

Alta Disponibilidade de Link

Autoria de Danilo Neves
22/12/2011
Última Atualização 22/12/2011

Introduçã o

Muitas vezes, provedores de internet ou até mesmo empresas, não têm 2 servidores para ter uma alta disponibilidade de link e servidor. E com isso, eles apenas garantem a disponibilidade de link em único servidor, isso garante que o usuário não vá ligar reclamando que não consegue navegar e bla bla bla, caso o link principal venha ficar indisponível.

Para fazer isso, vou usar 2 ferramentas no FreeBSD, uma vai ser o Ifstated para fazer o monitoramento dos link e alterar a rota. E a outra vai ser o Packet Filter, o famoso PF.

Caso o link venha ficar indisponível, o Ifstated vai alterar a rota para o outro link.

Caso o link indisponível venha ficar disponível automaticamente, a rota vai voltar para a rota default, para o link principal.

Já com o PF, vou usar para criar os NAT dos clientes em uma única linha. Ele também pode ser usado como Firewall, redundância de link (round-robin ou source-hash), redirecionamento de portas, e etc.

Vamos ao trabalho. Clique abaixo para continuar lendo o artigo todo.

Preparando o Ambiente Precisamos compilar o KERNEL para o PF e instalar o Ifstated.

Primeiro vou compilar o Kernel e para isso vou descobrir qual o nome do meu Kernel atual. `#uname -a/usr/obj/usr/src/sys/GENERIC` Veja que no final temos o nome do Kernel que no meu caso o nome padra o GENERIC. Precisamos agora editar e adicionar algumas opc'o es para que o PF funcione corretamente. Para editar: `# ee /usr/src/sys/i386/conf/GENERIC`

Va ate o final do arquivo e adicione as seguintes opc'o es: `device pf options ALTQ options ALTQ_CBQ options ALTQ_RED options ALTQ_RIO options ALTQ_HFSC options ALTQ_CDNR options ALTQ_PRIQ` Feito isso e so salvar, agora vamos compilar o Kernel modificado. `#cd /usr/src #make buildkernel KERNCONF=GENERIC` Espere porque isso demora um pouco, assim que termina execute o comando abaixo `# make installkernel KERNCONF=GENERIC` Depois e so reiniciar o servidor.

Agora que ja compilamos nosso Kernel para o PF, vamos agora instalar o Ifstated pelo ports. `# cd /usr/ports/net/ifstated/ # make install clean` Finalizamos o ambiente e agora vamos para configurac'a o.

Configurac'o es No ambiente que estou configurando o servidor tem 3 interfaces, duas interfaces do link e uma dos clientes.

Agora precisamos definir o ip de cada uma delas.

No meu caso a interface do meu link principal e a re1, vou configurar ela na ma o e depois configurar no rc.conf para que na hora do boot do servidor ja fique configurada. `Ip da interface #ifconfig re1 187.64.174.186/30 Gateway#route add default 187.64.174.85` Obs: Devemos configurar o DNS `# ee /etc/resolv.conf nameserver 8.8.8.8` Feito isso o servidor ja deve esta disponi vel para internet e agora devemos configurar o segundo link em uma outra interface, que no meu caso e a bge0. `#ifconfig bge0 186.68.173.182/30` Nesse momento na o precisamos colocar o rota default porque ja colocamos no passo anterior e no momento na o precisamos dessa rota default. Por fim vamos configurar a interface do gateway do seus clientes, no meu caso a re2. `# ifconfig re2 192.168.1.1/24` Agora todas as interfaces esta o com IP, agora vou colocar todas essas configurac'o es no rc.conf, na o coloquei antes para fins dida tico. Na o e preciso fazer as configurac'o es das interfaces que eu fiz acima, basta apenas seguir o exemplo do rc.conf e reiniciar o servidor que o mesmo ja vai iniciar com todas as configurac'o es das interfaces. Meu rc.conf ficou da seguinte forma:

```
/etc/rc.conf # CONFIGURACAO DAS INTERFACES#Link 1
ifconfig_re1="inet 187.64.174.186/30&rdquo; #Link 2
ifconfig_bge0="inet 186.68.173.182/30" #Clientes
ifconfig_re1="inet 192.168.1.1/24" #GATEWAY DEFAULT
defaultrouter="187.64.174.85"
```

Pronto, agora esta o todas as interfaces configuradas. Vamos agora configura o PF, para fazer o NAT dos clientes e assim eles poderem sairem para a internet.

Para o PF funcionar e necessa rio adicionar no rc.conf `pf_enable=”YES”`; e tambem o `gateway_enable=”YES”`; , pode ser logo abaixo da configurac'a o do gateway. `#ee /etc/rc.conf pf_enable=”YES”`;

`gateway_enable=”YES”`; Feito isso vamos fazer backup do arquivo de configurac'a o do pf.conf e criar um novo.

