

## Тестирование производительности сети с помощью iperf

Документ соответствует iperf версии 2.0.14a (20 Jan 2020) pthreads.

### Синтаксис

```
iperf -s [options]
iperf -c server [options]
iperf -u -s [options]
iperf -u -c server [options]
```

### Описание

Пакет iperf служит для измерения производительности работы сети. Программа позволяет измерять пропускную способность на основе протоколов TCP и UDP. Для проведения теста нужен сервер, принимающий и отбрасывающий пакеты, и клиент, который генерирует тестовые пакеты. Клиент и сервер могут размещаться на одном (локальные тесты) или разных (тесты сети) хостах локальной или распределенной сети.

Для создания пакетов клиент использует многопоточковый (multithread) режим, что позволяет эффективно загрузить загрузить имеющиеся в системе процессорные ядра.

### Опции

#### **-b, --bandwidth**

Задаёт полосу текстового потока и может также задавать стандартное отклонение от нормального распределения в форме `<mean>.[<stdev>]`, которое обычно указывается в выводе. Значения могут задаваться с символьными суффиксами<sup>1</sup>.

#### **-e, --enhanced**

Задаёт расширенный формат вывода. В тестах UDP при расширенном выводе предполагается синхронизация часов клиента и сервера по протоколу NTP или PTP. На точность измерения задержки UDP оказывает влияние точность эталонных (опорных) часов.

#### **-f, --format [abkmgBKMG]**

Задаёт формат вывода и может включать значения a (адаптивный), b (биты), B (байты), k (килобиты), m (мегабиты), g (гигабиты), K (килобайты), M (мегабайты), G (гигабайты).

#### **-h, --help**

Выводит справочную информацию о программе.

#### **-i, --interval <n[p] | f >**

Задаёт интервал выборки или отображения n секунд (принято по умолчанию) или n пакетов (суффикс p). При использовании f интервал задаёт группа (burst) или кадр.

#### **-l, --len n[kmKM]**

Задаёт размер буфера чтения-записи (TCP) или размер пакетов (UDP) и может использовать суффиксы k, m, K, M<sup>1</sup>. По умолчанию для TCP принято n = 128K, для UDP - n = 1470.

#### **--l2checks**

Задаёт проверку размера кадров L2 для принятых пакетов UDP (требуется поддержка сокета пакетов).

#### **-m, --print\_mss**

Задаёт вывод максимального размера сегментов TCP (MSS<sup>2</sup>, MTU - заголовок TCP/IP).

#### **--NUM\_REPORT\_STRUCTS <count>**

Переопределяет принятый по умолчанию размер общей памяти для потоков (thread) трафика и блока отчетов для снижения числа конфликтов блокировки семафора (mutex). Принятого по умолчанию значения 5000 должно быть достаточно для сетей 1 Гбит/с. Значение следует увеличить при наличии предупреждений о слишком медленных потоках. При отсутствии таких предупреждений увеличение параметра приведет лишь к дополнительному расходу памяти.

#### **-o, --output filename**

Задаёт запись вывода и сообщений об ошибках в указанный файл.

#### **-p, --port n**

Задаёт порт, используемый сервером (по умолчанию 5001).

#### **-u, --udp**

Задаёт использование протокола UDP вместо принятого по умолчанию TCP.

#### **-w, --window n[kmKM]**

Задаёт размер окна TCP (размер буфера сокета).

#### **-z, --realtime**

Запрашивает использование планировщика в реальном масштабе времени (если он поддерживается).

#### **-B, --bind host[:port][%dev]**

Задаёт привязку к IP-адресу хоста или групповому адресу, а также может задавать привязку к порту.

#### **-C, --compatibility**

Служит для совместимости со старыми версиями, не передающими дополнительных сообщений.

#### **-M, --mss n**

Задаёт максимальный размер сегмента TCP (MTU - 40 байтов).

#### **-N, --nodelay**

Отключает задержку TCP (алгоритм Nagle).

<sup>1</sup>Некоторые числовые опции поддерживают указание единиц в форме `<value>s` (например, 10M), где s задаёт единицу измерения и может принимать значения k, m, g, K, M, G. Символы нижнего регистра указывают единицы на основе десятичных значений (10<sup>3</sup>, 10<sup>6</sup>, 10<sup>9</sup>), а символы верхнего регистра - на основе двоичных (2<sup>n</sup> - 1K = 1024, 1M = 1048576, 1G = 1073741824).

