

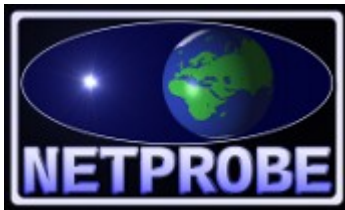
# IP Quality Monitor

## IQM-агент

---

### Руководство пользователя

Версия 3.11 редакция 2025-03-18



ООО «Нетпроб»  
123557, г.Москва,  
пер. Электрический, дом 3/10 стр. 3,  
офис 306А

---

Москва, 2025

# 1 Введение

IP Quality Monitor (далее в тексте IQM) – это аппаратно-программный комплекс, предназначенный для расчёта качественных метрик сетей TCP/IP. АПК состоит из двух частей — некоторого числа установленных на сети IQM-агентов (IQMA) и ядра системы (IQM менеджер, IQMM). Между агентами проводятся тесты, итоги которых поступают в ядро. Определим основные понятия:

## Определение 1.

Метрика качества	измеряемая характеристика сети TCP/IP как цельного материального объекта, полученная в результате статистического наблюдения с использованием технических средств.
------------------	--

## Определение 2.

Статистическая величина	количественная характеристика размеров явлений, их соотношения, степени изменения, взаимосвязи.
-------------------------	---

## Определение 3.

Статистическое наблюдение	сбор сведений, заключающийся в регистрации признаков и фактов, которые характеризуют каждую единицу исследуемой совокупности.
---------------------------	---

## Определение 4.

Генеральная совокупность	совокупность всех признаков и фактов, относительно которых предполагается делать выводы. Для нашего статистического наблюдения — это весь трафик, пропущенный по сети.
--------------------------	--

## Определение 5.

Выборка	часть генеральной совокупности элементов, которая охватывается наблюдением. Для нашего статистического наблюдения — это часть трафика, пропущенного по сети.
---------	--

## Определение 6.

Тест	трафик от точки создания до точки получения, созданный искусственно и предназначенный для расчёта метрик качества сети.
------	---

## Определение 7.

Агент-инициатор	точка создания <i>теста</i> (АПК).
-----------------	------------------------------------

## Определение 8.

Сопряжённый агент	точка назначения <i>теста</i> (АПК).
-------------------	--------------------------------------

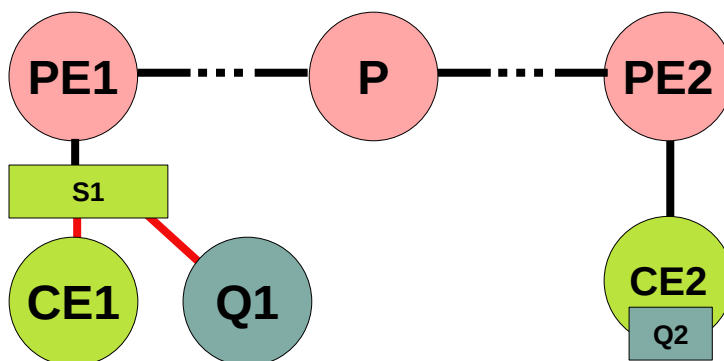
IQMA способен выступать как в роли агента-инициатора, так и в роли сопряжённого для проведения тестов и сбора метрик. Дополнительно он поддерживает широкий спектр протоколов, позволяющих пользователю системы выполнять необходимые задачи по расчёту *метрик качества* с уже установленными на сети сопряжёнными агентами. Руководство по протоколам доступно отдельно от данной документации.

Настоящее руководство предназначено для системных администраторов, сопровождающих IQM. От администратора требуются следующие навыки:

- уверенное понимание принципов работы IQM,
- опыт работы со стеком протоколов TCP/IP,
- знание операционной системы Linux на уровне системного администратора.

## 2 IQM-агент

Типовая схемой включения IQM-агента в сеть TCP/IP представлена на [рисунке 1](#). Где Q1 это агент-инициатор, а Q2 — сопряжённый. Возможны иные варианты включения, например, сопряжённый так же может включаться Т-образно, либо агент может быть встроен в саму аппаратную платформу в случае использования стандартных протоколов.



**Рисунок 1.** Агент-инициатор включен безопасно Т-образно, сопряжённый ему использует пользовательское оборудование как программную платформу.

Сам же агент представляет собой программный процесс, запускаемый на операционной системе общего назначения. Чаще всего используется операционная система (ОС) Linux, однако, возможны и иные. При приобретении АПК агент уже установлен. В случае поставки агента отдельно от аппаратной платформы, он содержится в пакете нужного для ОС формата. Для Linux это, как правило, **dpkg** либо **rpm**. Установка таких пакетов (равно как и обновление) осуществляется стандартным для ОС способом (например, через запуск `dpkg -i package-name.deb` либо `rpm -ihv package-name.rpm`).

После установки агент автоматически запускается. Для каждого из дистрибутивов это реализовано стандартным способом, что позволяет исполнять текущие задачи уже знакомым для администраторов ОС способами.

Приведём примеры, характерные для разных версий ОС Linux.

Узнать состояние процесса:

```
systemctl status iqma.service
```

```
service iqma status
```

```
/etc/init.d/iqma status
```

```
ps -ax|grep iqm_agent
```

Запустить процесс:

```
systemctl start iqma.service
```

```
service iqma start
```

```
/etc/init.d/iqma start
```

Остановить процесс:

```
systemctl start iqma.service
```

```
service iqma start
```

```
/etc/init.d/iqma start
```

Обращаем внимание, что сам исполнимый файл называется **iqm\_agent**, а системный модуль, который производит запуск-останов — **iqma**. Исполнимый файл и дополнительные файлы для ОС класса Unix (включая ОС Linux) после установки располагаются в каталоге `/usr/local/iqm_agent`. Для ОС иных классов производитель следует традиции размещения. Возможно изменение администраторами на более удобное им, однако, мы это не рекомендуем, так как может усложниться обновление.

### 3 Стандартные ключи командной строки

При обычной работе IQM не требуется запускать агент вручную, его текущая конфигурация и создание или модификация *тестов* производится из системы управления (IQMM). В данном тексте мы описываем дополнительные возможности, доступные в агенте, если администратору системы (либо отдельных хостов, содержащих агента) требуется то, что в IQMM либо недоступно, либо затруднено.

На ОС класса Unix запуск производится через указание полного пути исполнимого файла, так как включать агента в переменную среды PATH нецелесообразно. В качестве параметров командной строки можно указывать дополнительные ключи, модифицирующие поведение агента. Выглядит это традиционным методом:

```
/usr/local/iqm_agent/iqm_agent -O --opt1 --opt2=va2
```

Перечислим те ключи, которые выполнимы при любой версии агента. Отметим, что мы следуем традициям Unix и GNU и для коротких версий ключей возможное значение указывается через пробел, а для длинных версий через знак равенства.

Ключ командной строки	Описание
--help	Выдать подсказку по поддерживаемым ключам командной строки с указанием формата, а так же значений по умолчанию и завершить работу
-h	—"__
--version	Выдать версию агента, список модулей, протоколов, версий сторонних библиотек и завершить работу
-V	—"__
--copyright	Выдать лицензионный договор присоединения и завершить работу
--tech-support	Выдать краткую информацию об аппаратной платформе и завершить работу
--pidfile=/path/file.pid	Сохранять цифровой идентификатор процесса в файле /path/file.pid Если путь к файлу не указан, будет использоваться /var/run/iqm_agent.pid Данный ключ рекомендуется использовать с именем, так как файл с идентификатором создаётся в любом случае.
-P /path/file.pid	—"__
--level=N	Уровень журналирования (1-7), чем больше, тем подробнее.

Для исправления тех ключей командной строки, которые нужны в процессе работы, мы рекомендуем не ручной запуск агента, а модификацию файлов дополнительной настройки. Точное местоположения таких файлов зависит от аппаратной платформы и дистрибутива ОС. Это могут быть /etc/default/iqma, или /etc/sysconfig/iqm\_agent, или /lib/systemd/system/iqma.service и иные. Администратору соответствующей ОС формат таких файлов известен, мы следуем сложившимся в отрасли стардартам.

Версии ключей, которые приводят к немедленному окончанию работы следует употреблять по одиночке только при ручном запуске агента. В файлах дополнительной настройки их не следует указывать, так как это может привести к останову агента и формированию цикла перезапуска без реального старта.

## 4 Ручной запуск агента

Если администратору требуется ручной запуск агента, следует в первую очередь остановить стандартными методами встроенное в пакет исполнение, как указано в [разделе 2](#). Если этого не сделать, то возможная привязка к общим ресурсам, которые использует агент, может привести к аварийному завершению процесса.

