

Дополнительные инструменты

- [Проверка маршрутизации портов](#)
 - [nc](#)
 - [tcpdump](#)
- [Утилита jstack](#)
- [Тестирование пропускной способности канала при помощи iperf](#)

В этом разделе представлены вспомогательные инструменты, которые могут быть использованы для администрирования WCS-сервера.

Проверка маршрутизации портов

WCS-сервер может располагаться за NAT и требовать диапазона портов, открытого для внешней сети, например UDP 31000-32000. Это означает, что UDP-пакет, отправленный из внешней сети на порт из этого диапазона должен дойти до сервера, на котором расположен WCS.

Таким образом имеем простой тест. Отправляем UDP пакет извне с помощью nc и принимаем его на сервере с помощью tcpdump. Если пакет дошел, значит порт открыт.

nc

```
echo -n "hello" | nc -4u -w1 wcs1.com 31000
```

или для Debian:

```
echo -n "hello" | nc -u -w1 wcs1.com 31000
```

Эта команда отправит простой UDP пакет в нужном направлении.

tcpdump

```
tcpdump udp port 31000
```

Эта команда заставит слушать нужный порт и немедленно выведет в консоль информацию о прибытии пакета:

```
17:50:21.932509 IP myhost.39194 > host.31000: UDP, length 5
```

Утилита jstack

Это Java-утилита, которая дает важную информацию о Java-процессе и потоках выполнения. При вызове программы jstack из консоли должна отобразиться краткая информация о jstack:

```
[root@localhost bin]# jstack
Usage:
  jstack [-l] <pid>
    (to connect to running process)
  jstack -F [-m] [-l] <pid>
    (to connect to a hung process)
  jstack [-m] [-l] <executable> <core>
    (to connect to a core file)
  jstack [-m] [-l] [server_id@]<remote server IP or hostname>
    (to connect to a remote debug server)

Options:
  -F to force a thread dump. Use when jstack <pid> does not respond (process is hung)
  -m to print both java and native frames (mixed mode)
  -l long listing. Prints additional information about locks
  -h or -help to print this help message
```

Если информация не отображается или утилита jstack не найдена, воспользуйтесь [инструкцией к установке последней версии JDK](#). После установки jdk нужно будет создать символическую ссылку для jstack чтобы была возможность его быстро вызвать:

```
ln -sf /usr/java/default/bin/jstack /usr/bin/jstack
```

Пример:

```
jstack 8888 > jstack.report
```

В этом примере 8888 - ID Java-процесса.

Тестирование пропускной способности канала при помощи iperf

Качество картинки публикуемого потока напрямую зависит от пропускной способности канала между публикующей стороной и сервером, то же касается и подписчиков. Проверить пропускную способность канала можно при помощи утилиты [iperf](#). Эта программа выпущена под все основные операционные системы: Windows, MacOS, Ubuntu/Debian, CentOS. iperf в режиме сервера может быть установлена вместе с WCS, что позволяет тестировать канал целиком, от публикатора до зрителя.

Установка iperf на CentOS 7 производится следующим образом:

```
yum install iperf3
```

Запуск iperf в режиме сервера

```
iperf3 -s -p 5201
```

здесь 5201 - порт, на который iperf ожидает соединений от тестирующих клиентов

На стороне клиента iperf запускается следующим образом:

1. Для проверки канала на отправку данных серверу по UDP (пример для Windows)

```
iperf3.exe -c test2.flashphoner.com -p 5201 -u
```

