta i els emmagatzema en la base de dades de l’eina ELSA.

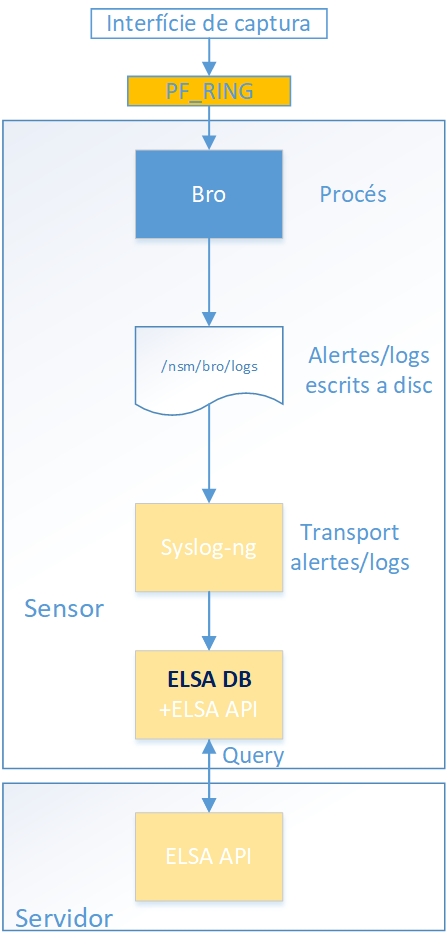


Figura 3.10 - Esquema del funcionament de Bro dins de l’arquitectura de Security Onion

### Sensors i servidor Security Onion

En el següent esquema (figura 3.11) podem veure com s’integren totes les eines i elements que hem descrit anteriorment en l’arquitectura de client/servidor de Security Onion. També veiem quins són els fluxos de dades de que s’alimenten les diferents eines d’anàlisi i visualització, principalment Sguil i ELSA. (Lambert, s.f.)

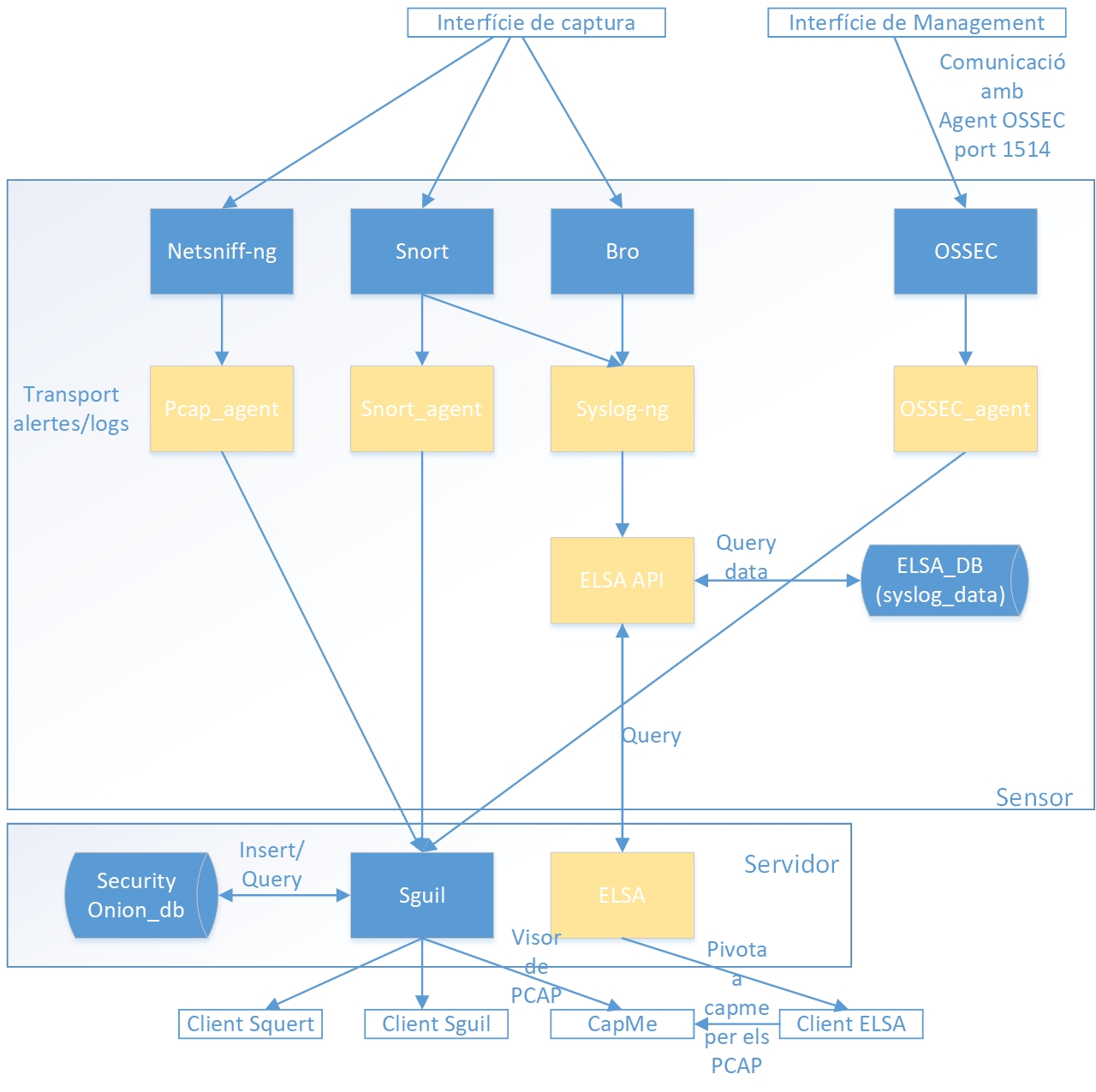


Figura 3.11 - Esquema d’arquitectura general de Security Onion i els seus elements

### Altres eines dins Security Onion

A banda de les eines mencionades, Security Onion disposa d’altres eines, per ajudar en l’anàlisi de seguretat i la detecció de programari maliciós. Programes dins de Security Onion com Sguil es recolzen i poden pivotar cap a aquestes eines per ajudar a estructurar i treure més partit de la informació capturada.

#### WireShark

Wireshark és un conegut analitzador de protocols de xarxa. Les principals característiques són:

* Proporciona una interfície gràfica amb la que permet visualitzar el tràfic de xarxa capturat
* Permet guardar la informació de capturada
* Permet llegir dades de xarxa (capturades prèviament) des d’un arxiu pcap
* Permet reconstruir sessions TCP (el flux de paquets intercanviats entre client i servidors)
* Es programari lliure

#### NetworkMiner

NetworkMiner és una eina d’anàlisi forense de xarxa, que a partir d’una captura de tràfic entre equips, permet determinar els sistemes operatius implicats, les sessions, els hostnames, els fitxers i altres informacions implicades en el flux. Igual que Wireshark, permet fer un anàlisi offline a partir d’un arxiu de captura pcap.

#### Squert

Squert és una aplicació web que permet veure els esdeveniments IDS guardats a la base de dades de Sguil. A partir dels esdeveniments visualitzats, permet extreure informació de context, tal com metadades, agrupacions de la informació, etc. És un complement a Sguil.

#### Capme

CapME ens mostra el flux de paquets tcp en format web a partir d’un arxiu pcap. Dins de Security Onion, es pot pivotar a CapMe des de ELSA o des de Squert, i permet crear transcripcions pcap a partir de les captures de Bro.

## Casos d’us real

En aquest apartat exposarem com la plataforma de monitorització haurà de donar resposta a les necessitats plantejades en la fase d’anàlisi de requisits.

La plataforma constarà de:

* Sensor des d’on es podrà capturar el tràfic de les xarxes o VLANs que volem vigilar.
* Switch configurat amb port-mirroring per enviar tràfic al sensor
* Agents OSSEC que controlaran la seguretat en servidors.

Gràcies a la plataforma de seguretat amb HIDS/NIDS, haurem de ser capaços de detectar i controlar els problemes de seguretat que sorgeixin. Des de la consola Sguil podrem veure les alertes generades i les regles que les han activat. Mitjançant l’eina Bro, i gràcies al cercador de logs ELSA, podrem aprofundir en l’incident i tractar d’entendre què ha passat (qui l’ha provocat. a qui ha afectat, accions malicioses realitzades, etc.). (Bezborodov, 2016)

A continuació enumerem els principals problemes de seguretat que es poden produir i com les eines de monitorització de seguretat ens podran ajudar a resoldre’ls o mitigar-los.

***Servidors compromesos per un programari maliciós***

* OSSEC, mitjançant la vigilància de l’integritat dels fitxers (o de canvis en el registre si és una màquina Windows) ens avisa si es produeixen canvis en algun dels fitxers importants del sistema
* OSSEC permet la detecció de rootkits en màquines compromeses.
* Snort vigila el tràfic maliciós i sospitós. Snort pot detectar tràfic generat per programari de tipus trojan (atac que permet a l’atacant controlar remotament l’equip infectat) i llençar una alerta.

***Detectar intents de hacking o intrusions en servidors***

* Snort, amb mòduls com per exemple sfPortscan, permet detectar un escaneig de ports en un servidor.
* OSSEC també permet detectar un escaneig de ports. (Ochobits, 2015)

***Detectar atacs per força bruta***

* L’anàlisi de logs dins OSSEC permet detectar atacs de força bruta, ja que es detectaran molts missatges fallits d’autenticació.
* Snort també disposa de firmes que li permeten detectar atacs de força bruta.

***Servidors amb errors de configuració que puguin comprometre la seguretat***

* OSSEC fa anàlisi de logs del servidor i de les aplicacions dins el servidor, i ens avisa si detecta errors o fallades en el sistema o les aplicacions deguts a errors de configuració.

***Detecció i prevenció d’atacs en una aplicació web, Cross-site scripting (XXS) o SQL Injection (SQLi)*** (Deuble, 2016)

* Snort disposa de firmes per detectar atacs de XXS i SQL Injection.

***Detectar vulnerabilitats en servidors, serveis i equips***

* Dins la plataforma Security Onion, ni Snort, ni OSSEC, ni Bro porten integrada cap funcionalitat de detecció de vulnerabilitats (vulnerability assesment).
* Malgrat això, amb la informació recollida sobre els servidors més atacats o el tipus d’atacs més comuns es poden detectar els punts més vulnerables de la xarxa.

***Detectar fluxos de dades sospitosos o poc habituals***

* Snort disposa de signatures per detectar aquest tipus de tràfic i avisar.
* Bro pot enviar alertes si detecta que el tràfic capturat és anormal o sospitós (per exemple un usuari es registra des d’un país estranger on no hi tenim cap treballador).

***Tenir evidències i alertes en el cas d’accessos no autoritzats o robatoris de dades per part d’usuaris maliciosos.***

* ELSA ens permet fer recerques dins el tràfic de xarxa capturat per Bro, i permetre descobrir quines dades s’han robat.
* Bro permet fer anàlisi forense de l’activitat de xarxa, i ens ajuda a entendre que ha passat.
* Snort ens avisa de l’activitat de xarxa sospitosa.

***Impedir o mitigar els efectes dels atacs de ransomware i malware***

* Bro permet detectar i llençar una avís en cas arxius descarregats d’internet perillosos. Això ho fa calculant el hash de l’arxiu i comparant-lo amb la base de dades de hash d’arxius maliciosos de Team Cymru.
* Snort també conté signatures que permeten detectar comportaments anormals que puguin ser indici de màquines infectades

***En cas d’atac amb èxit, tenir eines d’anàlisi per poder determinar que ha passat i establir l’abast del problema.***

* ELSA ens permet fer recerques basades en el tràfic capturat per Bro, i fer un anàlisi forense per entendre que ha passat.
* Eines com wireshark o networkminer, integrades dins Security Onion, permeten seguir el flux TCP dels paquets de xarxa i analitzar el tràfic entre l’atacant i la víctima

# DESENVOLUPAMENT

En la fase de desenvolupament s’instal·larà Security Onion en un entorn de proves i en mode avaluació. El procés d’instal·lació i les proves de validació realitzades estan detallats en el Apèndix I.

## Instal·lació i avaluació de Security Onion

La instal·lació de Security Onion es farà en una màquina virtual. La instal·lació en mode avaluació ja porta incorporats el servidor i el sensor integrats en una mateixa màquina, junt amb totes les utilitats de HIDS, NIDS, anàlisi i visualització.

En el següent esquema (figura 4.1) es mostra la col·locació de la sonda de captura de tràfic i dels agents OSSEC.

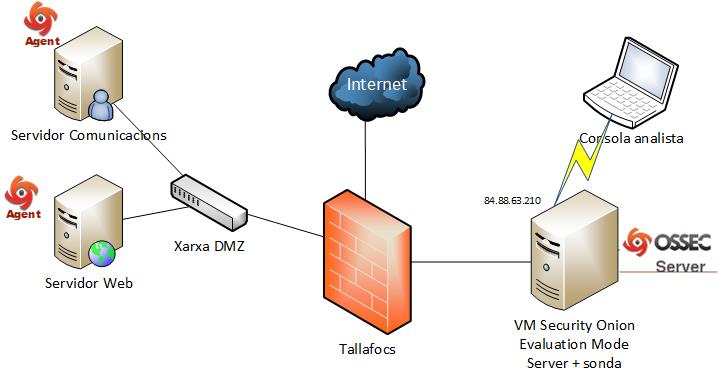


Figura 4.1 - Esquema de la instal·lació de la VM Security Onion en mode avaluació

Les especificacions de la màquina virtual on s’ha instal·lat el sistema d’avaluació són les següents:

* Hostname: seconion1
* Maquinari
  + CPU: 2
  + Memòria RAM: 3Gb
  + Disc dur: 40Gb
  + 2 adaptadors de xarxa, un per monitorització i l’altre per l’administració del servidor
  + IP management: 84.88.63.210/255.255.255.192
  + Interfície xarxa monitorització.
* Programari
  + Host: VmWare versió 6.5
  + Guest: Security Onion 14.04.5.2 ISO Image

Veure [Apèndix I](#_APENDIX_I:_Fase) per el detall.

## Valoracions inicials en la fase de desenvolupament

En aquest apartat valorarem la instal·lació i les proves fetes amb Security Onion en la fase de desenvolupament.

* La instal·lació de Security Onion en mode avaluació és molt senzilla i funciona correctament, en el sentit de que s’obté un entorn funcional en poca estona.
* Les proves realitzades amb OSSEC han sigut satisfactòries. S’han provat les funcionalitats de detecció d’atacs, d’integritat de fitxers i de resposta activa amb resultats satisfactoris. També s’ha provat la modificació de regles amb èxit.
* Les proves realitzades amb Snort han sigut satisfactòries. Snort ha detectat correctament els atacs simulats. S’ha provat amb èxit la funcionalitat de creació i modificació de regles.
* Les eines de visualització i anàlisi s’estimen útils i funcionals. Sguil és una consola de visualització pràctica, amb moltes funcions integrades que permeten complementar la informació aportada per l’alerta.
* Hem comprovat la informació recollida per Bro i la seva utilitat. També hem comprovat com visualitzar-la amb ELSA.
* La consola d’analista és molt útil per accedir de manera remota al servidor i realitzar la gestió de les alertes.
* La combinació de Ossec i Snort és molt potent i permet molta visibilitat del que està passant i dels problemes de seguretat.
* Gràcies a totes les informacions recollides per Bro i per Snort, és possible descobrir tots els passos realitzats per un atacant: màquina atacant i màquina objectiu, ports i protocols utilitzats, comandes executades, fitxers intercanviats, etc. Cal dir però que descobrir totes aquestes dades entre els molts logs, alertes, i tràfic capturat no és una tasca senzilla, si no que requereix certa pràctica i entrenament propi d’un analista de seguretat.
* En la fase de desenvolupament s’han pogut veure les tasques que caldrà realitzar en fase d’implantació per ajustar el sistema al nostre entorn

# IMPLANTACIÓ

En aquest capítol s’exposaran els passos realitzats per implantar la plataforma de seguretat Security Onion.

En el primer apartat explicarem la col·locació del sensor dins la xarxa, i els passos per fer la instal·lació de la plataforma (detallats en el [Apèndix II](#_APENDIX_II:_Fase)). Seguidament explicarem quines són les proves per verificar que el programari funciona correctament i quines són les tasques realitzades després de la instal·lació per deixar el sistema a punt.

En els següents apartats, explicarem les tasques pendents que hem identificat després de fer la instal·lació i posar la plataforma a monitoritzar tràfic.

Finalment acabarem amb les valoracions de la fase d’implantació i les futures evolucions.

## Descripció de la xarxa i col·locació del sensor

En el següent esquema (figura 5.1) es mostra on situarem el sensor, que capturarà el tràfic que circula entre el clúster de servidors virtualitzats i la resta de la xarxa, incloent usuaris, laboratoris i Internet. Situem el sensor entre el switch-core de la xarxa i el clúster de servidors.

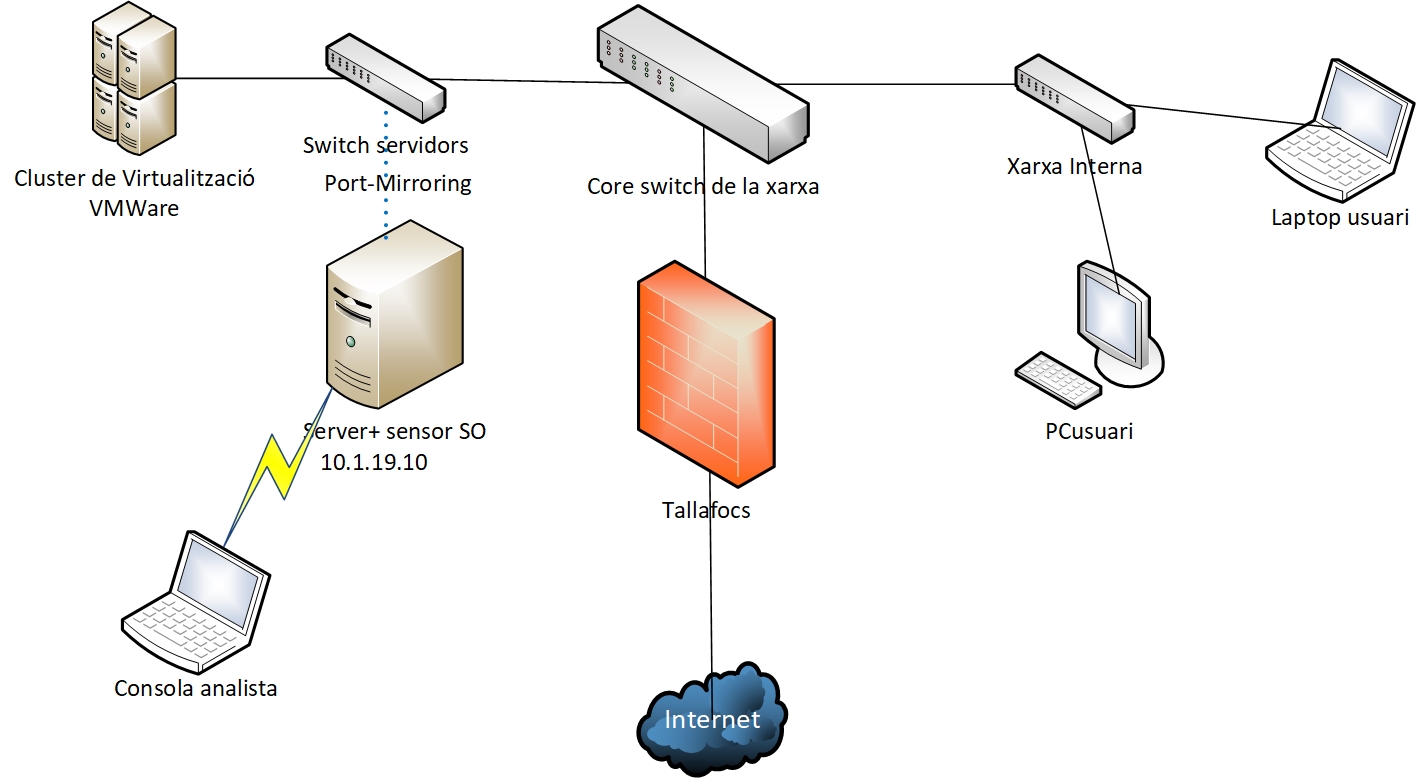


Figura 5.1 - Diagrama físic amb la posició del sensor dins la xarxa del CTTC

En el projecte es disposa d’un sol sensor amb 3 interfícies de xarxa Gigabit Ethernet per capturar tràfic. En un primer moment només configurarem una de les tres interfícies de xarxa per capturar tràfic. També tindrem una interfície de gestió.

