



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

LUCAS CANDIDO SIMEÃO

**USO DA FERRAMENTA OWNCLOUD COM A TECNOLOGIA
RASPBERRY PI**

**Assis/SP
2018**



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

LUCAS CANDIDO SIMEÃO

**USO DA FERRAMENTA OWNCLOUD COM A TECNOLOGIA
RASPBERRY PI**

Projeto de pesquisa apresentado ao curso de Bacharelado em Ciências da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando (a): Lucas Candido Simeão
Orientador (a): Fábio Eder Cardoso

**Assis/SP
2018**

FICHA CATALOGRÁFICA

S589u SIMEÃO, Lucas Candido
Uso da ferramenta owncloud com a tecnologia raspberry pi /
Lucas Candido Simeão. – Assis, 2018.

47.

Trabalho de conclusão do curso (Ciência da Computação). –
Fundação Educacional do Município de Assis-FEMA

Orientador: Ms. Fábio Eder Cardoso

1.Software 2.Owncloud 3.Raspberry PI

CDD 005.12

USO DA FERRAMENTA OWNCLOUD COM A TECNOLOGIA RASPBERRY PI

LUCAS CANDIDO SIMEÃO

Projeto de pesquisa apresentado ao curso de Bacharelado em Ciências da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientador:

FÁBIO EDER CARDOSO

Examinador:

LUIZ CARLOS BEGOSSO

DEDICATÓRIA

Dedico esse trabalho a meus pais e irmãs pelo apoio, e incentivo em todas as etapas de realização deste projeto.

AGRADECIMENTOS

Primeiramente agradeço a Deus, meu Mestre. Aos meus familiares por todo o apoio ao longo do curso, ao professor Fábio Eder pela sabedoria com que conduziu a orientação deste trabalho e aos professores pelos conhecimentos transmitidos.

“Conheço muitos que não puderam, quando deviam,
porque não quiseram quando podiam. ”

François Rabelais
(1494-1553)

RESUMO

A *Cloud Computing* é uma tecnologia nova, mas o conceito surgiu em 1961 quando já se falava sobre uma rede global de computadores (LICKLIDER, 1962). O objetivo deste projeto é de apresentar uma forma mais segura de se armazenar os dados pessoais e o método de criação de uma nuvem privada com o uso da tecnologia Owncloud e a placa Raspberry Pi, de forma a tornar mais prático e rápido o acesso ao mesmo. Com isso o usuário não ficará limitado a apenas um dispositivo e terá disponibilidade do mesmo em qualquer lugar, que tenha acesso à Internet, e uma maior segurança dos dados podendo fazer *backup*.

Palavras-chave: Owncloud; computação em nuvem; backup; segurança; Raspberry Pi.

ABSTRACT

Cloud Computing is a new technology, but the concept arose in 1961 when we were talking about a global network of computers (LICKLIDER, 1962). However, the objective of this project is to present a safer way to store your personal data and the method of building a private cloud with the use of technology Owncloud and plate Raspberry Pi, in order to become more convenient and fast access to the same. With this, the user will not be limited to only one device and will have availability of the same anywhere that has Internet access and a greater security of the data being able to back up.

Keywords: Owncloud; Cloud Computing; backup; security; Raspberry Pi.

LISTA DE ILUSTRAÇÕES

| | |
|--|----|
| Figura 1 - Representação da Estrutura da Nuvem | 18 |
| Figura 2 - Placa Raspberry Pi 3 Model B..... | 23 |
| Figura 3 - Tela de exibição do arquivo de configuração de interfaces | 24 |
| Figura 4 – Tela de exibição do conteúdo do arquivo de configuração de interfaces | 25 |
| Figura 5 – Tela de exibição do comando de reinicialização do serviço de rede | 25 |
| Figura 6 – Tela de exibição do comando de atualização do repositório e instalação das atualizações dos programas | 25 |
| Figura 7 – Tela de exibição do comando para configuração do Raspberry Pi..... | 26 |
| Figura 8 – Tela de exibição para selecionar a opção de interfaces | 26 |
| Figura 9 – Tela de exibição para escolha da interface SSH | 27 |
| Figura 10 – Tela de exibição para confirmar a ativação do serviço SSH..... | 27 |
| Figura 11 – Tela de exibição do comando para atribuir o usuário www-data ao grupo www-data..... | 28 |
| Figura 12 – Tela de exibição dos comandos de instalação do servidor Nginx e o PHP7 com todos os pacotes necessários..... | 28 |
| Figura 13 – Tela de exibição do comando para limpar as configurações do servidor | 28 |
| Figura 14 – Tela de exibição do comando para acessar o arquivo de configuração do servidor Nginx..... | 29 |
| Figura 15 – Tela de exibição da configuração do servidor Nginx para o funcionamento do Owncloud..... | 30 |
| Figura 16 – Tela de exibição do comando para acessar as configurações do PHP7 | 31 |
| Figura 17 – Tela de exibição para alteração do comando <code>post_max_size</code> | 31 |
| Figura 18 – Tela de exibição para alteração do comando <code>upload_max_filesize</code> | 31 |
| Figura 19 – Tela de exibição do comando para acessar o arquivo <code>conf</code> | 31 |
| Figura 20 – Tela de exibição para atualizar o comando <code>listen</code> | 31 |
| Figura 21 – Tela de exibição do comando para acessar o arquivo de swap do Raspberry Pi 3..... | 32 |
| Figura 22 – Tela de exibição para alteração do comando <code>CONF_SWAPSIZE</code> | 32 |
| Figura 23 – Tela de exibição do comando para reinicialização do Raspberry Pi 3..... | 32 |
| Figura 24 – Tela de exibição dos comandos para criação de um certificado SSL..... | 32 |
| Figura 25 – Tela de exibição do comando para alterar a permissão no arquivo <code>cert.pem</code> | 33 |
| Figura 26 – Tela de exibição do comando para alterar a permissão no arquivo <code>cert.key</code> | 33 |

| | |
|--|----|
| Figura 27 – Tela de exibição do comando de criação de diretório para o Owncloud..... | 33 |
| Figura 28 – Tela de exibição do comando de <i>download</i> da ferramenta Owncloud..... | 34 |
| Figura 29 – Tela de exibição do comando de descompactar arquivos compactados..... | 34 |
| Figura 30 – Tela de exibição do comando de alteração de diretório | 34 |
| Figura 31 – Tela de exibição do comando para alteração das permissões do diretório | 34 |
| Figura 32 – Tela de exibição do comando de remoção do arquivo Owncloud..... | 34 |
| Figura 33 – Tela de exibição do comando para acessar o diretório Owncloud..... | 35 |
| Figura 34 – Tela de exibição para acessar o arquivo <i>htaccess</i> | 35 |
| Figura 35 – Tela de exibição dos comandos modificados do arquivo <i>htaccess</i> | 35 |
| Figura 36 – Tela de exibição do comando para acessar o arquivo <i>user.ini</i> | 35 |
| Figura 37 – Tela de exibição dos comandos modificados do arquivo <i>user.ini</i> | 36 |
| Figura 38 – Tela de exibição do comando de instalação do pacote NTFS | 36 |
| Figura 39 – Tela de exibição do comando para criação do diretório ownclouddrive..... | 36 |
| Figura 40 – Tela de exibição dos comandos para descobrir o <i>UID</i> e o <i>GID</i> | 37 |
| Figura 41 – Tela de exibição do comando para descobrir o <i>UUID</i> da unidade externa..... | 37 |
| Figura 42 – Tela de exibição para obter o <i>UUID</i> da unidade externa conectada no Pi | 37 |
| Figura 43 – Tela de exibição do comando para acessar o arquivo <i>fstab</i> | 38 |
| Figura 44 – Tela de exibição do arquivo para adicionar a unidade externa com suas permissões | 38 |
| Figura 45 – Tela de exibição da página de configurações finais da ferramenta Owncloud..... | 39 |
| Figura 46 – Tela de exibição da página de bem-vindo da interface gráfica do Owncloud | 39 |
| Figura 47 – Tela de exibição do ambiente gráfico da tela de administrador do Owncloud | 40 |
| Figura 48 – Tela de exibição de configuração do aplicativo de sincronização 1..... | 41 |
| Figura 49 – Tela de exibição de configuração do aplicativo de sincronização 2..... | 41 |
| Figura 50 - Tela de exibição de configuração do aplicativo de sincronização 3 | 42 |
| Figura 51 - Tela de exibição de configuração do aplicativo de sincronização 4 | 43 |

LISTA DE ABREVIATURAS E SIGLAS

| | |
|-------|--------------------------------------|
| DDR | Double Data Rate |
| GID | Group IDentification |
| HDMI | High-Definition Multimedia Interface |
| IaaS | <i>Infrastructure-as-a-Service</i> |
| IP | Internet Protocol |
| NTFS | New Technology File System |
| PaaS | <i>Platform-as-a-Service</i> |
| PHP | Personal Home Page |
| RAID | Redundant Array of Independent Disks |
| RJ-45 | Registered Jack |
| SaaS | <i>Software-as-a-Service</i> |
| SSH | Secure Shell |
| SSL | Secure Sockets Layer |
| TI | Tecnologia da Informação |
| UID | User IDentifier |
| USB | Universal Serial Bus |
| UUID | Universally Unique Identifier |

