**Вопросы к контрольной**

1. ***Назначение интерфейса Named Pipe***

Все функции ***Named Pipe API*** можно разбить на три группы:

- функции управления каналом (создать канал, соединить сервер с каналом, открыть канал, получить информацию об именованном канале, получить состояние канала, изменить характеристики канала);

- функции обмена данными (писать в канал, читать из канала, копировать данные канала);

- функции для работы с транзакциями.

Именованный канал представляет собой объект операционной системы Windows, позволяющий создавать между распределенными в локальной TCP/IP-сети процессами дуплексные и полудуплексные каналы, по которым может осуществляться передача данных в синхронном или асинхронном режимах.

1. ***Назначение и основные принципы работы Proxy-сервера***

**Прокси-сервер** (от англ. proxy — «представитель, уполномоченный») — служба в компьютерных сетях, позволяющая клиентам выполнять косвенные запросы к другим сетевым службам.

***Использование***

1. ***Обеспечение доступа с компьютеров локальной сети в Интернет***
2. ***Кэширование данных***
3. ***Сжатие данных***
4. ***Защита локальной сети от внешнего доступа***
5. ***Ограничение доступа из локальной сети к внешней***
6. ***Анонимизация доступа к различным ресурсам.***
7. ***Назначение интерфейса Mailslot***
8. Механизм Mailslots (почтовый ящик) является одним из IPC-механизмов операционной системы Windows, позволяющий создавать распределенные приложения архитектуры клиент-сервер в локальной сети TCP/IP.
9. Почтовый ящик представляет собой объект операционной системы, предоставляющий возможность пересылать данные в одном направлении: от клиента к серверу.
10. Почтовый ящик идентифицируется своим именем. Сервером называется процесс создающий почтовый ящик. Клиентом – процесс, который подключается к почтовому ящику и записывает в него данные.
11. Обмен данными осуществляется сообщениями и может происходить в синхронном и асинхронном режимах. Если клиент и сервер находятся на разных компьютерах, доставка сообщений не гарантируется.
12. Допускается создание нескольких ящиков с одним и тем же именем. Если пересылаемые сообщения не превышают 425 байт, то возможна передача данных одновременно нескольким почтовым ящикам.
13. В состав Mailslots API входят функции создания почтового ящика, подсоединения клиента к почтовому ящику, функции записи и чтения сообщений, а также функции для получения и установки характеристик почтового ящика.
14. ***Поясните понятие «Фальцовка » и «Расфольцовка»***

Для удобства и с учетом ограничения размеров строки (998/78 символов), значение поля может быть разбито на несколько строк; это называется «фальцовкой» (folding).

Процесс преобразования фальцованного многострочного представления поля в обычное однострочное называется расфальцовкой (unfolding) и выполняется путем простого удаления всех последовательностей CRLF, непосредственно за которыми следуют пробельные символы (WSP)

1. ***Назначение и основные принципы работы (с указанием уровней модели ISO/OSI) межсетевого экрана***

***Межсетевой экран или сетевой экран*** — комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами

Основной задачей сетевого экрана является защита компьютерных сетей или отдельных узлов от несанкционированного доступа. Также сетевые экраны часто называют фильтрами, так как их основная задача — не пропускать (фильтровать) пакеты, не подходящие под критерии, определённые в конфигурации.

В зависимости от уровня, на котором происходит контроль доступа, существует разделение на сетевые экраны, работающие на:

1. **сетевом** уровне, когда фильтрация происходит на основе адресов отправителя и получателя пакетов, номеров портов транспортного уровня модели OSI и статических правил, заданных администратором;
2. **сеансовом уровне** (также известные как stateful) — отслеживающие сеансы между приложениями, не пропускающие пакеты нарушающих спецификации TCP/IP, часто используемых в злонамеренных операциях — сканировании ресурсов, взломах через неправильные реализации TCP/IP, обрыв/замедление соединений, инъекция данных.
3. **уровне приложений**, фильтрация на основании анализа данных приложения, передаваемых внутри пакета. Такие типы экранов позволяют блокировать передачу нежелательной и потенциально опасной информации на основании политик и настроек.

Некоторые решения, относимые к сетевым экранам уровня приложения, представляют собой прокси-серверы с некоторыми возможностями сетевого экрана, реализуя прозрачные прокси-серверы, со специализацией по протоколам. Возможности прокси-сервера и многопротокольная специализация делают фильтрацию значительно более гибкой, чем на классических сетевых экранах, но такие приложения имеют все недостатки прокси-серверов (например, анонимизация трафика)

