

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
CENTRO UNIVERSITÁRIO DO RIO GRANDE DO NORTE
ESPECIALIZAÇÃO EM REDES DE COMPUTADORES**

JOIR MEDEIROS RAMALHO

DESDUPLICAÇÃO:ELIMINANDO A REDUNDÂNCIA DO DADO

NATAL/RN

2013

JOIR MEDEIROS RAMALHO

DESDUPLICAÇÃO:ELIMINANDO A REDUNDÂNCIA DO DADO

Monografia apresentada ao curso de Especialização em Redes de Computadores do Centro Universitário do Rio Grande do Norte como requisito final para a obtenção do título de Especialista em Redes de Computadores.

Orientador:

Aluizio Ferreira da Rocha Neto, M.Sc.

NATAL/RN

2013

JOIR MEDEIROS RAMALHO

DESDUPLICAÇÃO:ELIMINANDO A REDUNDÂNCIA DO DADO

Monografia apresentada ao curso de Especialização em Redes de Computadores do Centro Universitário do Rio Grande do Norte – UNI-RN, como requisito final para a obtenção do título de Especialista em Redes de Computadores.

Aprovado em: 03/03/2013

BANCA EXAMINADORA:

Prof.M.Sc.Aluizio Ferreira da Rocha Neto.

Membro examinador

Membro examinador

Dedico este trabalho a meus pais, minha esposa e filha, que souberam entender minha ausência, me incentivaram nesta contínua busca pelo conhecimento na certeza de que os objetivos serão alcançados.

AGRADECIMENTO

Agradeço a Deus e a Nossa Senhora por terem me proporcionado mais essa conquista em minha vida.

Aos meus pais Jairo e Ivone, por terem me educado e incentivado a estudar em todos os momentos, e em especial durante essa pós graduação.

A minha esposa Monna Lisa, que por saber da importância desse curso para mim, esteve presente ao meu lado com paciência e carinho.

A minha filha Marianna, que sentiu o pai ausente em alguns momentos, ocupado nas tarefas de estudo, mas entendeu e usou como exemplo.

Ao Professor Aluizio Rocha, que apesar das inúmeras atividades acadêmicas, nunca se furtou a prestar esclarecimentos e orientações fundamentais ao desenvolvimento deste trabalho e no decorrer de todo o curso.

E por fim, ao Professor Ricardo Kleber, amigo de longa data, responsável por eu ter participado desta Especialização em Redes.

RESUMO

Devido ao crescimento da quantidade de informação em formato digital que dispomos surgiu um desafio para os administradores de TI, que são responsáveis por armazená-la, disponibilizá-la e mantê-la de acordo com as normas vigentes. As técnicas convencionais de cópia e armazenamento de dados já não conseguem mais atender as demandas atuais e por isso foram desenvolvidas uma série de técnicas denominadas de deduplicação, que permitem manter a informação protegida, podendo ser transportada e replicada com mais facilidade, segurança e rapidez, uma vez que a duplicidade é descartada. Além disso, graças a essas técnicas, é possível utilizar de forma racional os equipamentos de armazenamento de dados disponíveis nas organizações bem como os links de dados, evitando-se aquisições desnecessárias e aumentando a vida útil dos mesmos. Este trabalho aborda como essas técnicas são aplicadas, seus benefícios e demonstra o funcionamento de alguns serviços disponíveis na internet que existem graças a utilização das mesmas.

Palavras-chave: Deduplicação, *Backup*, *Restore*, Replicação, Recuperação de desastre.

ABSTRACT

Due to the growing amount of digital information that we have emerged a challenge for IT administrators, who are responsible for storing it, make it available and keep it in line with current standards, because the conventional techniques of copying and storing data can no longer meet the demands more current and therefore have been developed a number of techniques referred to as deduplication, which allow to keep information secure and can be transported and replicated more easily, safely and quickly, since the duplication is discarded. Moreover, thanks to these techniques, it is possible to use rationally the equipment storage available in organizations as well as data links, avoiding unnecessary acquisitions and increasing the useful life thereof. This work presents how these techniques are applied, its benefits and demonstrates the functioning of some services that are available on the internet through their use.

Keywords: Deduplication, Backup, Restore, Replication, Disaster Recovery

LISTA DE FIGURAS

Figura 1 Dispositivos eletrônicos se comunicam usando internet.....	11
Figura 2 Site remoto, replicação de dados, Recuperação de disastre	15
Figura 3 Fluxograma da deduplicação	21
Figura 4 Informação original x Redundância do dado retirada	21
Figura 5 Evitando blocos duplicados	22
Figura 6 Métodos de deduplicação	23
Figura 7 Momento da deduplicação	25
Figura 8 Técnica de backup x tempo de restauração.....	27
Figura 9 Linha do tempo para RTO e RPO	29
Figura 10 Tela de cadastro para criar conta no Dropbox.....	31
Figura 11 Área de trabalho do Dropbox.....	31
Figura 12 Área de trabalho do Google Drive.....	32
Figura 13 Tela do aplicativo Truecrypt	33
Figura 14 Estatísticas de conversação do arquivo de 1 MByte	34
Figura 15 Estatísticas de conversação do arquivo de 2 MBytes	34
Figura 16 Concatenando arquivos criptografados para upload	35
Figura 17 Arquivos preparados para upload	35
Figura 18 Estatísticas de conversação do arquivo de 4 Mbytes (concatenado uma vez).....	36
Figura 19 Estatísticas de conversação do arquivo de 8 Mbytes (concatenado duas vezes)	36
Figura 20 Estatísticas de conversação do arquivo de 16 Mbytes (concatenado três vezes)	36
Figura 21 Estatísticas de conversação do arquivo de 1 MByte	37
Figura 22 Estatísticas de conversação do arquivo de 2 MBytes	37
Figura 23 Estatísticas de conversação do arquivo de 4 Mbytes (concatenado uma vez).....	38
Figura 24 Estatísticas de conversação do arquivo de 8 Mbytes (concatenado duas vezes)	38
Figura 25 Estatísticas de conversação do arquivo de 16 Mbytes (concatenado três vezes)	38
Figura 26 Arquivo de 8 Megabytes sendo enviado para Dropbox.....	39
Figura 27 Arquivo de 8 Megabytes sendo enviado para Dropbox após edição	39
Figura 28 Arquivo de 8 Megabytes sendo enviado para Google Drive	40
Figura 29 Arquivo de 8 Megabytes sendo enviado para Google Drive após edição..	40

LISTA DE TABELAS

Tabela 1 Simulacao de rotina diária de backup full	14
Tabela 2 Backup Diferencial em relação ao primeiro backup FULL	16
Tabela 3 Backup Incremental após backup FULL e ao último Incremental.....	17
Tabela 4 Características dos tipos de backups	18

SUMÁRIO

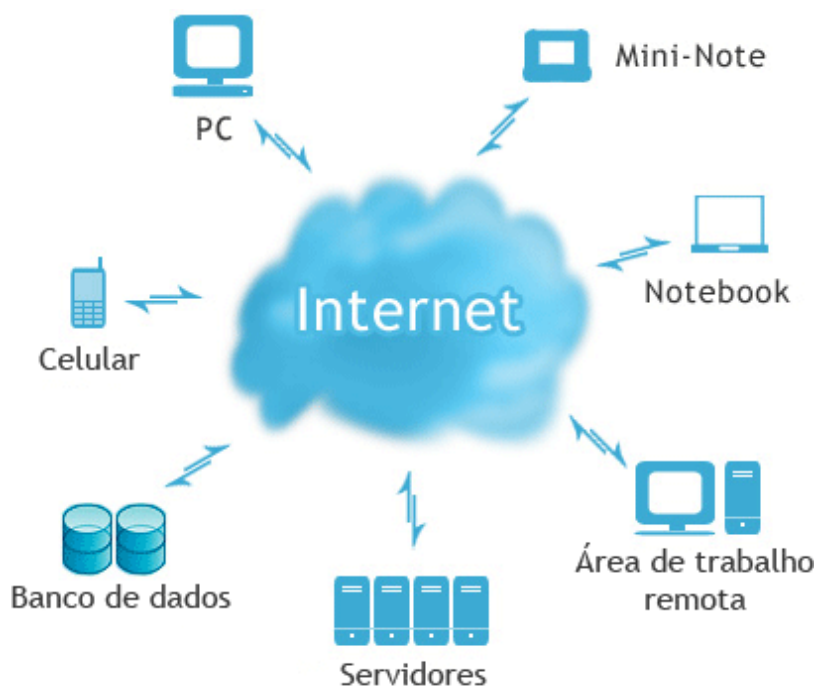
1	INTRODUÇÃO.....	11
1.1	Objetivos	13
1.1.1	Objetivos específicos.....	13
1.2	Organização do trabalho	13
2	REFERENCIAL TEÓRICO.....	14
2.1	Funcionamento básico do backup.....	14
2.2	Compressão, criptografia e transferência de dados.....	19
2.3	Desduplicação.....	20
2.3.1	Métodos da desduplicação: <i>File-level x Block-level x Byte-level</i>	22
2.3.2	Momentos da desduplicação: <i>in-line x post-process x on Client</i>	24
2.4	Políticas de desduplicação.....	26
2.5	Impacto indireto da desduplicação.....	26
2.6	Recursos viabilizados pela desduplicação	28
2.7	RTO e RPO.....	29
3	ESTUDO DE CASO	30
3.1	Dropbox.....	30
3.2	Google drive	31
3.3	Metodologia dos testes	32
3.4	Sequência dos testes.....	33
4	CONCLUSÃO	42
	REFERÊNCIAS BIBLIOGRÁFICAS	43

1 INTRODUÇÃO

Cada vez mais depende-se de algum equipamento eletrônico, começando pelo rádio, o televisor e vários outros utensílios domésticos que são controlados por circuitos eletrônicos, e desde que ocorreu a popularização dos microcomputadores, tornaram-se realidade em nossos lares produtos que vão desde os desktops, notebooks, *tablets*, máquinas fotográficas digitais até os celulares que são utilizados para acessar informações através da internet.

Com todos esses equipamentos permeando nossas vidas ocorreu uma digitalização da informação de quase tudo que se vê, ouve e se produz em nosso dia-a-dia, e isto gera cada vez mais dados que precisam ser armazenados para que possam ser acessados quando necessário.

Figura1: Dispositivos eletrônicos se comunicando através da Internet



Fonte: Autor

Então surgiu um problema não só para os usuários domésticos como também para as empresas: Como armazenar e controlar toda essa informação em formato digital que são produzidos e recebidos? Devem ser feitas cópias de segurança?

De acordo com [10],

Cópia de segurança (em inglês: *backup*) é a cópia de dados de um dispositivo de armazenamento a outro para que possam ser restaurados em caso da perda dos dados originais, o que pode envolver apagamentos acidentais ou corrupção de dados.

A indústria, ao longo dos últimos anos, tem respondido a essa demanda fabricando dispositivos cada vez mais rápidos e como maior capacidade de armazenamento, mas a quantidade de informação produzida está crescendo mais rápido que a capacidade dos equipamentos que ora dispomos, forçando-se que se invista cada vez mais ao invés de utilizá-los de uma forma mais racional.

Quando a indústria de software percebeu essa questão foram desenvolvidos aplicativos que fossem capazes de comprimir as informações que estão armazenadas, os conhecidos compactadores, que são utilizados diariamente, como o padrão ZIP e tantos outros, para que fosse possível seu armazenamento e transporte quando necessário de forma mais eficiente.

