Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра автоматизированных систем управления (АСУ)

А.Я. Суханов

Сети и телекоммуникации

Учебно-методическое пособие по самостоятельной работе для студентов направления бакалавриата 09.03.01

2023

Суханов А.Я.

Сети и телекоммуникации: Учебно-методическое пособие по самостоятельной работе для студентов направления бакалавриата 09.03.01, – 18 с.

Учебное методическое пособие содержит программу обучения и перечень контрольных вопросов.

© ТУСУР, каф. АСУ

© Суханов А.Я., 2023

Оглавление

Программа дисциплины

Целью дисциплины является обучение студентов основам построения и функционирования вычислительных сетей (ВС) и телекоммуникационных систем (ТКС), изучение основных современных протоколов взаимодействия удаленных систем.

Основной задачей изучения дисциплины является приобретение студентами прочных знаний и практических навыков в области, определяемой основной целью курса. В результате изучения дисциплины студенты должны усвоить следующие понятия и определения: классификация информационно-вычислительных сетей, способы коммутации, взаимодействие программного и аппаратного обеспечения сетей, протоколы и интерфейсы, эталонная модель взаимосвязи открытых систем, аналоговые и цифровые каналы передачи данных, модемы, базовые технологии локальных сетей, глобальные сети, технологии современных телекоммуникаций. В части организации программного обеспечения сетей изучаются способы адресации в протоколах TCP/IP, алгоритмы маршрутизации, протоколы файлового обмена, электронной почты, дистанционного управления, Web-технологии, способы организации распределенных вычислений, основные возможности сетевых операционных систем. Рассматриваются как низкоуровневые (сокеты) так и высокоуровневые программные технологии для работы в вычислительных сетях.

При изучении курса студент выполняет лабораторные работы, также изучая теоретические вопросы и сдавая их при защите лабораторной работы.

Содержание курса.

1 Введение в сети и телекоммуникации. Основы передачи данных.

Предмет и содержание курса. Вычислительные машины, сети и системы телекоммуникаций - важный фактор научно-технического прогресса и прогресса цивилизации. Различие понятий вычислительная машина и вычислительная сеть. История и современные тенденции развития вычислительных сетей. Задачи, решаемые современными вычислительными сетями: файловый сервис, сервис печати, сервис сообщений, сервис приложений, сервис баз данных. Понятие среды передачи данных. Характеристики сред. Шкала электромагнитных колебаний. Стандарты сред передачи данных. Понятие полосы пропускания. Количество информации и энтропия, единицы измерения. Законы Найквиста, Шеннона, Котельникова. Аналоговая и цифровая формы представления информационного сигнала. Способы модуляции. Информационная и техническая скорость передачи. Алгоритмы кодирования и сжатия информации.

2 Принципы построения сетей ЭВМ. Модель взаимодействия открытых систем (ВОС).

Классификация сетей. Многоуровневый подход к организации сетей. Протоколы и

интерфейсы. Стандарты и источники стандартов ВС. Открытые системы. Модель взаимодействия открытых систем (ВОС). Понятие стека протоколов. Взаимодействие различных уровней стека. Физический уровень ВОС. Задачи физического уровня. Типы соединения. Физическая топология. Аналоговое и цифровое представление сигнала. Синхронизация бит. Использование полосы пропускания. Мультиплексирование (TDMA, FDMA, CDMA, OFDMA). Канальный уровень. Логическая топология. Методы доступа к среде передачи данных. Адресация канального уровня. Синхронизация передачи. Сервис соединения канального уровня. Сетевой уровень. ервис шлюзов. Адресация в сетях. Задача маршрутизации. Методы маршрутизации. Коммутация. Виртуальные каналы. Протоколы и алгоритмы групповой маршрутизации. Транспортный, сеансовый, представительский уровни. Разрешение имен. Адресация транспортного соединения. Сегментация, блокирование, сцепление данных. Алгоритм медленного пуска. Сервис транспортного соединения. Задачи уровня представления. Шифрование. Прикладной уровень. Сервисы прикладного уровня. Оповещение о сервисах. Использование сервисов.

3 Стеки сетевых протоколов

TCP/IP (IPv4, IPv6, ICMP, DHCP, DNS, ARP, RARP, SCTP, UDP), IPX/SPX, SMB/NetBIOS, DNA, SNA, AppleTalk, DecNet, стек OSI.

