Documentation de stage : installation et configuration de Snort.

Documentation : Installation, configuration et enrôlement d'un NIDS Snort au sein du SIEM Prelude.

Auteur : Guillaume Orlando
Date : 15/02/2018

I. Introduction :

Snort est un NIDS (Network Intrusion Detection System) open-source, permettant de filtrer et d'appliquer des règles sur l'ensemble du trafic bas niveau d'un réseau. De nombreuses règles sont disponibles gratuitement, et il est possible de facilement en créer de nouvelles de toutes pièces. Ce document décrit les différentes étapes d'installation et de configuration d'une sonde Snort capable de s'intégrer dans l'environnement d'un SIEM Prelude. Les logs générés par le NIDS Snort seront stockés hors de la machine, et les alertes de sécurité vont être remontées au format IDMEF à destination du Prelude-Manager, capable d'interpréter correctement ce format. La machine cible est une RedHat7.

II. Téléchargement de Snort :

La première étape consiste à installer l'outil Snort sur la machine cible. Snort nécessite plusieurs dépendances pour fonctionner correctement, et pour être compilé sans erreur. Les paquets suivants peuvent donc êtres installés via le gestionnaire de paquets de la machine :

```
yum install gcc flex bison zlib libpcap pcre libdnet tcpdump libnghttp2
```

Ainsi que les versions devel :

```
yum install -y zlib-devel libpcap-devel pcre-devel libdnet-devel
libnghttp2-devel
```

Pour plus de clarté, nous allons travailler dans un répertoire spécifique, dans le répertoire personnel de notre compte utilisateur, sous le nom de 'Snort' par exemple :

```
mkdir /home/local/PROD/<user-id>/Snort
```

Une fois au sein de ce répertoire, nous pouvons procéder au téléchargement des modules nécessaires au fonctionnement de Snort :

```
wget https://www.snort.org/downloads/snort/daq-2.0.6.tar.gz
wget https://www.snort.org/downloads/snort/snort-2.9.11.1.tar.gz
wget https://www.snort.org/documents/snort-startup-script-for-centos
```

Nous pouvons désormais les décompresser :

```
tar xf daq-2.0.6.tar.gz
tar xf snort-2.9.11.1.tar.gz
```

III. Compilation des outils Snort :

Puisque nous venons de récupérer les sources de Snort, nous allons devoir les compiler manuellement. Commençons par le module DAQ de Snort. Nous allons devoir nous placer dans le répertoire '*daq-2.0.6*' avant de lancer la compilation :

```
./configure && make && make install
```

Si une erreur survient durant la compilation, le message d'erreur indiquera si un outil de compilation n'est pas présent sur la machine.
Le même procédé peut être effectué avec les sources de Snort, dans le répertoire '*snort.2.9.11.1*' :

```
./configure --enable-sourcefire && make && make install
```

Attention, il sera certainement nécessaire de modifier le path du système pour que Snort puisse trouver les outils DAQ et être correctement compilé :

```
which daq-modules-config
export PATH=$PATH:/usr/local/bin
```

Pour terminer, nous allons installer au bon endroit le script de démarrage automatique téléchargé plus tôt :

```
mv snort-startup-script-for-centos/ snortd/
chmod 755 snortd/
mv snortd/ /etc/init.d/
```

Snort est désormais installé sur la machine, nous allons donc commencer à configurer la sonde.