Tal iniciativa não foi suficiente uma vez que além de compactar as informações era preciso evitar arquivos em duplicidade e por isso surgiram várias técnicas para analisá-los conhecidas como *deduplication*, em português, desduplicação, que pretende manter apenas uma instância de cada um, tornando possível seu armazenamento e transporte mais eficiente. Segundo[5],

Em computação, a desduplicação de dados é uma técnica de compressão de dados para eliminar cópias duplicadas de dados repetidos. A técnica é utilizada para melhorar a utilização do armazenamento e também pode ser aplicada aos dados que a rede transfere para reduzir o número de bytes a serem enviados. No processo de redução de redundância, pedaços únicos de dados, ou padrões de byte, são identificados e armazenados durante um processo de análise. Como a análise continua, pedaços outros são comparados com a cópia armazenada e, sempre que ocorrer uma correspondência, o pedaço redundante é substituído por uma referência pequena que aponta para o bloco armazenado. Dado que o padrão de byte pode ocorrer mesmo dezenas, centenas ou mesmo milhares de vezes (a frequência de partida é dependente do tamanho do bloco), a quantidade de dados que devem ser armazenados ou transferidos pode ser grandemente reduzido.

Essa tecnologia também é aplicada em ferramentas de *backup* e *Data Archiving* que se tornam bastante eficientes e por vezes até são viabilizadas somente se tais técnicas forem utilizadas.

Visando-se utilizar as técnicas de deduplicação, este trabalho mostrará a melhor forma de aplicá-las para o uso racional do equipamento existente, buscando-se melhores resultados em face das limitações orçamentárias que as empresas e órgãos públicos dispõem para investir em TI.

1.1 Objetivos

Este projeto tem por objetivo fazer uma análise das técnicas e apontar as melhores práticas quando da aplicação da deduplicação aos arquivos tanto nos storages, às rotinas de backup e aos procedimentos de Archive utilizados no dia-a-dia nas organizações e em nossas vidas.

1.1.1 Objetivos específicos

Para alcançar o objetivo deste trabalho seguiremos as seguintes etapas:

- Descrição das técnicas de backup e deduplicação;
- Análise do momento, local e como são aplicadas as técnicas;
- Apresentação das vantagens alcançadas com deduplicação;
- Verificação dos benefícios indiretos com a utilização;
- Outras tarefas viabilizadas após deduplicação

1.2 Organização do trabalho

Este trabalho encontra-se organizado em quatro capítulos. O Capítulo 1 apresenta a contextualização do trabalho, o problema e os objetivos da monografia. O Capítulo 2 apresenta o referencial teórico de todas as tecnologias aplicadas, como funcionam, quando e como devem ser utilizadas, os ganhos diretos e indiretos e aquilo que sua aplicação torna possível. O Capítulo 3 apresenta-se um estudo de caso, ou seja, aplicações práticas que permitiram que serviços fossem prestados aos usuários. O Capítulo 4 apresenta as conclusões deste trabalho.

2 REFERENCIAL TEÓRICO

2.1 Funcionamento básico do backup

Anteriormente a aplicação das técnicas de deduplicação nas empresas, gerentes de TI enfrentavam problemas para manter de forma segura e organizada a informação produzida, preocupavam-se em fazer o *backup* diário, permitindo a restauração dos dados em caso de perda. No entanto, as ferramentas existentes (NTBackup - Windows 2003 Server e outras) se limitavam a verificar se os arquivos e pastas selecionados para cópia haviam sido alterados em função de um *flag*, uma propriedade do arquivo, que quando marcado como "A" significava que o mesmo havia sido criado ou alterado desde a última cópia e portanto deveria ser copiado.

Existem três formas básicas de se fazer *backup*, o *FULL* ou completo, que seleciona para cópia arquivos existentes em uma pasta selecionada, o Diferencial que copiará somente os arquivos de uma pasta que foram criados ou alterados desde a última cópia completa e o do tipo incremental, que selecionará apenas os arquivos manipulados desde o último *backupFULL* ou incremental.

Criar e executar diariamente essas rotinas de *backup*, quer sejam completos, diferenciais ou incrementais, não representa uma tarefa difícil, o problema é manter esses arquivos e suas diversas versões em locais seguros, uma vez que se tornam cada maiores e exigem novas mídias para armazenamento.

Tabela 1 Simulação de rotina diária de *backup full*

Tipo FULL	Dia 1 (FULL-1)	Dia 2 (FULL-2)	Dia 3 (FULL-3)	Total
Arquivos	Mídia_1	Mídia_2	Mídia_3	3 Mídias
Arquivo .doc	2MBytes	2MBytes	2MBytes	6MBytes
Arquivo .xls	3MBytes	3MBytes	3MBytes	9MBytes
Arquivo .pdf	5MBytes	5MBytes	5MBytes	15 MBytes
Arquivo .ppt (10 MBytes)	-	-	novo 10 MBytes	10 MBytes
Total de espaço em disco	10 MBytes	10 MBytes	20 MBytes	40 MBytes

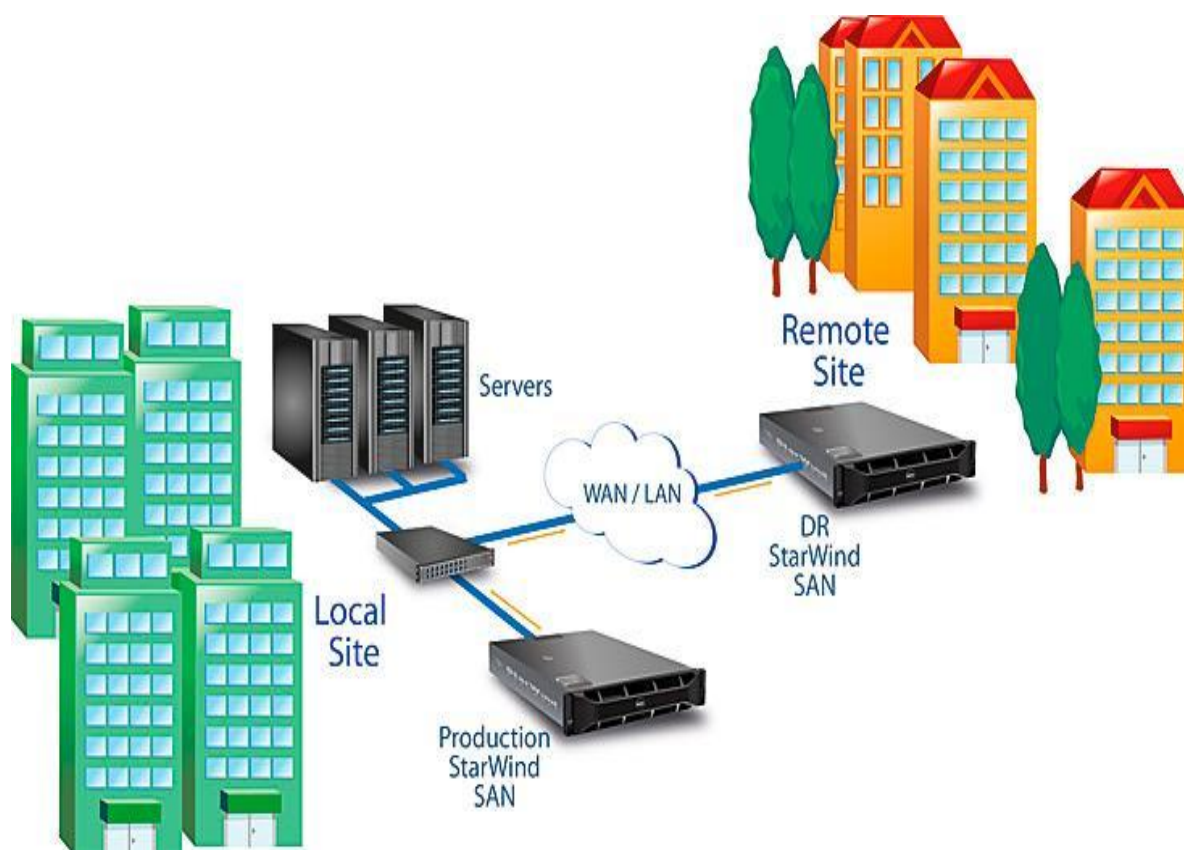
Fonte: Autor

Analisando a Tabela 1 percebe-se que, executando-se um *Backup* do tipo *FULL* (completo) diariamente, ao final de três dias estarão armazenados 20

MBytes de arquivos no servidor mais 40 MBytes de *Backup* em mídias removíveis. Além da redundância de informação contida nos arquivos de Backups, precisamos ainda transportar uma dessas mídias removíveis para um local diverso (*offsite*) daquele onde se encontram os dados, ou seja, devem ser mantidas protegidas caso seja necessário proceder um *disasterrecovery*. Segundo a [16],

Plano de Recuperação de Desastre (*Disasterrecovery* - DR)"é o processo, políticas e procedimentos relacionados à preparação para recuperação ou manutenção de infraestrutura de tecnologia fundamental para uma organização após um desastre natural ou induzido pelo homem. A recuperação de desastres é um subconjunto de continuidade de negócios. Enquanto a continuidade dos negócios envolve planejamento para manter todos os aspectos de uma empresa funcionando no meio de eventos perturbadores, recuperação de desastres se concentra nos sistemas de TI ou tecnologia que suportam funções de negócios."

Figura2 Site remoto, replicação de dados, Recuperação de desastre



Fonte: [25]

Tabela2 Backup Diferencial em relação ao primeiro backup FULL

Tipo Diferencial	Dia 1 (Diferencial-1)	Dia 2 (Diferencial-2)	Dia 3 (Diferencial-3)	Total
Arquivos	Mídia_1	Mídia_1	Mídia_1	1 Mídia removível
Arquivo .doc (2MBytes)	Alterado 2MBytes	Alterado Dia 1 2MBytes	Alterado Dia 1 2MBytes	6MBytes
Arquivo .xls (3MBytes)	-	Alterado 3MBytes	Alterado Dia 2 3MBytes	6MBytes
Arquivo .pdf (5MBytes)	-	-	-	-
Arquivo .ppt (10 MBytes)	-	-	novo 10 MBytes	10 MBytes
Total de espaço em disco	2MBytes	5MBytes	15 MBytes	22 MBytes

Fonte: Autor

Observando a Tabela 2 percebe-se que, executando-se um *Backup* do tipo Diferencial apenas o que foi adicionado ou alterado diariamente em relação ao último backup FULL será copiado, após três dias estarão armazenados 20 MBytes de arquivos no servidor mais 22 MBytes de *Backup* em mídia removível. Foi evitada parcialmente a redundância de informação contida nos arquivos de Backups e por serem menores ocupam menos espaço e necessitam de menos mídias removíveis. Nesse caso precisa-se apenas guardar o último backup FULL e o último backup Diferencial caso seja preciso restaurá-lo por completo. Se fosse feito o backup FULL seriam armazenados 40 MBytes. Além disso, não existe versionamento, ou seja, não são mantidas versões dos arquivos a cada dia que é feito o backup, impossibilitando a restauração de um determinado arquivo em dado momento do tempo.

Entendendo a Tabela 3 percebe-se que, executando-se um *Backup* do tipo Incremental apenas o que foi adicionado ou alterado diariamente em relação ao último backup FULL ou incremental será copiado, após três dias estarão armazenados 20 MBytes de arquivos no servidor mais 17 MBytes de *Backup* em mídia removível. Foi evitada a redundância de informação contida nos arquivos de Backups. Nesse caso, é necessário guardar o último *backup FULL* e todos os

backups incrementais caso seja preciso restaurá-lo por completo. Dessa forma, existe versionamento, ou seja, foram mantidas versões dos arquivos a cada dia que foi feito o backup, permitindo a restauração de um determinado arquivo em dado momento do tempo, por exemplo: é possível restaurar a versão do Arquivo.doc tanto no dia 1 quanto no dia 3, e isso pode ser o que o usuário está precisando.

