

Раздел 1. Основы защиты информации в компьютерных сетях

Тема 1.1. Защита информации в сетях: актуальность, проблемы, подходы к их решению

Место курса среди других дисциплин учебного плана. Предмет и задачи курса. Современная ситуация и проблемы в области информационной безопасности.

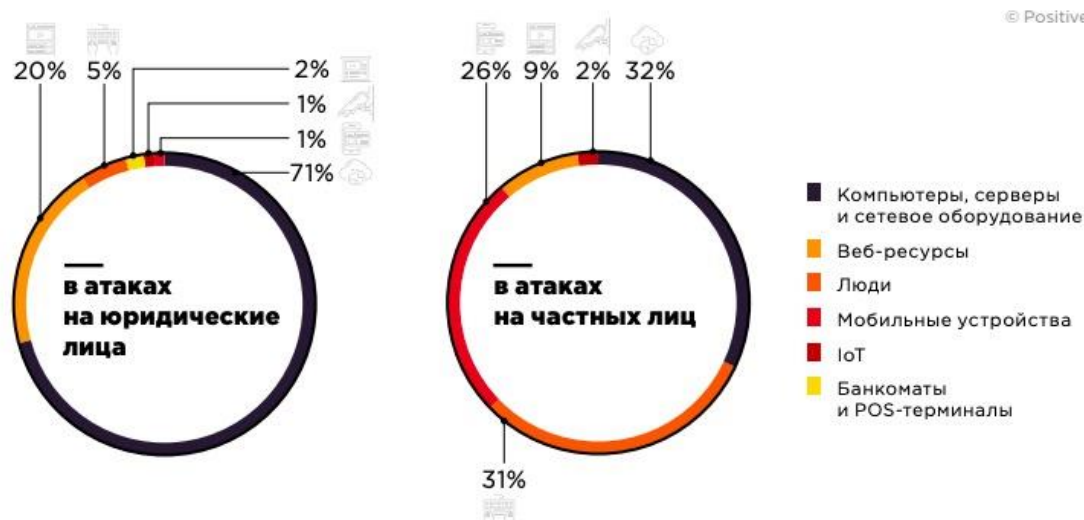
Эволюция и развитие локальных компьютерных сетей привели к глобализации сетей передачи данных и появлению Интернета, предоставляющему пользователям компьютеров новые возможности для оперативного обмена информацией в повседневной жизни практически каждого человека.

По мере развития и усложнения средств, методов и форм автоматизации процессов обработки информации повышается зависимость общества от степени безопасности используемых им информационных технологий. Сетевой обмен данными является неотъемлемой частью большинства информационных сервисов и автоматизированных систем и любые нарушения штатной работы этого процесса могут привести к полной остановке сервиса или АС.

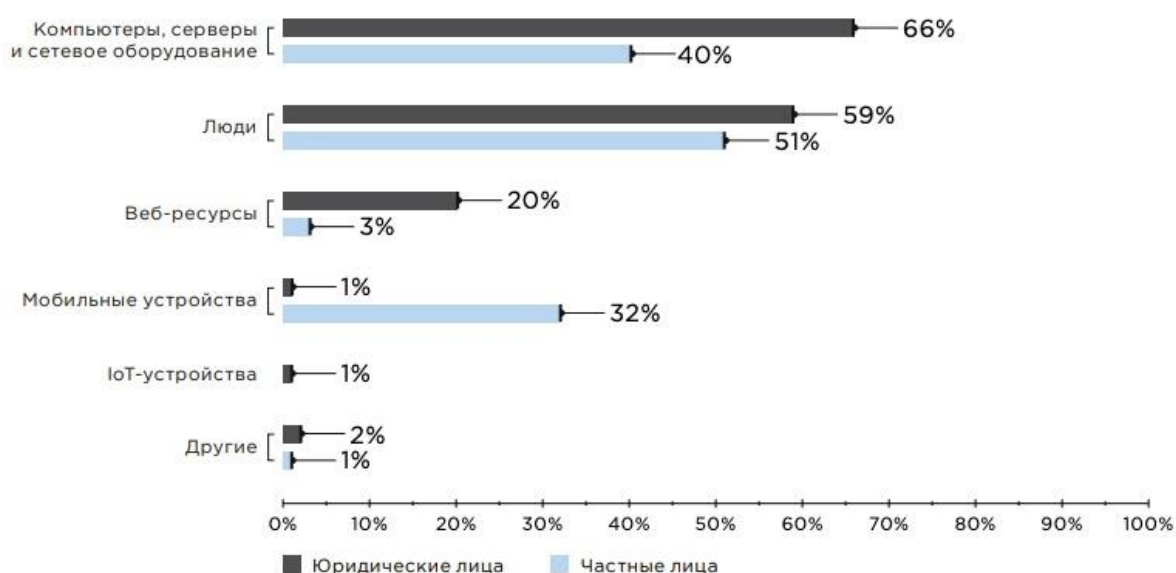
Современные методы обработки, передачи и накопления информации способствовали появлению угроз, связанных с возможностью потери, искажения и раскрытия данных, адресованных или принадлежащих конечным пользователям, организациям и целым государствам. Поэтому, сегодня на передний план вышли проблемы защиты компьютерных сетей, которые являются не менее важными, чем проблемы защиты программного обеспечения компьютера, которым традиционно уделяется больше внимания.

Чтобы усилить абстрактность понятия «важный» следует привести статистические данные, ведь количество киберинцидентов с каждым годом и месяцем продолжает расти. На конец 2019 года, по данным компании Postive Technologies, доля реализованных атак на сетевую инфраструктуру составила рекордные 71% на сети юридических лиц и 32% на частных лиц. К примеру, к концу 2017 года доля атак на сетевую инфраструктуру составляла лишь 51%. А уже по итогам 1 полугодия 2020 года атаки на домашние сети частных лиц выросла на 8%.

Киберпреступники преследовали самые разные цели - от установки майнеров до кибершпионажа в сетях крупных компаний.



