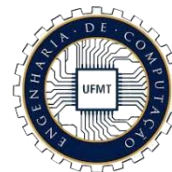




UNIVERSIDADE FEDERAL DE MATO GROSSO – UFMT
CAMPUS UNIVERSITÁRIO DE VÁRZEA GRANDE – CUVG
FACULDADE DE ENGENHARIA – FAENG
ENGENHARIA DA COMPUTAÇÃO – EC



GUILHERME DE OLIVEIRA TIBURCIO

**Implementação de um Servidor de Backup Inteligente com
Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e
BTRFS no Linux: Um Estudo de Caso**

Orientadora:

DR^a GRACYELI SANTOS SOUZA GUARIENTI

Cuiabá-MT
2023

GUILHERME DE OLIVEIRA TIBURCIO

**Implementação de um Servidor de Backup Inteligente com
Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e
BTRFS no Linux: Um Estudo de Caso**

Trabalho de Conclusão de Curso – TCC,
apresentado a FAENG/CUVG/UFMT (área de
concentração: Engenharia de Computação), no qual
foi elaborado em conjunto com a orientadora,
como parte dos requisitos básicos para a obtenção
do título de Bacharel em Engenharia de
Computação.

Orientadora: Dr^a Gracyeli Santos Souza Guarienti.

Cuiabá-MT

2023

Título Original

Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux:

Um Estudo de Caso

Primeira Publicação em

Cuiabá, Mato Grosso, Brasil.

1ª Edição

Todos os direitos da obra

Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux:

Um Estudo de Caso

reservados ao Autor.

Copyright do texto © Guilherme de Oliveira Tiburcio, 2023

Autorizo a reprodução e divulgação total ou parcial deste trabalho, por qualquer meio convencional ou eletrônico, para fins de estudo e pesquisa, desde que citada a fonte.

Universidade Federal de Mato Grosso (UFMT) / Biblioteca Central.

TIBURCIO, Guilherme de Oliveira. Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux: Um Estudo de Caso. 1ª Edição, UFMT, 2023. 59, p.

Dados Internacionais de Catalogação da Publicação na Fonte.

UFMT / Biblioteca Central

T554i Tiburcio, Guilherme de Oliveira.

Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux: Um Estudo de Caso. -- 1. Ed -- / Guilherme de Oliveira Tiburcio - Cuiabá, MT : Ed, do Autor 2023
59 p.

Orientadora: Gracyeli Santos Souza Guarienti

Coorientador: César Eduardo Guarienti

Trabalho de Conclusão de Curso (graduação) - Universidade Federal de Mato Grosso. Campus Universitário de Várzea Grande. Faculdade de Engenharia. Curso de Graduação em Engenharia de Computação, 2023.

1. BACKUP INTELIGENTE; 2. BACULA; 3. LINUX; 4. ARMAZENAMENTO EM NUVEM; 5. BTRFS.

I. Guarienti, Gracyeli Santos Souza, *orientadora*. II. Guarienti, César Eduardo, *coorientador*. III. Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux: Um Estudo de Caso.

MT/UF/BC

CDU 004.932(043.2)

**Implementação de um Servidor de Backup Inteligente com Capacidade de Armazenamento Local e na Nuvem utilizando Bacula e BTRFS no Linux:
Um Estudo de Caso**

Trabalho de Conclusão de Curso – TCC, apresentado a FAENG/CUVG/UFMT (área de concentração: Engenharia de Computação), no qual foi elaborado em conjunto com a orientadora, como parte dos requisitos básicos para a obtenção do título de Bacharel em Engenharia de Computação.

Guilherme de Oliveira Tiburcio

TCC aprovado em: _____ / _____ / _____ .

Dr. Fabrício Barbosa de Carvalho (Coord. Do Curso)

BANCA EXAMINADORA

Prof. Dr^a. Gracyeli Santos Souza Guarienti (Orientadora) FAENG/CUVG/UFMT

Prof. Dr. Diogo Henrique Duarte Bezerra FAENG/CUVG/UFMT

Prof. Dr Raoni Florentino da Silva Teixeira FAENG/CUVG/UFMT

“Dedico esse trabalho a todas as pessoas notáveis, enlouquecedoras, desafiadoras, até mesmo as frustrantes que me inspiraram para fazer as grandes coisas e me lembraram de que as estrelas estão acima de nós, mas o mundo a nossos pés, então, sigamos em frente.”

AGRADECIMENTOS

Agradeço, primeiramente a Deus, por ser a fonte de toda sabedoria. Por ter me dado energia e benefícios para ter concluído essa jornada acadêmica, e principalmente por ter me dado saúde, forças e inspirações para superar as dificuldades em minha vida.

A esta unidade da Universidade Federal do Mato Grosso (UFMT), ao seu corpo docente, direção e administração que me deram essa janela de oportunidade, no qual hoje vislumbro um horizonte superior; pelo aprendizado que obtive ao longo dos anos de estudo; eivado (infectado) pela acendrada (pura) confiança e ética aqui presentes.

À minha Orientadora Dr^a Gracyeli Santos Souza Guarienti, que auxiliou nesse trabalho com destreza, fazendo com que esse momento não fosse tão estressante e desgastante, você é simplesmente sensacional; meu Coorientador César Eduardo Guarienti pelo suporte. Ambos estiveram disponíveis para me ajudar e guiar durante a elaboração deste trabalho, a vocês expresso minha profunda gratidão. Suas orientações, correções, incentivos, direções, sugestões e críticas foram fundamentais para a construção de um trabalho de qualidade.

Aos meus pais e irmãos, pelo amor, incentivo e apoio incondicional. A eles expresso meu imenso amor e carinho, pois esta família, sempre esteve ao meu lado, me apoiando e incentivando em todas as etapas da minha jornada acadêmica. Sem o amor, a paciência e a compreensão de vocês, eu não teria chegado até aqui. Obrigado por serem a minha base sólida e por me ensinarem a nunca desistir dos meus sonhos. Este trabalho é dedicado a vocês com todo o meu amor. Vocês são a minha maior inspiração e motivação.

Aos meus colegas de academia, que mesmo indiretamente, me apoiaram e contribuíram sem saber para que esse trabalho se realizasse.

Enfim, agradeço a todos que direta ou indiretamente fizeram parte dessa etapa decisiva da minha vida, da minha formação, o meu muito obrigado.

RESUMO

Este trabalho apresenta a implementação de um servidor de backup inteligente utilizando Bacula, BTRFS e armazenamento em nuvem, com a integração do grub-btrfs. O objetivo é fornecer uma solução abrangente e eficiente para a proteção e recuperação de dados em ambientes Linux. O servidor de backup utiliza o Bacula como plataforma principal, aproveitando suas funcionalidades avançadas de backup e recuperação. O sistema de arquivos BTRFS é adotado para armazenar os dados de forma eficiente, utilizando recursos como snapshots incrementais. Além disso, a integração com serviços de armazenamento em nuvem expande as opções de proteção e disponibilidade dos dados. O estudo de caso realizado demonstra a eficácia da solução proposta, oferecendo insights valiosos para aprimorar o sistema e adaptá-lo a outros ambientes de trabalho.

Palavras-chave: Backup inteligente, Bacula, Linux, BTRFS, Armazenamento em Nuvem.

ABSTRACT

This paper presents the implementation of an intelligent backup server using Bacula, BTRFS, and cloud storage, with the integration of grub-btrfs. The objective is to provide a comprehensive and efficient solution for data protection and recovery in Linux environments. The backup server utilizes Bacula as the main platform, leveraging its advanced backup and recovery capabilities. The BTRFS file system is adopted to store data efficiently, utilizing features such as incremental snapshots. Furthermore, the integration with cloud storage services expands the options for data protection and availability. The conducted case study demonstrates the effectiveness of the proposed solution, providing valuable insights for system improvement and adaptation to other work environments.

Keywords: Intelligent backup, Bacula, Linux, BTRFS, Cloud Storage.

“Mas buscai primeiro o Reino de Deus, e
a sua Justiça, e as demais coisas vos
serão acrescentadas.”
(JESUS CRISTO)

SUMÁRIO

RESUMO	7
ABSTRACT	8
1 INTRODUÇÃO	13
1.1 OBJETIVOS GERAIS	15
1.2 OBJETIVOS ESPECÍFICOS	15
1.3 JUSTIFICATIVA	16
2 REVISÃO DE LITERATURA	17
2.1 CONCEITO DE BACKUP E RECUPERAÇÃO DE DADOS	17
2.2 BACULA: ARQUITETURA E FUNCIONALIDADES	18
2.3 SISTEMAS DE ARQUIVO LINUX	19
2.4 BTRFS: CARACTERÍSTICAS E VANTAGENS.....	22
2.5 AMBIENTES DE MICRO E PEQUENAS EMPRESAS	23
2.6 BACKUP INTELIGENTE	23
2.7 ARMAZENAMENTO EM NUVEM : BENEFÍCIOS E DESAFIOS.....	24
3 METODOLOGIA	28
3.1 DEFINIÇÃO DOS REQUISITOS DO SERVIDOR DE BACKUP	29
3.1.1 Capacidade de Armazenamento.....	29
3.1.2 Políticas de Backup e Recuperação.....	30
3.1.3 Recursos de Backup Inteligente	30
3.1.4 Requisitos de Hardware e Software.....	30
3.2 CONFIGURAÇÃO DO AMBIENTE LINUX COM BACULA E BTRFS	31
3.2.1 Instalação do Bacula.....	31
3.2.2 Configuração do Bacula	31
3.2.3 Configuração de Políticas de Backup e Recuperação.....	31
3.2.4 Testes da Solução.....	32
3.3 INTEGRAÇÃO COM SERVIÇOS DE ARMAZENAMENTO EM NUVEM.....	32
3.3.1 Seleção do Serviço de Armazenamento em Nuvem.....	32
3.3.2 Configuração da Integração	32
3.4 IMPLEMENTAÇÃO DO SERVIDOR DE BACKUP INTELIGENTE.....	33
3.4.1 Arquitetura do Sistema.....	33
3.4.2 Componentes de Hardware.....	33
3.4.3 Software Utilizado.....	33
3.4.4 Configuração do Ambiente	33
3.5 CRIAÇÃO DE UMA ESTRUTURA DADOS A SEREM RECOLHIDOS	34
4 ESTUDO DE CASO.....	35

4.1	PASSO A PASSO PARA MOLDAR O GERENCIADOR DE BACKUP.....	35
4.2	DESCRIÇÃO DO CENÁRIO TESTE	40
4.2.1	Requisitos mínimos para o Servidor de backup:	40
4.2.2	Requisitos mínimos para o Dispositivo de Armazenamento:.....	41
4.2.3	Requisitos mínimos para a Rede:.....	41
4.2.4	Verificar as configurações de firewall:.....	41
4.2.5	Teste a transferência de dados:	42
4.2.6	Componentes Relevantes:	42
4.3	EXECUÇÃO DOS TESTES E COLETA DE DADOS	43
4.4	DISCUSSÃO DOS RESULTADOS	43
4.4.1	Análise e discussão dos resultados	44
4.4.2	Análise dos resultados	44
5	CONCLUSÃO	45
5.1	TRABALHOS FUTUROS	46
6	REFERÊNCIAS BIBLIOGRÁFICAS	48
7	APÊNDICES.....	51
7.1	APÊNDICE A: EXEMPLO DE SCRIPT DE BACKUP.....	51
7.2	APÊNDICE B: EXEMPLO DE SCRIPT DE LIMPEZA DE BACKUPS ANTIGOS	52
8	ANEXOS.....	53
8.1	ANEXO A: IMPLEMENTAÇÃO	53
8.2	ANEXO B: DIFICULDADES.....	55

LISTA DE FIGURAS

Figura 1 – Arquitetura Bacula.....	19
Figura 2 – Funcionamento de um SA.....	20
Figura 3 – Estrutura do SA BTRFS.	23
Figura 4 – Integração do Ambiente ao Armazenamento em Nuvem.....	27

LISTA DE SÍMBOLOS E ABREVIATURAS

ARPANET	–	Rede de Agências de Projetos de Pesquisa Avançada
Big Data	–	Algoritmo de aprendizado de máquina para a capacidade de coletar as informações dos consumidores e aplica-los na inteligência dos negócios.
Brain	–	Servidor (Cérebro autônomo)
CC	–	Cloud Computing (Computação em Nuvem)
CEO	–	Chief Executive Officer (Diretor executivo)
FEM	–	Fórum Econômico Mundial
FSF	–	Free Software Foundation (Fundação de Softwares Livres)
FTP	–	File Transfer Protocol (Protocolo de Transferência de Pastas)
GNU GPL	–	General Public License (Licença Pública Geral),
HD	–	Hard Disk (Disco Rígido)
IA/AI	–	Inteligência Artificial
MPEs	–	Micro e Pequenas Empresas
SA	–	Servidor de Arquivos
SCP	–	Secure Copy Protocol (Protocolo de Cópia de Segurança)
Serverless	–	Computação sem Servidor
SO	–	Sistema Operacional
SSH	–	Secure Socket Shell (Shell de Soquete Seguro)
TI	–	Tecnologia da Informação
TCC	–	Trabalho de Conclusão de Curso
UPS	–	Uninterruptible Power Supply (Fonte de Energia Ininterrupta)

1 INTRODUÇÃO

Quem nunca viu a famosa mensagem "Espaço Insuficiente"? A recomendação do Fórum Econômico Mundial – FEM, é que as empresas e usuários devam se conscientizar a respeito do que deve ser guardado e por quanto tempo. Técnicas de redução de dados estão sendo desenvolvidas para eliminar redundâncias, duplicatas e aplicações de compressão. É fundamental um planejamento para a guarda desse material, dados acessados com mais frequência terão prioridades e acesso a mídias mais velozes, enquanto dados secundários poderão ser guardados em mídias mais lentas.

