Zarządzanie bezpieczeństwem w sieciach

dr inż. Robert Banasiak, mgr inż. Rafał Jachowicz,
Instytut Informatyki Stosowanej PŁ

Temat: Dynamiczny IDS (prosty IPS) z użyciem oprogramowania Guardian

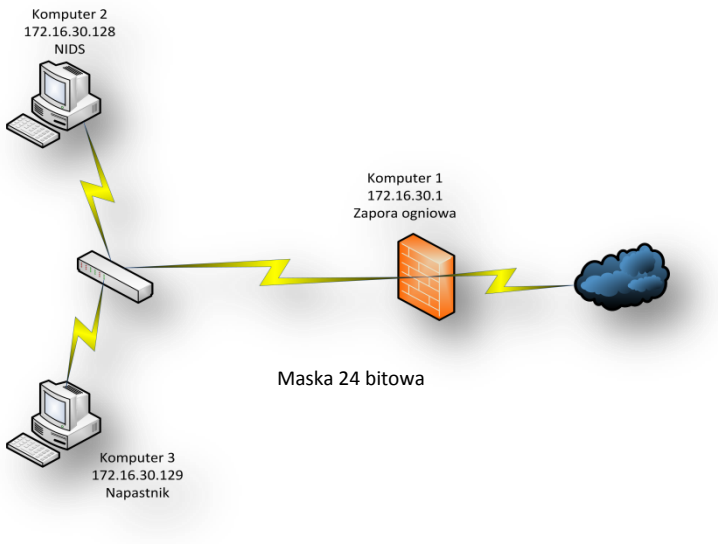
Na poprzednich ćwiczeniach, mimo że udało nam się zidentyfikować atakującego i poznać szczegóły ataku, który przeprowadził, nie udało nam się na ten atak w żaden sposób zareagować. Aby móc coś z tym zrobić, potrzebujemy narzędzia, które na podstawie tego, co wykrywa NIDS będzie modyfikowało reguły naszej zapory sieciowej. Do tego celu służy program Guardian. W niniejszym ćwiczeniu spróbujemy skonfigurować Guardiana tak by skutecznie blokował ataki.

Uwaga: Konfiguracja i uruchomienie Dyn IDS wymaga konta **root**!!!

Guardian to program darmowy, podobnie jak Snort. Źródło tego programu można pobrać z jego strony domowej http://www.chaotic.org/guardian/guardian-1.7.tar.gz. Sposób rozpakowania:

```
tar xzf guardian-1.7.tar.gz
```

Guardian jest skryptem napisanym w języku Pearl, który współpracuje z systemem Snort. Jego główne zadanie polega na zarządzaniu bezpieczeństwem w sieci komputerowej poprzez automatyczna zmianę zabezpieczeń systemu zapory sieciowej. Funkcję tę realizuje na podstawie alertów generowanych przez program Snort.



Sprawdź czy masz zainstalowany pakiet Snort (**jeżeli nie to sudo apt-get install snort -> w trakcie instalacji ustaw sieć nasłuchiwania jako 172.16.30.0 z maską 24 bit oraz interfejs nasłuchujący zgodny z siecią wewnętrzną w VirtualBox -> (*)**).

po zainstalowaniu snorta restartujemy to środowisko: /etc/init.d/snort stop -> /etc/init.d/snort start
Nie uruchamiamy snorta bezpośrednio z linii poleceń!!!.

Po uprzednim pobraniu i rozpakowaniu źródeł Guardiana możemy przystąpić do konfiguracji tego programu. Plikiem konfiguracyjnym Guardiana jest *guardian.conf*. Przed uruchomieniem należy dostosować konfigurację do naszych potrzeb.

Pierwszą opcją, którą musimy ustawić, jest opcja `HostIpAddr`. Powinna ona wskazywać na adres IP komputera z zainstalowanym systemem NIDS. **Zostawiamy tę linię zakomentowaną**!!!

