NIC.BR

CURSO DE IPV6 A DISTÂNCIA

MIKROTIK ROUTEROS V5.20 E O SUPORTE AO IPV6

BOM JARDIM – RJ

2013

LUIZ NOGUEIRA LEMOS

MIKROTIK ROUTEROS V5.20 E O SUPORTE AO IPV6

Trabalho de conclusão de curso da instituição NIC.br como parte dos requisitos necessários para obtenção do certificado de conclusão de curso de IPv6 a distância. Sob a orientação dos professores Eduardo Barasal Morales, Edwin Cordeiro e Antônio Marcos Moreira.

BOM JARDIM – RJ

2013

AGRADECIMENTOS

Agradeço a todos os integrantes do NIC.br que no decorrer no curso apresentaram boa vontade e atenção para com todos os alunos e principalmente mediantes as dúvidas e insistências ao erro.

Agradeço também a todos os alunos do curso que se propuseram ajudar uns aos outros sejam nas dúvidas ou nos materiais compartilhados.

Agradeço a todos da empresa Dalmar Medicamentos Ltda por compreender o tempo dispensado durante o expediente.

Agradeço a toda a minha família por compreender a minha "ausência" durante as minhas horas de estudo.

RESUMO

O fim da disponibilidade de endereços IPv4 é uma realidade, o *IANA*, a instituição que gerencia o endereçamento *IPv4* de todo o mundo já previu este esgotamento há alguns anos, no entanto, mesmo com o *IPv6* já operante tendemos a deixar sempre para implementar quando não tivermos mais opção, seja por custo de implantação, desconhecimento deste protocolo ou mesmo preguiça, deste modo estamos longe de saber dos problemas que enfrentaremos a nos deparar com este novo protocolo, qual a melhor solução para determinados casos, falhas de segurança ou melhorias em relação ao *IPv4*. Por ser relativamente novo muitas limitações tanto de hardware como de software poderão ser encontradas, mesmo em soluções profissionais ou semiprofissionais como o *RouterOS* da *Mikrotik*, devemos então explorar as possibilidades antes que o IPv6 deixe de ser uma opção e se torne uma obrigação.

LISTA DE ABREVIATURAS E SIGLAS

CTBC – Companhia de Telecomunicações Brasil Central
DHCP – Dynamic Host Configuration Protocol
DHCPv4 – Dynamic Host Configuration Protocol Version 4
DHCPv6 – Dynamic Host Configuration Protocol Version 6
DNS – Domain Name Service
DNSv4 – Domain Name Service Version 4
DNSv6 – Domain Name Service Version 6
EUI-64 – Extended Unique Identifier 64 bits
GRE – Generic Routing Encapsulation
HE – Hurricane Eletric
IP – Internet Protocol
IPv4 – Internet Protocol Version 4
IPv6 – Internet Protocol Version 6
ISP – Internet Service Provider
MAC – Media Access Control
MMS – Manufacturing Message Specification
MRU – Maximun Receive Unit
MTU – Maximun Transmition Unit
NAT – Network Address Translation
ND – Neighbor Discovery
POP – Point of Presence
PPC – Per Connection Classifier
PPPoe – Point-to-Ooint Protocol over Ethernet
RA – Router Advertisement
SSH – Secure Shell
TCP – Transmission Control Protocol
UDP – User Datagram Protocol

SUMÁRIO

1. INTRODUÇÃO

O número de aparelhos conectados a internet tem crescido geometricamente, hoje temos *Tablets*, Celulares, Computadores, Câmeras de vigilância, todos em sua maioria utilizando *NAT* (*Network Address Translation*) para driblar o problema do esgotamento do *IPv4 (Internet Protocol Version 4)*, o que tem resolvido parcialmente o problema, no entanto, aumenta significativamente o processamento dos servidores como também quebra o modelo fim a fim. Empresas do mundo inteiro tem se preocupado com o fim de endereços *IPv4* disponíveis como também com a implantação do *IPv6* num contexto global. O *IPv6* (*Internet Protocol Version 6*) não possui retro compatibilidade direta para com o *IPv4*, no entanto, os dois podem ser implantados paralelamente caso haja compatibilidade com *IPv6* do hardware e software utilizados. Outro fator importante são os equipamentos que não possuem disponibilidade para o *IPv6*, sendo assim, até que ponto dependeremos do *IPv4*?