4 Базовые технологии сетей.

Ethernet, TokenRing, Frame Relay, ATM, FDDI, 100VG ANYLAN, Wifi, WiMax. Территориальные сети. Аппаратное обеспечение сетей: сетевые интерфейсные карты, концентраторы, коммутаторы, мосты, маршрутизаторы. Сегментация. Широковещательный шторм. Протокол покрывающего дерева.

5 Современные телекоммуникационные системы

Коммутируемые телефонные сети. Интегральные сети цифрового обслуживания. Сотовая телефония. Спутниковые системы связи и навигации. Низкоорбитальные и высокоорбитальные системы. Системы глобального позиционирования и синхронизации. Спутниковый Интернет. Система Iridium, GlobalStar, GPS, VSAT. Технологии сотовой связи. UMTS, GSM.

6 Программное обеспечение сетей. Сетевые операционные системы.

Определение сетевой ОС. Одноранговые сети и сети «клиент\сервер». Обзор сетевых ОС. Служба сетевых каталогов как средство интеграции сетевых продуктов. Драйверы сетевых устройств. Сокеты Беркли. Программирование на уровне сокетов.

7 Безопасность в вычислительных сетях.

Общие правила безопасности. Классы безопасности. Безопасность в ВС по ГОСТ. Понятие Firewall. Аутентификация. Авторизация. SASL. AUTH 2.0. SSL. Безопасность

беспроводных сетей, WEP, атаки на WEP, WPA, WPA2. Квантовые алгоритмы шифрования. DoS, DDoS атака, спуфинг, атака не переполнение буфера. Луковая и чесночная маршрутизация (TOR, I2P). DPI. IPSec.

Экзаменационные вопросы.

1. Виды сервисов, предоставляемых современными сетями.
2. Глобальные, муниципальные и локальные сети. Другие способы классификации сетей.
3. Необходимость стандартизации сетей ЭВМ. Источники официальных стандартов и рекомендаций. (iso, ieee (802.3, 802.11), isoc, iana (rir)), способы принятия стандартов.
4. Источники стандартов Интернет. Модель взаимодействия открытых систем. Принцип построения. Понятия: протокол, интерфейс.
5. Понятие среды передачи данных. Шкала электромагнитного спектра. Достоинства и недостатки каждого из диапазонов.
6. Проводные технологии передачи данных. Основные характеристики.
7. Беспроводные технологии передачи данных. Основные характеристики.
8. Спутниковые системы передачи данных. Низко- и высокоорбитальные системы.
9. Аппаратное обеспечение сетей ЭВМ (сетевые карты, мосты, коммутаторы, маршрутизаторы, модемы, мультиплексоры). Разделение устройств по уровням OSI.
10. Понятие стека протоколов. Существующие стеки протоколов (назвать не менее 4-5 стеков).
11. Стеки TCP/IP и IPX/SPX и OSI. Принципиальное отличие этих стеков.
12. Пример взаимодействие двух компьютеров в сети. Клиент и сервер.
13. Задачи решаемые, сетевыми операционными системами.
14. Коммутация пакетов и коммутация каналов. Сравнительная характеристика, области и технологии применения.
15. Аналоговое и цифровое кодирование. Связь частоты, методов кодирования и информационной скорости передачи сигнала, энтропия: формула Шеннона.
16. Физический уровень модели OSI. Единицы данных физического уровня. Модуляция, Манипуляция QAM. Методы синхронизации бит (скрэмблирование, 4b5b). Потенциальное кодирование, импульсное кодирование.
17. Физическая топология. Достоинства и недостатки различных топологий. Мультиплексирование физического уровня: методы мультиплексирования, области применения. CDMA, TDMA, FDMA.
18. Методы помехоустойчивого кодирования.
19. Канальный уровень модели OSI. Логическая топология сетей. Единицы данных канального уровня. Методы доступа к среде передачи данных. Синхронизация байт. Сервис соединений.

