Serviço (Servidor) Proxy em plataforma Linux

Instalando o Squid

O Squid é composto de um único pacote, por isso a instalação é simples. Instale o pacote "**squid**" usando o apt-get, como em:

 # apt-get install squid

Toda a configuração do Squid é feita em um único arquivo, o "**/etc/squid/squid.conf**". Caso você esteja usando uma versão antiga do Squid, como a incluída no Debian Woody, por exemplo, o arquivo pode ser o "/etc/squid.conf". Apesar da mudança na localização do arquivo de configuração, as opções descritas aqui vão funcionar sem maiores problemas.

O arquivo original, instalado junto com o pacote, é realmente enorme, contém comentários e exemplos para quase todas as opções disponíveis. Ele pode ser uma leitura interessante se você já tem uma boa familiaridade com o Squid e quer aprender mais sobre cada opção, mas, de início, é melhor começar com um arquivo de configuração mais simples, apenas com as opções mais usadas.

Em geral, cada distribuição inclui uma ferramenta diferente para a configuração do proxy. Uma das mais usadas é o **Webmin**, disponível em várias distribuições. A função dessas ferramentas é disponibilizar as opções através de uma interface gráfica e gerar o arquivo de configuração com base nas opções escolhidas. Em alguns casos, essas ferramentas ajudam bastante, mas, como elas mudam de distribuição para distribuição, acaba sendo mais produtivo aprender a trabalhar direto no arquivo de configuração, que, afinal, não é tão complicado assim. Assim como em outros tópicos do livro, vamos aprender a configurar o Squid "no muque", sem depender de utilitários de configuração.

Comece renomeando o arquivo padrão, de forma a conservá-lo para fins de pesquisa:

mv /etc/squid/squid.conf /etc/squid/squid.conf.orig
Em seguida, crie um novo arquivo "**/etc/squid/squid.conf**", contendo apenas as quatro linhas abaixo:
http_port 3128
visible_hostname **gdh**
acl all src 0.0.0.0/0.0.0.0
http_access allow all

Estas linhas são o suficiente para que o Squid "funcione". Como viu, aquele arquivo de configuração gigante tem mais uma função informativa, citando e explicando as centenas de opções disponíveis. Apenas um punhado das opções são realmente necessárias, pois, ao omití-las, o Squid simplesmente utiliza os valores default. É por isso que acaba sendo mais simples começar com um arquivo vazio e ir inserindo apenas as opções que você conhece e deseja alterar.

As quatro linhas dizem o seguinte:

http_port 3128: A porta onde o servidor Squid vai ficar disponível. A porta 3128 é o default, mas muitos administradores preferem utilizar a porta 8080, que soa mais familiar a muitos usuários.
visible_hostname gdh: O nome do servidor, o mesmo que foi definido na configuração da rede. Ao usar os modelos desse capítulo, não se esqueça de substituir o "gdh" pelo nome correto do seu servidor, como informado pelo comando "hostname".

acl all src 0.0.0.0/0.0.0.0 e **http_access allow all**: Estas duas linhas criam uma acl (uma política de acesso) chamada "all" (todos), incluindo todos os endereços IP possíveis. Ela permite que qualquer um dentro desta lista use o proxy, ou seja, permite que qualquer um use o proxy, sem limitações.

Para testar a configuração, reinicie o servidor Squid com o comando:

/etc/init.d/squid restart

Se estiver no CentOS, Fedora ou Mandriva, pode utilizar o comando "service", que economiza alguns toques no teclado:

 # service squid restart

No Slackware, o comando será "/etc/rc.d/rc.squid restart", seguindo a lógica do sistema em colocar os scripts referentes aos serviços na pasta /etc/rc.d/ e inicializá-los automaticamente durante o boot, desde que marcada a permissão de execução.

Para testar o proxy, configure um navegador (no próprio servidor) para usar o proxy, através do endereço 127.0.0.1 (o localhost), porta 3128. Se não houver nenhum firewall pelo caminho, você conseguirá acessar o proxy também através dos outros micros da rede local, basta configurar os navegadores para usarem o proxy, fornecendo o endereço do servidor na rede local.

Caso necessário, abra a porta 3128 na configuração do firewall, para que o Squid possa receber as conexões. Um exemplo de regra manual do Iptables para abrir a porta do Squid apenas para a rede local (a interface eth0 no exemplo) é:

iptables -A INPUT -i eth0 -p tcp --dport 3128 -j ACCEPT

Criando uma configuração básica

O problema com o modelo de configuração minimalista que acabamos de ver é que com apenas estas quatro linhas o proxy ficará muito aberto. Se você deixar o servidor proxy ativo no próprio servidor que compartilha a conexão e não houver nenhum firewall ativo, qualquer um na internet poderia usar o seu proxy, o que naturalmente não é desejado. O proxy deve ficar ativo apenas para a rede local.

