Uso de APIs para implementação de páginas de avisos para WISPs que utilizam PPPoE como método de autenticação de usuários em RADIUS

Clique para editar o estilo do subtítulo mestre

Eng. Sérgio Ferreira de Brito

Adm. Jorge Fernando Matsudo Iwano

ALOO INTERNET BANDA LARGA

Nossa Empresa

- Fundada em 2003, operação conjunta com operadora de TV a cabo.
- Redes metropolitanas Wi-Fi desde 2004.
- Implantações de HotSpots, Outsourcing, VPN IP e outras soluções IP desde 2004.
- Experiência com Mikrotik desde 2005.
- Experiência em desenvolvimento de sistema ERP corporativo de forma automatizada com Mikrotik desde 2005.
- Rede metropolitana WiMAX não licenciada desde 2007.
- Rede metropolitana de fibra óptica GigabitEthernet desde 2007.

Nossa Empresa

Nossa Cidade

Maceió – AL, Brasil



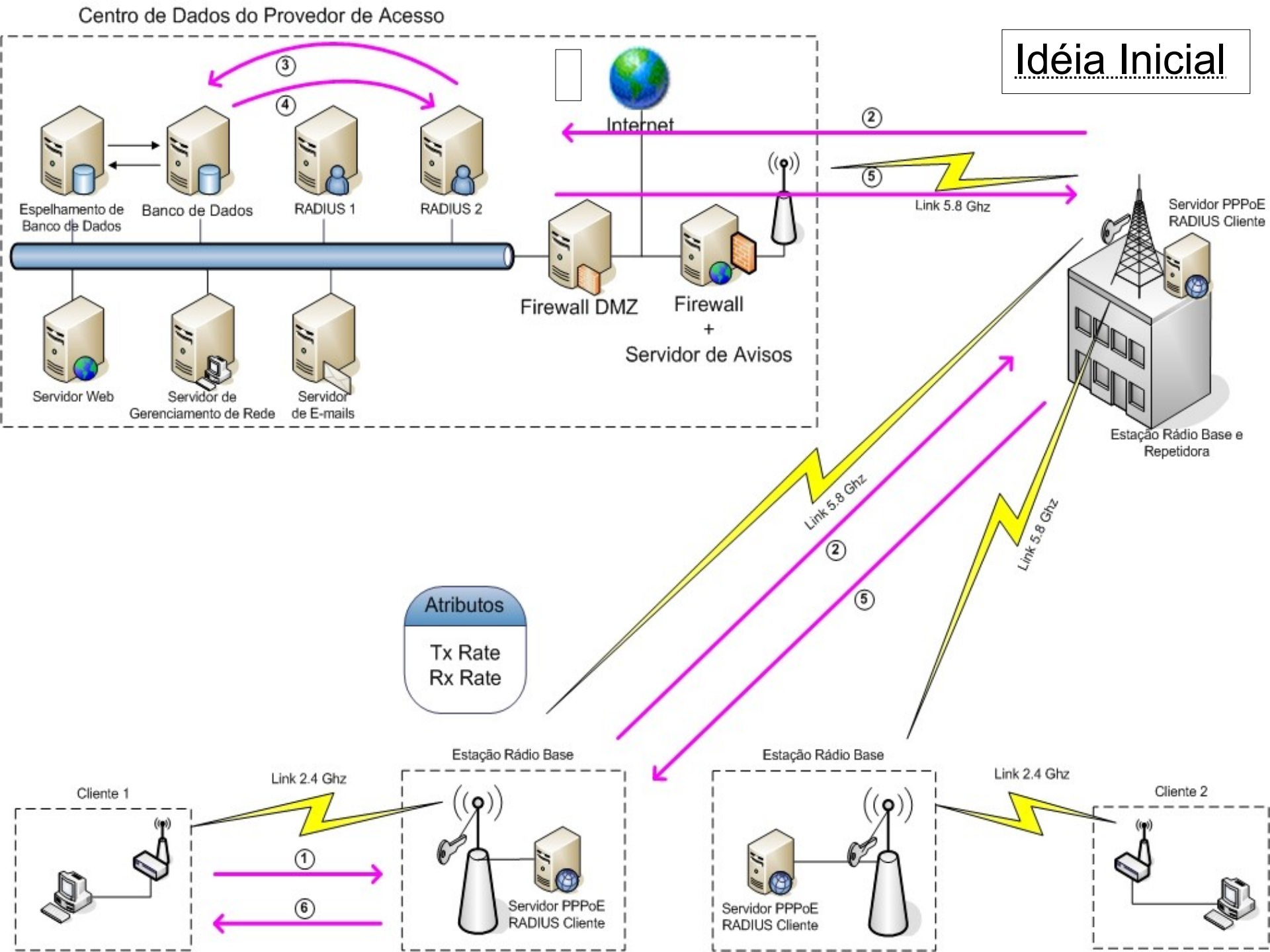


- Público-alvo: Provedores de serviço de internet banda larga wireless que utilizam PPPoE como método de autenticação de usuários em RADIUS integrado com Banco de Dados.

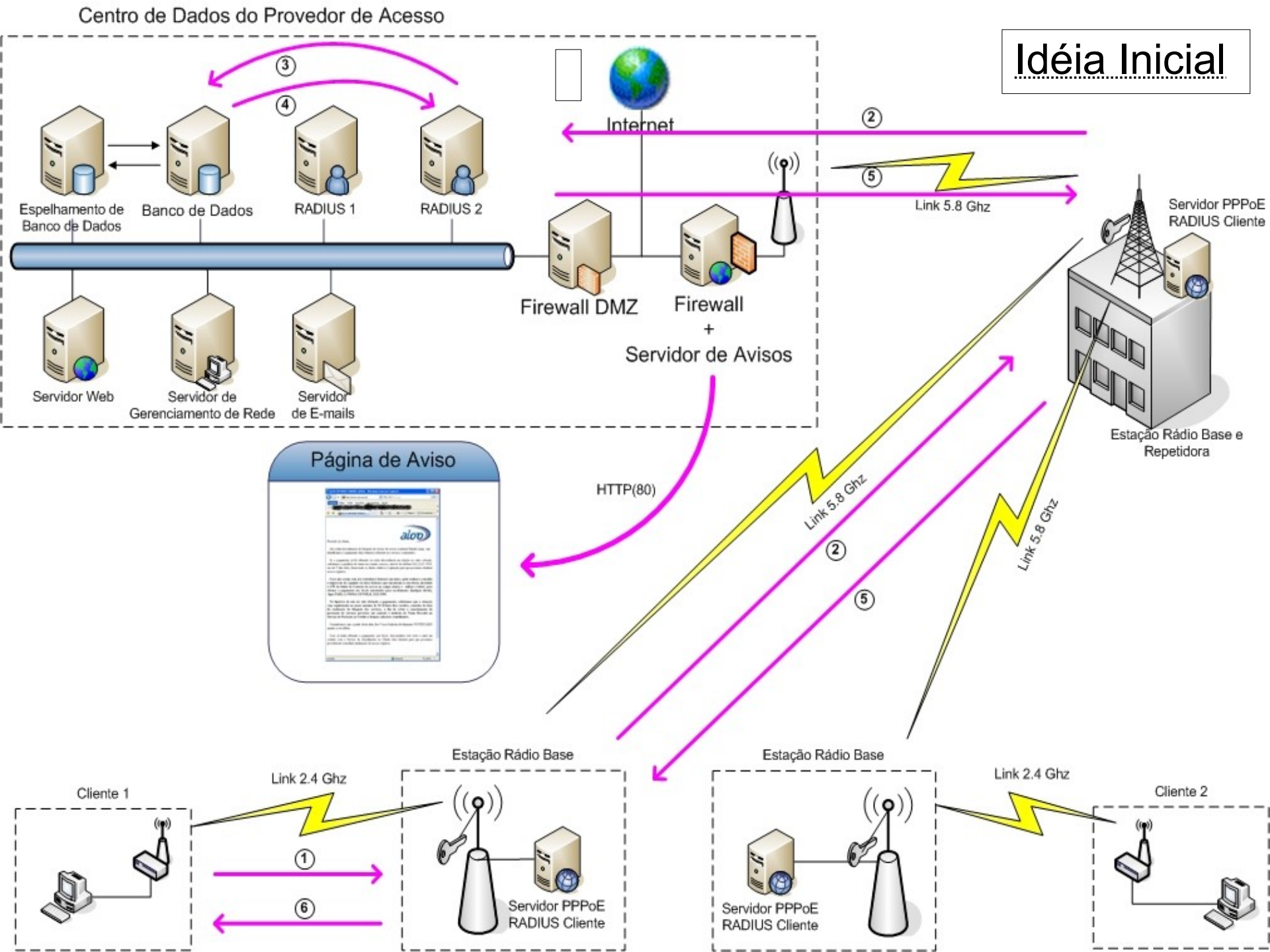
- Pré-requisitos:
  - Conhecimento detalhado de como funciona o método de autenticação PPPoE em RADIUS integrado com Banco de Dados.
  - Uso de RouterOS v3

