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

PABLO SMITH DA SILVA SANTOS

ALTA DISPONIBILIDADE UTILIZANDO O FIREWALL PFSENSE

NATAL/RN
2015

PABLO SMITH DA SILVA SANTOS

ALTA DISPONIBILIDADE UTILIZANDO O FIREWALL PFSENSE

Trabalho de Conclusão de Curso apresentado ao Centro Universitário do Rio Grande do Norte (UNI-RN), como requisito final para obtenção do título de Especialista em Redes de Computadores.

Orientador: Prof. M.Sc. Aluízio Ferreira da Rocha Neto

NATAL/RN

2015

Catalogação na Publicação – Biblioteca da UNI-RN
Setor de Processos Técnicos

Santos, Pablo Smith da Silva.
Alta disponibilidade utilizando o firewall pfsense / Pablo Smith da Silva Santos. – Natal, 2015.
57 f.

Orientador: Prof. M.Sc. Aluízio Ferreira da Rocha Neto.
Monografia (Especialização em Redes de Computadores) – Centro Universitário do Rio Grande do Norte.

1. Alta disponibilidade – Monografia. 2. CARP – Monografia. 3. PFSYNC – Monografia. 4. pfSense – Monografia. I. Rocha Neto, Aluízio Ferreira da. II. Título.

RN/UNI-RN/BC CDU 004.72

PABLO SMITH DA SILVA SANTOS

ALTA DISPONIBILIDADE UTILIZANDO O FIREWALL PFSENSE

Trabalho de Conclusão de Curso apresentado ao Centro Universitário do Rio Grande do Norte (UNI-RN), como requisito final para obtenção do título de Especialista em Redes de Computadores.

Aprovado em:_____/_____/_____.

BANCA EXAMINADORA

_____

Prof. Aluízio Ferreira da Rocha Neto

Orientador

_____

Professor Convidado

Membro

_____

Professor Convidado

Membro

Dedico este trabalho à minha grande mãe, por ela ter sido a minha principal incentivadora, sempre me estimulando a seguir em frente, ficando ao meu lado nas minhas escolhas mesmo quando não concordava. Ela foi a grande responsável por essa conquista.

AGRADECIMENTO

Agradeço em primeiro lugar a Deus por ter me proporcionado mais essa conquista.

Ao principal responsável pela concepção desse trabalho, o meu amigo e por muito tempo companheiro de trabalho José Romualdo Barros de Santana, apresentando-me a ferramenta chave da elaboração desse trabalho e propondo-me o desafio do aprendizado.

Ao professor Aluízio Ferreira da Rocha Neto, pela dedicação em suas orientações prestadas na elaboração deste trabalho, incentivando-me e colaborando no desenvolvimento de meu trabalho.

RESUMO

A internet da atualidade exige que os sistemas informatizados estejam sempre disponíveis. Em virtude disso, a Alta Disponibilidade tem se tornado necessidade em empresas de TI, além da contínua preocupação em sempre manter elevados níveis de segurança. No decorrer deste trabalho, será demostrado a implementação de um ambiente em Alta Disponibilidade com o uso do firewall pfSense, ferramenta que vem caindo no gosto dos profissionais da área por possuir diversos recursos de rede e boa reputação em relação à segurança. Esta ferramenta oferece a possibilidade de manter um ou mais serviços em funcionamento mesmo que ocorra o desligamento de um servidor, para isso, faz uso de mecanismos como o CARP e o PFSYNC.

Palavras-chave: Alta disponibilidade. CARP. PFSYNC. pfSense.

ABSTRACT

Nowadays the internet requires that computerized systems are always available. Because of this, the High Availability has become a need for IT companies, as well as the continuing concern in mind to always ensure high levels of safety. In the course of this work, it will be demonstrated the implementation of an environment for High Availability using the pfSense firewall, a tool that has been pleased the professionals because it has various network resources and good reputation related to security. This tool offers the possibility to maintain one or more services in operation even if there is the shutting down of a server, for this, it makes use of mechanisms such as CARP and PFSYNC.

Keywords: High Availability. CARP. PFSYNC. pfSense.

LISTA DE FIGURAS

SUMÁRIO

1 INTRODUÇÃO

Não é difícil encontrar em estabelecimentos comerciais ou empresariais funcionários que desempenham diversas funções. Em grande parte, eles são os mais antigos ou detém maior conhecimento sobre determinados assuntos relacionados às suas funções na empresa, tornando-os mais requisitados. Quando isso ocorre, tem-se a percepção que eles parecem sempre estar correndo, ou até mesmo estressados, na realidade, essa percepção é verdadeira, pois, eles sempre estão realizando o dobro da função a que lhe foi atribuída. Esse tipo de funcionário dificilmente consegue tirar férias devido a sua importância e, quando isso acontece geralmente o seu setor ou algumas das funções que ele desempenhava, ficam parados sem ser realizados por falta de alguém que os realize com a mesma qualidade.

Pode-se usar o exemplo dado para explicar o que acontece com a maioria das máquinas tidas como servidores em grande parte das empresas espalhadas ao redor do mundo. Em analogia a situação mostrada, tem-se a estrutura informatizada de algumas empresas, onde os funcionários representam os servidores e, suas funções serão os serviços que cada servidor tem que manter em funcionamento.

Assim como os funcionários trabalham de forma sobrecarregada, os servidores também, só que de uma maneira muito pior, porque os servidores geralmente são os concentradores dos sistemas e, se por alguma eventualidade vierem a parar todos os sistemas que estejam atrelados a eles também irão parar.

Em um ambiente informatizado onde temos o servidor firewall como concentrador principal de acesso à internet, desempenhado além do firewall, funções como: Proxy, Balanceador de Carga e VPN, se ele parar, consequentemente toda a comunicação com o meio externo que era feito por intermédio deste servidor, vai ficar prejudicada. Esta situação torna-o um ponto de falha de alto risco! Levando em consideração a dependência da internet, ficar sem acesso a ela dependendo do segmento empresarial, pode significar um prejuízo financeiro enorme.

Devido a esse tipo de problema, ao longo dos anos foram utilizadas diversas maneiras de minimizar os riscos de se ter um host concentrador, onde tudo fica dependente dele para funcionar. Uma forma bastante utilizada, mais que vem caindo em desuso é, distribuir funções entre vários hosts, mas isso acaba sento muito caro

além de bastante trabalhoso, pois, é muito mais difícil de gerenciar e manter diversos equipamentos ao invés de apenas um. Com a intenção de pôr fim a esse problema, grandes empresas de desenvolvimento de *hardware* e *software* desenvolveram mecanismos para tornar seus sistemas mais seguros em relação a falhas, chegando ao que chamamos de Alta Disponibilidade, um sistema informatizado resistente a falhas de *hardware*, *software* e energia, cujo objetivo é manter os serviços disponibilizados o máximo de tempo possível. Inicialmente isso era muito caro porque essas alternativas só eram vendidas junto a soluções proprietárias e, poucas eram as alternativas para fugir do grande investimento, mas, a necessidade fez com que fossem desenvolvidas outras técnicas para se chegar ao mesmo objetivo utilizado *hardware* mais baratos e *softwares gratuitos*.

Diversas técnicas e medidas foram adotadas durante o passar dos anos, no entanto, atualmente o que vem caindo no gosto das empresas de médio e pequeno porte e que buscam por uma solução de firewall é, o pfSense. A sua busca tem aumentado por ser uma alternativa aos *softwares* de segurança em firewalls de alto custo. Diferente de algumas outras opções do mercado, o pfSense pode ser adquirido gratuitamente direto de sua página na internet, além de, possuir uma interface de fácil gerenciamento que ajuda no aprendizado e facilita sua utilização, possui uma excelente reputação em relação a segurança e diversos outros recursos.

1.1 OBJETIVOS

1.1.1 Geral

Os objetivos gerais deste trabalho são demonstrar a configuração de um ambiente informatizado em regime de Alta Disponibilidade com o firewall pfSense, apresentando os conceitos das principais tecnologias envolvidas.

1.1.2 Objetivos Específicos

Para atingir o objetivo geral, as seguintes etapas foram desenvolvidas para a confecção deste documento.

- Apresentar conceitos, protocolos e ferramentas utilizadas;

- Apresentar o ambiente virtual proposto;
- Definir hierarquias dos dispositivos necessários;
- Configurar dispositivos em ambiente de Alta Disponibilidade.

1.2 ORGANIZAÇÃO DO TRABALHO

Este trabalho foi organizado de forma sequencial partindo dos conceitos necessários para o bom entendimento sobre o assunto, assim como o entendimento sobre os protocolos e equipamentos utilizados, até chegar ao objetivo final. O capítulo 2 apresenta o referencial teórico, contendo os conceitos, protocolos, ferramentas necessárias. O capítulo 3 apresenta o desenvolvimento do trabalho, mostrando o que precisa ser feito para deixar o ambiente configurado em Alta Disponibilidade, o capítulo 4 terá a conclusão e em sequência as referências.

2 REFERENCIAL TEÓRICO

É preciso conhecer um pouco sobre os componentes envolvidos para que a situação proposta possa funcionar. Ao ter a noção de como cada componente atua e qual a necessidade de cada um dentro da configuração desejada, será necessário partir para configuração do ambiente, demostrando a configuração básica para deixar tudo funcionando.

2.1 PLANEJAMENTO

Sabe-se que a Alta Disponibilidade não é um programa, muito menos um produto e, que cada ambiente tem uma necessidade diferente, com isso o planejamento se torna algo indispensável, com ele a solução poderá ser implementada e aproveitada da forma correta, reduzindo os pontos de falhas e os custos de implantação, além de, gerar uma visão sobre o ambiente ao qual se quer melhorar, mostrando o que realmente é crítico e o que deve ser atingido com o estudo.

É no planejamento que será possível identificar a possibilidade de utilizar equipamentos de diferentes fabricantes e se pode ser usado *desktops* ao invés de grandes servidores. O pfSense, solução adotada neste trabalho, se encaixa perfeitamente a essas caraterísticas, tornando possível a redução de custos da implantação.

2.2 DISPONIBILIDADE

Segundo o dicionário online Michaelis (2015), em uma de suas definições disponibilidade é: “Qualidade daquele ou daquilo que é ou está disponível”.

Mais na área de tecnologia da informação, Taylor e Ranganathan (2013) descreve disponibilidade como, “[...] a probabilidade de um sistema executar a função a que se destina, em condições operacionais e ambientais especificados.”, ou seja, é a probabilidade de um sistema está disponível sempre que for solicitado, caso ele esteja em condições normais a que foi projetado. É importante deixar claro que, o nível de disponibilidade poderá ser diferente de acordo com ambiente em que será implantada, havendo necessidades diferentes em cada situação, ficando

explícita a importância do planejamento bem feito e detalhado.

Considera-se que disponibilidade de um sistema é o tempo durante o qual ele está em operação em relação ao tempo em que ele deve estar em operação. Pode-se calcular o nível de disponibilidade utilizando a fórmula.

$$Disponibilidade = \frac{MTBF}{MTBF + MTTR} = \frac{tempo\ do\ sistema\ em\ operação}{tempo\ total\ incluindo\ falhas}$$

Sendo:

- MTBF (Mean Time Between Failures): tempo médio entre falhas.
- MTTR (Mean Time to Rapair): tempo médio de reparo.

Para sistemas altamente confiáveis, este número deve estar muito próximo de 1,000 (ou 100%).

2.1.1Classes de Disponibilidade

Parece algo simples mas, a partir de um olhar atento torna-se é um conteúdo extenso, então, seguindo essa forma de pensamento, chega-se a uma segmentação.

> A Disponibilidade de um sistema computacional, indicada por A(t), é a probabilidade de que este sistema esteja funcionando e pronto para uso em um dado instante de tempo t. Esta disponibilidade pode ser enquadrada em três classes, de acordo com a faixa de valores desta probabilidade. As três classes são: Disponibilidade Básica, Alta Disponibilidade e Disponibilidade Contínua (SEVERICH, 2012),

2.1.1.1 Disponibilidade Básica

Se encaixam aplicações simples de baixo nível de importância, geralmente encontradas em máquinas consideradas comuns onde não há nenhum sistema de detecção de falhas. Nesta situação, quando há falhas, o tempo em que essas máquinas ficam paradas pode durar por dias sem que o serviço seja prejudicado.

2.1.1.2 Alta Disponibilidade

É neste nível onde esse trabalho se encaixará, pois é aqui que se encontra

grande parte das empresas. Neste nível de disponibilidade já é possível encontrar diversas formas e mecanismos de detecção, recuperação e mascaramento de falhas, devido a uma necessidade de disponibilidade maior dos serviços em funcionamento. Esse nível de disponibilidade não permite que serviços fiquem muito tempo inacessível, são aceitáveis apenas minutos ou no máximo poucas horas fora do ar.

2.1.1.3 Disponibilidade Contínua

Este é o nível mais crítico entre todos os níveis, aqui se encaixam sistemas como transações financeiras, transmissão de satélites, controle de voo, bolsa de valores e outros. Nesta situação segundos fora do ar pode causar perdas irreparáveis.

2.3 REDUNDÂNCIA

Em TI, redundância significa ter mais de um hosts realizando a mesma tarefa. O mais comum é que haja redundância de *hardware* mas também pode existir de *software*. Neste trabalho a redundância ocorrerá nos firewalls, serão duas máquinas configuradas com o pfSense, uma será considerada *master* e a outra *backup*, quando ocorrer uma falha no *host* considerado *master* e ele parar de responder as solicitações, o *host backup* irá assumir o papel até que tudo seja normalizado. Assim o serviço continuará sendo prestado sem interrupção perceptível ao usuário.

2.4 TOLERÂNCIA A FALHA

Pode ser considerada o coração da Alta Disponibilidade, se caracteriza pela utilização de redundância dos componentes, sejam eles de *hardware*, *software* ou outros, que manterá os serviços funcionando mesmo em casos de falhas, sem que o usuário perceba alguma alteração ou interrupção.

“Tolerância a falhas é uma abordagem pela qual a fiabilidade de um sistema de computador pode ser aumentada para além do que pode ser conseguido por meio de métodos tradicionais” (JALOTE, 1994).

2.5 SISTEMA DE ALTA DISPONIBILIDADE

Alta disponibilidade é um daqueles termos que todo mundo parece saber o que é, mas é difícil encontrar uma definição amplamente aceita e precisa (Schmidt, 2006).

Segundo Schimidt (2006), “A alta disponibilidade é a característica de proteger-se ou recuperar-se de falhas pequenas em um curto espaço de tempo com meios amplamente automatizados”.

Pode-se dizer que Sistema de Alta Disponibilidade é um sistema informatizado tolerante a falhas e que utiliza mecanismos e técnicas para manter um ou mais serviços disponíveis o máximo de tempo possível, mesmo que haja alguma mudança interna ou alguma falha, causada por erro humano ou não.

2.6 CONCEITOS IMPORTANTES

2.6.1 Failover

Nome dado ao processo no qual uma máquina assume os serviços de outra, quando esta última apresenta falha. Ele pode ser automático ou manual, sendo o automático o que normalmente se espera de uma solução de Alta Disponibilidade. Mas, em algumas aplicações não críticas o tempo de recuperação do serviço pode suportar um tempo maior, e portanto, podem ser utilizados *failover* manual. Além do tempo entre a falha e a sua detecção, existe também o tempo entre a detecção e o reestabelecimento do serviço. Grandes bancos de dados, por exemplo, podem exigir um considerável período de tempo até que atualizem os índices de suas tabelas e, durante este tempo, o serviço ainda estará indisponível.

Para o *failover* de um serviço funcionar corretamente, é necessário que as duas máquinas envolvidas possuam recursos equivalentes. Neste trabalho, as duas máquinas que estarão com o firewall *pfSense* instalados devem, possuir as mesmas configurações de *hardware* e *software*. Um recurso pode ser uma placa de rede, um disco rígido, os dados presentes no disco, e todo e qualquer elemento necessário à prestação de um determinado serviço. Caso as configurações não sejam as mesmas o reestabelecimento dos serviços podem não acontecer conforme esperado.

Dependendo da natureza do serviço, executar um *failover* significa

interromper as transações em andamento, sendo necessário reiniciá-las. Em outros casos, significa apenas um retardo até que o serviço esteja novamente disponível. Dependendo do tipo de aplicação o *failover* pode ou não ser um processo transparente.

2.6.2 Failback

É o processo de retorno de um determinado serviço após uma falha. Digamos que o servidor principal venha a sofrer com uma falha, neste caso, o servidor que no momento estava funcionando como secundário irá assumir o posto de principal e, manterá o serviço em funcionamento até que o servidor que sofreu a falha se recupere, ou seja, recuperado e volte a funcionar assumindo novamente o papel de principal.

O processo que envolve a retomada do posto pode ser automático, manual ou até mesmo indesejado. Em alguns casos, em função da possível interrupção na prestação de serviços, o *failback* pode não ser atraente. No trabalho proposto esse processo será automático.

