Sessão 7: Sistema de detecção/prevenção de intrusos

Todas as atividades desta sessão serão realizadas na máquina virtual *FWGW1-G*, com pequenas exceções destacadas no enunciado de cada exercício.

As atividades apresentadas nesta seção foram baseadas no excelente tutorial de Don Mizutani, acessível em http://donmizutani.com/ , com adaptações para o cenário de laboratório deste curso.

1) Instalação do Snort

1. A seção 1.5 do manual oficial do Snort, *Packet Acquisition*, alerta para o fato que duas características de placas de rede e de processamento do kernel Linux podem afetar negativamente o funcionamento do IDS: LRO (*large receive offload*) e GRO (*generic receive offload*). Em particular, o fato de que as placas de rede podem remontar pacotes antes do processamento do kernel pode ser problemático, pois o Snort trunca pacotes maiores que o *snaplen* de 1518 bytes; em adição a isso, essas *features* podem causar problemas com a remontagem de fluxo orientada a alvo [1] do Snort.

   Na máquina *FWGW1-G*, instale o pacote `ethtool` e desative as *features* `lro` e `gro` da interface `eth0`. Se houver algum erro desativando as características, não se preocupe; siga para o próximo passo.

   ```
   # hostname
   FWGW1-A
   ```

   ```
   # apt-get install ethtool
   ```

   ```
   # ethtool -K eth0 gro off
   # ethtool -K eth0 lro off
   Cannot change large-receive-offload
   ```

2. Agora, vamos instalar o Snort. Mas, antes, um problema: note que o Snort não está disponível nos repositórios do `apt-get`:

   ```
   # apt-cache search snort | grep '^snort '
   ```

   Assim sendo, vamos ter que fazer a instalação do Snort por código-fonte. Primeiro, vamos instalar as dependências de compilação. Quando perguntado: *Install these packages without verification? [y/N]*, responda `y`.

```
# apt-get install bison           \
                  build-essential \
                  ca-certificates \
                  flex            \
                  libdumbnet-dev  \
                  libpcap-dev     \
                  libpcre3-dev    \
                  zlib1g-dev
```

Crie um diretório para download dos fontes do Snort, no qual trabalharemos, e entre nesse diretório.

```
# mkdir ~/src
# cd ~/src
# pwd
/root/src
```

3. Vamos compilar a instalar o DAQ (*Data Acquisition Library*) do Snort, usado para I/O de pacotes. Essa biblioteca permite ao Snort substituir chamadas diretas a funções da `libpcap` com uma camada de abstração que facilita operações em uma quantidade variada de interfaces de hardware e software sem serem necessárias mudanças ao Snort em si.

   Quando da escrita deste material, a versão mais recente da DAQ era a 2.0.6. Faça o (1) download, (2) extração, (3) configuração, (4) compilação e (5) instalação como indicam os passos a seguir.

```
# wget https://www.snort.org/downloads/snort/daq-2.0.6.tar.gz
(...)
```

```
# tar zxf daq-2.0.6.tar.gz
# cd daq-2.0.6/
```

```
# ./configure
```

```
# make
```

```
# make install
```

4. Volte ao diretório-pai (`/root/src`) e proceda com a instalação do Snort em si. Quando da escrita deste material, a versão mais recente era a 2.9.11.1. Faça o (1) download, (2) extração, (3) configuração, (4) compilação e (5) instalação como indicam os passos a seguir.

```
# cd ~/src
```

```
# wget https://www.snort.org/downloads/snort/snort-2.9.11.1.tar.gz
(...)
```

```
# tar zxf snort-2.9.11.1.tar.gz
# cd snort-2.9.11.1/
```

```
# ./configure --enable-sourcefire --enable-reload
```

```
# make
```

```
# make install
```

Vamos recriar os links e a *cache* para as bibliotecas dinâmicas do sistema, já que a instalação do Snort criou novas dessas bibliotecas. Em adição a isso, vamos criar um link simbólico apontando para o binário do Snort.

```
# ldconfig
# ln -s /usr/local/bin/snort /usr/sbin/snort
```

