



INSTALAÇÃO E CONFIGURAÇÃO DO FIREWALL WIPFW



Universidade Federal do Rio de Janeiro
Instituto de Matemática
Departamento de Ciência da Computação
Grupo de Resposta a Incidentes de Segurança

Rio de Janeiro, RJ – Brasil

INSTALAÇÃO E CONFIGURAÇÃO DO FIREWALL WIPFW

GRIS-2007-T-001

Guilherme Alves Cardoso Penha
Rafael de Oliveira Costa

A versão mais recente deste documento pode ser obtida na página oficial do GRIS

GRIS – Grupo de Resposta a Incidentes de Segurança
CCMN Bloco E 2º andar
Salas: E2000 e E2003
Av. Brigadeiro Trompowski, s/nº
Cidade Universitária - Rio de Janeiro/RJ
CEP: 21949-900
Telefone: +55 (21) 2598-3309

Este documento é Copyright© 2007 GRIS. Ele pode ser livremente copiado desde que sejam respeitadas as seguintes condições:

É permitido fazer e distribuir cópias inalteradas deste documento, completo ou em partes, contanto que esta nota de copyright e distribuição seja mantida em todas as cópias, e que a distribuição não tenha fins comerciais. Se este documento for distribuído apenas em partes, instruções de como obtê-lo por completo devem ser incluídas. É vedada a distribuição de versões modificadas deste documento, bem como a comercialização de cópias, sem a permissão expressa do GRIS.

Embora todos os cuidados tenham sido tomados na preparação deste documento, o GRIS não garante a correção absoluta das informações nele contidas, nem se responsabiliza por eventuais consequências que possam advir do seu uso.

Índice

- 1) Introdução
- 2) Instalando e Configurando o WIPFW
- 3) Configuração Avançada
- 4) Exemplos
- 5) Dúvidas e Soluções
- 6) Referências Bibliográficas

1) Introdução

Hoje em dia devemos ter muito cuidado ao conectarmos um computador na internet, principalmente computadores com o sistema operacional Windows, pois este sistema é o mais visado para atividades maliciosas. Por isso além de já termos instalado um antivírus e um anti-spyware nós necessitamos de um firewall.

Um firewall é uma ferramenta que controla o tráfego de uma rede, ou seja, ele é capaz de permitir ou bloquear o acesso a uma rede de computadores. Para o uso doméstico, um firewall irá funcionar simplesmente para regular o tráfego da internet que entra e que sai do seu computador.

Neste tutorial é proposto o uso do WIPFW como um firewall para uso doméstico para usuários que utilizam Windows 2000/2003/XP/Vista.

O WIPFW é um firewall baseado no IPFW que é o padrão do FreeBSD considerado como um sistema robusto e estável e muito utilizado por empresas funcionando como servidor de Internet ou de Proxies.

2) Instalando e configurando o WIPFW

Primeiramente acesse a página oficial do projeto <http://wipfw.sourceforge.net/> e faça o download da última versão do software. Para este tutorial utilizaremos a versão 0.28 que foi lançada no mês de Dezembro de 2006.

Após o download, descompacte o arquivo .zip em um local desejado. Neste exemplo foi utilizado o diretório <c:\Arquivos de Programas\wipfw> para a descompactação do software



