Информационная безопасность

1. Модели безопасности ОС. Дискреционные и мандатные модели доступа.

Модели безопасности ОС

Модель безопасности операционной системы – это формальное описание политики безопасности, определяющее правила и ограничения доступа субъектов (пользователей, процессов) к объектам (файлам, устройствам, процессам).

Основные компоненты моделей безопасности:

- **Субъекты** – активные сущности, осуществляющие доступ к информации
- **Объекты** – пассивные сущности, к которым осуществляется доступ
- **Операции** – действия, выполняемые субъектами над объектами (чтение, запись, выполнение)
- **Правила** – условия, определяющие возможность выполнения операций

Дискреционные модели доступа (DAC - Discretionary Access Control)

Дискреционная модель основана на принципе назначения прав доступа к объектам по усмотрению их владельцев.

Характеристики дискреционных моделей:

- Владелец объекта определяет права доступа к нему для других субъектов
- Информация о правах доступа хранится в матрице доступа
- Проверка доступа осуществляется при каждом обращении к объекту
- Права доступа могут передаваться (делегироваться) другим субъектам

Матрица доступа представляет собой таблицу, где строки соответствуют субъектам, столбцы – объектам, а на пересечении указаны разрешенные операции (r – чтение, w – запись, x – выполнение).

На практике матрица доступа реализуется в виде:

- Списков контроля доступа (ACL) – для каждого объекта хранится список субъектов с их правами
- Списков возможностей – для каждого субъекта хранится список объектов с правами на них

Модели типа Харисона–Рузо–Ульмана

Модель Харисона–Рузо–Ульмана (HRU) формализует изменение состояний системы с дискреционным управлением доступом.

Основные особенности:

- Представляет систему как набор состояний, определяемых матрицей доступа
- Определяет примитивные операции для изменения матрицы доступа
- Позволяет анализировать безопасность системы при изменении прав доступа
- Доказывает, что проблема безопасности в общем случае алгоритмически неразрешима

Мандатные модели доступа (MAC - Mandatory Access Control)

Мандатная модель основана на централизованном контроле доступа, осуществляемом системой, а не пользователями.

Характеристики мандатных моделей:

- Информации и субъектам присваиваются метки безопасности (уровни доступа)
- Доступ определяется на основе сравнения меток
- Пользователи не могут изменять метки безопасности и политику доступа
- Обеспечивается строгий иерархический контроль информационных потоков

Модели типа Белла–Лападулы

Модель Белла–Лападулы (BLP) – классическая модель мандатного контроля доступа, направленная на обеспечение конфиденциальности.

Основные принципы:

- **Простое свойство безопасности (No Read Up)**: субъект может читать только объекты с уровнем доступа не выше своего
- **Свойство * (No Write Down)**: субъект может записывать данные только в объекты с уровнем доступа не ниже своего

Эти принципы предотвращают утечку информации от высокоуровневых субъектов к низкоуровневым.

Ролевая модель (RBAC - Role-Based Access Control)

Ролевая модель основана на доступе к ресурсам через роли, а не напрямую.

Характеристики ролевой модели:

- Пользователям назначаются роли
- Ролям назначаются права доступа к объектам
- Пользователи получают доступ к объектам через роли
- Упрощается администрирование доступа в больших системах

SELinux

SELinux (Security-Enhanced Linux) – реализация мандатного контроля доступа для Linux, разработанная АНБ США.

Основные особенности:

- Обеспечивает принцип наименьших привилегий
- Использует контексты безопасности для субъектов и объектов
- Разделяет политику безопасности и механизм её реализации
- Поддерживает различные модели безопасности
- Реализует контроль на уровне типов (Type Enforcement)

SELinux имеет три режима работы:

- **Enforcing** – политика безопасности принудительно применяется
- **Permissive** – нарушения регистрируются, но не блокируются
- **Disabled** – SELinux отключен

2. Критерии безопасности информационных систем. Стандарты безопасности информационных систем.

Критерии безопасности информационных систем

Критерии безопасности – это набор требований, используемых для оценки уровня защищенности информационных систем (ИС). Они определяют необходимые механизмы защиты и методы их верификации.

Основные критерии безопасности ИС:

1. **Конфиденциальность** – защита от несанкционированного доступа к информации
    - Разграничение доступа
    - Шифрование
    - Контроль информационных потоков
2. **Целостность** – защита от несанкционированной модификации информации
    - Контроль целостности данных
    - Электронная подпись
    - Журналирование изменений
3. **Доступность** – обеспечение доступа к информации и системам для авторизованных пользователей
    - Отказоустойчивость
    - Резервное копирование
    - Предотвращение DoS-атак
4. **Неотказуемость** – невозможность отрицания факта отправки или получения информации
    - Цифровая подпись
    - Аудит действий
5. **Подотчетность** – однозначное прослеживание действий пользователя в системе

- Идентификация и аутентификация
- Регистрация событий

Стандарты безопасности информационных систем

Стандарты безопасности – это формализованные наборы требований и рекомендаций, применяемые для обеспечения защиты информационных систем.