IV. Mise en place de l'environnement Snort :

Commençons par ajouter le groupe et l'utilisateur '*snort*' sur le système :

```
groupadd snort
useradd snort -r -s /sbin/nologin -c SNORT_IDS -g snort
```

Nous allons ensuite préparer les répertoires de Snort :

```
mkdir /etc/snort
mkdir /etc/snort/rules
mkdir /var/log/snort
mkdir /usr/local/lib/snort_dynamicrules
```

Et y attribuer les permissions nécessaires :

```
chmod -R 5775 /etc/snort
chmod -R 5775 /var/log/snort
chmod-R 5775 /usr/local/lib/snort_dynamicrules
chown -R snort:snort /etc/snort
chown -R snort:snort /var/log/snort
chown -R snort:snort /usr/local/lib/snort_dynamicrules
```

Les répertoires sont maintenant prêts à accueillir les différents fichiers de configurations provenants des sources Snort téléchargés plus tôt :

```
cp ~/snort_src/snort-2.9.11.1/etc/*.conf* /etc/snort
cp ~/snort_src/snort-2.9.11.1/etc/*.map /etc/snort
```

Nous pouvons désormais installer les règles Snort.

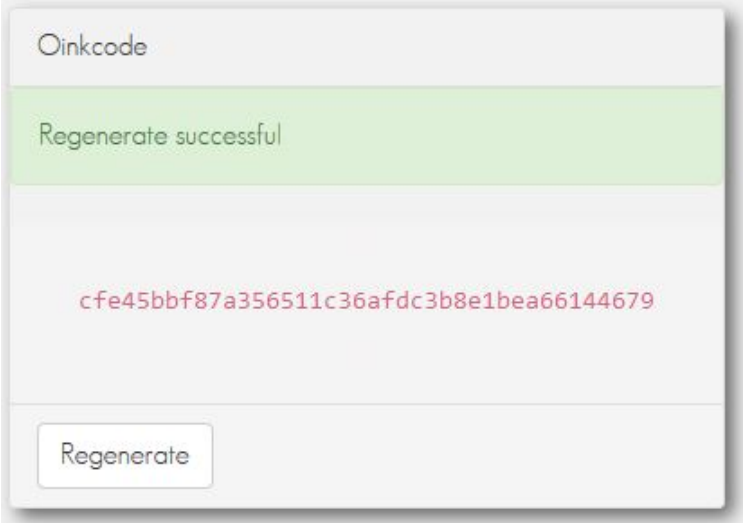
V. Ajout des règles Snort :

Il existe deux façons d'ajouter des règles Snort sur notre machine : en téléchargeant un package de règles communautaires limitées, ou en créant un compte sur le site officiel de Snort pour avoir accès à d'autres règles, plus précises et modulaires. Nous choisirons ce deuxième cas, afin d'avoir plus de matière sur laquelle travailler avec Snort.

Pour s'inscrire sur le site de Snort, visitez simplement l'adresse https://snort.org/.

Il sera nécessaire de fournir une adresse mail, et de valider un message de configuration.

Une fois l'inscription réalisée, nous allons pouvoir récupérer un identifiant unique qui nous permettra de télécharger les règles Snort étendues :



Il nous est désormais possible de télécharger ces règles Snort :

```
wget https://www.snort.org/rules/snortrules-snapshot-2990.tar.gz?oinkcode=
```

Bien Sûr, le code récupéré plus haut doit être placé à la suite de ce lien.
Une fois le fichier de règles décompressé, nous pouvons déplacer l'ensemble des fichiers dans le répertoire /etc/snort/ :

```
mv snortrules-snapshot-2990/* /etc/snort/
```

Les règles Snort sont maintenant valides et au bon endroit. Nous pouvons terminer par la configuration réseau du futur NIDS.

VI. Configuration réseau de la machine :

Pour que la Snort soit efficace, il faut que celle-ci soit située sur un point du réseau où l'ensemble du trafic passe. En positionnant la sonde en front sur le réseau, directement sur le point d'entrée et de sortie de tous

flux internet, juste derrière les différents firewall, notre rayon d'accès sera le plus haut possible.

Pour de meilleurs performances, il est conseillé d'utiliser Snort sur une interface réseau configuré en mode promiscuous. De cette façon, la machine sera capable d'observer le trafic en un point précis du réseau.
Enfin, un mirroring de port doit être mis en place sur le premier switch d'entrée du réseau, entre le port de trafic entrant et le port de la carte Snort promiscuous, afin de fournir toutes les traces d'évènements réseaux à la sonde Snort.