- Objetivos:
  - Apresentar as vantagens e pré-requisitos de implantação desta solução em comparação com outros métodos

Idéia Inicial



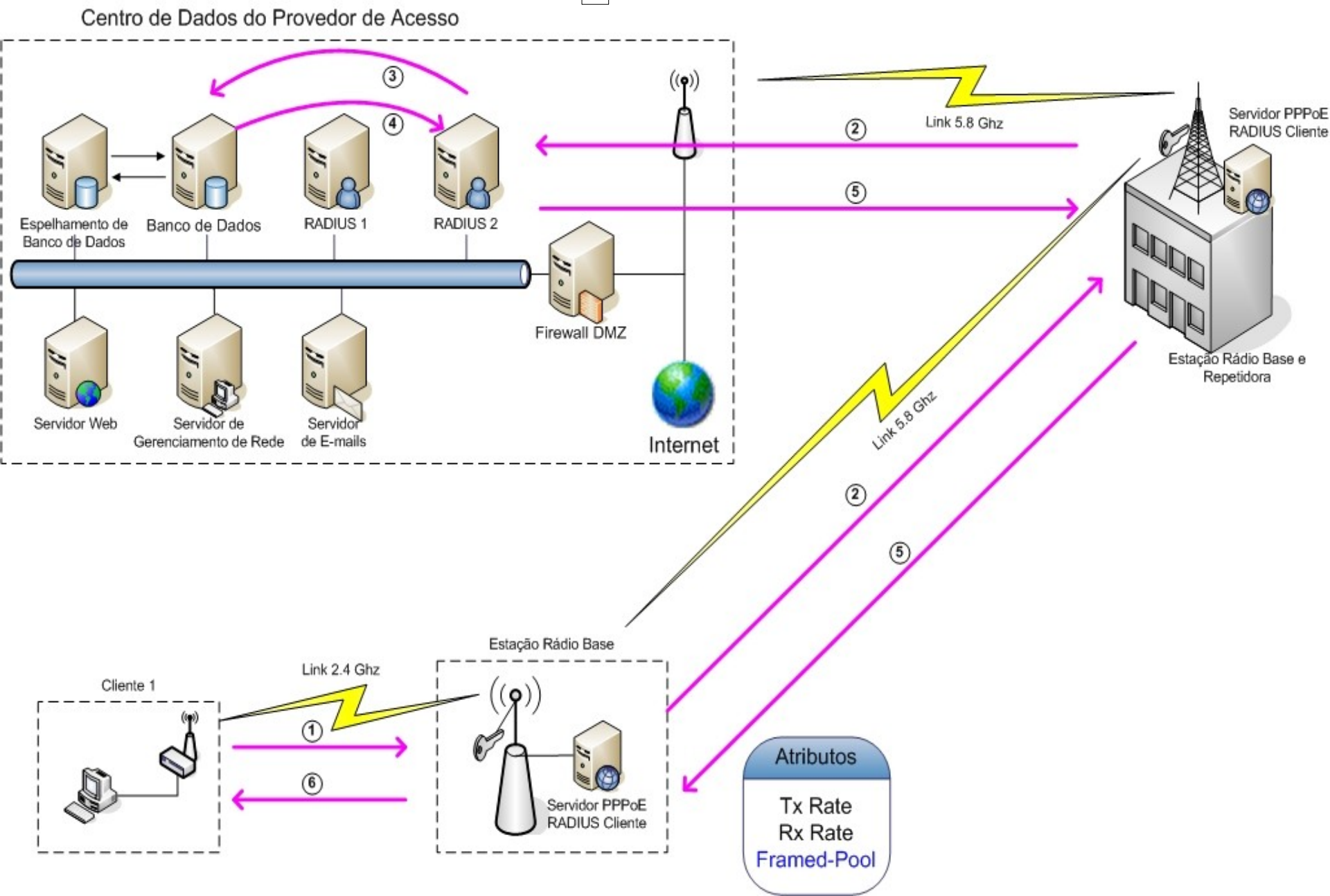
Idéia Inicial



Idéia Inicial – Pontos Fracos

- Servidor de redirecionamento centralizado no POP de interconexão com as operadoras.
- Escalabilidade comprometida, pois com o aumento de usuários teríamos que aumentar a capacidade de processamento e *throughput* do servidor central.
- Mais um ponto de falha, pois trata-se de servidor onde o MTBF de seus componentes é bem menor que equipamentos de Telecomunicações.
- Alto custo de estoque de sobressalentes, devido a necessidade de termos servidor e equipamentos *backup* pois todo o tráfego do provedor passa por ele. Se ocorrer interrupção, teríamos que ter um de igual configuração para substituir.
- Solução não muito fácil para automatização do sistema de avisos com ERP

Cenário Atual – Redirecionamento Estático



Cenário Atual – Redirecionamento Estático

- Conexão PPPoE em RADIUS e Banco de Dados
- Atributos do RADIUS
  - Rate-Limit
  - Framed-IP-Address
  -

7311	4554	radius	Rate-Limit	150k/300k	(Null)
7312	4555	radius	Rate-Limit	2048k/4096k	(Null)
7319	92	radius	Framed-Pool	POOL_BLOQUEADO	(Null)

- Mil