20. Технологии канального уровня: Ethernet,
21. Token Ring.
22. Технология 100 VG-AnyLAN,
23. FDDI.
24. WiFi, CSMA-CA. RTS. CTS. Режим работы с точкой доступа и без нее.
25. CSMA-CD.
26. Сегментация локальных сетей.
27. Широковещательный шторм. STA и STP.
28. Сетевой уровень модели OSI. Единицы данных, адресация сетевого уровня. Сервис шлюзов. Маршрутизация.
29. Коммутация. Коммутация пакетов, каналов, сообщений.
30. Транспортный уровень модели OSI. Сервис транспортного уровня. Сегменты.
31. Сеансовый, уровень представления и прикладной уровни.
32. Адресация в сетях. Адресация в сетях TCP/IP. CIDR. NAT. Проблемы с NAT при использовании протоколов прикладного уровня.
33. Протоколы ARP, DHCP.
34. Маршрутизация. Алгоритм Дейкстры нахождения кратчайшего пути.
35. Маршрутизация. Метод заливки.
36. Архитектура и службы электронной почты. Пользовательский агент и агент передачи сообщения. Перечислите протоколы электронной почты Internet. Форматы сообщений. RFC822. MIME. Правила кодировки base64.
37. Протоколы SMTP, POP3, IMAP4.
38. Sasl аутентификация, CRAM-MD5, Digest-MD5.
39. Основные виды сетевых атак.
40. Динамические веб-документы на стороне сервера. Назначение. Примеры реализации. Технологии CGI, ISAPI, PHP, ASP, JSP. Примеры кода. Достоинства и недостатки.
41. Saas, Iaas, Paas.
42. Динамические веб-документы на стороне клиента. JavaScript, Java, ActiveX, SWF, Haxe, Dart. Примеры реализации. Достоинства и недостатки.
43. Описание сервиса в модели OSI. Функции уровней.
44. Протокол маршрутизации RIP (триггерные обновления, расщепление горизонта, замораживание изменений).
45. OSPF.

46. IGRP. EIGRP.
47. Автономные системы.
48. BGP.
49. Протокол IP версии 6. Отличия от версии 4. IP адреса v6.
50. Форматы пакетов IPv4 и IPv6.
51. Протокол SCTP, мультихоуминг, многопоточность, процедура четверного рукопожатия.
52. Форматы сегментов tcp, udp.
53. Протокол dns.
54. Протокол ICMP v4.
55. Процедура тройного рукопожатия tcp. TCP flooding.
56. Луковая маршрутизация. TOR.
57. Таймеры TCP.
58. Алгоритм медленного пуска TCP.
59. Протоколы и алгоритмы групповой маршрутизации. RPF
60. Протоколы и алгоритмы групповой маршрутизации. CBT .
61. Протокол IGMP.
62. Основные способы обеспечения QoS.
63. Интеллектуальный коммутатор.

Перечень дополнительных вопросов на экзамене.

1. Уровни OSI. Назначение уровней. Сетевые сервисы (сервис печати, сервис СУБД (файлсерверная, серверная архитектура, трехзвенная), сервис сообщений, файловый сервис, сервис приложений (iaas, saas, paas, для чего предназначены данные сервисы, виртуализация на уровне ядра), приведите примеры протоколов обеспечивающих данные сервисы. Понятие интерфейса и протокола. Источники стандартов в телекоммуникациях, сетях и интернете (iso, ieee (802.3, 802.11), isoc, iana (rir)), способы принятия стандартов. Команды ping, ipconfig (ifconfig), tracert (traceroute).

Записать примеры определений функций сервисов (описание их параметров и названия), реализуемые каждым из уровней (предоставляемые вышестоящему уровню). Пример, интерфейсных функций сокетов для транспортного уровня.

2. Основные характеристики протоколов электронной почты. Команды протоколов SMTP, POP3. Система DNS. Mx запись. (как применяется mx dns запрос при отправке почты, какие еще dns запросы бывают, рекурсивный и итеративный запросы, корневые сервера, локальные DNS сервера) Взаимодействие почтовых серверов. MIME формат.(Boundary) . Протокол IMAP. Команда nslookup. Схема взаимодействия почтовых агентов и серверов. Round robin DNS. Кодировка base64. Зачем нужна. Как сделать mx запрос командой nslookup.