<sup>2</sup>Maximum segment size.

**-v, --version**

Выводит информацию о версии программы и завершает работу.

**-x, --reportexclude [CDMSV]**

Исключает отчеты о соединениях (C), данных (D), групповых пакетах (M), настройках (S) и сервере (V).

**-y, --reportstyle C|c**

Установка значения C или c задает вывод в формате CSV<sup>1</sup>.

**-Z, --tcp-congestion**

Задаёт используемый по умолчанию алгоритм контроля насыщения для новых соединений. Платформа должна поддерживать setsockopt TCP\_CONGESTION. (см. sysctl и tcp\_allowed\_congestion\_control).

## Опции сервера

**-b, --bandwidth n[kmgKMG]**

Задаёт целевую скорость чтения n и может использовать описанные выше суффиксы (только для сервера TCP).

**-s, --server**

Задаёт работу в режиме сервера.

**--histogram[=binwidth[u],bincount,[lowerci],[upperci]]**

Задаёт вывод гистограмм задержки для пакетов (опция -u) или групп (burst) и кадров (опция --trip-times или --isochronous). binwidth - продолжительность элемента (по умолчанию 1 мсек, для мсек суффикс u), bincount - общее число элементов (по умолчанию 1000), ci - доверительный интервал между 0-100% (по умолчанию от 5% до 95%).

**-B, --bind ip | ip%device**

Задаёт привязку к IP-адресу получателя, а также может задавать привязку к порту и входному интерфейсу. Приниматься будут лишь пакеты, соответствующие заданным опцией параметрам. Опция полезна также при групповой адресации. Например, iperf -s -B 224.0.0.1%eth0 будет задавать прием групповых пакетов на входном интерфейсе eth0.

**-D, --daemon**

Задаёт работу сервера в режиме демона. В Windows это ведёт к запуску заданной команды как IperfService с установкой службы при необходимости. Служба не настраивается на автоматический запуск или перезапуск и при необходимости это можно организовать с помощью сценарий инициализации или команды Windows sc.

**-H, --ssm-host host**

Задаёт хост отправителя (адрес IP) для групповых пакетов SSM (т. е. S в S,G)

**-R, --remove**

Удаляет службу IPerfService (только Windows).

**-U, --single\_udp**

Задаёт работу в режиме UDP с одним потоком (thread).

**-V, --ipv6\_domain**

Включает прием пакетов IPv6 путем установки домена и сокета AF\_INET6 (можно принимать сразу IPv4 и IPv6).

## Опции клиента

**-b, --bandwidth n[kmgKMG] | npps**

Задаёт целевую полосу в бит/с (по умолчанию 1 Мбит/с) или пакет/с для трафика TCP или UDP. Значение параметра может указываться с суффиксом, задающим единицу измерения. Кроме того, поддерживается возможность задать среднее и стандартное отклонение от нормального распределения (mean,standard)

**-c, --client host | host%device**

Задаёт работу в режиме клиента с сервером host. Необязательный параметр %device указывает выходной интерфейс (SO\_BINDTODEVICE).

**--connect-only**

Задаёт лишь организацию соединений TCP без передачи реального трафика, что может быть полезно для измерения времени TCP connect().

**-d, --dualtest**

Задаёт выполнение теста одновременно в обоих направлениях.

**--fq-rate n[kmgKMG]**

Задаёт скорость, используемую при беспристрастных очередях на уровне сокетов, в битах или байтах в секунду. Параметр может содержать суффикс для задания единиц измерения (строчные буквы указывают единицы в битах, прописные - в байтах). Опция доступна лишь на платформах, поддерживающих опцию сокетов SO\_MAX\_PACING\_RATE.

**--incr-dstip**

Задаёт инкрементирование IP-адреса получателя при использовании опции -P.

**--ipg n**

Задаёт межпакетный интервал (в миллисекундах) в изохронном кадре (burst). Требуется опция --isochronous

**--isochronous[=fps:mean,stdev]**

Задаёт передачу изохронного трафика с заданным числом кадров в секунду и нагрузкой, указанной средним и стандартным отклонением (mean, stdev) от нормального распределения (по умолчанию 60:20m,0). Скорость может указываться с суффиксом для задания единиц измерения (строчные буквы указывают единицы в битах, прописные - в байтах).