5. Teste o funcionamento do Snort.

```
# snort -V

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.11.1 GRE (Build 268)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2017 Cisco and/or its affiliates. All rights
reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using libpcap version 1.6.2
           Using PCRE version: 8.35 2014-04-04
           Using ZLIB version: 1.2.8
```

2) Configuração inicial do Snort

1. Vamos agora fazer a configuração do Snort. Como o software foi instalado manualmente, via código-fonte, temos que fazer diversos passos que normalmente são realizados pelo gerenciador

de pacotes da distribuição, quais sejam:

- Configurar uma conta de sistema não-privilegiada.
- Criar arquivos e diretórios padrão, vazios.
- Todos os arquivos de configuração serão salvos em `/etc/snort`, que será um *symlink* para `/usr/local/etc/snort`.
- Os registros de eventos serão gravados em `/var/log/snort`.

O *script shell* abaixo irá tratar de configurar os aspectos descritos acima:

```
#!/bin/bash

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

mkdir /usr/local/etc/snort
mkdir /usr/local/etc/snort/rules
mkdir /usr/local/etc/snort/preproc_rules
ln -s /usr/local/etc/snort /etc/snort

mkdir /usr/local/lib/snort_dynamicrules
mkdir /var/log/snort

touch /etc/snort/rules/white_list.rules
touch /etc/snort/rules/black_list.rules
touch /etc/snort/rules/local.rules

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

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

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

2. Iremos agora desabilitar (via comentários) todas as regras padrão do Snort já que iremos, em um passo futuro, usar o PulledPort para atualizar as regras pela Internet.

```
# sed -i 's/^\(include \$RULE\_PATH.*\)/#\1/' /etc/snort/snort.conf
```

3. Edite o arquivo de configuração do Snort e configure as redes a serem protegidas (variável `HOME_NET`), e as redes consideradas externas (variável `EXTERNAL_NET`).

```
# sed -i 's/^\(ipvar HOME\_NET\).*/\1 \[172.16.1.0\/24,10.1.1.0\/24\]/'
/etc/snort/snort.conf
```

```
# grep '^ipvar HOME_NET' /etc/snort/snort.conf
ipvar HOME_NET [172.16.1.0/24,10.1.1.0/24]
```

```
# sed -i 's/^\(ipvar EXTERNAL\_NET\).*/\1 \!\$HOME\_NET/' /etc/snort/snort.conf
```

```
# grep '^ipvar EXTERNAL_NET' /etc/snort/snort.conf
ipvar EXTERNAL_NET !$HOME_NET
```

4. Agora, vamos corrigir os caminhos de busca de regras do Snort, que encontram-se incorretos no arquivo de configuração original.

```
# sed -i 's/^\(var RULE\_PATH\).*/\1 \/etc\/snort\/rules/' /etc/snort/snort.conf
# sed -i 's/^\(var SO\_RULE\_PATH\).*/\1 \/etc\/snort\/so\_rules/'
/etc/snort/snort.conf
# sed -i 's/^\(var PREPROC\_RULE\_PATH\).*/\1 \/etc\/snort\/preproc\_rules/'
/etc/snort/snort.conf
# sed -i 's/^\(var WHITE\_LIST\_PATH\).*/\1 \/etc\/snort\/rules/'
/etc/snort/snort.conf
# sed -i 's/^\(var BLACK\_LIST\_PATH\).*/\1 \/etc\/snort\/rules/'
/etc/snort/snort.conf
```

   Verifique que as substituições funcionaram como esperado:

```
# grep '^var
[RULE_PATH\|SO_RULE_PATH\|PREPROC_RULE_PATH\|WHITE_LIST_PATH\|BLACK_LIST_PATH]'
/etc/snort/snort.conf
```

```
var RULE_PATH /etc/snort/rules
var SO_RULE_PATH /etc/snort/so_rules
var PREPROC_RULE_PATH /etc/snort/preproc_rules
var WHITE_LIST_PATH /etc/snort/rules
var BLACK_LIST_PATH /etc/snort/rules
```

5. Finalmente, vamos descomentar a linha que habilita regras customizadas locais, que usaremos em breve para testar o funcionamento do Snort.