Доля реализованных атак на IT-объекты в 2019 году

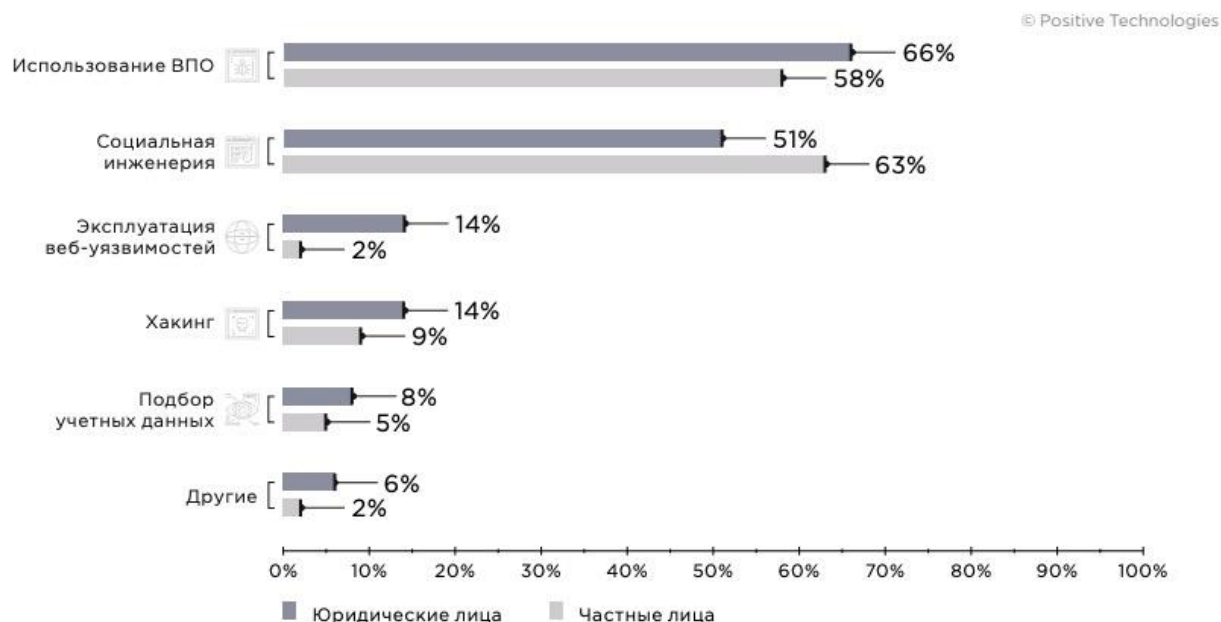


Доля реализованных атак на IT-объекты за 6 месяцев 2020 года

Примечательными являются также данные по используемым методам атак. С 2017 года значительно выросла доля использования вредоносного ПО и социальной инженерии, в то время как компрометации учетных данных уменьшилась более чем в 2 раза, а количество DDoS-атак и вовсе перешли в общий разряд «другие».

Не смотря на со временем уменьшающуюся и относительно небольшую долю в 14%, хакинг является наиболее опасным вектором проникновения в государственные учреждения и частные компании во всем мире. Для реализации атак хакеры используют уязвимости ПО и недостатки механизмов защиты. Здесь всё очевидно: после публикации информации о серьезной уязвимости злоумышленники незамедлительно пытаются использовать ее в атаках. В 2019-2020 годах было обнаружено сразу несколько критических

уязвимостей, позволяющих удаленно выполнять код без авторизации. Широко известны случаи проникновения в сети муниципальных и финансовых учреждений в США, а также энергетической компании Elexon (Великобритания) в указанный период.



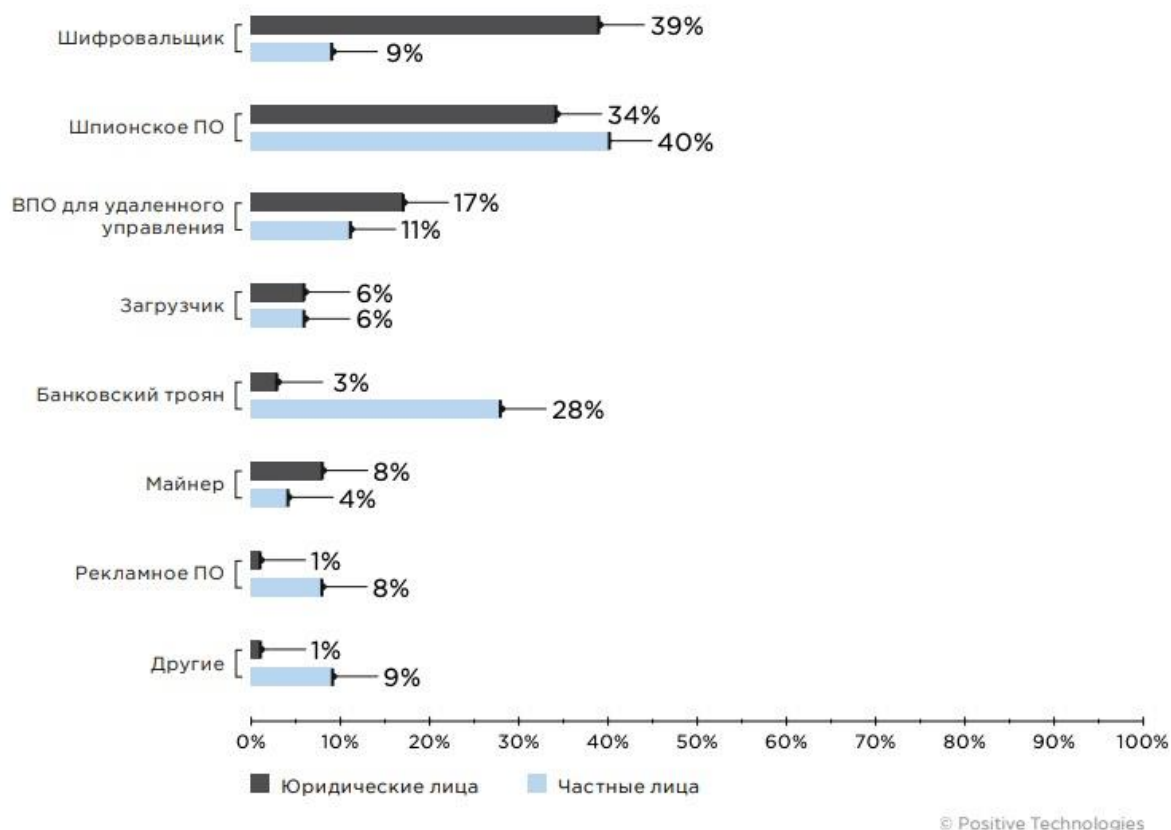
Доля используемых методов атак (по итогам 2019 года)

Для общего понимания приведу также сведения по способам распространения ВПО. Как видно из диаграмм наибольшую нишу занимает электронная почта, где через фишинговые рассылки наиболее часто атакуется энергетический и промышленный сектор стран СНГ.

Способы распространения ВПО



Видов ВПО огромное множество, но стоит перечислить наиболее популярные, в число которых по-прежнему входят шифровальщики, программы-шпионы, программы удаленного управления, различные загрузчики, adware-софт и, с недавних пор, трояны-майнеры.



Доля ВПО

В качестве итога, вся представленная на слайдах статистика свидетельствует о важности и месте защиты компьютерных сетей в комплексных системах обеспечения информационной безопасности.

Основной целью изучения дисциплины является обучение студентов технологиям, принципам, методам и средствам защиты данных и программного обеспечения от различных типов угроз с привлечением программных и аппаратных средств защиты, а также аналитического анализа серьезности угроз информационной безопасности сети.

Основными задачами дисциплины являются:

- изучение основных понятий и положений проведения исследований, связанных с обеспечением информационной безопасности в компьютерных сетях;
- знакомство с программными и программно-аппаратными средствами защиты программ и данных от типовых угроз информационной безопасности в сети;
- знакомство с основными подходами к активному аудиту компьютерных сетей.

В данном курсе будут рассмотрены основные понятия и определяющие факторы информационных угроз и уязвимостей компьютерных сетей; технологии защиты информации в сетях; основные средства защиты информации в компьютерных сетях, такие как межсетевые экраны, системы обнаружения атак, виртуальные частные сети (VPN) и др.; отдельным

разделом будут рассмотрены системы анализа защищенности компьютерных сетей на основе различных сканеров безопасности.

Тема 1.2. Угрозы ИБ в компьютерных сетях

Основные понятия: угрозы, уязвимости, инциденты.

Новые информационные технологии активно внедряются во все сферы народного хозяйства. Появление локальных и глобальных сетей передачи данных предоставило пользователям компьютеров новые возможности для взаимодействия. Развитие Интернета привело к использованию глобальных сетей передачи данных в повседневной жизни практически всех людей на планете. По мере развития и усложнения средств, методов и форм автоматизации процессов обработки информации повышается зависимость общества от степени безопасности используемых им информационных технологий.