За исключением способов выдачи версии, подсказки и технической информации, администратору не требуется ручной запуск, тем не менее мы не ограничиваем его возможность. Из стандартной Unix-оболочки (shell) это осуществляется следующим

образом:

```
/usr/local/iqm_agent/iqm_agent -P /run/iqm_agent.pid
```

Агент не использует стандартный ввод, ошибки на стандартный вывод и стандартный канал ошибок выводит только в случае, если невозможно запустить журнал, либо он выключен вручную. Если в ОС есть поддержка `fork(2)`, агент становится демоном (постоянно исполнимым процессом без привязки к терминалу). В случае же отсутствия такой поддержки (напишем сразу, такую ОС вам ещё следует постараться найти, это большая редкость) следует использовать иной способ вызова, хотя и сохраняющий привязку к стандартным потокам, но позволяющий выйти в `shell` для дальнейшей работы.

```
/usr/local/iqm_agent/iqm_agent -P /run/iqm_agent.pid &
```

Запуск на ОС классов, отличных от Unix, ввиду отсутствия текущего спроса будет описана дополнительно.

Окончание процесса предусмотрено только в случае ошибок или остановка всех системных модулей. Агент регулярно проводит внутренние проверки с тем чтобы минимизировать риски окончания работы. В случае необходимости агент перезапускается внешним модулем, встроенным в пакет ОС. Нулевой код возврата в ОС производится только в случае корректного запроса на окончание администратором. В случае ошибок код возврата ненулевой.

## 5 Права процесса

Штатно агент работает под пользовательским идентификатором супервизора (`root`, `uid 0`). Мы не рекомендуем это понижать. Для большого числа сетевых протоколов требуется такой доступ. Однако, в случае поставки ограниченной версии агента или при использовании пользователями только определённых версий протоколов, возможно понижение до выбранного вами уровня. Гарантии полной работоспособности ООО «Нетпроб» в этом случае не даёт. Возможные проблемы могут быть рассмотрены по отдельным запросам.

В процессе начала работы агент понижает приоритеты и файловые маски. Все создаваемые файлы не могут быть прочтены пользователями, не входящими в группу `iqm`. Члены группы `iqm` могут читать создаваемые файлы, но не модифицировать их. Супервизору доступны все права. Дополнительные ограничения, например, устанавливаемые модулями SELinux, следует рассматривать по мере необходимости, в традиционной поставке проблем не отмечалось.

Файловый сокет для связи с внешними программами имеет права, позволяющие запись группе `iqm`. Это позволяет тем внешним программам, которые запускаются под пользователями данной группы или имеют выставленный в правах `setgid` бит, производить нормальную работу без запроса прав супервизора.

## 6 Модули агента

Агент имеет модульную структуру. Большинство модулей при создании бинарного файла агента согласно техническим требованиям конечного пользователя может быть включено или выключено. В данном разделе мы перечисляем все возможные модули, так как документация общая для всех поставок.

<b>Имя модуля</b>	<b>Описание модуля</b>	<b>Режим работы без модуля</b>
FOPT	Чтение ключей командной строки из файла.	Чтение ключей командной строки только из штатных переменных
FCFG	Чтение и сохранение настроек агента и тестов из файла.	Чтение настроек агента из ключей командной строки, сохранение отсутствует. Конфигурация тестов проводится либо через систему управления, либо через файловый сокет связи.
SRV-CFG	Чтение настроек агента и тестов с внешнего сервера (системы управления) по HTTP/FTP.	Отсутствие чтения настроек агента и теста с внешнего сервера.
SCHD	Планировщик для запуска тестов по расписанию.	Запуск тестов возможен только синхронно, через файловый сокет связи с внешними программами.
CRON	Запуск из планировщика по стоп-шаблонам времени, а не только по истечению периода	Отсутствие запуска по стоп-шаблонам времени
SCDP	Возможность отладочной записи в журнал данных и событий в планировщике. Предоставляется только в рамках договора технической поддержки.	Отсутствие записи отладочной информации планировщика
CTRL	Принимать запросы от системы управления и агентов-инициаторов для конфигурации, исполнения тестов по запросу и регулярных тестов.	Агент не использует контрольный протокол, тесты могут запускаться только через файловый сокет связи.
U0R	Поддерживать в рамках модуля CTRL запросы на сопряжённый агент по протоколу U0	Не принимать запросы на сопряжённый агент протокола U0.
U1R	Поддерживать в рамках модуля CTRL запросы на сопряжённый агент по протоколу U1	Не принимать запросы на сопряжённый агент протокола U1.
CDR	Записывать итоговые метрики каждого теста, запущенного по расписанию в файл для отправки в систему управления.	Не записывать итоговые метрики.
CDR-HTTP	Отправка файлов в систему управления по протоколу HTTP.	Оправлять файлы иными способами.
CDR-FTP	Отправка файлов в систему управления по протоколу FTP без использования внешних программ.	Отправлять файлы иными способами.

Имя модуля	Описание модуля	Режим работы без модуля
OND	Поддерживать в рамках модуля CTRL запросы на тесты по требованию из системы управления.	Не принимать запросы на тесты по требованию.
SSL	Поддерживать протоколы с TLS/SSL (HTTPS, FTPS и иные). Требуется внешняя библиотека. Статическое встраивание возможно в рамках договора поставки или техподдержки.	Не поддерживать протоколы с TLS/SSL.
LIVEMON	Внедрять в агент быструю реакцию на превышение метриками указанных нормативов без использования системы управления.	Не использовать по итогам теста быструю реакцию.
USOCK	Использовать файловый сокет связи с внешними программами. Например, для синхронного тестирования, web-iqm или программ третьих лиц.	Не использовать файловый сокет связи.
HIST	Хранить историю синхронного тестирования в агенте.	Не хранить историю синхронного тестирования.
MCOR	Использовать несколько точек доставки файлов с метриками итогов теста. Это могут быть как несколько систем управления, так и одна штатная система управления и несколько хостов третьих лиц.	Использовать одну точку доставки файлов с метриками итогов теста.
RSDP	Записывать итоги теста не только в файл данных, но и в журнал.	Записывать итоги теста только в файл данных, либо в случае синхронного тестирования в файловый сокет связи.
TCONG	Использовать алгоритмы управления потоком для протоколов, где есть TSP. Требуется поддержка в ОС.	Использовать алгоритмы по умолчанию.
TMS	Выдавать в журнал состояние счётчиков статистики ОС (/proc/stat для ОС Linux) до и после запуска теста.	Не использовать счётчики ОС.
TPTY	По команде из web-iqm через файловый сокет связи иметь возможность запускать программы третьих фирм (iperf3, bwping) и iqm-test для выполнения сравнительного тестирования разными средствами.	Не использовать запуск программ третьих фирм.

Имя модуля	Описание модуля	Режим работы без модуля
G107	Включить расчёт синтетический MOS и Rfactor согласно G.107 при указании в настройках теста профиля трафика с пользовательским ожиданием.	Не рассчитывать MOS и Rfactor согласно G.107
CCC	Core Control Connector. Автоматическое соединение с внешней прокси-программой для доступа к управлению агентом из системы управления, если агент находится за сетевой трансляцией NAT или межсетевым экраном.	Не соединяться автоматически с внешним прокси
U0	Поддерживать фирменный протокол тестирования ООО «Нетпроб» между двумя агентами либо между IQM-агентом и универсальным сопряжённым агентом.	Не поддерживать протокол U0.
U1	Поддерживать фирменный протокол тестирования ООО «Нетпроб» между двумя агентами либо между IQM-агентом и универсальным сопряжённым агентом, при этом агент-инициатор может находиться за точкой трансляции адресов (NAT).	Не поддерживать протокол U1.
U9	Поддерживать фирменный протокол тестирования ООО «Нетпроб» между двумя агентами, при этом агент-инициатор может находиться за точкой трансляции адресов (NAT). В качестве протокола управления используется UDP.	Не поддерживать протокол U9.
U7	Поддерживать протокол UDP echo	Не поддерживать протокол UDP echo.
C0	Поддерживать протокол третьей фирмы №1 с расширениями ООО «Нетпроб» с миллисекундной точностью.	Не поддерживать протокол C0.
C1	Поддерживать протокол третьей фирмы №1 с миллисекундной точностью без расширений ООО «Нетпроб».	Не поддерживать протокол C1.

Имя модуля	Описание модуля	Режим работы без модуля
C2	Поддерживать протокол третьей фирмы №1 с расширениями ООО «Нетпроб» с микросекундной точностью.	Не поддерживать протокол C2.
C3	Поддерживать протокол третьей фирмы №1 с микросекундной точностью без расширений ООО «Нетпроб».	Не поддерживать протокол C3.
J0	Поддерживать протокол третьей фирмы №2 с расширениями ООО «Нетпроб».	Не поддерживать протокол J0.
J1	Поддерживать протокол третьей фирмы №2 без расширений ООО «Нетпроб».	Не поддерживать протокол J1.
T1	Поддерживать протокол TWAMP (RFC-5357)	Не поддерживать протокол T1
IO	Поддерживать протокол ICMP echo	Не поддерживать протокол ICMP echo.
PPPOE	Поддерживать протокол PPPoE.	Не поддерживать протокол PPPoE.
URL	Поддерживать загрузку файла с сервера по HTTP/FTP/др. Тип адреса выбирается автоматически.	Не поддерживать протокол URL.
URLUP	Поддерживать загрузку файла на сервер по HTTP/FTP/др. Тип адреса выбирается автоматически.	Не поддерживать протокол URLUP.
URL4	Поддерживать загрузку файла с сервера по HTTP/FTP/др. Тип адреса IPv4.	Не поддерживать протокол URL4.
URLUP4	Поддерживать загрузку файла на сервер по HTTP/FTP/др. Тип адреса IPv4.	Не поддерживать протокол URLUP4.
URL6	Поддерживать загрузку файла с сервера по HTTP/FTP/др. Тип адреса IPv6.	Не поддерживать протокол URL6.
URLUP6	Поддерживать загрузку файла на сервер по HTTP/FTP/др. Тип адреса IPv6.	Не поддерживать протокол URLUP6.
DNS	Поддерживать тестирование DNS-серверов выбранным типом запроса.	Не поддерживать тестирование DNS-серверов.