SUMÁRIO

| | |
|--|-----------|
| 1. INTRODUÇÃO | 14 |
| 1.1. OBJETIVOS | 15 |
| 1.2. JUSTIFICATIVAS | 15 |
| 1.3. MOTIVAÇÕES | 15 |
| 1.4. METODOLOGIA | 16 |
| 2. REVISÃO BIBLIOGRÁFICA | 17 |
| 2.1. COMPUTAÇÃO EM NUVEM | 17 |
| 2.2. DEFINIÇÃO | 17 |
| 2.3. MODELOS DE SERVIÇOS | 18 |
| 2.3.1. SaaS | 18 |
| 2.3.2. Paas | 19 |
| 2.3.3. IaaS | 19 |
| 2.4. MODELOS DE IMPLANTAÇÃO | 19 |
| 2.4.1. Nuvem Pública | 20 |
| 2.4.2. Nuvem Privada | 20 |
| 2.4.3. Nuvem Comunitária | 20 |
| 2.4.4. Nuvem híbrida | 21 |
| 2.5. SEGURANÇA | 21 |
| 2.6. OWNCLOUD | 21 |
| 2.7. RASPBERRY PI | 22 |
| 2.7.1. Sistemas Operacionais | 23 |
| 3. INSTALAÇÃO E CONFIGURAÇÃO DA FERRAMENTA OWNCLOUD | 24 |
| 3.1. CONFIGURAÇÃO DE REDE | 24 |
| 3.2. SERVIÇO DE ACESSO REMOTO SSH | 25 |
| 3.3. SERVIDOR HTTP NGINX E PHP7 | 28 |
| 3.4. ATRIBUIÇÃO DE SEGURANÇA | 32 |
| 3.5. INSTALAÇÃO DA FERRAMENTA OWNCLOUD | 33 |
| 3.6. CONFIGURAÇÃO DE UMA UNIDADE DE ARMAZENAMENTO | 36 |
| 3.7. CONFIGURAÇÃO DA INTERFACE GRÁFICA | 38 |
| 3.8. INSTALAÇÃO DO APLICATIVO DE SINCRONIZAÇÃO | 40 |

| | |
|----------------------------|-----------|
| 4. CONCLUSÃO | 44 |
| 5. REFERÊNCIAS..... | 45 |

1. INTRODUÇÃO

O homem vem remodelando a realidade global, a partir da Revolução Industrial com seu início no século XVII, vários aspectos foram modificados. Recentemente, encontra-se em sua terceira etapa, que se destaca a globalização, conseqüentemente as máquinas fortificaram os braços e pernas, à medida que a sociedade da informação e do conhecimento intensificaram nossa capacidade de criação (TAURION, 2009)

Com os avanços tecnológicos a informação tornou-se a base de tudo, e por meio dela toma-se decisões. Portanto, seu valor é significativo e perdê-la pode custar muito para as organizações. Com isso o investimento na área de tecnologia da informação, ou TI vem expandindo cada vez mais, com o objetivo de preservar, compartilhar e garantir que sempre esteja disponível (LOPES, MARTINS e ANDRADE, 2016).

De acordo com Vaquero *et al.* (2009), Computação em nuvem é um grupo de recursos virtuais de fácil acesso, especificamente o *hardware*, *software*, plataformas de desenvolvimento e serviços. Podendo ser dinamicamente reconfigurados para que possa se ajustar a trabalhos variáveis, melhorando seu desempenho de uso.

O propósito da nuvem é reduzir os gastos operacionais e manter a equipe de TI voltada para desenvolvimento de projetos estratégicos e não preocupados com o funcionamento do *Datacenter* (VELTE, VELTE e ELSENPETER, 2012).

Segundo Veras (2012), *Cloud Computing* representa inovações na área de TI, partindo de um modelo com base em compra de equipamentos para um modelo de aquisições de serviços, proporcionando que aplicações de *Datacenters* sozinhos possam ser integrados na nuvem, em um local de larga escala com utilização elástica dos recursos, que vai se adaptando conforme a demanda atual.

Todavia, a partir de 04 de novembro de 2013 com o Decreto 8.135, deixou de ser lícito o armazenamento de dados institucionais utilizando nuvens fora do Brasil (PRESIDÊNCIA DA REPÚBLICA, 2013). Contudo o *Owncloud* é uma aplicação *web* cujo propósito é sincronização e compartilhamento de arquivos, e por ser *open source* permite que o usuário altere o código conforme a sua necessidade (KARLITSCHKEK, 2017).

É neste contexto que o trabalho está inserido, tendo como objetivo o desenvolvimento de uma nuvem privada para armazenamento de informações confidenciais, visando uma maior segurança e disponibilidade do mesmo.

1.1. OBJETIVOS

O presente projeto tem por objetivo o desenvolvimento de uma nuvem privada de armazenamento utilizando a ferramenta *Owncloud* utilizando a tecnologia *Raspberry PI* como solução de hardware, por ser mais acessível aos usuários. A nuvem aqui proposta possui uma sincronização de arquivos e um método de backup de dados.

1.2. JUSTIFICATIVAS

A justificativa para o desenvolvimento deste trabalho é a possibilidade de apresentar um novo modelo que ainda não foi muito explorado pelas organizações, e que já é uma grande tendência. O modelo *Cloud Computing* é novo e proporciona uma série de vantagens e funcionalidades, trazendo maior conforto ao criar um ambiente em que os usuários poderão guardar seus arquivos de todos os tipos em um local seguro, podendo acessá-los em qualquer lugar, que tenha acesso à Internet.

1.3. MOTIVAÇÕES

A ideia básica da computação em nuvem é que o fornecimento de recursos computacionais seja de responsabilidades de empresas especializadas, em que apenas eles se preocupem em gerencia-los e mantê-los, e ainda sejam disponibilizados como serviços (CARR, 2008).

Logo, esta proposta é desafiadora pois representa uma quebra de paradigma, visto que até pouco tempo atrás, empresas e pessoas utilizavam os recursos computacionais de forma proprietária e eram responsáveis pela manutenção e atualização dos recursos. Com isso, o desenvolvimento deste trabalho consiste no fato de ser um assunto ainda pouco explorado e por possuir um progresso exponencial.

1.4. METODOLOGIA

Para a realização do presente trabalho, foi feita a revisão da literatura para melhor entendimento das ferramentas que serão utilizadas. Com isso o Sistema Operacional Raspbian¹ foi instalado em um Raspberry Pi, e foi inserido os pacotes necessários para o funcionamento. Esses pacotes serão o servidor web Nginx², a linguagem PHP. Visando, a implementação de uma nuvem privada para armazenamento dinâmico de dados com o uso da tecnologia Owncloud e Raspberry Pi.

¹ Raspbian – é uma distribuição Linux criada para rodar nos Raspberry Pi derivada do Debian.

² Nginx - é responsável por disponibilizar páginas e todos os recursos que podem ser acessados pelo usuário.

2. REVISÃO BIBLIOGRÁFICA

2.1. COMPUTAÇÃO EM NUVEM

A Computação em Nuvem é um termo que vem sendo utilizado amplamente desde que começou a ser popularizado em 2008, tornando-se mais conhecido desde que virou realidade aos usuários da Internet. Seu novo modelo de computação em que os recursos são dinamicamente escaláveis e podendo ser virtualizados e disponibilizados como serviços pela Internet, que consistem nas infraestruturas computacionais contidas nos datacenters pelo mundo (BITTENCOURT e MANOLA, 2011).

Contudo este termo foi difundido em 2006, pelo Eric Schmidt o então CEO (*Chief Executive Officer* – Diretor Executivo) da Google para descrever os serviços da própria empresa, e posteriormente a Amazon utilizou o mesmo termo para lançar seu serviço EC2 (*Elastic Compute Cloud*) (GILDER, 2006).

De acordo com Rushel, Zanotto e Mota (2010) a computação em nuvem é um ambiente projetado para trabalhar em redundância onde o processamento é centralizado e seus usuários estão movendo suas informações e dados a todos os momentos, possibilitando o acesso ao mesmo em todos os lugares e de forma simples.

2.2. DEFINIÇÃO

Para Cearly (2009), computação em nuvem pode ser definido como um paradigma de infraestrutura que permite o estabelecimento do SaaS (software como serviço), sendo um conjunto de serviços com base na *web* e com o objetivo de fornecer funcionalidades, que até o momento precisava de grandes investimentos em *hardware* e *software*, e seu funcionamento era através de um modelo de pagar pelo uso. A definição é um modelo de computação onde os recursos são escaláveis e elásticos podendo ser disponibilizados como serviço para os usuários finais pela Internet.

De acordo com Buyya (2008), a nuvem é um conjunto de computadores virtualizados e interconectados em um tipo de sistema paralelo e distribuído que são providos de forma dinâmica e unificado. Os recursos são fornecidos e regulados através de acordos entre as partes.

Das várias propostas para se definir computação em nuvem, para a agencia governamental dos Estados Unidos *National Institute of Standards and Technology* (NIST 2011) diz ser um modelo que permitir acessos a rede sob demanda de modo adequado a empresa ou usuário, sendo recursos configuráveis que podem ser alterados com poucos ajustes.

2.3. MODELOS DE SERVIÇOS

A Internet é imprescindível para o modelo de computação em nuvem, porem o mesmo é algo muito maior e complexo. Podendo ser tanto um software quanto uma infraestruturra (MOELLER, 2010).

Para Veras (2012), o ambiente de computação em nuvem é composto por serviços, sendo três modelos importantes.

A estrutura da nuvem é definida como mostra a Figura 1, onde um serviço serve de suporte ao outro.



Figura 1 - Representação da Estrutura da Nuvem
Fonte: Adaptado de VERAS (2011), p 31.

2.3.1. SaaS

Software como um serviço, *Software-as-a-Service* (SaaS): basicamente se refere ao *software* na nuvem. Para Reese (2009), é um modelo de *software* implantado na web, onde seu objetivo é ser disponibilizado por um navegador web. Com isso não há

necessidades de se preocupar com os requisitos do sistema que esse *software* será processado, porém é de suma importância ter uma conexão com serviços de internet. Um exemplo é o pacote Microsoft Word e o Google Docs, por serem *software* local e web respectivamente. Enquanto o Word deve ser instalado e se preocupar com espaço em disco HD (*Hard Drive*) e sistema operacional, enquanto o Google Docs tem a mesma funcionalidade, porém é necessário apenas conexão com a Internet e um navegador instalado na máquina.

2.3.2. Paas

Para McGrath(2012), a computação em nuvem PaaS (*Platform-as-a-Service*) significando Plataforma como Serviço. Podendo-se dizer que essa é uma camada intermediária, cujo o objetivo é facilitar o desenvolvimento de aplicações voltadas a usuários de uma nuvem, ocasionando um ambiente de desenvolvimento para as aplicações e fornecendo um sistema operacional e linguagens de programação, com o objetivo de agilizar o processo.

2.3.3. IaaS

Representando a camada inferior de modo a compor a base, pois contém uma plataforma para o desenvolvimento de aplicações e testes. De acordo com Sousa (2009), seu principal intuito é tornar o fornecimento dos recursos como servidores, redes, armazenamento mais fácil e acessível. Seu modelo de infraestrutura é baseado na virtualização de *hardware* que pode ser dinamicamente escalável podendo aumentar ou diminuir seu recurso de acordo com as necessidades das aplicações.

2.4. MODELOS DE IMPLANTAÇÃO

O tipo de nuvem depende da necessidade da aplicação a ser utilizada e que serão implantadas. Logo os tipos de acesso e restrições dependem unicamente do tipo de negócio e do nível desejado. Percebe-se que algumas entidades não querem que todos os usuários possam acessar e utilizar determinados recursos em seu ambiente em nuvem. Nos dias de hoje existem vários tipos de implantações de nuvem, porém quatro deles são as mais utilizadas (JADEJA e MODI, 2012).

