

Implementação de Firewalls em Redes de Computadores: Teoria e Prática

Ana Clara Reis¹, Áthina Galassi Strini¹, Lucas Rodrigues Silva Flores¹

Faculdade de Computação – Universidade Federal de Uberlândia (UFU)
Rodovia LMG 746, km 1, s/n, Bairro Araras – 38500-000 – Monte Carmelo – MG –
Brasil
{anaclarareis, athinags, lucas.flores}@ufu.br

Resumo

Este artigo analisa a implementação de firewalls em redes de computadores, fazendo uma combinação de conhecimento teórico e uma prática que reforça a teoria. O estudo aborda conceitos essenciais de segurança de rede, as arquiteturas de firewall (como filtro de pacotes, filtros de estado e gateways de aplicação) e o uso de uma Zona Desmilitarizada (DMZ). Por meio de uma simulação no Cisco Packet Tracer e uma implementação em hardware real com um firewall baseado em Linux, demonstramos a eficácia dessas soluções na proteção da infraestrutura de rede. Os resultados obtidos comprovam que a configuração de firewalls melhora significativamente a segurança de uma rede, reforçando a importância da implementação.

Abstract

This paper analyzes the implementation of firewalls in computer networks, combining theoretical knowledge with practical application to reinforce the theory. The study addresses essential concepts of network security, firewall architectures (such as packet filters, stateful filters, and application gateways), and the use of a Demilitarized Zone (DMZ). Through a simulation in Cisco Packet Tracer and a real-hardware implementation with a Linux-based firewall, we demonstrate the effectiveness of these solutions in protecting network infrastructure. The results obtained prove that firewall configuration significantly improves network security, reinforcing the importance of implementation.

1. Introdução

A segurança em redes de computadores virou uma preocupação fundamental no contexto da Internet moderna. Com as empresas dependendo cada vez mais da conexão online para realizar suas atividades, garantir a segurança dos ativos de rede se transformou em algo essencial, ultrapassando as questões meramente técnicas.

Um firewall isola a rede interna de uma organização da internet, permitindo que alguns pacotes passem e outros não [1]. Essa descrição define o firewall como uma ferramenta para controlar o acesso, permitindo que o administrador da rede controle o tráfego de dados que entra e sai dos recursos sob sua responsabilidade.

A justificativa para o estudo mais aprofundado de firewalls é baseado em três objetivos fundamentais que estes sistemas devem cumprir: primeiro, todo o tráfego de fora para dentro, e vice-versa, deve passar por um firewall; segundo, somente o tráfego autorizado, como definido pela política de segurança local, pode passar; e terceiro, o próprio firewall deve ser imune à penetração.

Esses princípios formam a base para compreender as capacidades e os limites dos firewalls modernos. A escolha desse tema é importante porque ele se encontra no ponto de intersecção entre os protocolos de rede básicos (IP, TCP, UDP) e as políticas de segurança das organizações. O estudo dos firewalls possibilita compreender de forma prática como os conceitos teóricos de redes são aplicados na criação de soluções de segurança reais.

O objetivo principal deste projeto é realizar uma análise detalhada das três categorias de firewalls: filtros de pacotes tradicionais, filtros de estado e gateways de aplicação. Para isso, unimos teoria e prática, empregando uma simulação no Cisco Packet Tracer e uma aplicação em hardware real com um firewall baseado em Linux. O objetivo é mostrar, por meio desses testes, como a implementação de regras de filtragem e o uso de arquiteturas como a Zona Desmilitarizada (DMZ) satisfazem as variadas necessidades de segurança de uma rede.

2. Fundamentação Teórica

2.1. Definição e Objetivos dos Firewalls

Um firewall é uma combinação de hardware e software que funciona como uma barreira protetora, isolando a rede interna de uma organização da internet. Sua função principal é controlar o fluxo de tráfego, permitindo que apenas certos pacotes passem e bloqueando outros [1]. Com isso o administrador consegue gerenciar o acesso aos recursos internos em relação ao mundo externo, gerenciando o fluxo de tráfego de e para esses recursos.

Os firewalls possuem três objetivos fundamentais:

Controle Total do Tráfego: Todo o tráfego de fora para dentro, e vice-versa, passa por um firewall. Essa concentração simplifica o gerenciamento e a aplicação de uma política de acesso segura, viabilizando que as empresas estabeleçam um ponto central de controle.

Aplicação de Políticas de Segurança: Somente o tráfego considerado seguro, em conformidade com a política de segurança interna, tem permissão para prosseguir. Essa funcionalidade essencial possibilita que as organizações transformem suas diretrizes gerais em regras técnicas de filtragem bem definidas.

Resistência à Penetração: O firewall em si deve ser impenetrável. Por ser um componente ligado à rede, um firewall mal projetado ou mal implementado pode dar uma falsa sensação de proteção, chegando a ser mais arriscado do que não ter proteção

nenhuma.

2.2. Filtros de Pacotes Tradicionais

Os filtros de pacotes tradicionais analisam cada unidade de dados individualmente, decidindo se a transmitem ou barram, de acordo com normas concretas estabelecidas pelo administrador. Uma organização normalmente possui um roteador de borda que conecta sua rede interna com seu ISP, e é neste ponto que ocorre a filtragem de pacotes.

CrITÉRIOS de Filtragem:

- Endereço IP de origem e de destino
- Tipo de protocolo no campo do datagrama IP (TCP, UDP, ICMP, OSPF)
- Porta TCP ou UDP de origem e de destino
- Bits de flag do TCP (SYN, ACK)
- Tipo de mensagem ICMP
- Regras diferentes para datagramas que entram e saem da rede
- Regras diferentes para diferentes interfaces do roteador

Implementação de Políticas Organizacionais: Um administrador de rede configura o firewall baseado na política da organização, considerando produtividade do usuário, uso de largura de banda e preocupações de segurança. Por exemplo, se uma empresa não quer conexões TCP chegando, a não ser para seu site, ela pode impedir todos os envios TCP SYN que chegam, menos os que vão para a porta 80 e para o IP do servidor do site.