Имя модуля	Описание модуля	Режим работы без модуля
CMD	Поддерживать запуск внешних программ для расширения возможностей агента во время теста.	Не поддерживать запуск внешних программ.
MPGCMD	Поддерживать запуск <code>mreceiver</code> с трансляцией метрик теста.	Не поддерживать запуск с трансляцией метрик.
DHCP	Поддерживать тестирование DHCP-серверов версии 4 выбранным типом запроса.	Не поддерживать тестирование DHCP-серверов.
G0	Поддерживать протокол Unicast ETH-LB согласно G.8013	Не поддерживать протокол G0.
G1	Поддерживать протокол Multicast ETH-LB согласно G.8013	Не поддерживать протокол G1.
EL2	Поддерживать протокол тестирования сети Eltex ESR.	Не поддерживать протокол EL2.
SG	Поддерживать протокол SNMP GET.	Не поддерживать протокол SG.
SW	Поддерживать протокол SNMP GETNEXT.	Не поддерживать протокол SW.
SBW	Поддерживать протокол SNMP GETBULK.	Не поддерживать протокол SBW.
B0	Поддерживать протокол Bundle 0.	Не поддерживать протокол B0.
I3SDU	Поддерживать протокол iperf3 UDP от агента-инициатора к сопряжённому.	Не поддерживать протокол I3SDU.
I3DSU	Поддерживать протокол iperf3 UDP от сопряжённого к агенту-инициатору.	Не поддерживать протокол I3DSU.
I3SDT	Поддерживать протокол iperf3 TCP от агента-инициатора к сопряжённому.	Не поддерживать протокол I3SDT.
I3DST	Поддерживать протокол iperf3 TCP от сопряжённого к агенту-инициатору.	Не поддерживать протокол I3DST.

Теперь рассмотрим модули подробнее.

## 7 Ключи командной строки модулей

Помимо модульной структуры, в зависимости от возможностей ОС агент может поддерживать дополнительные возможности. Некоторыми из них можно управлять через командную строку.

Если администратору не нужен начальный вывод предупреждений и ошибок, можно на старте указать следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--nstdout</code>	Выводить на начальном этапе ошибки и предупреждения.	Вывод на начальном этапе запуска ошибок

Если в системе доступно ведение журнала в файл, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--log=S</code>	Не вести журналирование.	Записывать журнал в файл

Если в системе доступно ведение журнала через системный демон, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--syslog</code>	Не вести журналирование.	Записывать журнал в syslog

Если в системе доступно ведение журнала по UDP-протоколу, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--udp-syslog=H:P</code>	Не вести журналирование.	Отсылать журнал в удалённый хост H на UDP-порт P

Если в системе есть `fork(2)`, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--nodaemon</code>	Становиться демоном	Не отсоединяться от контрольного терминала, сохраняя работу агента без выхода в shell-сессию. Полезно для отладки либо на экзотических ОС. При штатной работе не применяется.

Если в системе есть IP\_RECVTOS либо IPV6\_RECVTCLASS, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
--notos	Обрабатывать поле TOS или TCLASS каждого пакета UDP либо ICMP.	Не использовать чтение TOS и TCLASS. Как правило это не нужно, но возможно для отладки либо в экзотических конфигурациях.

Если в системе есть IP\_RECVTTL либо IPV6\_RECVNOPLIMIT, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
--notttl	Обрабатывать поле TTL или NOPLIMIT каждого пакета UDP либо ICMP.	Не использовать чтение TTL и NOPLIMIT. Как правило это не нужно, но возможно для отладки либо в экзотических конфигурациях.

Если в системе есть возможность фиксации входящего адреса на каждом пакете, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
--nosamesend	Отсылать ответы с того же адреса, на который они направлены.	Отсылать ответы согласно маршрутной таблице. Как правило это не нужно, но возможно для отладки либо в экзотических конфигурациях.

Если в системе есть поддержка IPv6, можно на старте использовать следующие ключи:

Ключ	Значение по умолчанию	Описание
--ipv4-only	Работать с двумя типами адресов одновременно.	Включить работу только для адресов IPv4.
--ipv6-only	Работать с двумя типами адресов одновременно.	Включить работу только для адресов IPv6.

Если в системе есть файловые сокеты и в агенте разрешена данная возможность при сборке, можно на старте использовать следующий ключ:

Ключ	Значение по умолчанию	Описание
--pid-socket=S	/tmp/ia35-pids	Установить имя для файлового сокета, контролирующего повторный запуск.

## FOPT

Если при запуске агента в конце командной строки передавать в качестве ключа имя, которому предшествует «коммерческая at» (@filename), то filename будет рассматриваться как имя файла, в котором в текстовом виде перечислены ключи. Это может быть удобно для делегирования прав по записи определённых ключей, чтобы не редактировать системные файлы. После исправления такого файла и перезапуска, агент будет работать в новом режиме. Кроме того, в случае отсутствия модуля FCFG в таком файле можно перечислять настройки агента в виде ключей командной строки.

## FCFG

Как правило, в агенте этот модуль включен. Однако, возможны поставки, где на устройстве нет хранилища для файлов. В этом случае предопределённые настройки могут быть указаны в ПЗУ в модуле FOPT, а регулярное обновление может осуществляться в модуле SRV-CFG. Если модуль представлен в агенте, обрабатывается следующий ключ:

Ключ	Значение по умолчанию	Описание
--config=name -C name	/etc/iqm_agent.conf либо иное, если аппаратная платформа имеет ограничения	Файл хранения конфигурации агента и тестов

В случае отсутствия модуля, обрабатываются следующие ключи:

Ключ	Значение по умолчанию	Описание
--agent-name=S	@auto@	Имя агента, строка @auto@ автоматически создаёт имя с префиксом IQMA и суффиксом в виде адреса. Agent name в IQMM, SID в FCFG
--server-timeout=F -T F	3.0	Время ожидания пакета от источника до фиксации таймаута/ Server timeout в IQMM, ServerTimeout в FCFG.
--enable-key=S	отсутствует	Лицензионный ключ

Если в агенте нет модуля FCFG, но есть модули CTRL либо USOCK либо SCHD, обрабатывается следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--measure-shared-secret=S</code>	Смотрите в информации поставки. Строка может отличаться.	Разделяемая секретная строка для аутентификации в U0, U1 и иных. Agent password в IQMM, Password в FCFG.

Если в агенте нет модуля FCFG, но есть модуль CTRL, обрабатываются следующие ключи:

Ключ	Значение по умолчанию	Описание
<code>--control-port=N</code>	1189	TCP-порт контрольного протокола управления из IQMM и для протоколов U0, U1. Listen port в IQMM, ListenPort в FCFG.
<code>--control-shared-secret=S</code>	Смотрите в информации поставки. Строка может отличаться.	Разделяемая секретная строка для конфигурации (поддерживается в системах 3.5 и выше). CtlPassword в FCFG.
<code>--remote-admin-ip=S</code>	Не установлено.	Список адресов, которым разрешён доступ через точку с запятой. List of administrator's IPs в IQMM, RemoteAdminIP в FCFG.

Если в агенте нет модуля FCFG, но есть модули CDR либо SRV-CFG, обрабатывается следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--core-ip=S</code>	iqmm	Адрес системы управления для отправки CDR. Core IP в IQMM, CoreIP в FCFG.

Если в агенте нет модуля FCFG, но есть модуль CDR, обрабатываются следующие ключи:

Ключ	Значение по умолчанию	Описание
--cdr-num-records=N	50	Число записей в CDR, после достижения которых будет проводиться отправка в систему управления. CDR filesize (NoR) в IQMM, ResFileNOR в FCFG.
--cdr-timeout=F	5.0	Время накопления данных в CDR в минутах, после превышения которого будет проводиться отправка в систему управления вне зависимости от числа накопленных записей, если оно больше нуля. CDR file timeout (min) в IQMM, ResFileTimeOut в FCFG.
--cdr-send-timeout=F	5.0	Время ожидания в минутах, если отсылка CDR в систему управления в очередной раз прошла неудачно. CDR send timeout (min) в IQMM. SendTimeOut в FCFG.
--spool-dir=S	/tmp	Каталог для накопления CDR, должен быть доступен на запись агенту. Spool dir в IQMM. SpoolDir в FCFG.
--send-command=S	FTP	Команда отсылки CDR. CDR transport script в IQMM. SendCmd в FCFG.
--ftp-user=S	iqm	FTP-пользователь, под которым проводится пересылка CDR. FTP user в IQMM, FtpUser в FCFG.
--ftp-password=S	Случайная строка	FTP-пароль, который используется для пересылки CDR. FTP password в IQMM, FtpPassword в FCFG.

