Firewall fácil com o Shorewall

Domando o fogo, parte 2

Na segunda parte de nosso tutorial de uso do poderoso Shorewall, aprenda a criar um firewall mais complexo e a proteger sua rede com muita praticidade.

por Tarcísio C. Espínola

Ramon Gonzalez - www.sxc.hu

O primeiro artigo desta série[1] apresentou o *Shorewall*, uma solução de Código Aberto para criação e gerenciamento de firewalls em diversos sistemas operacionais, incluindo, é claro, o Linux. Neste segundo artigo, vamos explicar algumas configurações mais avançadas dessa poderosa ferramenta.

Mascaramento

Para que o servidor firewall (veja na **figura 1**) funcione como um roteador e permita a passagem de pacotes entre as zonas *LOC* (rede local) e *NET* (a Internet), é necessário ativar o encaminhamento (*forward*) e o mascaramento (*postrouting*) entre as suas respectivas interfaces. No Shorewall, essa configuração deve ser feita no arquivo `/etc/shorewall/masq`, como no **exemplo 1**.

Com essa configuração, ativamos o encaminhamento das requisições vindas da interface da rede local (`eth1`) para a interface da Internet (`eth0`), assim como seu mascaramento. Não é necessário ativar o roteamento entre as zonas *DMZ* e *NET*; para isso, usaremos outra técnica: *Proxyarp*. Neste artigo, vamos abrir apenas os aspectos mais relevantes no contexto local O manual do arquivo de configuração (`man shorewall-masq`) contém todos os detalhes.

Proxyarp

O Shorewall não exige qualquer configuração do tipo *DNAT*, em que as requisições ao servidor firewall são desviadas para máquinas internas para montar uma DMZ. Por meio da função Proxyarp, cada servidor da DMZ será configurado com o seu próprio IP fixo e válido, pertencente à mesma rede do IP válido do servidor firewall. Isso significa que é como se cada servidor da DMZ estivesse conectado diretamente ao roteador, exatamente como fizemos com a interface `eth0` do servidor firewall[1].

Assim, poderemos adicionar facilmente mais servidores à DMZ sempre que necessário, bastando, para isso, configurá-los com os IPs fornecidos pela operadora e conectá-los no mesmo switch dos demais membros da DMZ. O limite, como se vê, é o número de endereços disponibilizados pela operadora. Vale lembrar que a interface `eth2` do servidor firewall também deverá estar conectada a esse switch.

Para configurar o Proxyarp, é preciso definir no arquivo `/etc/shorewall/proxyarp` os IPs de cada servidor da DMZ. Os servidores *mail.meu_site.com.br* e `www.meu_site.com.br`, em nosso exemplo

Exemplo 1: Mascaramento

```
#INTERFACE  SOURCE  ADDRESS  PROTO  PORT(S) IPSEC  MARK
eth0        eth1
# Não apagar esta última linha.
```

Exemplo 2: Configuração do Proxyarp

```
#ADDRESS      INTERFACE  EXTERNAL  HAVEROUTE  PERSISTENT

# servidor mail.meu_site.com.br:
200.166.161.194  eth2       eth0      no
# servidor www.meu_site.com.br:
200.166.161.195  eth2       eth0      no
# Não apagar esta última linha.
```

(**figura 1**) deveriam ser configurados conforme o **exemplo 2**.

Note que a opção `INTERFACE` precisa representar a interface do servidor firewall que foi vinculada à zona DMZ, enquanto a opção `EXTERNAL` sinaliza a interface do servidor firewall ligada à zona NET. A opção `HAVEROUTE` deve ser definida como `no` para que o próprio Shorewall se encarregue de criar as rotas para os servidores da DMZ na tabela de roteamento do servidor firewall.

Arquivo de configuração

Com a DMZ definida com facilidade, vamos agora definir algumas configurações específicas do Shorewall que devem ser realizadas em seu arquivo de configuração `/etc/shorewall/shorewall.conf`. Nele, são definidos alguns parâmetros utilizados na inicialização do firewall e, para o nosso exemplo, será necessário alterar as seguintes opções:

- `STARTUP_ENABLED=Yes`: Permite que o Shorewall seja iniciado;
- `SHOREWALL_COMPILER=perl`: Define o interpretador que será utilizado pelo Shorewall na compilação de suas regras (uma novidade na versão 4). A opção `perl` é a mais recente e também muito rápida. No entanto, ainda existem algumas restrições ao seu uso. A opção `shell` é mais antiga e lenta, e será descontinuada no futuro. Em nosso exemplo, usaremos `perl`;
- `DISABLE_IPV6=No`: Por padrão, o suporte ao protocolo IPv6 vem desativado no Shorewall, o que gera uma mensagem de alerta durante a compilação em distribuições como *RHEL*, *Fedora* e *CentOS*. Já no *Debian* e no *Ubuntu*, não existe esse problema.