Здесь

- test2.flashphoner.com - WCS сервер
- 5201 - порт iperf в режиме сервера

Результат выполнения команды будет следующим:

```
Connecting to host test2.flashphoner.com, port 5201
[ 4] local 192.168.0.120 port 51365 connected to 95.191.131.65 port 5201
[ ID] Interval           Transfer     Bandwidth   Total Datagrams
[ 4]  0.00-1.01   sec      312 KBytes    2.54 Mbits/sec    39
[ 4]  1.01-2.00   sec     96.0 KBytes    790 Kbits/sec    12
[ 4]  2.00-3.01   sec     80.0 KBytes    654 Kbits/sec    10
[ 4]  3.01-4.01   sec     88.0 KBytes    718 Kbits/sec    11
[ 4]  4.01-5.01   sec     88.0 KBytes    722 Kbits/sec    11
[ 4]  5.01-6.00   sec     112 KBytes    922 Kbits/sec    14
[ 4]  6.00-7.00   sec     128 KBytes    1.05 Mbits/sec    16
[ 4]  7.00-8.01   sec     120 KBytes    976 Kbits/sec    15
[ 4]  8.01-9.01   sec     128 KBytes    1.05 Mbits/sec    16
[ 4]  9.01-10.00  sec     128 KBytes    1.05 Mbits/sec    16
-- -- -- -- --
[ ID] Interval           Transfer     Bandwidth   Jitter    Lost/Total Datagrams
[ 4]  0.00-10.00  sec     1.25 MBytes    1.05 Mbits/sec    0.000 ms    0/0 (0%)
[ 4] Sent 0 datagrams

iperf Done.
```

2. Для проверки канала на прием данных от сервера по UDP

```
iperf3.exe -c test2.flashphoner.com -p 5201 -u -R
```

Здесь

- test2.flashphoner.com - WCS сервер
- 5201 - порт iperf в режиме сервера

Результат выполнения команды будет следующим:

```
Connecting to host test2.flashphoner.com, port 5201
Reverse mode, remote host test2.flashphoner.com is sending
[ 4] local 192.168.0.120 port 63892 connected to 95.191.131.65 port 5201
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[ 4]  0.00-1.00   sec      144 KBytes    1.18 Mbits/sec  58.999 ms    0/18 (0%)
[ 4]  1.00-2.00   sec      128 KBytes    1.05 Mbits/sec  24.781 ms    0/16 (0%)
[ 4]  2.00-3.00   sec      128 KBytes    1.05 Mbits/sec  17.640 ms    0/16 (0%)
[ 4]  3.00-4.01   sec      128 KBytes    1.04 Mbits/sec  29.843 ms    0/16 (0%)
[ 4]  4.01-5.00   sec      128 KBytes    1.05 Mbits/sec  45.138 ms    0/16 (0%)
[ 4]  5.00-6.00   sec      128 KBytes    1.05 Mbits/sec  29.477 ms    0/16 (0%)
[ 4]  6.00-7.01   sec      128 KBytes    1.04 Mbits/sec  31.473 ms    0/16 (0%)
[ 4]  7.01-8.00   sec      128 KBytes    1.05 Mbits/sec  40.623 ms    0/16 (0%)
[ 4]  8.00-9.00   sec      120 KBytes    984 Kbits/sec   50.511 ms    0/15 (0%)
[ 4]  9.00-10.00  sec      136 KBytes    1.12 Mbits/sec  25.318 ms    0/17 (0%)
-----
[ ID] Interval           Transfer     Bandwidth       Jitter    Lost/Total Datagrams
[ 4]  0.00-10.00  sec      1.27 MBytes    1.06 Mbits/sec  25.318 ms    0/162 (0%)
[ 4] Sent 162 datagrams

iperf Done.
```

По умолчанию, iperf тестирует канал в течение 10 секунд. Это время рекомендуется увеличить, например, до 120 секунд

```
iperf3.exe -c test2.flashphoner.com -p 5201 -u -t 120
```

Результат тестирования канала на отправку данных серверу по UDP показывает максимальное значение битрейта публикации видео, при котором не будет потерь пакетов. В вышеприведенном примере битрейт публикации следует ограничить 1000 кбит/с, например, на стороне сервера

```
webrtc_cc_max_bitrate=1000000
```

Необходимо отметить, что версии iperf на сервере и на тестирующем клиенте должны совпадать. В настоящее время актуальной является версия 3, хотя в репозиториях есть и версия 2.