ACADEMIA MILITAR DAS AGULHAS NEGRAS
ACADEMIA REAL MILITAR (1811)
CURSO DE CIÊNCIAS MILITARES

Fábio Ribeiro Rodrigues Junior

PROCEDIMENTOS OPERACIONAIS PADRÃO PARA ENDURECIMENTO DE SISTEMAS OPERACIONAIS LINUX (HARDENING)

Resende
2019

Fábio Ribeiro Rodrigues Junior

PROCEDIMENTOS OPERACIONAIS PADRÃO PARA ENDURECIMENTO DE SISTEMAS OPERACIONAIS LINUX (HARDENING)

Monografia apresentada ao Curso de Graduação em Ciências Militares, da Academia Militar das Agulhas Negras (AMAN, RJ), como requisito parcial para obtenção do título de **Bacharel em Ciências Militares.**

Orientador: Ten Cel Inf Nivio Paula de Souza

Resende
2019

Fábio Ribeiro Rodrigues Junior

PROCEDIMENTOS OPERACIONAIS PADRÃO PARA ENDURECIMENTO DE SISTEMAS OPERACIONAIS LINUX (HARDENING)

Monografia apresentada ao Curso de Graduação em Ciências Militares, da Academia Militar das Agulhas Negras (AMAN, RJ), como requisito parcial para obtenção do título de **Bacharel em Ciências Militares.**

Aprovado em ____ de _____ de 2019:

Banca examinadora:

_____

Nívio Paula de Souza**, Tenente-coronel**
(Presidente/Orientador)

_____

Marco Antônio da Silva**, Coronel R1**

_____

Antônio Fernando Pires **Patury** Junior**, Major**

Resende
2019

Dedico esse trabalho, primeiramente a Deus, meu refúgio bem presente nos momentos de angústia, que me guiou por todos esses anos me dando força, paciência e sabedoria por essa jornada de 5 anos rumo ao oficialato. Dedico também aos meus pais, que com seu amor e afeto sem fim, sempre me aconselhando qual o melhor caminho a seguir por todos os bons e maus momentos. Por fim dedico a minha esposa, companheira e ouvinte desde antes mesmo da preparatória.

AGRADECIMENTOS

Agradeço primeiramente a Deus, meu refúgio bem presente nos momentos de angústia, que me guiou por todos esses anos me dando força, paciência e sabedoria por essa jornada de 5 anos rumo ao oficialato. Tudo que conquistei até o momento foi por que Ele nunca me abandonou.

Agradeço também a minha família, principalmente meus pais, que com amor e afeto sem fim, sempre me aconselhando qual o melhor caminho a seguir por todos os bons e maus momentos, meus irmãos com seu amor alegrando os meus dias.

Agradeço ao meu orientador, por todo o esforço e comprometimento com o trabalho, com seu grande conhecimento na área ampliou meus horizontes para o campo de T.I., mostrando as infinitas possibilidades que se pode seguir. O trabalho não teria sido feito sem a orientação dele.

RESUMO

PROCEDIMENTOS OPERACIONAIS PADRÃO PARA ENDURECIMENTO DE SISTEMAS OPERACIONAIS LINUX (HARDENING)

AUTOR: Fábio Ribeiro Rodrigues Junior
ORIENTADOR: Ten Cel Inf Nivio Paula de Souza

Com a expansão em ritmo acelerado dos conflitos no âmbito do ciberespaço, medidas de proteção da informação são cada vez mais necessárias. A informação nada mais é do que um processo de organização de dados que de maneira geral se relaciona com indivíduos e sistemas por meio de inúmeros canais de comunicação. Na atualidade, a internet é o principal canal por onde trafega esse conhecimento. Em sua grande parte, essa informação detém um alto valor, podendo ser um ativo financeiro, segredos comerciais de grandes corporações ou informações pessoais. Em decorrência da expansão do fluxo de dados e de seu valor, surge o questionamento de como proteger contra problemas de invasões de sistemas, vazamento e furto de dados por indivíduos mal-intencionados. Uma forma de proteger os ativos desses ataques é aplicando uma técnica conhecida como *hardening* em tradução literal "endurecimento". Consiste basicamente na blindagem do sistema operacional aplicando técnicas específicas com o objetivo de torná-lo mais seguro, através também da redução de suas possíveis vulnerabilidades. A adoção da solução livre, ou aberta, é considerada definitiva para todo o Exército Brasileiro. Portanto, a obtenção do índice máximo de sua utilização deve ser um objetivo permanente para todas as Unidades do Exército, em opção à solução fechada, sem ônus à plena operacionalidade das atividades específicas da OM. O sistema operacional Linux é a solução livre citada pelo plano de migração. Por ser aberta, não tem dependência tecnológica com nenhuma empresa, fomenta o desenvolvimento de conhecimento, elimina a necessidade de troca do sistema operacional em decorrência de sua obsolescência, entre outras vantagens. A pesquisa tem o propósito de demonstrar as técnicas de "hardening" que podem ser empregadas em *desktops* nas organizações militares do Exército Brasileiro, evidenciando assim a importância da aplicação da "blindagem" nos sistemas. Dessa maneira, se evita que ocorra possíveis vazamentos de informação decorrentes de invasões a computadores. A produção de um documento padrão, que operacionaliza esse endurecimento auxilia na adoção de medidas que visam à proteção do sistema, tendo em vista que ele será utilizado por militares sem um conhecimento técnico específico.

Palavras-chave: Segurança. Blindagem. Software livre. Linux. *Hardening*.

ABSTRACT

AUTHOR: Fábio Ribeiro Rodrigues Junior
ADVISOR: Ten Cel Inf Nivio Paula de Souza

The rapid expansion of cyberspace conflicts, information protection measures are increasingly needed. Information is nothing more than a process of data organization that in general relates to individuals and systems through numerous channels of communication. Nowadays, the internet is the main channel through which this knowledge travels. For the most part, this information holds a high value, and can be a financial asset, trade secrets of large corporations or personal information. As a result of the expansion of data flow and its value, the question arises as to how to protect against problems of system intrusion, data leakage and theft by malicious individuals. One way to protect assets from these attacks is by applying a technique known as hardening in literal translation "hardening." It basically consists of the shielding of the operating system applying specific techniques in order to make it safer, also reducing its possible vulnerabilities. The adoption of the free or open solution is considered definitive for the entire Brazilian Army. Therefore, obtaining the maximum utilization rate should be a permanent objective for all Army Units, in option to the closed solution, without burden to the full operation of the specific OM activities. The Linux operating system is the free solution cited by the migration plan. Because it is open, it has no technological dependence on any company, it fosters the development of knowledge, eliminates the need to change the operating system due to its obsolescence, among other advantages. The research has the purpose of demonstrating the hardening techniques that can be used in desktops in the military organizations of the Brazilian Army, thus evidencing the importance of the application of the "shielding" in the systems. In this way, it prevents the occurrence of possible leaks of information resulting from computer intrusions. The production of a standard document, which operates this hardening, assists in the adoption of measures aimed at protecting the system, since it will be used by military personnel without specific technical knowledge.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

AMAN Academia Militar das Agulhas Negras
POP Procedimento Operacional Padrão
ABNT Associação Brasileira de Normas Técnicas
ISO *International Organization of Standardization*
IEC *International Electrotechnical Commission*
DNS *Domain Name System*
BSD *Berkeley Software Distribution*
GPL *General Public license*
OSI *Open System Interconnection*

Sumário

1 INTRODUÇÃO

Apesar de ser um fator que muitas vezes passa despercebido pelo usuário final, a segurança no ciberespaço é uma variável importante a se considerar. A preocupação com os dados pessoais, a privacidade e o sigilo deve ser uma preocupação equivalente às medidas que se tomam nas residências.

Os sistemas operacionais que utilizamos para navegar pela internet, digitar textos, editar conteúdos, assistir filmes entre outras atividades. Normalmente traz consigo um pacote básico de segurança que é muitas vezes configurado apenas quando se instala pela primeira e posteriormente é ignorado pelo usuário.

