Совет пользователям: **фундаментальные принципы Ethernet.**

Фундаментальные принципы Ethernet.

I. Различные значения/адреса, используемые для связи по Ethernet 3

II. Обзор модели уровня OSI 7 3

III. Соединение MAC-адреса идентификатора ID 4

A. Что такое идентификатор (ID) MAC-адреса 4

B. Роль MAC-адреса ID с коммутаторами и концентраторами Ethernet 5

C. Как MAC-адрес получен из IP-адреса (ARP) 6

IV. IP-адресация 6

A. Как IP-адреса работают с масками подсети 10

B. IP-адрес используется для маршрутизации сообщений по правильному «пути» 11

C. Адрес шлюза используется для доступа к маршрутизатору для пакетов, предназначенных для другой подсети 12

V. TCP против UDP 14

VI. DNS система доменных имён 16

I. Различные значения/адреса, используемые для связи по Ethernet.

Существует три основных значения или адреса, которые обсудим в этом документе, чтобы объяснить, как сообщения по Ethernet поступают от одного устройства к другому:

- Идентификатор MAC-адреса
- IP-адрес (вместе с маской подсети и адресом шлюза по умолчанию)
- Номер порта UDP/TCP

Это не означает, что это единственные важные моменты, используемые в сети Ethernet, но они играют ключевую роль в попытке понять, как устройства маршрутизируется по сети.

II. Обзор 7 уровней модели OSI

Важно понять модель 7 уровней OSI по которой работает Ethernet, чтобы понять, как Ethernet-пакеты перемещаются по типичной сети. Многие устройства, через которые проходит Ethernet-пакет, определяют, куда отправлять пакет на основе одного или нескольких слоев Ethernet. Многие коммутаторы и маршрутизаторы Ethernet сети, классифицируются, как устройства уровня L2 или L3.

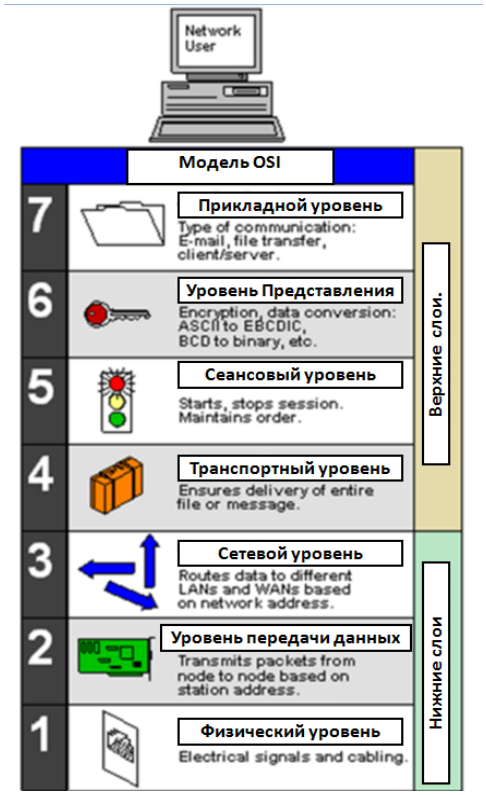
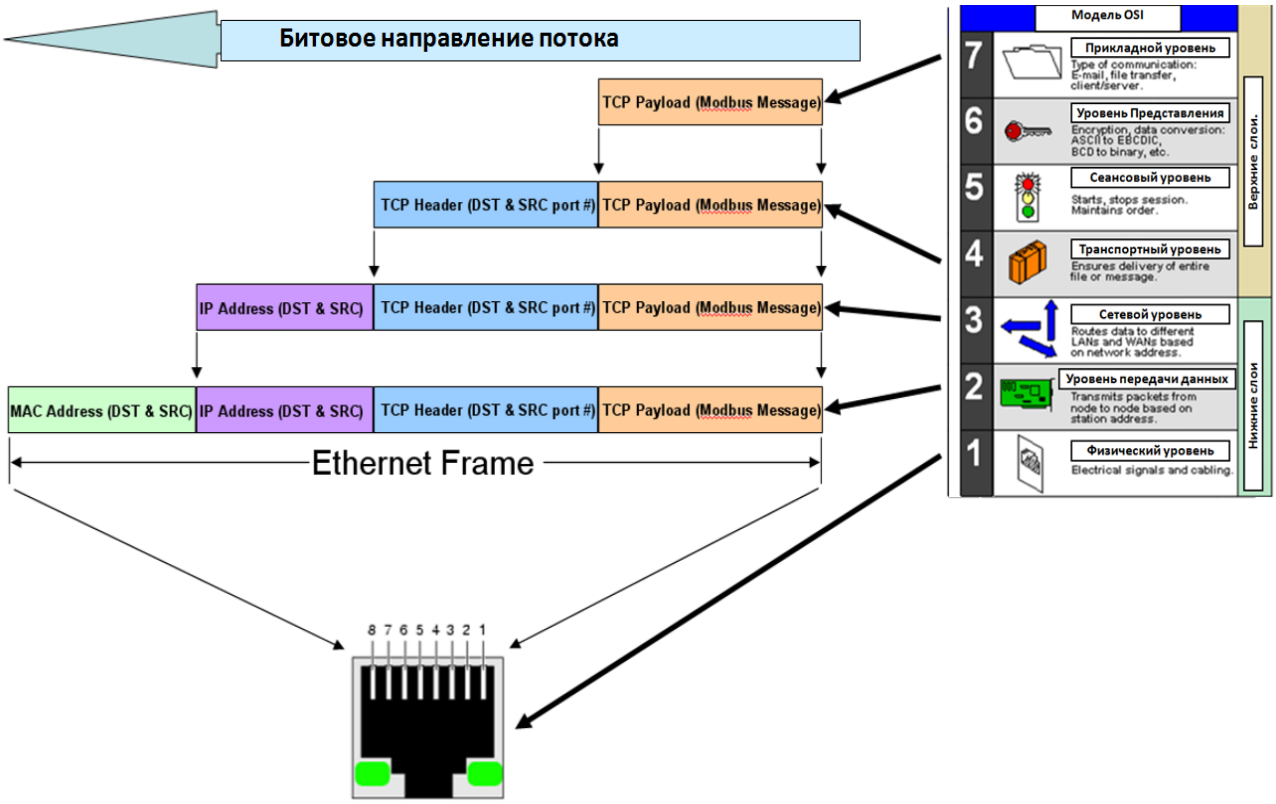