**--no-connect-sync**

По умолчанию параллельные потоки трафика (-P больше 1) будут синхронизироваться до организации соединений TCP и реальной передачи трафика, т. е. потоки (thread) сначала завершают согласование TCP 3WHS (возможно с ошибкой) и лишь после этого начинается передача трафика. Эта опция отключает такую синхронизацию и каждый поток начинает передаваться сразу после организации соединения.

**--no-udp-fin**

Отключает выполнение завершающего обмена UDP от сервера к клиенту, в результате чего у клиента не будут выводиться сообщения от сервера. Все пакеты в тесте будут передаваться только от клиента к серверу без передачи пакетов в обратном направлении. Эта опция устанавливается клиентом и передается серверу (начиная с версии 2.0.14).

<sup>1</sup>Comma separated values - разделенные запятыми значения.

**-n, --num n[кмКМ]**

Число байтов для передачи (вместо -t)

**-r, --tradeoff**

Выполнять двухсторонние тесты по направлениям, сначала от клиента к серверу, затем обратно.

**-t, --time n**

Время прослушивания новых соединений (в секундах), приема или передачи трафика. По умолчанию передача длится 10 секунд, а прием и прослушивание не ограничены во времени.

**--trip-times**

Включает измерение задержки записи (или передачи данных) в тесте TCP. Требуется синхронизация часов.

**--txdelay-time**

Время (в секундах) удержания или задержки между организацией соединения TCP и записью в сокет, а для UDP - задержки между стартом потока трафика и первой записью.

**--txstart-time n.n**

Устанавливает начало передачи (n.n) по времени unix или epoch (с поддержкой наносекундного разрешения, например, 1536014418.839992457).

**-B, --bind ip | ip:port | ipv6 -v | [ipv6]:port -V**

Задаёт IP-адрес отправителя, а также позволяет задать порт отправителя и выходное устройство (%device) для передачи пакетов. Опция влияет на системные вызовы bind() и обычно служит для привязки к определённому адресу IP и порту отправителя (например, iperf -c <host> -B 192.168.100.2:6002). Это задаёт источник пакетов но не применяется при маршрутизации. Здесь может возникнуть путаница при просмотре маршрутов и устройств. Например, если IP-адрес интерфейса eth0 указан в опции -B, а таблица маршрутизации для IP-адреса получателя (опция -c) указывает выходной интерфейс eth1, хост будет передавать через интерфейс eth1 пакеты с IP-адресом интерфейса eth0. Для задания выходного интерфейса в системе с несколькими подключениями следует применять форму -c <host>%device (требуются полномочия root) для обхода поиска в таблице маршрутизации хоста или настроить таблицу маршрутизации хоста для каждой опции -B соответствующим образом и задать выходные интерфейсы в правилах.

Указание выходного интерфейса требуется при использовании адресов IPv6 link-local.

**-F, --fileinput name**

Задаёт считывание передаваемых данных из файла.

**-l, --stdin**

Задаёт считывание передаваемых данных со стандартного устройства ввода (stdin).

**-L, --listenport n**

Задаёт порт для приема возвращаемых пакетов.

**-P, --parallel n**

Задаёт число параллельных потоков (thread), запускаемых клиентом.

**-R, --reverse**

Задаёт обращение (реверс) потока трафика после обмена заголовками и может быть полезно при тестировании через межсетевые экраны<sup>1</sup>.

**-S, --tos**

Устанавливает значение поля IP\_TOS для сокета (1 байт).

**-T, --ttl n**

Задаёт TTL для группового трафика (по умолчанию 1)

**-V, --ipv6\_domain**

Задаёт домен для IPv6 (передача пакетов по IPv6).

**-X, --peerdetect**

Задаёт определение версии сервера до начала обмена трафиком.

**-Z, --linux-congestion algo**

Задаёт алгоритм контроля насыщения TCP (только для Linux).

## Примеры

### Тест TCP (клиент)

```
iperf -c <host> -e -i 1
```

```
-----
Client connecting to <host>, TCP port 5001 with pid 5149
Write buffer size: 128 KByte
TCP window size: 340 KByte (default)
-----
```