Limitações da Filtragem Tradicional: Analisar só os endereços de fora não protege contra dados com remetente falso (IP spoofing). Além disso, decidir só com base em pacotes separados dificulta a descoberta de ataques complexos que usam sequências de pacotes.

Uma maneira comum de implementar esses filtros de pacotes em roteadores e switches é através das Listas de Controle de Acesso (ACLs). As ACLs são conjuntos de regras ordenadas que instruem o roteador a permitir ou negar o tráfego com base em critérios como endereços IP e portas. Elas atuam como a “linguagem” de programação do firewall, permitindo ao administrador aplicar as políticas de segurança da organização de forma precisa e granular, definindo exatamente quais pacotes podem entrar ou sair da rede.

2.3. Filtros de Estado (Stateful Filters)

Os filtros de estado (Stateful) representam um grande avanço nos firewalls de pacotes convencionais, pois conseguem acompanhar o andamento das conexões em tempo real. Eles

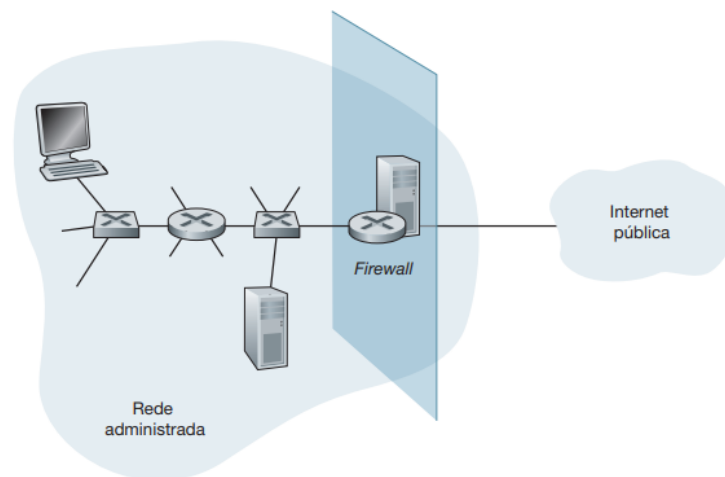


Figura 1: Arquitetura de um roteador de borda com filtro de pacotes.

utilizam uma tabela de estado, onde registram dados importantes de cada comunicação, como os IPs e as portas usadas, desde o começo até o fim.

Essa funcionalidade possibilita uma proteção mais eficaz e inteligente. Ao invés de examinar cada pacote separadamente, o firewall stateful considera todo o contexto da conexão para decidir se o libera ou não. Por exemplo, ele observa a troca de mensagens inicial do TCP (SYN, SYN/ACK, ACK) para confirmar se a comunicação é realmente legítima. Se um pacote com o flag ACK ligado chegar de fora, o firewall verifica se ele faz parte de uma conexão que começou de dentro da rede. Caso não encontre nada na tabela, o pacote é barrado, mesmo que as portas e os IPs pareçam corretos. A figura 2 exemplifica como isso funciona. Nela, o pacote de resposta do servidor (com o flag ACK) passa porque a conexão já foi validada e está na tabela do firewall, mas um pacote falso de um invasor seria bloqueado por não ter registro. Essa capacidade de analisar o estado da conexão corrige a falha dos firewalls antigos, que não sabiam diferenciar uma resposta verdadeira de um ataque externo.

Ação	Endereço de origem	Endereço de destino	Protocolo	Porta de origem	Porta de destino	Bit de flag
Permitir	222.22/16	Fora de 222.22/16	TCP	> 1023	80	Qualquer um
Permitir	Fora de 222.22/16	222.22/16	TCP	80	> 1023	ACK
Permitir	222.22/16	Fora de 222.22/16	UDP	> 1023	53	—
Permitir	Fora de 222.22/16	222.22/16	UDP	53	> 1023	—
Negar	Todos	Todos	Todos	Todos	Todos	Todos

Figura 2: Diagrama do rastreamento de estado em um Firewall.

2.4. Gateways de Aplicação

Os gateways de aplicação, que também podem ser chamados de proxies de aplicação, funcionam na camada de aplicação. A sua utilização é necessária quando precisamos de uma supervisão de segurança mais precisa, que não seja limitada aos cabeçalhos de rede e examine o conteúdo da aplicação.

Estes gateways funcionam como intermediários ou proxies, administrando toda

a circulação da aplicação. Cada gateway é um servidor distinto, criado especificamente para uma aplicação (como um gateway para HTTP, Telnet ou FTP).

2.5. Zona Desmilitarizada (DMZ)

Uma Zona Desmilitarizada (DMZ) é uma sub-rede intermediária que serve como uma camada de segurança entre uma rede interna privada e uma rede externa pública, como a Internet. O objetivo é isolar e proteger a infraestrutura interna, permitindo que serviços públicos, como servidores web, operem em um ambiente controlado e separado. Essa separação é aplicada e fiscalizada por políticas de segurança configuradas em um firewall. O aspecto do isolamento físico da DMZ é crucial para a segurança da rede. Ele garante que a Internet só possa acessar os servidores que estão isolados na DMZ, como servidores de e-mail, FTP e HTML. Isso impede que qualquer acesso externo chegue diretamente à sua rede interna, protegendo a infraestrutura principal da sua organização.

