

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA SUL-RIO-
GRANDENSE - IFSUL, CÂMPUS PASSO FUNDO
CURSO DE TECNOLOGIA EM SISTEMAS PARA INTERNET**

FERNANDO JOSÉ SIMPLICIO

**IMPLEMENTAÇÃO DE FIREWALL DE ALTA DISPONIBILIDADE
ATRAVÉS DO PFSENSE**

Orientador: Prof. Me. Lisandro Lemos Machado

**PASSO FUNDO,
2013**

FERNANDO JOSÉ SIMPLICIO

**IMPLEMENTAÇÃO DE FIREWALL DE ALTA DISPONIBILIDADE
ATRAVÉS DO PFSENSE**

Monografia apresentada ao Curso de Tecnologia em Sistemas para Internet do Instituto Federal Sul-Rio-Grandense, *Campus* Passo Fundo, como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet.

Orientador: Prof. Me. Lisandro Lemos Machado

**PASSO FUNDO,
2013**

FERNANDO JOSÉ SIMPLICIO

**IMPLEMENTAÇÃO DE FIREWALL DE ALTA DISPONIBILIDADE
ATRAVÉS DO PFSENSE**

Trabalho de Conclusão de Curso aprovado em __10__ / __12__ / 2013__ como requisito parcial para a obtenção do título de Tecnólogo em Sistemas para Internet

Banca Examinadora:

Prof. Me. Lisandro Lemos Machado (Orientador)

Prof. Me. Carlos Alberto Petry (Convidado)

Prof. Esp. José Antônio Oliveira de Figueiredo (Convidado)

Prof. Dr. Alexandre Tagliari Lazzaretti

Coordenação do Curso

PASSO FUNDO, 2013

*A minha família,
pela compreensão e pelo estímulo
em todos os momentos.*

AGRADECIMENTOS

Agradeço aos meus pais Valdir Alves Simplicio e Edi Terezinha Didoné Simplicio, pelo apoio e pelo caráter que me conduz por esta vida, sempre buscando o lado correto. À minha esposa, Giovana Maria de Oliveira, e às minhas filhas, Daniele de Oliveira Simplicio e Fernanda de Oliveira Simplicio, pela compreensão, paciência e incentivo nas horas mais difíceis e tensas.

Aos professores e funcionários do Instituto Federal Sul-Rio-Grandense (IFSul) Câmpus Passo Fundo que têm grande colaboração nesta etapa da minha vida, especialmente ao Diogo Nelson Rovadosky, Luciano Rodrigo Ferreto e Fernanda Milani integrantes da COTIN, setor de tecnologia da instituição, no qual fui estagiário por dois anos e que tem grande contribuição no êxito da minha formação.

Aos meus colegas, André Luciano Ciota e Vinícius Michelin, do setor de Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS), por permitirem e contribuírem no desenvolvimento e implementação deste trabalho.

Agradecimento especial para o meu professor e orientador Mestre Lisandro Lemos Machado, pela sua dedicação e empenho desde o curso Técnico de Informática na Escola Estadual de Educação Profissional João de César, passando pelo curso Técnico em Sistemas de Informação até chegar ao Curso Superior de Tecnologia em Sistemas para Internet.

Aos meus colegas de aula, com quem vivenciei excelentes momentos ao longo do caminho dentro do IFSul.

“Se você vai tentar, vá até o fim,
caso contrário, nem comece”.

Charles Bukowski

RESUMO

O trabalho Implementação de *Firewall* de Alta Disponibilidade Através do *PFSense* apresenta um estudo a partir dos conceitos de virtualização, redundância, segurança da informação, *firewall* e alta disponibilidade com base em serviços e softwares livres, visando à construção de um *firewall* que implemente o conceito de alta disponibilidade no ambiente do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, Câmpus Erechim. A análise foi realizada com base neste estudo de campo e através de métodos, equipamentos e ferramentas utilizados para implementação. Apresentando, posteriormente, os testes de bloqueios no *firewall* e de alta disponibilidade nos serviços e servidores.

Palavras-chave: *Firewall*; Alta disponibilidade; Segurança da informação; Software livre.

ABSTRACT

The Implementation of Work High Availability Firewall Through pfSense presents a study based on the concepts of virtualization, redundancy, information security, firewall and high availability based on free software and services, aiming to build a firewall that implements the concept of High availability in the Federal Institute of Education, Science and Technology of Rio Grande do Sul, Campus Erechim environment. The analysis was performed based on this field study and by methods, equipment and tools used for implementation. Introducing tests locks on the firewall and high availability services and servers later.

Keywords: Firewall; High Availability; Information Security; Free Software.

LISTA DE TABELAS

Tabela 1 – O Significado dos Noves	17
Tabela 2 – Serviços	29
Tabela 3 – Interfaces do Servidor Mestre.....	39
Tabela 4 – Interfaces do Servidor Escravo	44
Tabela 5 – TESTE 1 - Mestre e Escravo Ligado	48
Tabela 6 – TESTE 2 - Escravo Ligado	50

LISTA DE FIGURAS

Figura 1 – Arquitetura de um Servidor executando o Xen.....	23
Figura 2 – Servidor Power Edge R-710	24
Figura 3 – Virtualização	25
Figura 4 – Tela Principal <i>PFSense</i>	26
Figura 5 – CARP	28
Figura 6 – Ambiente Atual.....	30
Figura 7 – Ambiente construído.....	32
Figura 8 - Arquitetura.....	34
Figura 9 – <i>Aliases</i> IP	36
Figura 10 – Regras	36
Figura 11 – Edição de Regra	37
Figura 12 – Teste UltraSurf.....	38
Figura 13 – Regra Sincronismo	40
Figura 14 – Configuração CARP	40
Figura 15 – Opções de Sincronismo	41
Figura 16 – IP Virtual WAN	42
Figura 17 – IP Virtual LAN	42
Figura 18 – IPs Virtuais Mestre	43
Figura 19 – Status Mestre	43
Figura 20 – Sincronismo Escravo	44
Figura 21 – Configuração CARP Escravo.....	45
Figura 22 – IP Virtual WAN Escravo.....	46
Figura 23 – IP Virtual LAN Escravo	46
Figura 24 – IPs Virtuais	47
Figura 25 – Status	47
Figura 26 – Servidor Mestre em Execução.....	48
Figura 27 - Servidor Escravo em Execução.....	49
Figura 28 – Ping no Servidor Mestre.....	49
Figura 29 - Ping no Servidor Escravo	49
Figura 30 – Ping IP Virtual	49
Figura 31 - Servidor Mestre Desligado.....	50
Figura 32 – Servidor Escravo Ligado	51

Figura 33 – IP Virtual.....	51
Figura 34 – Servidor Mestre sendo desligado	52
Figura 35 - Status Servidor Escravo	52
Figura 36 – Servidor Mestre Religado	53
Figura 37 - Servidor Escravo	53

LISTA DE ABREVIATURAS E SIGLAS

ABNT - Associação Brasileira de Normas Técnicas
BSD - Berkeley Software Distribution
CIA - Confidentiality, Integrity and Availability
CPU - Unidade central de processamento
D - Disponibilidade
DMZ - Demilitarized Zone
DT - Down Time
FTP - File Transfer Protocol
HA - High Availability
HD - Hard Disk
IEC - International Engineering Consortium
ISO - International Organization for Standardization
MAC - Media Access Control
NIC BR - Núcleo de Informação e Coordenação do Ponto BR
NAT - Network Address Translation
OCS NG - Software Inventory Next Generation
RAM - Random Access Memory
RAID - Redundant Array of Inexpensive Disk
TCP - Transmission Control Protocol
TUP - Total de Unidade de Tempo
UDP - User Datagram Protocol
URL - Uniform Resource Locator
VHID - Virtual Host ID
VMM - Monitores de máquina virtual
Web - World Wide Web

SUMÁRIO

1	INTRODUÇÃO	12
1.1	MOTIVAÇÃO	12
1.2	OBJETIVOS	13
1.2.1	Objetivo Geral.....	13
1.2.2	Objetivos Específicos	13
2	SEGURANÇA DE SISTEMAS	14
2.1	SEGURANÇA DA INFORMAÇÃO	14
2.1.1	Gestão de Segurança da Informação	15
2.1.2	Política de Segurança.....	15
2.2	ALTA DISPONIBILIDADE	16
2.2.1	Disponibilidade	17
2.3	TOLERÂNCIA A FALHAS	18
2.3.1	Redundância.....	18
2.3.1.1	Redundância via Hardware	18
2.3.1.2	Redundância via Software.....	19
2.3.1.3	Redundância via Informação.....	19
2.3.1.4	Redundância via Tempo.....	19
2.4	FIREWALL	20
2.4.1	A Escolha do Firewall	20
2.4.2	A Localização do Firewall	21
2.5	VIRTUALIZAÇÃO.....	21
2.5.1	Virtualizador.....	22
2.5.1.1	Xen.....	23
2.5.1.2	Arquitetura Xen	23
3	IMPLEMENTAÇÃO DE UM FIREWALL DE ALTA DISPONIBILIDADE	24
3.1	MÉTODOS, EQUIPAMENTOS E FERRAMENTAS	24
3.1.1	Métodos.....	24

3.1.2	Equipamentos.....	24
3.1.3	Ferramentas	25
3.1.3.1	PFSense.....	25
3.1.3.2	CARP	27
3.1.3.3	PFSync.....	28
3.2	CONFIGURAÇÃO ATUAL	28
3.2.1	Política de Segurança da Instituição.....	29
3.2.2	Ambiente atual.....	29
3.2.2.1	PFSense (Firewall).....	30
3.2.2.2	Bolicho (Samba).....	30
3.2.2.3	Impressão (Cups).....	31
3.2.2.4	Sistemas (Apache).....	31
3.2.2.5	Chasque (GLPI)	31
3.2.2.6	Sentinela (OCS)	31
3.3	FIREWALL DE ALTA DISPONIBILIDADE	32
3.3.1	Arquitetura	33
3.3.2	Serviços	35
3.3.3	Configurações Firewall	35
3.3.3.1	Testes e Resultados Firewall	38
3.3.4	Configurações de alta disponibilidade	39
3.3.4.1	Mestre	39
3.3.4.2	Escravo	44
3.3.4.5	Testes e Resultados.....	47
4	CONSIDERAÇÕES FINAIS	54
	REFERÊNCIAS.....	56

1 INTRODUÇÃO

A área de segurança de informações e disponibilidade em computadores e sistemas vem tendo um amplo crescimento nos últimos anos. Empresas ou instituições, mesmo pequenas, estão fazendo uso de sistemas para gerenciamento e automação de suas atividades, o que se torna um grande desafio para os gestores de tecnologia, pois os serviços e recursos computacionais devem estar o maior tempo possível disponíveis para o uso da empresa.

Outro fator também é a expansão da internet e das redes de computadores que trazem vários problemas de segurança e confiabilidade das informações para estas empresas e instituições que utilizam de tecnologia, através de computadores e dispositivos interconectados em rede, nos seus negócios.

Diante deste cenário, pretende-se elaborar um *firewall* de alta disponibilidade a partir dos conceitos de segurança da informação, *firewall* e alta disponibilidade, bem como, todas as especificações de hardware e software, desenvolvimento e implementação virtual deste *firewall*. Todo o desenvolvimento prático do *firewall* utilizará software livre e sem custos. Tomar-se-á por base um cenário semelhante ao construído e implantado no Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, Câmpus Erechim, onde estão implementados servidores virtualizados e interligados em alta disponibilidade que gerenciam todo o controle de acessos à rede externa e interna.

1.1 MOTIVAÇÃO

Um fator visível para a necessidade de proteção da informação é o aumento de serviços de armazenamento, processamento e recuperação de informações em empresas ou instituições que usam redes computacionais. Outro fator relevante é que, cada vez mais, o homem se torna dependente de tecnologia e conectividade com redes internas e externas, ou seja, redes internas de instituições ou empresas e da internet.

Para garantir o acesso a essas informações e para que não ocorra a interrupção nos acessos dos serviços, que podem causar grandes prejuízos financeiros e de imagem à empresa, uma solução possível é a utilização de um *firewall* de alta disponibilidade com tolerância a falhas.

1.2 OBJETIVOS

Através da implementação de um *firewall* de alta disponibilidade, pretende-se atingir os objetivos a seguir.

1.2.1 Objetivo Geral

Pesquisar e aplicar conceitos sobre segurança da informação, *firewall* e alta disponibilidade em um sistema de *firewall*, o qual será construído utilizando ferramentas de virtualização e softwares livres.

1.2.2 Objetivos Específicos

- Conceituar segurança em redes, alta disponibilidade e *firewall*.
- Estudar softwares e hardwares usados para implementação de um sistema seguro baseado em um *firewall* de alta disponibilidade;
- Implementar um ambiente organizacional protegido por *firewall* de alta disponibilidade;
- Realizar testes de funcionalidades de disponibilidade e tolerância a falhas do sistema implementado.

2 SEGURANÇA DE SISTEMAS

2.1 SEGURANÇA DA INFORMAÇÃO

A segurança da informação é a referência principal que uma empresa ou pessoa tem sobre a proteção de determinadas informações, pessoais ou corporativas, ou seja, todo conteúdo ou dado com valor. Estas informações podem estar guardadas nos mais diversos meios e para os mais diversos usos, podendo ser restritas ou expostas ao público, para a consulta e o download, entre outras operações. Conforme a norma ABNT NBR ISO/IEC 17799:2005.

A informação é um ativo que, como qualquer outro ativo importante, é essencial para os negócios de uma organização e conseqüentemente necessita ser adequadamente protegida. Isto é especialmente importante no ambiente dos negócios, cada vez mais interconectado.