1. ***Последовательное выполнение при вызове удаленной процедуры***
2. Процедура клиента вызывает клиентскую заглушку.
3. Клиентская заглушка создает сообщение и вызывает службу RPC локальной операционной системы.
4. Служба RPC пересылает сообщение службе RPC удаленной операционной системе.
5. Служба RPC удаленной операционной системы вызывает серверную заглушку и передает ей сообщение.
6. Серверная заглушка извлекает из сообщения параметры и вызывает удаленную процедуру.
7. Удаленная процедура исполняет код и возвращает параметры и значение возврата серверной заглушке.
8. Серверная заглушка формирует сообщение и вызывает службу RPC своей локальной операционной системе.
9. Служба RPC своей локальной операционной системы сервера пересылает сообщение службе RPC локальной операционной системы клиента.
10. Служба RPC локальной операционной системы клиента возвращает сообщение клиентской заглушке.
11. Заглушка извлекает результаты и возвращает их.
12. ***Какие ограничения вносит спецификация REC5322(формат сообщения Internet) на размерность сообщений***

Сообщения, соответствующие данной спецификации, включают символы с десятичными кодами от 1 до 127, интерпретируемые в соответствии с кодировкой US-ASCII

Данная спецификация вносит два ограничения на число символов в строке. Строка должна содержать не более 998 символов; следует использовать строки размером не более 78, без учета CRLF.

Поля заголовков представляют собой строки, начинающиеся с имени поля, за которым следует двоеточие (":"), содержимое поля и знак завершения строки CRLF. Имя поля должно состоять только из печатаемых символов US-ASCII (т. е., символов с кодами от 33 до 126, включительно), исключая двоеточие. Значение поля может включать печатаемые символы US-ASCII, символы пробела (SP, код ASCII - 32) и горизонтальной табуляции.

1. ***Назначение и основные принципы работы NAT-устройства***

Существует 3 базовых концепции трансляции адресов: статическая (Static Network Address Translation), динамическая (Dynamic Address Translation), маскарадная (NAPT, NAT Overload, PAT).

*Статический NAT* — Отображение незарегистрированного IP-адреса на зарегистрированный IP-адрес на основании один к одному. Особенно полезно, когда устройство должно быть доступным снаружи сети.

*Динамический NAT* — Отображает незарегистрированный IP-адрес на зарегистрированный адрес от группы зарегистрированных IP-адресов. Динамический NAT также устанавливает непосредственное отображение между незарегистрированным и зарегистрированным адресом, но отображение может меняться в зависимости от зарегистрированного адреса, доступного в пуле адресов, во время коммуникации.

*Перегруженный NAT* (NAPT, NAT Overload, PAT, маскарадинг) — форма динамического NAT, который отображает несколько незарегистрированных адресов в единственный зарегистрированный IP-адрес, используя различные порты. Известен также как PAT (Port Address Translation). При перегрузке каждый компьютер в частной сети транслируется в тот же самый адрес, но с различным номером порта.

*NAT выполняет три важных функции.*

1. Позволяет сэкономить IP-адреса (только в случае использования NAT в режиме PAT), транслируя несколько внутренних IP-адресов в один внешний публичный IP-адрес (или в несколько, но меньшим количеством, чем внутренних). По такому принципу построено большинство сетей в мире: на небольшой район домашней сети местного провайдера или на офис выделяется 1 публичный (внешний) IP-адрес, за которым работают и получают доступ интерфейсы с приватными (внутренними) IP-адресами.
2. Позволяет предотвратить или ограничить обращение снаружи ко внутренним хостам, оставляя возможность обращения изнутри наружу. При инициации соединения изнутри сети создаётся трансляция. Ответные пакеты, поступающие снаружи, соответствуют созданной трансляции и поэтому пропускаются. Если для пакетов, поступающих снаружи, соответствующей трансляции не существует (а она может быть созданной при инициации соединения или статической), они не пропускаются.
3. Позволяет скрыть определённые внутренние сервисы внутренних хостов/серверов. По сути, выполняется та же указанная выше трансляция на определённый порт, но возможно подменить внутренний порт официально зарегистрированной службы (например, 80-й порт TCP (HTTP-сервер) на внешний 54055-й). Тем самым, снаружи, на внешнем IP-адресе после трансляции адресов на сайт (или форум) для осведомлённых посетителей можно будет попасть по адресу http://example.org:54055, но на внутреннем сервере, находящемся за NAT, он будет работать на обычном 80-м порту. Повышение безопасности и скрытие «непубличных» ресурсов.
4. ***Опишите процесс получения элементов IP-адресов DHCP-сервера с указанием протоколов, адресов и получателей***

DHCP. Протокол, предназначенный для автоматизации конфигурирования стека TCP/IP на узлах. С его помощью можно назначить узлам IP адрес, маску подсети, адрес шлюза по умолчанию, сконфигурировать многие другие настраиваемые параметры стека TCP/IP, сделав соответствующие настройки на сервере, а не на каждом узле.

DHCP использует принцип «аренды» или определенного времени, в течение которого IP-адрес будет действителен для компьютера. Это время может различаться, в зависимости от того, как долго пользователю требуется сетевое подключение в определенной точке.

