Dhcdrop - блокировка сторонних DHCP-серверов в сети

Опубликовано muff в Втр, 2012-05-08 15:44

DHCP (***Dynamic Host Configuration Protocol*** — протокол динамической конфигурации узла) — это сетевой протокол, позволяющий компьютерам автоматически получать **IP-адрес** и другие параметры, необходимые для работы в сети **TCP/IP**.

Чтобы не говорили, а использование автоматического назначения IP-адресации в сети намного упрощает жизнь и администратору, и конечному пользователю. Однако, при эксплуатации такой сети, можно столкнуться с появлением в сети сторонних **DHCP-серверов** (маршрутизатор, включенный в сеть **LAN-портом**; сетевые устройства в режиме моста; **DHCP-сервер** на компьютере абонента и тд.). В случае, когда сеть построена на управляемых коммутаторах, бороться с этим явлением довольно просто - с помощью **ACL**. А что делать, если сеть построена на неуправляемых коммутаторах? Как блокировать работу сторонних **DHCP-серверов** без ущерба для работы сети и не "мешая" основному **DHCP-серверу**?

Одним из наиболее действенных инструментов выступает программа **dhcdrop**. Она обеспечивает поиск сторонних **DHCP-серверов** и их подавление путем исчерпания пула **IP-адресов** (**DHCP starvation**). Более подробная информация - на сайте проекта [1].

Что радует - так это то, что **dhcdrop** есть в портах. Выполним установку:

```
# cd /usr/ports/net-mgmt/dhcdrop && make install clean && rehash
```

По завершению установки советую обратиться к страницам man-руководства... Ознакомимся со списком доступных опций (свои комментарии выделил красным):

-h - показывает help-сообщение.
-D - просмотр списка имён и индексов сетевых интерфейсов. Актуально в ОС Windows.
-t - режим теста. В этом режиме dhcdrop не выполняет подавление сервера. Производится лишь посылка DHCPDISCOVER, если на него приходит ответ нелегального сервера, то программа заверается возвращаяя код 200 и выводя на экран строку вида DHCP SRV: 10.7.7.1 (IP-hdr: 10.7.7.1) SRV ether: 00:02:44:75:77:E4, YIP: 10.7.7.205 содержащую минимум информации о создающем проблему DHCP сервере.
-y - подразумевается ответ "yes" на любой вопрос программы. *(позволяет автоматически соглашаться с предложениями «давить» найденные сервера)*
-r - отключает рандомизацию MAC адреса источника. Каждый последующий MAC адрес источника увеличивается на 1.
-b - указывает на необходимость использования флага BROADCAST в отправляемых DHCP пакетах.
-a - всегда ожидать ответа сервера на порт DHCP клиента по умолчанию (68), даже если задано значение номера порта клиента отличное от значения по умолчанию.
-A - всегда ожидать ответа с порта DHCP сервера по умолчанию (67), даже если задано значение номера порта сервера отличное от значения по умолчанию.
-f - режим флуда запросами DHCPDISCOVER. ПРИМЕНЯТЬ С ОСТОРОЖНОСТЬЮ. Удобен для стресс-тестирования сервера. В случае указания опции -r все отправляемые пакеты имеют одинаковый MAC адрес.
-R - отправляет сообщение DHCPRELEASE с MAC адресом источника указанном при запуске программы и IP адресом указанным при помощи опции -F к серверу указанному опцией -s.
-q - "тихий" режим работы. Выводится минимум информации.
-m count - максимальное число попыток получения ответа от сервера. *(по умолчанию 255, в редких случаях необходимо увеличивать)*
-c count - максимальное число адресов арендуемых у сервера.