Name	Addresses	Next Pool
DHCP_INTERNO	192.168.0.200-192.168.0.254	none
POOL_BLOQUEADO	192.168.10.2-192.168.10.254	none

Cenário Atual – Redirecionamento Estático

- Mikrotik Firewall
  - Redirecionamento dos acessos na porta de protocolo TCP/80 para o servidor.

```
/ip firewall nat add chain=dstnat src-address=192.168.10.0/24 protocol=tcp dst-port=80
action=dst-nat to-ddresses=IP_SRV_WEB to-ports=80
```

  - Com isso, todos os clientes que receberem o atributo pré-configurado POOL_BLOQUEADO, serão redirecionados para um servidor WEB.

Cenário Atual – Redirecionamento Estático

- Servidores de redirecionamentos descentralizados, pois cada estação Mikrotik fará os redirecionamentos de forma descentralizada otimizando custos e investimentos.
- Escalabilidade, pois com a descentralização de processamento de redirecionamentos não teríamos necessidade de aumentar a capacidade de processamento dos equipamentos nas Estações Rádio Base.
- Sem pontos de falhas, a topologia original será mantida.
- Não teríamos que manter estoque de equipamentos do projeto do servidor de avisos.
- Solução de fácil automatização do sistema de avisos com ERP de nosso ambiente corporativo, pois trata-se somente de mudança de “status”dos clientes em Banco de Dados.