Международные стандарты:

1. **ISO/IEC 27000** – семейство стандартов по управлению информационной безопасностью
    - ISO/IEC 27001 – требования к системам управления информационной безопасностью (СУИБ)
    - ISO/IEC 27002 – свод практик для управления информационной безопасностью
    - ISO/IEC 27005 – управление рисками информационной безопасности
2. **Common Criteria (ISO/IEC 15408)** – стандарт оценки безопасности информационных технологий
    - Определяет 7 уровней доверия (EAL1-EAL7)
    - Устанавливает требования к функциональности и гарантиям безопасности
    - Обеспечивает механизм для признания сертификации разными странами
3. **PCI DSS** (Payment Card Industry Data Security Standard) – стандарт безопасности данных индустрии платежных карт
    - Определяет требования к безопасности при обработке, передаче и хранении данных о банковских картах
    - Включает 12 обязательных требований

Российские стандарты:

1. **ГОСТ Р ИСО/МЭК 15408** – российский аналог Common Criteria
2. **Руководящие документы ФСТЭК России**
    - Руководящий документ "Защита от несанкционированного доступа к информации. Термины и определения"
    - Руководящий документ "Автоматизированные системы. Защита от несанкционированного доступа к информации. Классификация автоматизированных систем и требования по защите информации"
    - Руководящий документ "Средства вычислительной техники. Защита от несанкционированного доступа к информации. Показатели защищенности от несанкционированного доступа к информации"
3. **Приказы ФСТЭК России**
    - Приказ №21 "Об утверждении Состава и содержания организационных и технических мер по обеспечению безопасности персональных данных при их обработке в информационных системах персональных данных"

- Приказ №17 "Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах"

Эволюция стандартов безопасности:

1. **Оранжевая книга** (TCSEC) – первый значимый стандарт безопасности компьютерных систем, разработанный Министерством обороны США
   - Классы безопасности: D, C1, C2, B1, B2, B3, A1
   - Фокус на конфиденциальность и мандатное управление доступом
2. **Европейские критерии** (ITSEC) – европейский подход к оценке безопасности ИТ
   - Разделение функциональности (F) и уровней гарантии (E)
   - Более гибкий и широкий подход, чем TCSEC
3. **Common Criteria** – объединение и развитие TCSEC и ITSEC
   - Профили защиты (PP) – требования безопасности для категории продуктов
   - Задания по безопасности (ST) – требования безопасности для конкретного продукта
   - Цели безопасности (TOE) – предмет оценки

3. Применение межсетевых экранов для защиты корпоративных сетей

Межсетевые экраны для защиты корпоративных сетей

Межсетевой экран (МЭ, файрвол, firewall) – это система или комбинация систем, обеспечивающая защитный барьер между различными сетевыми средами и реализующая политику разграничения доступа между ними.

Функции межсетевых экранов:

- Фильтрация сетевого трафика
- Разграничение доступа между сетями
- Сокрытие внутренней структуры сети (NAT)
- Мониторинг и регистрация событий
- Кэширование данных (для прокси-серверов)
- Аутентификация доступа к ресурсам

Виды межсетевых экранов:

1. **Пакетные фильтры**
   - Фильтрация на сетевом и транспортном уровнях
   - Анализ IP-адресов, портов, флагов протоколов
   - Быстрая работа, низкое потребление ресурсов
   - Ограниченные возможности анализа

2. **Шлюзы сеансового уровня (Stateful Inspection)**
    - Отслеживание состояния сеансов связи
    - Создание динамических таблиц соединений
    - Более высокий уровень защиты, чем у пакетных фильтров
    - Большее потребление ресурсов
3. **Шлюзы прикладного уровня (прокси-серверы)**
    - Анализ трафика на прикладном уровне
    - Понимание специфики протоколов (HTTP, FTP, SMTP)
    - Наиболее полная защита
    - Высокое потребление ресурсов, снижение производительности
4. **Инспекторы состояния**
    - Комбинирование возможностей пакетных фильтров и прикладных шлюзов
    - Высокая производительность
    - Хорошая степень защиты

Пакетный фильтр на базе ОС Linux

В Linux встроена подсистема фильтрации пакетов **iptables** (для IPv4) и **ip6tables** (для IPv6), заменённые в новых версиях на **nftables**. В Ubuntu 20.04 и новее, а также в RHEL8, используется **firewalld** как интерфейс к nftables/iptables.

