Minicurso Tópicos em Segurança da Informação
Parte 2 – Tratamento de Incidentes

I Workshop de Tecnologia de Redes do POP-BA
Ponto de Presença da RNP na Bahia

Italo Valcy <italo@pop-ba.rnp.br>

20 e 21 de setembro de 2010

Licença de uso e atribuição

Todo o material aqui disponível pode, posteriormente, ser utilizado sobre os termos da:

Creative Commons License:
Atribuição - Uso não comercial - Permanência da Licença

http://creativecommons.org/licenses/by-nc-sa/3.0/

Agenda

- Tratamento de Incidentes
  - Estatísticas do CERT.Bahia
  - Construindo Firewalls com IPTables
  - Detecção de máquinas usando NAT
  - Sistema de *logging* remoto

Estatísticas do CERT.Bahia

Sobre o CERT.Bahia

- CSIRT (Centro de Estudos, Respostas e Tratamento de Incidentes de Segurança)

- Grupo responsável pelo tratamento e resposta dos incidentes relacionados com a comunidade baiana

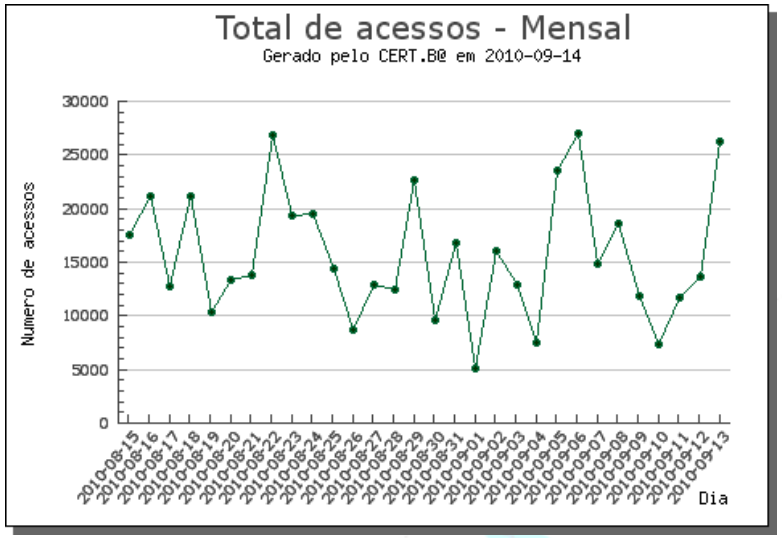
Serviços do CERT.Bahia

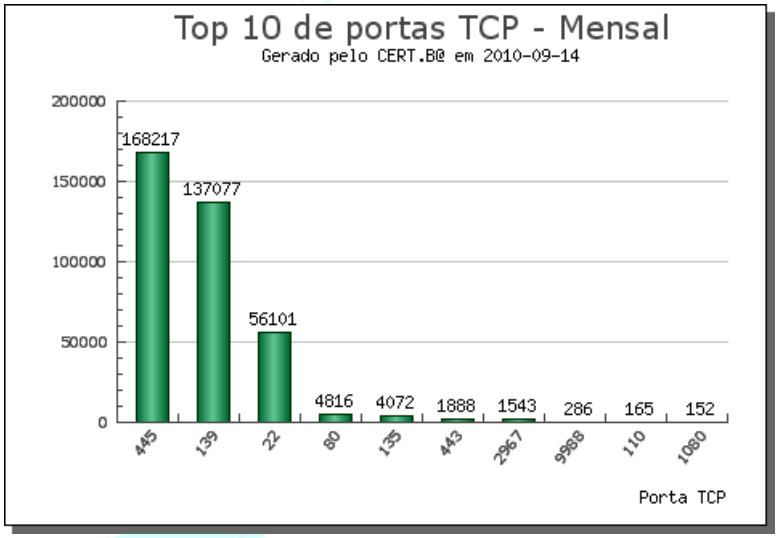
- Tratamento de incidente
- Educação e treinamento
- Alertas de segurança
- Estatísticas

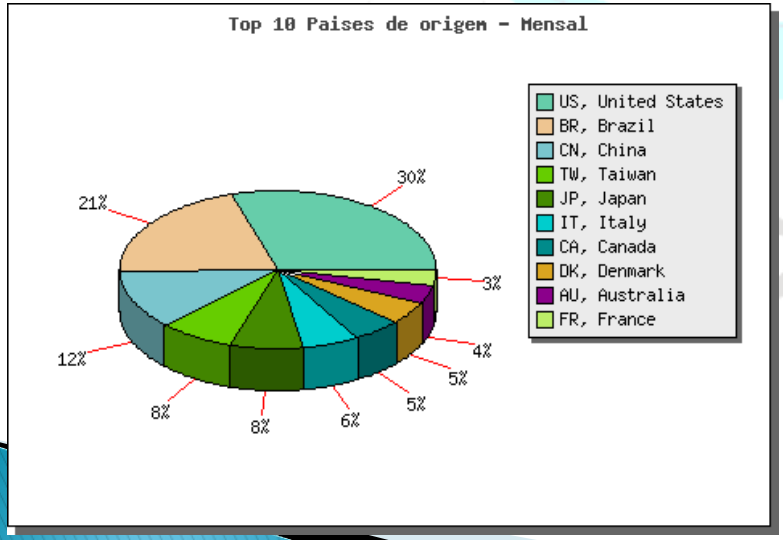
Serviços do CERT.Bahia

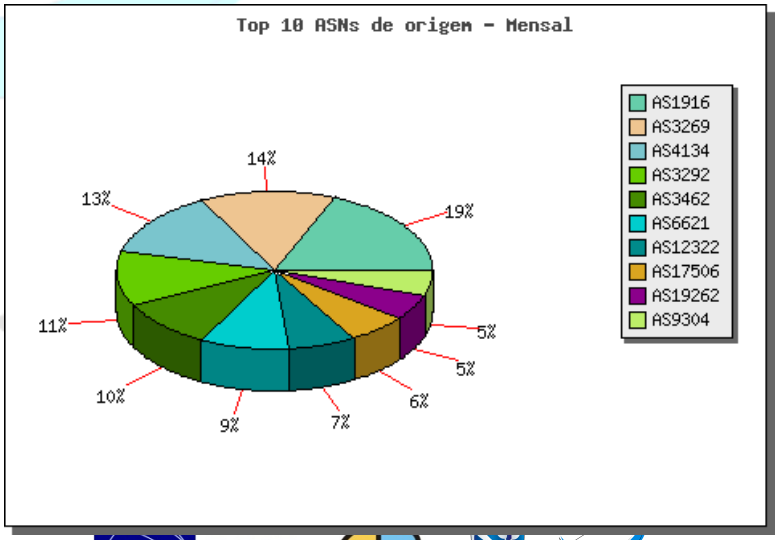
- Tratamento de incidente
- Educação e treinamento
- Alertas de segurança
- Estatísticas

