SERVIDOR PROXY COM SQUID3 em GNU/Linux Debian7

Por: Prof. Roitier Campos Gonçalves

O Proxy é um serviço de rede através do qual é possível estabelecer um alto nível de controle/filtro de tráfego/conteúdo e armazenamento em cache de navegação. Através de serviço, os administradores de rede podem contar com varias possibilidades, aumentando sua capacidade de adminsitração da rede, bem como aumentando a dinamica das regras e politicas implantadas nas organizações.

O Proxy atua como um intermediário entre o usuário e servidores de conteúdo, seja ele local ou remoto. Através dele, o administrador pode decidir, entre outras coisas, como, quando e o quê cada usuário o grupo de usuários pode acessar, bem como estabelecer regras baseadas em hardwares, protocolos, serviços e contéudo.

Simplificando um pouco, o usuário deixa de ter contato direto com a internet e passa a contar com o Servidor Proxy para esse fim.

Segue uma ilustração do que está sendo dito:

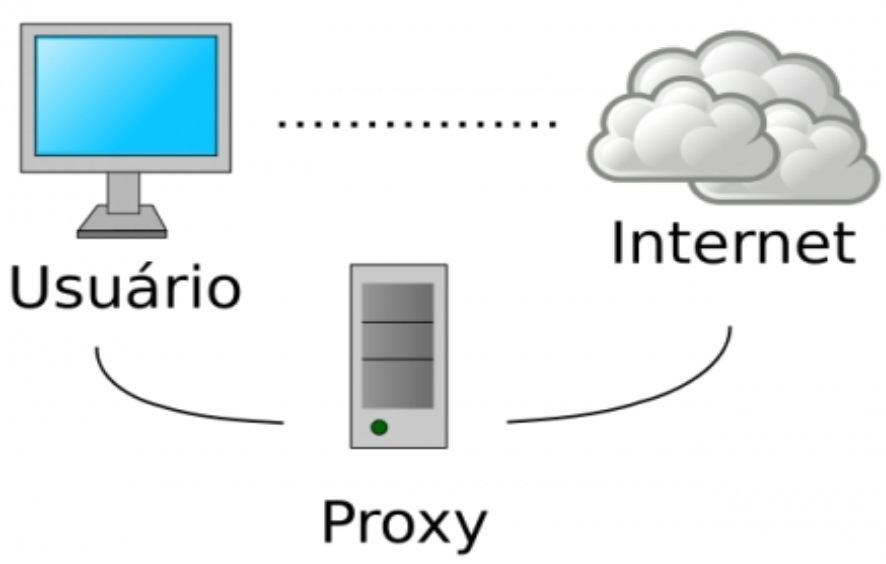

Figura 1 – Visão do Proxy

Neste tutorial, utilizaremos o Software Livre Squid-Cache para mostrar o que se pode fazer com um Servidor Proxy. É importante ressaltar que existem várias opções de softwares para isso, tanto proprietérios como Softwares Livres, e que a opção pelo Squid é estritamente técnica e conceitual, e não envolve ideais mercadológicas.

Instalação do Squid

A instalação do serviço no GNU/Linux Debian 7 (ou superior), como em todo sistema que utiliza o Gerenciador de Pacotes APT pode ser feita através dos comandos abaixo:

 # apt-get update && apt-get install squid3

Após a instalação do programa squid3, vamos acessar o local de instalação para podermos iniciar as devidas configurações renomeando ou ate mesmo excluindo o arquivo original:

 # cd /etc/squid3/

Um procedimento bastante comum é fazermos um backup do arquivo de configuração original, aquele que vem junto do software no momento da instalação, para que o mesmo possa ser consultado em necessidades futuras:

 # mv squid.conf squid.conf.backup

Após a remoção ou renomeação do arquivo squid.conf, como acima, vamos criar um novo arquivo “squid.conf”

 # nano squid.conf

O arquivo de configuração:”squid.conf” é criado respeitando uma estrutura top/down, ou seja, ele é lido de cima para baixo, o que implica dizer que a primeira regra que atenda à requisição feita será utilizada para a decisão.