Devemos analisar então como os nossos equipamentos responderão as necessidades impostas, se servirão para tudo de novo que há para vir e se estamos realmente aptos a trabalhar com este novo protocolo. Neste documento analisaremos o software *RouterOS v5.20* da *Mikrotik.*

2. METODOLOGIA

Para realizar nossos testes sobre o protocolo IPv6 utilizaremos dois *RouterBoards* com o software *RouterOS* instalado, sendo o segundo apenas para demostração do funcionamento do *DHCPv6* (*Dynamic Host Configuration Protocol Version 6*), dois computadores um com o Sistema Operacional Microsoft Windows 8 64 bits e outro com o Ubuntu 13.04 64 bits e dois provedores de *Tunnel Broker:* a *SixXS* e a *Hurricane Eletric*. Descreveremos passo a passo os **comandos executados, dificuldades, problemas e melhorias** que encontraremos ao implantar o *IPv6* tanto para a implantação do IPv6 com suporte direto do *ISP (Internet Service Provider),* com técnicas de transição como também para interligação por *IPv6* entre filiais de empresas.

Por uma questão de compatibilidade num primeiro momento demostraremos como seria feita a disponibilização do *IPv4*, a partir de então iniciaremos todo o processo de configuração *IPv6* com suporte direto do ISP e com técnicas de transição.

3. COMPATIBILIDADE

A maior parte dos provedores ainda não fornecem *IPv6* para seus clientes, muitos estando ainda sem previsão para disponibilização, os provedores na região de Bom Jardim, Rio de Janeiro não são diferentes, além disso apenas uma pequena parcela dos provedores de conteúdo oferecem suporte *IPv6* em seus serviços, sendo assim dependemos e dependeremos ainda por mais alguns anos do IPv4 até que a maiores dos serviços na internet ofereçam seus serviços pelo *IPv6*.

Para redes locais ainda temos outros agravantes, por exemplo, existem relógios de ponto, impressoras de rede e outros dispositivos que trabalham apenas com o *IPv4*, portanto, para que consigamos trabalhar somente com o *IPv6* teremos que atualizar o *firmware* ou substituir, caso necessário, cada equipamento que apresentar esta limitação ou manter a compatibilidade através da técnica de pilha dupla (*Dual Stack*) em todos os computadores que necessitem acessar o recurso de um destes dispositivos, no qual cada equipamento manterá dois *IPs* um sendo *IPv4* e outro sendo IPv6.

Pelos fatos citados acima fica indispensável falar do IPv4 para poder falar do

IPv6, pois não justifica ainda ter somente IPv6 sem alguma técnica que permita o acesso a ambos os protocolos.

Considerando um *RouterOS* v5.20 recém-instalado, iniciaremos a configuração inicialmente pelo console.

3.1 CONFIGURAÇÃO BÁSICA IPV4

Por padrão, o usuário do *RouterOS* será o “admin” e a senha será em branco, caso não haja interface gráfica em seu *RouterBoard*, o *RouterOS* poderá ser acessado através de uma aplicação chamada *WinBox,* que é uma ferramenta extraordinária para configuração do *Router* por interface gráfica baseada na plataforma *Microsoft Windows*, onde será possível o acesso ao roteador através do endereço *MAC* ou mesmo pelo *IP (Internet Protocol)* se estiver configurado.

Para os profissionais menos habituados com o *RouterOS*, na versão 5.20 deste software existe t*ambém* o *WebFig* que é uma interface gráfica nativa acessada pelo navegador tanto pelo *IPv4 como IPv6* para configuração do *RouterOS.* Apesar de ambas as ferramentas citadas anteriormente funcionarem perfeitamente na maioria dos casos, sempre será mais aconselhável a um profissional mais experiente que configure o *RouterOS* pelo terminal seja por *SSH, Telnet* ou mesmo pelo *Console*, pois ambas as aplicações possuem *bugs* e diferenças quanto as possibilidades de operações exibidas nas ferramentas e as que o *RouterOS realmente oferece e* como por experiência própria quanto ao bloqueio acidental das portas *TCP (Transmission Controle Protocol)* 80 e 22 na interface errada, respectivamente HTTP e SSH, perdendo assim o acesso remoto as configurações do roteador, sendo solucionado apenas pela retirada da regra do *firewall* através do console. Para acessar o WebFib abra o navegador e digite em sua barra de endereços: http://192.168.0.1 (IP do roteador configurador anteriormente), no entanto, só mostraremos neste documento a configuração do RouterOS através do Terminal *SSH (Secure Shell)*, Console, ou *Telnet*.