[ ID]	Interval	Transfer	Bandwidth	Write/Err	Rtry	Cwnd/RTT	NetPwr
[ 3]	0.00-1.00 sec	126 MBytes	1.05 Gbits/sec	1006/0	0	56K/626 us	210636.47
[ 3]	1.00-2.00 sec	138 MBytes	1.15 Gbits/sec	1100/0	299	483K/3884 us	37121.32
[ 3]	2.00-3.00 sec	137 MBytes	1.15 Gbits/sec	1093/0	24	657K/5087 us	28162.31
[ 3]	3.00-4.00 sec	126 MBytes	1.06 Gbits/sec	1010/0	284	294K/2528 us	52366.58
[ 3]	4.00-5.00 sec	117 MBytes	980 Mbits/sec	935/0	373	487K/2025 us	60519.66
[ 3]	5.00-6.00 sec	144 MBytes	1.20 Gbits/sec	1149/0	2	644K/3570 us	42185.36
[ 3]	6.00-7.00 sec	126 MBytes	1.06 Gbits/sec	1011/0	112	582K/5281 us	25092.56
[ 3]	7.00-8.00 sec	110 MBytes	922 Mbits/sec	879/0	56	279K/1957 us	58871.89
[ 3]	8.00-9.00 sec	127 MBytes	1.06 Gbits/sec	1014/0	46	483K/3372 us	39414.89
[ 3]	9.00-10.00 sec	132 MBytes	1.11 Gbits/sec	1054/0	0	654K/3380 us	40872.75
[ 3]	0.00-10.00 sec	1.25 GBytes	1.07 Gbits/sec	10251/0	1196	-1K/3170 us	42382.03

где (с учетом -e)

<sup>1</sup>Опции --reverse (-R), -r и -d вызывают путаницу. Если нужно выполнить тест через шлюз NAT, следует применять опцию --reverse (или -R в системах, отличных от Windows). Опции -d и -r сохранены для совместимости. Вновь открытые и исходные сокеты работают в полнодуплексном режиме. Работа через межсетевой экран обычно требует использовать -d, опция -r нужна при работе через шлюз NAT. Кроме того, установка --reverse -b <rate> даёт несколько отличающийся эффект. Для TCP это будет ограничивать скорость на читающей стороне, т. е. скорость чтения клиентом iperf из полнодуплексного сокета. Это будет приводить к использованию стандартного контроля насыщения TCP для реверсированного трафика. Опции --reverse -b <rate> должны применяться на передающей стороне (т. е., на обратном сервере) для трафика UDP, поскольку здесь нет управления потоком трафика.

**ct=**

Время соединения TCP (время трехэтапного согласования 3WHS).

**Write/Err**

Общее число успешных записей в сокет и общее число не критических ошибок записи в сокет.

**Rtry**

Общее число попыток TCP.

**Cwnd/RTT (только \*nix)**

Окно насыщения TCP и время кругового обхода (выборка)

**NetPwr (только \*nix)**

Отношение пропускной способности к RTT.

## Тест TCP (сервер)

```
iperf -s -e -i 1 -l 8K
```

```
-----
Server listening on TCP port 5001 with pid 13430
Read buffer size: 8.00 KByte
TCP window size: 85.3 KByte (default)
-----
```

```
[ 4] local 45.33.58.123 port 5001 connected with 45.56.85.133 port 49960
[ ID] Interval          Transfer          Bandwidth        Reads          Dist (bin=1.0K)
[ 4] 0.00-1.00 sec      124 MBytes       1.04 Gbits/sec   22249          798:2637:2061:767:2165:1563:589:11669
[ 4] 1.00-2.00 sec      136 MBytes       1.14 Gbits/sec   24780          946:3227:2227:790:2427:1888:641:12634
[ 4] 2.00-3.00 sec      137 MBytes       1.15 Gbits/sec   24484          1047:2686:2218:810:2195:1819:728:12981
[ 4] 3.00-4.00 sec      126 MBytes       1.06 Gbits/sec   20812          863:1353:1546:614:1712:1298:547:12879
[ 4] 4.00-5.00 sec      117 MBytes       984 Mbites/sec   20266          769:1886:1828:589:1866:1350:476:11502
[ 4] 5.00-6.00 sec      143 MBytes       1.20 Gbits/sec   24603          1066:1925:2139:822:2237:1827:744:13843
[ 4] 6.00-7.00 sec      126 MBytes       1.06 Gbits/sec   22635          834:2464:2249:724:2269:1646:608:11841
[ 4] 7.00-8.00 sec      110 MBytes       921 Mbites/sec   21107          842:2437:2747:592:2871:1903:496:9219
[ 4] 8.00-9.00 sec      126 MBytes       1.06 Gbits/sec   22804          1038:1784:2639:656:2738:1927:573:11449
[ 4] 9.00-10.00 sec     133 MBytes       1.11 Gbits/sec   23091          1088:1654:2105:710:2333:1928:723:12550
[ 4] 0.00-10.02 sec     1.25 Gbytes      1.07 Gbits/sec   227306         9316:22088:21792:7096:22893:17193:6138:120790
```

где (с учетом -e)

**Reads**

Общее число считываний сокета.