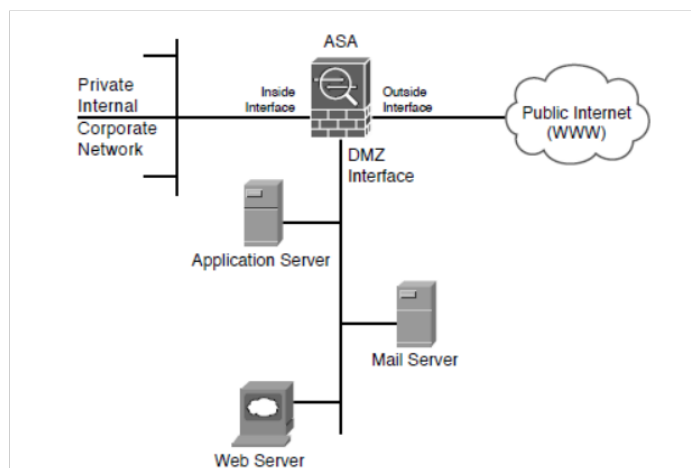


Figura 3: O isolamento físico do DMZ[5].

2.5.1 Arquiteturas de Implementação

A arquitetura de uma Zona Desmilitarizada (DMZ) pode ser implementada de diferentes formas para atender a variados níveis de segurança e complexidade. As duas principais abordagens são a de Firewall Único e a de Múltiplos Firewalls.

A arquitetura de Firewall Único é a maneira mais simples de criar uma DMZ. Ela utiliza apenas um firewall que possui pelo menos três conexões: uma conectando o firewall à internet através do ISP, outra para a rede interna e a terceira para a DMZ. Neste modelo, o firewall é o único ponto de defesa, centralizando o controle de todo o tráfego entre as redes. Embora essa abordagem seja mais fácil e econômica de configurar e gerenciar, ela também é considerada mais vulnerável. Se esse firewall for comprometido, toda a rede fica exposta, tornando-o um alvo crítico para ataques.

A arquitetura de Múltiplos Firewalls é a mais segura para uma DMZ, utiliza no mais de um firewall (geralmente dois) para criar camadas de proteção. O primeiro firewall,

chamado de firewall externo (ou front-end), é posicionado entre a internet e a DMZ para controlar o tráfego de entrada. E os demais, ficam entre a DMZ e a rede principal, regulando a comunicação entre elas. A grande vantagem dessa abordagem é a segurança significativamente maior; um atacante precisa invadir mais de um firewall para conseguir acessar a rede interna. Por essa razão, é comum o uso de firewalls de fabricantes diferentes, o que ajuda a reduzir o risco de vulnerabilidades comuns. Porém, essa arquitetura é mais complexa de gerenciar e exige um investimento maior em equipamentos e configuração.

3. Vantagens, Desvantagens e Limitações

3.1. Desvantagens da implementação de um firewall

Falsa sensação de segurança: Esse é um risco mencionado no texto do Jim Kurose, na seção sobre firewall, em que há a afirmação de que um firewall mal projetado ou instalado pode oferecer apenas uma “falsa sensação de segurança (pior do que não ter nenhum firewall!)”.

Impacto no desempenho da rede: Pode haver impacto no desempenho da rede, uma vez que todo o tráfego deve passar e ser inspecionado em um único ponto. Fazendo uma analogia com uma peneira, ela pode ficar sobrecarregada.

Complexidade de implementação e de gerência: Um firewall não é algo plug-and-play, ou seja, simples e trivial de colocar em funcionamento. A sua eficácia tem relação direta com a configuração feita e com a manutenção dada. Ele pode ser considerado uma aplicação complexa pela configuração técnica detalhada, em que há diferentes tipos de firewall e necessidade de balanceamento entre segurança e usabilidade. Caso a configuração seja muito restritiva, pode bloquear serviços legítimos que não deveriam ser bloqueados e prejudicar a produtividade.

3.2. Limitações do firewall

Susceptibilidade a penetração: O firewall não possui “imunidade à penetração”. A própria frase “mecanismo conectado à rede” evidencia essa limitação, pois, quando exposto à rede, ele também está exposto aos problemas e ameaças que busca combater.

Portanto, podem ocorrer falhas, ainda mais explícitas caso o firewall seja mal projetado ou instalado. Isso se relaciona diretamente com a falsa sensação de segurança, que pode levar usuários e administradores a baixarem a guarda enquanto, na realidade, o sistema possui uma porta dos fundos aberta para invasores.

Limitações inerentes aos métodos de filtragem: A existência de três categorias de firewall (filtros de pacotes, stateful e gateway) já indica que cada uma foi criada para suprir ou superar limitações da anterior.

Filtros de pacotes: método mais básico, sofre de “cegueira contextual”, pois analisa cada pacote isoladamente, baseando-se apenas em endereços IP e portas. Isso significa que não consegue diferenciar um pacote legítimo de um malicioso.

Além disso, não é capaz de inspecionar o conteúdo real dos pacotes, onde podem

estar ameaças escondidas.

Para suprir essa falha, houve a evolução para os firewalls stateful e gateways.

3.3. Vantagens dos tipos de Firewall

Filtros de Pacotes Tradicionais: possui filtragem baseada em cabeçalhos, isso examina campos específicos como os endereços IP de origem e destino, tipo de protocolo, portas TCP/UDP, bits de flag TCP e tipo de mensagem ICMP. além disso o controle granular permite regras diferentes para datagramas que entram e saem da rede. Há uma flexibilidade de interface, isso no caso, possibilita regras diferentes para diferentes interfaces do roteador, melhorando a flexibilidade das aplicações, como vantagem, também possui uma configuração baseada em política, ou seja o administrador pode configurar o firewall conforme a política organizacional, e por fim, pode bloquear de conexões específicas, como por exemplo impedir conexões TCP de entrada enquanto permite conexões originadas internamente, tudo isso citado ele possui uma facilidade de implementação, pois pode ser criado facilmente usando sistemas Linux com iptables, porém o fato de ser simples de implementar e ser mais básico pode ser um ponto negativo, como dito acima.