Vive-se nos dias de hoje numa realidade aonde a aparência exposta é que o armazenamento é infinito, quando na verdade não é! Nem no computador, ou Smartphone, quem dirá no banco de dados das gigantes da tecnologia. Logo, logo, esse cenário deverá se esbarrar em seus limites, gerando assim uma redução no armazenamento em diversos serviços (E-mail; Redes Sociais; entre outros), mesmo com o advento de novas tecnologias.

No ambiente empresarial, a quantidade de dados a serem protegidos tem aumentado exponencialmente. Além disso, a diversidade de fontes e formatos dos dados torna o processo de backup um pouco mais complexo. A abordagem tradicional de backup, que envolve a cópia dos dados para mídias físicas, pode ser demorada, propensa a erros e apresentar limitações de capacidade de armazenamento. Além disso, a necessidade de armazenamento seguro e confiável dos dados em caso de desastres ou falhas de hardware é fundamental.

Segundo o site TI Inside, 51% dos brasileiros consumidores de serviços tecnológicos não tem esse hábito de realização de Backup. O processo de backup é uma prática comum para preservar informações importantes e recuperá-las em caso de falhas, perdas ou desastres. No entanto, a eficiência e a confiabilidade dos sistemas de backup são fatores críticos a serem considerados. Os desafios envolvem o gerenciamento adequado dos recursos de armazenamento, a minimização do tempo de backup e recuperação, além da redução do impacto no desempenho do sistema em operação

A crescente dependência das organizações em relação aos dados digitais tem gerado uma demanda cada vez maior por soluções de backup eficientes e confiáveis. A perda de dados pode acarretar consequências graves, como prejuízos financeiros, danos

à reputação da empresa e interrupção de suas operações. Esse avanço tecnológico tem gerado um enorme aumento na geração e no armazenamento de dados em ambientes corporativos e pessoais.

Com o aumento exponencial significativo das informações, tornou-se essencial garantir a segurança e a integridade desses dados por meio de estratégias eficientes de backup. Nesse contexto, a implementação de um servidor de backup inteligente com capacidade de armazenamento local e na nuvem utilizando Bacula e BTRFS no Linux surge como uma inovação promissora.

Utilizando o Sistema Operacional (SO), Linux Ubuntu, é possível facilitar a recuperação de dados em uma pesquisa futura. Essa distro é simples de obter e é gratuita, o que a torna uma boa opção para tal implementação.

A solução utilizando o Bacula foi decidida devido ela ser uma ferramenta de backup de código aberto com interface simples de usar, permitindo gestão de backups, restaurações e verificação de dados através de rede ou localmente. Sua estrutura cliente-servidor permite que um backup centralizado em uma máquina e controle remoto de várias máquinas clientes. Ele é modular, independente e é uma General Public License – GNU (Licença Pública Geral), GNU GPL ou simplesmente GPL, que é a designação da licença para software livre idealizada por Richard Stallman em 1989, no âmbito do projeto GNU da Free Software Foundation (FSF), com suporte na comunidade e personalização de backups. Suporta a maioria dos dispositivos gestores de mercado, incluindo mídias ópticas, e possui versão corporativa com recursos exclusivos. Por sua vez, pode ser acessado através de um navegador web, permitindo controle remoto de qualquer computador na rede.

A importância deste trabalho reside na necessidade de proteger os dados dos mais diversos usuários, que muitas vezes são vitais para a continuidade dos projetos e negócios. A solução proposta pode ajudar a garantir a proteção e disponibilidade dos dados em caso de perda ou desastre. Ao ler esse documento iremos nos aprofundar em cada tópico citado de forma mais aplicada, visando sempre relatar e mostrar como foi feito.

1.1 OBJETIVOS GERAIS

O objetivo geral deste trabalho é desenvolver e implementar um servidor (Brain – Cérebro Autônomo) de backup inteligente utilizando o Bacula com o arquivo Linux BTRFS, dando assim a possibilidade de salvar os dados tanto localmente quanto na nuvem. Dessa forma, mantendo todo e quaisquer dados dos usuários de uma empresa a salvo, evitando de possíveis imprevistos, ações indesejáveis de terceiros, ou até mesmo de ataques terroristas de Crackers – Hackers com índole duvidosa.

1.2 OBJETIVOS ESPECÍFICOS

A ideia do projeto consiste na implementação de um servidor de backup automático com capacidade de armazenamento local e na nuvem, utilizando Bacula e BTRFS no Linux, onde esse servidor terá a missão de realizar o processo de backup dos arquivos de trabalho (planilhas, textos, imagens, banco de dados etc.) de todos os usuários contidos em uma rede de computadores, de forma eficaz e eficiente utilizando computação aplicada voltada na área Segurança de Dados.

Os arquivos salvos estarão em um diretório compartilhado no computador do usuário, ou então na nuvem adquirida. Com o servidor desenvolvido e implementado, a segurança dos dados administrativos da empresa irá aumentar significativamente. Para atingir esse objetivo, os seguintes objetivos específicos são propostos:

- + Investigar os fundamentos teóricos de backup, Bacula, BTRFS e armazenamento em nuvem, compreendendo seus conceitos, recursos e funcionamento.
- + Definir os requisitos necessários para a implementação de um servidor de backup inteligente que combine Bacula e BTRFS, garantindo a proteção abrangente dos dados.
- + Configurar um ambiente Linux com Bacula e o BTRFS, definindo as políticas e configurações adequadas para o correto funcionamento do servidor de backup.
- + Integrar o servidor de backup com serviços de armazenamento em nuvem, possibilitando a cópia dos dados para um ambiente externo e seguro.
- + Implementar um estudo de caso, simulando um cenário de teste e avaliando a eficiência e eficácia do servidor de backup inteligente proposto.

1.3 JUSTIFICATIVA

A escolha desse tema para o Trabalho de Conclusão de Curso (TCC), se dá pela relevância e atualidade da área de backup de dados. A crescente necessidade de proteger e recuperar informações valiosas, impulsiona a busca por soluções inovadoras e eficientes. A combinação do Bacula, reconhecido por sua robustez e flexibilidade, com o BTRFS, que oferece recursos avançados de snapshots e cópia de arquivos, apresenta uma alternativa criativa para o desenvolvimento de um servidor de backup inteligente.

A implementação de um servidor de backup automático apresenta-se como uma solução vantajosa diante dos desafios enfrentados pelas organizações em relação à proteção dos dados. Ao unir as vantagens de cada artifício é possível obter um resultado abrangente e eficaz para o restabelecimento dos dados com a capacidade de salvá-los na nuvem proporcionando maior segurança e disponibilidade, permitindo que as informações sejam recuperadas mesmo em casos de desastres físicos, invasões ou falhas no ambiente local.

A realização desta pesquisa permitirá a compreensão mais aprofundada das tecnologias envolvidas, bem como a análise dos resultados obtidos, auxiliando na identificação de benefícios, limitações e possíveis aprimoramentos da abordagem proposta. Além disso, contribuirá para a compreensão da implementação de um servidor de backup inteligente e fornecerá insights sobre a eficácia dessa abordagem em relação à proteção dos dados. A partir dos resultados obtidos, as organizações poderão tomar decisões embasadas na escolha e implementação de soluções de backup mais eficientes e confiáveis, aumentando a segurança de suas informações críticas.

2 REVISÃO DE LITERATURA

2.1 CONCEITO DE BACKUP E RECUPERAÇÃO DE DADOS

O backup é um processo que consiste em copiar os dados de um sistema para um local seguro, para que eles possam ser recuperados em caso de perda ou desastre. O backup e a recuperação de dados são práticas fundamentais para a proteção de dados de empresas, garantindo uma continuação dos negócios em caso de perda de dados. O processo de backup pode ser realizado de formas distintas contendo suas próprias vantagens e desvantagens.

"O processo de backup pode ser realizado de diferentes formas, como backup completo, incremental, diferencial e espelhamento, cada um com suas vantagens e desvantagens" (Santos & Martins, 2019, p. 5).

Enquanto a recuperação de dados, como o próprio nome diz, é o processo de restaurar os dados a partir do backup após uma falha no sistema. Ambos os processos são vitais para garantir a disponibilidade e integridade dos dados em empresas de qualquer porte. Uma recuperação de dados, assim como o processo de backup, também pode ser realizada de diferentes maneiras.

"A recuperação de dados pode ser realizada por diferentes maneiras, como restauração completa, restauração de arquivos individuais e recuperação de desastres" (Carissimi, Fracalossi & Segala, 2015, p. 66).

Logo, entende-se que o backup é um processo importante para garantir a disponibilidade e integridade dos dados de uma empresa. Segundo Dhillon e Backhouse (2001), "o backup é uma das atividades mais importantes em qualquer organização, pois os dados são um ativo crítico e valioso". A recuperação de dados, por sua vez, é o processo de restaurar os dados a partir do backup após uma falha no sistema. Groth e Willett (2010) destacam que "a recuperação de dados é uma tarefa crítica para as organizações, e o tempo necessário para a recuperação pode ter um impacto significativo nas operações do negócio".

Com o fim de garantir a proteção e disponibilidade das informações, o backup de dados se torna fundamental. Diversos sistemas de backup estão disponíveis, cada um com suas características e funcionalidades. Nesse contexto, a implementação de um servidor de backup inteligente se torna crucial para atender às necessidades de diferentes ambientes, como MPEs. Um servidor de backup inteligente deve ser capaz de

gerenciar e proteger os dados de forma eficiente, combinando recursos avançados de armazenamento e recuperação.

Segundo Preston et al. (2019, p. 15), "o backup é a cópia dos dados originais para um local secundário", enquanto Dubey e Mishra (2018, p. 45) destacam que "empresas sem uma estratégia adequada de backup enfrentam riscos significativos".

2.2 BACULA: ARQUITETURA E FUNCIONALIDADES

O Bacula é um software de backup de código aberto, que oferece recursos de backup e recuperação de dados avançados, além de permitir a integração com diferentes tipos de dispositivos de armazenamento. Composto pelos componentes Diretor, Console, Armazenamento e Clientes, o Bacula permite a coordenação das operações de backup e recuperação de dados em uma rede de computadores heterogênea. Ele é uma ferramenta de backup altamente escalável e personalizada, que oferece recursos de backup inteligente e recuperação de desastres. Também é amplamente utilizado em empresas de portagens diferentes, desde pequenas até grandes corporações, devido à sua flexibilidade e ao seu custo. Sendo assim, pode-se dizer que além de econômico, ele pode suportar um aumento de carga, adequado para diferentes portes de empresa.

