

Распределение протоколов по уровням модели OSI

7	Прикладной	напр., HTTP, SMTP, SNMP, FTP, Telnet, SSH, SCP, SMB, NFS, RTSP, BGP
6	Представления	напр., XDR, AFP, TLS, SSL
5	Сеансовый	напр., ISO 8327 / CCITT X.225, RPC, NetBIOS, PPTP, L2TP, ASP
4	Транспортный	напр., TCP, UDP, SCTP, SPX, RTP, ATP, DCCP, GRE
3	Сетевой	напр., IP, ICMP, IGMP, CLNP, OSPF, RIP, IPX, DDP, ARP, RARP
2	Канальный	напр., Ethernet, Token ring, HDLC, PPP, X.25, Frame relay, ISDN, ATM, MPLS
1	Физический	напр., электрические провода, радиосвязь, волоконно-оптические провода, Wi-Fi

Распределение протоколов по уровням модели TCP/IP

5	Прикладной «7 уровень»	напр., HTTP, RTP, FTP, DNS <i>(RIP, работающий поверх UDP, и BGP, работающий поверх TCP, являются частью сетевого уровня)</i>
4	Транспортный	напр., TCP, UDP, SCTP, DCCP <i>(протоколы маршрутизации, подобные OSPF, что работают поверх IP, являются частью сетевого уровня)</i>
3	Сетевой	Для TCP/IP это IP (IP) <i>(вспомогательные протоколы, вроде ICMP и IGMP, работают поверх IP, но тоже относятся к сетевому уровню; протокол ARP является самостоятельным вспомогательным протоколом, работающим поверх физического уровня)</i>
2	Канальный	Ethernet, IEEE 802.11 Wireless Ethernet, SLIP, Token Ring, ATM и MPLS
1	Физический	напр., физическая среда и принципы кодирования информации, T1, E1

Физический уровень

Физический уровень описывает среду передачи данных (будь то коаксиальный кабель, витая пара, оптическое волокно или радиоканал), физические характеристики такой среды и принцип передачи данных (разделение каналов, модуляцию, амплитуду сигналов, частоту сигналов, способ синхронизации передачи, время ожидания ответа и максимальное расстояние).

Канальный уровень

Канальный уровень описывает, каким образом передаются пакеты данных через физический уровень, включая *кодирование* (то есть специальные последовательности бит, определяющих начало и конец пакета данных). Ethernet, например, в полях заголовка пакета содержит указание того, какой машине или машинам в сети предназначен этот пакет.

Примеры протоколов канального уровня — Ethernet, IEEE 802.11 Wireless Ethernet, SLIP, Token Ring, ATM и MPLS.

Сетевой уровень

Сетевой уровень изначально разработан для передачи данных из одной (под)сети в другую. Примерами такого протокола является X.25 и IPС в сети ARPANET.

С развитием концепции глобальной сети в уровень были внесены дополнительные возможности по передаче из любой сети в любую сеть, независимо от протоколов нижнего уровня, а также возможность запрашивать данные от удалённой стороны, например в протоколе ICMP (используется для передачи диагностической информации IP-соединения) и IGMP (используется для управления multicast-потоками).

ICMP и IGMP расположены над IP и должны попасть на следующий — транспортный — уровень, но функционально являются протоколами сетевого уровня, и поэтому их невозможно вписать в модель OSI.

Пакеты сетевого протокола IP могут содержать код, указывающий, какой именно протокол следующего уровня нужно использовать, чтобы извлечь данные из пакета. Это число — уникальный *IP-номер протокола*. ICMP и IGMP имеют номера, соответственно, 1 и 2.

К этому уровню относятся: DHCP^[1], DVMRP, ICMP, IGMP, MARS, PIM, RIP, RIP2, RSVP

Транспортный уровень

Протоколы транспортного уровня могут решать проблему негарантированной доставки сообщений («дошло ли сообщение до адресата?»), а также гарантировать правильную последовательность прихода данных. В стеке TCP/IP транспортные протоколы определяют, для какого именно приложения предназначены эти данные.

TCP — «гарантированный» транспортный механизм с предварительным установлением соединения, предоставляющий приложению надёжный поток данных, дающий уверенность в безошибочности получаемых данных, перезапрашивающий данные в случае потери и устраняющий дублирование данных. TCP позволяет регулировать нагрузку на сеть, а также уменьшать время ожидания данных при передаче на большие расстояния. Более того, TCP гарантирует, что полученные данные были отправлены точно в такой же последовательности. В этом его главное отличие от UDP.

UDP протокол передачи датаграмм без установления соединения. Также его называют протоколом «ненадёжной» передачи, в смысле невозможности удостовериться в доставке сообщения адресату, а также возможного перемешивания пакетов. В приложениях, требующих гарантированной передачи данных, используется протокол TCP.

UDP обычно используется в таких приложениях, как потоковое видео и компьютерные игры, где допускается потеря пакетов, а повторный запрос затруднён или не оправдан, либо в приложениях вида запрос-ответ (например, запросы к DNS), где создание соединения занимает больше ресурсов, чем повторная отправка.

Прикладной уровень

На прикладном уровне работает большинство сетевых приложений.

Эти программы имеют свои собственные протоколы обмена информацией, например, HTTP для WWW, FTP (передача файлов), SMTP (электронная почта), SSH (безопасное соединение с удалённой машиной), DNS (преобразование символьных имён в IP-адреса) и многие другие.

В массе своей эти протоколы работают поверх TCP или UDP и привязаны к определённому порту, например:

- HTTP на TCP-порт 80 или 8080,
- FTP на TCP-порт 20 (для передачи данных) и 21 (для управляющих команд),
- SSH на TCP-порт 22,
- запросы DNS на порт UDP (реже TCP) 53,
- обновление маршрутов по протоколу RIP на UDP-порт 520.

Эти порты определены Агентством по выделению имен и уникальных параметров протоколов (IANA).

К этому уровню относятся: Echo, Finger, Gopher, HTTP, HTTPS, IMAP, IMAPS, IRC, NNTP, NTP, POP3, POPS, QOTD, RTSP, SNMP, SSH, Telnet, XDMCP.