MANUAL COMPLETO PARA CONFIGURAÇÃO DO FIREWALL IPTABLES/NETFILTER VIA SHELL-SCRIPT

Paulo Afonso, 16 de julho de 2007

Apresentação

O netfilter/iptables vem se tornando umas das referências no que se refere a segurança da tecnologia da informação, pois o mesmo oferece sua flexibilidade técnica e estrutural por está carregado diretamente no núcleo do sistema operacional tornando o software mais eficiente em sua categoria. É necessário a criação de um script de firewall com as regras do iptables é motivado por que o iptables grava as regras nas tabelas e quando é reiniciado o computador as mesmas são apagadas isto acarreta a necessidade da criação de scripts executáveis que iniciem automaticamente junto ao carregamento do computador, isso é um fator importante por que se o iptables guardassem as configurações na memória nem criaria scripts e eles podem tornar as regras bem mais dinâmicas do que se fossem configuradas diretamente sem este recurso.

Paulo Afonso-BA, 06 de agosto de 2007

Título: Construção de Shell-Script de Firewall completíssimo em IPTABLES/NETFILTER

Descrição: Este script possui diversas características como: mascaramento da rede interna e a rede externa, bloquear acesso de sites, Radio UOL mesmo com proxy transparente, liberar as portas principais, bloqueio de ataques como: ping da morte, SYN-FLOOD, ssh de força bruta, Anti-Spoofings e entre outros. Faltou dizer a importante amarração do ip ao mac.

1. Introdução

Através da implantação, adaptação e melhorias de diversos scripts encontrados na internet, em livros e outros que formulei, crie este script com o mesmo propósito da comunidade do linux em que cada um oferece sua parte para um grande sistema, onde a união de suas partes é maior do que a soma deles "Este um Principio da Administração". Bom o script oferece diversas vantagens além da que já foram ditas na descrição, mas uma grande diferencial que ofereço é o tamanho das possibilidades de gerenciar este script e com a flexibilidade dele, pois contém diversas variáveis que facilitam a re-configuração, pois basta alterar uma linha do script e todo o restante será afetado.

Este script foi testado durante muito tempo e pode ter certeza que contém a união de diversos scripts encontrados na internet e em livros. Outro item importante que falta dizer é que com os testes foi possível identificar e excluir regras que bloqueavam coisas indesejadas e que são regras testadas na pratica. A minha intenção não era construir um artigo que apenas possua um monte de regras com as quais ninguém entende o porquê a existência delas, mas um artigo que mostre diversas formas de configurar um Firewall, justamente para atender as diversas necessidades dos administradores de TI, tanto uma empresa rígida com outra mais leve. Assim tentei construir um artigo sobre script-de-firewall que atendesse essas necessidades com exemplos claros e explicação detalhada de cada item e também que possuísse a flexibilidade da re-configuração.

Pré-requisitos básicos, se você está pensando que para o hardware? Enganou-se já que eu faço meus testes em um Pentium I mmx 200 Mhz a placa mãe tem a data de 1997 o linux possui uma grande versatilidade e funciona praticamente em qualquer hardware, mas os requisitos são para as pessoas mesmo e não para as máquinas segue então os requisitos:

- Conhecimento Prévio do Iptables;
- Conhecimento de Shell-script;
- Conhecimento Básico de Linux;
- Conhecimento sobre comandos em internos do sistema.

Para aqueles já sabem a nível médio ou iniciante em Linux que ainda não compreende o Iptables sugiro a Leitura de um artigo que no seguinte endereço: http://www.vivaolinux.com.br/artigos/verArtigo.php?codigo=6834, de forma prática e bem concisa o autor ensina a estrutura de funcionamento do Iptables, porém não ensina a sua sintaxe e nem configurar o mesmo. Este artigo se limita a ensinar como construir o script que inicializa as configurações do iptables para manipular as tabelas do netfilter, em breve estarei lançado meu próprio artigo que ensinará o funcionamento do firewall em questão e sua sintaxe de forma bem didática.

A implementação deste script foi feita em computador com o Debian-Etch, mas segue um exemplo de implementação no slackware. Das Características completas do que ele implementa:

- Trafego de prioridade de saída e entrada pela tabela mangle;
- Ativa o redirecionamento no arquivo ip_forward;
- Habilita o fluxo interno entre os processos;
- Libera as portas principais do servidor;
- Bloqueia acesso de sites negados a rede interna;
- Bloqueio de ataques conhecidos como: ping da morte, SYN-FLOOD, ssh de força bruta, Back Orifice, TRIN00, bloqueio para o servidor X, Anti-Spoofings;
- Bloqueio de scanners ocultos (Shealt Scan);
- **AMARRA IP AO MAC**;
- **Libera Radio UOL mesmo com o proxy transparente**;
- Direciona as requisições da porta 80 para a 3128 (proxy transparente);
- Ativa o mascaramento da rede interna para a externa;
- Exemplos de regras que fazem diversos arranjos como: Redirecionar conexões do VNC, bloqueio do emule, bloqueio Kazaa, Morpheus and AudioGalaxy.

2. Construindo o Firewall – O Básico

O script está muito bem comentado para o entendimento melhor do leitor e para que compreenda a finalidade da tal regra ou comando

Identifica qual é o interpretado que será usado na execução do script;

```
#!/bin/sh
```

Título do script, eu coloquei ele por que o que está comentado nesta linha que está mediadamente depois da declaração do shell, será usado na descrição do webmim no modulo de iniciar e encerrar;

```
#Configuração do Firewall através do iptables
#Autoria do Script
#"::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
#"| Script de Firewall - IPTABLES"
#"| Criado por: Marcelo Magno"
#"| Contribuindo por: Josemar, Marcelo, Urubatan Neto e todos os"
#"| membros da comunidade viva o linux"
#"| Técnico em Informática"
#"| marcelo.espindola@gmail.com"
#"| Uso: firewall start|stop|restart"
#"::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
```

Nesta parte eu declaro a variável **PATH** por que ela indica as pastas onde estão os diversos softwares existentes no sistema que não serão necessários a indicações completa do caminho deles, a variável **IPTABLES** recebe o caminho completo do executável do firewall, a variável **MACLIST** recebe o caminho completo do arquivo que contem os macs e os ips que estão amarrados pelo firewall, na variável **PROGRAMA** recebe o caminho absoluto do próprio script. Lembrando que geralmente eu particularmente gosto de fazer minhas alterações em uma pasta separada, a fim de obter mais organização e não me perder sem saber onde estão os scripts, então eu faço assim, coloco a maioria deles no diretório "/etc/configuracao_personalizada/" só que aqueles scripts que iniciam no boot, sempre deixo na pasta "/etc/init.d/"

```
#Declaração de variaveis
PATH=/sbin:/bin:/usr/sbin:/usr/bin
IPTABLES="/sbin/iptables"
MACLIST="/etc/configuracao_personalizada/macsliberadosfirewall"
PROGRAMA="/etc/init.d/firewall"
```

PORTSLIB e **PORTSBLO** recebem respectivamente o caminho absoluto das portas liberadas e bloqueadas pelo firewall

```
#portas liberadas e bloqueadas
PORTSLIB="/etc/configuracao_personalizada/portslib"
PORTSBLO="/etc/configuracao_personalizada/portsblo"
```

Identificar a interface de rede onde está a **LAN** e a **WAN** através de variáveis é muito importante para dar certa flexibilidade no script, pois uma pequena modificação na rede basta mudar a variável onde está **LAN** e a **WAN** e todo o restante do script vai mudar, isto serve também para a variável **REDE**.

```
#Interfaces de Rede
LAN=eth1
WAN=eth0
REDE="192.168.253.0/24"
```

Neste variável receberá o caminho completo onde está localizado o arquivo que contém sites que serão negados pelo firewall.

```
SITESNEGADOS=/etc/configuracao_personalizada/sitesnegados
```

```
# Os diversos módulos do iptables são chamados através do modprobe
modprobe ip_tables
modprobe iptable_nat
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ip_nat_ftp
modprobe ipt_LOG
modprobe ipt_REJECT
modprobe ipt_MASQUERADE
modprobe ipt_state
modprobe ipt_multiport
modprobe iptable_mangle
modprobe ipt_tos
modprobe ipt_limit
```

```
modprobe ipt_mark
modprobe ipt_MARK
```

3. Construindo o Firewall – Corpo Principal do Script/Start – parte I

Neste Capítulo contém os comandos para a opção de Start do Script - /etc/init.d/firewall start

```
case "$1" in
start)

#mensagem de inicialização
echo ":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo "| Script de Firewall - IPTABLES"
echo "| Criado por: Marcelo Magno"
echo "| Contribuido por: Josemar, Marcelo, Urubatan Neto e todos os"
echo "| membros da comunidade viva o linux"
echo "| Técnico em Informática"
echo "| marcelo.espindola@gmail.com"
echo "| Uso: firewall start|stop|restart"
echo ":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo
echo "============================================= |"
echo "|:INICIANDO A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS  :|"
echo "|:                              DO IPTABLES                              :|"
echo "============================================= |"
```