-n hostname - значение DHCP опции HostName (по умолчанию - "DHCP-dropper")
-N clientname - значение DHCP опции Vendor-Class (по умолчанию - "DHCP-dropper")
-p port - порт используемый клиентом для отправки DHCP сообщений. По умолчанию - 68.
-P port - порт сервера, на который отправляются DHCP сообщения. По умолчанию - 67.
-w секунд - задаёт таймаут рестарта процесса получения IP адресов в случае использования агрессивного режима. По умолчанию - 60 секунд.
-T timeout - устанавливает таймаут ожидания ответа сервера (в секундах). По умолчанию - 3 секунды.
-M хостов-максимум - максимально допустимое количество сканируемых хостов в случае использования агрессивного режима.
-l MAC-address - Ethernet адрес сервера который необходимо игнориновать при выполненении поиска ложных DHCP серверов в сети. В этой опции следует указать адрес DHCP сервера ответственного за раздачу адресов в данном сегменте сети. Может быть указано несколько адресов - каждый должен предваряться ключом -l.
-L легальная-сеть - указывает легальную IP подсеть для выбранного интерфейса. Использование этой опции автоматически включает агрессивный режим получения IP адресов. Может быть указано несколько сетей - каждая должна предваряться ключом -L. Подробное описание смотрите ниже.
-S сеть/маска - ARP сканирование сети 'сеть' с использованием сетевой маски 'маска' в CIDR нотации. IP адрес источника задаётся опцией -F. Если IP адрес источника не задан - используется случайный адрес из диапазона указанной подсети. Пример использования смотрите ниже.
-F исходящий-IP-адрес - указывает IP адрес источника для сканирования сети (опция -S), либо IP адрес DHCP клиента для отправки сообщения DHCPRELEASE (опция -R).
-s IP-адрес-сервера - задаёт IP адрес DHCP сервера. Используется с опцией -R.
-C count - число порождаемых процессов-потомков. Совместим только с флагом -f. Используется для увеличения числа отправляемых пакетов за единицу времени. При значении этого параметра равном 30, 10000 пакетов генерировалось менее чем за 1,5 секунды.
-i interface - имя либо индекс сетевого интерфейса (см. ключ -D). Не может быть "any"! Единственный обязательный параметр программы.
initial MAC address - задаёт MAC адрес источника используемый при отправке первого DHCP сообщения, либо используемый постоянно, в случае использования опции '-f' (flood) вместе с опцией '-r'. Если не указан, то используется случайный MAC адрес источника.

Отметим также коды выхода программы (надеюсь, что все читатели "дружат" с английским языком):

0 - Exit success. Illegal DHCP server not found.
10 - invalid user ID. You must be root for running programm.
20 - failed to set signal handler.
30 - configuration error. See usage.
40 - memory allocation error. Insufficient memory?
50 - error opening ethernet device.
51 - error listing devices.
60 - pcap filter overflow.
70 - pcap compile error.
80 - pcap set filter error.
90 - error sending packet.
100 - error getting packet.
110 - set non blocked mode error.
120 - invalid device.
200 - illegal DHCP server was found.

Соответственно, подберем следующие ключи для запуска **dhcdrop** на интерфейсе **vlan51**:

```
# dhcdrop -y -r -m 3 fe:fe:fe:fe:fe:00 -l 00:0b:cd:68:78:cc -i vlan51
```

Здесь:

- **00:0b:cd:68:78:cc** - **MAC-адрес** легитимного **DHCP-сервера**.
- **vlan51** - интерфейс **FreeBSD-сервера**, который "смотрит" в сеть с посторонним **DHCP-сервером**