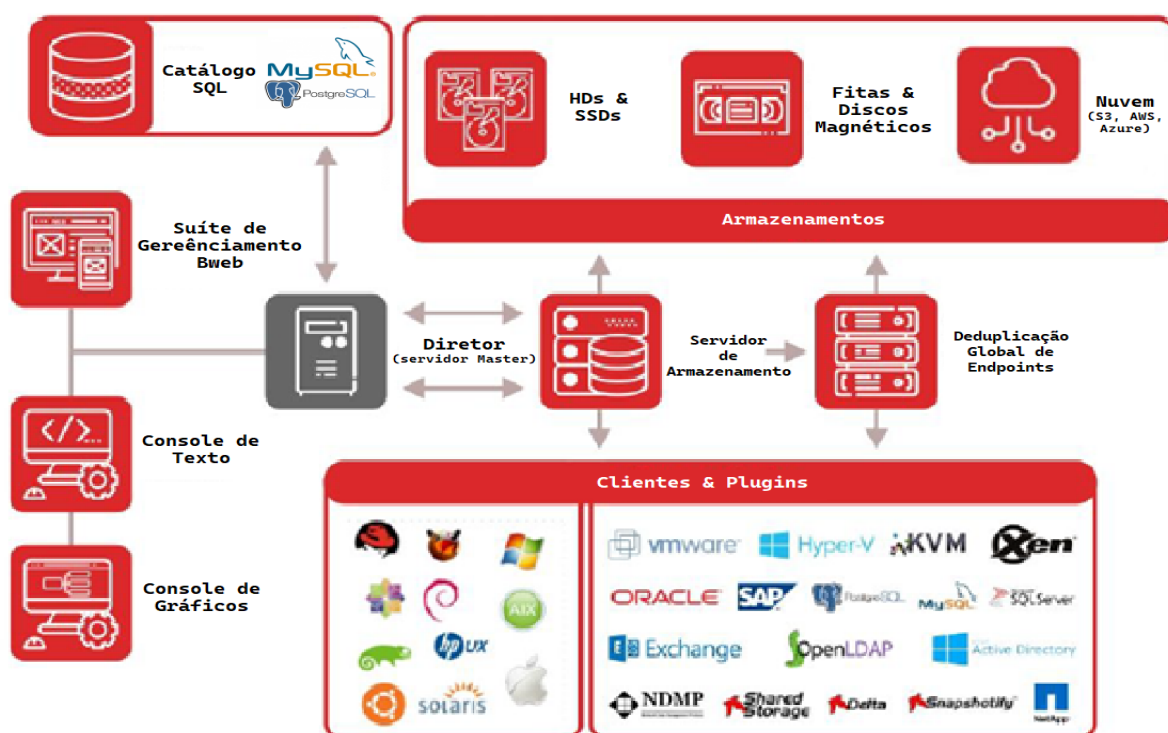
Carissimi et al. (2015, p. 6) destacam que "o Bacula é uma solução de backup de código aberto, que permite a integração com diferentes tipos de dispositivos de armazenamento, tornando-se uma opção econômica e escalável para empresas de diferentes portes". Santos e Martins (2019, p. 8) também enfatizam que "o Bacula é uma solução de backup e recuperação de dados adequada para pequenas e médias empresas".

Além disso, ele oferece funcionalidades como agendamento flexível, compressão de dados, criptografia e verificação de integridade. A configuração adequada do Bacula é essencial para garantir um servidor de backup inteligente eficiente. De acordo com a Bacula Systems (2021, p. 10) e Kern et al. (2020, p. 2123), "o Bacula é um sistema de backup de código aberto que oferece uma arquitetura flexível e recursos avançados".

O Bacula possui uma série de funcionalidades que o tornam adequado para a implementação de um servidor de backup inteligente. Ele suporta agendamento flexível, permitindo a definição de políticas de backup de acordo com as necessidades

específicas. Além disso, ele oferece recursos avançados, como compressão de dados para otimizar o uso do armazenamento, criptografia para garantir a segurança dos dados em trânsito e suporte para múltiplas tecnologias de armazenamento, incluindo discos, fitas e armazenamento em nuvem (Bacula Systems, 2021, p. 14; Kern et al., 2020, p. 2123). Na **Figura 1**, encontra-se a arquitetura do software, exemplificando seu funcionamento com as tecnologias citadas.

Figura 1 – Arquitetura Bacula.



Fonte: Website Target.

2.3 SISTEMAS DE ARQUIVO LINUX

Um Sistema de Arquivos (SA) é uma estrutura utilizada pelos sistemas operacionais para organizar, armazenar e recuperar dados em dispositivos de armazenamento, como discos rígidos e unidades de estado sólido. Ele fornece um meio de acesso aos arquivos, permitindo que os usuários e aplicativos localizem, leiam, escrevam e excluam dados armazenados.

De acordo com Silberschatz et al. (2013, p. 588), "um sistema de arquivos é uma coleção de estruturas de dados que administra como os arquivos estão armazenados e organizados em um dispositivo de armazenamento".

Os SA desempenham um papel importante na organização e armazenamento dos dados em um dispositivo. Eles utilizam estruturas de metadados para rastrear informações sobre os arquivos, como seus nomes, tamanhos, datas de criação e permissões de acesso. Como explicado por Tanenbaum e Woodhull (2014, p. 89), "o sistema de arquivos mantém informações sobre cada arquivo, incluindo seu nome, localização, tamanho, tipo de arquivo, data de criação, data de modificação e permissões de acesso".

Além disso, os SA definem a estrutura física dos dados no dispositivo de armazenamento. Eles dividem o espaço em blocos ou setores, onde os arquivos são armazenados. Isso permite a alocação eficiente do espaço disponível, evitando fragmentação e garantindo o acesso rápido aos dados.

Conforme destacado por McKusick et al. (2015, p. 26), "o sistema de arquivos alocam espaço em unidades lógicas de armazenamento chamadas blocos ou setores, e depois aloca esses blocos para os arquivos".

Na imagem retratada na **Figura 2**, podemos observar um resumo de como funciona um SA. Ela é uma camada de software que em seu funcionamento normal gerencia a organização, armazenamento e recuperação dos dados em um dispositivo de armazenamento. Ele fornece uma interface para o acesso aos arquivos, mantém informações sobre os metadados e controla a alocação eficiente do espaço de armazenamento. Silberschatz et al. (2013, p. 588), menciona que o sistema de arquivos é uma parte crucial no SO que permite aos usuários armazenar e acessar informações de forma organizada e eficiente.

Figura 2 – Funcionamento de um SA.



Fonte: Página web CódigoFonte².

Elas desempenham um papel essencial no gerenciamento e armazenamento de dados em sistemas operacionais Linux. Existem diferentes SA que estão disponíveis, cada um com suas características e vantagens. Isso significa que os administradores de sistemas têm a liberdade de escolher o sistema de arquivos mais adequado às necessidades específicas de um ambiente. "diferentes sistemas de arquivos estão disponíveis no Linux, cada um com suas características e vantagens" (Silva, 2021, p. 72).

Como dito, existem diferentes tipos de SA, cada um com suas próprias características, funcionalidades e otimizações. Alguns exemplos populares incluem o sistema de arquivos FAT32, NTFS (utilizado no Windows), ext4 (comum em sistemas Linux), APFS (utilizado pela Apple) e o sistema de arquivos HFS+ (utilizado em versões mais antigas do macOS). No contexto do Linux, diferentes sistemas de arquivos estão disponíveis, cada um com suas características e vantagens. O BTRFS é um exemplo de sistema de arquivos avançado e robusto que pode ser utilizado para implementar um servidor de backup inteligente no Linux.

A compreensão dos sistemas de arquivo no Linux é importante para escolher a melhor solução para as necessidades de backup e recuperação dos dados. Kaur e Gupta (2017, p. 112) e Kern et al., 2020, p. 2125 afirmam que "o BTRFS é um sistema de arquivos nativo do Linux com recursos avançados". Ele utiliza uma estrutura de árvore balanceada (B-tree) para organizar e acessar os dados de maneira eficiente. Além disso, ele oferece recursos como snapshots (instantâneos), compressão, criptografia e verificação de integridade. A escolha do BTRFS como sistema de arquivos pode trazer benefícios adicionais para a implementação de um servidor de backup inteligente. Ao combiná-lo com soluções de backup como o Bacula, é possível obter um servidor de backup inteligente e eficiente.

Outro sistema de arquivos amplamente utilizado é o EXT4, que é uma evolução do sistema de arquivos EXT3. Segundo Silva (2021, p. 78), o EXT4 é considerado o sistema de arquivos padrão para a maioria das distribuições Linux atualmente. Ele apresenta melhorias em relação ao EXT3, como suporte a tamanhos de arquivos maiores, melhor desempenho e confiabilidade. O EXT4 é uma opção confiável e amplamente adotada para o armazenamento de dados em sistemas Linux.

Contudo, pessoalmente, o BTRFS está em um nível diferente se comparado ao EXT4. O EXT4 é um sistema de arquivos “puro”, enquanto o BTRFS possui gerenciamento de disco e volume incorporado. O intuito não é ter um sistema de arquivos EXT4 que se estende por vários discos sem alguns truques “sujos” (que ainda não realizam o que se deseja). Por outro lado, os BTRFS podem fazê-lo sem problemas, porque foi projetado para fazer isso.

2.4 BTRFS: CARACTERÍSTICAS E VANTAGENS

Uma das principais características do BTRFS é a capacidade de criar snapshots (instantâneos) dos dados. Isso permite a criação de cópias pontuais dos sistemas de arquivos, facilitando a recuperação de versões anteriores dos dados em caso de falhas ou exclusões acidentais. Além disso, ele oferece mecanismos de verificação de integridade dos dados, permitindo detectar e corrigir erros de forma automática (Kaur & Gupta, 2017, p. 643; Souza et al., 2019, p. 87).

Muitos outros recursos, como somas de verificação, snapshots do sistema, recursos de invasão, recursos interessantes de reversão, entre diversos outros, também são recursos que diferenciam o BTRFS de outros sistemas de arquivos, e esse é o ponto que eu quero apresentar com firmeza.

Vale citar que o BTRFS é um sistema de arquivos nativo do Linux que oferece recursos avançados de gerenciamento e proteção de dados. Utilizando uma estrutura de árvore balanceada (B-tree), como mostrado na **Figura 3**, o BTRFS organiza e acessa os dados de forma eficiente. Ele suporta recursos como snapshots (instantâneos), compressão, criptografia e verificação de integridade. A utilização do BTRFS como parte da implementação do servidor de backup inteligente proporciona benefícios adicionais, como a capacidade de restaurar versões anteriores dos arquivos e verificar a integridade dos dados. "O BTRFS é um sistema de arquivos nativo do Linux com recursos avançados" (Kaur e Gupta, 2017, p. 112; Souza et al., 2019, p. 87).

Figura 3 – Estrutura do SA BTRFS.

Fonte: Elaboração Própria

2.5 AMBIENTES DE MICRO E PEQUENAS EMPRESAS

As micro e pequenas empresas (MPEs) são um segmento importante da economia, representando uma parcela significativa do total de empresas em muitos países. As empresas de pequeno porte enfrentam desafios específicos em relação à proteção de dados, a falta de recursos financeiros e de pessoal especializado em TI. Isso não significa que seja o fim, elas geralmente têm recursos limitados para investir em tecnologia e infraestrutura, o que pode tornar a proteção de dados um desafio. No entanto, existem soluções de backup e recuperação de dados de acessos e efeitos disponíveis no mercado, que podem ser atendidos às necessidades específicas das empresas. Sendo assim, é possível afirmar que as MPEs enfrentam desafios específicos na proteção de dados, devido a recursos limitados de investimento em tecnologia e infraestrutura.

"as MPEs geralmente têm baixos níveis de investimento em tecnologia da informação, o que pode dificultar a proteção de dados e a realização de backups eficientes" afirmam Santos e Martins (2019). Dhillon e Backhouse (2001) também apontam que "as MPEs têm menos recursos para investir em tecnologia, o que pode levar a problemas com a proteção de dados".

2.6 BACKUP INTELIGENTE

O backup inteligente é uma técnica de backup que utiliza uma abordagem de algoritmos com inteligência artificial e aprendizagem de máquina para analisar, identificar e priorizar os dados críticos para decidir quais arquivos devem ser copiados

em um determinado momento para backup. Isso ajuda a economizar tempo e espaço necessários para realiza-lo, além de permitir uma recuperação mais rápida e eficiente do armazenamento, garantindo uma proteção adequada dos dados vitais da empresa.

Junior e Cunha (2018) afirmam que "o backup inteligente é uma abordagem que utiliza técnicas de análise de dados para decidir quais arquivos devem ser copiados, permitindo reduzir o tempo e o espaço necessários para o backup". Groth e Willett (2010) também destacam que "o backup inteligente é uma técnica que pode melhorar significativamente a eficiência do backup, permitindo a realização de backups mais frequentes e reduzindo o tempo necessário para a recuperação de dados".

A implementação de um servidor de backup inteligente, combinando tecnologias como Bacula, BTRFS e armazenamento em nuvem, proporciona uma solução abrangente para a proteção de dados. O Bacula oferece recursos avançados de backup e recuperação, o BTRFS fornece um sistema de arquivos robusto e flexível, e o armazenamento em nuvem adiciona redundância e segurança aos dados. Essa combinação permite uma estratégia de backup inteligente, garantindo a disponibilidade e a integridade dos dados (Preston et al., 2019, p. 32; Dubey & Mishra, 2018, p. 120).

Essa combinação de tecnologias oferece uma solução abrangente para a proteção de dados, permitindo um servidor de backup inteligente que atenda às necessidades de diferentes ambientes e requisitos de segurança. Através dessa implementação, é possível garantir a disponibilidade, integridade e segurança dos dados, tanto localmente quanto na nuvem (Preston et al., 2019, p. 32; Dubey & Mishra, 2018, p. 120).