Aqui o finalmente o iptables começa a trabalhar, nesta parte ele zera todas as regras do iptables (isso caso exista), para que as regras deste script sejam únicas presente na tabela do netfilter

```
$IPTABLES -F
$IPTABLES -F INPUT
$IPTABLES -F OUTPUT
$IPTABLES -F FORWARD
$IPTABLES -t mangle -F
$IPTABLES -t nat -F
$IPTABLES -X
```

A política padrão do iptables é aceitar qualquer pacote que medidamente não exista nenhuma regra para a bloqueando, e nesta parte o iptables é instruído a mudar esta política padrão (-P), no caso bloqueado as conexões de entrada e redirecionamento, menos a saída, que ficará aceitando o envio de pacotes originados pelo o nosso computador firewall, se não fosse assim ele não ficaria funcional.

```
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT ACCEPT
$IPTABLES -P FORWARD DROP
```

Obs.: Se você colocar para negar (DROP) por padrão o redirecionamento e entrada (recomendado) você deverá por obrigação liberar as portas principais do servidor se não ele ficará incomunicável, mas também não cometa o erro de bloquear algo e depois liberar o mesmo, só estará consumido recursos desnecessários do servidor. A estrutura ideal das regras é liberar e depois bloquear, isto que foi o que fiz neste script!

```
# ativar o redirecionamento no arquivo ip_forward
echo "1" > /proc/sys/net/ipv4/ip_forward
echo "ativado o redirecionamento no arquivo ip_forward"
echo "ON .......................................................................................[ OK ]"

#habilitando o fluxo interno entre os processos
$IPTABLES -I INPUT -i lo -j ACCEPT
$IPTABLES -I OUTPUT -o lo -j ACCEPT
echo "ativado o fluxo interno entre os processos"
echo "ON .......................................................................................[ OK ]"
```

Nesta parte o laço "for" credita na variável i, uma linha de cada vez o conteúdo do arquivo, onde o caminho deste foi atribuído na variável **PORTSLIB**, então o comando **cat $PORTSLIB** terá como a saída o conteúdo do arquivo portslib (portas liberadas) e linha deste será atribuída de cada vez a variável i, assim fica possível configura diversas portas de uma vez só.

```
#liberar as portas principais do servidor
for i in `cat $PORTSLIB`; do
$IPTABLES -A INPUT -p tcp --dport $i -j ACCEPT
$IPTABLES -A FORWARD -p tcp --dport $i -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --sport $i -j ACCEPT
done
$IPTABLES -I INPUT -m state --state ESTABLISHED -j ACCEPT
$IPTABLES -I INPUT -m state --state RELATED -j ACCEPT
$IPTABLES -I OUTPUT -p icmp -o $WAN -j ACCEPT
$IPTABLES -I INPUT -p icmp -j ACCEPT
echo "ativado as portas abertas para estabelecer conexões"
echo "ativado a liberação das portas principais do servidor $HOSTNAME"
echo "ON ..........................................................................[ OK ]"

#ATIVAR O REDIRECIONAMENTO (para ambiente flexível)
#LIBERAR RADIO UOL (para ambiente flexível)
```

Este caso é muito parecido com anterior, mas este bloqueia sites que serão negados pelo firewall com o intuito de proteger a rede e o servidor.

```
#bloquear acesso de sites negados a rede interna
for i in `cat $SITESNEGADOS`; do
$IPTABLES -t filter -A FORWARD -s $REDE -d $i -j DROP
$IPTABLES -t filter -A FORWARD -s $i -d $REDE -j DROP
$IPTABLES -t filter -A INPUT -s $i  -j DROP
$IPTABLES -t filter -A OUTPUT -d $i -j DROP
done
echo "ativado o bloqueio de envio de pacotes com origem aos sites negados"
echo "ON ..........................................................................[ OK ]"


#Bloqueio ping da morte
echo "0" > /proc/sys/net/ipv4/icmp_echo_ignore_all
$IPTABLES -N PING-MORTE
$IPTABLES -A INPUT -p icmp --icmp-type echo-request -j PING-MORTE
$IPTABLES -A PING-MORTE -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A PING-MORTE -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo ping da morte"
echo "ON ..........................................................................[ OK ]"


#bloquear ataque do tipo SYN-FLOOD
echo "0" > /proc/sys/net/ipv4/tcp_syncookies
$IPTABLES -N syn-flood
$IPTABLES -A INPUT -i $WAN -p tcp --syn -j syn-flood
$IPTABLES -A syn-flood -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A syn-flood -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo SYN-FLOOD"
echo "ON ..........................................................................[ OK ]"


#Bloqueio de ataque ssh de força bruta
$IPTABLES -N SSH-BRUT-FORCE
$IPTABLES -A INPUT -i $WAN -p tcp --dport 22 -j SSH-BRUT-FORCE
$IPTABLES -A SSH-BRUT-FORCE -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A SSH-BRUT-FORCE -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo SSH-BRUT-FORCE"
echo "ON ..........................................................................[ OK ]"
```

a mesma coisa que no caso anterior, mas neste as portas serão bloqueadas, a fim de evitar ataques que serão explicados no capitulo posterior onde estão as sintaxes do arquivos sendo explicados. Mesmo com a política padrão de bloquear as portas cuja liberação só seria possível com regras do IPTABLES, eu coloquei este exemplo de como bloquear portas apenas para ilustrar a possibilidade esta possibilidade de bloqueio e para enriquecer este artigo, pois o mesmo poderá ser facilmente mudado a gosto de quem quiser.

```
#Bloqueio de portas
for i in `cat $PORTSBLO`; do
$IPTABLES -A INPUT -p tcp -i $WAN --dport $i -j DROP
$IPTABLES -A INPUT -p udp -i $WAN --dport $i -j DROP
$IPTABLES -A FORWARD -p tcp --dport $i -j DROP
$IPTABLES -A FORWARD -p udp --dport $i -j DROP
$IPTABLES -A OUTPUT -p tcp --dport $i -j DROP
$IPTABLES -A OUTPUT -p udp --dport $i -j DROP
done
```

Este tipo de ataque origina-se da seguinte forma: um ataquante que visa destabilizar o servidor, tentará enviar pacotes cuja origem deste não existe ou são inválidos e assim o servidor enviar respostas a destinos inalcançáveis pela rede. Estas regras bloqueiam esta tentativa de ataque

```
#bloqueio Anti-Spoofings
$IPTABLES -A INPUT -s 10.0.0.0/8 -i $WAN -j DROP
$IPTABLES -A INPUT -s 127.0.0.0/8 -i $WAN -j DROP
$IPTABLES -A INPUT -s 172.16.0.0/12 -i $WAN -j DROP
$IPTABLES -A INPUT -s 192.168.1.0/16 -i $WAN -j DROP
echo "ativado o bloqueio de tentativa de ataque do tipo Anti-spoofings"
echo "ON ..........................................................................[ OK ]"
```

Esta regra Limitará os pacotes através do modulo **limit** do iptables

```
#Bloqueio de scanners ocultos (Shealt Scan)
$IPTABLES -A FORWARD -p tcp --tcp-flags SYN,ACK, FIN,  -m limit --limit 1/s -j ACCEPT
echo "bloqueado scanners ocultos"
echo "ON ..........................................................................[ OK ]"
```

4. Construindo o Firewall – Corpo Principal do Script/Start – parte II

Aqui está uma regra onde muitos procuram como fazer, basicamente ela bloqueia um computador na rede em que o ip não seja o que está liberado para aquele mac, isto é, ele restringe o mac para aquele ip e vice versa (amarra o IP ao mac). A lógica do comando for é semelhante as anteriores, a variável **i** receberá de cada vez, cada linha existente no arquivo maclist, no laço for ele executará os comandos tanta vezes forem a quantidade de linhas existe no arquivo maclist. A sintaxe deste será explicada mais adiante, mas eu posso tecer algumas sobre isto, status;ip_do_computador;mac_do_computador;nome_do_computador é a sintaxe básica do maclist, funciona assim: a sintaxe é separada por ";", na primeira subparte representa o status do computador que poder ser a (liberado) ou b (bloqueado), é possível observar nas linhas seguintes como isto ocorrerá:

```
#amarrar ip ao mac
for i in `cat $MACLIST`; do
#aqui cada linha do maclist é atribuída de cada vez

STATUS=`echo $i | cut -d ';' -f 1`
```

