Фотоотчет прислать на эл.почту spo203@mail.ru (В теме указать фамилию, группу, предмет, дату)

Тема: Протоколы транспортного уровня: TCP и UDP. Их характеристика и применение

5.6. Протоколы транспортного уровня

User Datagram Protocol

User Datagram Protocol — UDP — один из двух протоколов транспортного уровня, используемых в стеке протоколов TCP/IP. UDP позволяет прикладной программе передавать свои сообщения по сети с минимальными издержками, связанными с преобразованием протоколов уровня приложения в протокол IP. Однако при этом прикладная программа сама должна обеспечивать подтверждение того, что сообщение доставлено по месту назначения. Заголовок UDP-датаграммы (сообщения) имеет вид, показанный на рис. 5.11.

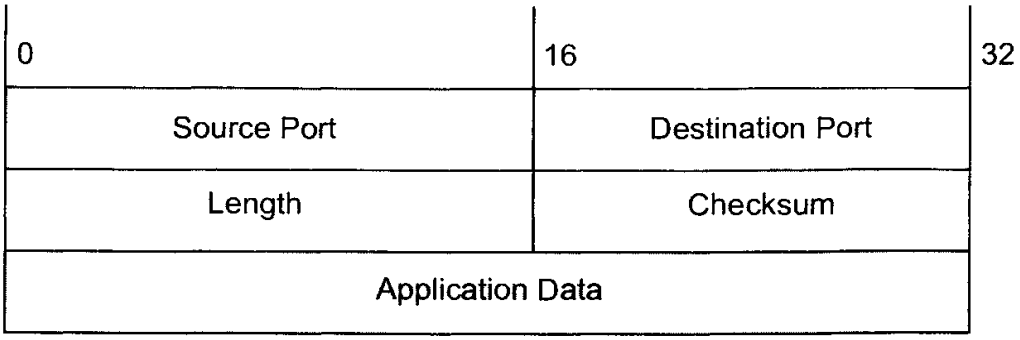

Рис. 5.11. Структура заголовка UDP-сообщения

Порты в заголовке определяют протокол UDP как мультиплексор, который позволяет собирать сообщения от приложений и отправлять их на уровень протоколов. При этом приложение использует определенный порт. Взаимодействующие через сеть приложения могут использовать разные порты, что и отражает заголовок пакета. Всего можно определить 2^{16} разных портов. Первые 256 портов закреплены за так называемыми «well known services (WKS)» (см., например, табл. 5.4).

Поле `Length` определяет общую длину сообщения. Поле `Checksum` служит для контроля целостности данных. Приложение, которое использует протокол UDP, должно поддерживать целостность данных, анализируя поля `Checksum` и `Length`. Кроме этого, при обмене данными по UDP прикладная программа сама должна заботиться о контроле получения данных адресатом. Обычно это достигается обменом подтверждениями о доставке между прикладными программами.

Наиболее известными сервисами, основанными на UDP, являются служба доменных имен BIND и распределенная файловая система NFS. Собственно, именно сообщение UDP и посылается в сеть, но при этом используется такой порт, который не имеет обслуживания, поэтому порождается ICMP-пакет, который и определяет отсутствие сервиса на принимающей машине, когда пакет наконец достигает машины-адресата.

В том случае, когда контроль качества передачи данных по сети имеет особое значение для приложения, используется протокол TCP. Этот протокол также называют надежным, ориентированным на соединение потокоориентированным протоколом. Рассмотрим формат передаваемой по сети датаграммы (рис. 5.12). Согласно этой структуре в TCP, как и в UDP, используются порты. В поле `Sequence Number` определен номер пакета в последовательности пакетов, которая составляет сообщение, затем идет поле подтверждения `Asknowledgment Number` и другая управляющая информация.