2.6.3 Missão

Quando se calcula a disponibilidade de um sistema, é importante que se observe o conceito de missão. Missão de um sistema é o período de tempo no qual ele deve desempenhar suas funções sem interrupção. Por exemplo, uma farmácia, que funcione das 8h às 20h, não pode ter seu sistema fora do ar durante este período de tempo. Se este sistema vier a apresentar defeitos fora deste período, ainda que indesejados, esses defeitos não atrapalham em nada o andamento correto do sistema. Uma farmácia 24h obviamente tem uma missão contínua, de forma que qualquer tipo de parada deve ser mascarada.

A Alta Disponibilidade visa eliminar as paradas não planejadas. Porém, no caso da primeira farmácia, as paradas planejadas não devem acontecer dentro do período de missão. Paradas não planejadas decorrem de defeitos, já paradas planejadas são aquelas que se devem a atualizações, manutenção preventiva e atividades correlatas. Desta forma, toda parada dentro do período de missão pode ser considerada uma falha no cálculo da disponibilidade.

Uma aplicação de Alta Disponibilidade pode ser projetada inclusive para suportar paradas planejadas, o que pode ser importante, por exemplo, para permitir a atualização de programas por problemas de segurança, sem que o serviço deixe de ser prestado.

2.7 PFSYNC

O pfsync (*packet filter state table sychronisation interface*) é uma interface de rede usada para expor certas alterações feitas na tabela de estados do *pf* (OPENBSD, 2015c). A interface pfsync pode enviar mensagens de alterações de estados para a rede, de modo que outros nós executando o *pf* possam mesclar as alterações recebidas com suas próprias tabelas de estados. Da mesma forma, o *pfsync* também pode receber essas mensagens.

Por padrão, o *pfsync* não envia ou recebe atualizações da tabela de estados na rede, mais quando está configurado para enviar e receber atualizações na rede, o comportamento padrão é enviar atualizações em *multicast* na rede local e todas as atualizações serão enviadas sem autenticação, o que pode ser perigoso, por isso o mais indicado é ligar duas interfaces fisicamente com o uso de um cabo de rede.

2.8 CARP

O *Common Address Redundancy Protocol* (CARP) é um protocolo de rede que permite a um grupo de *hosts* de um mesmo segmento compartilharem um mesmo endereço de rede.

O CARP utiliza um endereço IP virtual que é compartilhado entre os nós membros de um grupo. Nesse grupo um dos nós é designado *master* e os demais *backups.* Apenas o nó *master* responderá as requisições *ARP* (PLUMES, 1982). Em tempos determinados o nó *master* publica notificações em *multcast* na rede informando seu estado. Quando por algum motivo ele parar de publicar, os *hosts backups* perceberão e o nó com maior prioridade no momento assume a função até que o nó *master* retorne ao estado de *UP*.

Ele é uma alternativa gratuita e segura para os protocolos *Virtual Router Redundancy Protocol* (VRRP) (NADAS, 2010), e o *Hot Standby Router Protocol* (HSRP) (LI, 1998) com patente pertencente a CISCO. O CARP não possui *RFC* nem

é reconhecido pela *IANA*, pois ele não passou pelas etapas de reconhecimento necessárias. Ele apenas se encaixa no *IP 112* que é o mesmo do *VRRP* e por isso os dois não podem coexistir em um mesmo seguimento de rede devido a conflitos de MAC.

O CARP se diferencia do VRRP e HSRP em alguns pontos importantes, o principal deles é a segurança. Ele foi concebido para ser seguro, por isso, utiliza um controle de segurança usando *SHA-1 HMAC* (VEGA, 2006), oculta o endereço *IP virtual* (WIKIPEDIA, 2010) dentro do pacote e realiza uma verificação de integridade e autenticidade das informações processadas pelo sistema de *Failover*. Além disso, ele já tem suporte nativo ao IPv6, dispensando a necessidade de ativação para o seu uso.

Tabela 1 – Tabela comparativa de recursos entre os protocolos CARP, HSRP e VRRP.

RECURSOS	CARP	HSRPv0	VRRPv2
Suporte a IPv6	Sim	Não	Em planejamento
Troca de mensagens autenticadas	SHA-1 HMAC	Texto limpo; Password; MD5 HMAC (somente na versão 2)	Texto limpo; Password; MD5 HMAC
Patenteado	Não	Sim	Em reivindicação

Fonte: DANHIEUX, 2004, p 11.

2.9 XMLRPC

O XMLRPC é o mecanismo de sincronização utilizado pelo protocolo CARP para que seja possível sincronizar informações e comando contidos no nó *master* com os nós *backups*. O arquivo de extensão .php contendo as funções de sincronismo pode ser encontrado no diretório /usr/local/www/, conforme mostra a figura 1.

Figura 1 – Trecho do arquivo xmlrpc.php que é responsável pelo sincronismos das informações entre os nós CARP.

```
##|+PRIV
##|*IDENT=page-xmlrpclibrary
##|*NAME=XMLRPC Library page
##|*DESCR=Allow access to the 'XMLRPC Library' page.
##|*MATCH=xmlrpc.php*
##|-PRIV

require("config.inc");
require("functions.inc");
require_once("filter.inc");
require("ipsec.inc");
require("vpn.inc");
require("shaper.inc");
require("xmlrpc_server.inc");
require("xmlrpc.inc");
require("array_intersect_key.inc");

function xmlrpc_loop_detect() {
        global $config;

        /* grab sync to ip if enabled */
        if ($config['hasync'])
                $synchronizetoip = $config['hasync']['synchronizetoip'];
        if($synchronizetoip) {
                if($synchronizetoip == $_SERVER['REMOTE_ADDR'])
                        return true;
        }
```

Fonte: Elaborado pelo autor.

2.10 PFSENSE

Criado por *Chris Buechler* e *Scott Ullrich,* foi lançado em 2004. Hoje já pode ser considerado como *Unified Threat Management* (UTM) (KASPERSKY, 2015), Central Unificada de Gerenciamento de Ameaças, devido ao seu repleto leque de recursos nativos e pacotes que podem ser instalados para aumentar o seu poder, podendo ser implantado desde pequenas redes domésticas à grandes corporações.

> PfSense é um sistema operacional de código aberto usado para transformar um computador em um firewall, roteador ou uma variedade de outros dispositivos de rede. PfSense é uma distribuição FreeBSD [FREE...2015] personalizado baseado no projeto m0n0wall, uma distribuição de firewall leve e poderoso. O PfSense foi construído sobre os fundamento do m0n0wall [M0N0WALL, 2015] e possuiu vários traços de suas funcionalidades, acrescentando uma variedade de outros serviços de redes populares (WILLIAMSON, 2011).

Seus serviços são abrangentes no quesito segurança de redes, tornando-o

uma evolução do firewall tradicional, unindo em um só lugar diversos recursos, tais como: VPN, balanceamento de carga, monitoramento de tráfego, controle de banda, HA e etc. Difere de alguns concorrentes ele é gratuito e pode ser executado a partir de *hardware* com configurações simples a grandes servidores.

3 DESENVOLVIMENTO

Aqui será apresentado o ambiente modelo para o tema proposto, além de todas as configurações necessárias para que as máquinas com pfSense possam funcionar em Alta Disponibilidade.

É importante ressaltar que o ambiente apresentado abaixo é virtual, montado com a ajuda do software VirtualBox, mais todas as configurações feitas nesse ambiente se aplicam perfeitamente ao mudo real.

Não será abordado neste trabalho a instalação de nenhuma das máquinas virtuais por entender que esse conteúdo foge do escopo. Partiremos do princípio que as máquinas virtuais já estarão instaladas de forma básica sem nenhuma configuração adicional a não ser as informadas em cada sessão.

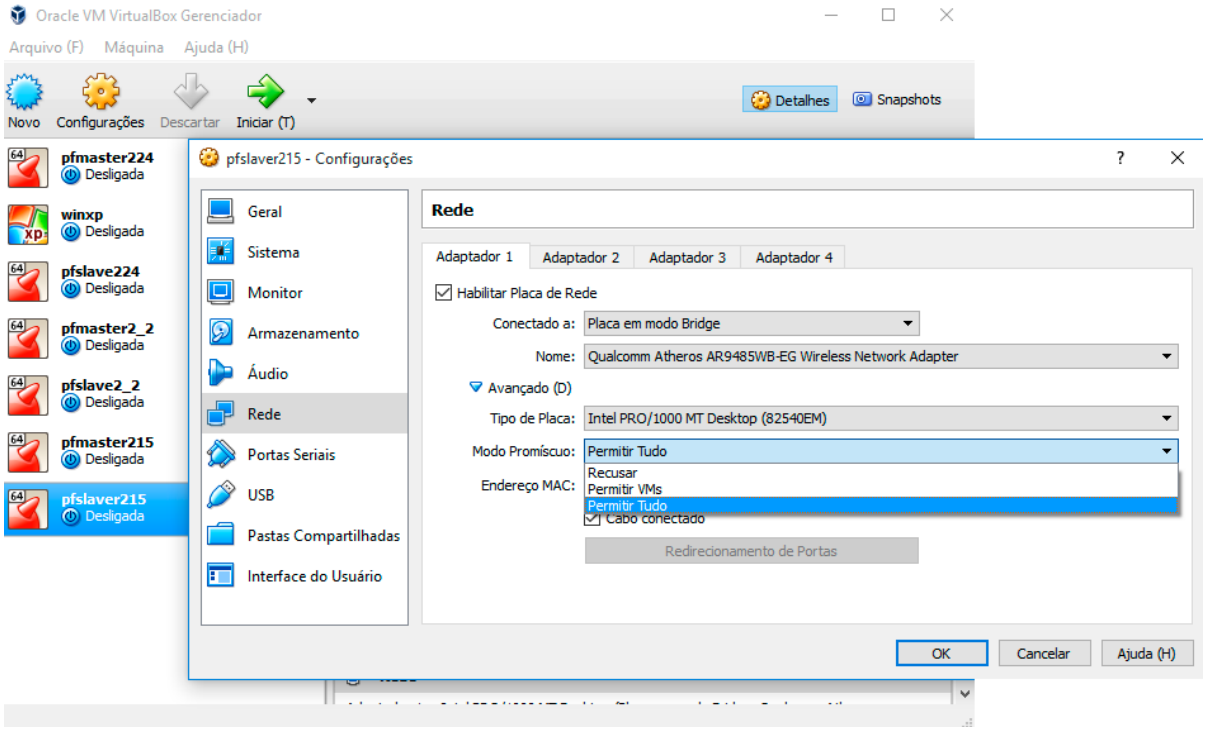
Na máquina virtual Windows XP, foi adotada a instalação básica e a adição do *browser* Firefox.

Para os firewall pfSense, adotaremos a instalação básica e limpa, não será preciso instalar nenhum complemento ou pacote adicional, mais será preciso que o firewalls já estejam com acesso à internet, por exemplo, com resposta ao PING para o site *www.google.com*. Quanto as interfaces de rede, não será mostrado como atribuir um IP a cada interface, apenas será mostrado os IP que serão utilizados para que o trabalho possa ser desenvolvido. Será mostrada em forma de imagens apenas as configuração dos *IPs Virtuais* e da configuração da HA (Alta Disponibilidade), que serão necessários para a sincronização entres os servidores, estas imagens por sua vez serão todas explicadas.

Devido a uma característica de operação, o pfSense coloca as suas placas de rede em *MODO PROMÍSCUO*[1]*,* para que seja possível coletar dados da rede necessários para o funcionamento de algumas de suas aplicações. Em máquinas reais esta configuração é feita automaticamente, mas, como este é um ambiente virtual será preciso configurar isso manualmente. No virtualBox, esta configuração é bem simples. Quando a máquina virtual desejada estiver selecionada, ficará disponível em configurações o modo promiscuo, conforme mostrado na figura 2.

[1] Modo em que placa de rede capta todos os pacotes do seu segmento de rede mesmo que os pacotes não seja destinado a ele.

Figura 2 – Ativação do modo promíscuo das placas de redes virtuais no VirtualBox.



Fonte: Elaborado pelo autor.

3.1 MÁQUINAS VIRTUAIS

3.1.1 Máquina Virtual Windows XP

Utilizada para acessar o firewall pfSense via *browser*, e assim poder realizar e testar as configurações que serão aplicadas, simulando uma estação de trabalho da rede interna.

3.1.2 Máquina Virtual PFSENSE MASTER

Funcionará como firewall principal, todas configurações para que a Alta Disponibilidade funcione serão aplicadas e criadas inicialmente nesta máquina, que posteriormente serão replicadas para servidor secundário.

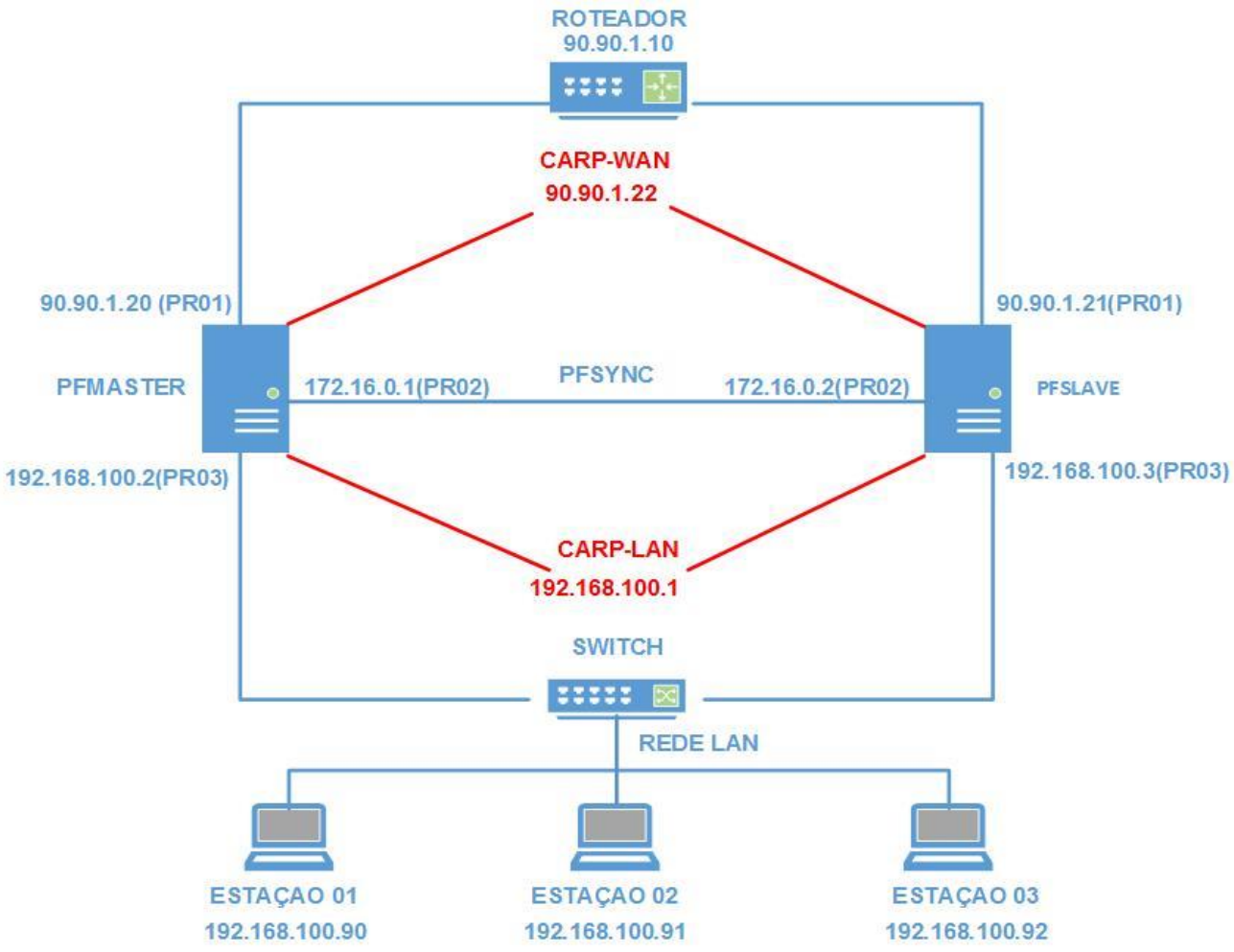
Por padrão, apenas os recursos nativos do pfSense serão sincronizados automaticamente após a configuração. Pacotes adicionais como *Squid* ou *Sarg* devem ser instalados manualmente exatamente iguais, tanto no servidor primário como no secundário, além disso, deverão ter suporte ao XMLRPC.

3.1.3 Máquina Virtual PFSENSE SLAVE

Backup do servidor principal, todas as configurações que forem realizadas no servidor principal serão automaticamente sincronizadas com ela, assim, quando houver a necessidade do servidor *slave* assumir o papel do *master,* todas as configurações estarão presente. Para que isso aconteça corretamente a máquina secundária deve ser exatamente igual a máquina principal. Essa regra se aplica tanto para as configurações de *hardware*, *software* quanto aos pacotes e atualizações instaladas.