Vamos gerar, então, um arquivo mais completo, permitindo que apenas os micros da rede local possam usar o proxy e definindo mais algumas políticas de segurança. Neste segundo exemplo já aproveitei algumas linhas do arquivo original, criando regras que permitem o acesso a apenas algumas portas específicas e não mais a qualquer coisa, como na configuração anterior:

```
http_port 3128
visible_hostname gdh

acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl SSL_ports port 443 563
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
acl Safe_ports port 443 563 # https, snews
acl Safe_ports port 70 # gopher
acl Safe_ports port 210 # wais
```

```
acl Safe_ports port 280 # http-mgmt
acl Safe_ports port 488 # gss-http
acl Safe_ports port 591 # filemaker
acl Safe_ports port 777 # multiling http
acl Safe_ports port 901 # swat
acl Safe_ports port 1025-65535 # portas altas
acl purge method PURGE
acl CONNECT method CONNECT

http_access allow manager localhost
http_access deny manager
http_access allow purge localhost
http_access deny purge
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports

acl redelocal src 192.168.1.0/24
http_access allow localhost
http_access allow redelocal

http_access deny all
```

As acl's "SSL_ports" e a "Safe_ports" são as responsáveis por limitar as portas que podem ser usadas através do proxy. Neste exemplo, usei a configuração-modelo indicada na documentação do Squid, que prevê o uso de diversos protocolos conhecidos e também o uso de portas altas, acima da 1024. Ela é tão extensa porque cada porta é especificada em uma linha diferente. Podemos simplificar isso agrupando as portas na mesma linha, o que resulta em um arquivo de configuração muito menor, mas que faz exatamente a mesma coisa:

```
http_port 3128
visible_hostname gdh

acl all src 0.0.0.0/0.0.0.0
acl manager proto cache_object
acl localhost src 127.0.0.1/255.255.255.255
acl SSL_ports port 443 563
acl Safe_ports port 21 80 443 563 70 210 280 488 59 777 901 1025-65535
acl purge method PURGE
acl CONNECT method CONNECT

http_access allow manager localhost
http_access deny manager
http_access allow purge localhost
http_access deny purge
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports

acl redelocal src 192.168.1.0/24
http_access allow localhost
http_access allow redelocal

http_access deny all
```

Veja que em ambos os exemplos adicionei duas novas acl's. A acl "localhost" contém o endereço 127.0.0.1, que você utiliza ao usar o proxy localmente (ao navegar usando o próprio servidor), e a acl "rede local", que inclui os demais micros da rede local. Substitua o "**192.168.1.0/24**" pela faixa de endereços IP e a máscara de sub-rede usada na sua rede local (o 24 equivale à mascara 255.255.255.0).

Depois de criadas as duas políticas de acesso, vão duas linhas no final do arquivo que especificam que os micros que se enquadrarem nelas poderão usar o proxy:

http_access allow localhost
http_access allow redelocal

Lembra-se da acl "all", que contém todo mundo? Vamos usá-la para especificar que os clientes que não se enquadrarem nas duas regras acima (ou seja, clientes não-autorizados, vindos da Internet) não poderão usar o proxy:

http_access deny all

Esta linha deve ir no final do arquivo, depois das outras duas. A ordem é importante, pois o Squid interpreta as regras na ordem em que são colocadas no arquivo. Se você permite que o micro X acesse o proxy, ele acessa, mesmo que uma regra mais abaixo diga que não.

Se você adicionasse algo como:

acl redelocal src 192.168.1.0/24
http_access allow redelocal
http_access deny redelocal

... os micros da rede local continuariam acessando, pois a regra que permite vem antes da que proíbe.

Existem alguns casos de sites que não funcionam bem quando acessados através de proxies, um exemplo comum é o "Conectividade Social", da Caixa. Normalmente nesses casos o problema está em algum recurso fora do padrão usado pelo sistema do site e não no servidor proxy propriamente dito, mas, de qualquer forma, você pode solucionar o problema de forma muito simples orientando o servidor proxy a repassar as requisições destinadas ao site diretamente.

Para isso, adicionamos uma ACL na configuração, especificando a URL do site e usando a opção "always_direct":

acl site dstdomain siteproblematico.com
always_direct allow site

Esta regra deve vir antes da regra que libera os acessos provenientes da rede local, como em:

acl site dstdomain siteproblematico.com
always_direct allow site

acl redelocal src 192.168.1.0/24
http_access allow localhost
http_access allow redelocal
http_access deny all

Depois de configurar o arquivo, não se esqueça de reiniciar o serviço para que a configuração entre em vigor:

 # /etc/init.d/squid restart

Nesse ponto o seu proxy já está completamente funcional. Você pode começar a configurar os navegadores nos PCs da rede local para utilizá-lo e acompanhar o desempenho da rede.