Elvio de Sousa

Segurança para acesso a Rede da Brasiltelecom

Monografia apresentada ao Departamento de Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pósgraduação Latu Sensu em Administração de Redes Linux, para obtenção do título de especialista em Administração em Redes Linux.

Orientador:
Prof. Fernando Cortez Sica
Co-orientador:
Joaquim Quinteiro Uchôa

LAVRAS
MINAS GERAIS – BRASIL
2005

Elvio de Sousa

Segurança para acesso a Rede da Brasiltelecom

Monografia apresentada ao Departamento de Computação da Universidade Federal de Lavras, como parte das exigências do curso de Pósgraduação Latu Sensu em Administração de Redes Linux, para obtenção do título de especialista em Administração em Redes Linux.

APROVADA em _____ de _____ de _____

_____
Prof. Ricardo Martins de Abreu Silva (Dr – CC/UFPE)

_____
Prof. Joaquim Quinteiro Uchoa (Msc – CC/UFLA)
(Co-orientador)

_____
Prof. Fernando Cortez Sica (Msc - CC/UFOP)
(Orientador)

LAVRAS
MINAS GERAIS – BRASIL

RESUMO

A Brasil Telecom tem um Centro Nacional de Redes e Serviços que tem como missão assegurar a qualidade dos serviços oferecidos, zelando pelo cumprimento dos prazos requeridos pelo negócio. O CNRS lida com a gerência de plataformas, equipamentos e serviços de alto refinamento tecnológico, buscando atingir níveis de qualidade compatíveis com as exigências do mercado. Tais responsabilidades exigem que certas ações sejam executadas a qualquer momento, independente da situação. Contudo para se ter acesso aos equipamentos gerenciados pelo CNRS, seus funcionários ou qualquer pessoa que precise dar manutenção nestes, atualmente, deve estar fisicamente dentro do CNRS. Esta situação algumas vezes gera custos altos para a empresa e atrasos em soluções que podem prejudicar muito o cliente final.

Este trabalho pretende apresentar uma solução segura para este problema, utilizando algumas ferramentas que fazem parte do cotidiano dos vários Engenheiros e Técnicos responsáveis por zelar desta rede.

O acesso externo à rede deve ser feito de forma bastante pensada, pois além de implementar segurança de rede, por meio de *Firewalls*, IDS, *acces-lists* e outras ferramentas afins, deve ser implantada uma política de segurança bastante forte e clara, onde todos os envolvidos tenham o compromisso de segui-la.

Lista de Figuras

1 Introdução

O Centro Nacional de Redes e Serviços - CNRS insere-se numa complexa malha de eventos operacionais, que envolve diversos segmentos da Brasil Telecom: *Call Center*, Projetos, Consultorias, Clientes Especiais, Fornecedores, Centros de Operação e manutenção (COM), Filiais etc. O CNRS tem como missão assegurar a qualidade dos serviços oferecidos, zelando pelo cumprimento dos prazos requeridos pelo negócio, trazendo o máximo de benefícios à Brasil Telecom e aos clientes.

O CNRS lida com a gerência de plataformas, equipamentos e serviços de alto refinamento tecnológico, buscando atingir níveis de qualidade compatíveis com as exigências do mercado.

As responsabilidades:

- Gerenciar as Redes de Comunicação de Dados e Rede Inteligente da BrasilTelecom;
- Gerenciar os serviços de Comunicação de Dados e de Rede Inteligente da Brasil Telecom;
- Garantir os níveis de qualidade dos serviços, no que tange aos prazos de atendimento às solicitações de serviços e reparos;
- Prestar suporte técnico às áreas de operação e manutenção, consultorias e projetos;
- Interagir com os Centros de Gerência de outras Empresas Operadoras para o atendimento a serviços e reparos.

Essas responsabilidades exigem que certas ações sejam executadas a qualquer momento, independente da situação. Contudo para se ter acesso aos equipamentos gerenciados pelo CNRS, seus funcionários ou qualquer pessoa que precise dar manutenção nestes, atualmente, deve estar fisicamente dentro do CNRS.

Esta situação algumas vezes gera custos altos para a empresa e atrasos em soluções que podem prejudicar muito o cliente final.

Atualmente a agilidade é muito importante, pois gera lucro ou diminui gastos. Grandes empresas precisam ser ágeis pois seus clientes estão cada dia mais exigentes e não podem ser penalizados por soluções demoradas à problemas em serviços dos quais eles utilizam.

Atualmente a internet nos permite conectar a qualquer lugar e a qualquer momento, contudo estamos compartilhando o mesmo ambiente com inúmeras outras pessoas, algumas, donas de conhecimentos que com informações corretas podem destruir sistemas, tudo através na internet.

Este trabalho pretende apresentar uma solução segura para **prover acesso seguro àqueles profissionais através da internet**, utilizando algumas ferramentas que fazem parte do cotidiano dos vários Engenheiros e Técnicos responsáveis por zelar desta rede.

O acesso externo à rede deve ser feito de forma bastante pensada, pois além de implementar segurança de rede, por meio de *Firewalls*, IDS, *acces-lists* e outras ferramentas afins, deve ser implantada uma política de segurança bastante forte e clara, onde todos os envolvidos tenham o compromisso de segui-la.

Se tratando de assunto muito delicado, do ponto de vista de uma empresa como esta, visto que um acesso deste tipo nas mãos de

uma pessoa mal intencionada pode causar um problema de proporções catastróficas à empresa.

Portanto neste trabalho serão abordados os principais problemas enfrentados atualmente, quais serão os benefícios trazidos por esta solução e quais serão os desafios para deixá-la o mais completa possível, mostrando desde os passos para a implantação do processo de política de segurança, conceitos básicos até a configuração de ferramentas necessárias para a implantação da segurança dos equipamentos de rede utilizados para prover este tipo de acesso externo aos profissionais envolvidos.

2 POLÍTICAS DE USO

A política de uso aceitável (AUP--*Acceptable* Use *Policy*) é o documento que define como os recursos computacionais da organização podem ser utilizados. Ela deve ser pública e estar disponível a todos os que utilizam a infra-estrutura computacional da organização, sendo recomendável que a autorização para o uso dos recursos seja condicionada a uma concordância expressa com os seus termos.

A AUP é geralmente parte integrante da política de segurança global. Para muitas organizações, ela será composta pelos itens da política que afetam diretamente os usuários de recursos computacionais, principalmente os que definem seus direitos e responsabilidades.

Por outro lado, organizações que oferecem acesso a usuários externos devem definir uma política de uso aceitável para esses usuários que seja independente da AUP à qual estão sujeitos os seus usuários internos. É importante que os usuários externos tomem conhecimento dessa política e saibam que o uso dos recursos está condicionado ao seu cumprimento.

2.1 POLÍTICAS DE SEGURANÇA

Atualmente o crescimento da interatividade nos mais diferentes ramos dentro de empresas e mesmo entre empresas, gera uma demanda elevada em relação as facilidades de acessar e compartilhar informações entre grupos de trabalho, muitas vezes envolvidos em projetos conjuntos. Esta facilidade deve ser transparente

o suficiente para permitir agilidade e em contra partida oferecer segurança para evitar roubo ou cópia ilícita de informação.

Os administradores de sistemas estão adotando posturas controladoras, optando por estruturar documentos que apresentam de forma explícita a posição da empresa perante seus dados, descrevendo como será o controle, os níveis de segurança entre outros detalhes cruciais para total comprometimento do usuário com o sistema.

Este documento é conhecido por política de segurança e define o que é permitido e o que é proibido em um sistema, ou seja, a expressão formal das regras pelas quais é fornecido acesso aos recursos tecnológicos da empresa. Existem basicamente duas filosofias sustentando qualquer política de segurança:

- **Proibitiva**: tudo que não é expressamente permitido é proibido.
- **Permissiva**: tudo que não é expressamente proibido é permitido.

As decisões tomadas dentro de uma empresa relacionadas a segurança irão determinar quão segura ou insegura é a rede, quantas funcionalidades ela irá oferecer e qual será a facilidade de utilizá-la. Porém, não se consegue tomar boas decisões sem antes determinar quais são as metas em relação à segurança, comprometendo a utilização efetiva das ferramentas de segurança.

Os objetivos da segurança devem ser determinados observando as seguintes condições:

- **Serviços oferecidos versus segurança oferecida**: cada serviço oferecido para os usuários carrega seus

próprios riscos de segurança. Para alguns serviços o risco é superior ao benefício oferecido, o administrador deve optar por eliminar o serviço ao invés de tentar torná-lo mais seguro.

- **Finalidade do uso versus segurança**: o sistema mais fácil de usar deveria permitir acesso a qualquer usuário e não exigir senha, isto é, não haveria segurança. Solicitar senhas torna o sistema pouco conveniente porém mais seguro.
- **Custo da segurança versus o risco da perda**: Há muitos custos (equipamento, treinamento etc) diferentes para segurança:
- **Monetário:** o custo da aquisição de hardware e software.
- **Desempenho**: cifrar e decifrar dados, mensagens, informações etc.

Muitos níveis de risco devem ser analisados:

- Perda de privacidade.
- Perda de serviços.
- Perda de dados.

2.2 ELEMENTOS DE UMA POLÍTICA DE SEGURANÇA

Um sistema de computadores pode ser considerado um conjunto de recursos que são disponibilizados para serem utilizados por usuários autorizados. São descritos alguns elementos que devem ser contemplados em uma política de segurança:

- **Disponibilidade**: o sistema deve estar disponível para uso quando o usuário precisar. Dados críticos devem estar disponíveis de forma ininterrupta.
- **Utilização**: o sistema e os dados devem ser utilizados para as devidas finalidades.
- **Integridade**: o sistema e os dados devem estar completamente íntegros e em condições de serem utilizados.
- **Autenticidade**: o sistema deve ter condições de verificar a identidade do usuário e o usuário deve ter condições de verificar a identidade do sistema.
- **Confidencialidade**: dados privados devem ser apresentados somente para os donos dos dados ou grupo de usuários para o qual o dono dos dados permitir.
- **Posse**: o dono do sistema deve ter condições de controlá-lo.

2.3 PROBLEMAS EM PROVER SEGURANÇA

Muitos são os fatores que influenciam o processo de criação de um plano de segurança destacando, os seguintes:

- Falta de treinamento dos usuários.
- Avaliação errônea dos riscos.
- Funcionários insatisfeitos.
- Vírus.
- Invasões/Ataques.

- Lentidão na detecção de ataques/tentativas de ataque.

Os dois últimos são em especial preocupantes por se tratar de situações onde o administrador não consegue humanamente manter uma análise ininterrupta dos pacotes TCP (*Transmission Control Protocol*) que trafegam pela internet/intranet.

2.4 MEDIDAS DE SEGURANÇA

Foram tratadas medidas basicamente procedimentais, mas faz-se necessário um conjunto de medidas, tais como (NIC BR, 2003):

- Definição da política de segurança e política de uso aceitável;
- Elaboração de uma arquitetura de rede segura;
- Elevação do nível de segurança dos *hosts*;
- Monitoração do tráfego da rede e dos serviços;
- Definição de testes periódicos à procura de vulnerabilidades.

3 AMEAÇAS E CONTRAMEDIDAS

Um invasor procura dispositivos de rede mal configurados para explorar. As vulnerabilidades comuns incluem configurações de instalação padrão inadequadas, controles de acesso totalmente abertos e dispositivos sem atualização. As ameaças de alto nível à rede são as seguintes:

- Coleta de informações
- Escuta por *sniffer*
- *Spoofing*
- Seqüestro de sessão
- Negação de serviço

Conhecendo as ameaças que podem afetar a rede, é possível aplicar contramedidas eficazes.

3.1 COLETA DE INFORMAÇÕES

A coleta de informações pode revelar informações detalhadas sobre a topologia da rede, configurações dos sistemas e dispositivos de rede. Um invasor usa essas informações para montar ataques certeiros contra as vulnerabilidades descobertas.

3.1.1 VULNERABILIDADES

As vulnerabilidades comuns que tornam a rede suscetível a ataque incluem:

- A natureza inerentemente desprotegida do conjunto de protocolos TCP/IP
- As informações de configuração contidas nos cabeçalhos.
- Serviços expostos que deveriam ser bloqueados.

3.1.2 ATAQUES

Ataques comuns de coleta de informações incluem:

- O uso do **Tracert** para detectar nova topologia da rede.
- O uso do **Telnet** para abrir portas e capturar cabeçalhos.
- O uso de verificações de porta para detectar portas abertas.
- Uso de solicitações de difusão para enumerar os *hosts* de uma sub–rede.

3.1.3 CONTRAMEDIDAS

Pode-se empregar as seguintes contramedidas:

- Usar cabeçalhos de serviço genéricos que não contenham informações de configuração, como versões ou nomes de software.
- Usar *firewalls* para mascarar os serviços que não devem ser expostos publicamente.

3.2 INVASÃO POR SNIFFER

A invasão por sniffer, também chamada de *escuta*, é o ato de monitorar o tráfego na rede em busca de dados, como senhas ou informações de configuração com texto não criptografado. Com um simples *sniffer* de pacotes, todo o tráfego com texto não criptografado

pode ser lido facilmente. Além disso, os algoritmos de *hash* superficiais podem ser violados e as cargas, que supostamente estariam seguras, podem ser decifradas.