Основные компоненты iptables:

- **Таблицы** (tables) – группы цепочек с определенным назначением (filter, nat, mangle, raw)
- **Цепочки** (chains) – последовательности правил, применяемых к пакетам (INPUT, OUTPUT, FORWARD)
- **Правила** (rules) – условия и действия для обработки пакетов (ACCEPT, DROP, REJECT, LOG)

Пример базовой настройки iptables:

```
# Очистка текущих правил
iptables -F

# Установка политик по умолчанию
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT ACCEPT

# Разрешение локальных соединений
iptables -A INPUT -i lo -j ACCEPT

# Разрешение установленных соединений
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

# Разрешение SSH
iptables -A INPUT -p tcp --dport 22 -j ACCEPT

# Разрешение веб-сервера
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT

# Сохранение правил
iptables-save > /etc/iptables/rules.v4
```

Фильтрация пакетов: параметры и правила фильтрации

При настройке правил фильтрации используются следующие параметры:

1. **Сетевые адреса**
   - Источник (source) и получатель (destination)
   - Могут задаваться как отдельные адреса, диапазоны или подсети
2. **Порты**
   - Порт источника (source port) и порт назначения (destination port)
   - Могут задаваться как отдельные порты, диапазоны или списки
3. **Протоколы**
   - TCP, UDP, ICMP и другие
   - Для каждого протокола могут быть свои специфические параметры
4. **Флаги TCP**
   - SYN, ACK, FIN, RST, PSH, URG
   - Важны для определения состояния соединения
5. **Интерфейсы**
   - Входящий (in) и исходящий (out)
   - Позволяют фильтровать пакеты по интерфейсу

Правила фильтрации:

1. **Правила должны быть однозначными и не противоречить друг другу**
2. **Правила обрабатываются по порядку до первого срабатывания**
3. **Порядок правил критически важен**:
    - Более специфичные правила должны предшествовать более общим
    - Наиболее часто используемые правила лучше размещать в начале для повышения производительности
4. **Типичные правила**:
    - Блокирование известных вредоносных IP-адресов
    - Разрешение доступа только к определенным сервисам
    - Блокирование непроверенных входящих соединений
    - Ограничение скорости соединений (защита от DoS)

Шлюзы прикладного уровня

Шлюзы прикладного уровня (Application-level gateways, ALG) – межсетевые экраны, которые работают на прикладном уровне модели OSI.

Характеристики:

- Полностью разрывают соединение между клиентом и сервером
- Анализируют содержимое пакетов с учетом специфики протоколов
- Обеспечивают аутентификацию пользователей
- Предоставляют кэширование и оптимизацию трафика

Примеры шлюзов прикладного уровня:

- **Squid** – прокси-сервер HTTP, HTTPS, FTP
- **NGINX** – может использоваться как обратный прокси
- **Microsoft ISA/TMG** – корпоративный прокси-сервер
- **HAProxy** – специализированный прокси для балансировки нагрузки

Противодействие сетевым атакам при помощи межсетевых экранов

1. Защита от сканирования портов

- Блокирование большого количества неудачных попыток соединения
- Ограничение скорости новых соединений
- Скрытие внутренней структуры сети с помощью NAT

2. Защита от DoS/DDoS-атак

- Ограничение числа одновременных соединений с одного IP
- Ограничение скорости соединений
- Блокирование известных паттернов атак
- Фильтрация неправильно сформированных пакетов

3. Защита от атак уровня приложений

- Глубокий анализ трафика (Deep Packet Inspection)
- Обнаружение и фильтрация атак типа SQL-инъекций, XSS
- Проверка корректности форматов данных

4. Защита от инсайдерских угроз

- Ограничение исходящего трафика по протоколам и портам
- Мониторинг и регистрация подозрительной активности
- Применение VPN для удаленного доступа

5. Построение комплексной защиты

- Многоуровневая защита (defense-in-depth)
- Комбинирование различных типов межсетевых экранов
- Интеграция с системами обнаружения/предотвращения вторжений (IDS/IPS)
- Регулярное обновление правил фильтрации

4. Электронные цифровые подписи. Система PGP. Система S/MIME.

Электронные цифровые подписи

Электронная цифровая подпись (ЭЦП) – реквизит электронного документа, обеспечивающий проверку его целостности и подтверждающий подлинность. ЭЦП получается в результате криптографического преобразования информации с использованием закрытого ключа.