Seguem algumas possibilidades que o Squid nos dá:

############################## INICIO ##################################

#Porta padrao proxy

http_port 3128

#Define o tamonho maximo de um objeto para seu armazenamento no cache local:

maximum_object_size 4096 KB

#Define o tamanho minimo de um objeto para seu armazenamento no cache local

minimum_object_size 0 KB

#Define o tamanho maximo de um objeto para seu armazenamento no cache de memoria

maximum_object_size_in_memory 64 KB

#Definicao da quantidade de memoria RAM a ser alocada para cache

cache_mem 60 MB

#Para nao bloquear downloads

quick_abort_min -1 KB

 # Resolve um problema com conexõs insistentes que ocorre com certos servidores,

 # e que provoca delays em nosso cache.

detect_broken_pconn on

 # Provoca um ganho de performance ao usar conexões pipeline (requisiçem em #paralelo)

pipeline_prefetch on

#Para cache de fqdn

fqdncache_size 1024

#Tempo de atualizacao dos objetos relacionados aos protocolos ftp, gopher e http.

refresh_pattern ^ftp:// 1440 20% 10080

refresh_pattern ^gopher: 1440 0% 1440

refresh_pattern -i (/cgi-bin/|\?) 0 0% 0

refresh_pattern . 0 20% 4320

#Definicao da porcentagem do uso do cache que fará o squid descartar os arquivos #mais antigos

cache_swap_low 90

cache_swap_high 95

#Arquivos de adminsitração de Logs

access_log /var/log/squid3/access.log squid

cache_log /var/log/squid3/cache.log

cache_store_log /var/log/squid3/store.log

#Define a localizacao do cache no disco, o tamanho e a quantidade de diretorios #pais e diretórios filhos

cache_dir ufs /var/spool/squid3 100 16 256

#Controle do arquivo de Log

logfile_rotate 10

#Arquivo que contem os nomes de maquinas

hosts_file /etc/hosts

#Maquinas que nao precisaram de autenticacao "Colocar IP"

acl liberados src "/etc/squid3/liberados/liberados"

http_access allow liberados

#liberar o acesso ao site da caixa que está problemas

acl bancos dstdomain "/etc/squid3/liberados/bancos"

always_direct allow bancos

cache deny bancos

#MACS que estao liberados.

acl macliberado arp "/etc/squid3/liberados/mac_liberado"

http_access allow macliberado

ACL Padroes

acl manager proto cache_object

acl localhost src 127.0.0.1/32

acl SSL_ports port 443 # https

acl SSL_ports port 444 # https

acl SSL_ports port 447 # https

acl SSL_ports port 563 # https

acl SSL_ports port 873 # https

acl SSL_ports port 7443 # https

acl SSL_ports port 1000 # https

acl Safe_ports port 80 # http

acl Safe_ports port 21 # ftp

acl Safe_ports port 22 # ftp

acl Safe_ports port 20 # 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 1025-65535 # unregistered ports

```
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 631 # cups
acl Safe_ports  port 873 # rsync
acl Safe_ports  port 901 # SWAT
acl Safe_ports  port 1080
acl Safe_ports  port 1863
acl Safe_ports  port 8443 # https
acl Safe_ports  port 5222 # gTalk
acl Safe_ports  port 5223 # gTalk
acl Safe_ports  port 47057 # torrent

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

# Seguranca (Protecao do Cache) ---- Não exite mais no
acl manager proto cache_object

#Limita conexeos HTTP
acl connect_abertas maxconn 8
```

#sites para os quais não serão feitos cache. Geralmente bancos

acl NOCACHE url_regex "/etc/squid3/liberados/direto" \?

no_cache deny NOCACHE

 # WHITE-LIST

acl sites-liberados url_regex -i "/etc/squid3/liberados/sites_liberados"

acl sites-almoco url_regex -i "/etc/squid3/liberados/sites_almoco"

 # BLACK-LIST

acl downloads url_regex -i "/etc/squid3/bloqueados/downloads"

acl sites-proibidos url_regex -i "/etc/squid3/bloqueados/sites_proibidos"

acl sites-bloqueados url_regex -i "/etc/squid3/bloqueados/sites_bloqueados"

acl sites-videos url_regex -i "/etc/squid3/bloqueados/sites_videos"

acl sites-redes-sociais url_regex -i "/etc/squid3/bloqueados/sites_redes_sociais"

acl malware url_regex -i "/etc/squid3/bloqueados/sites_malware"

acl extencoes urlpath_regex -i "/etc/squid3/bloqueados/extencoes"

#Bloquear determinados usuarios autenticados

acl usu_bloqueados proxy_auth "/etc/squid3/bloqueados/usu_bloqueados"