3.2 AMBIENTE PROPOSTO

Figura 3 – Ambiente modelo do projeto de HA.



Fonte: Elaborado pelo autor.

O ambiente proposto simula a saída da rede interna para a internet. Sendo composto por um roteador que servirá de gateway para os firewall pfSense, já estes dois estão configurados em Alta Disponibilidade e serão o gateway da rede interna, que será representado pelas três estações de trabalho, tendo ainda um switch que será o concentrador da rede.

3.3 CONFIGURAÇÃO DO ROTEADOR

O roteador será utilizado para ser o intermediador entre o firewall pfSense e a rede externa (Internet). Para atender à necessidade do trabalho foi escolhido um roteador que se encaixasse nos requisitos mínimos necessários para configuração, que são:

- Função de roteamento simples;
- Uma porta WAN;
- Duas portas LAN.

O modelo utilizado foi o **N 150 MBPS** do fabricante **INTELBRAS**. Este modelo atende muito bem a necessidade, pois, possui quatro portas LAN e uma porta WAN, além de diversas outras funcionalidades que não serão utilizadas, estas devem ficar como padrão.

Por segurança não é indicado deixar equipamento com acesso direto a internet configurados de forma padrão, neste caso, como é apenas uma exemplificação e o ambiente foi montado apenas para confecção deste documento, não será necessário se preocupar com o fator segurança.

Para a rede externa (WAN) o endereço IP foi configurado para ser obtido de forma automática do provedor. Para a rede entre o pfSense e o roteador, foi colocar o endereço IP 90.90.1.10, mascara 255.255.255.0. O DHCP deve ficar desativado.

Resumo das configurações para o roteador:

- REDE WAN: IP Automático
- REDE LAN: IP 90.90.1.10/24
- DHCP: Desativado.
- Demais opções: Deixar de forma padrão.

3.4 CONFIGURAÇÃO DO SWITCH

Neste trabalho o switch é virtual e quem desempenha essa função é o próprio VirtualBox, mais caso queira montar este mesmo ambiente de modo real, há a possibilidade de ser usado um switch simples ou gerenciável, esta escolha fica a critério. A função do switch na situação proposta é apenas ser um ponto de concentração e convergência para que as estações de trabalho possam se comunicar com o firewall.

Para que as máquinas virtuais possam se comunicar, basta apenas que, as interfaces de redes estejam configuradas no mesmo modo, *REDE INTERNA* e que estejam na mesma faixa de rede.

3.5 CONFIGURAÇÃO DA ESTAÇÃO DE TRABALHO

Para instalar a máquina virtual Windows XP no virtualizado VirtualBox, foi utilizado uma *ISO*[2]*,* as configurações adotadas para rodar essa máquina virtual de forma aceitável e assim poder realizar as atividades propostas seguem abaixo:

- Quantidade de núcleos virtuais: 01 núcleo;
- Quantidade de memória virtual: 512 MB;
- Tamanho do disco virtual: 40 GB;
- Quantidade de interfaces de rede: 01 Interface de rede.

Nesta máquina foi instalado apenas o Mozila Firefox, esse *browser* foi escolhido por:

- Consumir poucos recursos;
- Ser compatível com as tecnologias web mais atuais;
- Ser compatível com diversos sistemas operacionais;
- Ser gratuito e poder ser baixado facilmente do seu site.

A interface de rede foi configurada para funcionar no modo *REDE INTERNA* disponível no VirtualBox.

Para que a máquina virtual possa acessar o *WebGUI* do pfSense é necessário que a mesma esteja na mesma faixa de rede do pfSense. Seguindo o que é apresentado na imagem do ambiente proposto a máquina virtual terá as

[2] ISO é um formato de arquivo que contém todas as informações sobre o conteúdo de um CD ou DVD, seja ele de áudio, vídeo ou dados.

seguintes configurações de rede:

- IP: 192.168.100.10
- Mascara: 255.255.255.0
- Gateway: 192.168.100.1
- DNS: 192.168.100.1

As estações de trabalho mostradas na figura do ambiente proposto serão representadas por uma máquina virtual com o sistema operacional Windows XP, que será usada para acessar a interface WebGUI do pfSense, a partir daí, será possível acessar os menus necessários e realizar as configurações. Após a conclusão das configurações no pfSense será utilizado o *Prompt de Comandos* do Windows para testar e verificar o que acontece quando o pfSense primário é desligado.

3.6 CONFIGURAÇÃO DO PFMASTER

É neste servidor onde ocorrerá a maior quantidade de configurações, ele será o mais importante do processo, pois, será o principal e, a partir dele que as informações serão publicadas, por isso, a configuração deste servidor demandará uma maior concentração, além disso, se as configurações realizadas nele não funcionarem as demais configurações também não terão efeito.

3.6.1 Checagem Inicial do PFMASTER

Inicialmente deve-se saber que em vez do identificador genérico ethX, que o Linux usa para identificar uma interface de rede, o FreeBSD no qual o pfSense é baseado, usa o nome do driver do dispositivo de rede seguido por um número como identificador, por isto, não estranhe se encontrar interface como le0, ed0 ou re0. No trabalho em questão as interfaces de rede serão exibidas como: em0, em1 e em2, sendo que é em0 será considerada a primeira interface de rede, em1 a segunda e em2 a terceira.

Todos os IPs devem estar devidamente atribuídos como mostrado na Figura 3, assim como as interfaces de rede devem estar na mesma ordem, por exemplo: se em0 estiver configurada para ser a interface WAN do servidor *pfMaster*, então a interface WAN do servidor *pfSlave* também deve ser a em0. Por isso, a necessidade do fabricante das placas de rede ser o mesmo. Caso isso não ocorra haverá

grandes chances do sincronismos não funcionar.

3.6.1.1 Interface WAN do PFMASTER

A interface WAN funciona como gateway para as demais interfaces de rede presentes no servidor. Por este trabalho ter sido desenvolvido em um ambiente virtual e esta interface pertencer a uma máquina virtual, a interface deverá trabalhar em modo *BRIDGE* e assim se comunicar com o Roteador, que por sua vez se comunicará com a internet.

A interface WAN deve ficar com as seguintes configurações:

- IP: 90.90.1.2;
- Máscara: 255.255.255.0;
- Gateway: 90.90.1.10.

Também é necessário que seja criada uma regra no firewall liberando todo o acesso de qualquer destino para qualquer destino na interface WAN, para que não haja nenhum bloqueio de tráfego de dados durante a configuração.

Por medidas de segurança, não é aconselhável que haja regras liberando todo o tráfego na interface WAN, isto foi feito apenas porque no ambiente de trabalho apresentado a interface WAN não é ligada diretamente à Internet e sim ao Roteador, que neste caso será o responsável por intermediar todo o tráfego que entra e sai com destino ao meio externo.

3.6.1.2 Interface LAN do PFMASTER

Será usada para configurar grande parte do que é preciso para deixar tudo funcionando. É necessário que a interface esteja no modo *REDE INTERNA,* para que seja possível se comunicar com a estação de trabalho Windows XP.

O IP desta interface deve ser:

- IP: 192.168.100.2;
- Máscara: 255.255.255.0.

3.6.1.3 Interface PFSYNC do PFMASTER

Usada apenas para realizar o sincronismo entre os servidores *pfMaster* e *pfSlave*. É aconselhável não só por medidas de segurança mais também por questões de consumo de tráfego de rede, e até mesmo para evitar problema de compatibilidade com alguns modelos *Switchs* gerenciáveis, que a ligação entre as interfaces PFSYNC seja feita com o um cabo crossover quando as placas de rede forem no padrão 10/100 ou com um cabo Categoria 5e caso as placas sejam do padrão Gigabit 10/100/1000.

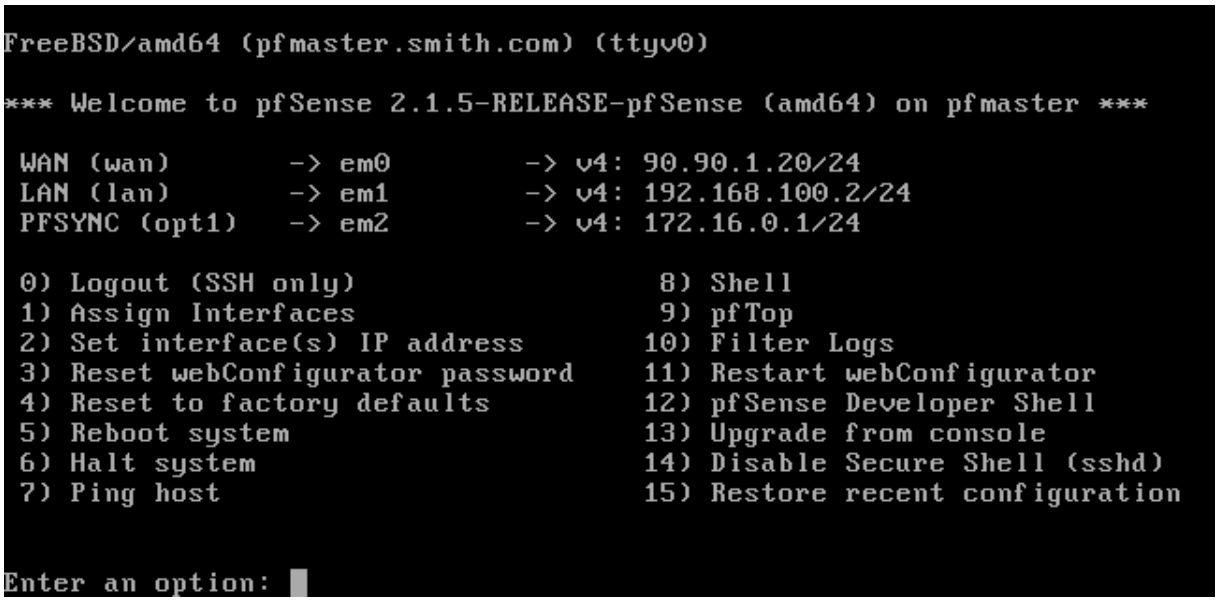
Neste trabalho não será necessário a utilização de nenhum cabo de rede, pois isto será feito automaticamente pelo VirtualBox, precisando apenas que esta interface esteja configurada no modo *REDE INTERNA.*

Esta interface deve ficar com o seguinte IP:

- IP: 172.16.0.1
- Máscara: 255.255.255.0

Após a aplicação dos IPs nas interfaces, a tela do console do *pfMaster* será semelhante à mostrada na figura 4.

Figura 4 – Console do pfMaster já com os IP configurados.

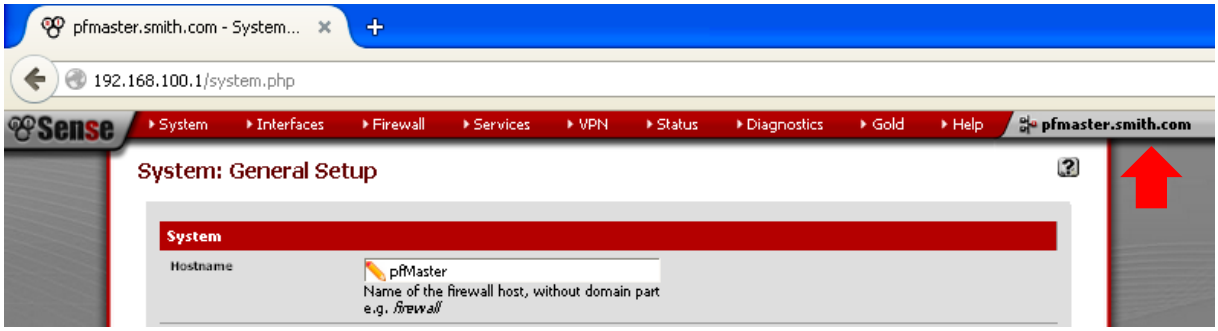


Fonte: Elaborado pelo autor.

3.6.1.4 Nome do Servidor Primário

Para uma melhor identificação e organização, o servidor primário foi renomeado para *pfMaster*, assim, ao acessa-lo via *browser* o nome aplicado será mostrado no canto superior direto da tela de configuração, facilitando a identificação, como mostrado na Figura 5.

Figura 5 - Verificação do nome do pfMaster.

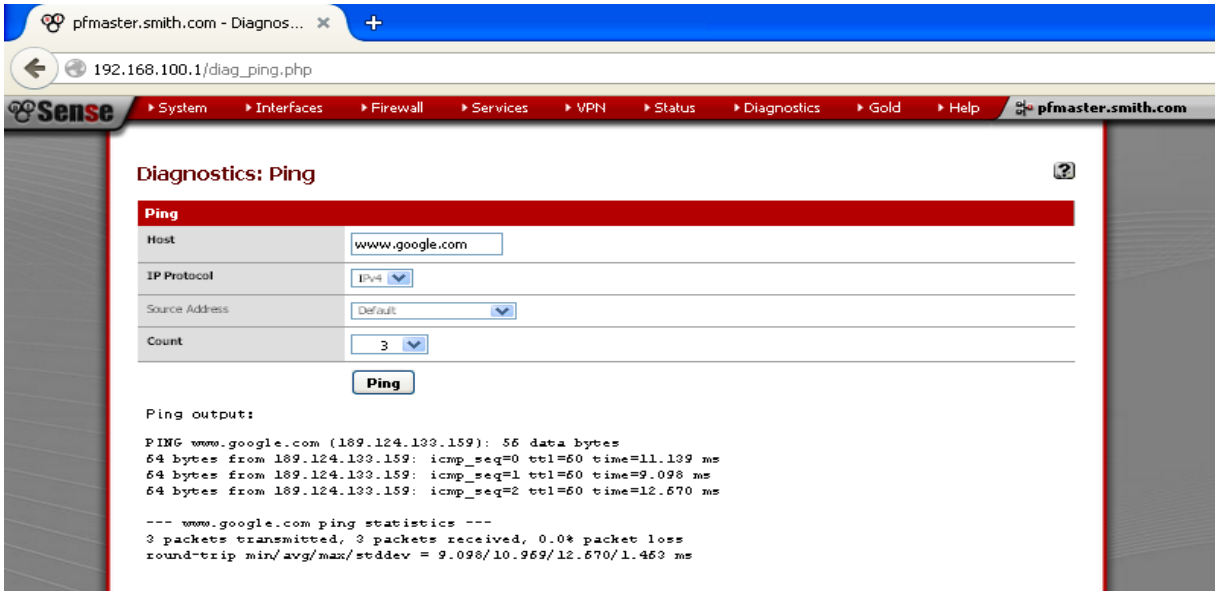


Fonte: Elaborado pelo autor.

3.6.1.5 Acesso à Internet a partir do PFMASTER

O servidor *pfMaster* deve estar com acesso à internet e respondendo ao *ping* a algum site da internet, por exemplo: *www.google.com*, como mostra a figura 6.

Figura 6 – Teste de ping do pfMaster para o site do google.



Fonte: Elaborado pelo autor.

3.6.2 Configuração dos IPs VIRTUAIS do PFMASTER

Será utilizado a máquina virtual Windows XP para acessar o *pfMaster* através do endereço 192.168.100.1, usando o navegador Firefox. Após o *login,* deve-se acessar a tela do *Virtual IP* localizada no menu *Firewall.*

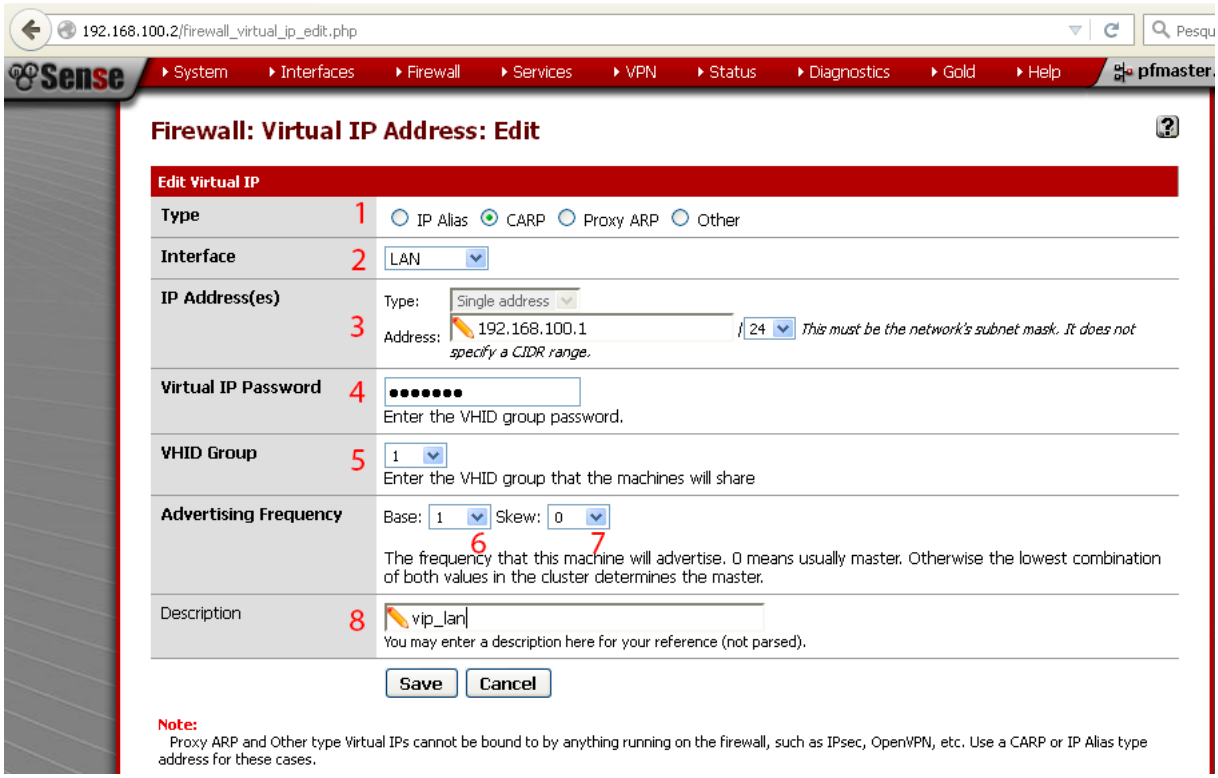
Figura 7 – Tela de configuração do Virtual IP do pfMaster.