Passem a descriure els diferents elements implicats en la posada en marxa:

***Clúster de servidors VmWare:***

* **Arquitectura:** Clúster de 3 Hosts i la corresponent cabina de discs.
* **Nombre de servidors virtuals:** Uns 40
* **IPs de les Xarxes de servidors:** 84.88.62.192/26 (VLAN 20), 84.88.63.192/26 (VLAN 21), 84.88.61.128 (VLAN 120), 10.2.1.0/24 (VLAN 22 - DMZ)
* **Tipus de servidors:** web, email, fitxers, desenvolupament, repositoris subversion, telefonia VoIP (Asterisk), Intranet, Cloud privat, SAP/ERP.
* **Sistemes operatius dels servidors:** Linux (varies distribucions), i Windows Server

***Switch de servidors:***

* **Nom:** *hiperion*
* **Arquitectura:** 3 switchos Cisco WS-C2960S-24TS-L configurats en stack
* **Connexió monitoritzada:** PortChannel 5, amb 3 ports que el connecten amb el switch-core de la xarxa (Helium) (figura 5.2)

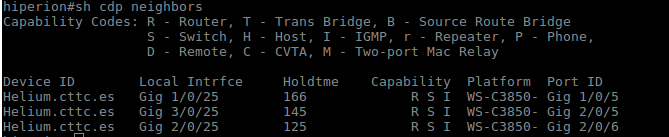


Figura 5.2 - Captura del switch Hiperion amb connexió contra el switch core Helium

* **Port de captura:** PortSpan del tràfic del PortChannel 5 cap a la interfície de captura del sensor (figura 5.3).

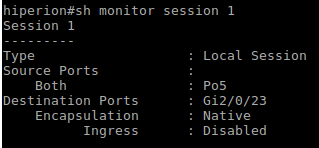


Figura 5.3 - Captura del switch amb el PortSpan

* Tràfic mig a la interfície Po5 (figura 5.4)
  + **IN** (des del switch-Core): 13,23Mbps
  + **Out** (cap al switch-core): 2,4Mbps

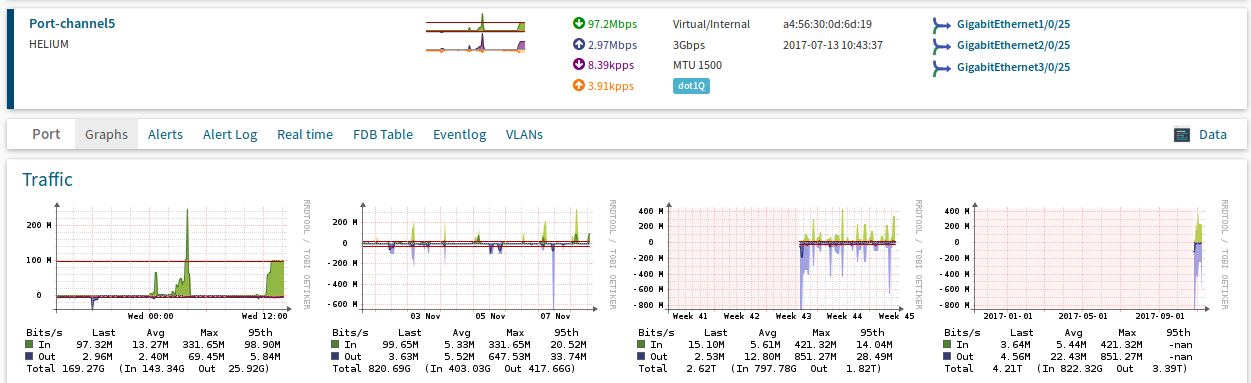


Figura 5.4 - Tràfic mig a la interfície de xarxa a monitoritzar

***Plataforma de monitorització - Servidor més sonda Security Onion:***

* **Mode d’instal·lació:** Security Onion instal·lat en un equip físic en mode *standalone* (servidor i sensor en un mateix equip).
* **Hostname:** *seconion*.
* **IP:** 10.1.19.10
* **Tràfic capturat:** Captura tot el tràfic de totes les VLANs entre el switch dels servidors (Hiperion) i el switch-core (Helium).

***Switch Core de la xarxa:***

* **Nom:** *Helium*
* **Arquitectura:** Stack de 2 switchos WS-C3850-12S
* **Xarxes connectades**
  + Xarxes de usuaris i laboratoris: 10.1.16.0/24 (usuaris de recerca), 10.1.15.0/24 (usuaris administració), 10.1.1-14.0/24 (xarxes de laboratoris)
  + Firewall i Internet

## Descripció de la Instal·lació pas a pas

La configuració escollida per la instal·lació de Security Onion en producció és:

* **Mode standalone**: aquesta configuració ens permet tenir el sensor i el servidor en la mateixa màquina. L’avantatge és que ens permet estalviar maquinari, i si tenim una màquina prou potent i una xarxa petita com la nostre, és una bona solució. No ens limita per poder escalar la solució, ja que més endavant podem afegir més sensors al servidor.
* Durant la instal·lació, podem escollir la opció d’implementar les “**best practices**”. La opció “best practices” deshabilita serveis considerats innecessaris que dupliquen feina i dades, i doncs permet tenir una configuració més optima per les nostres necessitats estàndard.
* Escollirem **Snort** com a NIDS (hem justificat la decisió en la fase de disseny).
* El set de regles que escollirem per Snort és **Emerging threats**, ja que no necessiten subscripció.

En el [Apèndix II](#_APENDIX_II:_Fase) s’expliquen quins són els passos que cal seguir per dur a terme la instal·lació.

## Consola de l’analista

Per preparar la consola d’analista, una opció és instal·lar Security Onion en una màquina virtual, però sense executar la part del setup ja que no és necessari que capturi tràfic. D’aquesta manera s’obté una màquina amb totes les eines necessàries per fer anàlisi de seguretat.

Un cop la consola d’analista instal·lada, cal donar-li accés al servidor Security Onion amb la comanda ***sudo so-allow***, opció [a], per obrir els ports del Firewall (figura 5.5).

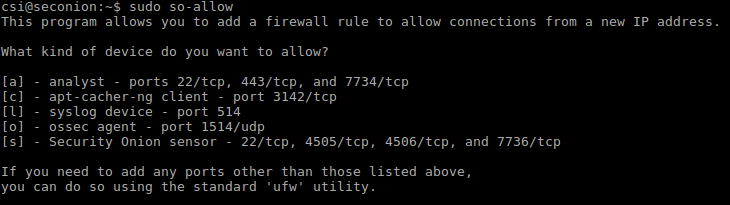


Figura 5.5 - Captura permetre consola analista

## Ajustaments i manteniment del sistema

### Ajustaments inicials del sistema

Un cop instal·lat el sistema, podem realitzar alguns ajustos:

* Enviament de emails en cas d’alertes: podem ajustar-ho segons el tipus o la gravetat de la alerta.
* Ajustar la variable daystokeep de la base de dades de Sguil per que esborri les entrades cada cert temps. Per defecte són 30 dies.
* Deshabilitar processos de sensor innecessaris
* Configurar la hora NTP del sistema.

### Manteniment diari de Security Onion

Aquestes són les tasques que cal realitzar diàriament per la correcta explotació de Security Onion:

1. Controlar la carrega del sistema, per assegurar-nos de que no s’estan descartant paquets. Usar la comanda ***sudo sostat | less*** .També podem afegir aquesta comanda en el crontab i enviar un email amb el resultat
2. Actualització regular del programari del sistema amb la comanda ***sudo soup***
3. Revisió diària de les alertes, classificant-les en categories segons el tipus (scan, força bruta, accés no autoritzat, etc.).
4. Fer els informes dels incidents de seguretat que es vagin produint (per exemple un atac que hagi tingut èxit).

Quan es detecta un incident de seguretat, cal seguir una plantilla per fer el report corresponent.

Un informe d’incident ha d’incloure com a mínim:

* Data, hora d’inici i d’acabament de l’activitat maliciosa, en UTC (GMT).
* Adreça IP i hostname de l’equip víctima
* Adreça Mac del host
* usuari si és pertinent
* Descripció de l’activitat maliciosa detectada.
* Indicadors de l’activitat maliciosa detectada (adreça IP, nom de domini, hashes dels fitxers implicats, …)
* Resum del que ha passat

### Ajustaments de la plataforma i tractament dels falsos positius

En posar en marxa la plataforma Security Onion i començar a capturar tràfic i analitzar la xarxa, apareixen gran nombre d’alertes, algunes de les quals no volem que apareguin al no comportar cap problema de seguretat. Tenim varies maneres de reduir el nombre d’alertes per només quedar-nos amb les que realment són rellevants, entre elles hem utilitzat les següents:

* Deshabilitar regles de Snort
* Modificar regles de Snort
* Crear noves regles amb Snort
* Crear Autocategories amb Sguil
* Excloure de les verificacions els fitxers que canvien sovint.

Aquest procés d’ajustament és una tasca que s’ha d’anar fent de manera continuada durant l’explotació del sistema per tal de que les alertes irrellevants no emmascarin les que mereixen atenció.

En el [Apèndix III](#_APENDIX_III:_Posada)- Posada a punt i operació es descriu com s’ha realitzat la posta a punt de la plataforma i el tractament dels falsos positius i les alertes no importants.

## Tasques pendents per després de la implantació

Un cop la plataforma de producció està instal·lada i es comença a monitoritzar la xarxa, queden pendents algunes tasques, com ja s’ha anticipat en la fase de desenvolupament.

### Equip i formació de l’equip de seguretat

Serà necessària la formació d’un equip de seguretat, on les persones designades tinguin rols ben definits:

* Designació d’un responsable de seguretat, que tindrà les següents responsabilitats
  + conèixer el funcionament de la plataforma de seguretat
  + definició dels procediments d’us de la plataforma
  + Comprovar i monitoritzar el correcte funcionament del sistema
  + Realitzar contínuament l’ajust del sistema per que tingui un rendiment òptim
  + Donat un incident de seguretat, decidir les accions a emprendre per corregir-lo.
* Membres de l’equip de seguretat, que tindran les següents funcions
  + Analitzar les alertes de la plataforma de seguretat
  + conèixer el funcionament de la plataforma de seguretat.
  + Identificar i classificar les alertes per categories.
  + Investigar els d’incidents de seguretat
  + redacció d’informes per els incidents de seguretat

### Contingències i recuperació

Cal elaborar un pla de contingències i recuperació de la plataforma de seguretat. Les tasques principals seran:

* Fer backup diari dels arxius de configuració de la plataforma de seguretat
* Fer backup de la base de dades de la plataforma de seguretat

### Tasques de consolidació i explotació de la plataforma

Algunes de les tasques més importants que s’aniran realitzant per fer evolucionar la plataforma són:

* Fer el desplegament d’agents OSSEC en els servidors.
* Desplegar OSSEC agent-less, de manera que equips on no es pugui instal·lar l’agent OSSEC enviïn els seus logs a la plataforma Security Onion per que puguin ser analitzats. Es tracta típicament d’equips de xarxa com switchos i tallafocs.
* Usar Squert per realitzar de manera periòdica un informe executiu sobre les amenaces de seguretat al CTTC, per mostrar-lo a la direcció del centre.
* Continuar ajustant el sistema per aconseguir reduir el nombre d’alertes a descartar: modificar regles, ajustar configuracions, desactivar conjunts de regles, etc.
* Fer decoders i regles específiques de OSSEC per monitoritzar algunes aplicacions pròpies, com són el sistema de backup i la plataforma de gestió administrativa.
* Connectar les altres dues interfícies GigaEthernet del sensor que encara no s’estan utilitzant:
  + Amb la segona interfície, monitoritzar el tràfic entre el Firewall i el core-switch, ja que per aquest enllaç passa tot el tràfic que va i ve d’ Internet.
  + Amb la tercera interfície, monitoritzar el tràfic entre el core-switch i un switch d’usuaris, el que tingui més tràfic. D’aquesta manera podrem monitoritzar el tràfic intercanviat dins la xarxa d’usuaris.
* Entrenar-se, com a analista de seguretat per detectar atacs, tant amb atacs reals com amb els exemples didàctics disponibles a Internet.

# RESULTATS I VALORACIONS

En aquest capítol presentem els resultats obtinguts durant l’explotació de la plataforma de seguretat Security Onion, un cop instal·lada. També valorarem quines són les tasques i millores referents a la seguretat informàtica que hem inferit a partir de la implantació de la plataforma, enumerarem quines creiem que són les evolucions futures de la plataforma de seguretat a implementar al CTTC, i inclourem una valoració econòmica del projecte.

La plataforma de seguretat es va connectar a la xarxa de producció el dia 7 de novembre. El resultats obtinguts comprenen un període de temps d’unes tres setmanes, durant les quals s’han anat fent ajustaments al sistema per reduir el nombre de falsos positius.

## Deteccions i alertes rebudes

### Detecció d’aplicacions vulnerables

Un cop en la fase d’explotació del sistema, analitzem les alertes del sistema per avaluar l’estat de la xarxa, els sistemes monitoritzats i les vulnerabilitats detectades.

A continuació enumerem algunes de les vulnerabilitats en els nostres sistemes que la plataforma de seguretat ens ha permès descobrir:

* **Equips amb versions antigues i vulnerables:** la regla amb **sid 2019401:20** “et policy vulnerable java version 1.8.x detected” ens avisa en el cas de que detecti clients amb una versió del client java vulnerable.
* **Equips amb protocols d’encriptació vulnerables:** La regla snort amb **sid: 2019415:3** “ET POLICY SSLv3 Outbound Connection From Client Vulnerable To POODLE Attack” ha permès detectar alguns servidors web tenien SSLv3 activat per defecte. Aquest fet pot comportar perill de robatori de dades xifrades amb un atac MItM (Man in the Middle).

### Detecció d’errors de configuració

A partir de l’anàlisi de les alertes de Security Onion, s’han pogut determinar errors de configuració que afecten a la seguretat.

* **Ports oberts en el Firewall no necessaris**: a partir de l’anàlisi dels accessos i de les alertes per intents d’atac, s’han detectat alguns ports i accessos oberts en el Firewall perimetral que no eren necessaris i que podrien estar tancats.
* **Contrasenyes que viatgen en clar:** Algunes alertes ens han fet adonar-nos que, dins la xarxa interna, les contrasenyes d’usuari per validar-se en algunes aplicacions viatgen en clar. Això constitueix un error de configuració i no és recomanable ja que podria permetre a un atacant obtenir les contrasenyes si aconseguís capturar aquest tràfic. Algunes de les regles de Snort que ens avisen d’aquests casos són SID 2012483, 2006402:9 i 2006380:12
* **Accessos SNMP amb el password per defecte en clar**: algunes impressores tenen activat el protocol SNMP amb password en clar, i Snort ens ha avisat i ens ha permès resoldre el problema. Regles de Snort amb SID 36101411.
* **Arxius amb els permisos incorrectes:** OSSEC ens ajuda a detectar si tenim arxius que tenen els permisos incorrectes i amb més privilegis dels recomanats, i ens permet corregir-ho.

### Polítiques de seguretat

Les alertes de la plataforma ens poden ajudar a detectar alguns incompliments de les polítiques de seguretat i d’us de les tecnologies d’informació del centre, i ajudar-nos en el seu compliment:

* **Us de Dropbox**: les regles amb SID 2012648:3 i 2020565 ens indiquen si usuaris estan usant Dropbox dins el centre.
* **Us de TeamViewer**: les regles amb SID 2009475:8 ens indiquen si usuaris estan usant TeamViewer dins el centre
* **Us de la xarxa d’Internet anònima I2P**: la regla amb ID 2019988:2 ens indica accessos a la xarxa d’Internet Invisible Project.
* **Accés a pàgines web amb possible contingut inapropiat**: Accés a dominis que poden contenir pirateria ja que tenen baixa reputació, per exemple la regla amb SID 2016778:4
* **Us de programari no aprovat**: per exemple ens pot avisar d’usuaris que usin navegadors de la xarxa TOR.

### Principals tipus d’atacs detectats i alertes

La instal·lació de la plataforma de seguretat ens permet veure el tipus d’atacs més comuns als que ha estat sotmesa la nostre xarxa en l’espai de temps de tres setmanes que fa que està en explotació. El tipus d’atacs detectats van dirigits als serveis que publica el CTTC a Internet, sobretot a servidors que contenen pàgines web, que en la majoria de casos estan fetes amb Wordpress o Joomla.

Cal remarcar que en el espai de temps estudiat no s’han detectat atacs des de dintre de la xarxa.

#### Atacs de força bruta contra servidors Wordpress

En la següent gràfica podem veure els atacs de força bruta rebuts per els servidors que contenen llocs web de Wordpress, principalment 10.2.1.15 (www.cttc.es), i 84.88.62.226 (pàgines web de projectes). Aquests atacs consisteixen en intentar registrar-se en el servidor provant diferents contrasenyes. La següent gràfica mostra els atacs d’aquest tipus rebuts en el període de temps del dia 8/11/2017 al 29/11/2017 (figura 6.1). En aquest període la web del centre ha rebut de l’ordre de 26.000 atacs d’aquest tipus.