#o comando echo exibe o conteúdo da variável e o pipe "|" repassa a saída para outro comando, o cut por sua vez reparte cada linha em pedaços onde o delimitador (-d) é o ';' no parâmetro –f imprime na tela conteúdo da 1º coluna (status), a saída deste é enviada para **STATUS;**

```
IPSOURCE=`echo $i | cut -d ';' -f 2`
MACSOURCE=`echo $i | cut -d ';' -f 3`
MARK= `echo $IPSOURCE | cut -d '.' -f 4`
```

neste caso o **IPSOURCE** e o **MACSOURCE** recebem as outras colunas da mesma linha, faço uma ressalva para o nome do computador que eu coloquei apenas para a organização do maclist, pois neste do script contará até a 3º coluna.

```
#aqui neste caso o comando if está dentro do laço for
#Se status = a então iptables libera a conexão através destes comandos construídos na tabela filter

if [ $STATUS = "a" ]; then
$IPTABLES -t filter -A FORWARD -d 0/0 -s $IPSOURCE -m mac --mac-source $MACSOURCE -j ACCEPT
$IPTABLES -t filter -A FORWARD -d $IPSOURCE -s 0/0 -j ACCEPT
$IPTABLES -t filter -A INPUT -s $IPSOURCE -d 0/0 -m mac --mac-source $MACSOURCE -j ACCEPT
$IPTABLES -t filter -A OUTPUT -s $IPSOURCE -d 0/0 -j ACCEPT
$IPTABLES -t filter -A OUTPUT -s 0/0 -d $IPSOURCE -j ACCEPT
$IPTABLES -t mangle -A PREROUTING -s $IPSOURCE -j MARK --set-mark $MARK


# Se for = b então bloqueia o MAC, ele só executa este comandos se STATUS não for igual a "a".

elif [ $STATUS = "b" ]; then
$IPTABLES -t filter -A FORWARD -m mac --mac-source $MACSOURCE -j DROP
$IPTABLES -t filter -A INPUT -m mac --mac-source $MACSOURCE -j DROP
$IPTABLES -t filter -A OUTPUT -m mac --mac-source $MACSOURCE -j DROP
$IPTABLES -t filter -I INPUT -s $IPSOURCE -j DROP
$IPTABLES -t filter -I FORWARD -s $IPSOURCE -d 0/0 -j DROP
$IPTABLES -t filter -I FORWARD -s 0/0 -d $IPSOURCE -j DROP
$IPTABLES -t filter -I OUTPUT -d $IPSOURCE -j DROP


# Senão for igual a "a" nem "b" então bloqueia todos os ips que não estão sendo usados

else
$IPTABLES -t filter -I INPUT -s $IPSOURCE -j DROP
$IPTABLES -t filter -I FORWARD -s $IPSOURCE -d 0/0 -j DROP
$IPTABLES -t filter -I FORWARD -s 0/0 -d $IPSOURCE -j DROP
$IPTABLES -t filter -I OUTPUT -d $IPSOURCE -j DROP


fi #fim do elif
done #fim do for
echo "Ativado a amarração do ip ao mac"
echo "ON ..............................................[ OK ]"


Na versão original do script de amarrar ip ao mac não inclui bloquear todos os ips


#proxy transparente
$IPTABLES -t nat -A PREROUTING -i $LAN -p tcp --dport 80 -j REDIRECT --to-port 3128
echo "Proxy Transparente ativado"
echo "ON ..................................................................................[ OK ]"
# ativar o mascaramento
$IPTABLES -t nat -A POSTROUTING  -o $WAN -j MASQUERADE

# Carrega controlador de banda
/etc/init.d/cbq start #Para debian
#/etc/rc.d/init.d/cbq start #Para slackware
echo
echo "============================================= |"
echo "::TERMINADA A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS::|"
echo "::                          DO IPTABLES                                  :: |"
echo "============================================= |"
echo "FIREWALL ATIVADO - SISTEMA PREPARADO"
echo "SCRIPT DE FIREWALL CRIADO POR :-) MARCELO MAGNO :-)"
;;
```

5. Construindo o Firewall – Corpo Principal do Script – FINAL

```
stop)
$IPTABLES -F
$IPTABLES -F INPUT
$IPTABLES -F OUTPUT
$IPTABLES -F FORWARD
$IPTABLES -t mangle -F
$IPTABLES -t nat -F
$IPTABLES -X
$IPTABLES -Z

$IPTABLES -P INPUT ACCEPT
$IPTABLES -P OUTPUT ACCEPT
$IPTABLES -P FORWARD ACCEPT

/etc/init.d/cbq stop #Para debian
#/etc/rc.d/init.d/cbq stop #Para slackware
echo "FIREWALL DESCARREGADO - SISTEMA LIBERADO"
;;

restart)
$PROGRAMA stop
$PROGRAMA start
;;
;;
*)
echo "Use: $N {start|stop|restart}" >&2
echo -e "\033[01;31mATENÇÃO"; tput sgr0
echo "Você não colocou nenhum argumento ou algum conhecido, então Por Padrão será dado em 5 segundos um restart no firewall"
sleep 5
$PROGRAMA restart
exit 1
esac
exit 0
```

6. Exemplos dos arquivos de configuração personalizada

macsliberadosfirewall

status;ip_do_computador;mac_do_plrede;nomepc

A letra (a) indica que o computador que está com esse status está liberado, se for (b) estará bloqueado e se for (c) bloqueará toda tipo de conexão originado,destinado e redirecionado para o ip correspondente. O nome o computador não irá importar para o script de firewall só serve para organizar melhor a nossa lista.

```
c;192.168.253.1;;bloqueio-total
a;192.168.253.2;00:0E:A6:8E:1F:F2;marcelo-home
a;192.168.253.3;00:0F:EA:2A:E0:36;na
a;192.168.253.4;00:20:ED:09:8B:53;tania-home
a;192.168.253.5;00:14:2A:42:9F:C2;elidia-home
a;192.168.253.7;00:17:31:40:04:37;jaira-home
a;192.168.253.8;00:16:EC:D9:29:13;luana-home
c;192.168.253.9;;bloqueio-total
c;192.168.253.11;;bloqueio-total
c;192.168.253.12;;bloqueio-total
c;192.168.253.13;;bloqueio-total
c;192.168.253.14;;bloqueio-total
c;192.168.253.15;;bloqueio-total
```

.

```
.
.
c;192.168.253.254;;bloqueio-total
```

Nos testes que fiz mesmo um computador não cadastrado ele só não conseguia acessar a internet, porém acessava normalmente os outros serviços do servidor como, por exemplo, ssh e ftp. Então para garantir a integridade do servidor e da segurança da rede decidi colocar todos os ips da rede sendo bloqueados por este script garantido com estas regras as seguintes situações:

- Se o usuário possuir permissão de trocar o ip do computador e fizer isso ele não navegará, pois as regras irão apenas liberar pacotes se os ips e macs se correspondem entre si;
- Se o usuário possuir um computador cujo mac não esteja cadastrado é preferível que o dhcp não dê dinamicamente o ip a computadores que estejam nesta situação, porém se ele possuir conhecimentos sobre redes de computadores e configurar estaticamente o computador mesmo assim o mac não estará liberado pelas regras, no entanto se não bloqueasse todos os ips da rede ele poderia acessar os outros serviços podendo assim comprometer a segurança do mesmo, pois o ip estaria liberado;
- Se o status de um computador for "b" o computador estará totalmente bloqueado tanto o mac como o ip e se trocar o ip por um que esteja liberado ele estará travado pelo mac;
- A única possibilidade de ele conseguir navegar pela rede e na internet será emular virtualmente um mac que esteja liberado, mas lembrando que as quantidades de endereços macs são na unidade de vários milhões e ele conseguir achar um que esteja liberado pelo servidor é uma missão quase impossível;
- Para garantir mais ainda a segurança deverá ser feito um bloqueio no nível do squid negando acesso a macs não liberados, autenticação com login e senha e se for uma rede wireless usar criptografia forte como a do wpe e wpa.

Para facilitar a recriação do arquivo citado isso se o exemplo dado for diferente da situação de sua rede, bastar executar adaptando o seguinte script:

```
#!/bin/sh
REDE=192.168.253
for i in `seq 254`; do
echo "c;$REDE.$i;;bloqueio-total" >> /etc/configuracao_personalizada/macsliberadosfirewall
done
```

Para baixar o exemplo deste arquivo baixe neste link: http://marcelo.espindola.googlepages.com/macsliberadosfirewall

portsblo

Neste arquivo estará contido as portas a serem bloqueadas pelo firewall, a sintaxe basicamente é colocar em cada linha a porta que deverá ser bloqueada pelo sistema, mas também é possível colocar intervalos de portas, como exemplo, porta_inicio:porta_fim. Segue um exemplo de configuração de bloqueio de portas e os motivos dos bloqueios.
Obs. se você colocar a política padrão como DROP, isto é, bloqueando todo o redirecionamento de entrada e saída, não é necessário colocar portas bloqueadas, pois todas elas já estarão bloqueadas por default, mas se quiser se certificar é que a rede está segura fica por sua conta levar em consideração bloquear as portas ou não. Mas se você colocar ACCEPT aceitando os pacotes é uma prioridade colocar estas portas como bloqueadas para garantir segurança.

traceroute

33435:33525

bloqueio Back Orifice

31337

bloqueio TRIN00

1524
27665
27444
31335

bloqueio para o servidor X

5999:6003
7100

Lembrando que estes bloqueios só servem principalmente para a proteção do próprio servidor e não da rede em si.