Принципы работы ЭЦП:

1. **Формирование подписи**:
    - Вычисление хеш-функции от исходного документа
    - Шифрование полученного хеша закрытым ключом отправителя
2. **Проверка подписи**:
    - Расшифровка подписи с использованием открытого ключа отправителя
    - Вычисление хеш-функции от полученного документа
    - Сравнение расшифрованного и вычисленного хешей

Свойства ЭЦП:

- **Аутентичность** – подтверждение авторства документа
- **Целостность** – обнаружение изменений в документе
- **Неотказуемость** – невозможность отрицания авторства
- **Невозможность подделки** – защита от создания поддельной подписи

Используемые типы криптографических примитивов

1. Алгоритмы хеширования:

- **MD5** – 128-битный хеш (устарел, не рекомендуется)
- **SHA-1** – 160-битный хеш (устарел, не рекомендуется)
- **SHA-2** (SHA-256, SHA-384, SHA-512) – современные алгоритмы
- **SHA-3** – новейшее семейство хеш-функций

2. Алгоритмы асимметричного шифрования:

- **RSA** – алгоритм, основанный на сложности факторизации больших чисел
- **DSA** (Digital Signature Algorithm) – стандарт подписи США
- **ECDSA** (Elliptic Curve DSA) – эллиптическая криптография
- **ГОСТ Р 34.10** – российский стандарт ЭЦП

Система PGP (Pretty Good Privacy)

PGP – система шифрования и подписи электронной почты и файлов, разработанная Филом Циммерманном.

Особенности PGP:

- **Гибридная криптосистема** – использует симметричное шифрование для данных и асимметричное для ключей
- **Web of Trust** – модель доверия, основанная на сети взаимных подтверждений подлинности ключей
- **Открытый стандарт** – спецификация OpenPGP (RFC 4880)
- **Многоплатформенность** – реализации для различных ОС

Процесс подписи в PGP:

1. Вычисление хеша сообщения (SHA-1, SHA-2)
2. Шифрование хеша закрытым ключом отправителя (RSA, DSA, ECDSA)
3. Добавление подписи к сообщению (открытое или вложенное)

Процесс шифрования в PGP:

1. Генерация случайного сеансового ключа для симметричного шифрования
2. Шифрование сообщения сеансовым ключом (AES, CAST, 3DES)
3. Шифрование сеансового ключа открытым ключом получателя
4. Объединение зашифрованного сообщения и зашифрованного сеансового ключа

Система S/MIME (Secure/Multipurpose Internet Mail Extensions)

S/MIME – стандарт для шифрования и подписи MIME-данных (в основном электронной почты), основанный на инфраструктуре открытых ключей (PKI).

Особенности S/MIME:

- **Иерархическая модель доверия** – основана на сертификатах X.509

- **Интеграция с почтовыми клиентами** – поддерживается большинством популярных почтовых программ
- **Международный стандарт** – IETF RFC 5751
- **Использование CA** – сертификационные центры удостоверяют подлинность ключей

Используемые алгоритмы:

1. **Для хеширования**:
   - SHA-1 (устаревший)
   - SHA-256, SHA-384, SHA-512
2. **Для цифровой подписи**:
   - RSA
   - DSA
   - ECDSA
3. **Для шифрования**:
   - RSA (асимметричное)
   - AES, 3DES (симметричное)

Процесс работы S/MIME:

1. **Подписание**:
   - Создание хеша содержимого
   - Шифрование хеша закрытым ключом отправителя
   - Формирование MIME-пакета с подписью
2. **Шифрование**:
   - Генерация случайного ключа для симметричного шифрования
   - Шифрование содержимого симметричным ключом
   - Шифрование симметричного ключа открытым ключом получателя
   - Формирование MIME-пакета с зашифрованными данными
3. **Подтверждение подлинности**:
   - Проверка сертификата отправителя через центр сертификации
   - Расшифровка подписи с помощью открытого ключа отправителя
   - Сравнение хешей

Сравнение PGP и S/MIME

Характеристика	PGP	S/MIME
Модель доверия	Web of Trust (децентрализованная)	PKI (централизованная)
Сертификаты	Самостоятельно подписанные	X.509 от CA

Характеристика	PGP	S/MIME
Интеграция	Часто требует дополнительных плагинов	Встроена в большинство почтовых клиентов
Стандартизация	OpenPGP (RFC 4880)	RFC 5751
Гибкость	Высокая	Средняя
Удобство для пользователя	Может быть сложным	Относительно простое
Распространенность	Среднее	Высокое в корпоративной среде

5. Инфраструктура открытых ключей. Техники управления ключами. Основные концепции.

Инфраструктура открытых ключей (PKI)

Инфраструктура открытых ключей (Public Key Infrastructure, PKI) – комплекс технических средств, организационных мер и нормативно-методического обеспечения для создания и управления сертификатами открытых ключей.

Компоненты PKI:

1. **Удостоверяющий центр (Certificate Authority, CA)**
    - Выпускает и подписывает сертификаты
    - Ведет списки отозванных сертификатов (CRL)
    - Является доверенной стороной для всех участников
2. **Центр регистрации (Registration Authority, RA)**
    - Проверяет личность заявителей
    - Формирует запросы на сертификаты для CA
    - Выполняет первичную валидацию данных
3. **Хранилище сертификатов (Certificate Repository)**
    - Обеспечивает публичный доступ к сертификатам
    - Обычно реализуется через LDAP или HTTP
4. **Центр валидации (Validation Authority, VA)**
    - Проверяет действительность сертификатов
    - Поддерживает протоколы OCSP и CRL
5. **Конечные пользователи (End Entities)**
    - Запрашивают и используют сертификаты
    - Выполняют криптографические операции

Сертификаты открытых ключей X.509:

Сертификат X.509 – стандартизированная структура данных, которая связывает открытый ключ с идентификационной информацией о его владельце.