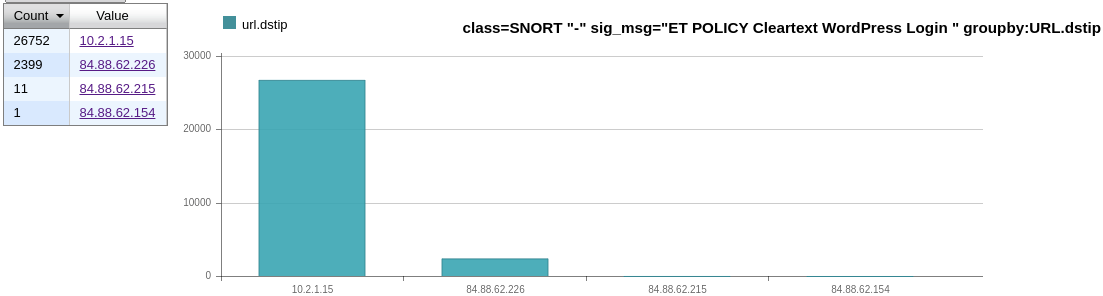


Figura 6.1 - Captura de Elsa amb els atacs de força bruta per servidor

#### Atacs a aplicacions web

Aquest tipus d’atac, el segon més comú que ens hem trobat, és el que intenta aprofitar possibles vulnerabilitats en aplicacions web amb PHP.

***Atac contra xmlrpc.php***

Dintre d’aquesta categoria, el més comú és l’intent d’atac contra *xmlrpc.php*. Aquest atac pretén forçar la contrasenya del servidor per força bruta, provocant en molts casos de passada una denegació de servei en l’equip atacat.

En la següent figura podem veure les peticions a l’arxiu *xmlrpc.php* en el període de temps del 8/11/2017 al 21/11/2017 (figura 6.2).

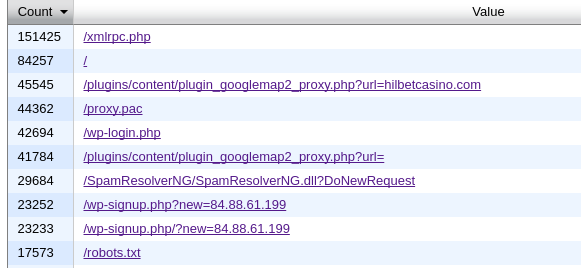


Figura 6.2 - Captura de ELSA amb els intents d'accés a l’arxiu xmlrpc.php

***Atac contra el plugin de Joomla googlemap2***



Veiem també l’atac al plugin de Joomla plugin\_googlemap2\_proxy.php, que es va produir entre els dies 7/11/2017 i 8/11/2017 (figura 6.3). En aquest cas l’atac va tenir èxit, i l’atacant va poder usar la vulnerabilitat en aquest plugin per fer atacs de DDoS usant com a relay proxy el servidor atacat. El plugin estava desactivat, però tot i així era explotable.

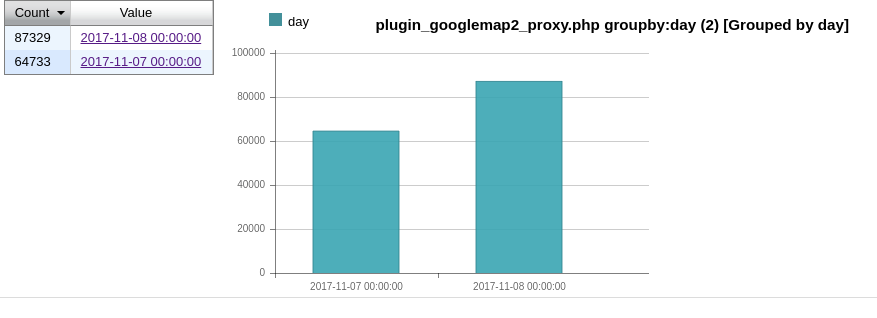


Figura 6.3 - Captura de ELSA amb els accessos al plugin vulnerable ordenats per dia

***Nombre d’atacs d’aplicacions web als servidors***

La següent gràfica ens dona un idea de quins són els servidors que han rebut més intents d’atacs d’aplicacions web en el període del 8/11/2017 al 24/11/2017, on veiem les respostes dels nostres servidors a peticions *uri* prohibides (figura 6.4). El més atacat en aquest cas és el servidor 84.88.62.226 que conté nombroses pàgines web Joomla i Wordpress.

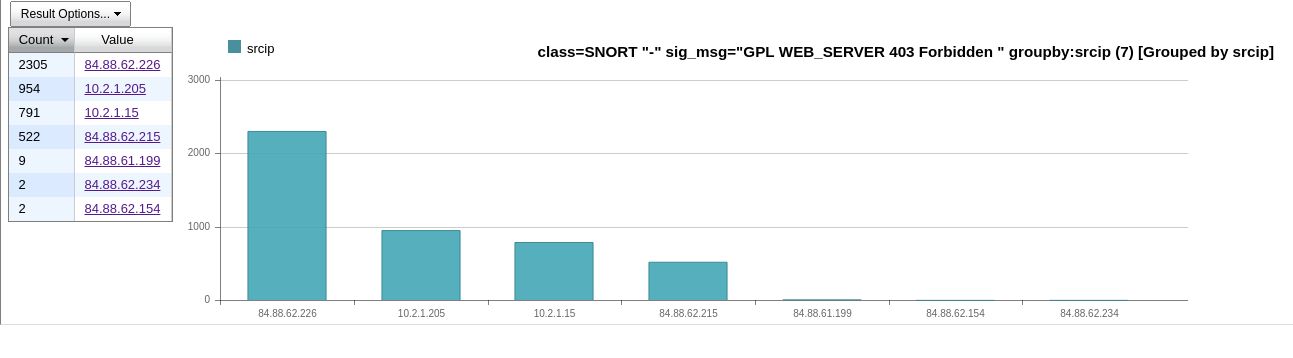


Figura 6.4 - Captura de ELSA amb els atacs a aplicacions web ordenats per servidor

I en la següent gràfica podem veure la distribució d’aquesta atacs al llarg del temps (figura 6.5)

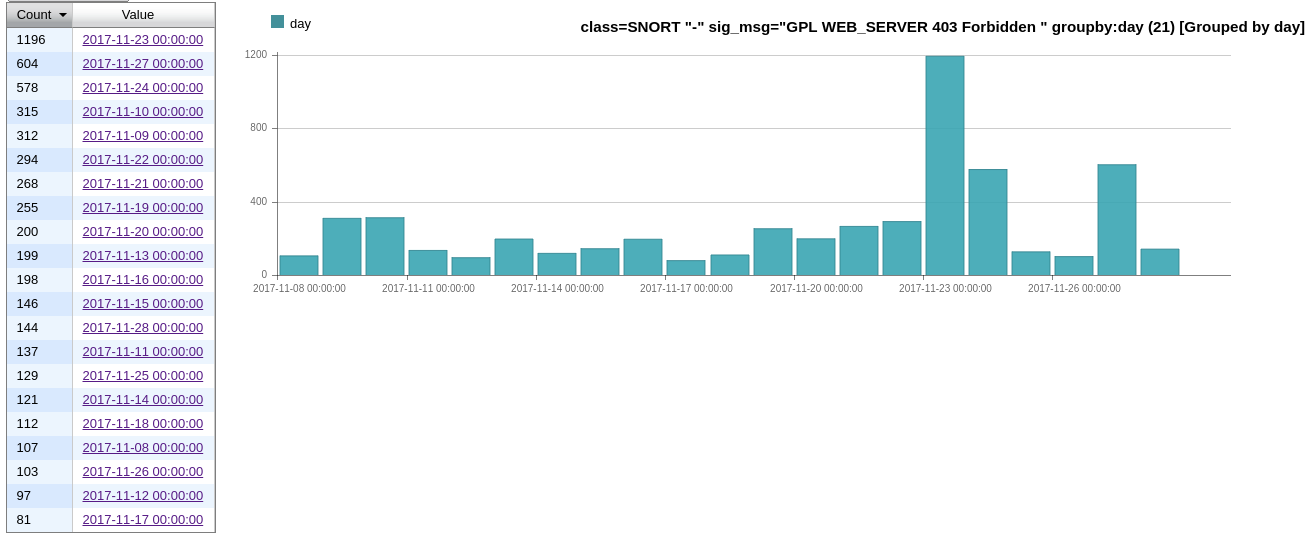


Figura 6.5 - Captura de ELSA amb els atacs a servidors per dia

#### Atacs d’injecció SQL

S’han pogut detectar també atacs d’injecció SQL, on l’atacant intenta executar codi SQL per tal de poder manipular la base de dades del servidor de l’aplicació (figura 6.6).

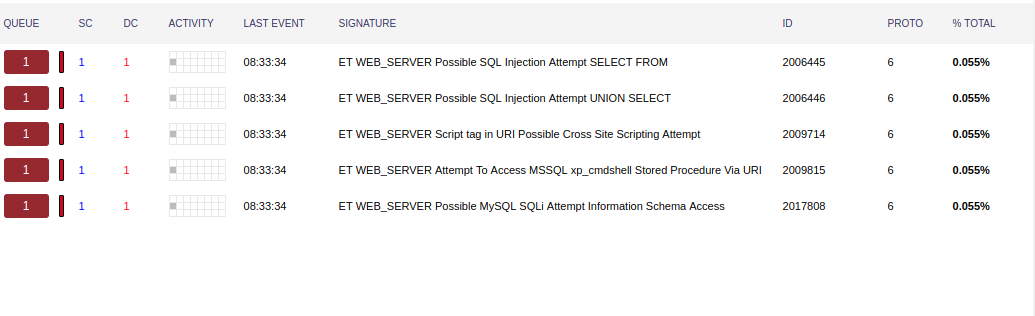


Figura 6.6 - Captura Squert amb exemples d'atacs SQLi

Possibles atacs d’injecció SQL agrupats per llocs web del 8/11/2017 al 28/11/2017 (figura 6.7)

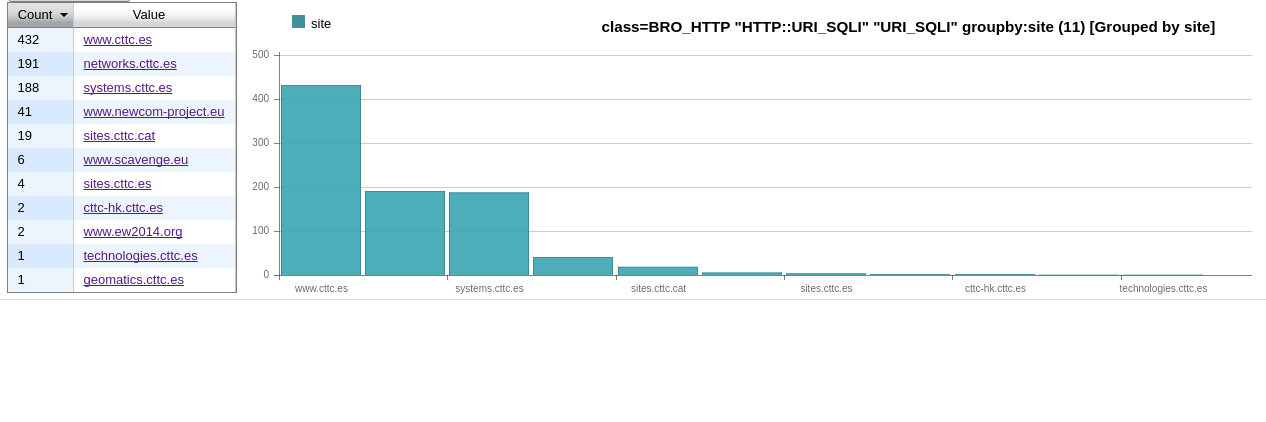


Figura 6.7 - Captura ELSA amb atacs SQLi agrupats per lloc web

### Estadístiques d’alertes

A continuació presentem algunes estadístiques recollides durant el període en que s’ha dut a terme l’explotació de la plataforma de seguretat.

* Total d’alertes generades durant la setmana del 20/11 al 26/11, classificades segons la seva prioritat (figura 6.8).

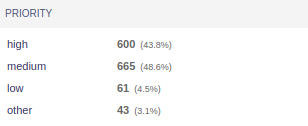
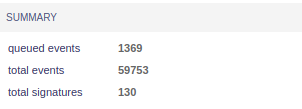


Figura 6.8 - Alertes en el període d’una setmana per prioritat

* Classificació de les alertes durant una setmana (del 20/11/2017 al 26/11/2017) (figura 6.9). Quasi el 50% són intents d’accés no autoritzat a servidors web. Un 42% de les alertes són falsos positius, però ha mesura que avanci el temps i es vagi ajustant millor la plataforma aquesta xifra haurà de disminuir.

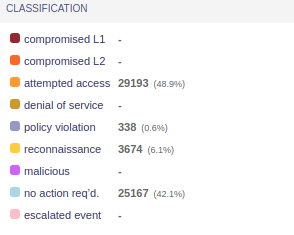


Figura 6.9 - Classificació d'alertes segons categoria

* Alertes amb més aparicions en el període de una setmana (del 20/11/2017 al 26/11/2017) (figura 6.10). Veiem que la majoria d’alertes són per intents d’accés per força bruta a servidors Wordpress. Aquest tipus d’atacs s’han pogut reduir instal·lant el Firewall d’aplicacions *mod-security, mod-evasive* i l’aplicació *fail2ban*. La segona alerta de OSSEC és un fals positiu i es pot descartar. La tercera alerta també fa referència a l’intent d’explotar vulnerabilitats en servidors web.

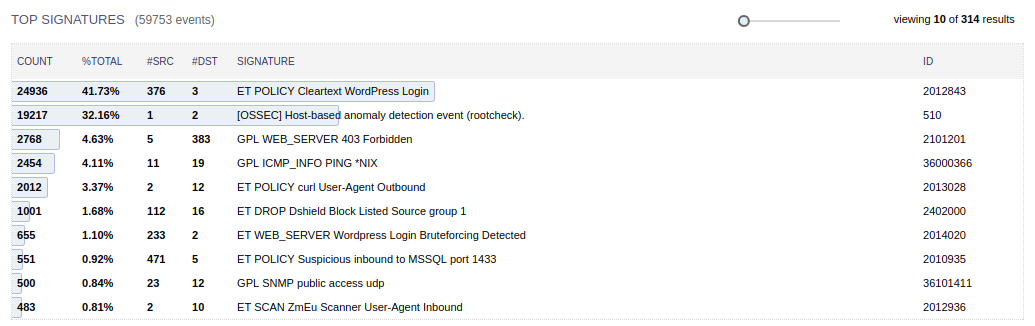


Figura 6.10 - Alertes amb més aparicions durant el període de una setmana

* Top IPs de origen/destí de les alertes d’atacs en el període de una setmana (figura 6.11). Si descartem les 4 primeres que són internes i no corresponen a atacs, la IP que ha realitzat més atacs és 123.206.197.121 que ve de Xina. Per altre banda podem veure que el servidor que ha rebut més atacs és el servidor que conté la pàgina web corporativa, www.cttc.es.

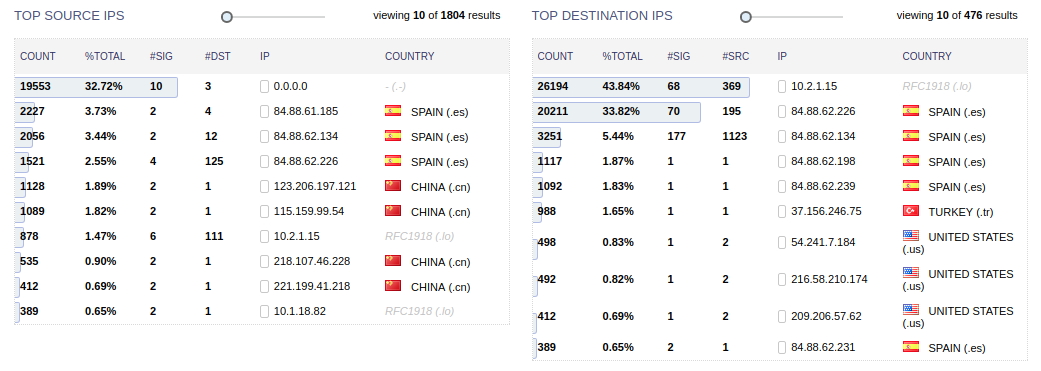


Figura 6.11 - Top IPs origen/destí dels atacs

* El tipus d’atacs que ha fet la IP 123.206.197.121 són de força bruta contra la nostre pàgina web corporativa, com podem veure a la següent figura (figura 6.12).

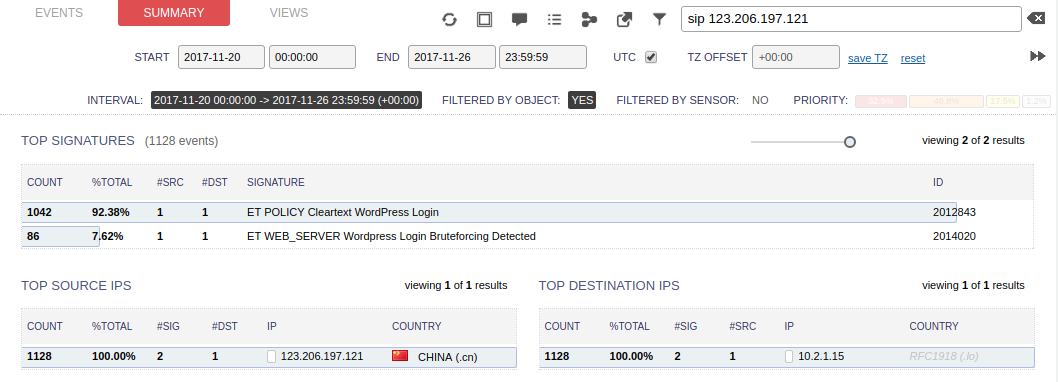


Figura 6.12 - Captura Squert amb el tipus d'atacs realitzats per la IP 123.206.197.121

* En la següent gràfica podem veure els ports més atacats, i els ports d’origen dels atacs més comuns. El port més atacat és el TCP/80 (http) (figura 6.13).

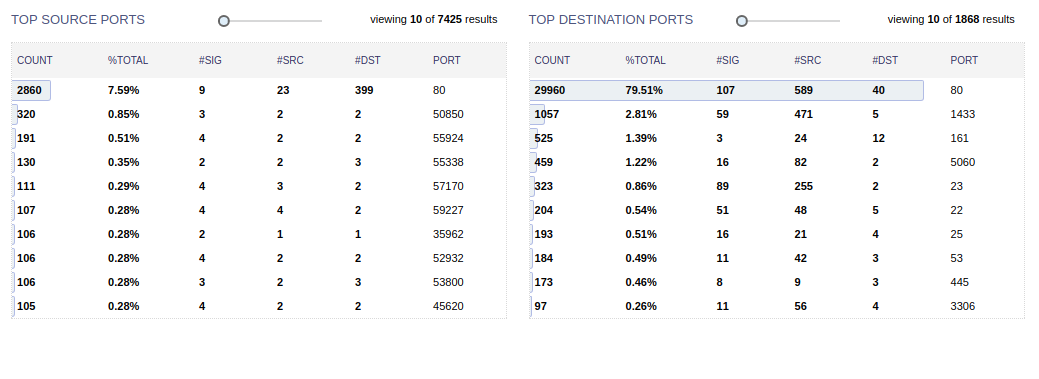


Figura 6.13 - Top de ports origen/destí dels atacs

* En la següent gràfica podem veure la classificació dels atacs segons el seu país d’origen (figura 6.14). Xina és el país d’on provenen la majoria dels atacs.