3.2.1 VULNERABILIDADES

As vulnerabilidades comuns que tornam a rede suscetível à invasão de dados por *sniffer* incluem:

- Segurança física ineficaz.
- Falta de criptografia ao serem enviados dados confidenciais.
- Serviços que se comunicam por meio de texto sem formatação ou com criptografia ou *hash* de pouca segurança.

3.2.2 ATAQUES

O invasor coloca a ferramenta de invasão por *sniffer* de pacotes na rede para capturar todo o tráfego.

3.2.3 CONTRAMEDIDAS

As contramedidas incluem:

- Segurança física eficiente, que evita que dispositivos invasores sejam colocados na rede.
- Credenciais e tráfego de aplicativo criptografados na rede.

3.3 SPOOFING

O spoofing, também chamado de *camuflagem de identidade*, é um meio de ocultar a identidade real de alguém na rede. É usado um endereço de origem falso que não representa o endereço real do originador do pacote. O *spoofing* pode ser usado para ocultar a origem de um invasor ou para contornar as ACLs (Listas de Controle de Acesso) usadas para limitar o acesso ao host segundo regras de endereço.

3.3.1 VULNERABILIDADES

As vulnerabilidades comuns que tornam a rede suscetível ao *spoofing* incluem:

- A natureza inerentemente desprotegida do conjunto de protocolos TCP/IP.
- Falta de filtragem de entrada e de saída. A filtragem de entrada é a filtragem de qualquer pacote IP com endereço de origem não confiável antes que ele tenha a chance de entrar e afetar o sistema ou a rede. Filtragem de saída é o processo de filtragem do tráfego que sai da rede.

3.3.2 ATAQUES

Um invasor pode usar várias ferramentas para modificar pacotes enviados de modo que pareçam ter sido originados por uma outra rede ou um outro host.

3.3.3 CONTRAMEDIDAS

A filtragem de entrada e de saída pode ser usada em roteadores de perímetro. Roteadores que ficam no perímetro(entrada) da rede fazendo a segurança na maioria das vezes com *access-list* ou com um sistema operacional que funcione como *firewall*.

3.4 SEQÜESTRO DE SESSÃO

Com o seqüestro de sessão (*hijack*), também conhecido como ataque de interceptação, o invasor usa um aplicativo que se faz passar pelo cliente ou pelo servidor. Isso faz com que o servidor ou o cliente seja levado a pensar que o *host* upstream é o *host* legítimo. Entretanto, o *host upstream* é, na verdade, o host do invasor que está manipulando a rede de modo a parecer que ele é o destino desejado. O seqüestro de sessão pode ser usado para obter informações de logon que depois poderão ser usadas para acessar um sistema ou informações confidenciais.

3.4.1 VULNERABILIDADES

As vulnerabilidades comuns que tornam a rede suscetível ao seqüestro de sessão incluem:

- Segurança física ineficaz.
- A inerente falta de segurança do conjunto de protocolos TCP/IP.

- Comunicação não criptografada.

3.4.2 ATAQUES

Um invasor pode usar várias ferramentas para combinar alterações de roteamento e manipulação de pacotes.

3.4.3 CONTRAMEDIDAS

As contramedidas incluem:

- Criptografia de sessão.
- Inspeção com informações de estado no *firewall*.

3.5 NEGAÇÃO DE SERVIÇO

Um ataque de negação de serviço é o ato de negar aos usuários legítimos o acesso a um servidor ou serviço. Os ataques de negação de serviço na camada de rede geralmente tentam negar o serviço inundando a rede de tráfego, o que consome a largura de banda e os recursos disponíveis.

3.5.1 VULNERABILIDADES

As vulnerabilidades que aumentam as oportunidades de negação de serviço incluem:

- A inerente falta de segurança do conjunto de protocolos TCP/IP.
- Configuração ineficaz de roteador e switch.

- Comunicação não criptografada.
- *Bugs* do software de serviço.

3.5.2 ATAQUES

Os ataques comuns de negação de serviço incluem:

- Inundação de pacotes a força bruta, como ataques de difusão em cascata.
- Ataques SYN flood.
- Explorações de serviços, como estouros de buffer.

3.5.3 CONTRAMEDIDAS

As contramedidas incluem:

- Filtragem de solicitações de difusão.
- Filtragem de solicitações ICMP (Internet Control Message Protocol).
- Aplicação de *patches* e atualizações ao software de serviço.

4 ARQUITETURA DE REDE SEGURA

Segundo Uchôa (2003), uma ferramenta de *software* o *hardware* situada entre duas redes(uma interna e outra externa), responsável por filtrar os pacotes, evitando o acesso externo a determinados serviços é chamada de *Firewall*. Entretanto *firewalls* não são mecanismos exclusivos para proteger a rede interna da rede externa (que pode ser

qualquer rede, a Internet é apenas o exemplo mais significativo de redes).

Atuantes como barreiras de segurança, *firewalls* são úteis em qualquer ponto estratégico às redes ou sub-redes. Em algumas situações as organizações podem necessitar proteger partes da rede interna de outras partes da mesma rede corporativa. Nesse caso, pode-se utilizar *firewalls* internos configurados de forma apropriada à segurança interna.

Existem componentes básicos com os quais se pode construir uma infinidade de arquiteturas de *firewall*. Em redes de computadores, *firewalls* são barreiras interpostas entre a rede privada e a rede externa com a finalidade de reduzir os riscos de intrusos (ataques); ou seja, são mecanismos (dispositivos) de segurança que protegem os recursos de *hardware* e *software* da organização dos perigos (ameaças) aos quais o sistema está exposto.

Estes mecanismos de segurança são baseados em *hardware* e *software* e seguem a política de segurança estabelecida pela organização.

4.1 VISÃO GERAL

A rede é um ponto de entrada para seu aplicativo. Ela possui os primeiros *gateways* que controlam o acesso aos vários servidores do ambiente. Por exemplo, os servidores são protegidos pelos *gateways* do próprio sistema operacional, mas é importante não deixar que sejam inundados por ataques provenientes da camada de rede. É igualmente importante garantir que os *gateways* da rede não sejam substituídos nem reconfigurados por impostores. Em resumo, a segurança de rede

envolve a proteção dos dispositivos de rede e dos dados que eles encaminham.

Os componentes básicos de uma rede, que atuam como *gateways* de linha de frente, são o roteador, o *firewall* e o switch. A Figura 1 mostra esses componentes principais.

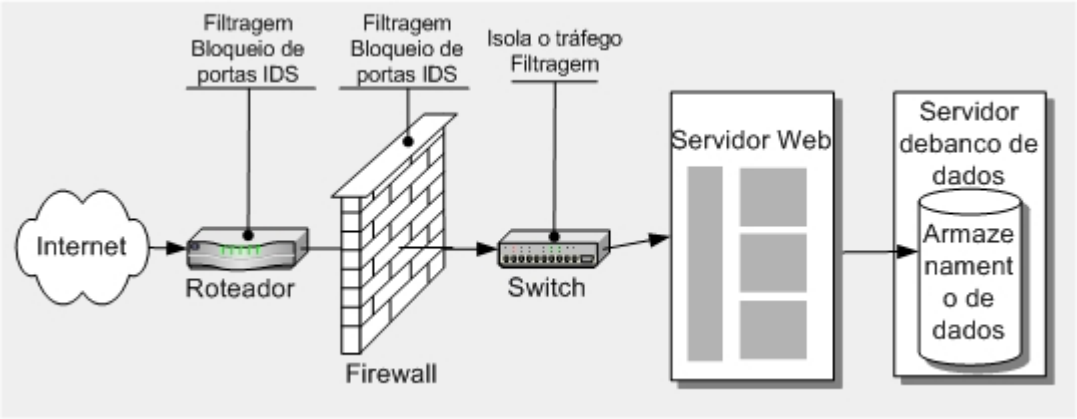

Figura 1 – Componentes de Rede: roteador, firewall e switch

4.2 METODOLOGIA

A segurança começa com a compreensão de como funciona o sistema ou a rede que precisa ser protegida. Este capítulo subdivide a segurança de rede por dispositivos, o que permite enfocar pontos

individuais de configuração. Em sintonia com a filosofia deste guia, este capítulo aborda a análise das ameaças potenciais. Sem tais análises, é impossível aplicar a segurança de forma adequada.

A infra–estrutura da rede pode ser subdividida nestas três camadas: acesso, distribuição e núcleo. Essas camadas contêm todo o hardware necessário para controlar o acesso aos recursos internos e externos e o acesso a partir deles. As recomendações aplicam–se a uma zona da Web que tem interface com a Internet ou com a intranet e, portanto, podem não se aplicar a sua rede interna ou corporativa.

Estes são os principais componentes de rede:

- Roteador
- *Firewall*
- Switch

4.2.1 ROTEADOR

O roteador é a entrada mais externa da segurança. Ele é responsável por encaminhar pacotes IP às redes a que ele se conecta. Esses pacotes podem ser solicitações recebidas de clientes da Internet para o servidor Web, respostas a solicitações ou solicitações enviadas de clientes internos. O roteador deve ser usado para bloquear o tráfego não autorizado ou indesejado entre redes. Não se esqueça de proteger o próprio roteador contra reconfiguração usando interfaces de administração seguras e confirmando se os *patches* e as atualizações de software mais recentes foram instalados.

4.2.2 FIREWALL

A função do *firewall* é bloquear todas as portas desnecessárias e só permitir o tráfego proveniente de portas conhecidas. O *firewall* deve ser capaz de monitorar as solicitações recebidas para evitar que ataques conhecidos atinjam o servidor Web. Associado à detecção de invasão, o *firewall* é uma ferramenta útil para evitar ataques e detectar tentativas de invasão, ou nos piores casos, detectar a origem de um ataque.

Como o roteador, o *firewall* é executado em um sistema operacional que precisa receber *atualização* regularmente. Suas interfaces de administração devem ser protegidas e os serviços não usados devem ser desativados ou removidos.

As atribuições de um *firewall* são:

- Checkpoint; ou seja, ele é um foco para as decisões referentes à segurança, é o ponto de conexão com o mundo externo, tudo o que chega à rede interna passa pelo firewall;
- Aplicar a política de segurança;
- Logar eficientemente as atividades da e para Internet;
- Limitar a exposição da empresa ao mundo externo; Proteger a rede contra vírus.

Tarefas que um *firewall* não pode realizar:

- Proteger a empresa contra usuários internos mal intencionados;
- Proteger a empresa de conexões que não passam por ele;

- Proteger contra ameaças completamente novas como vulnerabilidade e vírus.

4.2.3 SWITCH

O switch tem uma função mínima no ambiente de rede seguro. Os switches são projetados para melhorar o desempenho da rede a fim de facilitar a administração. Por essa razão, você pode facilmente configurar um switch enviando a ele pacotes formatados.

4.3 CONSIDERAÇÕES SOBRE ROTEADORES

O roteador é a primeiríssima linha de defesa. Sua função é o roteamento de pacotes, mas ele também pode ser configurado para bloquear ou filtrar o encaminhamento de tipos de pacotes que são conhecidos como vulneráveis ou que são usados de maneira mal–intencionada, como ICMP ou SNMP (*Simple Network Management Protocol*).

Se você não controlar o roteador, não haverá muito a ser feito para proteger a rede além de perguntar ao seu provedor que mecanismos de defesa ele usa nos roteadores dele.

As categorias de configuração do roteador são:

- Patches e atualizações
- Protocolos
- Acesso administrativo
- Serviços
- Auditoria e log
- Detecção de invasão

4.3.1 PATCHES E ATUALIZAÇÕES

Inscreva–se nos serviços de alerta fornecidos pelo fabricante do seu equipamento de rede para se manter atualizado sobre problemas de segurança e *patches* de serviço. À medida que as vulnerabilidades são descobertas, e, inevitavelmente elas são, os bons fornecedores disponibilizam *patches* rapidamente e anunciam essas atualizações por email ou em seus sites. Sempre teste as atualizações antes de implementá–las no ambiente de produção.

4.3.2 PROTOCOLOS

Ataques de negação de serviço geralmente aproveitam as vulnerabilidades dos protocolos, por exemplo, inundando a rede. Para combater esse tipo de ataque, você deve:

- Usar filtragem de entrada e saída.
- Fazer uma triagem do tráfego ICMP proveniente da rede interna.

4.3.3 USAR FILTRAGEM DE ENTRADA E SAÍDA

Os pacotes falsos são indicadores de sondagens, de ataques e de um invasor experiente. Pacotes recebidos com endereços internos podem indicar uma tentativa de invasão ou sondagem, e sua entrada deve ser impedida na rede de perímetro. Da mesma forma, configure o roteador para rotear os pacotes enviados somente se eles tiverem um endereço IP interno válido. Verificar os pacotes enviados não o protege contra ataques de negação de serviço, mas impede que esse tipo de ataque seja originado em sua rede.

Esse tipo de filtragem também permite que o originador seja rastreado facilmente até sua verdadeira origem, pois o atacante teria que usar um endereço de origem válido e legitimamente alcancável.