Estatísticas do CERT.Bahia

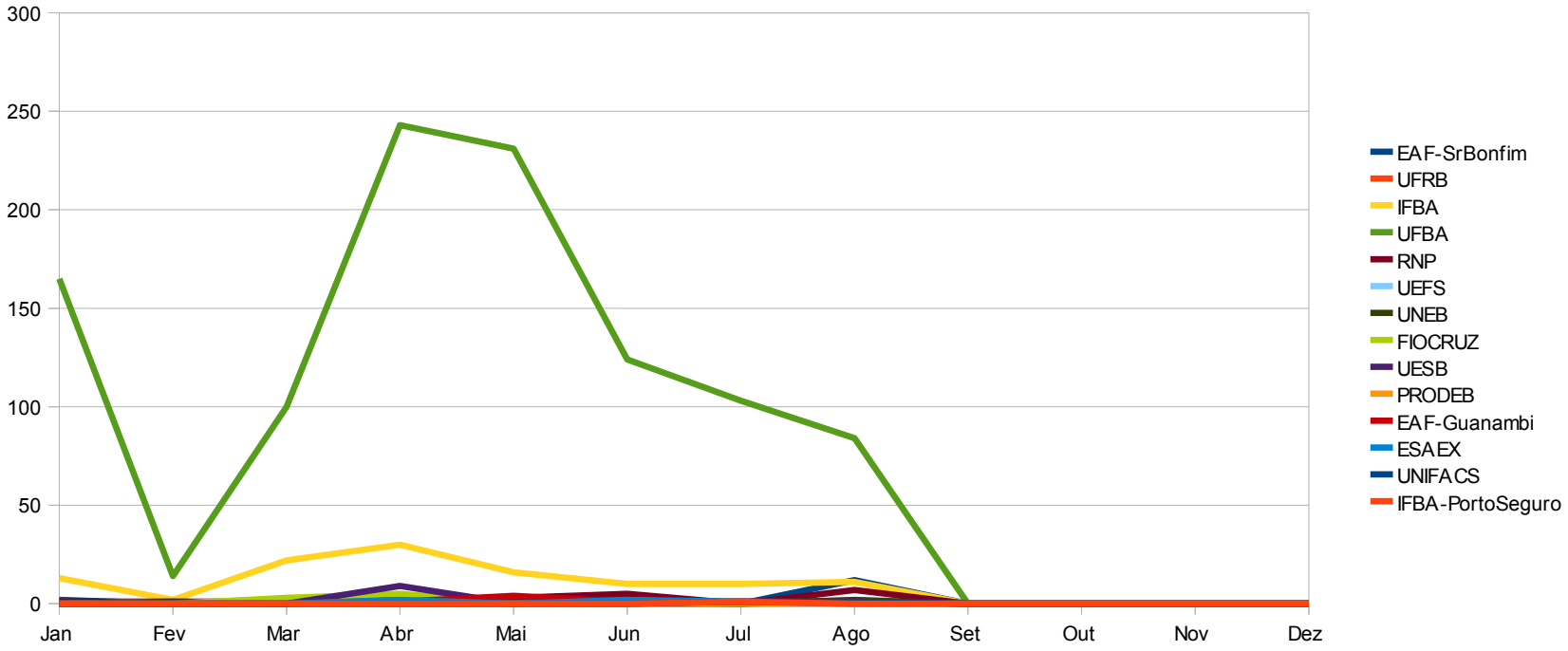








Estatísticas do CERT.Bahia



Estatísticas do CERT.Bahia

- Incidentes reportados em 2010 (até agosto)

Tipo de incidente	Qtd. Tickets
Tentativas de obter acesso não autorizado a sistemas ou dados	12
Host possivelmente infectado com Virus/Worm	1187
Scans	5
Envio de Spam	40
Mudanças nas configurações do sistema sem o consentimento prévio	4
Violação de copyright	17
Total	1265

Estatísticas do CERT.Bahia

Incidentes reportados pelo POP-BA / CERT.Bahia em 2010 (até agosto)

- Total de incidentes: 1265
  - Fechado por falta de resposta do cliente: 503
  - Incidente resolvido: 762

Estatísticas do CERT.Bahia

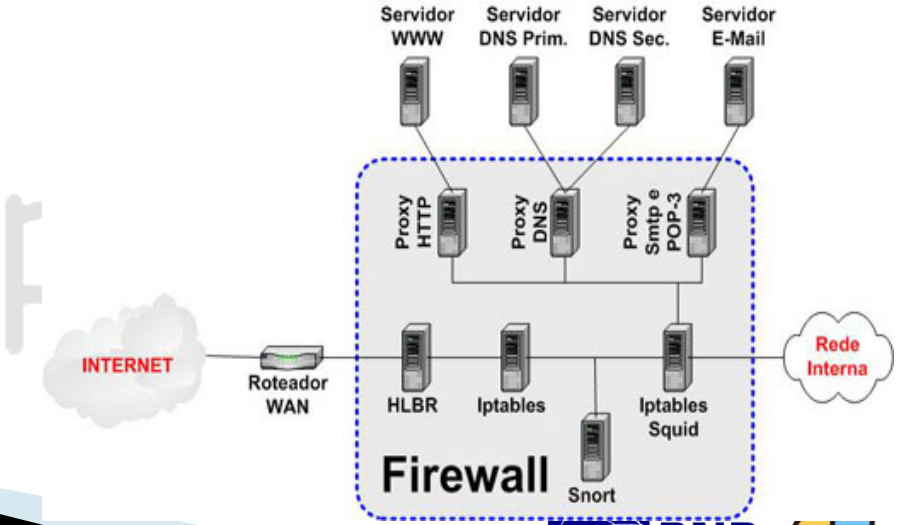
- Sistema de controle de chamados
  - https://suporte.pop-ba.rnp.br
- Palestras e treinamentos
- Software de tratamento de incidentes (em desen.)
  - Detecção e armazenamento de NATs
  - Detecção da máquina que gerou o incidente
  - Bloqueio da máquina para tratamento futuro

Configuração de Firewalls com IPTables

IPTables

O que é um firewall

- Software cujo objetivo é proteger a máquina de acesso/tráfego indesejado, proteger serviços, evitar que dados sigilosos sejam acessados
- Tal sistema é composto por filtros de pacotes, filtros de estados, IDS, IPS, proxies etc.



IPTables

Tipos de firewall

- Filtro de pacote
- Gateway de aplicação
- Gateway a nível de circuito
- Proxy Server

IPTables

IPTables: características

- Especificação de portas/endereço de origem/destino
- Suporte a protocolos TCP/UDP/ICMP (incluindo tipos de mensagens icmp)
- Suporte a interfaces de origem/destino de pacotes
- Tratamento de tráfego dividido em chains (para melhor controle do tráfego que entra/sai da máquina e tráfego redirecionado.
- Permite um número ilimitado de regras por chain
- Muito rápido, estável e seguro
- Possui mecanismos internos para rejeitar automaticamente pacotes duvidosos ou mal formados.
- Suporte a módulos externos para expansão das funcionalidades padrões oferecidas pelo código de firewall

IPTables

Regras

- São como comandos passados ao *iptables* para que ele realize uma determinada ação.
- As regras são armazenadas dentro dos ***chains*** e processadas na ordem que são inseridas.
- As regras são armazenadas no kernel
- Exemplos:
  - ACCEPT
  - DROP
  - REJECT
  - LOG

IPTables

Tabelas

- São os locais usados para armazenar os chains e conjunto de regras com uma determinada característica em comum
  - ***Filter***: filtro de pacotes. Admite as chains INPUT, OUTPUT e FORWARD.
  - ***NAT***: Network Address Translation. Admite as chains PREROUTING, OUTPUT e POSTROUTING.
  - ***Mangle***: Modificação do cabeçalho TCP (QoS). Admite as chains PREROUTING, POSTROUTING, OUTPUT, INPUT e FORWARD