A informação pode existir de diversas maneiras, pode ser falada, impressa, escrita em papel, armazenada ou transmitida por meio digital e independente da forma apresentada, ela deve ser adequadamente protegida. Neste contexto, entra a proteção da informação, visando a garantir a continuidade do negócio, minimizar os riscos e maximizar o retorno e as oportunidades de negócio. Para atingir a segurança da informação, é preciso construir um conjunto de ações que façam o controle de acordo com o nível exigido pela empresa, incluem-se neste conjunto políticas, processos, procedimentos, estruturas organizacionais e funções de software e hardware, os quais devem ser estabelecidos, implementados, analisados e melhorados.

A segurança da informação é de extrema importância para o negócio da empresa ou instituição, sendo considerados como ativos para os negócios os sistemas, os processos e as redes, entre outros. Quando alcançados todos os objetivos de definição, alcance, manutenção e melhorias da segurança, a competitividade e a imagem da empresa tende a estar melhor assegurada perante o mercado, sob o ângulo da confiabilidade.

2.1.1 Gestão de Segurança da Informação

Segundo Lopes Filho (2008, p.19), a política de segurança ou gestão de segurança em uma instituição ou empresa deve respeitar a tríade CIA (*Confidentiality, Integrity and Availability*). A confidencialidade, a integridade e a disponibilidade são a base na implementação da segurança em um determinado conjunto de informações que deve ser protegido, sendo que além destes, a autenticidade e a privacidade também aparecem como fatores importantes e geram grande preocupação para os gestores.

Estes elementos indicados abaixo, segundo o mesmo autor, são definidos segundo os padrões internacionais como:

- **Confidencialidade** – É a garantia de resguardo das informações contra a divulgação não autorizada.
- **Integridade** – Garantia de que a informação não foi alterada de maneira não autorizada, inclui mecanismos de controle de mudanças durante o ciclo de vida da informação, ou seja, nascimento, manutenção e destruição.
- **Disponibilidade** – Informação disponível para o uso legal, com acesso autorizado pelos donos da informação.

Portanto, o sistema de segurança implementado no presente trabalho tem como ponto central para o seu desenvolvimento o requisito disponibilidade, e a direção das etapas desenvolvidas estará voltada para a implementação da disponibilidade ou alta disponibilidade em software e hardware.

2.1.2 Política de Segurança

A definição de Cheswick (2005, p. 26) diz que a política de segurança é um conjunto de regras definidas coletivamente que determinam a postura da empresa ou instituição em relação à segurança das informações, a qual deve deixar claro quais os limites aceitáveis de comportamento e quais medidas podem ser tomadas em desvios.

As políticas de segurança definem o que deve ser preservado, mas não como vai ser preservado, segundo ALBERTIN (2010, p.83) elas podem ser definidas de duas formas:

- **Regras** – Determina o tipo de acesso que pode ser feito;
- **Segurança** – Controla e especifica o que cada usuário pode ler, escrever ou gravar nos recursos da instituição.

2.2 ALTA DISPONIBILIDADE

Além da segurança do sistema, outro conceito importante quando se trata de serviços oferecidos em uma rede de computadores é o de alta disponibilidade, que é a capacidade de um sistema funcionar sem interrupções por um longo período de tempo. Este tempo deve ser superior à fiabilidade dos equipamentos individuais do sistema, em qualquer cenário rotineiro de funcionamento ou em circunstâncias de falhas e erros, sendo o equilíbrio entre as ações de proteção e o custo de inatividade. Segundo Lopes Filho (2008, p.19), em sistemas de informações, pode-se dizer que o serviço proposto ficará sempre disponível independentemente da hora, do dia ou local, a alta disponibilidade tem por seu maior objetivo a continuidade do serviço sem interrupções por longos períodos de tempo. Outro fator importante é a capacidade de proteção e recuperação de interrupções em um curto espaço de tempo, o que faz com que o serviço volte a funcionar o mais rápido possível, com isso o usuário pode até perceber o ocorrido, mas não ficará sem o serviço.

Com o crescimento rápido da internet, a pressão sobre gestores de tecnologia da informação tem crescido em proporção semelhante, pois se o sistema ou o serviço não estiver disponível, o negócio pode ser ameaçado e sofrer com perdas de produtividade, receita e bom relacionamento com o cliente. Alta disponibilidade é uma solução que pode combinar baixo investimento, tanto financeiro como de manutenção (dependendo das ferramentas e recursos utilizados), para qualquer empresa que necessite implantar redes, sistemas e segurança. Ela auxilia na redução da perda de informações e da negação do serviço, tendo como suas principais características a disponibilidade, o sistema tolerante a falhas e a redundância.

2.2.1 Disponibilidade

A disponibilidade é a descrição de um sistema que executa um serviço específico e com configurações estabelecidas. De acordo com Brandão (2013, p.1), em informática, disponibilidade é entendida como o espaço de tempo em que estes serviços estão disponíveis, por exemplo, 12 horas por dia, 5 dias por semana. A disponibilidade pode ser calculada através de uma fórmula simples:

$$D = (TUP - DT) / TUP$$

Onde,

D é a Disponibilidade;

TUP é o Total de Unidade de Tempo

DT é o *DownTime* (tempo parado)

Para exemplificar, considere-se o cálculo da disponibilidade de um equipamento (servidor) em um período de 1 ano, levando em consideração 3 horas de DT (*DownTime*) para manutenção.

$$TUP = 1 \text{ ano ou } 8760 \text{ horas} \quad \text{DowTime} = 3 \text{ horas}$$

Aplicando a fórmula, tem-se o seguinte: $D = (8760 - 3) / 8760 = 99,9658 \%$

O resultado é 99,9658 por cento, neste caso, tem-se uma grande disponibilidade do serviço, sendo que chegar aos 100 % é praticamente impossível por inúmeros fatores como, por exemplo, um HD que possui vida útil de 20.000 horas, que quando chegar ao seu limite de uso deverá ser trocado, podendo causar um *DownTime* no equipamento, caso este não conte com técnicas auxiliares como RAID. Este tempo indisponível do equipamento é medido em porcentagem, então quanto mais noves (9), da direita para a esquerda no resultado da fórmula da disponibilidade, aparecerem no cálculo, menor será o *DownTime*. Conforme Brandão (2013, p.1), isso representa “O significado dos noves” que estão indicados na tabela 1.

Tabela 1 – O Significado dos Noves

%	<i>Downtime</i> (por ano)
100	Nenhum <i>downtime</i>
99,999 (5 noves)	Menos de 5,26 min.
99,99 (4 noves)	De 5,26 a 52 min.
99,9 (3 noves)	De 52 min. a 8 horas e 45 min.
99 (2 noves)	De 8 horas e 45 min. a 87 horas e 56 min
90,0 - 98,9 (1 nove)	87 horas e 56 min a 875 horas 54 min

Fonte: BRANDÃO, 2013.

2.3 TOLERÂNCIA A FALHAS

Essa técnica é utilizada para descobrir, mascarar e tolerar falhas em um sistema computacional, sua aplicação quando feita do modo correto, tende a garantir o bom funcionamento do sistema, deixando o mesmo confiável e disponível, conforme conceito de Avizienis (1967). Este conceito considera duas classes separadas que podem ser aplicadas na tolerância a falhas, o mascaramento, onde a ocorrência de falhas não se manifesta como erros, porque na sua origem é aplicada uma máscara, ou seja, o usuário não percebe a falha – nesta classe geralmente usa-se redundância de hardware e software; a segunda classe envolve a detecção, localização e reconstrução do sistema, onde, na detecção é feito o diagnóstico da situação ocorrida, a localização trata de identificar o porquê e quando ocorreu o erro no sistema, e a reconstrução é o reparo feito no sistema.

Neste projeto, será considerada a classe de mascaramento, pois seu objetivo é a resposta rápida à ocorrência de erro ou falha em um sistema de *firewall*.

2.3.1 Redundância

A palavra redundância em informática, conforme Weber (p.11), é usada para definir um sistema de tolerância a falhas e todas as técnicas usadas em um sistema tolerante a falhas, que de alguma forma usa redundância, segundo o mesmo autor, pode ser de hardware, software, informação ou tempo, conforme segue.

2.3.1.1 Redundância via Hardware

Constitui na replicação de equipamentos e pode ser construída respeitando três maneiras. A primeira delas é a passiva, que é usada no mascaramento de falhas e os elementos redundantes fazem o mascaramento. A ativa é usada na técnica de detecção, localização e recuperação, sendo mais usada em aplicações que suportam um estado errôneo, mesmo que por um breve tempo, ela usa módulos estepes para reconstruir a aplicação. A híbrida combina a ativa e a passiva, garantindo o mascaramento de falhas e um suporte a erros, tornando o sistema menos suscetível a interrupções não planejadas.

2.3.1.2 Redundância via Software

Em software com replicação é uma estratégia errônea e inútil, pois programas iguais irão apresentar erros iguais, portanto não adianta copiá-lo. As formas de redundância em software são a diversidade – onde se tem várias versões do sistema e pode ser aplicada desde o levantamento de requisitos até a fase de testes. Outra forma de redundância em software é a de blocos de recuperação, que consiste em programas ou versões secundárias que só entram em atividade após a detecção de um erro no programa ou sistema primário.

2.3.1.3 Redundância via Informação

A redundância de informação é provida pelo uso de códigos de correção de erros, usada em transferências entre memória e processador e também só em memória, nela a codificação da informação aumenta o número de bits, mas estes bits não aumentam a capacidade de representação das informações do código. Desta forma, a codificação pode ser considerada uma forma de redundância.

2.3.1.4 Redundância via Tempo

Redundância de tempo é a repetição da computação ou do processo em execução. Utilizada em sistemas onde o processador trabalha com ociosidade e quando o tempo não é crítico. Tem redução de custo em hardware, mas aumenta o tempo de execução de um processo.

2.4 FIREWALL

Separando-se a palavra nos termos “fire” e “wall”, tem-se como tradução mais próxima **parede corta-fogo**, ou seja, uma barreira para evitar a propagação de incêndios. Conforme Cheswick (2005), este termo é definido como sendo um filtro de tráfego de rede, podendo ser um dispositivo, arranjo ou software que impõe limite para acessar a rede. Um *firewall* pode ser uma forma de defesa, mas devem ser adotados outros mecanismos e procedimentos para aumentar a segurança da rede a ser protegida. O *firewall* protege uma rede interna de ataques com origem de fora da rede, quando o ataque vem de dentro da rede, ele nada pode fazer, necessitando de outros mecanismos de defesa complementares. Dentro de um ambiente computacional, é um instrumento importantíssimo para a implementação da política de segurança, pois, quando bem configurado, restringe as informações sobre a rede, tornando mais custoso o acesso não autorizado.

2.4.1 A Escolha do Firewall

Inúmeros tipos de *firewall* encontram-se disponíveis no mercado, quase sempre a decisão da escolha está ligada a fatores como recursos, flexibilidade e custo. Mas um ponto crucial é a plataforma operacional onde ele vai ser instalado, geralmente a escolha é feita com os produtos de *firewall* que executem sobre uma plataforma com a qual os administradores da rede tenham conhecimento e experiência, ou seja, quando os administradores usam Linux é aconselhável que seja usado um *firewall* que execute em Linux. As razões para essas recomendações, conforme NIC BR (2003, p.31), são duas: o primeiro fator diz respeito à familiarização com o sistema para que possa configurar o *firewall* de maneira segura, e a segunda é porque os produtos de maneira geral tendem a seguir a filosofia onde executam, por exemplo, plataformas Windows costumam executar com janelas e em modo gráfico e Linux em modo texto e sem interface gráfica.

2.4.2 A Localização do Firewall

A localização do *firewall* normalmente é dependente da política de segurança, pois esta define onde o *firewall* vai atuar, mas como regra geral todo o tráfego de entrada e saída da rede deve passar pelo firewall. Segundo Stallings (2008, p.44), na maioria dos casos, o *firewall* está localizado entre a rede local e a rede externa, mas pode ser necessário o uso de *firewalls* internos, onde este tem o objetivo de isolar e proteger sub-redes umas das outras, para evitar ou conter a propagação de ataques bem-sucedidos.

Outra prática recomendada é manter os servidores externos que proveem serviços como Web, FTP, Correio Eletrônico em uma DMZ (zona desmilitarizada), ou seja, esses servidores ficam em uma pequena rede localizada entre uma rede confiável e uma rede não confiável, com outras palavras, entre a rede local e a internet.

2.5 VIRTUALIZAÇÃO

A ideia de virtualização não é nova, sua origem foi no início de 1970, nesta época cada computador (mainframe) tinha o seu sistema operacional e para que softwares legados pudessem ser executados nestes computadores surgiu então a virtualização. Conforme Carissimi (2008), nos dias atuais com o avanço do poder computacional dos processadores e os sistemas distribuídos fizeram com que a virtualização seja uma opção para os administradores de sistemas e redes, pois na maioria dos casos é preciso manter um grande e heterogêneo conjunto de servidores.

Para compreender virtualização, é preciso entender o que é real e o que é virtual em tecnologia, sendo que o real é o físico, concreto, por outro lado, o virtual é o abstrato ou o simulado. Então, virtualização é definida como um ambiente virtual, pois ela simula um ambiente real, com isso tem-se acesso a sistemas e aplicativos sem a relação de dependência do software com o hardware, visto que pode se ter um sistema operacional completamente diferente, executando de forma virtualizada sobre outro sistema operacional instalado na máquina real.