Рассмотрим основные понятия защиты информации и информационной безопасности компьютерных систем и сетей с учетом определений стандарта ГОСТ Р 50922-96.

ГОСТ Р 50922-96 «Защита информации. Основные термины и определения» устанавливает основные термины и определения понятий в области защиты информации. Термины, установленные настоящим стандартом, обязательны для применения во всех видах документации и литературы по защите информации, входящих в сферу работ по стандартизации и (или) использующих результаты этих работ.

Защита информации (ЗИ) – деятельность по предотвращению утечки защищаемой информации, несанкционированных и непреднамеренных воздействий на защищаемую информацию

Объект защиты – информация, или носитель информации, или информационный процесс, в отношении которого необходимо обеспечивать защиту в соответствии с поставленной целью ЗИ.

Цель защиты информации – это желаемый результат защиты информации. Целью ЗИ может быть предотвращение ущерба собственнику, владельцу, пользователю информации в результате возможной утечки информации и/или несанкционированного и непреднамеренного воздействия на информацию.

Эффективность ЗИ – степень соответствия результатов ЗИ поставленной цели.

Защита информации от утечки – деятельность по предотвращению неконтролируемого распространения защищаемой информации от ее разглашения, НСД к защищаемой информации и от получения защищаемой информации злоумышленниками.

Защита информации от разглашения – деятельность по предотвращению несанкционированного доведения защищаемой информации до неконтролируемого количества получателей информации.

Защита информации от несанкционированного воздействия – деятельность по предотвращению воздействия на защищаемую информацию с нарушением установленных прав и правил на изменение информации, приводящего к искажению, уничтожению, копированию, блокированию доступа к информации, а также к утрате, уничтожению или сбою функционирования носителя информации.

Защита информации от непреднамеренного воздействия – деятельность по предотвращению воздействия на защищаемую информацию ошибок пользователя информацией, сбоя технических и программных средств информационных систем, а также природных явлений или иных нецеленаправленных на изменение информации воздействий, связанных с функционированием технических средств, систем или с деятельностью людей и приводящих к искажению, уничтожению, копированию, блокированию доступа к информации, а также к утрате, уничтожению или сбою функционирования носителя информации.

Защита от несанкционированного доступа (НСД) – деятельность по предотвращению получения защищаемой информации заинтересованным субъектом с нарушением установленных правовыми документами или собственником, владельцем информации прав или правил доступа к защищаемой информации. Заинтересованным субъектом, осуществляющим несанкционированный доступ к защищаемой информации, может выступать государство, юридическое лицо, группа физических лиц, в т.ч. общественная организация, отдельное физическое лицо.

Под **информационной безопасностью** (сетевой безопасностью) понимают защищенность информации от незаконного ознакомления, преобразования и уничтожения, а также защищенность информационных ресурсов от воздействий, направленных на нарушение их работоспособности. Примеры воздействий: попытки проникновения злоумышленников, ошибки персонала, выход из строя аппаратных и программных средств, стихийные бедствия и т.д.

Информационная безопасность компьютерных систем достигается обеспечением *конфиденциальности, целостности и достоверности* обрабатываемых данных, а также доступности информационных компонентов и ресурсов системы.

Конфиденциальность данных – это статус, предоставленный данным и определяющим требуемую степень их защиты. К конфиденциальным данным можно отнести, например, следующие: личная информация пользователей, учетные записи; данные о кредитных картах; данные о разработках и различные внутренние документы; бухгалтерские сведения.

Конфиденциальная информация должна быть известна только допущенным и прошедшим проверку (авторизованным) субъектам системы (пользователям, процессам, программам). Для остальных субъектов системы эта информация должна быть неизвестной.

Установление градаций важности защищаемой информации называют категоризированием защищаемой информации.

Целостность информации – это свойство информации сохранять свою структуру и/или содержание в процессе передачи и хранения. Целостность обеспечивается, если данные не отличаются в семантическом отношении от данных в исходных документах, т.е. если не произошло их случайного или преднамеренного искажения или разрушения. Обеспечение целостности данных является одной из сложных задач ЗИ.

Достоверность информации – свойство, выражающееся в строгой принадлежности субъекту, которой является ее источником, либо тому субъекту, от которого эта информация принята.

Доступность данных – возможность доступа к информации.

Различают санкционированный и несанкционированный доступ к информации.

Санкционированный доступ – это доступ к информации, не нарушающий установленные правила разграничения доступа. Правила разграничения доступа служат для регламентации права доступа к компонентам системы.

Несанкционированный доступ (НСД) к информации характеризуется нарушением установленных правил разграничения доступа. Лицо или процесс, осуществляющие НСД, являются нарушителями правил разграничения доступа. НСД является наиболее распространенным видом компьютерных нарушений.

С допуском к информации и ресурсам системы связан группа таких важных понятий, как идентификация, аутентификация, авторизация.

С каждым субъектом системы (сети) связывают некоторую информацию, идентифицирующую субъект. Эта информация является идентификатором субъекта. Субъект, имеющий зарегистрированный идентификатор, является законным (легальным) субъектом.

Таким образом, **идентификация субъекта** – это процедура распознавания субъекта по его идентификатору. Идентификация выполняется при попытке субъекта войти в систему (сеть).

Следующий шаг взаимодействия – **аутентификация субъекта** – проверка подлинности субъекта с данным идентификатором. Процедура аутентификации устанавливает, является ли субъект именно тем, кем он себя объявил.

После идентификации и аутентификации выполняется процедура **авторизации субъекта** – предоставление законному субъекту, успешно

прошедшему идентификацию и аутентификацию, соответствующих полномочий и доступных ресурсов системы (сети).

Уязвимость компьютерной системы – это присущее системе неудачное свойство, которое может привести к реализации угрозы. Например, отсутствие антивируса в системе, не установлены последние обновления и т.д.

С понятием уязвимости компьютерной системы (сети) тесно связано понятие угрозы безопасности. Под **угрозой безопасности АС** понимаются возможные действия, способные прямо или косвенно нанести ущерб ее безопасности. *Ущерб безопасности подразумевает нарушение состояния защищенности информации, содержащейся и обрабатываемой в системе (сети).* Другими словами, **ущерб** – негативное влияние на систему, оказываемое проведенной атакой.

Риск – это одновременное наличие угрозы и соответствующей ей уязвимости в системе.

Атака на компьютерную систему – это поиск и/или использование злоумышленником той или иной уязвимости системы. Иными словами, атака – это реализация угрозы безопасности, т.е. **киберинцидент**.

Противодействие угрозам безопасности является целью средств защиты компьютерных систем и сетей.

Средство защиты информации – техническое, программное средство, вещество и/или материал, предназначенные или используемые для защиты информации.

Комплекс средств защиты (КСЗ) представляет собой совокупность программных и технических средств, создаваемых и поддерживаемых для обеспечения информационной безопасности системы (сети).

Политика безопасности – это совокупность норм, правил и практических рекомендаций, регламентирующих работу средств защиты компьютерной системы от заданного множества угроз.

Защищенность является одним из важнейших показателей эффективности функционирования АС, наряду с такими показателями как *надежность, отказоустойчивость, производительность* и т. п.

Под **защищенностью АС** будем понимать степень адекватности реализованных в ней механизмов защиты информации существующим в данной среде функционирования рискам, связанным с осуществлением угроз безопасности информации.

Определяющие факторы и классификация угроз.