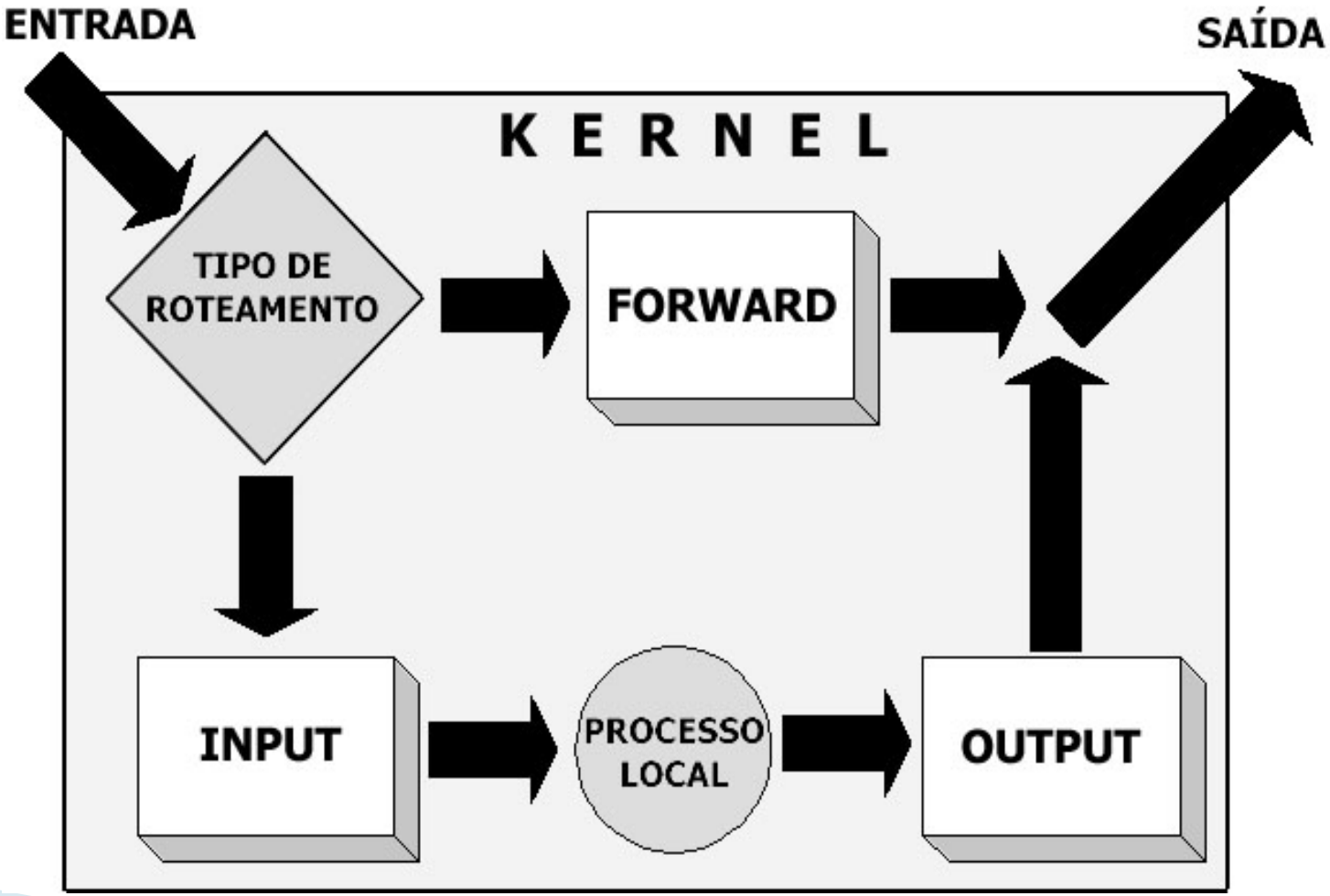
IPTables

Chains da tabela Filter

- ***INPUT***: utilizada quando o destino final é a própria máquina filtro;
- ***OUTPUT***: qualquer pacote gerado na máquina fltro e que deva sair para a rede será tratado pela chain OUTPUT;
- ***FORWARD***: qualquer pacote que atravessa o fltro, oriundo de uma máquina e direcionado a outra, será tratado pela chain FORWARD.

IPTables

Chains da tabela Filter



IPTables

Regras de filtragem

- As regras (rules) de filtragem, geralmente, são compostas assim:
  - ***iptables [-t tabela] [opção] [chain] [dados] -j [ação]***
- Exemplo:
  - ***iptables -A FORWARD -d 192.168.1.1 -j DROP***
- A linha acima determina que todos os pacotes destinados à máquina 192.168.1.1 devem ser descartados.

IPTables

Regras de filtragem – opções

- **-P**: Policy (política). Altera a política da chain. Só aceita DROP e ACCEPT
  - ***iptables -P FORWARD DROP***
  - ***iptables -P INPUT ACCEPT***
- **-A**: Append (anexar). Acresce uma nova regra à chain.
  - ***iptables -A OUTPUT -d 172.20.5.10 -j ACCEPT***
  - ***iptables -A FORWARD -s 10.0.0.1 -j DROP***
  - ***iptables -A FORWARD -d www.chat.com.br -j DROP***
- **-D**: Delete (apagar). Apaga uma regra. A regra deve ser escrita novamente:
  - ***iptables -D FORWARD -s 10.0.0.1 -j DROP***

IPTables

Regras de filtragem – opções

- **-L:** List (listar). Lista as regras existentes.
  - ***iptables -L***
  - ***iptables -t nat -L***
  - ***iptables -L FORWARD***
- **-F**: Flush (esvaziar). Remove todas as regras existentes. No entanto, não altera a política (-P).
  - ***iptables -F***
  - ***iptables -F FORWARD***

IPTables

Regras de filtragem – dados

- **-s:** Source (origem). Estabelece a origem do pacote. Geralmente é uma combinação do endereço IP com a máscara de sub-rede, separados por uma barra.
  - -s 172.20.5.10
  - -s 192.168.0.0/24
- **-d**: Destination (destino). Estabelece o destino do pacote. Funciona exatamente como o -s, incluindo a sintaxe.

IPTables

Regras de filtragem – dados

- **-p**: Protocol (protocolo). Especifica o protocolo a ser filtrado. Os protocolos mais utilizados são udp, tcp e icmp.
  - ***-p icmp***
- **-i**: In-Interface (interface de entrada). Especifica a interface de entrada.
  - ***-i ppp0***
  - ***-i eth+*** (várias interfaces)
- **-o**: Out-Interface (interface de saída). Especifica a interface de saída.

IPTables

Regras de filtragem – dados

- **!**: Exclusão. Utilizado com -s, -d, -p, -i, -o e outros, para excluir o argumento.
  - ***-s ! 10.0.0.1***
  - ***-p ! tcp***
- **--sport**: Source Port. Porta de origem. Só funciona com as opções -p udp e -p tcp.
  - ***-p tcp --sport 80***
  - ***-p tcp --sport 1000:2000***
- **--dport**: Destination Port. Porta de destino. Só funciona com as opções -p udp e -p tcp. Similar a --sport.

