FTP + NAT

Введение

FTP (File Transfer Protocol - Протокол передачи файлов). FTP является протоколом высокого уровня, а именно, уровня приложений (7й уровень модели OSI). FTP служба построена по хорошо известной схеме «клиент-сервер», которая позволяет пользователю передавать/получать файлы с удаленного сервера.

Клиент посылает запросы серверу и принимает файлы. В качестве клиентов могут использоваться - internet explorer, Windows Commander, NetVampir, gftp и т.д.

Сервер обрабатывает запросы клиента на получение файла. В качестве серверов могут использоваться - vsftpd, IIS, wuftpd, proftpd и т.д.

FTP сервер - компьютер, на котором запущена программа FTP сервера. Общедоступные FTP сайты обычно доступны любому, зашедшему под именем anonymous или ftp. Имеется много отличных FTP сайтов, на которых есть архивы свободно распространяемого программного обеспечения для Unix.

FTP отличается от других приложений тем, что он использует два TCP соединения для передачи файла:

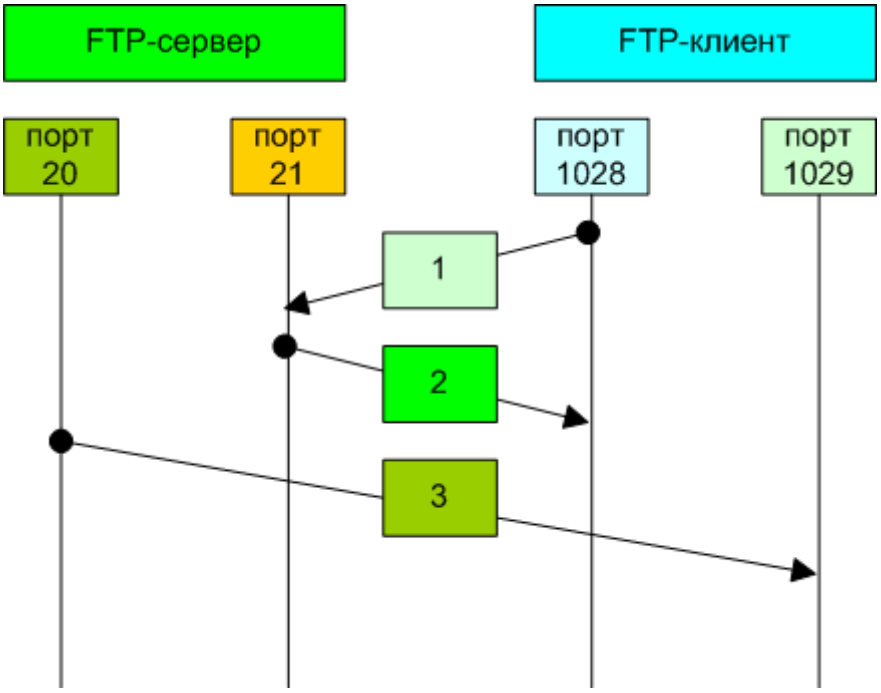
- Управляющее соединение - соединение для посылки команд серверу и получение ответов от него.
- Соединение данных - соединение для передачи файлов.

FTP был впервые разработан в калифорнийском университете для включения в 4.2BSD (Berkeley Unix). Для более детального ознакомления рекомендую прочитать RFC959.