Filtros de pacotes com controle de estado: Uma das suas vantagens é que ele mantém uma tabela das conexões TCP ativas, permitindo um rastreamento de conexões e consequentemente uma maior segurança. Ele possui também uma prevenção contra ataques mais sofisticados, antes, não tão bem lidado no filtro de pacotes tradicionais. Outro ponto é que ele por meio das vias de SYN, SYNACK, ACK e pacotes FIN ele tem uma detecção e controle do início e fim das conexões, assim como a gestão automática de timeouts. Outro ponto relevante é que o mesmo consegue impedir tentativas de destruir sistemas internos com pacotes defeituosos

3.4. Vantagens dos gateways

Possui uma inspeção profunda, uma vez que examina os dados da camada de aplicação e não apenas cabeçalhos, além disso, permite uma autenticação de usuários, uma vez que permite o controle de acesso baseado na identidade de usuários específicos. Outro ponto a se citar é a funcionalidade de proxy, na qual ele atua como um intermediário entre usuários internos e servidores externos, além dele conseguir gerir múltiplas aplicações, no sentido de que pode haver vários gateways para diferentes serviços, como HTTP, FTP, e-mail dentre outros. Isso tudo passa por uma validação de credenciais, na qual o gateway solicita e verifica a identificação e a senha dos usuários.

3.5. Limitações dos gateways

Acerca das suas limitações podemos citar 3 principais desvantagens de seu uso, a primeira é a necessidade de gateway específico, ou seja, requer um gateway diferente para cada aplicação, além disso em segundo, seria o impacto que há no desempenho final, pois todos os dados devem passar pelo gateway, o que pode causar um possível gargalo, em terceiro e por último seria a sua complexidade na configuração, uma vez que o software cliente deve saber como contratar o gateway e assim especificar o servidor de destino.

4. Aplicação Proposta

A aplicação proposta consistiu na simulação, configuração e implementação física de um firewall utilizando o software Cisco Packet Tracer. Ela foi dividida em três fases: a configuração básica do ambiente para melhor visualização dos conceitos fundamentais, e uma configuração mais avançada para refletir um cenário corporativo mais realista. A aplicação demonstra na prática os conceitos teóricos como políticas de segurança, filtragem de tráfego e criação de regras granulares.

4.1. 1ª Fase: Topologia Básica

4.1.1 Topologia

Na fase inicial foram utilizados 3 PCs e 1 PT-Server, dispostos ao redor de um PT-Switch e configurados como mostra a Figura 4 e Tabela 1:

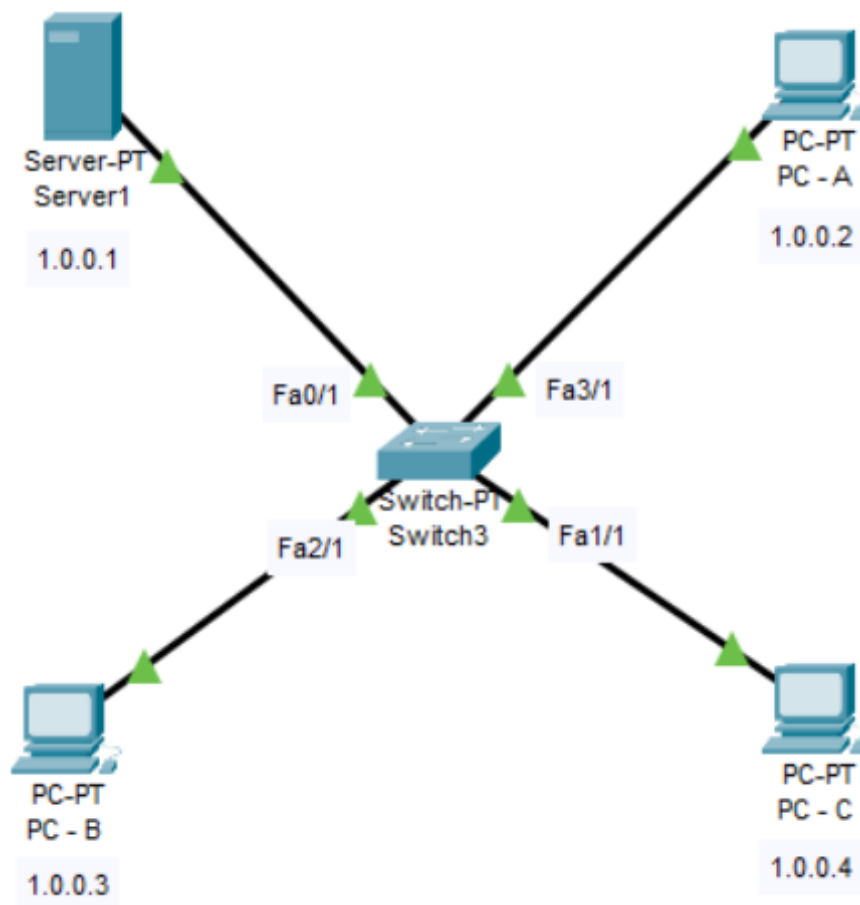


Figura 4: Topologia dos dispositivos na Fase 1 da aplicação.

Tabela 1: Configuração base dos dispositivos na Fase 1 da aplicação.