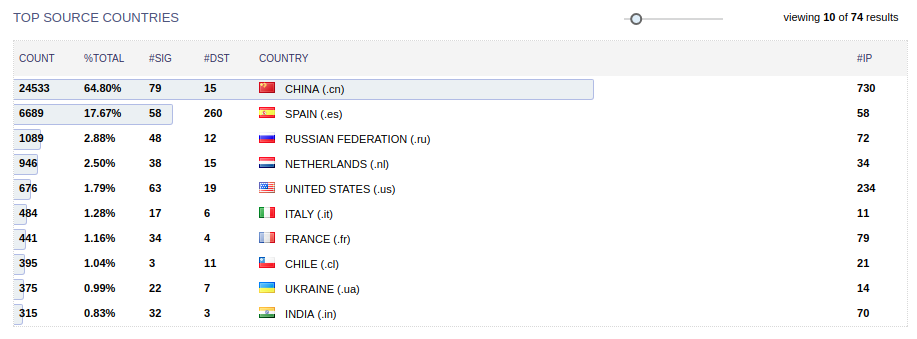


Figura 6.14 - Classificació d'atacs per països

## Millores en la seguretat informàtica al CTTC gràcies a la implantació de la plataforma

Després d’analitzar els primers resultats obtinguts de l’explotació del sistema, s’han detectat algunes accions necessàries i urgents a realitzar, que exposem a continuació.

### Enfortir la seguretat dels servidors web que reben més atacs

S’ha detectat que el servidor web del centre, basat en Wordpress, i el servidor web de projectes, que incorpora pàgines web amb Joomla, Wordpress, i alguna Wiki, reben contínuament intents d’atac de força bruta, d’injecció SQL o d’intents d’aprofitar vulnerabilitats amb el PHP. Es fa necessari enfortir-ne la seguretat.

Per fer-los més segurs s’utilitzen les següents estratègies:

* Activar els mòduls *mod-security* i *mod-evasive* d’apache. Mod-security és un Firewall d’aplicacions web que ajuda a detectar i rebutjar intents d’atac. Mod-evasive permet contrarestar atacs de DDoS.
* Activar Fail2ban i resposta activa amb OSSEC per tal de posar remei als atacs de força bruta.
* *htacces* per restringir accés a alguns arxius, com per exemple el xmlrpc.php que rep nombrosos atacs, o wp-login.php dels servidors Wordpress.

Després d’enfortir la seguretat dels servidors hem notat una disminució de la intensitat dels atacs des del dia 23/11 ja que al cap de pocs intents els atacants són rebutjats (figura 6.15).

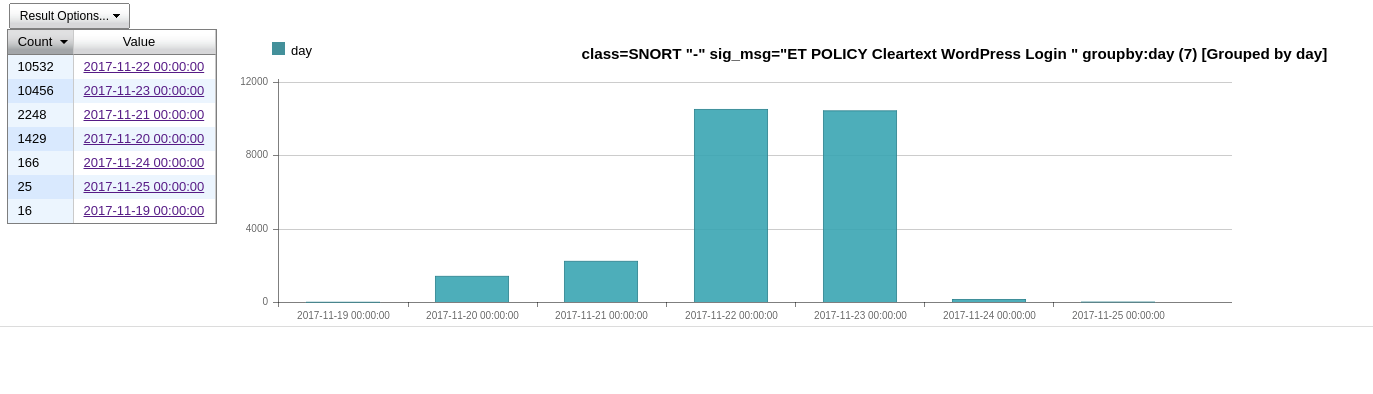


Figura 6.15 - Captura de ELSA que mostra l'evolució dels atacs de força bruta en els servidors web

### Actualització dels equips, serveis i plugins

Es important tenir les aplicacions web actualitzades, i la plataforma de seguretat ens ha ajudat a detectar possibles problemes de seguretat derivats d’aquest fet.

* Els servidors més desactualitzats són en general els que tenen més intents d’atac, ja que és més probable detectar-hi una vulnerabilitat explotable. Així la plataforma ens permet centrar el focus d’atenció on hi han més problemes a resoldre.
* De vegades alguns serveis no es poden actualitzar degut a que tenen dependències amb programes antics i poden deixar de funcionar. En aquests casos, i mentre no es puguin migrar, s’ha hagut d’enfortir-ne la seguretat per altres mitjans, com per exemple tallafocs d’aplicacions com mod-security.
* Algunes aplicacions web no actualitzades ni actualitzables de vegades han arribat al final de la seva vida útil i és més efectiu desactivar-les.
* Hem pogut detectar alguns plugins de Joomla vulnerables, com googlemap2, que hem procedit a desinstal·lar del sistema.
* Hem detectat alguns serveis com HTTPS que usaven serveis de xifrat insegurs o poc robustos, com SSLv3, vulnerable a atacs tipus “Poodle”, o RC4, també insegur. Hem desactivat aquests protocols.
* Hem esborrat plugins de Joomla i Wordpress que no estan en ús.
* Ens ha avisat de clients amb versions de Java vulnerables, que hem procedit a actualitzar.

### Evitar que les contrasenyes viatgin en clar

La plataforma de seguretat ens ha servit per adonar-nos que alguns serveis i aplicacions web estan usant contrasenyes que viatgen en clar per la xarxa. Es el cas per exemple d’alguns repositoris de fitxers (subversion). Un atacant intern que pogués capturar el tràfic de xarxa podria descobrir aquestes contrasenyes. Ens cal activar SSL en tots aquests serveis.

## Valoracions del programari Security Onion implantat

En la fase d’implantació hem instal·lat la plataforma de seguretat Security Onion, hem començat a monitoritzar una part de la xarxa i hem pogut començar a interactuar i treballar amb el sistema.

Les primeres valoracions durant aquesta fase són:

* El sistema s’instal·la amb facilitat i totes les eines dins la solució s’integren correctament
* El sistema és estable i funciona com s’espera
* En connectar el sensor a una xarxa en producció i amb un nivell de tràfic important, el nombre d’alertes que apareix és molt alt i pot desbordar-nos una mica. Al principi cal una tasca bastant intensiva d’ajust del sistema per reduir el nombre d’alertes a tractar.
* També en un principi, com no estem entrenats, no és fàcil distingir les alertes importants de les que no ho són.
* La modificació i gestió de les regles de snort és complexa i no massa amigable, al menys en un primer moment.

Respecte a la detecció d’atacs i intrusions, la nostre experiència ha sigut la següent:

* Al poc de posar en marxa la plataforma, un dels nostres servidors web ha patit un atac amb èxit. El servidor contenia un web Joomla que tenia un plugin de googlemaps (plugin\_googlemap2) vulnerable, i que tot i que estava deshabilitat, permetia a un atacant usar el servidor com a Open Proxy i llençar atacs de denegació de servei contra altres webs.
* Hem pogut detectar l’atac ja que el servidor deixava de respondre (estava saturat), i mirant els logs d’apache hem pogut detectar el problema i corregir-lo. Snort, però, no ha fet saltar cap alerta.
* Mirant les regles de Snort, hem vist que la regla existeix (sid:2023574, "ET WEB\_SPECIFIC\_APPS Joomla GoogleMaps Plugin Open Proxy Access"). Examinant l’activitat de la IP de l’atacant amb ELSA, hem comprovat que Bro sí ha processat el tràfic implicat en l’atac i ha detectat les connexions i el tràfic http.
* La raó per la que suposem que snort no ha detectat el problema està en la construcció de la regla, que inclou la variable $HTTP\_SERVERS i que pensem que potser no ha sigut interpretada correctament en no estar encara explícitament definida en el nostre entorn.
* Veiem doncs que cal ajustar el comportament de snort per adaptar-lo a la nostre xarxa, i que sempre pot deixar escapar algunes alertes.
* Tot i això destaquem el sistema té un alt nivell de detecció i en poca estona ens mostra tot el que està passant per la nostre xarxa.

## Evolucions futures

Les possibles evolucions futures de la nostre instal·lació de Security Onion són:

* Instal·lar equips que tinguin funcionalitats de IPS, com per exemple instal·lar un Snort en línia per protegir els servidors web més vulnerables, de manera que pugui no només alertar si no parar atacs en temps real.
* Activació de la integració de Bro amb la base de dades de Team Cymru per poder ser capaç de detectar fitxers maliciosos referenciats.
* Integrar Security Onion amb les eines d’anàlisi i visualització Elasticsearch per augmentar les funcionalitats de intel·ligència i visualització de la plataforma.

Les possibles coses a millorar i funcionalitats addicionals que seria interessant que incorporés el programari en el futur són:

* Millorar de les funcionalitats de “autocategoria” de Sguil: Un cop creades les categories amb la interfície de Sguil, es poden desactivar, però no modificar ni esborrar. Per fer-ho s’ha d’anar a la base de dades de Sguil i seguir un procediment i no és pràctic.
* L’estabilitat del programa Sguil es pot millorar ja que sembla congelar-se en moments puntuals si ha de tractar moltes alertes a la vegada.
* Millorar les opcions de *reporting* per poder realitzar diagrames de manera més àgil amb Squert.
* Afegir funcionalitats de realització de informes periòdics d’incidents de seguretat de manera automatitzada.
* *Vulnerabilities assesment*: Avisos automàtics de les vulnerabilitats dels nostres equips i aplicacions.

## Valoració econòmica

El projecte és econòmic, i es pot tenir desplegada la plataforma de seguretat en molt poc temps.

A continuació s’enumeren els costos aproximats del projecte si és fa per part de personal expert acostumat a treballar amb Security Onion i fer aquest tipus de projectes:

* Preu del programari Security Onion: llicència GNU GPLv2 sense cost
* Preu del maquinari escollit: 800€
* Fase de disseny: 12 hores
* Fase d’implantació: 16 hores
* Fase principal de l’explotació: la posada a punt del sistema i la depuració de les alertes: 8 hores
* Formació per la operació i entrega de procediments: 4 hores
* Total hores: 40 hores
* Total cost humà: 4000€ (cost per hora estimat de 50€) taula 1

|  |  |
| --- | --- |
| Cost material | 800€ |
| Cost humà | 4000€ |
| **Cost econòmic del projecte** | **4800€** |

# CONCLUSIONS

La implantació de Security Onion ens ha permès d’una manera relativament ràpida i senzilla disposar d’una plataforma de seguretat NIDS/HIDS per monitoritzar la xarxa, els equips i els serveis informàtics oferts. Al mateix temps també ens proporciona les eines per la visualització, gestió i anàlisi de les alertes de seguretat generades.

Disposar d’aquesta plataforma ens ha permès:

* Detectar intents de Hacking de servidors
* Detectar servidors compromesos
* Detectar vulnerabilitats en servidors, serveis i equips
* Detectar errors de configuració que puguin comprometre la seguretat dels servidors
* Detectar fluxos de dades sospitosos o poc habituals
* Tenir alertes en el cas d’accessos no autoritzats o robatoris de dades per part d’usuaris maliciosos.
* Impedir o mitigar els efectes dels atacs de ransomware i malware
* Detectar i parar intents d’atac per força bruta
* En cas d’atac amb èxit, tenir eines d’anàlisi per poder determinar que ha passat i l’abast del problema.
* Poder tenir informació de manera senzilla sobre el tipus d’atacs rebuts, el equips més atacats, l’origen dels atacs, i obtenir una visibilitat del que està passant a la nostre xarxa per poder protegir-la millor.

Un cop entès el funcionament dels diferents elements que composen la plataforma, i com interactuen entre ells, la implantació del sistema no és complicada. La documentació existent en la web del projecte és detallada i està actualitzada, i existeixen nombrosos tutorials a Internet. El programa està prou madur i en el nostre cas no hem de mencionar cap dificultat important durant la posada en marxa

Un cop el sistema es connecta per primer cop a la xarxa en producció i comença a analitzar el tràfic, la quantitat d’alertes detectades és molt gran i pot arribar a ser desbordant. Cal una fase inicial de posada a punt del sistema, desactivant i modificant regles, que pot arribar a consumir bastant de temps. Aquesta fase però és indispensable per poder començar a analitzar de manera efectiva el tipus de tràfic que es produeix a la nostre xarxa.

La gestió de les alertes de seguretat, la seva organització en categories, la investigació dels atacs i la realització dels informes corresponents és una tasca que s’ha de fer diàriament i que consumeix temps. Cal preveure dedicar els recursos de personal suficients per poder realitzar aquesta treball. També cal preveure formació en seguretat informàtica per poder ser capaç d’analitzar i valorar les alertes de seguretat correctament.

Les principals problemàtiques detectades amb la plataforma són les següents:

* No sempre es pot analitzar tot el tràfic de la xarxa, hi ha sectors que se’ns poden escapar.
* Per la manca de recursos, no sempre es poden analitzar totes les alertes, i poden haver-hi atacs que se’ns passin per alt.
* Poden haver-hi atacs que no es detectin per que el sistema no disposa de les signatures corresponents.

# GLOSSARI

**ACL:**  La llista de control d’accés és una llista de permisos associats a un objecte. Per exemple pot ser una llista de adreces IP que tenen accés a un recurs.

**Adreça IP:** Nombre que identifica un dispositiu connectat a una xarxa.

**Asterisk:** Programari que fa una implementació d’una centraleta telefònica i que permet treballar amb VoIP (veu sobre IP).

**Atac per força bruta:** Atac basat en provar moltes contrasenyes en un sistema informàtic per tractar d’endevinar la correcte.

**Checksum MD5:** Técnica que permet garantir la integritat de les dades mitjançant l’aplicació d’una funció criptogràfica a les dades origen per extreure’n una empremta.

**Clúster de virtualització VmWare:** Conjunt de dos o més Host on gràcies al programari VMWare podem assignar recursos i crear màquines virtuals.

**Cross-site scripting:** (XXS) és una tècnica d’atac que aprofita una aplicació web vulnerable per insertar scripts que s’executaran en la màquina del client.

**DDoS:** Atac informàtic en el que l’atacant usa moltes adreces IP per encaminar tràfic cap a la víctima i desbordar la seva capacitat.

**Defacing:** Atac en una pàgina web que en modifica l’aparença.

**DMZ:** la zona desmilitaritzada és una subxarxa d’àrea local situada entre la xarxa privada interna d’una organització i Internet.

**Domini de Broadcast:** Divisió lògica d’una xarxa d’ordinadors, on tots els ordinadors es poden accedir dins la capa data link, és a dir sense necessitat d’encaminador/router.

**Eduroam:** (Education ROAMing) és una iniciativa de l'associació de xarxes acadèmiques i de recerca europees, TERENA que permet a qualsevol usuari d'una institució participant tenir accés a Internet en qualsevol de la resta d'institucions que participen en la iniciativa.

**Fail2ban:** Programari de prevenció d’intrusions que protegeix un equip d’atacs de força bruta.

**Fals positiu:** similar a falsa alarma, és una alerta que indica un problema de seguretat quan en realitat no n’hi ha cap.

**FTP:** (File Transfer Protocol) Protocol de xarxa per l’enviament de fitxers entre ordinadors

**Hacking:** conjunt de tècniques informàtiques destinades a posar a prova o vulnerar la seguretat d’un sistema informàtic.

**HIDS:** Sistema de detecció de intrusos en un Host

**HTTP:** Hypertext Transfer Protocol es un protocol d’aplicació fonamental per les comunicacions de dades a la World Wide Web.

**I2P:** Invisible Internet Project permet publicar continguts a Internet de forma anònima per tal de resistir a la censura.

**IDS:** Sistema de detecció d’intrusions.

**IPS:** Sistema de prevenció d’intrusions. No només detecta, si no que para l’atac.

**Iptables:** Tallafocs integrat en el kernel de Linux d’un equips.

**ISO:** tipus de fitxer que conté les dades contingudes en un disc òptic. Equivalent en fitxer d’un disc d’instal·lació.

**ISP:** Proveïdor de serveis d’Internet.

**Joomla:** Sistema de gestió de continguts que permet publicar continguts a Internet.

**Llista blanca:** Llista d’elements autoritzats per defecte, sense ser sotmesos a controls de seguretat.

**Llista negra:** Llista d’elements denegats per defecte ja que es consideren maliciosos.

**Macro de MsOffice:** Seguit de comandes i instruccions que permet realitzar una tasca automatitzada. Una macro maliciosa en MsOffice es pot utilitzar per infectar un equip.

**Malware:** Programari maliciós, defineix qualsevol tipus de programari hostil o intrusiu, tal com virus, spyware, adware, trojan, ransomware, etc.

**Mode promiscu:** mode en que una targeta de xarxa processa la totalitat del tràfic que rep, i no només el que li va destinat.

**Nextcloud/OwnCloud:** Programari client/servidor que permet crear un servei de hosting de fitxers en cloud privat.

**NIDS:** Sistema de detecció d’intrusos de xarxa

**Phising:** Intent d’aconseguir informació sensible, típicament dades d’usuari o de targetes de crèdit, mitjançant la suplantació d’una entitat de confiança

**Pla de seguretat:** conjunt de mesures preventives i reactives en una organització per tal de garantir la seguretat de la informació.

**Port-mirroring:** El port mirroring s’utilitza en un switch per replicar el tràfic d’un port del switch a un altre.

**Ransomware:** Tipus de programari maliciós que amenaça de bloquejar les dades de la víctima en perpetuïtat a no ser que es pagui un rescat.

**Rootkit:** Conjunt de tècniques que un programa maliciós que ha infectat un equip utilitza per ocultar-se i no poder ser detectat.

**Rsync:** Utilitat de transferència i sincronització de fitxers entre ordinadors.

**SMTP:** (Simple Mail Transfer Protocol) Protocol de xarxa per l’intercanvi de missatges de correu electrònic entre ordinadors

**Sniffer de paquets:** programa que permet capturar el tràfic que passa per una xarxa.

**SQLinjection:** Técnica d’injecció de codi usada per atacar aplicacions informàtiques basades en bases de dades.

**Subversion:** Programari de control de versions de codi lliure.

**Switch-nivell 3:** Switch que també té capacitats de treballar com a encaminador/router.

**Switch-core:** Switch principal de la xarxa per on passen totes les connexions troncals.

**Team Cymru:** Organització de recerca dedicada a millorar la seguretat d’Internet. Proveu una base de dades d’empremtes d’arxius reconeguts com maliciosos (malware hash registry)

**Unified2:** Format de fitxer de tipus binari usat per Snort

**VLAN:** Virtual LAN, permet crear xarxes lògiques independents dins d’una mateixa xarxa física. Permeten segmentar la xarxa.