2.7 ARMAZENAMENTO EM NUVEM : BENEFÍCIOS E DESAFIOS

De modo mais grosso, o armazenamento através da Cloud Computing (CC) – Computação em Nuvem, é quase como se fosse um Hard Disk Drive (HDD) – Drive de Disco Rígido , ou um Solid State Drive (SSD) – Disco de Estado Sólido, porém longe de sua casa. O que a diferencia é que a CC é mais que uma fila de HDs em uma sala quietinha com ar-condicionado no super gelado. Tem processadores trabalhando 24 horas por 7 dias por semana, ou seja, o trabalho é “Nonstop” (do inglês, sem parar) e não há registros indicando que isso irá mudar, pelo menos não antes de 2030.

"A diferença fundamental entre os HDs tradicionais e a Cloud Computing está na forma como os dados são armazenados e acessados. Enquanto os HDs são dispositivos físicos localizados no local do usuário ou em servidores dedicados, a Cloud Computing utiliza recursos de armazenamento virtualizados e distribuídos em data centers remotos, acessíveis pela internet" (Silva, 2021, p. 15).

De modo intuitivo e bem analisado, entende-se que a Computação em Nuvem tem muito a ver com os serviços de Inteligência Artificial – IA/AI; Machine Learning; Big Data. E ela está mudando a natureza desses negócios devido ao poder de processamento disponível externamente. Sendo assim, desafios que seriam impossíveis estão sendo resolvidos hoje. Por isso a escolha de criar um servidor de Backup em Nuvem, pois não é de se admirar que diversos especialistas nessas áreas afirmam que aquilo que temos hoje está apenas no início, graças a capacidade de armazenamento e processamento.

"Não é de se admirar que diversos especialistas nas áreas de armazenamento em nuvem afirmem que aquilo que temos hoje está apenas no início, graças à capacidade de armazenamento e processamento disponibilizada pela Cloud Computing" (Martins, 2022, p. 78).

Primeiramente, o americano John McCarthy (1961) discutiu o uso compartilhado do computador, de forma simultânea, por dois ou mais usuários, no qual diz que "Compartilhar o computador entre muitos usuários é uma ideia tão óbvia que parece estranho ter sido necessário inventá-la.". O conceito foi chamado por ele de "Utility Computing". Anos mais tarde, McCarthy ficaria conhecido como o "pai da inteligência artificial" e inventor da programação Lisp.

Passados mais alguns anos, foi a vez de Joseph Carl Robnett Licklider estudar novas formas de usar o computador. Lick ajudou a desenvolver a Rede de Agências de Projetos de Pesquisa Avançada (ARPANET). Inclusive, a ARPANET foi a primeira rede que permitiu o compartilhamento de informações entre computadores que não estavam no mesmo local físico. O objetivo do pesquisador era que a conexão pudesse ser feita de qualquer lugar e que estivesse disponível a qualquer horário utilizando a Serverless (Computação sem Servidor), que é o termo para indicar a arquitetura de software em Nuvem que dispensa o gerenciamento de infraestrutura. Por meio dele, todos os recursos necessários para desenvolver e implementar um software são gerenciados pelo provedor da Nuvem. Em síntese, são estes os conceitos da computação em nuvem: disponibilidade e acessibilidade.

"A computação em nuvem se destaca pela sua capacidade de oferecer alta disponibilidade e acessibilidade aos usuários. Através de uma rede de servidores distribuídos em data centers geograficamente dispersos, os serviços em nuvem garantem a disponibilidade contínua dos recursos, minimizando o tempo de inatividade. Além disso, a acessibilidade é ampliada, permitindo o acesso aos dados e aplicativos hospedados na nuvem de qualquer lugar, a qualquer momento, por meio de dispositivos conectados à Internet" (Ferreira, 2022, p. 27).

Em 2015 Mark Hurd, Chief Executive Officer (CEO – Diretor Executivo) da renomada empresa Oracle, fez o seguinte comentário “Em 2025, 80% de todas as aplicações de ambiente corporativo estariam rodando na nuvem”. Ousado?! Com certeza não! Essa adoção em larga escala permite que empresas manipulem e analisem dados muito maiores do que seria possível em seus próprios data centers. O que estamos falando aqui é sobre o nosso futuro, ou melhor nosso presente, que popularmente conhecemos como Big Data – algoritmo de aprendizado de máquina para a capacidade de coletar as informações dos consumidores e aplicá-los na inteligência dos negócios.

O armazenamento em nuvem oferece uma abordagem alternativa e complementar para a proteção de dados. Com a capacidade de salvar e gerenciar os dados em servidores remotos acessíveis pela Internet, o armazenamento em nuvem proporciona benefícios como escalabilidade, disponibilidade e redundância. No entanto, também apresenta desafios em termos de segurança e privacidade dos dados, como a segurança no trânsito e armazenamento de dados, bem como questões de privacidade. Enquanto os HDs têm sido a opção tradicional para o armazenamento de dados, a abordagem alternativa do armazenamento em nuvem oferece uma solução inovadora. Através da utilização de recursos de armazenamento virtualizados em data centers remotos, a Cloud Computing proporciona vantagens como escalabilidade, flexibilidade e acesso conveniente aos dados, sem depender da presença física de discos rígidos locais (Rodrigues et al., 2022, p. 32).

A **Figura 4**, exemplifica de modo claro como funciona a integração do ambiente ao nosso redor com o armazenamento em nuvem como parte da implementação do servidor de backup inteligente, o que proporciona uma camada adicional de segurança e disponibilidade dos dados. Ao integrar o armazenamento em nuvem com soluções de backup como o Bacula e o BTRFS, é possível obter uma estratégia de backup mais

abrangente, combinando a proteção local com a redundância e a disponibilidade dos dados na nuvem (Mell e Grance, 2011, p. 5; Ristenpart et al., 2016, p. 102).

Figura 4 – Integração do Ambiente ao Armazenamento em Nuvem.



Fonte: Google Images.

No entanto, o armazenamento em nuvem também apresenta desafios, especialmente em termos de segurança e privacidade dos dados. É essencial garantir que os provedores de serviços em nuvem adotem medidas adequadas de segurança, como criptografia dos dados em repouso e em trânsito, para proteger as informações armazenadas. Além disso, é importante considerar questões de conformidade regulatória e escolher provedores confiáveis e respeitáveis (Mell & Grance, 2011, p. 5; Ristenpart et al., 2016, p. 102).

Esses tópicos são Insights que só são possíveis obter quando se faz algo com esse intuito, levando o poder de processamento para outro patamar. Os resultados se tornam processos mais dinâmicos dentro das empresas, menos achismos, maior redução de custos, mais clientes satisfeitos, menos tarefas repetitivas o qual é o intuito. Novamente citando o Mark Hurd, ele diz que a chave da CC é a Automação, com máquinas cada vez mais poderosas e cada vez mais inteligentes as tarefas repetitivas estão com os dias contados. Segundo o CEO 60% das vagas de Tecnologia da Informação – TI, ainda nem existem. Um novo cenário assim, gera a necessidade de um novo profissional.

3 METODOLOGIA

Para este trabalho, foi preciso criar uma estrutura de dados a ser relacionada com o Backup, pois ela servirá como banco de dados a serem extraídos durante o processo de execução do Brain; e desenvolver a implementação dele de modo com que o servidor inteligente venha a ter capacidade de armazenamento local e na nuvem através do Bacula e do BTRFS no Linux. Esse projeto foi desenvolvido com o fim de manter e recuperar todos os dados dos funcionários e entidades, deixando-os a salvo, evitando possíveis imprevistos, ações indesejáveis de terceiros, ou até mesmo de ataques terroristas de Crackers.

Primeiramente, é necessário identificar a localização da pasta alvo, nesse caso foi escolhida a pasta 'Documentos' no sistema de arquivos do computador do usuário. Isso pode ser feito por meio de pesquisa da estrutura de diretórios padrão do sistema operacional ou por configuração personalizada definida pelo usuário. Uma vez identificada a pasta, é realizado um processo de listagem de todos os arquivos contidos nela. Essa listagem pode ser obtida por meio de chamadas de sistema ou bibliotecas de manipulação de arquivos, dependendo da linguagem de programação utilizada no desenvolvimento do servidor de backup.

Após a listagem dos arquivos, são aplicados filtros para selecionar apenas os arquivos relevantes a serem incluídos no backup. Por exemplo, podem ser aplicados filtros baseados em extensões de arquivo específicas, data de modificação ou tamanho do arquivo, de acordo com as preferências e requisitos do usuário. Com base nos arquivos selecionados, é criada uma estrutura de dados adequada para armazenar as informações necessárias para o backup. Isso pode envolver a criação de metadados, como nome do arquivo, localização, tamanho, data de modificação e outras informações relevantes.

Uma vez criada a estrutura de dados, é implementado o processo de backup, utilizando a solução Bacula. Esse processo envolve a transferência dos arquivos selecionados para o armazenamento de backup, seja local ou em nuvem, por meio de protocolos de comunicação adequados, como FTP – File Transfer Protocol (Protocolo de Transferência de Pastas), SSH – Secure Socket Shell (Shell de Soquete Seguro) ou outros protocolos suportados pelo Bacula.

Em suma, esse capítulo descreve os recursos e as abordagens utilizadas para realizar essas etapas a serem concluídas no projeto, contendo os seguintes subtítulos: Passo a passo para moldar o gerenciador de backup; Definição dos requisitos do servidor de backup; Configuração do ambiente Linux com Bacula e BTRFS; Integração com serviços de armazenamento em nuvem; Implementação do servidor de backup inteligente. Tais temas trarão clareza e objetividade para realizar o molde do Brain almejado, visto que o intuito geral é realizar proteção e segurança para os usuários de MPEs, com a possibilidade de manter e recuperar os dados tanto localmente quanto na nuvem.

Logo, a finalidade, então, é idealizar um gerenciador de backups utilizando o Bacula. Vale salientar que o Bacula é um software e pode levar tempo para configurar corretamente. É recomendável consultar a documentação do Bacula ou contratar um profissional experiente para ajudar na configuração e gerenciamento do backup. Ao seguir essa metodologia, é possível implementá-lo de modo com que ele seja eficiente, capaz de proteger os dados de forma confiável, utilizar recursos avançados de backup automático e integrar-se com serviços de armazenamento em nuvem para garantir redundância e disponibilidade dos dados.

3.1 DEFINIÇÃO DOS REQUISITOS DO SERVIDOR DE BACKUP

Para implementar um servidor de backup inteligente, é essencial identificar e definir os requisitos do sistema. Isso inclui:

3.1.1 Capacidade de Armazenamento

O primeiro passo é determinar a capacidade de armazenamento necessária para acomodar os backups dos dados. Isso envolve analisar o tamanho dos dados a serem protegidos, considerar a expansão futura e garantir que haja espaço suficiente para retenção adequada dos backups.

Pensando nisso, todos os dados criados terão de ser alocados de acordo com o tamanho de armazenamento (HD/SSD externo, USB, Nuvem, Disco Magnético) obtido. No caso desse trabalho, foi levado em conta apenas os documentos obtidos na pasta

Documentos do computador, a partir do momento em que são salvas nessa área de destino já estão sendo contabilizadas e entrarão nas políticas de backup e recuperação necessários.

3.1.2 Políticas de Backup e Recuperação

Após se ter uma ideia mínima de espaço que será utilizado para a realização do backup, é de suma importância definir as políticas de backup e recuperação. Isso inclui estabelecer a frequência dos backups, determinar os tipos de dados a serem protegidos (como arquivos, bancos de dados, configurações do sistema), definir a retenção dos backups (quanto tempo eles serão mantidos) e estabelecer as políticas de recuperação para diferentes cenários (recuperação de arquivos individuais, restauração do sistema completo); sendo que, cada um desses tópicos são fundamentais para tal ocasião, para então aplicar os recursos necessários para sua automação.

3.1.3 Recursos de Backup Inteligente

Em seguida, têm-se que identificar e aproveitar os recursos de backup inteligente disponíveis nas tecnologias utilizadas visto que o objetivo é automatizar todo o processo, o que pode incluir duplicação de dados para reduzir o espaço de armazenamento necessário, compressão dos backups para otimizar a eficiência, criptografia para garantir a segurança durante transferência e alocação dos arquivos, além de recursos avançados de recuperação, como snapshots e pontos de verificação.

3.1.4 Requisitos de Hardware e Software

Para que tudo ocorra de acordo com o planejado, a análise dos requisitos de hardware e software, por certo, é essencial para garantir um desempenho adequado do servidor de backup. No qual inclui considerar a capacidade de processamento, quantidade de memória RAM, espaço em disco, interfaces de rede e compatibilidade com os sistemas operacionais, como Linux. Também é válido verificar os requisitos específicos das tecnologias utilizadas, como Bacula e BTRFS, e ter convicção de que o

hardware e o software irão atendê-los e, a partir de então fazer as configurações necessárias do ambiente de produção.