Fonte: Elaborado pelo autor.

3.6.2.1 Configuração do IP VIRTUAL da Interface LAN do PFMASTER

Até o momento não há nenhum *IP Virtual* conforme mostra a figura 7, mais será preciso criar um para a interface WAN e outro para interface LAN. É indicado, por questões de organização, criar primeiro o *IP Virtual* da interface LAN. A figura 8, exibe a tela de configuração com os parâmetros necessários para a criação do *IP virtual*.

Figura 8 - Tela de configuração do IP Virtual da interface LAN.



Fonte: Elaborado pelo autor.

As opções numeradas na Figura 8 são as seguintes:

1 - Define o tipo de interface a ser usada. O tipo CARP permite que o *IP Virtual* seja compartilhado. Para esta configuração deve ser escolhido tipo CARP.

2 - Interface onde o *IP Virtual* será aplicado. Como é a interface LAN que está sendo configurada é ela que deve ser selecionada.

3 - *IP Virtual* que será compartilhado entre os nós do mesmo segmento de rede, ele será o gateway para todas as estações de trabalho da rede interna, a máscara de rede também deve ser igual à do segmento de rede interna.

4 - Senha usada na autenticação entre os nós grupo.

5 - Define um identificador para o grupo. Cada grupo deverá ter um identificador diferente. Pode existir vários grupos em um mesmo segmento de rede desde que possuam *IDs* diferentes.

6 - Tempo em segundos que as informações serão atualizadas entre os nós.

7 - Nível de prioridade dos nós, este valor define quem será o nó mestre ou o nó escravo. O valor 0 (zero) é o de maior prioridade.

8 - Descrição da interface virtual. Usar uma descrição que facilite facilmente a

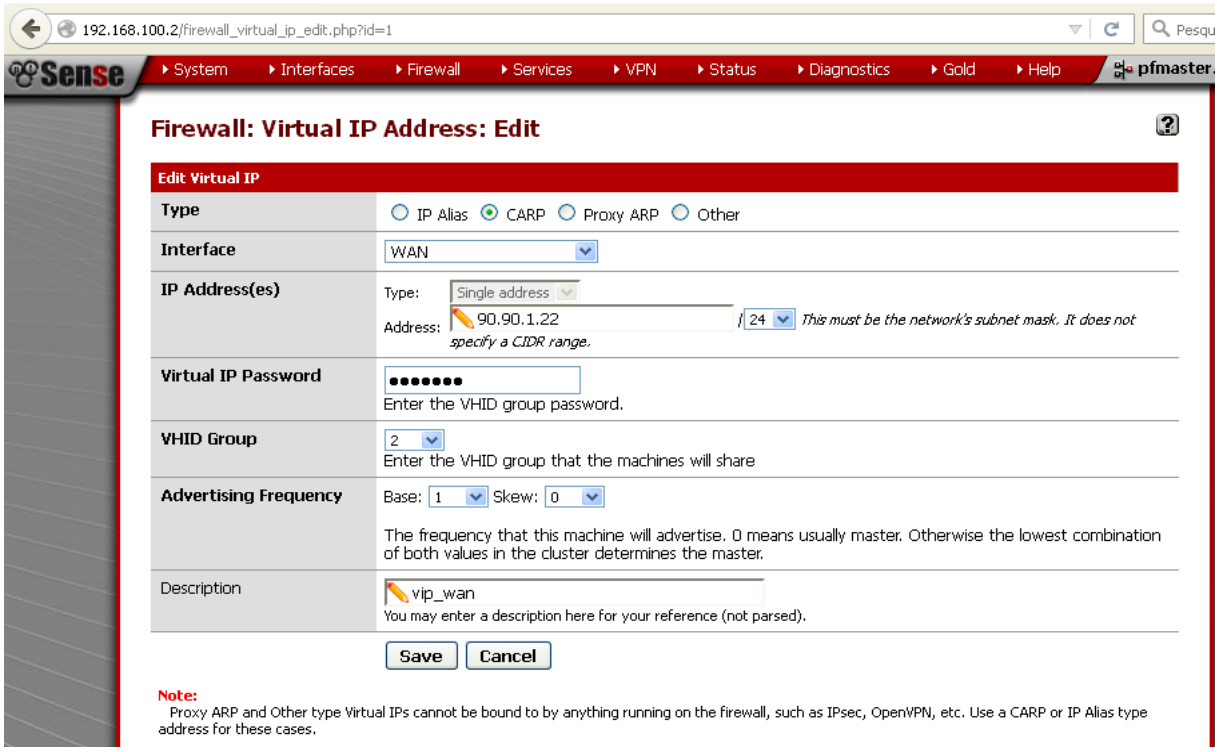
interface. Para a interface LAN a descrição é *"vip_lan"*.

O *Virtual IP* da interface LAN deve ficar igual a Figura 9 para que tudo possa funcionar.

3.6.2.2 Configuração do IP VIRTUAL da Interface WAN do PFMASTER

Já com *IP Virtual* da LAN criado, agora será necessário criar o da interface WAN, serão praticamente os mesmos parâmetros mostrados na sessão 3.5.2.1. Na interface WAN, o *endereço IP* será 90.90.1.22, *id do grupo* terá o valor 2 e "*IP_WAN*" como *descrição*. Após a criação, a interface deverá ser semelhante à mostrada na Figura 9.

Figura 9 – Tela de configuração do IP Virtual da interface WAN.



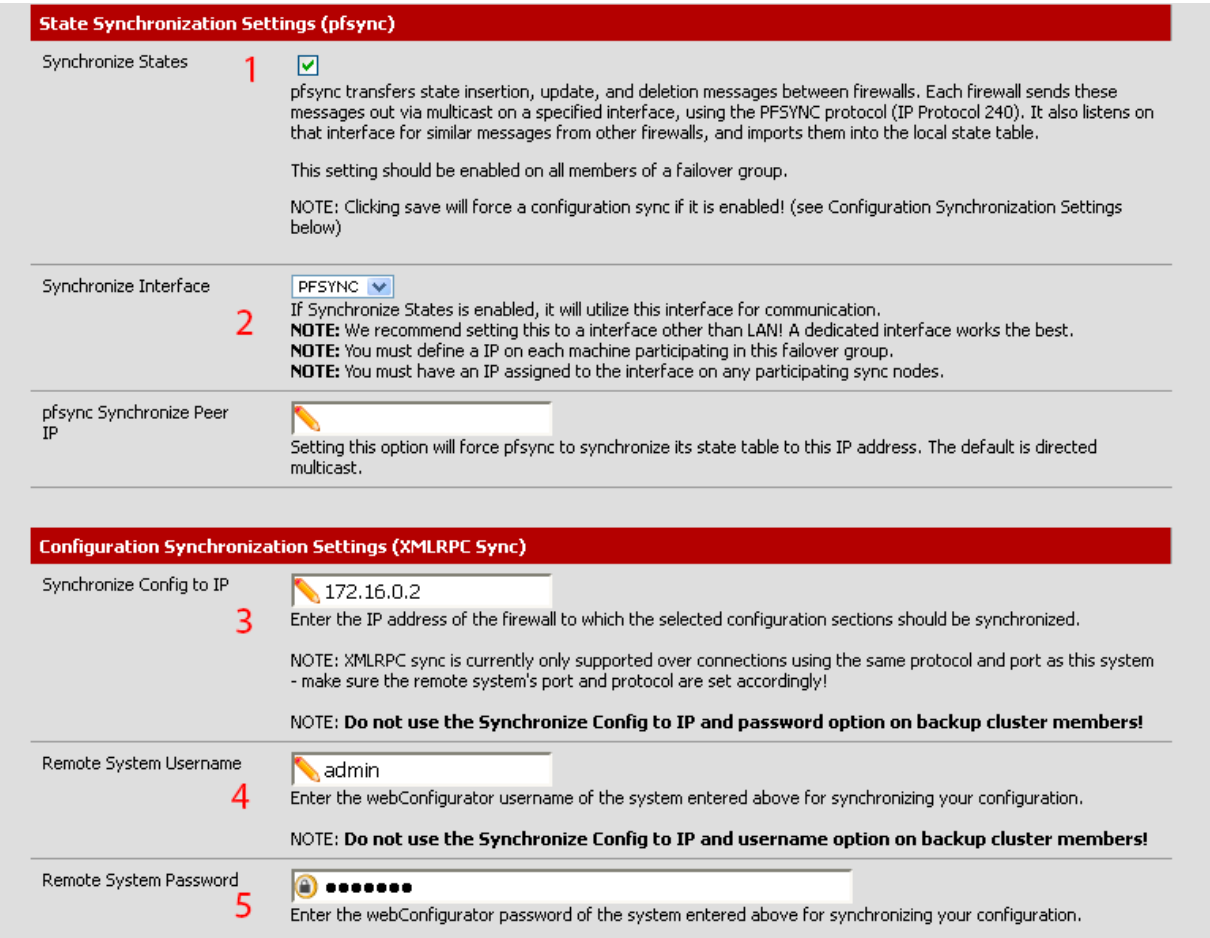
Fonte: Elaborado pelo autor.

6.2 Configuração do sincronismo do PFMASTER

Por padrão vem desativado, neste caso, será necessário ativa-lo. É possível chegar a tela de configuração por meio de dois caminhos, o submenu *High Availability* Sync disponível no menu *System* será adotado por ser o mais curto. A

tela de configuração é mostrada na Figura 10.

Figura 10 – Tela de configuração do sincronismo do pfMaster.



Fonte: Elaborado pelo autor.

As opções numeradas na Figura 10 são as seguintes:

1 – Esta opção ativa o sincronismo. Se a intenção for ativar, deve-se deixar este *checkbox* marcado;

2 – Define a interface que será utilizada para o sincronismo das informações. Neste trabalho a interface selecionada deve ser a PFSYNC;

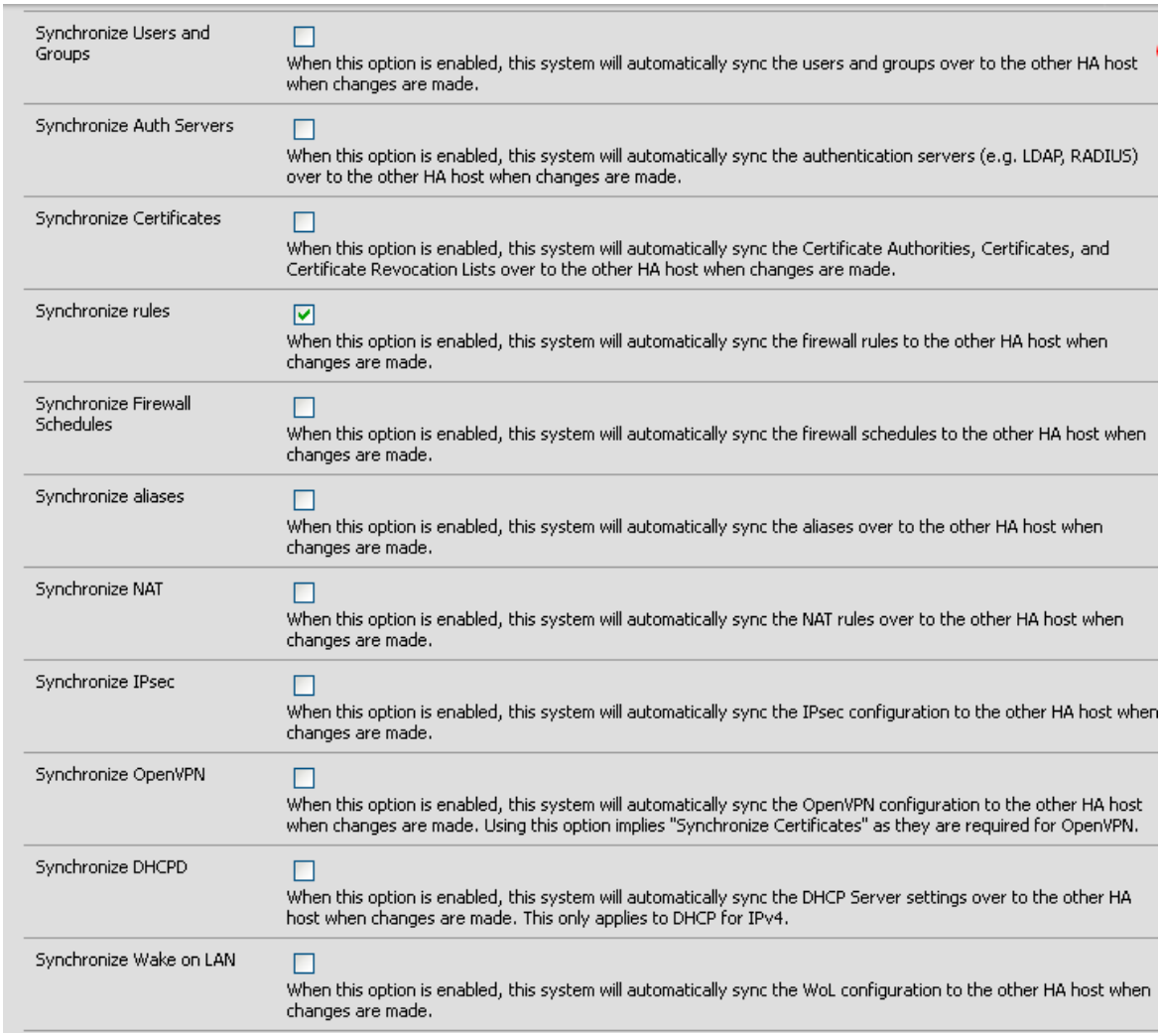
3 – Define o IP da interface de sincronismo do servidor *backup* como o qual as informações serão sincronizadas. Neste caso coloque o IP 172.16.0.2, que é o da interface PFSYNC do *pfSlave*;

4 – Usado para a autenticação feita entre os nós. Deve se colocar neste campo o usuário *root* de acesso ao *WebGUI* do servidor *backup*, neste caso o usuário do pfSlave é "*admin".*

5 – Senha usada para autenticação feita entre os nós. Mesma senha do

usuário *root*, utilizado para acessar o *WebGUI* do *pfSlave*. Foi usado usuário e senha padrão, “*admin”*, para o usuário e “*pfsense”* para senha.

Figura 11 – Lista de opções nativas do pfMaster que podem ser sincronizadas.



Fonte: Elaborado pelo autor.

A figura 11 mostra as opções que podem ser sincronizadas nativamente entre os nós. Foi ativado apenas duas dessas opções, *Synchronize rules*, ativa o sincronismo das regra de firewall, e *Synchronize Virtual IPs*, sincroniza os *IPs virtuais*.

Com essas configurações o sincronismo já vai está ativo mas, ainda não vai ser possível a sincronização, porque o servidor secundário, *pfSlave,* ainda não foi configurado.