Dispositivo	Endereço IPv4	Máscara de Subrede
Server1	1.0.0.1	255.0.0.0
PC - A	1.0.0.2	255.0.0.0
PC - B	1.0.0.3	255.0.0.0
PC - C	1.0.0.4	255.0.0.0

4.1.2 Implementação

Em seguida, o Firewall IPv4 do servidor foi ligado e configurado de forma que negue o protocolo ICMP de origem 0.0.0.0 e máscara coringa de 255.255.255.255 e permita o protocolo IP com os mesmos parâmetros.

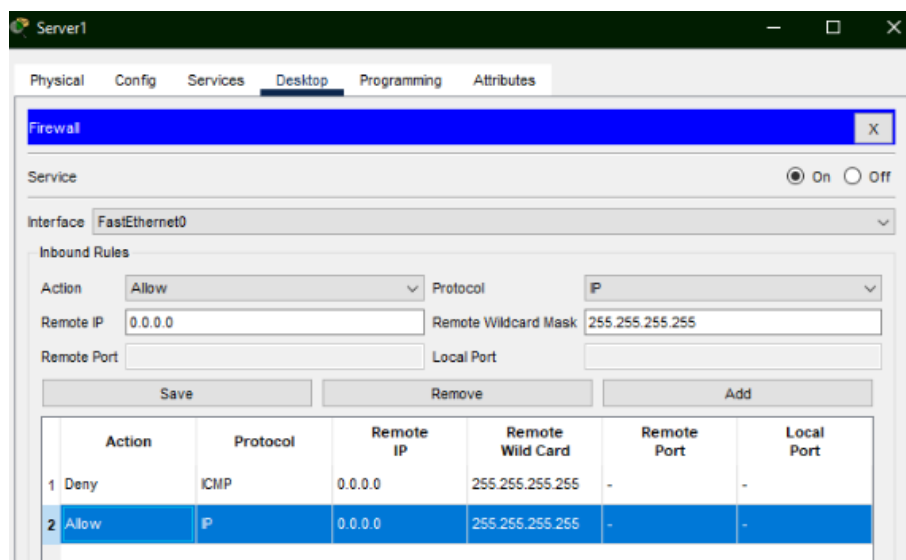


Figura 5: Configuração das políticas do Firewall na Fase 1.

4.1.3 Resultados

Como resultado, ao executar o comando ping 1.0.0.1 em qualquer um dos computadores da aplicação ele falha (da “Request timed out”). Isso ocorre pois a regra de negação do protocolo ICMP está funcional. Paralelamente a isso, ao acessar via navegador web o mesmo servidor, o processo é bem-sucedido, já que a regra imposta pelo firewall permite o protocolo IP.

4.2. 2ª Fase: Topologia Avançada

4.2.1 Topologia

A topologia da fase 2 da aplicação consistiu em uma rede segmentada com duas sub-redes (LAN interna e DMZ) interconectadas por um roteador. Os dispositivos utilizados

foram: 2 PCs conectados a um Switch 2950-24 (formando a LAN interna), 1 Server-PT conectado a outro Switch 2950-24 (formando a zona desmilitarizada) e 1 Router 2911 com duas interfaces G0/0 (LAN) e G0/1 (DMZ). O firewall foi implementado no roteador utilizando Listas de Controle de Acesso (ACLs) estendidas, prática mais comum em cenários do mundo real.

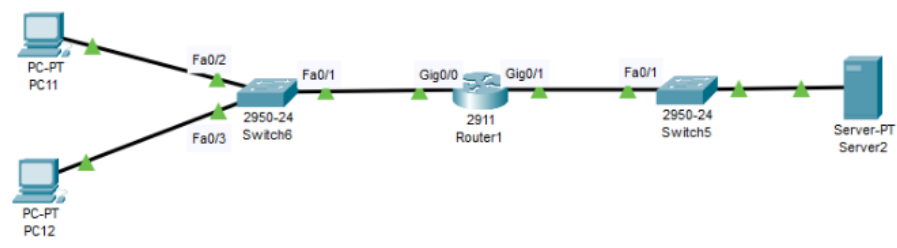


Figura 6: Topologia dos dispositivos na Fase 2 da aplicação.

Tabela 2: Configuração base dos dispositivos na Fase 2 da aplicação.

Dispositivo	Interface	Endereço IPv4	Máscara	Gateway
Router1	G0/0	10.0.0.1	255.255.255.0	N/A
Router1	G0/1	192.168.1.1	255.255.255.0	N/A
PC11	NIC	10.0.0.10	255.255.255.0	10.0.0.1
PC12	NIC	10.0.0.11	255.255.255.0	10.0.0.1
Server2	NIC	192.168.1.10	255.255.255.0	192.168.1.1

4.2.2 Implementação

Após a montagem da topologia física e a configuração dos endereços IP conforme mostra a Tabela 2, as interfaces do roteador foram configuradas via CLI com os comandos:

Listing 1: Configuração das interfaces do roteador

```
Router> enable
Router# configure terminal
Router(config)# interface GigabitEthernet0/0
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface GigabitEthernet0/1
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# no shutdown
Router(config-if)# end
Router# write memory
```

Além disso, no Server 2, o serviço HTTP foi ativado para simular um servidor web real.