Tabela3 Backup Incremental após backup FULL e ao último Incremental

Tipo Incremental	Dia 1 (Incremental-1)	Dia 2 (Incremental-2)	Dia 3 (Incremental-3)	Total
Arquivos	Mídia_1	Mídia_1	Mídia_1	1 Mídia removível
Arquivo .doc (2MBytes)	Alterado 2MBytes	-	Alterado Dia 3 2MBytes	4MBytes
Arquivo .xls (3MBytes)	-	Alterado 3MBytes	-	3MBytes
Arquivo .pdf (5MBytes)	-	-	-	-
Arquivo .ppt (10 MBytes)	-	-	novo 10 MBytes	10 MBytes
Total de espaço em disco	2MBytes	3MBytes	12 MBytes	17 MBytes

Fonte: Autor

Após serem executados os três tipos básicos de backup pode ser feita uma comparação entre os resultados obtidos, as vantagens e desvantagens de cada um deles, conforme Tabela 4.

Percebe-se após a análise das características e resultados de cada tipo de backup que devido ao crescimento da quantidade e tamanho dos arquivos a serem copiados, torna-se necessária a implementação de novas técnicas que consigam evitar ainda mais a duplicidade de arquivos a serem armazenados, uma vez que comparar arquivos por data e hora ou mesmo pelo *flag* torna a tarefa diária do backup pouco eficiente.

Tabela4 Características dos tipos de backups

Tipos de backup	FULL	Diferencial	Incremental
Tempo de execução (janela de backup)	maior	crescente	menor
Periodicidade de execução (política da empresa)	mensal	semanal	diária
Quantidade de cópias armazenadas	limitada ao espaço das mídias	maior	maior ainda
Tempo de retenção	limitada ao espaço das mídias	até próximo FULL	até próximo FULL
Capacidade de armazenamento das mídias	menor	maior	maior ainda
Rotatividade da mídia removível	menor	maior	maior ainda
<i>Overhead</i> sobre servidores em produção	maior	menor	menor ainda
Transferência de dados (<i>Backupoffsite</i> (Nuvem))	mais demorada	demorada	mais rápida
Pontos de restauração (versionamento)	único	dois	vários

Fonte: Autor

Por isso, foram desenvolvidas as técnicas de deduplicação que visam comparar arquivos através de uma assinatura digital calculada para cada um ou partes deles. Essa assinatura, conhecida por *HASH*, permite aos softwares de backup comparar arquivos e considerá-los iguais mesmo que tenham nome, data de criação ou alteração (*timestamp*) diferentes. Segundo a [20],

HASH significa "uma sequência de bits geradas por um algoritmo de dispersão, em geral representada em base hexadecimal, que permite a visualização em letras e números (0 a 9 e A a F). O conceito teórico diz que hash é a transformação de uma grande quantidade de informações em uma pequena quantidade de informações.

Essa sequência busca identificar um arquivo ou informação unicamente. Por exemplo, uma mensagem de correio eletrônico, uma senha, uma chave criptográfica ou mesmo um arquivo. É um método para transformar dados de tal forma que o resultado seja (quase) exclusivo. Além disso, funções

usadas em criptografia garantem que não é possível a partir de um valor de hash retornar à informação original.

Se for gerado o *HASH* MD5 da *string* “Hash gerado a partir do algoritmo MD5” o resultado do algoritmo será: c5788ce8218239c6a1e9da4261e6874d

Mas, se a *string* for “Hash gerado a partir do algoritmo MD5.”, apenas acrescentando um ponto no final, o *HASH* MD5 gerado será totalmente diferente: ec79eda8e96df85bde4455fbcbbddca59

A geração do *HASH* para um arquivo ou parte dele (*chunk*) é a base de todos os algoritmos usados nas técnicas de deduplicação.

2.2 Compressão, criptografia e transferência de dados

Após a definição da política de backup, quando são escolhidos os tipos de backup a serem utilizados, feita a seleção dos arquivos a serem copiados, o horário de execução dos *jobs*, o prazo e local de retenção de mídias além de outras questões, percebeu-se que poderiam ser utilizadas técnicas de compressão de dados, permitindo que fosse possível armazenar ainda mais informação nas mídias disponíveis. Segundo a [13],

Compressão de dados "é o ato de reduzir o espaço ocupado por dados num determinado dispositivo. Essa operação é realizada através de diversos algoritmos de compressão, reduzindo a quantidade de Bytes para representar um dado, sendo esse dado uma imagem, um texto, ou um arquivo (ficheiro) qualquer. Comprimir dados destina-se também a retirar a redundância.

Uma vez que os dados estavam sendo submetidos a processamento para compressão, os fabricantes de *softwares de backup* perceberam que poderiam, caso o administrador de TI determinasse, fazer a criptografia dos dados que estavam sendo copiados. Tal iniciativa permitia que as informações contidas nos arquivos dos usuários ou organizações pudessem ser transportados para locais offsite em mídias óticas e ou magnéticas evitando problemas caso ocorra extravio ou interceptação.

Após submeter os dados aos processos de compressão e criptografia, torna-se seguro o envio e armazenamento para sites existentes na web, tais como Dropbox.com, AmazonSimpleStorage Service (Amazon S3), Google Drive entre outros. Tais serviços oferecem quantidades limitadas gratuitas ou cotas maiores caso o usuário desejar pagar por isso, e como será visto a seguir se utilizam de

técnicas de deduplicação para que possam fornecer esse tipo de serviço de forma viável, tanto técnica quanto financeiramente.

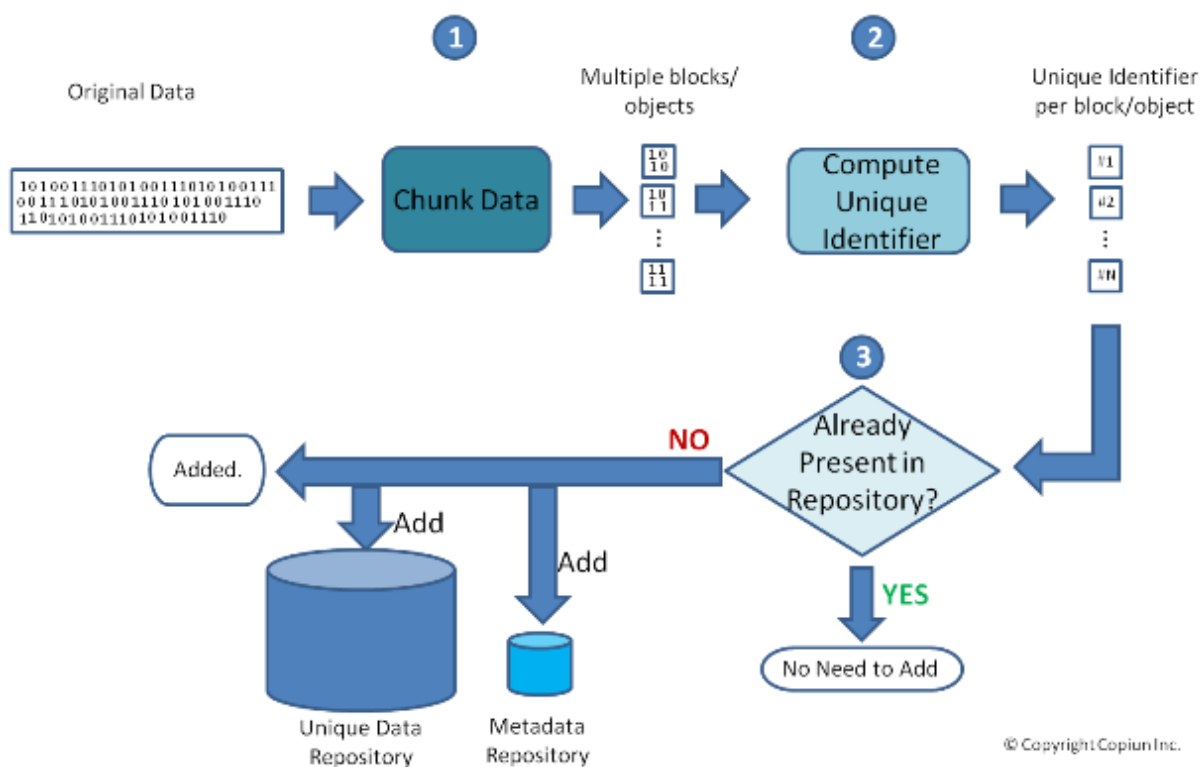
2.3 Deduplicação

Graças a evolução da indústria de software e a utilização de técnicas deduplicação tornou-se possível a economia de espaço em disco, a redução dos investimentos em *storages* e a melhor utilização dos links de dados, evitando-se o armazenamento de dados redundantes.

Durante o processo de deduplicação, o fluxo de dados é dividido em blocos e uma assinatura digital (*HASH*) é criada para cada um, o que permitirá identificá-los unicamente. Essa assinatura é armazenada em um repositório indexado, e cada vez que nova informação chegar para ser gravada no *storage* é feita uma checagem para verificar se ela já existe. Se for encontrada pelos algoritmos significa que é duplicata e não deve ser armazenada novamente, apenas um ponteiro será gravado apontando para o bloco pré-existente toda vez que isso ocorrer, caso seja uma nova informação será gravada normalmente no disco e sua assinatura no repositório de *HASH*.

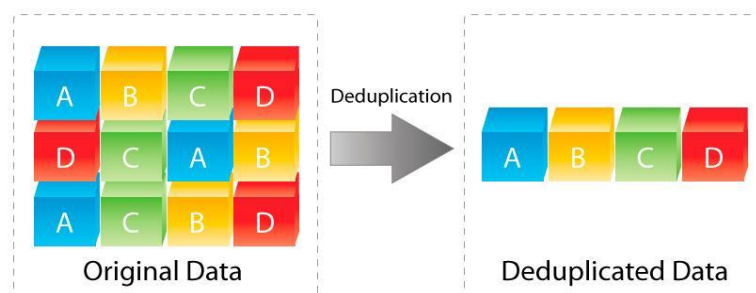
Algumas técnicas de deduplicação dividem cada arquivo em blocos de tamanho fixo e outros em blocos de tamanho variável. Quando menor forem os blocos mais repetições serão encontradas e maior será a economia de espaço em disco, contudo a redução da performance das ferramentas de *backup* e *restore* será significativa. Por isso, fabricantes de software tentam determinar o tamanho ideal do bloco fixo para cada tipo de dado ou usar um bloco de tamanho variável para qualquer tipo de dado.

Figura 3 Fluxograma da deduplicação



Fonte: [3]

Figura 4 Informação original x Redundância do dado retirada



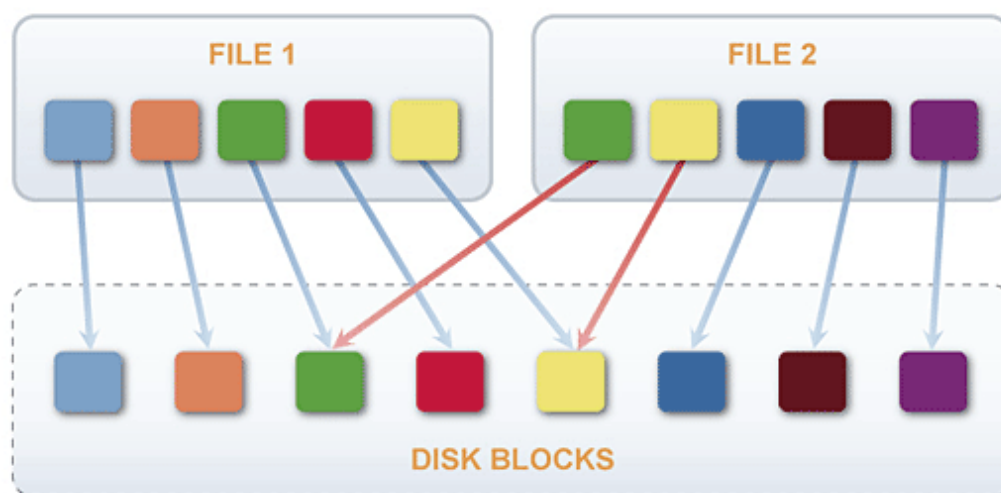
Fonte: [2]

A aplicação dessas técnicas, segundo os fabricantes, permite otimizar a capacidade de armazenamento em alguns casos entre 80 e 90%.