Observação: neste arquivo não são suportados comentários com inicio com o símbolo "#"

portslib

A intenção das portas liberadas é justamente aceitar somente aquelas que realmente forem necessárias tanto para uma rede corporativa ou uma outra mais flexível. Lembrando que as portas liberadas servem para tanto para rede quanto ao servidor. É importante liberar essas portas se você as bloqueou na política padrão, mas se não, então não precisa.

ftp=21
ssh=22
SMTP=25
Dns=53
http=80
POP3=110
https=443
squid=3128
vnc=5900
webmin=10000
apache=8080

Esta foi a Configuração básica do script de firewall

Obs.: Lembrando mais uma vez que os arquivos portslib e portsblo devem conter somente as portas em formato de texto puro para as respectivas finalidades

7. Dicas para um ambiente fechado como empresas com políticas rígidas de acesso

BLOQUEAR MSN

Para bloquear o msn basta acrescentar a porta 1863 no final do arquivo **portslib** e colocar o endereço 64.4.13.0/24 no arquivo onde estão os sites negados (/etc/configuração_personalizada/sitesnegados)

Bloqueando AIM:

Neste caso basta acrescentar no arquivo (/etc/configuração_personalizada/sitesnegados) o seguinte endereço: **login.oscar.aol.com**

Bloqueando ICQ:

Para o ICQ basta acrescentar a porta **5190** nas portas bloqueadas e negar o site **login.icq.com**

Bloqueando Yahoo Messenger:

O Yahoo Messenger trabalha com os endereços **cs.yahoo.com** e **scsa.yahoo.com** então para bloquear o messenger do yahoo basta negar estes endereços

iMesh:

basta acrescentar bloquear o seguinte endereço: **216.35.208.0/24**

BearShare e o ToadNode:

acrescente a porta **6346** nas portas bloqueadas

WinMX:

Bloquear os endereços **209.61.186.0/24** e **64.49.201.0/24**

Napigator:

O endereço: **209.25.178.0/24**

Morpheus:

O Endereço: **206.142.53.0/24**
Porta: **1214**

KaZaA:
O Endereço: **213.248.112.0/24**
Porta: **1214**

Limewire:
Porta: **6346**

Audiogalaxy:
O Endereço: **64.245.58.0/23**

Bittorrent:
Porta: **6881:6889**

Bate papo do gmail
O endereço: **chatenabled.mail.google.com**

Obs.: este endereço é para o bate-papo que ocorre no site do gmail e não para o gloogle talk

Google Talk
O endereço: 72.14.0.0/16, 216.239.0.0/16 e talk.google.com
A porta: 5222

Todos os softwares p2p em CIA

As portas: 1214:1215, 1981, 2037, 3501, 3531, 3587, 3955, 4242, 4661:4672, 4688, 5121, 5662, 6085:6086, 6346:6347, 6699, 6881:6889, 8473

8. Black List – Lista Negra – Sites pornográficos

Segue uma lista de sites pornográficos que poderão ser bloqueados ou não pelo firewall, é bom lembra que qualquer bloqueio ou restrição de qualquer espécie, é melhor que seja configurado no firewall, pois nenhum Proxy terá tanta velocidade na execução das regras de bloqueios como o firewall-iptables.

Para facilitar clique no link para baixar o arquivo correspondente: http://marcelo.espindola.googlepages.com/sitesnegados

Veje o conteúdo dele:

realsexcash.com
mature-videos.net
sexoagora.net
sexojovem.com
fucksakes.net
gorgeousboobies.com
fode-me.com
cidadeamadora.com.br
clubedaninfeta.com
porninlife.net

gatadasemana.com
ninfetasbr.com
sexo24h.net
transa-na.net
pausil9.freehostia.com
supercalcinhas.com
lucasex.com
mrjogos.com
sexslick.com
daily-videos.com
filhadaputa.com
forumdesexo.com
sexodigital.info
bangbrosnetwork.com
apornmovie.com
pornkingmovies.com
fotosdeputas.com
animations.com.br
mediapickle.com
mu-red.no-ip.info:8090
peudroweb.tatyvideos.com
diarioprive.com
voyeurxxxvideos.com
americandaydreamspage.com
filmesadultos.com.br
sexo.vtudo.com.br
adultovip.com
clubedaputaria.org
rebecaw.rg9.net
gonzo-movies.net
longvideos.net
pornografia.com.br
amadoresbrasil.com
tgp.pornaccess.com
pimpfreevids.com
sexmoviess.com
videosdesexo.com.br
filmecaseiro.com.br
putavenganza.com
imperiodevideos.com
sexonabanheira.com
gogetsome.com
loucosporsexo.com.br
fineassmovies.com
supersafadas.com
galinhas.com.br
sexoliberado.com.br
diariocaseiras.com.br
filmesgratis.com.br
clicksex.com.br
achesexo.com.br
flogsexual.com
xnxx.net
quevideos.com
putascaseiras.com
rabao.com
bigassadventure.com
sexocaseiro.net
laizinha.com.br
filezinhas.com
marcelinha.com
viciointerracial.com
sexxyclub.net

videosgratis.tv
topfotos.com
new-anal-movies.com
bucetasnet.com
sexolandia.org
moviesdelight.com
japxxx.com
topcaseiras.com
simplyclips.com
mafiadaputaria.com
sexocaseiro.net
paparazzo.purerotica.com
clubedosexo.com.br
flogsexual.com
flogsexual.blogspot.com
garotasfloripa.net
pornvideofinder.com
sexy.com
sexo.com.br
amateur-usa.com
amadorasweb.com
onlyamateurvids.com
revengetv.com
revengex.com
sapphicerotika.com
mylesbianclips.com
pornislove.com
dirtybot.com
sexobis.com.br
sexocaseiro.net
sexobr.com.br
moviediario.gratishost.com
blogdasamadoras.info
bucetasnet.com
baudevideos.com
baudevideos.com.br
sexmatures.com
filmesgratis.com
bustyz.com
vipermovies.com
putasdefavela.com
putascaseiras.com
biazinha.com
filmesproibidos.com
laizinha.com
mythumbs.com
camilinha.com
sexpornhost.com
orgiasbr.net
sexosuruba.com
diariocaseiras.com
rabuda.net
naughtyamericavip.com
xnxx.com
sexpornhost.com
galinhas.com
ultragayvideos.com
pridemov.com
youngerdudes.com
mansites.com
leatherbang.com
gaygravy.com
top100egregio.com

gaymoviedome.com
sexonomotel.com
filmecompleto.com
sexo-brasil.com
quevideos.com
siririca.com
silveirareisfilmes.com
virtualsexy.com
amazonsexy.com
tufosex.com
sexxyclub.net
kazona.com
nexxx.com
diabinhas.com.br
galinhas.com.br
sextracker.com
livesexbar.com
mexicanmassages.com
pornosphere.com
blackvideoz.com
diariodoprazerblog.blogspot.com
musasdanet.com
sexpornhost.com
desejoscarnais.zip.net
diariodoprazer.com
famozinhas.xpg.com.br
marcelinha.com
onionbooty.com
bootycollection.com
bangbros.com
sexonavan.com
sandrinha.com.br
garotasdavan.com.br
garotasdavan.uol.com.br
SoCaseiras.com
playboy.abril.com.br
playboy.com
revistasexy.com
Jujuzinha.com
RevistaSexy.com.br
brunasurfistinha.com
amazonsexy.com
jujuzinha.com
belasninfetas.com
robertinha.com
galerias.com.br
mulhergostosa.net
conventodassafadas.com.br
img147.imageshack.us
priscillinha.com

9. Para um ambiente mais flexível

Descrição

Como exemplos Lan Houses ou provedores de acesso que precisam liberar outras coisas mais importantes. Uma dica para estes ambientes, é não bloquear tudo com aquela política padrão (-P), pois não é necessário tanto controle, mas se mesmo assim quiser bloquear alguma coisa tem muitos exemplos no capítulo de regras para ambientes rígidos.