Para implementação do Firewall a política de segurança estabelecida foi: permitir que a LAN gerencie o roteador, permitir acesso HTTP da LAN para a DMZ e negar

explicitamente todo resto do tráfego. Para isso, foi criada uma ACL estendida digitando os seguintes comandos na CLI do roteador:

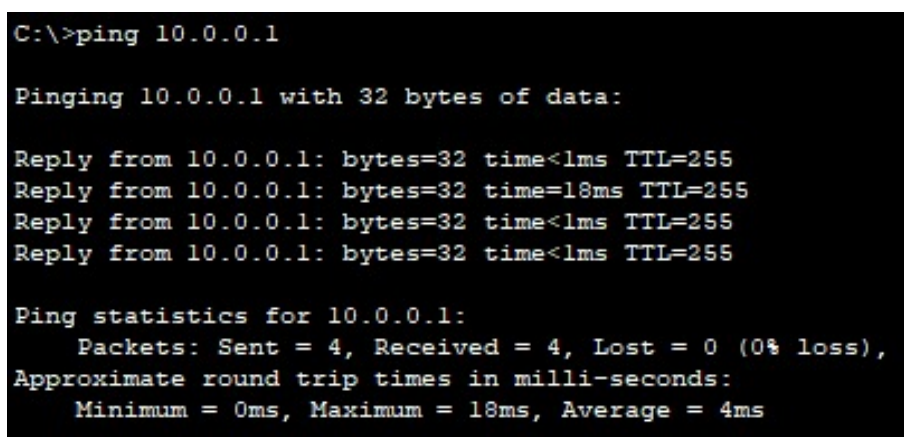
Listing 2: Configuração da ACL para implementação do firewall

```
Router> enable
Router# configure terminal
Router(config)# access-list 100 permit ip 10.0.0.0 0.0.0.255
host 10.0.0.1
Router(config)# access-list 100 permit tcp 10.0.0.0 0.0.0.255
any eq www
Router(config)# access-list 100 deny ip any any
Router(config)# interface GigabitEthernet0/0
Router(config-if)# ip access-group 100 in
Router(config-if)# end
Router# write memory
```

Dessa forma, a ACL foi aplicada como filtro de entrada na LAN.

4.2.3 Resultados

As configurações foram validadas através de uma série de testes para verificar a eficácia da política de segurança. No início, foi executado no PC 11 o comando `ping 10.0.0.1`, que resultou em sucesso, como mostra a Figura 7.



```
C:\>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time=18ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255
Reply from 10.0.0.1: bytes=32 time<1ms TTL=255

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 18ms, Average = 4ms
```

Figura 7: Sucesso no teste 1 da Fase 2.

Isso mostrou que o pacote ICMP correspondeu à primeira regra da ACL (`permit ip 10.0.0.0 0.0.0.255 host 10.0.0.1`), que permite explicitamente qualquer tráfego da rede LAN para o IP específico da interface do roteador.

O próximo teste consistiu na tentativa de acessar o endereço `http://192.168.1.10` no Web Browser no mesmo dispositivo, resultando em sucesso novamente, como mostra a Figura 8.

Isso ocorreu pois o pacote TCP de origem 10.0.0.10 (PC 11) com destino à porta 80 do servidor 192.168.1.10 correspondeu perfeitamente à segunda regra da ACL feita

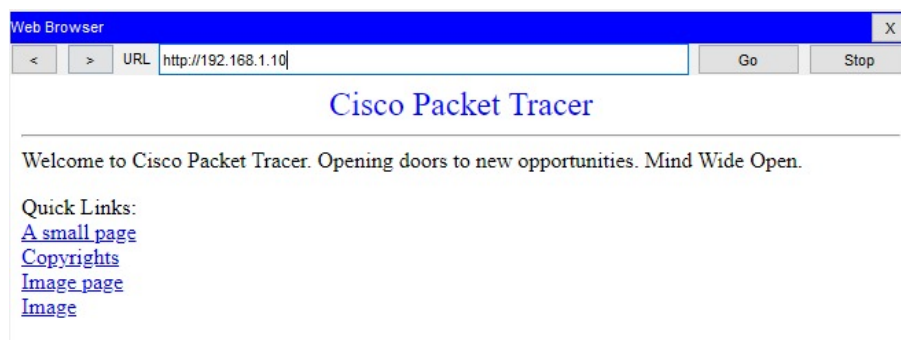


Figura 8: Sucesso no teste 2 da Fase 2.

(`permit tcp 10.0.0.0 0.0.0.255 any eq www`), que foi criada justamente para autorizar esse tipo de tráfego.