**VPN:** Una xarxa privada virtual permet una connexió remota segura entre dos xarxes privades però a través d’una xarxa pública.

**Wordpress:** Sistema de gestió de continguts que permet publicar continguts a Internet.

# BIBLIOGRAFIA

Aanval. (2016). *Snort vs Suricata*. Obtenido de http://wiki.aanval.com/wiki/Snort\_vs\_Suricata

Bezborodov, S. (6 de Mayo de 2016). Intrusion Detection System and Intrusion Prevention System with Snort provides by Security Onion - Bachelor Thesis. *Intrusion Detection System and Intrusion Prevention System with Snort provided by Security Onion*. Mikkeli, Finlandia: MAMK - University of Applied Sciences. Obtenido de http://www.theseus.fi/bitstream/handle/10024/109245/Snort+Thesis+Bezborodov+Sergey.pdf;jsessionid=D691F812C821F9D14C551F52D2C0A319?sequence=1

Brad. (2017). *Blog Malware-traffic-analysis.net*. Obtenido de Blog Malware-traffic-analysis.net: www.malware-traffic-analysis.net

*Bro pàgina web*. (2014). Obtenido de Bro Web Site: https://www.bro.org/

Burks, D. (2017). *ELSA Security Onion*. Obtenido de ELSA Security Onion: https://github.com/Security-Onion-Solutions/security-onion/wiki/ELSA

Burks, D. (2017). *Squert pàgina web*. Obtenido de Squert pàgina web: https://github.com/Security-Onion-Solutions/security-onion/wiki/Squert

Burks, D. (2017). *Wiki Security Onion*. Obtenido de Wiki Security Onion: https://github.com/Security-Onion-Solutions/security-onion/wiki/

Cid, D. B. (2007). *Log analysis using OSSEC*. Obtenido de Log analysis using OSSEC: http://ossec.net/ossec-docs/auscert-2007-dcid.pdf

Cid, D. B. (2017). *OSSEC Pàgina web del programari*. Obtenido de OSSEC WebSite: http://www.ossec.net/

Cisco. (2017). *Snort Users Manual 2.9.11*. Obtenido de Snort Users Manual 2.9.11: http://manual-snort-org.s3-website-us-east-1.amazonaws.com/

Conde, P. C. (s.f.). *Curso de Seguridad informàtica práctica - Universidad de Valencia* . Obtenido de Miriadax: https://miriadax.net/web/seguridad-informatica-practica

Deuble, A. (26 de Julio de 2016). *Detecting and Preventing Web Application Attacks with Security Onion - SANS Institute*. Obtenido de Detecting and Preventing Web Application Attacks with Security Onion - SANS Institute: https://www.sans.org/reading-room/whitepapers/detection/configuring-security-onion-detect-prevent-web-application-attacks-33980

*Elasticsearch Site*. (2017). Obtenido de Elasticsearch Site: https://www.elastic.co/products

Jahangiri, A. (30 de Abril de 2012). *How to Test Snort with Penetration Testing Tools*. Obtenido de How to Test Snort with Penetration Testing Tools - Blog: http://blog.alijahangiri.org/2012/04/how-to-test-snort-with-penetration-testing-tools/

Lambert, W. (s.f.). *Security Onion Architecture*. Obtenido de Security Onion Architecture: https://cloud.githubusercontent.com/assets/1659467/18955044/a8a4d744-8623-11e6-8116-b9cbc43daed3.png

*Logwatch web site*. (2017). Obtenido de Logwatch web site: https://sourceforge.net/projects/logwatch/

Ochobits. (Octubre de 2015). *Detección de instrusos con OSSEC - ochobits*. Obtenido de Detección de instrusos con OSSEC: https://www.ochobitshacenunbyte.com/2015/10/21/deteccion-instrusos-ossec/

*OSSEC HIDS Configuration*. (s.f.). Obtenido de OSSEC HIDS Configuration - Syngress: http://www.ossec.net/ossec-docs/OSSEC-book-ch3.pdf

*OSSEC Manual*. (2010). Obtenido de OSSEC Manual: http://ossec-docs.readthedocs.io/en/latest/manual/

Robertston, C. (5 de Julio de 2011). *OSSEC - Practical OSSEC - SANS Institute*. Obtenido de OSSEC - Practical OSSEC - SANS Institute: https://pen-testing.sans.org/resources/papers/gcih/practical-ossec-115906

*Sguil - Pàgina web del programari*. (2017). Obtenido de Sguil - Pàgina web del programari: https://github.com/Security-Onion-Solutions/security-onion/wiki/Sguil

Slagell, A. (s.f.). *The Bro Monitoring Platform - National Center for Supercomputing Applications*. Obtenido de The Bro Monitoring Platform - National Center for Supercomputing Applications: https://www.bro.org/current/slides/broverview-2015.pdf

*Snort pàgina del programari*. (2017). Obtenido de Snort Web Site: https://www.snort.org/

Solutions, S. O. (2017). *Security Onion pàgina web del programari*. Obtenido de Security Onion pàgina web del programari: https://securityonion.net/

*Suricata Pàgina del programari*. (2017). Obtenido de Suricata web Site: https://suricata-ids.org/

Wazuh, I. (2017). *OSSEC - Testing decoders and rules* . Obtenido de https://documentation.wazuh.com/2.0/user-manual/ruleset/testing.html

# APENDIX I: Fase de desenvolupament - Instal·lació de Security Onion en mode avaluació

En aquest apèndix es detallarà com s’ha realitzat la instal·lació i les proves de Security Onion en una màquina virtual en mode avaluació (Burks, Wiki Security Onion, 2017). El mode avaluació ja porta incorporats el servidor i el sensor integrats en una mateixa màquina, junt amb totes les utilitats de HIDS, NIDS, anàlisi i visualització. Es descriu també la instal·lació i avaluació d’un agent OSSEC.

En algunes de les proves d’aquesta fase el tràfic capturat per avaluar les alertes serà simulat mitjançant arxius de captura de tràfic pcap. No capturarem encara tràfic real ja que en la solució de virtualització on hem instal·lat la màquina no tenim la possibilitat de posar la interfície de xarxa en mode “bridge” per que capturi tot el tràfic de la interfície de xarxa del host VMWare.

Per avaluar la funcionalitat de OSSEC s’instal·larà un agent en un host i es comprovarà que en el servidor es reben les alertes corresponents i que la monitorització dels servidors és correcte. (Wazuh, 2017) (Robertston, 2011)

## Passos a realitzar per la instal·lació

Per realitzar la instal·lació cal seguir els següents passos:

1. Creem la màquina virtual amb el programari especificat
2. Obtenim la distribució ISO del programari Security Onion des de la web de Security Onion, i verifiquem que la descarrega es correcte i el checksum coincideix
3. Engeguem la màquina virtual amb la ISO de Security Onion i avancem en la instal·lació
4. Un cop feta la instal·lació de Security Onion, actualitzem el programari del sistema amb l’script d’actualització “sudo soup”

### Setup Fase 1

1. Cliquem en la icona del “setup”
2. Quan ens pregunta si volem configurar les interfícies, li diem que sí
3. Especifiquem quina és la interfície de management i li configurem l’adreça IP
4. Especifiquem la interfície de monitorització. No li posem cap IP.
5. Se’ns presentarà un sumari dels canvis (figura I.2). Cliquen “yes, make changes!”
6. L’equip reinicia i els canvis definits s’aplicaran

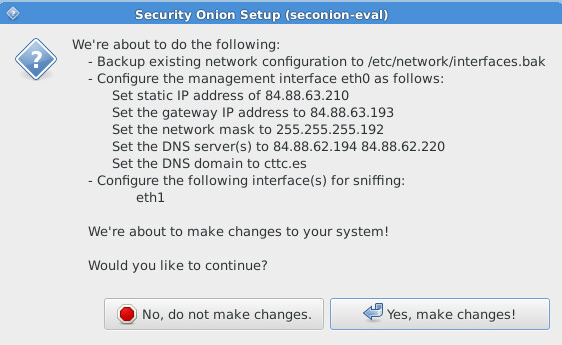


Figura 10.1 - llista de canvis aplicats en la instal·lació de Security Onion

### Setup Fase 2

1. Escollim de nou la l’opció “setup”
2. Ens demana si ens volem saltar la configuració de xarxa. Diem sí ja que ja l’hem configurat en el pas previ
3. En aquest punt ens pregunta si volem mode d’avaluació o de producció. Escollim “Evaluation”.
4. Ens presentarà un sumari (figura I.3) amb la configuració que tindrà el sistema i només mancarà acceptar. El sistema quedarà configurat amb OSSEC com a HIDS; Snort i Bro com a NIDS en la interfície eth1; ELSA com a eina per visualitzar els logs de Bro; Sguil com a eina de visualització de les alertes de Snort i OSSEC.
5. Els serveis interns que quedaran corrent per defecte són: netsniff-ng, pcap\_agent, snort\_agent, barnyard2, Xplico.

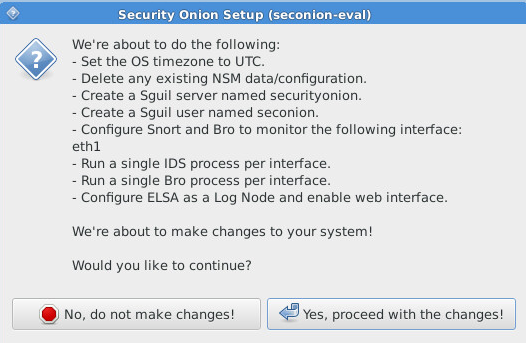


Figura 10.2 - llista de canvis aplicats en la instal·lació de Security Onion

### Verificació post-instal·lació

Un cop feta la instal·lació, la comprovem executant ***sudo sostat*** i veiem que els serveis estiguin corrent, que les regles de Snort s’hagin descarregat, i que tot hagi anat com s‘esperava.

seconion@seconion-virtual-machine:~/Downloads/exercice4$ sudo sostat

[sudo] password for seconion:

=========================================================================

Service Status

=========================================================================

Status: securityonion

\* sguil server[ OK ]

Status: HIDS

\* ossec\_agent (sguil)[ OK ]

Status: Bro

Name Type Host Status Pid Started

bro standalone localhost running 4420 12 Oct 10:54:24

Status: seconion-virtual-machine-eth1

\* netsniff-ng (full packet data)[ OK ]

\* pcap\_agent (sguil)[ OK ]

\* snort\_agent-1 (sguil)[ OK ]

\* snort-1 (alert data)[ OK ]

\* barnyard2-1 (spooler, unified2 format)[ OK ]

Una altre comanda que podem utilitzar per verificar que els serveis estan corrent és:

***sudo service nsm status (stop/start)***

## Instal·lació de l’agent OSSEC

Instal·lem l’agent OSSEC en els equips a vigilar. El servidor de OSSEC, per la seva banda, està integrat en el equip Security Onion en mode avaluació. (OSSEC HIDS Configuration, s.f.) (OSSEC Manual, 2010)

La comunicació entre el servidor i els agents és xifrada i autenticada. Per aquest motiu, per cada agent és necessari crear una clau d’autentificació en el servidor OSSEC, per després exportar-la, i importar-la en el agent. El servidor de OSSEC escolta per el port 1514/UDP.

### Instal·lació de l’agent

Per fer la prova instal·larem l’agent OSSEC en un equip Radius que no està en producció:

* Nom de l’equip: freeradius3
* Sistema operatiu: Ubuntu 16.04
* Ip: 10.2.1.113

Per instal·lar un agent OSSEC en un equip Ubuntu, podem instal·lar el paquet des del repositori. Cal fer la següent comanda:

***apt-get install ossec-hids-agent***

### Alta de l’agent en OSSEC Server

Primer hem d’obrir el Firewall de l’equip Security Onion per permetre la comunicació amb l’agent OSSEC. Security Onion disposa d’un script per fer-ho.

***sudo so-allow***

i escollim la opció “[o] - ossec agent - port 1514/udp”

Posem la IP on estarà l’agent (10.2.1.113), i el Script afegirà en el Firewall del servidor la regla.

***sudo ufw allow proto udp from 10.2.1.113 to any port 1514***

Seguidament cal donar d’alta l’agent OSSEC i crear la clau en el OSSEC server. Per fer-ho s’utilitza la comanda

***sudo /var/ossec/bin/manage\_agent*s**

L’agent quedarà donat d’alta en el servidor. El servidor li assignarà a l’agent una ID i una clau. Aquesta clau la podrem extreure després i importar-la en l’agent OSSEC.

Cal reiniciar el servidor OSSEC per que tingui en compte el nou agent.

***sudo /var/ossec/bin/ossec-control restart***

Finalment a l’agent executem també ***manage\_agents***, i importem la clau i reiniciem l’agent OSSEC.

En aquest punt l’agent ja estarà preparat per connectar-se amb el servidor i començar a funcionar com agent HIDS enviant alertes al servidor. La posada en marxa d’un nou agent genera un missatge en el servidor.

### Configuració de l’agent OSSEC

Dins l’agent, en el directori ***/var/ossec/etc/ossec.conf*** podem ajustar la configuració:

* Escollim els directoris i arxius dels quals volem vigilar la integritat

***<directories check\_all="yes">/etc,/usr/bin,/usr/sbin</directories>***

* Escollim els directoris i arxius dels quals volem ignorar la integritat

***<ignore>/etc/httpd/logs</ignore>***

* Indiquem els arxius de log de les aplicacions que volem controlar

***<localfile>***

***<log\_format>apache</log\_format>***

***<location>/var/www/logs/access\_log</location>***

***</localfile>***

## Verificacions

### Comprovació funcionament HIDS OSSEC

La comanda executada en el OSSEC server ens permet veure els agents OSSEC connectats (figura 10.4):

***agent\_control -l***

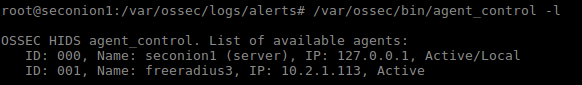


Figura 10.34 - Captura de pantalla del servidor

En l’arxiu següent indiquem quin ha de ser el nivell de les alarmes dins OSSEC per que apareguin en la consola de Sguil (figura 10.5).

***/etc/nsm/securityonion.conf***

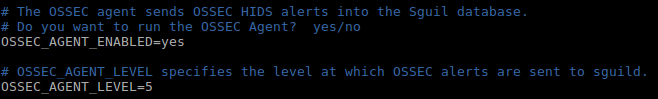


Figura 10.4 - Captura de pantalla del servidor

#### Reiniciar l’agent

Reiniciem l’agent que tenim en el equip freeradius3:

***sudo /var/ossec/bin/ossec-control restart***

Comprovem l’alerta rebuda en el servidor OSSEC (var/ossec/logs/alerts/alerts.log) (figura 10.6). Es de nivell 3 i doncs no apareix en la consola Sguil.

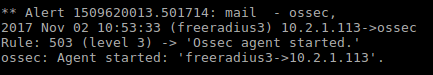


Figura 10.56 - Captura de l’alerta

#### Canvi en un fitxer vigilat

Fem un canvi en un fitxer dins del directori ***/etc*** del equip amb l’agent, i veiem les alertes que ens arriben degut a canvis en la integritat dels fitxers (figura 10.7). Són alertes OSSEC de nivell 7.

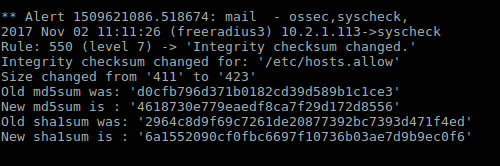


Figura 10.6 - Captura de l'alerta d'integritat de fitxers

Veiem també una alerta a Sguil (figura 10.8).

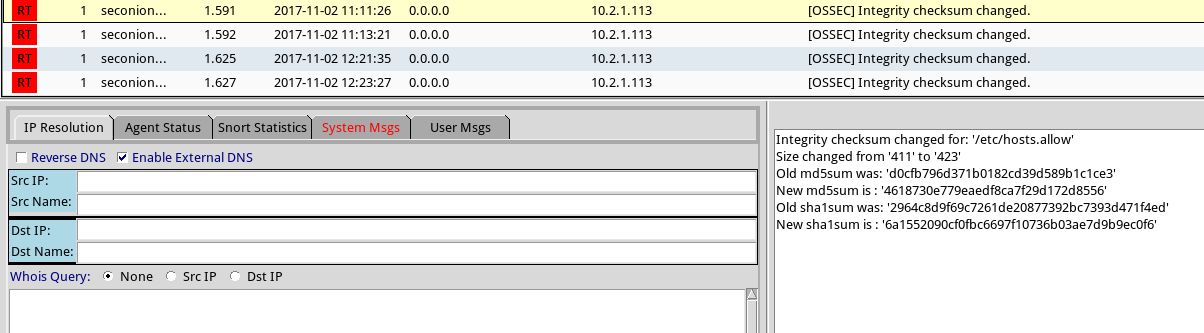


Figura 10.7 - Captura de l'alerta a Sguil

#### Atac de força bruta mitjançant SSH

Intentem registrar-nos en l’equip amb un usuari inexistent varies vegades

***ssh -p 22 tr@freeradius3***

Comprovem una alerta rebuda en els logs de OSSEC (figura 10.9).

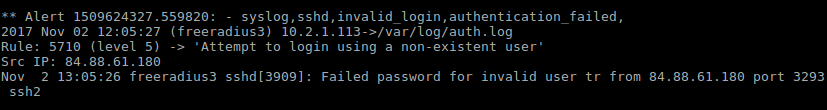


Figura 10.8 - Captura de l'alerta a Ossec

Comprovem també les alertes en la consola Sguil, ja que són alertes de nivell 5 (figura 10.10).