A virtualização em hardware tem como característica principal a criação de vários sistemas operacionais na mesma máquina, programas específicos como o *VMWare*, *Citrix XenServer*, entre outros, são usados para criarem estas máquinas virtuais, possibilitando que um sistema operacional possa ser instalado em cada uma das máquinas virtuais, ou seja, pode-

se ter vários computadores virtuais dentro de um real. As principais vantagens são resolver problemas de incompatibilidade entre aplicativos e sistemas operacionais, bem como, melhor aproveitar servidores onde o hardware geralmente é mais robusto, com boa capacidade de processamento, armazenamento e memória. Com a virtualização podem ser criadas várias máquinas virtuais, otimizando o espaço de alocação das máquinas, redução na mão de obra técnica e economia em gastos com energia.

Por outro lado, segundo Carissimi (2008), existem também desvantagens na virtualização, as principais são:

- De segurança, pois se tiver uma falha na camada de software que faz a abstração entre os sistemas operacionais e o hardware, todos as máquinas virtuais estarão vulneráveis;
- De gerenciamento, os ambientes virtuais precisam ser implementados, configurados, monitorados e salvos, vários produtos realizam estas tarefas, mas ainda com deficiências e contratempos na implementação da virtualização.
- De desempenho, ainda não existem métodos para medir o desempenho de ambientes virtualizados, não sendo possível saber exatamente quantas máquinas podem ser executadas em determinado hardware sem que haja prejuízo do desempenho.

2.5.1 Virtualizador

É a plataforma que permite o controle da virtualização, pois eles são uma camada de software entre o hardware e a máquina virtual na qual estará instalado o sistema operacional. Também são chamados de monitores de máquina virtual (VMM) são responsáveis por fornecer ao sistema operacional visitante a abstração da máquina virtual e prover todo suporte para o acesso aos dispositivos de hardware.

Para implementação deste trabalho, como referido antes, todas as ferramentas necessárias devem ser livres, então a opção de escolha foi pelo *hipervisor Xen*, conforme exposto a seguir.

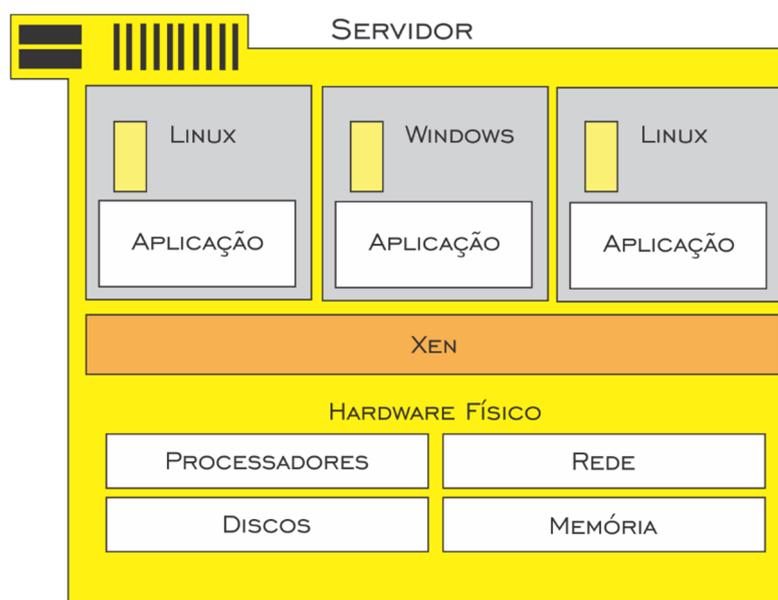
2.5.1.1 Xen

Plataforma de virtualização em software livre, desenvolvido pela Universidade de Cambridge, em meados de 2007, foi adquirido pela empresa *Citrix System Inc* a qual mantém o desenvolvimento do projeto tanto *Open Source*, como também vende uma versão empresarial do software.

2.5.1.2 Arquitetura Xen

A figura 1 mostra um servidor físico rodando o virtualizador *Xen*, fica claro que o *Xen* é a camada entre o hardware e os sistemas operacionais, no exemplo aparece três máquinas virtuais e cada máquina executa um sistema operacional e aplicativos independentes das outras máquinas, mas compartilha os mesmos recursos físicos.

Figura 1 – Arquitetura de um Servidor executando o Xen



Fonte: IBM, 2013.

3 IMPLEMENTAÇÃO DE UM FIREWALL DE ALTA DISPONIBILIDADE

3.1 MÉTODOS, EQUIPAMENTOS E FERRAMENTAS

A seguir serão descritos os métodos, equipamentos e ferramentas utilizados para a realização da pesquisa, com a implementação do *firewall* de alta disponibilidade, também as técnicas de captação, análise e interpretação das informações coletadas.

3.1.1 Métodos

Este estudo foi desenvolvido por meio de uma pesquisa bibliográfica sobre estudos de casos de uso e posterior elaboração e implementação em ambiente real através da virtualização, sempre abordando e respeitando o conceito de alta disponibilidade.

3.1.2 Equipamentos

Para a implementação do firewall foi usado um servidor *Dell PowerEdge R710* (figura 2), contendo 16 GB de memória RAM e podendo ser expandida até 192 GB, processado pelo CPU *E5645@2.40GHz Intel (R) Xeon (R) Quad-core*, com capacidade de armazenamento de cada disco rígido de 300 GB cada. O servidor possui dois destes trabalhando em *RAID 1*, também conhecido como espelhamento, onde é feita uma cópia das informações do primeiro disco para o segundo, ou seja, faz com que o sistema grave os dados ao mesmo tempo nos dois discos, este servidor foi projetado para o uso da tecnologia de virtualização (figura 3), com ênfase no desempenho e escalabilidade.

Figura 2 – Servidor Power Edge R-710



Fonte: DELL, 2013.

Figura 3 – Virtualização



Fonte: DELL, 2013.

3.1.3 Ferramentas

Para a escolha dos softwares, foi respeitado aqueles que atendem aos padrões de virtualização e que sejam distribuídos com código *open source* ou gratuitos.

3.1.3.1 PFSense

PfSense é uma distribuição linux, licenciada sob BSD *licence*, que tem por base o sistema operacional FreeBSD, tendo sido adaptado para assumir o papel de um *firewall* e/ou roteador de rede. Concebido em meados de setembro de 2004, por Cris Buechler e Scott Ullrich, foi construído com base no projeto *m0n0wall*¹, o que o torna uma distribuição de *firewall* poderoso e leve.

Conforme Williamsom e Persaud (2012), em seu nível mais básico, uma máquina desktop simples, cujo sistema operacional seja *PfSense*, pode ser usada para substituir um roteador doméstico e/ou comercial com a funcionalidade que deseja. Em configurações mais avançadas, o *PfSense* pode ser usado para estabelecer um túnel seguro para um escritório remoto ou para equilibrar a carga de tráfego em uma rede, existindo centenas de formas de se configurar um *firewall PfSense*.

A utilização de uma máquina com sistema operacional *PfSense* visa manter o foco na segurança e administração da rede, atuando ativamente como um *firewall* e roteador. Além

¹ O *m0n0wall* tem basicamente as mesmas pretensões técnicas do *pfSense*, mas desde o seu surgimento até os dias de hoje, é focado em appliances, ou seja, equipamentos específicos para desempenhar uma determinada função.

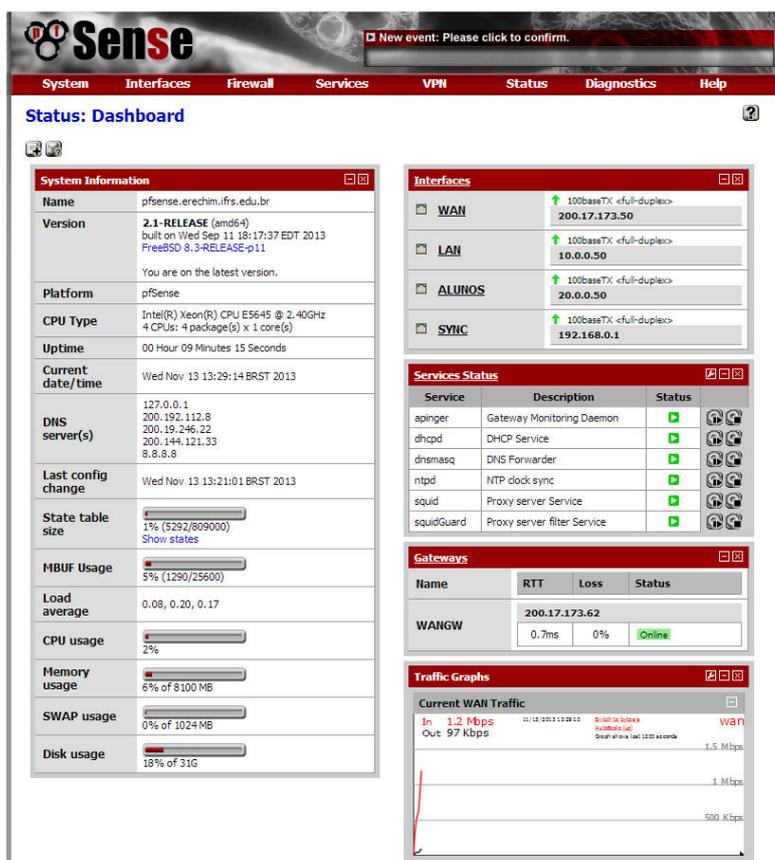
disso, a ferramenta é flexível, pois inclui uma longa lista de recursos relacionados e um sistema de pacotes, permitindo a expansão de serviços sem acrescentar novas vulnerabilidades para o sistema. Outra facilidade do *PFsense* é o backup das configurações, o que torna a administração muito mais fácil e segura, isso quando o backup das configurações é feito regularmente.

De acordo com a configuração e/ou as necessidades, a ferramenta pode assumir o papel de:

- *Firewall*;
- LAN ou WAN *Router*;
- Ponto de Acesso Wireless;
- Serviços específicos, tais como VPN *Appliance*, *sniffer Appliance*, Servidor DHCP *Appliance*, Servidor DNS *Appliance*.

A figura 4 apresenta a tela principal do sistema *PFsense*, onde aparecem as informações do sistema, interfaces, status dos serviços sendo executados, *gateways* e gráficos das redes.

Figura 4 – Tela Principal *PFsense*



Fonte: do autor

Nesse trabalho, serão abordadas as definições básicas para implementação de um *firewall* através do *PfSense* em uma rede local.

3.1.3.2 CARP

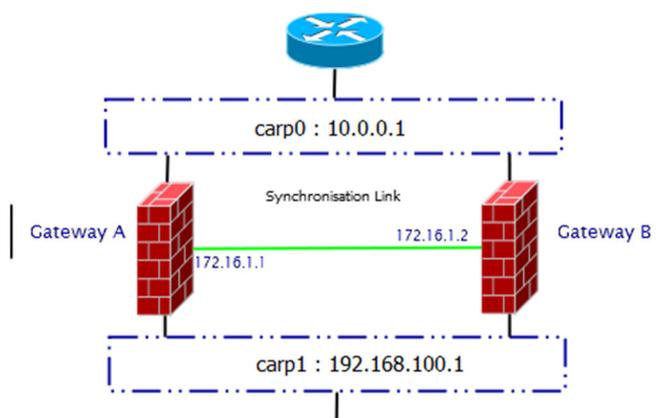
O Protocolo de Redundância de Endereço Comum ou Common Address Redundancy Protocol, com suporte para IPV6 ou IPV4, conforme o *Manual do FreeBSD* (cap. 30), é o software responsável pelo sincronismo dos servidores conectados em alta disponibilidade. Esta ferramenta destina-se a prover a redundância do sistema, com vários computadores compartilhando uma única interface virtual de rede entre eles, garantindo o sincronismo de estado para que em grande parte do tempo os serviços estejam sendo disponibilizados na rede, ou seja, os servidores estejam ativos.

A tecnologia tem por base um endereço IP que fica fluando entre os computadores que fazem parte do grupo. Um integrante do grupo é eleito o mestre e fica respondendo todos os pacotes destinados ao grupo, os outros ficam aguardando para serem substitutos automáticos do mestre, não importando o endereço IP ou endereço MAC da interface física.

O mestre usa a porta IP para comunicar a sua situação ao grupo e a sua frequência é zero (0), sendo esse o valor de maior prioridade. Caso estiver indisponível por um determinado período de tempo, todos do grupo são avisados e a máquina que está programada com um valor de frequência menor e acima de zero torna-se o novo mestre. Quando o mestre volta a ser ativo na rede, tem-se duas alternativas, a padrão é que ele se torne uma máquina secundária, a outra é ele se tornar novamente o mestre, retomando todas as funcionalidades, desta forma, os clientes da rede não são afetados e os serviços continuam disponíveis.

Na figura 5, uma configuração de amostra, aparece um gateway com redundância, o qual faz a separação da rede externa da rede interna. Uma placa de rede dos gateways está com o IP 10.0.0.1 e com o nome de “carp0” e faz a ligação dos gateways com a rede externa, a segunda placa de rede dos gateways com o IP 192.168.100.1 e de nome “carp1”, sendo responsável pela comunicação com a rede interna. A terceira placa de rede dos gateways com o IP 172.16.1.1/172.16.1.2 faz as consultas e a sincronização entre os gateways.

Figura 5 – CARP



Fonte: NOMOA

3.1.3.3 PFSync

PFSync é uma interface de rede usada em máquinas para compartilhar e atualizar o estado, ou seja, faz a certificação de que um *firewall* redundante tenha as mesmas configurações do *firewall* principal, sincronizando dois ou mais *firewalls* que utilizam esta interface, ficando apto a aceitar conexões sem perda mesmo com o *firewall* principal desligado. Deve ser executado em uma rede confiável, pois a mesma não faz autenticação do protocolo e de preferência com um cabo *crossover* entre os dois *firewalls*, conforme *Manual do FreeBSD*.