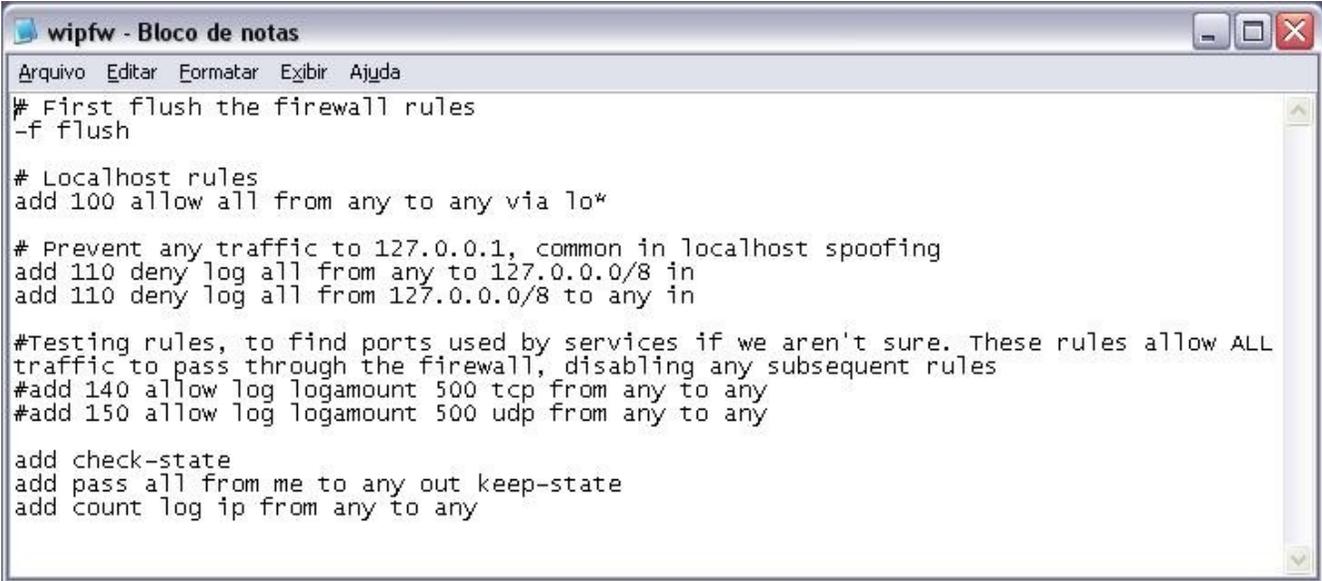
A instalação do WIPFW é bastante simples e é feita através da execução de um único arquivo. Porém esta instalação pode ser feita de duas maneiras: utilizando o `install.cmd` ou o `install-deny.cmd`.

A instalação deve ser feita por usuário com privilégio de administrador. Caso escolha a execução do primeiro, será feita uma instalação permitindo todas as entradas e saídas dos dados entre o seu computador e a internet mas caso escolha o segundo a instalação do wipfw será feita com uma configuração padrão que bloqueará todas as entradas e saídas entre seu computador e a internet.

O uso do arquivo `uninstall.cmd` é condicionado à duas situações: quando o usuário deseja desinstalar o WIPFW e quando o usuário deseja trocar o tipo de instalação e por isso antes de instalar de outra maneira a execução deste arquivo é essencial.

Apesar de podermos utilizar o firewall das duas maneiras possíveis através da instalação, nós sabemos que a grande maioria dos firewalls não possui a política de permitir tudo e nem de bloquear tudo com isso para configurar o seu firewall da maneira que você precisa basta editar o arquivo de regras do WIPFW (wipfw.conf).

A seguir teremos o wipfw.conf comentado para que você possa entender melhor o funcionamento das regras do WIPFW para que mais tarde você possa criar as suas próprias regras.



```
wipfw - Bloco de notas
Arquivo  Editar  Formatar  Exibir  Ajuda
# First flush the firewall rules
-f flush

# Localhost rules
add 100 allow all from any to any via lo*

# Prevent any traffic to 127.0.0.1, common in localhost spoofing
add 110 deny log all from any to 127.0.0.0/8 in
add 110 deny log all from 127.0.0.0/8 to any in

#Testing rules, to find ports used by services if we aren't sure. These rules allow ALL
traffic to pass through the firewall, disabling any subsequent rules
#add 140 allow log logamount 500 tcp from any to any
#add 150 allow log logamount 500 udp from any to any

add check-state
add pass all from me to any out keep-state
add count log ip from any to any
```

Com esta regra, todas as regras anteriores serão apagadas, ou seja, o quadro de regras estará vazio
-f flush

Esta regra permite o tráfego de dados na máquina local, esta regra é recomendada para que processos e aplicativos possam se comunicar internamente.

add 100 allow all from any to any via lo*

Estas duas regras previnem ataques do tipo spoofing

add 110 deny log all from any to 127.0.0.0/8 in

add 110 deny log all from 127.0.0.0/8 to any in

Estas duas regras só se diferenciam pelo tipo de protocolo do pacote que será permitido na sua entrada ou na sua saída. Além disso, esta regra ainda gera log.