2.4.1. Nuvem Pública

Pode ser definido como nuvem publica as infraestruturas que são disponibilizadas para o público em geral e que podem ser acessadas por qualquer usuário que conheça a localização do serviço, no qual não possui nenhuma técnica para restrição de acesso. A nuvem publica busca fornecer aos clientes um serviço de TI sem complexidades, no qual o provedor assume todas as obrigações para com a manutenção, gerenciamento e instalação e disponibilidade. Geralmente os serviços são disponibilizados de forma simples com configurações definidas para deixar o mais conveniente para os usuários, portanto, a nuvem publica não é o mais pertinente para usuários que almejam mais segurança e restrições (VERAS, 2011).

2.4.2. Nuvem Privada

Este modelo de infraestrutura é proprietário ou alugado por alguma entidade no qual ela é unicamente operada pela mesma, podendo ser local ou remota possuindo suas políticas de acessos.

De acordo com Taurion (2009), a peculiaridade que diferencia as nuvens privadas são as restrições de acesso, pois cada uma se encontra por de trás do *firewall* de cada organização, adotando a tecnologia e desfrutando de suas vantagens, todavia mantendo as regras de segurança da empresa. O custo e os obstáculos para o estabelecimento de uma nuvem privada são as vezes excessivo, e as operações constantes da nuvem pode ultrapassar os custos de uma nuvem pública. Contudo as vantagens de uma nuvem privada é a opção de maior detalhamento sobre os recursos que a constituem permitindo a empresa acesso as configurações possíveis.

2.4.3. Nuvem Comunitária

A nuvem comunitária tem por objetivo o compartilhamento de uma mesma nuvem para inúmeras empresas, que dispõem interesses em comuns como os quesitos de segurança e políticas da mesma. O modelo aceita acesso local e remoto, normalmente é gerenciado por um grupo da comunidade ou terceiros (JADEJA e MODI, 2012).

2.4.4. Nuvem híbrida

A infraestrutura desta nuvem é composta por pelo menos duas nuvens, que mantem as características originais de seu modelo, contudo estarão interligadas por uma tecnologia padronizada ou que permite a portabilidade dos dados e aplicações (VERAS, 2011).

Uma nuvem híbrida bem construída poderia atender processos críticos e seguros, como pagamento de clientes ou de serviços secundários, tais como processamento da folha de pagamento de funcionários. Porém a restrição desta nuvem é a dificuldade de se criar e administrar algo desse porte. Os serviços de várias fontes devem ser recebidos e fornecidos como se fossem do mesmo local de origem, e a comunicação entre os componentes públicos e privados tendem a tornar a implementação ainda mais complexa.

2.5. SEGURANÇA

A segurança em computação em nuvem é uma questão um pouco polêmica ainda, pois há usuários que acreditam na segurança da nuvem, enquanto outros tem um certo pessimismo. Todos os sistemas estão sujeitos a falhas, e com isso as empresas escolhem armazenar seus dados localmente por não confiar nos sistemas operados por provedores de nuvem, e que seus riscos são grandes o suficiente a ponto de eles escolherem sua própria infraestrutura negando totalmente a possibilidade de disponibilizá-los em uma nuvem (CIO, 2010).

2.6. OWNCLOUD

Owncloud é um *software* de armazenamento que independe de onde esteja o servidor que armazena os dados pois ele é gratuito, *open-source* e é baseado em nuvem. Logo sua plataforma pode proporcionar um mecanismo que possibilita o usuário ver, compartilhar e sincronizar facilmente os contatos, calendários e repositórios através de todos os dispositivos e possui uma edição básica de web. Com isso o Owncloud agrada a maioria dos seus usuários, pois agora todas suas fotos, músicas, tarefas e arquivos estão em uma nuvem privada, o usuário escolhe onde hospedar, podendo ser uma plataforma alugada ou um servidor local o qual foi escolhido para esse trabalho (GLAUBER G. F, 2013).

2.7. RASPBERRY PI

Atualmente as empresas querem criar um dispositivo que possa fazer o mesmo que um computador comum, mas com um preço que seja relativamente acessível a todos, e para atrair jovens interessados. Com isso, o desenvolvimento do microcomputador Raspberry Pi revolucionou toda a indústria com Eben Upton e sua equipe em 2006, pois em uma placa do tamanho de um cartão possui toda a tecnologia de um desktop e ainda uma flexibilidade superior. É neste contexto que as empresas perceberam que um novo mercado de dispositivos estava se formando, logo o Raspberry Pi veio conquistar o seu lugar em um meio totalmente novo. A primeira diferença marcante entre a arquitetura é a memória ram, que para o Raspberry Pi seus modelos variam de 256Mb a 1Gb com tecnologia DDR2 e conta com a tecnologia Swap. As conexões USB possuem uma diferença também pois ele conta com 4 entradas sendo 2 de 2.0 e 2 de 3.0. A saída HDMI possui uma diferença também nas versões aceitas, o Raspberry aceita versões 1.3 e 1.4. O periférico de rede RJ45 possui barramento de 10/100 e suporta operações *full-duplex* e *half-duplex*. Possuindo o processador Cortex-A53 baseado no Atmel ATmega organização que produz os processadores ARM, mas possuem algumas peculiaridades, pois o seu *clock* chega a 1.2 GHz com uma placa gráfica *Broadcom VideoCore IV*. Possuindo também saída de áudio (3.5 mm) e slot para cartão para aumentar a memória interna e possui 40 pinos Pi-2. A fonte de alimentação de 5V conectada a uma entrada micro-USB. E um grande diferencial do Raspberry Pi é que ele possui Wi-Fi e Bluetooth integrados na placa. Após essa análise detalhada pode-se perceber que a arquitetura é semelhante a um computador, mas com algumas diferenças. Contudo o Raspberry Pi por ter sido o primeiro a sair no mercado ele conta com um número grande de pesquisas e trabalhos relacionados a ele tanto no meio acadêmico quanto os próprios usuários finais (UPTON, 2013).

Na Figura 2, ilustra a placa Raspberry Pi Model B com todos os seus componentes e periféricos de entrada e saídas.



Figura 2 - Placa Raspberry Pi 3 Model B³

2.7.1. Sistemas Operacionais

Além das várias diferenças dos *hardwares*, uma outra composição importante também é o sistema operacional, que por sua vez é o que controla o equipamento. Hoje em dia a maioria dos *desktops* e *notebooks* utilizam os sistemas operacionais Microsoft Windows ou Apple OS X, contudo, essas plataformas são conhecidas por terem os códigos fechados. Todavia os sistemas operacionais de código fechado que guardam seus segredos a sete chaves, podem apenas demonstrar seus produtos finais sem poder verificar como são feitos. Enquanto o Raspberry Pi foi planejado para utilizar o sistema operacional Linux, que diferente dos de código fechado é possível fazer o download do código-fonte pertencente ao sistema operacional e fazer seja qual for as alterações necessárias. Logo com essa oportunidade de modificação, foi possibilitado que o Linux se adaptasse para que pudesse ser executasse no Raspberry Pi, tendo algumas versões como o Debian, Fedora Remix e Arch Linux foram utilizados. Cada uma atendendo uma necessidade específica, porem todas com uma semelhança, a de terem o código aberto. A essência do Raspberry Pi é a de proporcionar e estimular o ensino de computação básica, pois o dispositivo disponibiliza facilidades, pois muitos projetos o estão utilizando, inclusive este trabalho (HALFACREE e UBTON, 2013).

³ Disponível em < <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>> Acesso em 15 mar. 2018.

3. INSTALAÇÃO E CONFIGURAÇÃO DA FERRAMENTA OWNCLOUD

O presente projeto teve por objetivo o desenvolvimento de uma nuvem privada utilizando a tecnologia Owncloud, pois o mesmo ofereceu uma maior disponibilidade de recursos que o próprio administrador da nuvem pode optar por usar ou não, logo, este processo resultou em um ganho significativo em relação aos outros tipos de nuvem que o usuário só tem acesso ao produto final.

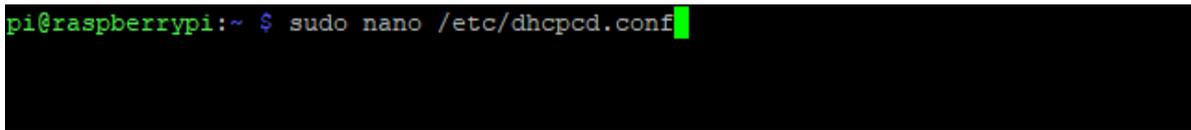
A ferramenta foi instalada e configurada em um servidor com o Sistema Operacional Linux Raspbian Stretch. A versão do Owncloud para este trabalho foi a 10.0.8. As versões da ferramenta e do sistema operacional implementada no estudo de caso são umas das mais estáveis até o momento. Primeiramente, foi testado em laboratório, realizadas as validações necessárias e quando todos os requisitos foram cumpridos então foi realizada a implementação prática. Para a instalação e configuração do servidor foi usado o manual oficial⁴ disponível no site da comunidade Owncloud e alguns conhecimentos adquiridos durante todo o curso de Ciências da Computação.

Na implementação prática percebeu-se que o Raspberry PI atendeu todas as expectativas oferecendo recursos computacionais satisfatórios para a implementação da tecnologia bem como a apresentação de baixo custo financeiro e baixo consumo de energia.

3.1. CONFIGURAÇÃO DE REDE

Para uma maior disponibilidade e comodidade foi usado um IP estático, que será mostrado a seguir.

Para defini-lo foi editado o arquivo de interfaces `dhcpcd.conf`, como ilustra a figura 3.



```
pi@raspberrypi:~ $ sudo nano /etc/dhcpcd.conf
```

Figura 3 - Tela de exibição do arquivo de configuração de interfaces

Fonte: do autor

⁴ https://doc.owncloud.org/server/latest/admin_manual/installation/source_installation.html

No arquivo foram incluídas as seguintes linhas, como ilustra a figura 4.

```
iface lo inet loopback

auto eth0
interface eth0
static ip_address=192.168.0.62/24
static routers=192.168.0.1
static domain_name_servers=192.168.0.1 8.8.8.8
```

Figura 4 – Tela de exibição do conteúdo do arquivo de configuração de interfaces

Fonte: do autor

Para a configuração entrar em vigor foi reiniciado o serviço de rede com o comando *restart*, como ilustra a figura 5.

```
root@raspberrypi:/home/pi# service networking restart
```

Figura 5 – Tela de exibição do comando de reinicialização do serviço de rede

Fonte: do autor

3.2. SERVIÇO DE ACESSO REMOTO SSH

O serviço SSH foi habilitado para proporcionar o acesso remoto ao servidor de armazenamento em nuvem. Através deste serviço é possível gerenciar a aplicação de outro computador local ou até mesmo externo, através da internet.