A técnica de *hardening*, que se traduz em algo como endurecer, reforçar ou blindar; como seu nome sugere, implementa medidas que impeça ou até mesmo se anule qualquer possibilidade que invasão do Sistema.

As Organizações Militares como um todo utilizam o meio informatizado para todo tipo de atividade pelas infovias. Muitas vezes circulam documentos de caráter sigiloso por meio delas, dessa maneira a exposição a perigos de vazamento de informações ou até mesmo roubo de segredos de estado, ficam a mercê de indivíduos mal-intencionados.

Para a implementação da técnica de blindagem do sistema, será através da geração de um Procedimento Operacional Padrão, ou apenas pela sigla POP, nada mais é do que um passo a passo que como será aplicado. Exemplificando a partir de imagens para que o usuário ou até mesmo o técnico que detenha esse documento possa aplicá-lo de maneira correta.

O POP será aplicado em uma máquina que utiliza o kernel Linux como base. Isso se deve ao fato de que o Exército Brasileiro segue a diretriz do “Plano de Migração para o *software* livre”, com isso o mais adequado e preciso é utilizar seguindo os parâmetros estabelecidos no documento em questão.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Analisar a viabilidade da aplicação da técnica de *Hardening* em distribuições do Sistema Linux, com a geração de Procedimento Operacional Padrão (POP). De maneira que o usuário final utilize o procedimento em computadores de Organizações Militares do Exército Brasileiro.

1.1.2 Objetivos específicos

Conceituar *hardening*, definindo os conceitos básicos, restringindo o escopo dos processos existentes para o Sistema Operacional Linux que poderão ser utilizados para a "blindagem" do sistema.

Selecionar a distribuição Linux que será aplicada o processo de *hardening*, inserido no plano de migração para software livre no Exército Brasileiro, para o desenvolvimento do Procedimento Operacional Padrão a ser utilizado.

Gerar e descrever de maneira detalhada, obedecendo critérios técnicos e observando normas e legislação das áreas pertinentes, na criação do Procedimento Operacional Padrão necessário para a realização do procedimento de endurecimento do sistema.

2 REFERENCIAL TEÓRICO

2.1 DISTRIBUIÇÃO LINUX

Primeiramente devemos conceituar o que é *software* livre. O *software* livre é o *software* em se que respeita a liberdade e senso comunitário de seus dos usuários, isso significa que eles possuem a liberdade de executar, copiar, distribuir, estudar, mudar e fazer melhoramentos no software. Desse modo, o "*software* livre" é uma questão primordialmente de liberdade, não de preço. Para entender esse conceito, devemos pensar em algo como "liberdade de expressão". O acesso ao código-fonte é uma condição necessária para o *software* livre. Na prática, código-fonte é um texto, com sua própria sintaxe e utiliza letras, números e caracteres especiais, exigindo ou não um editor especial.

As principais licenças para *software* livres são o GPL e BSD. A vantagem do BSD é que ela é mais livre que o GPL, porque garante praticamente a mesma liberdade. O GPL impõe certas restrições que fazem com que seja incompatível com muitos outros software livres, inclusive aqueles com licenças similares. A sua vantagem é que a obrigatoriedade de que *softwares* derivados sejam licenciados sobre GPL fomentando o crescimento do *software* livre. Apesar de o BSD ter uma liberdade maior, pode-se argumentar que a GPL tem maior liberdade no sentido em que garante a liberdade nos trabalhos derivados. Por causa desse conceito de "liberdade" do *copyleft.*

> Exemplificando de maneira específica, Linux é apenas um núcleo (ou *kernel*), a parte primordial do *software* que faz a intermediação entre o hardware e os programas, controlando ambos. Uma "distribuição Linux" é um sistema operacional completo, que normalmente inclui o kernel do Linux, um programa para instalação e, o fundamental, aplicativos e outros *softwares* necessários para transformar um computador em uma ferramenta realmente útil de trabalho (Hertzog, 2015; Mas, 2015).

As principais distribuições linux no mercado são: Linux Mint, Ubuntu, Debian, Manjaro entre outras. Distribuição Linux é um sistema operacional criado a partir de uma coleção de software, com o uso do núcleo Linux, um sistema gestor de pacotes, e um repositório de programas. Na maioria das distribuições Linux, a maior parte do software disponível em seus repositórios é livre e de código aberto, estando disponíveis na forma de pacotes previamente compilados (binários), e em código-fonte. Na maior parte dos casos, as distribuições Linux

utilizam bibliotecas e utilidades criadas pela GNU. Há distribuições Linux para uma variedade de casos de uso, desde sistemas embarcados, computadores pessoais, para supercomputadores.

> A maioria das distribuições Linux são financiadas por empresas com fins lucrativos. O Linux reuniu um volume significativo de cobertura da mídia ao longo dos anos e isto beneficia principalmente as distribuições apoiadas por um departamento de marketing real – em outras palavras, para distribuições baseadas em empresas (Hertzog, 2015; Mas, 2015).

2.2 PROCEDIMENTO OPERACIONAL PADRÃO (POPs)

Os Procedimentos Operacionais Padrão (POPs) são documentos essenciais para a execução de qualquer tarefa realizada com qualidade, obedecendo critérios técnicos e observando normas e legislação. Eles Servem de veículo para que as informações acerca dos mais diversos processos cheguem com segurança ao executor.

Deve conter as instruções sequenciais das operações e a frequência de execução, especificando o responsável pela execução, listagem dos equipamentos; peças e materiais utilizados na tarefa, descrição dos procedimentos da tarefa por atividades críticas.

> Convém que os procedimentos de operação sejam documentados, mantidos atualizados e disponíveis a todos os usuários que deles necessitem. Importa que os procedimentos documentados sejam preparados para as atividades de sistemas associados a recursos de processamento e comunicação de informações (ABNT NBR ISO/IEC 27002, 2005, p.40).

Os POPs para as atividades de sistemas devem ser tratados como documentos formais e as mudanças devem ser autorizadas pela autoridade competente. Quando tecnicamente possível, convém que os sistemas de informação sejam gerenciados de maneira uniforme, usando os mesmos procedimentos e ferramentas.

2.3 HARDENING

A palavra *hardening* significa algo como “endurecimento”, mas no contexto da Segurança cibernética é o processo de proteger um sistema através da redução de possíveis vulnerabilidades, por meio de configurações que implementam controles específicos. Esse processo acarreta em realizar várias configurações, instalações de pacote e procedimentos de segurança e modificações com o objetivo de melhorar e reforçar a segurança do ambiente.

Se considerar a principal tradução de *hardening*, que significa “endurecer”, é exatamente isso o que se deseja fazer com um Sistema Operacional. Essa ideia fica mais clara se for considerada a tradução como “fortalecimento”, pode ser explicado como um conjunto de configurações e melhoramentos, ajustes finos que vão gerar controles para que o sistema se torne mais seguro.

O administrador de redes deve analisar muito bem essas grandezas e encontrar um estado de harmonia entre elas, levando o sistema a uma alta produtividade e segurança, pois quanto maior a segurança menor o risco e também a flexibilidade. É fundamental que haja um estudo completo do cenário e serviços em questão.

Inicialmente recomenda-se sempre instalar versões atuais do sistema operacional, que contenham correções e *patches* de segurança pois será um grande risco de segurança cibernética utilizar uma versão antiga sem atualizações, deixando o sistema temporariamente vulnerável no caso da existência de pacotes com falhas. Serviços críticos como web, e-mail e DNS devem estar sempre nas versões mais atuais. Softwares desnecessários devem ser desinstalados e pacotes inseguros devem ser substituídos por alternativas mais confiáveis.

3 REFERENCIAL METODOLÓGICO

3.1 TIPOS DE PESQUISA

Foi realizada uma pesquisa com a metodologia explicativo, devido ao fato do POP descrever de maneira elucidativa um passo a passo para se alcançar o objetivo proposto. A abordagem foi de maneira qualitativa, de maneira que utilizamos dos dados provenientes da técnica de *hardening*.