Como em todos os outros arquivos de configuração citados, a página de manual do `shorewall.conf` (`man shorewall.conf`) lista todas as suas funcionalidades.

Regras

As regras do firewall são a parte que pode mudar com freqüência, diferentemente de todas as configurações até agora realizadas.

O firewall construído até aqui já pode ser iniciado, mas suas regras de política impedirão a passagem de qualquer tráfego entre zonas diferentes. É no arquivo de regras, `rules`, que são definidos os serviços que cada zona poderá acessar nas demais, da liberação de endereços e portas. Isso exige um estudo minucioso de cada regra, o que pode ser inconveniente a princípio, mas é muito importante.

A sintaxe de cada linha do arquivo de configuração é simples e consiste em informar a ação a ser tomada, origem e destino do pacote, seu protocolo e sua porta de destino. Vejamos como definir cada um desses campos (a página de manual *shorewall-rules* oferece todos os detalhes).

A **ação** determina o que a regra vai fazer com as conexões que coincidirem com o restante dos campos. As ações mais comuns são *ACCEPT* (aceitar o acesso), *REJECT* (rejeitar o acesso e informar ao solicitante) e *DROP* (simplesmente rejeitar o acesso).

A **origem** representa a zona e, opcionalmente, a máquina que acessará o serviço. Caso a máquina seja omitida, a regra será válida para todas as máquinas daquela zona. É permitido o uso do curinga *all* para representar todas as zonas.

O **destino** define a zona à qual pertence o serviço. Assim como no campo *origem*, pode-se especificar também a máquina nesse campo. Da mesma forma, o curinga *all* também

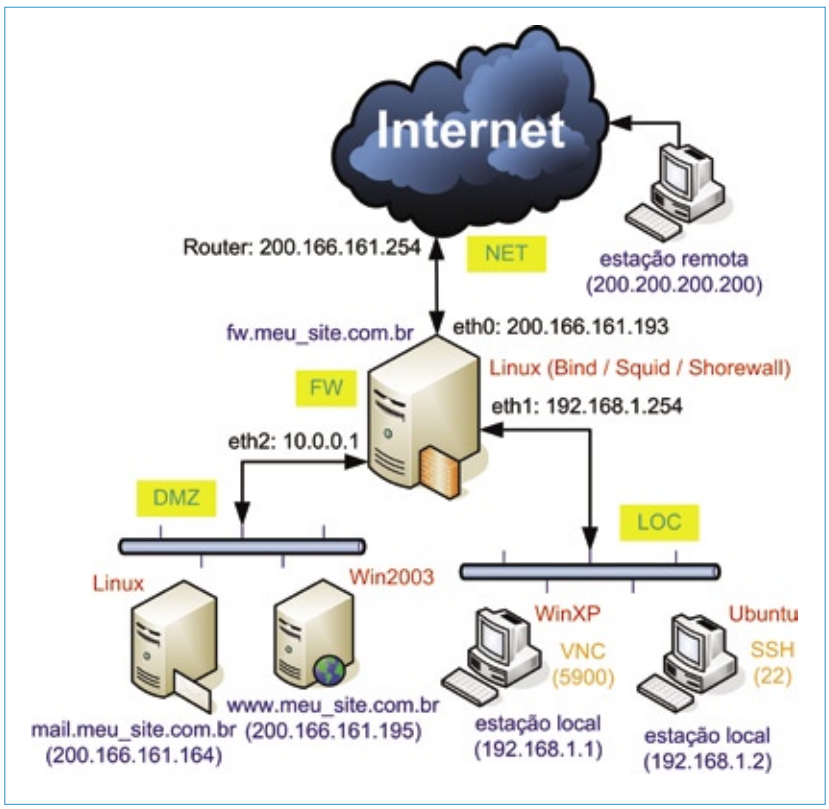

Figura 1 A rede padrão numa pequena empresa usada neste artigo possui dois servidores numa DMZ e duas estações na rede interna, ambas conectadas ao firewall, que as liga à Internet.

pode ser usado para representar todas as zonas.

No campo **protocolo**, auto-explicativo, as possibilidades são *tcp*, *udp* e *icmp*.