4.3.4 FAZER A TRIAGEM DO TRÁFEGO ICMP DA REDE INTERNA

O ICMP é um protocolo independente que fica por cima do IP e permite que as informações de disponibilidade do host sejam verificadas entre *hosts*. As mensagens ICMP comumente usadas são mostradas na Tabela 1.

Mensagem	Descrição
Solicitação de eco	Determina se o nó IP (um *host* ou roteador) está disponível na rede.
Resposta de eco	Responde à solicitação de eco do ICMP.
Destino de envio inacessível	Informa ao host que não é possível entregar um datagrama.
Retardamentos de origem	Informa ao *host* que ele deve reduzir a taxa de envio de datagramas devido a um congestionamento.
Redirecionar	Informa ao host uma rota preferencial.
Tempo excedido	Indica que a vida útil de um datagrama IP expirou.

Tabela 1 - Mensagens ICMP comumente usadas

O bloqueio do tráfego ICMP no roteador de perímetro externo protege contra ataques como inundações de ping em cascata. Existem outras vulnerabilidades do ICMP que justificam o bloqueio desse protocolo. Embora o ICMP possa ser usado para a solução de

problemas, ele também pode ser utilizado na descoberta e no mapeamento de rede. Portanto, controle o uso do ICMP. Se for necessário ativá–lo, use–o apenas no modo eco–resposta.

4.3.5 EVITE MENSAGENS DE VIDA ÚTIL EXPIRADA COM VALORES DE 1 OU 0

O roteamento com rastreamento usa valores de vida útil de 1 e 0 para contar saltos entre um cliente e um servidor. O roteamento com rastreamento é um meio de coletar informações sobre a topologia da rede. Com o bloqueio de pacotes desse tipo, você evita que um invasor descubra detalhes sobre sua rede por meio de rotas de rastreamento.

4.3.6 NÃO RECEBA NEM ENCAMINHE O TRÁFEGO DE DIFUSÃO DIRECIONADO

O tráfego de difusão direcionado pode ser usado para enumerar os *hosts* de uma rede e como veículo para um ataque de negação de serviço. Por exemplo, ao bloquear endereços de origem específicos, você impede que solicitações de eco mal–intencionadas gerem fluxos de ping em cascata. Os endereços de origem devem ser filtrados como mostra a Tabela 2.

Endereço de origem	Descrição
0.0.0.0/8	Difusão histórica
10.0.0.0/8	Rede privada RFC 1918
127.0.0.0/8	Auto–retorno
169.254.0.0/16	Link para redes locais
172.16.0.0/12	Rede privada RFC 1918
192.0.2.0/24	TEST–NET

Endereço de origem	Descrição
192.168.0.0/16	Rede privada RFC 1918
224.0.0.0/4	Multicast classe D
240.0.0.0/5	Class E reserved
248.0.0.0/5	Não–alocado
255.255.255.255/32	Difusão

Tabela 2 - Endereços de origem que devem ser filtrados

4.3.7 ACESSO ADMINISTRATIVO

De onde o roteador será acessado com fins administrativos? Decida por que interfaces e portas a conexão administrativa será permitida e a partir de que rede ou *host* a administração será feita. Restrinja o acesso a esses locais específicos. Não deixe a interface de administração com a Internet disponível sem criptografia e contramedidas para evitar seqüestro. Além disso:

- Desative as interfaces não usadas.
- Aplique diretivas rígidas de senhas.
- Use roteamento estático.
- Faça auditoria nas interfaces de administração com a Web.

4.3.8 DESATIVE AS INTERFACES NÃO USADAS.

Somente as interfaces necessárias devem estar ativadas no roteador. Uma interface que não é usada também não é monitorada nem controlada e, provavelmente, não é atualizada. Isso pode expô–lo a ataques desconhecidos contra essas interfaces.

4.3.9 APLIQUE DIRETIVAS RÍGIDAS DE SENHAS

Software de senha de força bruta pode iniciar mais do que apenas ataques de dicionário. Pode descobrir senhas comuns em que uma letra é substituída por um número. Por exemplo, se "s3nh4" for usado como senha, ela pode ser violada. Sempre use combinações de letras maiúsculas, minúsculas, números e símbolos ao criar senhas.

4.3.10 USE ROTEAMENTO ESTÁTICO

O roteamento estático evita especialmente que pacotes formados alterem as tabelas de roteamentos do roteador. Um invasor poderá tentar alterar as rotas para causar negação de serviço ou para encaminhar solicitações a um servidor invasor. Por meio do uso de rotas estáticas, a interface administrativa deve primeiro fazer alterações no roteamento.

4.3.11 FAÇA AUDITORIA NAS INTERFACES DE ADMINISTRAÇÃO COM A WEB

Determine também se o acesso interno pode ser configurado. Quando possível, encerre a interface de administração interna e use métodos de acesso interno com as ACLs.

4.3.12 SERVIÇOS

Em um roteador implantado, cada porta aberta é associada a um serviço de escuta. Para reduzir a área de ataque, os serviços padrão que não forem necessários deverão ser encerrados. Exemplos: **bootps** e **Finger**, que raramente são necessários. Também verifique o roteador para detectar que portas estão abertas.

4.3.13 AUDITORIA E LOG

Por padrão, um roteador registra todas as ações de negação. Esse comportamento padrão não deve ser alterado. Além disso, proteja os arquivos de *log* em um local central. Os roteadores modernos têm uma matriz de recursos de *log* que inclui a capacidade de definir severidades com base nos dados registrados. Uma agenda de auditoria deve ser estabelecida para inspecionar periodicamente os *logs* em busca de sinais de invasão ou sondagem.

4.3.14 DETECÇÃO DE INTRUSÃO

Estabelecidas as restrições no roteador para evitar ataques de TCP/IP, o roteador deve ser capaz de identificar quando está ocorrendo um ataque e notificar o administrador do sistema.

Os invasores sabem onde estão as prioridades de segurança e tentam contorná–las. Os IDSs (Sistemas de Detecção de Intrusão) podem mostrar onde o invasor está tentando atacar.

4.4 CONSIDERAÇÕES SOBRE FIREWALLS

Um *firewall* deve existir em todos os pontos de interação com uma rede não confiável, especialmente a Internet. Também é recomendável separar os servidores Web, os servidores de aplicativos e de bancos de dados *downstream* com um *firewall* interno.

Depois do roteador, com seus filtros amplos e *gatekeepers*, o *firewall* é o próximo ponto de ataque. Em muitos casos (se não na

maioria), você não tem acesso administrativo ao roteador *upstream*. Muitos filtros e ACLs que se aplicam ao roteador podem também ser implementados no *firewall*. As categorias de configuração do *firewall* são:

- *Patches* e atualizações
- Filtros
- Registro e auditoria
- Redes de perímetro

4.4.1 PATCHES E ATUALIZAÇÕES

Inscreva–se nos serviços de alerta fornecidos pelo fabricante do seu *firewall* e sistema operacional para se manter atualizado sobre problemas de segurança e *patches* de serviço.

4.4.2 FILTROS

A filtragem de portas publicadas em um *firewall* pode ser um método eficaz de bloquear pacotes e cargas mal–intencionados. Os filtros variam de simples filtros de pacotes que restringem o tráfego na camada de rede com base nos endereços IP e números de portas de origem e destino a filtros complexos que inspecionam cargas específicas de aplicativos. Uma defesa profunda que use filtros em camadas é uma maneira eficaz de bloquear ataques. Há seis tipos comuns de filtros de *firewall*.

Quando você usa filtros em vários níveis na pilha da rede, isso ajuda a tornar o ambiente mais seguro. Por exemplo, um filtro de

pacote pode ser usado para bloquear tráfego IP destinado a qualquer porta que não seja a porta 80 e o filtro de aplicativo pode restringir ainda mais o tráfego com base na natureza do verbo HTTP. Por exemplo, poderiam ser bloqueados os verbos HTTP DELETE.

4.4.3 FILTROS DE PACOTE

Esses podem filtrar pacotes com base no protocolo, número de porta de origem e destino e endereço de origem e destino ou nome do computador. Os filtros de pacotes IP são estáticos e a comunicação por uma porta específica é permitida ou bloqueada. Pacotes bloqueados geralmente são registrados e um filtro de pacote seguro nega acesso por padrão.

No nível da camada de rede, a carga é desconhecida e pode ser perigosa. Tipos de filtragem mais inteligentes devem ser configurados para inspecionar a carga e tomar decisões baseadas em regras de controle de acesso.

4.4.4 FILTROS NO NÍVEL DO CIRCUITO

Eles inspecionam sessões em vez de dados de carga. Um cliente de entrada ou saída faz uma solicitação diretamente ao *firewall*/*gateway*, e o *gateway*, por sua vez, inicia uma conexão com o servidor e atua como um intermediário entre as duas conexões.

Conhecendo as regras de conexão de aplicativo, os filtros no nível do circuito garantem interações válidas. Eles não inspecionam a carga real, mas contam quadros para garantir a integridade dos pacotes e evitar seqüestro de sessões ou ataques por repetição.

4.4.5 FILTROS DE APLICATIVO

Os filtros de aplicativo inteligentes permitem analisar um fluxo de dados de um aplicativo e oferecem processamento específico do aplicativo, incluindo inspeção, triagem ou bloqueio, redirecionamento e até mesmo modificação de dados, à medida que eles passam pelo *firewall*. Os filtros de aplicativo protegem contra ataques como os seguintes:

- Comandos SMTP desprotegidos.
- Ataques contra servidores DNS internos.
- Ataques baseados em HTTP (por exemplo, Code Red e Nimda, que usam conhecimentos específicos do aplicativo).

Por exemplo, um filtro de aplicativo pode bloquear um HTTP DELETE, mas permitir um HTTP GET. Os recursos de triagem de conteúdo, incluindo detecção de vírus, análise lexical e categorização de sites, tornam os filtros de aplicativo muito eficazes em cenários da Web, tanto como medidas de segurança quanto na aplicação de regras de negócios.

4.4.6 INSPEÇÃO COM INFORMAÇÕES DE ESTADO

Os filtros de aplicativo se limitam ao conhecimento da carga de um pacote e, portanto, tomam decisões de filtragem com base apenas na carga. A inspeção com informações de estado usa tanto a carga quanto seu conteúdo para determinar regras de filtragem.

Usando a carga e o conteúdo do pacote, as regras de inspeção com informações de estado garantem a integridade das sessões e da comunicação. A inspeção dos pacotes, de sua carga e seqüência, limita a escalabilidade da inspeção com informações de estado.

4.4.7 FILTROS DE APLICATIVO PERSONALIZADOS

Esses filtros garantem a integridade da comunicação servidor/cliente do aplicativo.

4.4.8 LOG E AUDITORIA

O log de todas as solicitações recebidas e enviadas, independentemente das regras de *firewall*, permitem detectar ataques invasores ou, até pior, ataques bem–sucedidos que não haviam sido detectados anteriormente. Historicamente, os administradores de rede, às vezes, tinham de analisar os logs de auditoria para determinar como um ataque havia tido êxito. Naqueles casos, os administradores não podiam aplicar soluções para vulnerabilidades, saber como elas haviam sido comprometidas nem descobrir outras vulnerabilidades que existiam.

Aplique as seguintes diretivas para criar log e auditar logs.

- Registre todo o tráfego que passa pelo *firewall*.
- Mantenha um ciclo de log saudável que permita a análise rápida dos dados. Quanto mais dados você tiver, maior será o arquivo de log.

- Certifique–se de que o relógio do *firewall* esteja sincronizado com o resto do hardware da rede.

4.5 CONSIDERAÇÕES SOBRE SWITCHES

Um switch é responsável por encaminhar pacotes diretamente a um *host* ou segmento de rede, em vez de compartilhar os dados com toda a rede. Portanto, o tráfego não é compartilhado entre segmentos comutados. Essa é uma medida preventiva contra a invasão por *sniffer* de pacotes entre as redes. Um invasor pode contornar essa medida de segurança reconfigurando as regras de comutação com o uso de interfaces administrativas de fácil acesso, incluindo nomes de contas e senhas conhecidos nos pacotes SNMP.

As categorias de configuração a seguir são usadas para garantir uma configuração de switch segura:

- Patches e atualizações
- VLANs (Redes Locais Virtuais)
- Padrões desprotegidos
- Serviços
- Criptografia

4.5.1 PATCHES E ATUALIZAÇÕES

Patches e atualizações devem ser testados e instalados assim que estiverem disponíveis.

4.5.2 VLANS

As VLANs permitem que você separe segmentos de rede e aplique o controle de acesso de acordo com regras de segurança. Entretanto, uma VLAN aumenta o desempenho da rede, mas não proporciona, necessariamente, segurança. Limite o uso de VLANs à rede de perímetro (atrás do firewall) já que existem muitas interfaces desprotegidas para facilitar a administração.

4.5.3 PADRÕES DESPROTEGIDOS

Para ter certeza de que padrões desprotegidos fiquem seguros, altere todas as senhas padrão e seqüências de caracteres de comunidade SNMP para evitar a enumeração da rede ou o controle total do switch. Além disso, investigue e identifique contas possivelmente não documentadas e altere os nomes e senhas padrão. Esses tipos de contas costumam ser encontrados em tipos de switches bem conhecidos e são muito divulgados e conhecidos por invasores.