O teste 3 foi feito executando o comando `ping 192.168.1.10` novamente no PC 11 resultando em falha, como mostra a Figura 9.

```
C:\>ping 192.168.1.10

Pinging 192.168.1.10 with 32 bytes of data:

Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.

Ping statistics for 192.168.1.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Figura 9: Falha no teste 3 da Fase 2.

O comportamento era previsto, visto que o pacote ICMP não correspondia a nenhuma das regras de permissão (não era para o roteador nem era TCP porta 80). Portanto, foi desconsiderado pela regra `deny ip any any`, que implementa a política de “negar por padrão”.

O último teste foi realizar o comando `telnet 192.168.1.10 22` na tentativa de conexão SSH. O resultado foi falha, pois embora o pacote TCP se originasse na rede LAN, seu destino era a porta 22 (SSH). Como a ACL só permite a porta 80 (WWW), o pacote não correspondia à regra de permissão e foi negado pela regra `deny ip any any`. Isso demonstra granularidade da filtragem baseada em portas.

4.3. 3ª Fase: Implementação Física

4.3.1 Motivação e objetivos

Para complementar a segunda parte da simulação e enriquecer o estudo, buscamos implementar uma aplicação prática utilizando hardwares reais, não apenas simulações. Nessa simulação, agora via hardware, buscamos fazer uma implementação de um firewall via

terminal de comando do Linux. Uma vez que, por padrão, há a presença do iptables nos sistemas base Linux, o iptables é uma ferramenta de linha de comando usada para configurar filtros de pacotes.

Nossa simulação faz correlação com a parte 2 da simulação feita no Cisco Packet Tracer. Agora, faremos o uso do firewall para bloquear uma URL específica.

4.3.2 Definição da plataforma de execução

A aplicação consiste, primeiramente, em conseguirmos uma máquina que rodasse o Linux. Inicialmente, consideraríamos rodar uma VM (máquina virtual) apenas para teste. Para entender e compreender, acreditamos que não haveria complicações. Porém, para uma prática mais realista, queríamos de fato rodar uma aplicação Linux com controle total sobre os elementos do computador.

No momento, uma opção seria instalar o Linux diretamente no dispositivo de armazenamento. Porém, devido à inviabilidade de excluir o sistema operacional atual, resolvemos rodar a distro Linux diretamente no pendrive. Esta solução possui limitações de desempenho devido à transferência de dados, mas oferece uma visão mais precisa de como o sistema funcionaria na prática.

4.3.3 Sistema operacional

Quanto ao sistema escolhido, optamos pelo MiniOS por ser uma distro extremamente leve, com interface gráfica e, principalmente, focada em rodar em pendrives. Essa escolha visa evitar eventuais problemas que distros usuais como Ubuntu ou Debian poderiam trazer, mesmo que tais problemas fossem improváveis.

4.3.4 Configuração do ambiente de teste e execução

Após definir o sistema operacional, definimos o local de execução: um dos notebooks dos estudantes do grupo. É importante que este dispositivo possua uma placa de rede adequada para visualizarmos a filtragem que o firewall realizará.

Com o notebook em mãos, temos o núcleo da nossa simulação. Nele, utilizando o MiniOS (nossa distro Linux), realizaremos via terminal: primeiro, a conexão com a internet; segundo, a configuração do firewall para bloquear o acesso a uma URL específica.

Para verificarmos se a aplicação está funcional, realizaremos testes de ping dentro do próprio sistema operacional, de forma que se quisermos configurar um ponto de acesso, o computador com MiniOS funcionará como um roteador, comportando-se especificamente como um roteador com firewall bloqueando acesso a determinadas URLs.

Os comandos utilizados para a implementação, considerando que já se estava conectado à internet, foram:

Abaixo temos o processo realizado, em que como primeiro passo, na Figura 10, temos o uso do `nslookup` para conseguir obter o endereço utilizado pelo Facebook,

aplicação que será bloqueada. Com isso, obtemos o endereço 157.240.12.35.

Após isso, na Figura 11 é realizado o teste de `ping` para verificar se o sistema consegue se conectar com o Google, no qual o processo foi interrompido após 3 pings, utilizando o comando `Ctrl+C` para encerrar.

A Figura 12 é uma continuidade da anterior, mostrando o mesmo teste, porém agora com o `facebook.com`, que falha.

Na Figura 13, podemos observar que o navegador de internet não consegue acessar o Facebook; entretanto, consegue acessar o YouTube, mas não o WhatsApp. Isso ocorre porque, como pode ser visto na imagem, o WhatsApp possui o mesmo endereço do Facebook, que por sua vez difere do YouTube.

Por último, na Figura 14, temos o terminal de comando realizando novamente o `ping` com o Google, mostrando lado a lado um ping bem-sucedido e, em seguida, um ping para o Facebook em que o firewall bloqueia o acesso. O processo foi encerrado com `Ctrl+C`, e é possível observar que, em um curto espaço de tempo, houve 16 tentativas de conexão. Finalmente, foi realizado um último ping para o Facebook, porém agora estando desconectado da internet.

4.3.5 Resultados da Implementação Física

O processo realizado e seus resultados são documentados através das seguintes figuras:



```
Terminal - live@minios: ~
File Edit View Terminal Tabs Help

live@minios:~$ nslookup facebook.com
Server:      192.168.1.254
Address:     192.168.1.254#53

Non-authoritative answer:
Name:   facebook.com
Address: 157.240.12.35
Name:   facebook.com
Address: 2a03:2880:f105:283:face:b00c:0:25de

live@minios:~$ sudo iptables -I OUTPUT -d 157.240.0.0/16 -j DROP
live@minios:~$
```

Figura 10: Obtenção do endereço por meio do `nslookup` e uso do `iptables` para bloquear a faixa de endereços.

```
Terminal - live@minios: ~
File Edit View Terminal Tabs Help
live@minios:~$ ping google.com
PING google.com (142.251.133.14) 56(84) bytes of data.
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=1 ttl=117 time=22.5 ms
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=2 ttl=117 time=18.9 ms
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=3 ttl=117 time=19.1 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 18.873/20.177/22.525/1.663 ms
live@minios:~$
```

Figura 11: Verificação da conexão por meio do ping com o Google.