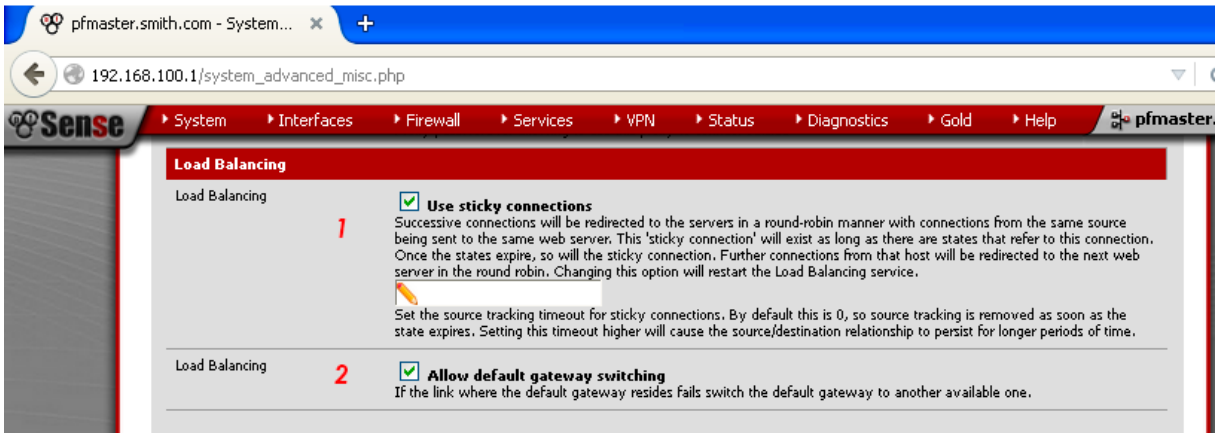
3.6.4 Load Balance

Um dos grandes diferenciais do pfSense. De forma extremamente fácil é possível ativar o *Load Balance*. Este recurso é imprescindível na configuração. Em ambientes de Alta Disponibilidade as falhas no sistema devem ser transparentes para o usuário final.

Caso as opções mostradas na figura 12 não estejam ativadas, em um eventual desligamento do servidor *pfMaster*, todas as sessões que estavam abertas terão que ser refeitas pelo usuário, sendo possível a percepção da falha. Por exemplo, um pagamento que esteja sendo feito em uma página banco online terá que ser todo refeito.

Ativa-lo torna possível ao servidor secundário manter as sessões abertas em um possível desligamento do servidor primário. A figura 12 mostra as opções que devem ser marcadas. O caminho *Systems/Advanded/Miscellaneous*, submenu *Load Balancing* leva a tela de configuração do *Load Balance.*

Figura 12 – Tela de ativação do LoadBalance



Fonte: Elaborado pelo autor.

Em ambiente com apenas dois servidores, *master e slave,* apenas o principal precisar ativar o *Load Balance*.

As opções numeradas na Figura 12 são as seguintes:

1 – Ativa o algoritmo *Round-Robin*[3]. Redes onde há mais de um gateway, geralmente apresenta problemas quando precisam acessar sites que trabalham com

[3] Round Robin é uma expressão usada para situações que não deva haver hierarquia. Em tecnologia da informação esta expressão tenta se reproduzia em forma de algoritmo.

sessões a partir do IP de origem, este problema é causado devido as sucessivas conexões que são feitas a partir da rede interna com destino a um mesmo endereço *web*, ocorre com maior frequência em sites de bancos. Como há mais de um gateway, quando uma estação de trabalho da rede interna faz uma solicitação ao site do banco, cada gateway tenta abrir uma conexão, com isso os sites dos bancos podem achar que estão sendo invadidos e optam por fechar a sessão. Mais com opção *"use stricky connection"* ativa o *packet filter* por meio do algoritmo *Round-Robin* tenta manter o estado das conexões relacionado a apenas um endereço, criando uma relação gateways máquina, assim quando um dos gateways cair o outro assumirá e o estado ainda será mantido.

2 – Esta opção permite a troca automática do gateway padrão para o gateway *backup* em caso de falhas, mantendo as sessões que por eventualidade estejam abertos.

3.7 CONFIGURAÇÃO DO PFSLAVE

Este servidor funcionará como *backup* do primário, mais em nenhum momento será menos importante, porque se ele não estiver funcionando perfeitamente quando o servidor primário falhar ou precisar parar, todo o esforço realizado terá sido perdido. Com isso, o *pfSlave* será o *backup* de segurança em caso de um eventual problema, tornando-se uma peça fundamental do projeto.

3.7.1 Checagem inicial do PFSLAVE

Assim como no *pfMaster,* todos os IPs do *pfSlaver* devem ter sido devidamente atribuídos como mostrado na *figura 3*. A sequência das interface de rede no *pfSlave* deve ser exatamente igual à do servidor primário, caso contrário o sincronismo não funcionará. Deve-se atribuir *em0* para a interface WAN, *em1* a interface LAN e *em2* a interface PFSYNC.

3.7.1.1 Interface WAN do PFSLAVE

A interface WAN, como dito antes, funciona como o gateway para as demais interfaces de rede presentes no servidor. Deverá trabalhar em modo *BRIDGE*, assim

poderá se comunicar como o Roteador, que por sua vez se comunicará com a internet.

A interface WAN deve ficar com as seguintes configurações:

- IP: 90.90.1.3/24;
- Máscara: 255.255.255.0;
- Gateway: 90.90.1.10.

A regra de firewall liberando todo o acesso de qualquer destino para qualquer destino na interface WAN, para que não haja nenhum bloqueio de tráfego de dados durante a configuração, também deve ser criada.

3.7.1.2 Interface LAN do PFSLAVE

Trabalhará no modo *REDE INTERNA* para que seja possível se comunicar com a estação de trabalho Windows XP, sendo possível acessar o *WebGUI* do *pfSlave* e realizar as configurações.

O IP desta interface deve ser:

- IP: 192.168.100.3;
- Máscara: 255.255.255.0.

3.7.1.3 Interface PFSYNC do PFSLAVE

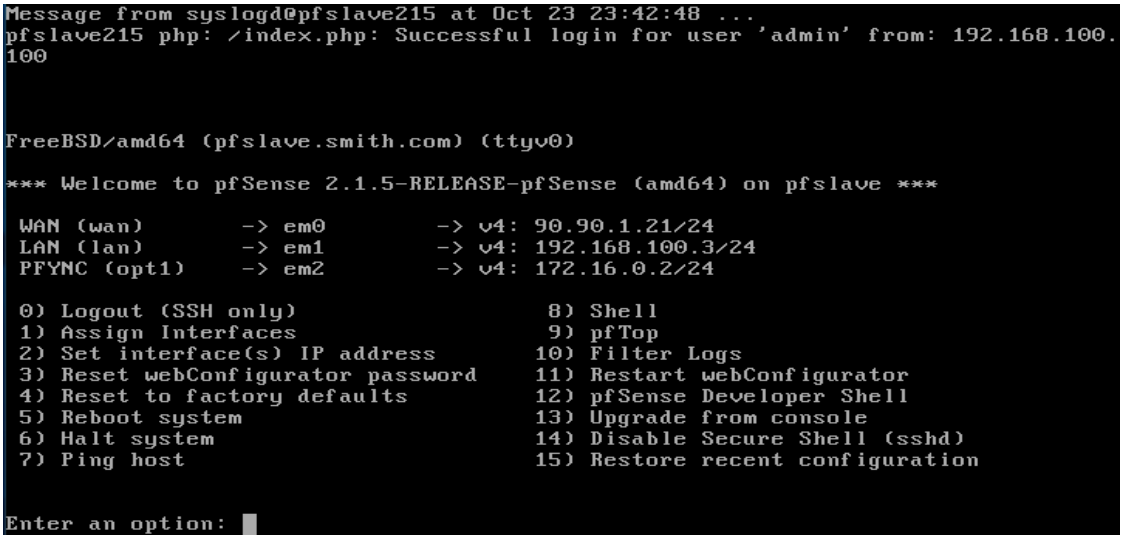
Utilizada para o sincronismo entre os servidores, a interface PFSYNC do pfSlave deve estar na mesma faixa de rede do pfMaster, assim como configurada no mesmo modo de operação, neste caso REDE INTERNA.

Esta interface deve ficar com o seguinte IP:

- IP: 172.16.0.2
- Máscara: 255.255.255.0

Depois de atribuir os IPs, o console do pfSlave apresentará um tela semelhante a apresentada na figura 13.

Figura 13 – Console do pfSlave já com os IPs configurados.

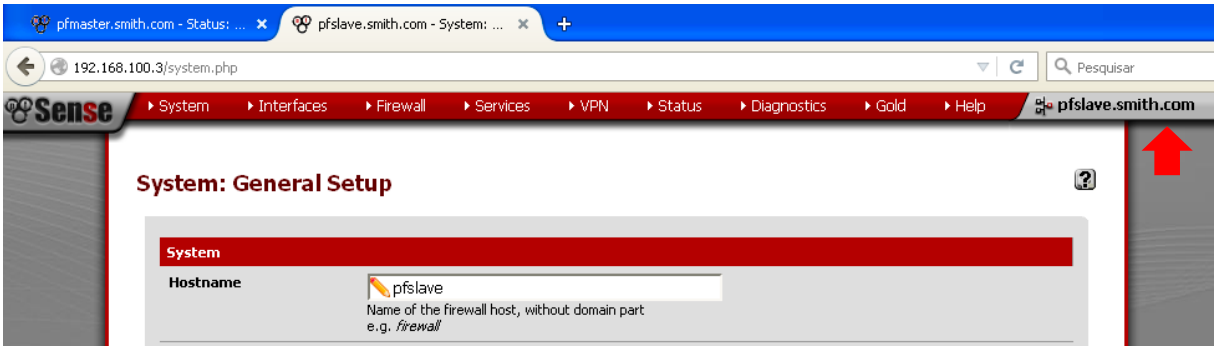


Fonte: Elaborado pelo autor.

3.7.1.4 Nome do Servidor Secundário

Por questão de identificação e organização, o servidor secundário foi renomeado para *pfSlave*, assim, ao acessa-lo via *browser* o nome aplicado será mostrado no canto superior direto da tela do *WebGUI*, facilitando a identificação, como mostrado na figura 14.

Figura 14 – Verificação do nome do pfSlave.

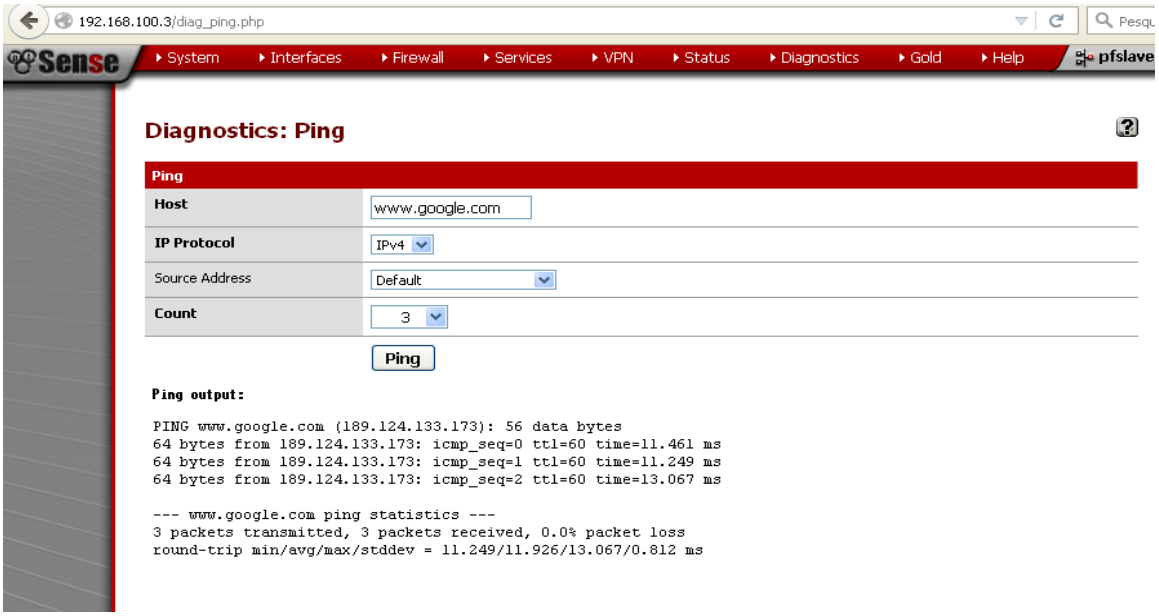


Fonte: Elaborado pelo autor.

3.7.1.5 Acesso à internet a partir do PFSLAVE

O servidor *pfSlave* deve estar com acesso à internet e respondendo ao *ping* a algum site da internet, por exemplo: *www.google.com*, como mostra a figura 15.

Figura 15 – Teste de ping do pfSlave para o site do google.



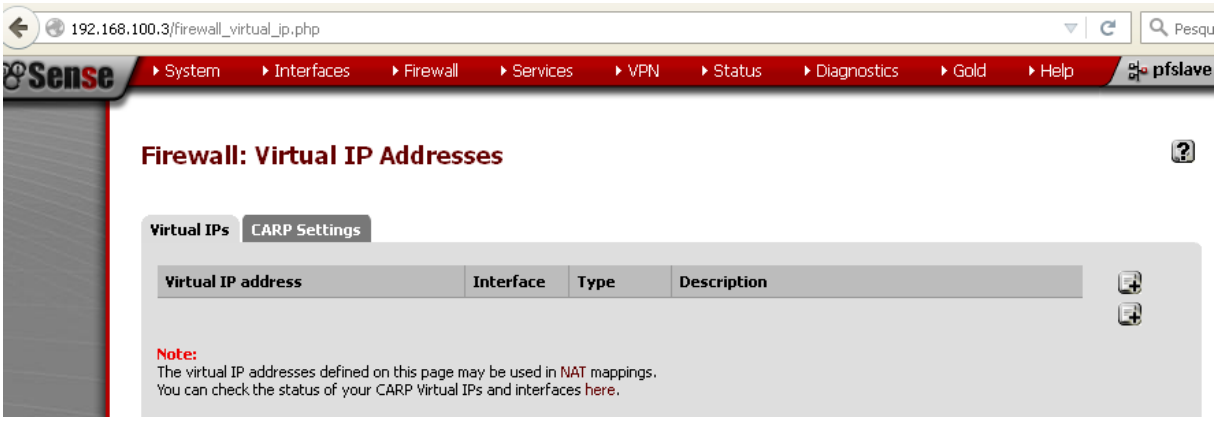
Fonte: Elaborado pelo autor.

3.7.2 Configuração dos IPs VIRTUAIS do PFSLAVE

Não será necessário criar os *IPs virtuais* no *pfSlave*, eles serão criados automaticamente quando o sincronismo for ativado, isso se as configurações estiverem corretas. Lembrando que as interfaces devem estar do jeito que foi solicitado, as regras de firewall necessárias criadas e o acesso à internet funcionando.

No momento a tela de configuração deve ser deixada como padrão, como mostrado na figura 16.

Figura 16 – Tela de configuração do IP Virtual do pfSlave



Fonte: Elaborado pelo autor.

3.7.2.1 Configuração do IP VIRTUAL da interface LAN do PFSLAVE

Não será necessário criar os *IPs Virtuais* no *pfSlave* LAN, eles serão criados automaticamente quando o sincronismo for ativado, caso as configurações estiverem corretas.

3.7.1.2 Configuração do IP VIRTUAL da interface WAN do PFSLAVE

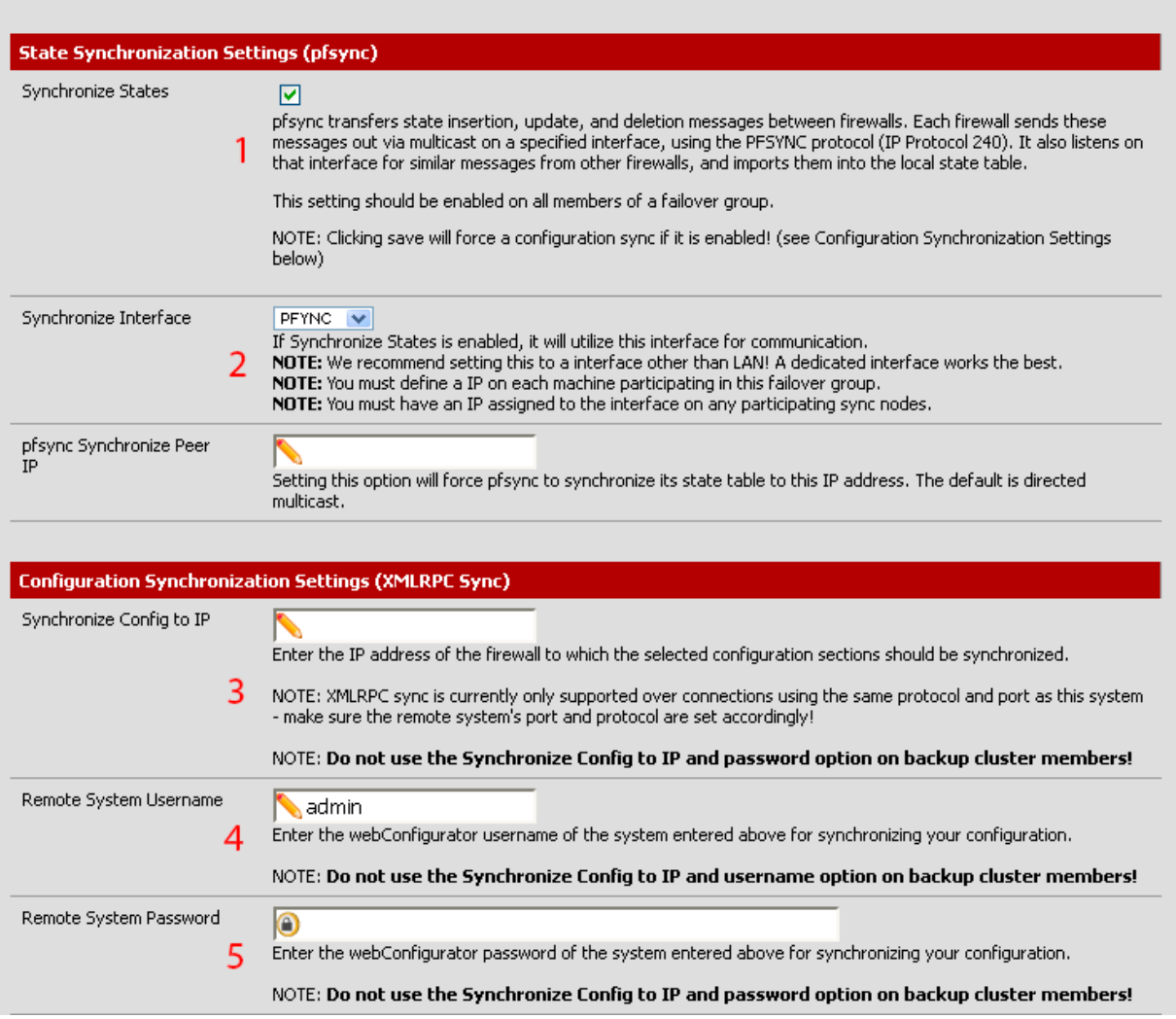
Também não será necessário criar os *IP Virtual* no *pfSlave* para a interface WAN, eles serão criados automaticamente quando o sincronismo for ativado.