```
# sed -i 's/^\#\(include \$RULE\_PATH\/local\.rules\)/\1/' /etc/snort/snort.conf
```

```
# grep '^include \$RULE\_PATH\/local\.rules' /etc/snort/snort.conf
include $RULE_PATH/local.rules
```

6. Teste o arquivo de configuração do Snort procurando por erros de sintaxe. Se tudo estiver correto, a penúltima linha deverá dizer `Snort successfully validated the configuration!`.

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

```
(...)
Snort successfully validated the configuration!
Snort exiting
```

7. Vamos criar uma regra customizada no Snort para testar se tudo está a contento. No arquivo `/etc/snort/rules/local.rules`, insira a linha:

```
alert icmp any any -> any any (msg:"ICMP packet from all, to all"; sid:10000001;
rev:001;)
```

Esta regra irá simplesmente levantar um alerta se o Snort detectar um pacote ICMP vindo de qualquer IP, qualquer porta, para qualquer IP, qualquer porta.

8. Descubra o IP público da máquina *FWGW1-G*:

```
# ip a s eth0 | grep '^ *inet ' | awk '{ print $2 }'
192.168.29.103/24
```

Agora, vamos rodar o Snort em modo console e testar o funcionamento da regra.

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

Em sua máquina física, envie alguns pacotes ICMP para o IP público da máquina *FWGW1-G*:

```
C:\>ping 192.168.29.103

Pinging 192.168.29.103 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.29.103:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

De volta à máquina *FWGW1-G*, note que o Snort gerou registros para cada um dos pacotes recebidos, como esperado:

```
09/04-09:10:33.691493  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:38.278164  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:43.279523  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
09/04-09:10:48.283261  [**] [1:10000001:1] ICMP packet from all, to all [**]
[Priority: 0] {ICMP} 192.168.29.102 -> 192.168.29.103
```

Observe, ainda, que os ICMP `echo-reply` enviados por sua máquina física não foram respondidos porque o firewall interno permite tráfego ICMP oriundo apenas das redes 172.16.1.0/24 e 10.1.1.0/24, como configurado na sessão 5.

```
# iptables -vn -L INPUT | grep ' prot\|icmp '
 pkts bytes target     prot opt in     out     source               destination
    1    84 ACCEPT     icmp --  *      *       172.16.1.0/24        0.0.0.0/0
icmptype 255
    0     0 ACCEPT     icmp --  *      *       10.1.1.0/24          0.0.0.0/0
icmptype 255
```

Finalize o Snort com CTRL+C, e comente a regra inserida no arquivo `/etc/snort/rules/local.rules`.

3) Habilitando o Snort no boot

1. Ainda devido ao fato de termos instalado o Snort via código-fonte, não temos instalado nenhum script de inicialização que permita iniciar/reiniciar/parar o Snort de forma automática (via comando `systemctl`), bem como configurá-lo para ser iniciado durante o boot da máquina.

   Crie o arquivo novo `/lib/systemd/system/snort.service`, com o seguinte conteúdo:

```
[Unit]
Description=Snort NIDS Daemon
After=syslog.target network.target

[Service]
Type=simple
ExecStart=/usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i
eth0 -D

[Install]
WantedBy=multi-user.target
```

2. Verifique que as permissões, usuário e grupo dono do arquivo estão corretos. Em seguida, crie um *symlink* do mesmo para o diretório `/etc/systemd/system`.

```
# chown root.root /lib/systemd/system/snort.service
# chmod 0644 /lib/systemd/system/snort.service
```

```
# ls -ld /lib/systemd/system/snort.service
-rw-r--r-- 1 root root 223 Sep  4 09:22 /lib/systemd/system/snort.service
```

```
# ln -s /lib/systemd/system/snort.service /etc/systemd/system/snort.service
```

```
# ls -ld /etc/systemd/system/snort.service
lrwxrwxrwx 1 root root 33 Sep  4 09:24 /etc/systemd/system/snort.service ->
/lib/systemd/system/snort.service
```