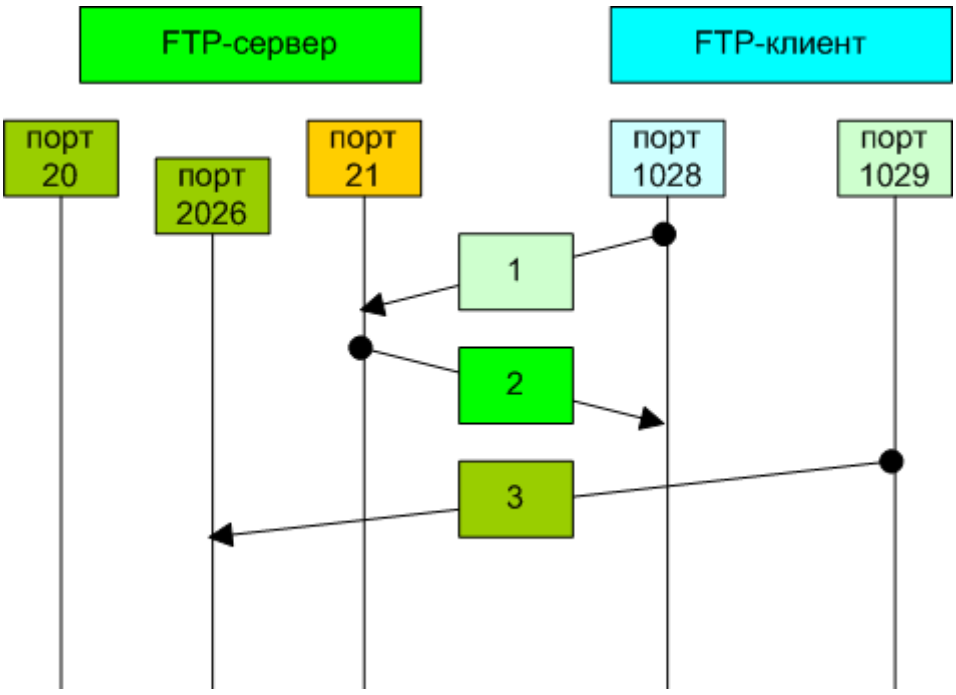
Режимы работы FTP: активный и пассивный

При работе по протоколу FTP между клиентом и сервером устанавливается два соединения - управляющее (по нему идут команды) и соединение передачи данных (по нему передаются файлы). Управляющее соединение одинаково для Активного и Пассивного режимов. Клиент инициирует TCP-соединение с динамического порта (1024-65535) к порту номер 21 на FTP-сервере и говорит «Привет! Я хочу подключится к тебе. Вот мое имя и мой пароль». Дальнейшие действия зависят от того, какой режим FTP (Активный или Пассивный) выбран.

В Активном режиме, когда клиент говорит «Привет!» он так же сообщает серверу номер порта (из динамического диапазона 1024-65535) для того, чтобы сервер мог подключиться к клиенту для установки соединения для передачи данных. FTP-сервер подключается к заданному номеру порта клиента используя со своей стороны номер TCP-порта 20 для передачи данных.



В Пассивном режиме, после того как клиент сказал «Привет!», сервер сообщает клиенту номер TCP-порта (из динамического диапазона 1024-65535), к которому можно подключится для установки соединения передачи данных. Главное отличие между Активным режимом FTP и Пассивным режимом FTP - это сторона, которая открывает соединение для передачи данных. В Активном режиме, клиент должен принять соединение от FTP-сервера. В Пассивном режиме, клиент всегда инициирует соединение.



Активный FTP «выгоден» для FTP-сервера, но «вреден» для стороны клиента. FTP сервер пытается соединиться со случайными высокими (по номеру) портами на клиенте, такое соединение наверняка будет блокировано брандмауэром на стороне клиента.

Пассивный FTP «выгоден» для клиента, но «вреден» для FTP-сервера. Клиент будет делать оба соединения к серверу, но одно из них будет к случайному высокому порту, такое соединение наверняка будет блокировано брандмауэром на стороне сервера.