**Dist(bin=size)**

8 элементов (bin) гистограммы чтения, возвращенных клиентом и разделяемых двоеточиями. В примере это элементы 0-1K, 1K-2K, ..., 7K-8K.

## Тест TCP (сервер) с опцией --trip-times на стороне клиента

```
iperf -s -e -i 1
```

```
-----
Server listening on TCP port 5001 with pid 30369
Read buffer size: 128 KByte
TCP window size: 85.3 KByte (default)
-----
```

```
[ 4] local 10.19.87.7 port 5001 connected with 10.19.87.10 port 43338 (trip-times)
[ ID] Interval          Transfer          Bandwidth        Reads          Dist (bin=16.0K)          Burst Latency avg/min/max/stdev (cnt/size) inP
NetPwr
[ 4] 0.00-1.00 sec      112 MBytes       941 Mbites/sec   7000           1552:5447:1:0:0:0:0:0      8.749/ 1.583/10.340/ 1.011 ms (897/131127) 1029057 bytes
13444.08
[ 4] 1.00-2.00 sec      112 MBytes       941 Mbites/sec   7015           1562:5453:0:0:0:0:0:0      8.790/ 7.131/10.443/ 0.878 ms (898/131050) 1034467 bytes
13387.92
[ 4] 2.00-3.00 sec      112 MBytes       941 Mbites/sec   7009           1543:5466:0:0:0:0:0:0      8.799/ 7.050/10.389/ 0.869 ms (897/131170) 1035306 bytes
13371.80
[ 4] 3.00-4.00 sec      112 MBytes       941 Mbites/sec   7032           1589:5442:1:0:0:0:0:0      8.810/ 7.128/10.437/ 0.877 ms (898/131047) 1036818 bytes
13356.91
[ 4] 4.00-5.00 sec      112 MBytes       941 Mbites/sec   7013           1556:5457:0:0:0:0:0:0      8.805/ 7.244/10.352/ 0.874 ms (898/131050) 1036239 bytes
13365.03
[ 4] 5.00-6.00 sec      112 MBytes       941 Mbites/sec   6999           1554:5440:3:1:0:0:0:1      10.384/ 7.257/12.712/ 1.284 ms (898/131050) 1222077 bytes
11332.64
[ 4] 6.00-7.00 sec      112 MBytes       941 Mbites/sec   7015           1568:5447:0:0:0:0:0:0      10.682/ 8.714/12.711/ 1.121 ms (898/131045) 1257085 bytes
11016.23
[ 4] 7.00-8.00 sec      112 MBytes       941 Mbites/sec   7010           1557:5453:0:0:0:0:0:0      10.683/ 8.681/12.695/ 1.125 ms (898/131050) 1257237 bytes
11015.71
[ 4] 8.00-9.00 sec      112 MBytes       941 Mbites/sec   7016           1570:5446:0:0:0:0:0:0      10.674/ 8.704/12.679/ 1.128 ms (897/131193) 1256177 bytes
11024.46
[ 4] 9.00-10.00 sec     112 MBytes       941 Mbites/sec   7062           1624:5438:0:0:0:0:0:0      10.693/ 8.624/12.681/ 1.127 ms (898/131047) 1258342 bytes
11005.49
[ 4] 10.00-10.01 sec    1.28 MBytes      939 Mbites/sec   80             17:63:0:0:0:0:0:0          11.582/ 8.761/12.361/ 1.191 ms (11/121860) 1359148 bytes
10131.78
[ 4] 0.00-10.01 sec     1.10 GBytes      941 Mbites/sec   70251          15692:54552:5:1:0:0:0:1    9.699/11.582/11.582/ 0.000 ms (8988/131072) 1141261 bytes
12133.03
```

где (с учетом -e)

**Burst Latency**

Односторонняя задержка TCP от write() до read() в формате среднее/минимальное/максимальное/стандартное отклонение. Требуется синхронизация часов клиента и сервера от одного источника (например, по протоколу PTP). Рекомендуется применять опорный источник GPS ОСХО.