3.7.3 Configuração do sincronismo no PFSLAVE

Após a configuração que será mostrada neste tópico, o sincronismo da Alta Disponibilidade começará a funcionar segundos depois de ativado, os *IPs Virtuais* já estarão presente no *pfSlave.*

A máquina virtual Windows XP deverá ser usada para acessar o *WebGUI* do *pfSlave* via Firefox no endereço 192.168.100.3. Após o *login,* a opção *High Availability Sync* do menu *System* levará a tela de ativação do sincronismo, como mostrado na figura 17.

Figura 17 – Tela de configuração da Alta Disponibilidade do pfSlave.



Fonte: Elaborado pelo autor.

As opções numeradas na Figura 17 são as seguintes:

1 – Ativa o sincronismo, deve-se deixar este *checkbox* marcado;

2 – Interface que será utilizado para o sincronismo das informações. A interface selecionada deve ser a PFSYNC;

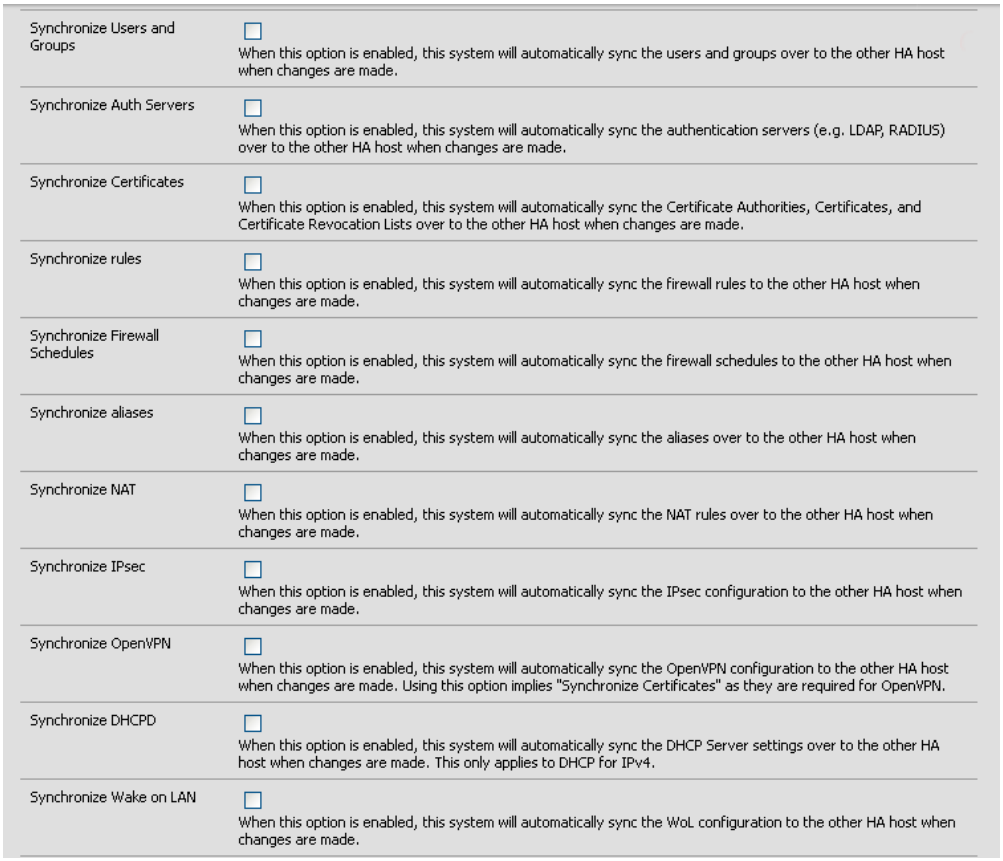
3 – Define o IP da interface de sincronismo com o qual o servidor ira sincronizar as informações. No *pfSlave* não será necessário colocar nada, este campo deve ficar em branco, porque em ambiente com dois servidores, primário e secundário, se o primário cair restará apenas o secundário, e ele não vai ter com quem sincronizar, então este campo não será utilizado.

4 – Usuário usado na autenticação feita entre os nós. Deve ser o mesmo usuário do pfMaster. Seguindo a mesma lógica do item 3 desta sessão, não se faz necessário preencher um campo que não vai ser utilizado. Então, este campo deve

ficar como padrão;

5 – Senha usada na autenticação feita entre os nós. Também segue a mesma lógica do item 3 dessa sessão. Este campo deve ficar em branco.

Figura 18 – Lista de opções nativas do pfSlave que podem ser sincronizadas.



Fonte: Elaborado pelo autor.

A Figura 18 mostras as opções que podem ser sincronizadas nativamente entre os nós. Aqui também não marcaremos nada, todos os *checkbox* ficarão desmarcados. Por fim, salve as configurações para o sincronismo começar a funcionar.

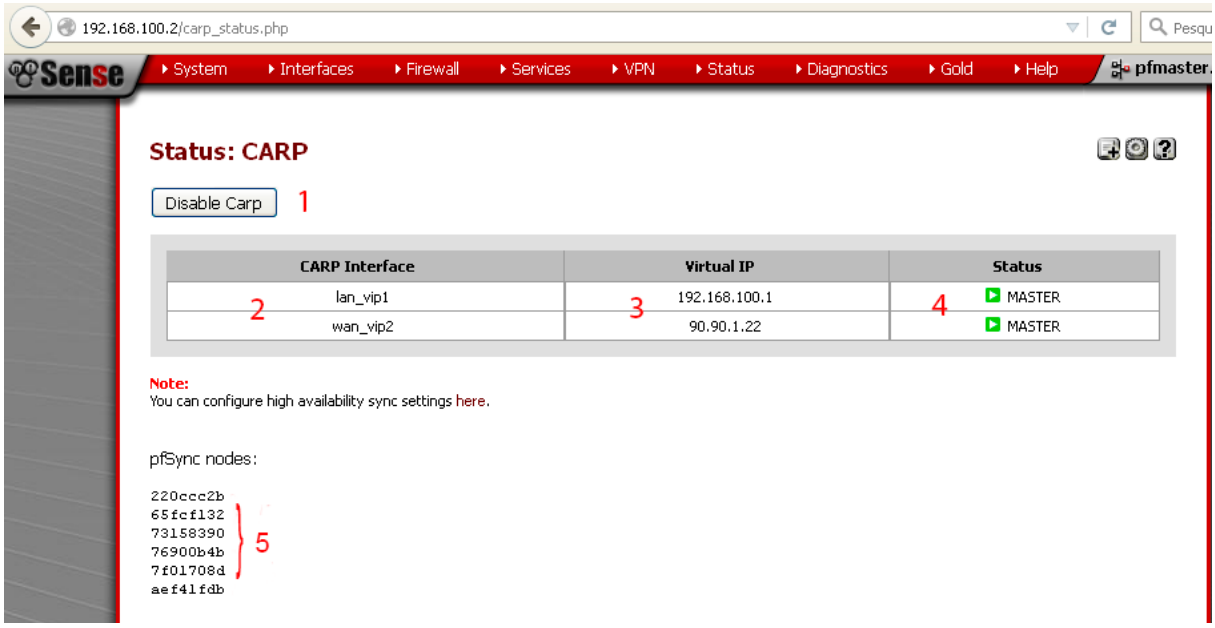
3.8 TESTES

Após tudo está devidamente configurado, tem-se agora que verificar e testar tudo que foi feito e ver ser está funcionando como esperado. Depois do sucesso nas configurações o ambiente em Alta Disponibilidade com pfSense estará pronto.

3.8.1 Status CARP do PFMASTER

O pfSense oferece um menu com opções onde é possível verificar os status de alguns recursos disponíveis nele, facilitando o trabalho de *debug* em algumas situações, o CARP é um deles. A tela de status está localizada no menu *Status/CARP (failover),* semelhante à figura 19, que mostra uma tabela com informações sobre o funcionamento do protocolo CARP.

Figura 6 – Status do protocolo CARP do pfMaster



Fonte: Elaborado pelo autor.

As opções numeradas na Figura 17 são as seguintes:

1 – Botão desativar, pode ser usado para desativar temporariamente o serviço CARP, funciona um *Stand By*. Não é indicado o seu uso e não será utilizado neste trabalho;

2 – Lista de interface CARP, lista as interfaces virtuais criadas;

3 – *IPs virtuais,* exibe os *IPs virtuais* que foram atribuídos a cada interface virtual;

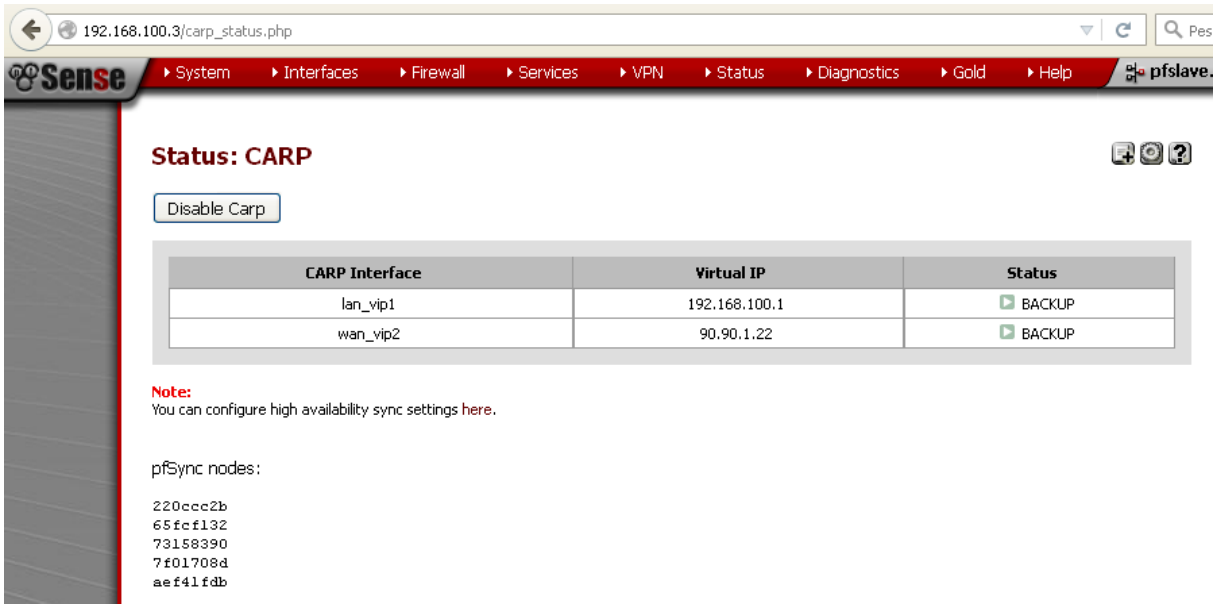
4 – Status, exibe um ícone com status do protocolo referente ao servidor acessado. Se o servidor for o *master* o ícone será verde, se for *slave* o ícone será cinza. Neste caso, o servidor é o *master* e o ícone é verde.

5 – Identificador dos nós, gerado automaticamente pelo pfSense.

3.9 Status CARP do PFSLAVE

Assim como foi verificado o status do *pfMaster*, também deve-se verificar o status do *pfSlave*. A tela apesenta praticamente as mesmas informações, a diferença é que a coluna *Status* agora mostra o nome *backup* ao lado do ícone *cinza,* como esperado.

Figura 20 – Status do protocolo CARP do pfSlave.



Fonte: Elaborado pelo autor.

3.10 TESTANDO OS IP VIRTUAIS

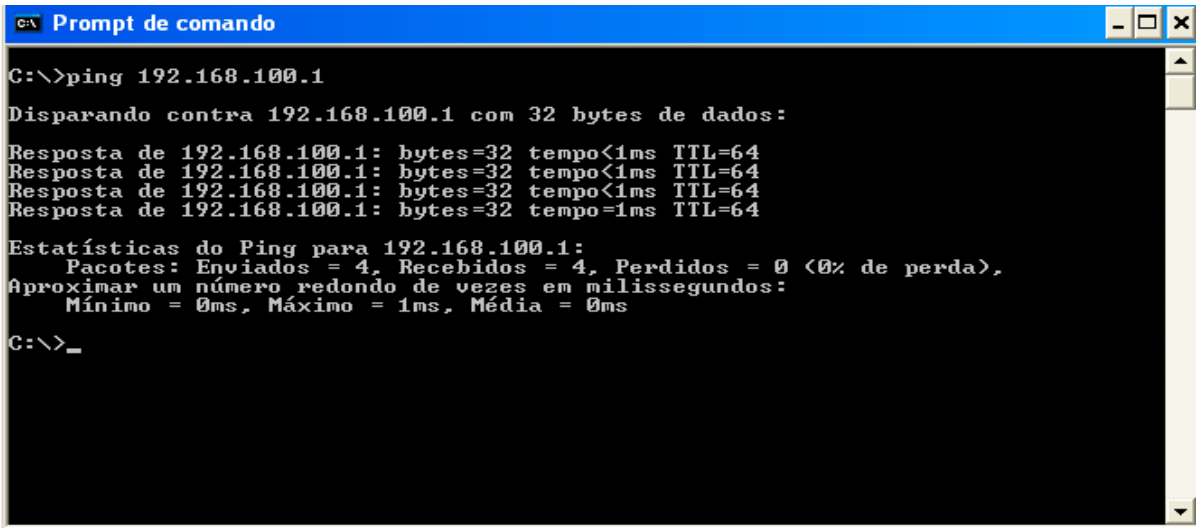
Foi visto que o status dos servidores está como esperado, agora é preciso saber se os *IPs Virtuais* estão respondendo as solicitações. Será utilizado o *ping*, um comando utilizado para testar a conectividade entre equipamentos de rede utilizando o protocolo ICMP, presente em praticamente todos os sistemas operacionais existentes hoje. Como foi criado dois *IPs Virtuais*, um para a interface LAN e outro para interface WAN, será preciso testar todos os dois e saber se eles estão funcionando.

3.10.1 Testando IP VIRTUAL da interface LAN

Pode-se usar o teste de *ping* no *prompt* de comando do Windows para

verificar se o *IP Virtual*, 192.168.100.1, da interface LAN está funcionando. É esperado que o resultado seja semelhante ao mostrado na figura 21.

Figura 21 – Teste de PING para o IP virtual da interface LAN.

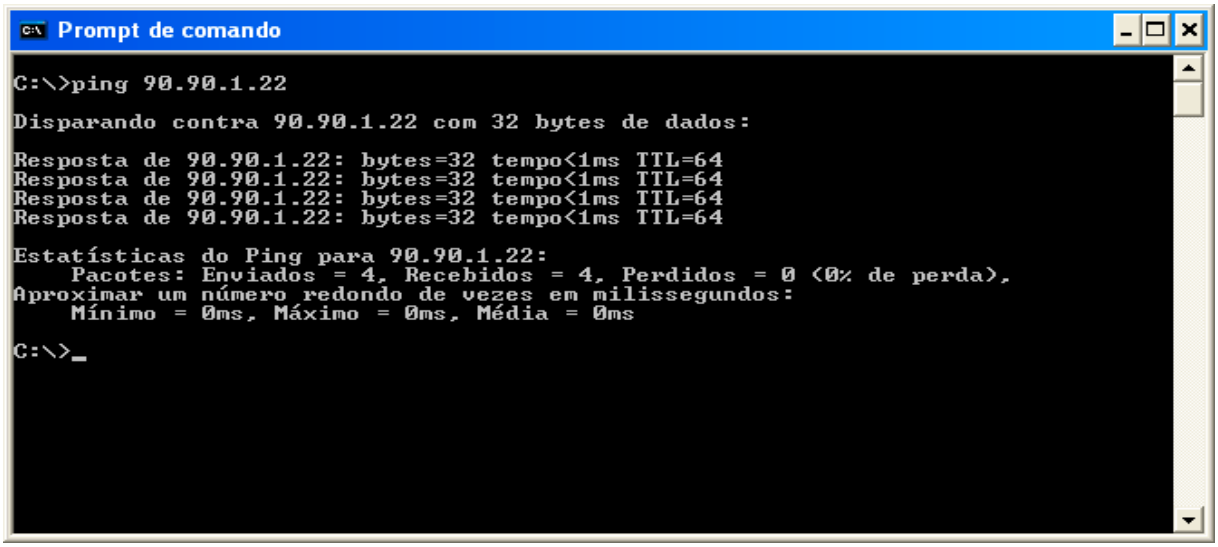


Fonte: Elaborado pelo autor.