Tendo acessado o console com sucesso iniciaremos algumas configurações básicas: definição da faixa de *IPv4* que será utilizada na rede local, alteração dos nomes das interfaces para nomes mais descritivos e configuração da conexão com ISP.

Para alterar o nome das interfaces para nomes mais descritivos execute os

comandos abaixo descritos, isso ajudará na identificação posteriormente das regras do *firewall* como rotas de cada interface, no entanto, não é obrigatório:

/interface ethernet set name="intranet" ether1
/interface ethernet set name="isp" ether2

Tendo escolhido a rede e a máscara, no nosso caso será a rede 192.168.0.0 com máscara de 24 *bits* executaremos o seguinte comando que adicionará o *IPv4* 192.168.0.1 com máscara 24 (255.255.255.0) na interface *intranet*:

/ip address add address=192.168.0.1/24 interface=intranet

3.2 CONECTIVIDADE IPV4

Configuraremos agora a conexão *IPv4* com o nosso *ISP.*

3.2.1 CONECTIVIDADE IPV4 COM PPPOE

Criaremos uma interface PPPoe interligada a uma interface física pela qual conectaremos ao servidor PPPoe informando usuário e senha de acesso através do seguinte comando:

/interface pppoe-client add name="isppppoe" max-mru=1452 max-mtu=1452 [service-name=ispservice] [ac-name=ispac] user="user" password="abc123" add-default-route=yes use-peer-dns=yes interface=isp profile="default-encryption"

O*s atributos entre colchetes são em algumas situações obrigatórios pelo ISP. O atributo add-default-route se encarregará de criar a rota padrão de saída para o ISP, o atributo use-peer-dns adicionará o DNS* (*Domain Name Service) fornecido pelo ISP no servidor DNS do RouterOS, o atributo max-mru e max-mtu são atributos de controle do tamanho máximo de um pacote que poderá ser enviado ou recebido respectivamente. Os atributos max-mtu e max-mru deverão estar de acordo com o*

seu ISP, sendo por padrão 1452, mas podendo ser alterado pelo próprio ISP.

O atributo *profile* executado no comando acima tem uma característica especial, nele podemos administrar como o nosso cliente *DHCP (Dynamic Host Configuration Protocol)* se comportará, por padrão utilizaremos um *profile* nativo do RouterOS, mas poderemos criar o nosso, neste exemplo abaixo apenas desativamos o atributo change-tcp-mss o qual permite que procedimento de adaptação de *MMS* (*Manufacturing Message Specification*) seja feito automaticamente para que não haja descarte de pacotes de ambos os lados se o *MTU (Maximun Transmition Unit)* e o *MRU* (*Maximun Receive Unit*) estiverem configurados diferentemente de seu *ISP*:

/ppp profile add name="pppoe_profile" remote-ipv6-prefix-tool=none use-ipv6=yes use-mpls=default use-compression=default use-vj-compression=default use-encryption=default only-one=default change-tcp-mss=no

Obs.: Não aconselhável, apenas para demostração.

Liberaremos a acesso ao *DNS* internamente na instituição, lembrando sempre de bloquear no *firewall* a porta 53 *UDP (User Datagram Protocol)* e *TCP* tanto para *IPv6* como para *IPv4* para que outras pessoas não possam usar nosso *DNS*:

/ip dns set allow-remote-request= yes max-udp-packet-size=512
/ip firewall filter add action=drop in-interface=isppppoe protocol=tcp dst-port=53
/ip firewall filter add action=drop in-interface=isppppoe protocol=udp dst-port=53

Caso o atributo add-default-route esteja "false" na configuração da conexão PPPoe precisaremos executar o seguinte comando para que uma rota padrão de saída seja criada:

/ip route add dst-address=0.0.0.0/0 gateway isppppoe

Caso o atributo use-per-dns esteja "false" na configuração da conexão PPPoe precisaremos executar o seguinte comando para definir um servidor DNS para

nosso roteador, no entanto, não necessariamente precisaremos que sejam estes valores (Google DNS), porém é um dos que apresenta menor latência, lembrando-se sempre de permitir acesso internamente e bloqueando as tentativas de acesso externo:

```
/ip dns set servers=8.8.8.8,8.8.4.4 allow-remote-request=yes max-udp-packet-size=512
/ip firewall filter add action=drop in-interface=isp protocol=tcp dst-port=53
/ip firewall filter add action=drop in-interface=isp protocol=udp dst-port=53
```