#add 140 allow log logamount 500 tcp from any to any

#add 150 allow log logamount 500 udp from any to any

Com esta regra, cada vez que um pacote passa pelo firewall, ele recebe um “tempo de vida” e com isso mesmo que um certo tráfego seja bloqueado no firewall após um dos pacotes já ter chegado este vai demorar um certo tempo ainda ativo

add check-state

Esta regra inclui um parâmetro ao pacote que sai de localhost afim de que seja verificado se é um pacote válido quando este retornar.

add pass all from me to any out keep-state

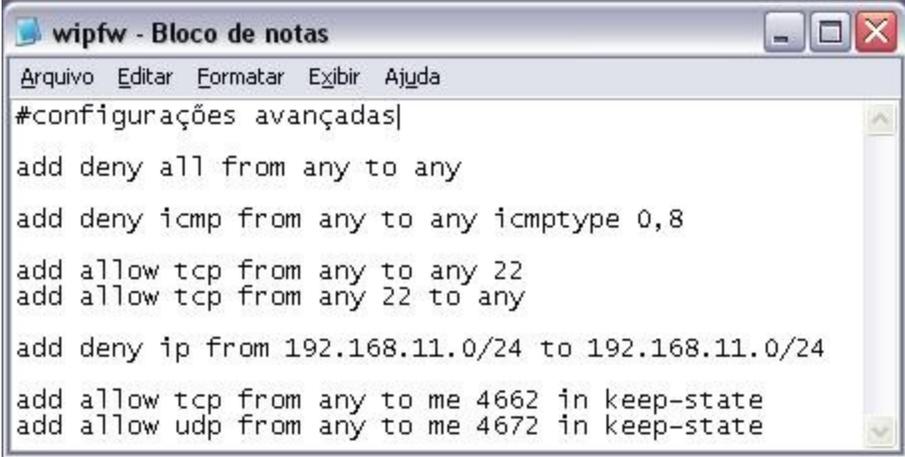
Com esta regra aplica-se log a todos os dados dos pacotes que passaram pelas regras anteriores.
Obs.: O arquivo de log por padrão se encontra em c:\WINDOWS\security\logs\wipfwaaaammdd.log

add count log ip from any to any

3) Configuração avançada

Com as configurações básicas que temos após a instalação já temos um firewall funcionando porém existem muitas outras possibilidades de regras a serem criadas para que o wipfw seja o seu firewall, pois um firewall é algo muito específico para cada usuário, máquina, rede e etc.

Abaixo existem algumas regras que não estão incluídas no wipfw.conf básico e por isso são chamadas de configurações avançadas.



```
wipfw - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
#configurações avançadas|
add deny all from any to any
add deny icmp from any to any icmptype 0,8
add allow tcp from any to any 22
add allow tcp from any 22 to any
add deny ip from 192.168.11.0/24 to 192.168.11.0/24
add allow tcp from any to me 4662 in keep-state
add allow udp from any to me 4672 in keep-state
```

Esta regra bloqueia todos os pacotes mesmo independentes da origem ou do destino. O recomendado é inserir esta regra ao final do seu arquivo de regras, logo após a regra que faz com que o firewall crie um log de todo o tráfego que passa pela interface de rede pois se um determinado pacote não for filtrado pelo seu firewall, ele será automaticamente bloqueado

add deny all from any to any

Com esta regra bloqueamos tanto a entrada quanto a saída de pacotes que possuem o protocolo icmp, este protocolo é o protocolo utilizado pelo ping e por outras ferramentas que podem ser utilizados maliciosamente para fazer varreduras em redes.

add deny icmp from any to any icmptype 0,8

As duas regras abaixo permitem que haja conexão ssh de qualquer origem e destino desde que esta conexão seja feita através da porta 22

add allow tcp from any to any 22

add allow tcp from any 22 to any

Caso seja necessário em sua rede bloquear conexão entre as máquinas de uma mesma rede utilizamos a regra abaixo que nega as conexões de uma máquina da rede 192.168.11.0/24 que tenta se conectar à uma máquina da rede 192.168.11.0/24, ou seja, da sua própria rede.

add deny ip from 192.168.11.0/24 to 192.168.11.0/24

Caso seja necessário liberar um determinado serviço que utiliza a rede devemos fazer uma liberação em nosso firewall. As regras abaixo permitem que a conexão do serviço eMule à rede Kad seja feita, para isso permitimos a entrada dos pacotes com protocolo tcp pela porta 4662 e com protocolo udp pela porta 4672.