4.5.4 SERVIÇOS

Certifique–se de que todos os serviços não utilizados estejam desativados. Certifique–se também de que o TFTP esteja desativado, que os pontos de administração na interface com a Internet foram removidos e que as ACLs foram configuradas para limitar o acesso administrativo.

4.5.5 CRIPTOGRAFIA

Embora não seja tradicionalmente implementada no switch, a criptografia de dados por cabo garante que os pacotes submetidos à invasão por *sniffer* sejam inúteis quando um monitor é colocado no mesmo segmento comutado ou quando o switch é comprometido permitindo a invasão por *sniffer* de segmentos.

4.5.6 CONSIDERAÇÕES ADICIONAIS

As seguintes considerações podem aumentar ainda mais a segurança das redes:

- Certifique–se de que os relógios de todos os dispositivos da rede estejam sincronizados. Consulte a hora da rede e sincronize todas as fontes com uma fonte de hora conhecida e confiável.
- Use autenticação TACACS (Terminal Access Controller Access Control System) ou RADIUS (Remote Authentication Dial–In User Service) para ter ambientes altamente seguros como meio de limitar o acesso administrativo à rede.
- Defina uma rede IP que possa facilmente ser protegida com o uso de ACLs em sub–redes ou limites da rede, sempre que possível.

4.5.7 MODELO DE UMA REDE SEGURA

A Tabela 3 apresenta um instantâneo das características de uma rede segura. As configurações de segurança foram resumidas a partir de informações de especialistas em segurança do setor e de aplicativos em ambientes seguros do mundo real. Este instantâneo pode ser usado como referência para avaliar sua solução.

Componente	**Característica**
Roteador	
Patches e atualizações	O sistema operacional do roteador está corrigido com software atualizado.
Protocolos	Os protocolos e as portas não usados estão bloqueados. A filtragem de entrada e saída foi implementada. O tráfego ICMP é triado a partir da rede interna. As mensagens de vida útil expirada com valores 1 ou 0 são bloqueadas (o rastreamento de rota está desabilitado). O tráfego de difusão direcionado não é encaminhado. Pacotes de ping grandes são triados. Pacotes RIP, se usados, são bloqueados no roteador mais externo.

Componente	Característica
Acesso administrativo	As interfaces de gerenciamento do roteador não usadas foram desativadas.
	É aplicada uma diretiva rigorosa de senha administrativa. É usado roteamento estático. A interface de administração com a Web está desativada.
Serviços	Os serviços não usados estão desativados (por exemplo, **bootps** e **Finger**).
Auditoria e log	O log está ativado para todo o tráfego negado.
	Os logs são armazenados de maneira centralizada e segura. É feita uma auditoria nos logs em busca de padrões incomuns.
Detecção de intrusão	Existe um IDS para identificar e notificar sobre um ataque ativo.
Firewall	
Patches e atualizações	O software e o sistema operacional do *firewall* receberam os *patches* e as atualizações de segurança mais recentes.
Filtros	A diretiva de filtragem de pacote bloqueia todo o tráfego, exceto o necessário, em ambas as direções. Filtros específicos de aplicativos estão implantados para limitar o tráfego desnecessário.
Log e auditoria	Todo o tráfego permitido é registrado.

Componente	Característica
	Tráfego negado é registrado. O ciclo dos dados é feita com uma freqüência tal que permite a rápida análise dos dados. Todos os dispositivos da rede estão sincronizados com uma fonte de tempo comum.
Redes de perímetro	Existe uma rede de perímetro se várias redes precisam acessar o servidor.
	Existe um *firewall* entre as redes não confiáveis.
Switch	
Patches e atualizações	Os *patches* de segurança mais recentes foram testados e instalados ou a ameaça de vulnerabilidades conhecidas foi atenuada.
VLANs	Certifique–se de que as VLANs não são usadas em excesso ou se gozam de excesso de confiança.
Padrões desprotegidos	Todas as senhas de fábrica foram alteradas. Está disponível o mínimo de interfaces administrativas. Os controles de acesso estão configurados para proteger as seqüências de caracteres de comunidade SNMP.
Serviços	Os serviços não usados estão desativados.
Criptografia	O tráfego comutado é criptografado.
Outros	
Sincronização	Os relógios de todos os dispositivos com

Componente	Característica
do log	recurso de log estão sincronizados.
Acesso administrativo à rede	TACACS ou RADIUS é usado para autenticar usuários administrativos.
ACLs da rede	A rede está estruturada de modo que as ACLs possam ser usadas em *hosts* e redes.

Tabela 3: Instantâneo de uma rede segura

5 PROJETO PARA A REDE DA BRASILTELECOM

Tendo em vista a grande necessidade apresentada pelos profissionais da Brasiltelecom e pela própria empresa visando minimizar os gastos com viagens de terceiros para dar manutenção nos equipamentos é que apresentamos a proposta de disponibilizar acesso seguro a qualquer equipamento inserido dentro do *backbone* da brasiltelecom a partir de qualquer ponto da internet.

Devemos ter o cuidado de preservar a segurança da rede e faremos isso levando em consideração os parâmetros de segurança apresentados nos capítulos anteriores.

Os endereços de IP e nomenclatura dos equipamentos utilizados serão mascarados, mas tentaremos mostrar como é a topologia e a configuração dos equipamentos e máquinas da forma mais clara possível.

Abaixo a figura 2 apresenta o *backbone* da BrasilTelecom, onde estão os equipamentos que deverão ser acessados remotamente.

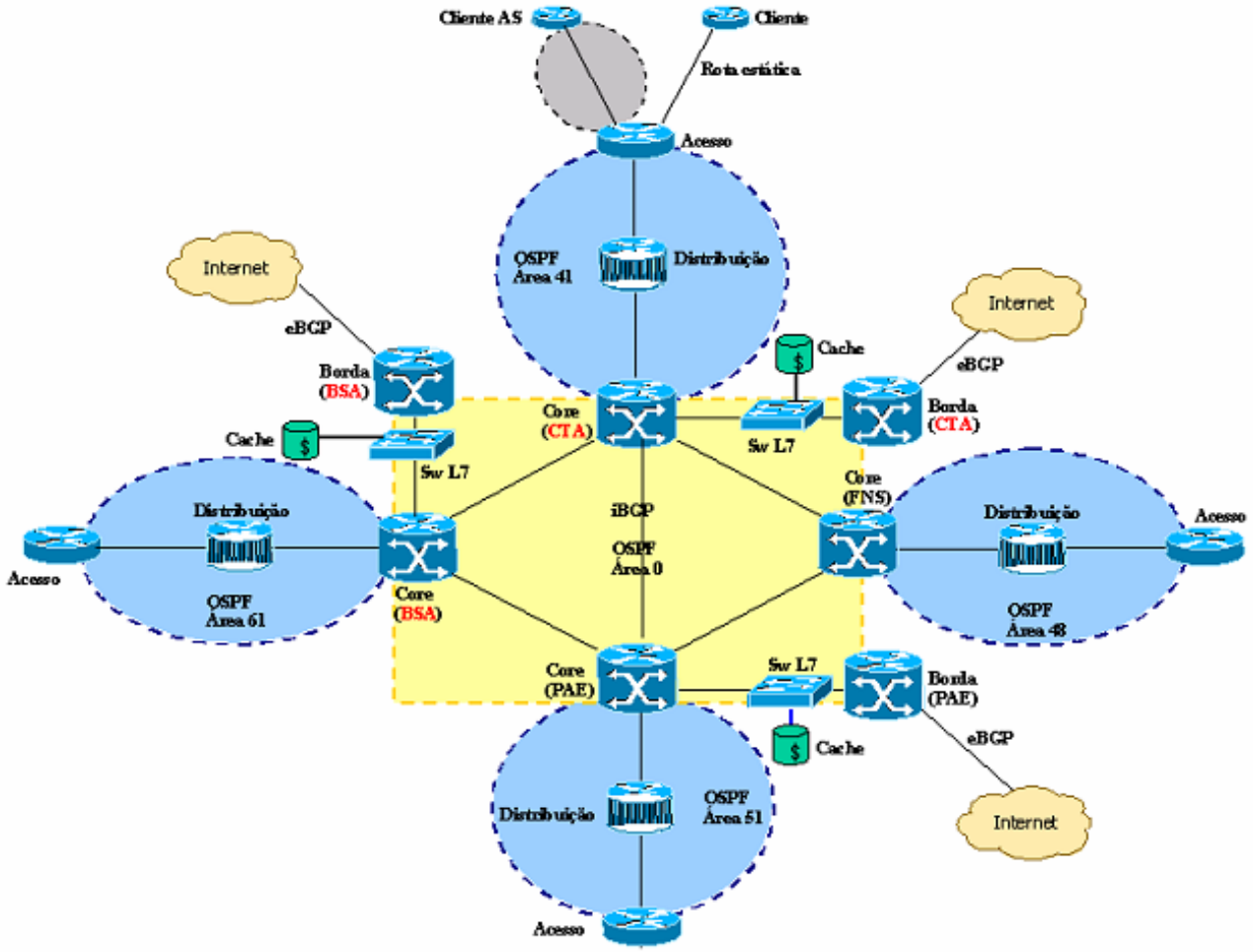

Figura 2 – Equipamentos do Backbone da BrasilTelecom

5.1 CONFIGURAÇÃO DA REDE ANTES DA IMPLEMENTAÇÃO DO PROJETO

No princípio o acesso aos equipamentos do *Backbone* da Brasiltelecom, só era possível à partir de máquinas que estavam dentro

da rede coorporativa do Centro Nacional de Redes e Serviços(CNRS) e das Filiais.

Segue abaixo a topologia da rede com os servidores, antes da implementação da máquina *Gateway*:

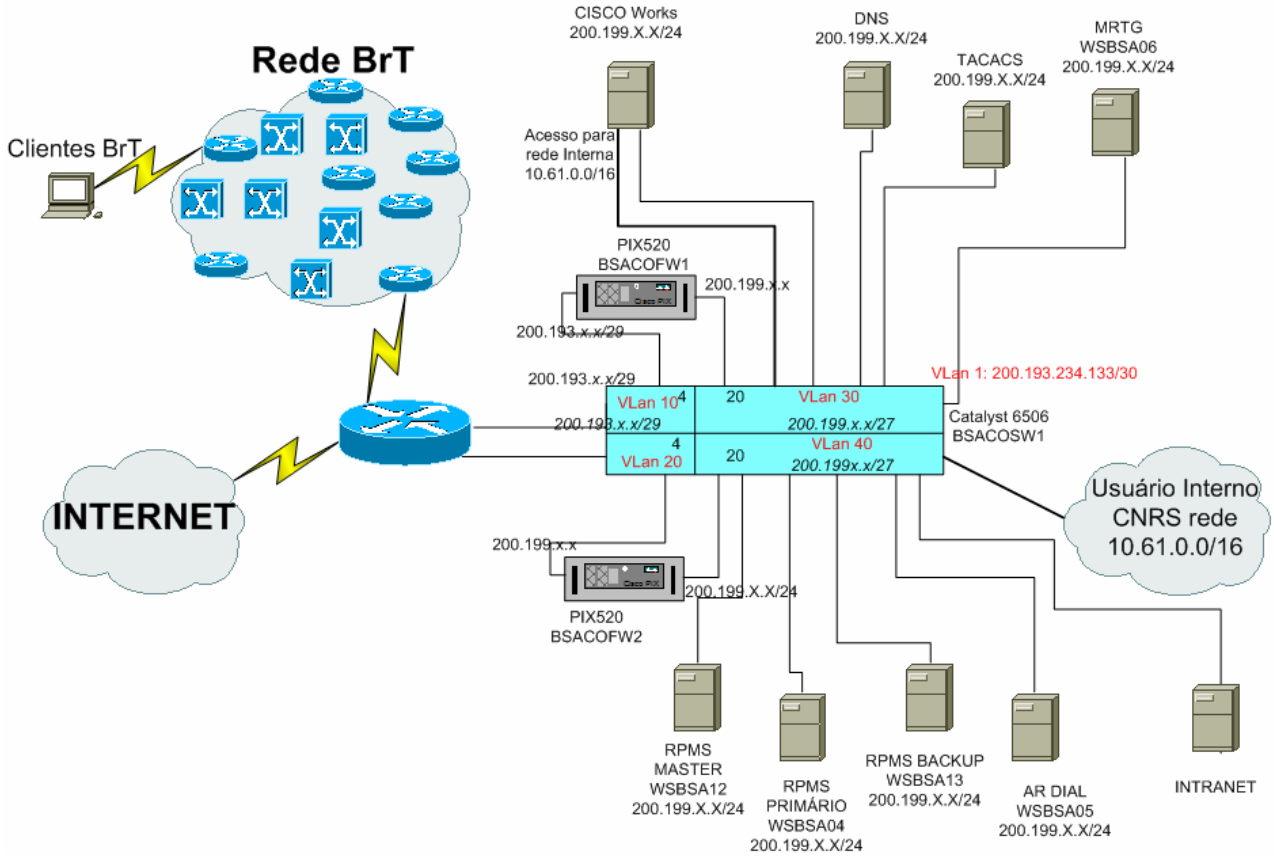

Figura 3 – Topologia da rede antes

A rede de servidores internos era composta por várias máquinas conectadas a um switch Cisco Catalyst 6506(BSACOSW1), onde os segmentos eram separados por Vlans.