Основные поля сертификата:

- **Версия** – версия формата сертификата
- **Серийный номер** – уникальный идентификатор
- **Алгоритм подписи** – идентификатор алгоритма подписи
- **Издатель** – информация об удостоверяющем центре
- **Срок действия** – период валидности (с/по)
- **Субъект** – информация о владельце ключа
- **Информация об открытом ключе** – алгоритм и значение ключа
- **Расширения** – дополнительные поля и ограничения
- **Цифровая подпись** – подпись издателя (CA)

Техники управления ключами

Управление ключами – набор процессов и технологий для создания, хранения, распределения, использования, архивирования и уничтожения криптографических ключей.

1. Генерация ключей

Методы генерации ключей:

- **Аппаратные генераторы случайных чисел** (HRNG)
- **Программные генераторы псевдослучайных чисел** (PRNG)
- **Комбинированные методы**

Требования к процессу:

- Достаточная случайность/энтропия
- Соответствие требованиям алгоритма
- Безопасная среда генерации

2. Распределение ключей

Методы распределения ключей:

- **Физическое распределение** – передача ключей на физическом носителе
- **Распределение с помощью центра** – доверенная третья сторона
- **Прямой обмен** – протоколы согласования ключей (Диффи-Хеллмана)
- **Распределение через PKI** – использование сертификатов

3. Хранение ключей

Способы хранения:

- **Аппаратные модули безопасности (HSM)**
- **Смарт-карты и USB-токены**

- **Защищенные хранилища ключей**
- **Разделение секрета** (схема Шамира)

4. Обновление ключей

Процедуры обновления:

- **Периодическое обновление** – регулярная замена ключей
- **Обновление по требованию** – замена при подозрении компрометации
- **Изменение длины ключа** – увеличение при росте вычислительных мощностей

5. Отзыв ключей

Механизмы отзыва:

- **Списки отозванных сертификатов (CRL)** – периодически публикуемые списки
- **Онлайн-протокол проверки статуса (OCSP)** – проверка в реальном времени
- **Сертификаты с коротким сроком действия** – самоотзыв по истечении срока

6. Архивирование и восстановление ключей

Методы архивирования:

- **Резервное копирование в защищенном хранилище**
- **Эскроу ключей** – хранение копий у доверенной третьей стороны
- **Шифрование архивов ключей**

Основные концепции PKI

1. Иерархическая модель доверия

Организация PKI в виде дерева, где:

- **Корневой CA** – верхний уровень доверия
- **Промежуточные CA** – подчиненные центры сертификации
- **Конечные сертификаты** – листья дерева

Преимущества:

- Четкая структура управления
- Масштабируемость
- Локализация компрометации

2. Сетевая модель доверия (Web of Trust)

Децентрализованная модель, используемая в PGP:

- Пользователи сами подтверждают подлинность ключей друг друга
- Формируется сеть взаимных доверительных отношений
- Нет единой точки отказа

3. Кросс-сертификация

Установление доверительных отношений между различными PKI:

- Взаимная сертификация корневых CA
- Создание мостовых CA
- Построение федеративной инфраструктуры

4. Политики сертификатов

Набор правил, определяющих:

- Процедуры идентификации субъектов
- Требования к генерации и хранению ключей
- Процедуры выпуска и отзыва сертификатов
- Сроки действия сертификатов
- Области применения ключей

5. Жизненный цикл сертификата

Этапы:

- **Регистрация** – проверка личности и данных
- **Выпуск** – создание и подписание сертификата
- **Распространение** – публикация в хранилище
- **Использование** – криптографические операции
- **Обновление** – продление срока действия
- **Отзыв** – досрочное прекращение действия
- **Архивирование** – хранение истекших сертификатов

6. Протоколы PKI

Основные протоколы:

- **PKCS#10** – формат запроса на сертификат
- **PKCS#7/CMS** – формат подписанных/зашифрованных данных
- **PKCS#12** – формат для хранения и обмена личными ключами
- **X.509** – формат сертификатов
- **CRL** – списки отозванных сертификатов
- **OCSP** – онлайн-проверка статуса сертификатов

6. Характеристика и механизмы удаленных атак на распределённые вычислительные системы.

Характеристика удаленных атак на распределённые вычислительные системы

Удаленные атаки на распределенные вычислительные системы – это атаки, осуществляемые злоумышленником из удаленной точки сети на целевую систему без

физического доступа к ней.