Первоначальная настройка брандмауера

Создаем минимальный набор правил в нашем фаерволе. Этот пример создан лишь для того, чтобы показать принцип работы iptables. Для более углубленного понимания работы iptables рекомендую прочитать статью Оскара Андерсона в переводе Андрея Кисилева - Iptables Tutorial 1.1.19, которая считается классикой.

```
#
# /usr/local/firewall.sh
#

#!/bin/sh
IPTABLES="/sbin/iptables"

# Отчищаем все правила в таблицах filter, nat и mangle

$IPTABLES -t filter -F
$IPTABLES -t nat -F
$IPTABLES -t mangle -F

# Удаляем все пользовательские цепочки в таблицах filter, nat и mangle

$IPTABLES -t filter -X
$IPTABLES -t nat -X
$IPTABLES -t mangle -X

# Задаем политики по умолчанию

$IPTABLES -t filter -P INPUT DROP
$IPTABLES -t filter -P FORWARD DROP
$IPTABLES -t filter -P OUTPUT ACCEPT

# Создаем пользовательские цепочки. Как строить firewall каждый решает сам.
# Но лично мне удобно настраивать фаервол, когда все разбито по цепочкам
# и как бы разложено по своим полочкам. Тогда я точно знаю, что и где надо
# искать или исправлять. Думаю, названия цепочек говорят сами за себя.

$IPTABLES -N eth0-eth1
$IPTABLES -N eth1-eth0
$IPTABLES -N eth1-in
$IPTABLES -N eth0-in

# Направляем все входящие пакеты в соответствующие цепочки.

$IPTABLES -A INPUT -d 192.168.1.1 -j eth0-in
$IPTABLES -A INPUT -d 212.42.65.100 -j eth1-in

# Для удобства фильтрации, направляем все транзитные пакеты,
# в соответствующие цепочки.
# eth0-eth1 локальная сеть -> мир
```

```
# eth1-eth0 мир -> локальная сеть

$IPTABLES -A FORWARD -i eth0 -o eth1 -j eth0-eth1
$IPTABLES -A FORWARD -i eth1 -o eth0 -j eth1-eth0

# eth0-eth1. В данную цепочку попадают все транзитные пакеты,
# направленные из локальной сети в мир.

$IPTABLES -A eth0-eth1 -p tcp --dport 21 -j ACCEPT
$IPTABLES -A eth0-eth1 -m state --state RELATED,ESTABLISHED -j ACCEPT
$IPTABLES -A eth0-eth1 -j LOG --log-prefix "eth0-eth1 " --log-level 7
$IPTABLES -A eth0-eth1 -j DROP

# eth1-eth0. В данную цепочку попадают все транзитные пакеты,
# направленные из мира в локальную сеть.

$IPTABLES -A eth1-eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
$IPTABLES -A eth1-eth0 -j LOG --log-prefix "eth1-eth0 " --log-level 7
$IPTABLES -A eth1-eth0 -j DROP

# eth0-in. В данной цепочке открываем порты тех служб, которые
# должны быть доступны из локальной сети.

$IPTABLES -A eth0-in -i lo -j ACCEPT
$IPTABLES -A eth0-in -p tcp --dport 21 -j ACCEPT
$IPTABLES -A eth0-in -p tcp --dport 22 -j ACCEPT
$IPTABLES -A eth0-in -p tcp --dport 25 -j ACCEPT
$IPTABLES -A eth0-in -p udp --dport 53 -j ACCEPT
$IPTABLES -A eth0-in -p tcp --dport 80 -j ACCEPT
$IPTABLES -A eth0-in -p tcp --dport 110 -j ACCEPT
$IPTABLES -A eth0-in -m state --state RELATED,ESTABLISHED -j ACCEPT
$IPTABLES -A eth0-in -j LOG --log-prefix "eth0-in " --log-level 7
$IPTABLES -A eth0-in -j DROP

# eth1-in. В данной цепочке открываем порты тех служб, которые
# должны быть доступны из мира.

$IPTABLES -A eth1-in -p tcp --dport 25 -j ACCEPT
$IPTABLES -A eth1-in -p tcp --dport 80 -j ACCEPT
$IPTABLES -A eth1-in -m state --state RELATED,ESTABLISHED -j ACCEPT
$IPTABLES -A eth1-in -j LOG --log-prefix "eth1-in " --log-level 7
$IPTABLES -A eth1-in -j DROP

# Производим сетевую трансляцию адресов (NAT)

$IPTABLES -A POSTROUTING -t nat -s 192.168.1.0/24 -o eth1 -j SNAT --to-
source 212.42.65.100
```

eth0 - смотрит в локальную сеть, eth1 - смотрит в мир. Данным скриптом (firewall.sh) удобно пользоваться на стадии отладки правил.

```
# chmod +x /usr/local/firewall.sh
# /usr/local/firewall.sh
```

После того, как у вас все настроено, лучше воспользоваться штатными средствами управления iptables. Для этого воспользуемся скриптом для сохранения текущих правил в файл. В разных дистрибутивах набор правил может сохраняться по отличному от /etc/sysconfig/iptables пути. Данный путь является стандартным для Red Hat дистрибутивов и его клонов, например CentOS.

```
# iptables-save > /etc/sysconfig/iptables
# chkconfig --level 35 iptables on
# service iptables restart
Flushing firewall rules:                                   [  OK  ]
Setting chains to policy ACCEPT: mangle filter nat         [  OK  ]
Unloading iptables modules:                                [  OK  ]
Applying iptables firewall rules:                          [  OK  ]
```

Но при таких настройках фаервола ftp через nat еще не будет работать. Для облегчения нашей задачи воспользуемся специальным модулем, который и был разработан как раз для этих целей. Загрузим модуль ip_nat_ftp, который и будет выполнять всю черную работу вместо нас.

```
# modprobe ip_nat_ftp
# lsmod | grep ftp
ip_nat_ftp              4336  0
ip_conntrack_ftp       71728  1 ip_nat_ftp
iptable_nat            19772  2 ip_nat_ftp
ip_conntrack           34740  4
ip_nat_ftp,ip_conntrack_ftp,iptable_nat,ipt_state
```

Обратите внимание, что в цепочке eth0-eth1, через которую проходят все транзитные пакеты, мы открыли только порт 21 и разрешили прохождение пакетов с состоянием RELATED и ESTABLISHED. Для работы ftp как в активном так и в пассивном режимах больше ничего не требуется, все остальные заботы берут на себя модули ip_nat_ftp и ip_conntrack_ftp.