3. Команды протокола HTTP. (отличие PATCH и PUT). Коды ответов HTTP (200, 304, 302, 401, 403, 402). Заголовки range, referer, host, expires, vary. Команды протокола FTP. Устройство NAT (Full cone NAT, restricted NAT). Как NAT меняет пакет и сегмент. Привести пример для подмены адресов устройством NAT для различных сетей (192.168.0.0-192.168.255.255, 172.16.0.0-172.32.255.255, 10.0.0.0, 100.0.0.0, какие префиксы у данных сетей), сколько компьютеров может поддерживать NAT с узлами открывшими сколько-то соединений. Проблемы клиента за устройством NAT в активном режиме FTP. Зачем нужен активный и пассивный режим. Почему реализован переход с HTTP1 на HTTP2 и далее на HTTP3. Проблема HOL для HTTP и TCP. В чем проблема текстового протокола HTTP1 и недостатки использования base64. Есть ли у FTP проблема с base64. Зачем FTP возвращает на команду PASV еще и IP, что это дает, зачем вообще FTP два и более соединений в отличие от HTTP.

4. Основные технологии на стороне сервера и клиента. CGI, FAST CGI (чем FAST CGI лучше CGI, через что передаются данные, привести примеры), PHP, ASP, JSP, Фреймворки Node JS, django, ruby on rails, Java script, Dart, SWF, ACTIVEX, Haxe, Wasm. Зачем нужен балансировщик нагрузки типа NGINX, что такое медленные клиенты и в чем преимущество дополнительного использования NGINX, зачем над веб-сервером на fastapi или flask доп WSGI сервера типа uvicorn, проблема GIL python. JAVA spring, аннотации и декораторы в современных фреймворках для web.

CSRF – межсайтовая подделка запроса. Способы защиты. Xml. json отличие от xml

Xhtml, DOM sax парсер, RDF, OWL семантическая паутина, XQuery, чем foreach отличается от match в Xsl. Способы ускорения выполнения запросов. (отправка данных авторизации в самом запросе). Привести пример xml файла и шаблона преобразования в HTML.

5. Proxy. socks5, цепочка proxy (нарисовать схему цепочки), UDP Associate (нарисовать таблицы преобразования IP и портов), http proxy, http connect proxy, if-modified-since last-modified if-none-match tag, http connect proxy. VPN. IPSec. SSL, TLS. HTTPS. Принцип создания сертификата.

6. Физический уровень.

Методы кодирования. Модуляция (амплитудная, фазовая, частотная, QAM), манипуляция. Потенциальные коды. (AMI, MLT-3, PAM, почему в MLT-3 три уровня, а в AMI 2 на лог. единицу) Импульсные коды. Синхронизация (как достигается). Коды 4b-5b, 8b-6T. Скрэмблирование. Кодирование на четность, по паритету, код хэмминга. Сверточные коды. Мультиплексирование. (TDMA, FDMA, CDMA). Почему OFDMA на исходящее, а SC-FDMA на входящее. Зачем в CDMA ортогональность кодов псевдослучайной последовательности. Хаб, концентратор. Виды проводных физических сред передачи — витая пара, оптоволоконный кабель, коаксиальный кабель, как устроены. Классификация сетей. Локальные, муниципальные и глобальные сети. Сети отделов, кампусов, корпоративные.

7. Протоколы и технологии канального уровня.

Доступ к среде CSMA/CD. Технология Ethernet. Что такое двойной экспоненциальный откат. Что такое коллизия. Почему есть ограничение на минимальный размер кадра, как рассчитывается минимальный размер кадра на основе размера сети, пропускной способности сети. MAC адрес. Формат кадра. Что такое интеллектуальный

коммутатор, зачем нужен STA. Что такое широковещательный шторм, зачем нужна сегментация сетей Ethernet, как изменяется интенсивность трафика при сегментации.

Технология Token Ring. Для чего MSAU. Что такое маркерный доступ. Основные скорости передачи. Приоритеты.

Технология FDDI. Зачем двойное кольцо. Размер сети. Скорость передачи.

Технология WiFi. Доступ к среде. CSMA/CA. Difs, sifs, pifs интервалы. ACK, RST, CST.

8. Маршрутизация

Дистанционны векторные протоколы (DVA), RIP протокол, триггерное обновление, замораживание изменений и расщепление горизонта, route poisoning, poison reverse протокол BGP, IGRP, EIGRP внутренний протокол и внешний протокол маршрутизации IGP EGP. Маршрутные метрики, Алгоритм маршрутизации на основе состояний линий связи (LSA – link state). OSPF, автономные системы в интернете, групповые протоколы маршрутизации, CBT RPF PIM SM, DM, IGMP.