## SRV-CFG

Это модуль включается согласно техническому заданию. Возможны поставки, где он отсутствует. Он отвечает за возможность чтения конфигурации с системы управления, что позволяет сохранять единую точку конфигурации в случае массовой миграции агентов между точками присутствия в сети. Обрабатываются следующие ключи:

Ключ	Значение по умолчанию	Описание
<code>--server-config=URL</code>	Не установлено	Адрес ресурса системы управления, с которого читается конфигурация агентов и тестов. Config source в IQMM. ConfigSrc в FCFG.
<code>--srv-cfg-cli-only</code>	Использовать и командную строку и данные системы управления, которые могут затереть конфигурацию командной строки.	Использовать только URL, указанный в командной строке.
<code>--srv-cfg-once-only</code>	Запускать серверную конфигурацию как на старте, так и регулярно.	Читать конфигурацию из системы управления только на старте.
<code>--srv-cfg-no-override</code>	Затирать конфигурацию успешно прочтенной каждый раз.	Не затирать текущую конфигурацию кроме самого первого чтения. Полезно для экзотических случаев.
<code>--srv-cfg-long-timeout=F</code>	3600.0	Время между циклами чтения конфигурации из системы управления в секундах.

Если представлен ключ командной строки `--server-config` и в принятой конфигурации так же есть Config source, но иной, то после первого обновления для следующих будет использоваться URL из системы управления. Это можно купировать ключами `--srv-cfg-cli-only` или `--srv-cfg-once-only`. Вполне рабочим вариантом может считаться агент только с модулем SRV-CFG и без модуля FCFG (настройки каталогов и иных можно в этом случае вынести в `/etc/default/iqma`). При возможной недоступности системы управления на старте это, разумеется, сокращает время возможной работы, однако позволяет постоянно иметь синхронную с системой управления настройку агента.

Если в агенте есть модуль FCFG, обрабатываются следующие ключи:

Ключ	Значение по умолчанию	Описание
<code>--srv-cfg-first</code>	Читать файл конфигурации, а обновлять из системы управления после этого.	Вначале читать конфигурацию из системы управления, а затем из файла.

Ключ	Значение по умолчанию	Описание
<code>--srv-cfg-nofile</code>	Перед подстановкой переменных (см. ниже) читать файл конфигурации для подстановки нужных переменных. Дальнейшее использование прочтенной конфигурации не проводится.	См. ниже, после описания переменных.

Если модуль SRV-CFG отсутствует, конфигурация читается из файла при наличии модуля FCFG, либо из командной строки. Помимо этого, текущая работа из системы управления модифицирует конфигурацию по требованию администратора.

Важной особенностью модуля SRV-CFG являются переменные автоподстановки в URL. Для гибкой настройки пакета агента возможно использование в URL при ключе командной строки `--server-config` следующих переменных:

<code>\$COREIP</code>	При соединении с серверов вместо переменной подставится значение CoreIP (адрес сервера) из конфигурации. Тем самым одной и той же поставкой пакета агенты вы можете обращаться к разным серверам с хранящейся конфигурацией.
<code>\$AGENTNAME</code>	При соединении с серверов вместо переменной подставится значение SID (имя агента) из конфигурации.
<code>\$FTPUSER</code>	При FTP-соединении с сервером в качестве входного имени использовать FtpUser из конфигурации.
<code>\$FTPPWD</code>	При FTP-соединении с сервером в качестве пароля использовать FtpPassword из конфигурации.
<code>\$AGENTIP</code>	<p>В тех случаях, когда имя агента вносить в пакет поставки нецелесообразно, например, из-за большого их числа, либо для автоматического обновления, возможно использование данной переменной. В этом случае мы рекомендуем использовать уникальные IP-адреса для каждого агента в системе управления. Например, при указании в файле модуля FOPT <code>@/usr/local/iqm_agent/iqm_agent.args</code> ключа командной строки <code>--server-config=<a href="http://10.10.10.10/iqm/getconfig?login=loadcfg&amp;pass=Secret&amp;agent_ip=\$AGENTIP">http://10.10.10.10/iqm/getconfig?login=loadcfg&amp;pass=Secret&amp;agent_ip=\$AGENTIP</a></code> (одной строкой!) будут проводиться запросы указанного URL с этими параметрами. Если в IQMM модуль <code>getconfig</code> существует и находит агента с нужным IP-адресом в базе, то будет возвращена текущая конфигурация. Если же нет, будет выдана ошибка.</p> <p>В качестве IP-адреса будет использоваться <u>ближайший адрес пробника к запрошенному серверу</u>. (Для профессионалов: результат <code>getsockname(2)</code> после <code>connect(2)</code>). При традиционно настроенной маршрутизации это обычно шлюз по умолчанию. При статическом маршруте на пробнике в сторону</p>

системы управления (например, на PPPoE пробниках), это будет адрес интерфейса, через который будут отправляться пакеты на систему управления.

Будьте аккуратны, если в качестве адреса сервера используется имя, будет обращение к системному резолверу (`getaddrinfo(3)`). Это может вести к увеличению длительности ответа!

Подробные консультации по данной возможности доступны дополнительно по согласованным каналам связи.

В процессе автоподстановки может потребоваться сохранение URL для всех видов запуска агента, и модификация реального обращения в сеть через конфигурацию агента. Это удобно для регулярного обновления уже существующих установок агентов, где конфигурация тем или иным образом сохранена в файл, а настройка обновления с сервера либо ставится заново, либо имеет новые настройки. В этом случае либо через файл модуля `FOPT @/usr/local/iqm_agent/iqm_agent.args` ключа командной строки загрузки, либо через настройки запуска, выставленные в ОС (`/etc/default/iqma`, `/lib/systemd/system/iqma.service` или аналогичными), можно указать для всех агентов одинаковые настройки.

Например, при ключах `--server-config=http://$COREIP/iqm/getconfig?login=loadcfg&pass=Secret&agent_name=$AGENTNAME` (одной строкой!) `--srv-cfg-first` первое чтение файла конфигурации (допустим, это может быть `/etc/iqm_agent.conf`) будет проводиться один раз, оттуда использованы настройки CoreIP для переменной `$COREIP` и `SID` для переменной `$AGENTNAME`, однако реальная работа начнётся только после чтения конфигурации с сервера и, если там хранится иная, то в файле конфигурации будет проведено обновление. Рекомендуется сохранять один CoreIP в рамках домена тестирования, а `SID` (имя агента) делать уникальным.

Когда агенты в домене тестирования используют уникальные IP-адреса, то можно использовать ключ `--server-config=http://$COREIP/iqm/getconfig?login=loadcfg&pass=Secret&agent_ip=$AGENTIP` (одной строкой!). В этом случае система управления IQMM будет искать настройки и конфигурацию агента по IP-адресу. Это удобно для единообразной настройки всех агентов, можно даже имя агента в пакете ПО выставить одинаковое, так как при чтении из IQMM будет проведено обновление. Так же, данным путём можно легко менять вышедшие из строя пробники на исправные. Первичная конфигурация может не включать тесты, тогда до первого успешного обновления запуска реальных тестов проводиться не будет.

Ключ `--srv-cfg-nofile` по умолчанию не используется. Он введён для сложных случаев, когда чтение файла конфигурации для автоподстановок переменных не нужно. В этом случае для переменных будут использованы значения по умолчанию. Пока мы не можем сформулировать случаи, когда это необходимо, однако, оставляем за пользователями системы возможность гибко управлять агентом.

В целях упрощения поставок предусмотрены следующие специальные значения URL:

ftp или FTP	Осуществляется автоподстановка на: <code>ftp://\$FTPUSER:\$FTPPWD@\$COREIP/cfg/\$AGENTNAME_iqm_agent.conf</code>
http или HTTP	Осуществляется автоподстановка на: <code>http://\$COREIP/cfg/\$AGENTNAME_iqm_agent.conf</code>

tftp                   Осуществляется автоподстановка на:  
tftp://\$COREIP/\$AGENTNAME\_iqm\_agent.conf

## SCHD

Как правило, этот модуль поставляется в агенте. Возможно удешевление поставки, если данный модуль отсутствует, однако в этом случае можно будет выполнять только синхронное тестирование, либо тесты по требованию при наличии нужных модулей. Отдельных настроек модуль планировщика не требует.