`# mv /etc/pf.conf /etc/pf.conf.old` Vamos criar um novo arquivo com as seguintes configurac'o es. `# ee /etc/pf.conf nat on`

re1 from 192.168.1.0/24 to any -> 187.64.174.186 nat on bge0 from 192.168.1.0/24 to any -> 186.68.173.182 Agora so salvar, para carregar o arquivo de configurac'o es devemos digitar `# pfctl -f /etc/pf.conf# pfctl -e /etc/pf.conf` primeira regra no arquivo de configurac'a o fazem que os clientes da faixa 192.168.1.0/24 saiam nateado pelo ip 187.64.174.186 que e meu IP do link principal.A outra regra so vai funcionar caso o primeiro link fique indisponi vel e assim o segundo link vai assumir e todos os clientes va o sair pelo ip 186.68.173.182.

Neste momento todos seus clientes esta o navegando. Agora vem o ifstated a parte mais importante desse artigo.

Vamos fazer um backup do arquivo de configurac'a o padra o ifstated.conf. `#mv /usr/local/etc/ifstated.conf /usr/local/etc/ifstated.conf.old` Vamos criar um novo ifstated.conf com as novas configurac'o es. `#ee /usr/local/etc/ifstated.conf init-state master if_up="re1.link.up" net = '("ping -q -c 1 -t 1 187.64.174.85 > /dev/null" every 5)' state master {`

`if ($net) { run "route delete default && route add default 187.64.174.85" }`

`if (! $net) {`

`run "route delete default && route add default 186.68.173.181" }` } Veja que o arquivo de configurac'a o na o tem segredo nenhum.

Ele faz uma checagem de ping no seu gateway principal 187.64.174.85 a cada 5 segundos e caso nesse tempo o gateway fique indisponi vel ele vai alterar a rota para o 186.68.173.181. Depois de 5 segundo ele faz um checagem novamente o gateway principal e caso ele volte a ficar disponi vel, automaticamente vai alterar a rota para o gateway principal.

Voce pode escolher para fazer a checagem de ping no gateway do seu roteador e tambe m diminuir ou aumentar o tempo de checagem, isso depende de voce . Feito tudo isso preciso colocar daemon para fazer a checagem, digitando:`#ifstated -f /usr/local/etc/ifstated.conf` Pronto, agora o daemon esta rodando e checando seu link a cada 5 segundos e assim temos um alta disponibilidade. Mas agora voce se pergunta, e se o servidor reiniciar o que vai acontecer?

Isso e um problema porque esse daemon na o vai ser carregado automaticamente e sendo carregado ele manualmente.

Calma que para tudo tem uma soluc'a o!

Para isso arrumei um script e com algumas alterac'o es minhas e do meu colega de trabalho Renan Fagundes fizemos um script simples para resolver esse problema. Dentro do direto rio de inicializac'a o edite o seguinte script#`ee`

`/usr/local/etc/rc.d/link.sh #!/bin/sh #echo -n ' link ' case "$1" in start)`

`start=`ls -la /var/run/link.pid 2>&1``

`if [$? -eq 1]`

`then`

`/usr/local/sbin/ifstated -f /usr/local/etc/ifstated.conf`

`ps -auxw | grep /usr/local/sbin/ifstated | grep -v grep |awk '{print $2}' -> /var/run/link.pid echo "Iniciando Alta Disponibilidade" else`

`echo "Disponivel" fi ;; stop) stop=`ls -la /var/run/link.pid 2>&1` if [$? -eq 0]`

`then`

`kill -9 `cat /var/run/link.pid` rm /var/run/link.pid`

`echo "Desativando Alta Disponibilidade" else`

`echo "Indisponivel"`

`fi`

`;; echo "Usage: `basename $0` {start|stop}" >&2 exit 64`

`;; esac exit` Agora so salvar e alterar permissao es do arquivo e permissao de execucao o: `#chmod 755`

`/usr/local/etc/rc.d/link.sh #chmod +x /usr/local/etc/rc.d/link.sh` Agora vou explicar o que o script faz de uma forma ra pida e simples. Todo daemon tem um PID, entao eu usei esse PID para poder matar o processo e com esse script posso dar um start ou stop na hora que eu quiser, alem de ele carregar automaticamente na hora da inicializac'a o do servidor.

Ao iniciar o script ele gera um arquivo chamado de /var/run/link.pid dentro desse arquivo esta o PID do daemon Ifstated.

Quando voce quer para o servico ele vai ler esse arquivo verificar qual o PID que esta dentro dele e matar esse PID e excluir o arquivo /var/run/link.pid .

E isso que o script faz, sem segredo nenhum. Start no servico#`/usr/local/etc/rc.d/link.sh start` Stop no

servico#`/usr/local/etc/rc.d/link/sh stop` Pronto, agora e so fazer os testes.Espero que gostem e aceito criticas, sugestoes e elogios. :)