Travaillant sur une machine RedHat 7, la procédure de déploiement du mode promiscuous su l'interface eno50 et ens2f0 est la suivante :

```
ip link set eno50 promisc on
ip link set ens2f0 promisc on
```

Afin de rendre cette configuration persistante, même après un redémarrage, il sera nécessaire d'inscrire ces deux lignes sous */etc/rc.d/rc.local*, puis :

```
chmod u+x /etc/rc.d/rc.local
systemctl enable rc-local
systemctl start rc-local
```

VII. Configuration initiale de la sonde :

Avant de pouvoir tester l'installation de Snort, il est nécessaire d'effectuer quelques dernières modification au niveau des fichiers de configurations de Snort.

Il nous faudra éditer le fichier de configuration '*/etc/snort/snort.conf*', et modifier les lignes suivantes :

```
# Path to your rules files (this can be a relative path)
var RULE_PATH rules
var SO_RULE_PATH so_rules
var PREPROC_RULE_PATH preproc_rules
[...]
```

```
# Set the absolute path appropriately
var WHITE_LIST_PATH /etc/snort/rules
var BLACK_LIST_PATH /etc/snort/rules
[...]
# Recommended for most installs
output unified2: filename snort.log, limit 128
[...]
include $RULE_PATH/local.rules
```

Ces modifications indiquent le chemin jusqu'au fichier de règles Snort, et permettent de mettre en place des règles personnalisées via le fichier '*local.rules*'.

Pour valider notre configuration, il est possible de demander à Snort de tester le fichier de configuration avec la commande suivante :

```
snort -T -c /etc/snort/snort.conf
```

Si la configuration est validée par Snort, nous allons pouvoir mettre en place notre premier test.

Si la configuration est invalidée par snort, essayez d'effectuer les modifications suivantes :

```
# whitelist $WHITE_LIST_PATH/white_list.rules, \
#var WHITE_LIST_PATH /etc/snort/rules
blacklist $BLACK_LIST_PATH/blacklist.rules
```

VIII. Première phase de tests :

Pour tester l'installation de Snort, nous allons créer une règle temporaire dans le fichier '*/etc/snort/rules/local.rules*'. Cette règle va générer une alerte "icmp test" dès qu'un paquet ICMP est détecté par le NIDS :

```
alert icmp any any -> $HOME_NET any (msg:"ICMP test";
sid:10000001; rev:001;)
```

Lançons maintenant Snort en mode IDS, sur l'interface en mode promiscuous, en chargeant le fichier de configuration modifié plus tôt :

```
snort -A console -i eth0 -c /etc/snort/snort.conf -g snort -u snort
```

Attention à bien modifier le path du système de façon à ce qu'il pointe vers le binaire Snort :

```
PATH=$PATH:/usr/bin/
```

Détaillons la commande Snort :

- L'argument -A indique sous quel format le résultat de l'alerte est affiché. Ici, nous allons observer l'alerte directement dans la console.
- L'argument -i indique sur quel interface Snort doit écouter.
- L'argument -c pointe vers le fichier de configuration de Snort.
- L'argument -g et -u permettent de définir l'uid et le guid du processus.

Testons notre règle en effectuant un ping entre deux machines. Nous devrions observer une alerte Snort s'afficher dans la console :

```
07/12-11:20:33.501624  [xx] [1:10000001:1] ICMP test [**] [Priority: 0] {ICMP} xx.xx.xx.xx -> xx.xx.xx.xx
```

Si l'alerte est correctement générée, Snort dispose d'une configuration initiale stable. Attention à bien commenter la règle ICMP une fois le test effectué.

IX. Installation de Barnyard2 :

Depuis la version 2.9.3, Snort ne supporte plus nativement la communication avec les modules Prelude. Nous allons donc devoir passer par une couche tierce pour transformer les alertes Snort en alertes IDMEF compatibles avec les modules Prelude.