Под **угрозами безопасности информации** традиционно понимается возможность нарушения таких свойств информации, как конфиденциальность, целостность и доступность.

На практике всегда существует большое количество неподдающихся точной оценке возможных путей осуществления угроз безопасности в отношении ресурсов АС. В идеале каждый путь осуществления угрозы должен быть перекрыт соответствующим механизмом защиты. Данное условие является **первым фактором**, определяющим защищенность АС. **Вторым фактором** является прочность существующих механизмов защиты, характеризующаяся степенью сопротивляемости этих механизмов попыткам их обхода либо преодоления. **Третьим фактором** является величина ущерба, наносимого владельцу АС в случае успешного осуществления угроз безопасности.

На практике получение точных значений приведенных характеристик затруднено, т.к. понятия угрозы, ущерба и сопротивляемости механизма защиты трудноформализуемы.

Для современных информационных технологий подсистемы защиты являются неотъемлемой частью АС обработки информации. Атакующая сторона должна преодолеть эту подсистему защиты, чтобы нарушить, например, конфиденциальность АС. Надо понимать, что не существует абсолютно стойкой системы защиты, вопрос лишь во времени и средствах, требуемых на ее преодоление.

Под **утечкой информации** подразумевается получение несанкционированным субъектом доступа к конфиденциальной информации.

Под **модификацией** подразумевается несанкционированное изменение информации, либо подмена источника информации.

Под **отказом в доступе** подразумевается невозможность получения авторизованным субъектом доступа к информации в установленные сроки.

Для противодействия утечкам информации необходимо использовать методы и средства, повышающие конфиденциальность информации.

Защита от модификации информации может обеспечиваться средствами обеспечения целостности информации, например, защитой от несанкционированных изменений.

Достоверность источника информации или его аутентичность может быть обеспечена путем введения СЗИ идентификации и аутентификации.

Механизмы обеспечения доступности ресурсов системы повышают ее устойчивость от возможных отказов в доступе к сети.

Классификация угроз.

Угроза ИБ реализуется в результате образования канала реализации угроз между источником угрозы и носителем, что создает условия для нарушения одного или нескольких критериев ИБ.

Основными элементами канала реализации угроз ИБ являются:

- источник угроз ИБ – субъект, материальный объект или физическое явление, создающие угрозы;
- среда (путь) распространения информации или воздействий, в которой физическое поле, сигнал, данные или программы могут распространяться и воздействовать на защищаемые свойства (конфиденциальность, актуальность, целостность и доступность) информации;
- носитель – физическое лицо или материальный объект, в том числе физическое поле, в котором информация находит свое отражение в виде символов, образов, сигналов, технических решений и процессов, количественных характеристик физических величин.

Перечень угроз, оценки вероятностей их реализации, а также модель нарушителя служат основой для анализа риска реализации угроз и формулирования требований к системе защиты. В настоящее время известен обширный перечень угроз информационной безопасности компьютерных систем и сетей, содержащий сотни позиций. Поэтому для защищаемой системы обычно определяют не полный их перечень, а только классы угроз по ряду базовых признаков.

1) по природе возникновения различают:

- естественные угрозы, вызванные воздействиями на АС объективных физических процессов или стихийных природных явлений;
- искусственные угрозы безопасности, вызванные деятельностью человека;

2) по степени преднамеренности проявления различают:

- угрозы, вызванные случайными ошибками или халатностью персонала, например некомпетентное использование средств защиты, ввод ошибочных данных и т.п. Причинами случайных воздействий при эксплуатации АС могут быть: аварийные ситуации из-за стихийных бедствий, отказы и сбои аппаратуры, ошибки в ПО, помехи в линиях связи и т.д. Ошибки в ПО являются самым распространённым видом компьютерных нарушений. Они и являются частыми уязвимостями приводящими к серьезным последствиям. Обычно ошибки ПО устраняются с помощью пакетов обновлений;
- угрозы преднамеренного действия, например действия злоумышленников.

3) по непосредственному источнику угроз, среди которых:

- природная среда, например стихийные бедствия, магнитные бури и пр.
- человек, например вербовка путем подкупа персонала, разглашение конфиденциальных данных и т.п.
- санкционированные программно-аппаратные средства, например удаление данных, отказ в работе операционной системы;
- несанкционированные программно-аппаратные средства, например заражение компьютера вирусами с деструктивными функциями.

4) по положению источника угроз:

- вне контролируемой зоны АС, например перехват данных, передаваемых по каналам связи, перехват побочных электромагнитных, акустических и других излучений устройств;
- в пределах контролируемой зоны АС, например применение подслушивающих устройств, хищение распечаток, записей, носителей информации и т.п.;
- непосредственно в АС, например некорректное использование ресурсов АС.

5) по степени зависимости от активности АС. Угрозы проявляются:

- независимо от активности АС, например вскрытие шифров криптозащиты информации;
- только в процессе обработки данных, например угрозы выполнения и распространения программных вирусов.

6) по степени воздействия на АС различают:

- пассивные угрозы, которые при реализации ничего не меняют в структуре и содержании АС, например угроза копирования секретных данных;
- активные угрозы, которые при воздействии вносят изменения в структуру и содержание АС, например внедрение «троянов» и вирусов.

7) по этапам доступа пользователей или программ к ресурсам АС различают:

- угрозы, проявляющиеся на этапе доступа к ресурсам АС, например угрозы НСД в АС;
- угрозы, проявляющиеся после разрешения доступа к ресурсам АС, например угрозы несанкционированного или некорректного использования ресурсов АС.

8) по способу доступа к ресурсам АС различают:

- угрозы с использованием стандартного пути доступа к ресурсам АС, например незаконное получение паролей и других реквизитов разграничения доступа с последующей маскировкой под зарегистрированного пользователя;
- угрозы с использованием скрытого нестандартно пути доступа к ресурсам АС, например НСД к ресурсам АС путем использования недокументированных возможностей ОС.

9) по текущему месту расположения информации, хранимой и обрабатываемой в АС, различают:

- угрозы доступа к информации на внешних запоминающих устройствах, например несанкционированное копирование секретной информации с жесткого диска;

- угрозы доступа к информации в оперативной памяти, например чтение остаточной информации из оперативной памяти, доступ к системной области оперативной памяти со стороны прикладных программ;

- угрозы доступа к информации, циркулирующей в линиях связи, например незаконное подключение к линиям связи с последующим вводом ложных сообщений или модификацией передаваемых сообщений; незаконное подключение к линиям связи с целью прямой подмены законного пользователя с последующим вводом дезинформации и навязыванием ложных сообщений;

- угрозы доступа к информации, отображаемой на терминале или печатаемой на принтере, например запись отображаемой информации на скрытую видеокамеру.

При рассмотрении вопросов защиты компьютерных сетей (систем, АС) целесообразно использовать четырехуровневую градацию доступа к хранимой, обрабатываемой и защищаемой системе информации, которая поможет систематизировать как возможные угрозы, так и меры по их нейтрализации. Различают следующие уровни: уровень носителей информации; уровень средств взаимодействия с носителем; уровень представления информации; уровень содержания информации.

Данные уровни были введены исходя из того, что:

- 1) информация для удобства манипулирования чаще всего фиксируется на некотором материальном носителе, которым может быть бумага, дискета или иной носитель;

- 2) Если способ представления информации таков, что она не может быть непосредственно воспринята человеком, возникает необходимость в преобразователях информации в доступный для человека способ представления. Например, для чтения с USB-флешки требуется компьютер, оборудованный USB-портом;

- 3) Как уже было отмечено, информация может быть охарактеризована способом своего представления или тем, что еще называется языком в обиходном смысле;

- 4) Человеку должен быть доступен смысл представленной информации, ее семантика.