Inicialmente foi atualizado a lista de repositórios do Sistema Operacional com o comando *update* e posteriormente instalando os pacotes de atualizações dos programas com o comando *upgrade*, como ilustra a figura 6.

```
root@raspberrypi:/home/pi# apt-get update && apt-get upgrade
```

Figura 6 – Tela de exibição do comando de atualização do repositório e instalação das atualizações dos programas

Fonte: do autor

Logo após foi habilitado a interface de acesso remoto SSH com os seguintes comandos, como ilustra a figura 7.

```
pi@raspberrypi:~ $ sudo raspi-config
```

Figura 7 – Tela de exibição do comando para configuração do Raspberry Pi

Fonte: do autor

Com isso foi escolhido a opção 5 no menu, que se trata da configuração de interfaces do Raspberry Pi, como mostra na figura 8.

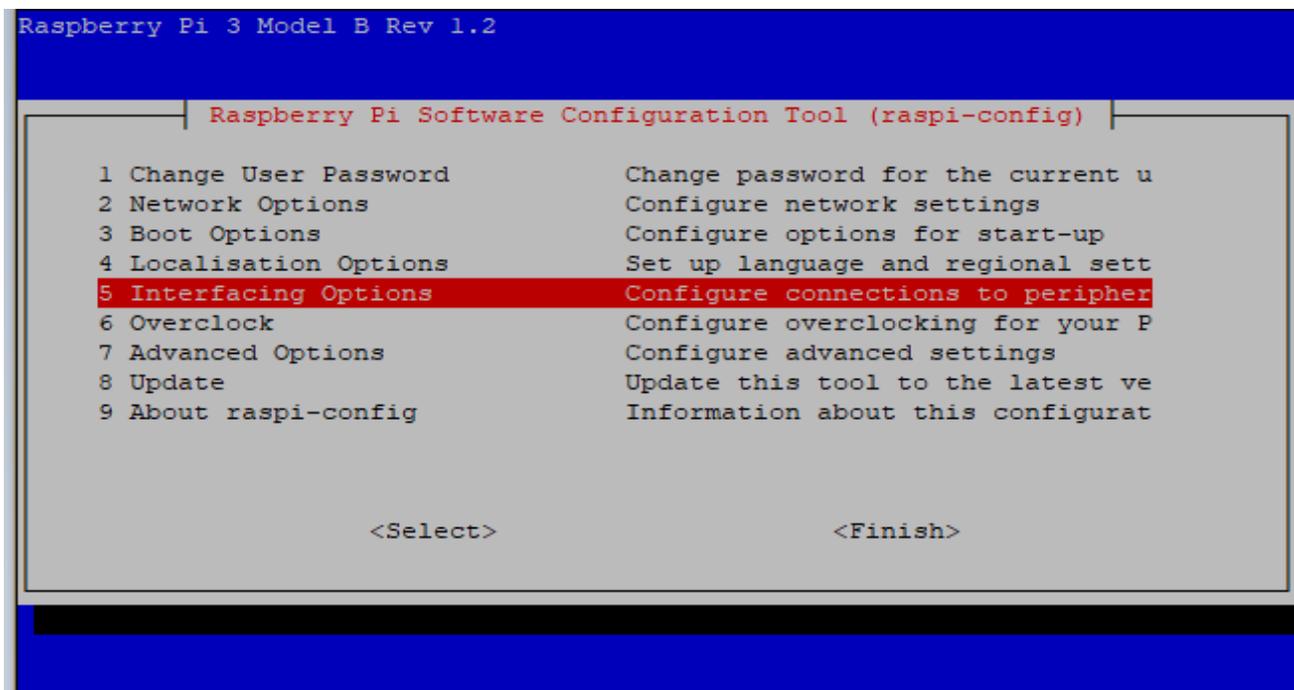


Figura 8 – Tela de exibição para selecionar a opção de interfaces

Fonte: do autor

E para finalizar foi feita a escolha da interface SSH para ser habilitada, como ilustra a figura 9.

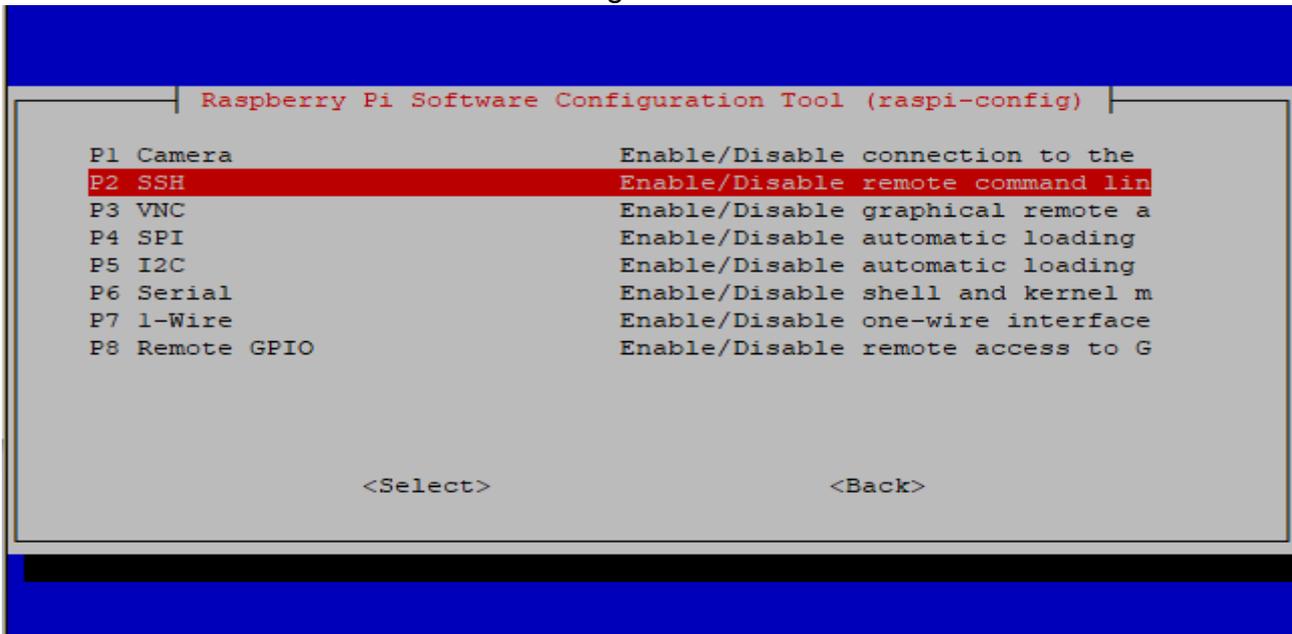


Figura 9 – Tela de exibição para escolha da interface SSH
Fonte: do autor

E por fim foi confirmado a habilitação do serviço SSH, como mostra a figura 10.

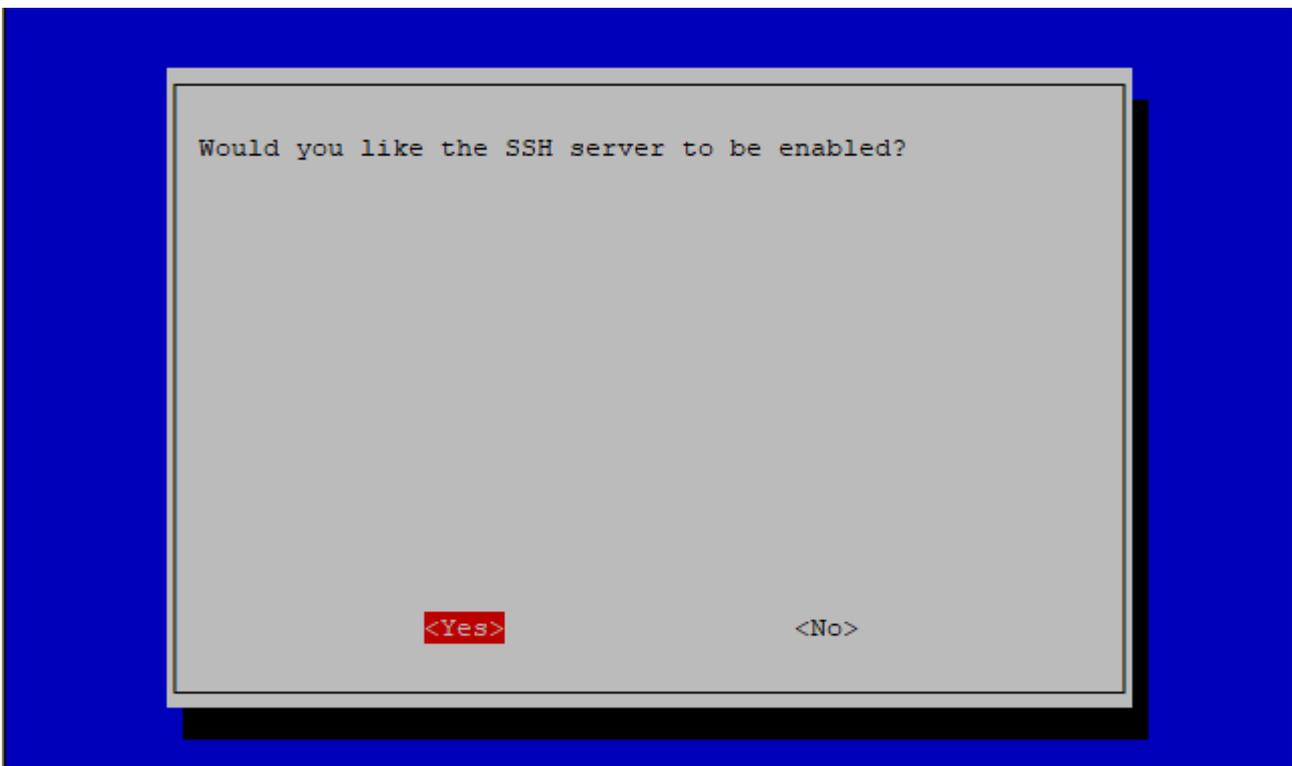


Figura 10 – Tela de exibição para confirmar a ativação do serviço SSH
Fonte: do autor

Após ser habilitado o serviço SSH, o servidor já pode ser acessado remotamente através de uma ferramenta específica, como por exemplo o PuTTY⁵ que foi utilizada neste estudo.