## CRON

Как правило, этот модуль поставляется в агенте. Возможно удешевление поставки с выключением модуля, однако в этом случае будет невозможно запускать тесты на базе cron-таблиц, то есть в часы, минуты и дни, выбранные администратором, что удобно для выполнения тестирования в нужные часы, например при наименьшей нагрузке на сеть. Формат соответствующего поля Cron-like template (RunAt в FCFG) в тесте:

МИНУТЫ;ЧАСЫ;ДНИ\_МЕСЯЦА;МЕСЯЦЫ;ДНИ\_НЕДЕЛИ

Каждое поле представляет из собой список из ЭЛЕМЕНТов, разделённых запятой:

ЭЛЕМЕНТ[ЭЛЕМЕНТ[ЭЛЕМЕНТ...]]

либо звёздочку или пустую строку, что означает выбор любого возможного ЭЛЕМЕНТа.

ЭЛЕМЕНТ представляет из себя либо число, либо два числа разделённых дефисом (минусом):

ОДНО\_ЧИСЛО либо МЕНЬШЕЕ\_ЧИСЛО-БОЛЬШЕЕ\_ЧИСЛО.

Допустимые ЧИСЛА для МИНУТ от 0 до 59, для ЧАСОВ от 0 до 23, для ДНЕЙ\_МЕСЯЦА от 1 до 31 (неверные дни игнорируются), для МЕСЯЦЕВ от 1 до 12, для ДНЕЙ\_НЕДЕЛИ от 0 до 7 (0 и 7 — воскресенье). Аналогично crontab (5).

Запуск теста проводится, когда данные gmtime(3), то есть всемирное координированное время (UTC) на аппаратной платформе агента совпадает с указанными МИНУТАМИ, ЧАСАМИ, (ДНЯМИ\_МЕСЯЦА **или** ДНЯМИ\_НЕДЕЛИ) и МЕСЯЦАМИ. Пример:

**1-3,10;3,4,6-8;17,18,20-22;1;0** означает:

запускать 17, 18, 20, 21 и 22 января **и** в каждое воскресенье января в минуты: 03:01, 03:02, 03:03, 03:10, 04:01, 04:02, 04:03, 04:10, 06:01, 06:02, 06:03, 06:10, 07:01, 07:02, 07:03, 07:10, 08:01, 08:02, 08:03, 08:10.

Точные секунды запуска агент определяет автоматически, распределяя все тесты внутри минуты, с тем чтобы не было большого всплеска одновременных соединений. Настроек командной строки модуль CRON не требует.

## SCDP

Если включен модуль SCHD, данный дополнительный модуль позволяет исполнять отладочный вывод в журнал о состоянии планировщика по мере необходимости. Для текущей эксплуатации он не рекомендован, так как вносит дополнительные задержки, однако в отведённой среде тестирования (в лаборатории) допустим к использованию. Модулем обрабатывается следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--scheduler-debug</code>	Не проводить отладочного журналирования.	Включить возможность отладочного вывода о состоянии планировщика. Сам вывод проводится при изменении состояния, либо по сигналу SIGUSR2.

## CTRL

В подавляющем большинстве поставок агента этот модуль включен. Он отвечает за две важные задачи — конфигурацию агента из системы управления и проведение тестирования по протоколам U0 и U1 между агентами. Не имеет большого смысла экономить при поставке на этом модуле, так как вы будете искусственно ограничивать возможности агента. Однако, если вам интересно только синхронное тестирование, а в качестве сопряжённых агентов вы используете либо универсальный сопряжённый агент, либо оборудование сторонних производителей (например, фирмы №1 или №2), то возможно заказать сборку без данного модуля.

Дополнительных ключей, кроме изложенных в разделе про FCFG, не предусмотрено.

## U0R

Данный модель отвечает за функционал сопряжённого агента протокола U0. Отключение данного модуля возможно только при наличии в поставке универсального сопряжённого агента. Модуль требует наличия CTRL.

## U1R

Данный модель отвечает за функционал сопряжённого агента протокола U1. Отключение данного модуля возможно только при наличии в поставке универсального сопряжённого агента. Модуль требует наличия CTRL.

## CDR

Важной частью агента является сохранение результатов тестирования в формате файла, готового для загрузки как в систему управления, так и в сторонние системы, например, статистического анализа. За это отвечает модуль CDR. Он формирует во временных файлах в указанном каталоге информацию о проведённых тестах, распознаёт необходимость отсылки в систему управления, завершает накопление и производит регулярную отсылку.

## CDR-HTTP

Если в качестве CDR transport script (SendCmd) указывать строки, начинающиеся с `http://` вместо вызова программы будет установлено соединение с HTTP-сервером (PUT запрос). Если указать HTTP либо `http`, будет использоваться автосоздаваемый URL вида: `http://${CoreIP}/iqm/recvstat35.cgi?USER=${FtpUser}&PASS=${FtpPassword}`

Это полезно для случаев, когда нет возможности исполнять на аппаратной платформе агента пользовательские программы. В остальных случаях рекомендуется для гибкости использовать стандартную программу `/usr/local/iqm_agent/sender.pl`, так как она уже достаточно отработана. Так же, в случае когда невозможно настроить FTP-сервер на системе управления, можно использовать данный модуль для доставки результатов.

Отдельные от уже перечисленных ключи командной строки данным модулем не используются.

## CDR-FTP

Если в качестве CDR transport script (SendCmd) указывать строки, начинающиеся с `ftp://`, то вместо вызова программы будет установлено соединение с FTP-сервером в пассивном режиме. Если указать FTP либо `ftp`, будет использоваться автосоздаваемый URL вида: `ftp://${FtpUser}:${FtpPassword}@${CoreIP}/stat/`

Если в качестве CDR transport script (SendCmd) указывать строки, начинающиеся с `ftpra://`, то вместо вызова программы будет установлено соединение с FTP-сервером в активном режиме. Если указать FTPA либо `ftpra`, будет использоваться автосоздаваемый URL вида: `ftpra://${FtpUser}:${FtpPassword}@${CoreIP}/stat/`

Это полезно для случаев, когда нет возможности исполнять на аппаратной платформе агента пользовательские программы. В остальных случаях рекомендуется для гибкости использовать стандартную программу `/usr/local/iqm_agent/sender.pl`, так как она уже достаточно отработана. Для принудительного активного режима можно использовать стандартную программу `/usr/local/iqm_agent/asender.pl`

Отдельные от уже перечисленных ключи командной строки данным модулем не используются.

## OND

Весьма важной части агента является модуль запуска тестов по требованию администратора. Данный модуль отвечает именно за это. Экономия на его выключении из сборки не является разумной, он весьма полезен. Ключи командной строки данным модулем не используются.

## SSL

В настоящее время очень часто приходится использовать протоколы с поддержкой SSL (например HTTPS). Этот дополнительный модуль, оказывающий влияние на модули SRV-CFG и URL. Рекомендован для использования, если у вас есть такая потребность.

## LIVEMON

Одним из модулей, который используется редко. Его создание было обусловлено желанием пользователей получать реакцию от агента сразу после проведения теста. Для этого предусмотрены поля теста Live alarm expression (ThresholdExpression в FCFG) и URL for live alarm action (ThresholdAction в FCFG). В первом можно использовать простые выражения вида:

```

ПЕРЕМЕННАЯ ОПЕРАТОР ПЕРЕМЕННАЯ
ПЕРЕМЕННАЯ ОПЕРАТОР ЧИСЛО
ЧИСЛО ОПЕРАТОР ПЕРЕМЕННАЯ

```

где

ПЕРЕМЕННАЯ это одна из метрик теста (MinRtt, AvgRtt, MaxRtt, SDLostPercent, DSLostPercent, SDLost, DSLost, SDRemarkedPercent, DSRemarkedPercent, SDRemarked, DSRemarked, SDJitter, DSJitter, ServiceCode, SDBytes, DSBytes, SDBW, DSBW, SDMinDelay, SDAvgDelay, SDMaxDelay, DSMinDelay, DSAvgDelay, DSMaxDelay)

ЧИСЛО — число с фиксированной запятой (1.23456)

Во втором можно использовать URL вида `http://usr:pass@ip/iqm/recvlivemon.cgi?USER=u1&PASS=x1`. Переменные TID, VAR, VAL, STATE добавляются автоматически.

## USOCK

Модуль файлового сокета позволяет проводить управление агентом с помощью дополнительного ПО. Это могут быть как утилиты синхронного тестирования или web-iqm, так и программы третьих фирм, использующие агента для реализации форм тестирования, которые не поддерживаются IQM.

Ключ	Значение по умолчанию	Описание
<code>--usock-path=S</code>	<code>/tmp/iqm-ctl</code>	Путь к сокету связи
<code>--usock-secret=S</code>	Смотрите в информации поставки. Строка может отличаться.	Разделяемая секретная строка для аутентификации.

## HIST

Если включен модуль USOCK, то стандартным методом общения при синхронном тестировании является запрос результатов по итогам теста. Однако, для тех случаев, когда это неудобно, либо пользователь желает накопить некоторое число результатов внутри агента и только потом запросить итоги, можно использовать данный модуль. Число хранящихся записей обычно невелико и определяется на этапе технического задания. Ключи командной строки данным модулем не используются.