9. UDP. TCP алгоритм медленного пуска, тройного рукопожатия, обеспечение надежности, таймеры TCP, как рассчитывается таймер повторной передачи, SCTP, алгоритм четверного рукопожатия, многопоточность, мультихоуминг, TCP SYN Flood атака. Формат сегмента TCP, SCTP и UDP.

Формат пакета IPV4 и IPV6, отличие v4 от v6, формат адресов, фрагментация, протокол ARP, RARP, NDS, связь адреса IPv6 с MAC, контрольная сумма в 4 и 6 версии, недостатки, формат адреса 6,. Команды ICMP. ICMP флуд атака. Перенаправление шлюза. Эхо запрос, эхо ответ. Подавление трафика.

10. Digest-авторизация. CRAM-md5. AUTH 2.0. Authorization credentials flow + PKCE, Client Credentials Flow, implicit flow, device code flow. JWT токен, scope. Для чего может понадобиться фингер принт. Refresh token. Access token. Что возможно при их перехвате.

Примеры тестов

ОПК-3.1. Знает принципы, методы и средства решения стандартных задач профессиональной деятельности, основы информационной и библиографической культуры, современные информационно-коммуникационные технологии для поиска и анализа информации, основные требования информационной безопасности в профессиональной деятельности

1 Что представляет собой DoS TCP флуд атака ?

- = постоянную отправку TCP SYN сегментов серверу, что ограничивает доступ к серверу легальных клиентов
- постоянную отправку TCP SYN сегментов серверу, что приводит к ответу ACK от сервера и переводит его в нерабочее состояние
- постоянные ответы TCP ACK серверу, что приводит его к отказу в обслуживании
- участие в процедуре тройного рукопожатия распределенного бот-нета

2 Для чего нужен refresh токен ?

- = Чтобы затем по refresh токену можно было запросить новый access и refresh токены, что позволит автоматически проводить их обновление в случае компрометации или устаревания первого и получать доступ к ресурсу
- Чтобы затем по refresh токену можно было запросить новый access токен, что позволит автоматически проводить его обновление и в случае утери access токена сессию труднее будет взломать
- Чтобы по refresh токену можно было проверить валидность access токена
- Refresh токен отправляется на ресурс для обновления данных ресурса, а access для доступа к ресурсу

3 Что представляет собой Authorization code flow в OAUTH 2.0 ?

- = Отправка авторизационным сервером специального кода владельцу ресурса, например, через браузер, который отправляется затем приложению клиента, например на бэк часть приложения, далее код отправляется на авторизационный сервер и клиентскому приложению выдается аксес и рефреш токен на доступ к ресурсу
- Отправка авторизационным сервером специального кода владельцу ресурса, например, через браузер, который отправляется затем на авторизационный сервер и пользователю выдается аксес и рефреш токен на доступ к ресурсу
- Отправка авторизационным сервером специального кода клиентскому приложению, например, на бэк часть веб-приложения, который отправляется затем на авторизационный сервер и клиентскому приложению выдается аксес и рефреш токен на доступ к ресурсу и авторизационному приложению на сервере
- Отправка клиенту аксес и рефреш токена на клиентское приложение со стороны авторизацинного сервера, по которым клиентского приложение получается специальный код доступа к ресурсу

4 Что представляет собой модификация PKCE Authorization code flow в OAUTH 2.0 ?

- = Добавление отправки челендж строки сформированной по верифай строке на авторизационный сервер, после получения на клиентское приложение кода, код отправляется на авторизационный сервер вместе с верифай строкой для получения аксес и рефреш токена, предотвращает от уязвимости при возможной утери кода и клиентского секрета
- Добавление отправки верифай строки сформированной по челендж строке на авторизационный сервер, после получения на клиентское приложение кода, код отправляется на авторизационный сервер вместе с челендж строкой для получения аксес и рефреш токена, предотвращает от уязвимости при возможной утери аксес токена и клиентского секрета
- Добавление отправки челендж строки сформированной по верифай строке на авторизационный сервер, после получения на клиентское приложение кода, код отправляется на авторизационный сервер вместе с верифай строкой для получения аксес и рефреш токена, предотвращает от уязвимости при возможной утери клиентского секрета и аксес токена
- Добавление отправки челендж строки сформированной на основе клиентского секрета на авторизационный сервер, после получения на клиентское приложение кода, код отправляется на авторизационный сервер вместе с клиентским секретом для получения аксес и рефреш токена, предотвращает от уязвимости при возможной утери кода

ОПК-3.2. Умеет решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

1 Как получить ip адрес по доменному имени командой предназначенной для этого?