Após as configurações deve-se adicionar o usuário `www-data` ao grupo `www-data`, como mostra a figura 11.

```
root@raspberrypi:/home/pi# usermod -a -G www-data www-data
```

Figura 11 – Tela de exibição do comando para atribuir o usuário `www-data` ao grupo `www-data`

Fonte: do autor

3.3. SERVIDOR HTTP NGINX E PHP7

O Nginx é um serviço utilizado como servidor *web*, rápido e leve com inúmeras possibilidades de configuração para melhorar seu desempenho. E para que tudo funcione corretamente precisamos instalar todos os pacotes necessários juntamente com o PHP7, onde pode-se acompanhar a seguir na figura 12.

```
root@raspberrypi:/home/pi# sudo apt-get install nginx openssl ssl-cert php7.0-cl  
i php7.0-sqlite php7.0-gd php7.0-common php7.0-cgi sqlite3 php-pear curl libapr1  
libtool curl libcurl4-openssl-dev php7.0-xml php7.0 php7.0-dev php7.0-curl php7  
.0-gd php7.0-fpm memcached php7.0-memcache php7.0-zip php7.0-intl php7.0-mbstrin  
g varnish
```

Figura 12 – Tela de exibição dos comandos de instalação do servidor Nginx e o PHP7 com todos os pacotes necessários.

Fonte: do autor

O servidor de armazenamento em nuvem depende da instalação destes serviços para que seu funcionamento seja eficaz.

Ainda na parte de configuração do Nginx foi editado o arquivo `sites-available/default`, como ilustra a figura 13.

```
root@raspberrypi:/home/pi# sh -c "echo '' > /etc/nginx/sites-available/default"
```

Figura 13 – Tela de exibição do comando para limpar as configurações do servidor

Fonte: do autor

⁵ <https://putty.org/>

Primeiramente foi apagado todas as configurações que vieram na instalação do servidor Nginx, e após isso foram colocadas as configurações certas para que funcione corretamente o serviço Owncloud, como ilustra a figura 14.

```
root@raspberrypi:/home/pi# nano /etc/nginx/sites-available/default
```

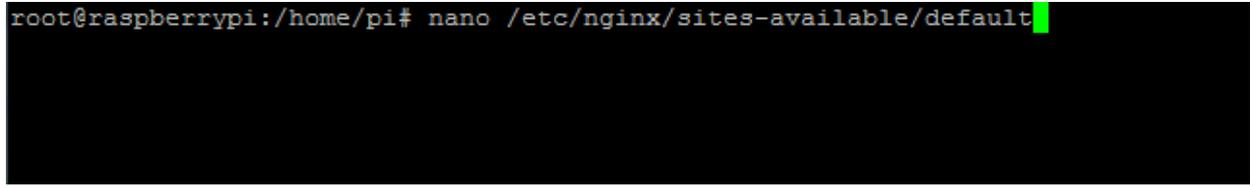
A terminal window with a black background and white text. The prompt is 'root@raspberrypi:/home/pi#'. The command entered is 'nano /etc/nginx/sites-available/default'. A green cursor is visible at the end of the command line.

Figura 14 – Tela de exibição do comando para acessar o arquivo de configuração do servidor Nginx

Fonte: do autor

Configuração necessárias do servidor Nginx, como ilustra a figura 15.

```
GNU nano 2.7.4 Arquivo: /etc/nginx/sites-available/default

upstream php-handler {
    server 127.0.0.1:9000;
}
server {
    listen 80;
    server_name 192.168.0.62;
    return 301 https://$server_name$request_uri; # enforce https
}

server {
    listen 443 ssl;
    server_name 192.168.0.62;
    ssl_certificate /etc/nginx/cert.pem;
    ssl_certificate_key /etc/nginx/cert.key;
    # Path to the root of your installation
    root /var/www/owncloud;
    client_max_body_size 1000M; # set max upload size
    fastcgi_buffers 64 4K;
    rewrite ^/caldav(.*)$ /remote.php/caldav$1 redirect;
    rewrite ^/carddav(.*)$ /remote.php/carddav$1 redirect;
    rewrite ^/webdav(.*)$ /remote.php/webdav$1 redirect;
    index index.php;
    error_page 403 /core/templates/403.php;
    error_page 404 /core/templates/404.php;
    location = /robots.txt {
        allow all;
        log_not_found off;
        access_log off;
    }
    location ~ ^/(?:\.htaccess|data|config|db_structure\.xml|README) {
        deny all;
    }
    location / {
        # The following 2 rules are only needed with webfinger
        rewrite ^/.well-known/host-meta /public.php?service=host-meta last;
        rewrite ^/.well-known/host-meta.json /public.php?service=host-meta-json$;
        rewrite ^/.well-known/carddav /remote.php/carddav/ redirect;
        rewrite ^/.well-known/caldav /remote.php/caldav/ redirect;
        rewrite ^(/core/doc/[\^\./]+)$ $1/index.html;
        try_files $uri $uri/ index.php;
    }
    location ~ \.php(?:$|/) {
        fastcgi_split_path_info ^(.+\.(php))(/.+)$;
        include fastcgi_params;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        fastcgi_param PATH_INFO $fastcgi_path_info;
        fastcgi_param HTTPS on;
        fastcgi_pass php-handler;
    }
    # Optional: set long EXPIRES header on static assets
    location ~* \.(?:jpg|jpeg|gif|bmp|ico|png|css|js|swf)$ {
        expires 30d;
        # Optional: Don't log access to assets
        access_log off;
    }
}
}
```

Figura 15 – Tela de exibição da configuração do servidor Nginx para o funcionamento do Owncloud

Fonte: do autor

Ainda na parte de configurações precisa-se modificar alguns arquivos de configuração do PHP7, como ilustra a figura 16

```
root@raspberrypi:/home/pi# nano /etc/php/7.0/fpm/php.ini
```

Figura 16 – Tela de exibição do comando para acessar as configurações do PHP7

Fonte: do autor

Foi modificado o comando “post_max_size = 512M” para “post_max_size = 2000M”, como mostra a figura 17

```
post_max_size = 2000M
```

Figura 17 – Tela de exibição para alteração do comando post_max_size

Fonte: do autor

E de semelhante modo o comando “upload_max_filesize = 512M” foi modificado para “upload_max_filesize = 2000M”, como ilustra a figura 18

```
upload_max_filesize = 2000M
```

Figura 18 – Tela de exibição para alteração do comando upload_max_filesize

Fonte: do autor

Ainda na parte de configuração foi modificado o arquivo conf, como mostra a figura 19

```
root@raspberrypi:/home/pi# sudo nano /etc/php/7.0/fpm/pool.d/www.conf
```

Figura 19 – Tela de exibição do comando para acessar o arquivo conf

Fonte: do autor

Logo após acessado o arquivo *conf* deve-se modificar o comando “listen:127.0.0.1:9000”, como mostra a figura 20.

```
listen = 127.0.0.1:9000
```

Figura 20 – Tela de exibição para atualizar o comando *listen*

Fonte: do autor

O Raspberry Pi 3 possui 1Gb de memória ram, porém em alguns momentos pode ser que falte memória. Logo vamos modificar o arquivo responsável por swapfile de memória, como ilustra a figura 21.

```
root@raspberrypi:/home/pi# nano /etc/dphys-swapfile
```

Figura 21 – Tela de exibição do comando para acessar o arquivo de swap do Raspberry Pi 3

Fonte: do autor

Após acessar o arquivo swapfile o mesmo foi alterado de “CONF_SWAPSIZE=100” para “CONF_SWAPSIZE=512”, como mostra a figura 22.

```
CONF_SWAPSIZE=512
```

Figura 22 – Tela de exibição para alteração do comando CONF_SWAPSIZE

Fonte: do autor

E por fim o Raspberry Pi 3 deve ser reiniciado para que todas as modificações entrem em vigor, como ilustra a figura 23.

```
root@raspberrypi:/home/pi# reboot
```

Figura 23 – Tela de exibição do comando para reinicialização do Raspberry Pi 3

Fonte: do autor

3.4. ATRIBUIÇÃO DE SEGURANÇA

Para este estudo a habilitação do serviço SSL de conexão segura foi uma das configurações mais importantes para segurança, como mostra a figura 24.

```
root@raspberrypi:/home/pi# openssl req $@ -new -x509 -days 730 -nodes -out /etc/nginx/cert.pem -keyout /etc/nginx/cert.key
```

Figura 24 – Tela de exibição dos comandos para criação de um certificado SSL

Fonte: do autor

Depois de feita a habilitação do serviço de segurança, foi necessário mais algumas configurações básicas para que o processo fosse concluído com êxito.

Após as configurações, foram aplicadas algumas permissões como vemos a seguir, na figura 25.

```
root@raspberrypi:/home/pi# chmod 600 /etc/nginx/cert.pem
```

Figura 25 – Tela de exibição do comando para alterar a permissão no arquivo cert.pem

Fonte: do autor

Mais algumas permissões foram aplicadas, como ilustra a figura 26.

```
root@raspberrypi:/home/pi# chmod 600 /etc/nginx/cert.key
```

Figura 26 – Tela de exibição do comando para alterar a permissão no arquivo cert.key

Fonte: do autor

3.5. INSTALAÇÃO DA FERRAMENTA OWNCLOUD

Inicialmente foi criado um diretório para o Servidor Owncloud como pode-se ver a seguir, na figura 27.

```
root@raspberrypi:/home/pi# sudo mkdir -p /var/www/owncloud
```

Figura 27 – Tela de exibição do comando de criação de diretório para o Owncloud

Fonte: do autor

Em seguida foi acessado diretamente o site para fazer *download* da ferramenta, como mostra a figura 28.

```
root@raspberrypi:/home/pi# wget https://download.owncloud.org/community/owncloud-10.0.8.tar.bz2
```

Figura 28 – Tela de exibição do comando de *download* da ferramenta Owncloud

Fonte: do autor

Após o termino do *download*, o arquivo foi descompactado, como ilustra a figura 29.

```
root@raspberrypi:/home/pi# tar xvf owncloud-10.0.8.tar.bz2
```

Figura 29 – Tela de exibição do comando de descompactar arquivos compactados

Fonte: do autor

E então foi movido para o devido diretório onde deve ser colocado para ser acessado através da rede, como mostra na figura 30.

```
root@raspberrypi:/home/pi# mv owncloud/ /var/www/
```

Figura 30 – Tela de exibição do comando de alteração de diretório

Fonte: do autor

Posteriormente foi dada as devidas permissões para o diretório, como ilustra a figura 31.

```
root@raspberrypi:/home/pi# chown -R www-data:www-data /var/www
```

Figura 31 – Tela de exibição do comando para alteração das permissões do diretório

Fonte: do autor

E por fim foi removido o arquivo compactado do diretório, como ilustra a figura 32.

```
root@raspberrypi:/home/pi# rm -rf owncloud owncloud-10.0.8.tar.bz2
```

Figura 32 – Tela de exibição do comando de remoção do arquivo Owncloud

Fonte: do autor

Neste ponto pode-se dizer que a ferramenta já foi instalada, porém ainda foi preciso definir algumas configurações.