3.2 MÉTODOS

3.2.1 Escolha da técnica de "*hardening*"

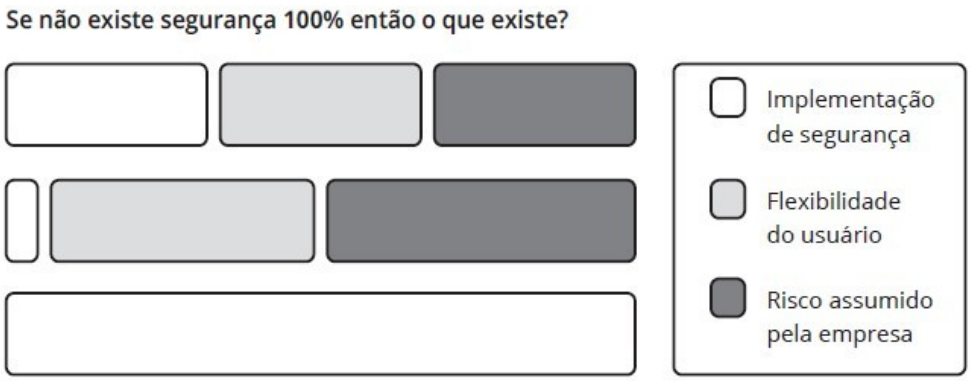
Deve-se lembrar que o conjunto de técnicas e ferramentas para implementação de um *hardening*, normalmente utilizadas em no Linux sejam interessantes do ponto de vista de garantia dos serviços e realmente agreguem à segurança. Não será incomum o fato de que um conjunto de ações pré-estabelecidos em uma *baseline* não se apliquem em todas as situações. Cada caso é um caso. É preciso analisar e descobrir que controles serão mais adequados para a necessidade em questão e também qualificar o quanto de segurança foi estabelecido no respectivo contexto (Melo, 2014, p. 05).

Dessa maneira, é conveniente lembrar que ferramentas são importantes, mas não são o fim e sim o meio para a execução dos procedimentos do processo, que, somados a outras ações, como a capacitação de pessoal envolvido, é fator determinante de sucesso, pois a imperícia é um inimigo de qualquer trabalho. Explicando de um ponto de vista prático, esse processo envolve: A remoção ou desativação de serviços desnecessários, Remoção de contas de usuários-padrão, Desinstalação de pacotes desnecessários, Definição do processo de atualização, Definição de

controles para auditoria, Definir controles para limites do uso dos recursos pelas aplicações e/ou usuários, Instalação de pacotes de ferramentas de segurança e auditoria.

No momento que se inicia a implementação das técnicas de hardening, é necessário pensar em três fatores: segurança, risco e flexibilidade, como ilustrado na figura a seguir:

Figura 1 – Segurança, flexibilidade e risco



Fonte: Melo (2014, p. 4)

A questão é saber balancear esses três fatores para definir um conjunto de controles que possam proporcionar ao sistema um equilíbrio. De maneira geral perguntas são levantadas durante o processo, como por exemplo: O nível de segurança que se deseja ou a conformidade do sistema quanto as políticas de segurança definidas e etc.

Pode se perceber que os fatores são inversamente proporcionais, assim, os fatores como “segurança” e “flexibilidade” ou “risco” e “segurança” tem impacto direto no risco. Não se pode ter 100% de segurança mas, quanto mais segurança, menores serão o risco e a flexibilidade. Já que não existe a possibilidade de se ter 100% de segurança, deve-se criar o maior número de controles para tornar mais seguro. Assim, o hardening permite atenuar possíveis vulnerabilidades.

3.2.1.1 Análise do ambiente

Após o planejamento da instalação e sua execução, o administrador do sistema vai iniciar a execução do *hardening*. Uma boa técnica tem como princípio “menor recurso e menor

privilégio". Um bom exemplo da utilização desse princípio é checar a lista de pacotes instalados, pois serão identificados possíveis pacotes que não necessários para a utilização do servidor ou aplicações que terão a necessidade de ter a permissão revista.

Figura 2 – Verificação de pacotes instalados

```
# dpkg -l | awk '{print $2,$3}' | sed '1,7d'


gravando o resultado em um arquivo

# dpkg -l | awk '{print $2,$3}' | sed '1,7d' > /root/pacotes
```

Fonte: MELO (2014, p. 6)

A remoção de programas que não são úteis e podem ser usados por um *exploit* (código que explora a vulnerabilidade de um programa para ganhar um acesso não autorizado ao sistema) para um ataque local ou remoto, dependendo do programa.

Exemplos de programas que podem ser não necessariamente importantes, dependendo do tipo de servidor: (Melo, 2014, p. 06)

- **lynx:** cliente http/ftp que possibilita transferência de *malware*;
- **wget:** cliente *http/ftp* que possibilita transferência de *malware*;
- **netcat (nc):** é um "canivete suíço" que possibilita transferência de *malware* ou até mesmo criar *backdoors*;
- **hping:** montador de pacotes que possibilita criar *backdoors* via *rawsocket*.

Muitos outros aplicativos podem entrar nesta lista, tais como: telnet, cliente ftp, rshd, rlogind, rwhod, ftpd, sendmail, tcpdump, nmap, pois, devido aos recursos que proporcionam, devem ser devidamente avaliados. (Fagundes, 2017, p. 07)

3.2.1.2 Controles de Segurança para Contas de Usuários

A criação de usuários e grupos no Linux é importante para definir quais conteúdos podem ser acessados, por quais usuários e/ou grupos e ainda permite ao Sistema Operacional gerenciar

os processos de cada usuário. Monitorar o que os usuários estão fazendo é uma parte fundamental para o gerenciamento. O Linux possui vários módulos que aumentam a segurança e a administração sobre seus usuários. Administradores inexperientes de maneira geral preparam seus servidores com uma instalação básica e nenhum procedimento é feito para manter a estrutura do sistema.

3.2.1.2.1 Controles de autenticação com a utilização do PAM

Módulos de Autenticação Plugáveis (PAM) são uma interface de módulo que corresponde ao tipo de autorização baseada em um módulo. Um módulo PAM pode usar apenas uma ou todas as quatro interfaces possíveis. Cada uma delas será especificada no arquivo de configuração de um serviço, se for apropriada para o módulo e o administrador desejar usar essa interface. Essas interfaces são:

- ***account:*** essa interface verifica se uma conta tem permissão para usar o sistema, o que pode significar a verificação da existência, se está vencida ou se tem autorização de acesso em determinado horário, vinculada a um grupo ou através de determinado serviço;
- ***auth:*** interface que serve para autenticar usuários. Isso pode ocorrer através de senhas, banco de dados ou outro mecanismo. Além disso, esses módulos também têm permissão para definir credenciais, como membros de grupo ou mesmo baseados em tokens Kerberos;
- ***password:*** a interface *password* é usada para verificar e definir a autenticação de senha ou o critério de definição de senha;
- ***session:*** é responsável pela configuração e gerenciamento de sessões de usuário. Pode incluir tarefas de organização, como montar diretórios, criar arquivos etc.

3.2.1.2.2 Procura por senhas fracas

A senha do login de um usuário é algo muito pessoal, até mesmo os próprios administradores não devem saber as senhas dos usuários. De maneira geral, quando as contas são cadastradas em um sistema, ou quando o cadastro é feito com uma senha inicial, tem de ser trocada no primeiro login feito pelos usuários.

Se não existem controles que definem o critério para ter uma senha forte, acaba criando a oportunidade para o usuário digitar uma senha fraca, que pode ser totalmente composta por números e com poucos dígitos, palavras existentes em dicionários de *bruteforce* e até mesmo uma data simbólica. Essas são senhas muito fáceis de serem descobertas, às vezes nem sendo necessário um programa de Força Bruta para serem quebradas.

Para realizar um trabalho de busca de senhas fracas, pode ser interessante usar uma ferramenta da mesma categoria e funcionalidade utilizada por um *cracker*. No Linux, a ferramenta recomendada para essa tarefa é o *John the Ripper*.