Как отмечалось ранее, для АС рассматривают три основных вида угроз: нарушения конфиденциальности, нарушения целостности и работоспособности. Данные виды угроз считаются первичными, поскольку реализация этих угроз ведет к непосредственному воздействию на защищаемую информацию. Угрозы раскрытия параметров АС можно считать опосредованной угрозой, т.к. последствия ее реализации не причиняют какой-либо ущерб обрабатываемой информации, но дают возможность реализовать первичные угрозы, позволяя сократить временные и ресурсные затраты на преодоление системы защиты.

Для достижения требуемого уровня информационной безопасности АС необходимо обеспечить противодействие различным техническим угрозам и минимизировать возможное влияние человеческого фактора.

Основные методы реализации угроз представлены в виде таблицы (см. презентацию).

Классификация уязвимостей. Модель OSI/ISO. Стек TCP/IP.

Уязвимости.

В компьютерной безопасности термин "уязвимость" (англ. vulnerability) используется для обозначения недостатка в системе, используя который злоумышленник может намеренно нарушить её целостность и вызвать неправильную работу. Уязвимости присущи компьютерным системам, неотделимы от них и обуславливаются недостатками процесса функционирования, свойствами архитектуры автоматизированных систем, особенностями протоколов обмена и интерфейсов, применяемыми программным обеспечением и аппаратной платформой, условиями эксплуатации и расположения.

Источники угроз используют уязвимости для нарушения безопасности информации, получения незаконной выгоды (нанесения ущерба собственнику, владельцу, пользователю информации). Другими словами, любая угроза направлена на поиск и использование уязвимостей системы. В некоторых случаях злоумышленник работает наощупь, пытаясь обнаружить тот или иной дефект системы. А система реагирует на такого рода угрозы выдачей сообщений о мелких, но странных неполадках, а также флуктуациями в статистических характеристиках работы системы, на основании которых администратор сети или специалист по безопасности может заподозрить подготовку атаки. Кроме того, возможны действия источников угроз по активизации тех или иных уязвимостей, не связанные со злым умыслом.

Уязвимостями являются, например, ошибка в программе, примитивный пароль, неправильное назначение прав доступа к файлу с важными данными и множество других дефектов в разработке, эксплуатации или настройке системы. Наиболее распространенными **причинами возникновения уязвимостей** являются:

- ошибки при проектировании, разработке и эксплуатации программно-аппаратного обеспечения;
- преднамеренные действия по внесению уязвимостей в ходе проектирования, разработки и эксплуатации программно-аппаратного обеспечения;
- неправильные настройки оборудования и ПО, недопустимое изменение режимов работы устройств и программ;
- несанкционированное внедрение и использование неучтенных программ с последующим необоснованным расходом ресурсов (например, загрузка процессора, захват оперативной памяти, памяти на внешних носителях);

- внедрение вредоносных программ, создающих уязвимости в программном и программно-аппаратном обеспечении;
- несанкционированные неумышленные действия пользователей;
- сбои в работе оборудования и ПО (вызванные сбоями в электропитании, выходом из строя аппаратных элементов в результате старения и снижения надежности, внешними воздействиями электромагнитных полей технических устройств и др.).

Общая классификация уязвимостей. Уязвимости ИБ можно разделить на объективные, субъективные и случайные.

Объективные уязвимости основываются на особенностях построения и технических характеристиках оборудования и ПО, применяемых на защищаемом объекте. Полное устранение этих уязвимостей невозможно, но они могут существенно ослабляться техническими и инженерно-техническими методами парирования угроз ИБ.

Субъективные уязвимости зависят от действий субъектов (например, разработчиков оборудования и ПО, системных администраторов и пользователей организации). Уязвимости данного типа в большинстве случаев устраняются организационными и программно-аппаратными методами.

Случайные уязвимости обуславливаются особенностями окружающей объект информатизации среды и непредвиденными обстоятельствами. Многие из факторов, обеспечивающих наличие таких уязвимостей ИС, в целом предсказуемы, но полное их устранение либо невозможно, либо затруднено и достижимо только при проведении целого комплекса организационных и инженерно-технических мероприятий.

Уязвимости системы могут быть также **скрытыми**, т.е. еще не обнаруженными, они могут быть **известными**, но только теоретически, или же **общеизвестными** и активно используемыми злоумышленниками. Для общеизвестных уязвимостей в программных продуктах производители регулярно выпускают исправления, называемые патчами (заплатками). Например, компания Microsoft даже назначила специальный день – каждый второй вторник месяца, когда она объявляет о новых исправлениях в семействе ОС Windows. Многие из этих исправлений направлены на устранение уязвимостей. Однако к этой рутинной процедуре – регулярному обновлению не все и не всегда относятся с должным вниманием, из-за этого общеизвестные, но неисправленные ошибки в ПО являются одним из самых распространенных типов уязвимостей.

Другой тип уязвимостей, которыми часто пользуются злоумышленники, это **ошибки в конфигурировании** программных и аппаратных средств. Например, имена «администратор» и «гость», установленные по умолчанию во многих ОС, могут облегчить злоумышленникам доступ к системе, поэтому они должны быть сразу, при начальном конфигурировании ОС, заменены на другие, менее очевидные имена. С этой же целью администратор должен настроить подсистему интерактивного входа на то, чтобы она не отображала последнего имени пользователя, систему аудита – чтобы фиксировала все

успешные и неуспешные попытки входа пользователей, а также выполнить другие столь же простые, но необходимые настройки.

Поиск уязвимостей – важная часть задачи обеспечения безопасности. Эта работа включает в себя регулярное тестирование системы с привлечением программных инструментов. Существующие многочисленные программные средства обнаружения уязвимостей не требуют от пользователя особой квалификации, как правило, они выдают на выходе довольно длинный перечень потенциальных брешей в защите системы. Однако без включения в этот процесс человека, профессионала, обладающего знаниями обо всех аспектах функционирования системы, трудно рассчитывать на успех.

Злоумышленники в компьютерных сетях.

Вопреки распространенному мнению о том, что основную опасность для компании представляют внешние нарушители, действующие из сети Интернет, так называемые хакеры, реальная угроза современной компании исходит от внутренних нарушителей. По многочисленным исследованиям около 70-80% всех нарушений в корпоративной среде приходится на долю внутренних нарушителей.

Нарушителем в общем смысле является лицо, по ошибке, незнанию или осознанно предпринявшее попытку выполнения запрещенных операций и использующее для этого различные возможности, методы и средства. Внутренний нарушитель представляет собой легитимного сотрудника организации, имеющего определенный доступ к ее информационным ресурсам. Причем, причинами нарушений внутри организации могут быть как ошибки персонала, так и умышленные действия с их стороны. Таким образом, согласно общемировой статистике на долю внутренних нарушителей, умышленно совершающих противоправные действия, приходится около 20% всех инцидентов в компании, в то время как внешние нарушители повинны только в 5% подобных случаев.

Нарушители могут быть разбиты на две категории:

Outsiders (англ. чужой, посторонний) - это нарушители из вне, которые атакуют внутренние ресурсы корпоративной сети (удаление информации на корпоративном веб-сервере, пересылка спама через почтовый сервер и т.д.) и которые обходят МЭ и СОА для того, чтобы проникнуть во внутреннюю корпоративную сеть. Злоумышленники могут атаковать из Интернет, через модемные линии, через физическое подключение к каналам связи или из сети партнеров (поставщиков, заказчиков, дилеров и т.д.).