Uma observação importante que deve ser feita é que ao final dessas regras, o modo keep-state é ativado para que o wipfw possa de tempos em tempos verificar se as portas que foram abertas estão sendo utilizadas. Caso o eMule seja fechado, esta regra faz com que a porta seja fechada.

add allow tcp from any to me 4662 in keep-state

add allow udp from any to me 4672 in keep-state

4) Exemplos

Estes exemplos foram desenvolvidos para uma maior facilidade de implementação do firewall para certos tipos específicos de aplicações. Estes são atualizados e disponibilizados pela equipe do GRIS e para se manter sempre atualizado com os exemplos mais novos, basta acessar o nosso website.

Para a utilização dos exemplos a seguir, basta escolher o que melhor se adequa ao seu perfil e copiar o conteúdo para o arquivo wipfw.conf.

Exemplo 1:

#--Autor: Guilherme Alves – última atualização: 26/07/07

#--Recomendado para quem só deseja navegar na internet.

#--Modo: Fecha todas as portas e libera apenas as portas para navegação (http), navegação segura (https) e atualização da hora pelo servidor (time - ntp). É permitido o PING e o acesso ao DNS. Todo o tráfego não aplicado nas regras está sendo registrado no log.

```
#--Autor: Guilherme Alves – última atualização: 26/07/07
```

```
#--Recomendado para quem só deseja navegar na internet.
```

```
-f flush
```

```
add pass all from any to any via lo0
```

```
add pass icmp from any to me icmptype 0,8
```

```
add pass all from me to any via eth0
```

```
add deny all from any to 127.0.0.0/8 in
```

```
add deny all from 127.0.0.0/8 to any in
```

```
add check-state
```

```
add pass ip from me to any out keep-state
```

```
add allow udp from any 53 to me in keep-state
```

```
add allow tcp from any 80 to me in keep-state
```

```
add allow tcp from any 443 to me in keep-state
```

```
add allow udp from any 123 to me in keep-state
```

```
add allow tcp from any 123 to me in keep-state
```

```
add count log ip from any to any
```

```
add deny all from any to any
```

Exemplo 2:

#--Autor: Guilherme Alves – última atualização: 26/07/07

#--Recomendado para quem não possui rede local e deseja navegar na internet, usar MSN e Gtalk

#--Modo: Fecha todas as portas e libera as portas para navegação (http), navegação segura (https) e atualização da hora pelo servidor (time - ntp). É permitido o PING e o acesso ao DNS. Todo o tráfego não aplicado nas regras está sendo registrado no log. Neste exemplo é permitido o uso do MSN (apenas voz e texto), e GTalk (protocolo Jabber).

```
#--Autor: Guilherme Alves – última atualização: 26/07/07
```

```
#--Recomendado para quem não possui rede local e deseja navegar na internet, usar MSN e Gtalk
```

```
-f flush
```

```
add pass all from any to any via lo0
```

```
add allow icmp from any to me icmp type 0,8
```

```
add pass all from me to any via eth0
```

```
add deny all from any to 127.0.0.0/8 in
```

```
add deny all from 127.0.0.0/8 to any in
```

```
add check-state
```

```
add pass ip from me to any out keep-state
```

```
add allow udp from any 53 to me in keep-state
```

```
add allow tcp from any 80 to me in keep-state
```

```
add allow tcp from any 443 to me in keep-state
```

```
add allow udp from any 123 to me in keep-state
```

```
add allow tcp from any 123 to me in keep-state
```

```
add allow tcp from any 5222 to me in keep-state
```

```
add allow tcp from any 1863 to me in keep-state
```

```
add allow tcp from any 7001 to me in keep-state
```

```
add count log ip from any to any
```

```
add deny all from any to any
```

Exemplo 3:

#--Autor: Guilherme Alves – última atualização: 26/08/07

#--Recomendado para quem possui uma rede doméstica e deseja permitir o tráfego interno. Assim como navegar na internet, usar Gtalk e o MSN (apenas texto e voz).