Neste exemplo eu incluo regras de redirecionamento que podemos fazer de forma mais simples basta acrescentar a porta, o protocolo e o ip de pacotes de serviços destinados a serem redirecionados a outros computadores. Em linhas gerais, sendo este script implementado a um gateway os computadores que estão fora da rede Interna, isto na internet, que não podem se comunicarem

diretamente aos computadores da rede interna, ao não ser que se faça redirecionamento como, por exemplo: acessar por vnc de forma externa algum pc da rede interna, emule com id baixa ou acessar outro serviço qualquer oferecido pela rede interna, neste formato fica mais fácil se você faz muitos redirecionamentos, assim não precisa várias com a mesma finalidade.

Para acrescentar esta funcionalidade basta acrescentar este linha a seguir no inicio do script junto às outras variáveis; sugiro que seja no final.

```
REDILIST="/etc/configuracao_personalizada/listaderedirecionamento"
```

É recomendado que estas próximas linhas que se seque sejam colocadas antes dos bloqueios de portas e de ataques

```
for i in `cat $REDILIST`; do

REDIPROTO=`echo $i | cut -d ';' -f 1` #recebe o protocolo a ser redirecionado
REDIPORTA=`echo $i | cut -d ';' -f 2` #recebe a porta a ser redirecionado
REDIP=`echo $i | cut -d ';' -f 3` #recebe o ip a ser redirecionado
REDISERVICO=`echo $i | cut -d ';' -f 4` #recebe o nome do serviço
REDIHOST=`echo $i | cut -d ';' -f 5` #recebe o nome do host


$IPTABLES -A FORWARD -p $REDIPROTO --dport $REDIPORTA -j ACCEPT
$IPTABLES -t nat -A PREROUTING -p $REDIPROTO -i $WAN --dport $REDIPORTA -j DNAT --to $REDIP
echo "ativado o serviço $REDISERVICO para o  $REDIHOST"

done # fim do comando laço for
echo "ativado o redirecionamento das portas da WAN para LAN"
echo "ON ..........................................................................[ OK ]"
```

A sintaxe do arquivo é “protocolo;porta;ip;nome_do_servico;nome_do_host” muito semelhante ao do exemplo de amarrar ip ao mac, eu coloquei nome do serviço e nome do host, apenas para fim organizacional do arquivo.

Exemplo do arquivo “/etc/configuracao_personalizada/listaderedirecionamento”

Redireciona as conexões do VNC

```
tcp;5900;192.168.253.2;VNC;debian-home
```

Redirecionar as portas do emule para o host 192.168.253.2

```
tcp;4662;192.168.253.2;EMULE;debian-home
udp;4672;192.168.253.2;EMULE;debian-home
```

Coloque estas em caso se gatway pertence a uma lan house, estou dando prioridade a porta 80 e 3128

```
# ativando o trafego de prioridade de saída e entrada pela tabela mangle
$IPTABLES -t mangle -A OUTPUT -o $WAN -p tcp --dport 80 -j TOS --set-tos 16
$IPTABLES -t mangle -A INPUT -i $LAN -p tcp --sport 80 -j TOS --set-tos 16
$IPTABLES -t mangle -A FORWARD -i $LAN -p tcp --sport 80 -j TOS --set-tos 8

$IPTABLES -t mangle -A INPUT -i $LAN -p tcp --sport 3128 -j TOS --set-tos 16
$IPTABLES -t mangle -A OUTPUT -o $LAN -p tcp --dport 3128 -j TOS --set-tos 16

$IPTABLES -t mangle -A INPUT -i $WAN -p tcp --sport  22 -j TOS --set-tos 8
$IPTABLES -t mangle -A OUTPUT -o $WAN -p tcp --dport 22 -j TOS --set-tos 8
echo "ativado o trafego de prioridades mangle TOS:"
echo "ON ..........................................................................[ OK ]"
```

Estas regras foram encontradas na internet e serve para libera a radio da uol mesmo com o proxy transparente, a fim de não passar por ele. O curioso sobre a radio uol, é que nem mesmo pelo ISA SERVER, que o proxy da Microsoft, não funciona. Lembre-se de colocar

estas regras antes do Proxy transparente. Outra coisa importante referente a isto é que eu não coloquei as portas e endereços sendo liberadas nos arquivos portslib e nem no listaderedirecionamento por que elas são especificas para a rádio uol.

```
# Liberar Radio UOL
$IPTABLES -t nat -I PREROUTING -i $LAN -m tcp -p tcp -d 200.221.0.0/16 --dport 80 -j ACCEPT
$IPTABLES -A FORWARD -s $REDE -p tcp --dport 554 -j ACCEPT
$IPTABLES -A FORWARD -d $REDE -p udp --sport 554 -j ACCEPT
$IPTABLES -t nat -I PREROUTING -d app.radio.musica.uol.com.br -j ACCEPT
$IPTABLES -t nat -I PREROUTING -d radio.musica.uol.com.br -j ACCEPT
$IPTABLES -A INPUT -p tcp --dport 111 -j ACCEPT
$IPTABLES -A INPUT -p udp --dport 111 -j ACCEPT
$IPTABLES -A INPUT -p udp --dport 2049 -j ACCEPT
echo "Liberado a Radio Uol"
echo "ON ..............................................................................[ OK ]"
```

No arquivo (portslib) coloque estas portas

```
MSN=1863
ICQ=5190
```

10. Como fazê-lo funcionar no boot

No caso do debian: coloque este script com o nome firewall na pasta /etc/init.d/ e verifique com o comando ls –l /etc/init.d/firewall se este tem as permissões de execução, ele deverá ter o seguinte formato:

-rwxr-xr-x 1 /etc/init.d/firewall

se ele não estiver com as permissões execute o seguinte comando:

chmod +x /etc/init.d/firewall

Depois basta criar um link simbólico na pasta /etc/rc2.d/ com o seguinte comando
ln –s /etc/init.d/firewall /etc/rc2.d/S99firewall, o link apontará para o arquivo /etc/init.d/firewall que o nosso script, o S99 do arquivo de link significa: O “S” de Start (iniciar) e o 99 é a ordem que ele será executado juntamente com o sistema.

O Debian usa o sistema de inicialização SYSV, segundo a apostila do técnico cidadão da Utan, outras distribuições também usam esse sistema como: RedHat, Suse, Mandrake, Conectiva, Ubuntu, etc ..). Então provavelmente essa dica valerá para a maioria das distribuições.

No caso do slackware: esta distribuição linux utiliza o sistema init BSD, neste caso coloque-o na pasta /etc/rc.d/init.d/ e o link na pasta /etc/rc.2/, valendo as mesmas dicas ditas acima.

11. O script básico completo

Para baixa-lo é clicar neste link: http://marcelo.espindola.googlepages.com/firewall