Диаграмма выше - хорошо объясняет разные уровни в модели OSI. Важно отметить, что не все Ethernet сообщения будут включать все эти уровни. Большая часть, по сути, будет содержать только уровни L4 или L5 . На приведенном ниже рисунке показана структура сообщения Modbus TCP и его соответствие модели OSI.



Когда в приложении создается сообщение Ethernet, обычно первым шагом является формирование сообщения приложения, которое в показанном выше случае будет сообщением Modbus. Сообщение приложения затем будет отправлено на то, что оно обычно называется «стек». «Стек» - это обычно используемое имя для кода, который существует в Операционной системе большинства устройств и которые фактически заботятся о формировании пакета TCP/UDP, а затем добавляя надлежащий IP-заголовок, заголовок MAC-адреса и отправляя весь этот Ethernet-кадр (**frame**) физическому чипу (PHY), который фактически посылает электрические сигналы на провод. Как видно из изображения выше, «уровень L2» соответствует уровню MAC-адреса, а «уровень L3» соответствует уровню IP-адреса.

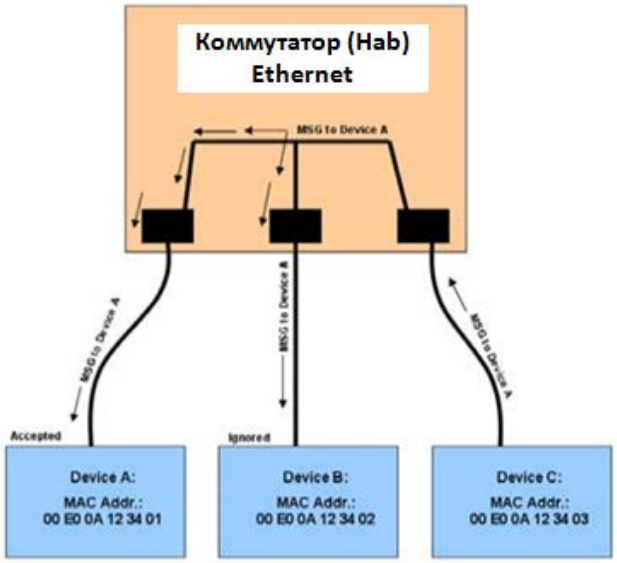
III. Соединение по MAC-адресу с идентификатором (ID)

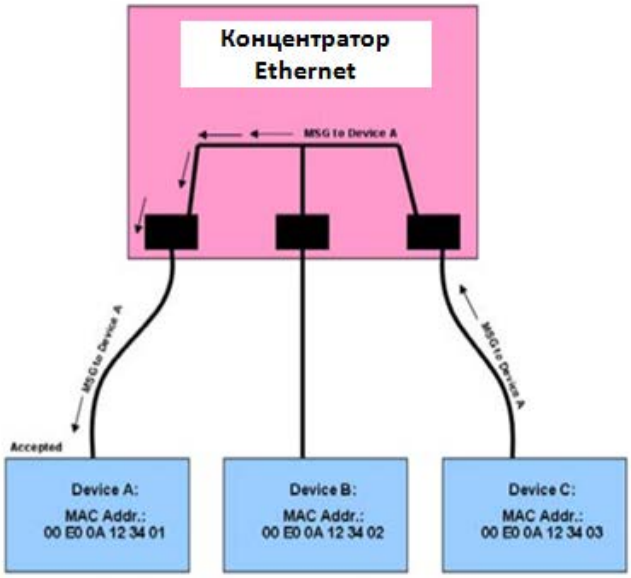
A. Что такое идентификатор MAC-адреса?

Значение идентификатора (ID) MAC-адреса - физически аппаратный адрес устройства. Это первое значение, которое приходит в порт Ethernet. Это значение обычно не настраивается в отличие от IP-адреса или номера порта.

B. Роль идентификатора MAC-адреса в коммутаторах (switches) и концентраторах (hubs) Ethernet.

Стандартные неуправляемые коммутаторы используют, что называется, переключение “уровня L2". Другими словами, коммутатор использует уровень канала передачи данных, который содержит MAC-адрес, чтобы решить, какой физический порт должен отправить пакет, который он получил. Таким образом, неуправляемый коммутатор должен только читать первые 6 байт Ethernet-пакета, чтобы решить, куда его отправить. Это предотвращает ненужные коллизии, возникающие в концентраторе (hub) Ethernet. На приведенных ниже двух диаграммах показана разница между концентратором Ethernet и коммутатором Ethernet. Коммутатор использует MAC-адрес, чтобы разумно решить, какие порты отправят сообщения по сравнению с концентратором, который в свою очередь всегда отправляет каждое сообщение, полученное из порта, во все его порты.





C. Как MAC-адрес получается из IP-адреса (ARP)

Несмотря на то, что значением, необходимым для связи с устройством, является MAC-адрес, пользователь почти никогда не указывает MAC-адрес при отправке сообщения на данное устройство. Обычно IP-адрес нужного объекта указывается при отправке сообщения на другое устройство в приложении. Так как же передающее устройство знает, что такое MAC-адрес, если только указан IP-адрес? Ответ будет - через **ARP** (**Address Resolution Protocol** — протокол разрешения адресов). Всякий раз, когда отправляющее устройство хочет отправить сообщение другому устройству, оно отправит широковещательное сообщение (**MAC-адрес = FF FF FF FF FF FF** и **IP-адрес = FF.FF.FF.FF**), используя протокол ARP, запрашивающий MAC-адрес указанного IP-адреса. Устройство, у которого есть указанный IP-адрес, ответит своим MAC-адресом, чтобы отправляющее устройство теперь могло сформировать надлежащее сообщение с MAC-адресом назначения.

