

Protótipo de aplicação web para gerenciamento de Firewall Linux

Acadêmico: Régis Maciel Borscheid

Orientador: Francisco Adell Péricas

Roteiro da apresentação

- Introdução
- Segurança de redes de computadores
- Firewall
- Iptables
- Contexto atual
- Desenvolvimento (requisito, especificação, implementação, validação)
- Conclusão e extensões

Introdução

- Proteção dos recursos da rede
- Manutenção constante das configurações de segurança das redes

Objetivo do trabalho

O objetivo do trabalho é criar um protótipo web para administração de Firewall Linux, facilitando assim a manipulação das regras de Firewall baseadas no Iptables.

Segurança de redes

- Objetivo de proteger as informações
- Garantir a consistência e confiabilidade
- Confidencialidade e isolamento
- Disponibilidade

Política segurança

- Conjunto formal de regras
- Definir o que vai ser protegido
- Eleger conjunto de pessoas responsáveis pela segurança
- Procedimentos pós violação
- Plano de continuidade de negócio

Firewall

- Sistema que impõe uma política de controle de acesso
- Principais tipos de Firewalls:
 - Filtros de pacotes
 - Filtros de pacotes com base no estado da conexão
 - Filtro de pacotes na camada de aplicação

Iptables

- Compões a quarta geração de sistemas de Firewall do Linux
- Incorporada a partir do kernel 2.4
- Ferramenta Front-End para o Netfilter
- Netfilter módulo agregado ao kernel
- Grande flexibilidade na configuração