3.2 CONFIGURAÇÃO DO AMBIENTE LINUX COM BACULA E BTRFS

Para a etapa de configuração do ambiente Linux com Bacula e BTRFS, foi feito o seguinte:

3.2.1 Instalação do Bacula

Baixar os pacotes necessários, configurar as dependências e seguir as instruções específicas para o sistema operacional em uso, são os princípios de instalação do Bacula. Observa-se que ele deve ser instalado no servidor central e nos clientes Bacula, ou seja, nos dispositivos que contêm os dados a serem protegidos, fazendo com que a configuração planejada funcione em conjunto.

3.2.2 Configuração do Bacula

Ao obtê-lo, é necessário configurar de acordo com a necessidade. Isso envolve a definição do servidor Bacula, configuração dos clientes Bacula, estabelecimento de conexões entre eles, definição de armazenamento e recursos de backup, configuração de políticas de retenção e agendamento de backups. Depois disso feito, têm-se que definir suas regras de backup e recuperação para o funcionamento.

3.2.3 Configuração de Políticas de Backup e Recuperação

Tal como no item 3.1.2, definir políticas de backup e recuperação adequadas às necessidades do ambiente, também é crucial. E ele é aplicado a partir da definição dos horários de execução dos backups, tipos de backup a serem realizados (incremental, diferencial, completo), regras de retenção dos backups e configuração de estratégias de

recuperação. Ao concluir cada requisito anterior, é feito um teste que serve para verificar o desempenho do projeto.

3.2.4 Testes da Solução

Após a configuração, é fundamental realizar testes para verificar se a solução está funcionando corretamente. Isso envolve executar backups de teste, realizar restaurações de dados para validar a integridade dos backups e garantir que todas as políticas de backup e recuperação estejam operando conforme o esperado.

3.3 INTEGRAÇÃO COM SERVIÇOS DE ARMAZENAMENTO EM NUVEM

Para finalizar, será tratado de um assunto um pouco mais complexo, que é integração do servidor de backup com serviços de armazenamento em nuvem, pois é ele irá proporcionar uma camada adicional de proteção e redundância para os dados. Os passos a serem seguidos para tal feito, incluem:

3.3.1 Seleção do Serviço de Armazenamento em Nuvem

Identificar um serviço de armazenamento em nuvem confiável e adequado às necessidades desse projeto é o que tornará tal realização possível. Existem alguns provedores populares, como Amazon S3, Google Cloud Storage, Microsoft Azure entre outros serviços compatíveis com o Bacula. Sendo assim, a escolha fica sendo pessoal, obtendo o livre arbítrio para decidir qual agrada, ou então, qual possui mais compatibilidade e conhecimento, e então, configurá-lo para tal.

3.3.2 Configuração da Integração

Assim como comentado em algum dos itens anteriores, configurar as credenciais, é a base, pois é por meio disso que iremos obter acesso ao serviço de armazenamento em nuvem no Bacula. Ela envolve a definição de políticas de envio de backups para a nuvem, como frequência, tipos de dados e regras de retenção específicas.

3.4 IMPLEMENTAÇÃO DO SERVIDOR DE BACKUP INTELIGENTE

Iniciando a partir deste ponto o processo de automatização e implementação do servidor de backup inteligente, detalhes específicos do sistema requerem um pouco de atenção, como:

3.4.1 Arquitetura do Sistema

Descrever a arquitetura do servidor, destacando o Brain central Bacula; os clientes Bacula e o armazenamento em nuvem; e, explicar como esses componentes se interconectam e trabalham em conjunto para garantir a proteção dos dados; exigem um pouco de critério, pois é mediante a arquitetura que o exercício planejado criará forma. Nesse sentido, a definição dos componentes básicos de Hardware fica com um planejamento mais simples de ser definido graças a arquitetura planejada.

3.4.2 Componentes de Hardware

Especificar os componentes de hardware necessários, como servidores, dispositivos de armazenamento em nuvem, switches de rede e outros equipamentos que compõem o ambiente do servidor de backup inteligente; fazem parte de toda metodologia a ser aplicada, com o fim de designar os softwares utilizados no processo.

3.4.3 Software Utilizado

Agora, basta enumerar os softwares utilizados, incluindo o sistema operacional Linux, as versões específicas do Bacula e do BTRFS, bem como outros softwares auxiliares necessários para a configuração e operação do servidor de backup no ambiente.

3.4.4 Configuração do Ambiente

O intuito aqui é detalhar as configurações específicas realizadas no ambiente, incluindo as configurações de rede, as configurações do Bacula, as configurações do

BTRFS, as políticas de backup e recuperação, bem como quaisquer outros ajustes relevantes feitos para garantir a eficiência e a segurança do servidor de backup inteligente.

3.5 CRIAÇÃO DE UMA ESTRUTURA DADOS A SEREM RECOLHIDOS

A criação de dados de teste é fundamental para a implementação do backup, pois permite simular situações reais e verificar a eficácia do processo. Esses dados de teste podem ser criados utilizando diferentes abordagens, como a geração de arquivos fictícios com tamanhos variados e formatos diversificados, representando a diversidade de dados encontrados na pasta "Documentos" do usuário. Além disso, podem ser inseridos arquivos com estruturas de diretórios aninhados, nomes longos e caracteres especiais para testar a capacidade de recuperação e restauração do sistema. A criação desses dados de teste possibilita a validação do fluxo de backup, a verificação de erros ou falhas, e o ajuste de parâmetros e configurações necessários para garantir a integridade e a confiabilidade do processo de backup.

4 ESTUDO DE CASO

Nesta seção, está descrito o estudo de caso realizado para avaliar a solução proposta de implementação de um servidor de backup inteligente utilizando Shell Script para direcionar o Bacula, BTRFS e então, realizar a integração com serviços de armazenamento em nuvem. Para isso, foi feito algumas etapas importantes, para em seguida ser explicado o cenário de teste, a execução dos testes e coleta de dados, bem como a análise e discussão dos resultados obtidos.

4.1 PASSO A PASSO PARA MOLDAR O GERENCIADOR DE BACKUP

1º Conhecendo o Bacula:

- + O que é Bacula?
- + Principais recursos do Bacula.
- + Importância do backup para empresas e organizações.

2º Instalação do Bacula:

- + Verificação dos requisitos do sistema para instalação do Bacula.
- + Baixar e instalar o Bacula. No qual é feito através do site oficial do Bacula. Dependendo do SO, as etapas de instalação podem variar.
- + Configuração do ambiente de instalação.

Script de instalação do Bacula:

```
#!/bin/bash
```

```
# Comando utilizado para atualizar os pacotes do sistema
```

```
sudo apt update
```

```
# Comando utilizado para instalar os pacotes do Bacula
```

```
sudo apt install bacula-server bacula-director-pgsql bacula-common-pgsql bacula-console bacula-fd
```

3º Configuração do Bacula:

- + Após a instalação, configurar o Bacula para atender às necessidades é o processo que envolve definir os recursos de backup, tais como as máquinas a serem incluídas nos backups, os tipos de backup, os horários e a frequência dos backups.
- + Entender as opções de configuração do Bacula.
- + Configurar o horário e a frequência dos backups.

Figura 5 – Processo de Instalação e Configuração do Bacula.

```

root@ServidorBrain: /home/oem/Downloads

oem@ServidorBrain:~$ ls
Desktop Documentos Downloads Imagens Modelos Música Público snap Vídeos
oem@ServidorBrain:~$ sudo su
[sudo] senha para oem:
root@ServidorBrain:/home/oem# ls
Desktop Documentos Downloads Imagens Modelos Música Público snap Vídeos
root@ServidorBrain:/home/oem# cd Downloads/
root@ServidorBrain:/home/oem/Downloads# ls
bacula-13.0.3.tar.gz
root@ServidorBrain:/home/oem/Downloads# apt update
Atingido:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Obter:2 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Obter:3 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Obter:4 http://archive.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Baixados 337 kB em 2s (135 kB/s)
Lendo listas de pacotes... Pronto
Construindo árvore de dependências... Pronto
Lendo informação de estado... Pronto
Todos os pacotes estão atualizados
root@ServidorBrain:/home/oem/Downloads# tar -xzf bacula-13.0.3.tar.gz
bacula-13.0.3/
bacula-13.0.3/scripts/
bacula-13.0.3/scripts/mtx-changer.conf
bacula-13.0.3/scripts/bat.desktop.xsu.in
bacula-13.0.3/scripts/bacula-ctl-fd.in
bacula-13.0.3/scripts/tapealert
bacula-13.0.3/scripts/bconsole.in
bacula-13.0.3/scripts/breload.in
bacula-13.0.3/scripts/bacula-tray-monitor.desktop.in

bacula-13.0.3/COPYRIGHT
bacula-13.0.3/ChangeLog
bacula-13.0.3/LICENSE-FAQ
root@ServidorBrain:/home/oem/Downloads# mv bacula-13.0.3 /usr/src/
root@ServidorBrain:/home/oem/Downloads# apt-get install -y build-essential libread
line6-dev zlib1g-dev liblz2-dev mt-st mt-x postfix libacl1-dev libssl-dev libmysql
++-dev mysql-server

A processar 'triggers' para install-info (6.8-4build1) ...
root@ServidorBrain:/home/oem/Downloads# cd ..
root@ServidorBrain:/home/oem# cd ..
root@ServidorBrain:/home# ls
oem
root@ServidorBrain:/home# cd ..
root@ServidorBrain:/# ls
bin dev lib libx32 mnt root snap sys var
boot etc lib32 lost+found opt run srv tmp
cdrom home lib64 media proc sbin swapfile usr
root@ServidorBrain:/# cd usr/src/bacula-13.0.3/
root@ServidorBrain:/usr/src/bacula-13.0.3# chmod +x configure
root@ServidorBrain:/usr/src/bacula-13.0.3#

```

Fonte: Autoria Própria.

Script de configuração do Bacula:

```
#!/bin/bash
```

```
# Comando utilizado para configurar o Bacula de acordo com a necessidade  
sudo dpkg-reconfigure bacula-server
```

```
# Comando utilizado para configurar o arquivo de configuração do Bacula Director  
sudo nano /etc/bacula/bacula-dir.conf
```

```
# Comando utilizado para configurar o arquivo de configuração do Bacula Storage  
sudo nano /etc/bacula/bacula-sd.conf
```

```
# Comando utilizado para configurar o arquivo de configuração do Bacula File Daemon  
sudo nano /etc/bacula/bacula-fd.conf
```

```
# Comando utilizado para reiniciar os serviços do Bacula  
sudo service bacula-director restart  
sudo service bacula-sd restart  
sudo service bacula-fd restart
```

4º Criação dos arquivos de configuração:

- + Para gerenciar os backups com o Bacula, é necessário criar arquivos de configuração para os Jobs (trabalhos/etapas) de backup e para o pool de storage (espaços de armazenamento), que serve para criar um ou mais discos virtuais. Os arquivos de configuração especificam os dados que são salvos e como os dados estão sendo armazenados, no caso o BTRFS.
- + Entender como criar os arquivos de configuração para os jobs de backup e para o pool de storage.
- + Definir os dados que serão salvos e como os dados serão armazenados.

Script de configuração do BTRFS:

```
#!/bin/bash
```

```
# Comando utilizado para criar o sistema de arquivos BTRFS em um dispositivo
```

```
sudo mkfs.btrfs /dev/sdb
```

Comando utilizado para montar o dispositivo BTRFS

```
sudo mount /dev/sdb /mnt/btrfs
```

Comando utilizado para configurar o sistema de arquivos BTRFS para snapshots incrementais

```
sudo btrfs subvolume create /mnt/btrfs/snapshots
```

```
sudo btrfs subvolume create /mnt/btrfs/data
```

Comando utilizado para configurar o arquivo /etc/fstab para montar automaticamente o dispositivo BTRFS

```
echo "/dev/sdb /mnt/btrfs btrfs defaults 0 0" | sudo tee -a /etc/fstab
```

Reiniciar o sistema

```
sudo reboot
```

5º Criação e Execução dos scripts de backup:

- + Criar scripts de backup que o Bacula executará para criar backups. Esses scripts definem as etapas que o Bacula deve seguir para realizar backups e como os dados devem ser tratados durante o processo.
- + Entender como criar scripts de backup que o Bacula executará.
- + Definir as etapas que o Bacula deve seguir para realizar backups e como os dados devem ser tratados durante o processo.
- + Depois que tudo estiver configurado corretamente, inicia-se os backups, definindo como o Bacula deve executá-los; sendo manualmente ou se será para agendar backups automáticos.
- + Entender como executar backups manualmente e como agenda-los para ser realizados de forma automática.
- + Verificar o status dos backups e solucionar problemas.