- Вызвать команду ping с доменным именем
- Вызывать команду nslookup с доменным именем и типом запроса mx
- =Вызывать команду nslookup с доменным именем
- Вызывать команду nslookup с типом запроса PTR

2 Как получить маршрут следования пакета до узла назначения?

- =Использовать команду traceroute
- Использовать команду ping
- Использовать команду nslookup
- Использовать команду ifconfig или ipconfig

3 Из какого списка адресов назначить адреса узлов за устройством NAT, если количество узлов порядка 2 в степени 19 ?

- =172.16.0.0-172.31.255.255
- 192.168.x.x
- 10.x.x.x
- 100.x.x.x

4 Какую информацию к DNS системе не позволяет получить команда nslookup ?

- IP адрес по доменному имени
- список почтовых серверов для данного доменного имения
- доменное имя по IP адресу
- =описание сервиса предоставляемого данным доменом

ОПК-3.3. Владеет навыками подготовки и оформления информационных ресурсов, например, в виде обзоров, рефератов, докладов по вопросам профессиональной деятельности, с применением современных технологий и с учетом основных требований информационной безопасности

1 Что представляет собой сервис Google документы ?

- = SaaS, сетевой сервис приложений
- PaaS, платформа как сервис
- IaaS, инфраструктура как сервис
- DaaS, десктоп как сервис

2 Чем отличается шаблон match от for each при формировании документа на основе xslt шаблона ?

- = Match проходит по всем уровням вложения, а for each по определенному уровню вложения
- for each проходит по всем уровням вложения, а match по определенному уровню вложения
- for each является циклом проходящим по определенному количеству тэгов, а match проходят по совпадениям
- Нет никаких отличий

3 Чем отличается SAX от DOM парсера ?

- = SAX использует событийный механизм доступа, не требует загрузки всего xml файла, DOM парсер загружает весь файл, но может изменять узлы, требует больше памяти
- SAX использует событийный механизм доступа, требует загрузки всего xml файла, DOM парсер загружает частично файл, но может изменять узлы, требует меньше памяти
- SAX использует событийный механизм доступа, не требует загрузки xml файла, DOM может изменять узлы, требует меньше памяти
- Нет никаких отличий

4 Какая технология позволяет обеспечить документирование веб-сервиса?

- = Open API, RAML
- Google Docs
- JSON, XML
- XML RPC

ПКС-1.1. Знает методики разработки программного обеспечения для решения конкретных производственных и научно-исследовательских задач

1 Какой будет правильной последовательность вызова функций при создании сервера?
- = socket, bind, listen, accept, recv, send
- socket, listen, bind, accept, send
- socket, connect, recv, send
- socket, accept, bind, listen, recv, send

2 Какие функции не нужны при создании дейтаграммного сокета?
- = listen, accept
- recv, send
- recv_from, send_to
- socket, bind

ПКС-1.2. Умеет принимать проектные решения при выполнении производственных и научно-исследовательских задач

1 Чем обусловлен переход с HTTP 1x на HTTP 2x?
- = HTTP 2 бинарный протокол, что позволило сократить затрачиваемые ресурсы на передачу бинарных данных, а так же реализовать мультиплексирование в рамках одного TCP соединения
- HTTP 2 избавил от проблемы блокировки начала строки (HOLB) присущей использованию протоколу TCP
- HTTP 2 позволяет нарезать данные на отдельные части и отсылать их по разным TCP соединениям и сетевым интерфейсам
- HTTP 2 бинарный протокол, что позволяет ему не тратить время на кодировку в base64, кроме того он полностью избавился от проблем с блокировкой начала строки (HOLB)