Большинство устройств обычно содержат таблицу ARP, которая содержит предопределенный список MAC-адресов для IP-адресов, чтобы уменьшить количество трафика в сети. Вместо отправки ARP-сообщения каждый раз, когда отправляющему устройству необходимо отправить сообщение, он сначала проверит свою таблицу ARP, чтобы увидеть, существует ли эта запись. Другим широко используемым методом является то, что устройство отправляет сообщение «**безвозмездное ARP**» («gratuitous ARP») при его включении или подключении к сети, которая объявляет свой IP-адрес и MAC-адрес, и любое прослушивание устройства может затем добавить эту запись в свою таблицу ARP.

IV. IP- адресация

Следующий очевидный **вопрос:** если нужно использовать MAC-адрес для связи с устройством, почему нужно беспокоиться о IP-адресе? **Ответ**: для сетевой маршрутизации. IP-адрес - это не только номер устройства, (также называемый «Адресом узла» или «Host Address»), но также содержит номер сети **network number** (см. изображение ниже).

A. Как IP-адреса работают с масками подсети

Наиболее распространенная причина, по которой 2 устройства IP (UDP или TCP) не могут взаимодействовать друг с другом, заключается в том, что IP-адреса и маски подсети не настроены должным образом.

IP-адрес и маски подсети используются вместе для создания «Сетевого адреса» и «Адрес узла», как указано выше. Это позволяет в большинстве случаев увеличить количество сетей и больше устройств на сеть и сократить трафик на сеть. IP-адреса и маски подсети обычно отображаются в десятичной системе:

- IP-адрес: 192.168.10.1
- Маска подсети: 255.255.255.0

Чтобы лучше понять, как эти два адреса используются вместе, их необходимо просмотреть в двоичной форме:

- IP-адрес: 11000000.10101000.00001010.00000001
- Маска подсети: 11111111.11111111.11111111.00000000

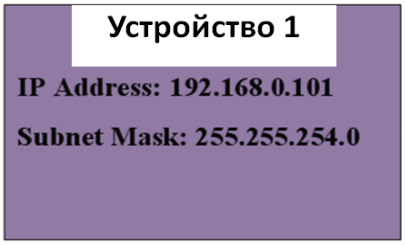
Где бы ни находилась 1 (единица) в маске подсети, это используемый сетевой адрес (**Network Address**). Где есть 0, это указывает адрес хоста **(Host Address):**

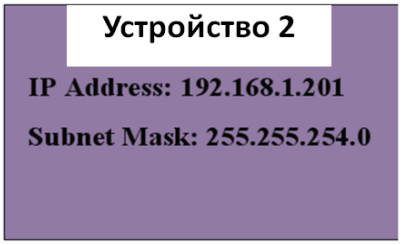
- IP-адрес: 11000000.10101000.00001010.00000001
- Маска подсети: 11111111.11111111.11111111.00000000

|← Network Address (сетевой адрес) →|← Host →|
(адрес хоста)

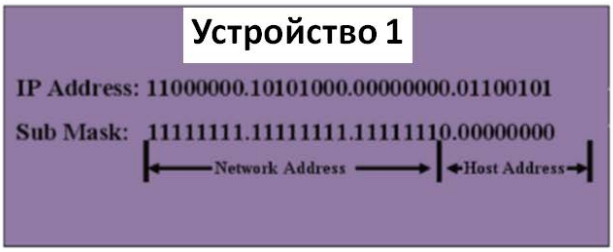
- Network Address (сетевой адрес)= 192.168.10.0
- Host Address (адрес хоста) = 0.0.0.1

Ключевыми моментами для того, чтобы 2 устройства могли обмениваться данными друг с другом, являются: их сетевые адреса (**Network Addresses**) должны быть одинаковыми, а их адреса хостов (**Host Addresse**) должны быть разными.





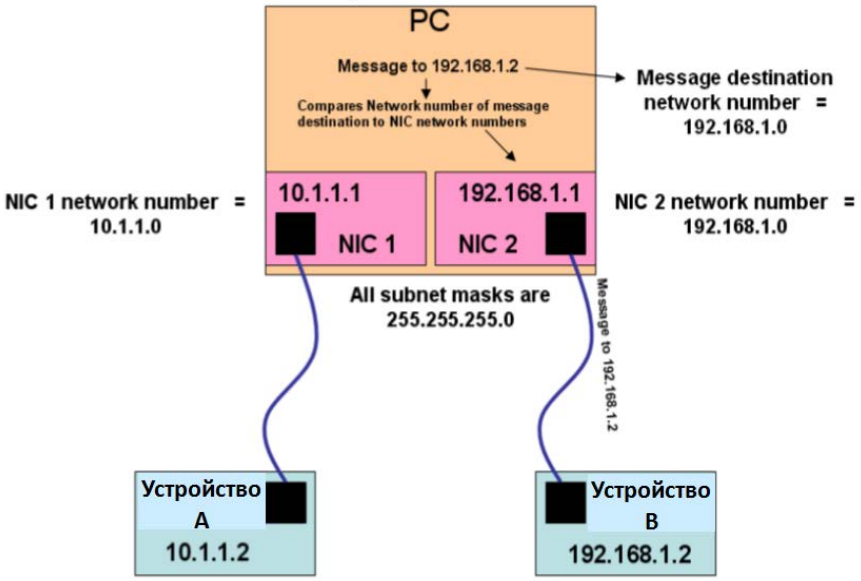
Хотя может показаться, что эти 2 устройства, показанные выше, не могут разговаривать друг с другом, они фактически находятся в одной сети и могут общаться друг с другом. Это более отчетливо видно, если смотреть на адреса в двоичной форме (показано ниже). Обратите внимание, что они имеют один и тот же сетевой адрес, но разные адреса хостов (**Host addresses**).