3.2 CONFIGURAÇÃO ATUAL

A configuração atual não dispõe em nenhum nível de redundância e alta disponibilidade, então qualquer parada no servidor de *firewall* para manutenção ou alguma falha técnica ocorrer, acarretará na indisponibilização da rede, prejudicando o andamento das aulas, e da instituição como um todo.

3.2.1 Política de Segurança da Instituição

O Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, através da sua direção e juntamente com o setor de tecnologia da informação, definiram uma política de segurança apropriada para a instituição, respeitando os princípios de confidencialidade, integridade, disponibilidade, autenticidade, não repúdio, responsabilidade, ética, legalidade, proporcionalidade e conhecimento.

A Política de Segurança da Informação e Comunicações do IFRS é uma declaração formal da organização acerca do seu compromisso com a proteção das informações de sua propriedade e/ou sob sua guarda, devendo ser cumprida por todos os servidores, colaboradores, consultores externos, estagiários e prestadores de serviço que exerçam atividades no âmbito do IFRS ou quem quer que tenha acesso a dados ou informações no ambiente da Organização. O seu propósito é estabelecer diretrizes, normas, procedimentos, e responsabilidades adequadas para o manuseio, tratamento, controle e proteção das informações pertinentes à Organização.

3.2.2 Ambiente atual

Na atualidade, o *firewall* está instalado na borda da rede e é responsável direto por todas as conexões externas e internas da instituição. A configuração atual conta com um servidor físico *Dell R710*, o qual foi virtualizado através do software *Xen (version 6.2)*, o qual executa várias máquinas virtuais e cada máquina realiza um serviço específico. Os principais serviços no servidor estão apresentados na tabela 2.

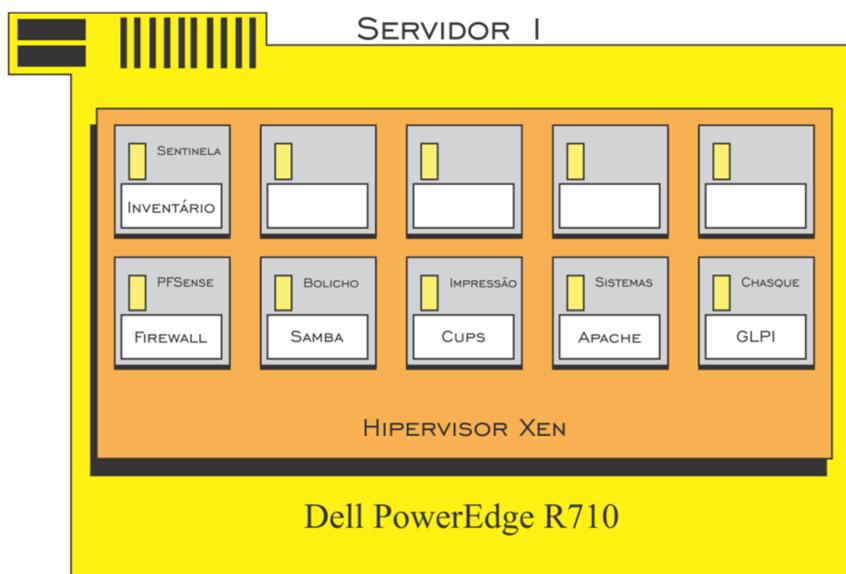
Tabela 2 – Serviços

Nome	Serviço	Descrição
PFsense	Firewall	Firewall/DHCP/Proxy/NTP/DNS
Bolicho	Samba	Servidor de arquivos
Impressão	Cups	Servidor de Impressão
Sistemas	Apache	Sistemas de Inscrição, Patrimônio e Recursos Humanos.
Chasque	GLPI	Sistema de atendimento (help-desk)
Sentinela	OCS	Inventário e monitoramento de máquinas

Fonte: do autor

Esta configuração fica mais clara na figura 6, onde se observa todos os servidores e serviços sobre um único hardware, não provendo nenhum sistema de redundância e nem alta disponibilidade, estando sujeito a qualquer momento à ocorrência de uma interrupção em serviços disponibilizados na rede, gerando assim problemas técnicos e insatisfação dos usuários.

Figura 6 – Ambiente Atual



Fonte: do autor

3.2.2.1 PFSense (Firewall)

O *firewall* implementado na instituição com o sistema *PFSense (FreeBSD)* segue o padrão da maioria dos *firewalls*, que é de liberação total, ou seja, aceita todos os pacotes e somente bloqueia o que for definido na política de segurança como proibido ou potencialmente perigoso. No momento, esta filosofia supre as necessidades da instituição, então será seguida a mesma para a construção do *firewall* em alta disponibilidade.

3.2.2.2 Bolicho (Samba)

Máquina executando distribuição Debian 6.0, que armazena e disponibiliza na rede arquivos, através do software samba, de diversos setores da instituição. O acesso aos documentos compartilhados ocorre mediante usuário e senha, este servidor dispõe de

armazenamento com redundância, sendo executados backups semanalmente em discos rígidos externos.

3.2.2.3 Impressão (Cups)

Servidor de impressão que usa o software Cups e está instalado em uma distribuição Debian 6.0, concentrando todos os acessos às impressoras neste servidor; executa também um sistema que faz o controle (computador + usuário) de quem está enviando o arquivo para a impressão, quantidade de cópias, hora e data, para, se necessário, futura auditoria e responsabilização.

3.2.2.4 Sistemas (Apache)

Máquina executando uma distribuição Debian 6.0 e também o servidor web Apache. Nesta máquina estão hospedados os scripts de sistemas usados internamente na instituição, como o Sistema para Controle de Recursos Humanos (COREH) e o Sistema de Controle Patrimonial (SICOP).

3.2.2.5 Chasque (GLPI)

Neste servidor, está instalado o sistema de atendimento (Help Desk), para isso é usado o software GLPI (Gestão Livre de Parque de Informática), uma solução web Open-source completa para gestão e suporte ao usuário, instalado em uma distribuição Debian 6.0.

3.2.2.6 Sentinela (OCS)

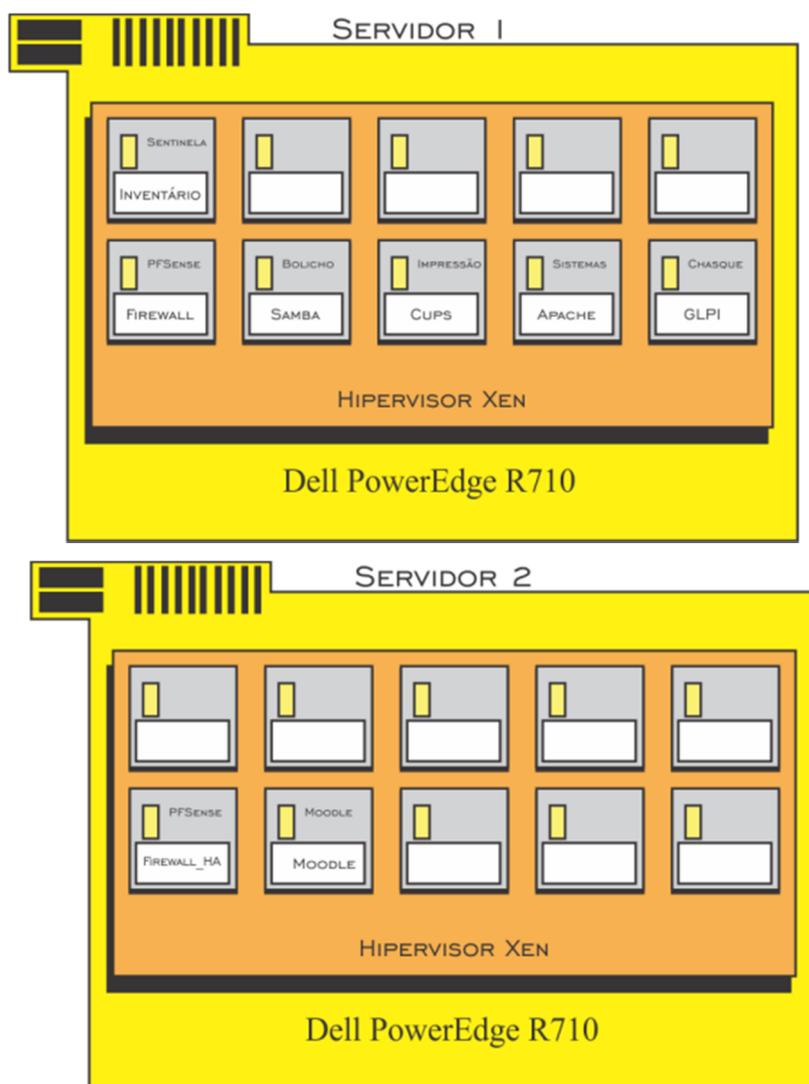
Executando uma distribuição Debian 6.0 e executando o software OCS *Inventory* NG, este usado para gerenciar as máquinas do parque computacional e fazer o inventário automatizado de todo o hardware e softwares das máquinas da rede que possuam um cliente OCS instalado.

3.3 FIREWALL DE ALTA DISPONIBILIDADE

O ambiente proposto neste trabalho deve respeitar a alta disponibilidade e a redundância na construção de um firewall, sendo assim, a figuras 7 apresenta os ambientes. Nela, é possível identificar dois servidores físicos (Dell R710), virtualizados com o *Xen Server version 6.2* e com a possibilidade de criação de várias máquinas virtuais.

O servidor do ambiente atual não vai ser modificado, permanecendo os servidores e serviços, somente foi agregado mais uma máquina *Pfsense* no servidor 2 (*PFSense2*) para que seja possível um *firewall* de alta disponibilidade, neste servidor, também é possível criar outras máquinas necessárias para suprir outras necessidades.

Figura 7 – Ambiente construído



Fonte: do autor

Para a construção do *firewall* em alta disponibilidade precisa-se de duas máquinas virtualizadas onde devem ser instalados e configurados o sistema de *firewall FreeBSD PFsense 2.1-Release* ou versão mais nova. Em cada servidor deve ter obrigatoriamente três placas de rede no mínimo e uma estação de trabalho para realizar os testes. Os principais objetivos que se pretende atingir são:

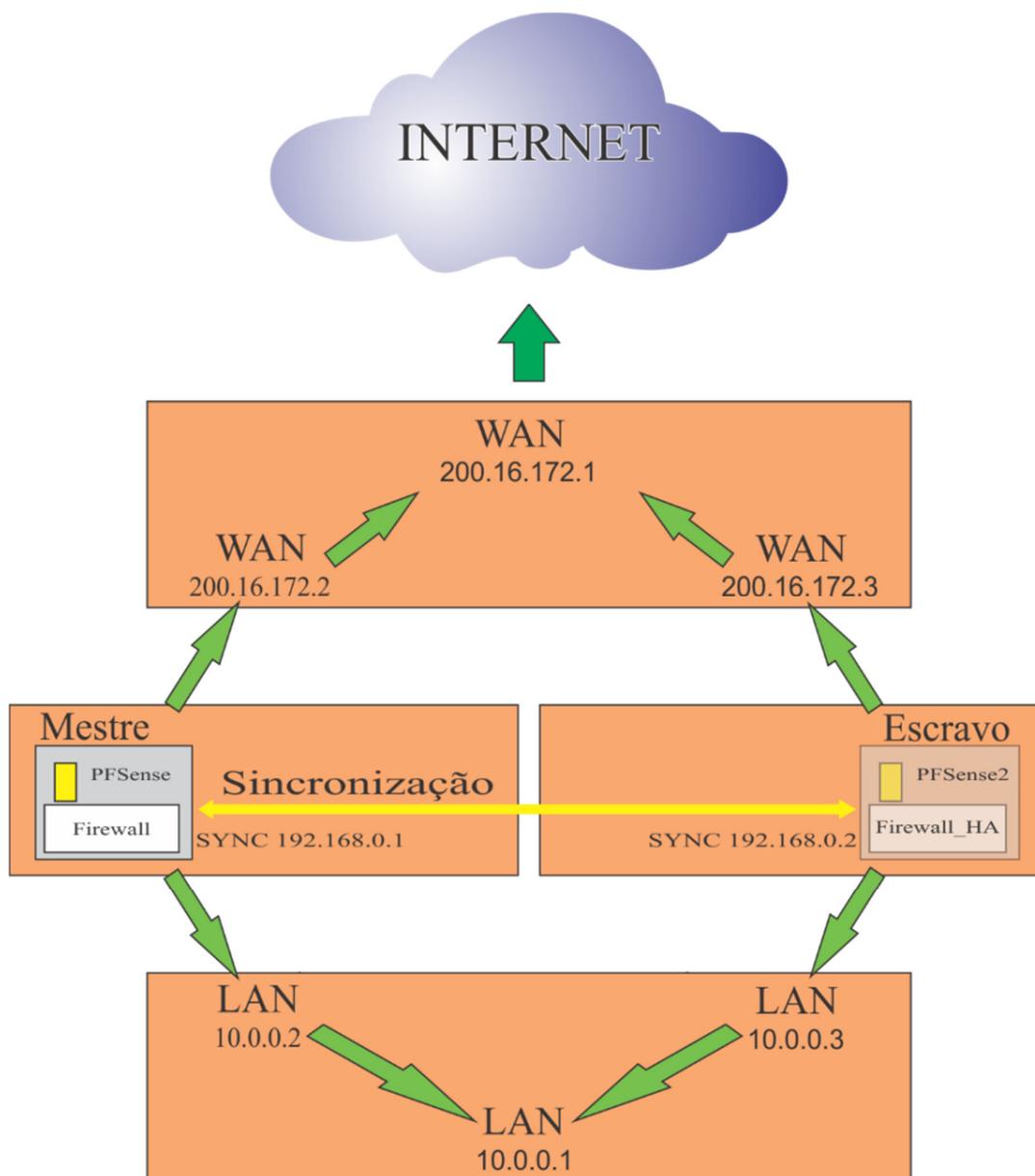
- Quando a máquina principal (mestre), por algum motivo for desligada, a máquina secundária (escravo) deve assumir todas as funcionalidades.
- Quando a máquina principal (mestre) voltar, ela deve retomar suas atividades normalmente.
- O usuário não deve perceber esta mudança.