Aplicações de backup que utilizam deduplicação são as que têm apresentado os melhores resultados, uma vez que a execução de rotinas diárias de backup, como apresentadas anteriormente, retornam pouca alteração nos arquivos selecionados para serem copiados e, portanto apenas apontadores para os arquivos

previamente copiados serão armazenados. E caso a forma como ocorre a comparação seja *Block-level*, apenas as partes dos arquivos que foram alterados serão armazenados. Tamaña eficiência se traduzirá em economia cada vez maior em relação aos tipos convencionais de se fazer backup.

Figura 5 Evitando blocos duplicados



Fonte: [2]

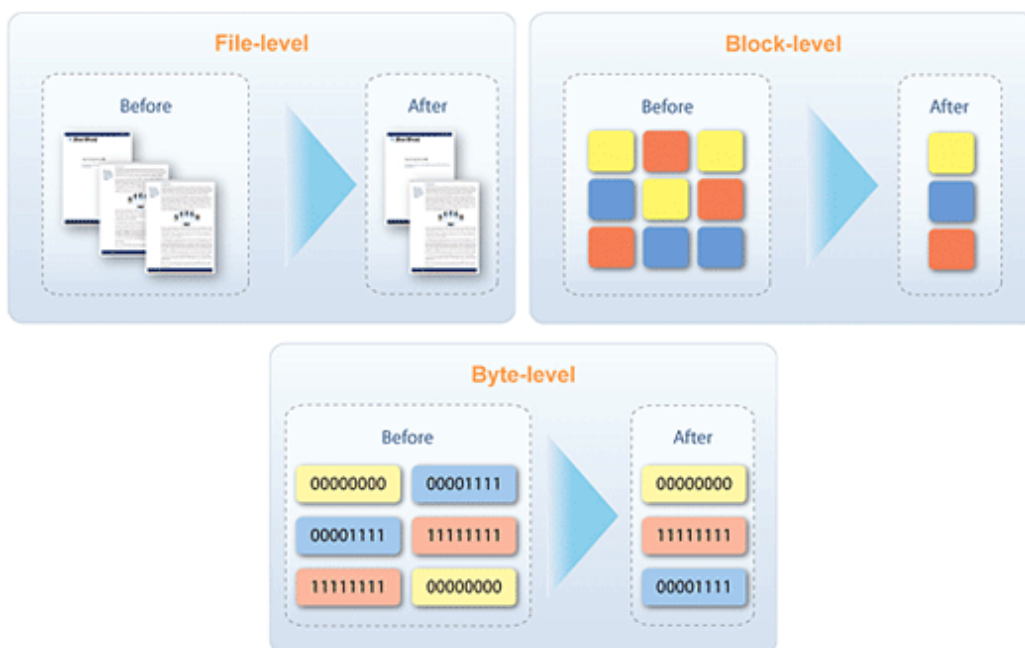
2.3.1 Métodos da desduplicação: *File-level* x *Block-level* x *Byte-level*

A desduplicação de dados pode operar sobre arquivos, blocos de dados ou em nível de byte, e assim definirá o tamanho do fragmento de dado que será checado para evitar duplicidade. Para cada pedaço será criado um *HASH* que será comparado com uma tabela de assinaturas armazenadas.

Teoricamente, quanto mais detalhada a análise, mais alta será a taxa de duplicidade evitada, mas na prática cada forma de fazer essa divisão da informação tem suas particularidades.

A desduplicação *File-level*, que significa em nível de arquivo, é executada mais rapidamente, pois requer menos poder de processamento e as assinaturas são relativamente fáceis de serem geradas, no entanto, se apenas um byte de um arquivo for alterado, o *HASH* gerado será diferente e todo o arquivo será armazenado, ou seja, será praticamente duplicado.

Figura 6 Métodos de deduplicação



Fonte: [2]

Caso seja utilizada a técnica de comparação *Block-level*, que quer dizer em nível de bloco, cada arquivo a ser gravado é dividido em várias sequências de blocos de tamanho fixo ou variável e, se apenas pequenas diferenças são encontradas entre os arquivos antes armazenados e o que está sendo copiado agora, apenas fragmentos alterados serão armazenados. Na média, segundo um fabricante, técnicas de *file-level* resultam em taxas de economia de espaço de 5:1 enquanto as de *Block-level* retornam taxas de até 20:1.

Esta técnica necessita de mais poder de processamento e memória que a anterior, uma vez que o número de pedaços identificados, ou seja, que tiveram assinaturas armazenadas cresce rapidamente, exigindo que o *storage* e o *host* que executam o software de deduplicação tenham capacidade de executar os algoritmos sem degradar o funcionamento dos sistemas como um todo.

É possível que o mesmo *HASH* seja gerado para pedaços de informação diferentes, o que traria problemas para as soluções que utilizam deduplicação. Isso é chamado de "colisão de *hash*" e caso ocorra fará com que nova informação não seja gravada criando inconsistência nos dados armazenados no *storage*.

Com relação ao tamanho do bloco de informações a ser processado, poderá ser definido entre 4 e 256kb ou qualquer outro tamanho. Quanto maior o tamanho do bloco menos recursos de processamento serão necessários, mas a compressão

será menor. Por outro lado, quanto menor o tamanho do blocos a serem analisados, mais processador e memória serão exigidos e os resultados obtidos serão melhores.

Quando se aplica o método de comparação Byte-level, ou seja, os arquivos são comparados byte a byte, não se faz necessária uma maior capacidade de processamento, e embora seja mais precisa tal técnica é a mais demorada. Deve ser utilizada apenas na fase de *post-process*, como veremos adiante.

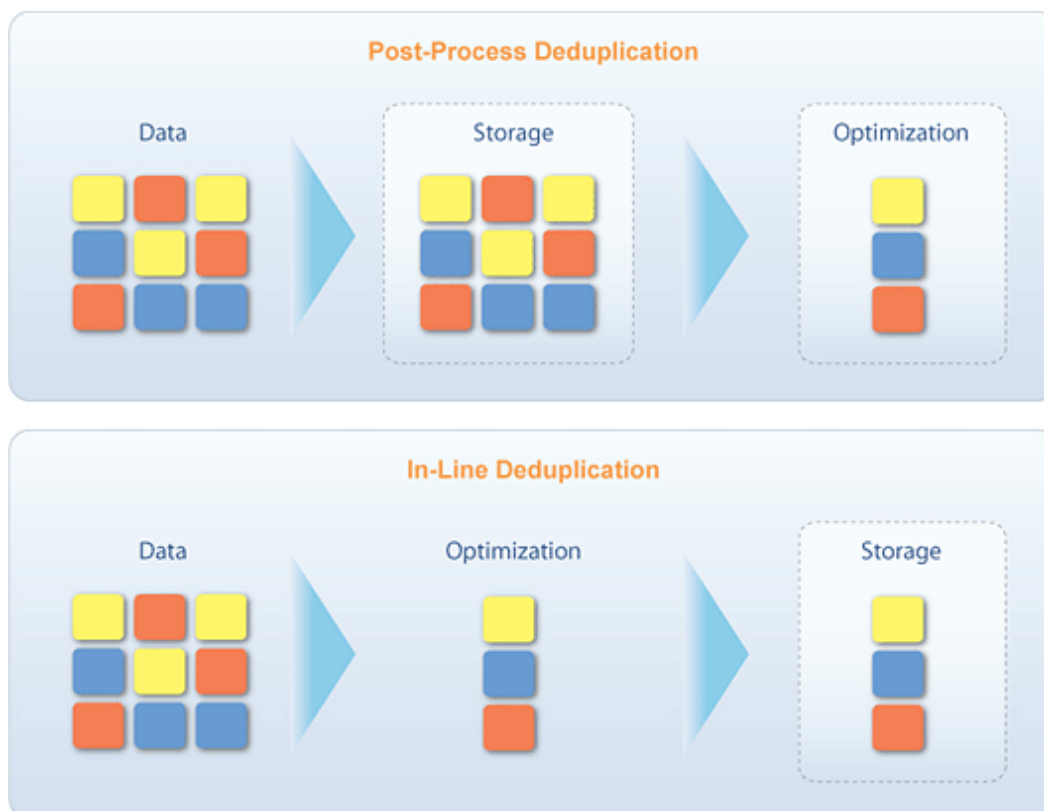
2.3.2 Momentos da deduplicação: *in-line x post-process x onClient*

A deduplicação pode ser aplicada aos arquivos em três momentos: *in-line*, que quer dizer antes que o dado seja gravado ou *on-client*, e a *Post-process*, na qual os dados serão analisados em busca de duplicidade após terem sido gravados. Cada uma das formas a serem utilizadas tem suas vantagens e desvantagens.

Quando utiliza-se *Post-process*, cada bloco de dados que chega ao *storage* é gravado nos discos integralmente, como se não existisse qualquer recurso de deduplicação. Então, depois um processo separado faz a leitura dos blocos recentemente gravados e verifica se encontra duplicidade. Se o mesmo já foi gravado anteriormente, ou seja, se a informação é duplicata, o arquivo gravado por último é apagado e uma referência ao arquivo original é gravada no disco. Caso contrário, se a informação é nova, não sendo duplicata, nenhuma alteração é executada.

Dessa forma o processo de gravação de dados no *storage* ocorre de forma mais rápida, no entanto será necessário dispor de mais espaço em disco livre para que seja gravada a informação nos discos e, quando é feita a comparação ocorre ainda mais leitura e gravação em busca das duplicidades.

Figura 7 Momento da deduplicação



Fonte: [2]

Por outro lado, podem ser utilizadas as técnicas de deduplicação *in-line* que atuam ou do lado do cliente ou quando a informação está sendo transferida da origem para o servidor de backup, que é o destino da informação.

Se a verificação for *on-client*, o processamento do *hash* ocorre na máquina cliente, que o envia pela rede até o servidor que vai compará-lo com o repositório de assinaturas enviando ou não o arquivo pela rede.

Assim, quando a informação chega ao *appliance*, que é um *hardware* customizado para determinada aplicação com *software* específico embarcado, é feita a análise para ver se aquela informação foi processada anteriormente, se sim apenas é gravada a referência para o dado original, senão o dado é único e deve ser gravado nos discos.

Portanto, é feita a verificação de duplicidade antes da gravação, dessa forma é necessária mais memória RAM, que vai armazenar a tabela de assinatura digitais, por outro lado ocorre menos leitura e gravação nos discos (*overhead*) e espaço em disco não é utilizado em vão. Contudo, a rede por onde trafegam os dados é bastante utilizada e pode gerar gargalos. Mesmo assim, o método *in-line* aumenta a

eficiência global da deduplicação desde que o dado tenha várias origens e toda a duplicidade seja removida de forma instantânea.

Post-process é mais utilizada em aplicações de *backup* e *virtual tape library*, nas quais a redução do tempo para execução do *backup* é desejada.

2.4 Políticas de deduplicação

Após a análise dos métodos de deduplicação é necessário destacar que nem todos os tipos de arquivos são passíveis de serem processados, ou seja, cada fabricante de acordo com o ambiente do usuário e a configuração do equipamento, vai sugerir a política a ser aplicada aos dados daquela organização.