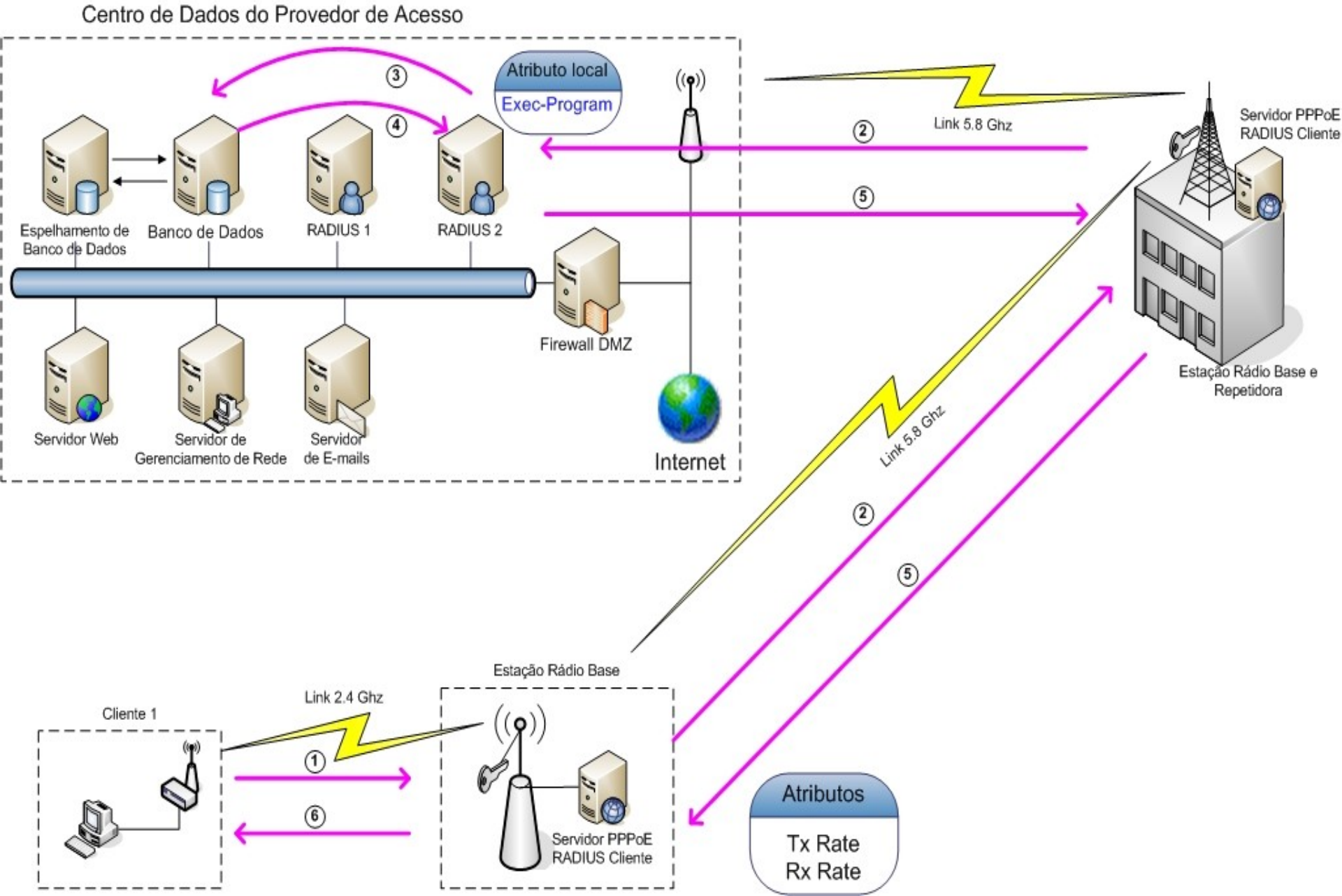
Cenário Atual – Redirecionamento Dinâmico

- Desafios com surgimento de novas necessidades
  - Divulgar avisos a clientes que estão em atraso de pagamento.
  - Divulgar avisos distintos simultaneamente a grupos de clientes diferentes.
  - Automatizar usando o Banco de Dados + RADIUS + PPPoE Server.
  - E como avisar os clientes e estes continuarem acessando internet normalmente, como acontece no HotSpot Mikrotik?