3.2.2 CONECTIVIDADE IPV4 COM IP ESTÁTICO

considerando que o ISP forneceu os seguintes dados:

IP do cliente: 200.200.200.200/24
IP do gateway: 200.200.200.1
Servidores DNS: 200:200:200:2, 200:200:200:3

Primeiramente definiremos o endereço IPv4 do cliente:

```
/ip address add address=200.200.200.200/24 interface=isp
```

Criaremos agora a rota padrão:

```
/ip route add dst-address=0.0.0.0/0 gateway 200.200.200.1
```

Precisaremos executar o seguinte comando para definir um servidor *DNS* para nosso roteador, utilizaremos os endereços dos servidores *DNS* informados pelo *ISP*, lembrando de permitir acesso internamente apenas bloqueando as tentativas de acesso externo:

/ip dns set servers=200.200.200.2, 200.200.200.3 allow-remote-request=yes max-udp-packet-size=512
/ip firewall filter add action=drop in-interface=isp protocol=tcp dst-port=53
/ip firewall filter add action=drop in-interface=isp protocol=udp dst-port=53

3.2.3 CONECTIVIDADE IPV4 COM IP DINÂMICO

Criamos um serviço de cliente DHCPv4 para capture as informações de configuração de IPv4 pela interface selecionada através do seguinte comando:

/ip dhcp-client add add-default-route=yes use-peer-dns=yes use-peer-ntp=yes interface=isp

Precisaremos executar o seguinte comando para definir um servidor DNS para nosso roteador, no entanto, não necessariamente precisaremos que sejam estes valores (Google DNS), porém é um dos que apresenta menor latência, lembrando de permitir acesso internamente apenas bloqueando as tentativas de acesso externo:

/ip dns set allow-remote-request= yes max-udp-packet-size=512
/ip firewall filter add action=drop in-interface=isp protocol=tcp dst-port=53
/ip firewall filter add action=drop in-interface=isp protocol=udp dst-port=53

3.2.4 CONECTIVIDADE IPV4 PARA A SUB-REDE

Configuraremos agora um pool de endereços e um servidor DHCP associado a ele para nossa rede interna (intranet) para que todos os dispositivos possam ter conectividade com a internet IPv4, mas caso prefira, poderá configurar manualmente o endereço IP em cada máquina:

/ip pool add name="DHCPv4Pool" ranges=192.168.0.30-192.168.0.100
/ip dhcp-server network add address=192.168.0.0/24 dns-server=192.168.0.1 domain=casa gateway=192.168.0.1 netmask=24
/ip dhcp-server add address-pool="DHCPv4Pool" authorative=after-2sec-delay interface=intranet lease-time lease-time=1d name="DHCPv4"

Para que nossos computadores possam acessar a internet pelo protocolo IPv4 normalmente precisaremos ativa o NAT com o seguinte comando:

Caso seja conectado pelo PPPoe:

/ip firewall nat add action=masquerade out-interface=isppppoe

Caso seja conectado diretamente (IP estático / dinâmico):

/ip firewall nat add action=masquerade out-interface=isp

4. CONECTIVIDADE IPV6

Pronto, a configuração para o *IPv4* está pronta, a partir de agora realizaremos a configuração do protocolo *IPv6* para que os dispositivos em nossa rede possam se comunicar com qualquer dispositivo na internet, seja ele *IPv4* ou *IPv6*.

As configurações se assemelham muito com as do *IPv4*, portanto, não repetirei as explicações pertinentes aos dois protocolos pois todas estão citados nos subitens do item de número 3.