3.3.1 Arquitetura

Cada máquina virtual construída para o *firewall* no *Xen Server* possui um disco rígido com 30 Gb, 2 núcleos de Processamento (CPU), memória de 1 Gb e três placas de rede.

A figura 8 demonstra a arquitetura do *firewall* com alta disponibilidade, nota-se que cada *firewall* possui três placas de rede, sendo uma para o acesso externo (WAN), a segunda para o acesso interno (LAN) e a terceira para sincronização (SYNC), esta deve ser conectada através de um cabo *crossover*, fazendo a conexão entre os dois *firewalls* diretamente um ao outro, ou seja, uma conexão ponto a ponto.

Figura 8 - Arquitetura



Fonte: do autor

3.3.2 Serviços

Os principais serviços instalados em uso em cada o *firewall* são:

- NTP - sincronismo de horário em computadores.
- NAT - Tradutor dos endereços IP e portas TCP da rede local para a Internet.
- DHCP - Software que distribui IPs na rede pode ser automático, dinâmico ou estático, por motivos de segurança vai ser usado em modo estático.
- DNS – Sistema de tradução de endereços IPs para nomes de domínios.
- Gráficos RRD - Visualização de informações sobre a utilização da CPU, entrada e saída das placas de rede, pode ser visto o histórico ou em tempo real.
- *Squid* - Servidor *Proxy* que suporta HTTP, HTTPS, FTP e outros. Melhora os tempos de resposta fazendo cache de requisições frequentes de páginas web numa rede de computadores.
- *SquidGuard* - Redirecionador de URLs usado para a utilização de *blacklists* com o proxy *Squid*.
- *Lightsquid* - Analisador de logs do *Squid*, mantém os logs diários dos acessos dos usuários a internet.

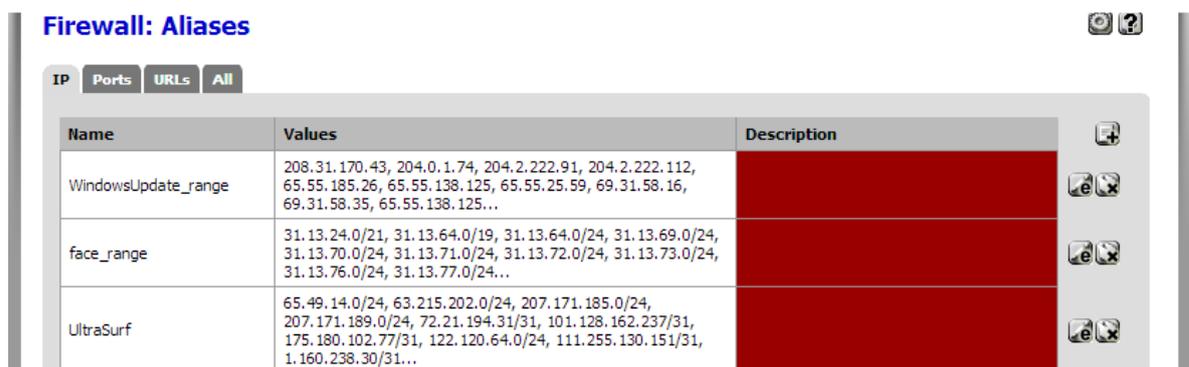
Muitos outros serviços, como VPN, *Captive Portal*, etc..., podem ser instalados e configurados no *firewall PFSense*. Cada instituição ou empresa deve analisar e definir o que de útil deve ser instalado e configurado para o seu firewall.

3.3.3 Configurações Firewall

Na aba ***Firewall*** do *PFSense* concentra-se todas as configurações de filtragem por endereço IP de origem e destino, protocolo IP, porta de origem e destino para tráfego TCP e UDP, sendo capaz de limitar conexões simultâneas para cada regra.

O local para configurar os IPs, Portas e URLs que serão bloqueados, rejeitados ou aceitos fica em ***Firewall*** → ***Aliases***. Na figura 9 há uma amostra de faixas de IPs que estão configurados para os sites *Windows Update*, *FaceBook* e o programa *Ultrasurf*. Para portas e URLs não foram criadas *Aliases*.

Figura 9 – Aliases IP

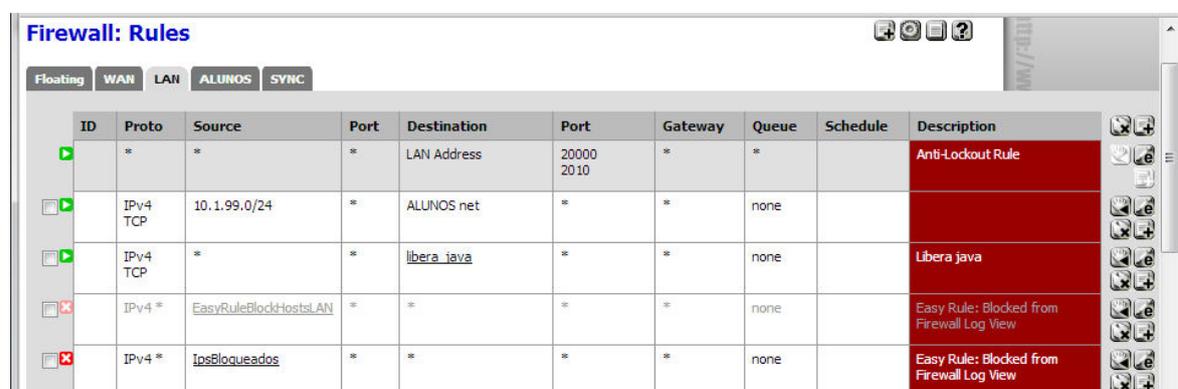


Name	Values	Description
WindowsUpdate_range	208.31.170.43, 204.0.1.74, 204.2.222.91, 204.2.222.112, 65.55.185.26, 65.55.138.125, 65.55.25.59, 69.31.58.16, 69.31.58.35, 65.55.138.125...	
face_range	31.13.24.0/21, 31.13.64.0/19, 31.13.64.0/24, 31.13.69.0/24, 31.13.70.0/24, 31.13.71.0/24, 31.13.72.0/24, 31.13.73.0/24, 31.13.76.0/24, 31.13.77.0/24...	
UltraSurf	65.49.14.0/24, 63.215.202.0/24, 207.171.185.0/24, 207.171.189.0/24, 72.21.194.31/31, 101.128.162.237/31, 175.180.102.77/31, 122.120.64.0/24, 111.255.130.151/31, 1.160.238.30/31...	

Fonte: do autor

Em **Firewall** → **Rules** configura-se as regras, elas podem ser para permitir (seta verde), rejeitar (“x” amarelo) ou ainda bloquear (“x” vermelho) o acesso à rede. Esta aba é visível na figura 10 que mostra para qual interface a regra vai ser aplicada, sendo possível a criação de novas regras ou a edição de regras já construídas, vale lembrar que aqui a ordem na posição da lista importa, ou seja, se tiver uma regra liberando um determinado site e logo após outra regra bloqueando o mesmo site, o que valerá será a primeira, liberando o acesso ao site.

Figura 10 – Regras



ID	Proto	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description
	*	*	*	LAN Address	20000 2010	*	*		Anti-Lockout Rule
	IPv4 TCP	10.1.99.0/24	*	ALUNOS net	*	*	none		
	IPv4 TCP	*	*	libera_java	*	*	none		Libera java
	IPv4 *	EasyRuleBlockHostsLAN	*	*	*	*	none		Easy Rule: Blocked from Firewall Log View
	IPv4 *	IpsBloqueados	*	*	*	*	none		Easy Rule: Blocked from Firewall Log View

Fonte: do autor

Para a criação de uma regra, basta clicar no ícone com o sinal de adicionar (+); já para editar, deve-se clicar no ícone com a letra “e”. A figura 11 mostra a edição da regra que bloqueia o acesso do programa “*Ultrasurf*”, os principais campos são:

- **Action:** pode assumir um dos três valores, Block (bloquear), Reject(rejeitar) ou Pass(passar), no caso está sendo usado o Block, bloqueando o acesso.
- **Disable:** habilitar ou desabilitar a regra.

- **Interface:** seleciona à qual interface de rede vai ser aplicada a regra, no caso a interface LAN.
- **TCP/IP Version:** seleção da versão do protocolo de internet, IPv4 ou IPv6.
- **Protocol:** a escolha de qual protocolo a regra vai corresponder, no caso foi configurado como todos (*any*).
- **Source:** quando selecionado, inverte o sentido da regra na origem, para determinada rede e determinado endereço IP.
- **Destination:** quando selecionado, troca o sentido da regra para o destino, determinando o tipo e o endereço; no exemplo, o tipo assume um *alias* e o com o nome *UltraSurf*, portanto todos os IPs definidos lá no *Aliases* e vistos na figura 9 vão estar passivos desta regra.
- **Log:** ativação de logs para a regra, não é aconselhável ativar, pois o limite de espaço de log local é pequeno.
- **Description:** a descrição da regra, para ficar mais fácil a interpretação na tela de regras.

Figura 11 – Edição de Regra

The screenshot shows the 'Edit Firewall rule' configuration page in pfSense. The page has a navigation bar with 'System', 'Interfaces', 'Firewall', 'Services', 'VPN', 'Status', 'Diagnostics', and 'Help'. The main content area is titled 'Firewall: Rules: Edit' and contains the following configuration fields:

- Action:** Block (dropdown menu). Hint: Choose what to do with packets that match the criteria specified below. Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.
- Disabled:** Disable this rule. Set this option to disable this rule without removing it from the list.
- Interface:** LAN (dropdown menu). Choose on which interface packets must come in to match this rule.
- TCP/IP Version:** IPv4 (dropdown menu). Select the Internet Protocol version this rule applies to.
- Protocol:** any (dropdown menu). Choose which IP protocol this rule should match. Hint: in most cases, you should specify TCP here.
- Source:** not. Use this option to invert the sense of the match. Type: LAN subnet (dropdown menu). Address: / (dropdown menu).
- Destination:** not. Use this option to invert the sense of the match. Type: Single host or alias (dropdown menu). Address: UltraSurf (dropdown menu).
- Log:** Log packets that are handled by this rule. Hint: the firewall has limited local log space. Don't turn on logging for everything. If you want to do a lot of logging, consider using a remote syslog server (see the Diagnostics: System logs: Settings page).
- Description:** BloqUltraSurf (text input field). You may enter a description here for your reference.

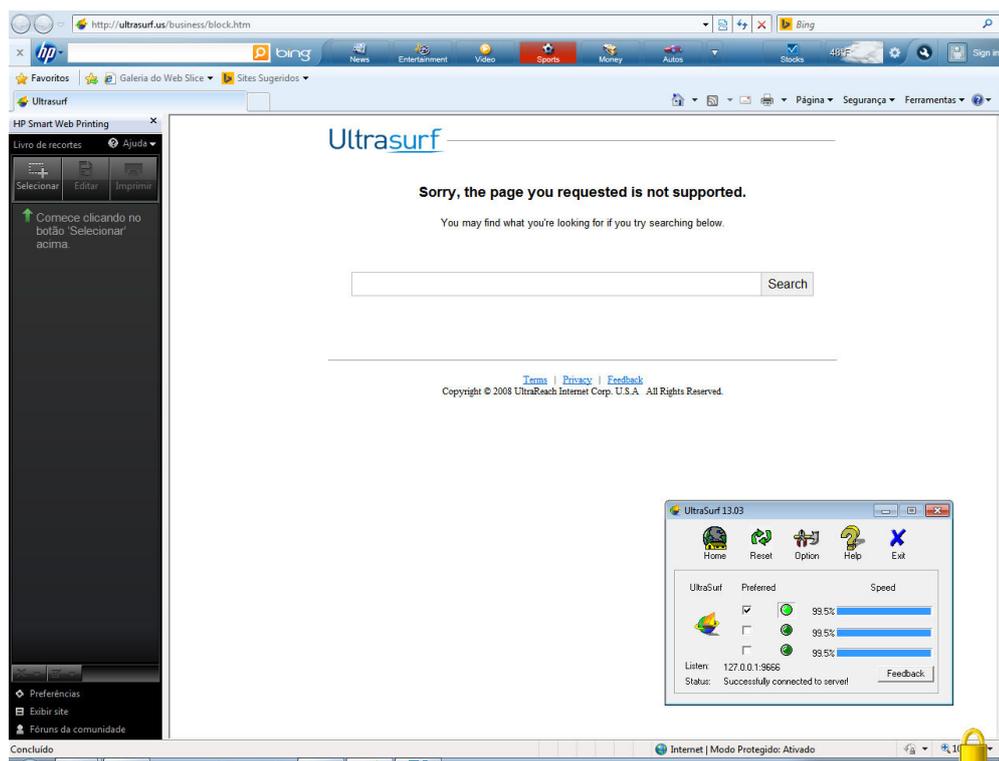
Fonte: do autor

No *Firewall*, se necessário, ainda é possível configurar os redirecionamentos na tabela NAT, configurar regras com definição de dias e horários para o acesso ou bloqueio, limitar o tráfego da rede e configurar os IPs Virtuais, este último será explicado na seção seguinte.