Os equipamentos do *backbone* eram acessados através da máquina WSBS01, onde os usuários da rede coorporativa faziam uma conexão via telnet para esta máquina.

A máquina WSBSA01 utilizava o sistema operacional Windows NT e rodava o servidor de telnet. Não existia nenhuma política de gerenciamento de usuários e nenhuma diferenciação de acesso entre os usuários das filiais e os usuários do Centro de Gerência.

Um dos problemas enfrentados era que qualquer manobra na rede, mesmo que muito emergencial, deveria ser feita por pessoas que estivessem dentro do espaço físico da BrasilTelecom, em muitos casos os profissionais precisavam se deslocar de casa até o prédio da BrasilTelecom para poder atuar sobre eventuais problemas.

Outro problema bastante comum era o acesso de parceiros como, Cisco, NEC e Siemens, responsáveis pela manutenção de alguns equipamentos na rede da BrasilTelecom, na maioria das vezes estes parceiros estão em cidades fora da abrangência da BrasilTelecom, o que impossibilita que seus profissionais atuem em problemas ou executem manutenções a apartir de suas próprias cidades, isto gera despesa com viagens e hotéis.

Visando solucionar estes problemas e implantar uma política de segurança eficaz no nível que uma empresa de grande porte como a BrasilTelecom precisa foi que apresentamos este projeto de implementação de uma máquina *gateway* capaz de fornecer acesso à rede à partir da internet e gerenciar os usuários, internos e externos, que acessam esta rede, bem como monitorar todas as ações gerando logs de comandos.

O projeto da máquina *gateway* consiste em configurar uma máquina que servirá como portal de acesso para todos os equipamentos

do *backbone* da BrasilTelecom, para isso será necessário implementar algumas alterações de topologia da rede de acesso.

5.2 INFORMAÇÕES GERAIS

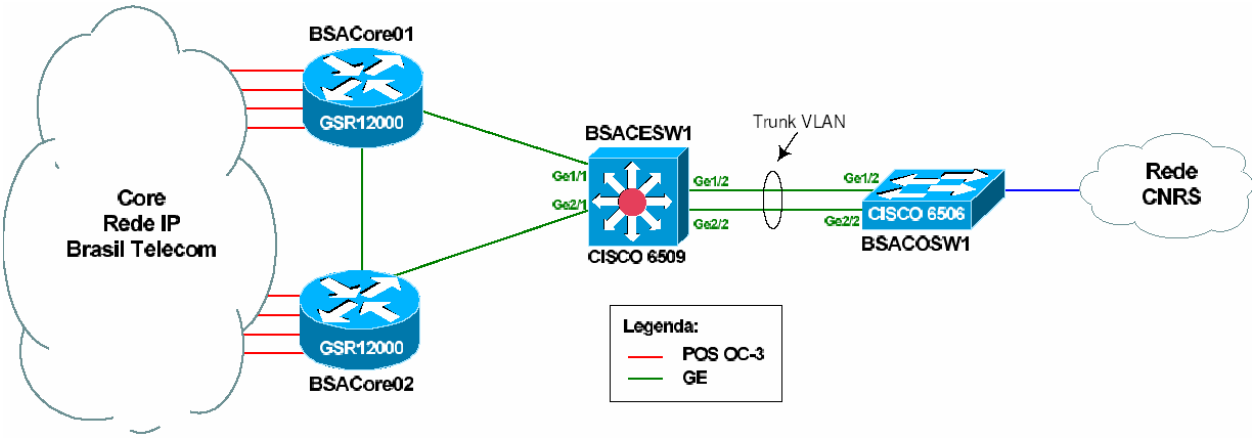

Figura 4 – Topologia da rede

A Figura 4 apresenta o esboço da conexão entre o SWITCH ROUTER 6509 da estação centro (BSACESW1) e o SWITCH ROUTER 6506 da estação norte (BSACOSW1), onde são usadas 2 fibras monomodo para interligação dos equipamentos, fazendo um TRUNK de VLAN's usando o protocolo ISL proprietário da CISCO.

A Rede do CNRS, esboçada acima como uma nuvem, é formada por 1 VLAN:

- VLAN30 - Rede 200.199.x.0/24

Cada um destes segmentos é roteado por um PIX 520 da CISCO, operando com redundância. Na VLAN30 o PIX BSACOFW1 protege a rede 200.199.x.0/24. Através do BSACOFW1 é feito o acesso à BrtNet, cujo endereço IP deverá ser informado pelo CNRS.

O Trunk VLAN entre o BSACESW1 e o BSACOSW1 deverá ser modificado para o protocolo IEEE802.1Q.

Todas as máquinas da Rede IP e demais redes que sejam acessadas via Telnet (access list 1) e gerenciados via SNMP (access list 8), deverão possuir as *access-list* abaixo:

access-list 1 permit 200.199.x.0 0.0.0.255
access-list 8 permit 200.199.x.0 0.0.0.255

5.2.1 ORIENTAÇÕES PARA EXECUÇÃO

A seguir serão definidos os principais pontos a serem seguidos no processo. Algumas alterações são solicitadas neste documento com o intuito de padronizar a configuração, bem como desabilitar algumas configurações não necessárias e não utilizadas pelos equipamentos.

Atividades a serem executadas:

- A descrição do ***vtp domain*** do BSACOSW1 e BSACESW1.

- As descrições das interfaces Giga Ethernet Ge1/2 e Ge2/2 de BSACESW1 para:
    - Ge1/2 - Description: BSACOSW1 6506 Ge1/2
    - Ge2/2 - Description: BSACOSW1 6506 Ge2/2

- O protocolo utilizado no Trunk VLAN deverá ser IEEE 802.1Q nos SWITCHES BSACESW1 (G1/2 e Ge2/2) e BSACOSW1 (G1/2 e Ge2/2), para isso utilizar o comando:
    - *Set trunk [módulo/porta] on dot1q [VLAN's a serem transportadas no trunk]*

- Em BSACOSW1, configurar o vtp mode para o modo transparente:
    - Set vtp mode transparent

5.3 CONFIGURAÇÃO DA REDE APÓS A IMPLEMENTAÇÃO DO PROJETO

A rede da Brasiltelecom é composta por várias máquinas que desempenham funções específicas, tais como:

- Monitoração de alarmes na rede;
- Monitoração de números de usuários conectados pelo serviço Dialnet;
- Autenticação de usuários Dialnet e ADSL;
- Geração de gráficos de desempenho do *Backbone*;

A máquina *gateway* será inserida em meio à estas máquinas e ficará atrás de uma Vlan que está configurada em um Switch, que segmenta os equipamentos do *Backbone* dos equipamentos de gerência.

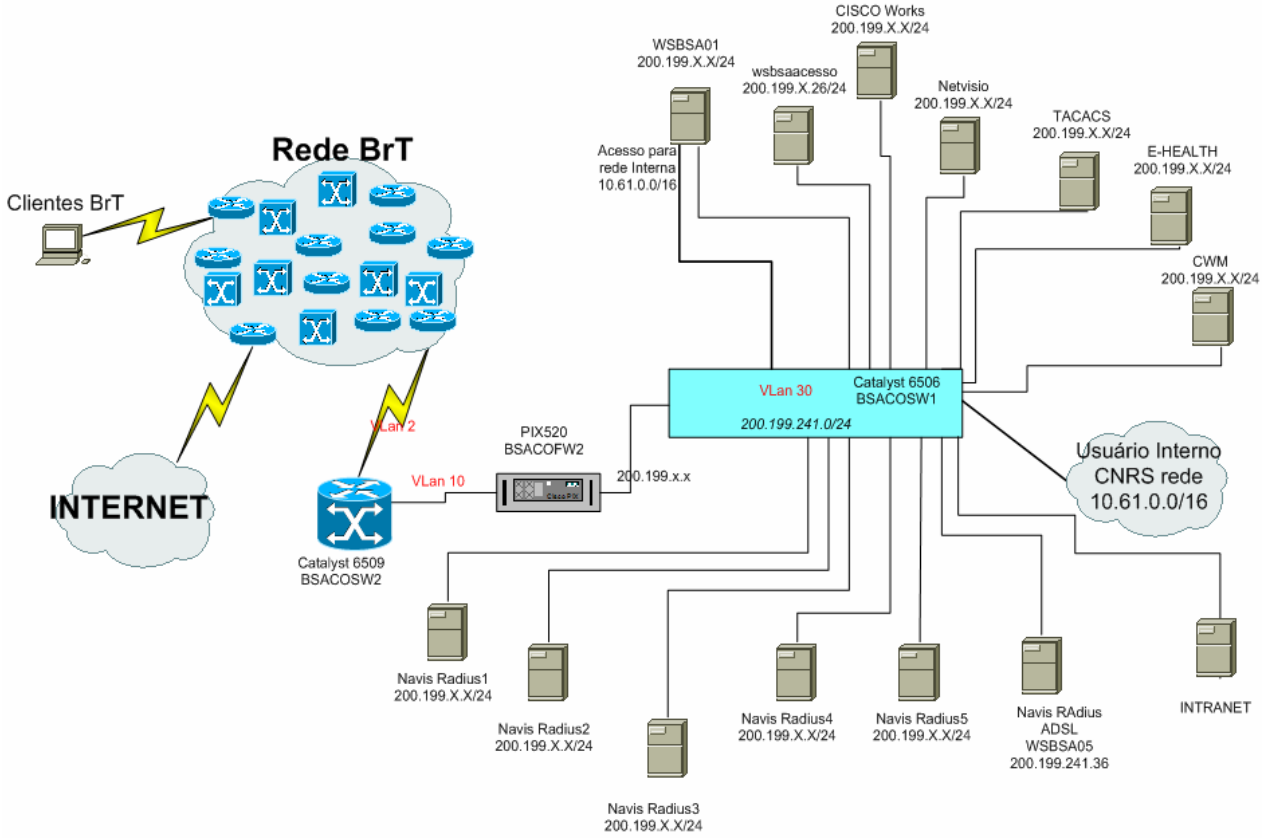

Figura 5 – Topologia atual da rede

A figura 5 apresenta a topologia atual da rede acesso bem como a representação dos equipamentos que serão acessados por meio da máquina *gateway*, a qual foi chamada de wsbsaacesso,

Não será apresentado um detalhamento maior sobre a configuração do switch e do PIX *Firewall*, pois o objetivo principal é a configuração do servidor, que será apresentada a seguir.

6 CONFIGURAÇÃO DA MÁQUINA GATEWAY

Esta máquina estará ligada ao PIX 520, utilizando um IP da rede 200.199.241.0/24. Utilizaremos um hardware de PC com alta capacidade memória e processamento.

Escolhemos o debian versão 3.0 como sistema operacional, para este compilamos o kernel 2.4.27, o qual conseguimos fazendo um *download* no site www.kernel.org/pub/linux/kernel/v2.4.

O kernel foi compilado com uma configuração mais simples possível, utilizando apenas as funções básicas. A máquina está configurada para escutar apenas a porta 22 e a porta 2222 destinada apenas para alguns IP's específicos, utilizados apenas por terceiros para acessar a rede da Brasiltelecom.

Foi configurado também o IPTABLES para fazer a segurança local da máquina, a configuração deste será descrita adiante.

Em seguida foram instalados os pacotes necessários para algumas aplicações.

Ferramentas para Scripts:

- GCC
- Perl

Ferramentas para gerenciamento de usuários:

- Jail
- PAN

Algumas ferramentas para *troubleshooting* de problemas de rede:

- Traceroute

- NAMP

O processo de HARDENING foi feito seguindo as orientações propostas no site do Sistema operacional Debian: www.debian.org/doc/manuals/securing-debian-howto.

Este processo consiste em fazer uma configuração segura e estável para a máquina em questão proporcionando maior clareza em torno do parâmetros configurados para cada aplicação.

A segurança deste sistema não difere muito da segurança de outros sistemas operacionais. Contudo a importância de se ter uma seqüência de idéias a serem seguidas é muito grande, pois elas nos direcionam a utilizar apenas o necessário do sistema, sem sobrecarregar a máquina.

6.1 PASSO 1: O QUE QUEREMOS DO SISTEMA?

A primeira decisão que devemos tomar é escolher o que queremos do nosso sistema, quais os serviços que serão realmente necessários. Partindo deste princípio, para este sistema, compilamos uma versão de Kernel 2.4.27, que era a mais recente desabilitando todas as funções que consideramos desnecessárias em termos práticos, foi compilado um kernel enxuto com uma versão estável.

Apenas as portas 22 e 2222 do protocolo tcp foram liberadas, sendo esta última liberada apenas para algumas máquinas de fabricantes que acessariam nossa rede de pontos externos à Brasiltelecom.

Em seguida foram configurados os usuários e as permissões que cada um teria sobre o sistema. Para tanto utilizamos a ferramenta Jail para administração das contas do usuários.

Foram disponibilizados apenas o telnet, que é a principal função usada pelos usuários, e o ftp para aqueles eventualmente necessitem baixar algum arquivo de backup de configuração para a máquina, também foi configurado quota para os usuários prevenindo a sobrecarga do disco rígido.