Observe que para todas as configurações *IPv6* abaixo descritas nenhuma utiliza NAT, sendo o mesmo nem disponibilizado no *RouterOS* v5.20

4.1 CONECTIVIDADE IPV6 COM PPPOE

Criaremos uma interface PPPoe interligada a uma interface física pela qual conectaremos ao servidor PPPoe informando usuário e senha de acesso através do seguinte comando:

/interface pppoe-client add name="ispppoev6" max-mru=1452 max-mtu=1452 user="user" password="abc123" [service-name=ispservice] [ac-name=ispac] add-default-route=yes use-peer-dns=yes interface=isp profile="default-encryption"

Caso o atributo add-default-route esteja "false" na configuração da conexão PPPoe precisaremos executar o seguinte comando para que uma rota padrão de saída seja criada:

/ipv6 route add dst-address= ::/0 gateway isppppoev6

Caso o atributo use-per-dns esteja "false" na configuração da conexão PPPoe precisaremos executar o seguinte comando para definir um servidor DNS para nosso roteador, no entanto, não necessariamente precisará ser estes valores (Google DNS), porém é um dos que apresenta menor latência, lembrando de permitir acesso internamente apenas bloqueando as tentativas de acesso externo:

/ip dns set servers=8.8.8.8,2001:4860:4860::8888,8.8.4.4,2001:4860:4860:8444 allow-remote-request=yes max-udp-packet-size=512
/ipv6 firewall filter add action=drop in-interface=isppppoev6 protocol=tcp dst-port=53
/ipv6 firewall filter add action=drop in-interface=isppppoev6 protocol=udp dst-port=53

No comando acima mantivemos dos quatro servidores *DNS* informados dois *IPv4* e dois *IPv6*, para que o *RouterOS* dê preferência para conectar a servidores *DNSv6* pelo protocolo *IPv6* e os servidores *DNSv4* pelo protocolo *IPv4*, no entanto, caso sejam informados apenas servidores *IPv4* ou somente servidores IPv6, o servidor *DNS* funcionará perfeitamente desde que o mesmo protocolo esteja configurado corretamente pois há compatibilidade entre os servidores DNSv4 e DNSv6.

Para conexões *PPPoe* teremos que configurar um cliente *DHCPv6* para que o mesmo possa capturar o prefixo para um pool e posteriormente divulgá-lo através do *ND (Neighbor Discovery)* para os demais dispositivos:

/ipv6 dhcp-client add interface=isppppoev6 pool-name=ispv6pool pool-refix-length=64
/ipv6 address ::1/64 advertise=yes eui-64=no from-pool=ispv6pool interface=intranet

Observe que no comando anterior adicionamos o IP ::1/64 na interface utilizando o prefixo do *pool, sendo assim o roteador pegara os 64 bis relativos ao prefixo a adicionará ao final os 64 bits do IP acima citado. Configurando a opção advertise* como "*true*" o *ND* divulgará o prefixo utilizado para os dispositivos abaixo da interface intranet, observe ainda que deixamos a opção *EUI-64 (Extended Unique Identifier 64 bits)* como "*false*", sendo assim o roteador utilizará os últimos 64 bits do IP informado para formar o IPv6 da interface intranet, caso o EUI-64 seja configurado para "true" o roteador utilizar do cálculo EUI-64 para configurar um IPv6 baseado no endereço MAC da interface intranet.

Executando o comando abaixo exibiremos a tabela de prefixos a serem divulgados, no nosso caso desejamos verificar a configuração do prefixo que será divulgado na rede interna, supondo que seja 2001:1450:1234:1234::/64 este prefixo será configurado automaticamente:

/ipv6 nd prefix print

Deverão ser exibidos os seguintes dados com a letra D no início, informando que a configuração deste prefixo:

prefix=2001:1450:1234:1234::/64 interface=intranet on-link=yes autonomos=yes valid-lifetime=4w2d preferred-lifetime=1w

Há dois atributos extremamente importantes, o atributo "*on-link*" informa aos dispositivos que o prefixo informado não deverá ser resolvido pelo *gateway*, por se tratar de um prefixo local e o atributo "*autonomous*" informa aos clientes através do *ND* que eles poderão utilizar autoconfiguração *stateless* com o prefixo informado. Neste ponto o usuário já terá acesso a um endereço *IPv6 global* e um endereço *IPv6 link-local.* Configuraremos agora o *RA (Router Advertise) :*

/ipv6 nd set interface=intranet advertise-dns=yes advertise-mac-address=yes other-configuration=yes managed-address-configuration=no

O atributo "advertise-dns" permite que o roteador inclua no RA através do ICMPv6 tipo 25 o endereço do servidor DNS que o cliente deverá utilizar, no entanto, será informado o servidor DNS IPv6 que foi cadastrado para DNS do próprio

roteador e não repassado o endereço IPv6 do roteador para que a solicitação seja centralizada para fins de controle de cache e redução do fluxo de mensagens de pesquisa de nomes, portanto, cada cliente fará uma busca para cada domínio a ser pesquisado em vez de aproveitar o resultado da busca de outros clientes. O atributo *"advertise-mac-address"* quando setado informará pelo RA o endereço do gateway a ser utilizado. O atributo "*other-configuration*" informa ao cliente que estes deverão utilizar configuração *stateful*. O atributo "*managed-address-configuration*" informa ao cliente que este deverá utilizar um servidor DHCP.