3.10.2 Testando IP VIRTUAL da interface WAN

Agora o teste de *ping* será realizado para o endereço 90.90.1.22, que é o da interface WAN. O procedimento será praticamente o mesmo adotado no teste da interface LAN. Espera-se que o resultado seja semelhante ao da figura 22.

Figura22 – Teste de ping para o IP Virtual da interface WAN.



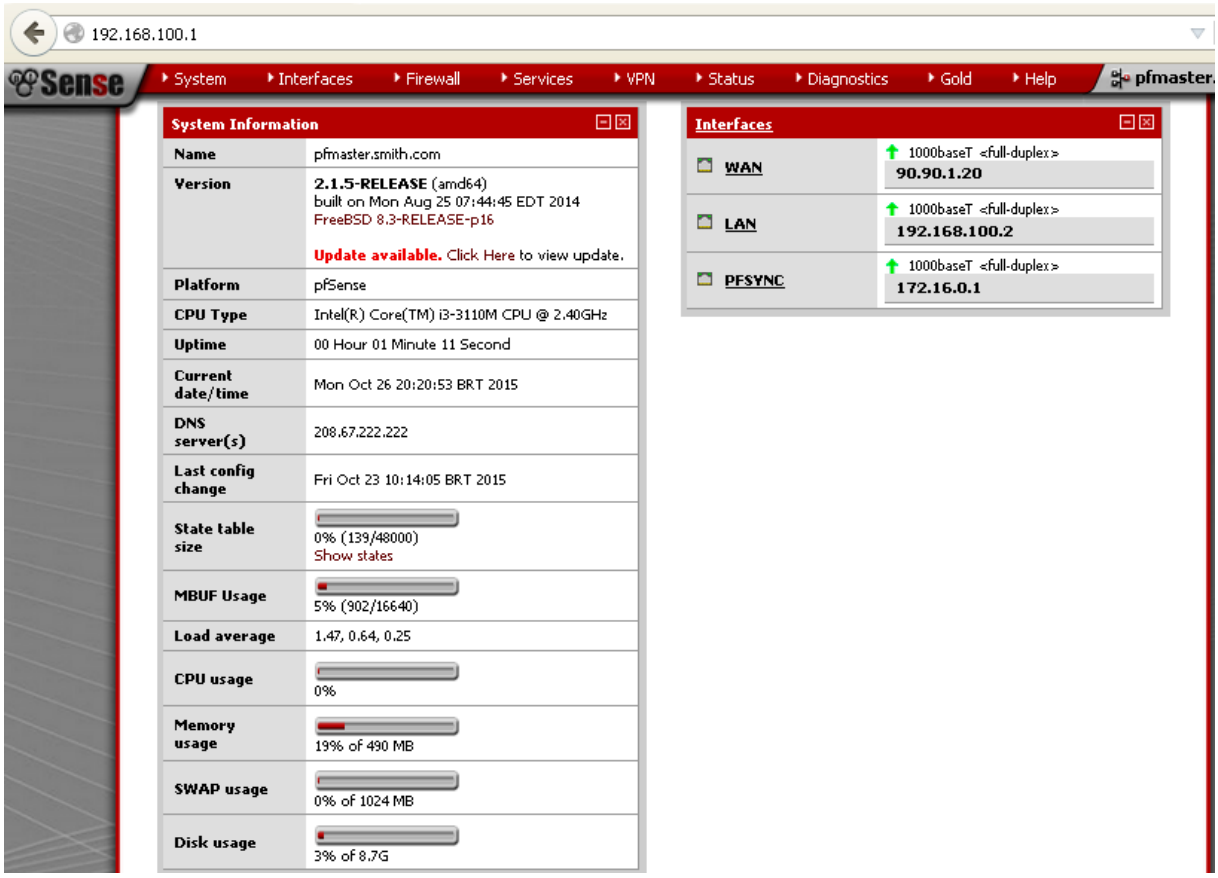
Fonte: Elaborado pelo autor.

3.11 ACESSANDO O WEBGUI PELO IP VIRTUAL

Como os *IPs Virtuais* funcionando como esperado, todas as configurações que precisarem ser feitas devem ser realizadas no *WebGUI* utilizando-os, pois, eles representarão o servidor *master*.

Utilizando a máquina virtual Windows XP, é possível acessar o *WebGui* através do endereço 192.168.100.1. Ao realizar este procedimento a tela de *login* será exibida. Após o *login* a primeira tela a ser apresentada será o *dashboard*, semelhante ao que é mostrado na figura 23.

Figura 23 – Dashboard do pfMaster acessado via IP Virtual CARP.



Fonte: Elaborado pelo autor.

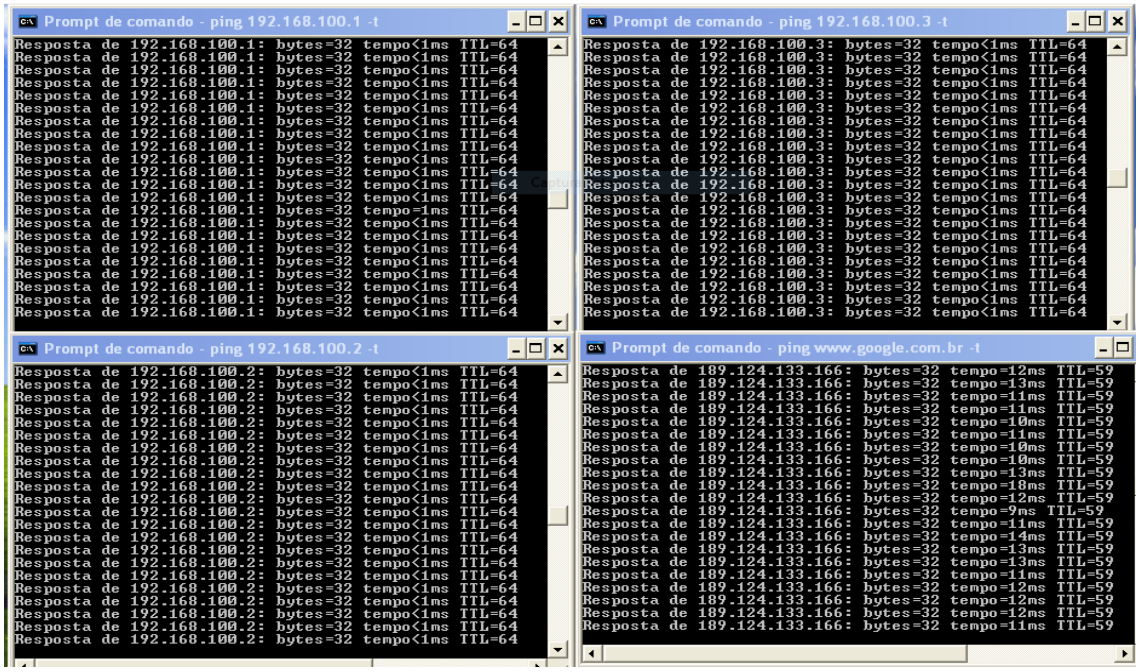
3.12 TEMPO DE REPOSTA EM CASO DE FALHA

Simulando o desligamento do servidor *pfMaster* pode-se observar o tempo que leva para o *pfSlave* assumir o papel de *máster*, ou seja, o tempo de resposta em caso de falha.

Uma forma simples de observar é abrir quatro janelas do *prompt* de comando e disparar *pings* para os *IPs desejados* de forma simultânea, 192.168.100.1, 192.168.100.2, 192.168.100.3 e ***www.google.com***. As janelas dos *prompts* devem ser dispostas lado a lado de forma que seja possível visualizar o conteúdo das mesmas simultaneamente.

O primeiro *prompt* dispara *pings* para o *IP Virtual* da LAN com o uso do comando *ping 192.168.100.1 –t*, o segundo *prompt* realiza *pings* para o IP real do pfMaster, *ping 192.168.100.2 -t*, o terceiro para o IP real do pfSlave, *ping 192.168.100.3* e por último o *ping* para o google, *ping www.google.com –t*. A figura 24 exibe o resultado.

Figura 24 – Verificando tempo de resposta em caso de falha do servidor pfMaster



Fonte: Elaborado pelo autor.

Percebe-se que os quatro endereços estão respondendo ao teste, esta é a situação esperada para o ambiente normal.

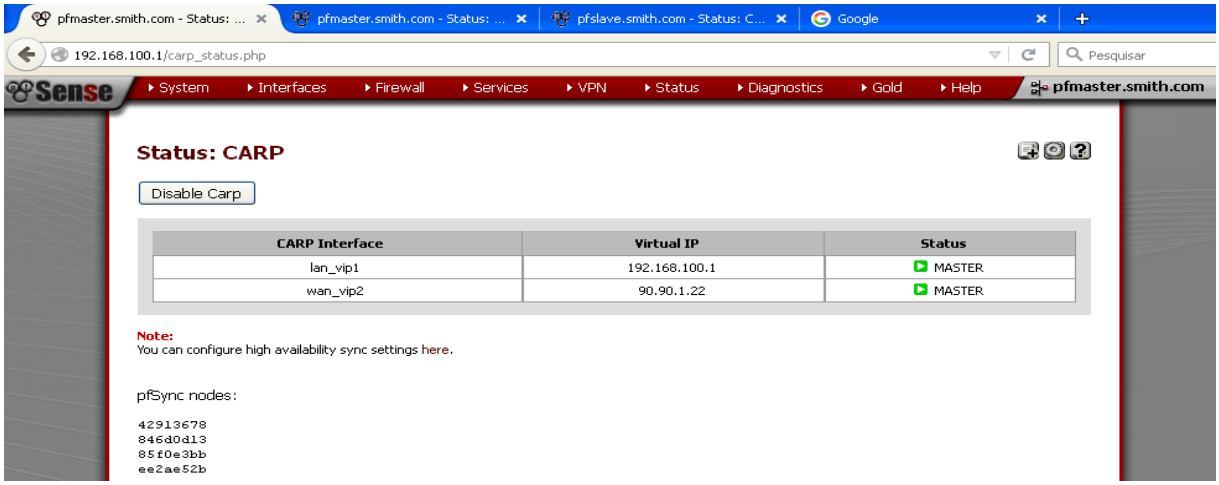
É possível acompanhar também o funcionamento por intermédio do navegador Firefox, acessando respectivamente o status do CARP no *WebGUI* do *IP Virtual* da LAN, *pfMaster*, *pfSlave* e o site do google.

No Firefox a primeira aba será a do *WebGUI* do *IP Virtual* da LAN, 192.168.100.1. Na segunda o *WebGUI* do *pfMaster*, 192.168.100.2. A terceira janela

abra o *WebGUI* do *pfSlave*, 192.168.100.3, a quarta e última a do google.

Observa-se que na tela de status do *IP Virtual da LAN* está como *master*, o que está correto, pois este endereço virtual foi criado para representar o *cluster* de servidores pfSense e mesmo que o *pfMaster* real seja desligado este endereço deverá continuar funcionando e apresentando o status de *master.* O resultado esperado será semelhante ao mostrado na figura 25.

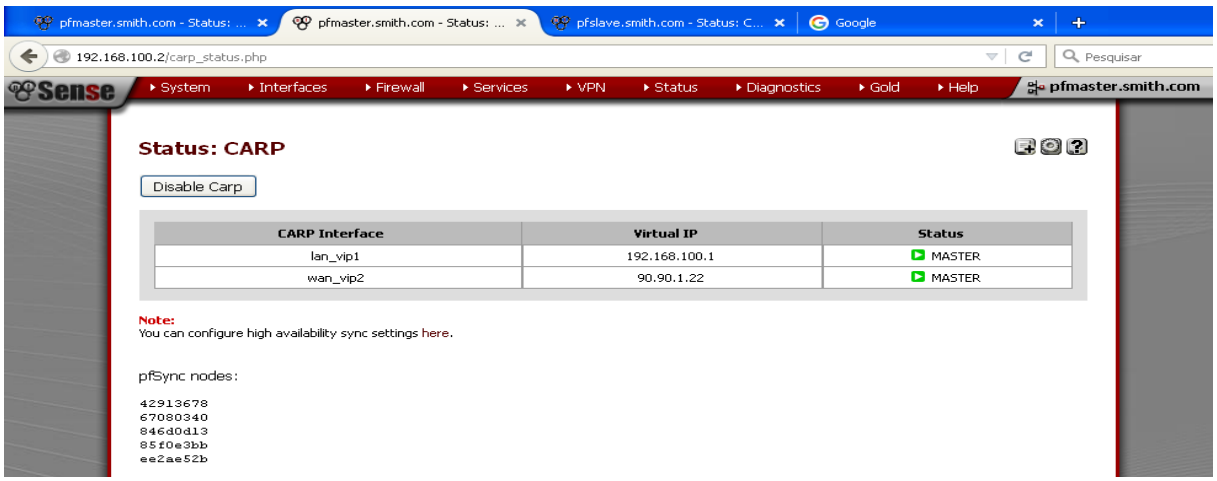
Figura 7 – Tela de status do IP virtual IP da interface LAN



Fonte: Elaborado pelo autor.

Na segunda aba, a do *pfMaster*, observa-se que ele também apresenta o status de *master*, pois, devido a configuração ele é o que tem maior prioridade. A figura 26 exibe uma tela semelhante.

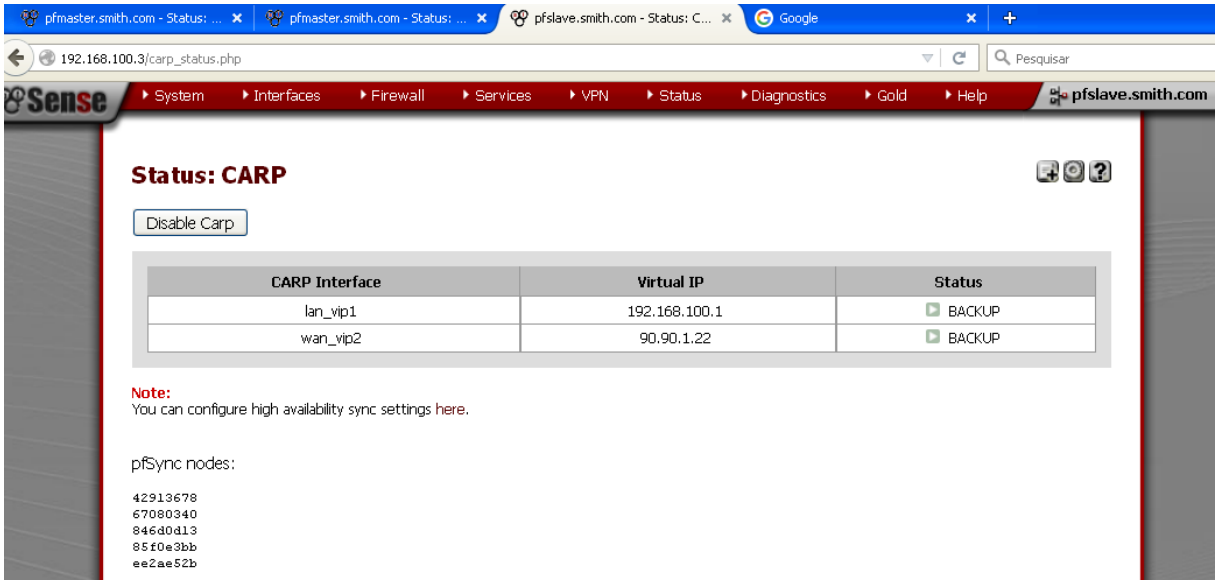
Figura 26 – Tela de status CARP do pfMaster.



Fonte: Elaborado pelo autor.

Na terceira aba temos o status do *pfSlave*, ver-se que o status aparece como *backup*, isto acontece porque o prioridade do *pfSlave* é menor do que a do *pfMaster*, este status apenas mudará para o papel de *master* quando o *pfMaster* for desligado. A figura 27 mostra o status do *pfSlave*.

Figura 27 – Tela de status CARP do pfSlave



Fonte: Elaborado pelo autor.

A quarta aba tem-se a página do google aberta, como mostra figura 28.