**cnt**

Число принятых завершенных групп (burst), использованных для расчета задержки.

**size**

Средний размер группы (burst) в байтах (только для оценки).

**inP**Сокращение для in progress (в работе). Указывает среднее число байтов, находящихся в обработке или «на лету» (в сети) с точки зрения записывающего приложения<sup>1</sup>.**NetPwr**

Отношение пропускной способности к задержке в одном направлении.

## Тест UDP (клиент)

```
iperf -c <host> -e -i 1 -u -b 10m
```

```
-----
Client connecting to <host>, UDP port 5001 with pid 5169
Sending 1470 byte datagrams, IPG target: 1176.00 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
```

<sup>1</sup>Закон Литтла (Little) в теории очередей определяет среднее число элементов (L) в стационарной системе очередей на основе средне-взвешенного времени (W) нахождения элемента в системе и среднего числа элементов, прибывающих в систему за единицу времени (lambda). Математически это выражается в форме  $L = \lambda * W$ . Здесь элементами TCP являются байты, а UDP - пакеты.

```

[ 3] local 45.56.85.133 port 32943 connected with 45.33.58.123 port 5001
[ ID] Interval      Transfer      Bandwidth    Write/Err    PPS
[ 3] 0.00-1.00 sec  1.19 MBytes  10.0 Mbits/sec  852/0        851 pps
[ 3] 1.00-2.00 sec  1.19 MBytes  10.0 Mbits/sec  850/0        850 pps
[ 3] 2.00-3.00 sec  1.19 MBytes  10.0 Mbits/sec  850/0        850 pps
[ 3] 3.00-4.00 sec  1.19 MBytes  10.0 Mbits/sec  851/0        850 pps
[ 3] 4.00-5.00 sec  1.19 MBytes  10.0 Mbits/sec  850/0        850 pps
[ 3] 5.00-6.00 sec  1.19 MBytes  10.0 Mbits/sec  850/0        850 pps
[ 3] 6.00-7.00 sec  1.19 MBytes  10.0 Mbits/sec  851/0        850 pps
[ 3] 7.00-8.00 sec  1.19 MBytes  10.0 Mbits/sec  850/0        850 pps
[ 3] 8.00-9.00 sec  1.19 MBytes  10.0 Mbits/sec  851/0        850 pps
[ 3] 0.00-10.00 sec 11.9 MBytes  10.0 Mbits/sec  8504/0       850 pps
[ 3] Sent 8504 datagrams
[ 3] Server Report:
[ 3] 0.00-10.00 sec 11.9 MBytes 10.0 Mbits/sec 0.047 ms 0/ 8504 (0%) 0.537/ 0.392/23.657/ 0.497 ms 850 pps 2329.37

```

где (с учетом -e)

#### Write/Err

Общее число успешных записей в сокет и некритичных ошибок при записи в сокет.

#### PPS

Число переданных в секунду пакетов.

## Тест UDP (сервер)

```
iperf -s -e -i 1 -u
```

```

-----
Server listening on UDP port 5001 with pid 13496
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)

```

```

[ 3] local 45.33.58.123 port 5001 connected with 45.56.85.133 port 32943
[ ID] Interval      Transfer      Bandwidth    Jitter      Lost/Total   Latency avg/min/max/stddev  PPS    NetPwr
[ 3] 0.00-1.00 sec  1.19 MBytes  10.0 Mbits/sec  0.057 ms  0/ 851 (0%)  0.475/ 0.408/ 1.898/ 0.090 ms  851 pps 2633.56
[ 3] 1.00-2.00 sec  1.19 MBytes  10.0 Mbits/sec  0.039 ms  0/ 851 (0%)  0.669/ 0.405/16.256/ 1.375 ms  850 pps 1869.32
[ 3] 2.00-3.00 sec  1.19 MBytes  10.0 Mbits/sec  0.038 ms  0/ 850 (0%)  0.795/ 0.395/23.657/ 2.138 ms  850 pps 1572.05
[ 3] 3.00-4.00 sec  1.19 MBytes  10.0 Mbits/sec  0.045 ms  0/ 850 (0%)  0.475/ 0.403/ 3.477/ 0.148 ms  850 pps 2628.58
[ 3] 4.00-5.00 sec  1.19 MBytes  10.0 Mbits/sec  0.043 ms  0/ 851 (0%)  0.463/ 0.400/ 1.458/ 0.068 ms  850 pps 2639.88
[ 3] 5.00-6.00 sec  1.19 MBytes  10.0 Mbits/sec  0.032 ms  0/ 850 (0%)  0.486/ 0.404/ 2.658/ 0.154 ms  850 pps 2572.21
[ 3] 6.00-7.00 sec  1.19 MBytes  10.0 Mbits/sec  0.055 ms  0/ 850 (0%)  0.469/ 0.404/ 2.768/ 0.108 ms  850 pps 2664.82
[ 3] 7.00-8.00 sec  1.19 MBytes  10.0 Mbits/sec  0.039 ms  0/ 851 (0%)  0.571/ 0.400/12.452/ 0.855 ms  850 pps 2192.68
[ 3] 8.00-9.00 sec  1.19 MBytes  10.0 Mbits/sec  0.083 ms  0/ 850 (0%)  0.475/ 0.392/ 3.702/ 0.196 ms  850 pps 2628.29
[ 3] 9.00-10.00 sec 1.19 MBytes  10.0 Mbits/sec  0.047 ms  0/ 850 (0%)  0.493/ 0.396/ 6.010/ 0.343 ms  850 pps 2534.89
[ 3] 0.00-10.00 sec 11.9 MBytes  10.0 Mbits/sec  0.047 ms  0/ 8504 (0%) 0.537/ 0.392/23.657/ 0.867 ms  850 pps 2329.37