4.2 CONECTIVIDADE IPV6 COM IP ESTÁTICO

considerando que o ISP forneceu os seguintes dados:

IP do cliente: fe80:1234::2/127
IP da sub-rede: 2001.1234.1234.1234::/64

IP do gateway: fe80:1234::1/127
Servidores DNS: 2001.1234.1234.1000::1, 2001.1234.1234.1000::2

Primeiramente definiremos o endereço IPv6 do cliente:

/ipv6 address add address=fe80:1234::2/64 interface=isp advertise=no eui-64=no
*/ipv6 address add address=*2001.1234.1234.1234*::1/64 interface=intranet advertise=yes eui-64=no*

Criaremos agora a rota padrão:

/ipv6 route add dst-address=::0/0 gateway=fe80:1234::1/64

Configuraremos agora o *RA (Router Advertise) :*

/ipv6 nd set interface=intranet advertise-dns=yes advertise-mac-address=yes other-

configuration=yes managed-address-configuration=no

Precisaremos executar o seguinte comando para definir um servidor DNS para nosso roteador, lembrando de permitir acesso internamente apenas bloqueando as tentativas de acesso externo:

/ip dns set servers=200.200.200.2, 2001.1234.1234.1000::1*, 200.200.200.3,* 2001.1234.1234.1000::2 *allow-remote-request=yes max-udp-packet-size=512*
/ipv6 firewall filter add action=drop in-interface=isp protocol=tcp dst-port=53
/ipv6 firewall filter add action=drop in-interface=isp protocol=udp dst-port=53

4.3 CONECTIVIDADE IPV6 COM IP DINÂMICO

A configuração de *IP* dinâmico se dará sempre por *DHCPv6* para roteadores visto que os roteadores não implementam a autoconfiguração. Inicialmente precisaremos configurar o cliente *DHCPv6*, o qual capturará o prefixo fornecido pelo *ISP* e o adicionará em um *pool* para ser utilizado pelos dispositivos da rede interna.

/ipv6 dhcp-client add interface=isp pool-name=isppoolv6

Agora adicionaremos um IPv6 na interface selecionando um pool dinâmico:

/ipv6 address ::1/64 from-pool=isppoolv6 interface=intranet advertise=yes eui-64=no

4.4 TUNNEL BROKER

Tunnel Broker é uma técnica que permite que clientes de ISPs que não ofereçam conectividade IPv6 possam receber um prefixo /64 ou mesmo um /48 através de um túnel 6in4, o qual encapsula todo o tráfego *IPv6* dentro de pacotes *IPv4*, então para que seja possível a utilização do *Tunnel Broker* será necessário uma conexão funcional *IPv4* com o seu ISP.

Para configurar da técnica *Tunnel Broker*, primeiramente será necessário realizar um cadastro no site da empresa SixXS ou semelhante para obter um prefixo IPv6, se seu usuário e sua solicitação forem aprovados a SixXS te encaminhará um e-mail contendo seu IPv4, o IPv4 da SixxS, o IPv6 da SixXS, o IPv6 do seu roteador para enlace e o prefixo IPv6 /64 para sua subnet.

Dois fornecedores deste serviço foram testados, *Hurricane Eletric* (http://www.tunnelbroker.net/) e a *SixXS* (http://www.sixxs.net /), ambos oferecem boa estabilidade, no entanto, a *HE* (*Hurricane Eletric*) oferece maior flexibilidade quanto a criação de outros túneis (limite de 5 túneis) que são criados imediatamente bem como a aquisição de um prefixo /48 para a possibilidade de criação de sub-redes sem maiores esforços, a *SixXS* tem mais restrições, todo o qualquer túnel tem de ser solicitado formalmente e justificado, porém o *POP* do *SixXS* mais próximo é fornecido pela CTBC (Companhia de Telecomunicações Brasil Central) em Minas Gerais, enquanto o Hurricane Eletric apenas em outros continentes explicando assim a menor latência da *SixXS* e minha escolha.

