



Скрипт автоматического переключения между основным и резервным каналом

Опубликовано muff в Ср, 2010-10-27 17:00

Есть офисный маршрутизатор на базе **FreeBSD** и два приходящих канала **Internet**.

Поставлена задача: автоматизировать переключения с основного канала на резервный в случае его падения и обратно, после возобновления работы основного канала.

В "закромах Родины" завалялся **perl**-скрипт, происхождение которого установить не получилось. Скорее всего был скачан "на всякий случай" из инета.

Итак, проанализировав содержимое скрипта, выполняем установку необходимого расширения **Perl - Net-Ping**:

```
# cd /usr/ports/net/p5-Net-Ping-External && make install clean && rehash
```

Далее правим переменные в скрипте. Листинг скрипта следующий:

```
#!/usr/local/bin/perl -wuse strict;use warnings;use Net::Ping::External (qw(ping));  
# 1 - автоматический режим переключение канала# 2 - принудительное переключение на второй канал.my $action = 1;my $host_gw = "192.168.213.129";  
# default gatewaymy $gw = "192.168.178.209"; # backup gateway  
my $now = localtime time;my $log = "/var/log/route-change.log";if($action == 1){my $command = `netstat -rn | grep default`;my @b = split('\s+', $command, 3);if (ping(hostname =>$host_gw, count=>5)){ print "host $host_gw is ok\n"; if($b[1] ne $host_gw){  
    if($b[1] eq ""){ `route add default $host_gw`;}else{ `route change default $host_gw`;  
    open(LOG,>>$log); print LOG "[!] $now Route change to $host_gw\n"; close(LOG); } }else{ print "host $host_gw is bad.\n"; if($b[1] ne $gw){ `route change default $gw`;  
    open(LOG,>>$log); print LOG "[!] $now Route change to $gw\n"; close(LOG); } } }if($action == 2){my $command = `netstat -rn | grep default`;my @b = split('\s+', $command, 3);if($b[1] ne $gw){  
    if($b[1] eq ""){ `route add default $gw`;}else{ `route change default $gw`;  
    open(LOG,>>/change_route.log); print LOG "[!] $now Route change to $gw\n";  
    close(LOG); } }}
```

Разместим данный скрипт под названием **route-change.pl** в каталоге **/usr/local/etc**.

Проверим дефолт до запуска скрипта:

```
# netstat -rn | grep default  
default      192.168.213.129      UGS      335 51206943 vlan32
```

Запустим скрипт:

```
# perl /usr/local/etc/route-change.pl  
host 192.168.213.129 is ok
```

Проверяем значение маршрута по умолчанию:

```
# netstat -rn | grep default  
default      192.168.213.129      UGS      346 51212415 vlan32
```

Проверим, выполняется ли переключение, выполнив имитацию падения канала.



```
# ipfw add 1 deny icmp from any to 192.168.213.129
00001 deny icmp from any to 192.168.213.129

# perl /usr/local/etc/route-change.pl
host 91.204.213.129 is bad.

# netstat -rn | grep default
default      192.168.178.209    UGS      521 51732384 wlan33
```

Все работает! Осталось только проверить, происходит ли обратное переключение на основной канал:

```
# ipfw 1 delete

# perl /usr/local/etc/route-change.pl
host 91.204.213.129 is ok

# netstat -rn | grep default
default      192.168.213.129    UGS      374 51751262 wlan32
```

Проверим, ведется ли логгирование переключений:

```
# cat /var/log/route-change.log
[!] Thu Oct 28 16:13:45 2010 Route change to 85.223.178.209
[!] Thu Oct 28 16:16:41 2010 Route change to 91.204.213.129
```

Все гуд... Последний штрих - добавим запуск скрипта в cron с ежеминутным запуском:

```
# echo '# Route check and change' >> /etc/crontab
# echo '* * * * *      root    /usr/bin/perl /usr/local/etc/route-change.pl > /dev/null 2>&1'
>> /etc/crontab
# killall -HUP cron
```

Преимущество данного скрипта - простота и надежность.

Недостатком можно считать то, что он проверяет только доступность шлюза. В случае, если инет "упал" за шлюзом, переключение выполняться не будет. Как вариант - можно прописать статический маршрут к какому-то надежному серверу (например, к одному из серверов google.com [1]) через основной канал и выполнять проверку доступности именно этого сервера. Выполнить модификацию скрипта в таком случае не так уж и сложно...

Источник (получено 2026-01-30 06:00):

<http://muff.kiev.ua/content/skript-avtomaticheskogo-pereklyucheniya-mezhdu-osnovnym-i-rezervnym-kanalom>

Ссылки:

[1] <http://www.google.com>



Скрипт автоматического переключения между основным и резервным каналом

Опубликовано muff.kiev.ua (<http://muff.kiev.ua>)