К внешним злоумышленникам относят:

- разведывательные службы государств;
- криминальные структуры;
- конкуренты;
- недобросовестные партнеры;
- внешние субъекты (физические лица).

Insiders (англ. свой, хорошо осведомленный человек) - это те, кто находится внутри корпоративной сети, и имеют определенный доступ к корпоративным серверам и рабочим станциям. Они включают пользователей, неправильно использующих свои привилегии, или исполняющих роль привилегированного пользователя. Эти люди изначально находятся в преимущественном положении, чем Outsiders. Поскольку они уже владеют конфиденциальной информацией о фирме, недоступной для внешних нарушителей. В отличие от внешних нарушителей, для которых в общем случае атакуемая корпоративная сеть изначально представляет «черный ящик», внутренние нарушители - это люди, которые знают, как работает фирма, и понимают ее слабости. Знают, что пароль у шефа записан на бумажке, которая лежит у него на столе, что пароль секретарши - имя ее собачки, знают, когда администратор идет пить чай и т.д.

К внутренним злоумышленникам относят:

- пользователи АС;
- персонал АС;
- сотрудники службы безопасности;
- руководители различных уровней.

Основные классы злоупотреблений в компьютерных сетях.

1. Несанкционированный доступ к сервисам или информации ограниченного доступа путем обхода систем разграничения доступа.
2. Несанкционированное использование ресурсов сети для целей, не относящихся к бизнесу организации (использование социальных сетей, посещение сайтов с музыкой, видео, порнографией и др.)
3. Прослушивание каналов связи
4. Отказ в доступе
5. Проникновение в компьютерные сети
6. Сканирование компьютерных сетей

Уязвимость компонентов распределенных АС. Типовая структура компьютерных сетей.

Аппаратные средства корпоративных сетей включают физическую среду и оборудование передачи данных. В общем случае ЛВС состоит из следующих основных структурно-функциональных элементов:

- рабочих станций;
- серверов;
- межсетевых коммуникационных узлов (шлюзов, мостов, маршрутизаторов);
- каналов связи;
- Демилитаризованные зоны;
- Интернет-ресурсы;
- Пользователи Интернет;
- Мобильные пользователи.

Рабочие станции считаются наиболее доступными компонентами сетей и именно с них могут быть предприняты наиболее многочисленные попытки совершения несанкционированных действий.

С рабочих станций осуществляется управление процессами обработки информации, запуск программ, ввод и корректировка данных, на дисках рабочих станций могут размещаться важные данные и программы обработки. На мониторы и печатающие устройства рабочих станций выводится информация при работе пользователей, выполняющих различные функции и имеющих разные полномочия по доступу к ресурсам системы.

Серверы и коммуникационное оборудование нуждаются в особой защите, поскольку наиболее привлекательны с точки зрения злоумышленников. Первые - как концентраторы больших объемов информации, вторые - как элементы, в которых осуществляется преобразование (возможно через открытую, нешифрованную форму представления) данных при согласовании протоколов обмена в различных участках сети.

Сетевые устройства служат для объединения компьютеров в локальные сети. Концентраторы в настоящее время почти полностью вытеснены коммутаторами и маршрутизаторами. Концентратор или хаб - ретранслирует входящий сигнал с одного из портов в сигнал на все остальные (подключённые) порты. Понятно, что о конфиденциальности при таком способе ретрансляции сигналов и говорить не приходится. *Работает на физическом уровне модели ISO/OSI.*

Коммутатор представляет собой более сложное сетевое устройство. И как следствие различаются по набору поддерживаемых функций. Коммутатор - хранит в памяти таблицу коммутации, в которой указывается соответствие MAC-адреса узла порту коммутатора. *Работает на канальном уровне модели ISO/OSI.*

Маршрутизаторы – это наиболее сложные модели управляемыми интеллектуальными коммутаторами и обладают собственным IP-адресом, поддержкой удаленного администрирования, средствами организации виртуальных сетей (VLAN) и развитым набором средств защиты. Маршрутизатор использует адрес получателя, указанный в заголовке пакета, и определяет по таблице маршрутизации путь, по которому следует передать данные. *Работает на сетевом уровне модели ISO/OSI.*

Каналы связи, в силу большой пространственной протяженности через неконтролируемую или слабо контролируемую территорию, представляют возможность как прямого подключения к ним, так и вмешательства в процесс передачи данных.

К типовым компонентам компьютерных сетей также можно отнести **серверы, расположенные в DMZ**; Демилитаризованная зона, ДМЗ — сегмент сети, содержащий общедоступные сервисы и отделяющий их от частных. В качестве общедоступного может выступать, например, веб-сервис: обеспечивающий его сервер, который физически размещён в локальной сети, должен отвечать на любые запросы из внешней сети, при этом другие

локальные ресурсы (например, файловые серверы, рабочие станции) необходимо изолировать от внешнего доступа.

Структура MAC-адреса

Практически у каждого устройства есть свой локальный и сетевой адрес, используемые для сетевой идентификации. Локальные (аппаратные) адреса представляют собой вшитые уникальные идентификаторы (MAC-адреса), присваиваемые каждой единице активного оборудования или некоторым их интерфейсам в компьютерных сетях Ethernet, как правило еще на заводе-изготовителе. Уникальность MAC-адресов достигается тем, что каждый производитель получает в координирующем комитете IEEE Registration Authority диапазон из 16 777 216 адресов и, по мере исчерпания выделенных адресов, может запросить новый диапазон. Поэтому по трём старшим байтам MAC-адреса можно определить производителя. Существуют таблицы, позволяющие определить производителя по MAC-адресу; в частности, они есть в свободном доступе в Интернете. Узнать MAC-адрес сетевого устройства в ОС Windows можно, например, консольной командой `ipconfig /all`. Обычно он записывается как шесть шестнадцатеричных чисел, разделенных двоеточием: 00:AB:CD:EF:11:22, хотя некоторые производители оборудования предпочитают запись вида 00-AB-CD-EF-11-22 и даже 00ab.cdef.1122.

Структура IP-адреса.

Сетевые адреса (IP-адреса) представляют собой адрес узла в компьютерной сети, построенной на основе стека протоколов TCP/IP. IP-адрес узла в протоколе IP назначается независимо от MAC-адреса узла. Этот адрес представляет собой набор чисел, состоящий из 4-х байтов (или октетов), например, 192.168.1.12. IP-адрес состоит из двух логических частей – номера сети и номера узла в сети, длина которых варьируется маской подсети.

В глобальных сетях к цифровым адресам добавились еще и знакомые всем доменные имена DNS, которые более удобно использовать в обычной жизни. Примером доменного имени может быть: bb.kai.ru

Модель OSI.

В силу приведенных ранее особенностей современных компьютерных сетей, существует значительное число различных видов угроз. Критически важно представлять себе архитектуру современных компьютерных сетей. Но для того чтобы понимать, как сетевые устройства, программное обеспечение и оборудование взаимодействуют друг с другом требуется понять стек коммуникационных протоколов, т.е. каким образом компьютеры согласуют язык общения: уровни и форму электрических сигналов, перечень сообщений и формат, методы контроля достоверности и т.д.