3.2.1.2.3 Comando “sudo”

O comando “sudo” permite aos usuários executar comandos como outro usuário. Uma vantagem em usar o sudo é a de poder conceder aos usuários definidos acesso restrito, embora privilegiado, a programas como Super Usuário. Dessa forma o uso do comando “sudo” acaba sendo uma alternativa interessante ao uso da permissão especial de suid bit. Entre as vantagens do uso do “sudo”, pode se destacar:

- A possibilidade de tornar a conta *root* uma conta compartilhada, normalmente em um ambiente onde existam múltiplos administradores com difícil controle do acesso *root*, existe a necessidade de várias pessoas executarem atividades que demandam recursos administrativos é um complicador. Isso pode motivar algo não recomendado. Todavia, nem todos que necessitam executar recursos administrativos precisam ter acesso *root*;
- Ser configurado de modo que os membros de um grupo designado não precise de autenticação adicional para executar um determinado comando como *root*, o que resulta em maior produtividade. Frequentemente se argumenta em lista de segurança que o uso do sudo pode ser um problema de segurança, pois a configuração errada do sudo fatalmente traz risco de conformidade com uma política de segurança. Um bom exemplo é o fato de o programa sudo. No entanto, deve ser avaliado pela equipe de segurança deve-se ou não ser permitida essa “produtividade”;

3.2.1.3 Registro de eventos (log)

Podemos analisar algumas das configurações-padrão dos sistemas Linux relacionadas ao histórico (trilha) de comandos. Por exemplo, é possível usar o comando "*history"* para visualizar o histórico do usuário que estamos utilizando.

Existem algumas variáveis de ambiente importantes, entre elas a "HISTFILE", que armazena a localidade do arquivo de histórico do usuário – que pode ser visualizado para que tenhamos a certeza da localidade do arquivo de histórico (trilha) de comandos.

3.2.1.3.1 Registro de eventos – Logs do sistema

Tem a função de registrar as atividades dos usuários e serviços do sistema:

- De grande importância para os administradores.
- A administração se torna muito mais auditável e detalhada quando se tem controle sobre todos os logs que o Linux pode oferecer.

Logs:

- Ajudam a descobrir se no sistema houve algum acesso proibido, registra as tentativas que podem sugerir uma possibilidade de invasão ou mesmo intrusões.

As distribuições Linux já trazem habilitada grande parte dos *logs* necessários a uma administração, ou seja, muitas das atividades que acontecem já são registradas pelo sistema. Entretanto, se recomenda refinar os controles de *logs* para que seja possível ter registro ainda mais detalhado e organizado.

3.2.1.3.2 Conformidade com as recomendações

O registro de eventos é uma necessidade de normas de segurança como NBR IEC 27002 e PCI DSS, entre outros. Normas e documentos de segurança dedicam vários tópicos do quão importante são os *logs*.

> Convém que registros *(log)* de auditoria contendo atividades dos usuários, exceções e outros eventos de segurança da informação sejam produzidos e mantidos por um período de tempo acordado para auxiliar em futuras investigações e monitoramento de controle de acesso. (ABNT NBR ISO/IEC 17799, 2005, p. 63)

Assim, é prioridade adotar uma política de segurança na qual os registros devam atender a características como:

- Identificação dos usuários;
- Datas e horários de entrada (login e logout);
- Identidade do terminal, nome da máquina ou IP;
- Registro das tentativas de acesso aos aceitos e rejeitados;
- Registro das tentativas de acesso a outros recursos e dados aceitos e rejeitados;
- Alteração de arquivos;
- Uso de privilégios, aplicativos e utilitários do sistema.

3.2.1.4 Proxy web

A função básica de um *proxy* na rede é servir como um intermediário, entre a conexão de um cliente e um servidor. Normalmente, entre uma rede local e um serviço de rede na internet, existe a possibilidade de criação de *proxy* para vários tipos de protocolo, como smtp, pop3, http, entre outros. O servidor *proxy Squid* tem a função de *proxy* web, ou seja, mantém um cache de conteúdo web acessado de todos os clientes web de uma determinada rede.

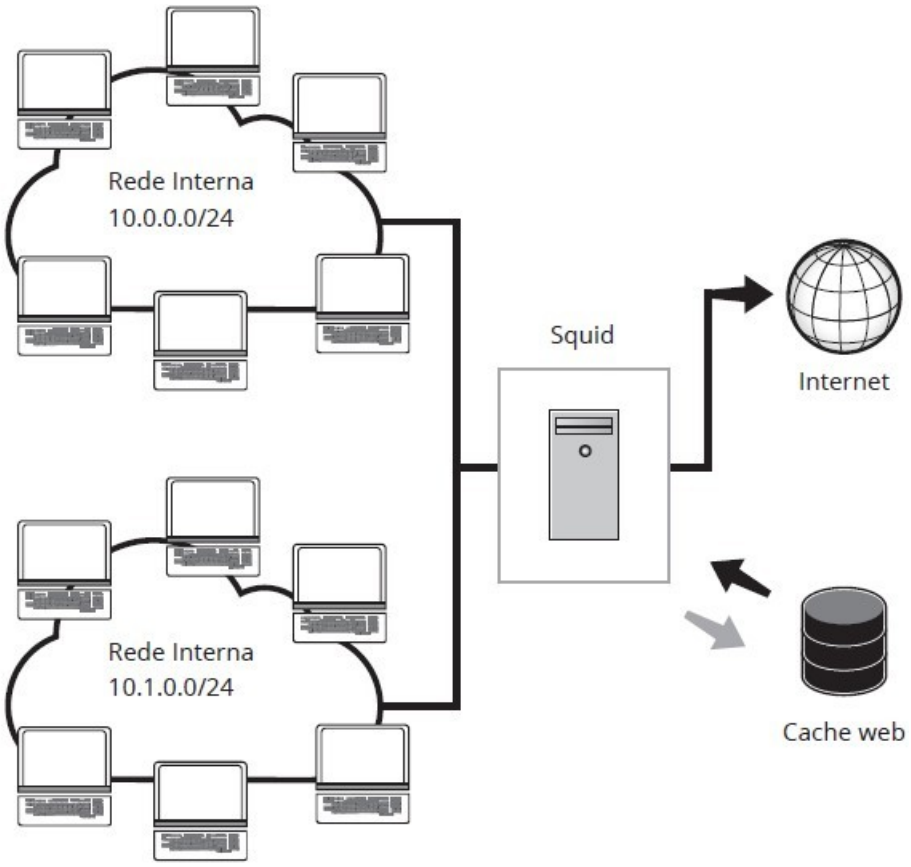
> O *Squid* é uma solução de software livre para *proxy* cache para a Web com suporte a protocolos HTTP, HTTPS e FTP. A implementação desse servidor em uma rede possibilita reduzir a largura de banda e melhora os tempos de resposta fazendo cache e reutilização de páginas da web frequentemente acessadas. Permite também a definição de controles de acesso amplos e pode ser um acelerador web. Existem versões para diversos Sistemas Operacionais, incluindo Windows, e está licenciado sob a GNU GPL. (Melo, 2014, p. 132)

O *Squid*, como servidor *proxy* cache, funciona de forma direta, quando uma solicitação de estação na rede solicitar um mesmo conteúdo web, ele será fornecido por meio da estrutura de cache. Para o funcionamento adequado, recomenda-se definir regras de Firewall no perímetro da rede que redirecione conexões web para a porta do servidor *proxy squid* utilizado, que normalmente é a porta padrão 3128. O uso de servidor *proxy web* agrega valor à segurança da rede, pois a única máquina que fará conexão web diretamente para a internet é o servidor proxy, e os clientes da rede se comunicam diretamente com ele.

Entretanto, se o conteúdo não tiver sido acessado antes e consequentemente não estiver contido no cache, então o servidor vai baixar esse conteúdo para o seu cache. Assim, para que lá exista uma cópia para uma consulta futura proveniente dos clientes, reduzindo assim o tempo de acesso e o consumo de banda para uso de internet.