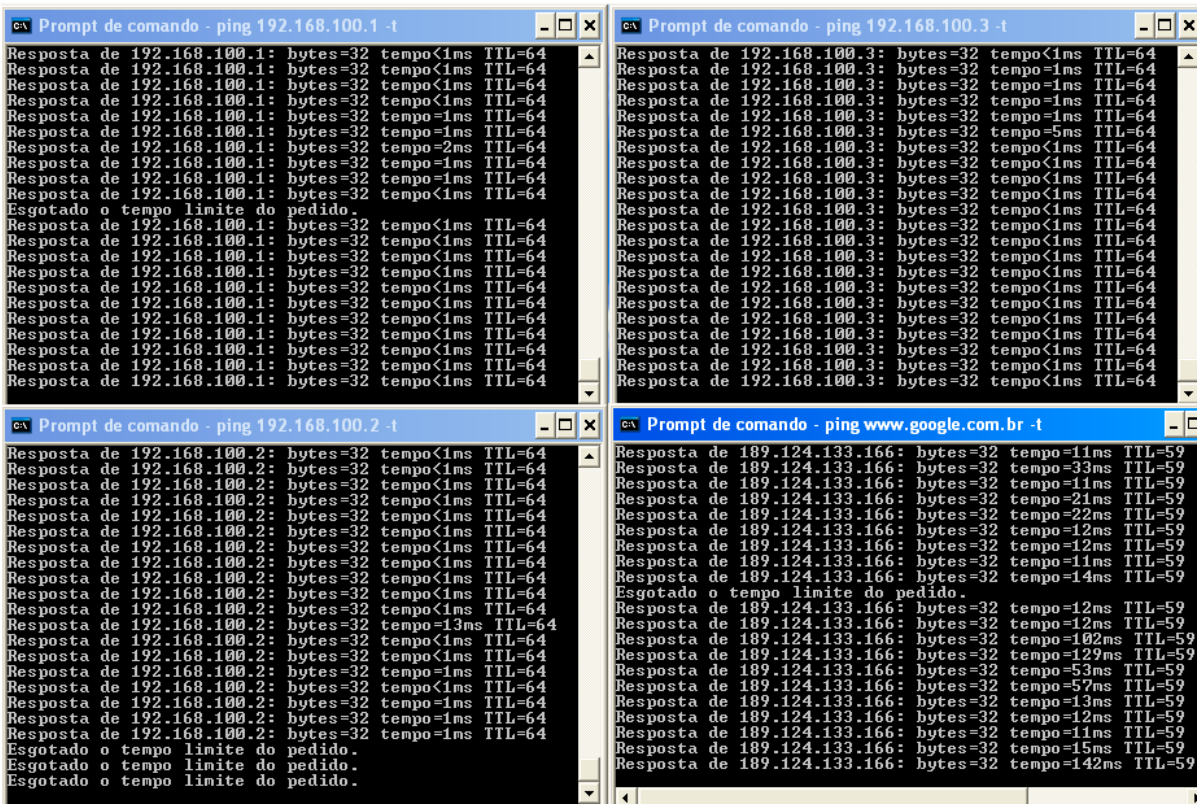
Figura 28 – Página do google.

Fonte: Elaborado pelo autor.

Observa-se que tudo está funcionando como esperado, todos os endereços estão respondendo ao *ping*, abrindo o *WebGUI* e acessando a internet normalmente, com isso pode-se agora desligar o *pfMaster* real, utilizando para isso a opção *Halt System* do menu *Diagnostics*.

Durante o processo de desligamento do servidor, já é possível observar na área de trabalho, onde encontra-se os prompts, que *pfMaster* vai começar a perder pacotes, como mostrado na figura 29.

Figura 29 – Teste de ping após o pfMaster ser desligado



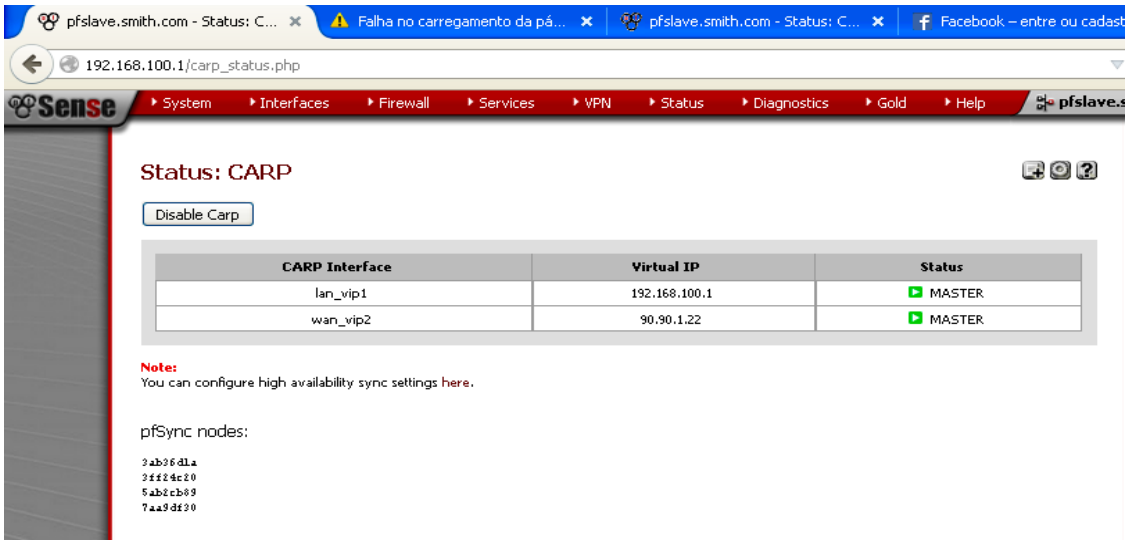
Fonte: Elaborado pelo autor.

Na mesma figura, na janela do *prompt* onde está sendo executado o *ping* para o endereço 192.168.100.1, vai perder apenas 1 (um) pacote e em seguida voltará a responder normalmente, assim como o *ping* para o google.

Ao recarregar as abas do Firefox. Nota-se que a primeira aba onde estava aberto o IP CARP, 192.168.100.1, vai voltar para a tela de *login*, pois anteriormente ele estava representando a interface do *pfMaster*, mais como ele foi desligado a interface que será representada agora será a do *pfSlave* e assim será necessário um novo *login*. Indo novamente até a tela de status do CARP, observa-se que o

status continuará como *master,* mais agora o nome do servidor ser *pfSlave*, como mostra a figura 30.

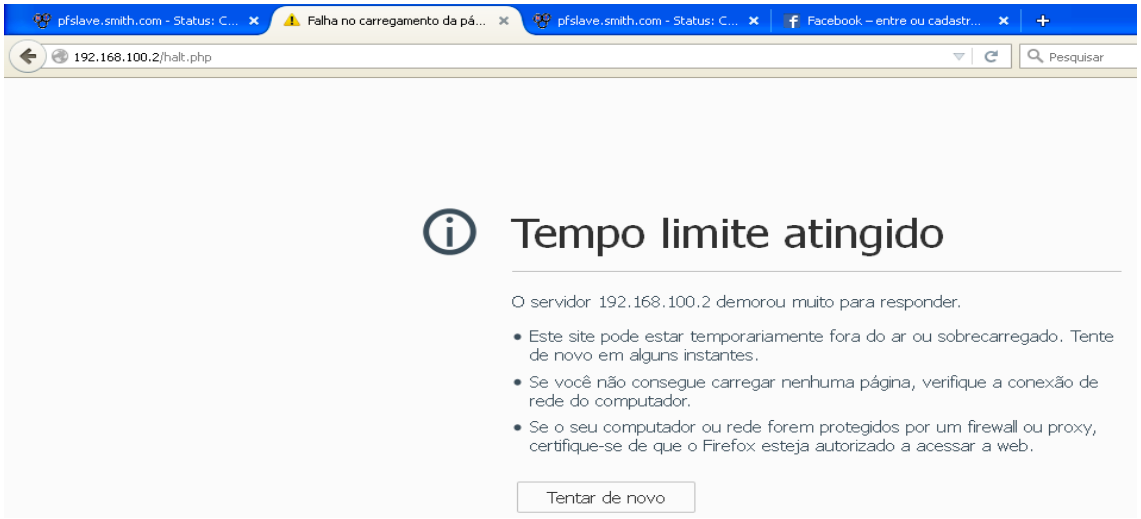
Figura 30 – WebGUI do IP Virtual após o pfMaster ter sido desligado.



Fonte: Elaborado pelo autor.

Na aba 2 (dois), onde estava aberta o *pfMaster*, 192.168.100.2, após a página ter sido recarregada não conseguirá mais encontrar a página do *WebGUI* e vai apresentar uma mensagem de página não encontrada, como mostrado na figura 31.

Figura 31 – WebGUI do pfMaster após ter sido desligado não carrega mais.

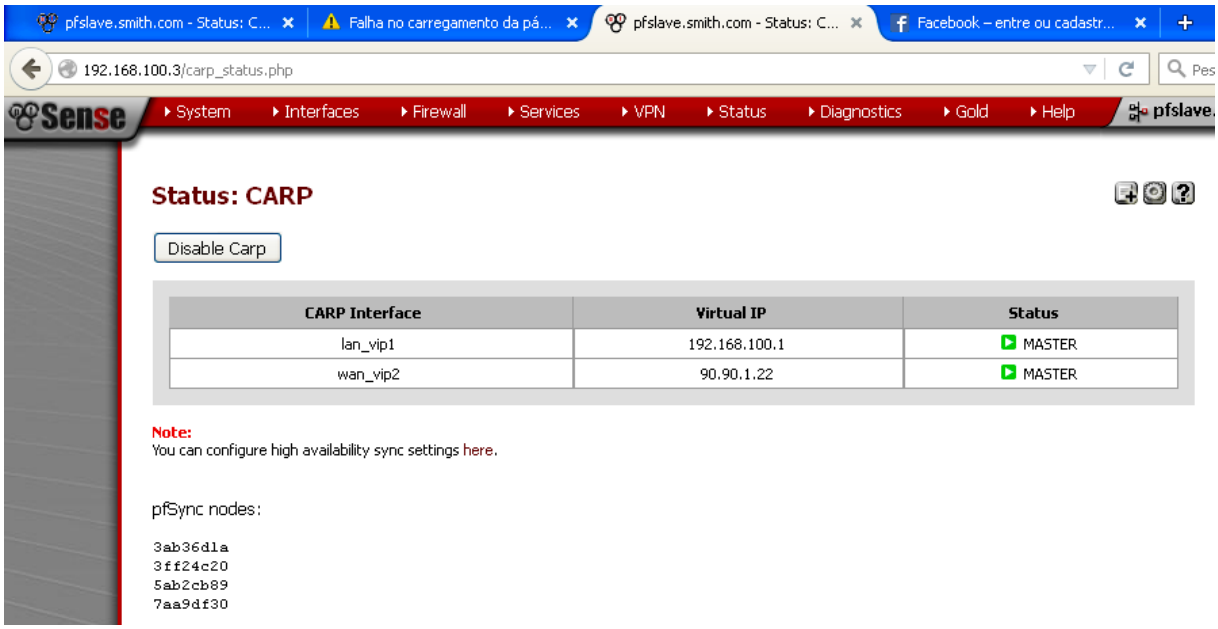


Fonte: Elaborado pelo autor.

Na aba 3 (três), onde estava aberto o *pfSlave*, 192.168.100.3, continuará

funcionando normalmente, só que agora onde era apresentado o status *backup,* será mostrado o status *master,* como mostrado na figura 32.

Figura 32 – pfSlave assume função de master após o pfMaster ter sido desligado.



Fonte: Elaborado pelo autor.

Ainda no Firefox, percebe-se que a navegação vai estar funcionando normalmente. Em questão de segundos o servidor *backup* assume o papel de *master,* mantendo a navegação normal. Dessa forma a falha torna-se praticamente imperceptível ao usuário.

Com a realização desses procedimentos, chega-se ao fim a configuração do ambiente em Alta Disponibilidade com o uso do firewall pfSense.

4 CONCLUSÃO

Ao fim do trabalho, conclui-se que o pfSense é uma excelente alternativa para quem está planejando implantar a solução de Alta Disponibilidade. Percebeu-se que com poucos procedimentos é possível montar um ambiente que se manterá disponível, mesmo que haja o desligamento do servidor considerado primário. O pfSense mostrou-se também muito fácil de configurar e de implementar, tento ainda um ótimo custo benefício, pois, sua obtenção é feita de forma gratuita, deixando os custos por parte do maquinário, caso os equipamentos disponíveis não possam ser reutilizados, gerando a necessidade de aquisição de novos equipamentos que irão compor a estrutura desejada.

REFERÊNCIAS

BLOG SEJA LIVRE. **Conhecendo e configurando o Pfsense.** Disponível em: <http://sejalivre.org/conhecendo-configurando-pfsense/>. Acesso em: 3 maio de 2015.

BLOGSPOT. **Redes e Servidores:** Alta Disponibilidade – Introdução. Disponível em: <http://redes-e-servidores.blogspot.com.br/2011/02/alta-disponibilidade-introducao.html>. Acesso em: 3 abr. 2015.

CONCEITOS BÁSICOS DA ALTA DISPONIBILIDADE. Disponível em: <http://ucbweb2.castelobranco.br/webcaf/arquivos/12822/1156/Traducao_HA_1.pdf> . Acesso em: 20 jul. 2015.

FREE BSD. **The Freebsd Project**. Disponível em: <https://www.freebsd.org/>. Acesso em: 11 dez. 2015.

JALOTE, Pankaj. **Fault Tolerance in Distributed Systems.** São Paulo: Prentice-Hall; 1994.

KASPERSKY. **O que é a gestão unificada de ameaças (UTM, Unified Threat Management)?.** 2015. Disponível em: <http://www.kaspersky.com/pt/internet-security-center/definitions/utm>. Acesso em: 11 dez. 2015.

LI, T. et al. **Cisco Hot Standby Router Protocol (HSRP).** 1998. Disponível em: <https://www.ietf.org/rfc/rfc2281.txt>. Acesso em: 11 dez. 2015.

M0N0WALL.Disponivel em: <http://m0n0.ch/wall/index.php>. Acesso em: 11 dez. 2015.

MICHAELIS. Disponibilidade. In:_____. **Dicionário da Língua Portuguesa.** Disponível em: <http://michaelis.uol.com.br/moderno/portugues/index.php?lingua=portugues-portugues&palavra=disponibilidade>. Acesso em: 19 nov. 2015.

NADAS, S. Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6. **Proposed Standard,** 2010. Disponível em: <https://tools.ietf.org/html/rfc5798>. Acesso em: 11 dez. 2015.

OPENBSD. **PF:** Firewall Redundancy with CARP and pfsync. Disponível em: <http://www.openbsd.org/faq/pf/carp.html>. Acesso em: 11 nov. 2015a.

OPENBSD. **PF:** Redundância de Firewall com CARP e pfsync. Disponível em: <http://www.tw.openbsd.org/faq/pf/pt/carp.html#pfsyncintro>. Acesso em: 11 nov. 2015b.

OPENBSD. **PF:** The OpenBSD Packet Filter. Disponível em: <http://www.tw.openbsd.org/faq/pf/pt/carp.html#pfsyncintro>. Acesso em: 11 dez. 2015c.

PFSENSE-BR. **O que é o pfSense?.** Disponível em: <http://www.pfsense-br.org/blog/o-que-e-o-pfsense/>. Acesso em: 3 maio 2015.

PLUMES, David C. An Ethernet Address Resolution Protocol. **Internet Standard,** 1982. Disponível em: < https://tools.ietf.org/html/rfc826>. Acesso em: 11 dez. 2015.

SCHMIDT, Klaus. **High availability and disaster recovery:** concepts, design, implementation. Berlin: Springer, 2006.

SEVERICH, Mauricio. **Alta Disponibilidade**. 2012. Disponível em: <http://mauricio.severich.com.br/linux/refs/serv*id*or/ha.html>. Acesso em: 20 Jul. 2015.

TECHCENTER. **Balanceamento de carga baseado no método "Round-Robin":** Agility Tech Center. Disponível em: <http://techcenter.agilitynetworks.com.br/index.php?option=com_content&view=article&id=174:balanceamento-de-carga-baseado-no-metodo-round-robin&catid=94&Itemid=878>. Acesso em: 11 nov. 2015.

VEGA, Michael. How to Implement SHA-1/HMAC Authentication for bq26100. **Texas Instruments,** 2006. Disponível em: <http://www.ti.com/lit/an/slua389a/slua389a.pdf>. Acesso em: 11 dez. 2015.

WIKIPEDIA**. Virtual IP address.** 2010. Disponível em: <https://en.wikipedia.org/wiki/Virtual_IP_address>. Acesso em: 11 dez. 2015.

WILLIAMSON, Matt. **pfSense 2 CookBook:** A practical, example-driven guide to configure even the most advanced features of pfSense 2. Packt Publishing; 2011.