Por isso, são consideradas informações sobre a última vez que o arquivo foi acessado ou modificado, se o arquivo está entre um tamanho máximo e mínimo, o tipo do arquivo. Arquivos que são acessados frequentemente, arquivos contendo imagens, áudio e vídeo, que já são comprimidos não são submetidos a deduplicação.

No entanto, quando utilizamos diversas máquinas virtuais armazenadas no *storage*, ocorre muita repetição do conteúdo de cada uma delas portanto é um tipo de dado que após deduplicado apresenta altas taxas de economia de espaço em disco.

Já quando tratamos de grandes arquivos de bancos de dados, não são obtidos bons resultados após a aplicação das técnicas, uma vez que o conteúdo desses arquivos é constantemente modificado e não pode ser mantido compactado.

2.5 Impacto indireto da deduplicação

Graças as tecnologias de virtualização de servidores e desktops, máquinas físicas que executavam únicos serviços e se multiplicavam, puderam ser consolidadas em poucos *hosts* de grande capacidade, mas de menor tamanho. Por isso, ocorreu uma liberação do espaço físico dos *Datacenters*, ativo de custo elevado de imobilização, e também se evitou que fossem feitos investimentos em *Racks*, *no-breaks*, ativos de rede, cabeamento estruturado, *storage* e discos.

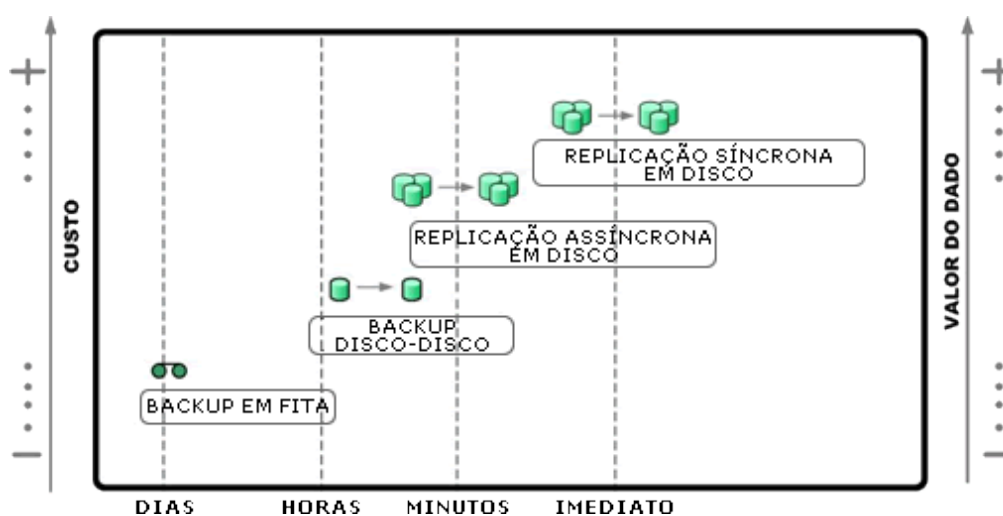
A economia obtida com a virtualização dos servidores e demais equipamentos faz reduzir um dos maiores gastos mensais de um Datacenter, que é o consumo de energia elétrica, tanto dos servidores quanto dos equipamentos de refrigeração.

Seguindo essa tendência, com a adoção das técnicas de deduplicação no *storage* primário, aquele onde ficam armazenados todos os arquivos dos usuários, bancos de dados das aplicações corporativas e até mesmo os arquivos das máquinas virtuais que estão sendo executadas nos *Hosts*, possibilitou uma melhor utilização da capacidade instalada dos *storages* disponíveis, evitando que fossem feitas novas aquisições ou mesmo ampliações dos equipamentos existentes.

Além disso, anteriormente a principal mídia de armazenamento de backups, que podiam ser armazenadas por longo tempo, dispunham de grande capacidade e atendiam as políticas de backup e *restore* das organizações, que eram as unidades de fita magnética, por exemplo DAT, foram substituídas pelo chamado backup D2D ou *disk-to-disk backup*.

A evolução dos discos rígidos, que tiveram sua capacidade aumentada e a vantagem de oferecer acesso randômico, possibilitou que as ferramentas de deduplicação evitassem ao máximo a repetição dos dados e isso tornou mais rápidos os *backups*, que passaram a exigir uma menor janela de execução, e permitiu a realização de *restores* de forma instantânea, evitando assim o investimento nos dispositivos de fita e cartuchos, o que também possibilitou a automatização de todo o processo de backup e transferência de dados.

Figura 8 Técnica de backup x tempo de restauração



Fonte: [19]

Deve ser considerada a forma como os dados são transferidos. Se a transferência for síncrona, isto é, ocorre de forma imediata, não ocorre perda de dados em caso de desastre, mas se for assíncrona, poderá ocorrer perda que será proporcional ao tempo que a mesma levar para ser replicada. Quanto maior pior.

Outro ponto de economia verificado foi com relação aos *links* de internet. Com o crescimento da quantidade e tamanho dos arquivos tornou-se necessário contratar *links* de dados cada vez maiores para que fosse viável a transferência das informações entre o site principal e o site *backup*. Por isso, são aplicadas as mais avançadas técnicas de replicação para se manter um *site recovery* atualizado.

2.6 Recursos viabilizados pela deduplicação

Importante destacar que, além de todos os benefícios alcançados com a aplicação das técnicas de deduplicação, as atividades cotidianas desempenhadas pelos responsáveis de um datacenter podem ser melhoradas, ou seja, pode ocorrer um melhor gerenciamento do tempo do administrador, uma vez que após planejar e implementar as rotinas de *backup* e replicação remota dos dados, o responsável terá mais tempo para implementar sistemas de monitoramento, poderá se capacitar e implantar padrões como NBR ISO/IEC 27002 – Código de Prática para a Gestão de Segurança da Informação, entre outros.

Esta norma trata de temas como Política de Segurança da Informação, Organizando a Segurança da Informação, Gestão de Ativos, Segurança em Recursos Humanos, Segurança Física e do Ambiente.

Dessa forma, verifica-se que após a implementação das ferramentas de *backup* torna-se possível uma otimização das rotinas de *backup*, quando é definido o que deve ser copiado, em que ordem, para qual destino, se deve ser criptografado ou não, para onde será replicado e como, etc. Após isso, pode ser feita uma checagem das rotinas de backup, verificando se o que foi selecionado foi copiado e, o mais importante, devem ser feitas restaurações programadas e permanentes de tudo que foi copiado. Somente a realização dessa checagem garante que, caso seja necessário, os arquivos sejam restaurados de forma rápida e segura.

2.7 RTO e RPO

RTO (*Recovery Time Objective*) compreende o tempo máximo que o negócio pode suportar sem a solução tecnológica e RPO (*Recovery Point Objective*) representa o ponto de recuperação dos dados, ou seja, uma vez recuperada a solução qual a quantidade de dados máxima que poderá ser perdida sem que o negócio seja afetado.

Graças a maior facilidade em armazenar e transferir dados, outro benefício obtido com a utilização das técnicas de deduplicação foi a redução do RTO e o aumento do RPO, como mostra a Figura 9.



Fonte: [4]

Assim, esses dois parâmetros são fundamentais para definição dos requisitos dos sistemas de missão crítica e dependem da política de backup.

3 ESTUDO DE CASO

Para demonstrar a aplicação das técnicas de deduplicação é possível explorar as mais diversas aplicações. Dentre elas, pode-se citar: uma ferramenta de backup, suas *features*, tempos de execução e resultados alcançados, instalação e configuração de um NAS, no caso o FreeNAS-8.3.0, baseado em FreeBSD, que a partir dessa versão incorpora o sistema de arquivos ZFS e implementa técnicas de deduplicação, ou ainda testar ferramentas de sincronização síncrona ou assíncrona de dados para site remoto.

No entanto, foi escolhido demonstrar o funcionamento de serviços como Dropbox e Google Drive, que só são possíveis tecnicamente e viáveis financeiramente graças a adoção em alto grau de recursos de deduplicação.

3.1 Dropbox

Tal serviço encontra-se disponível na Internet de forma gratuita com algumas limitações (<http://www.dropbox.com>), mas atende às necessidades da grande maioria dos usuários de dispositivos móveis, dentre eles os celulares *smartphones* e os *tablets*, que passaram a dispor de equipamentos com grande poder computacional, e que fizeram surgir uma série de aplicativos de todos os tipos, desde tocadores de música, visualizadores de fotos e filmes, jogos e tantos outros, que de uma forma ou de outra precisavam acessar os arquivos do usuário.

Com o surgimento da demanda por serviços de armazenamento de dados como fazer para atendê-la? Devido a grande quantidade de usuários que muitas vezes nem pagam para utilizar tais serviços e o enorme número de arquivos que são armazenados, tornou-se inevitável utilizar toda a tecnologia disponível para tornar realidade essa nova forma de manter e transportar informações.

Para aqueles que estão em casa ou no trabalho tudo parece ser muito simples. Conforme pode ser visto na Figura 10, basta fazer um cadastro com nome, e-mail e senha, instalar um software cliente do serviço se quiser e pronto, está disponível um disco virtual inicialmente de 2GB de tamanho, que pode ser acessado pelo *browser* (Figura 11) ou como se fosse um drive pelos usuários de vários sistemas operacionais ou automaticamente pelos aplicativos instalados nos *smartphones* ou *tablets*.

Figura 10 Tela de cadastro para criar conta no Dropbox





Create an account [\(or sign in\)](#)

First name

Last name

Email

Password

☐ I agree to [Dropbox Terms](#)

[Create account](#)

Fonte: Autor

Figura 11 Área de trabalho do Dropbox

Dropbox

Sharing

Links

Events

Get Started

Space Race

[Get free space!](#)

Dropbox

[Shared folder](#) [Shared folder options](#) [Share link](#) [Download](#) [Delete](#) [Rename](#) [Move](#)

Getting Started.pdf	document pdf	1/16/2012 6:06 PM
Photos	folder	--
Public	folder	--
Shared folder	shared folder	--

Fonte: Autor

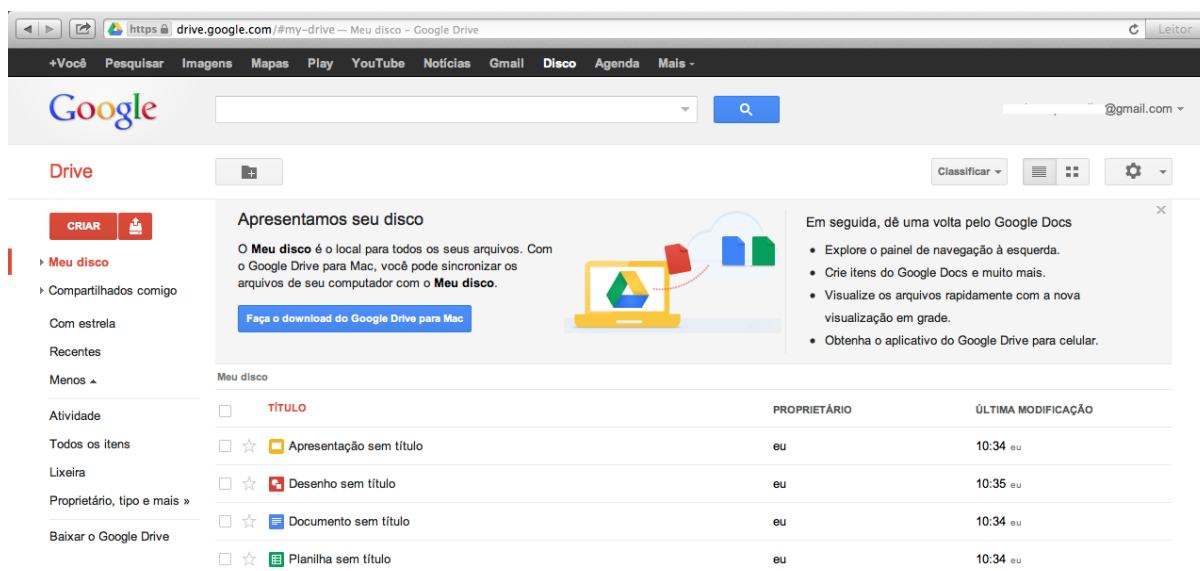
Pelo lado de quem presta o serviço existem muitas preocupações: Como armazenar tanta coisa? Como transferir tudo isso? e a privacidade das informações? e quanto as questões legais? e ainda como tornar esse serviço lucrativo? Todas essas questões foram resolvidas e fizeram com que esse serviço se tornasse indispensável para muitos usuários de dispositivos móveis.