Barnyard2 est le processus capable d'une tel fonction. Les source du projet sont disponibles sur la page github : https://github.com/firnsy/barnyard2.

Nous allons donc télécharger Barnyard2 :

```
git clone https://github.com/firnsy/barnyard2
cd barnyard2
chmod 777 autogen.sh
./autogen.sh
```

ATTENTION : De nombreuses erreurs sont susceptibles d'apparaître à partir de cette étape, merci de consulter la section 'Troubleshooting' pour les résoudre de manière optimal.

Et le compiler en activant un mode de configuration compatible Prelude :

```
./configure --enable-prelude && make && make install
```

L'outil Barnyard2 est maintenant prêt à être configuré puis utilisé en tant que sous-couche entre Prelude-Manager et Snort.

X. Configuration de Barnyard2 :

La configuration de Barnyard2 ne nécessite pas beaucoup de modifications, puisqu'il s'agit uniquement d'indiquer que l'output de Snort doit être redirigé vers Prelude :

```
#/usr/bin/etc/barnyard2.conf
output alert_prelude: profile=snort
```

Et de rediriger les logs Snort vers un fichier sur lequel Barnyard2 écoute:

```
#/etc/snort/snort.conf
output unified2: filename merged.log, limit 128
```

Il ne reste plus qu'à enrôler le profil Snort aux profils Prelude.

XI. Enrôlement de la machine auprès du SIEM Prelude :

Pour enrôler le profil Snort à Prelude, il est nécessaire d'effectuer des manipulations en parallèle sur le Prelude-Manager et sur l'agent. Prelude-Manager n'autorise les enrôlements qu'en fournissant un mot

de passe unique à l'agent, et en communiquant sur un port spécifique à l'enrôlement.

Pour initier l'enrôlement, sur le serveur :

```
prelude-admin registration-server prelude-manager
```

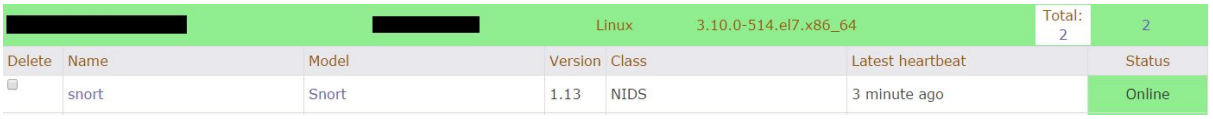
Et sur l'agent :

```
prelude-admin register snort "idmef:w admin:r" dctpldvlp01 --uid snort
--gid snort
```

Il sera ensuite demandé de coller le mot de passe temporaire généré par le serveur prelude sur l'agent. Il faudra confirmer l'enrôlement sur le serveur pour terminer la procédure.
Pour vérifier si l'agent dispose bien d'un profil Prelude valable :

```
prelude-admin list -l
```

La sonde enregistrée à l'instant ainsi que les identifiants qui y sont relatifs doivent apparaitre.
En lançant très rapidement Snort et Barnyard2, nous devrions générer un heartbeat, plaçant ainsi notre sonde Snort dans la liste des hôtes 'vivants' du SIEM :



XII. Phase de test n°2

Pour tester la 'sous-couche' Barnyard2 en temps qu'intermédiaire entre Snort et Prelude-manager, nous allons retirer le commentaire de la règles Snort customisée sous */etc/snort/rules/local.rules (voir partie VIII).*
Vérifier que le répertoire */var/log/snort* ne comporte que les répertoires et fichiers "*archived*" et "*alerts*".
Lancer ensuite, de manière séquentielle, Snort puis Barnyard2 :

```
/usr/local/bin/snort -c /etc/snort/snort.conf -i enoX
```

ou enoX est le nom de l'interface en bridge définie en partie IV.