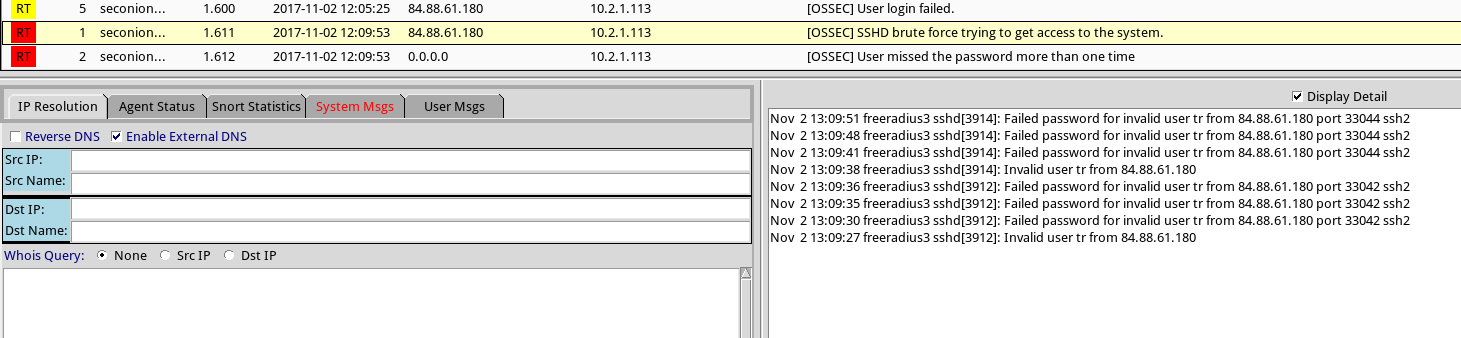


Figura 10.9 - Captura de l'alerta a Sguil

#### Mesures de resposta activa

Amb la següent comanda en el OSSEC server, comprovem les mesures de resposta activa disponibles

***/var/ossec/bin/agent\_control -L***

***OSSEC HIDS agent\_control. Available active responses:***

***Response name: host-deny600, command: host-deny.sh***

***Response name: firewall-drop600, command: firewall-drop.sh***

Quan OSSEC detecti que un equip fa una acció que fa saltar una regla, inclourà la IP atacant en /etc/hosts.deny i la prohibirà en el Firewall durant 600s (valor per defecte).

***Testejar la resposta activa***

Intentem registrar-nos des de la màquina 10.2.1.112 i amb diferents usuaris a la màquina freeradius3 (IP 10.2.1.113).

El resultat és que quan provem amb el tercer usuari, el sistema atacat deixa de respondre’ns (figura 10.11).

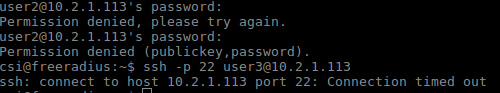


Figura 10.10 - Captura de pantalla amb ssh sense resposta

En el sistema atacat, apareix una regla al Firewall que ens talla l’accés des de la IP atacant 10.2.1.112 (figura 10.12).

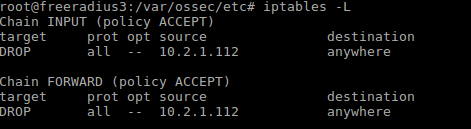


Figura 10.11 - Captura de pantalla amb regla del Firewall

També en el fitxer */etc/hosts.deny* s’inclou la IP de l’atacant (figura 10.13).

***vim /etc/hosts.deny***

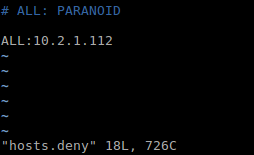


Figura 10.12 - Captura de pantalla fitxer hosts.deny

#### Modificar una regla

Per canviar una regla, és millor no modificar l’arxiu de regla original, ja que qualsevol actualització del sistema n’esborraria les modificacions. Perquè els canvis siguin efectius i persistents cal fer les modificacions a:

***/var/ossec/rules/local\_rules.xml***

OSSEC disposa de la utilitat ***ossec-logtest*** que ens ajuda a construir i veure com interaccionen els missatges de log del sistema amb les regles creades.

***Exemple de modificació d’una regla:***

Modifiquem una regla perquè no es generi una alerta ni s’enviï un email quan es produeix un determinat missatge de warning en el syslog

* log que produeix la alerta

*Nov 2 17:03:41 seconion1 vmsvc[3062]: [ warning] [guestinfo] Failed to get vmstats.*

* regla que desencadena la generació de la alerta

*Rule id: '1002'*

* Resultat del tractament del log si usem ossec-logtest, on veiem que es genera la alerta “Unknown problem somewhere in the system” de nivell 2 (figura 10.14).

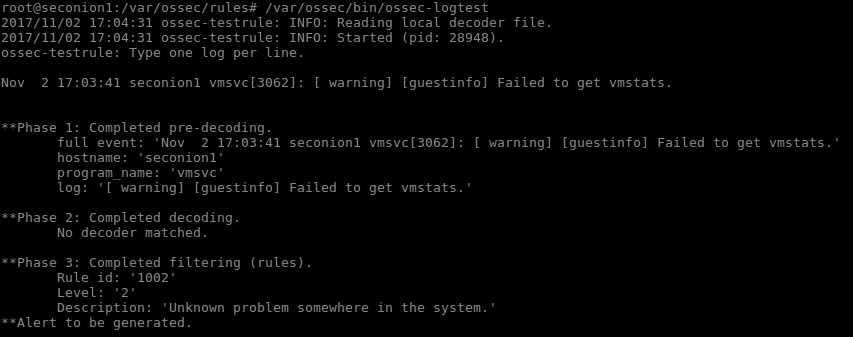


Figura 10.13 - Tractament del log amb ossec-logtest abans d'aplicar regla

* A ***/var/ossec/rules/local\_rules.xml*** afegim el següent paràgraf, on “match” fa que si en el “log event” posa vmstats, la regla no llenci cap alerta (level=0) (figura 10.15).

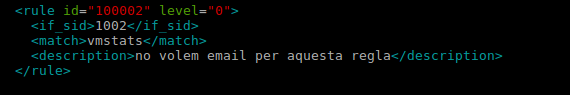


Figura 10.14 - Regla nova dins local\_rules.xml

* Resultat de ossec-logtest un cop introduïda la modificació, on es veu que la regla que intervé és la nova (1002) i no es genera cap alerta (figura 10.16).

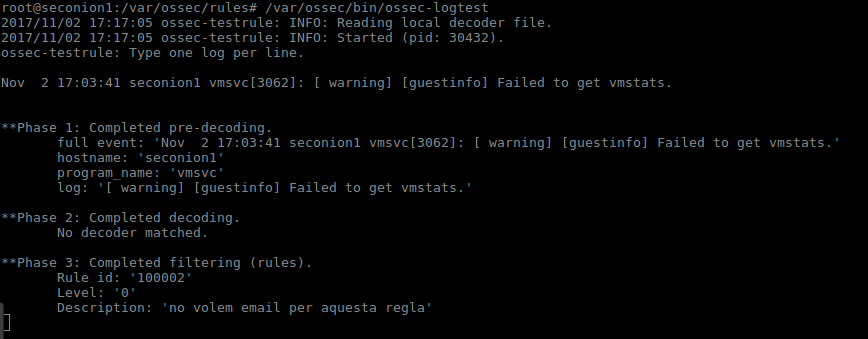


Figura 10.15 - Sortida de ossec-logtest un cop aplicada la regla nova

* De la mateixa manera, podem deshabilitar la regla original canviant-li el level a 0.

### Verificació de funcionament NIDS Snort i Bro

#### Creació d’una regla de snort

Crearem una regla que ens servirà per provar el funcionament de snort i que consistirà en que l’equip llenci una alerta cada cop que fem ping des de la nostre màquina.

Creem la alerta dins de ***/etc/nsm/rules/local.rules***

*Alert icmp 84.88.61.180 any -> any (msg:”ICMP”; sid:100002;)*

Fem el ping i revisem a la consola de sguil que tenim un alarma (figura 10.17).



Figura 10.16 - Captura d’alerta consola Sguil

#### Modificació d’una regla

La principal raó que ens pot portar a modificar una regla és que generi moltes alertes. Per exemple, a continuació mostrem algunes de les regles que generen més alertes (figura 10.18).

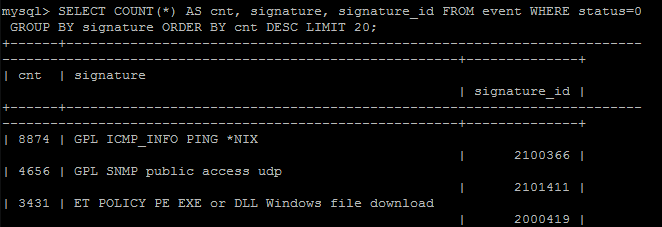


Figura 10.17 - Regles amb més alertes

Si ens fixem en la regla amb id 2100366, veiem que ha saltat 8874 vegades. Aquesta alerta implica l’adreça IP origen 84.88.62.209 (figura 10.19), que coneixem i donem per vàlida. Es un equip de monitorització i és normal que faci *pings* als equips per verificar que estiguin actius.

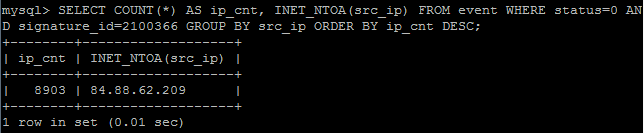


Figura 10.18 - Captura amb IP de la màquina que fa generar les alertes

Modificarem la regla amb id 2100366 (figura 10.20) per que no salti cap alerta si la IP 84.88.62.209 està implicada.



Figura 10.19 - captura amb la regla a modificar

Els passos a seguir són:

1. Editem el fitxer ***sudo vi /etc/nsm/rules/local.rules***, on s’editen les regles locals, i creem una nova regla per que no generi alertes si la IP origen és 84.88.62.209. Li assignem un SID 9000366

*alert icmp !84.88.62.209 any -> $HOME\_NET any (msg:"GPL ICMP\_INFO PING \*NIX"; itype:8; content:"|10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F|"; depth:32; classtype:misc-activity; sid:9000366; rev:1; metadata:created\_at 2017\_11\_04, updated\_at 2017\_11\_04;)*

1. Deshabilitem la regla original afegint en el fitxer /etc/nsm/pulledpork/disablesid.conf:

1:2100366

1. Reiniciem les regles: *sudo rule-update*
2. Comprovem que aquesta alerta ja no ha de sortir més cops.

### Verificació de les eines de visualització

Per avaluar Snort i Bro, així com les eines d’anàlisi i visualització associades, es poden usar arxius de tràfic capturat *pcap*, que reproduirem amb l’eina *tcpreplay*. Security Onion inclou mostres de tràfic pcap i també es poden descarregar mostres de tràfic amb captures d’atacs de malware, amb objectius educatius o d’estudi. Un exemple d’aquestes webs és per exemple www.malware-traffic-analysis.net. (Brad, 2017)

#### Descripció de la prova

S’analitzarà el tràfic que conté una infecció per malware en un equip Windows, i es veurà quina informació es pot extreure amb les eines que ens proporciona Security Onion (Snort, Bro, Sguil, ELSA, Capme i NetworkMiner)

Executem el pcap en el servidor Security Onion instal·lat:

***tcpreplay -ieth1 -M10 2015-11-24-traffic-analysis-exercise.pcap***

#### Alertes Snort i visualització des de Sguil

Obrim la consola de Sguil per veure les alertes que Snort ha fet saltar (figura 10.21).

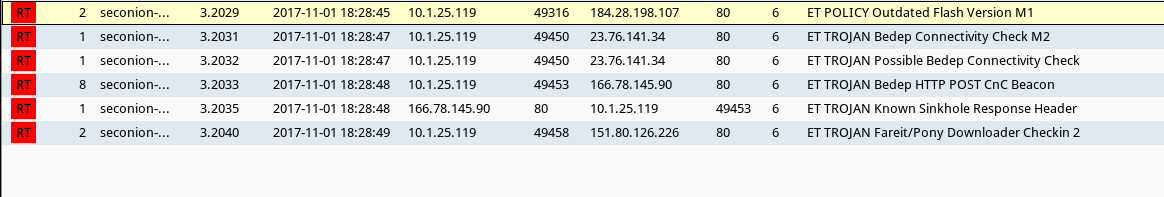


Figura 10.20 - alertes de Sguil després d'executar el pcap amb tràfic maliciós

Veiem que hi ha una alerta de una versió de flash desactualitzada. També hi han alertes de trojans que intenten accedir al CnC (Comand and Control, des d’on es controlen les xarxes de trojans). Així doncs sembla que efectivament la infecció ha tingut èxit.

Si cliquem en l’alerta, podem veure la regla de Snort que l’ha fet saltar, i les dades del paquet (figura 10.22).

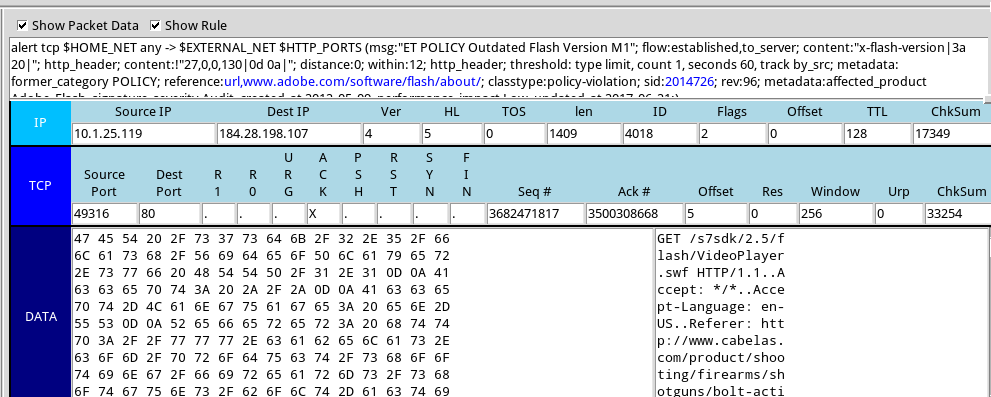


Figura 10.21 - Regla de Snort que ha fet saltar la alerta

També tenim integrat un whois query que ens dona informació sobre la IP implicada (184.28.198.107). En aquest cas és una IP del proveïdor de hosting Akamai (figura 10.23).

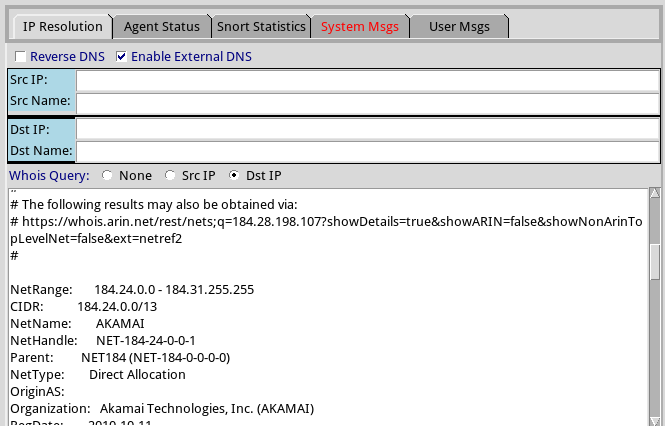


Figura 10.22 - Captura amb la informació WHOIS de la IP implicada

Si fem botó dret sobre el Id de la alerta, podem veure la transcripció del tràfic TCP que ha provocat l’alerta (figura 10.24). El blau és l’origen i el vermell el destí. Si analitzem el tràfic es pot veure que en un moment donat es descarrega en el client infectat alguna aplicació flash.

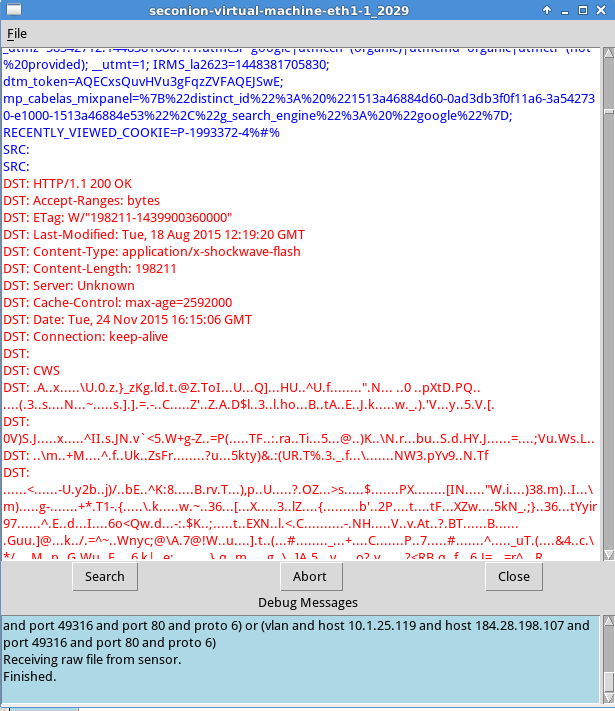


Figura 10.23 - Captura amb la transcripció del tràfic TCP de la alerta.

Des de Sguil, podem pivotar cap a Network Miner per extreure les dades intercanviades, i doncs el fitxer flash que ha causat la infecció, que és VideoPlayer.swf (figura 10.25).

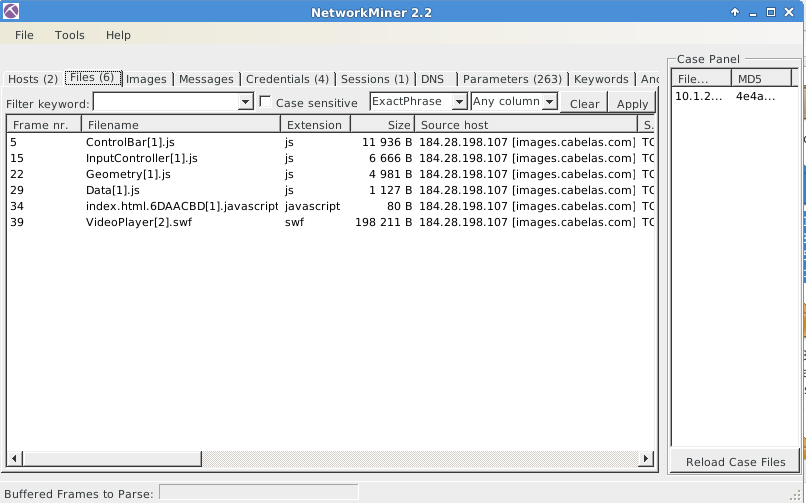


Figura 10.24 - Captura de NetworkMiner amb els fitxers implicats

#### ELSA

Basculem a l’agregador de logs ELSA per veure quina informació ha recopilat Bro d’aquesta IP (figura 10.26).