Если одно из устройств, показанных выше, было настроено с маской подсети 255.255.255.0, они тогда будут в разных сетях и не смогут общаться друг с другом.

B. IP-адрес используется для маршрутизации сообщений на правильный «путь».

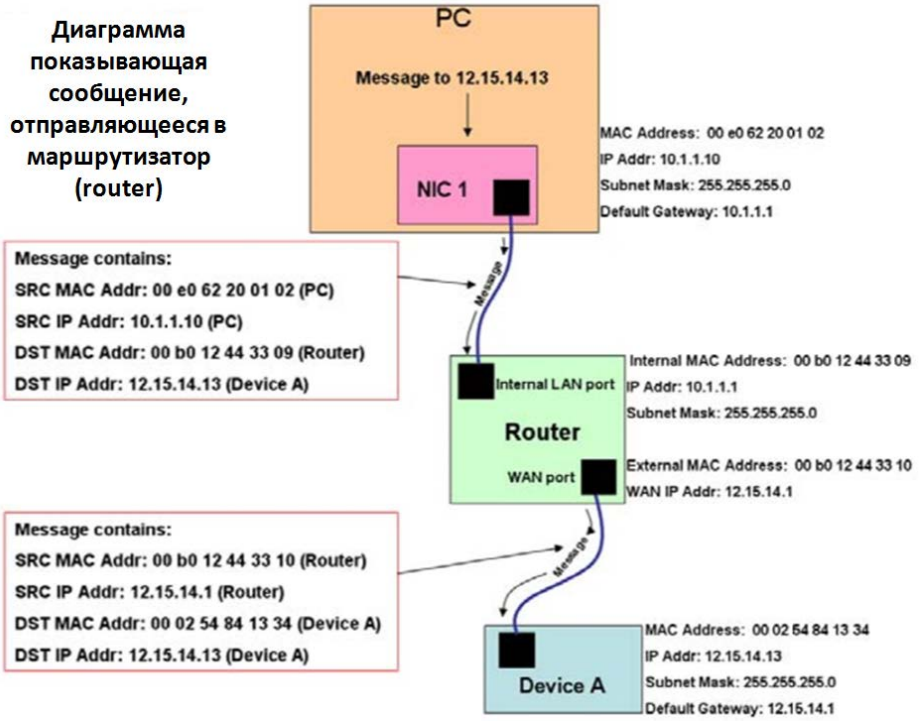
Устройство может использовать IP-адрес (совместно с маской подсети) несколькими способами, чтобы решить, каким способом отправить сообщение. Очень распространенная ситуация, когда ПК необходимо расшифровать сетевой адрес от IP и маски подсети для , когда установлено несколько сетевых интерфейсных карт.



Итак, в приведенном выше примере на ПК есть сообщение, которое необходимо отправить на устройство B. Он должен решить, какую сетевую карту интерфейса (NIC) отправить сообщение. PC просматривает номер сети для сообщения и 2 сетевых адаптера, обнаруживает, что сетевой номер соответствует NIC 2, поэтому решает отправить сообщение из этого порта. Это довольно простая концепция. Следующий вопрос: что, если номер сети назначения сообщений не соответствует ни одному из сетевых номеров сетевых адаптеров? Это подводит нас к следующей теме адресации шлюза (**Gateway**).

C. Адрес шлюза (Gateway) используется для доступа к маршрутизатору (Router) для пакетов, предназначенных для другой подсети

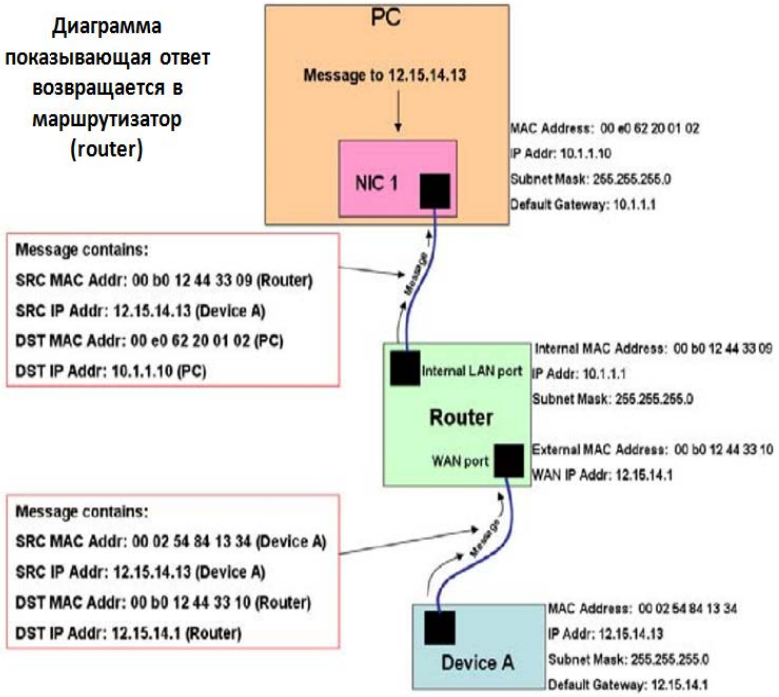
Адрес шлюза по умолчанию обычно является внутренним IP-адресом локальной стороны маршрутизатора. Интересно, что IP-адрес шлюза по умолчанию фактически не отображается в сообщении, отправляемом маршрутизатору. Устройство фактически использует IP-адрес шлюза по умолчанию для разрешения MAC-адреса маршрутизатора, а MAC-адрес маршрутизатора используется в качестве MAC-адреса назначения в отправляемом сообщении. Иллюстрация этого показана ниже:



Следуя приведенной выше схеме, мы рассмотрим шаги устройства, отправляющего сообщение Ethernet через маршрутизатор.