Организация взаимодействия устройств сети является сложной задачей. Для решения сложных задач часто используется известный универсальный прием декомпозиции, т.е. разбиение одной сложной задачи на несколько более простых задач-модулей. Развитием идеи декомпозиции является

многоуровневый подход, где исходная задача представляется в виде множества модулей, сгруппированных и упорядоченных по уровням, образующим иерархию.

К концу 70-х годов в мире уже существовало большое разнообразие в стеках коммуникационных протоколов. Однако компьютеры и другие сетевые устройства разных производителей, поддерживающие разные стеки протоколов, будучи помещенными в одну сеть, отказывались работать друг с другом. Необходимость преодоления несовместимости привела к созданию стандартной модели взаимодействия открытых систем (Open System Interconnection, OSI) организации International Standard Organization (ISO). *Модель ISO/OSI* сыграла важную роль в развитии компьютерных сетей.

Понимание модели OSI является полезным для понимания того, каким образом различные сетевые протоколы включены в мозаику компьютерных сетей. Как мы увидим далее, основная часть сетевых атак может быть отнесена к одному (или более) из уровней модели OSI.

Схема модели.

Модель OSI

Данные	Прикладной работа прикладных сервисов
Данные	Представления представление и кодирование данных
Данные	Сеансовый Управление сеансом связи
Блоки	Транспортный безопасное и надёжное соединение точка-точка
Пакеты	Сетевой Определение пути и IP (логическая адресация)
Кадры	Канальный MAC и LLC (Физическая адресация)
Биты	Физический кабель, сигналы, бинарная передача данных

Модель OSI определяет различные уровни взаимодействия систем и указывает, какие функции должен выполнять каждый уровень. Всего модель делит семь уровней: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический.

Самый верхний уровень – прикладной. На этом уровне пользователь взаимодействует с приложениями. Самый нижний уровень – физический. Этот уровень обеспечивает обмен сигналами между устройствами.

Модель открытых систем OSI				
Уровень (layer)	Тип данных	Функции уровня	Особенность адресации	Примеры протоколов
Уровни хоста (узла)	7. Прикладной (application)	Доступ к сетевым службам	URL	HTTP(S), FTP(S), RPC, POP3
	6. Представительский (presentation)	Представление (кодировка) и шифрование данных		ASCII, EBCDIC
	5. Сеансовый (session)	Управление сеансом связи		RAP
	4. Транспортный (transport)	Прямая связь между конечными пунктами и надёжность	Порт	TCP, UDP, SCTP, PORTS
Уровни связи (сети)	3. Сетевой (network)	Определение маршрута и логическая адресация	IP-адрес	IPv4, IPv6, IPsec, AppleTalk
	2. Канальный (data link)	Физическая адресация	MAC-адрес (физический адрес компьютера)	PPP, IEEE 802.22, Ethernet, DSL, ARP, L2TP, сетевая карта.
	1. Физический (physical)	Работа со средой передачи, сигналами и двоичными данными		USB, кабель ("витая пара", коаксиальный, оптоволоконный), радиоканал

Для обеспечения необходимой совместимости на каждом из уровней архитектуры КС действуют специальные стандартные **протоколы**. Они представляют собой формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах сети.

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле сети, должны взаимодействовать друг с другом также в соответствии с четко определенными правилами и с помощью стандартизированных форматов сообщений. Эти правило принято называть межуровневым **интерфейсом**. В сущности, протокол и интерфейс являются близкими понятиями, но традиционно в сетях за ними закреплены разные области действия: протоколы определяют правила взаимодействия модулей одного уровня в разных узлах сети, а интерфейсы определяют правила взаимодействия модулей соседних уровней в одном узле.

Описание уровней по схеме.

Рассмотрим подробнее уровни модели OSI.

- **Уровень 1: физический (Physical) уровень:** Этот уровень определяет электрические, механические, процедурные и функциональные технические условия для активации и поддержки физической связи между взаимодействующими сетевыми системами. Технические условия физического уровня определяют такие характеристики, как уровни питания, физические скорости передачи данных и физические соединители.
- **Уровень 2: канальный (Data Link) уровень:** Этот уровень обеспечивает синхронизацию, контроль ошибок и управление потоками данных по физическим каналам, включая физические и логические соединители с местами доставки пакетов, как правило, с помощью сетевых адаптеров (NIC, network interface card). Этот уровень разделен на два подуровня: уровень управления логической связью (MAC, Media Access Control) и уровень управления доступа к устройствам (LLC,

Logical Link Control). На этом уровне работают такие протоколы, как Point-to-Point Protocol (PPP), Layer 2 Tunneling Protocol (L2TP) и Fiber Distributed Data Interface (FDDI). MAC-адреса устройств формируют основу сетей на канальном уровне модели OSI, которую используют протоколы более высокого (сетевого) уровня. Для преобразования MAC-адресов в адреса сетевого уровня и обратно применяются специальные протоколы, например, ARP.

- **Уровень 3: уровень сети (Network):** Этот уровень определяет сетевой адрес и обеспечивает маршрутизацию и перенаправление (forwarding) данных. На этом уровне работают такие протоколы, как Internet Protocol (IP), Internet Control Message Protocol (ICMP) и Routing Information Protocol (RIP).
- **Уровень 4: уровень транспорта (Transport):** Этот уровень обеспечивает сквозное (end-to-end) управление, включая проверку на наличие ошибок и управление потоками. Он получает данные с уровня сеанса (см. уровень 5) и разбивает данные для транспортировки по сети. На уровне транспорта работают протоколы Transmission Control Protocol (TCP) и User Datagram Protocol (UDP). TCP отслеживает состояние соединения, например, порядок доставки пакетов и то, какие пакеты должны передаваться следующими. UDP, напротив, является протоколом, который не устанавливает соединения и не проверяет, дошел ли пакет до адресата.
- **Уровень 5: уровень сеанса (Session):** Этот уровень устанавливает, поддерживает и прекращает сеансы связи. Сеансы связи состоят из запросов служб и их ответов, происходящих между приложениями, расположенными на различных сетевых устройствах. Эти запросы и ответы согласовываются с помощью протоколов, реализованных на уровне сеанса. На этом уровне работают протоколы Secure Socket Layer (SSL), Remote Procedure Call (RPC) и AppleTalk Protocol.
- **Уровень 6: уровень представления данных (Presentation):** Этот уровень форматирует данные, представляемые уровнем приложения. Он может рассматриваться в качестве переводчика для сети (упорядочение битов данных). Этот уровень может переводить данные из формата, используемого уровнем приложения в общий для посылающей и принимающей стороны формат. На этом уровне работают такие хорошо известные графические форматы, как Graphics Interchange Format (GIF) и Joint Photographic Experts Group (JPEG). Этот уровень также выполняет сжатие и шифрование данных.
- **Уровень 7: уровень приложения (Application):** Этот уровень обычно обеспечивает идентификацию партнеров по взаимодействию, определяет доступность ресурсов и синхронизирует взаимодействие. Этот уровень не включает самих приложений, но только протоколы, поддерживающие их. На уровне приложения работают такие протоколы, как Simple Mail Transfer Protocol (SMTP), Hypertext Transfer Protocol (HTTP) Telnet и FTP.

Обмен данными через каналы связи происходит путем перемещения данных с верхнего уровня на нижний, затем транспортировки по линиям связи и наконец обратным воспроизведением данных в компьютере клиента в результате их перемещения с нижнего уровня на верхний.