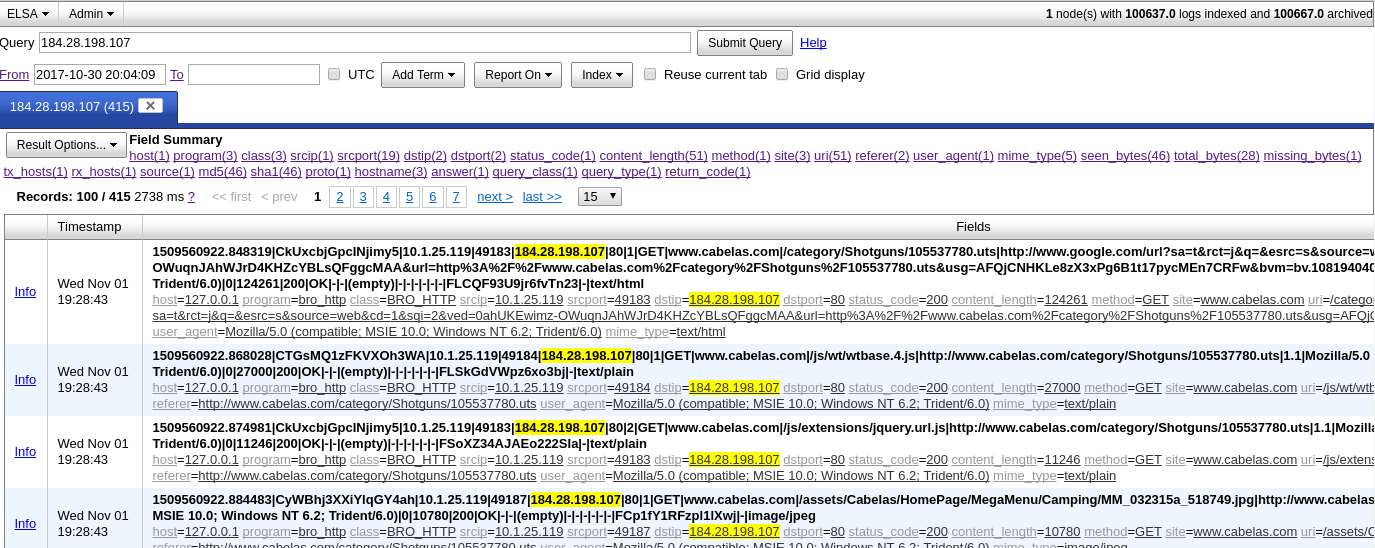


Figura 10.26 - Captura de ELSA amb informació de la IP implicada

Podem veure la informació capturada per Bro agrupada per programes (figura 10.27).

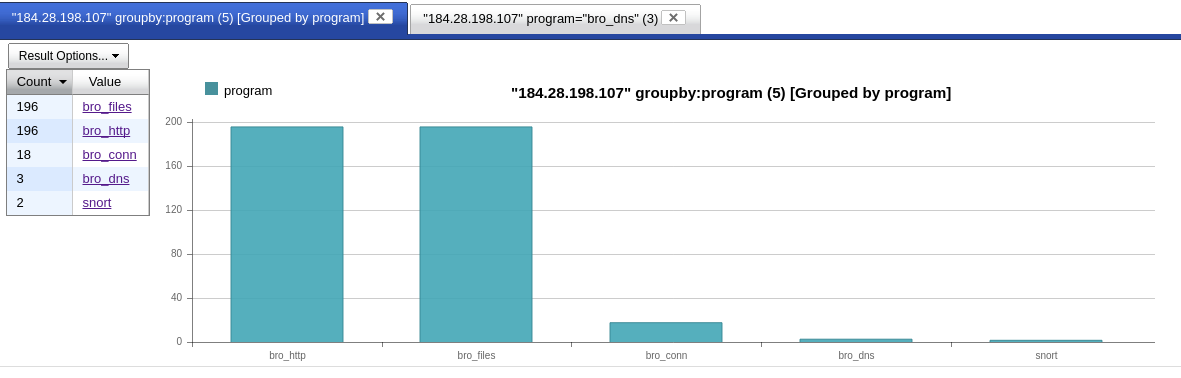


Figura 10.25 - Captura amb el nombre de captures de Bro agrupat per programes

Mirant el tràfic DNS, podem veure el hostname de la Ip maliciosa, “cabelas.com”

Si mirem el tràfic http, podem veure el “referer”, és a dir des d’on s’ha redirigit al client cap al servidor maliciós (figura 10.28).

“http://www.cabelas.com/product/shooting/firearms/shotguns/bolt-action-single-shot-shotguns%7C/pc/104792580/c/553829580/sc/105537780/i/104801”

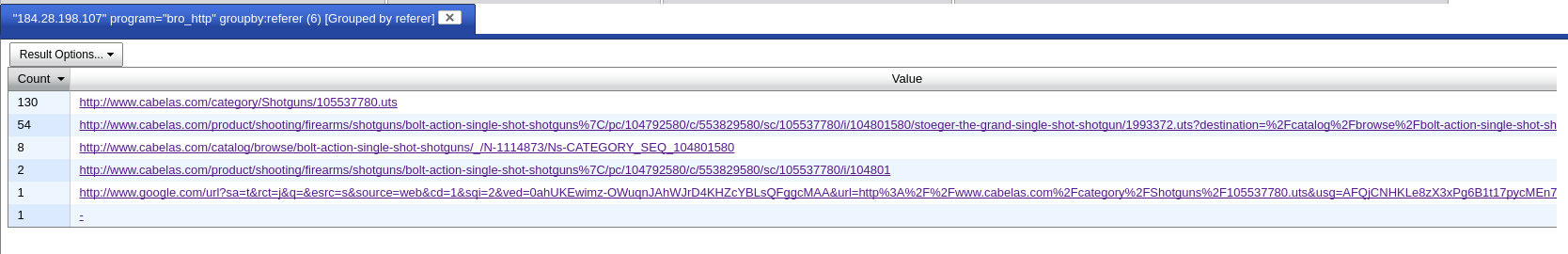


Figura 10.26 - Captura de Bro amb els referers

# APENDIX II: Fase d‘implantació - Instal·lació de Security Onion en producció

En aquest apèndix es detallarà com s’ha realitzat la instal·lació de Security Onion en una màquina virtual en mode de producció (Burks, Wiki Security Onion, 2017).

## Passos a realitzar per la instal·lació

Primer de tot arrenquem l’equip i comencem la instal·lació de la distribució. Escollim el mode de producció (figura 11.1).

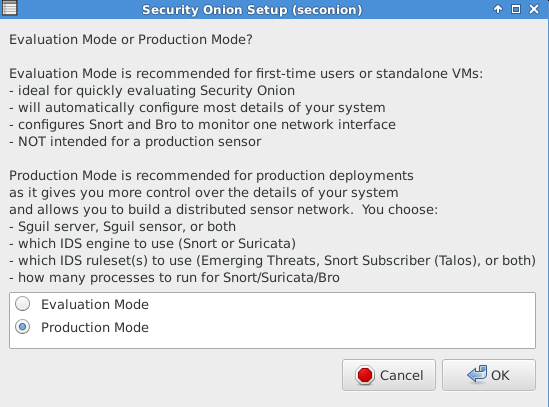


Figura 11.1 - Captura Security Onion setup Production Mode

Escollim el mode de desplegament *standalone,* per que sensor i servidor comparteixin maquinari (figura 11.2)

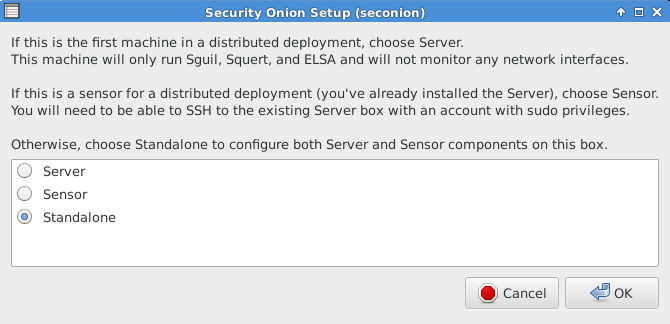


Figura 11.2 - Captura Security Onion setup Deployment Standalone

Escollim Snort com a NIDS (figura 11.3)

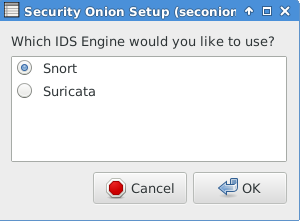


Figura 11.3 - Captura Security Onion setup IDS Snort

Escollim el conjunt de regles de *Snort Emerging Threats Open* (figura 11.4)

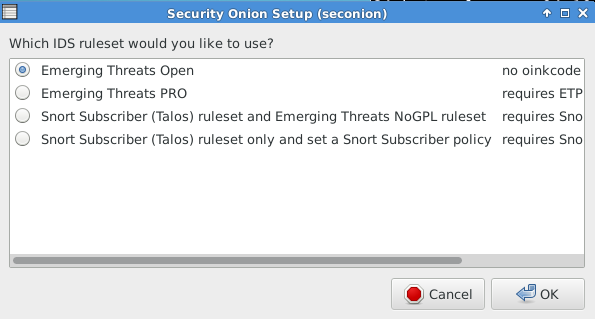


Figura 11.4 - Captura Security Onion setup ruleset

Paràmetre per defecte per la captura de paquets (figura 11.5), que podrem ajustar més endavant si cal.

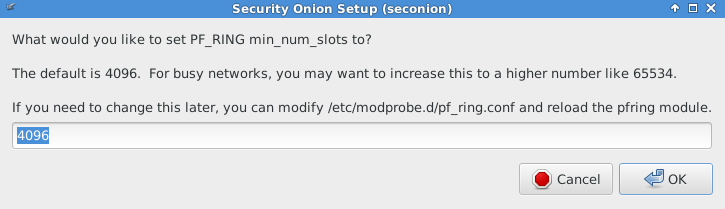


Figura 11.5 - Captura Security Onion setup PF\_RING

Indiquem les interfícies de xarxa que seran per monitorització (figura 11.6), per que siguin configurades en mode “promiscu”.

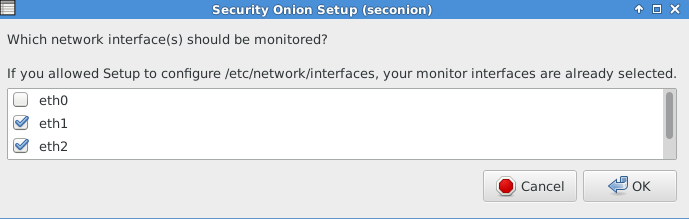


Figura 11.6 - Captura Security Onion setup monitor interfaces

Afegim les Home\_nets (figura 11.7), que seran les xarxes locals.

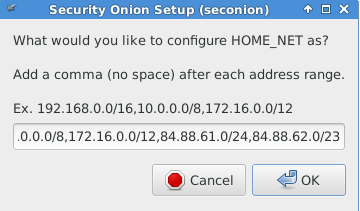


Figura 11.7 - Captura Security Onion setup HOME\_NET

Acceptem els canvis i repassem la configuració final (figura 11.8).

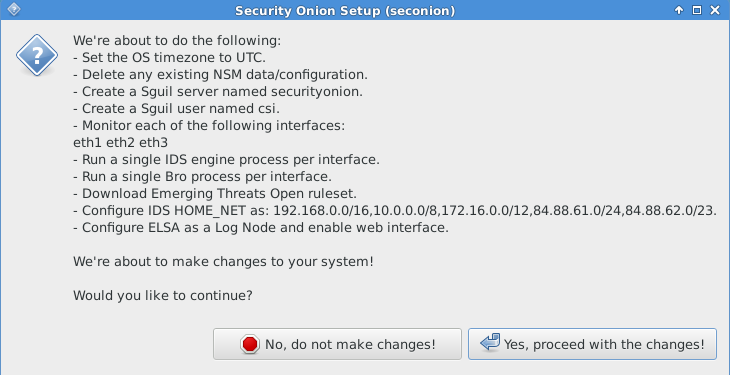


Figura 11.8 - Captura Security Onion setup Resum Configuració

Es baixen les regles de pulled pork (figura 11.9)

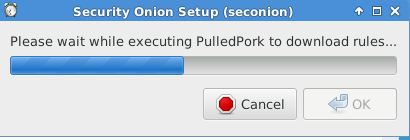


Figura 11.9 - Captura Security Onion setup regles Pulled Pork

I s’ acaba la instal·lació (figura 11.10)

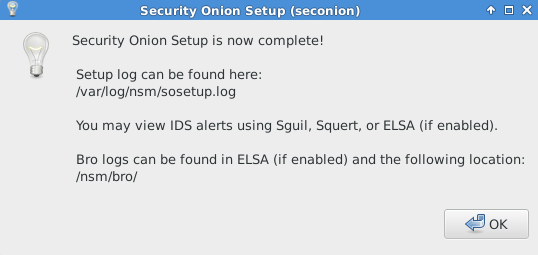


Figura 11.10 - Captura Security Onion setup complete

## Post instal·lació i proves

Un cop feta la instal·lació i amb l’equip configurat i connectat a la xarxa, es fan les verificacions inicials per comprovar que el sistema funciona i està monitoritzant el tràfic correctament. Després preparem l’entorn i les eines per poder començar a treballar amb la plataforma i començar a realitzar anàlisis de seguretat.

### Estat dels serveis

Comprovem l’estat dels serveis ***sudo service nsm status*** (figura 11.11)

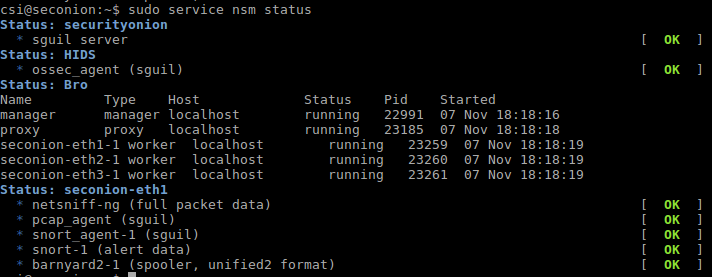


Figura 11.11 - Captura del SO server estat dels serveis

### Consola sguil

Ens registrem a sguil i verifiquem que tenim alertes (figura 11.12).

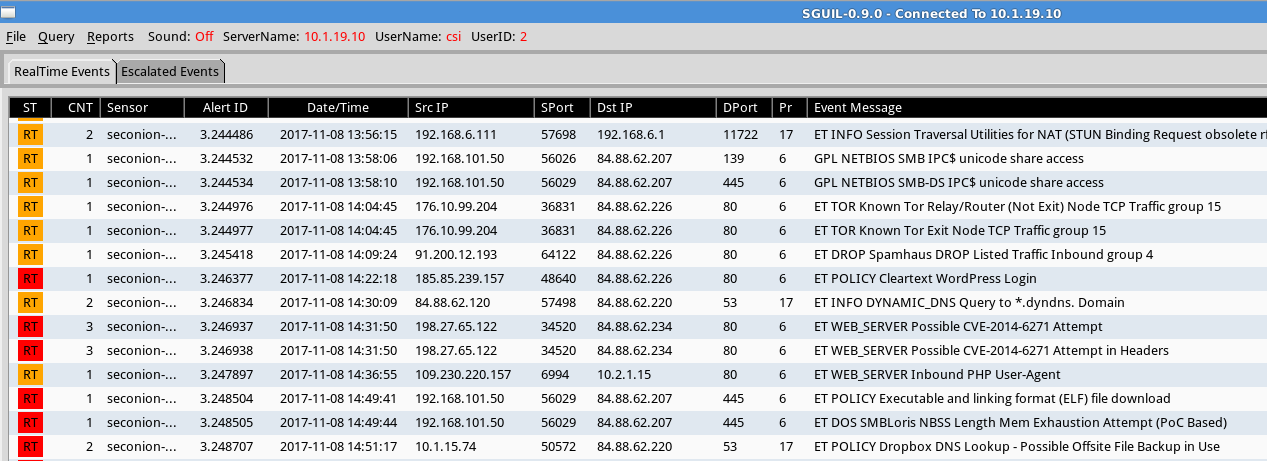


Figura 11.12 - Captura de les alertes a Sguil

### Testmyids

Anem a http://testmyids.com i veiem que apareix una alerta (figura 11.13).

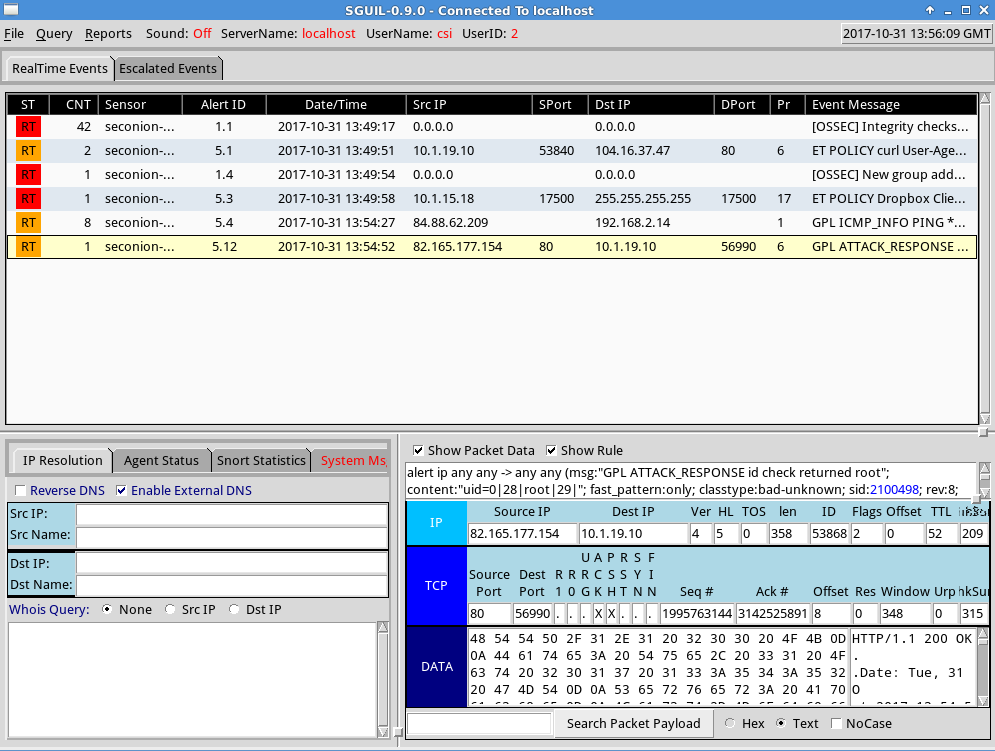


Figura 11.13 - Captura Sguil amb alerta tesmyids.com

### Port-mirroring en el switch

Mirem el switch que fa port-mirroring per veure el tràfic que està monitoritzant . En aquesta gràfica (figura 11.14) veiem el tràfic de la interfície monitoritzada, interfície Portchannel 5.

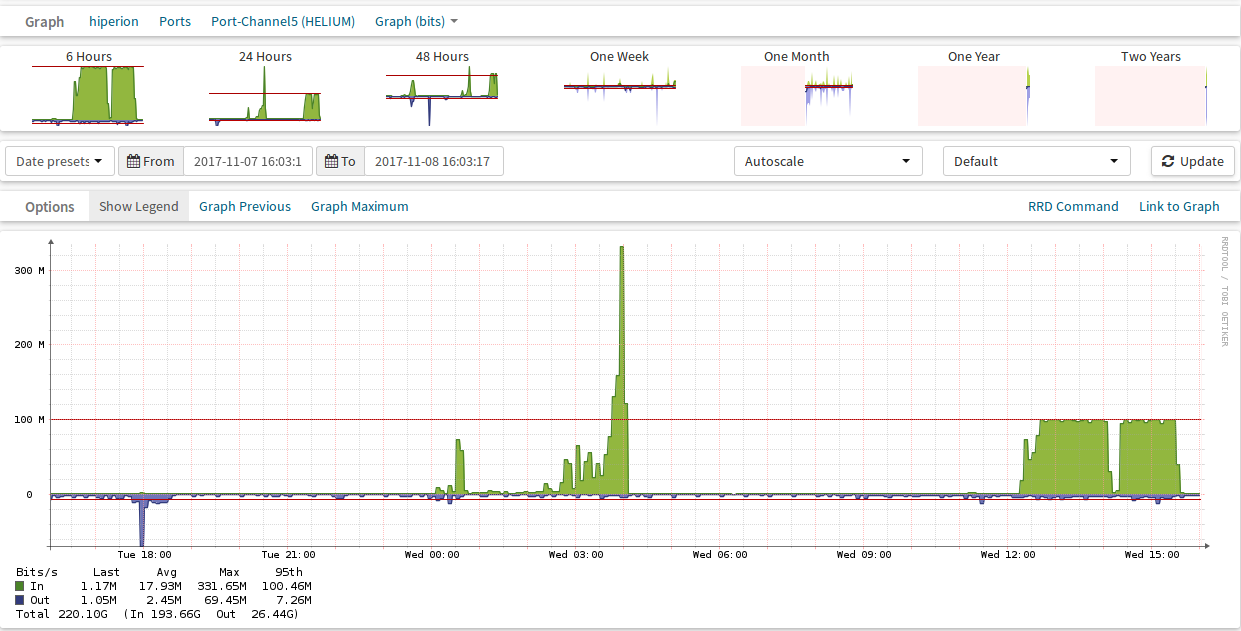


Figura 11.14 - Captura Port monitoritzat en el switch

I en aquesta gràfica veiem la interfície destí Gi2/0/23, on està connectada la sonda (figura 11.15). Veiem que el switch està replicant el tràfic correctament ja que coincideixen.