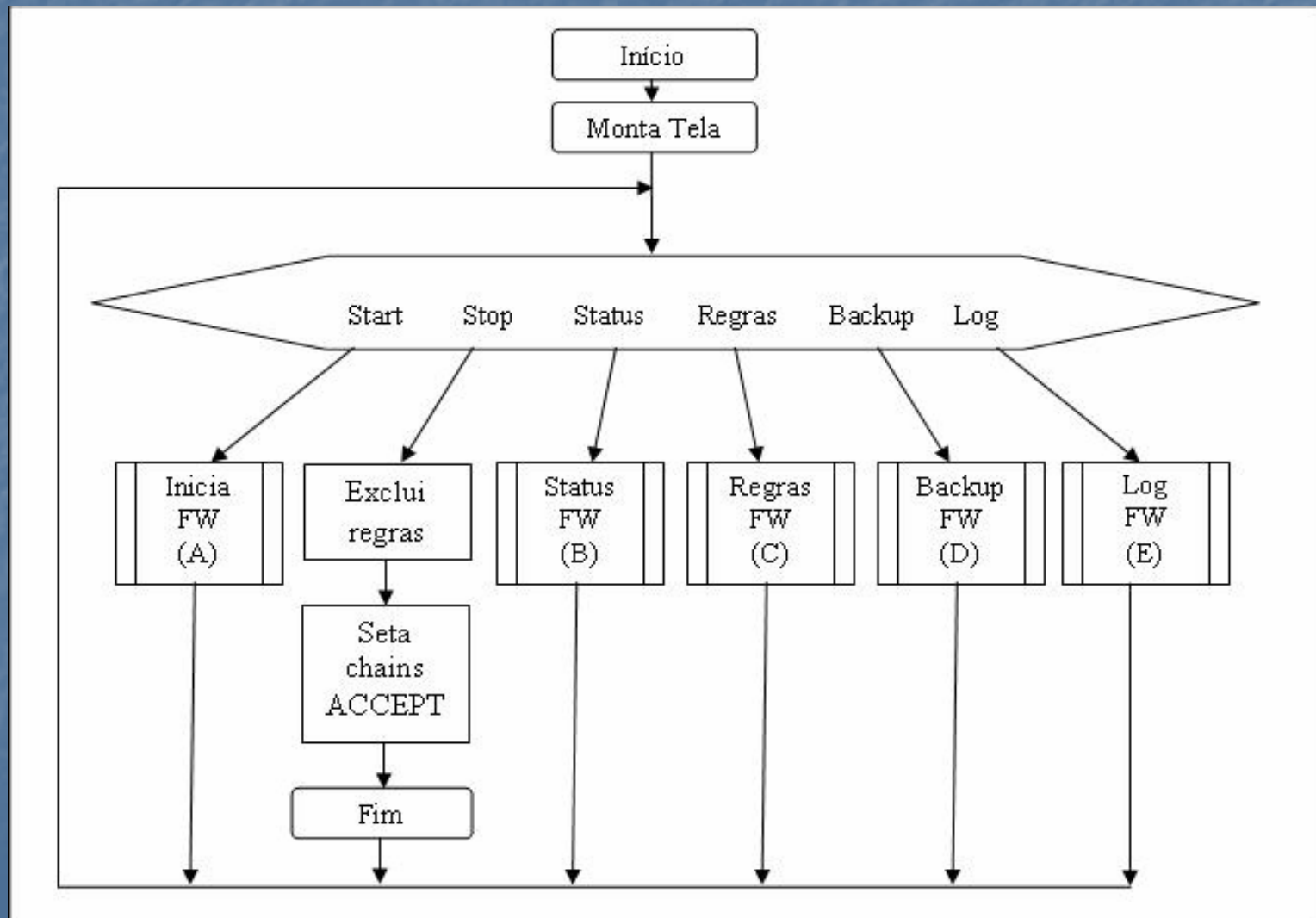
Contexto atual do tema

- Trabalhos correlatos:
 - Guarddog (hosts)
 - StarLink Xfwall (servidores)

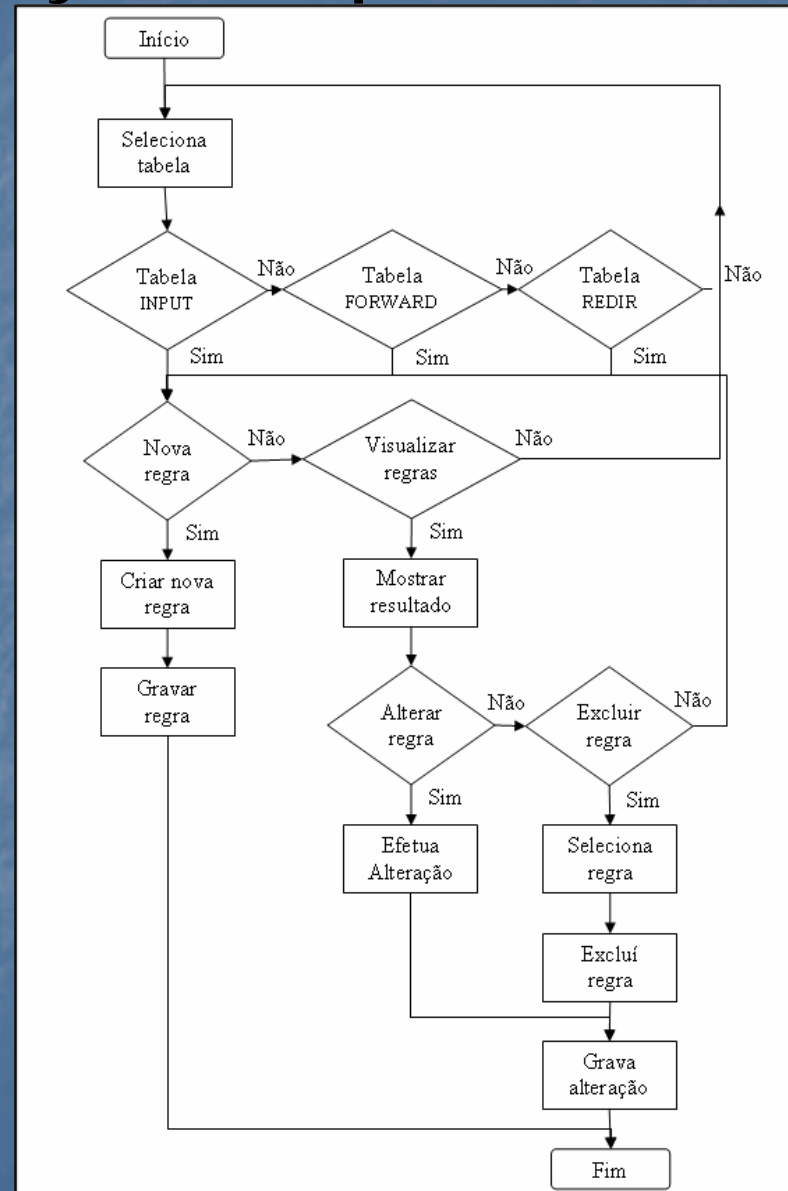
Requisitos principais

- Disponibilizar mecanismo para autenticação de usuários;
- Interface amigável, com exemplos de regras de Firewall;
- Permitir criar, alterar e excluir regras de Firewall
- Análise de logs
- Backup e restauração de regras