3.2 Google drive

O *Google Drive*, que também fornece serviços de armazenamento funcionando como *drive* virtual, teve incorporado um pacote de aplicativos que

permite criar documentos em formato texto, planilhas de cálculo, apresentações e outros e faz parte de uma série de serviços oferecidos pelo Google, como buscador, *e-mail*, calendário, vídeos, etc.

Figura 12 Área de trabalho do Google Drive



Fonte: Autor

3.3 Metodologia dos testes

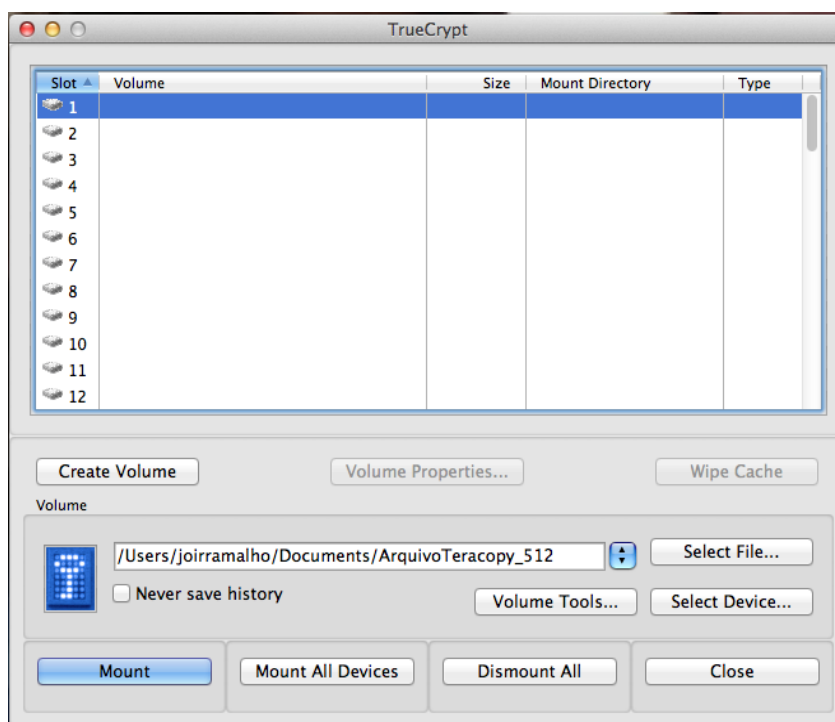
Para que fosse possível demonstrar a utilização das técnicas de deduplicação, aplicadas pelos sites que oferecem serviços de armazenamento, foram enviados arquivos de vários tamanhos, depois feitas alterações no conteúdo dos mesmos, a fim de avaliar a quantidade de bytes transferidos para completar cada tarefa.

Sabe-se que vários fatores devem ser considerados quando se utiliza serviços pela Internet, tais como velocidade da conexão de *upload/download*, qualidade da conexão, tamanho do arquivo, etc.

Sendo assim, os testes pretendem comprovar que as aplicações utilizadas na deduplicação nos processos de *upload* dos arquivos. Os testes foram feitos através da utilização da ferramenta *Wireshark* (<http://www.wireshark.org>). Não foi intenção do teste determinar qual é o melhor serviço, comparando as taxas de bytes que sobem no upload, apenas comprovar que alguma técnica de deduplicação foi utilizada.

Para este estudo de caso foram gerados arquivos de vários tamanhos usando o *TrueCrypt* (<http://www.truecrypt.org>). Este *software* cria e formata arquivos como se fossem um *container* com conteúdo aleatório, que pode ser montado como um *drive* e acessado para armazenar arquivos criptografados com segurança.

Figura 13 Tela do aplicativo Truecrypt



Fonte: Autor

3.4 Sequência dos testes

Foram gerados arquivos pelo *truecrypt* com tamanhos de 1 (um) *Megabyte*, 2 (dois) *Megabytes* e 8 (oito) *Megabytes*, e depois copiados para a pasta local do *Dropbox* e do *Google Drive*. Após a cópia, automaticamente foi feito o upload de cada um, e neste momento foram observados os funcionamentos dos agentes locais (*daemon*) de cada serviço. Isto é, foi analisada a quantidade de bytes que foram carregados para que seja possível comprovar a aplicação das técnicas de deduplicação.

Na primeira fase de testes foram enviados arquivos para o serviço *Dropbox*. Percebe-se, através da figura 14 gerada pelo *software Wireshark*, na linha de informações que foi destacada em vermelho, o endereço IP e porta da máquina de origem, o endereço IP do servidor localizado no *Datacenter* da

Amazon(<http://ec2.amazonaws.com>), o protocolo utilizado - HTTPS e a quantidade de bytes transferidos da estação de origem para o servidor destino, no caso, 1.289.888 bytes, que é a informação mais importante, além de outros dados.

Figura 14 Estatísticas de conversação do arquivo de 1MByte

Ethernet: 5 | Fibre Channel | FDDI | IPv4: 12 | IPv6 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 7 | Token Ring | UDP: 5 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	57512	199.47.219.159	https	11	726	11	726	0	0	36.9166
192.168.1.121	57514	199.47.218.159	https	24	7 939	13	2 612	11	5 327	17.5737
192.168.1.121	54129	74.125.140.125	xmpp-client	127	8 446	64	4 288	63	4 158	39.0322
192.168.1.121	57513	199.47.219.160	https	8	528	8	528	0	0	32.9124
192.168.1.121	57515	23.23.107.90	https	1 972	1 336 902	1 326	1 289 888	646	47 014	15.2133
192.168.1.121	54100	199.47.218.150	http	4	684	2	373	2	311	0.1711
192.168.1.121	57516	199.47.217.174	https	26	8 210	14	2 310	12	5 900	17.2364

☒ Name resolution ☐ Limit to display filter

Help Copy Follow Stream Close

Fonte: Autor

Figura 15 Estatísticas de conversação do arquivo de 2MBytes

Ethernet: 4 | Fibre Channel | FDDI | IPv4: 10 | IPv6: 1 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 7 | Token Ring | UDP: 5 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	57520	199.47.216.174	https	4	252	4	252	0	0	40.2153
192.168.1.121	54100	199.47.218.150	http	4	684	2	373	2	311	0.1759
192.168.1.121	54129	74.125.140.125	xmpp-client	148	9 842	74	4 958	74	4 884	40.0481
192.168.1.121	57522	199.47.218.160	https	4	252	4	252	0	0	44.4805
192.168.1.121	57535	199.47.219.160	https	23	7 861	12	2 534	11	5 327	27.0954
192.168.1.121	57536	23.23.223.138	https	3 501	2 653 544	2 313	2 569 690	1 188	83 854	25.7395
192.168.1.121	57537	199.47.218.159	https	21	7 831	11	2 100	10	5 731	1.2752

☒ Name resolution ☐ Limit to display filter

Help Copy Follow Stream Close

Fonte: Autor

Em seguida foi enviado um arquivo com dois *Megabytes* de tamanho (figura 15). É possível observar a transferência de 2.569.690 bytes.

Os próximos arquivos que foram transferidos foram gerados através da concatenação de um arquivo criptografado com ele mesmo. Para isso, foi utilizado o comando *cat* do *bash* do *terminal* do *MAC OS* para fazer um *append*, ou seja, concatenar um arquivo a outro, em seu final, de forma que o arquivo seja aumentado duas vezes, três vezes e até quatro vezes.

Figura 16 Concatenando arquivos criptografados para upload

```

Mesa — bash — 114x5
MacBook-Pro-de-Joir:Arquivos Teracopy Desduplicação joirramalho$ cp ArquivoTeracopy_1MB ArquivoTeracopy_1MBx2
MacBook-Pro-de-Joir:Arquivos Teracopy Desduplicação joirramalho$ cat ArquivoTeracopy_1MB >> ArquivoTeracopy_1MBx2
MacBook-Pro-de-Joir:Arquivos Teracopy Desduplicação joirramalho$

```

Fonte: Autor

Figura 17 Arquivos preparados para upload

Estudo de caso – Create-Append-Upload				
	Nome	Data de Modific.	Tamanho	Tipo
FAVORITOS Dropbox Todos o... AirDrop Aplicativos Mesa Docume... Transfer... Filmes Música Imagens joirrama... Google...	ArquivoTeracopy_1MB	30/01/2013 18:05	1 MB	Documento
	ArquivoTeracopy_1MB	30/01/2013 17:35	93 KB	Imagem Portable Network Graphics (PNG)
	ArquivoTeracopy_1MB.pcap	30/01/2013 16:51	1,4 MB	Pcap Network Capture
	ArquivoTeracopy_2MB	30/01/2013 18:05	2,1 MB	Documento
	ArquivoTeracopy_2MB	30/01/2013 17:35	98 KB	Imagem Portable Network Graphics (PNG)
	ArquivoTeracopy_2MB.pcap	30/01/2013 16:57	2,8 MB	Pcap Network Capture
	ArquivoTeracopy_2MBx2	30/01/2013 18:06	4,2 MB	Documento
	ArquivoTeracopy_2MBx2	30/01/2013 17:42	155 KB	Imagem Portable Network Graphics (PNG)
	ArquivoTeracopy_2MBx2.pcap	30/01/2013 17:06	5,8 MB	Pcap Network Capture
	ArquivoTeracopy_2MBx2x2	30/01/2013 17:46	8,4 MB	Documento
	ArquivoTeracopy_2MBx2x2	30/01/2013 17:51	98 KB	Imagem Portable Network Graphics (PNG)
	ArquivoTeracopy_2MBx2x2.pcap	30/01/2013 17:48	90 KB	Pcap Network Capture
	ArquivoTeracopy_2MBx2x2x2	Hoje 16:04	16,8 MB	Documento
	ArquivoTeracopy_2MBx2x2x2	30/01/2013 17:59	87 KB	Imagem Portable Network Graphics (PNG)
	ArquivoTeracopy_2MBx2x2x2.pcap	30/01/2013 17:57	21 KB	Pcap Network Capture

Fonte: Autor

Esse procedimento pretende demonstrar que quando o arquivo é analisado internamente, o mesmo é dividido em pedaços, sendo gerados *hashs* para comparação, e não só pela sua data e hora de criação. Sendo assim, pode-se armazenar o arquivo no *storage* principal, e enviar, se for o caso do Dropbox ou Google Drive, apenas partes do arquivo que difere daquilo que já fora armazenado anteriormente.

As figuras 18, 19 e 20 mostram o processo de aumentar gradativamente o arquivo para análise dos processos de *upload* e atualização no *storage* do serviço de armazenamento.