Harden ou a rigidez nas regras criadas minimizam o impacto de um evento que por ventura venha a parar um serviço. O uso apropriado das ferramentas garantem que usuários não autorizados sejam detectados rapidamente e que as ações cabíveis sejam tomadas de forma a impedir um estrago na segurança do sistema.

O administrador do sistema deve estar sempre atualizado sobre os problemas de segurança, é importante estar sempre lendo alguns documentos sobre segurança, pois o Debian GNU/Linux é baseado no Kernel do Linux, assim muitas informações a respeito de Linux, assim como de outras distribuições e da segurança geral de UNIX lhe aplique também (mesmo se as ferramentas usadas, ou os programas disponíveis, diferem).

6.2 PASSO 2: ANTES E DURANTE A INSTALAÇÃO

6.2.1 ESCOLHER UMA SENHA PARA A BIOS

Antes de instalar qualquer sistema operacional em seu computador, sete uma senha para a BIOS. Após a instalação você deve voltar à configuração da BIOS e alterar a sequência de Boot para

desabilitar o Boot pelo Floppy, cdrom ou outro dispositivo que não seja o disco rígido. Senão um *cracker* necessita somente do acesso físico e um disco de boot para alcançar seu sistema.

Desabilitando a inicialização a menos que uma senha seja fornecida é melhor, isto pode ser muito eficaz se você rodar em um servidor, porque eles não reiniciam muito freqüentemente. O outro lado desta tática é que a reinicialização do sistema necessita de uma intervenção humana o que pode ser um problema se a máquina não for facilmente acessível.

6.2.2 PARTICIONANDO O SISTEMA

Um esquema inteligente de particionamento depende de como a máquina é utilizada. Uma boa regra de *thumb* é ser razoavelmente liberal com as partições e atentar aos seguintes fatores:

- Toda árvore de diretórios que o usuário tiver permissões de escrita, como por exemplo: /home /tmp e /var/tmp/, deve estar em uma partição separada. Isto reduz o risco de um DoS de usuário lotar o "/" e deixar o sistema inutilizável.
- Toda partição que puder ser flutuante, por exemplo: /var (especialemente /var/log) deve também estar em uma partição separada.
- Toda partição onde serão instalados softwares que não são da distribuição estará seperada das demais. De acordo com o padrão de hierarquia de arquivos, isto é /opt ou /user/local. Se estas forem partições

separadas, não serão apagadas se o Debian for reinstalado.

6.2.3 SELECIONANDO UM SISTEMA DE ARQUIVOS APROPRIADO

Durante o particionamento do sistema é preciso escolher o sistema de arquivos que será usado. O sistema de arquivos default selecionado na instalação do Debian para partições Linux é o *ext2*. Contudo recomenda-se altera-lo paro um sistema de arquivos mais recente, como: *ext3, reiserfs, jfs* ou *xfs.* O sistema escolhido em nosso caso foi o *ext3,* devido ao fato de que em caso de crash no sistema ele levará menos tempo para se reestabelcer e checar o sistema de arquivos e menos dados serão perdidos.

6.2.4 NÃO PLUG À INTERNET ATÉ QUE ESTEJA TUDO PRONTO

O sistema não deve ser imediatamente plugado à internet durante a instalação. Se os serviços não estiverem corretamente configurados isto abrirá uma porta para um ataque.

Alguns serviços geralmente têm vulnerabiliades de segurança não repados nos pacotes que se está usando para a instalação.

No nosso caso todos os pacotes foram atualizados após a instalção.

6.2.5 SETANDO A SENHA DE ROOT

Setar uma boa senha para root é a exigência mais básica para se ter um sistema seguro. Para isto existem vários métodos, você pode utilizar programas que geram as senhas automaticamente. Em nosso caso utilizamos um padrão fornecido pela empresa.

6.2.6 ATIVE SENHAS DE SHADOW E SENHAS DE MD5

No fianal da instalação você será perguntado se a senha de Shadow deverá ser abilitada. Em nosso caso optamos por abilita-la, estas senhas ficarão no arquivo */etc/shadow.* Apenas o usuário root e o grupo shadow terão acesso de leitura à este arquivo, assim nenhum outro usuário poderá copiar este arquivo para poder rodar o cracker nele. É possível comutar entre senhas de Shadow e senhas normais a qualquer momento utilizando o shadowconfig.

Além disso, durante a instalação, habilitamos senhas de hashed MD5. Isto é geralmente uma idéia muito boa, que permite senhas mais longas uma criptografia melhor.

As senhas MD5 podem ser reconhecidas no arquivo */etc/shadow* pelo prefixo 1.

6.2.7 RODE O NÚMERO MÍNIMO DE SERVIÇOS REQUERIDOS

Serviços são programas como servidores de FTP e servidores de Web. Desde que eles estejam escutando as conexões entrantes os computadores externos podem acessá-los. Serviços são vulneráveis e por isso podem ser um de risco segurança.

Em nosso sistema não foram instalados serviços que não eram necessários à máquina.

6.2.8 DESABILITANTO DEAMON SERVICES

Desabilitar Deamon services é finalizá-los simplesmente. Há diferentes métodos, em nosso caso utilizamos o método manual, que está descrito abaixo:

- Remover os links do /etc/rc${runlevel}.d/
- Mover o script do arquivo /etc/init.d/_service_name_ para outro nome qualquer.
- Remover as permissões de executar o arquivo /etc/init.d/_service_name_
- Editar o script /etc/init.d/_service_name_ para pará-lo imediatamente.

6.2.9 DESABILITANDO INETD SERVICES

Foram parados todos os serviços desnecessários ao sistema, como echo, chargen, discard, daytime, time, talk, ntalk e r-services (rsh, rlogin e rcp) que são considerados altamente inseguros.

Pode-se desabilitar serviços editando o /etc/inetd-conf diretamente, mas o Debian provê uma alternativa melhor: update-inetd.

Pode-ser remover o telnet deamon utilizando o camando abaixo mudando a configuração do arquivo e reiniciando o deamon.

- /usr/sbin/update-inetd --disable telnet

6.2.10 INSTALE A QUANTIDADE MÍNIMA DE SOFTWARE REQUERIDA

O debian vem com inúmeros softwares e milhares de pacotes que podem ser utilizados em seu sistema. Contudo vários deles são desnecessários para a aplicação de sua máquina.

Desde que se saiba para que o sistema será utilizado deve-se instalar somente o software que é realmente necessário. Toda ferramenta desnecessária que for instalada pode ser usada por um intruso que queira comprometer o seu sistema.

6.2.11 IMPORTÂNCIA DO PERL

O perl é extremamente importante para o sistema e sua remoção é muito difícil, tendo em vista que ele é usado pela maioria dos utilitários do sistema Debian.

Os utilitários que usam perl podem ser vistos em:

- $ for i in /bin/* /sbin/* /usr/bin/* /usr/sbin/*; do [-f $i] && {
  type=`file $i | grep -il perl`; [-n "$type"] && echo $i;
  }; done

Isto inclui os seguintes utilitários nos pacotes com prioridade *requerido* ou *importante*:

- */usr/bin/chkdupexe of package util-linux.*
- */usr/bin/replay of package bsdutils.*
- */usr/sbin/cleanup-info of package dpkg.*
- */usr/sbin/dpkg-divert of package dpkg.*
- */usr/sbin/dpkg-statoverride of package dpkg.*
- */usr/sbin/install-info of package dpkg.*
- */usr/sbin/update-alternatives of package dpkg.*

- */usr/sbin/update-rc.d of package sysvinit.*
- */usr/bin/grog of package groff-base.*
- */usr/sbin/adduser of package adduser.*
- */usr/sbin/debconf-show of package debconf.*
- */usr/sbin/deluser of package adduser.*
- */usr/sbin/dpkg-preconfigure of package debconf.*
- */usr/sbin/dpkg-reconfigure of package debconf.*
- */usr/sbin/exigrep of package exim.*
- */usr/sbin/eximconfig of package exim.*
- */usr/sbin/eximstats of package exim.*
- */usr/sbin/exim-upgrade-to-r3 of package exim.*
- */usr/sbin/exiqsumm of package exim.*
- */usr/sbin/keytab-lilo of package lilo.*
- */usr/sbin/liloconfig of package lilo.*
- */usr/sbin/lilo_find_mbr of package lilo.*
- */usr/sbin/syslogd-listfiles of package sysklogd.*
- */usr/sbin/syslog-facility of package sysklogd.*
- */usr/sbin/update-inetd of package netbase.*

Assim, sem perl, e a menos que se refaça estes utilitários em sheel script, provavelmente não será possível gerenciar qualquer pacote, o que impossibilitará o upgrade do sistema.

6.2.12 EXECUTANDO UPDATE SEGURO

É impoprtante estar ciente das atualizações do sistema, pois sempre haverão novos bugs detectados e estes são corrigidos nas atualizações. deve-se sempre checar os sites de segurança para ver se

há *updates* de segurança. Depois que um BUG é corrigido um novo pacote pode ser encontrado em http://security.debian.org.

Para fazer a instalação do nosso sistema antes verificamos se não haviam updates de segurança, feito isso, criamos regras de *FireWall* para que o sistema possa se conectar apenas ao security.debian.org e então rodar o update, antes de se conectar à internet. A configuração é simples e segue abaixo.

Nota: acdicione primeiro o endereço de IP para http://security.debian.org em */etc/hosts* em seguida teste a configuração para ver se vai funcionar.

```
# iptables -F
# iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
# iptables -P INPUT DROP
# iptables -P FORWARD DROP
# iptables -P OUTPUT DROP
# iptables -A OUTPUT -d security.debian.org --dport 80
-j ACCEPT
# iptables -A INPUT -m state --state
ESTABLISHED,RELATED -j ACCEPT
# iptables -A INPUT -p icmp -j ACCEPT
# iptables -A INPUT -j LOG
# iptables -A OUTPUT -j LOG
# iptables -L
Chain INPUT (policy DROP)
Target rot opt source              destination
ACCEPT all  -- 0.0.0.0/0    0.0.0.0/0     state
RELATED,ESTABLISHED
ACCEPT icmp -- 0.0.0.0/0      0.0.0.0/0
```

```
LOG    all  -- anywhere       anywhere           LOG
level warning


Chain OUTPUT (policy DROP)
target     prot opt source           destination
ACCEPT     80   --  anywhere
security.debian.org
LOG        all  --  anywhere         anywhere
LOG level warning
```

Após fazer esta configuração podemos fazer o update do nosso sistema com mais tranquilidade.

Para atualizar manualmente o sistema ponha a seguinte linha em seu sources.list e o update começara automaticamente: *deb http://security.debian.org/ stable/updates main contrib non-free*

Uma vez feito isso é só usar o apt ou dselecte para o upgrade.

Em nosso sistema itilizamos o apt, para isso basta apenas, como root, dar os seguintes comandos:

- *# apt-get update*
- *# apt-get upgrade*

6.2.13 SETANDO A BIOS NOVAMENTE

Neste ponto reconfiguramos a BIOS para impedir o boot via disco removível, para tanto setamos a configuração default onde o boot é apenas feito por disco rígido.

6.2.14 SETANDO UMA SENHA PARA O LILO

Para prevenir que qualquer pessoa inicialize um root-sell e altere as senhas entrando com <name-of-your-bootimage> init=/bin/sh no prompt de boot, foi setada uma senha para o boot loader.

Como usamos o LILO, o procedimento foi o seguinte, editamos o arquivo de configuração /etc/lilo.conf e adicionar uma senha e a linha restricted, como segue abaixo:

```
i mage=/boot/2.2.14-vmlinuz
        label=Linux
        read-only
        password=hackme
        restricted
```

6.2.15 REMOVENDO O PROMPT ROOT DO KERNEL

O Linux kernel 2.4 fornece uma maneira de acessar o root shell enquanto inicia o sistema.

Em nosso sistema esta opção foi removida e fizemos isso alterando: */etc/mkinitrd/mkinitrd.conf* e setando:

- # DELAY The number of seconds the linuxrc script should wait to
- # allow the user to interrupt it before the system is brought up
- DELAY=0

Em seguida regeneramos a ramdiskimage. As duas maneiras possíveis seguem abaixo:

- # cd /boot

- # mkinitrd -o initrd.img-2.4.18-k7 /lib/modules/2.4.18-k7

ou

- # dpkg-reconfigure -plow kernel-image-2.4.x-yz

6.2.16 MONTANDO AS PARTIÇÕES

Quando montamos uma partição ext2, há uma porção de opções adicionais que podemser usadas na */etc/fstab*, vejamos o exeplo da partição */tmp*:

- */dev/hda7 /tmp ext2 defaults,nosuid,noexec,nodev 0 2*

A opção *nosuid* ignora os bits *setuid* e *setgid* completamente, enquanto *noexec* proíbe a execução de qualquer programa no ponto de montagem, e nodev, ignora os devices, mas isto:

- É aplicável apenas em sistemas de arquivos ext2
- Pode ser burlado facilmente

A opção noexec proíbe que binários sejam executados diretamente.