Controle de conteúdo acessado na web:

Figura 3 – Exemplificação do controle de conteúdo acessado



Fonte: MELO (2014, p. 133)

3.2.1.4.1 Registro de atividades

No que diz respeito a segurança de rede, o uso de um proxy cache permite também que sejam gerados registros de atividades dos usuários e que esses registros sejam mantidos por

tempo definido, possibilitando investigações futuras e também o monitoramento do controle de acesso. Esses registros têm a seguinte estrutura de dados:

- **Data e horários de entrada:** são registrados por padrão;
- **Identidade do terminal e localização:** essa diretriz de recomendação não se aplica;
- **Identificação dos usuários:** por padrão se obtém o registro de IP de acesso pelo cliente; entretanto, com o *proxy* por meio da autenticação, são atreladas informações com o respectivo login do usuário;
- **Registro das tentativas de acesso:** esses dados são registrados por padrão;
- **Registro das tentativas de acesso a outros recursos e dados:** esses dados são registrados por padrão.

3.2.1.5 Firewall

Normalmente o elemento mais importante para a defesa do *host* contra os ataques virtuais é o *firewall*. Na maioria dos casos, o *firewall* é a primeira linha de defesa. Ele pode ajudar a proteger de três maneiras principais: lidando com a entrada e a saída de tráfego indesejado, manipulando com o login suspeitos e com o tráfego que tenha a intenção de causar danos.

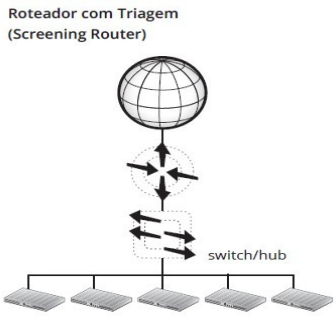
Sistemas de *firewalls* são importantes em um projeto de segurança; entretanto, somente ele não têm como garantir a segurança de uma rede de computadores. Tem a necessidade outros mecanismos como também um gerenciamento prévio. É um sistema de grande importância, há a recomendação a separação da rede, assim como a implementação de proteção dos serviços disponibilizados contra acessos não autorizados – um trabalho que é realizado pelos filtros *(firewall)*, para criar perímetros de redes devidamente protegidos e gerenciados.

3.2.1.5.1 Arquiteturas de firewall

Normalmente, as empresas preferem implementar um *firewall* baseado apenas em uma máquina, seja um *host* PC ou um roteador. Entretanto, os mais robustos são compostos por várias partes. Veja algumas arquiteturas a seguir:

- Um roteador com triagem (figura 4) é o tipo mais simples de firewall e usa apenas os recursos de filtrar pacotes para controlar e monitorar o tráfego da rede que passa pela fronteira. Em um servidor com filtragem de pacotes, esses roteadores podem bloquear o tráfego entre as redes ou, por exemplo, o tráfego destinado a ou proveniente de hosts específicos em um nível de porta IP. Em geral, a comunicação direta é permitida entre vários hosts na rede privada e na Internet.

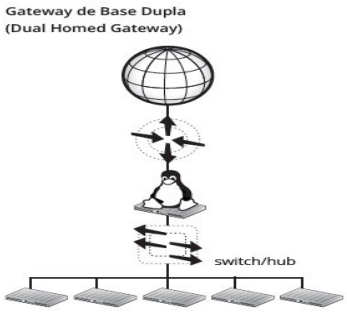
Figura 4 – Roteador com triagem



Fonte: MELO (2014, p. 150)

- No Gateway de base Dupla (figura 5) é colocado um computador com duas interfaces de rede entre as duas redes. Normalmente, o *gateway*, chamado de Bastion *host* ele tem um *proxy* de circuito para autenticar todo o processo da rede de uma empresa para a internet e filtra o acesso da Internet contra a rede da empresa.

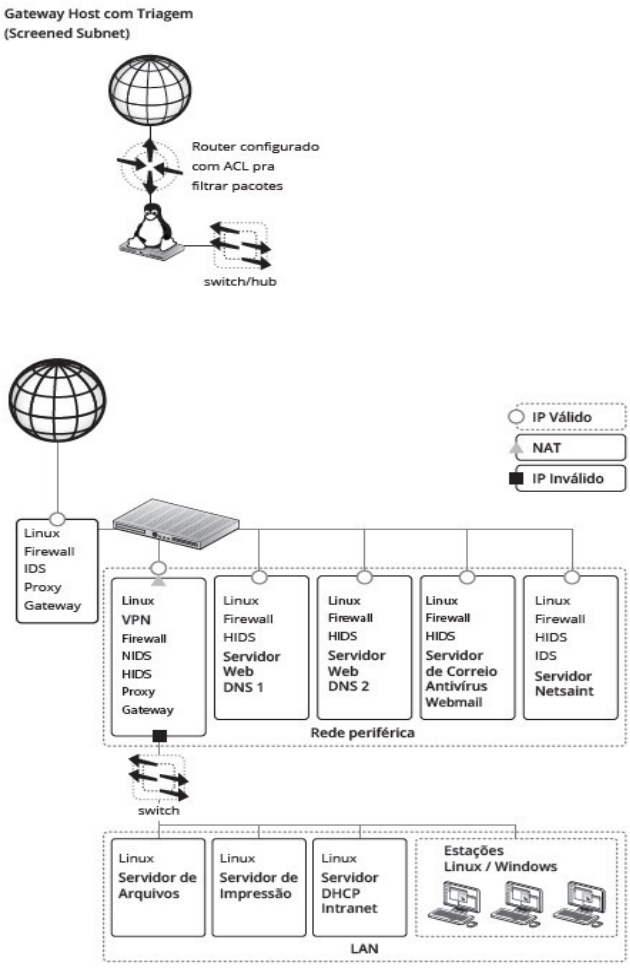
Figura 5 – Gateway de base dupla



Fonte: MELO (2014, p. 151)

- O Roteador e Gateway na figura 6, são usados de maneira conjunta em uma arquitetura, formando assim, duas camadas de proteção. A primeira camada, é a rede externa, que está interligada com a rede externa através de um roteador, nesta camada a rede só conta com o filtro de pacotes no roteador e tem como finalidade aceitar ou bloquear pacotes de rede através de regras definidas. A segunda camada, é a rede interna, e limita os acessos neste ponto é um Bastion Host, pois nele, tem um outro filtro de pacotes além de mecanismos de autenticação da própria rede interna.