3. Recarregue as configurações de *daemons* do `systemd`. Em seguida, tente iniciar/verificar o estado/parar o Snort de forma automática usando o *initsystem* do sistema. Finalmente, adicione-o à sequência de boot.

```
# systemctl daemon-reload
```

```
# systemctl start snort.service
```

```
# systemctl status snort.service
● snort.service - Snort NIDS Daemon
   Loaded: loaded (/lib/systemd/system/snort.service; linked)
   Active: active (running) since Tue 2018-09-04 09:30:16 EDT; 4s ago
 Main PID: 5215 (snort)
   CGroup: /system.slice/snort.service
           └─5215 /usr/local/bin/snort -q -u snort -g snort -c
/etc/snort/snort.conf -i eth0 -D
```

```
# ps auxwm | grep '^snort'
snort     5215  0.0  2.1 127420 44596 ?        -    09:30   0:00
/usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i eth0 -D
snort        -  0.0    -      -     - -        Ssl  09:30   0:00 -
snort        -  0.0    -      -     - -        Ssl  09:30   0:00 -
```

```
# systemctl stop snort.service
```

```
# systemctl enable snort.service
Created symlink from /etc/systemd/system/multi-user.target.wants/snort.service to
/lib/systemd/system/snort.service.
```

```
# systemctl is-enabled snort.service
enabled
```

4) Configurando atualizações de regras de forma automática

1. O programa PulledPork nos permite receber definições de regras atualizadas periodicamente pela Internet, sempre que novas vulnerabilidade e *exploits* forem descobertos e divulgados.

   Primeiro, vamos instalar as dependências do PulledPork:

   ```
   apt-get install git                             \
                   libcrypt-ssleay-perl            \
                   liblwp-useragent-determined-perl
   ```

2. Dentro do diretório `/root/src`, faça o download do código-fonte do PulledPork. Em seguida, copie seus binários e arquivos de configuração para os locais apropriados.

   ```
   # cd ~/src/
   ```

   ```
   # git clone https://github.com/shirkdog/pulledpork.git
   Cloning into 'pulledpork'...
   remote: Counting objects: 1323, done.
   remote: Total 1323 (delta 0), reused 0 (delta 0), pack-reused 1323
   Receiving objects: 100% (1323/1323), 331.28 KiB | 343.00 KiB/s, done.
   Resolving deltas: 100% (884/884), done.
   Checking connectivity... done.
   ```

   ```
   # cd pulledpork/
   ```

   ```
   # cp pulledpork.pl /usr/local/bin/
   # chmod +x /usr/local/bin/pulledpork.pl
   ```

   ```
   # cp ./etc/*.conf /etc/snort
   ```

3. Crie os diretórios e arquivos de configuração padrão do PulledPork, vazios.

```
# mkdir /etc/snort/rules/iplists
# touch /etc/snort/rules/iplists/default.blacklist
```

4. Teste o funcionamento do PulledPork, verificando sua versão.

```
# pulledpork.pl -V
PulledPork v0.7.4 - Helping you protect your bitcoin wallet!
```

5. Vamos agora configurar o PulledPork. O primeiro passo é a obtenção de um *Oinkcode*, que é basicamente um número de registro com o `snort.org` que nos permitirá o download de listas de regras geradas pela comunidade.

    1. Acesse https://www.snort.org/ , e clique em *Sign In* no canto superior direito.
    2. Se você não possuir uma conta, clique em *Sign up*.
    3. Preencha os campos *Email* (use um email válido e acessível), *Password* e *Password confirmation*, marque a caixa *Agree to Snort license* e finalmente clique em *Sign up*.
    4. Acesse o e-mail informado no passo (3). Dentro de algum tempo, você deverá receber uma mensagem com o título *Confirmation instructions*. Abra-a e clique no link *Confirm my account*.
    5. Com a conta confirmada, faça login no site https://www.snort.org/ usando os dados informados anteriormente.
    6. No canto superior direito da página, clique no seu e-mail cadastrado, logo ao lado do ícone de logout.
    7. Na nova página, clique no menu *Oinkcode*. Deverá aparecer uma *string* de cerca de 40 caracteres no centro da tela. Copie-a, pois a usaremos em seguida.