IPTables

Regras de filtragem – ações

- **ACCEPT**: Aceitar. Permite a passagem do pacote.
- **DROP**: Abandonar. Não permite a passagem do pacote, descartando-o. Não avisa a origem sobre o ocorrido.
- **REJECT**: Igual ao DROP, mas avisa a origem sobre o ocorrido (envia pacote icmp unreachable).
- **LOG**: Cria um log referente à regra, em /var/log/messages. Usar antes de outras ações.

IPTables

Regras de filtragem – logging

- ***LOG*** - Esse módulo provê logging dos pacotes submetidos. Possui as seguintes opçoes adicionais:
  - ***--log-level*** - Seguido de um número de nível ou nome. Os nome válidos (sensíveis a maiúsculas/minúsculas) são `debug', `info', `notice', `warning', `err', `crit', `alert' and `emerg', correspondendo a números de 7 até 0.
  - ***--log-prefix*** - Seguido de uma string de até 29 caracteres, esta será adicionada no início da mensagem de log, permitindo melhor identificação da mesma.

IPTables

Regras de filtragem – logging

- Exemplo:

```
# outras regras
iptables -A FORWARD -j LOG --log-prefix 'FORWARD NEGADO: '
iptables -A FORWARD -j DROP
```

- No /var/log/messages:

```
Sep 14 14:49:20 fwpop kernel: [4132057.694304] FORWARD NEGADO: IN=eth0
 OUT=eth1 SRC=192.168.0.1 DST=200.128.6.148 LEN=84 TOS=0x00 PREC=0x00
 TTL=63 ID=0 DF PROTO=ICMP TYPE=8 CODE=0 ID=4646 SEQ=1
Sep 14 14:49:21 fwpop kernel: [4132058.695938] FORWARD NEGADO: IN=eth0
 OUT=eth1 SRC=192.168.0.1 DST=200.128.6.148 LEN=84 TOS=0x00 PREC=0x00
 TTL=63 ID=0 DF PROTO=ICMP TYPE=8 CODE=0 ID=4646 SEQ=2
...
```

IPTables

Regras de filtragem – checagem de estado do pacote

- O módulo state interpreta o controle de conexão feito pelo ip_conntrack e fornece filtragem baseada no estado do pacote
- Usage: ***-m state --state <lista de estados>***
- Estados:
  - ***NEW***: Um pacote que cria uma nova conexão
  - ***ESTABLISHED***: Um pacote que pertence a uma conexão existente (um pacote de resposta, um pacote saindo por uma conexão na qual já houveram respostas).

IPTables

Regras de filtragem – checagem de estado do pacote

- Estados (cont.):
  - ***RELATED*** - Um pacote relacionado, mas que não faz parte, de uma conexão existente, como um erro ICMP, ou, um pacote estabelecendo uma conexão de dados FTP.
  - ***INVALID*** - Um pacote que não pôde ser identificado por alguma razão: isso inclui falta de memória e erros ICMP que não correspondem a qualquer conexão conhecida. Geralmente tais pacotes devem ser descartados.
- Exemplo:

```
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A FORWARD -m state --state RELATED,ESTABLISHED -j ACCEPT
```

IPTables

Exemplos

- Os pacotes oriundos da sub-rede 10.0.0.0 (máscara 255.0.0.0) e destinados aos hosts cujos endereços IP respondem pelo nome www.chat.com.br deverão ser descartados.
  - ***iptables -A FORWARD -s 10.0.0.0/8 -d www.chat.com.br -j DROP***

IPTables

Exemplos

- Os pacotes icmp oriundos do host 10.0.0.5 e destinados a qualquer host deverão ser descartados.
  - ***iptables -A FORWARD -s 10.0.0.5 -p icmp -j DROP***

IPTables

Exercício

- Qual o objetivo das regras abaixo?
  - ***iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT***
  - ***iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT***

IPTables

Exercício

- Qual o objetivo da regra abaixo?
  - ***iptables -A FORWARD -s 10.0.0.5 -p tcp --sport 80 -j LOG***

IPTables

Exercício

- Qual o objetivo da regra abaixo?
  - ***iptables -A FORWARD -p tcp --dport 25 -j ACCEPT***

IPTables

Impasse e ordem de processamento

- As regras serão interpretadas na ordem em que aparecerem. Sempre que um pacote se adequar a uma regra, tal regra processará o pacote e a sequência iptables será finalizada naquele instante, sem que as regras seguintes atuem. Isso não se aplicará às regras terminadas com ***-j LOG***.
- Exemplo:
  - ***iptables -A FORWARD -p icmp -j DROP*** #usada
  - ***iptables -A FORWARD -p icmp -j ACCEPT***