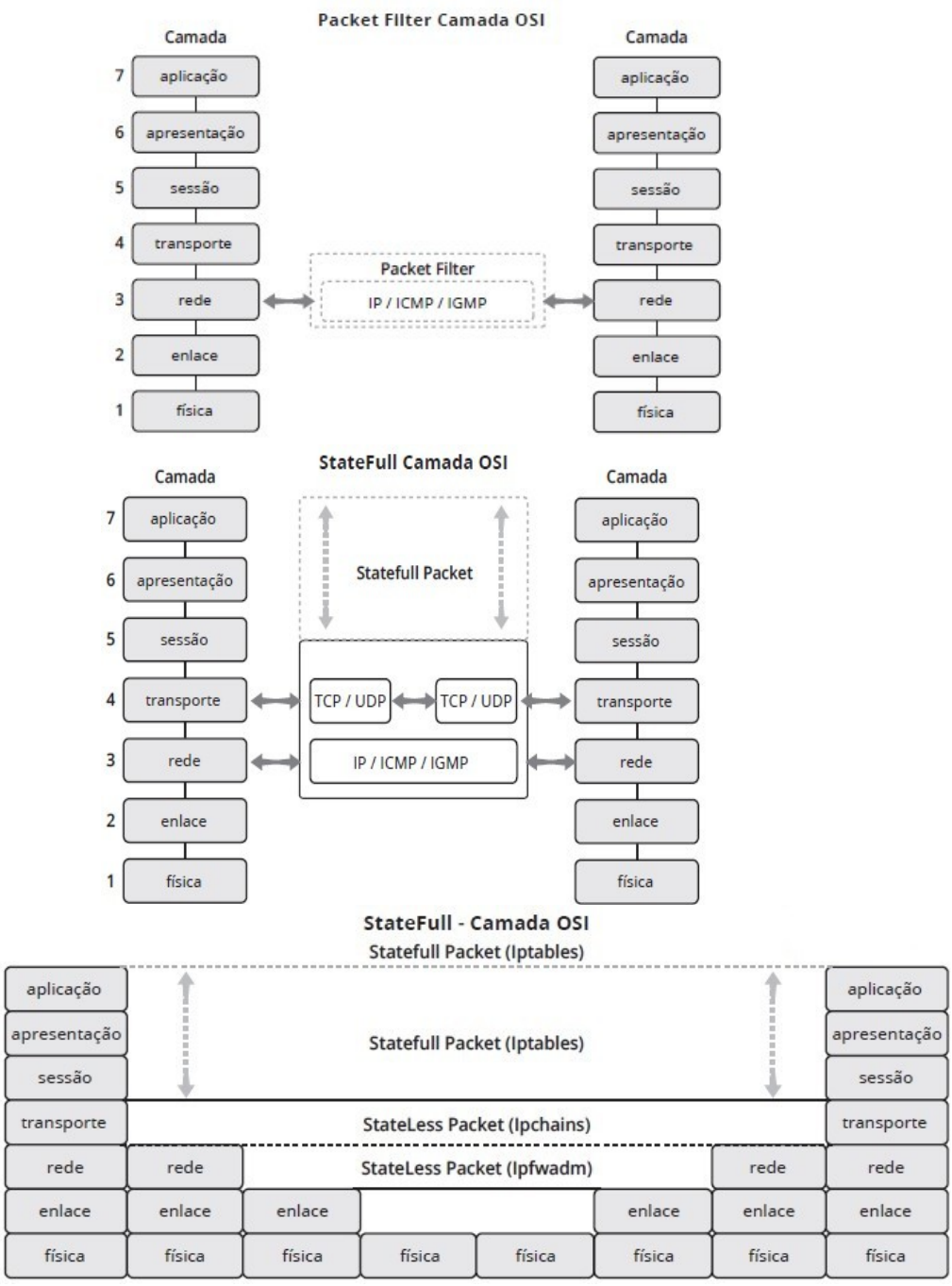
Figura 6 – Gateway Host com Triagem



Fonte: MELO (2014, p. 153)

Vejamos agora os tipos de *firewalls* e exemplificar como eles trabalham na parte estrutural de camada OSI:

Figura 7 – *Firewalls* e a estrutura de camada OSI

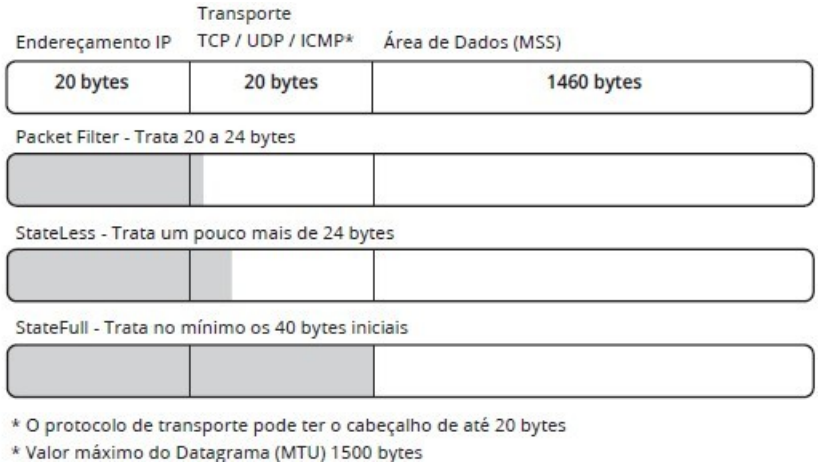


Fonte: MELO (2014, p. 154)

3.2.1.5.2 Datagrama

De uma forma direta e didática, pode-se classificar um firewall de acordo com seu nível de atuação em um datagrama, ou seja, o quanto em média do datagrama é efetivamente processado, ou quais os campos do respectivo cabeçalho podem ser relevantes para ação do firewall.

Figura 8 – O *firewall* e o datagrama



Fonte: MELO (2014, p. 154)

- **Packet Filter:** trata em média de 20 a 24 bytes iniciais de um pacote, ou seja, esse tipo de firewall trata e tem recursos para tratar todo o cabeçalho IP (os primeiros 20 bytes do pacote) e a parte do cabeçalho de transporte no que diz respeito à porta origem e porta destino. Lembrando que o campo “porta” tem 2 bytes (16 bits que correspondem a 2^16=65536 portas);
- **StateLess:** trata em média um pouco mais que 24 bytes iniciais de um pacote, ou seja, esse tipo de firewall seria um Packet Filter melhorado. Sendo relevante lembrar que esse tipo de firewall pode até ter um tratamento para algum campo específico do cabeçalho de transporte, mas não é capaz de tirar proveito do conceito de Estado de Conexão;
- **StateFull:** essa categoria de firewall trata no mínimo os 40 bytes iniciais de um datagrama, ou seja, todos o cabeçalho IP e, seja qual for o protocolo de transporte (UDP

ou TCP), são capazes de tratar o cabeçalho de Transporte. Sendo capaz de aproveitar as informações de estado de conexão do cabeçalho TCP.

3.2.2 Escolha da distribuição Linux

A parte central do sistema Linux é o seu kernel e também seu Sistema Operacional. Combinados, eles formam a base onde estão todas as suas aplicações que rodam no computador. O Linux e seu kernel são razoavelmente seguros. Um grande número de opções de segurança estão inclusas e uma grande variedade de ferramentas e configurações de segurança *open-source* já estão inclusas nas diversas distribuições. O Linux oferece excepcional controle sobre quem, como e quais recursos e aplicações os usuários podem ter acesso.

> Embora existam diversos Sistemas Operacionais Livres, o DCT (Departamento de Ciência e Tecnologia) recomenda, em ambiente *desktop*, distribuições derivadas do Debian (por exemplo, Kurumin, Ubuntu, Debian BR CDD e etc). Caso a Organização Militar opte pela solução Ubuntu, recomenda-se utilizar a última versão comprovadamente estável. (Plano de Migração para o Software Livre no Exército Brasileiro, p.22, 3ª Ed)

3.2.2.1 Ubuntu

O Ubuntu é um sistema de *software* livre. Este termo se origina de uma filosofia de origem africana, que fala sobre o significado da humanidade e o modo como vive. O sistema tem desenvolvimento comunitário e o produto pode ser compartilhado com qualquer pessoa. Quem desejar pode instalar gratuitamente o sistema operacional no computador, sem ter que pagar para o utilizar. O Ubuntu foi lançado em 2005, pela empresa Canonical.

Por ser um Sistema Operacional versátil é encontrado em diversos dispositivos, pode ser executado em servidores de rede, na sua versão Ubuntu *server*; em *smartphones* e *tablets* com o Ubuntu *touch*. Sua proposta é o oferecimento de um sistema que todas as pessoas possam usar sem dificuldades, não importando o país de origem, o nível de conhecimento a respeito do sistema ou até mesmo a limitação física do indivíduo.

Produzido pela Canonical, empresa britânica de software. Ela oferece atualizações gratuitas de segurança e suporte para as versões do Ubuntu, a partir da data de seu lançamento até

a data de seu fim de vida, essa data em questão é estabelecida pela empresa. Esse período varia, de 9 meses a partir do lançamento, até 5 anos nos casos das versões LTS (*Long Term Support*) que recebem suporte pelo período de 5 anos.

O objetivo do Ubuntu é ser seguro. Por padrão, os programas do usuário são executados com privilégios reduzidos. O *PolicyKit* também está sendo amplamente implementado no *desktop*. A maioria das portas de rede é fechada por padrão para evitar invasões. Um *firewall* vem integrado e este possui uma GUI disponível para configurá-lo, ele permite que usuários finais que instalam servidores de rede controlem o acesso. O Ubuntu compila seus pacotes usando recursos do GCC, como PIE, para proteger seu *software*.

3.2.3 Geração do procedimento operacional padrão

	PROCEDIMENTO OPERACIONAL PADRÃO - POP			Página 1 de 5
Código POP-CIBER-2019	Data de Emissão JAN/2019	Data de Vigência 31/DEZ/2019	Data de Revisão SET/2019	Versão nº. 01
ÁREA EMINENTE – SEÇÃO DE CIBERNÉTICA				
Assunto: Como configurar *firewall* na versão do Ubuntu 18.04 com o auxílio do UFW.				