Puis Barnyard2, pointant vers le fichier de log que génère Snort en temps réel :

```
/usr/local/bin/barnyard2 -c /etc/snort/barnyard2.conf -f /var/log/snort/merged.log.1519035656
-d /var/spool/prelude/snort/
```

Attendons maintenant qu'une requête icmp soit interceptée, et observons le résultat dans le SIEM :

Les alertes sont correctement retransmises à Prelude.
La configuration initiale de Snort est donc maintenant officiellement terminée ! Il ne reste plus qu'à affiner les règles, éliminer les faux positifs récurrents, et ajouter des règles customisées pour adapter Snort à son environnement réseau.

Additionnellement, des scripts de lancement en arrière plan peuvent être créés afin de faciliter l'administration du service sur la machine.

XIII. Troubleshoot :

• En suivant les documentations d'installation Snort classiques, il sera peut être demandé de configurer Snort avec l'option *--enabled-prelude*. Cette option n'est plus disponible depuis la version 2.9.3 de Snort. Merci de suivre ce document pour comprendre comment mettre en place une version récente de Snort avec le SIEM Prelude.

• En essayant de configurer Barnyard2, il sera possible de rencontrer plusieurs erreurs liées les unes aux autres. Si la commande *./configure --enable-prelude* retourne une erreur concernant libprelude, il sera nécessaire de compiler libprelude manuellement sur la machine, ou d'installer l'équivalent sous forme de paquets, si ceux-ci sont disponibles. Libprelude nécessite les dépendances suivantes :

libcrypt-devel, *libgnutls*. il sera également nécessaire d'ajouter */usr/lib64* au path de la session.

• Si une des étapes de compilation retournent une erreur de type :

```
_GL_WARN_ON_USE (gets, "gets is a security hole - use fgets instead");
```

Il manque certainement des outils de compilations. j'ai personnellement utilisé *gcc*, *gcc-c++*, *glibc-utils*, *glibc-header*, *glibc-common*, *glibc*.

• Si libprelude ne fonctionne pas, même après une compilation réussi sans message d'erreurs, il manque certainement un des langages requis à la librairie. Pour le vérifier, il suffit d'observer le résultat de la commande *./configure* sur la librairie Prelude :

```
*** Dumping configuration ***
        - Generate documentation   : no
        - LUA binding              : no
        - Perl binding             : yes
        - Python binding           : yes
        - Ruby binding             : no
        - Easy bindings            : yes
```

Si aucun des langages de la liste n'est compris dans la configuration, il sera nécessaire d'en installer au moins un.

• Si malgré tout, libprelude refuse de fonctionner avec la configuration de la documentation, ré-essayez avec la version 4.0.0rc3, depuis les build Github.

• Si barnyard2 indique qu'il ne peut être configuré sans avoir mis la main sur '*libprelude-config*', mais que libprelude est installé sur la machine :

```
cp [...]/libprelude.4.0.0rc3/libprelude-config /usr/bin/
```

• Si les commandes *snort* et *barnyard2* ne sont pas reconnues, ajouter */usr/local/bin* au PATH actuel.

• Si Barnyard2 n'arrive pas à ouvrir une session avec le manager Prelude distant, vérifiez le contenu de */usr/local/etc/prelude/profil/snort/config* et y indiquer l'adresse du Prelude-manager à contacter :

```
[prelude]
server-addr = x.x.x.x || y.y.y.y
```

• S'il vous est impossible de démarrer correctement Barnyard2 et Snort, vérifier que l'argument *-f* de Barnyard2 pointe bien vers le fichier de log au format unified2 de Snort.

• Si aucune des alertes n'est visible dans l'IHM du SIEM, faites pointer l'argument *-d* de Barnyard2 vers le spool Prelude : */var/spool/prelude/snort.*

• Dans le cas ou les configurations Preludes et Barnyard2 ne sont pas prises en compte, il est possible d'avoir un doublon des fichiers de configurations dans certains cas : une instance sous */etc/prelude*, ou une autre sous */usr/local/etc/prelude*.