Para isso foi preciso acessar o diretório do Owncloud para que sejam feitas as modificações necessárias, como ilustra a figura 33

```
root@raspberrypi:/home# cd /var/www/owncloud
```

Figura 33 – Tela de exibição do comando para acessar o diretório Owncloud

Fonte: do autor

Após isso foi editado o arquivo *htaccess* como pode-se ver a seguir, na figura 34

```
root@raspberrypi:/var/www/owncloud# nano .htaccess
```

Figura 34 – Tela de exibição para acessar o arquivo *htaccess*

Fonte: do autor

Os três comandos a seguir foram modificados de “512M” para “2000M” respectivamente, como ilustra a figura 35.

```
php_value upload_max_filesize 2000M  
php_value post_max_size 2000M  
php_value memory_limit 2000M
```

Figura 35 – Tela de exibição dos comandos modificados do arquivo *htaccess*

Fonte: do autor

Ainda no diretório do Owncloud foi modificado o arquivo *user.ini* como pode-se ver a seguir, na figura 36.

```
root@raspberrypi:/var/www/owncloud# nano .user.ini
```

Figura 36 – Tela de exibição do comando para acessar o arquivo *user.ini*

Fonte: do autor

E de semelhante modo foram modificados os três comandos que aparecem na Figura 37, seus valores foram modificados de “512M” para “2000M” respectivamente, como ilustra na figura 37.

```
upload_max_filesize=2000M  
post_max_size=2000M  
memory_limit=2000M
```

Figura 37 – Tela de exibição dos comandos modificados do arquivo *user.ini*

Fonte: do autor

3.6. CONFIGURAÇÃO DE UMA UNIDADE DE ARMAZENAMENTO

O armazenamento de dados é uma das principais características deste estudo, logo um dos objetivos é para permitir que o Owncloud possa armazenar seus arquivos em uma unidade externa.

Em primeiro lugar, deve-se ter uma unidade NTFS e será necessário instalar o pacote NTFS para que o mesmo possa funcionar corretamente, como mostra a figura 38.

```
root@raspberrypi:/home/pi# apt-get install ntfs-3g
```

Figura 38 – Tela de exibição do comando de instalação do pacote NTFS

Fonte: do autor

Após instalado o pacote necessário, precisamos criar um diretório, como ilustra a figura 39.

```
root@raspberrypi:/home/pi# mkdir /media/owncloudrive
```

Figura 39 – Tela de exibição do comando para criação do diretório owncloudrive

Fonte: do autor

Criado o diretório, precisa-se descobrir o *GID* e o *UID* pois será necessário logo mais, como ilustra a figura 40

```
root@raspberrypi:/home/pi# id -g www-data
33
root@raspberrypi:/home/pi# id -u www-data
33
```

Figura 40 – Tela de exibição dos comandos para descobrir o *UID* e o *GID*

Fonte: do autor

Agora precisa-se obter o *UUID* da unidade externa, para que mesmo que ela se conecte em uma porta USB diferente, o Raspberry Pi se lembre dele, como mostra a figura 41

```
root@raspberrypi:/home/pi# ls -l /dev/disk/by-uuid
```

Figura 41 – Tela de exibição do comando para descobrir o *UUID* da unidade externa

Fonte: do autor

Após buscar as unidades conectadas, deve se identificar a unidade externa na lista de dispositivos. A unidade possui um código que será usado logo mais e seu final deve terminar com “/sda1”, como ilustra a figura 42

```
root@raspberrypi:/home/pi# ls -l /dev/disk/by-uuid
total 0
lrwxrwxrwx 1 root root 15 ago  4 11:17 499A-594E -> ../../mmcblk0p1
lrwxrwxrwx 1 root root 10 ago  4 11:17 5634B7DF34B7BFF3 -> ../../sda1
lrwxrwxrwx 1 root root 15 ago  4 11:17 A150-9254 -> ../../mmcblk0p6
lrwxrwxrwx 1 root root 15 ago  4 11:17 c30e0807-4086-4c03-8fef-ee0919b8115d -> .
../../mmcblk0p5
lrwxrwxrwx 1 root root 15 ago  4 11:17 c8ba5006-8a87-4928-afe1-cc1efd1d7cf3 -> .
../../mmcblk0p9
lrwxrwxrwx 1 root root 15 ago  4 11:17 D6B4-2E6A -> ../../mmcblk0p8
lrwxrwxrwx 1 root root 15 ago  4 11:17 e70706a6-61f8-4fcf-ad29-50cc1ac707c8 -> .
../../mmcblk0p7
```

Figura 42 – Tela de exibição para obter o *UUID* da unidade externa conectada no Pi

Fonte: do autor

E por fim acessar o arquivo “fstab” para adicionar a unidade e que ela inicie com as permissões corretas, como mostra a figura 43.

```
root@raspberrypi:/home/pi# nano /etc/fstab
```

Figura 43 – Tela de exibição do comando para acessar o arquivo fstab

Fonte: do autor

Com isso deve-se adicionar uma linha no arquivo, utilizando o UID, o GID e o UUID obtidos acima, e deve ser colocado em apenas uma linha, como mostra na figura 44.

```
GNU nano 2.7.4                               Arquivo: /etc/fstab                               Modificado
proc /proc proc defaults 0 0
/dev/mmcblk0p6 /boot vfat defaults 0 2
/dev/mmcblk0p7 / ext4 defaults,noatime 0 1
UUID=5634B7DF34B7BFF3 /media/owncloudrive auto nofail,uid=33,gid=33,umask=0027,dmask=0027,noatime 0 0
```

Figura 44 – Tela de exibição do arquivo para adicionar a unidade externa com suas permissões

Fonte: do autor

E para que todas configurações entre em vigor e suas unidades possam ser montadas corretamente, o Raspberry Pi deve ser reiniciado.

3.7. CONFIGURAÇÃO DA INTERFACE GRÁFICA

Com todas as medidas tomadas para o devido funcionamento do servidor de armazenamento em nuvem, foram necessárias algumas configurações iniciais para começar sua utilização.

Na primeira vez que foi acessado o servidor pelo navegador como ilustra a figura 45, foram solicitados alguns dados utilizados anteriormente. Nesta parte o usuário vai criar o usuário e senha do administrador do servidor e irá informar o caminho para o armazenamento externo que já foi definido.

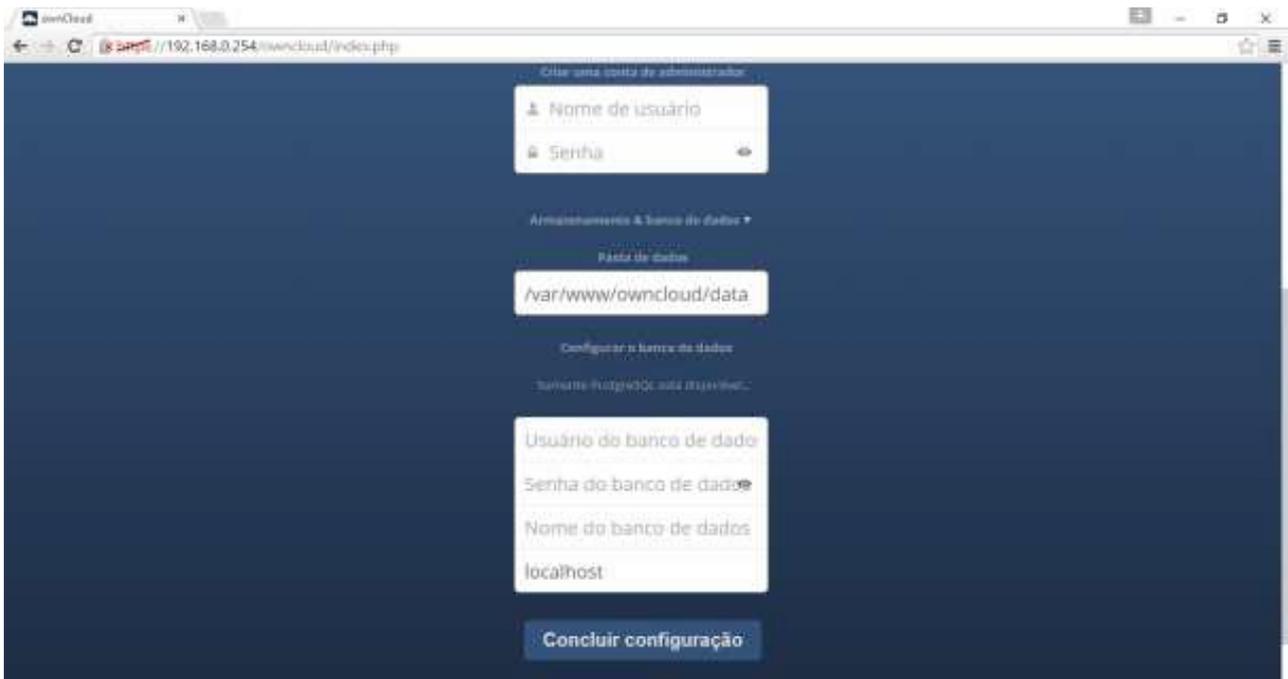


Figura 45 – Tela de exibição da página de configurações finais da ferramenta Owncloud

Fonte: do autor

Após o preenchimento dos dados, o servidor já está pronto para uso, como ilustra a figura 46.



Figura 46 – Tela de exibição da página de bem-vindo da interface gráfica do Owncloud

Fonte: do autor

E por fim como ilustra a figura 47, foram criados os usuários individuais cada um com suas devidas permissões.

| Nome de Usuário | Senha | user | Criar | | |
|-----------------|---------------|-------|--------|------------------|------|
| Nome de Usuário | Nome Completo | Senha | Grupos | Grupo Admin para | Cota |
| Allahu Akbar | Allahu Akbar | ***** | user | user | 1 GB |
| Joao | Joao | ***** | user | user | 2 GB |
| lucasimeao | lucasimeao | ***** | admin | nenhum grupo | 2 GB |
| Percy | Percy | ***** | user | user | 1 GB |
| Sundays | Sundays | ***** | user | user | 1 GB |
| Tattu | Tattu | ***** | user | user | 1 GB |
| Viny52 | Viny52 | ***** | user | user | 1 GB |

Figura 47 – Tela de exibição do ambiente gráfico da tela de administrador do Owncloud

Fonte: do autor

3.8. INSTALAÇÃO DO APLICATIVO DE SINCRONIZAÇÃO

Foi realizado a instalação do aplicativo que sincroniza os arquivos de uma determinada pasta para o servidor. O aplicativo ao detectar uma inclusão ou alteração de algum arquivo de uma pasta já configurada, faz automaticamente uma cópia do arquivo, ou apenas a alteração do mesmo no servidor.