OBJETIVO

Padronizar os Procedimentos Operacionais Padrão (POP) para a configuração do *firewall* no Sistema Operacional Ubuntu na versão 18.04.

APLICAÇÃO

Este procedimento operacional padrão (POP) se aplica a todo o usuário de sistemas baseados no kernel linux que contenham o UFW instalado.

DIVULGAÇÃO

Este POP é divulgado por meio da intranet das Organizações Militares do Exército Brasileiro para aqueles que tiverem interesse no método.

EMISSÃO, REVISÃO E APROVAÇÃO

Este POP foi:

- **Emitido por:**
- **Revisado por:**

- **Aprovado por:**

USUÁRIOS PRINCIPAIS

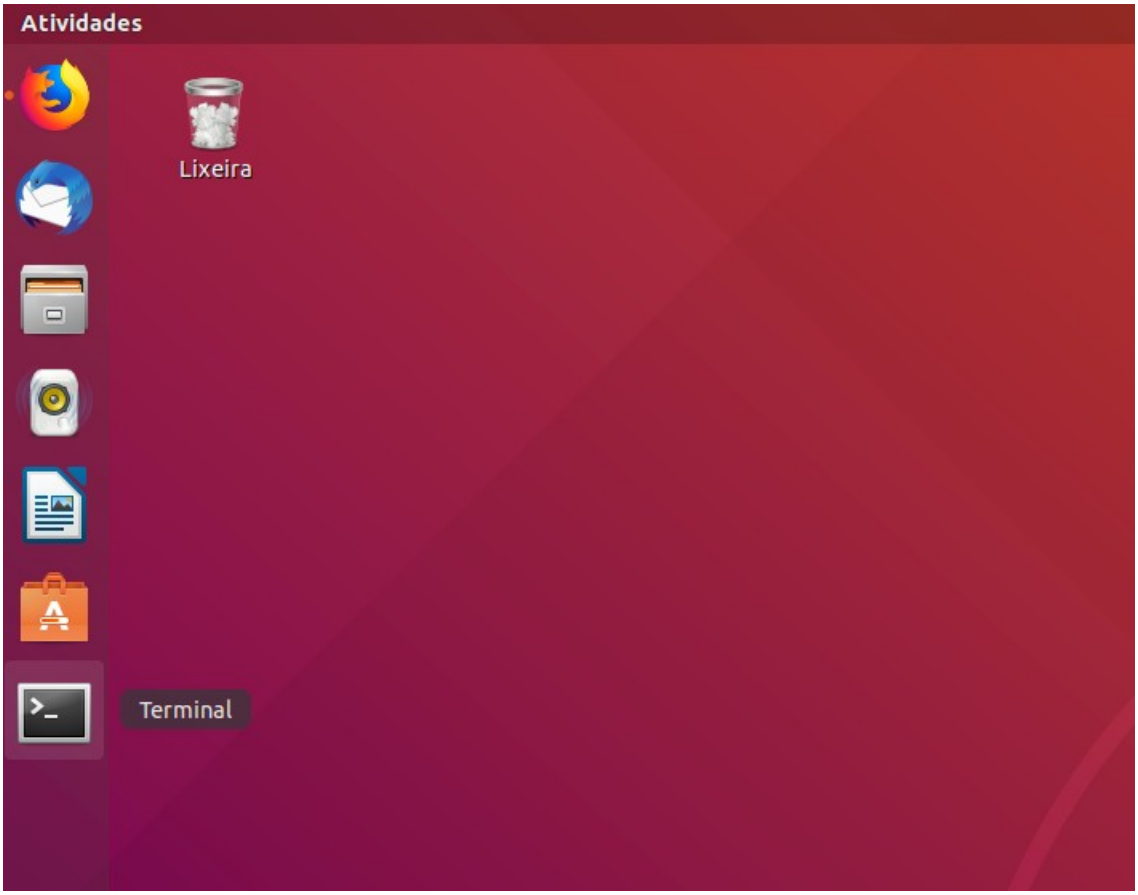
Acesso	Nome	Área

Procedimento operacional Padrão – Configuração do *firewall* com o auxílio UFW

Requisitos:

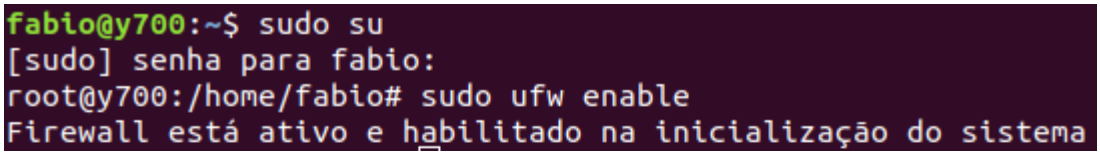
1. Ubuntu 18.04 LTS
2. UFW
3. Nmap
4. Iptables

Passo 1. Acessar o Terminal

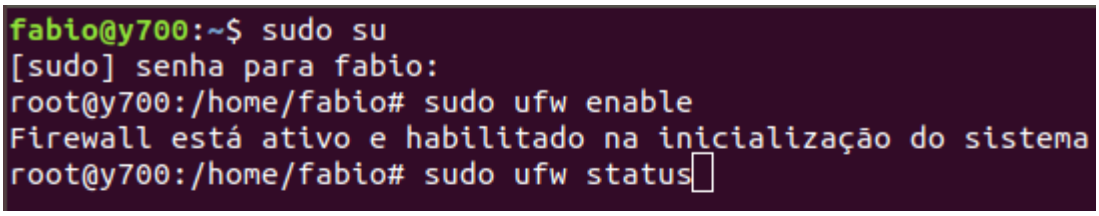


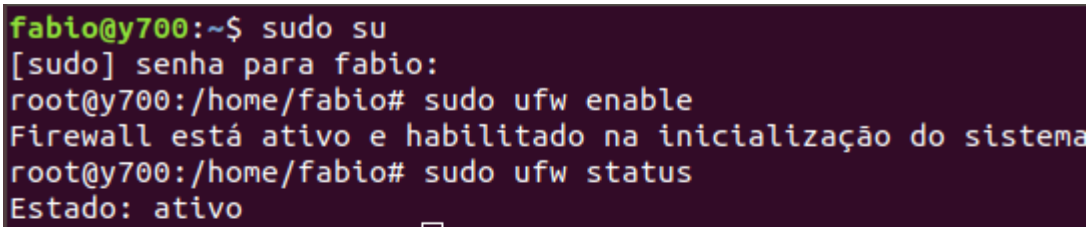
Passo 2. Acessar o modo de superusuário através do comando “sudo su”, colocar a senha se necessário.

Passo 3. Verificar se a função UFW (uncomplicated firewall) está habilitada por meio do comando “ufw enable”. Se não estiver, por meio desse comando será ativada a função.



Passo 4. Com o comando “ufw status”, verificar se *firewall* está ativo.