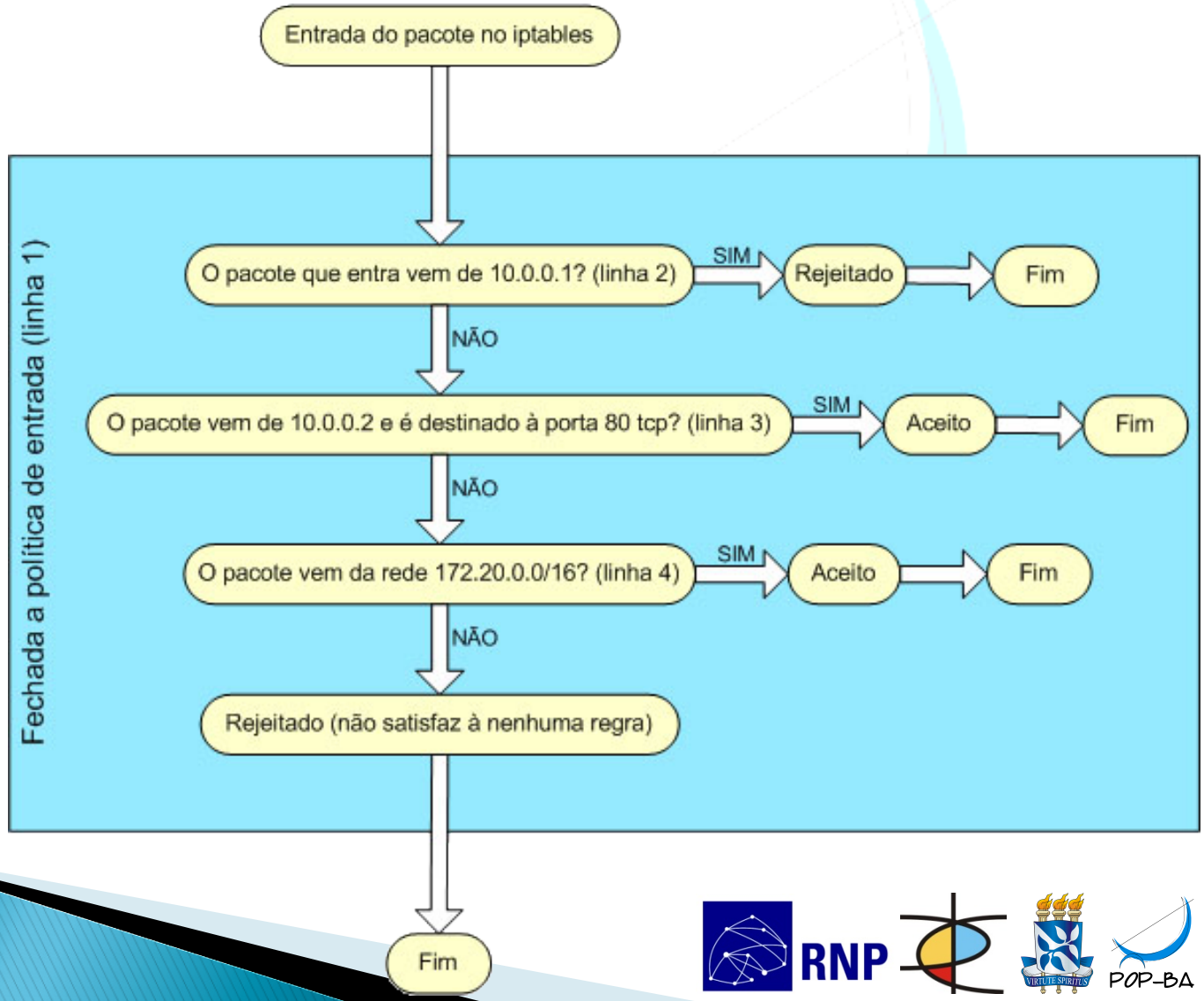
IPTables

Impasse e ordem de processamento

- Exemplo 1
  - ***iptables -P INPUT DROP***
  - ***iptables -A INPUT -s 10.0.0.1 -j DROP***
  - ***iptables -A INPUT -s 10.0.0.2 -p tcp --dport 80 -j ACCEPT***
  - ***iptables -A INPUT -s 172.20.0.0/16 -j ACCEPT***

IPTables

Impasse e ordem de processamento – fluxo do pacote



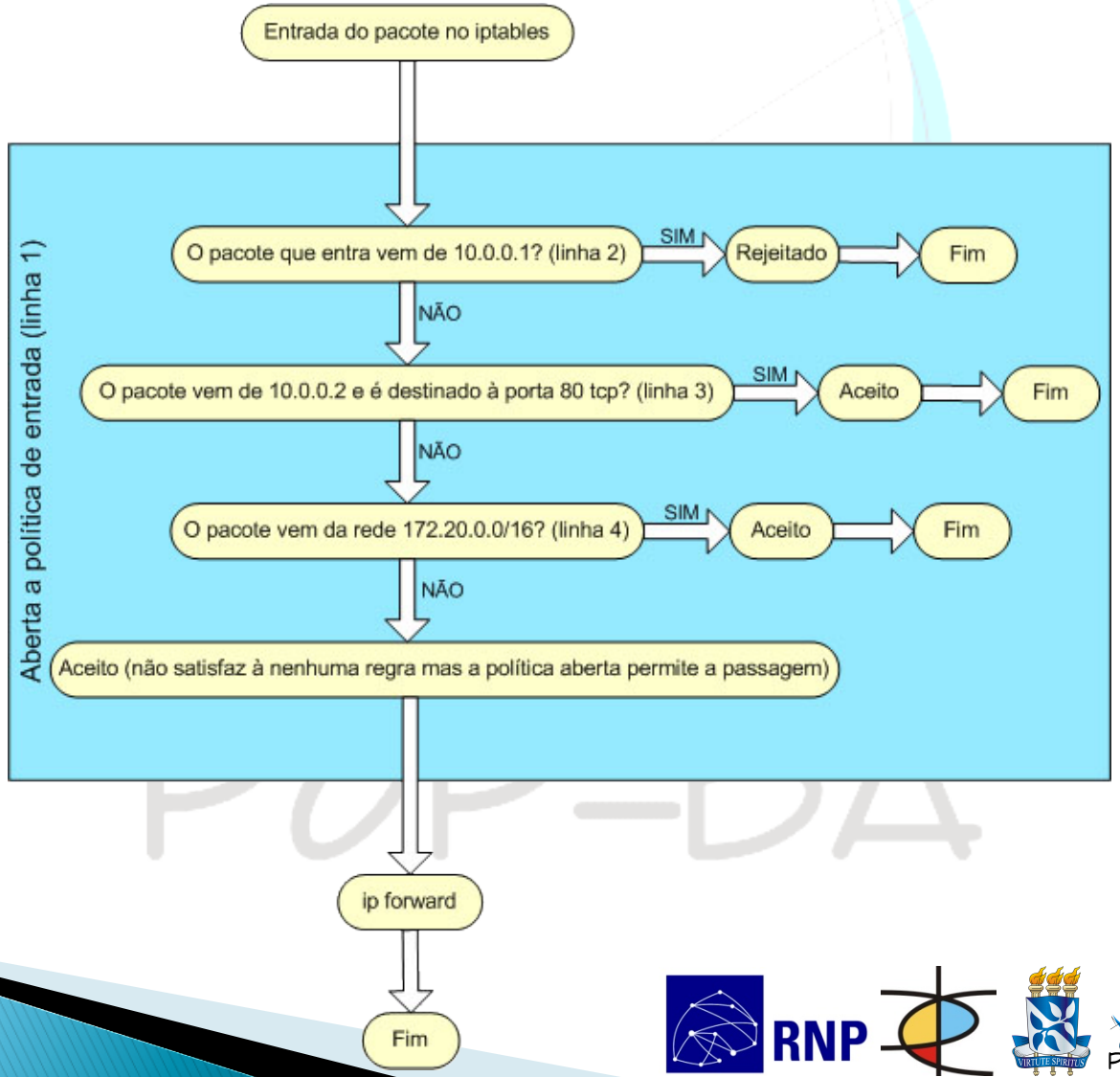
IPTables

Impasse e ordem de processamento

- Exemplo 2:
  - ***iptables -P INPUT ACCEPT***
  - ***iptables -A INPUT -s 10.0.0.1 -j DROP***
  - ***iptables -A INPUT -s 10.0.0.2 -p tcp --dport 80 -j ACCEPT***
  - ***iptables -A INPUT -s 172.20.0.0/16 -j ACCEPT***

IPTables

Impasse e ordem de processamento – fluxo do pacote



IPTables

Impasse e ordem de processamento

- Devemos planejar todo o fluxo do pacote antes de fazermos as regras, tanto na requisição quanto na resposta.
- Exemplo:
  - ***iptables -P FORWARD DROP***
  - ***iptables -A FORWARD -s 10.0.0.0/8 -d 172.20.0.0/16 -j ACCEPT***
- É preciso liberar a resposta da rede 172.20.0.0/16
  - ***iptables -P FORWARD DROP***
  - ***iptables -A FORWARD -s 10.0.0.0/8 -d 172.20.0.0/16 -j ACCEPT***
  - ***iptables -A FORWARD -d 10.0.0.0/8 -s 172.20.0.0/16 -j ACCEPT***

IPTables

Prática

- ***Prática***: Construindo firewalls no Linux com IPTables (ver roteiro de prática – prática 02)

IPTables

Tabela NAT – Network Address Translation

- A tabela nat serve para controlar a tradução dos endereços que atravessam o código de roteamento da máquina.
- Existem vários recursos que utilizam NAT. Os mais conhecidos são:
  - Mascaramento (masquerading)
  - Redirecionamento de portas (port forwarding ou PAT)
  - Redirecionamento de servidores (forwarding)
  - Proxy transparente (transparent proxy)
  - Balanceamento de carga (load balance)