Considerando a seguinte tabela:

Seu *IPv4*: 200.200.200.200
SixXS IPv4: 189.189.189.189
Seu *IPv6*: 2001:1234:1234:0234::2/64
SixXS IPv6: 2001:1234:1234:0234::1/64
IPv6 Sub-rede: 2001:1234:1234:1234::/64

Criaremos uma interface *6to4* no *RouterOS* informando o endereço *IPv4* local e o endereço *IPv4* Remoto (*SixXS IPv4*):

/interface 6to4 add local-address=200.200.200.200 mtu=1280 name=sixxs remote-address=189.189.189.189

Adicionaremos os *IPv6* fornecidos pela *SixXS* em cada interface:

/ipv6 address add 2001:1234:1234:0234::2/64 eui-64=no advertise=no interface=sixxs

/ipv6 address add 2001:1234:1234:1234::1/64 eui-64=no advertise=yes interface=intranet

Adicionaremos agora a rota padrão:

/ipv6 route add dst-address=::/0 gateway=2001:1234:1234:0234::1

Configuraremos agora o *RA (Router Advertise) :*

/ipv6 nd set interface=intranet advertise-dns=yes advertise-mac-address=yes other-configuration=yes managed-address-configuration=no

Precisaremos executar o seguinte comando para definir um servidor DNS para nosso roteador, lembrando de permitir acesso internamente apenas bloqueando as tentativas de acesso externo:

/ip dns set servers=200.200.200.2, 2001.1234.1234.1000::1*, 200.200.200.3,* 2001.1234.1234.1000::2 *allow-remote-request=yes max-udp-packet-size=512*
/ipv6 firewall filter add action=drop in-interface=sixxs protocol=tcp dst-port=53
/ipv6 firewall filter add action=drop in-interface=sixxs protocol=udp dst-port=53

Infelizmente a *SixXS* não liberou em meu caso um prefix /48, mas somente um /64 demostrarei a partir de agora o funcionamento do servidor *DHCPv6* com a limitação do *RouterOS* v5.20 utilizando a *Hurricane Eletric.*

Para configurar o RouterOS para um servidor DHCPv6 com o Tunnel Broker basta configurá-lo normalmente, mas ao final criaremos um pool de endereços dois quais divulgaremos para outros roteadores na rede. O comando a seguir criará um de endereços distribuindo prefixos /64, sendo assim teremos a possibilidade de criar 65536 sub-redes.

/ipv6 pool name="poolv6" prefix=2001:4321:4321::/48 prefix-length=64

Após a criação do pool de prefixos /64 criados o servidor DHCPv6:

/ipv6 dhcp-server add name="dhcpv6" interface=intranet address-pool="poolv6"

4.5 TÚNEL *GRE (Generic Routing Encapsulation)*

O túnel *GRE* é um túnel onde podemos trafegar *IPv6* ou até mesmo *IPv4* sobre o *IPv4*, por exemplo, caso necessitemos interligar duas filiais de uma mesma empresa, ambas, ou apenas uma delas não possuindo suporte a *IPv6* nativo pelos *ISPs*. Para implantar o GRE basta que ambas as filiais tenham acesso a internet pelo protocolo *IPv4*. Criaremos agora uma interface no *RouterOS* que permitirá realizar qualquer procedimento *IPv6* sobre a mesma:

/interface gre name="gre" local-address=189.189.189.189 remote-address=200.200.200.200

Tendo realizado este mesmo procedimento no roteador remoto o túnel estará funcional para o tráfego de dados. Existe ainda o *GRE6*, sendo sua única diferença que o túnel funcionará sobre o *IPv6*, mas também poderá trafegar *IPv4* e *IPv6*. Outros tuneis estão disponíveis no *RouterOS*, são eles: *EoIPv6* e *IPIPv6* tendo o funcionamento parecido com o *GRE sendo que o IPIPv6 somente trafegará IPv6 e o EoIPv6 trafegará todos os protocolos pois este último é um túnel entre interfaces.*.

5. DIFICULDADES E PROBLEMAS

Durante toda a pesquisa encontramos alguns fatores poucos citados na internet, que causaram alguns um certo transtorno para os usuários na rede:

Infelizmente o *RouterOS* da *Mikrotik* v5.20, apesar de ter a opção "gerenciar configuração de endereços" este somente está implementado para responder para outros roteadores, não para máquina de usuário, restando apenas a configuração sem estado pelo EUI-64.