Kolejna opcja to nazwa interfejsu (*), na którym działa NIDS

```
Interface  eth???
```

Następną informacją jest ścieżka dostępu do pliku, w którym Guardian będzie zapisywał wszystkie czynności związane z własnym uruchamianiem i modyfikowaniem zapory sieciowej.

```
LogFile          /var/log/guardian.log
```

Plik ten musimy stworzyć w systemie sami i ustawić dla niego odpowiednie prawa dostępu (polecam 777). Możemy zrobić to w następujący sposób z linii poleceń:

```
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ clear

rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ sudo touch /var/log/guardian.log
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ sudo chmod 700 /var/log/guardian.log
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$
```

Jeśli nie utworzylibyśmy tego pliku, Guardian wyświetlałby wszystkie komunikaty bezpośrednio na ekranie -> STDOUT. Kolejny parametr to ścieżka dostępu do pliku, w którym znajdują się logi Snorta:

```
AlertFile  /var/log/snort/alert
```

Ostatnim parametrem ważnym dla nas jest parametr TimeLimit wyrażony w sekundach (np. 120 sekund). Określa on, na jak długo atakujący będzie blokowany przez naszą zaporę sieciową.

```
TimeLimit  60
```

Kolejnym krokiem jest utworzenie plików *guardian_block.sh* i *guardian_unblock.sh* (pliki mają znajdować się w głównym katalogu Guardiana), które będą zawierały polecenia wykonywane w momencie wykrycia ataku i po upływie czasu określonego parametrem *TimeLimit*. W naszym przypadku będą to polecenia dla iptables, które będą odpowiednio dodawały regułę blokującą i odblokowującą ruch spod konkretnego adresu IP. Adres IP, który ma być blokowany lub odblokowywany, jest przekazywany do skryptu jako parametr $source. Skrypty te powinny mieć następującą zawartość:

```
Guardian_block.sh
source=$1
interface=$2
/sbin/iptables -A INPUT -s $source -i $interface -j DROP

Guardian_unblock.sh
source=$1
interface=$2
/sbin/iptables -D INPUT -s $source -i $interface -j DROP
```

Oczywiście, reakcja nie musi polegać na blokowaniu ruchu na zaporze sieciowej. W powyższych skryptach możemy umieścić dowolne ciągi poleceń podejmujące dowolną działalność. W katalogu scripts w źródłach Guardiana znajdziemy przykładowe skrypty przeznaczone dla różnych typów zapór sieciowych, a także takie, które modyfikują np. tablicę routingu w momencie wykrycia ataku.

W celu poprawnego uruchomienia Guardiana należy do zmiennej systemowej *PATH* dodać ścieżkę, na której znajdują się pliku Guardiana. W naszym przypadku pliki te znajdują się w katalogu */home/(zalogowany user)/Guardian-1.7/*. Katalog */home/(zalogowany user)/Guardian-1.7/* dodajemy do ścieżki systemowej w sposób pokazany na rysunku.

```
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ PATH=$PATH:/home/rbanasi/guardian-1.7
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$
```

Sposób uruchomienia programu Guardian widać na kolejnej ilustracji (w przypadku „świeżej" instalacji Linux-a może być niezbędne doinstalowanie pakietu libperl4-corelibs-perl -> poprzez apt-get)

```
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ ./guardian.pl -c /home/rbanasi/guardian-1.7/
guardian.conf
OS shows Linux
My ip address and interface are: 172.16.30.128       eth2
Loaded 0 addresses from /etc/guardian.ignore
Becoming a daemon..
```

Przed wykonaniem skanowania z komputera atakującego sprawdzamy tablice zapory sieciowej komputera z naszym systemem IDS. Jak można zobaczyć na poniższym rysunku nie zawiera ona żadnych reguł. Cały ruch przychodzący (łańcuch INPUT) oraz wychodzący (łańcuch OUTPUT) i przekazujący (łańcuch FORWARD) jest w tym momencie przepuszczany. Domyślna polityka zarządzania bezpieczeństwem sieci (policy) jest ustawiona na ACCEPT, co oznacza, że ruch we wszystkie strony nie jest w żaden sposób ograniczony.

```
rbanasi@rbanasi-VirtualBox:~/guardian-1.7$ sudo iptables -L -v
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination


Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination


Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

rbanasi@rbanasi-VirtualBox:~/guardian-1.7$
```

Zadania

1. Sprawdź zawartość łańcuchów IPTABLES (przykładowy rysunek powyżej)
2. Z komputera 172.16.30.129 wykonaj skanowanie typu NULL i sprawdź reakcję naszego systemu NIDS.
3. Sprawdź zawartość pliku */var/log/guardian.log*.
4. Korzystając z komputera 172.16.30.129 sprawdź komunikację ICMP z komputerem 172.16.30.128. Czy komunikacja funkcjonuje?
5. Odczekaj czas określony parametrem *TimeLimit*
6. Sprawdź ponownie zawartość łańcuchów IPTABLES
7. Korzystając z komputera 172.16.30.129 sprawdź ponownie komunikację ICMP z komputerem 172.16.30.128. Czy komunikacja tym razem funkcjonuje?