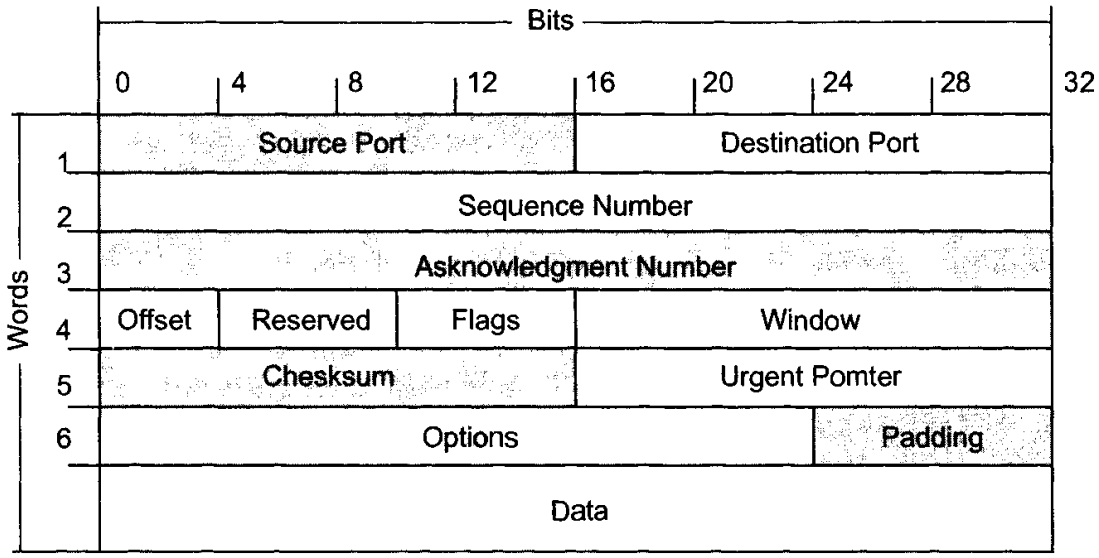

Рис. 5.12. Структура пакета TCP

Надежность TCP обеспечивается тем, что источник данных повторяет их передачу, если только не получит в определенный промежуток времени от адресата подтверждение об их успешном получении. Этот механизм называется *Positive Asknowledgement with Re-transmission (PAR).* В заголовке TCP предусмотрено поле контрольной суммы. Если при пересылке данные повреждены, то по контрольной сумме модуль, вычленяющий TCP-сегменты из пакетов IP, уничтожает поврежденный пакет (сообщение источнику не передается). Если же данные не были повреждены, то они пропускаются на сборку сообщения приложения, а в адрес источника отправляется подтверждение.

Ориентация на соединение определяется тем, что, прежде чем отправить сегмент с данными, модули TCP-источника и получателя обмениваются управляющей информацией. Такой обмен называется *handshake* (буквально «рукопожатие»). В TCP используется трехфазный handshake:

- источник устанавливает соединение с получателем, посылая ему пакет с флагом «синхронизации последовательности номеров» (`Synchronize Sequence Numbers — SYN`). Номер в последовательности определяет номер пакета в сообщении приложения. Это не обязательно должен быть 0 или 1, но все остальные номера будут использовать его в качестве базы, что позволит собрать пакеты в правильном порядке;
- получатель отвечает номером в поле подтверждения получения `SYN`, который соответствует установленному источником номеру. Кроме того, в поле «Номер в последовательности» может также сообщаться номер, который запрашивался источником;
- источник подтверждает, что принял сегмент получателя и отправляет первую порцию данных.

После установки соединения источник посылает данные получателю и ждет от него подтверждений об их получении, затем снова посылает данные и т. д., пока сообщение не закончится. Заканчивается сообщение, когда в поле флагов выставляется бит FIN, что означает «больше нет данных».

Потоковый характер протокола определяется тем, что `SYN` определяет стартовый номер для отсчета переданных байтов, а не пакетов. Это значит, что если `SYN` был установлен в 0 и было передано 200 байтов, то номер, установленный в следующем пакете, будет равен 201, а не 2.

Потоковый характер протокола и требование подтверждения получения данных порождают проблему скорости передачи данных. Для ее решения используется поле `Window` (`окно`). Идея применения *окна* достаточно проста: передавать данные, не дожидаясь подтверждения об их получении. Это значит, что источник передает некоторое количество данных, равное `Window`, без ожидания подтверждения об их приеме, и только после этого останавливает передачу и ждет подтверждения. Если он получит подтверждение только на часть переданных данных, то начнет передачу новой порции с номера, следующего за подтвержденным.

Контрольные вопросы

2. Какую функцию выполняет протокол TCP?

12. Какие протоколы транспортного уровня вы знаете?
13. Что такое инкапсуляция и фрагментация?
14. Что такое TCP/UDP-порт?

16. Какова структура пакета TCP?