Script de integração com armazenamento em nuvem:

```
#!/bin/bash
```

Comando utilizado para instalar o cliente de armazenamento em nuvem (por exemplo, Rclone)

```
sudo apt install rclone
```

ou

```
curl https://rclone.org/install.sh | sudo bash
```

Comando utilizado para configurar o cliente com o serviço de armazenamento em nuvem desejado

```
rclone config
```

Comando utilizado para criar um script para sincronizar os backups com o armazenamento em nuvem

```
echo "rclone sync /mnt/bacula/backups remote:/backup" | sudo tee  
/usr/local/bin/backup-sync.sh
```

```
sudo chmod +x /usr/local/bin/backup-sync.sh
```

Comando utilizado para agendar a sincronização dos backups para ser executada regularmente (por exemplo, diariamente)

```
sudo crontab -e
```

Adicione a seguinte linha ao arquivo crontab:

```
# 0 0 * * * /usr/local/bin/backup-sync.sh
```

6º Monitoramento do backup:

- + Para garantir que os backups estejam sendo executados corretamente, é necessário monitorá-los regularmente. Para isso, o Bacula permite a utilização de ferramentas para monitorar o status dos backups, verificando se ocorrem erros e fazer ajustes conforme o solicitado para o momento.
- + Entender como monitorar o status dos backups, verificar se há erros e fazer ajustes conforme necessário. Tal como a importância do monitoramento regular do backup.

7º Prática:

- + Criar dados a serem coletados.
- + Alocar os dados auxiliares que foram inventados na pasta “Documentos” do usuário.
- + Executar backups em uma máquina de teste.
- + Monitorar e solucionar problemas.

4.2 DESCRIÇÃO DO CENÁRIO TESTE

Para o estudo de caso, foi levado em conta um ambiente de uma empresa fictícia de médio porte, que possui uma infraestrutura de TI baseada em servidores Linux. O objetivo é proteger os dados críticos dessa empresa, o que garante sua disponibilidade, tal como a possibilidade de recuperação em caso de falhas. Incluindo a infraestrutura de hardware e software, as configurações dos servidores, dispositivos de armazenamento, rede e demais componentes relevantes, no qual depende das necessidades e do tamanho do ambiente em que o servidor de backup foi implementado. Também é mencionados os tipos e o volume de dados utilizados no cenário de teste.

4.2.1 Requisitos mínimos para o Servidor de backup:

- + **CPU:** Processador multicore para lidar com várias tarefas de backup simultâneas de forma eficiente.
- + **Memória RAM:** Ter uma quantidade adequada de RAM para permitir operações de backup rápidas e eficientes. Recomenda-se pelo menos 8 GB de RAM, mas isso pode variar dependendo do tamanho dos backups e da carga de trabalho.
- + **Armazenamento:** É necessário ter um armazenamento local para os backups. Pode ser um disco rígido interno ou um dispositivo de armazenamento externo conectado ao servidor.
- + **Sistema Operacional:** O uso de uma distribuição Linux, como Ubuntu Server, CentOS, ou Debian, com suporte ao Bacula e ao sistema de arquivos BTRFS.

- + **Software:** Bacula deve estar instalado e configurado corretamente no servidor de backup.

4.2.2 Requisitos mínimos para o Dispositivo de Armazenamento:

- + Para o armazenamento, pode ser usado ele externo, como discos rígidos USB, discos de rede (NAS) ou sistemas de armazenamento em anexo (SAN), para armazenar os backups. É válido certificar de que o dispositivo tenha capacidade de armazenamento suficiente para atender às necessidades de backup.

4.2.3 Requisitos mínimos para a Rede:

- + Ter uma rede confiável e de alta velocidade para garantir a transferência eficiente de dados durante o backup.
- + Para backups em nuvem, uma conexão de internet estável e de alta velocidade é necessária para transferir os dados para o provedor de armazenamento em nuvem.
- + Verificar as configurações de rede da máquina virtual para se obter a certeza de que o Brain está usando a opção de rede "Bridge" para que ela tenha um endereço IP na mesma rede local do computador principal com Windows.
- + Certificar de que a máquina virtual e o computador principal estejam conectados à mesma rede local ou ao mesmo roteador. Ou seja, fazer um teste a conectividade de rede. A partir da máquina virtual, recomenda-se fazer ping para o endereço IP do computador principal com Windows e verifique se há resposta. E, a partir do computador principal com Windows, refazer o ping para o endereço IP da máquina virtual e verifique se há resposta.

4.2.4 Verificar as configurações de firewall:

- + Certificar-se de que as portas necessárias para a comunicação entre a máquina virtual e o computador principal estejam abertas nos firewalls de ambos os sistemas. Por exemplo, as portas utilizadas pelo Bacula (como a porta 9101)

devem estar liberadas para permitir a comunicação entre o servidor Bacula na máquina virtual e o cliente Bacula no computador principal.

4.2.5 Teste a transferência de dados:

- + A partir do computador principal com Windows que é feita a conexão com a máquina virtual usando uma ferramenta de transferência de arquivos, como o Secure Copy Protocol (SCP) – Protocolo de Cópia de Segurança, para verificar se o acesso e a permissão para transferir arquivos entre os sistemas.

Por exemplo, o comando abaixo, se executado no Prompt de Comando do Windows ele irá servir para copiar um arquivo para a máquina virtual:

scp arquivo.txt user@<ip_da_maquina_virtual>:/caminho/destino/

Se todas essas etapas forem concluídas com sucesso, isso indica que a máquina virtual está configurada corretamente para se comunicar com o computador principal com Windows e permitir o acesso e a transferência dos backups conforme necessário. Como houve problemas de conectividade e transferência de dados durante o processo de implementação, algumas revisões das configurações de rede, firewall e permissões de acesso para garantir que tudo esteja configurado corretamente foram feitas.

O ambiente de teste consiste em um Brain central Bacula, que atua como o ponto central para o gerenciamento dos backups, e vários clientes Bacula distribuídos pela rede, representando os servidores e estações de trabalho a serem protegidos. Para isso foi configurada uma política de backup diário, realizando backups incrementais e backups completos semanais.

Além disso, é indispensável que a integração com um serviço de armazenamento em nuvem, onde uma cópia dos backups vai ser enviada periodicamente para garantir uma camada adicional de segurança e disponibilidade dos dados, não seja configurada.

4.2.6 Componentes Relevantes:

- + **Switches de rede:** Podem ser necessários switches adicionais para lidar com a carga de rede gerada pelos backups.

- + **Roteador:** Um roteador com recursos de firewall pode ser necessário para proteger a rede contra ameaças externas.
- + **UPS (Uninterruptible Power Supply):** É altamente recomendável ter uma UPS para garantir a disponibilidade contínua do servidor de backup em caso de queda de energia.

4.3 EXECUÇÃO DOS TESTES E COLETA DE DADOS

Os testes tem como meta ser executados ao longo de um período de um mês, durante o qual os backups são realizados de acordo com a política configurada. A ideia é que ao coletá-los, os dados automaticamente são relacionados tanto ao desempenho do servidor de backup, quanto ao tempo de execução dos backups, ao consumo de armazenamento e até mesmo com a eficácia da recuperação dos dados.

Para avaliar o desempenho do servidor de backup, as métricas monitoradas como taxa de transferência de dados, utilização de CPU e memória, e tempo de resposta do sistema durante a execução dos backups; são registradas regularmente para análise posterior. Os testes feitos incluem diferentes cenários, como backup de arquivos individuais, backup de bancos de dados, recuperação de dados, entre outros. Para cada teste, ficam registrados os tempos de execução, a taxa de transferência, o consumo de recursos (CPU, memória, armazenamento) e outros indicadores relevantes.

Além disso, testes de recuperação dos dados em diferentes cenários de falhas são simulados e realizados previamente. A recuperação será testada em nível de arquivo e diretório, bem como em nível de sistema completo, a fim de ter a certeza de uma eficácia e da velocidade do processo de restabelecimento. Durante a execução dos testes, as informações sobre o desempenho da solução, a eficácia das políticas de backup e recuperação, a integração com o armazenamento em nuvem e quaisquer problemas ou desafios encontrados durante o processo ficam registradas.

4.4 DISCUSSÃO DOS RESULTADOS

Os resultados analisados e discutidos para avaliar a eficácia da solução proposta, seguem os aspectos posteriores:

4.4.1 Análise e discussão dos resultados

- + **Desempenho do servidor de backup:** As métricas de desempenho coletadas durante a execução dos backups, incluem a taxa de transferência de dados, utilização de recursos e tempo de resposta do sistema. Os resultados foram comparados com os requisitos definidos anteriormente.
- + **Eficiência do armazenamento:** Avaliando o consumo de armazenamento dos backups utilizando o BTRFS e a integração com o serviço de armazenamento em nuvem. Fazendo em paralelo, a análise das taxas de compressão e de duplicação de dados, bem como a economia de espaço alcançada.
- + **Eficácia da recuperação:** Análise dos testes de recuperação para verificar a facilidade e a rapidez com que os dados puderam ser restaurados em caso de falhas. Considerando também, o tempo de recuperação e a integridade dos dados recuperados.

4.4.2 Análise dos resultados

Com base na análise dos resultados, torna-se possível avaliar o desempenho e a eficácia da solução de backup inteligente implementada, bem como identificar possíveis melhorias ou ajustes necessários no ambiente de backup. Sendo assim, os resultados comparados com os requisitos e metas estabelecidos na metodologia, se destacam pelos pontos fortes e limitações da solução proposta, bem como com as melhores práticas e padrões da indústria identificadas durante o estudo de caso. É importante ressaltar que o estudo de caso descrito acima é apenas um exemplo, e a execução dos testes pode variar de acordo com as características específicas do ambiente e dos requisitos do projeto.

5 CONCLUSÃO

Aqui, estão apresentadas as conclusões do estudo de caso, destacando a eficácia da solução proposta na implementação de um servidor de backup inteligente. Serão discutidas as contribuições do trabalho, as lições aprendidas e as recomendações para aprimorar ainda mais a solução.

Neste trabalho, foi proposta a implementação de um servidor de backup inteligente utilizando Bacula, BTRFS e armazenamento em nuvem, com a adição do grub-btrfs através da linguagem de programação Shell Script, para melhorar a recuperação do sistema. Ao longo do estudo, foram apresentados os conceitos fundamentais relacionados a backups, Bacula, BTRFS, sistemas de arquivo Linux e armazenamento em nuvem. A metodologia detalhada foi aplicada para configurar o ambiente Linux com Bacula e BTRFS, integrar serviços de armazenamento em nuvem e implementar o servidor de backup inteligente. Além disso, uma implementação funcional será conduzida para avaliar a solução proposta e os resultados obtidos vão ser analisados.

Este projeto contribuiu de forma significativa para o campo de backup e recuperação de dados, apresentando uma abordagem prática e eficiente para a implementação de um servidor de backup inteligente. As principais contribuições deste trabalho incluem:

- + **Princípios para a implementação do servidor de backup inteligente:** Foi demonstrado como inicializar o processo de implementação de um servidor de backup inteligente, utilizando tecnologias como Bacula, BTRFS e armazenamento em nuvem. Isso proporciona uma solução abrangente e escalável para empresas de micro e pequeno porte que desejam proteger seus dados de forma eficiente.
- + **Uso do grub-btrfs para recuperação do sistema:** A incorporação do grub-btrfs no servidor Bacula permite uma recuperação rápida e eficiente do sistema, por meio de snapshots incrementais do sistema de arquivos BTRFS. Essa funcionalidade acrescentou mais uma camada de segurança e flexibilidade ao processo de backup e recuperação.

5.1 TRABALHOS FUTUROS

- + **Escalabilidade em ambientes de grande porte:** O estudo de caso realizado abordou cenários de micro e pequenas empresas. No entanto, em ambientes de grande porte, pode ser necessário considerar questões adicionais, como o dimensionamento adequado dos recursos de hardware e a otimização de políticas de backup para lidar com grandes volumes de dados.
- + **Avaliação de outros sistemas de arquivos e tecnologias de armazenamento em nuvem:** Embora o BTRFS e o armazenamento em nuvem tenham sido abordados neste trabalho, existem outros sistemas de arquivos e serviços de armazenamento em nuvem disponíveis. Realizar uma comparação mais ampla e detalhada entre essas tecnologias pode fornecer insights adicionais sobre a melhor escolha para diferentes cenários.
- + **Implementação do Projeto:** Fazer uma amostragem prática e dinâmica de como todo esse projeto abordado funciona em um ambiente real.
- + **Realizar estudos de caso adicionais:** Fazê-lo em diferentes cenários e ambientes de trabalho, a fim de validar ainda mais a eficácia da solução proposta e identificar possíveis melhorias.
- + **Explorar técnicas avançadas de backup inteligente:** Investigar técnicas avançadas de backup inteligente, como a detecção automática de dados críticos e a otimização de algoritmos de duplicação e compressão, visando melhorar ainda mais a eficiência do sistema de backup.
- + **Aprofundar a segurança e criptografia:** Considerar a implementação de medidas adicionais de segurança, como criptografia de dados em repouso e em trânsito, garantindo uma proteção abrangente dos dados armazenados.
- + **Containers:** Os contêineres oferecem uma forma eficiente e isolada de empacotar e distribuir aplicativos e seus componentes, permitindo uma implantação consistente e escalável. Ao implementar um servidor de backup inteligente, a utilização de contêineres pode trazer benefícios

significativos (Portabilidade, Isolamento, Escalabilidade, Gerenciamento simplificado).