O campo **porta de destino** informa a(s) porta(s) afetada(s) pela regra. É permitido usar uma porta ou uma faixa de portas separadas por dois pontos, como `1024:1050`, por exemplo. Caso seja omitida a primeira ou a última porta na faixa, a porta 0 será usada como primeira ou a 65535 será considerada a última. Com `0:`, representamos todas as portas.

É importante ter em mente que um host pode representar tanto uma estação quanto um servidor, e que podemos utilizar tanto IPs quanto nomes totalmente qualificados em nossas regras.

Zona LOC

Vamos começar pelas regras referentes à zona LOC, definindo quais serviços suas estações devem conseguir acessar em cada uma das outras zonas.

O **exemplo 3** exibe o trecho inicial do arquivo `rules`, incluindo as definições de regras da zona LOC. Nos destinos da zona DMZ (**linhas 9 a 19**), está permitido o acesso aos serviços SSH (porta 22), SMTP (25), POP (110) e IMAP (445) no servidor *mail.meu_site.com.br*, e somente aos serviços web (portas 80 e 443) no servidor `www.meu_site.com.br`. Por último, fica permitido o uso do ping e mensagens de erro ICMP a partir de qualquer máquina da zona LOC para qualquer servidor da zona DMZ.

Note que nessas regras foi usada uma macro do Shorewall para facilitar ainda mais a escrita de regras. Com elas, podemos digitar, na **linha 16**:

```
Web/ACCEPT   loc   dmz:www.meu_
➥site.com.br
```

em vez de:

```
ACCEPT   loc   dmz:www.meu_site.
➥com.br   tcp   80
ACCEPT   loc   dmz:www.meu_site.
➥com.br   tcp   443
```

Existem macros para a maioria dos serviços mais comuns, o que realmente ajuda muito o administrador a usar melhor seu tempo. Elas estão disponíveis no diretório `/usr/share/shorewall/`.

Exemplo 3: Arquivo de regras, rede local

```
001 #ACTION  SOURCE  DEST  PROTO  DEST  SOURCE  ORIGINAL  RATE  USER/  MARK
002 #                            PORT  PORT(S) DEST      LIMIT GROUP
003 #SECTION ESTABLISHED
004 #SECTION RELATED
005 SECTION NEW
006
007 ###### Acessos da Zona LOC (Rede local)
008
009 ## Regras: LOC -> DMZ ##
010 # de todas as estações locais -> para mail.meu_site.com.br
011 SSH/ACCEPT      loc     dmz:mail.meu_site.com.br
012 SMTP/ACCEPT     loc     dmz:mail.meu_site.com.br
013 POP/ACCEPT      loc     dmz:mail.meu_site.com.br
014 IMAP/ACCEPT     loc     dmz:mail.meu_site.com.br
015 # de todos os hosts da LOC -> para www.meu_site.com.br
016 Web/ACCEPT      loc     dmz:www.meu_site.com.br
017 # de todos os hosts da LOC -> para: todos os hosts da DMZ
018 Ping/ACCEPT     loc     dmz
019 AllowICMPs      loc     dmz
020
021 ## Regras: LOC -> NET ##
022 # acesso total do host Ubuntu (tcp/udp) -> para: todos os hosts da NET.
023 ACCEPT        loc:192.198.1.2   net   tcp   0:  # todos os serviços tcp
024 ACCEPT        loc:192.198.1.2   net   udp   0:  # todos os serviços udp
025 # de: todos os hosts da LOC -> para: Redes da CEF.
026 Web/ACCEPT      loc     net:200.201.166.0/24
027 Web/ACCEPT      loc     net:200.201.173.0/24
028 Web/ACCEPT      loc     net:200.201.174.0/24
029 # de: todos os hosts da LOC -> para: todos os hosts da NET.
030 SSH/ACCEPT      loc     net
031 ACCEPT          loc     net     tcp     3456   # ReceitaNet
032 Ping/ACCEPT     loc     net
033 AllowICMPs      loc     net
034
035 ## Regras: LOC -> FW ##
036
037 # de: todos os hosts da LOC -> para: FW.
038 DNS/ACCEPT      loc     fw
039 SSH/ACCEPT      loc     fw
040 ACCEPT          loc     fw      udp     67     # DHCP Server
041 Ping/ACCEPT     loc     fw
042 AllowICMPs      loc     fw
```

As **linhas 21 a 33** do **exemplo 3** mostram que a estação com IP terminando em `.2` pode acessar qualquer serviço na zona NET, enquanto as demais estações têm permissão para acessar apenas os serviços web dos servidores da Caixa Econômica Federal, que se encontram nas sub-redes 200.201.166.0/24, 200.201.173.0/24 e 200.201.174.0/24. Além disso, todas as estações de LOC podem acessar os serviços SSH, *ReceitaNet* e ICMP.