## MCOR

Данный модуль позволяет осуществлять отсылку результатов тестирования в несколько точек получения. Допустим, пользователь IQM хочет получать CDR не только в систему управления, но и для собственного разбора в дополнительных модулях управления или статистики. Или система управления используется только для конфигурации агентов, а

отчётность и анализ проводятся в системах третьих фирм.

В случае необходимости можно указать в CoreIP через точку с запятой несколько хостов-получателей. Например так: `host1;host2;host3` или `ip1;ip2;ip3`. При этом возможна, хотя и не рекомендована, ситуация, когда штатная система управления вовсе не получает CDR.

Модуль дополнительно управляется следующими ключами командной строки:

Ключ	Значение по умолчанию	Описание
<code>--mcore-disable</code>	Модуль работает	Отключение модуля
<code>--mcore-success-all</code>	Считать сессию отсылки успешной, если первый хост ответил верно.	Считать сессию отсылки успешной, если все хосты ответили верно.
<code>--mcore-success-any</code>	Считать сессию отсылки успешной, если первый хост ответил верно.	Считать сессию отсылки успешной, если хотя бы один хост ответил верно. При одновременном указании ключа <code>--mcore-success-all</code> он имеет приоритет.

Проверка на правильность отправки предполагает, что штатная система управления (IQMM) используется в списке первой. Пользователь с помощью данных ключей может создавать гибкие отсылки файлов с результатами.

## RSDP

Если включен модуль RSDP, данный дополнительный модуль позволяет исполнять отладочный вывод в журнал о результатах проведённых тестов дополнительно к формированию файлов. Для текущей эксплуатации он не рекомендован, так как вносит дополнительные задержки, однако в отведённой среде тестирования (в лаборатории) допустим к использованию. Модулем обрабатывается следующий ключ:

Ключ	Значение по умолчанию	Описание
<code>--results-debug</code>	Модуль отключен	Выводить по итогам как асинхронного, так и синхронного тестирования, результаты измерений в журнал.

## TCONG

Данный модуль позволяет настроить алгоритмы управления потоков для тех протоколов, который используют TCP. Системные подробности в руководстве ОС Linux `tcp(7)`. На наш взгляд, этот модуль нужен не каждому пользователю, а только тем, кто глубоко разбирается в сетевом стеке. Кроме того, отметим, что отнюдь не в каждой сборке присутствуют альтернативные методы управления потоком. Проверьте, пожалуйста, сборку ядра прежде чем обращаться в нашу поддержку! Модулем обрабатываются ключи:

Ключ	Значение по умолчанию	Описание
<code>--ux-tcp-congestion=STR</code>	Системное внутри ядра ОС, обычно алгоритм Reno.	Использовать данный алгоритм для всех тестов класса Ux (U0 и U1 на данный момент).
<code>--tl-tcp-congestion=STR</code>	Системное внутри ядра ОС, обычно алгоритм Reno.	Использовать данный алгоритм для всех тестов TWAMP (RFC-5357)
<code>--url-tcp-congestion=STR</code>	Системное внутри ядра ОС, обычно алгоритм Reno.	Использовать данный алгоритм для всех тестов типа URL.

## TMS

Данный модуль был разработан по запросу одного из пользователей и фиксирует в журнале результаты текущего состояния процессоров в начале и в конце каждого теста. Это позволяет оценить с помощью стороннего программного обеспечения влияние генератора и приёмника тестов на доступные ресурсы оборудования пробника. Дополнительная информация по запросу. Модулем обрабатываются ключи:

Ключ	Значение по умолчанию	Описание
<code>--times-verbose</code>	Вывод не вести	Выводить в журнал на старте и окончании теста <code>sysconf(_SC_CLK_TCK) times(2) /proc/stat</code>
<code>--clock-verbose</code>	Вывод не вести	Выводить в журнал на старте агента расчётный минимально возможный интервал засыпания в микросекундах.

## TRTY

Если включен модуль USOCK, то данный модуль позволяет получать от `web-iqm` либо от иных команды для запуска программ третьих фирм (`iperf3`, `bwping`) и `iqm-test` для выполнения сравнительного тестирования разными средствами. Для традиционного использования агента модуль не требуется. Дополнительная информация по запросу. Ключи командной строки не обрабатываются.

## G107

Если пользователю нужен расчёт т.н. «синтетического» MOS или R-factor, предусмотренных стандартом G.107 для оценки качества голосового трафика, можно использовать данный модуль. Для использования этого модуля в тесте указывается профиль трафика с указанием кодека и пользовательского ожидания (то, что в англоязычной литературе называется Advantage factor, а в русском переводе переведено кем-то не очень грамотным как «коэффициент выигрыша»). При указании такого профиля агентом будут использоваться следующие настройки:

Кодек	Скорость IPv4	Скорость IPv6	Число пакетов	Размер пакета
G.711	79	86	1000	172
G.729	24	32	1000	32
G.729A	24	32	1000	32
G.726	47	55	1000	92
G.728	26	32	680	72
G.723 MPC	17	22	680	36
G.723 ACELP	16	21	680	32
ILBC	31	39	1000	50

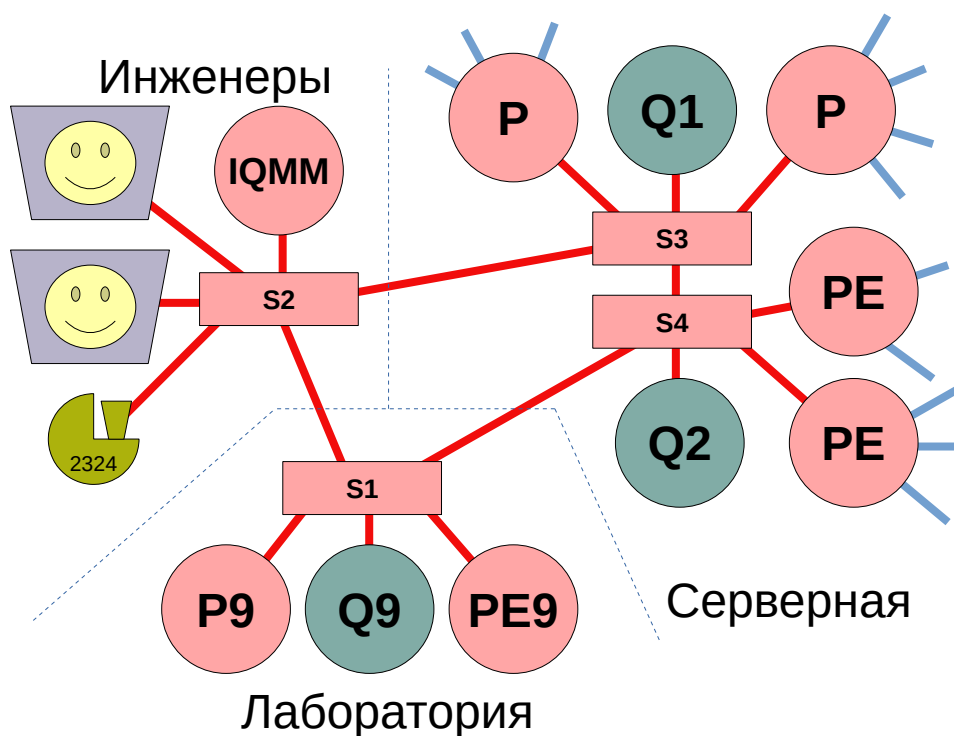
По итогам такого теста дополнительно к традиционным параметрам качества будут доступны MOS и R-factor в обоих направлениях. По итогам тестирования мы рекомендуем для этого модуля использовать только протоколы U0 или U9.

Модулем обрабатываются ключи:

Ключ	Значение по умолчанию	Описание
--g107-verbose	Вывод не вести	Выводить в журнал при расчёте дополнительные параметры, предусмотренные G.107. Это можно использовать для контроля правильности расчёта.

## ССС

Данный модуль был разработан по запросу одного из пользователей и позволяет вести управление агентом при нахождении того за точкой сетевой трансляции NAT или межсетевым экраном так, как будто бы агент находился в обычной сети. Это лучше всего проиллюстрировать на рисунках.