```

где (с учетом -e)

#### Latency

Сквозная задержка в формате средняя/минимальная/максимальная/стандартная. Для теста требуется синхронизация часов клиента и сервера от одного источника (например, по протоколу PTP). Рекомендуется источник синхронизации GPS OCXO.

#### PPS

Число принятых в секунду пакетов.

#### NetPwr

Отношение пропускной способности к задержке.

## Изохронный тест UDP (клиент)

```
iperf -c 192.168.100.33 -u -e -i 1 --isochronous=60:100m,10m --realtime
```

```

-----
Client connecting to 192.168.100.33, UDP port 5001 with pid 14971
UDP isochronous: 60 frames/sec mean= 100 Mbit/s, stddev=10.0 Mbit/s, Period/IPG=16.67/0.005 ms
UDP buffer size: 208 KByte (default)

```

```

[ 3] local 192.168.100.76 port 42928 connected with 192.168.100.33 port 5001
[ ID] Interval      Transfer      Bandwidth    Write/Err    PPS    frames:tx/missed/slips
[ 3] 0.00-1.00 sec  12.0 MBytes  101 Mbits/sec  8615/0        8493 pps 62/0/0
[ 3] 1.00-2.00 sec  12.0 MBytes  100 Mbits/sec  8556/0        8557 pps 60/0/0
[ 3] 2.00-3.00 sec  12.0 MBytes  101 Mbits/sec  8586/0        8586 pps 60/0/0
[ 3] 3.00-4.00 sec  12.1 MBytes  102 Mbits/sec  8687/0        8687 pps 60/0/0
[ 3] 4.00-5.00 sec  11.8 MBytes  99.2 Mbits/sec  8468/0        8468 pps 60/0/0
[ 3] 5.00-6.00 sec  11.9 MBytes  99.8 Mbits/sec  8519/0        8520 pps 60/0/0
[ 3] 6.00-7.00 sec  12.1 MBytes  102 Mbits/sec  8694/0        8694 pps 60/0/0
[ 3] 7.00-8.00 sec  12.1 MBytes  102 Mbits/sec  8692/0        8692 pps 60/0/0
[ 3] 8.00-9.00 sec  11.9 MBytes  100 Mbits/sec  8537/0        8537 pps 60/0/0
[ 3] 9.00-10.00 sec 11.8 MBytes  99.0 Mbits/sec  8450/0        8450 pps 60/0/0
[ 3] 0.00-10.01 sec 120 MBytes  100 Mbits/sec  85867/0       8574 pps 602/0/0
[ 3] Sent 85867 datagrams
[ 3] Server Report:
[ 3] 0.00-9.98 sec 120 MBytes 101 Mbits/sec 0.009 ms 196/85867 (0.23%) 0.665/ 0.083/ 1.318/ 0.174 ms 8605 pps 18903.85

```

где (с учетом -e)

#### frames:tx/missed/slips

Общее число изохронных кадров или групп (burst), общее число не переданных идентификаторов кадров, общее число проскальзываний (slip) кадров