Veja o conteúdo dele:

```
#!/bin/sh
#Configuração do Firewall através do iptables
#Autoria do Script
#":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
#"| Script de Firewall - IPTABLES"
#"| Criado por: Marcelo Magno"
#"| Contribuindo por: Josemar, Marcelo, Urubatan Neto e todos os"
#"| membros da comunidade viva o linux"
#"| Técnico em Informática"
#"| marcelo.espindola@gmail.com"
#"| Uso: firewall start|stop|restart"
#":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
```

```
#Declaração de variaveis
PATH=/sbin:/bin:/usr/sbin:/usr/bin
IPTABLES="/sbin/iptables"
MACLIST="/etc/configuracao_personalizada/macsliberadosfirewall"
PROGRAMA="/etc/init.d/firewall"

#portas liberadas e bloqueadas
PORTSLIB="/etc/configuracao_personalizada/portslib"
PORTSBLO="/etc/configuracao_personalizada/portsblo"

#Interfaces de Rede
LAN=eth1
WAN=eth0
REDE="192.168.253.0/24"

SITESNEGADOS=/etc/configuracao_personalizada/sitesnegados

# Os diversos módulos do iptables são chamados através do modprobe
modprobe ip_tables
modprobe iptable_nat
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ip_nat_ftp
modprobe ipt_LOG
modprobe ipt_REJECT
modprobe ipt_MASQUERADE
modprobe ipt_state
modprobe ipt_multiport
modprobe iptable_mangle
modprobe ipt_tos
modprobe ipt_limit
modprobe ipt_mark
modprobe ipt_MARK


case "$1" in
start)

#mensagem de inicialização
echo ":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo "| Script de Firewall - IPTABLES"
echo "| Criado por: Marcelo Magno"
echo "| Contribuindo por: Josemar, Marcelo, Urubatan Neto e todos os"
echo "| membros da comunidade viva o linux"
echo "| Técnico em Informática"
echo "| marcelo.espindola@gmail.com"
echo "| Uso: firewall start|stop|restart"
echo ":::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::"
echo
echo "============================================= |"
echo "|:INICIANDO A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS  :|"
echo "|:                              DO IPTABLES                          :|"
echo "============================================= |"

$IPTABLES -F
$IPTABLES -F INPUT
$IPTABLES -F OUTPUT
$IPTABLES -F FORWARD
$IPTABLES -t mangle -F
$IPTABLES -t nat -F
$IPTABLES -X

$IPTABLES -P INPUT DROP
```

```
$IPTABLES -P OUTPUT ACCEPT
$IPTABLES -P FORWARD DROP


# ativar o redirecionamento no arquivo ip_forward
echo "1" > /proc/sys/net/ipv4/ip_forward
echo "ativado o redirecionamento no arquivo ip_forward"
echo "ON ..........................................................................[ OK ]"


#habilitando o fluxo interno entre os processos
$IPTABLES -I INPUT -i lo -j ACCEPT
$IPTABLES -I OUTPUT -o lo -j ACCEPT
echo "ativado o fluxo interno entre os processos"
echo "ON ..........................................................................[ OK ]"

#liberar as portas principais do servidor
for i in `cat $PORTSLIB`; do
$IPTABLES -A INPUT -p tcp --dport $i -j ACCEPT
$IPTABLES -A FORWARD -p tcp --dport $i -j ACCEPT
$IPTABLES -A OUTPUT -p tcp --sport $i -j ACCEPT
done
$IPTABLES -I INPUT -m state --state ESTABLISHED -j ACCEPT
$IPTABLES -I INPUT -m state --state RELATED -j ACCEPT
$IPTABLES -I OUTPUT -p icmp -o $WAN -j ACCEPT
$IPTABLES -I INPUT -p icmp -j ACCEPT
echo "ativado as portas abertas para estabelecer conexões"
echo "ativado a liberação das portas principais do servidor $HOSTNAME"
echo "ON ..........................................................................[ OK ]"

#ATIVAR O REDIRECIONAMENTO (para ambiente flexível)
#LIBERAR RADIO UOL (para ambiente flexível)

#bloquear acesso de sites negados a rede interna
for i in `cat $SITESNEGADOS`; do
$IPTABLES -t filter -A FORWARD -s $REDE -d $i -j DROP
$IPTABLES -t filter -A FORWARD -s $i -d $REDE -j DROP
$IPTABLES -t filter -A INPUT -s $i -j DROP
$IPTABLES -t filter -A OUTPUT -d $i -j DROP
done
echo "ativado o bloqueio de envio de pacotes com origem aos sites negados"
echo "ON ..........................................................................[ OK ]"


#Bloqueio ping da morte
echo "0" > /proc/sys/net/ipv4/icmp_echo_ignore_all
$IPTABLES -N PING-MORTE
$IPTABLES -A INPUT -p icmp --icmp-type echo-request -j PING-MORTE
$IPTABLES -A PING-MORTE -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A PING-MORTE -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo ping da morte"
echo "ON ..........................................................................[ OK ]"


#bloquear ataque do tipo SYN-FLOOD
echo "0" > /proc/sys/net/ipv4/tcp_syncookies
$IPTABLES -N syn-flood
$IPTABLES -A INPUT -i $WAN -p tcp --syn -j syn-flood
$IPTABLES -A syn-flood -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A syn-flood -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo SYN-FLOOD"
echo "ON ..........................................................................[ OK ]"
```

```
#Bloqueio de ataque ssh de força bruta
$IPTABLES -N SSH-BRUT-FORCE
$IPTABLES -A INPUT -i $WAN -p tcp --dport 22 -j SSH-BRUT-FORCE
$IPTABLES -A SSH-BRUT-FORCE -m limit --limit 1/s --limit-burst 4 -j RETURN
$IPTABLES -A SSH-BRUT-FORCE -j DROP
echo "ativado o bloqueio a tentativa de ataque do tipo SSH-BRUT-FORCE"
echo "ON ..................................................................[ OK ]"

#Bloqueio de portas
for i in `cat $PORTSBLO`; do
$IPTABLES -A INPUT -p tcp -i $WAN --dport $i -j DROP
$IPTABLES -A INPUT -p udp -i $WAN --dport $i -j DROP
$IPTABLES -A FORWARD -p tcp --dport $i -j DROP
done

#bloqueio Anti-Spoofings
$IPTABLES -A INPUT -s 10.0.0.0/8 -i $WAN -j DROP
$IPTABLES -A INPUT -s 127.0.0.0/8 -i $WAN -j DROP
$IPTABLES -A INPUT -s 172.16.0.0/12 -i $WAN -j DROP
$IPTABLES -A INPUT -s 192.168.1.0/16 -i $WAN -j DROP
echo "ativado o bloqueio de tentativa de ataque do tipo Anti-spoofings"
echo "ON ..................................................................[ OK ]"


#Bloqueio de scanners ocultos (Shealt Scan)
$IPTABLES -A FORWARD -p tcp --tcp-flags SYN,ACK, FIN,  -m limit --limit 1/s -j ACCEPT
echo "bloqueado scanners ocultos"
echo "ON ..................................................................[ OK ]"

#amarrar ip ao mac
for i in `cat $MACLIST`; do
#aqui cada linha do maclist é atribuída de cada vez

STATUS=`echo $i | cut -d ';' -f 1`
#o comando echo exibe o conteúdo da variável e o pipe "|" repassa a saída para outro comando, o cut por sua vez reparte cada linha
em pedaços onde o delimitador (-d) é o ';' no parâmetro –f imprime na tela conteúdo da 1º coluna (status), a saída deste é enviada para
STATUS;

IPSOURCE=`echo $i | cut -d ';' -f 2`
MACSOURCE=`echo $i | cut -d ';' f 3`
MARK=`echo $IPSOURCE | cut -d '.' -f 4`

# neste caso o IPSOURCE e o MACSOURCE recebem as outras colunas da mesma linha, faço uma ressalva para o nome do
computador que eu coloquei apenas para a organização do maclist, pois neste do script contará até a 3º coluna.

#aqui neste caso o comando if está dentro do laço for
#Se status = a então iptables libera a conexão através destes comandos construídos na tabela filter

if [ $STATUS = "a" ]; then
$IPTABLES -t filter -A FORWARD -d 0/0 -s $IPSOURCE -m mac --mac-source $MACSOURCE -j ACCEPT
$IPTABLES -t filter -A FORWARD -d $IPSOURCE -s 0/0 -j ACCEPT
$IPTABLES -t filter -A INPUT -s $IPSOURCE -d 0/0 -m mac --mac-source $MACSOURCE -j ACCEPT
$IPTABLES -t filter -A OUTPUT -s $IPSOURCE -d 0/0 -j ACCEPT
$IPTABLES -t mangle -A PREROUTING -s $IPSOURCE -j MARK --set-mark $MARK

# Se for = b então bloqueia o MAC, ele só executa este comandos se STATUS não for igual a "a".

else #se não
$IPTABLES -t filter -A FORWARD -m mac --mac-source $MACSOURCE -j DROP
$IPTABLES -t filter -A INPUT -m mac --mac-source $MACSOURCE -j DROP
$IPTABLES -t filter -A OUTPUT -m mac --mac-source $MACSOURCE -j DROP
fi # fim do se
done # fim do comando laço for
```

```
echo "Ativado a amarração do ip ao mac"
echo "ON ..........................................................................[ OK ]"


#proxy transparente
$IPTABLES -t nat -A PREROUTING -i $LAN -p tcp --dport 80 -j REDIRECT --to-port 3128
echo "Proxy Transparente ativado"
echo "ON ..........................................................................[ OK ]"


# ativar o mascaramento
$IPTABLES -t nat -A POSTROUTING  -o $WAN -j MASQUERADE

# Carrega controlador de banda
/etc/init.d/cbq start #Para debian
#/etc/rc.d/init.d/cbq start #Para slackware
echo
echo "============================================= |"
echo "::TERMINADA A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS::|"
echo "::                          DO IPTABLES                          :: |"
echo "============================================= |"
echo "FIREWALL ATIVADO - SISTEMA PREPARADO"
echo "SCRIPT DE FIREWALL CRIADO POR :-) MARCELO MAGNO :-)"
;;

stop)
$IPTABLES -F
$IPTABLES -F INPUT
$IPTABLES -F OUTPUT
$IPTABLES -F FORWARD
$IPTABLES -t mangle -F
$IPTABLES -t nat -F
$IPTABLES -X
$IPTABLES -Z

$IPTABLES -P INPUT ACCEPT
$IPTABLES -P OUTPUT ACCEPT
$IPTABLES -P FORWARD ACCEPT

/etc/init.d/cbq stop #Para debian
#/etc/rc.d/init.d/cbq stop #Para slackware
echo "FIREWALL DESCARREGADO - SISTEMA LIBERADO"
;;

restart)
$PROGRAMA stop
$PROGRAMA start
;;
*)
echo "Use: $N {start|stop|restart}" >&2
echo -e "\033[01;31mATENÇÃO"; tput sgr0
echo "Você não colocou nenhum argumento ou algum conhecido, então Por Padrão será dado em 5 segundos um restart no firewall"
sleep 5
$PROGRAMA restart
exit 1
esac
exit 0
```