Figura 11.15 - Captura port mirror en el switch

### Comprovació de la carrega del sensor

Comprovem la carrega del sensor, per verificar que la interfície eth1 no descarti paquets (drops) (figura 11.16).

***sudo sostat | less***

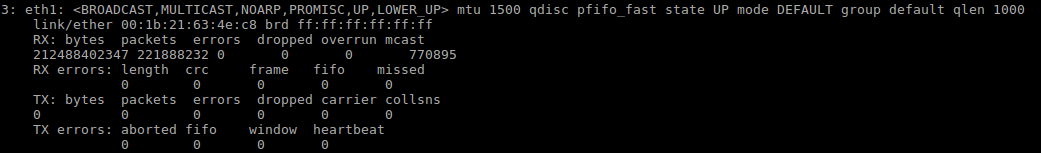


Figura 11.16 - Captura carrega del sensor

### Escaneig de ports

Es presenten els resultats de les deteccions realitzades amb el programa Snort treballant com a IDS. Usant la plataforma Kali (https://www.kali.org/), realitzem alguns atacs de prova. Kali és una distribució de seguretat que incorpora moltes eines per fer pentesting. (Jahangiri, 2012)

Des de la IP externa 213.96.1.216, fem un escaneig de ports al servidor web del CTTC:

* url www.cttc.es
* IP externa (NAT): 84.88.61.199
* IP interna (dins la DMZ): 10.2.1.15

***root@kali:~# nmap -v -A -sV 84.88.61.199***

Comprovem a la consola de Squert que Snort detecta el escaneig correctament.

* **SID de la regla**: 2009358
* **Missatge:** ET SCAN Nmap Scripting Engine User-Agent Detected (Nmap Scripting Engine)
* **IP origen**: 213.96.1.216
* **IP destí**: 10.2.1.15 (IP interna de la web)

A la següent figura veiem les deteccions de l’escaneig dins la consola de Squert (figura 11.17).

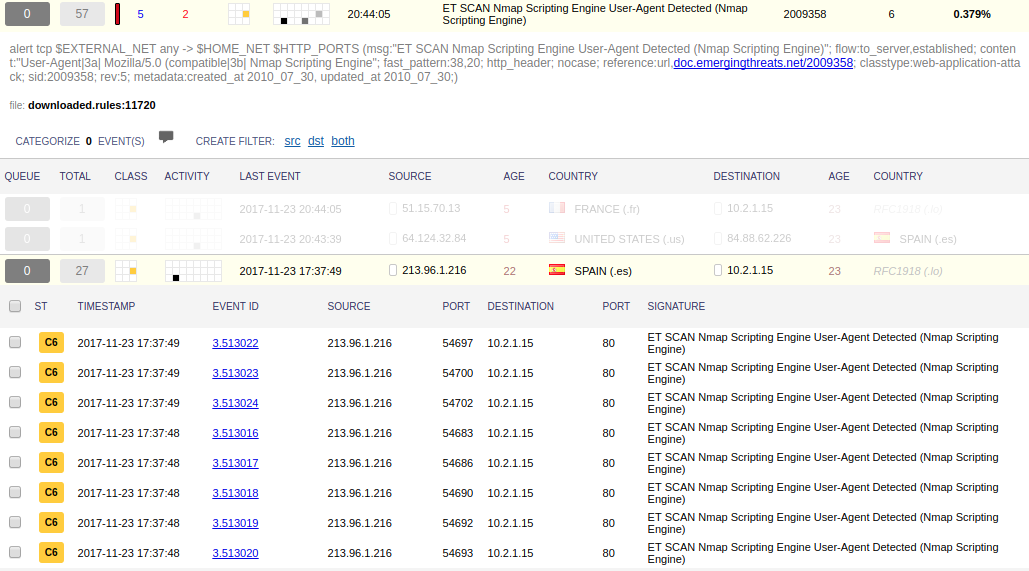


Figura 11.17- Alertes per la detecció de escaneig de ports

### Escaneig de vulnerabilitats

Kali disposa de l’eina WPscan, especialitzada en detectar vulnerabilitats en pàgines Wordpress. La usarem per escanejar la web del CTTC, que esta feta usant Wordpress, per cercar vulnerabilitats.

Enumerem els plugins de la web:

***wpscan --url http://www.cttc.es --enumerate p***

Enumerem els usuaris i descobrim l’usuari “*totbits*”

***wpscan --url http://www.cttc.es --enumerate u***

A partir dels usuaris obtinguts, fem un atac de força bruta amb un arxiu de passwords:

***wpscan --url www.cttc.es --wordlist /usr/share/nmap/nselib/data/passwords.lst --username totbits***

En la consola de Squert comprovem les alertes que aquests atacs han fet saltar (figura 11.18).

* Atac de força bruta (sid: 2014020)
* Scan WPscan (sid: 2020338)

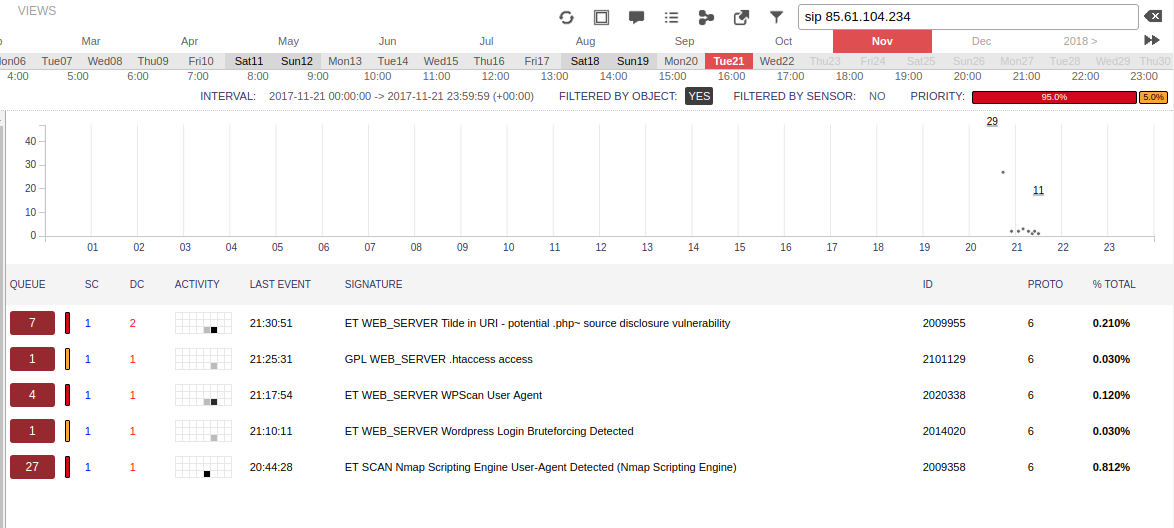


Figura 11.18 - Alertes de WPScan a la consola de Squert

Fem també un escaneig de la vulnerabilitat Poodle, que permet fer un atac Man in the Middle aprofitant la debilitat del protocol SSLv3, per veure si Snort el detecta (figura 11.19):

***nmap -sV --version-light --script ssl-poodle -p 443 84.88.61.199***

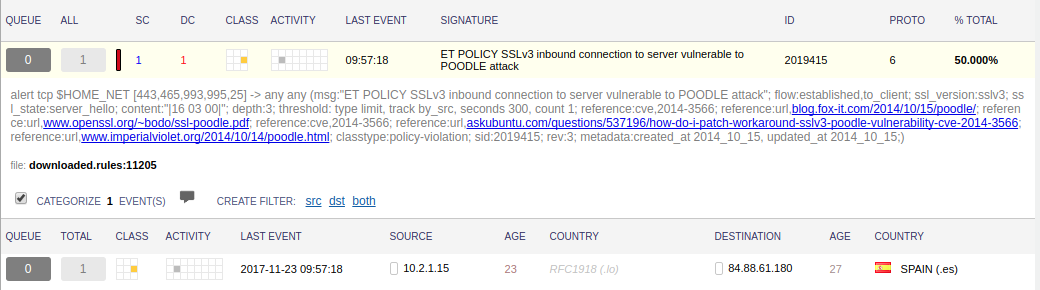


Figura 11.19 - Alerta per escaneig Nmap de la vulnerabilitat Poodle

### Resultats de OSSEC

Hem instal·lat OSSEC en un servidor web que serveix pàgines HTML de projectes i esdeveniments que es realitzen al centre.

* Nom del servidor: cms2
* IP: 84.88.62.226
* Algunes webs servides:
  + scavenge.eu (Wordpress)
  + euw2015.org (Joomla)
  + www.ict-emphatic.eu(Joomla)

Aquestes són algunes de les alertes que hem rebut després d’instal·lar OSSEC i que aquest faci un escaneig dels arxius del host.

#### Vulnerabilitats web

La següent alerta ens avisa d’un web vulnerable dins del servidor. Efectivament aquesta web va ser infectada l’any 2014 amb un malware del tipus backdoor eval-base64\_decode. La web afectada es va desactivar, però no es van esborrar els arxius afectats i OSSEC ens ha avisat.

Alerta OSSEC a Security Onion -> /var/ossec/logs/alerts/alert.log (figura 11.20):

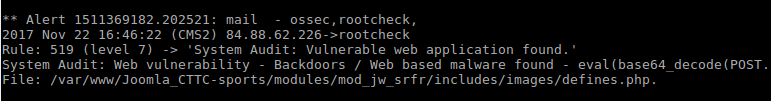


Figura 11.20 - Alerta OSSEC de pàgina web vulnerable

Alerta a la consola de Squert (figura 11.21)

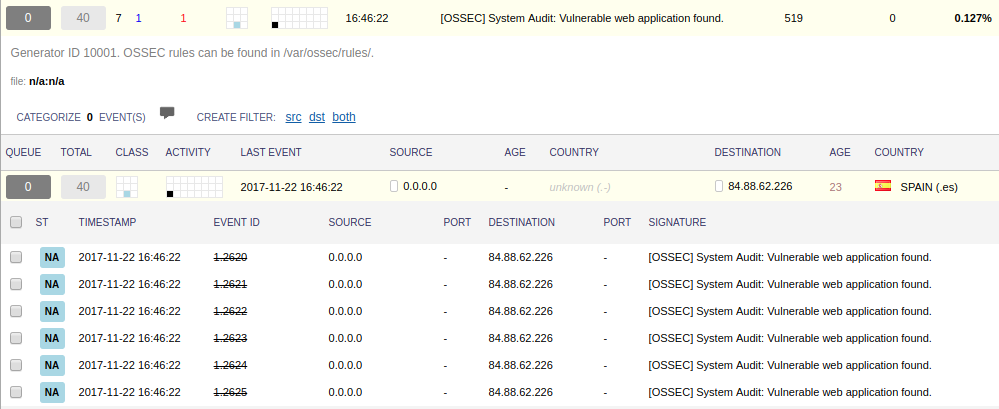


Figura 11.21 - alerta OSSEC a Squert per pàgina web vulnerable

#### Problemes de configuració i accessos incorrectes

OSSEC ens avisa de possibles problemes de configuració, en aquest cas d’un fitxer de certificat que no té els permisos adequats (figura 11.22).



Figura 11.22 - Alerta OSSEC de fitxer amb permisos a revisar

Finalment també ens avisa de les mesures de resposta activa que realitza, fent blacklisting d’alguna màquina que ha fet accessos incorrectes (figura 11.23).

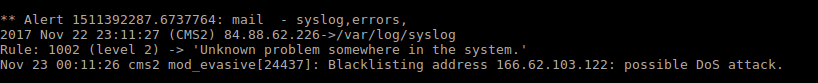


Figura 11.23 - Alerta OSSEC de IP banejada

# APENDIX III: Posada a punt del sistema Security Onion

En aquest apèndix s’expliquen els passos per posar a punt el sistema de monitorització un cop completada la instal·lació i connectada la sonda a la xarxa que volem monitoritzar. (Burks, Wiki Security Onion, 2017)

## Processos

Primer de tot deshabilitem processos no necessaris, per exemple els de les interfícies ethernet que no necessitem ja que encara no estem monitoritzant tràfic amb elles.

Per Bro, cal anar al fitxer

***/opt/bro/etc/node.cfg***

Per Snort, anem al fitxer

***/etc/nsm/sensortab***

i comentem en ambdós fitxers els processos seconion-eth2 i seconion-eth3 ja que encara no s’estan usant.

## Ajustaments a OSSEC

Configurem el sistema per que OSSEC ens envií emails en cas d’alertes greus.

Modifiquem l’arxiu

***/var/ossec/etc/ossec.conf***

<global>

    <email\_notification>yes</email\_notification>

    <email\_to>siem@cttc.es</email\_to>

    <smtp\_server>yes</smtp\_server>

Modificar manualment la configuració ***ossec.conf*** per excloure els fitxers que canvien sovint. Per defecte, si un fitxer canvia 3 vegades, els canvis posteriors són ignorats per OSSEC.

Finalment reiniciem OSSEC per que prengui en compte els canvis

***/var/ossec/bin/ossec-control restart***

## Ajustament de regles de Snort

En posar en marxa la plataforma Security Onion i començar a capturar tràfic i analitzar la xarxa, apareixen gran nombre d’alertes.

***Passos per reduir els falsos positius per deteccions i gestionar correctament les alertes dins de HIDS/NIDS en Security Onion*** (Burks, Wiki Security Onion, 2017)

1. Identificar les regles que produeixen més alertes
   1. Amb Sguil, la columna CNT permet ordenar les alertes segons el nombre d’ alertes correlades.
   2. Podem accedir a la base de dades de Security Onion (Dsecurityonion\_db) i llistar les firmes que més alertes han generat
2. Un cop identificades les alertes que potencialment més soroll poden introduir, les podem analitzar i descartar. Ara bé, si aquestes alertes són degudes a un equip mal configurat, la manera més pràctica de resoldre el nombre d’alarmes és resolent el problema de configuració de l’equip.
3. Identificar les categories de regles Snort que no apliquen en el nostre sistema i desactivar-les.
4. Modificar les regles existents per que s’adaptin millor al nostre entorn. En aquests casos, el procediment a seguir és crear una copia de la regla a modificar, modificar-la, i deshabilitar la original. No modificar les regles originals.

### Deshabilitar regles

Podem deshabilitar les regles que no ens apliquen. Per exemple, tenim alertes de snort avisant-nos sobre l’ús de dropbox, però si resulta que al nostre centre és un programari que està tolerat no té sentit considerar aquestes alertes.

Per deshabilitar una regla:

***sudo vi /etc/nsm/pulledpork/disablesid.conf***

***#regla a deshabilitar***

***1:sid\_de\_la\_regla***

i finalment fem

***sudo rule-update***

### Modificar regles

Hi ha alertes que no volem deshabilitar però si modificar per tal que s’adaptin millor al nostre entorn. Cal editar el següent fitxer:

***sudo vi /etc/nsm/pulledpork/modifysid.conf***

I seguim la sintaxi indicada per modificar una regla.

Per exemple podem posar una excepció per el cas de l’equip de monitorització identificat per $NAGIOS que no volem que ens provoqui alertes, i doncs de HOME\_NET s’exclourà aquest equip.

“\$HOME\_NET” “[$HOME\_NET,!$NAGIOS]”

Finalment fem:

***sudo rule-update***

### Crear noves regles

En alguns casos podem voler esborrar la regla original i reescriure-la segons la nostre conveniència. Per fer això, caldrà modificar el fitxer

***/etc/nsm/rules/local.rules***

I reescriure la regla. Per exemple en la següent capçalera de regla, l’origen no podrà ser cap de les xarxes o equips enumerats, i igual per el destí.

alert tcp ![$XARXA\_ISCSI,$HERCULES,$CSI,$VCENTER\_SERVERS,$ESXI\_SERVERS] !$HTTP\_PORTS -> [$HOME\_NET,!$SAMBA\_SERVERS] any

I per acabar deshabilitem la regla original a

***/etc/nsm/pulledprok/disablesid.conf***

i actualitzem les regles:

***sudo rule-update***

### Autocategories

Sguil ens ofereix la funcionalitat de autocategories, que permet que algunes regles es classifiquin automàticament per categories.

Les categories disponibles per defecte a Sguil són.

* F1: Category I: Unauthorized Root/Admin Access
* F2: Category II: Unauthorized User Access
* F3: Category III: Attempted Unauthorized Access
* F4: Category IV: Successful Denial-of-Service Attack
* F5: Category V: Poor Security Practice or Policy Violation
* F6: Category VI: Reconnaissance/Probes/Scans
* F7: Category VII: Virus Infection
* F8: No action necessary
* F9: Escalate

Alguns exemples d’alertes que ens pot interessar que automàticament es classifiquin són:

* Alertes de scans
* Avisos d’accessos d’usuaris a urls de baixa reputació
* Accessos des de Ips de baixa reputació

Són alertes que volem tenir en compte, però que no ens requereixen cap acció immediata ja que no ens avisen d’atacs que hagin pogut comprometre alguna de les nostres màquines. Els fet de autocategoritzar alertes ens estalvia feina i ens permet centrar-nos en les alertes més importants

En la següent figura 12.1 podem veure el quadre de sguil que permet crear autocategories.

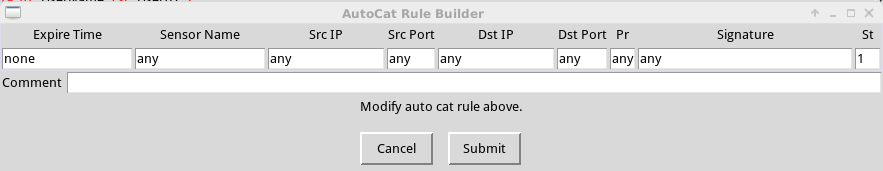


Figura 12.1 sguil - creació de autocategories d'alertes