Os acessos da zona LOC à zona FW são definidos nas **linhas 35 a 42** e permitem o tráfego dos serviços DNS, SSH, DHCP e ICMP.

Zona DMZ

Os servidores da DMZ somente terão acesso a serviços estritamente necessários: web na Internet, para o download de atualizações, e DNS no servidor firewall, além de ICMP nas duas zonas, como mostra o **exemplo 4**.

Note a facilidade para bloquear o acesso da zona DMZ à zona LOC: basta não permitir nada (**linha 60**), pois nossa política padrão já está definida para barrar esse tipo de acesso. O motivo dessa proibição de acesso é que, caso um servidor da DMZ seja invadido, este não terá acesso algum às estações da rede local.

Acesso remoto

É raro oferecer algum serviço nas estações da rede local. Uma possível exceção, no entanto, é o acesso remoto à área de trabalho, muito útil quando se está viajando, por exemplo. O **exemplo 5** mostra como permitir que uma estação remota na Internet (com IP 200.200.200.200) tenha acesso ao serviço *VNC* que será executado na estação *WinXP* e ao serviço *SSH* da estação Ubuntu, ambos na zona LOC.

Nas **linhas 67 e 70**, usamos o DNAT para permitir que as requisições feitas ao servidor nas portas 5900 e 2222 sejam redirecionadas, respectivamente, para as estações WinXP e Ubuntu, nas portas 5900 e 22. Além disso, o início das duas regras informa que tudo será registrado nos logs (`info`).

A sintaxe das regras do tipo DNAT é diferente da sintaxe padrão de regras de simples filtragem. Uma regra de DNAT exige os campos **ação**, **zona:máquina**, **destino:máquina:porta**, **protocolo** e portas de **origem** e **destino**.

É importante ressaltar que acessos do tipo DNAT às estações locais devem ser utilizados com bastante cautela, pois pode expor uma zona inteira ao acesso de invasores.

No acesso à DMZ, por outro lado, é necessário permitir o tráfego dos serviços oferecidos publicamente pelos servidores (**linhas 72 a 81**). Note, no entanto, que não permitimos o serviço SSH externamente, por medida de segurança.

Exemplo 4: Arquivo de regras, servidores

```
044 ###### Acessos da Zona DMZ
045
046 ## Regras: DMZ -> NET ##
047 # de: todos os hosts da DMZ
➥-> para: todos os hosts da NET.
048 Web/ACCEPT      dmz     net
049 Ping/ACCEPT     dmz     net
050 AllowICMPs      dmz     net
051
052 ## Regras: DMZ -> FW ##
053 # de: todos os hosts da DMZ
➥-> para: FW.
054 DNS/ACCEPT      dmz     fw
055 Ping/ACCEPT     dmz     fw
056 AllowICMPs      dmz     fw
057
058 ## Regras: DMZ -> LOC ##
059 # Acesso não permitido.
060
```

Exemplo 5: Arquivo de regras, acesso remoto

```
061 ###### Acessos da Zona NET (Internet).
062
063 ## Regras: NET -> LOC ##
064 # Atenção: Use estas regras com cautela!
065 # redirecionamento,
066 # de: fw.meu_site.com.br:5900 -> para: 192.168.1.1:5900 (WinXP)
067 DNAT:info net:200.200.200.200 loc:192.168.1.1:5900 tcp 5900 - fw.meu_site.com.br -  -
068 # redirecionamento,
069 # de: fw.meu_site.com.br:2222 -> para: 192.168.1.2:22 (Ubuntu)
070 DNAT:info net:200.200.200.200 loc:192.168.1.2:22   tcp 2222 - fw.meu_site.com.br -  -
071
072 ## Regras: NET -> DMZ ##
073 # de: todos os hosts da NET -> para: mail.meu_site.com.br
074 SMTP/ACCEPT     net     dmz:mail.meu_site.com.br
075 POP/ACCEPT      net     dmz:mail.meu_site.com.br
076 IMAP/ACCEPT     net     dmz:mail.meu_site.com.br
077 # de: todos os hosts da NET -> para: www.meu_site.com.br
078 Web/ACCEPT      net     dmz:www.meu_site.com.br
079 # de: todos os hosts da NET -> para: todos os hosts da DMZ.
080 Ping/ACCEPT     net     dmz
081 AllowICMPs      net     dmz
082
083 ## Regras: NET -> FW ##
084 # de: todos os hosts da NET -> para: FW.
085 DNS/ACCEPT      net     fw
086 Ping/ACCEPT     net     fw
087 AllowICMPs      net     fw
```