1. PC отправляет сообщение на устройство A. Поскольку PC и устройство A находятся в разных подсетях, сообщение ПК использует MAC-адрес маршрутизатора в качестве MAC-адреса получателя.

2. Маршрутизатор (router) принимает сообщение и изменяет MAC-адрес источника и IP-адрес источника на свой собственный MAC-адрес и IP-адрес. Он использует MAC-адрес и IP-адрес устройства A для своих адресов



Ниже приведены шаги для ответного сообщения в маршрутизаторе:

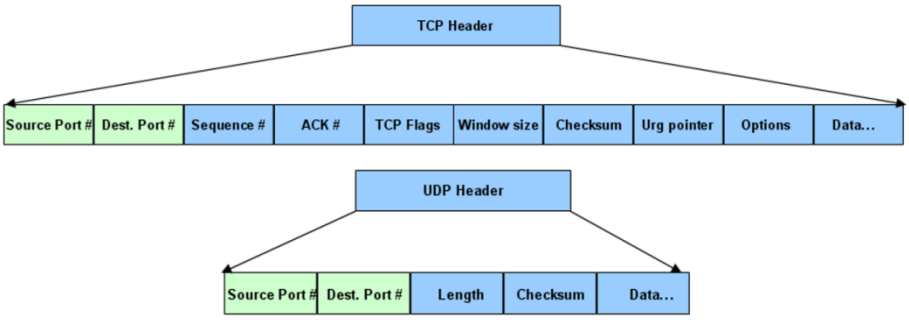
1. Устройство А отправляет ответ обратно на MAC-адреса и IP-адреса маршрутизаторов (router) с использованием собственных MAC-адресов и IP-адресов в качестве исходных адресов.

2. Маршрутизатор (router) изменяет MAC-адреса назначения и IP-адреса на MAC и IP-адреса ПК. Маршрутизатор изменяет MAC-адрес источника на внутренний MAC-адрес маршрутизатора, но использует исходный IP-адрес устройства А.

V. Номера портов TCP и UDP.

Номера портов TCP и UDP используются приложениями или процессами при отправке и приеме пакетов Ethernet.

A. Номера портов используются для управления приложениями и подключениями



Как видно из приведенных выше изображений, номера портов TCP и UDP являются первыми двумя полями в заголовке TCP и UDP. Номер исходного порта - 16 бит, а номер порта назначения - 16 бит.

Когда клиент отправляет Ethernet-пакет, номер целевого порта в его сообщении должен соответствовать номеру «прослушивающего» порта в устройстве, на которое отправляется сообщение. Обычно это приложения, которые выполняются на устройстве, которое «прослушивает» конкретный номер порта.

Например: если на устройстве запущено приложение веб-сервера, он чаще всего прослушивает порт 80. Порт 80 является стандартным или стандартным номером порта для HTTP. Если другой компьютер открывает веб-браузер (например, Internet Explorer), он обычно отправляет сообщение с портом назначения 80.

Номер исходного порта обычно менее важен как конкретное значение. Как правило, клиентское устройство выбирает номер порта источника из пула чисел, называемый диапазоном номеров портов «Эфемерный».

Существует 3 типа и/или диапазоны номеров портов, указанных в **IANA** (Internet Assigned Numbers Authority - Управление полномочиями в Интернете):

- «Хорошо известные порты» = 0 - 1023

- «Зарегистрированные порты» = 1024 - 49151

- «Динамические/частные порты» (обычно используется как эфемерные - Ephemeral) = 49152 - 65535

Хорошо известные порты - это порты, которые очень часто используются, такие как HTTP: 80, FTP: 21, SMTP: 25, Modbus TCP: 502 и т. д. Номера портов в этом диапазоне должны использоваться только приложением, если они зарегистрированы с помощью IANA.

Зарегистрированные порты могут также использоваться в качестве прослушивающих портов приложениями и очень часто используются для небольших приложений и процессов, которые не чувствуют необходимости регистрироваться в IANA.

И, как упоминалось выше, **динамические порты** должны использоваться как порт Ephemeral для клиентов, отправляющих сообщения TCP и UDP.