Пример работы **dhcdrop**:

```
# dhcdrop -y -r -m 3 fe:fe:fe:fe:fe:00 -l 00:0b:cd:68:78:cc -i vlan51

Using interface: 'vlan51'
Got response from server 192.168.85.65 (IP-header 192.168.85.65), server ethernet address:
B0:48:7A:F0:47:1C, lease time: 0.083h (300s)
Got BOOTREPLY (DHCPOFFER) for client ether: FE:FE:FE:FE:FE:00 You IP: 192.168.85.68/26
1. Got BOOTREPLY (DHCPACK) for client ether: FE:FE:FE:FE:FE:00 You IP: 192.168.85.68/26
2. Got BOOTREPLY (DHCPACK) for client ether: FE:FE:FE:FE:FE:01 You IP: 192.168.85.69/26
3. Got BOOTREPLY (DHCPACK) for client ether: FE:FE:FE:FE:FE:02 You IP: 192.168.85.70/26
4. Got BOOTREPLY (DHCPACK) for client ether: FE:FE:FE:FE:FE:03 You IP: 192.168.85.71/26
5. Got BOOTREPLY (DHCPACK) for client ether: FE:FE:FE:FE:FE:04 You IP: 192.168.85.72/26
Interrupted. Quit.
```

Также стоит отметить некоторые моменты работы с **dhcdrop**:

- в режиме флуда (флаг **-f**) программа не выбирает адреса из пула сервера, а просто "валит" сеть паразитным трафиком
- если запустить программу с числом запросов 1200 (например на час), то обнаруженный сервер будет подавлен только один раз. Особенностью встроенных в аппаратные роутеры **DHCP-серверов** является их автоматическое очищение пула адресов и восстановление работоспособности через определенный интервал времени. Очевидно, что нужно часто перезапускать программу

Итак, проблема восстановления работоспособности нелегальных серверов так и не решена. Как вариант - запуск **dhcdrop** с определенной периодичностью, используя **cron**. Однако более удобно использовать следующий скрипт:

```
#!/bin/shDATE="/bin/date"LOGDIR="/var/log/dhcdrop/"LEGAL_SERVERS="00:0b:cd:68:78:cc
"      # Legal DHCP Servers, space separated mac address
DROPPER="/usr/local/sbin/dhcdrop"IFNAME="vlan50 vlan51 vlan52
"         # Interfaces on our Router, space separated
TESTPARAMS="-t -m 3"PARAMS="-y -r -m 3 fe:fe:fe:fe:fe:00"

# Lets Go!# legal paramsfor mac in ${LEGAL_SERVERS}; do    LMAC="${LMAC} -l ${mac}"done
# Discovering on every interface
for IF in ${IFNAME}; do        echo "Processing interface ${IF}"        # test to any DHCP-Server
      ${DROPPER} -i ${IF} ${LMAC} ${TESTPARAMS}

         # Check for status 200
      if [ $? = 200 ]; then           echo "Illegal server found on ${IF}! Dropping him!"           ${DATE}
 >> ${LOGDIR}${IF}           ${DROPPER} ${PARAMS} ${LMAC} -i ${IF} >> ${LOGDIR}${IF}
else           echo "Illegal server not found on ${IF}."        fidoneecho "All done"
```

Для корректной работы скрипта необходимо создать каталог, куда будут писаться логи:

```
# mkdir /var/log/dhcdrop
```

После этого добавим в **cron** запуск скрипта каждые 10 минут. У меня путь к скрипту **/usr/local/etc/dhcdrop.sh**, соответственно:

```
# echo '# DHCDROP' >> /etc/crontab
```

```
# echo '*/10 * * * * root    /bin/sh /usr/local/etc/dhcdrop.sh > /dev/null 2>&1' >> /etc/crontab
```

После этого "забываем" о сторонних DHCP-серверах в неуправляемых сегментах сети. Однако, у злоумышленника еще остаются некоторые лазейки:

- увеличить пул выдаваемых адресов больше 255 - в таком случае необходимо будет установить переменную **PARAMS** в значение, например, "-y -r -m 3 -c 4096 fe:fe:fe:fe:fe:00". Значение ключа необходимо подобрать опытным путем.
- установить короткое время лизинга IP-адресов (до 1-2 минут). Тогда **dhcdrop** будет постоянно уходить в цикл.

Источник (получено *2026-01-14 17:08*):
http://muff.kiev.ua/content/dhcdrop-blokirovka-storonnikh-dhcp-serverov-v-seti

Ссылки:
[1] http://www.netpatch.ru/dhcdrop.html