Passo 5. Permitir que porta 56 com o protocolo TCP seja utilizada pelo usuário.

```
fabio@y700:~$ sudo su
[sudo] senha para fabio:
root@y700:/home/fabio# sudo ufw enable
Firewall está ativo e habilitado na inicialização do sistema
root@y700:/home/fabio# sudo ufw status
Estado: ativo
root@y700:/home/fabio# sudo ufw allow 56/tcp
```

Passo 6. Negar que algum usuário utilize a porta 56 através do protocolo TCP.

```
fabio@y700:~$ sudo su
[sudo] senha para fabio:
root@y700:/home/fabio# sudo ufw enable
Firewall está ativo e habilitado na inicialização do sistema
root@y700:/home/fabio# sudo ufw status
Estado: ativo
root@y700:/home/fabio# sudo ufw allow 56/tcp
Regra adicionada
Regra adicionada (v6)
root@y700:/home/fabio# sudo ufw deny 56/tcp
```

Passo 7. Permitir o acesso de diversas portas ao mesmo tempo

```
fabio@y700:~$ sudo su
[sudo] senha para fabio:
root@y700:/home/fabio# sudo ufw enable
Firewall está ativo e habilitado na inicialização do sistema
root@y700:/home/fabio# sudo ufw status
Estado: ativo
root@y700:/home/fabio# sudo ufw allow 56/tcp
Regra adicionada
Regra adicionada (v6)
root@y700:/home/fabio# sudo ufw deny 56/tcp
Regra atualizada
Regra atualizada (v6)
root@y700:/home/fabio# sudo ufw allow 300:310/tcp
```

Passo 8. Verificar se todos os comandos que foram digitados foram implementados, com o comando “ufw status”.

```
fabio@y700:~$ sudo su
[sudo] senha para fabio:
root@y700:/home/fabio# sudo ufw enable
Firewall está ativo e habilitado na inicialização do sistema
root@y700:/home/fabio# sudo ufw status
Estado: ativo
root@y700:/home/fabio# sudo ufw allow 56/tcp
Regra adicionada
Regra adicionada (v6)
root@y700:/home/fabio# sudo ufw deny 56/tcp
Regra atualizada
Regra atualizada (v6)
root@y700:/home/fabio# sudo ufw allow 300:310/tcp
Regra adicionada
Regra adicionada (v6)
root@y700:/home/fabio# sudo ufw status
Estado: ativo

Para                       Ação        De
----                       ----        --
56/tcp                     DENY        Anywhere
300:310/tcp                ALLOW       Anywhere
56/tcp (v6)                DENY        Anywhere (v6)
300:310/tcp (v6)           ALLOW       Anywhere (v6)
```

4 RESULTADOS E DISCUSÕES

A parte central do sistema Linux é o seu kernel. Como foi demonstrado, o sistema operacional é razoavelmente seguro. Um grande número de opções de segurança estão inclusas no kernel acompanhado de uma variedade de ferramentas. Adicionalmente, o Linux oferece um grande controle sobre quem e quais recursos dos usuários podem ter acesso. Normalmente, o problema está nos detalhes, nas pequenas brechas. A segurança do sistema depende de uma quantidade de elementos de configuração em nível, tanto de Sistema Operacional quanto de aplicações.

O Sistema Operacional em si é bastante complexo, e sua correta configuração não é um processo trivial. É possuidor de inúmeras possibilidades de configurações, e cada ajuste pode causar sérios problemas. As vulnerabilidades e falhas de segurança não são fáceis de encontrar ainda mais por pessoas não aptas a trabalhar com o Linux. Para estar em condições é preciso adquirir sólidos conhecimentos sobre os requisitos básicos de segurança do Sistema Operacional e seu núcleo. Assim, a técnica de Hardening em Linux é um processo de proteção do seu kernel e suas aplicações contra ameaças conhecidas ou não, através da aplicação de técnicas específicas.

O hardening nada mais é que um processo de mapeamento das ameaças, reduzindo de seus riscos por meio da execução de atividades corretivas. Assim, o hardening é o processo de otimização de configurações da segurança de um sistema. No ponto de vista da Segurança Computacional também é o processo de proteger um sistema por meio da redução de suas possíveis vulnerabilidades. Esse processo implica realizar inúmeras configurações, instalações corretas de pacotes destinados a algum procedimento de segurança com o objetivo de melhorar e reforçar a segurança do ambiente.

Como a principal tradução de hardening é “endurecer-se”, é exatamente isso o que desejávamos desde o princípio. Essa ideia fica mais clara se for considerada a tradução como “fortalecimento”, que do ponto de vista experimental pode ser explicado como um conjunto de configurações e melhoramento, ajustes finos que vão gerar controles para que o sistema se torne mais seguro.

5 CONSIDERAÇÕES FINAIS

O tema analisado tem uma diversidade de técnicas muito extensa com uma complexidade relativamente alta, com isso escolher o método mais adequado dentro do escopo de aplicação em Organizações Militares demanda uma análise minuciosa.

A técnica de *hardening* não se limita a apenas uma, mas a uma diversidade de métodos que dependem do objetivo proposto por quem deseja implementar. Podendo abranger uma parte do sistema ou ele com um todo.

Restringir apenas para os Sistemas Operacionais baseados no kernel Linux se deve ao fato para quem esse documento se direciona e por qual motivo. Ele focado mas não me maneira exclusiva para as Organizações Militares. Assim por seguirem o “Plano de Migração para o Software Livre no Exército Brasileiro” é natural que se acabe enveredando para esse Sistema.

Os Procedimentos Operacionais Padrão, é um documento que deve ser redigido de forma acessível, de mameira quem obtiver o documento consiga compreender seu objetivo, obtendo o resultado esperado estabelecido por quem o redigiu.

Assim esse trabalho acadêmico acaba sendo, de alguma maneira, uma forma de elucidar na medida do possível a respeito de que blindagem de um sistema vem a ser. O tema necessita de um conhecimento prévio por quem deseja lê-lo, por causa de suas nomenclaturas e termos mas não é algo que impeça sua leitura.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR ISO/IEC 27002 – Tecnologia da informação – Técnicas de segurança – Código de prática para a gestão de segurança da informação.** Rio de Janeiro, p. 40. 2005

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **ABNT NBR ISO/IEC 17799 - Tecnologia da informação – Técnicas de segurança – Código de prática para a gestão da segurança da informação.** Rio de Janeiro, p. 63, 2005

Beraldo, R; Fontenelle, R. **O que é o software livre?** GNU, 2017. Disponível em: <https://www.gnu.org/philosophy/free-sw.pt-br.html>. Acesso em: 02 abr 2019

BRASIL. Plano de Migração para o Software Livre no Exército Brasileiro. Brasília, DF, p.22. 3ª Ed. 2007.

FAGUNDES, B. **Estudo da aplicação da técnica de hardening nos servidores web do hospital de clínicas de porto alegre.** Artigo apresentado como trabalho de conclusão de curso de Gestão de Segurança da Informação da Universidade do Sul de Santa Catarina, como requisito parcial para obtenção de título de Especialista. Santa Catarina, p. 7. 2017

HERTZOG, R.; MAS, R. **O Manual do Administrador Debian.** 1ª Ed: Freexian, 2015. p. 21

IGTI BLOG. Arquitetura da Informação na Segurança em T.I.
Disponível em: <http://igti.com.br/blog/arquitetura-da-informacao-na-seguranca-em-t-i/>. Acesso em: 19 abr 2019

Manual de Padronização – Coordenado pela Secretaria Geral. Brasília, DF. p. 07. 1ª Ed: 2014

MELO, Sandro. **Hardening em Linux.** Rio de Janeiro: Escola Superior de Redes, 2014. p. 02

SILVA, E. G. **Entenda o que é Hardening.** Viva o Linux, 2015. Disponível em: <https://www.vivaolinux.com.br/artigo/Entenda-o-que-e-Hardening>. Acesso em: 08 Mar. 2019.

SIGNIFICADOS. Significado de Ubuntu.
Disponível em: <https://www.significados.com.br/ubuntu/>. Acesso em: 31 mai 2019

WIKIPÉDIA. **Hardening.** Wikipédia, 2015. Disponível em: <https://pt.wikipedia.org/wiki/Hardening>. Acesso em: 08 Mar. 2019.

WIKIPÉDIA. **Hardening (inglês).** Wikipédia, 2018. Disponível em: <https://en.wikipedia.org/wiki/Hardening_(computing)>. Acesso em: 08 Mar. 2019.

WIKIPÉDIA. **Ubuntu.** Disponível em: <https://pt.wikipedia.org/wiki/Ubuntu>. Acesso em: 31 mai 2019

WIKIPÉDIA. **Procedimentos operacionais Padronizados.** Wikipédia, 2018. Disponível em: <https://pt.wikipedia.org/wiki/Procedimento_operacional_padrão>. Acesso em: 08 Mar. 2019.

WIKIPÉDIA. **Licenças BSD e GPL.** Disponível em: <https://pt.wikipedia.org/wiki/Licen%C3%A7as_BSD_e_GPL>. Acesso em: 02 abr 2019

WIKIPÉDIA. **GNU General Public License.** Disponível em: <https://pt.wikipedia.org/wiki/GNU_General_Public_License >. Acesso em: 02 abr 2019