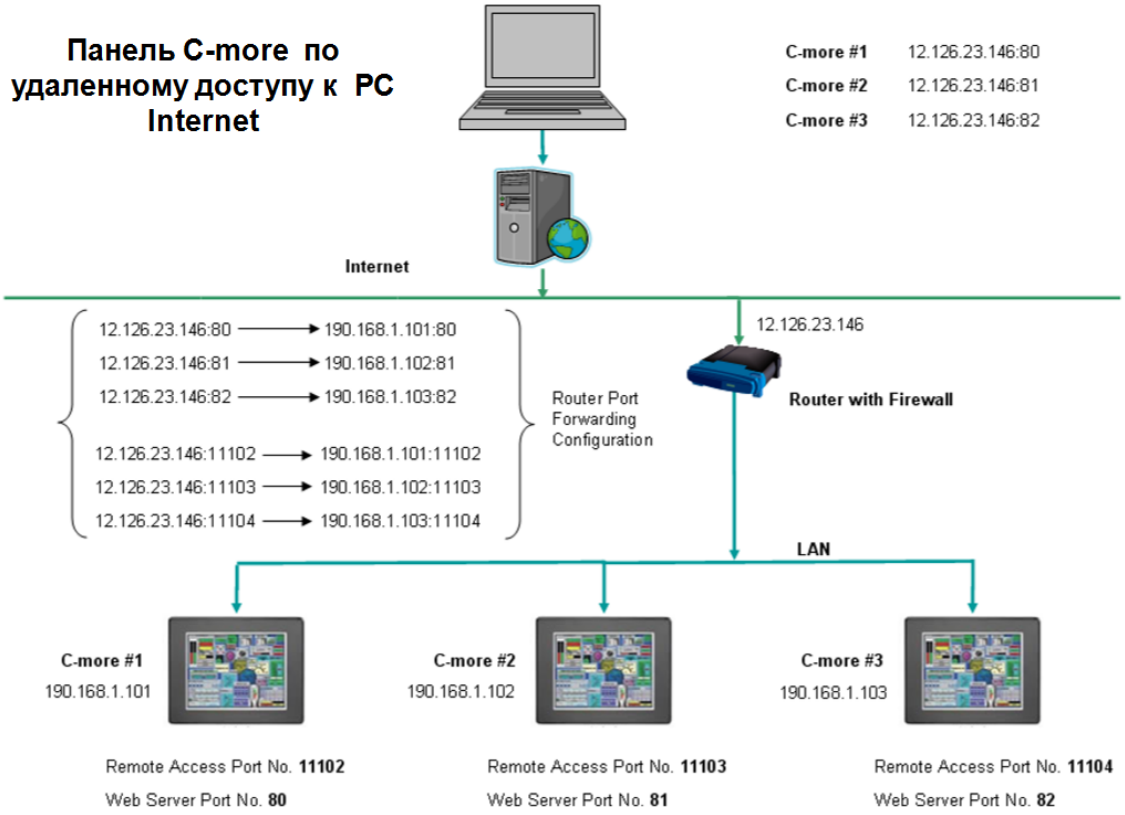
Номера портов также используются приложениями для «привязки» к сокету, который является внутренней структурой программного обеспечения. Это помогает поддерживать надежную связь между несколькими устройствами.

B. Номера портов используются для доступа и направления брандмауэра (Firewall).

Брандмауэры используют номер порта назначения сообщения Ethernet, ориентированного на маршрутизатор, чтобы определить, принимать или отклонять сообщение.

По умолчанию большинство брандмауэров будут иметь некоторые из известных портов Well Known, таких как HTTP (80) и SMTP (25), но большинство остальных нужно будет добавить в таблицу, обычно называемую таблицей «Переадресация портов» (Port Forwarding).

Таблица переадресации портов не только сообщает брандмауэру, какие номера портов разрешены внутри маршрутизатора, но также указывает маршрутизатору, на котором внутреннее устройство (IP-адрес) направляет сообщения, поступающие с указанными номерами портов. Таким образом, при использовании этой методологии тысячи внутренних устройств с частными IP-адресами могут получать сообщения только через один публичный IP-адрес.

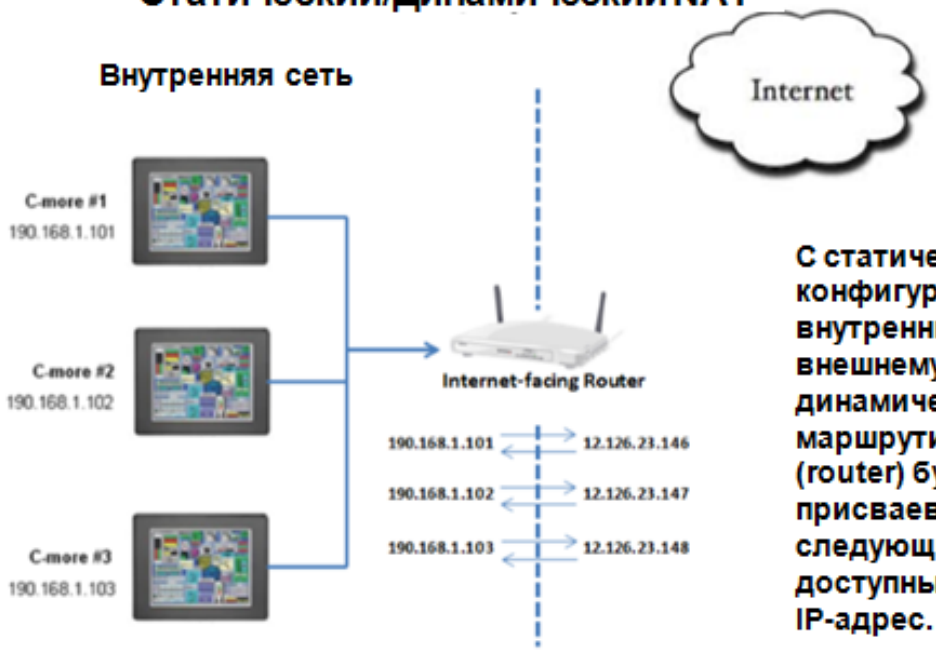


На изображении выше (взято из файла справки **C-more Remote Access**) была настроена таблица перенаправления портов маршрутизатора (Router's) для портов 80 - 82 (для web сервера панели C-more) и портов 11102 - 11104 (для панели C-more удаленное консольное приложение - **Remote Console Application**). Тут можно увидеть, что используя только один общедоступный IP-адрес (12.126.23.146), но используя несколько номеров портов TCP, можно получить доступ к 3-м различным панелям C- more внутри маршрутизатора (router).