Cenário Atual – Redirecionamento Dinâmico

- Exec-Program
  - Atributo local do RADIUS (freeradius.org) que nos permite executar um aplicativo ou programa após o cliente concluir sua conexão PPPoE com sucesso.
- API Class
  - Obrigado, Denis Basta.
  - Referência: http://wiki.mikrotik.com/wiki/API_PHP_class
  - Classe API que nos permite enviar comandos via script PHP para uma estação Mikrotik e receber respostas.
- Elaboração de script PHP utilizando API Class

Cenário Atual – Redirecionamento Dinâmico



Cenário Atual – Redirecionamento Dinâmico

- Quando o cliente conclui a conexão PPPOE com sucesso, o RADIUS executa um script local (script.sh), através do atributo *Exec-Program*, passando alguns argumentos os quais são variáveis internas do RADIUS .

nome	valor	op
Exec-Program	/usr/local/etc/raddb-teste/script.sh %{User-Name} %{NAS-IP-Address}	(Null)
Sistema-Operacional	Windows XP	(Null)
Rate-Limit	50k/100k	(Null)

```
#!/bin/sh

# arquivo: script.sh



# recebe os argumentos externos
```

Cenário Atual – Redirecionamento Dinâmico

Programa em PHP utilizando API Class e comandos API

```
<?php
// arquivo: radiusExec.php
// inclui a classe API de Denis Basta
include("mikrotikClassAPI.php");

// recebe os arqumentos externos
$fd_usuario = $argv[1];
$fd_estacao = $argv[2];

// cria uma instancia
$API = new routeros_api();
$API->debug = true;

// corrige problema com tempo de estabelecer a conexão pppoe
// do cliente com a torre
sleep(5);
```