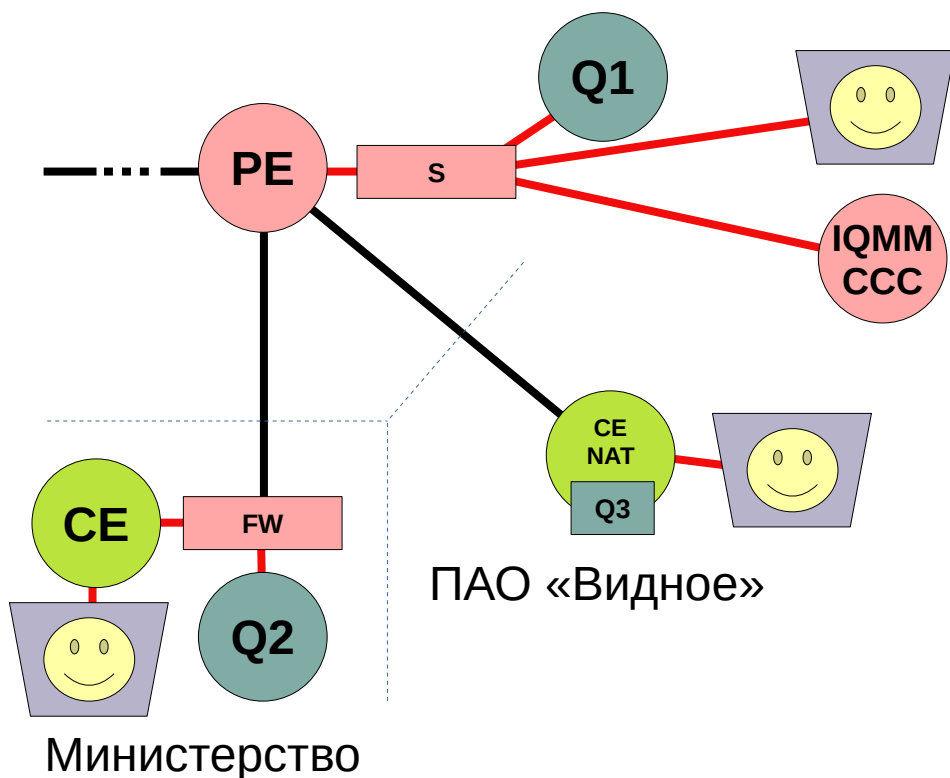
**Рисунок 2.** Агенты включены в сеть без использования сетевой трансляции и межсетевых экранов.

На рисунке 2 видна традиционная схема включения агентов, когда препятствий для пропуска трафика практически нет. Между Q1, Q2, Q9, системой управления IQMM и рабочими местами инженеров трафик проходит совершенно свободно. Рассмотрим подробнее:

Задача трафика	Точки прохода трафика	Сложности
Тестирование по расписанию между Q9 и Q1	TCP SD: Q9 → S1 → S4 → S3 → Q1 TCP DS: Q1 → S3 → S4 → S1 → Q9 UDP SD: Q9 → S1 → S4 → S3 → Q1 UDP DS: Q1 → S3 → S4 → S1 → Q9 TCP DS: Q1 → S3 → S4 → S1 → Q9	Проблемы могут быть только связанные с состоянием сети. Дополнительных ограничений нет.
Доступ от рабочего места инженера к IQMM	TCP SD: PM → S2 → IQMM TCP DS: IQMM → S2 → PM	-"-
Отправка CDR от Q2 к IQMM	TCP SD: Q2 → S4 → S3 → S2 → IQMM TCP DS: IQMM → S2 → S3 → S4 → Q2 data SD: Q2 → S4 → S3 → S2 → IQMM data DS: IQMM → S2 → S3 → S4 → Q2	-"-
Управление по команде инженера от IQMM на Q1 (конфигурация)	TCP SD: PM → S2 → IQMM TCP SD: IQMM → S2 → S3 → Q1 TCP DS: Q1 → S3 → S2 → IQMM TCP DS: IQMM → S2 → PM (реальное состояние стека несколько упрощено для лучшего понимания)	-"-

Задача трафика	Точки прохода трафика	Сложности
Управление по команде инженера от IQMM на Q2 (запуск теста по требованию на Q9)	TCP SD: PM → S2 → IQMM TCP SD: IQMM → S2 → S3 → S4 → Q2 TCP SD: Q2 → S4 → S1 → Q9 TCP DS: Q9 → S1 → S4 → Q2 TCP DS: Q2 → S4 → S3 → S2 → IQMM TCP DS: IQMM → S2 → PM UDP SD: Q2 → S4 → S1 → Q9 UDP DS: Q9 → S1 → S4 → Q2 TCP DS: Q9 → S1 → S4 → Q2 (так же есть некоторое упрощение)	-”-
Управление по команде инженера от IQMM на Q2 (чтение результатов теста по требованию)	TCP SD: PM → S2 → IQMM TCP SD: IQMM → S2 → S3 → S4 → Q2 TCP DS: Q2 → S4 → S3 → S2 → IQMM TCP DS: IQMM → S2 → PM	-”-

Теперь рассмотрим более приближенную к текущей реальности схему. Оператор связи в настоящее время, как правило, включает клиентов, находящихся либо за МСЭ, либо за точкой трансляции адресов (NAT). Примерная возможная схема показана на рисунке 3.



**Рисунок 3.** Агенты включены в сеть с использованием сетевой трансляции и межсетевых экранов.

У каждого клиента есть сетевое оборудование, рабочее место и агент тестирования сетевых характеристик в том или ином виде (АПК или встроенный в оборудование).

Вначале рассмотрим обычные случаи, без использования модуля CCC.

Задача трафика	Точки прохода трафика	Сложности
Доступ от рабочего места инженера министерства к IQMM	TCP SD: PM → CE → FW → PE → S → IQMM TCP DS: IQMM → S → PE → FW → CE → PM	Точка FW может как запрещать, так и разрешать трафик как на вход, так и на выход. Наиболее часто трафик на выход разрешён, трафик на вход запрещён. Поскольку инициатором соединения является инженер, <u>трафик разрешается</u> . Для обратного трафика от IQMM он разрешён только когда он запрошен (состояние connected).
Доступ от рабочего места инженера ПАО к IQMM	TCP SD: PM → CE <sub>nat</sub> → PE → S → IQMM TCP DS: IQMM → S → PE → CE <sub>nat</sub> → PM	Точка CE <sub>nat</sub> чаще всего настроена так, что трафик на выход разрешён, трафик на вход запрещён, дополнительно сетевая трансляция. Поскольку инициатором соединения является инженер, <u>трафик разрешается</u> . Для обратного трафика от IQMM он разрешён только когда он запрошен (состояние connected) плюс транслируется адрес. При переполнении таблицы соответствий возможны сложности.

Задача трафика	Точки прохода трафика	Сложности
Тестирование по расписанию протоколом U0 между Q2 и Q1	TCP SD: Q2 → FW → PE → S → Q1 TCP DS: Q1 → S → PE → FW → Q2 UDP SD: Q2 → FW → PE → S → Q1 UDP DS: Q1 → S → PE → FW → Q2	Точка FW может запретить исходящее соединение по несогласованному порту, но чаще <u>разрешает</u> . Обратное соединение TCP чаще всего <u>проблем не имеет</u> . Исходящий UDP-трафик так же <u>разрешён</u> . А вот входящий UDP-трафик с 99% вероятностью будет <u>запрещён</u> , так как о типе контрольного протокола тестирования FW не имеет представления. В результате тестирование будет либо <u>не выполнено</u> , либо выполнено в одну сторону.
Отправка CDR от Q2 к IQMM	TCP SD: Q2 → FW → PE → S → IQMM TCP DS: IQMM → S → PE → FW → Q2 data SD: Q2 → FW → PE → S → IQMM data DS: IQMM → S → PE → FW → Q2	Точка FW может запретить исходящее соединение по порту ftp, но чаще <u>разрешает</u> . Соединение ftp-data чаще всего <u>проблем не имеет</u> за счёт использования по умолчанию passive-режима, что в свою очередь <u>позволяет</u> пройти файлам данных.
Управление по команде инженера от IQMM на Q2 (конфигурация)	TCP SD: PM → CE → FW → PE → S → IQMM TCP SD: IQMM → S → PE → FW → Q2 TCP DS: Q2 → FW → PE → S → IQMM TCP DS: IQMM → S → PE → FW → CE → PM	Точка FW разрешит доступ к IQMM, тот попытается подсоединиться к Q2. Однако в традиционном МСЭ нормальная политика — не пропускать входящие соединения без специальной настройки. Таким образом, конфигурация <u>будет запрещена</u> уже на втором этапе. Вопросы маршрутизации опускаем.

Задача трафика	Точки прохода трафика	Сложности
Управление в режиме REMOTE_IQM по заранее настроенному расписанию от IQMM на Q2 (конфигурация)	IQMM создаёт файл по команде инженера. TCP SD: Q2 → FW → PE → S → IQMM TCP DS: IQMM → S → PE → FW → Q2 data SD: Q2 → FW → PE → S → IQMM data DS: IQMM → S → PE → FW → Q2	Файл, созданный на IQMM заранее, используется в том же режиме, что и загрузка CDR. Это <u>позволяет</u> пройти конфигурации как файлу данных. Однако требует дополнительных настроек на агенте.
Тестирование по расписанию протоколом U1 между Q2 и Q1	TCP SD: Q2 → FW → PE → S → Q1 TCP DS: Q1 → S → PE → FW → Q2 UDP SD: Q2 → FW → PE → S → Q1 UDP2 SD: Q2 → FW → PE → S → Q1 UDP2 DS: Q1 → S → PE → FW → Q2 UDP DS: Q1 → S → PE → FW → Q2	Точка FW может запретить исходящее соединение по несогласованному порту, но чаще <u>разрешает</u> . Обратное соединение TCP чаще всего <u>проблем не имеет</u> . Исходящий UDP-трафик так же <u>разрешён</u> . За счёт второго потока UDP-трафика происходит «фиксация» порта для входящего трафика. Поток UDP DS с высокой степенью вероятности будет пропущен МСЭ, если второй поток получил ответ. Итого тест <u>успешен</u> .
Управление по команде инженера от IQMM на Q2 (запуск теста по требованию на Q1)	TCP SD: PM → CE → FW → PE → S → IQMM TCP SD: IQMM → S → PE → FW → Q2 TCP DS: Q2 → FW → PE → S → IQMM TCP DS: IQMM → S → PE → FW → CE → PM ... остальные этапы опускаем, до них работа не дойдёт.	Точка FW разрешит доступ к IQMM, тот попытается подсоединиться к Q2. Однако в традиционном МСЭ нормальная политика — не пропускать входящие соединения без специальной настройки. Таким образом, конфигурация <u>будет запрещена</u> уже на втором этапе.
Управление по команде инженера от IQMM на Q2 (чтение результатов теста по требованию)	Случай аналогичен предыдущему	—”—