- + **Metodologias Ágeis:** As metodologias ágeis, como Scrum e Kanban, enfatizam a colaboração, a comunicação e a entrega incremental de funcionalidades. Ao aplicar essas metodologias no contexto da implementação do servidor de backup inteligente, podemos identificar alguns pontos de conexão (Entrega de valor incremental, Colaboração e comunicação, Flexibilidade e adaptação a mudanças, Melhoria contínua).

Em resumo, a implementação do servidor de backup inteligente utilizando Bacula, BTRFS e armazenamento em nuvem, juntamente com a integração do grub-btrfs, mostrou-se uma solução eficaz para a proteção e recuperação de dados em ambientes Linux. As vantagens proporcionadas, como a facilidade na gestão de backups, a rápida recuperação de sistemas e a maior disponibilidade dos dados, contribuem para aprimorar a segurança e a confiabilidade das operações de backup.

No entanto, é importante ressaltar que a adoção dessa solução deve levar em consideração as características específicas de cada ambiente e as necessidades individuais de cada organização. Além disso, é recomendado realizar testes e avaliações adicionais para verificar a escalabilidade, a segurança e o desempenho da solução em diferentes cenários.

Ao abordar essas “limitações” e realizar trabalhos futuros nessa linha, espera-se aprimorar e expandir ainda mais a solução proposta, contribuindo para a área de backup e recuperação de dados em ambientes Linux. Com base nos resultados obtidos e nas perspectivas futuras, a implementação de um servidor de backup inteligente utilizando Bacula, BTRFS e armazenamento em nuvem, com a integração do grub-btrfs, demonstra ser uma opção promissora para empresas que buscam aprimorar suas estratégias de proteção e recuperação de dados, garantindo a continuidade de suas operações e a preservação da integridade das informações.

6 REFERÊNCIAS BIBLIOGRÁFICAS

LIVROS E ARTIGOS:

JESUS, Guilherme Bindi Alencar e SCHIMIGUEL, Juliano(2018): “Implementação de Backup como processo de segurança da informação.”, Revista Atlante: Cuadernos de Educación y Desarrollo (fevereiro 2018). Disponível em: <http://www.eumed.net/2/rev/atlante/2018/02/backup-seguranca-informacao.html>

SCOPEL, Eduardo Longhi; FIORESE, Breno Carra; CERVINSKI, Natan Susin; CARLESSO, Guilherme Cavalheiro; XAVIER, Nicolás Lamb; TISOTT, Priscila Bresolin; ZANANDREA, Gabriela; DA SILVA, Eduardo Robini; CAMARGO, Maria Emilia (2018): “Importância da Segurança da Informação e Backup.” (novembro 2018)

INGIZZA, C. “Nuvem ou HD externo: qual o melhor tipo de backup?” Estadão.(julho 2017).Disponível em: <<https://link.estadao.com.br/noticias/geral,nuvem-ou-hd-externo-qual-omelhor-tipo-de-backup,70001895307>>.

FERREIRA, Bárbara Gonçalves e DA SILVA, Fábio Henrique Batista (dezembro 2019): "SEGURANÇA DA INFORMAÇÃO: backup em nuvem". Revista Agroveterinária, negócios e tecnologias, Coromandel, v. 4, n. 2, p. 25 - 41.

GONÇALVES, Marcelo de Oliveira: "Servidor de arquivos SAMBA: definição de um backup automático e controle de uso de disco". Niterói, 2018. 45 f. : il.

SOUZA, Frandecy R. N. de O.; FARIA, Neilson M.; DIAS, Taimara S.; VIANNA; Matheus B. M.: "Desenvolvimento de um Servidor de Backup Inteligente Utilizando a Linguagem Shell Script em Linux"

DE CARVALHO, Caio dos Santos Borsato; DOS SANTOS, Katia C. Lage; ALMEIDA, Fernanda Nascimento; ARBEX, Wagner: "Desenvolvimento e implementação de um sistema de backup para ambiente de computação científica com infraestrutura de baixo custo".

CARISSIMI, A. V., FRACALOSSO, A. e SEGALA, M. (2015). Bacula: um software livre para backup e recuperação de dados. Revista Brasileira de Computação Aplicada, 7 (1), 61-72.

DHILLON, G. S. e BACKHOUSE, J. (2001). Uma investigação dos fatores que afetam o processo de backup em pequenas e médias empresas. Journal of Information Technology, 16 (4), 209-218.

GROTH, K. e WILLETT, P. (2010). Backup inteligente: uma abordagem especializada do sistema para proteção de dados. Sistemas de especialistas com aplicações, 37 (7), 5272-5279.

JUNIOR, J. UMA. B. & CUNHA, M. C. R. (2018). Backup inteligente em ambientes de nuvem. Jornal Brasileiro de Desenvolvimento, 4 (6), 20592-20603.

- SANTOS, E. A. & MARTINS, P. S. (2019). Bacula: Uma solução de backup e recuperação de dados para pequenas e médias empresárias. Jornada Acadêmica da Universidade do Sul de Santa Catarina, 1 (1), 1-8.
- PATEL, N. Backup e recuperação: conceitos e estratégias de implementação. Boca Raton, FL: Publicações Auerbach, 2018.
- KAUR, H., & GUPTA, R. (2017). A comparative study of file systems BTRFS and ZFS. *International Journal of Computer Applications*, 160(6), 110-114.
- SILVA, E. F. (2021). *Sistemas de Arquivos no Linux: Uma Visão Geral*. Novatec Editora.
- SILVA, A. B. (2021). Armazenamento em nuvem: Uma análise comparativa entre HDs e Cloud Computing. *Revista de Tecnologia da Informação*, 8(2), 10-20.
- MCKUSICK, M. K., NEVILLE-NEIL, G. V., & WATSON, R. N. (2015). *The design and implementation of the FreeBSD operating system* (2nd ed.). Addison-Wesley Professional.
- SILBERSCHATZ, A., GALVIN, P. B., & GAGNE, G. (2013). *Operating system concepts* (9th ed.). Wiley.
- TANENBAUM, A. S., & WOODHULL, A. S. (2014). *Sistemas operacionais modernos* (4th ed.). Pearson Education.
- PRESTON, W., LUNDY, M., & ROBBINS, T. (2019). *Backup & Recovery: Inexpensive Backup Solutions for Open Systems* (2nd ed.). O'Reilly Media.
- MELL, P., & GRANCE, T. (2011). *The NIST Definition of Cloud Computing* (Special Publication 800-145). National Institute of Standards and Technology.
- DUBEY, A., & MISHRA, D. (2018). Backup and recovery: problems and solutions. *International Journal of Advanced Research in Computer Science*, 9(3), 45-48.
- RISTENPART, T., TROMER, E., SHACHAM, H., & SAVAGE, S. (2016). Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds. *ACM Transactions on Information and System Security*, 13(1), 1-33.
- SOUZA, F. C. R., CAMPOS, L. A. D. C., & ZANONI, N. (2019). A Comparative Analysis of Btrfs and ZFS File Systems. In *2019 IEEE 37th International Conference on Computer Design (ICCD)* (pp. 86-92). IEEE
- KERN, M., KARRENBURG, D., & DURIEUX, T. (2020). Bacula: The Open-Source Backup Solution. In *Proceedings of the 11th ACM SIGOPS Asia-Pacific Workshop on Systems* (pp. 10-15).
- RODRIGUES, C. A., SANTOS, L. M., OLIVEIRA, P. S., & COSTA, R. J. (2022). Armazenamento em nuvem: Uma abordagem alternativa ao uso de HDs. In *Anais do Congresso Brasileiro de Tecnologia da Informação* (pp. 30-40). São Paulo, Brasil.

FERREIRA, C. A. (2022). Disponibilidade e acessibilidade na computação em nuvem: Um estudo comparativo. *Revista de Tecnologia e Inovação*, 9(1), 25-35.

MARTINS, F. R. (2022). O futuro do armazenamento em nuvem: Perspectivas e avanços tecnológicos. *Revista de Tecnologia e Inovação*, 12(3), 70-85.

SITES:

KADANSKY, N. Soluções de backup para pequenas empresas. Disponível em: https://www.kadansky.com/files/newsletters/2018/2018_04_30.html.

BÁCULA SYSTEM. Sobre Bacula: Bacula Enterprise Edition Administrator's Guide (2021). Disponível em: <https://www.bacula.org/about-bacula>.

7 APÊNDICES

7.1 APÊNDICE A: EXEMPLO DE SCRIPT DE BACKUP

A seguir está um exemplo de script de backup para realizar o backup de um diretório específico usando o utilitário rsync:

```
#!/bin/bash
```

```
# Diretório de origem
```

```
SOURCE_DIR="/caminho/do/diretorio"
```

```
# Diretório de destino (local ou remoto)
```

```
DESTINATION_DIR="/caminho/do/destino"
```

```
# Executa o backup usando o rsync
```

```
rsync -avz --delete $SOURCE_DIR $DESTINATION_DIR
```

```
# Exibe uma mensagem de conclusão
```

```
echo "Backup concluído com sucesso!"
```

7.2 APÊNDICE B: EXEMPLO DE SCRIPT DE LIMPEZA DE BACKUPS ANTIGOS

Aqui está um exemplo de script de limpeza de backups antigos para remover arquivos de backup com base em uma determinada idade:

```
#!/bin/bash

# Diretório de backups
BACKUP_DIR="/caminho/do/diretorio/de/backups"

# Define a idade máxima dos backups em dias
MAX_AGE_DAYS=30

# Remove os backups mais antigos que a idade máxima
find $BACKUP_DIR -type f -mtime +$MAX_AGE_DAYS -exec rm {} \;

# Exibe uma mensagem de conclusão
echo "Limpeza de backups antigos concluída com sucesso!"
```

8 ANEXOS

8.1 ANEXO A: IMPLEMENTAÇÃO

Criação e automação de BACKUPS com Shell Script - Linux

🕒 Backup é importante para não perder dados

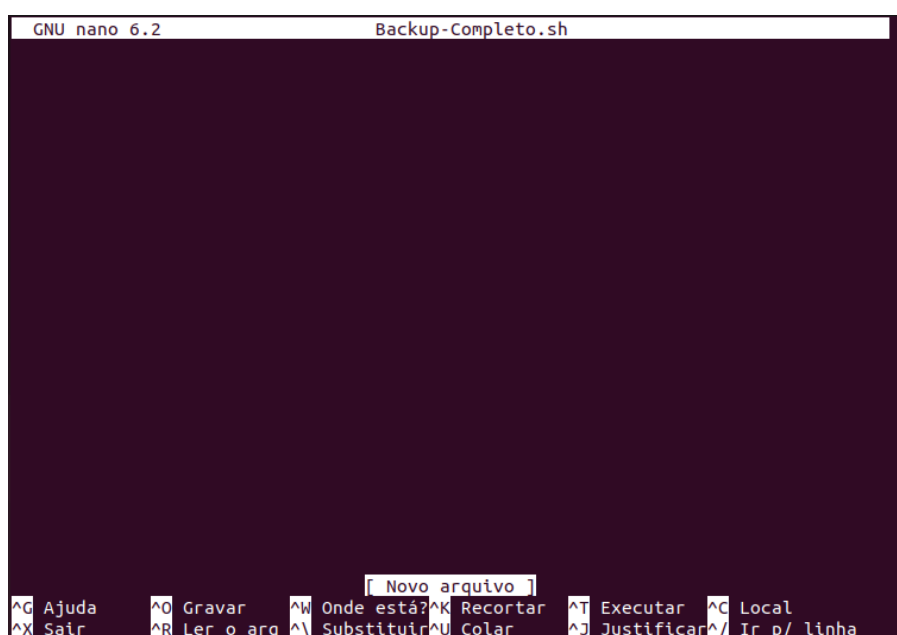
🔍 Existem tipos de backups como completo, incremental, diferencial e espelhado

📄 É possível criar um script em Shell para automatizar backups

💻 O script pode ser customizado para selecionar quais diretórios serão copiados e onde serão armazenados os backups

💾 É importante escolher um destino confiável para armazenar backups, como um HD externo ou pen drive.

```
oem@ServidorBrain:~$ nano Backup-Completo.sh
```



Como esse script é em shell, é necessário passar um *shebang* – ela serve para indicar qual tipo de shell que ele deve fazer a chamada.