Apresentação da especificação



Especificação do processo "Regras" C



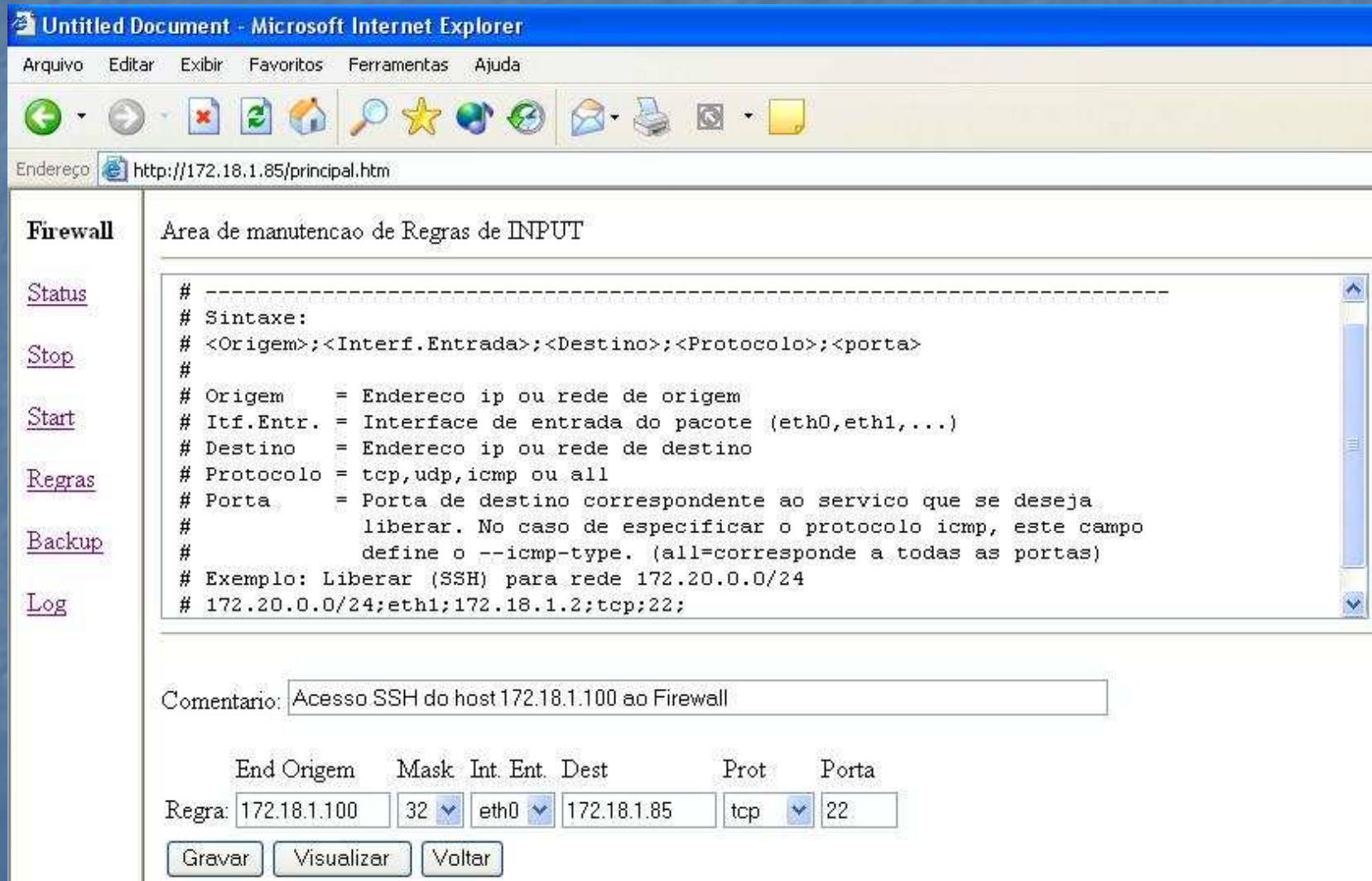
Implementação

- Programação estruturada, utilizando Perl-CGI, Shell Script
- Configuração servidor linux e web
- Autenticação dos usuários MD5 dos servidor Apache