Exemplo 6: Arquivo de regras, proxy

```
089 ## Redirecionamento (Proxy Transparente) ##
090 ####
091 # Parâmetros para o Squid 2.5 ou anterior:
092 #   http_port 3128
093 #   httpd_accel_host virtual
094 #   httpd_accel_port 80
095 #   httpd_accel_with_proxy  on
096 #   httpd_accel_uses_host_header on
097 #
098 ###
099 # Parâmetros para o Squid 2.6 ou posterior:
100 #   http_port 3128 transparent
101 #
102 ###
103 #
104 REDIRECT  loc  3128 tcp 80 - !200.201.166.0/24,200.201.173.0/24,
➥200.201.174.0/24
105 # Não apagar esta última linha.
```

Para que o site e os emails sejam acessíveis de fora, é necessário também permitir o tráfego do serviço DNS da zona NET para o firewall (**linhas 83 a 87**).

Acesso do firewall

Não precisamos definir regras para a zona FW, pois essa zona foi definida com a *tag* `ACCEPT` no arquivo de políticas para todas as zonas (*all*)[1]. Isso significa que o firewall poderá acessar quaisquer serviços em quaisquer zonas, sem restrições, o que é bastante lógico.

Simples demais

O arquivo de regras parece demasiadamente simples, e de fato é. Tarefas mais complexas, como o controle do estado das conexões, por exemplo, são realizadas automaticamente pelo Shorewall. Além disso, o uso das macros também simplifica a definição de regras para serviços que utilizam múltiplas portas e protocolos.

As regras do tipo DNAT (prerouting) também desfrutam de grande facilidade, pois seu uso dispensa a criação de regras de encaminhamento (*forward*) entre as zonas envolvidas. No *Iptables*, por exemplo, essas regras devem ser criadas manualmente, mas o Shorewall cuida de tudo automaticamente.

Regras de proxy

Um último serviço que ainda não ativamos é o proxy transparente, a ser alojado no servidor firewall[1]. Para ser ativado, precisamos fazer com que o Shorewall redirecione as conexões vindas da rede LOC na 80 para a porta 3128 do servidor firewall (**exemplo 6**, **linha 104**), pois o servidor proxy (*Squid*) está escutando nessa porta no firewall. Porém, precisamos excluir as três faixas de IP da Caixa Econômica, pois elas não passarão pelo proxy, já que foram tratadas de forma específica em um ponto anterior do arquivo de regras (**exemplo 3**).

Iniciando

A última etapa para a ativação do firewall é iniciá-lo. Para isso, basta executar o comando `shorewall start`. Caso toda a configuração esteja em ordem, a palavra "Compiling" será a primeira a aparecer, indicando que o programa está compilando as regras passadas a ele. Por último, a palavra "done" (ou "Shorewall Restarted", caso se use o compilador shell) informa que o firewall já está ativo.

Estendendo

Com a rede funcionando de acordo com a **figura 1**, talvez surja a necessidade de se executar o serviço *Samba* no servidor firewall, por exemplo. Nesse caso, após configurar o Samba no servidor, será necessário dar permissão às estações da rede local para acessarem esses serviços. Para isso, basta adicionar a seguinte regra às configurações de acesso da zona LOC à zona FW no arquivo `rules`:

```
ACTION    SOURCE   DEST
SMB/ACCEPT  loc      fw
```

Feito isso, basta reiniciar o Shorewall com o comando `shorewall restart` para que os usuários da rede local possam acessar os compartilhamentos no servidor.

Conclusão

O Shorewall é uma ótima ferramenta para a criação e a administração de firewalls, não apenas em máquinas dedicadas como também em estações individuais, com uma única interface.

O site do desenvolvedor[2] contém várias informações e dicas sobre a configuração e utilização do Shorewall, além de mostrar o potencial dessa grande ferramenta. ■

Mais informações

[1] Tarcísio C. Espínola, "Domando o fogo". http://www.linuxmagazine.com.br/article/1716

[2] Shorewall: http://www.shorewall.net/

Sobre o autor

Tarcísio Carvalho Espínola é coordenador de TI do Instituto Centec e vem trabalhando na implementação de soluções livres nos servidores da empresa. Em seu tempo livre, mantém o site Opção Linux (http://www.opcaolinux.com.br).