```
Terminal - live@minios: ~
File Edit View Terminal Tabs Help
live@minios:~$ ping google.com
PING google.com (142.251.133.14) 56(84) bytes of data.
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=1 ttl=117 time=22.5 ms
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=2 ttl=117 time=18.9 ms
64 bytes from pngrua-bv-in-f14.1e100.net (142.251.133.14): icmp_seq=3 ttl=117 time=19.1 ms
^C
--- google.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 18.873/20.177/22.525/1.663 ms
live@minios:~$ ping facebook.com
PING facebook.com (157.240.12.35) 56(84) bytes of data.
^C
--- facebook.com ping statistics ---
16 packets transmitted, 0 received, 100% packet loss, time 15350ms
live@minios:~$
```

Figura 12: Teste de conectividade via ping com o Facebook, falhando devido ao bloqueio.

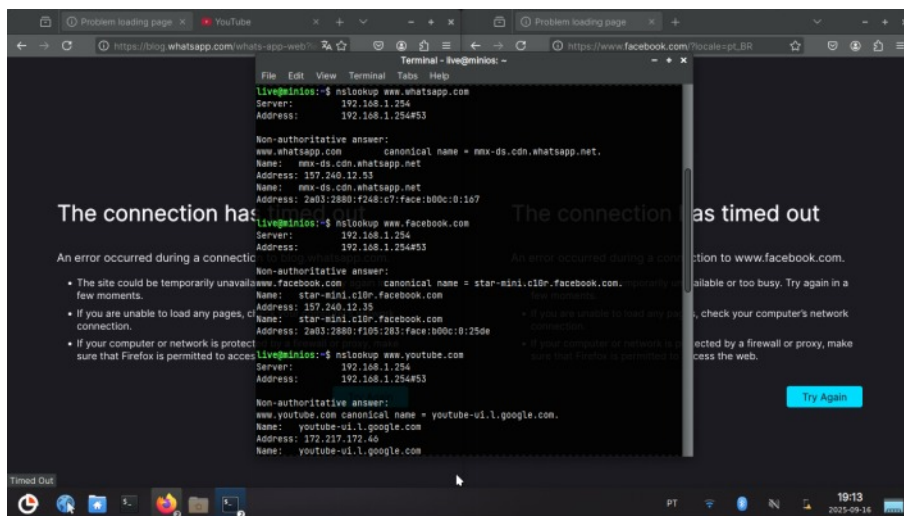


Figura 13: Acesso via navegador: Google e YouTube funcionando, Facebook e WhatsApp bloqueados.

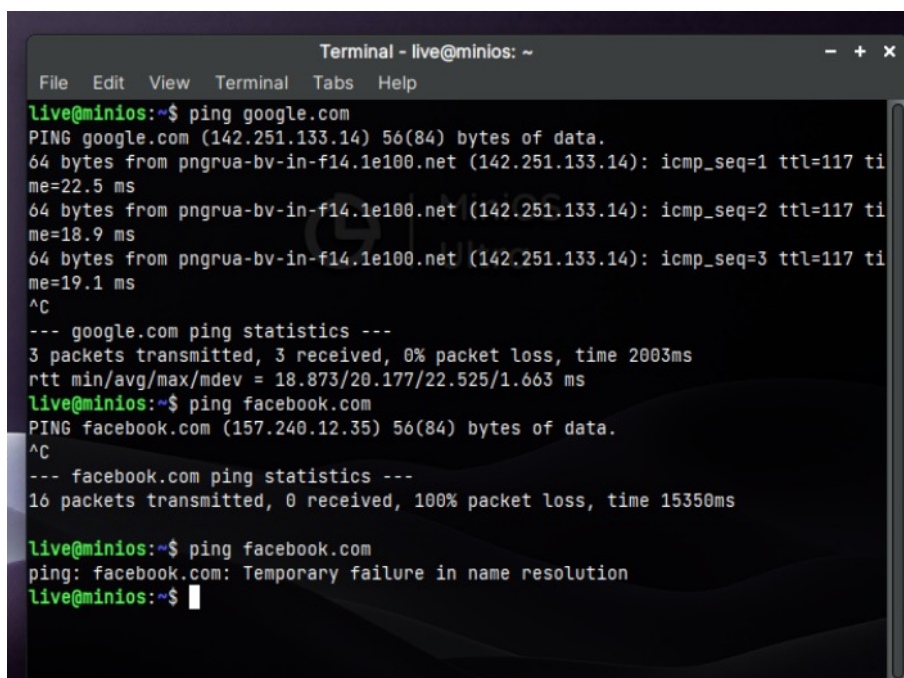


Figura 14: Comparação de pings: Google bem-sucedido e Facebook bloqueado pelo firewall.

5. Conclusão

Este estudo evidenciou a importância fundamental dos firewalls como instrumentos indispensáveis para a segurança das redes. Ao explorar os conceitos teóricos das diversas arquiteturas, desde os filtros de pacotes convencionais até gateways de aplicação, e sua implementação prática em contextos controlados, foi possível comprovar a eficácia dessas tecnologias na proteção da infraestrutura de rede.

A simulação realizada no Cisco Packet Tracer demonstrou de maneira clara como a segmentação de rede, particularmente por meio da utilização de uma DMZ, estabelece camadas extras de segurança, reduzindo os riscos de acesso não autorizado à rede interna.

A terceira fase, que envolveu a implementação de um firewall em um ambiente Linux real, confirmou a aplicação dos conceitos teóricos em um cenário prático, demonstrando que a elaboração de regras de filtragem é uma habilidade essencial para a segurança cibernética.

Em síntese, os resultados obtidos destacam que um firewall é um sistema que exige configuração e gerenciamento cuidadoso. A partir disso é possível concluir que a combinação de diferentes arquiteturas e a compreensão profunda de suas limitações são vitais para construir uma defesa robusta contra as ameaças digitais em constante evolução.

Referências

- [1] Kurose, J. F. and Ross, K. W. (2021). *Computer Networking: A Top-Down Approach*. 8th edition. Pearson Education.
- [2] Tanenbaum, A. S. and Wetherall, D. J. (2020). *Computer Networks*. 6th edition. Pearson Education.
- [3] Stallings, W. (2020). *Network Security Essentials: Applications and Standards*. 6th edition. Pearson Education.
- [4] GeeksforGeeks. (2024). *Introduction of Firewall in Computer Network*. Disponível em: <https://www.geeksforgeeks.org/computer-networks/introduction-of-firewall-in-computer-network/>
- [5] Cisco Systems. (2005). *Network Security Technologies and Solutions*. Disponível em: <http://ptgmedia.pearsoncmg.com/images/9781587204104/samplepages/158720410X.pdf>. Acesso em: 16/09/2025.