Knock: "O poderoso abre/fecha porta".

Autoria de Giancarlo Rubio
17/10/2006
Última Atualização 18/10/2006

Nesse artigo vamos conhecer o knock, daemon que permite que portas sejam abertas ou fechadas por demanda, dependendo "de quem bate à porta", ou seja de acordo com o IP originando a requisição. Esse artigo trata de um exemplo prático, simples e claro do funcionamento do knock.
Este título foi o que chegou mais perto para a definição da função do Knock.

O que é Knock?

Knock é um poderoso daemon, que gerencia a abertura/fechamento das portas de um determinado computador a apenas os usuários que obtiverem a permissão ou conhecerem.

Como funciona? O cliente envia requisições (bate) a uma série de portas predefinidas e a abre a porta desejada. No nosso caso a porta 22 do ssh será aberta.

Requisitos prévios para nossos exercícios são algum conhecimento básico em:
ipfw ports

Instalando Knock

Sempre é recomendável atualizar a árvore dos ports para a instalação de qualquer aplicação.

Para aqueles que ainda não usam portsnap vamos baixar e extrair

#portsnap fetch extract

Atualizando o ports

#portsnap fetch update

Instalando o Knock

#cd /usr/ports/security/knock
make install clean

São mostradas as seguintes opções:
Cliente, aquele que faz a requisição remota para abertura das portas; Servidor, aquele responsável por abrir a porta.

O cliente não é tão necessário, mais tarde explico porque, mas o servidor é obrigatório para o desenrolar do artigo.

O arquivo padrão está em /usr/local/etc/knockd.conf.sample

Copiamos para o nome certo, sem .sample

#cp /usr/local/etc/knockd.conf.sample /usr/local/etc/knockd.conf

Vamos ao arquivo de configuração, com meus comentários

#vi /usr/local/etc/knockd.conf

```
[options]
        logfile = /var/log/knockd.log     # Local que serão registrados os logs de abertura de portas
        interface = fxp0        #Interface que vai ouvir, normalmente a internet
#Abre o ssh
[openSSH]
        sequence    = 7000,8000,9000    #Sequencia de portas que o cliente deve "bater" para logar
        seq_timeout = 5           #Intervalo de tempo (milisegundos) para uma requisição entre portas
        command     = /sbin/ipfw -q add pass proto tcp src-ip %IP% dst-port 22    #Regra a ser adicionada no firewall
        tcpflags    = syn        #Podem ser usados as seguintes opções:syn,fin,ack e !ack,!syn,!fin
#Fecha o Ssh
[closeSSH]
```

```
sequence    = 9000,8000,7000
seq_timeout = 5
command     = /sbin/ipfw -q delete pass proto tcp src-ip %IP% dst-port 22
tcpflags    = syn
```

Iniciando o daemon

#/usr/local/etc/rd.d/knockd start

Não sei porque o meu não iniciou assim, mais em todo caso iniciei na mão(-d daemon, -v verbose

#knockd -dv

Vamos ao funcionamento
Primeiro bloquei o ssh na sua máquina

```
su-2.05b# ipfw show
00100   0     0 deny ip from any to me dst-port 22
65535     0     0 allow ip from any to any
su-2.05b#
```

De fora da minha rede fiz um portscan, para ver as portas abertas usando o nmap (/usr/ports/security/nmap)

su-2.05b# nmap -sS 201.21.140.208

```
Starting Nmap 4.01 ( http://www.insecure.org/nmap/ ) at 2006-10-17 21:45 BRT
Interesting ports on virtua-cwbas128-208.ctb.virtua.com.br (201.21.140.208):
(The 1661 ports scanned but not shown below are in state: closed)
PORT    STATE    SERVICE
21/tcp  filtered ftp
22/tcp  filtered ssh
25/tcp  filtered smtp
80/tcp  filtered http
135/tcp filtered msrpc
136/tcp filtered profile
137/tcp filtered netbios-ns
138/tcp filtered netbios-dgm
139/tcp filtered netbios-ssn
445/tcp filtered microsoft-ds
587/tcp open     submission

Nmap finished: 1 IP address (1 host up) scanned in 41.260 seconds
su-2.05b#
```

A porta está fechada ok?

Vamo abri-la

Existem 2 formas (De fora da rede)

1- Usando o knock cliente

#knock 201.21.140.208 7000 8000 9000

2-Usando telnet, atente para o fato de serem regras simlultaneas, pois definimos o tempo em 5 milisegundos

#telnet 201.21.140.208 7000;telnet 201.21.140.208 8000;telnet 201.21.140.208 9000

Nosso /var/log/knockd.log

[2006-10-17 21:49] 200.0.0.0: openSSH: Stage 1

[2006-10-17 21:49] 200.0.0.0: openSSH: Stage 2
[2006-10-17 21:49] 200.0.0.0: openSSH: Stage 3
[2006-10-17 21:49] 200.0.0.0: openSSH: OPEN SESAME
[2006-10-17 21:49] openSSH: running command: /sbin/ipfw -q add pass proto tcp src-ip 200.0.0.0 dst-port 22

meu ipfw

su-2.05b# ipfw show
00100 4 176 deny ip from any to me dst-port 22
65100 0 0 allow ip from any to any proto tcp src-ip 200.0.0.0 dst-port 22
65000 6488 310002 allow ip from any to any
su-2.05b#

Pronto agora a porta esta aberta. Ele adicionou a regra 65100 com o ip da maquina cliente. Use a vontade a aplicação até necessitar que seja fechada a porta [closeSSH].

1- Usando o knock cliente

#knock 201.21.140.208 9000 8000 7000

2-Usando telnet, atente para o fato de serem regras simlultaneas, pois definimos o tempo em 5 milisegundos

#telnet 201.21.140.208 9000;telnet 201.21.140.208 8000;telnet 201.21.140.208 7000

Vamos verificar

su-2.05b# nmap -sS 201.21.140.208

Starting Nmap 4.01 (http://www.insecure.org/nmap/) at 2006-10-17 22:03 BRT
Interesting ports on virtua-cwbas128-208.ctb.virtua.com.br (201.21.140.208):
(The 1661 ports scanned but not shown below are in state: closed)
PORT STATE SERVICE
21/tcp filtered ftp
22/tcp filtered ssh
25/tcp filtered smtp
80/tcp filtered http
135/tcp filtered msrpc
136/tcp filtered profile
137/tcp filtered netbios-ns
138/tcp filtered netbios-dgm
139/tcp filtered netbios-ssn
445/tcp filtered microsoft-ds
587/tcp open submission

Nmap finished: 1 IP address (1 host up) scanned in 20.377 seconds
su-2.05b#

Porta fechada!!!!!

Simples não??

Resalvas:
Nunca se sabe até aonde uma aplicação dessas deve gerenciar regras no firewall, lembre-se que ele executa como root, logo você é responsável pelo que faz.Evite usar essas portas default, coloque qualquer coisa diferente disso, ja existem scanner que varrem estas portas.