Adicionar Imagem1.jpg aqui

```
| Técnico em Informática
| marcelo.espindola@gmail.com
| Uso: firewall start|stop|restart
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

=========================================================|
|:INICIANDO A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS:|
|:                    DO IPTABLES                        :|
=========================================================|
ativado o redirecionamento no arquivo ip_forward
ON ...................................................[ OK ]
ativado o fluxo interno entre os processos
ON ...................................................[ OK ]
ativado as portas abertas para estabelecer conexões
ativado a liberação das portas principais do servidor proxy-server
ON ...................................................[ OK ]
ativado o serviço EMULR para o debian-home
ativado o serviço EMULE para o debian-home
ativado o redirecionamento das portas da WAN para LAN
ON ...................................................[ OK ]
ativado o trafego de prioridades mangle TOS:
ON ...................................................[ OK ]
Liberado a Radio Uol
ON ...................................................[ OK ]
ativado o bloqueio de envio de pacotes com origem aos sites negados
ON ...................................................[ OK ]
ativado o bloqueio a tentativa de ataque do tipo ping da morte
ON ...................................................[ OK ]
ativado o bloqueio a tentativa de ataque do tipo SYN-FLOOD
ON ...................................................[ OK ]
ativado o bloqueio a tentativa de ataque do tipo SSH-BRUT-FORCE
ON ...................................................[ OK ]
ativado o bloqueio de tentativa de ataque do tipo Anti-spoofings
ON ...................................................[ OK ]
bloqueado scanners ocultos
ON ...................................................[ OK ]
Ativado a amarração do ip ao mac
ON ...................................................[ OK ]
Proxy Transparente ativado
ON ...................................................[ OK ]

=========================================================|
::TERMINADO A CONFIGURAÇÃO DO FIREWALL NETFILTER ATRAVÉS::|
::                    DO IPTABLES                        ::|
=========================================================|
FIREWALL ATIVADO - SISTEMA PREPARADO
SCRIPT DE FIREWALL CRIADO POR :-) MARCELO MAGNO :-)
proxy-server:/etc/configuracao personalizada#
```

12. Facilidades para quem usa webmin

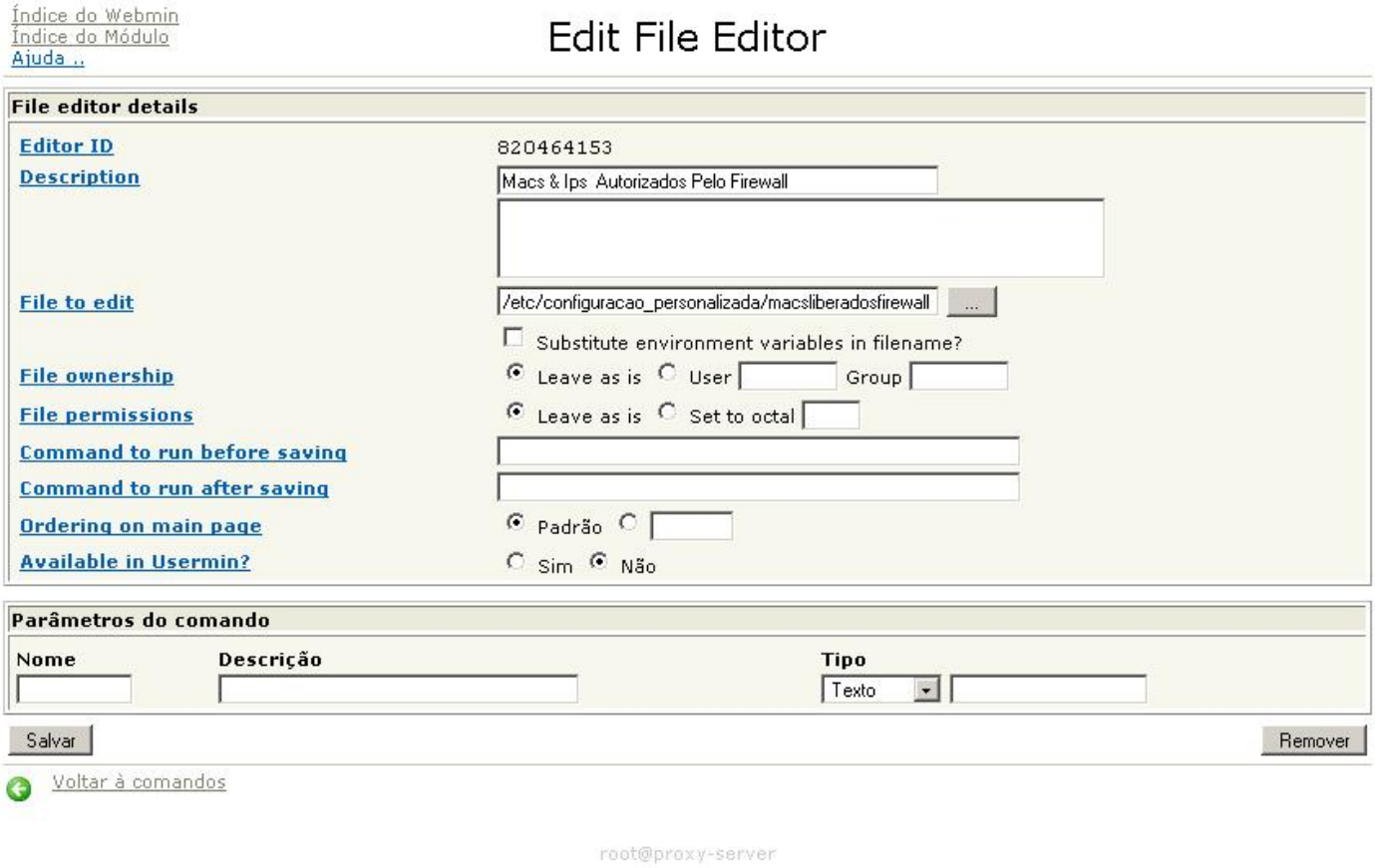
Realmente o webmin é uma ferramenta fantástica no que diz respeito na configuração e manutenção de servidores, então com intuito de oferecer uma demonstração desta ferramenta, que contribui grandemente aos que estão estudando como montar servidores com linux, também uma forma dos técnicos ensinarem aos seus padrões como configurar de forma mais simples o servidor sem a necessidade de aprender o "mapa do tesouro", isto é, sem a obrigação de compreender as linhas de comando do Shell. Assim, este tópico apenas tratará sobre um modulo do webmin que nos ajudará a configurar o servidor. Vamos imaginar a seguinte situação você é especialista em linux, uma empresa de pequeno porte (pode ser uma lan house ou um provedor de acesso) lhe contrata para montar um servidor com as seguintes especificações, servidor com proxy-squid, ativado com proxy transparente, controle de banda para ninguém ultrapassar os limites de consumo evitando a rede ficar lenta, segurança com o firewall iptables/netfilter incluindo o bloqueio a sites indesejados, amarração de ip ao mac e etc. Mas por acaso você decide utilizar este script para implementar como solução de parte dessas exigências, então você monta o sistema, configura-o deixando o mesmo funcionado perfeitamente, mas o gerente da lan house ou provedor decide acrescentar mais um pc na rede e como você é um ótimo profissional muito requisitado, não tem muito tempo disponível para apenas configurar um pc na rede, sendo assim você decide que o responsável pela empresa que lhe contratou pode ter certa autonomia no sentido de pelo menos configurar um pczinho a mais na rede, neste lhe aconselho até mesmo para você mesmo o uso do webmin. Neste contexto o uso deste script fará que um computador que não possui seu mac e ip cadastrado no arquivo maclist tenha acesso totalmente bloqueado pelo firewall, a solução então esta em um modulo do webmin chamado "comandos personalizados" ele está na guia outros do webmin. Neste modulo é possível criar pequenos editores de textos, criar comandos personalizados; chamando scripts com opções ou comandos do sistema, tudo para facilitar a configuração. No caso vou ensinar a criação de 4 editores de texto e um comando personalizado referentes a configuração do firewall presente neste shell-script, sendo para os seguintes arquivos: maclist, portslib, portsblo e sitesnegados; o comando personalizado ficará para o "/etc/init.d/firewall restart"

Vamos criar primeiro para o arquivo maclist siga os passos:

1 clique em adicionar "Create a new file editor";
2 No campo "Description" coloque a descrição do arquivo o ele faz sugiro: Macs & Ips Autorizados Pelo Firewall;

3 No campo “File to edit” coloque o endereço do arquivo, se você seguiu minhas orientações iniciais ele estará no endereço: /etc/configuracao_personalizada/macsliberadosfirewall, o restante deixe como está;
4 Clique em Salvar.