O Windows não adquire por padrão apenas um IPv6 Global e um IPv6 Link Local, mas também um IPv6 temporário para ambas os tipos que não se repetem de acordo com RFC4941 sobre a extensão que privacidade, porém para administradores de redes que desejam um controle maior sobre a definição de rotas e controle de acesso, uma solução seria evitar essa randomização de endereços IPv6, podendo ser desativada através dos comandos abaixo citados e reiniciando o computador após a execução:

netsh interface ipv6 set privacy state=disabled store=active
netsh interface ipv6 set privacy state=disabled store=persistent
netsh interface ipv6 set global randomizeidentifiers=disabled store=active
netsh interface ipv6 set global randomizeidentifiers=disabled store=persistent

O Windows mesmo em sua versão mais nova, o Windows 8, não implementa a RFC6106, ou seja, não carrega as informações do DNS (ICMPv6 Type 25 RDNSS) pelo Router Advertise (RA) no modo stateless autoconfiguration enquanto o Linux funciona normalmente desde o Ubuntu 11.04.

Um problema que temos enfrentando é que na rede IPv4 configurada com balanceamento de carga *PPC* (*Per Connection Classifier*), o *PPC* utiliza da função *PPC* da tabela *Mangle* do *Firewall IPv4* do *RouterOS* para realizar um cálculo no meu caso do IP de Origem com o IP de Destino e realizando uma divisão de números inteiros sobre um divisor informado na função e de acordo com o resto desta divisão será realizada a marcação da conexão e ao final a marcação da rota de saída de acordo com a marcação da conexão. Na tabela de roteamento *IPv4* as rotas padrões de menor distância serão atribuídas para os pacotes que tiverem as marcações anteriormente configurada, no entanto, a tabela de roteamento *IPv6 do RouterOS* v5.20 não possui ainda o atributo de marcação de rota inviabilizando assim o balanceamento de carga *PPC* similar ao *IPv4*.

A técnica Tunnel Broker tem causado instabilidade na conexão, visto que usuário que possuem IPv6 e por esse ser prioridade tem tido problemas com sites com segurança aprimorada como os bancos, principalmente o Itaú.

6. MELHORIAS

Testes realizados com IPv4 e o IPv6 pelo site http://teste-ipv6.com mostraram que apesar do IPv6 estar sendo conectado através de um túnel 6in4 ele ainda ganha por menor latência do IPv4 em alguns fatores como acesso ao servidor DNS e a procura de provedores de serviço.

Testa com um registro DNS IPv4	**ok** (1.242s) usando ipv4
Testa com um registro DNS IPv6	**ok** (0.978s) usando ipv6
Testa com um registro DNS duplo	**ok** (1.533s) usando ipv6
Testa DNS duplo e pacote grande	**ok** (2.418s) usando ipv6
Testa IPv4 sem DNS	**ok** (0.576s) usando ipv4
Testa IPv6 sem DNS	**ok** (0.728s) usando ipv6
Testa pacote IPv6 grande	**ok** (1.452s) usando ipv6
Testa se o servidor DNS de seu provedor usa IPv6	**ok** (0.988s) usando ipv6
Find IPv4 Service Provider	**ok** (1.444s) usando ipv4 ASN 28210
Find IPv6 Service Provider	**ok** (1.098s) usando ipv6 ASN 16735

Teste realizado as 16:00 de 19/08/2013 pelo site http://teste-ipv6.com.

7. CONCLUSÃO

Mesmo com todas as melhorias do IPv6 ainda é um protocolo em desenvolvimento, teremos ainda muitos problemas de segurança, instabilidades ainda não descobertos, limitações de hardware ou software como mostrado anteriormente.

Antes de qualquer aquisição de equipamento ou software devemos analisar cuidadosamente todas as características dos equipamentos e como será definida a nossa rede.

Concluindo, devemos ter calma e não migrarmos totalmente os dispositivos para o IPv6, pois a vida do IPv4 ainda será longa e por haver diversos problemas

ainda sem solução, mas o fato de retirar a interação do roteador para a realização do NAT para cada requisição reiterando o modelo fim a fim demostrou uma maior capacidade de processamento e de menor latência para as conexões com a internet.