Таким образом, мы видим, что без использования модуля CCC с помощью дополнительных средств решается проблема регулярной конфигурации агентов за NAT и FW. Так же, при использовании протоколов U1 или U9 решается проблема тестирования из-за NAT в сеть, где маршрутизация традиционна. Но без сложных дополнительных настроек (и возможно, согласований в СИБ!) трудно решить проблему доступа к тестам по требованию через web-интерфейс IQMM. Тем более, если площадка нам не принадлежит, а тестировать качественные характеристики всё-таки надо. Можно, конечно, применить паллиативы, например, синхронное тестирование утилитой **iqm-test**, но с её помощью обычно работают профессиональные администраторы, а нам требуется более упрощённое решение.

Поэтому и выбран следующий путь: на IQMM устанавливается демон **cccd** (описание доступно отдельно). Агент с модулем CCC имеет возможность автоматически проводить соединение с указанным демоном от запуска до конца работы, равно как и web-интерфейс IQMM с другой стороны так же осуществляет такое соединение. В результате появляется контролируемый демоном виртуальный канал, аналогичный каналу, устанавливаемому с модулем CTRL и позволяющий как проводить конфигурацию агента, так и выполнять тесты по требованию. Настройки безопасности указанного соединения обсуждаются отдельно.

Теперь подробно рассмотрим случай, когда в агенте работает модуль CCC.

Задача трафика	Точки прохода трафика	Сложности
Управление по команде инженера от IQMM на Q2 (конфигурация агента типа CIQM)	TCP SD CCC: Q2 → FW → PE → S → IQMM TCP SD: PM → CE → FW → PE → S → IQMM LOCAL SD: IQMM → CCCD OVER CCC DS: CCCD → S → PE → FW → Q2 OVER CCC SD: Q2 → FW → PE → S → CCCD LOCAL DS: CCCD → IQMM TCP DS: IQMM → S → PE → FW → CE → PM TCP SD CCC (автоматически): Q2 → FW → PE → S → IQMM	Точка FW разрешит доступ к IQMM, как для CCC, так и для инженера (если мы исходим из прошлых предположений о настройках). По уже установленному соединению с <b>cccd</b> IQMM может дать команды конфигурации агента Q2. Агент ответит на IQMM о корректности или ошибках по этому же каналу. Информацию IQMM сообщит инженеру итоги конфигурации. Итого, конфигурация традиционным методом <u>возможна</u> . Настройки REMOTE_IQM не нужны.

Задача трафика	Точки прохода трафика	Сложности
<p>Управление по команде инженера от IQMM на Q2 (запуск теста U1 по требованию на Q1)</p>	<p>TCP SD CCC: Q2 → FW → PE → S → IQMM TCP SD: PM → CE → FW → PE → S → IQMM LOCAL SD: IQMM → CCCD OVER CCC DS: CCCD → S → PE → FW → Q2 OVER CCC SD: Q2 → FW → PE → S → CCCD LOCAL DS: CCCD → IQMM TCP DS: IQMM → S → PE → FW → CE → PM TCP SD CCC (автоматически): Q2 → FW → PE → S → IQMM TCP SD: Q2 → FW → PE → S → Q1 TCP DS: Q1 → S → PE → FW → Q2 UDP SD: Q2 → FW → PE → S → Q1 UDP2 SD: Q2 → FW → PE → S → Q1 UDP2 DS: Q1 → S → PE → FW → Q2 UDP DS: Q1 → S → PE → FW → Q2 TCP DS: Q1 → S → PE → FW → Q2</p>	<p>Точка FW разрешит доступ к IQMM, как для CCC, так и для инженера (если мы исходим из прошлых предположений о настройках). По уже установленному соединению с <b>cccd</b> IQMM может дать команды конфигурации агента Q2. Агент ответит на IQMM о корректности или ошибках по этому же каналу. Информацию IQMM сообщит инженеру итоги запуска теста. Точка FW может запретить исходящее соединение по несогласованному порту, но чаще <u>разрешает</u>. Обратное соединение TCP чаще всего <u>проблем не имеет</u>. Исходящий UDP-трафик так же <u>разрешён</u>. За счёт второго потока UDP-трафика происходит «фиксация» порта для входящего трафика. Поток UDP DS с высокой степенью вероятности будет пропущен МСЭ. Итого тест <u>успешен</u>.</p>
<p>Управление по команде инженера от IQMM на Q2 (чтение результатов теста по требованию)</p>	<p>TCP SD CCC: Q2 → FW → PE → S → IQMM TCP SD: PM → CE → FW → PE → S → IQMM LOCAL SD: IQMM → CCCD OVER CCC DS: CCCD → S → PE → FW → Q2 OVER CCC SD: Q2 → FW → PE → S → CCCD LOCAL DS: CCCD → IQMM TCP DS: IQMM → S → PE → FW → CE → PM TCP SD CCC (автоматически): Q2 → FW → PE → S → IQMM</p>	<p>Точка FW разрешит доступ к IQMM, как для CCC, так и для инженера (если мы исходим из прошлых предположений о настройках). По уже установленному соединению с <b>cccd</b> IQMM может дать команды конфигурации агента Q2. Агент ответит на IQMM о корректности или ошибках по этому же каналу. Информацию IQMM сообщит инженеру итоги тестирования. Итого <u>успешно</u></p>

Что касается площадки за NAT, то её можно рассматривать как упрощённый случай МСЭ, когда исходящие с площадки соединения гарантированно пропускаются с исправлением адресов, а входящие, если они не предназначены для установленного соединения гарантированно отрасываются. Поэтому модуль ССС и демон-прокси **cccd** будут работать аналогично, настроек у NAT даже меньше, чем у МСЭ.

Модулем ССС обрабатываются ключи:

Ключ	Значение по умолчанию	Описание
<code>--ccc-disable</code>	Модуль включен	Отключить модуль. В этом случае агент работает традиционным методом.
<code>--ccc-core-ip=HOST</code>	Использовать CoreIP из текущей конфигурации	Для соединения с прокси использовать указанный хост в виде имени или IP-адреса.

## 8 Сигналы

Агент в процессе работы обрабатывает стандартные сигналы. Отдельные сигналы обрабатываются специальным образом:

**SIGHUP** - при наличии модуля FCFG перечитывается конфигурация агента и тестов из файла, при отсутствии — из стартовой командной строки.

**SIGINT** - завершить исполнение, удалить файл с идентификатором процесса.

**SIGQUIT** - завершить исполнение, сохранить файл с идентификатором процесса.

**SIGPIPE** - игнорируется.

**SIGALRM** - прервать ввод/вывод, выйти из системного вызова.

**SIGTERM** - завершить исполнение, удалить файл с идентификатором процесса.

**SIGUSR1** - при наличии модуля SRV-CFG принудительно перечитать конфигурацию агента и тестов с системы управления, при отсутствии — игнорируется.

**SIGUSR2** - при наличии модуля SCDP по договору техподдержки создать в журнале записи о текущем состоянии планировщика, при отсутствии — игнорируется.

**SIGCHLD** - обработать коды возврата дочерних процессов, если они сохранили группу процесса.

Остальные сигналы используют настройки ОС по умолчанию.

## Содержание

1 Введение.....	2
2 IQM-агент.....	3
3 Стандартные ключи командной строки.....	4
4 Ручной запуск агента.....	5
5 Права процесса.....	6
6 Модули агента.....	6
7 Ключи командной строки модулей.....	12
8 Сигналы.....	34

Настоящим подтверждается, что все исключительные авторские права на данную документацию принадлежат ООО «НетПроб». Предоставление прав на данную документацию осуществляется по лицензионному договору присоединения, ссылки на юридический текст которого указаны в данном тексте. Неотчуждаемые личные неимущественные права на данную документацию принадлежат физическим лицам – авторам, перечисленным в документации. Настоящим подтверждается, что все права на использованные системные и стандартные модули программного обеспечения принадлежат их авторам и используются правомерно в соответствии с предоставленными авторами лицензионными договорами, в том числе, но не ограничиваясь, GNU General Public License, Artistic License и т.д.

Copyright © 2008-2025



ООО «Нетпроб»

Copyright © 2010-2025



Сергей Александрович Еременко