Особенности удаленных атак на распределенные системы:

1. **Масштабный охват** – атаки могут быть направлены на несколько компонентов системы одновременно
2. **Многовекторность** – комбинирование различных уязвимостей и точек входа
3. **Распределенный характер** – атака может исходить из разных источников
4. **Сложность обнаружения** – из-за распределенности системы сложнее заметить аномалии
5. **Каскадный эффект** – компрометация одного компонента может привести к компрометации других

Классификация удаленных атак:

1. **По цели атаки**:
    - Нарушение конфиденциальности данных
    - Нарушение целостности данных
    - Нарушение доступности (Denial of Service)
    - Повышение привилегий
2. **По используемым уязвимостям**:
    - Уязвимости протоколов
    - Уязвимости программного обеспечения
    - Уязвимости конфигурации
    - Уязвимости в механизмах аутентификации
3. **По уровню взаимодействия**:
    - Пассивные (только сбор информации)
    - Активные (изменение данных или процессов)

Механизмы удаленных атак

1. Атаки отказа в обслуживании (DoS/DDoS)

Механизмы:

- **Флуд-атаки** – перегрузка системы множеством запросов
    - SYN-флуд – эксплуатация трехстороннего рукопожатия TCP
    - UDP-флуд – отправка множества UDP-пакетов
    - ICMP-флуд – перегрузка системы ICMP-запросами
- **Амплификационные атаки** – использование протоколов с усилением
    - DNS-амплификация
    - NTP-амплификация
    - SSDP-амплификация
- **Распределенные атаки (DDoS)** – атака из множества источников
    - Ботнеты

- Атаки с использованием отражателей

2. Атаки на уровне приложений

Механизмы:

- **Инъекции** – внедрение вредоносного кода
  - SQL-инъекции
  - XML-инъекции
  - Command-инъекции
- **Межсайтовый скриптинг (XSS)** – внедрение JavaScript-кода
  - Отраженный XSS
  - Хранимый XSS
  - DOM-based XSS
- **Межсайтовая подделка запросов (CSRF)**
- **Атаки на API** – эксплуатация недостатков проектирования API
- **Атаки на уровне сессий** – перехват и подделка сессионных токенов

3. Атаки "человек посередине" (MITM)

Механизмы:

- **ARP-спуфинг** – подмена ARP-таблиц
- **DNS-спуфинг** – подмена DNS-ответов
- **SSL/TLS-атаки**
  - SSL-stripping – понижение протокола HTTPS до HTTP
  - POODLE, BEAST, CRIME – атаки на протокол SSL/TLS
- **Перехват Wi-Fi** – создание поддельных точек доступа

4. Атаки на компоненты инфраструктуры

Механизмы:

- **Атаки на протоколы маршрутизации**
  - BGP hijacking – перехват и перенаправление трафика
  - Атаки на OSPF
- **Атаки на DNS**
  - Cache poisoning – отравление кеша DNS
  - Zone transfers – несанкционированное копирование зоны
- **Атаки на NTP** – манипуляции с временной синхронизацией

5. Эксплуатация уязвимостей ПО

Механизмы:

- **Использование известных уязвимостей**
  - Zero-day уязвимости

- Атаки на непропатченное ПО

- **Переполнение буфера** – выход за границы выделенной памяти
- **Использование уязвимостей десериализации**
- **Атаки на бизнес-логику приложений**

Характеристика и механизмы удаленных атак на хосты Internet

Хосты Internet – это компьютеры или устройства, подключенные к сети Internet и предоставляющие или использующие сетевые сервисы.

Характеристики атак на хосты Internet:

1. **Постоянная подверженность** – устройство доступно из Интернета 24/7
2. **Широкая поверхность атаки** – множество сервисов и портов
3. **Автоматизация атак** – использование ботов для сканирования и атак
4. **Множественность источников** – атаки приходят из разных стран и сетей

Механизмы атак на хосты Internet:

1. **Сканирование портов и сервисов**
   - Последовательное сканирование
   - SYN-сканирование
   - UDP-сканирование
   - Определение версий служб и ОС (fingerprinting)
2. **Атаки на уязвимые сервисы**
   - Эксплуатация уязвимостей веб-серверов
   - Атаки на FTP, SSH, SMTP и другие сервисы
   - Эксплуатация уязвимостей CMS (WordPress, Joomla и др.)
3. **Атаки на аутентификацию**
   - Брутфорс паролей
   - Словарные атаки
   - Credential stuffing – использование украденных учетных данных
   - Атаки на механизмы восстановления паролей
4. **Заражение вредоносным ПО**
   - Внедрение через уязвимости
   - Фишинг и социальная инженерия
   - Атаки типа "водопой" (watering hole)
   - Загрузка вредоносного ПО через скомпрометированные сайты

Системы обнаружения атак

Системы обнаружения атак (IDS - Intrusion Detection System) и системы предотвращения атак (IPS - Intrusion Prevention System) – это программно-аппаратные средства, предназначенные для обнаружения и предотвращения сетевых атак.