3.3.3.1 Testes e Resultados Firewall

Para testar a regra de bloqueio ao programa *UltraSurf* foi usado um computador alocado na sala de professores do curso de Vestuário, o programa foi executado e o resultado pode ser visto na figura 12. Com o acesso negado, o programa não conseguiu abrir nem a própria página, redirecionando para uma página de erro do programa. No futuro, o setor de tecnologia da informação da instituição pretende criar uma página de erro personalizada.

Figura 12 – Teste UltraSurf



Fonte: do autor

3.3.4 Configurações de alta disponibilidade

As configurações devem ser realizadas nas duas máquinas, tanto no mestre quanto no escravo, que farão parte do *firewall*, neste trabalho foi definido os nomes como “Firewall” e “*Firewall_HA*”, sendo mestre e escravo, respectivamente.

O protocolo redundante CARP e a interface de sincronismo *PFSync*, por padrão, vêm com o sistema *PFSense*, não sendo necessária nenhuma instalação adicional de pacotes.

A interface de rede Alunos aparece em várias figuras, esta interface foi desconsiderada no trabalho, pois as configurações são iguais a da interface de rede LAN, mudando apenas as regras de acesso no *firewall* e a faixa de IPs da rede, não agregando nada de novo.

3.3.4.1 Mestre

Este servidor dispõe de três interfaces de rede, elas estão configuradas conforme a tabela 3.

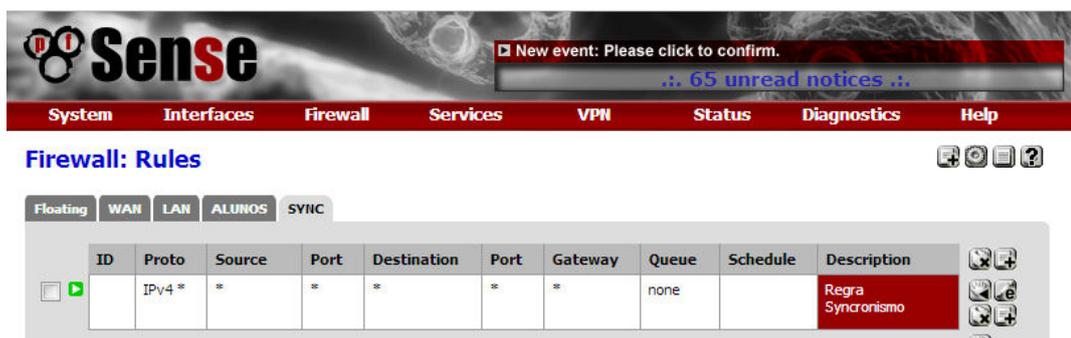
Tabela 3 – Interfaces do Servidor Mestre

Nome	IP	Função
WAN	200.16.172.2	Acesso externo
LAN	10.0.0.2	Endereço da máquina na rede interna
SYNC	192.168.0.1	Sincronismo entre os servidores

Fonte: do autor

O passo seguinte é configurar uma regra no *firewall* para que a interface de sincronismo possa ir de qualquer lugar de origem para qualquer lugar de destino, para fazer isso deve-se acessar a interface do *PFSense* e na aba **Firewall** → **Rules** → **SYNC**. Na figura 13, é possível perceber que esta regra permite acesso a todos os protocolos, portas e *gateway* da rede.

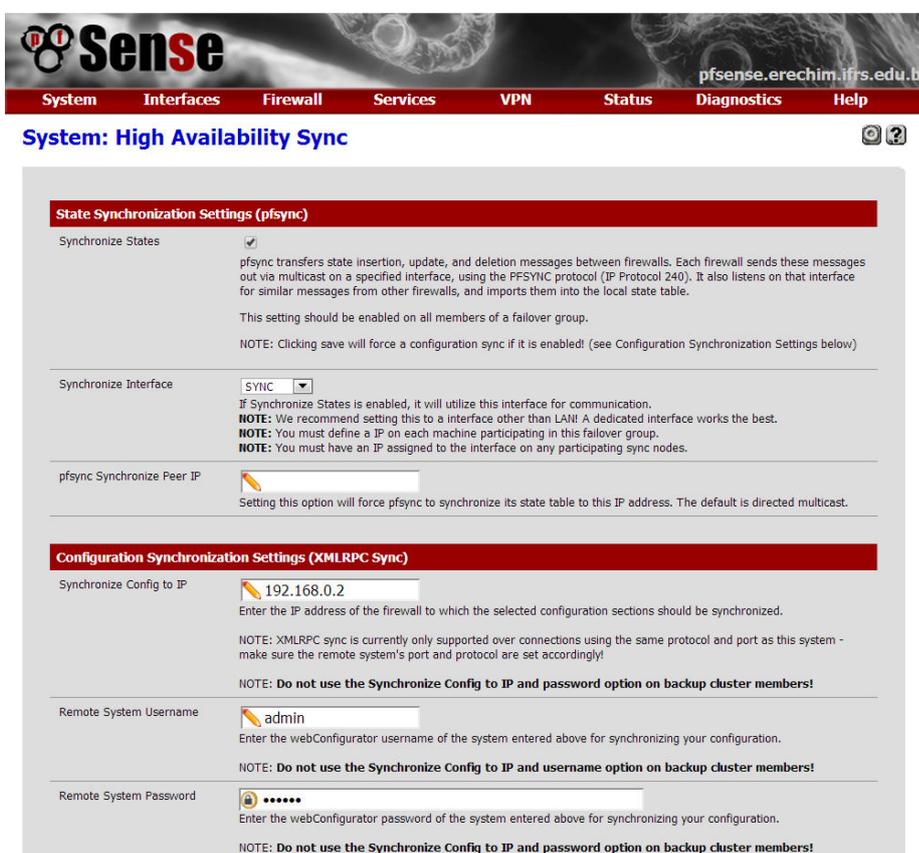
Figura 13 – Regra Sincronismo



Fonte: do autor

Para configurar o CARP, figura 14, acessa-se **Firewall** → **Virtual IPs** → **CARP Settings**, e para ativar a regra de sincronização deve ser marcada a opção de sincronismo de estados. Com essa opção ativa, tudo o que for alterado no *firewall* mestre, será reproduzido no *firewall* escravo (inserção, atualização e exclusão). É definido também qual interface, IP, usuário e senha para o sincronismo; no caso em estudo, a interface definida é a SYNC e o IP é 192.168.0.2, apontando para a interface SYNC do servidor escravo (*Firewall_HA*), nesta mesma tela também é configurado tudo o que se quer e pode ser sincronizado no *PFSense*.

Figura 14 – Configuração CARP



Fonte: do autor

São várias opções de sincronização, na figura 15 é possível ver as opções selecionadas, são elas, *Rules*, *Schedules*, *Aliases*, NAT, DHCP, Virtual IPs e DNS. No momento, estas sincronizações atendem às necessidades do *firewall* na instituição.

Figura 15 – Opções de Sincronismo

Synchronize rules	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the firewall rules to the other HA host when changes are made.
Synchronize Firewall Schedules	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the firewall schedules to the other HA host when changes are made.
Synchronize aliases	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the aliases over to the other HA host when changes are made.
Synchronize NAT	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the NAT rules over to the other HA host when changes are made.
Synchronize IPsec	<input type="checkbox"/>	When this option is enabled, this system will automatically sync the IPsec configuration to the other HA host when changes are made.
Synchronize OpenVPN	<input type="checkbox"/>	When this option is enabled, this system will automatically sync the OpenVPN configuration to the other HA host when changes are made. Using this option implies "Synchronize Certificates" as they are required for OpenVPN.
Synchronize DHCPD	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the DHCP Server settings over to the other HA host when changes are made. This only applies to DHCP for IPv4.
Synchronize Wake on LAN	<input type="checkbox"/>	When this option is enabled, this system will automatically sync the WoL configuration to the other HA host when changes are made.
Synchronize Static Routes	<input type="checkbox"/>	When this option is enabled, this system will automatically sync the Static Route configuration to the other HA host when changes are made.
Synchronize Load Balancer	<input type="checkbox"/>	When this option is enabled, this system will automatically sync the Load Balancer configuration to the other HA host when changes are made.
Synchronize Virtual IPs	<input checked="" type="checkbox"/>	When this option is enabled, this system will automatically sync the CARP Virtual IPs to the other HA host when changes are made.

Fonte: do autor

A configuração dos IPs Virtuais é feita através da aba **Firewall** → **Virtual IPs**, adiciona-se um novo IP virtual do tipo CARP, escolhe a interface, define o endereço IP virtual, senha, *VHID Group*, *Advertising Frequency* e uma descrição para o IP. As figuras 16, 17 e 18 trazem estas configurações dos IPs virtuais para as duas interfaces (WAN e LAN). Uma atenção especial deve ser dada para o campo “*Advertising Frequency*”, pois este é quem define a prioridade de ser o mestre ou escravo, quanto menor o valor da frequência, maior na hierarquia para assumir o lugar do mestre a máquina estará, o mestre sempre assume o valor “0”, definido na configuração “*Skew*”.

Figura 16 – IP Virtual WAN

The screenshot shows the 'Edit Virtual IP' configuration page for a WAN interface. The interface is part of the Sense firewall management system. At the top, there is a navigation menu with options: System, Interfaces, Firewall, Services, VPN, Status, Diagnostics, and Help. A notification bar at the top right indicates 'New event: Please click to confirm.' and '65 unread notices'. The main title is 'Firewall: Virtual IP Address: Edit'. The configuration form includes the following fields:

Edit Virtual IP	
Type	<input type="radio"/> IP Alias <input checked="" type="radio"/> CARP <input type="radio"/> Proxy ARP <input type="radio"/> Other
Interface	WAN
IP Address(es)	Type: Single address Address: 200.16.172.1 / 8 <small>This must be the network's subnet mask. It does not specify a CIDR range.</small>
Virtual IP Password Enter the VHID group password.
VHID Group	1 Enter the VHID group that the machines will share
Advertising Frequency	Base: 1 Skew: 0 The frequency that this machine will advertise. 0 means usually master. Otherwise the lowest combination of both values in the cluster determines the master.
Description	IP Virtual de sincronismo WAN You may enter a description here for your reference (not parsed).

Fonte: do autor

Figura 17 – IP Virtual LAN

The screenshot shows the 'Edit Virtual IP' configuration page for a LAN interface. The interface is part of the Sense firewall management system. At the top, there is a navigation menu with options: System, Interfaces, Firewall, Services, VPN, Status, Diagnostics, and Help. A notification bar at the top right indicates 'New event: Please click to confirm.' and '65 unread notices'. The main title is 'Firewall: Virtual IP Address: Edit'. The configuration form includes the following fields:

Edit Virtual IP	
Type	<input type="radio"/> IP Alias <input checked="" type="radio"/> CARP <input type="radio"/> Proxy ARP <input type="radio"/> Other
Interface	LAN
IP Address(es)	Type: Single address Address: 10.0.0.1 / 8 <small>This must be the network's subnet mask. It does not specify a CIDR range.</small>
Virtual IP Password Enter the VHID group password.
VHID Group	1 Enter the VHID group that the machines will share
Advertising Frequency	Base: 1 Skew: 0 The frequency that this machine will advertise. 0 means usually master. Otherwise the lowest combination of both values in the cluster determines the master.
Description	IP Virtual de sincronismo LAN You may enter a description here for your reference (not parsed).

Fonte: do autor

Figura 18 – IPs Virtuais Mestre

System > Interfaces > Firewall > Services > VPN > Status > Diagnostics > Help

Firewall: Virtual IP Addresses

Virtual IPs CARP Settings

Virtual IP address	Interface	Type	Description
200.16.172.1/25 (vhid 1)	WAN	CARP	Virtual Ip endereço Wan
10.0.0.1/8 (vhid 3)	LAN	CARP	Virtual IP - Endereço de LAN

Note:
The virtual IP addresses defined on this page may be used in NAT mappings.
You can check the status of your CARP Virtual IPs and interfaces here.

Fonte: do autor

Em *Status* → *CARP (failover)*, fica visível, conforme indicado pela seta na figura 19, o status do sistema de alta disponibilidade, esta tela traz também as informações sobre a interface, o IP Virtual e o status; no caso da máquina mestre, seu status aparece como “*Master*”.

Figura 19 – Status Mestre

System > Interfaces > Firewall > Services > VPN > Status > Diagnostics > Help

Status: CARP

Disable Carp

CARP Interface	Virtual IP	Status
wan_vip1	200.17.173.203	MASTER
lan_vip3	10.1.3.3	MASTER

Note:
You can configure high availability sync settings here.

pfSync nodes:

```

20e209d5
4b330f80
872ce406
bfa6e2b0
db60339a
f7a16bf5

```

Fonte: do autor

3.3.4.2 Escravo

A configuração no servidor escravo é relativamente mais simples, ele também dispõe de três interfaces de rede, configuradas conforme a tabela 4.

Tabela 4 – Interfaces do Servidor Escravo

Nome	IP	Função
WAN	200.16.172.3	Acesso externo
LAN	10.0.0.3	Endereço da máquina na rede interna
SYNC	192.168.0.2	Sincronismo entre os servidores

Fonte: do autor

A regra de sincronismo também tem que ser construída e aplica-se a mesma forma de construção citada na configuração do mestre, que pode ser vista na figura 20.

Figura 20 – Sincronismo Escravo



Fonte: do autor

A configuração do CARP, conforme figura 21, segue a mesma forma do servidor mestre, porém somente é ativado o sincronismo e definido a interface (SYNC), deixando os campos usuário, senha, número IP da interface de sincronismo e as opções de sincronização em branco.