Após a instalação, o sincronizador requer uma configuração para se conectar ao servidor de armazenamento em nuvem, precisando informar o IP de destino, como ilustra a figura 48.

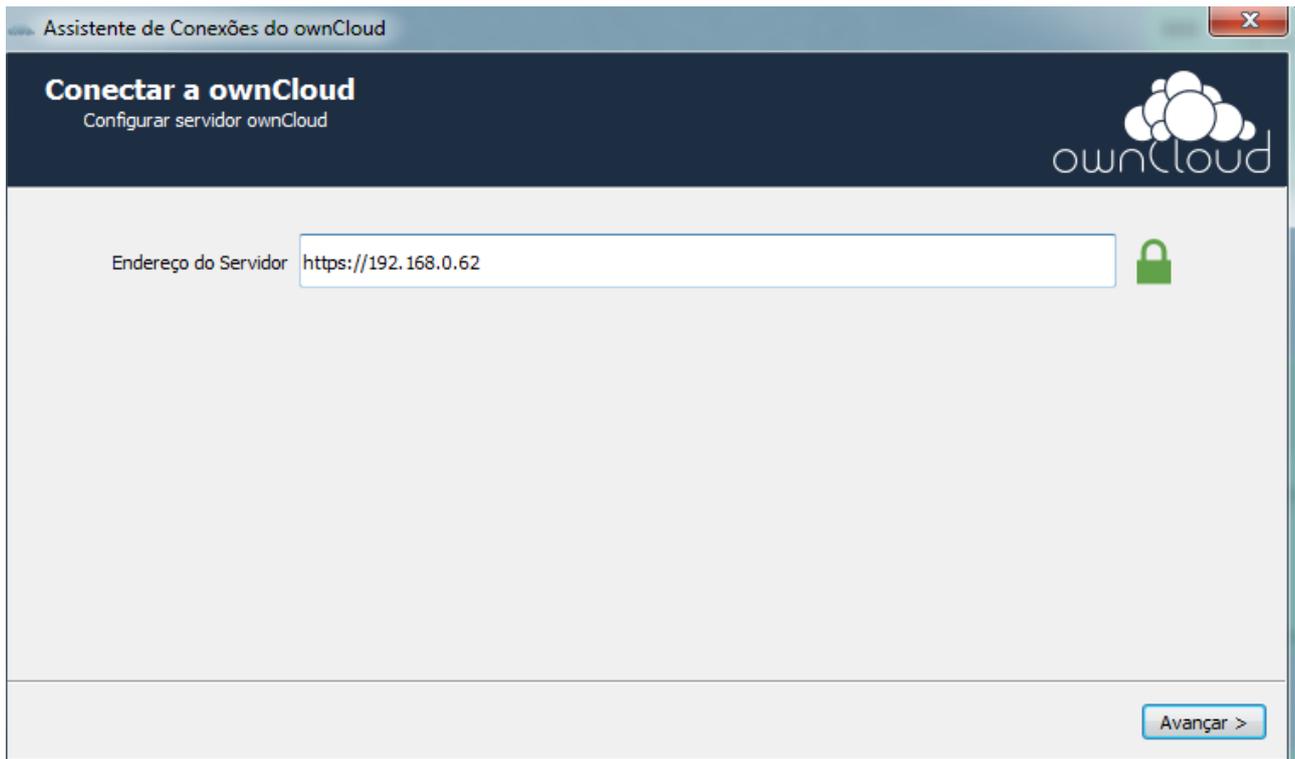


Figura 48 – Tela de exibição de configuração do aplicativo de sincronização 1

Fonte: do autor

Depois de informar o endereço do servidor, será solicitado usuário e senha criado anteriormente, como mostra na Figura 49.

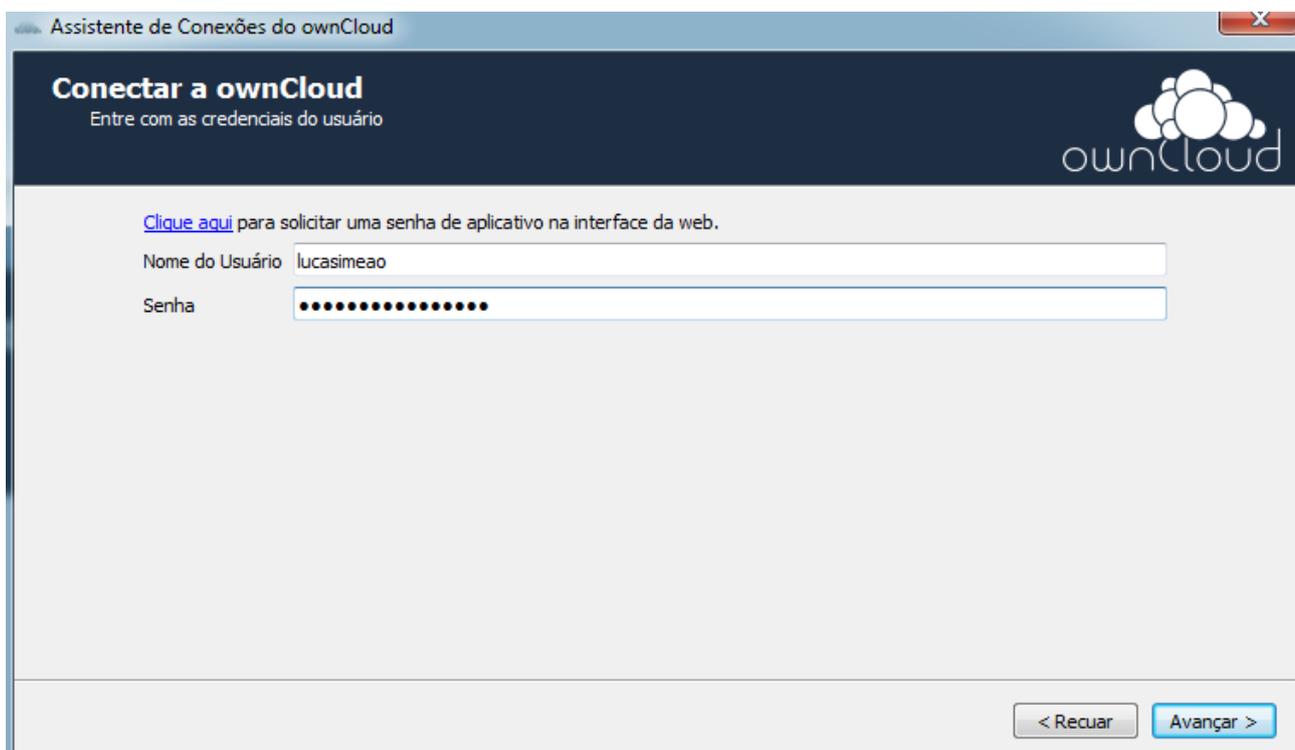


Figura 49 – Tela de exibição de configuração do aplicativo de sincronização 2

Fonte: do autor

Nesta etapa como mostra a figura 50, foi configurado para sincronizar os dados do usuário Lucas. Após esta configuração foi preciso escolher em qual diretório na máquina local será escolhido para ser sincronizado e com o servidor e o que seria sincronizado.

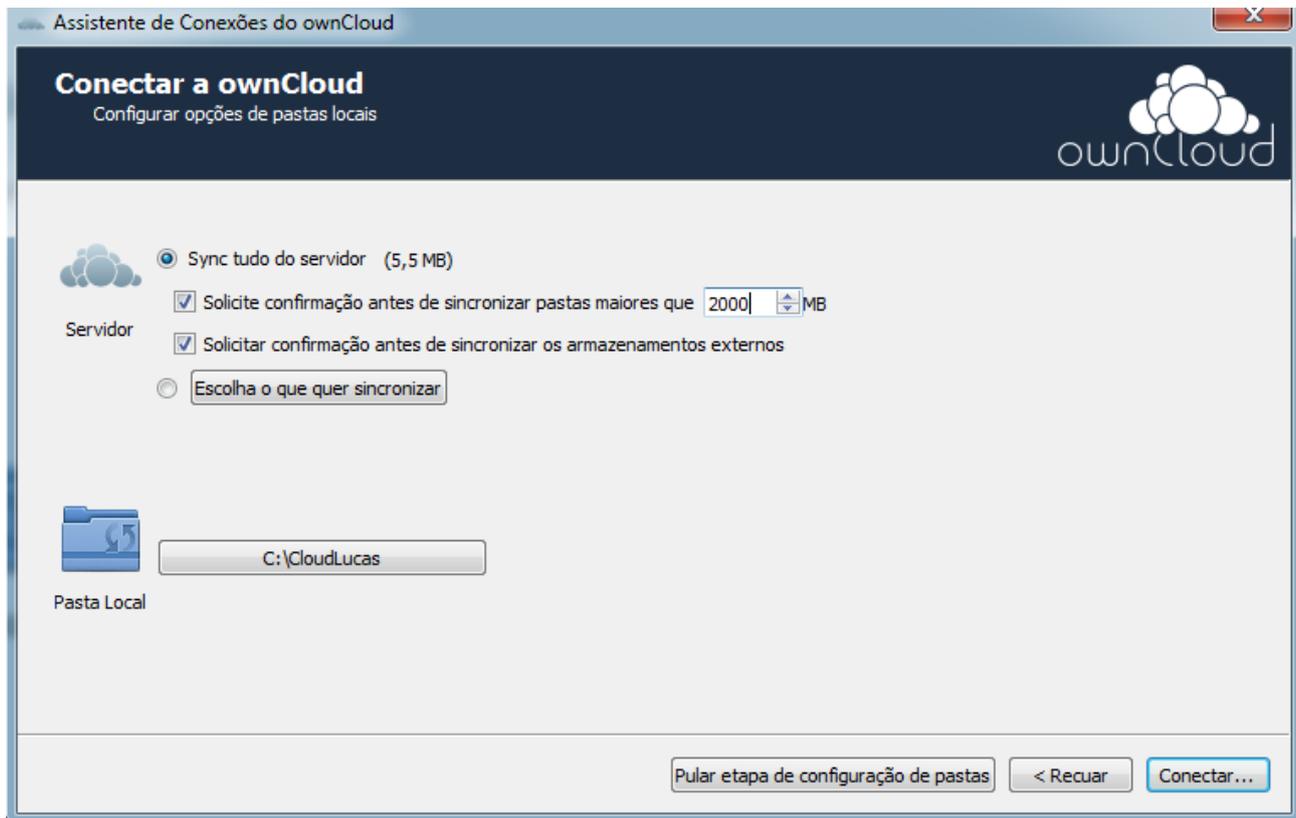


Figura 50 - Tela de exibição de configuração do aplicativo de sincronização 3

Fonte: do autor

Foi criado um diretório específico para a sincronização dos arquivos. Depois de se conectar todos os dados são automaticamente sincronizados e o aplicativo registra todos os processos.