O exemplo abaixo mostra que o */var* pode ser setado o *noexec*, mas alguns softwares sustentam programas abaixo do */var*. O mesmo aplica-se a opção *nosuid.*

```
dev/sda6    /usr        ext2  defaults,ro,nodev       0  2
/dev/sda12  /usr/share  ext2  defaults,ro,nodev,nosuid  0
    2
```

```
/dev/sda7       /var            ext2    defaults,nodev,usrquota,grpquota
        0       2
/dev/sda8       /tmp            extt2
        defaults,nodev,nosuid,noexec,usrquota,grpquota 02
/dev/sda9       /var/tmp        ext2
        defaults,nodev,nosuid,noexec,usrquota,grpquota  0       2
/dev/sda10      /var/log        ext2    defaults,nodev,nosuid,noexec
        0       2
/dev/sda11      /var/account    ext2    defaults,nodev,nosuid,noexec
        0       2
/dev/sda13      /home           ext2
        rw,nosuid,nodev,exec,auto,nouser,async,usrquota,grpquota
        0       2
/dev/fd0        /mnt/fd0        ext2    defaults,users,nodev,nosuid,noexec
        0       0
/dev/fd0        /mnt/floppy     vfat    defaults,users,nodev,nosuid,noexec
        0       0
/dev/had        /mnt/cdrom      iso9660 ro,users,nodev,nosuid,noexec
        0       0
```

6.3 PROVENDO ACESSO SEGURO AO USUÁRIO

6.3.1 AUTENTICAÇÃO DE USUÁRIO: PAM

PAM (Pluggable Authentication Modules) permite que o administrador do sistema escolha como as aplicações vão funcionar para os usuários autenticados.

Cada aplicação com suporte ao PAM fornece um arquivo de configuração em */etc/pam.id* que pode ser usado para modificar o comportamento:

- Que backend é usado para autenticação
- Que backend é usado para sessões
- Como fazer as verificações de senhas

O PAM oferece a possibilidade de atravessar etapas da autenticação de uma vez, sem o conhecimento do usuário. Pode-se autenticar em um banco de dados Berkley ou em um arquivo de senha normal e o usuário apenas logar se autenticar corretamente em ambos.

A primeira coisa que fizemos foi adicionar suporte MD5 para todas as aplicações que utilizam o PAM, isto ajuda contra dicionários cracks e em seguida adicionar duas linhas a todos os arquivos em */etc/pam.d* que garantem acesso à máquina, como login e ssh.

Be sure to install libpam-cracklib first or you will not be able to log in

password required pam_cracklib.so retry=3 minlen=12 difok=3

password required pam_unix.so use_authtok nullok md5

A primeira linha carrega a crackling PAM module, que permite password strength-checking, alerta para uma senha nova com comprimento mínimo de 12 caracteres. A segunda linha introduz o módulo padrão de autenticação com senhas MD5 e permite senha de tamanho zero.

Para ter certeza de que o usuário root pode logar no sistema apenas no terminal local, a seguinte linha foi abilitada em */etc/pam.d/login:*

- auth requisite pam_securetty.so

Então pode-se adicionar os terminais de que o usuário root pode registrar no sistema, em /etc/security/access.conf. Por último para abilitar limites de usuários utilizamos a linha:

- session required pam_limits.so

Editamos o arquivo */etc/pam.d/passwd* e modificamos a primeira linha. Adicionamos a opção "md5" para usar as senhas MD5, mudando o tamanho mínimo e máximo das senhas. A linha final ficou assim:

password required pam_unix.so nullok obscure min=6 max=11 md5

Protegemos o *SU*, para que apenas algumas pessoas pudessem acessar o sistema como *root*. Para isso, adicionamos o grupo w*heel* ao sistema. Adicionamos o *root* e outros usuários que deveriam ter permissão de acessar o *root* por *su* a esse grupo. Aí, adicionamos a seguinte linha ao */etc/pam.d/su*:

auth requisite pam_wheel.so group=wheel debug

Isso certificou que apenas pessoas do grupo *wheel* podiam usar o *su* para se tornar root. Outros usuários recebiam mensagem de *denied* ao tentarem se tornar *root* do sistema.

Para nos certificarmos que apenas certos usuários poderiam se autenticar em um serviço *PAM*, usamos arquivos onde estavam armazenados os usuários que estavam (ou não!) habilitados a fazer *login* via *ssh.* Criamos o arquivo */etc/sshusers-allowed* com os usuarios habilitados e escrevemos a seguinte linha dentro de */etc/pam.d/ssh*:

Auth required pam_listfile.so item=user sense=allow file=/etc/sshusers-allowed onerr=fail

Por ultimo, mas não menos importante, criamos o arquivo */etc/pam.d/other* e escrevemos as seguintes linhas:

auth required pam_securetty.so
auth required pam_unix_auth.so
auth required pam_warn.so
auth required pam_deny.so
account required pam_unix_acct.so
account required pam_warn.so
account required pam_deny.so
password required pam_unix_passwd.so
password required pam_warn.so
password required pam_deny.so
session required pam_unix_session.so
session required pam_warn.so
session required pam_deny.so

Essas linhas fornecem uma boa configuração *default* para todas as aplicações que que suportam PAM (O acesso é negado por default).

6.3.2 LIMITANDO O USO DE RECURSOS: O ARQUIVO *LIMITS.CONF*.

Essem arquivo e sua edição devem ser levados muito a sério, pois é nele que definimos a limitação ao uso dos recursos do sistema. Usando o PAM, o arquivo */etc/limits.conf* é ignorado e passamos a usar o */etc/security/limits.conf*.

Foi desse modo que resolvemos o problema de estouro de recursos. Isso acontece quando os usuários não têm limite de uso de armazenamento em disco, CPU, memória RAM, etc. Limitamos os recursos de alguns *Shells* (como o *BASH e o CHSH*).

6.3.3 AÇÕES EM LOGIN DE USUÁRIOS:

Editamos a configuração básica e as ações sobre os logins de usuários. Seguem as regras:

FAIL_DELAY 10: colocamos o tempo entre as tentativas de digitação de senha em 10 segundos. Se uma senha é digitada errada, somente após 10 segundos é que se pode tentar novamente. Isso dificulta a tentativa de quebra de senha usando a força bruta! Note que essa regra é inútil quando se está usando programas como *getty* ou *mingetty*.

FAILLOG_ENAB yes: Armazenamos as falhas de login. Assim, poderemos rastrear quem tentar invadir o sistema.

LOG_UNKFAIL_ENAB yes: Essa opção deve ser colocada em conjunto com a anterior. Essa regra armazena os logins inexistentes que forem usados nas tentativas de logins que falharam.

SYSLOG_SU_ENAB yes: Essa opção grava as tentativas do *SU* em fazer *syslog*.

MD5_CRYPT_ENAB *yes*: Essa opção habilita a criptografia *MD5*, o que reduz drasticamente o problema de ataques por dicionário, desde que tenhamos habilitado o uso de senhas maiores.

PASS_MAX_LEN *50:* Como configuramos o *MD5* no *PAM,* essa opção deve ser configurada com o mesmo valor do usado no *PAM.*

6.3.4 RESTRINGINDO O FTP: EDITANDO O */ETC/FTPUSERS*:

O arquivo */etc/ftpusers* contem uma lista dos usuários que têm permissão de fazer login ao *host* usando o *ftp*. Esse arquivo deve ser usado apenas se realmente exite a necessidade do uso de *ftp* no sistema.

6.3.5 USANDO O *SU*:

O comando *su* é usado para nos tornar o *Super User*, quando logados no sistema como um usuário comum. Mas, esse uso deve ser evitado por questões de segurança. A melhor opção é remover o *su* e mudar para o *sudo*. Esse possui mais configurações que o *su*, mas o *su* é o mais comumente usado.

6.3.6 USANDO O *SUDO*:

O *sudo* permite que o usuário execute comandos pré-definidos sob uma outra identidade, até mesmo como *root*. Adicionamos os usuários que poderiam ter permissão de usar o *sudo* em */etc/sudoers*. Nesse arquivo também definimos os comandos que deveriam estar habilitados para o *sudo*. Violações, tanto para senhas incorretas quanto para tentativas de executar programas que não se tem permissão são armazenadas e enviadas para o *root*.

6.3.7 DESABILITANDO O ACESSO ADMINISTRATIVO REMOTO:

Modificamos o arquivo */etc/security/access.conf* para desabilitarmos o acesso administrativo remoto. Dessa forma, somente usando o *su* ou *sudo* é que se pode usar poderes adminstrativos. E isso faz com sempre haja registros do uso de poderes admistrativos.

Adicionamos a seguinte linha no arquivo */etc/security/access.conf*:

-:wheel:ALL EXCEPT LOCAL

6.4 TRANSFERÊNCIA SEGURA DE ARQUIVOS:

Em uma administração normal, às vezes faz-se necessária a tranferência de arquivos para dentro e para fora do sistema. A cópia segura de arquivos de um *host* para outro, de forma segura, pode ser feita usando o servidor de pacotes *sshd*. Uma outra possibilidade seria o uso do *ftpd-ssl*, um servidor *ftp* que usa o *SSL (Secure Socket Layer)* para cifrar suas transmissões.

Todos esses métodos necessitam, claro, de *clients* especiais. O Debian possui alguns *clients*. Por exemplo, o *ssh* fornece o *scp*. Ele funciona como o *rcp*, mas completamente cifrado, evitando que os *bandidos* descubram o que se está copiando. Existe também o *ftp-ssl client* para o servidor equivalente. Existe desses *clients* para sistemas *não-Unix*, como o *Putty* e o Winscp para os sistemas operacionais da *Microsoft.*

6.5 LIMITE E CONTROLE DO SISTEMA DE ARQUIVOS:

6.5.1 USANDO *QUOTAS:*

Ter uma boa política de *quota* é importante para evitar que os usuários estourem o tamanho do *HD*. Podem ser usados dois sistemas de *quotas*: Por grupo e por usuário (adotado).

Os pontos mais importantes a serem lembrados para se montar um sistema de *quotas*:

- Manter as *quotas* pequenas o suficiente para que os usuários não acabem com o espaço em disco.
- Manter as *quotas* grandes o suficientes para que os usuários não reclamem ou que seus e-mails continuem recebendo e-mails por um longo período.
- Usar o sistema de *quotas* em todas as áreas de escrita de usuários; tanto no */tmp* quanto no */home*.

Cada partição ou diretório em que os usuários tenham pleno direito de escrita devem ter as *quotas* habilitadas. Os cálculos devem ser feitos de forma a preservar tanto a segurança quanto a funcionalidade. Lembrando que, se não foi habilitado o controle de *quotas* em seu kernel, o mesmo terá de ser recompilado.

Modificamos de *defaults* para *defaults,usrquota* no arquivo */etc/fstab*. Depois, criamos o arquivo vazio *quota.user* nas raízes do sistema de arquivos em que queríamos usar as *quotas,* usando *touch /home/quota.user /home/quota.group* do sistema de arquivos */home*.

Reiniciamos a *quota* fazendo */etc/init.d/quota stop;/etc/init.d/quota start*.

Editamos a *quota* de um usuário específico usando *edquota -u ref*.

6.6 MANTENDO A SEGURANÇA DO ACESSO À REDE.

6.6.1 CONFIGURANDO AS *FEATURES* DO *KERNEL* DA REDE:

Muitas *features* do *kernel* podem ser modificadas durante a execução "ecoando" alguma coisa para dentro do arquivo de sistema */proc* ou usando o *sysctl –w variable*=<*value*>. Mas, raramente será necessário editar alguma coisa aqui.

Bloqueamos os *pings* por *broadcast*, usando:

net/ipv4/icmp_echo_ignore_broadcasts = 1

Também bloqueamos os outros *ICMPS* usando essa configuração:

net/ipv4/icmp_echo_ignore_all = 0

6.6.2 MANTENDO A SEGURANÇA DA REDE DURANTE O *BOOT-TIME.*

Para que não precisemos configurar algumas opções do *kernel* da rede toda vez que o sistema é reiniciado, usamos o *script* a seguir para seja uma operação automática durante o *boot*. Esse *script* deve ser criado em */etc/network/interface-secure* e deve ser chamado por */etc/network/interfaces*. Segue o script:

```
auto eth0
iface eth0 inet static
    address xxx.xxx.xxx.xxx
    netmask 255.255.255.xxx
    broadcast xxx.xxx.xxx.xxx
    gateway xxx.xxx.xxx.xxx
    pre-up /etc/network/interface-secure
# Script-name: /etc/network/interface-secure
```

```
# Modifies some default behaviour in order to secure against
# some TCP/IP spoofing & attacks
#
# Contributed by Dariusz Puchalak
#
echo 1 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
                                    # broadcast echo protection enabled
echo 0 > /proc/sys/net/ipv4/ip_forward      # ip forwarding disabled
echo 1 > /proc/sys/net/ipv4/tcp_syncookies # TCP syn cookie protection enabled
echo 1 >/proc/sys/net/ipv4/conf/all/log_martians # Log strange packets
# (this includes spoofed Packets, source routed Packets, redirect Packets)
                                # but be careful with this on heavy loaded web servers
echo 1 > /proc/sys/net/ipv4/ip_always_defrag
                                    #  defragging protection always enabled
echo                    1                   >
/proc/sys/net/ipv4/icmp_ignore_bogus_error_responses
                                    # bad error message protection enabled

# now ip spoofing protection
```

echo 1 > /proc/sys/net/ipv4/conf/all/rp_filter

 # and finally some more things:
 # Disable ICMP Redirect Acceptance
echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects
echo 0 > /proc/sys/net/ipv4/conf/all/send_redirects

 # Disable Source Routed Packets
echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route
echo 1 > /proc/sys/net/ipv4/conf/all/log_martians

6.7 SERVIÇOS SEGUROS RODANDO NO SISTEMA

Os serviços, para serem seguros, devem ser configurados corretamente de modo que possam somente ser usados por usuários legitimamente autorizados.