## Изохронный тест UDP (сервер)

```
iperf -s -e -u --udp-histogram=100u,2000 --realtime
```

```

-----
Server listening on UDP port 5001 with pid 5175
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)

```

```

[ 3] local 192.168.100.33 port 5001 connected with 192.168.100.76 port 42928 isoch (peer 2.0.13-alpha)
[ ID] Interval      Transfer      Bandwidth    Jitter      Lost/Total   Latency avg/min/max/stddev  PPS    NetPwr  Frames/Lost
[ 3] 0.00-9.98 sec  120 MBytes  101 Mbits/sec  0.010 ms  196/85867 (0.23%) 0.665/ 0.083/ 1.318/ 0.284 ms 8585 pps 18903.85 601/1
[ 3] 0.00-9.98 sec  T8(f)-PDF:
bin(w=100us):cnt(85671)=1:2,2:844,3:10034,4:8493,5:8967,6:8733,7:8823,8:9023,9:8901,10:8816,11:7730,12:4563,13:741,14:1
(5.00/95.00%=3/12,Outliers=0,obl/obu=0/0)
[ 3] 0.00-9.98 sec F8(f)-PDF: bin(w=100us):cnt(598)=15:2,16:1,17:27,18:68,19:125,20:136,21:103,22:83,23:22,24:23,25:5,26:3
(5.00/95.00%=17/24,Outliers=0,obl/obu=0/0)

```

где (с учетом -e)

#### Frames/lost

Общее число полученных кадров (групп), общее число потерянных или ошибочных кадров.

#### T8-PDF(f)

Гистограмма задержки для пакетов.

#### F8-PDF(f)

Гистограмма задержки для кадров.



## Примечания

1. Установка параметров окружения в iperf не поддерживается должным образом, как можно видеть в исходном коде.
2. Опция -B задает привязку на логическом (ip) и физическом (%device) уровне для клиента и сервера. У клиента влияет на системные вызовы bind() и обычно служит для привязки к определенному адресу IP и порту отправителя (например, iperf -c <host> -B 192.168.100.2:6002). Это задает источник пакетов но не применяется при маршрутизации. Здесь может возникнуть путаница при просмотре маршрутов и устройств. Например, если IP-адрес интерфейса eth0 указан в опции -B, а таблица маршрутизации для IP-адреса получателя (опция -s) указывает выходной интерфейс eth1, хост будет передавать через интерфейс eth1 пакеты с IP-адресом интерфейса eth0. Для задания выходного интерфейса в системе с несколькими подключениями следует применять форму -s <host>%device (требуется полномочия root) для обхода поиска в таблице маршрутизации хоста или настроенной таблицы маршрутизации хоста соответствующим образом.
3. Время соединения (трехэтапного согласования) TCP можно увидеть на стороне клиента iperf при работе с опцией -e (--enhanced). Поле ct=<value> в сообщениях о соединении (например, [ 3] local 192.168.1.4 port 48736 connected with 192.168.1.1 port 5001 (ct=1.84 ms) показывает, что 3WHS составляет 1,84 мсек).
4. Параметр NetPwr<sup>1</sup> является экспериментальным. Значение поля определяется отношением пропускной способности к задержке в сети. Для TCP в качестве задержки применяется период кругового обхода (RTT), для UDP - измеренное время сквозной задержки. Не следует воспринимать слово «мощность» (power) буквально, как величину работы, выполненной за единицу времени. Следует также отметить, что должна использоваться опция -i interval с протоколом TCP для задания частоты выборки RTT.

## Сведения об ошибках

См. <https://sourceforge.net/p/iperf2/tickets/>

## Авторы

Программа iperf2, созданная на основе iperf (разработка Mark Gates и Alex Warshavsky), стала более удобной и функциональной. В разработке участвовали Ajay Tirumala, Jim Ferguson, Jon Dugan <jdugan at x1024 dot net>, Feng Qin, Kevin Gibbs, John Estabrook <jestabro at ncsa.uiuc.edu>, Andrew Gallatin <gallatin at gmail.com>, Stephen Hemminger <shemminger at linux-foundation.org>, Tim Auckland <tim.auckland at gmail.com>, Robert J. McMahon <rjmcMahon at rjmcMahon.com>.

## Исходный код

<http://sourceforge.net/projects/iperf2/>

Перевод на русский язык

Николай Малых

[nmalykh@protocols.ru](mailto:nmalykh@protocols.ru)

<sup>1</sup>Network power - «мощность» сети.