C. NAT (Network Address Translation –трансляция сетевых адресов) и PAT (Port Address Translation -перевод адресов портов)

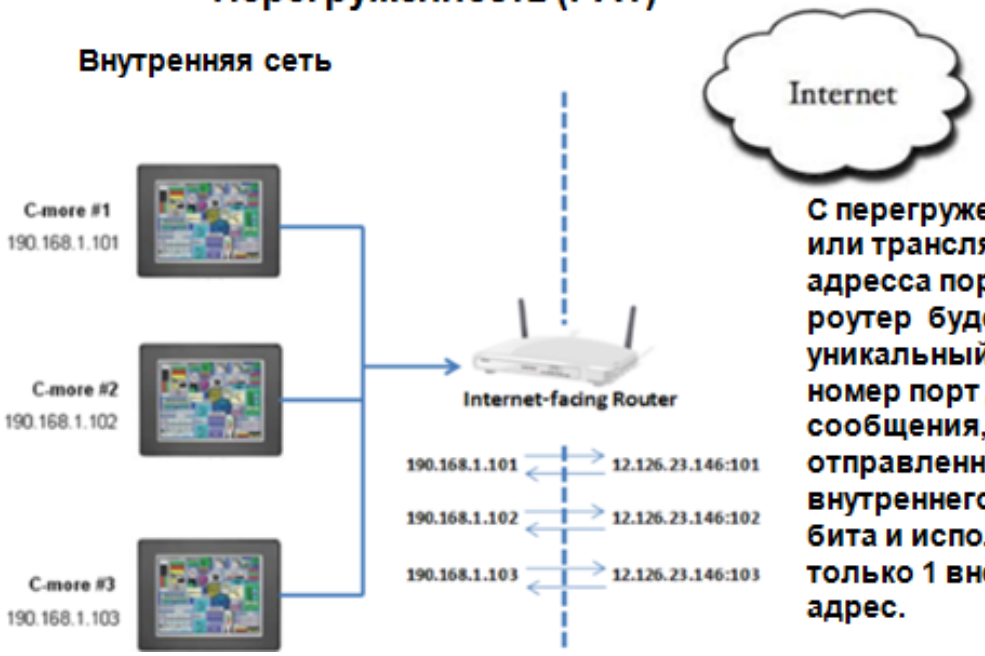
Одним из основных преимуществ использования маршрутизатора (router), обращенного к Internet, является возможность предоставления большому количеству устройств доступа в Internet без необходимости использования уникального адреса IANA. Обычно каждый публичный IP-адрес, который нужен, имеет ежемесячную стоимость, приписываемую ему. **NAT** и **PAT** позволяют нескольким внутренним, незарегистрированным IP-адресам «переводиться» на меньшую сумму (или 1) зарегистрированных публичных IP-адрес (адресов).

Статический/Динамический NAT



С статическим NAT вы конфигурируете внутренний IP к внешнему IP-адрес. С динамическим NAT, маршрутизатор (router) будет присваевать следующий доступный внешний IP-адрес.

Перегруженность (PAT)

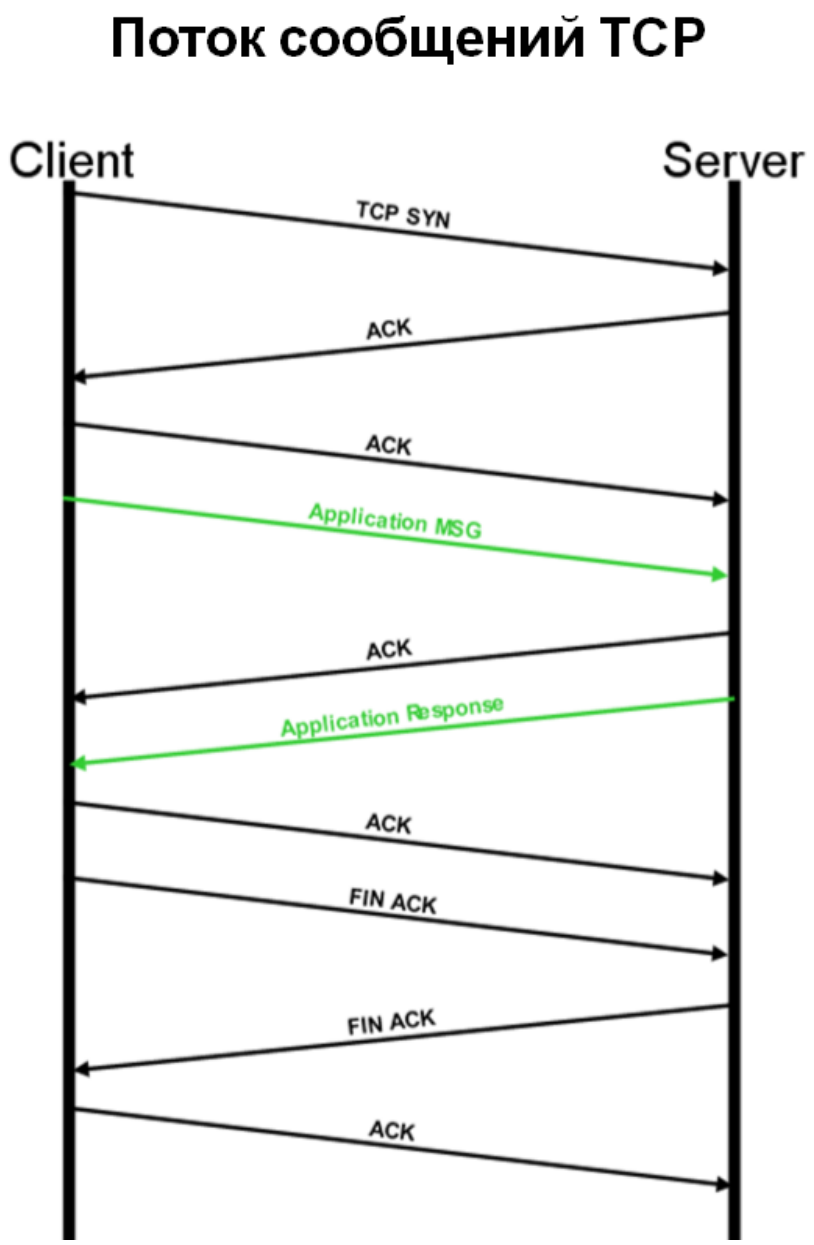


С перегруженностью или трансляцией адресса порта (PAT), роутер будет отмечать уникальный исходный номер порт для каждого сообщения, отправленного, для внутреннего крайнего бита и использовать только 1 внешний IP-адрес.