#--Modo: Fecha todas as portas e libera as portas para navegação (http), navegação segura (https) e atualização da hora pelo servidor (time - ntp). É permitido o PING e o acesso ao DNS. Todo o tráfego não aplicado nas regras está sendo registrado no log. Neste exemplo é permitido o compartilhamento e arquivos e impressoras na rede local.

```
#--Autor: Guilherme Alves – última atualização: 26/08/07
```

```
#--Recomendado para quem possui uma rede doméstica e deseja permitir o tráfego interno. Assim
```

```
#--como navegar na internet, usar Gtalk e o MSN (apenas texto e voz).
```

```
-f flush
```

```
add pass all from any to any via lo0
```

```
add allow icmp from any to me icmptype 0,8
```

```
add pass all from me to any via eth0
```

```
add deny all from any to 127.0.0.0/8 in
```

```
add deny all from 127.0.0.0/8 to any in
```

```
add check-state
```

```
add pass ip from me to any out keep-state
```

```
add allow udp from any 53 to me in keep-state
```

```
add allow tcp from any 80 to me in keep-state
```

```
add allow tcp from any 443 to me in keep-state
```

```
add allow udp from any 123 to me in keep-state
```

```
add allow tcp from any 123 to me in keep-state
```

```
add allow tcp from any 5222 to me in keep-state
```

```
add allow tcp from any 1863 to me in keep-state
```

```
add allow tcp from any 7001 to me in keep-state
```

```
add allow tcp from any to me 2869 in keep-state
```

```
add allow udp from any to 239.255.255.250 1900
```

```
add allow udp from any 137 to any
```

```
add allow tcp from any 137 to any
```

```
add allow udp from any 138 to any
```

```
add allow tcp from any 138 to any
```

```
add allow udp from any 139 to any
```

```
add allow tcp from any 139 to any
```

```
add allow tcp from any 445 to any
```

```
add count log ip from any to any
```

```
add deny all from any to any
```

5) Dúvidas e soluções

I) Qual a diferença entre WIPFW e o IPFW ?

O WIPFW, por enquanto, não troca os índices dos pacotes. Sendo assim, não é possível redirecionar pacotes. O WIPFW também não possui um controle de tráfego, porém no futuro, o WIPFW usará o driver ndis, que permitirá todas essas funcionalidades.

II) Em quais sistemas operacionais o WIPFW pode ser usado ?

O WIPFW pode ser usado em todas as plataformas Windows, começando pelo Windows 2000.

III) Eu posso utilizar o WIPFW em um projeto proprietário ou comercial ?

O WIPFW é baseado na licença BSD, logo você pode usar os binários sem limitações, porém se usado em códigos, é recomendado que você consulte a licença.

IV) O WIPFW é compatível com provedores de internet ADSL ?

O WIPFW é totalmente compatível com o ADSL, porém ele deve ser instalado com o a conexão ADSL já ativada.

V) Após instalar o WIPFW, a Internet não funciona. Como proceder ?

Neste caso, o principal problema pode ser a necessidade de se instalar o WIPFW com a conexão ativa. Se o WIPFW foi instalado com a conexão desativada, deve-se desinstalar o Firewall executando o arquivo uninstall.cmd e logo após instalando novamente como ensinado anteriormente.

VI) Quais as diferenças entre o Firewall Padrão do Windows e o WIPFW?

Primeiramente o Firewall do Windows trabalha com a liberação de conexões por aplicações (programas), enquanto o WIPFW trabalha por pacotes, onde deve-se especificar a porta e o protocolo usado na conexão desejada, seja para bloquear ou para permitir o tráfego. Uma vantagem do WIPFW é a facilidade de configurar bloqueios e também a facilidade de gerar logs para futuras análises das conexões. Uma vantagem para alguns usuários é que o WIPFW não possui caixas de diálogos, trabalhando assim de forma silenciosa no sistema.

VII) Devo desabilitar o Firewall do Windows para usar o WIPFW?

Para uma melhor performance do sistema, é aconselhável desabilitar o Firewall Padrão do Windows. Porém Os dois podem trabalhar juntos, mas o Firewall do Windows trabalha com filtragem por aplicação, enquanto o WIPFW por pacotes. Sendo assim, para um programa se conectar a internet, ele deve ser liberado no Windows e a sua respectiva porta no WIPFW.

6) Referências Bibliográficas

Página Oficial do projeto WIPFW

<http://wipfw.sourceforge.net/>

Artigo da Wikipédia em português sobre firewall

<http://pt.wikipedia.org/wiki/Firewall>

Manual completo do firewall IPFW

<http://www.hmug.org/man/8/ipfw.php>

Página de ajuda e exemplos do IPFW

http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/firewalls-ipfw.html