Cenário Atual – Redirecionamento Dinâmico

Programa em PHP utilizando API Class e comandos API

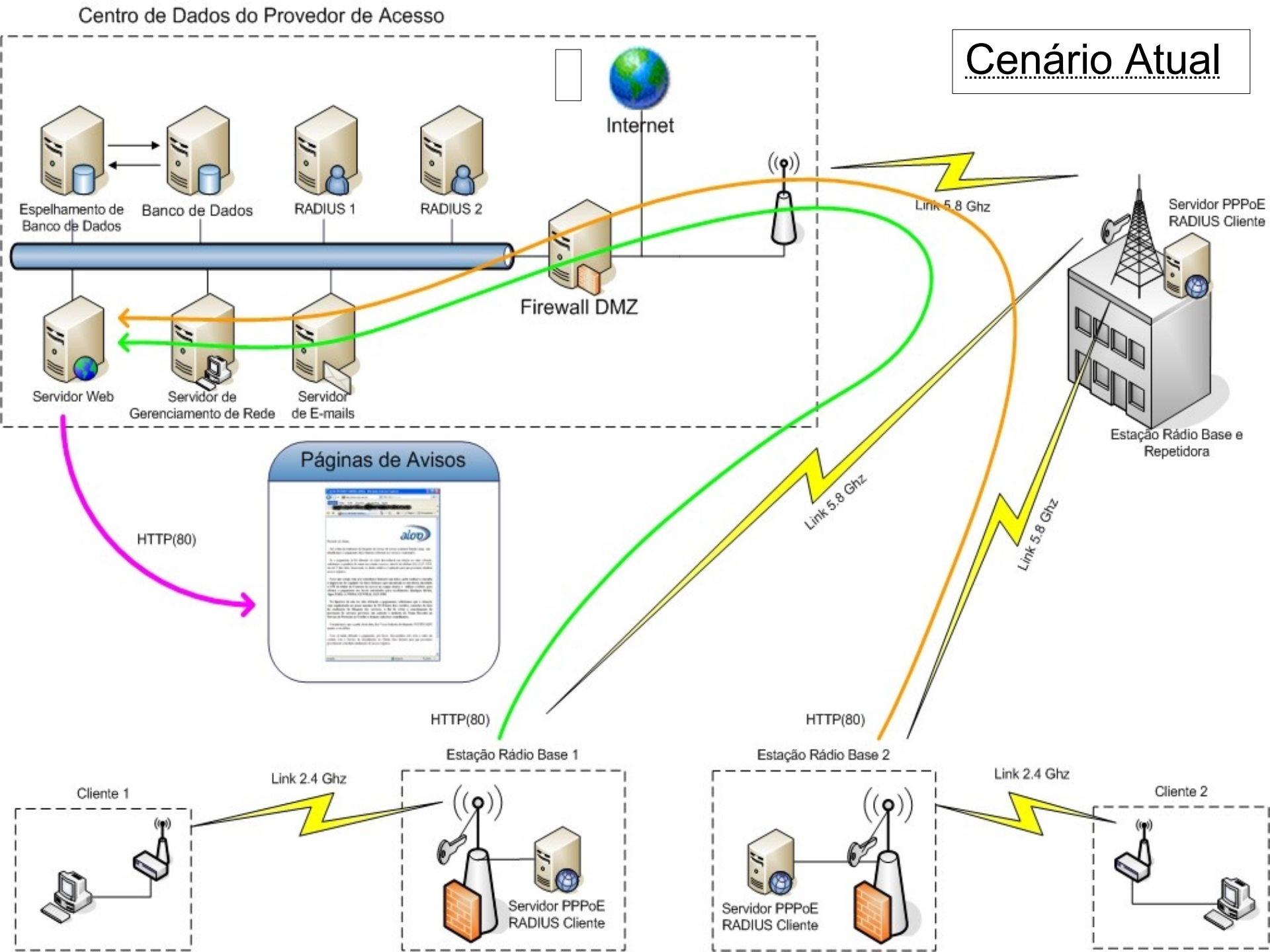
```
// Adicionando regra de firewall redirecionando apenas o cliente desejado.
// Observe que não redirecionamos IP, e sim, a interface que tem nome, por
// padrão <pppoe-login_do_usuário>
if ($API->connect("$fdMikrotikIP", "$fdUsuario", "$fdSenha")) {
   // cria a regra de firewall
   $API->write("/ip/firewall/nat/add",false);
   $API->write("=chain=dstnat",false);
   $API->write("=action=redirect",false);
   $API->write("=to-ports=8080",false);
   $API->write("=in-interface=<pppoe-$fd_usuario>",false);
   $API->write("=protocol=tcp",false);
   $API->write("=disabled=no");
   $READ = $API->read();
   $ARRAY = $API->parse_response($READ);
// identifica ID do firewall
   $codigo_remover = str_replace("=ret=","",$READ[1]);
   // tempo até remover a regra de firewall
   sleep(60);
```

Cenário Atual – Redirecionamento Dinâmico

Programa em PHP utilizando API Class e comandos API

```
// remove a regra de firewall
   $API->write("/ip/firewall/nat/remove",false);
   $API->write("=.id=$codigo_remover");
   $READ = $API->read();
   $ARRAY = $API->parse_response($READ);
   print_r($ARRAY);
   // desconecta do mikrotik
   $API->disconnect();
}
?>
```

Cenário Atual



Cenário Atual – Redirecionamento Dinâmico

- Servidores de redirecionamentos descentralizados;
- Escalabilidade;
- Sem pontos de falhas, a mesma topologia com redirecionamentos estáticos e dinâmicos;
- Não teríamos que manter estoque de equipamentos do projeto do servidor de avisos;
- Solução de fácil automatização do sistema de avisos com ERP de nosso ambiente corporativo.

- Latência na rede
- Processamento muito alto na estação
- Após adicionada a regra de firewall, a estação ficar “fora do ar” por qualquer motivo.

Projetos Futuros

- Como divulgar avisos apenas a um determinado grupo de usuários?
- Divulgação de novos produtos;
- Pesquisa de Pós-Reparo;
- Pesquisa de Pós-Venda;
- Pesquisa de Satisfação;
- Avisos a clientes tarifados que a cota esteja se esgotando (mesmo método utilizado no módulo de HotSpot do Mikrotik);

Muito obrigado !!!

Eng. Sérgio Ferreira de Brito

Adm. Jorge Fernando Matsudo Iwano

ALOO INTERNET BANDA LARGA