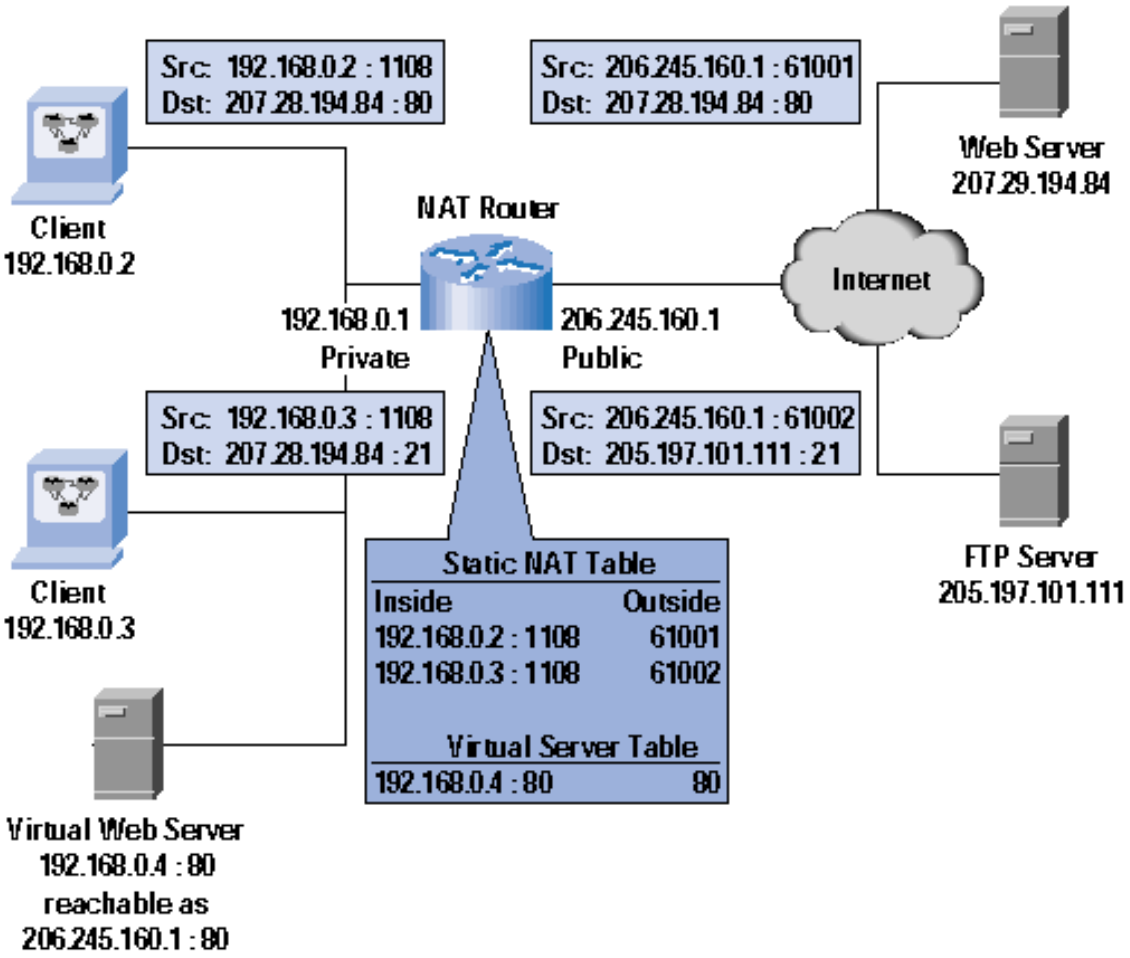
IPTables

NAT Masquerading

- Tráfico originado de todos os dispositivos em uma ou mais redes privadas serão encaminhados como se fossem originados de um simples endereço IP, roteável naquele segmento.
- Com isso, é possível fazer uma rede privada navegar na Internet.

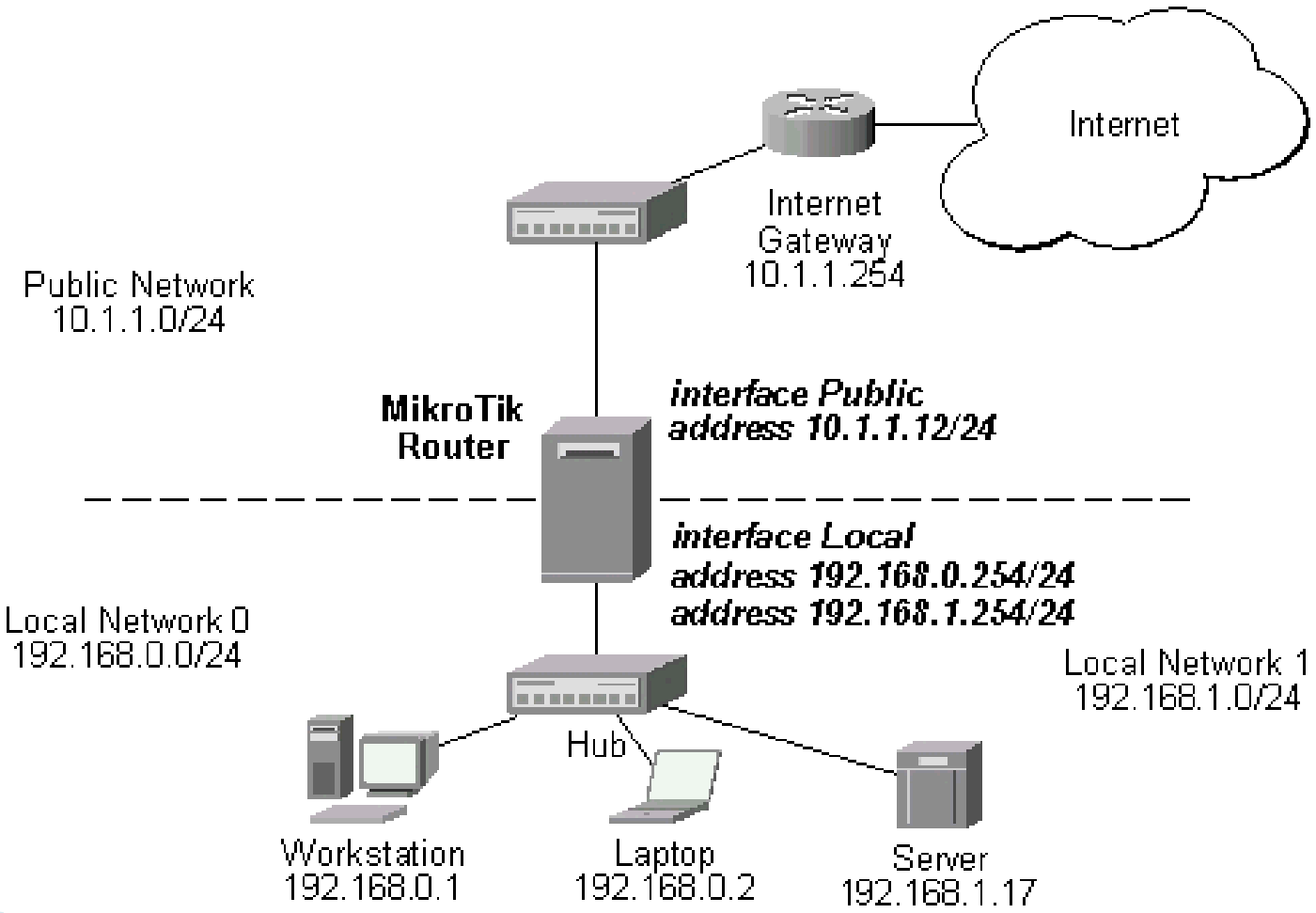
IPTables

NAT Masquerading



IPTables

NAT Masquerading



IPTables

NAT Masquerading

- Como fazer?
  - ***sysctl net.ipv4.ip_forward=1***
  - ***iptables -A POSTROUTING -t nat -o eth0 -s 192.168.1.0/24 -d 0/0 -j MASQUERADE***

IPTables

SNAT

- Consiste em modificar o endereço de origem das máquinas clientes antes dos pacotes serem enviados. A máquina roteadora é inteligente o bastante para lembrar dos pacotes modificados e reescrever os endereços assim que obter a resposta da máquina de destino, direcionando os pacotes ao destino correto
- Toda operação de SNAT é feita no chain POSTROUTING.