Figura 21 – Configuração CARP Escravo

System: High Availability Sync

State Synchronization Settings (pfsync)

Synchronize States

pfsync transfers state insertion, update, and deletion messages between firewalls. Each firewall sends these messages out via multicast on a specified interface, using the PFSYNC protocol (IP Protocol 240). It also listens on that interface for similar messages from other firewalls, and imports them into the local state table.

This setting should be enabled on all members of a failover group.

NOTE: Clicking save will force a configuration sync if it is enabled! (see Configuration Synchronization Settings below)

Synchronize Interface

If Synchronize States is enabled, it will utilize this interface for communication.

NOTE: We recommend setting this to a interface other than LAN! A dedicated interface works the best.

NOTE: You must define a IP on each machine participating in this failover group.

NOTE: You must have an IP assigned to the interface on any participating sync nodes.

pfsync Synchronize Peer IP

Setting this option will force pfsync to synchronize its state table to this IP address. The default is directed multicast.

Configuration Synchronization Settings (XMLRPC Sync)

Synchronize Config to IP

Enter the IP address of the firewall to which the selected configuration sections should be synchronized.

NOTE: XMLRPC sync is currently only supported over connections using the same protocol and port as this system - make sure the remote system's port and protocol are set accordingly!

NOTE: Do not use the Synchronize Config to IP and password option on backup cluster members!

Remote System Username

Enter the webConfigurator username of the system entered above for synchronizing your configuration.

NOTE: Do not use the Synchronize Config to IP and username option on backup cluster members!

Remote System Password

Enter the webConfigurator password of the system entered above for synchronizing your configuration.

NOTE: Do not use the Synchronize Config to IP and password option on backup cluster members!

Fonte: do autor

Os IPs Virtuais do escravo não necessitam ser configurados, o protocolo CARP é responsável pela criação e sincronização, sendo automático. As figuras 22, 23 e 24 mostram as configurações dos IPs Virtuais na máquina escravo. Nota-se que as interfaces WAN e LAN foram configuradas com os mesmos IPs virtuais do mestre, para que quando ocorra uma indisponibilidade no servidor mestre, o escravo possa assumir todas as configurações e serviços, mantendo assim a rede em funcionamento. O campo “*Skew*” foi definido com o valor de “100”, pois deve sempre ter um valor maior que o “*Skew*” do servidor mestre.

Figura 22 – IP Virtual WAN Escravo

The screenshot shows the 'Firewall: Virtual IP Address: Edit' configuration page in pfSense. The interface is set to 'WAN'. The IP address is '200.16.172.1' with a subnet mask of '/25'. The Vhid group is '1'. The advertising frequency base is '1' and skew is '100'. The description is 'Virtual Ip endereço Wan'.

Edit Virtual IP	
Type	<input type="radio"/> IP Alias <input checked="" type="radio"/> CARP <input type="radio"/> Proxy ARP <input type="radio"/> Other
Interface	WAN
IP Address(es)	Type: Single address Address: 200.16.172.1 / 25 <small>This must be the network's subnet mask. It does not specify a CIDR range.</small>
Virtual IP Password Enter the Vhid group password.
Vhid Group	1 Enter the Vhid group that the machines will share
Advertising Frequency	Base: 1 Skew: 100 <small>The frequency that this machine will advertise. 0 means usually master. Otherwise the lowest combination of both values in the cluster determines the master.</small>
Description	Virtual Ip endereço Wan <small>You may enter a description here for your reference (not parsed).</small>

Fonte: do autor

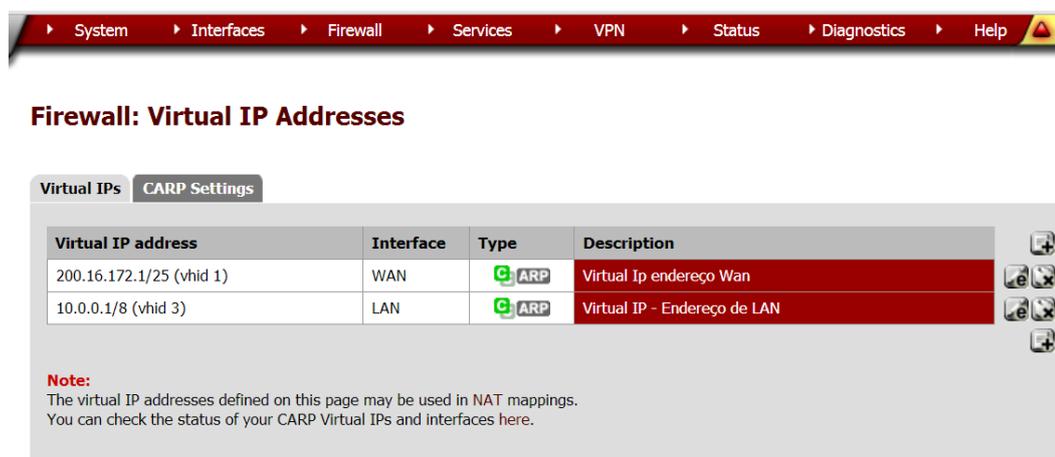
Figura 23 – IP Virtual LAN Escravo

The screenshot shows the 'Firewall: Virtual IP Address: Edit' configuration page in pfSense. The interface is set to 'LAN'. The IP address is '10.0.0.1' with a subnet mask of '/24'. The Vhid group is '3'. The advertising frequency base is '1' and skew is '100'. The description is 'Virtual IP - Endereço de LAN'.

Edit Virtual IP	
Type	<input type="radio"/> IP Alias <input checked="" type="radio"/> CARP <input type="radio"/> Proxy ARP <input type="radio"/> Other
Interface	LAN
IP Address(es)	Type: Single address Address: 10.0.0.1 / 24 <small>This must be the network's subnet mask. It does not specify a CIDR range.</small>
Virtual IP Password Enter the Vhid group password.
Vhid Group	3 Enter the Vhid group that the machines will share
Advertising Frequency	Base: 1 Skew: 100 <small>The frequency that this machine will advertise. 0 means usually master. Otherwise the lowest combination of both values in the cluster determines the master.</small>
Description	Virtual IP - Endereço de LAN <small>You may enter a description here for your reference (not parsed).</small>

Fonte: do autor

Figura 24 – IPs Virtuais



Firewall: Virtual IP Addresses

Virtual IPs CARP Settings

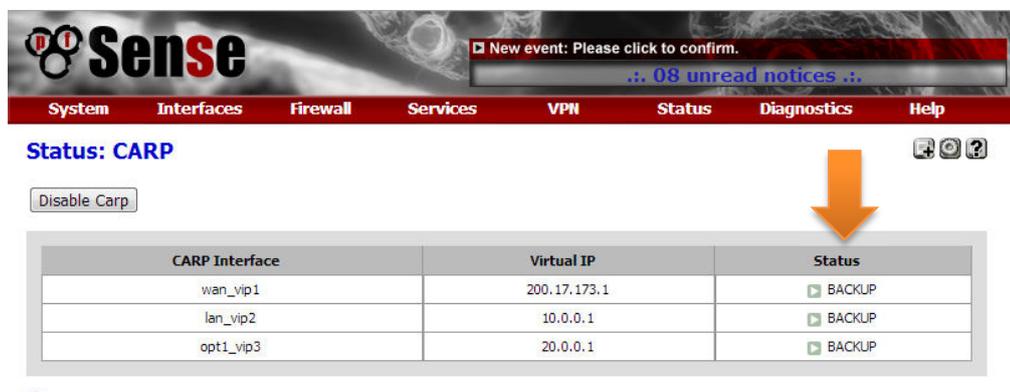
Virtual IP address	Interface	Type	Description
200.16.172.1/25 (vhid 1)	WAN	ARP	Virtual Ip endereço Wan
10.0.0.1/8 (vhid 3)	LAN	ARP	Virtual IP - Endereço de LAN

Note:
The virtual IP addresses defined on this page may be used in NAT mappings.
You can check the status of your CARP Virtual IPs and interfaces here.

Fonte: do autor

Em **Status** → **CARP (failover)**, fica visível, conforme indicado pela seta na figura 25, o status do sistema de alta disponibilidade, no caso do escravo é “*backup*”, ou seja, aguarda em execução.

Figura 25 – Status



Status: CARP

Disable Carp

CARP Interface	Virtual IP	Status
wan_vip1	200.17.173.1	BACKUP
lan_vip2	10.0.0.1	BACKUP
opt1_vip3	20.0.0.1	BACKUP

Fonte: do autor

3.3.4.5 Testes e Resultados

Os testes aplicados após as configurações feitas nos servidores mestre e escravo foram de três formas, a primeira com os dois servidores (mestre e escravo) ligados, a segunda e mais importante para a alta disponibilidade com o mestre desligado e somente o escravo ligado e a terceira o servidor mestre é religado e reassume o controle da rede. A conexão ou acesso a rede e os downloads são o foco dos testes.

O Esquema de Testes consta na tabela 5.

Tabela 5 – TESTE 1 - Mestre e Escravo Ligado

Servidor	Ligado	Desligado	Conexão	Download
Mestre	V		OK	OK
Escravo	V			

Fonte: do autor

As figuras 26 e 27 demonstram o ambiente de alta disponibilidade em execução, as páginas divididas ao fundo mostram o status dos dois servidores quanto à alta disponibilidade, o mestre no lado esquerdo com o status em “*Master*” e o escravo no direito em espera com o status “*Backup*”. Os terminais de comando, figuras 28, 29 e 30, aparecem para ficar visível quando a troca entre os servidores for efetuada, o comando “*ping*” mais os endereços IPs reais e virtual de cada servidor. Juntamente com o comando *ping*, foi usado a opção *-t*, para que seja contínuo, parando apenas quando forem pressionadas as tecla “*ctrl +c*”, ficando assim: “*ping -t endereço ip*”, sendo, desta forma, possível a visualização.

Figura 26 – Servidor Mestre em Execução



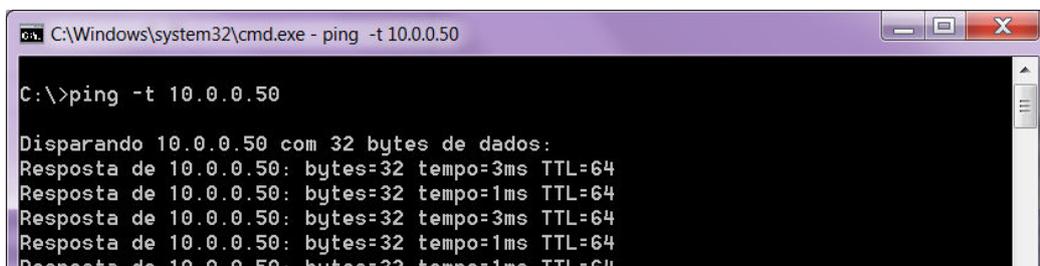
Fonte: do autor

Figura 27 - Servidor Escravo em Execução



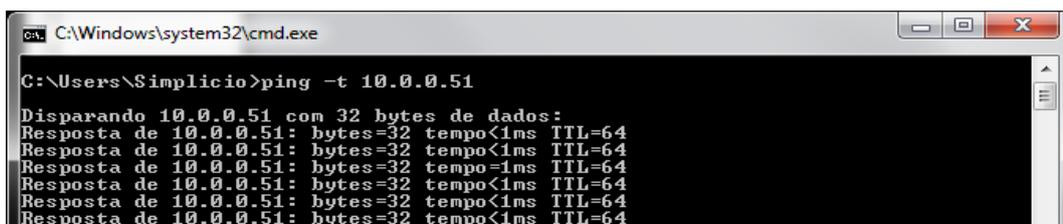
Fonte: do autor

Figura 28 – Ping no Servidor Mestre



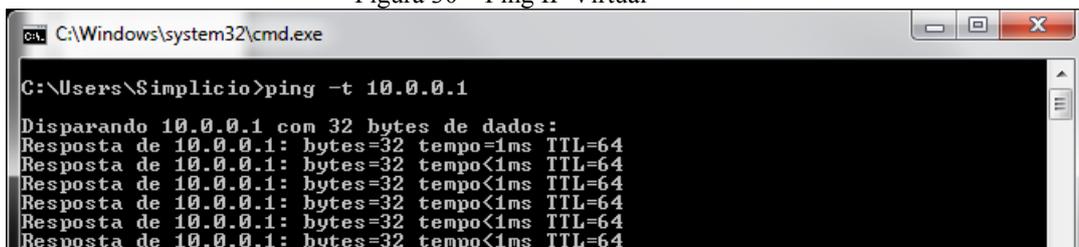
Fonte: do autor

Figura 29 - Ping no Servidor Escravo



Fonte: do autor

Figura 30 – Ping IP Virtual



Fonte: do autor

Nota-se, na figura 30, o último terminal, que o IP virtual configurado no protocolo CARP também responde ao comando *ping*. Na tabela 6, consta o esquema do teste 2.

Tabela 6 – TESTE 2 - Escravo Ligado

Servidor	Ligado	Desligado	Conexão	Download
Mestre		V		
Escravo	V		OK	X

Fonte: do autor

Ao desligar o servidor mestre (tabela 6) e o escravo assumir as funcionalidades configuradas na sincronização, observa-se um problema com os downloads longos e ativos. No momento da troca entre os servidores, eles perdem a conexão, sendo necessário o restabelecimento de forma manual pelo usuário. Os downloads iniciados após a ocorrência de troca entre os servidores mestre e escravo ocorrem de maneira regular e sem maiores problemas. As concessões de IPs da rede através de DHCP permaneceram estáveis e sem problema.