Тестирование

После того, как мы загрузили модуль и правил можно приступить непосредственно к тестированию. Для этого можно воспользоваться любимым ftp клиентом и попробовать соединиться с любым ftp сервером. В качестве клиента я использовал ftp клиент встроенный в far, который позволяет использовать как активный, так и пассивный режимы. Если под рукой нет никакого ftp клиента всегда можно воспользоваться встроенным в windows клиентом как показано ниже.

```
C:\Documents and Settings\Admin>ftp ftp.sys-adm.org.ua
Связь с ftp.sys-adm.org.ua.
220 Welcome To SYS-ADM.ORG.UA FTP Server.
Пользователь (ftp.sys-adm.org.ua:(none)): alex
331 Please specify the password.
```

```
Пароль: *******
230 Login successful.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
games
music
video
xxx
226 Directory send OK.
ftp: 26 байт получено за 0,00 (сек) со скоростью 26000,00 (КБ/сек).
ftp> quit
221 Goodbye.
```

Для наглядности можно посмотреть с помощью tcpdump что происходит при попытке подключиться к удаленному ftp серверу.

```
# tcpdump -npi eth1 port 21 and host ftp.sys-adm.org.ua
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 96 bytes
23:17:52.626734 IP 212.42.65.100.1327 > 212.42.65.15.ftp: S
3449417565:3449417565(0) win 65535
23:17:52.629494 IP 212.42.65.15.ftp > 212.42.65.100.1327: S
3892471215:3892471215(0) ack 3449417566 win 5840
23:17:52.630370 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 1 win 65535
23:17:52.632693 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 1:42(41) ack 1
win 5840
23:17:52.709940 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 42 win 65494
23:17:55.258372 IP 212.42.65.100.1327 > 212.42.65.15.ftp: P 1:12(11) ack 42
win 65494
23:17:55.260886 IP 212.42.65.15.ftp > 212.42.65.100.1327: . ack 12 win 5840
23:17:55.263883 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 42:76(34) ack 12
win 5840
23:17:55.347886 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 76 win 65460
23:17:56.906175 IP 212.42.65.100.1327 > 212.42.65.15.ftp: P 12:25(13) ack 76
win 65460
23:17:56.909129 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 76:99(23) ack 25
win 5840
23:17:57.000080 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 99 win 65437
23:18:01.089040 IP 212.42.65.100.1327 > 212.42.65.15.ftp: P 25:50(25) ack 99
win 65437
23:18:01.091282 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 99:150(51) ack
50 win 5840
23:18:01.094867 IP 212.42.65.100.1327 > 212.42.65.15.ftp: P 50:56(6) ack 150
win 65386
23:18:01.097930 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 150:189(39) ack
56 win 5840
23:18:01.098865 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 189:213(24) ack
56 win 5840
23:18:01.100303 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 213 win
65323
```

```
23:18:03.367299 IP 212.42.65.100.1327 > 212.42.65.15.ftp: P 56:62(6) ack 213
win 65323
23:18:03.370123 IP 212.42.65.15.ftp > 212.42.65.100.1327: P 213:227(14) ack
62 win 5840
23:18:03.372775 IP 212.42.65.15.ftp > 212.42.65.100.1327: F 227:227(0) ack
62 win 5840
23:18:03.372968 IP 212.42.65.100.1327 > 212.42.65.15.ftp: . ack 228 win
65309
23:18:03.375854 IP 212.42.65.100.1327 > 212.42.65.15.ftp: F 62:62(0) ack 228
win 65309
23:18:03.377860 IP 212.42.65.15.ftp > 212.42.65.100.1327: . ack 63 win 5840

24 packets captured
54 packets received by filter
0 packets dropped by kernel
```

Ну вот собственно и вся настройка. Для того, чтобы каждый раз при загрузке системы модуль ip_nat_ftp загружался автоматически необходимо внести небольшие изменения в файл /etc/sysconfig/iptables-config.

```
#
# /etc/sysconfig/iptables-config
#

# Загрузка дополнительных модулей iptables (nat помощников)
#  По умолчанию: -пусто-
# Разделенный пробелами список "nat помощников" (например 'ip_nat_ftp
ip_nat_irc'),
# которые загружаются после применения правил.
# Опции для "помощников" находятся в /etc/modprobe.conf.
IPTABLES_MODULES="ip_nat_ftp ip_conntrack_ftp"
```

После этого при перезагрузке iptables или системы в целом данный модуль будет загружаться автоматически, а также подгружать все необходимые модули.

```
# service iptables restart
Flushing firewall rules:                                   [  OK  ]
Setting chains to policy ACCEPT:  mangle filter nat        [  OK  ]
Unloading iptables modules:                                [  OK  ]
Applying iptables firewall rules:                          [  OK  ]
Loading additional iptables modules: ip_nat_ftp            [  OK  ]
```