6. Com o *Oinkcode* em mãos, vamos configurar o PulledPork. No comando abaixo, substitua o valor `OINKCODE` no começo do comando pelo código que você copiou no item (7) do passo anterior. Em seguida, execute-o no terminal.

```
# oc="OINKCODE" ; sed -i "s/^\(rule\_url\=https\:\/\/www\.snort\.org\/reg\-
rules\/|snortrules\-snapshot\.tar\.gz|\).*/\1${oc}/" /etc/snort/pulledpork.conf ;
unset oc
```

   Se tudo deu certo, você deverá ver seu *Oinkcode* ao final da linha de regras baixadas do site https://www.snort.org , como mostrado a seguir (nota: o *Oinkcode* abaixo é fictício):

```
# grep 'rule_url=https://www.snort.org/reg-rules' /etc/snort/pulledpork.conf
rule_url=https://www.snort.org/reg-rules/|snortrules-
snapshot.tar.gz|13eba036f37e80d0efb689c60af9e6daae810763
```

Falta substituir a distribuição-alvo padrão do PulledPork:

```
# sed -i 's/^\(distro=\).*/\1Debian-6-0/' /etc/snort/pulledpork.conf
```

```
# grep '^distro=' /etc/snort/pulledpork.conf
distro=Debian-6-0
```

7. Vamos testar as configurações do PulledPork, e fazer o download das listas de regras mais atualizadas.

```
# pulledpork.pl -c /etc/snort/pulledpork.conf -l
```

```
    https://github.com/shirkdog/pulledpork

      _____ ____
     `----,\    )
      `--==\\  /    PulledPork v0.7.4 - Helping you protect your bitcoin wallet!
       `--==\\/
     .-~~~~-.Y|\\_  Copyright (C) 2009-2017 JJ Cummings, Michael Shirk
  @_/        /  66\_  and the PulledPork Team!
    |    \   \   _(")
     \   /-| ||'--'  Rules give me wings!
      \_\  \_\\
 ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

(...)

Rule Stats...
        New:-------33914
        Deleted:---0
        Enabled Rules:----10841
        Dropped Rules:----0
        Disabled Rules:---23073
        Total Rules:------33914
IP Blacklist Stats...
        Total IPs:-----1470

Done
Please review /var/log/sid_changes.log for additional details
Fly Piggy Fly!
```

Se tudo deu certo, o PulledPork deve ter consolidado as regras baixadas no arquivo `/etc/snort/rules/snort.rules`. Verifique o tamanho e o número de linhas desse arquivo.

```
# du -sk /etc/snort/rules/snort.rules
18380   /etc/snort/rules/snort.rules
```

```
# wc -l /etc/snort/rules/snort.rules
38155 /etc/snort/rules/snort.rules
```

8. Finalmente, basta indicar ao Snort que esse arquivo seja usado em sua inicialização. Insira a linha `include $RULE_PATH/snort.rules` ao final do arquivo `/etc/snort/snort.conf`.

```
# echo 'include $RULE_PATH/snort.rules' >> /etc/snort/snort.conf
```

Pare todas as instâncias do Snort. Em seguida, inicie-o, e verifique seu uso de memória e processamento.

```
# systemctl stop snort
# ps auxwm | grep '^snort'
```

```
# systemctl start snort
```

```
# ps -eo 'rss,comm' | grep 'snort$'
548016 snort
```

```
# ps -eo 'cputime,comm' | grep 'snort$'
00:00:18 snort
```

9. Para que as regras se mantenham atualizadas, é necessário atualizá-las periodicamente. Crie um novo arquivo no diretório `/etc/cron.daily` que atualize as regras diariamente, com o seguinte conteúdo:

```
#!/bin/sh

test -x /usr/local/bin/pulledpork.pl || exit 0
/usr/local/bin/pulledpork.pl -c /etc/snort/pulledpork.conf -l
```

Verifique que o usuário/grupo dono e permissões do arquivo estão corretos.

```
# chown root.root /etc/cron.daily/pulledpork
# chmod 0755 /etc/cron.daily/pulledpork
```

Referências

[1] Novak, J. e Sturges, S. (2007). Target-Based TCP Stream Reassembly. [online] Pld.cs.luc.edu. Disponível em: http://pld.cs.luc.edu/courses/447/sum08/class5/novak,sturges.stream5_reassembly.pdf [Acessado em 4 Set. 2018].