Operacionalidade da implementação

- Aplicação deverá ser instalado em diretório onde somente o super-usuário tem permissão de acesso
- No servidor apache deverá ser feita a configuração do diretório dos Perl-CGI
- Configurado o modo de autenticação no servidor apache

Interface web manutenção regras



The screenshot shows a Microsoft Internet Explorer browser window with the address bar displaying `http://172.18.1.85/principal.htm`. The page content is titled "Area de manutencao de Regras de INPUT". On the left side, there is a navigation menu with links for "Status", "Stop", "Start", "Regras", "Backup", and "Log". The main content area contains a text editor with a firewall rule syntax. Below the editor is a "Comentario:" field with the text "Acesso SSH do host 172.18.1.100 ao Firewall". At the bottom, there is a form for configuring a rule with fields for "End Origem", "Mask", "Int. Ent.", "Dest", "Prot", and "Porta", and three buttons: "Gravar", "Visualizar", and "Voltar".

Firewall

[Status](#)

[Stop](#)

[Start](#)

[Regras](#)

[Backup](#)

[Log](#)

Area de manutencao de Regras de INPUT

```
# -----  
# Sintaxe:  
# <Origem>;<Interf.Entrada>;<Destino>;<Protocolo>;<porta>  
#  
# Origem      = Endereco ip ou rede de origem  
# Itf.Entr.  = Interface de entrada do pacote (eth0,eth1,...)  
# Destino    = Endereco ip ou rede de destino  
# Protocolo  = tcp,udp,icmp ou all  
# Porta      = Porta de destino correspondente ao servico que se deseja  
#             liberar. No caso de especificar o protocolo icmp, este campo  
#             define o --icmp-type. (all=corresponde a todas as portas)  
# Exemplo: Liberar (SSH) para rede 172.20.0.0/24  
# 172.20.0.0/24;eth1;172.18.1.2;tcp;22;
```

Comentario: Acesso SSH do host 172.18.1.100 ao Firewall

End Origem	Mask	Int. Ent.	Dest	Prot	Porta
172.18.1.100	32	eth0	172.18.1.85	tcp	22

Gravar Visualizar Voltar

Caso de uso criação de regra

- Parâmetros passados pelo administrador

#liberar ssh para rede 172.18.0.0

172.18.0.0/16 ; eth0 ; 172.18.1.100 ; tcp ; 22

- Regra Gerada

```
/sbin/iptables -A INPUT -s 172.18.0.0/16 -i eth0 -d  
172.18.1.100 -p tcp --dport 22 -j ACCEPT
```

```
/sbin/iptables -A OUTPUT -d 172.18.0.0/16 -o eth0 -s  
172.18.1.100 -p tcp --sport 22 -j ACCEPT
```


Trecho de código caso uso apresentado

```
cmd_input="$bin -A INPUT -s $orig"
cmd_output="$bin -A OUTPUT -d $orig"

if [ "$itin" != "any" ]; then
    cmd_input=$cmd_input" -i $itin -d $dest"
    cmd_output=$cmd_output" -o $itin -s $dest"
else
    cmd_input=$cmd_input" -d $dest"
    cmd_output=$cmd_output" -s $dest"
fi

if [ "$prot" != "all" ]; then
    if [ "$prot" = "icmp" ]; then
        cmd_input=$cmd_input" -p icmp"
        cmd_output=$cmd_output" -p icmp"
    elif [ "$port" = "all" ]; then
        cmd_input=$cmd_input" -p $prot"
        cmd_output=$cmd_output" -p $prot"
    else
        cmd_input=$cmd_input" -p $prot --dport $port"
        cmd_output=$cmd_output" -p $prot --sport $port"
    fi
fi
```

Conclusão

- Com o planejamento das etapas do trabalho, definição do cronograma e ferramentas que seriam utilizadas, a conclusão do trabalho foi consequência do cumprimento destas atividades
- Monitoração e configuração de regras Firewall Iptables tornou-se simples e sem erros
- Pontos decisivos: especificação, método autenticação, estrutura dos script de Firewall

Extensões

- Desenvolvimento de um módulo de bloqueio de ameaças e ataques
- Desenvolvimento de módulo para integração com uma ferramenta de IDS