IPTables

SNAT

- Como implementar
  - ***iptables -t nat -A POSTROUTING -s 192.168.1.2 -o eth1 -j SNAT --to 200.200.217.40***

IPTables

DNAT

- Utilizado com PREROUTING e OUTPUT para fazer ações de redirecionamento de portas e servidores, balanceamento de carga e proxy transparente.
  - ***iptables -t nat -A PREROUTING -s 200.200.217.40 -i eth0 -j DNAT --to 192.168.1.2***

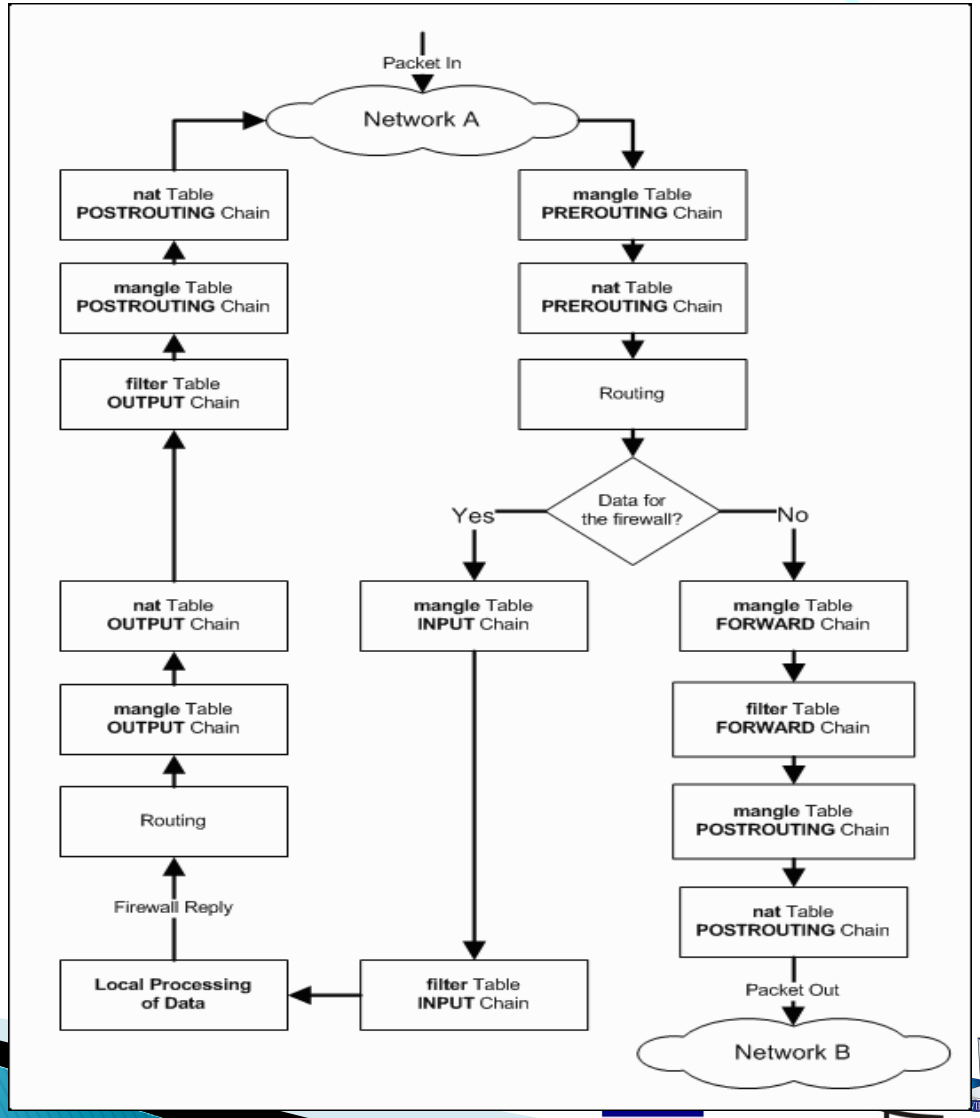
IPTables

DNAT

- Qual o objetivo das regras abaixo?
  - ***iptables -t nat -A OUTPUT -p tcp -d 10.0.0.10 -j DNAT --to 10.0.0.1***
  - ***iptables -t nat -A PREROUTING -i eth0 -j DNAT --to 172.20.0.1***

IPTables

Fluxo dos dados



IPTables

Prática

- ***Prática***: Configurar NAT Masquerade (ver roteiro de prática – prática 03)

Detecção de máquinas com NAT

Detecção de máquinas com NAT

- ***NAT*** é usado para traduzir endereços privados em endereços públicos
- *Network Address Port Translation* (NAPT)
  - Máquinas da intranet compartilhando um único IP público
- *NATificators* (firewalls, roteadores) em geral suportam *logging* das traduções
- Mas como detectar as máquinas da intranet?
  - IP dinâmico *versus* IP estático

Detecção de máquinas com NAT

- IP dinâmico *versus* IP estático
  - IP estático: documentação
  - IP dinâmico: DHCP + logging
- DHCP do windows não armazena log
- DHCP no Linux armazena log

Detecção de máquinas com NAT

- Exemplo de log do ***dhcp3-server***

```
Lease 192.168.0.8 {
  starts 5 2010/03/10 20:34:34;
  ends 5 2010/03/10 21:36:34;
  cltt 5 2010/03/10 20:34:34;
  binding state free;
  hardware ethernet 2a:81:58:e5:e3:f3;
  uid "\001*\201X\357\356#";
  client-hostname "teste-desktop";
}
```

Detecção de máquinas com NAT

- Alternativa: tabela ARP dos equipamentos
- ARP – Address Resolution Protocol
  - Em uma LAN onde as máquinas usam IPv4 sobre Ethernet, a comunicação é feita via MAC (L2)
  - O protocolo ARP mapeia endereços IPv4 em endereços MAC
  - Para melhorar o desempenho do mapeamento, utiliza-se das ***tabelas ARP*** (cache)

Detecção de máquinas com NAT

- Para visualizar a tabela ARP em sistemas GNU/Linux:

```
~# arp -n
Address         HWtype  HWaddress           Flags Mask   Iface
172.16.101.6    ether   00:1d:7e:c2:dd:a5   C            eth1
```

- Em equipamentos Cisco:

```
#show arp
Protocol  Address          Age (min)  Hardware Addr   Type   Interface
Internet  200.143.252.251         -   0002.7d9c.f020  ARPA   FastEthernet1/0/0
Internet  200.143.252.250         2   0004.9641.2520  ARPA   FastEthernet1/0/0
Internet  200.143.252.249        18   0012.1e80.2cfc  ARPA   FastEthernet1/0/0
Internet  200.143.252.253         0   0023.9c81.2280  ARPA   FastEthernet1/0/0
Internet  200.143.252.252         2   0004.961e.5600  ARPA   FastEthernet1/0/0
```

Detecção de máquinas com NAT

- Existe uma MIB SNMP para exibir as informações da tabela ARP?
  - *Sim, ipNetToMediaPhysAddress = .1.3.6.1.2.1.4.22.1.2*
- Porém, nem todos os equipamentos a suportam
  - Alternativa: execução remota de comandos
- Como gerenciar todas essas informações de IPs e MACs?