Типы систем обнаружения атак:

1. **По месту размещения**:
   - **Сетевые (NIDS)** – анализируют сетевой трафик
   - **Хостовые (HIDS)** – анализируют активность на конкретном компьютере
   - **Гибридные** – комбинируют оба подхода
2. **По методу анализа**:
   - **Сигнатурные** – ищут известные шаблоны атак
   - **Аномальные** – выявляют отклонения от нормального поведения
   - **Гибридные** – комбинация сигнатурного и аномального подходов

Основные функции систем обнаружения атак:

1. **Мониторинг**:
   - Сбор и анализ сетевого трафика
   - Мониторинг активности хостов
   - Проверка целостности файлов
2. **Обнаружение**:
   - Выявление известных атак по сигнатурам
   - Обнаружение аномалий
   - Поведенческий анализ
3. **Реагирование**:
   - Оповещение администраторов
   - Блокирование атак (для IPS)
   - Сбор доказательств
   - Автоматическое изменение конфигурации защиты

Примеры систем обнаружения атак:

1. **Open Source решения**:
   - **Snort** – сигнатурная NIDS/NIPS
   - **Suricata** – высокопроизводительная NIDS/NIPS
   - **OSSEC** – HIDS с функциями мониторинга целостности
   - **Wazuh** – расширенная версия OSSEC
2. **Коммерческие решения**:
   - **Cisco Secure IDS/IPS**
   - **Palo Alto Networks NGFW**
   - **McAfee Network Security Platform**
   - **Trend Micro Deep Security**
3. **Облачные и управляемые решения**:
   - **AWS GuardDuty**
   - **Azure Security Center**

- **Google Cloud Armor**

7. Идентификация и аутентификация, управление доступом.

Идентификация и аутентификация

Идентификация – процесс предъявления идентификатора (имени, номера, токена), который однозначно определяет пользователя или систему.

Аутентификация – процесс проверки подлинности предъявленного идентификатора, то есть подтверждение того, что субъект действительно является тем, за кого себя выдает.

Типы аутентификации:

1. **По знанию (what you know)**
   - Пароли
   - PIN-коды
   - Кодовые фразы
   - Ответы на секретные вопросы
2. **По владению (what you have)**
   - Смарт-карты
   - USB-токены
   - Физические ключи
   - Мобильные устройства (для получения OTP)
3. **По биометрии (what you are)**
   - Отпечатки пальцев
   - Распознавание лица
   - Сканирование сетчатки глаза
   - Голосовая аутентификация
   - Поведенческая биометрия (динамика нажатия клавиш, подпись)
4. **По местоположению (where you are)**
   - GPS-координаты
   - Принадлежность к сети
   - IP-адрес

Механизмы аутентификации:

1. **Парольная аутентификация**
   - Хранение паролей в хешированном виде
   - Солирование паролей
   - Политики сложности паролей
   - Защита от подбора (throttling, CAPTCHA)
2. **Многофакторная аутентификация (MFA)**

- Комбинация двух или более факторов
- Повышение безопасности по сравнению с однофакторной
- Примеры: пароль + SMS, пароль + приложение-аутентификатор

3. **Одноразовые пароли (OTP)**
   - Временные (TOTP) – генерируются на основе времени
   - Событийные (HOTP) – генерируются на основе счетчика
   - Механизм доставки: SMS, email, приложения
4. **Сертификаты X.509**
   - Аутентификация на основе PKI
   - Взаимная аутентификация клиента и сервера
   - Использование в TLS/SSL
5. **Протоколы аутентификации**
   - Kerberos – аутентификация на основе билетов
   - OAuth 2.0 – делегирование доступа
   - SAML – обмен аутентификационными данными между доменами
   - OpenID Connect – надстройка над OAuth 2.0 для аутентификации

Управление доступом

Управление доступом – процесс регулирования и контроля доступа субъектов (пользователей, процессов) к объектам (файлам, устройствам, сервисам) в соответствии с политикой безопасности.