VI. TCP по сравнению с UDP.

В этом документе упоминается UDP и TCP. В следующем разделе объясняются основные различия между этими двумя уровнями протокола.

На приведенных ниже диаграммах показан фактический поток сообщений в TCP и UDP для приложения-клиента (Client) для отправки сообщения на сервер (Server) и для приложения-сервер (Server) для отправки ответа.



`

Как видно из диаграмм выше, для отправки и получения запроса приложения в транспортном механизме TCP требуется гораздо больше сообщений Ethernet, чем в транспортном механизме UDP.

С учетом сказанного существует множество плюсов и минусов использования TCP и использования UDP.

Цель TCP протокола - гарантировать надежность при отправке сообщения с одного устройства на другое. TCP выполняет это, сначала создавая «соединение» между Клиентом и Сервером. После установления соединения Клиент и Сервер могут отправлять данные взад и вперед, а механизм подключения TCP «гарантирует», что данные будут поступать туда с помощью сообщений подтверждения и последовательности (для размещения сообщений в правильном порядке). Если Клиент отправляет сообщение на Сервер и не получает сообщение подтверждения (ACK) с этого устройства, он затем повторно передает сообщение. Настоящая красота этого метода заключается в том, что пользовательскому приложению не нужно выполнять все эти сложные процедуры надежности, это встроено в так называемый «код стека», который обычно поставляется с операционной системой. Пользовательскому приложению необходимо только указать, с какого устройства он хочет связываться и отправить сообщение, а все остальное обрабатывается под «стеком».

Недостатком этого является снижение производительности системы при отправке всех этих сообщений и выполнении всех вычислений, необходимых для поддержания надежности. Это также добавляет значительный сетевой трафик.

И наоборот, нет никакого «соединения», связанного с обменом сообщениями UDP. В UDP клиент и сервер просто отправляют сообщение на устройство и надеются, что сообщение туда попадет. Если требуется надежность, тогда тайм-аут потока сообщения и механизмы повторного запуска должны быть записаны в приложение пользователя как на стороне клиента, так и на стороне сервера.

Из-за этого отсутствия обработки соединений производительность UDP превосходит TCP. В сети гораздо меньше трафика, и «стек» должен выполнять гораздо меньше вычислений.

Означает ли это, что UDP лучше, чем TCP или наоборот? Ответ: это зависит от приложения.

Если бы UDP использовался, для перемещения больших файлов данных, между одним устройством, в другой загруженной сети с множеством коммутаторов и маршрутизаторов, сложностью приложений для обработки «пропущенных» пакетов и повторных передач, получения пакетов в правильном порядке и т. д., то это вызывало бы определенную сложность. Поэтому TCP, для такого приложения, будет иметь больше смысла.

С другой стороны, если приложение хочет отправлять потоковое видео или звук через эту же сеть, то применение UDP может иметь больше смысла. Если бы пакет был потерян в этой ситуации, лучше было бы продолжать идти и игнорировать потерянный пакет, так как было бы мало смысла, пытаться, повторно вводить старый звук или видео позже в потоке. Этот тип приложений будет лучше использоваться с более быстрой пропускной способностью и меньшей надежностью.

Некоторые примеры приложений этих двух протоколов в наших продуктах приведены ниже:

UDP протокол - просмотр данных в DirectSoft для ПЛК с прямой логикой: в 90% случаев DirectSoft просто считывает значения, поэтому, если 1 пакет пропущен, данные считываются так быстро, что это будет почти незаметно. DirectSoft, использует, простой способ написания пакета и получения ответа, поэтому, если пропущена запись, он просто отправит снова или выдаст предупреждение. Он не будет пытаться собрать и повторно еще раз собрать большие файлы.

TCP протокол - Программное обеспечение C-more для передачи проекта: программное обеспечение C-more Software отправляет файл проекта на панель с использованием TCP. Поэтому в случае пропуска 1 пакета он будет повторно передан, и файл будет повторно собран правильно.

VII. DNS система доменных имён

Другое поле адреса, которое часто можно увидеть при настройке Ethernet-устройств - это адрес DNS. В большинстве случаев это «Предпочтительный DNS-сервер» или «Первичный DNS-сервер» и «Альтернативный DNS-сервер» или «Вторичный DNS-сервер».

DNS - это доменная система. Именно этот механизм позволяет приложениям использовать более легко запоминающееся имя для удаленного устройства по сравнению с загадочным адресом или значением. Например: при создании интернет-браузера, такого как: Internet Explorer, Chrome или Firefox, большую часть времени не вводится IP-адрес в поле URL для доступа к удаленному веб-серверу. Обычно вводится что-то вроде: www.google.com. Однако это не соответствует правилам Ethernet, поэтому IP-адрес должен быть получен из этого имени.

Когда вводится адрес предпочтительного DNS-сервера и адрес вторичного DNS-сервера, это дает путь для вашего PC «найти» IP-адрес указанного имени веб-сайта. Приложение интернет-браузера введет имя, указанное в (www.google.com в этом случае), и отправит это имя на IP-адрес, указанный в поле Предпочтительный DNS-сервер. Затем предпочтительный DNS-сервер будет отвечать IP-адресом указанного имени. Далее приложение интернет-браузера может отправить запрос на подключение к указанному веб-серверу. Если первая попытка отправить это имя на предпочтительный DNS-сервер не удалась, приложение интернет-браузера попытается отправить имя на альтернативный DNS-сервер.

Сам DNS-сервер имеет сложный метод поддержания своей таблицы IP-адресов по сравнению с именами. Другим преимуществом этого механизма является то, что устройства с интернет-адресами IP-адресов могут использовать пул IP-адресов, и это может быть «затенено» с помощью поиска DNS. **Техническая помощь. Если у вас есть вопросы относительно этой заявки, пожалуйста, свяжитесь с нами по телефону 8-800-707-18-71 для получения дополнительной помощи.**