2 Чем обусловлен переход с HTTP 2x на HTTP 3x?
- = HTTP 2 хоть и реализует мультиплексирование, но в результате использования TCP имеет проблему с HOLB, протокол HTTP 3 планируют делать на базе дейтаграммного QUIC
- HTTP3 позволяет шифровать данные самостоятельно, что и обуславливает его основное преимущество перед HTTP 2
- HTTP3 текстовый протокол, который удобен для разбора и пользователям
- HTTP3 идея корпорации Google, направленная на захват рынка будущего Web 3.0

3 В чем отличие Fast CGI от CGI?
- = Fast CGI запускает один раз обрабатывающий процесс и передачу данных осуществляет через сокетные соединения, CGI запускает на каждый запрос новый процесс и передает данные через стандартный поток ввода вывода

- Fast CGI запускает несколько процессов для обработки каждого запроса и передачу данных осуществляет через сокетные соединения, CGI запускает на каждый запрос новый процесс, но данные передает данные через стандартный поток ввода вывода
- Fast CGI требует много памяти, но более быстрый при обработке запросов, потому что CGI выполняет обработку запросов через стандартные потоки ввода вывода
- CGI довольно старая и надежная технология, но требует много времени на обработку поступающих запросов через стандартные потоки ввода вывода, FAST CGI передает данные через сокетные соединения, что гораздо быстрее, что и обуславливает ее преимущество

4 Сколько потребуется открыть соединений на каждом прокси сервере, чтобы создать цепочку Прокси, если каждый Прокси по протоколу SOCKS5 возвращает новый порт, при этом в создано три эшелона прокси по схеме C->P->P->P->(W,W,W,W), буквами C обозначен клиент, буквами P – Прокси, W – четыре целевых веб сервера.

- = 7,6,5
- 6, 5, 4
- 4, 4, 4
- 5, 4, 4

5 Зачем нужна кодировка base64?

- = содержит группу символов ASCII, которая не совпадает с управляющими символами в текстовых протоколах, что дает возможность передачи бинарных данных
- За счет увеличения объема передаваемых данных на 1/3 можно передать дополнительную информацию
- Содержит 64 символа ASCII, которые содержат символы латинского алфавита и цифры, что делает их восприятие более удобным
- Уменьшить объем передаваемых данных за счет использования 6 бит на символ

6 Что дает дополнительное пассивное соединение и зачем оно было добавлено в протоколе FTP?

- = позволяет передавать по другому соединению данные, в том числе бинарные, позволяет провести балансировку нагрузки на FTP сервере
- Это устаревший протокол, который уже редко используется и данный недостаток был исправлен в его новых защищенных версиях
- Оно устраняет проблему, существующую в активных соединениях, когда устройство клиента находится за NAT
- Никакого преимущества нет, например, в протоколе HTTP одно соединение, как управляющие так и по которому получают данные

ПКС-1.3. Владеет современными языками и средствами разработки программного обеспечения в конкретных предметных областях

1 Какие получатся результаты при отправке отправителем 100 байт на получателя при SOCK STREAM соединении, при этом на получателе стоит вызов функций

```
int x1 = recv(s,buf,70);
int x2 = recv(s,buf,50);
int x3 = recv(s,buf,30);
```

- = x1=70, x2=30, x3 =0, если отправитель закроет соединение послав FIN
- x1=70, x2=30, x3 =0
- x1=70, x2=50, x3 =30
- x1=70, x2=50, третий recv будет ожидать

2 Что реализует ниже представленный код, написанный на языке python?

```
import socket
import ssl
from socket import *
mailserver = 'smtp.mail.ru'
cSock = socket(AF_INET, SOCK_STREAM)
cSock.connect((mailserver, 465))
cSockSSL = ssl.wrap_socket(cSock)
recv = cSockSSL.recv(1024)
print(recv)
cSockSSL.send("EHLO host\r\n")
recv = cSockSSL.recv(1024)
print(recv)
cSockSSL.close()
cSock.close()
```

- = позволяет реализовать отправку команды протокола ESMTP поверх защищенного соединения
- позволяет реализовать отправку команды протокола ESMTP поверх защищенного соединения, с использованием механизма START TLS
- позволяет отправить команду на сервер smtp.mail.ru, получить ответ один раз и зарывает соединение
- создает TCP незащищенное соединение и отправляет команду EHLO почтовому серверу
-