Модели управления доступом:

1. **Дискреционное управление доступом (DAC)**
   - Владелец объекта определяет права доступа
   - Доступ определяется на основе идентификатора пользователя
   - Примеры: файловые права в UNIX, ACL в Windows
2. **Мандатное управление доступом (MAC)**
   - Доступ определяется на основе меток безопасности
   - Централизованный контроль со стороны системы
   - Примеры: SELinux, AppArmor
3. **Ролевое управление доступом (RBAC)**
   - Доступ определяется на основе ролей пользователей
   - Роли связаны с набором разрешений
   - Пользователи назначаются на роли
   - Упрощает администрирование в крупных системах
4. **Атрибутное управление доступом (ABAC)**
   - Доступ определяется на основе атрибутов
   - Учитываются атрибуты субъекта, объекта, действия и среды

- Более гибкий подход по сравнению с RBAC
- Примеры: XACML

Механизмы управления доступом:

1. **Списки контроля доступа (ACL)**
   - Для каждого объекта хранится список субъектов и их прав
   - Гибкий, но сложный в управлении при большом количестве объектов
2. **Матрицы доступа**
   - Таблица, строки которой соответствуют субъектам, столбцы – объектам
   - На пересечении – разрешенные операции
3. **Возможности (Capabilities)**
   - Для каждого субъекта хранится список объектов и прав доступа к ним
   - Эффективнее при проверке прав для конкретного субъекта
4. **Многоуровневая защита**
   - Комбинирование различных механизмов
   - Реализация принципа "глубокой защиты"

Принципы управления доступом:

1. **Принцип наименьших привилегий**
   - Субъект должен иметь только необходимый минимум прав
2. **Разделение обязанностей**
   - Критические операции требуют участия нескольких субъектов
3. **Обязательная аутентификация**
   - Доступ предоставляется только после аутентификации
4. **Подотчетность действий**
   - Каждое действие субъекта должно быть зарегистрировано

Протоколирование и аудит, шифрование, контроль целостности

Протоколирование и аудит

Протоколирование (логирование) – процесс записи информации о событиях, происходящих в системе.

Аудит – процесс анализа записей о событиях для обнаружения нарушений безопасности.

Типы событий для протоколирования:

1. **События безопасности**

- Успешные и неуспешные попытки аутентификации
- Изменения в политиках безопасности
- Изменения привилегий пользователей

2. **Системные события**
    - Загрузка и остановка системы
    - Установка и удаление ПО
    - Системные ошибки
3. **События приложений**
    - Действия пользователей в приложениях
    - Ошибки приложений
    - Транзакции и изменения данных

Механизмы протоколирования:

- **Локальное протоколирование** – запись в локальные файлы
- **Централизованное протоколирование** – отправка логов на выделенный сервер
- **Защищенное протоколирование** – обеспечение целостности и конфиденциальности логов

Инструменты аудита:

- **SIEM-системы** (Security Information and Event Management)
- **Анализаторы логов**
- **Системы обнаружения вторжений**
- **Средства аналитики безопасности**

Шифрование

Шифрование – процесс преобразования данных в форму, недоступную для чтения без знания ключа.

Типы шифрования:

1. **Симметричное шифрование**
    - Один ключ для шифрования и дешифрования
    - Высокая скорость, но проблема распространения ключей
    - Алгоритмы: AES, 3DES, ChaCha20
2. **Асимметричное шифрование**
    - Пара ключей: открытый и закрытый
    - Медленнее симметричного, но решает проблему распространения ключей
    - Алгоритмы: RSA, ECC, DSA
3. **Гибридное шифрование**
    - Комбинация симметричного и асимметричного шифрования
    - Используется в большинстве современных систем (TLS, PGP)

Применение шифрования:

- **Шифрование данных в покое** – защита хранимых данных
- **Шифрование данных в передаче** – защита передаваемых данных
- **Шифрование на уровне файловой системы** – LUKS, BitLocker, FileVault
- **Шифрование на уровне приложений** – PGP, S/MIME
- **Шифрование на транспортном уровне** – TLS/SSL

Контроль целостности

Контроль целостности – проверка неизменности данных, обнаружение несанкционированных модификаций.

Механизмы контроля целостности:

1. **Хеш-функции**
    - Создание "отпечатка" данных фиксированной длины
    - Определение изменений: любое изменение данных изменяет хеш
    - Алгоритмы: SHA-256, SHA-3, BLAKE2
2. **Коды аутентификации сообщений (MAC)**
    - Хеш-функция с ключом
    - Обеспечивает аутентификацию источника и целостность
    - Алгоритмы: HMAC, CMAC
3. **Цифровые подписи**
    - Использование асимметричной криптографии
    - Обеспечивает целостность, аутентификацию и неотказуемость
    - Алгоритмы: RSA-PSS, ECDSA, EdDSA
4. **Системы обнаружения изменений файлов**
    - Мониторинг изменений файлов в системе
    - Периодическая проверка хешей критичных файлов
    - Примеры: AIDE, Tripwire, OSSEC

Применение контроля целостности:

- **Проверка целостности ПО** – верификация установленных программ
- **Защита конфигурационных файлов** – обнаружение неавторизованных изменений
- **Обеспечение целостности передаваемых данных** – обнаружение изменений при передаче
- **Верификация цифровых артефактов** – проверка подлинности скачанных файлов
- **Контроль целостности в блокчейне** – обеспечение неизменности цепочки блоков