Протокол DHCP предоставляет три способа распределения IP-адресов:

1. Ручное распределение. При этом способе сетевой администратор сопоставляет аппаратному адресу (для Ethernet сетей это MAC-адрес) каждого клиентского компьютера определённый IP-адрес. Фактически, данный способ распределения адресов отличается от ручной настройки каждого компьютера лишь тем, что сведения об адресах хранятся централизованно (на сервере DHCP), и потому их проще изменять при необходимости.
2. Автоматическое распределение. При данном способе каждому компьютеру на постоянное использование выделяется произвольный свободный IP-адрес из определённого администратором диапазона.
3. Динамическое распределение. Этот способ аналогичен автоматическому распределению, за исключением того, что адрес выдаётся компьютеру не на постоянное пользование, а на определённый срок. Это называется арендой адреса. По истечении срока аренды IP-адрес вновь считается свободным, и клиент обязан запросить новый (он, впрочем, может оказаться тем же самым). Кроме того, клиент сам может отказаться от полученного адреса.

Некоторые реализации службы DHCP способны автоматически обновлять записи DNS, соответствующие клиентским компьютерам, при выделении им новых адресов. Это производится при помощи протокола обновления DNS, описанного в RFC 2136.

1. ***Назначение записей «А» и «Мх» в таблице DNS***

|  |  |  |
| --- | --- | --- |
| **Тип ресурсной записи** | **Функция записи** | **Описание испоьзования** |
| A | Host Address  Адрес хоста, или узла | Отображает имя узла на IP-адрес  (например, для домена *microsoft.com* узлу с именем *www.microsoft.com* сопоставляется IP-адрес с помощью такой записи:  www A 207.46.199.60) |
| CNAME | Canonical Name (alias)  Какноническое имя (псевдоним) | Отображает одно имя на другое |
| MX | Mail Exchanger  Обмен почтой | Управляет маршрутизация почтовых сообщений для протокола SMTP |
| NS | Name Server  Сервер имен | Указывает на серверы DNS, ответственные за конкретный домен и его поддомены |
| PTR | Pointer  Указатель | Используется для обратного разрешения IP-адресов в имена узлов в домене *in-addr.arpa* |
| SOA | Start of Authority  Начальная запись зоны | Используется для указания основного сервера для данной зоны и описания свойств зоны |
| SRV | Service Locator  Указатель на службу | Используется для поиска серверов, на которых функционируют определенные службы (например, контроллеры доменов Active Directory или серверы глобального каталога) |

1. ***Назначение и основные принципы работы ремейтеров***

**Ремейлер** (англ. remailer) — это сервер, получающий сообщение электронной почты и переправляющий его по адресу, указанному отправителем. В процессе переадресации вся информация об отправителе уничтожается, поэтому конечный получатель лишён возможности выяснить, кто является автором сообщения

Ремейлеры делятся на анонимные и псевдо-анонимные.

При использовании псевдо-анонимного ремейлера, его оператор знает адрес электронной почты, который необходим для получения ответа на письмо. Тайна связи полностью зависит от оператора, который может стать жертвой угроз, шантажа или социальной инженерии. Преимуществом псевдо-анонимных ремейлеров является их юзабилити, за которое пользователь расплачивается меньшей защищённостью.

Анонимные ремейлеры обеспечивают гораздо более высокую секретность, но при этом они и сложнее в использовании. Их операторы не могут знать, какие данные пересылаются через них, а поэтому нет гарантии своевременной доставки сообщения, которое может и вовсе затеряться. В обмен на высокое время ожидания анонимные ремейлеры достаточно надёжно скрывают от посторонних глаз реальный адрес и содержимое сообщения.

1. ***Поясните понятие «Маршалинг параметров»***

Маршалинг – процесс передачи (копирования) параметров из адресного пространства клиента в адресное пространство компонента (ЕХЕ-модуля).

Маршалинг (marshaling) — это упаковка всех параметров и возвращаемых значений и передача их через границы процессов. СОМ поддерживает так называемый стандартный маршалинг, который надежно работает для большинства объектов, значительно снижая требования к программной реализации и делая процесс мар-шалинга полностью прозрачным.

Маршалинг (от англ. marshal — упорядочивать), по смыслу похож на сериализацию, в информатике — процесс преобразования представления объекта в памяти в формат данных, пригодный для хранения или передачи. Обычно применяется, когда данные необходимо передавать между различными частями одной программы или от одной программы к другой.

Маршалинг задействуется при использовании различных механизмов RPC, где есть необходимость в передаче данных между процессами и потоками. Переход от неуправляемого типа в тип CLR, как, например, в процессах P/Invoke, используемых в платформе .NET Framework, является подходящим примером, демонстрирующим применение маршалинга.

Кроме того, маршалинг широко используется в скриптах и приложениях, применяющих технологии XPCOM, поставляемой в составе Mozilla application framework. Браузер Mozilla Firefox — одно из самых известных приложений, созданных с применением этой платформы, что позволило скриптовым языкам использовать технологию XPCOM через Cross-Platform Connect (XPConnect).