Para restringir serviços que so podem ser alcançados de um dado local restringimos o acesso à eles a nível o Kernel – Firewall – configurando-os para ouvir apenas uma dada interface.

6.7.1 SECURING SSH

O ssh deve ser usado para todos os logins remotos. Em uma era onde o tráfego de internet pode ser facilmente capturado, devemos utilizar apenas protocolos que utilizem criptografia.

Utilizamos SSH versão 2 para maior segurança.

Utilizamos o pam_listfile para restringir no arquivo de controle do PAM para restringir inicio de sessão ssh, barrando qualquer

um não listado no /etc/loginusers adicionando linha abaixo em /etc/pam.d/ssh:

auth required pam_listfile.so sense=allow onerr=fail item=user file=/etc/loginusers

7 CONCLUSÃO

Ao fim deste trabalho podemos chegar a inúmeras conclusões que nos levam a crer que a segurançada de rede não é feita apenas na máquina na qual se quer proteger, é necessário uma política de segurança rígida e eficaz que conscientize cada usuário da rede, fazendo com que estes zelem pelo bom funcionamento da mesma, adotando cada regra da política implementada naquela rede.

As regras de uma boa política de segurança começam durante a instalação do servidor, onde o adminitrador deve conhecer cada passo a ser tomado para se instalar uma máquina segura. Estes passos foram descritos neste trabalho, obviamente isto não garante totalmente que a máquina está 100% segura, pois a cada dia surgem novas formas de invasão de sistemas ou se descobre novos bugs de software que proporcionam brechas para uma invasão.

A partir destas informações temos concluímos que é extremamente necessário que o administrador do sistema esteja sempre muito atualizado com relação às novidades sobre segurança que circulam na internet, para que a cada descoberta nova ele possa atualizar seu sistema rapidamente tornando-o cada vez mais estável e deixando-o cada vez mais protegido.

Foram apresentadas inúmeras formas de ataque existentes, bem como os métodos de proteção e contra-ataque, portanto deve-se estar atento às novidades, mas não se deve deixar de lado as vulnerabilidades já conhecidas.

Enfim mostramos neste trabalho como funciona a organização de uma política de segurança para redes coorporativas e apresentamos uma solução segura para uma necessidade da BrasilTelecom, empresa da qual fazemos parte. Esta solução pode ser adapitada e utilizada em diversas configurações de ambientes, não apenas nesta empresa. Contudo nela utilizamos equipamentos e meios de conexão de alto nível, possuídos apenas por empresas de grande porte nesta área.

Portanto uma boa política de segurança aliada á uma configuração bem feita das máquinas que protegem a rede, administrada por pessoas muito bem atualizadas é a receita para se ter uma rede segura.

8 BIBLIOGRAFIA

Uchôa, Joaquim Quinteiro Segurança em Redes e Criptografia / Joaquim Quinteiro Uchôa. - - Lavras: UFLA/FAEPE, 2003.

BARBOSA, André S. – Laboratório de Redes de Alta Velocidade COPPE/UFRJ – Sistemas de Detecção de Intrusão

CERT - Computer Emergency Respose Team - http://www.cert.org

ICSA - Internet Consortium Security Agency - http://www.icsa.net

CIDF Site – http://www.gidos.org

RFC2828 - Internet Security Glossary

RFC2196 - Site Security Handbook

IETF - Internet Engineering Task Force – http://www.ietf.org

SOARES, Luiz Fernando. Redes de Computadores: das Lan´s, Man´s e Wan´s às
redes ATM. 2ª Edição revisada e ampliada. Rio de Janeiro. Editora Campus, 1955.

TANENBAUM, Andrew S. Redes de Computadores. Rio de Janeiro. Editora
Campus, 1997.

HARDENING, disponível em : www.debian.org/doc/manuals/securing-debian-howto

SNORT PAGE, Disponível em: http://www.snort.org

SCHNEIDER, Kurt - SNORTFACE – Interface Para Configuração, Distribuição Em Rede E Coleta De Dados Do IDS Snort, Campus Frederico Westphalen Departamento de Engenharia e Ciências da Computação, Agosto de 2001.

CARTY, Aidan - Building An IDS Solution Using Snort - Systems and Security Architect. Entropy Ltd. http://www.entropy.ie/research/

OLIVEIRA, Gustavo Ferreira Résio, Brito, Gustavo Arantes – Sistema de Detecção de Intrusão. 2002. Projeto Final de curso – Escola de Engenharia Elétrica, Universidade Federal de Goiás, Goiânia.

Uchôa, Joaquim Quinteiro Segurança em Redes e Criptografia / Joaquim Quinteiro
UFLA/FAEPE, 2003. Curso de Pós-Graduação “Lato Sensu” (Especialização) a
Distância: Administração em Redes Linux.

9 ANEXO A: SCRIPTS DE CONFIGURAÇÃO DO IPTABLES, PAM E JAIL:

```
**************************************************************************
**************************************************************************
*******
******* SCRIPT do IPTABLES
*******
**************************************************************************
**************************************************************************
wsacesso-BSA:~# more /etc/rc2.d/S19firewall
#! /bin/sh

/etc/init.d/ssh restart
/usr/sbin/sshd -f /etc/ssh/sshd2_config
/usr/sbin/sshd -f /etc/ssh/sshd3_config


stopfw() {
        iptables -t filter -F
        iptables -t filter -P INPUT ACCEPT
        iptables -t filter -P OUTPUT ACCEPT
}

#### Tornar filesystem ro

##### Protec,a~o contra IP Spoofing #####
for i in /proc/sys/net/ipv4/conf/*/rp_filter; do
     echo 1 >$i
```

```
done


##### Ativamos o redirecionamento de pacotes (requerido para NAT)
#####
echo "0" >/proc/sys/net/ipv4/ip_forward

##### Definindo o no. maximo Conntrack
echo "2048" > /proc/sys/net/ipv4/ip_conntrack_max


startfw() {
##############################################
##
## Definição de Policiamento
##
##############################################
## Tabela filter

# Policy Default (DROP)
 iptables -t filter -P INPUT DROP
 iptables -t filter -P OUTPUT DROP
 iptables -t filter -P FORWARD DROP

# Contra ataques DoS
 iptables -t filter -A INPUT -p tcp --syn -m limit --limit 5/s -j ACCEPT
# Protecao contra MAC Spoofing
```

```
 iptables -t filter -A INPUT -m mac --mac-source 00:80:AD:42:37:F2 -j
DROP
 iptables -t filter -A INPUT -m mac --mac-source 00:10:B5:E4:16:42 -j
DROP
# Permite input para Porta TCP 22
 iptables -A INPUT -m state --state NEW,ESTABLISHED -p tcp --dport
22 -j ACCEPT
# Permite TRACEROUTE
 iptables -A INPUT -m state --state ESTABLISHED,RELATED -p icmp --
icmp-type echo-reply -m limit --limit 5/s -j ACCEPT
 iptables -A INPUT -m state --state ESTABLISHED,RELATED -p icmp --
icmp-type redirect -m limit --limit 5/s -j ACCEPT
 iptables -A INPUT -m state --state ESTABLISHED,RELATED -p icmp --
icmp-type time-exceeded -m limit --limit 5/s -j ACCEPT
 iptables -A INPUT -m state --state ESTABLISHED,RELATED -p icmp --
icmp-type destination-unreachable -m limit --limit 5/s -j ACCEPT

# Permite SSH na porta 2222(Tunnel SIEMENS)
        for i in `cat /etc/ssh.allow`
          do
iptables -A INPUT -m state --state NEW,ESTABLISHED -s $i -p tcp --
dport 2222 -j ACCEPT
          done

# Permite input Siemens CTA porta 1503/ssh
 iptables -A INPUT -m state --state NEW,ESTABLISHED,RELATED -s
200.103.174.248/29 -p tcp --dport 1503 -j ACCEPT
# Permite Monitoramento NAGIOS
```

```
 iptables -A INPUT -p icmp -s 200.199.241.5 -j ACCEPT
# Permite Conexões Estabelecidas
 iptables -A INPUT -m state --state ESTABLISHED -j ACCEPT

# Parte de LOG do Firewall
 iptables -A INPUT -m state --state INVALID -j LOG --log-prefix
"FIREWALL: "
 iptables -A INPUT -j LOG --log-prefix "FIREWALL: "
 iptables -A INPUT -j REJECT


#################################################################
##################
##
## Liberação de Acessos Necessarios para o Funcionamento da
Maquina
##
##########
#################################################################
########

###############################
## Liberação de Saida p/ Programas #
###############################
# Permite NTP
 iptables -A OUTPUT -m state --state NEW -d 200.199.241.12 -p udp --
dport 123 -j ACCEPT
```

```
 iptables -A OUTPUT -m state --state NEW -d 200.180.128.10 -p udp --
dport 123 -j ACCEPT
# Permite Acesso do TACACS
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED  -
d 200.199.241.37 -p udp --dport 1645 -j ACCEPT
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED  -
d 200.199.241.37 -p udp --dport 1646 -j ACCEPT
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -d
200.203.190.55 -p udp --dport 1645 -j ACCEPT
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -d
200.203.190.55 -p udp --dport 1646 -j ACCEPT

################################
## Liberação de Saida p/ USUARIO #
################################
# Permite Ping
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -p
icmp -j ACCEPT
# Permite SNMP
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED -p udp --
dport 161 -j ACCEPT
# Permite Consulta DNS
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED -p udp --
dport 53 -j ACCEPT
# Permite Telnet Reverso
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -p
tcp --dport 2001 -j ACCEPT
# Permite mandar mail via 200.199.241.12
```

```
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -d
200.199.241.12 -p tcp --dport 25 -j ACCEPT
# Permite Portas TCP conhecidas
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -p
tcp --dport 20:23 -j ACCEPT
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -p
tcp --dport 80 -j ACCEPT
 iptables -A OUTPUT -m state --state NEW,ESTABLISHED,RELATED -p
tcp --dport 443 -j ACCEPT

################################
## Permite Conexões Estabelecidas #
################################
 iptables -A OUTPUT -m state --state ESTABLISHED,RELATED -j
ACCEPT


}


# Configurable options:

case "$1" in
  start)
        echo "Starting Firewall"
        startfw
        ;;
  stop)
```

```
        echo "Stoping Firewall"
        stopfw
                        ;;

                reload|restart)
                        stopfw
        startfw
        ;;
  *)
        echo "Usage: /etc/init.d/firewall {start|stop|reload|restart}"
        exit 1
esac

exit 0
```

****

****

****

******* SCRIPT de Criação do JAIL

****

****

****

```
wsacesso-BSA:~# more /usr/local/bin/criajail
#!/bin/bash

if [ $1 ]
```

```
 then

      PROGS="telnet bash sh cat grep ls mkdir mv cp nslookup pwd rm
rmdir tail more touch vi head sftp scp ssh id"

      mkjailenv /home/.$1

      for i in `echo $PROGS`
      do
            addjailsw /home/.$1 -P $i
      done

      cp /etc/bash.bashrc /home/.$1/etc/profile
      cp /usr/bin/telnet /home/.$1/bin
      cp /usr/lib/sftp-server /home/.$1/usr/lib

      rm -r /home/.$1.

      more /home/.$1/etc/shadow | grep -v root >> /tmp/shadow.tmp
      mv /tmp/shadow.tmp /home/.$1/etc/shadow

 else
      echo -e '\n'
      echo -e 'USO:   criajail <jail>  \n'
      echo -e '\n Exemplo:    criajail <jail> \n'
      echo  -e  '(Neste  exemplo  é  criado  o  chroot  enviroment
"/home/.<jail>"\n'
```

fi

wsacesso-BSA:~# criajail

USO: criajail <jail>

Exemplo: criajail <jail>

(Neste exemplo é criado o chroot enviroment "/home/.<jail>"

****
****
****
******* SCRIPT de Inserção de Usuario no JAIL
****
****
****

wsacesso-BSA:~# criajailuser

USO: ./mkuserbt.sh <jail> <user>

Exemplo: ./mkuserbt.sh /home/.<jail> <userjail>

(Neste exemplo é criado o usuario "userjail" dentro do jail "jail"

```
wsacesso-BSA:~# more /usr/local/bin/criajailuser
#!/bin/bash

if [ $1 ] && [ $2 ]
  then

        addgroup $2
        useradd -s /usr/local/bin/jail -d /home/.$1/ -g $2 -G $2 $2
        addjailuser /home/.$1 /home/$2 /bin/bash $2
        chgrp $2 /home/.$1/home/$2

  else
        echo -e '\n'
        echo -e 'USO:   ./mkuserbt.sh <jail> <user> \n'
        echo -e '\n Exemplo:    ./mkuserbt.sh /home/.<jail> <userjail>\n'
        echo -e '(Neste exemplo é criado o usuario "userjail" dentro do jail
"jail"\n'

fi
```