Deixe-o como neste exemplo:
Adicionar Imagem2.jpg aqui



Sigas os mesmos passos acima para:

/etc/configuracao_personalizada/ portslib
/etc/configuracao_personalizada/portsblo
/etc/configuracao_personalizada/sitesnegados

No caso do comando siga estes passos

1º Clique em “Criar novo comando personalizado”;
2º No campo “Descrição” coloque: Reiniciar as configurações do Firewall e no outro campo abaixo dele coloque: Caso altere alguma configuração do firewall clique aqui;
3º No campo “Comando” coloque: /etc/init.d/firewall restart;
4º Na opção “Run in directory” marque como padrão;
5º No campo “Executar como utilizador” coloque root e marque-o;
6º Na opção “Resultados do comando em HTML?” marque como SIM, o restante deixe como está;
7º Clique em Salvar.
Deixe-o como neste exemplo:
Adicionar Imagem3.jpg aqui

Detalhes do comando

Command ID	1183263236
Descrição	Reiniciar as configurações do Firewall Caso altere alguma configuração do firewall clique aqui
Comando	/etc/init.d/firewall restart
Run in directory	Padrão
Executar como utilizador	Webmin user root ... Use user's environment?
Resultados do comando em HTML?	Sim Não
Hide command when executing?	Sim Não
Maximum time to wait for command?	Forever seconds
Ordering on main page	Padrão
Available in Usermin?	Sim Não
Clear environment variables?	Sim Não

Parâmetros do comando

Nome	Descrição	Tipo	Citar parâmetro?
		Texto	Sim Não

Salvar Remover

Voltar à comandos

Pronto! Agora Ele deverá ter um formato semelhante ao da seguinte imagem:
Adicionar Imagem4.jpg aqui

Determinado provedor de acesso usa esta amarração de ips & macs e certo cliente não pagou a mensalidade e como ele sabia a chave de criptografia do roteador wireless poderia acessar a internet sem dificuldades mesmo nesta situação de inadimplência, sendo que este provedor não usa ainda autenticação, e então como eles possuem essa amarração colocaram a letra "b" no status do maclist e reiniciariam o script, no final da história o cliente que não pagou também não acessou. Heheheh :-).

13. CONCLUSÃO E AS CONSIDERAÇÕES FINAIS

É notória a importância da flexibilidade na construção de uma solução empresarial qualquer que seja ela, já que não conhecemos totalmente todos os ambientes onde ele será inserido e também como é importante uma solução que atenda diversas

necessidades diferentes bastando algumas reconfigurações básicas, assim é este script, crie pensando em implementá-lo em diversos sistemas diferentes bastando pequenas mudanças e que fosse logicamente aplicável a escalabilidade.

Bom Fica a Dica para melhorar a interação entre os administradores e gerentes de redes ao sistema operacional linux, acredito que o artigo foi longo, mas é preciso, pois meu público alvo era atingir a quantidade possível de usuários, começando pelos iniciantes que ainda não entendem o que um script de shell e como configurar o iptables e também aos mais experientes passando pelo intermediários até aos avançados, com objetivo de incrementar novas idéias aos suas criações como também aprender mais sobre script e iptables, e logicamente repassei minha experiência e minha contribuição a comunidade do Viva o Linux, se baseando em tantas idéias e soluções encontradas em vários lugares diferentes e unindo em conjunto com a minha capacidade criativa e intelectual, crie este artigo para apreciação de todos membros da comunidade e todos profissionais em geral que querem se aprimorar na ciência da tecnologia da informação.

Basta Agora cada um fazer suas adaptações usando as diversas formas possíveis de combinações, arranjos e permutações destas regras, salientando que não é obrigado o uso de todas elas em um mesmo script o que importa é a verdadeira adequação e a relevância das regras, levando-se em conta as necessidades da rede corporativa da empresa ou qualquer outra rede.

Informativo: Lembrando que qualquer modificação das regras do iptables, só estarão valendo após o restart do script com o comando /etc/init.d/firewall restart, outra questão importante mesmo depois do restart na configuração ainda sim será possível que em algumas regras modificados ainda não fiquem valendo instantaneamente, um exemplo disto será um bloqueio a algum tipo de porta ou protocolo, mas se antes já existia uma conexão ativa ela em geral, não será desfeita e somente as regras valerão para os outros hosts que não estão com conexões ativas referentes ao bloqueio, somente depois que conexão fechar o mesmo host não poderá ativar novamente.

Atenção: Este script pode ser distribuído livremente e declaro que ele está sobre o regime da licença GNU/GPL, então qualquer modificação pode ser feita com exceção aos direitos de créditos dos autores, neste caso se você alterar o script para sua necessidade lhe peço que acrescente seu nome no cabeçalho de autoria no item "contribuindo por".

Obs.: Gostaria aos que implementarem esta solução que comentassem contribuindo com: elogios, criticas construtivas e sugestões de melhoria para um aprimoramento deste shell-script de firewall ou qualquer um que leia este artigo e deseje deixar sua opinião ou mensagem. Não vale digitar besteiras, é bom utilizar o bom senso antes de qualquer coisa. Lembre-se que estamos em forma comunitária e todos desejam a evolução do software livre.

Esta é a versão 1.0 deste manual em breve estarei revisando novamente em busca de novas funcionalidades e também como deixa-lo mais eficiente no uso dos recursos computacionais. Para mais novidades visite o meu blog http://marcelolinux.blogspot.com/

14. REFERÊNCIAS

Bom pessoal! Não posso desmerecer a credibilidade dos autores de scripts em Shell, e digo que diversos destes scripts foram baseados aos já existentes na internet, em livros, e em apostilas, porém eram regras únicas e não tem como mesmo se desviar delas para atingir o mesmo objetivo, porém como cada um faz a sua parte para melhorar o todo, a minha contribuição foi melhorar os scripts e reuni-los e um único lugar e explicar como funciona, uma parte destas regras eu mesmo crie com base no livro sobre o iptables na busca de atender as necessidades do meu cliente, o desenvolvimento deste script é de minha autoria sendo baseado nestas referências:

RIBEIRO, Uirá. Certificação Linux. Rio de Janeiro: Axcel Books, 2004. ISBN 85-7323-232-3.

NETO, Urubatan. Dominando Linux Firewall Iptables: Rio de Janeiro: Editora Ciência Moderna, 2004.ISBN 85-7393-320-8

Apostila Linux, Utan, projeto técnico cidadão apoiado pela Sun Microsystems.

Ao meu amigo Josemar por ter me oferecido o script de amarração de ip ao mac.

15. DOWNLOADS

Nestes links estão os scripts e arquivos para download referentes a este artigo

O script de firewall básico completo em texto puro: http://marcelo.espindola.googlepages.com/firewall (recomendado)
O script de firewall básico completo em .doc: http://marcelo.espindola.googlepages.com/firewall.doc
Exemplo do maclist em texto puro: http://marcelo.espindola.googlepages.com/macsliberadosfirewall
Exemplo do arquivo portsblo em texto puro: http://marcelo.espindola.googlepages.com/portsblo
Exemplo do arquivo portslib em texto puro: http://marcelo.espindola.googlepages.com/portslib
Exemplo do arquivo sitesnegados (black list) em texto puro: http://marcelo.espindola.googlepages.com/sitesnegados
O script do cbq versão 0.7.3: http://marcelo.espindola.googlepages.com/cbq

Versão Para Impressão do artigo

O artigo completo em pdf: http://marcelo.espindola.googlepages.com/firewallcompleto-inclusiveamarraripa.pdf
O artigo completo em .doc: http://marcelo.espindola.googlepages.com/firewallcompleto-inclusiveamarraripa.doc

16. CONTATO & DÚVIDAS

Qualquer dúvida sobre o script deixe um comentário aqui ou entre em contato através do meu e-mail! Ok?

-
"Jesus lâmpada para os meus pés é tua palavra, e luz para o meu caminho."
(Salmo 119:105 ACF)

==========
§Atenciosamente
Marcelo Magno
téc. em informática - Certificado www.colegiosete.com.br
Bacharelando em Administração - FASETE – www.fasete.edu.br
e-mail: marcelo.espindola@gmail.com
marcelo.newexp@gmail.com
Colunista e Membro Ativo da Comunidade VivaoLinux
endereço: http://www.vivaolinux.com.br/perfil/verPerfil.php?login=marceloespindola

Blog Mundo Linux
Dicas & Comentários sobre envolvendo o sistema operacional de código aberto LINUX
Blogger: http://marcelolinux.blogspot.com/

Visite também:

Portal do Técnico em Informática
Artigos, Dicas e Comentários sobre o mundo do Hardware dos computadores
Link:http://portaltecinformatica.blogspot.com/