Na troca de servidores, figuras 31,32 e 33, os terminais demonstram com clareza o mestre estourando o limite de tempo e perdendo os pacotes enviados; já o servidor escravo continua ativo e respondendo a todos os pacotes enviados, o IP virtual e flutuante esgota apenas dois limites de tempo e é transferido para o servidor redundante, ficando claro que a troca ocorre quase que instantaneamente, respeitando assim a disponibilidade dos serviços que estão sob responsabilidade do *firewall*.

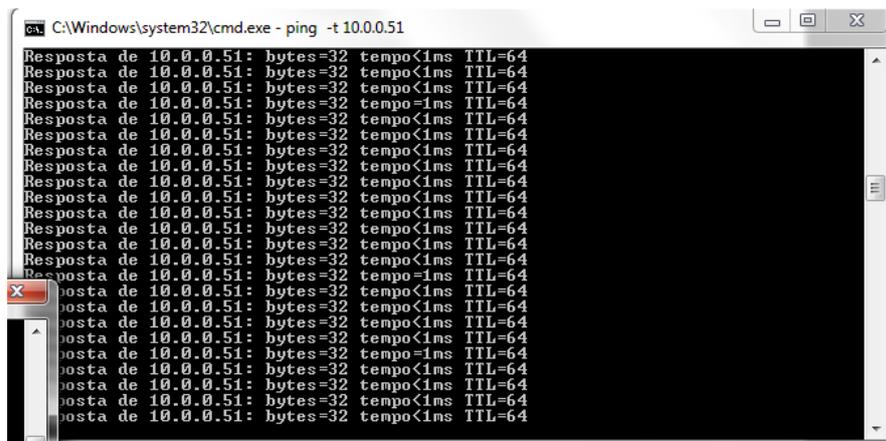
Figura 31 - Servidor Mestre Desligado

```

C:\Windows\system32\cmd.exe - ping -t 10.0.0.50
Resposta de 10.0.0.50: bytes=32 tempo<1ms TTL=64
Esgotado o tempo limite do pedido.
  
```

Fonte: do autor

Figura 32 – Servidor Escravo Ligado



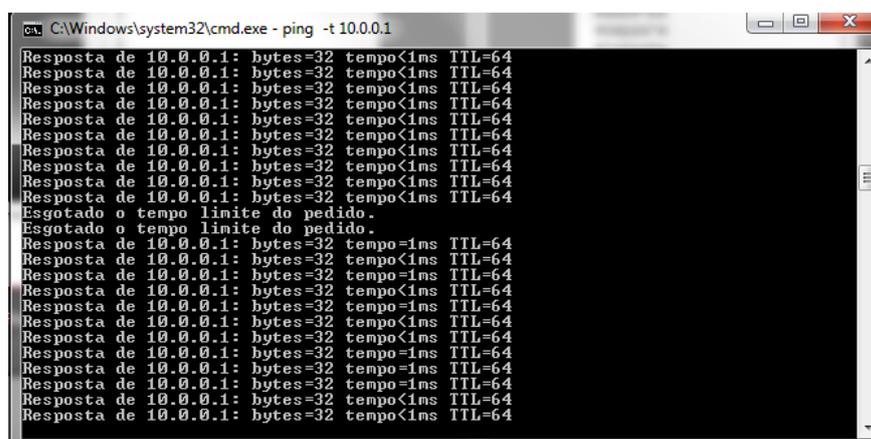
```

C:\Windows\system32\cmd.exe - ping -t 10.0.0.51
Resposta de 10.0.0.51: bytes=32 tempo<1ms TTL=64

```

Fonte: do autor

Figura 33 – IP Virtual



```

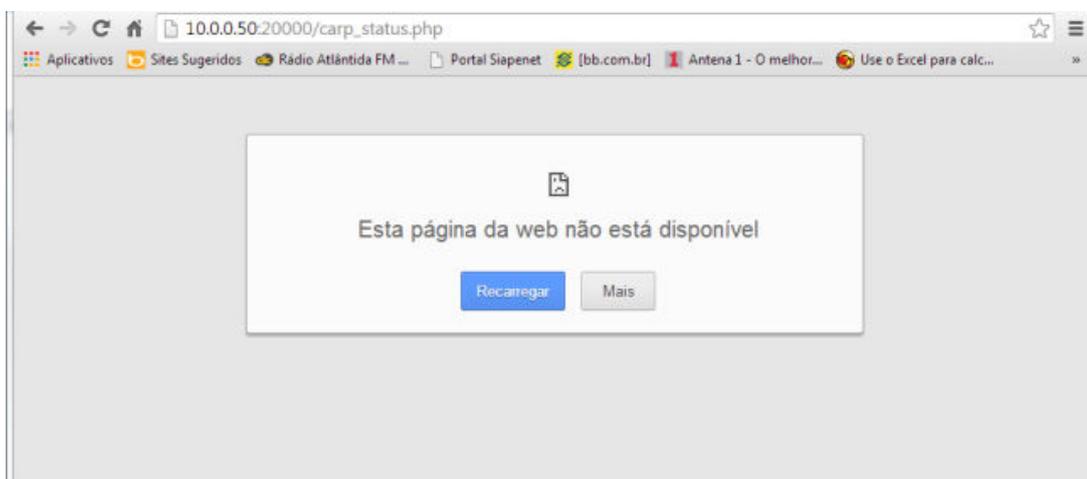
C:\Windows\system32\cmd.exe - ping -t 10.0.0.1
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Esgotado o tempo limite do pedido.
Esgotado o tempo limite do pedido.
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 10.0.0.1: bytes=32 tempo=1ms TTL=64

```

Fonte: do autor

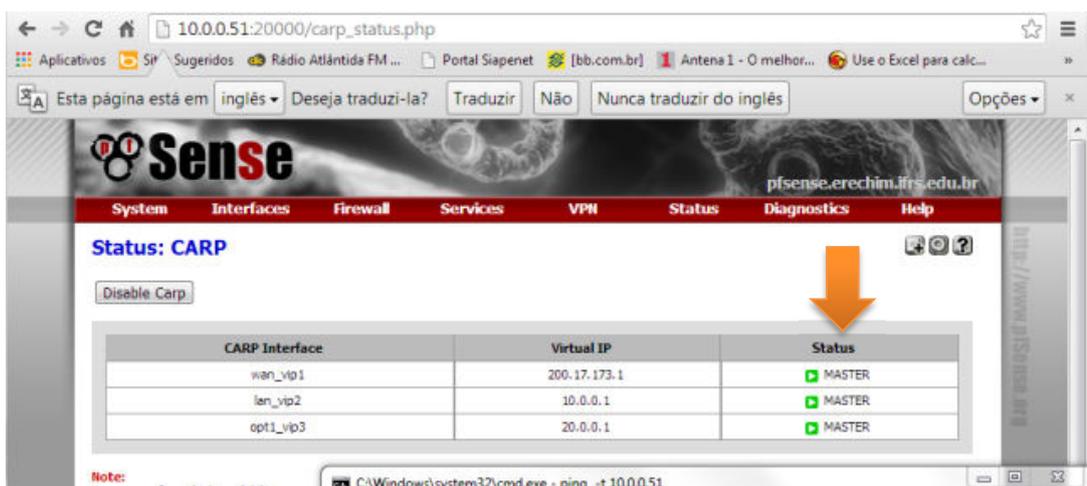
A figura 34 mostra o servidor mestre desligado e inacessível. Na figura 35. É possível verificar o servidor escravo assumindo todos os serviços do *firewall*, seu status passa de “Backup” para “Master”.

Figura 34 – Servidor Mestre sendo desligado



Fonte: do autor

Figura 35 - Status Servidor Escravo



Fonte: do autor

As figuras 36 e 37 demonstram o retorno do servidor mestre, restabelecendo todos os serviços da rede, a conexão e os downloads permaneceram estáveis e com bom desempenho, ocorre também a troca de status, conforme indicado.

Figura 36 – Servidor Mestre Religado



Fonte: do autor

Figura 37 - Servidor Escravo



Fonte: do autor

4 CONSIDERAÇÕES FINAIS

A distribuição *PFSense versão 2.1* mostra-se uma ótima ferramenta para uso em *firewall* com ou sem alta disponibilidade. Através de sua interface web, fica muito fácil e prático o gerenciamento da rede, trazendo inúmeras possibilidades de configuração e maneiras de usá-lo. A virtualização dos servidores também traz muitas vantagens quando se administra uma rede, pois torna o processo de criação e manutenção de máquinas eficiente, seguro e extremamente rápido.

O setor de Tecnologia da Informação do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul, Câmpus Erechim, agora dispõe de um sistema redundante, de alta disponibilidade e tolerante a falhas em seu *firewall*, proporcionando assim segurança e disponibilidade no acesso e controle da rede. Com este ambiente de virtualização construído, fica a possibilidade de, em um futuro próximo, também ter outras máquinas e sistemas em alta disponibilidade, pois este trabalho concentrou-se apenas no *firewall*, desconsiderando os outros serviços.

A principal dificuldade encontrada foi a implementação dos servidores de *firewall* em alta disponibilidade, pois exigiu que a rede fosse parada por um determinado tempo, para que as configurações de sincronismo entre os servidores mestre e escravo fossem feitas, o que acarretou em reclamações de usuários da rede, mesmo que avisados através de e-mail com antecedência da manutenção nos servidores e serviços.

Vale também ressaltar o quão importante foi a implementação na prática, ou seja, vivenciar a experiência prática, não ficando apenas na conceituação teórica, pois várias dificuldades surgem e precisam ser contornadas para que seja possível a continuação do projeto. Indo além dos domínios técnicos, surgem dificuldades referentes às relações interpessoais, o que torna a implementação mais difícil, pois cada indivíduo tem uma opinião e uma visão sobre o assunto.

Após tudo o que foi apresentado neste trabalho, acredita-se que os objetivos de disponibilidade, segurança e custo zero na implementação de um *firewall* de alta disponibilidade foram atingidos, sendo esta uma solução viável para empresas e instituições que tenham necessidade de um *firewall* com estas características, pois os estudos apontaram e os testes mostram que é possível chegar muito próximo aos 100% de disponibilidade dos serviços propostos em um *firewall*.

Futuramente, com o crescimento do câmpus e conseqüentemente o aumento de usuários e consumo de banda da rede, poderá ser necessário a configuração de outros serviços, como o balanceamento de carga (*load balance*), para que dois ou mais links de internet sejam distribuídos de forma alternativa garantindo assim a disponibilidade da rede de modo transparente para os usuários. Outro serviço que também pode ser configurado em uma distribuição *PFSense* é o *Captive portal*, este é um software responsável pelo gerenciamento e controle de acesso à internet em redes públicas através de autenticação de usuário e senha.

REFERÊNCIAS

- ABNT, Associação Brasileira De Normas Técnicas. Nbr Iso/Iec 17799. **Information technology -- Security techniques -- IT network security**, 2005.
- ALBERTIN, Alberto Luiz, PINOCHET Luis Hernan Contreras. **Política de Segurança de Informações**. Uma Visão Organizacional para a sua Formulação. São Paulo: Elsevier, 2010.
- AVIZIENIS, design of fault-tolerant computers. **Proceedings of fall joint computer conf.**, vol.31, Thompson Books, 1967.
- BRANDÃO, Robson. **Conceitos sobre Disponibilidade - Parte I**. Disponível em: <<http://technet.microsoft.com/pt-br/library/cc668492.aspx>>. Acesso em: 03 jun. 2013.
- CARISSIMI, Alexandre. Virtualização: da teoria a soluções. In: **Simpósio brasileiro de redes de computadores e sistemas distribuídos**, 2008, Rio de Janeiro. Livro texto dos minicursos. Rio de Janeiro: SBC, 2008. p. 174-199.
- CHESWICK, William R. **Firewalls e Segurança na Internet**. Repelindo o Hacker Arditoso. 2.ed. Porto Alegre: Bookman, 2005.
- DELL. Servidor em rack PowerEdge R710 11G. Disponível em:<<http://www.dell.com/br/empresa/p/poweredge-r710/pd>. Acesso em: 22 ago. 2013.
- IBM. Hypervisors, virtualização e nuvem: Aprofunde-se no Hypervisor Xen. Disponível em: <<http://www.ibm.com/developerworks/br/cloud/library/clhypervisorcompare-xen/>>. Acesso em: 20 ago. 2013.
- LOPES FILHO, Edmo. **Arquitetura de Alta Disponibilidade para Firewall e IPS baseada em SCTP**. Dissertação (Mestrado)-Universidade Federal de Uberlândia, Minas Gerais , 2008.
- MANUAL do FreeBSD: O Projeto de Documentação do FreeBSD. Revisão: 42953. Disponível em: <http://www.freebsd.org/doc/en_US.ISO8859-/books/handbook/carp.html>. Acesso em: 28 out. 2013.
- NIC BR, Security Office. **Práticas de Segurança para Administradores de Redes Internet**. Versão 1.2. 2003. Disponível em: < <http://www.nbso.nic.br/>> Acesso em 28 out. 2013.
- NOMOA. **Hot Failover with CARP** - Parte I. Disponível em:<http://nomoa.com/bsd/gateway/highavailability/carp.html> Acesso em: 12 jun. 2013.
- STALLINGS, William. **Criptografia e segurança de redes**. Princípios e práticas. 4.ed. São Paulo: Pearson Prentice Hall, 2008.
- WILLIAMSOM, Matt, PERSAUD, Christopher. **Livro do pfSense 2.0**. 2012. Disponível em: <<Http://pfsense.org>> Acesso em: 10 out. 2013.
- WEBER, Taisy Silva. **Tolerância a falhas: conceitos e exemplos**. Disponível em: <<http://www.inf.ufrgs.br/~taisy/disciplinas/textos/ConceitosDependabilidade.PDF> > Acesso em 28 out. 2013.