Detecção de máquinas com NAT

L2M – Layer 2 Manager

- Software em desenvolvimento pelo POP-BA/RNP para auxiliar no gerenciamento dos endereços IP/MAC e histórico de utilização na rede
- Com o L2M é possível saber de forma simples qual máquina estava com determinado IP em determinado momento

Detecção de máquinas com NAT

L2M – Layer 2 Manager

- Hora de entrada e saída de uma máquina na rede
- Quantidade total de máquinas utilizando a rede
- Quantidade total de máquinas que acessaram a rede nos últimos 15 dias
- Detecção de endereços IP duplicados na rede (plano futuro)
- Detecção de endereços MAC duplicados na rede (plano futuro)

Detecção de máquinas com NAT

L2M – Layer 2 Manager – Status atual

- Em funcionamento na UFBA
  - Dez roteadores: sete Cisco + um 3COM + dois DLINK
  - Consultas via SNMP e EXPECT

Sistema de logging remoto

Sistema de logging remoto

- Os registros de logs armazenam uma série de eventos do sistema (erros, alertas, violações, etc.)
- Logs são importantes para procedimentos de manutenção preventiva e corretiva
- Segurança da rede
  - Alerta de atividades suspeitas
  - Determinando a extensão das atividades de um intruso

Sistema de logging remoto

- Por padrão, são armazenados em arquivos de texto no próprio sistema
  - Vulnerável à alterações ou remoção em caso de o sistema estar comprometido
- Soluções de recuperação ou controle de integridade/versão ajudam mas não resolvem totalmente o problema
  - Solução: ***servidor de log centralizado***, armazenando de forma segura os registros de log

Sistema de logging remoto

- Servidor de log centralizado
  - Facilidade de gerenciamento e monitoramento dos logs
  - Centraliza a demanda de recursos computacionais para armazenamento dos logs
  - Facilidade para instalação de ferramentas de análise e geração de relatórios

Sistema de logging remoto

- Ferramentas utilizadas
  - Servidor de logs: ***syslog-ng***
  - Rotacionamento dos logs: ***logrotate***
  - Análise dos logs: ***swatch, logcheck, OSSEC***, etc.

Sistema de logging remoto

Syslog-ng: *Syslog New Generation*

- Pode atuar tanto como cliente quanto como servidor de logs
- Disponível para diversos sistemas: Linux,BSD, AIX, HP-UX e Solaris
- Permite filtragem por facility, level, data, origem, regex, etc.

Sistema de logging remoto

Syslog-ng: *Syslog New Generation*

- ***Facility*** é um tag que cada mensagem do syslog possui para identificar a parte do sistema que gerou o log.
- Exemplo: auth, cron, daemon, ftp, kern, mail, syslog, user, local0..local7, etc.

Sistema de logging remoto

Syslog-ng: *Syslog New Generation*

- ***Level*** indica quão crítica é a mensagem de log enviada ao sistema.
- Lista de severidade: emerg, alert, crit, err, warning, notice, info, debug, none

Sistema de logging remoto

Syslog-ng – Instalação e configuração

- Instalação em sistemas Debian:
  - ***aptitude install syslog-ng***
- Configuração
  - /etc/syslog-ng/syslog-ng.conf
- Reiniciar o daemon
  - ***/etc/init.d/syslog-ng restart***

Sistema de logging remoto

Syslog-ng - /etc/syslog-ng/syslog-ng.conf

- ***options {}***
  - time_reopen(60): tempo para reconectar com o cliente em caso de erro
  - use_fqdn(no)
  - keep_hostname(yes)
  - sync(0): quantidade de mensagens na memória antes de salvar no disco
  - create_dirs(yes)
  - owner(root), group(root), perm(0600)
  - dir_owner(root), dir_group(root), dir_perm(0700)

Sistema de logging remoto

Syslog-ng - /etc/syslog-ng/syslog-ng.conf

- ***log {}*** - responsável por, efetivamente, armazenar os logs, fazendo a ligação entre *origem*, *destino* e *filtro*.

```
log {
        source(s_local); source(s_udp);
        filter(f_auth);
        destination(d_auth);
};
```

Sistema de logging remoto

Syslog-ng - /etc/syslog-ng/syslog-ng.conf

- ***source {}*** - onde receber as mensagens

```
source s_udp { udp(port(514));};

source s_tcp { tcp(port(514)); };

source equipamentos {udp(ip(192.168.0.1) port(1514));};

source s_local {
        internal();
        unix-stream("/dev/log");
        file("/proc/kmsg" log_prefix("kernel: "));
};
```

Sistema de logging remoto

Syslog-ng - /etc/syslog-ng/syslog-ng.conf

- ***filter {}*** - permite a criação de filtros para organizar os logs

```
# filtros por facility
filter f_cron { facility(cron); };
filter f_daemon { facility(daemon); };
filter f_auth { facility(auth, authpriv); };
filter f_mail { facility(mail) and level(info..warn); };
filter f_mail_err {  facility(mail) and level(err..emerg); };

# filtros por regex
filter f_blacklist { match("regex" value("blacklist: ")); };
```

Sistema de logging remoto

Syslog-ng - /etc/syslog-ng/syslog-ng.conf

- ***destination {}*** - onde enviar os logs

```
# No servidor
destination d_syslog { file("/var/log/$HOST/syslog.log"); };
destination d_auth { file("/var/log/$HOST/auth.log"); };
destination d_daemon { file("/var/log/$HOST/daemon.log"); };
destination d_local { file("/var/log/$HOST/$FACILITY.log"); };


# No cliente
destination d_server {tcp(ip(172.16.34.2) port(514));};
```

Sistema de logging remoto

Logrotate – rotacionando os logs

- Instalação em sistemas Debian
  - ***aptitude install logrotate***
- Configuração:
  - /etc/logrotate.d/*
- Rotina no CRON
  - /etc/cron.daily/logrotate

Sistema de logging remoto

Logrotate – rotacionando os logs

- Exemplo de configuração:

```
/var/log/srv01/*.log {
   rotate 156
   weekly
   compress
   olddir /var/log/srv01/old
}
/var/log/dbserver/*.log {
   rotate 156
   weekly
   compress
   olddir /var/log/dbserver/old
}
```

Sistema de logging remoto

- ***Prática***: Configurar um servidor de logs remoto e um cliente (ver roteiro de prática – prática 04)

Dúvidas?