Diretório de Destino para onde o backup será transferido:

```
oem@ServidorBrain:~$ ls
Desktop Documentos Downloads Imagens Modelos Música Público snap Vídeos
oem@ServidorBrain:~$ cd Documentos/
oem@ServidorBrain:~/Documentos$ pwd
/home/oem/Documentos
```

Mkdir, make directory; Irá criar uma pasta, ou seja, permite com que eu esconda/guarde arquivos;

-v , verbose; é usada para ativar o modo verbose, em outras palavras o modo detalhado.

Esse é o diretório aonde será armazenado os backups:

```
oem@ServidorBrain:~$ ls
Desktop Documentos Downloads Imagens Modelos Música Público snap Vídeos
oem@ServidorBrain:~$ cd Documentos/
oem@ServidorBrain:~/Documentos$ pwd
/home/oem/Documentos
oem@ServidorBrain:~/Documentos$ sudo su
[sudo] senha para oem:
root@ServidorBrain:/home/oem/Documentos# mkdir -v /mnt/backup
mkdir: foi criado o diretório '/mnt/backup'
root@ServidorBrain:/home/oem/Documentos# |
```

Código para mostrar a Data Atualiza:

```
oem@ServidorBrain:~$ date "+%d"
31
oem@ServidorBrain:~$ date "+%d-%m-%y"
31-05-23
oem@ServidorBrain:~$ date "+%d-%m-%Y"
31-05-2023
```

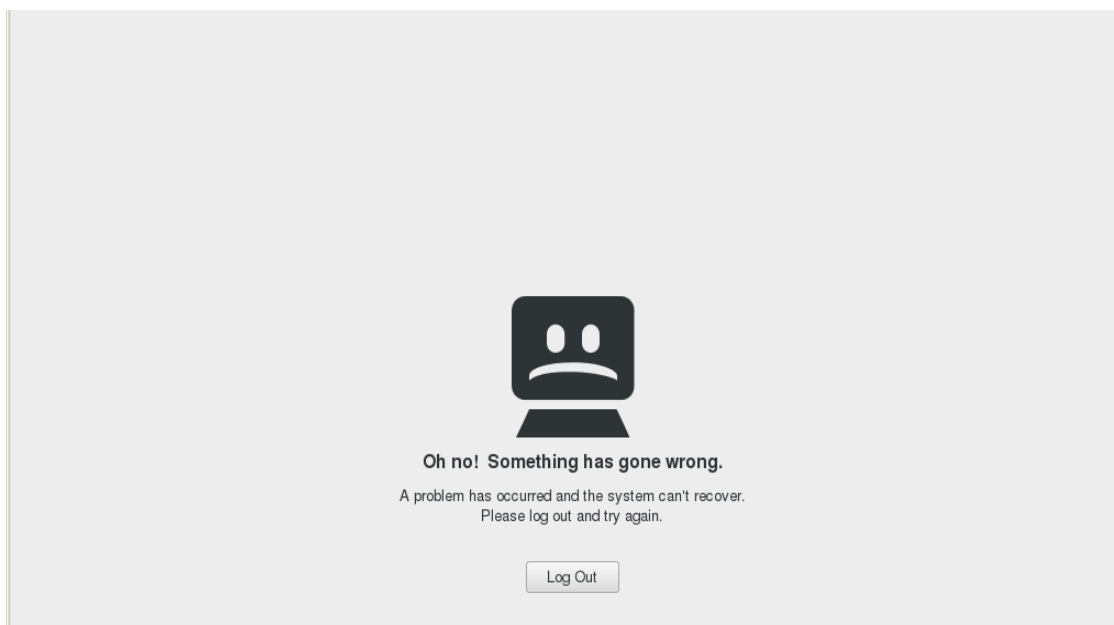
Vamos compactar com o formato “tar” e em seguida compactar no formato “gz”
– gunzip

Os Logs, são registros de todas as ações realizadas no sistema, como falhas, exceções e erros. Um servidor invadido enviará logs para um sistema remoto, dificultando que o invasor apague seus rastros. Para que ele obtenha êxito em

O comando que era para prosseguir, “lsblk” mostra todos os devices contidos no sistema. Contudo ocorreu alguns impasses.

8.2 ANEXO B: DIFICULDADES

Ao reiniciar o sistema para efetuar algumas atualizações necessárias que o sistema pediu ocorreu o seguinte erro:



Para a sua resolução foram feitas pesquisas no idioma inglês, no qual foi encontrado um resultado semelhante ao que está a seguir:

Alt + Ctrl + F3

Ao executar essa combinação de teclas seremos redirecionados ao terminal de comando. A partir daqui, faremos o login com nosso usuário e senha cadastrado previamente.

```
Ubuntu 22.04.1 LTS fulanosilva@brain:
fulanosilva@brain login: FulanoSilva
Password: Coxinha123
Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-50-generic x86_64)
```

```
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage
```

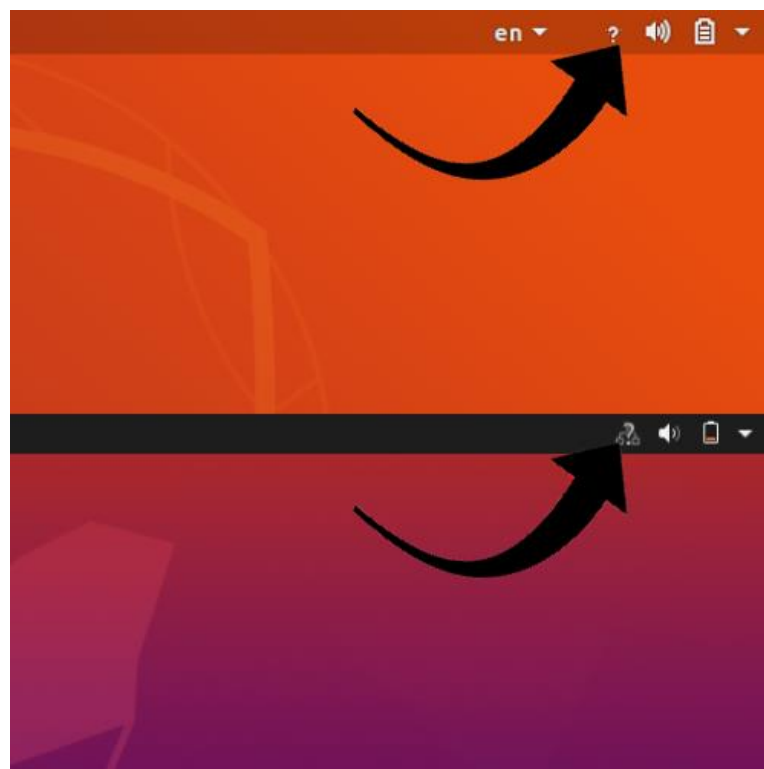
```
1277 updates can be applied immediately.
To see these additional updates run: apt list --upgradable
```

```
Last login: Fri Oct 31 19:26:14 SAST 2022 on tty5
fulanosilva@brain:~$
```

Após acessar o terminal escreva a seguinte série de comandos:

```
fulanosilva@brain:~$ sudo do-release-upgrade
fulanosilva@brain:~$ sudo apt-get update && sudo apt-get upgrade
fulanosilva@brain:~$ sudo apt-get update
fulanosilva@brain:~$ sudo apt-get upgrade
fulanosilva@brain:~$ sudo apt-get full-upgrade
fulanosilva@brain:~$ sudo apt install -f
fulanosilva@brain:~$ sudo apt autoremove && sudo apt clean
fulanosilva@brain:~$ sudo apt autoremove
fulanosilva@brain:~$ sudo apt autoclean
fulanosilva@brain:~$ clear
fulanosilva@brain:~$ sudo dpkg --configure -a
fulanosilva@brain:~$ reboot
```

Como nem tudo são rosas, mesmo após a restauração do sistema, por algum motivo não se conectava a rede de internet de jeito nenhum.



Para resolver esse pequeno e problemático detalhe, estão sendo feitas novas pesquisas. Foi encontrado a seguinte solução:

```
fulanosilva@brain:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
group default qlen 1000
```

```

link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: enp5s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state
UP group default qlen 1000
link/ether a4:5d:36:c2:48:2e brd ff:ff:ff:ff:ff:ff
inet 192.168.207.210/24 brd 192.168.207.255 scope global dynamic noprefixroute
enp5s0
valid_lft 2696sec preferred_lft 2696sec
inet6 fe80::d90e:f65a:e012:53c5/64 scope link noprefixroute
valid_lft forever preferred_lft forever
3: wlp4s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
state UP group default qlen 1000
link/ether 94:0c:6d:bf:53:27 brd ff:ff:ff:ff:ff:ff
inet 192.168.101.166/23 brd 192.168.101.255 scope global dynamic noprefixroute
wlp4s0
valid_lft 6294sec preferred_lft 6294sec
inet6 fe80::400d:b96:2f34:2fc7/64 scope link noprefixroute
valid_lft forever preferred_lft forever
fulanosilva@brain:~$ sudo nano /etc/netplan/00-installer-config.yaml

```

```

GNU nano 6.2 /etc/netplan/00-installer-config.yaml *
# This is network config written by 'subiquity'
network:
  ethernet:
    enp5s0:
      dhcp4: true
  version: 2

```

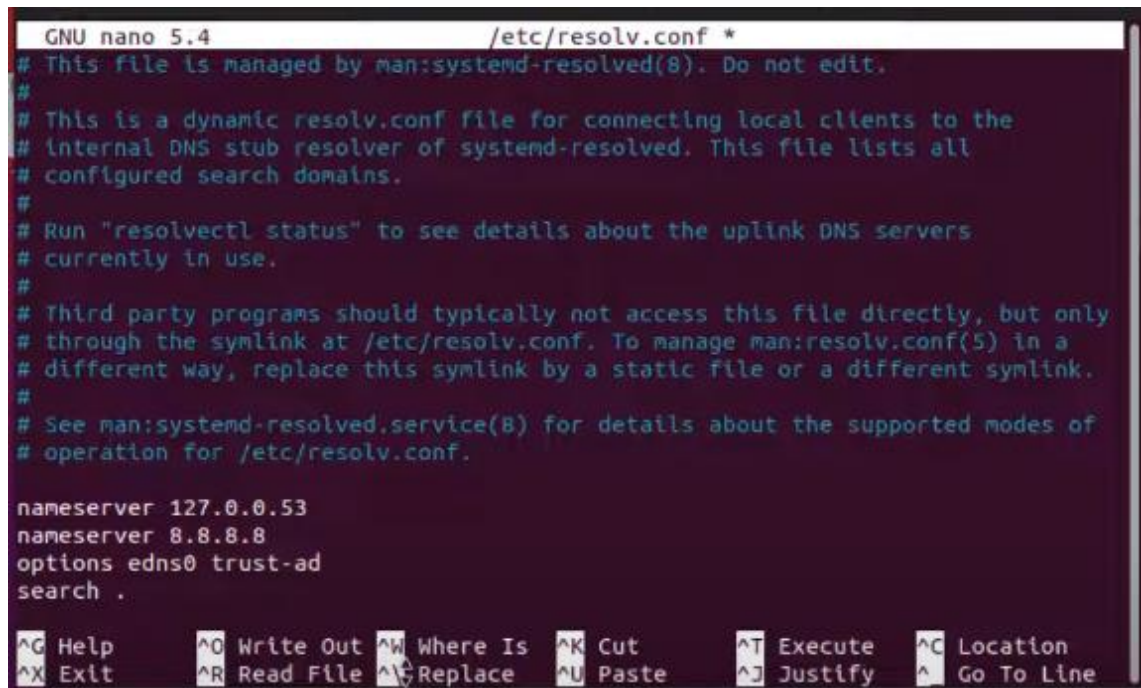
Ao entrar na tela exibida logo acima, se não estiver semelhante a isso, deixe de forma semelhante. Para salvar, aperte CTRL + X e ENTER.

```
fulanosilva@brain:~$ sudo netplan generate
```

```
fulanosilva@brain:~$ sudo netplan apply
fulanosilva@brain:~$ ip a
```

Após isso, basta ver se o ping do google irá responder, e, em seguida, colocar o servidor genérico dele (8.8.8.8).

```
fulanosilva@brain:~$ ping www.google.com
fulanosilva@brain:~$ sudo nano /etc/resolv.conf
```



```
GNU nano 5.4 /etc/resolv.conf *
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs should typically not access this file directly, but only
# through the symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a
# different way, replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
nameserver 8.8.8.8
options edns0 trust-ad
search .

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute  ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify  ^_ Go To Line
```

Para salvar, aperte CTRL + X e ENTER, e aplique o reboot.

```
fulanosilva@brain:~$ Reboot
```

Avaliação do Orientador

Avaliação do Projeto de Trabalho de Conclusão de Curso pela Orientadora:

Nome da orientadora:

A proposta é: () adequada () não adequada () precisa ser readequada.

Nota:

Parecer:

Cuiabá, ____/_____/2023.