#Controle de acesso por horáaqui, vamos liberar o acesso no horádo almoçqui os #usuávãder acessar alguns sites diferenciados entre as 12:00 até13:00

acl almoco time MTWHFAS 12:00-13:00

#Agora vamos criar uma regra para garantir que os usuários que vão acessar no #almoço #estão autenticados

acl autenticados proxy_auth REQUIRED

##################Permissões de Acesso##################

 # usuarios vips tem permissao total menos a sites proibidos

http_access allow acesso_vip

 # bloqueio de extencoes para todos menos usuario vip

http_access deny extencoes !acesso_vip

 # sites malware proibidos para todos os grupos

http_access deny malware

 # usuarios acesso normal e bloqueado em tudo

http_access allow acesso_normal !sites-proibidos !downloads !sites-bloqueados !sites-videos !sites-redes-sociais

 # usuarios acesso rede social acessa rede social menos os demais

http_access allow acesso_rede_social !sites-proibidos !downloads !sites-bloqueados ! sites-videos

#usuarios sites de video acessa videos menos os demais

http_access allow acesso_videos !sites-proibidos !downloads !sites-bloqueados !sites-redes-sociais

 # sites liberados para todos

http_access allow sites-liberados

#Aqui vamos cruzar as acls para garantir que os usuá que vãessar os sites no almoçtejam #autenticados

http_access allow almoco autenticados sites-almoco

acls de bloqueios

http_access deny downloads

http_access deny sites-bloqueados

http_access deny usu_bloqueados

http_access deny sites-videos

http_access deny sites-redes-sociais

 # sites proibidos para todos os grupos

http_access deny sites-proibidos

#regra de bloqueio geral (bloqueio de toda a empresa, libera o que eu liberar)

#http_access deny all

http_reply_access allow all

icp_access allow all

miss_access	allow all

 # nome visivel do servidor

visible_hostname SERVIDOR_ROITIER

 # diretorio de paginas de erro

error_directory /var/www/acessonegado/

 # erro personalizado por acl "regra"

#deny_info http://caminho malware

#cache_effective_group proxy

cache_effective_user proxy

coredump_dir /var/spool/squid3

Após o acesso ao arquivo squid.conf, e inserir o script temos que criar a pasta onde estará disponível para o squid “PROXY” verificar os bloqueios e as liberações de paginas dos grupos:

1 - Comando criar o diretório

mkdir liberados bloqueados

2 - Comando acessar o diretório criado

cd liberados

Ou

cd bloqueados

3 - Comando para criar os arquivos liberados

touch sites_liberados sites_almoco direto mac_liberado bancos liberados

4 - Comando para criar os arquivos bloqueados

touch downloads sites_proibidos sites_bloqueados sites_videos sites_redes_sociais sites_malware extencoes usu_bloqueados

5 - Vamos criar o diretorio virtual onde estara hospedado as paginas de erro do sistema:

mkdir /var/www/acessonegado

cp –a /usr/share/squid3/errors/pt-br/* /var/www/acessonegado/

Após todas as configuracções no arquivo squid.conf, vamos reiniciar o serviço no servidor mencionado com o seguinte comando:

service squid3 restart

Ou

/etc/init.d/squid3 restart

Firewall

Caso seu ambiente tenha dois servidores FIREWALL “gateways” para toda a rede interna compartilhando os links de internet você terá que adicionar um roteador adicional no proxy ou seja por onde a autenticação dos usuários será liberada para acesso a internet deverar ser criado um script em shell para ser iniciado no boot do sistema com o gateway adicional:

Criar o script com os seguintes comandos:

touch /etc/init.d/roteamento

Alterar as permissões do arquivo criado

chmod 777 /etc/init.d/roteamento

Transformar o arquivo em um arquivo executável

chmod +x vi /etc/init.d/roteamento

Depois digite o seguinte script no arquivo criado:
nano /etc/init.d/roteamento

```
################################## INICIO ###################################
#!/bin/bash
############################ Gateway de Internet  ############################
GATEWAY="IP DO FIREWALL"
########################### inicia as configuracao ############################
iniciar(){

route add default gw $GATEWAY
echo "Roteamento Adicional Ativado!"

}
parar(){
iptables -F -t nat
}
```

```
case "$1" in
"start") iniciar ;;
"stop") parar ;;
"restart") parar; iniciar ;;
*) echo "Use os parÃ¢ ou stop"
esac
################################ FIM DA REGRA ############################
```