Figura 18 Estatísticas de conversação do arquivo de 4Mbytes(concatenado uma vez)

Conversations: ArquivoTeracopy_2MBx2.pcap

Ethernet: 9 | Fibre Channel | FDDI | IPv4: 22 | IPv6: 3 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 19 | Token Ring | UDP: 11 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	57626	77.234.42.55	http	31	13 076	14	2 286	17	10 790	87.0667
192.168.1.121	57629	199.47.219.159	https	23	7 893	12	2 566	11	5 327	54.6455
192.168.1.121	57630	184.73.167.31	https	7 748	5 348 014	5 390	5 188 008	2 358	160 006	53.2614
192.168.1.121	54129	74.125.140.125	xmpp-client	222	14 763	111	7 437	111	7 326	76.5825
192.168.1.101	62078	192.168.1.121	57631	45	8 441	22	3 657	23	4 784	0.1404
192.168.1.121	57632	192.168.1.101	50380	29	4 406	15	2 570	14	1 836	11.6994
192.168.1.121	54100	199.47.218.150	http	8	1 390	4	746	4	644	54.1032
192.168.1.121	57621	74.125.140.125	xmpp-client	2	169	1	66	1	103	0.0002
192.168.1.121	57575	74.125.234.32	https	8	624	4	264	4	360	10.0381
192.168.1.121	57574	74.125.234.32	https	8	624	4	264	4	360	10.0260
192.168.1.121	57619	74.125.234.51	https	8	630	4	264	4	366	0.0808
192.168.1.121	57614	74.125.234.35	https	8	630	4	264	4	366	0.0835
192.168.1.121	57622	74.125.234.47	https	9	956	5	521	4	435	0.1053
192.168.1.121	57624	74.125.234.45	http	4	258	2	132	2	126	6.5357
192.168.1.121	57625	74.125.234.57	http	4	258	2	132	2	126	6.4350
192.168.1.101	62078	192.168.1.121	57633	45	8 441	22	3 657	23	4 784	0.1437
192.168.1.121	57634	192.168.1.101	50381	26	4 195	14	2 528	12	1 667	10.2562
192.168.1.121	57635	199.47.218.160	https	26	8 226	14	2 326	12	5 900	17.2566
192.168.1.121	57636	199.47.217.174	https	53	36 510	24	2 576	29	33 934	17.2178

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

Figura 19 Estatísticas de conversação do arquivo de 8Mbytes (concatenado duas vezes)

Conversations: ArquivoTeracopy_2MBx2x2.pcap

Ethernet: 6 | Fibre Channel | FDDI | IPv4: 12 | IPv6: 3 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 8 | Token Ring | UDP: 11 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	57747	199.47.219.159	https	25	8 148	13	2 248	12	5 900	17.9180
192.168.1.121	54129	74.125.140.125	xmpp-client	148	9 842	74	4 958	74	4 884	40.2096
192.168.1.121	54100	199.47.218.150	http	8	1 390	4	746	4	644	1.5819
192.168.1.121	57748	199.47.217.173	https	52	36 480	23	2 498	29	33 982	15.8765
192.168.1.121	57738	23.37.78.13	http	2	132	1	66	1	66	0.0003
192.168.1.121	57528	74.125.137.108	imaps	70	9 121	38	4 456	32	4 665	1.2266
192.168.1.121	57531	74.125.137.108	imaps	3	276	2	168	1	108	0.1635
192.168.1.121	57740	23.37.78.13	http	2	132	1	66	1	66	0.0001

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

Figura 20 Estatísticas de conversação do arquivo de 16 Mbytes (concatenado três vezes)

Conversations: ArquivoTeracopy_2MBx2x2x2.pcap

Ethernet: 5 | Fibre Channel | FDDI | IPv4: 8 | IPv6: 3 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 4 | Token Ring | UDP: 5 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	57769	192.168.1.101	50393	5	419	3	186	2	233	1.9857
192.168.1.121	57770	199.47.218.160	https	21	7 847	11	2 116	10	5 731	1.3008
192.168.1.121	54129	74.125.140.125	xmpp-client	74	4 921	37	2 479	37	2 442	5.1658
192.168.1.121	57767	77.234.43.56	http	4	755	2	359	2	396	0.2009

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

Analisando este processo, observou-se que a partir do momento em que o arquivo concatenado atinge quatro Megabytes, (figura 18) verifica-se que a quantidade de bytes transferidos não é proporcional ao tamanho do arquivo enviado, ou seja, foram aplicadas as técnicas de deduplicação. Caso o arquivo enviado seja menor que 4 (quatro) Megabytes ele não sofre processamento por não valer a pena a economia de tráfego e/ou espaço obtidos.

Em seguida, foram enviados os mesmos arquivos para o serviço *Google Drive* que os mantém armazenados em um dos *Datacenters* da *Google* (www.drive.google.com). Percebe-se, através da análise das figuras 21 a 25, que a quantidade de Bytes enviados pelo agente local do serviço do *Google* é proporcional aos tamanhos dos arquivos.

Figura 21 Estatísticas de conversação do arquivo de 1MByte

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59679	50.16.210.253	https	2	132	1	66	1	66	0.1106
192.168.1.121	59680	199.47.218.159	https	8	528	8	528	0	0	33.0925
192.168.1.121	59678	199.47.218.160	https	8	528	8	528	0	0	27.6388
192.168.1.121	59681	74.125.140.116	https	1 256	1 168 710	777	1 127 440	479	41 270	10.3091
192.168.1.121	59682	74.125.234.36	https	22	7 548	12	1 656	10	5 892	1.2907
192.168.1.121	58052	74.125.140.125	xmpp-client	80	6 252	40	3 115	40	3 137	8.4860
192.168.1.121	58039	199.47.218.150	http	4	684	2	373	2	311	0.1601
192.168.1.121	59665	74.125.140.125	https	2	617	1	66	1	551	0.0001
192.168.1.121	59683	74.125.234.35	https	24	7 134	13	1 739	11	5 395	0.7968

Fonte: Autor

Figura 22 Estatísticas de conversação do arquivo de 2MBytes

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59695	74.125.234.36	https	7	501	4	291	3	210	0.0740
192.168.1.121	58039	199.47.218.150	http	4	684	2	373	2	311	0.1600
192.168.1.121	59705	74.125.140.116	https	2 393	2 299 974	1 535	2 233 394	858	66 580	18.0805
192.168.1.121	58052	74.125.140.125	xmpp-client	80	6 252	40	3 115	40	3 137	11.7617
192.168.1.121	59706	74.125.234.36	https	28	8 018	15	1 854	13	6 164	2.3899
192.168.1.121	59665	74.125.140.125	https	2	617	1	66	1	551	0.0002
192.168.1.121	59707	74.125.234.36	https	28	7 453	15	1 871	13	5 582	0.6459

Fonte: Autor

Figura 23 Estatísticas de conversação do arquivo de 4Mbytes(concatenado uma vez)

Conversations: ArquivoTeracopy_2MBx2.pcap

Ethernet: 3 | Fibre Channel | FDDI | IPv4: 8 | IPv6 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 5 | Token Ring | UDP: 11 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59743	74.125.140.117	https	4 640	4 545 542	3 004	4 427 068	1 636	118 474	34.3381
192.168.1.121	58052	74.125.140.125	xmpp-client	80	6 252	40	3 115	40	3 137	28.9755
192.168.1.121	58039	199.47.218.150	http	4	684	2	373	2	311	0.6066
192.168.1.121	59719	173.194.73.125	xmpp-client	2	617	1	66	1	551	0.0002
192.168.1.121	59744	74.125.234.34	https	27	7 377	15	1 883	12	5 494	1.7888

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

Figura 24 Estatísticas de conversação do arquivo de 8Mbytes(concatenado duas vezes)

Conversations: ArquivoTeracopy_2MBx2x2.pcap

Ethernet: 6 | Fibre Channel | FDDI | IPv4: 13 | IPv6: 1 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 8 | Token Ring | UDP: 7 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	58052	74.125.140.125	xmpp-client	154	11 197	77	5 593	77	5 604	70.5527
192.168.1.121	59773	74.125.140.117	https	9 079	9 019 938	5 945	8 803 238	3 134	216 700	63.1049
192.168.1.121	58039	199.47.218.150	http	8	1 368	4	746	4	622	56.2588
192.168.1.121	59751	74.125.234.35	https	8	630	4	264	4	366	0.0851
192.168.1.121	59753	74.125.234.38	http	4	264	2	132	2	132	0.0951
192.168.1.121	59719	173.194.73.125	xmpp-client	2	617	1	66	1	551	0.0002
192.168.1.121	59774	74.125.234.46	https	22	6 986	12	1 673	10	5 313	0.6905
192.168.1.121	59754	74.125.234.47	http	2	132	1	66	1	66	0.0003

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

Figura 25 Estatísticas de conversação do arquivo de 16 Mbytes (concatenado três vezes)

Conversations: ArquivoTeracopy_2MBx2x2x2.pcap

Ethernet: 22 | Fibre Channel | FDDI | IPv4: 32 | IPv6: 10 | IPX | JXTA | NCP | RSVP | SCTP | TCP: 23 | Token Ring | UDP: 49 | USB | WLAN

TCP Conversations

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.108	53842	74.125.134.108	imaps	12	1 183	7	653	5	530	2.3135
192.168.1.108	53849	74.125.134.108	imaps	6	528	3	274	3	254	1.4667
192.168.1.108	53844	74.125.134.108	imaps	188	70 936	90	6 210	98	64 726	10.1074
192.168.1.108	53845	74.125.134.108	imaps	76	13 959	41	6 371	35	7 588	11.0574
192.168.1.108	54586	74.125.134.117	https	18 148	17 871 142	11 836	17 511 280	6 312	359 862	142.2398
192.168.1.108	53824	74.125.134.125	xmpp-client	2	750	1	684	1	66	0.3851
192.168.1.108	53821	74.125.134.125	xmpp-client	279	19 558	140	9 377	139	10 181	151.3856
192.168.1.108	54174	74.125.134.125	xmpp-client	16	2 648	8	1 150	8	1 498	137.1978
192.168.1.108	53827	74.125.134.125	xmpp-client	4	1 234	2	132	2	1 102	0.0240
192.168.1.108	53826	74.125.134.189	https	40	9 700	20	7 880	20	1 820	163.6754
192.168.1.108	54590	74.125.234.32	https	24	7 007	13	1 739	11	5 268	1.7334
192.168.1.108	54591	74.125.234.32	https	19	6 336	10	1 511	9	4 825	0.5293
192.168.1.108	54585	74.125.234.33	https	20	7 752	10	2 955	10	4 797	0.3399
192.168.1.108	54587	74.125.234.40	http	7	1 701	4	764	3	937	1.2420
192.168.1.108	54588	74.125.234.40	http	6	416	4	276	2	140	10.6189
192.168.1.108	54592	74.125.234.44	https	13	4 032	8	1 564	5	2 468	1.5824
192.168.1.108	54589	74.125.234.47	http	8	3 214	4	574	4	2 640	1.2390
192.168.1.108	54563	74.125.234.47	https	8	890	5	522	3	368	0.0665

☒ Name resolution ☐ Limit to display filter

[Help](#) [Copy](#) [Follow Stream](#) [Close](#)

Fonte: Autor

A fim de proceder a segunda fase de testes foram utilizados os arquivos de 8 (oito) Megabytes que foram enviados para o serviço *Dropbox*. Percebe-se através das figuras 26 e 27 geradas pelo *Wireshark*, na linha de informações que foi destacada em vermelho, a quantidade de *bytes* transferidos da estação de origem para o servidor destino, no caso, 10.229.636 bytes, e após a montagem do volume criptografado, alteração do conteúdo e desmontagem do mesmo, o serviço *Dropbox* envia uma quantidade mínima de bytes.