A Figura 51 mostra o aplicativo depois de instalado e configurado, onde mostra a sincronização dos arquivos do usuário Lucas no servidor com o diretório CloudLucas configurado na máquina local.

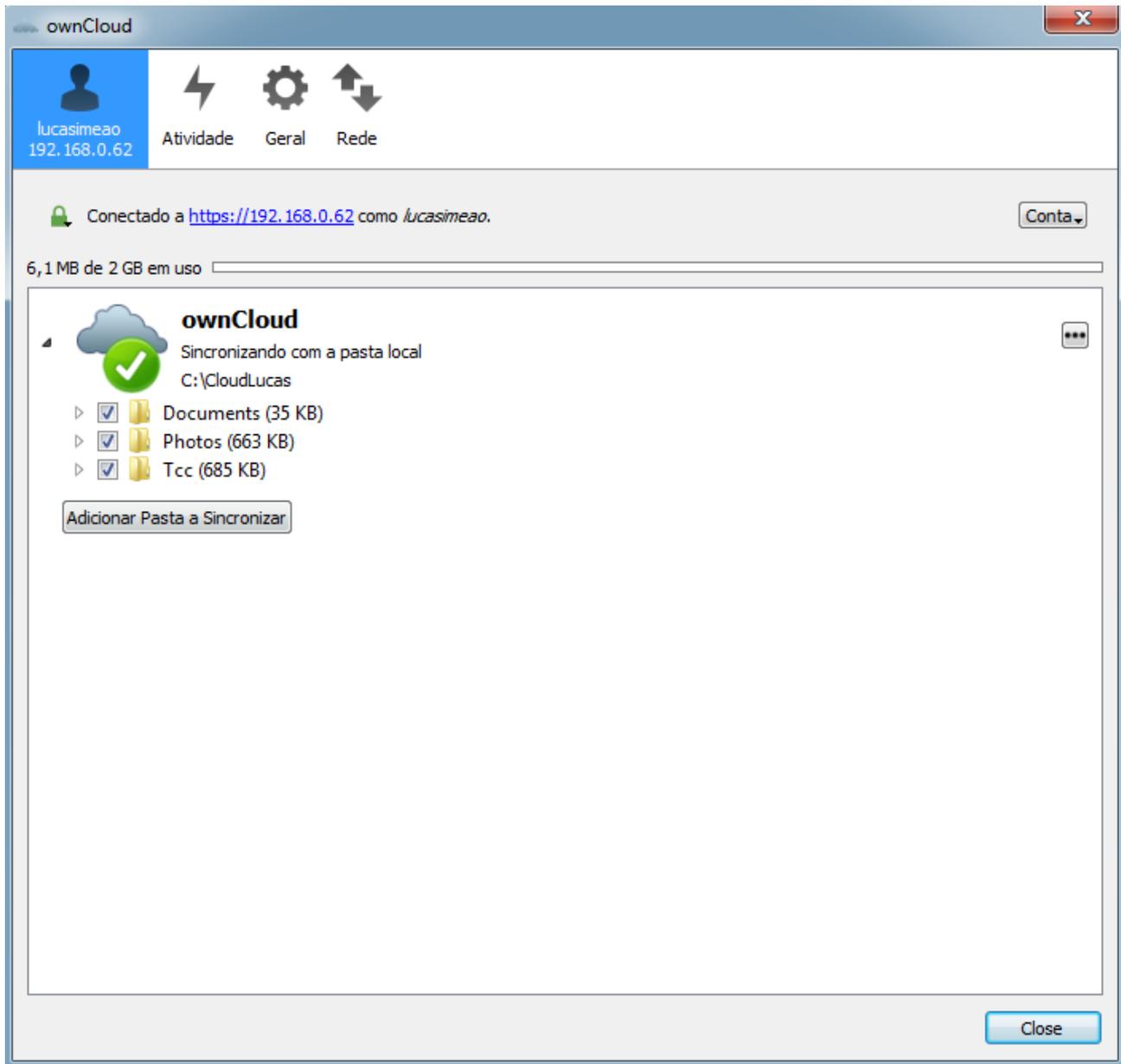


Figura 51 - Tela de exibição de configuração do aplicativo de sincronização 4

Fonte: do autor

4. CONCLUSÃO

O desenvolvimento do presente projeto foi de suma importância para ampliar os conhecimentos do autor sobre tema tão presente na realidade de sua área de estudo.

Logo, com o desenvolvimento final do servidor de armazenamento de dados, há uma quebra de paradigma, pois todas as informações se tornaram mais próximas e seguras, visto que sua disponibilidade se tornou algo real e o risco de perder qualquer que seja a informação reduziu drasticamente.

Contudo, o armazenamento em nuvem é uma tendência que chegou para proporcionar maior conforto as empresas e está cada vez mais dominando o mercado com suas vantagens.

Em relação a segurança, a ferramenta apresenta-se como grande estratégia para evitar a perda de dados, principalmente com os ataques de Ransomware que estão em alta, prejudicando a maioria das empresas no mercado local e regional. Como projeto futuro, pretende-se implementar a ferramenta para replicar seus backups na nuvem e também o uso das técnicas RAID0 voltadas para aumentar o desempenho da aplicação e o RAID1 para manter a segurança e disponibilidade dos mesmos.

5. REFERÊNCIAS

BITTENCOURT, A. A.; MANOLA, R. Segurança em computação em nuvem, Vitória, 2011. 18.

BUYYA, R.; YEO, C.; VNUGOPAL, S. Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities, 2008.

CARR, N. **Big Switch: Rewiring the World, from Edison to Google**. 1ª. ed. New York: W.W. Norton & Company, 2008.

CEARLEY, D.; AL, E. Hype Cycle for Applications Development. **Gartner Group Reporter**. Disponível em: <<http://www.gartner.com>>. Acesso em: 17 Fevereiro 2018.

GILDER, G. Wired Magazine, 2006. Disponível em: <<https://www.wired.com/2006/10/cloudware/>>. Acesso em: 05 Março 2018.

JADEJA, Y.; MODI, K. Cloud Computing - Concepts, Architecture and Challenges, Kumaracoil, 22 Mar 2012. 4. Disponível em: <<https://scihub.tw/10.1109/ICCEET.2012.6203873>>. Acesso em: 14 Jan 2018.

KARLITSCHKEK, F. Owncloud. **A safe home for all your data**, 2017. Disponível em: <<https://owncloud.org>>. Acesso em: 25 Out 2017.

LICKLIDER, J. C. R. Celebrating people who bring the internet to life. **Internet Hall of Fame**, 1962. Disponível em: <<https://www.internethalloffame.org/inductees/jcr-licklider>>. Acesso em: 23 Out 2017.

LOPES, F. R. B.; MARTINS, H. P.; ANDRADE, V. S. **Sistema de Arquivos Distribuídos: Comparativo Entre Ambiente Local e em Nuvem**. Faculdade de Tecnologia de Bauru. Bauru, p. 20. 2016.

MCGRATH, M. P. **Understanding PaaS**. Sebastopol: O'Reilly, 2012.

MOELLER, R. R. **IT Audit, Control, and Security**. Canada: John Wiley & Sons, v. 13, 2010.

PETER, M.; TIMOTHY, G. The NIST Definition of Cloud Computing. **Recommendations of the National Institute of Standards and Technology**. Disponível em:

<<https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>>. Acesso em: 22 Janeiro 2018.

PRESIDÊNCIA DA REPÚBLICA. Casa Civil. **DECRETO Nº8.135, DE 4 DE NOVEMBRO DE 2013**, 2013. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2013/Decreto/D8135.htm>. Acesso em: 20 Out 2017.

REESE, G. **Cloud Application Architectures - Building Applications ans Infrastructure in the Cloud**. Sebastopol: O'Reilly, 2009.

RUSCHEL, H.; ZANOTTO, M. S.; MOTA, W. C. *Computação em Nuvem*, Curitiba, 2010. Disponível em: <<http://www.ppgia.pucpr.br/~jamhour/RSS/TCCRSS08B/Welton%20Costa%20da%20Mota%20-%20Artigo.pdf>>. Acesso em: 20 Janeiro 2018.

SOUSA, F.; MOREIRA, L.; MACHADO, J. *Computação em Nuvem. Conceitos, Tecnologias, Aplicações e Desafios.*, Ceará, 20 Setembro 2009. 150-175. Disponível em: <https://www.researchgate.net/profile/Javam_Machado/publication/237644729_Computacao_em_Nuvem_Conceitos_Tecnologias_Aplicacoes_e_Desafios/links/56044f4308aea25fce3121f3/Computacao-em-Nuvem-Conceitos-Tecnologias-Aplicacoes-e-Desafios.pdf>. Acesso em: 10 Fevereiro 2018.

TAURION, C. **Computação em nuvem**: transformando o mundo da tecnologia da informação. Rio de Janeiro: Brasport, 2009.

UPTON, E.; HALFACREE, G. **Raspberry Pi Manual do Usuário**. São Paulo: Novatec, v. 1, 2013. Disponível em: <<https://s3.novatec.com.br/capitulos/capitulo-9788575223512.pdf>>. Acesso em: 13 Jan 2018.

VAQUERO, L. M. et al. A break in the Clouds: Towards a CLOUD Definition. **ACM**, New York, v. 39, n. 1, p. 50-55, Jan 2009.

VELTE, A. T.; VELTE, T. J.; ELSENPETER, R. **Computação em Nuvem**: Uma Abordagem Prática. Tradução de Gabriela Mei. 1ª. ed. Rio de Janeiro: Alta Books, 2012.

VERAS, M. **Cloud computing**: nova arquitetura da TI. 1. ed. Rio de Janeiro: Brasport, 2012.