Figura 26 Arquivo de 8 Megabytes sendo enviado para Dropbox

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59384	199.47.216.172	https	24	7 327	13	1 946	11	5 381	67.8598
192.168.1.121	59385	23.23.111.213	https	11 443	10 517 844	7 351	10 229 636	4 092	288 208	185.5373
192.168.1.121	58052	74.125.140.125	xmpp-client	427	28 432	214	14 337	213	14 095	183.2049
192.168.1.121	58039	199.47.218.150	http	16	2 758	8	1 492	8	1 266	123.8102
192.168.1.121	59386	199.47.216.173	https	32	8 670	20	2 770	12	5 900	112.6769
192.168.1.121	59387	108.160.161.177	https	59	36 954	30	2 972	29	33 982	110.9889
192.168.1.121	59388	108.160.161.178	https	28	6 967	18	1 812	10	5 155	92.9053
192.168.1.121	59381	74.125.234.47	https	8	630	4	264	4	366	0.0642
192.168.1.121	59383	74.125.234.36	https	8	630	4	264	4	366	0.0768

Fonte: Autor

Figura 27 Arquivo de 8 Megabytes sendo enviado para Dropbox após edição

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59631	199.47.217.174	https	3	186	3	186	0	0	27.2758
192.168.1.121	59552	200.157.208.233	http	4	264	2	132	2	132	2.9515
192.168.1.121	58052	74.125.140.125	xmpp-client	74	4 921	37	2 479	37	2 442	5.0671
192.168.1.121	59509	74.125.234.38	https	8	630	4	264	4	366	0.0747
192.168.1.121	59553	200.157.208.232	http	3	198	2	132	1	66	0.0937
192.168.1.121	59635	199.47.216.172	https	26	9 660	14	3 296	12	6 364	9.7080
192.168.1.121	58039	199.47.218.150	http	8	1 390	4	746	4	644	10.1975
192.168.1.121	59636	23.23.221.213	https	56	38 335	30	32 241	26	6 094	5.7700
192.168.1.121	58083	74.125.140.125	xmpp-client	2	750	1	684	1	66	0.1356
192.168.1.121	59637	199.47.216.173	https	48	36 191	21	2 378	27	33 813	2.0532

Fonte: Autor

Enquanto isso, o serviço *Google Drive* transmite o arquivo inteiro cada vez que o mesmo é desmontado, como pode ser observado através das figuras 28 e 29.

Figura 28 Arquivo de 8 Megabytes sendo enviado para Google Drive

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59719	173.194.73.125	xmpp-client	6	1 536	3	816	3	720	74.7094
192.168.1.121	59817	74.125.140.117	https	9 107	9 022 875	5 946	8 804 718	3 161	218 157	64.0618
192.168.1.121	58052	74.125.140.125	xmpp-client	152	11 028	76	5 527	76	5 501	56.4390
192.168.1.121	58039	199.47.218.150	http	8	1 368	4	746	4	622	56.1549
192.168.1.121	59815	50.17.243.189	https	6	433	3	198	3	235	7.5485
192.168.1.121	59816	199.47.218.159	https	2	132	1	66	1	66	0.1506
192.168.1.121	59814	199.47.217.174	https	2	126	1	66	1	60	0.1570
192.168.1.101	62078	192.168.1.121	59818	45	8 441	22	3 657	23	4 784	0.1194
192.168.1.121	59819	192.168.1.101	49292	29	4 406	15	2 570	14	1 836	10.1416
192.168.1.121	59820	199.47.217.174	https	21	7 101	11	1 882	10	5 219	17.6475
192.168.1.121	59821	74.125.234.40	https	21	6 129	12	1 687	9	4 442	2.1818

Fonte: Autor

Figura 29 Arquivo de 8 Megabytes sendo enviado para Google Drive após edição

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A→B	Bytes A→B	Packets A←B	Bytes A←B	Duration
192.168.1.121	59820	199.47.217.174	https	2	132	2	132	0	0	48.7131
192.168.1.121	58052	74.125.140.125	xmpp-client	152	11 028	76	5 527	76	5 501	70.8993
192.168.1.121	59828	74.125.234.38	https	23	6 766	13	1 733	10	5 033	1.6554
192.168.1.121	59829	74.125.140.117	https	9 075	9 019 607	5 945	8 804 590	3 130	215 017	64.9864
192.168.1.121	58039	199.47.218.150	http	8	1 368	4	746	4	622	55.7919
192.168.1.121	59719	173.194.73.125	xmpp-client	4	786	2	132	2	654	6.9186
192.168.1.121	59830	74.125.234.36	https	25	7 233	14	1 817	11	5 416	3.6635
192.168.1.101	62078	192.168.1.121	59831	45	8 441	22	3 657	23	4 784	0.1544
192.168.1.121	59832	192.168.1.101	49294	16	3 120	8	1 818	8	1 302	0.0612

Fonte: Autor

Ficou demonstrado que o serviço *Dropbox* é mais eficiente e envia para seus servidores, utilizando menos o *link* de dados, apenas *hashs* gerados localmente por seu agente de sincronização, que são comparados com os *metadados* armazenados em seus servidores, e portanto, evita-se que sejam enviados novamente arquivos, ou mesmo partes deles, que já se encontram armazenados remotamente.

Já o Serviço Google Drive, aparenta não proceder análise local do conteúdo dos arquivos a serem enviados, não sendo capaz de evitar a transferência de todo o arquivo novamente, consumindo banda e recursos do computador.

Portanto, fica comprovado o uso das técnicas de deduplicação, sua importância e necessidade de serem aplicadas por todos os motivos citados anteriormente.

4 CONCLUSÃO

Ao término deste trabalho, que teve como objetivo analisar as técnicas de deduplicação visando eliminar a redundância dos dados, comparando-as com as formas convencionais de armazenamento em *storages* primários, bem como os procedimentos de backup e arquivamento de dados, ficou demonstrado que tornou-se essencial a aplicação dessas técnicas, e o objetivo foi alcançado.

Observou-se a utilização da compressão de arquivos, a possibilidade de criptografar os dados armazenados e a viabilidade de transferi-los para locais seguros em mídias de armazenamento ou mesmo através de sítios de Internet.

Analisou-se como funciona o método e o momento quando devem ser aplicadas as técnicas de deduplicação, de acordo com o volume de dados, equipamentos disponíveis e a política de *Backup* e recuperação de desastres da organização.

Além disso, foi possível demonstrar uma série de benefícios indiretos, como por exemplo: economia de energia elétrica, a utilização racional dos recursos de hardware disponíveis e na otimização das rotinas administrativas do *Datacenter*, permitindo aos responsáveis pela TI planejar suas atividades, adequando-as à legislação vigente, adotando boas práticas do setor.

Foi visto também que a adoção de tais técnicas permite a redução do tempo de recuperação dos serviços após ocorrer um desastre ou mesmo a perda mínima de informação caso seja registrado algum incidente de grandes proporções.

Assim, foi possível concluir o trabalho com apoio da documentação oficial de alguns fabricantes, diversos sites, a *Wikipédia*, fóruns da internet, além dos vários conhecimentos adquiridos durante o Curso de Especialização em Redes de Computadores do Centro Universitário do Rio Grande do Norte – UNI-RN.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] How Does DropboxStore Data? Disponível em:<<http://blog.fosketts.net/2011/07/11/dropbox-data-format-deduplication>>.Acessoem: 01 set. de 2012

[2] Effective Technology of Disk Space Saving.Disponível em:
<<http://www.starwindsoftware.com/data-deduplication>>.Acesso em 03 set de 2012.

[3] Corporate PC backup: Anatomyof a Data DeduplicationSystem.Disponívelem :
<<http://www.copiun.com/blog/bid/45383/Corporate-PC-backup-Anatomy-of-a-Data-Deduplication-System>>.Acesso em 05 de set de 2012

[4] Disasterrecoveryplan. Disponível em:
<http://en.wikipedia.org/wiki/Disaster_recovery_plan>.Acesso em 05 de set de 2012

[5] Disasterrecovery.Disponível em:
<http://en.wikipedia.org/wiki/Data_deduplication>.Acesso em 05 de set de 2012

[6] DropboxAccusedOfMisleadingCustomersOn Security. Disponível em:
<<http://www.informationweek.com/storage/security/dropbox-accused-of-misleading-customers/229500683>>.Acesso em 08 de set de 2012

[7] HowtoChoose A DeduplicationSolution.Disponível em:
<http://www.wwpi.com/index.php?option=com_content&view=article&catid=99:cover-story&id=8477:how-to-choose-a-deduplication-solution&Itemid=2701018>.
Acesso em 08 de set de 2012

[8] EMC Glossary. Disponível em:
<<http://www.emc.com/corporate/glossary/index.htm>>.Acesso em 18 de set de 2012

[9] CERT.br. Cartilha de segurança para internet, versão 4.0. São Paulo: Comitê Gestor da Internet no Brasil, 2012. Disponível em:

<<http://cartilha.cert.br/livro/cartilha-seguranca-internet.pdf>>. Acesso em: 02 de novembro de 2012.

[10] Cópia de segurança. Disponível em:

<http://pt.wikipedia.org/wiki/C%C3%B3pia_de_seguran%C3%A7a>. Acesso em: 02 de novembro de 2012.

[11] Disaster recovery. Disponível em:

<http://en.wikipedia.org/wiki/Disaster_recovery>. Acesso em: 02 de novembro de 2012.

[12] Hash

Disponível em: <<http://pt.wikipedia.org/wiki/Hash>>. Acesso em: 02 de novembro de 2012.

[13] Compressão de dados. Disponível em:

<http://pt.wikipedia.org/wiki/Compress%C3%A3o_de_dados>. Acesso em: 02 de novembro de 2012.

[14] Backup disk-to-disk. Disponível em: <<http://en.wikipedia.org/wiki/Disk-to-disk>>. Acesso em: 02 de novembro de 2012.

[15] Site backup. Disponível em: <http://en.wikipedia.org/wiki/Backup_site>.

Acesso em: 02 de novembro de 2012.

[16] Off-site data protection. Disponível em: <http://en.wikipedia.org/wiki/Off-site_Data_Protection>. Acesso em: 02 de novembro de 2012.

[17] RTO – Recovery time objective. Disponível em:

<http://en.wikipedia.org/wiki/Recovery_time_objective>. Acesso em: 12 de novembro de 2012

[18] RPO – Recovery point objective. Disponível em:
<http://en.wikipedia.org/wiki/Recovery_point_objective>. Acesso em: 13 de novembro de 2012

[19] Missão Crítica em TI: Disponibilidade e Desastres. Disponível em:
<http://www.teleco.com.br/tutoriais/tutorialmissaoti/pagina_3.asp>. Acesso em: 13 de novembro de 2012

[20] Wireshark Disponível em: <<http://www.wireshark.org>>. Acesso em: 10 de janeiro de 2013

[21] Truecrypt. Disponível em: <<http://www.truecrypt.org>>. Acesso em: 19 de novembro de 2012

[22] FeeNAS. Disponível em: <<http://www.freenas.org>>. Acesso em: 13 de novembro de 2012

[23] Cinco maneiras de reduzir o storage. Disponível em:
<<http://computerworld.uol.com.br/tecnologia/2010/12/01/cinco-maneiras-de-reduzir-o-storage/>>. Acesso em: 20 de novembro de 2012

[24] md5 hashgenerator. Disponível em:
<<http://www.miraclesalad.com/webtools/md5.php>>. Acesso em: 01 de novembro de 2012

[25] Asynchronous WAN-replication Disponível em:
<<http://www.starwindsoftware.com/native-san-for-hyper-v-features>>. Acesso em: 10 de janeiro de 2013