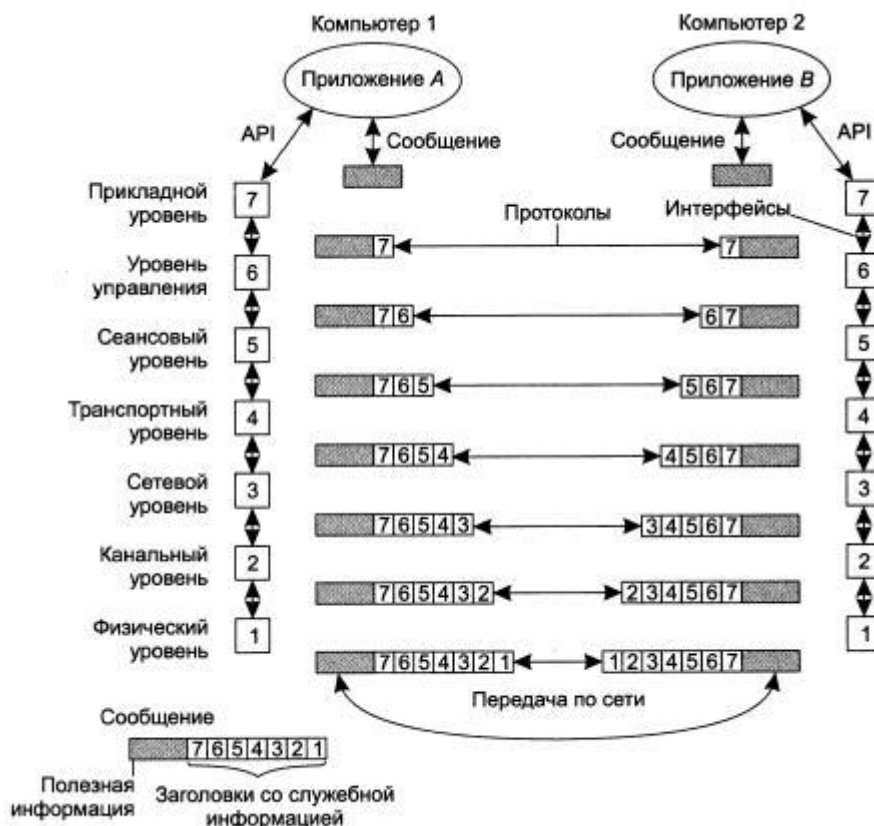


Рис. 1.44. Модель взаимодействия открытых систем OSI

Итак, пусть, например, приложению А требуется передать файл приложению В. Для этого приложение А посылает сообщение-запрос клиенту файловой службы, которая, в соответствии с разделением функций в модели OSI, относится к прикладному уровню. На основании этого запроса клиент в соответствии с принятым протоколом формирует сообщение для передачи серверу файловой службы, также работающему на прикладном уровне другого компьютера. Формат сообщения предусматривает наличие поля заголовка прикладного уровня (на рисунке он обозначен цифрой 7), в котором клиент размещает управляющую информацию, адресованную файловому серверу, например, полное имя файла в файловой системе сервера и тип операции «открыть файл». Но для того, чтобы доставить это сообщение по назначению, предстоит решить еще много задач, ответственность за которые несут нижележащие уровни.

Поэтому прикладной уровень, используя межуровневый интерфейс, передает запрос на выполнение необходимых для него действий расположенному ниже уровню представления. Запрос включает сформированное на прикладном уровне сообщение, а также некоторую дополнительную служебную информацию, без которой нижележащий уровень не может выполнить указанные действия, например, расшифровать

текст, который хранится на сервере в зашифрованном виде. Протокол уровня представления выполняет требуемые операции и добавляет к сообщению собственную служебную информацию – заголовок уровня представления, в котором содержатся указания для протокола уровня представления компьютера-адресата, например, тип шифра. Полученное в результате сообщение передается вниз сеансовому уровню, который, в свою очередь, добавляет свой заголовок и т.д. Наконец, сообщение достигает нижнего, физического уровня, который собственно и передает его по линиям связи компьютеру-адресату. К этому моменту сообщение «обрастает» заголовками всех уровней.

Физический уровень помещает сообщение на физический выходной интерфейс компьютера 1, и оно начинает свое «путешествие» по сети. Когда сообщение по сети поступает на входной интерфейс компьютера 2, оно принимается его физическим уровнем и последовательно перемещается вверх с уровня на уровень. Каждый уровень анализирует и обрабатывает заголовок своего уровня, выполняя соответствующие функции, а затем удаляет этот заголовок и передает сообщение вышележащему уровню.

Распределение функций между различными элементами сети.

Функциональность полного стека протоколов может быть востребована только конечными узлами, а коммуникационное оборудование – маршрутизаторы и коммутаторы, решающие задачу транспортировки сообщений между конечными узлами, как правило, ограничиваются поддержкой функциональности нижних трех уровней.

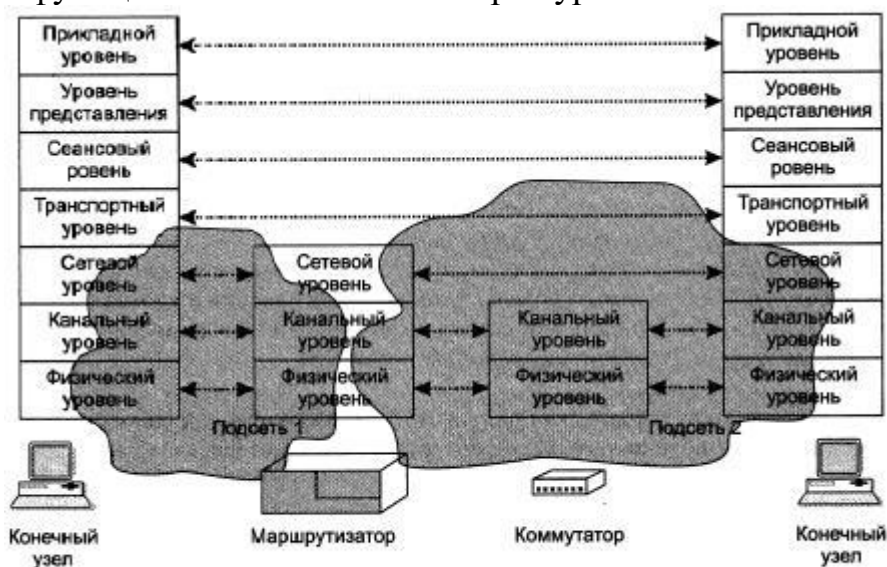


Рис. 1.45. Распределение функций между различными элементами сети

На рисунке показаны основные элементы компьютерной сети: конечные узлы (компьютеры) и промежуточные узлы (коммутаторы и маршрутизаторы).

Коммутаторы, как правило, поддерживают функции двух нижних уровней (физического и канального), что ограничивает их возможности передачи данных в пределах только одной подсети. Однако некоторые

коммутаторы, работающие на основе технологии виртуальных каналов, могут поддерживать как два уровня протоколов, так и три.

Маршрутизаторы служат пограничными устройствами, разделяя составную сеть на подсети. Они поддерживают функции всех трех нижних уровней, т.к. сетевой уровень нужен им для объединения подсетей различных технологий в составную сеть и нахождения маршрута между конечными узлами через составную сеть, а функции нижних уровней – для передачи данных в пределах отдельных подсетей.

Компьютеры, в общем случае поддерживают функции всех уровней модели. Протоколы прикладного уровня, пользуясь сервисами протоколов уровня представления и сеансового уровня, предоставляют приложениям набор сетевых услуг в виде сетевого интерфейса API.

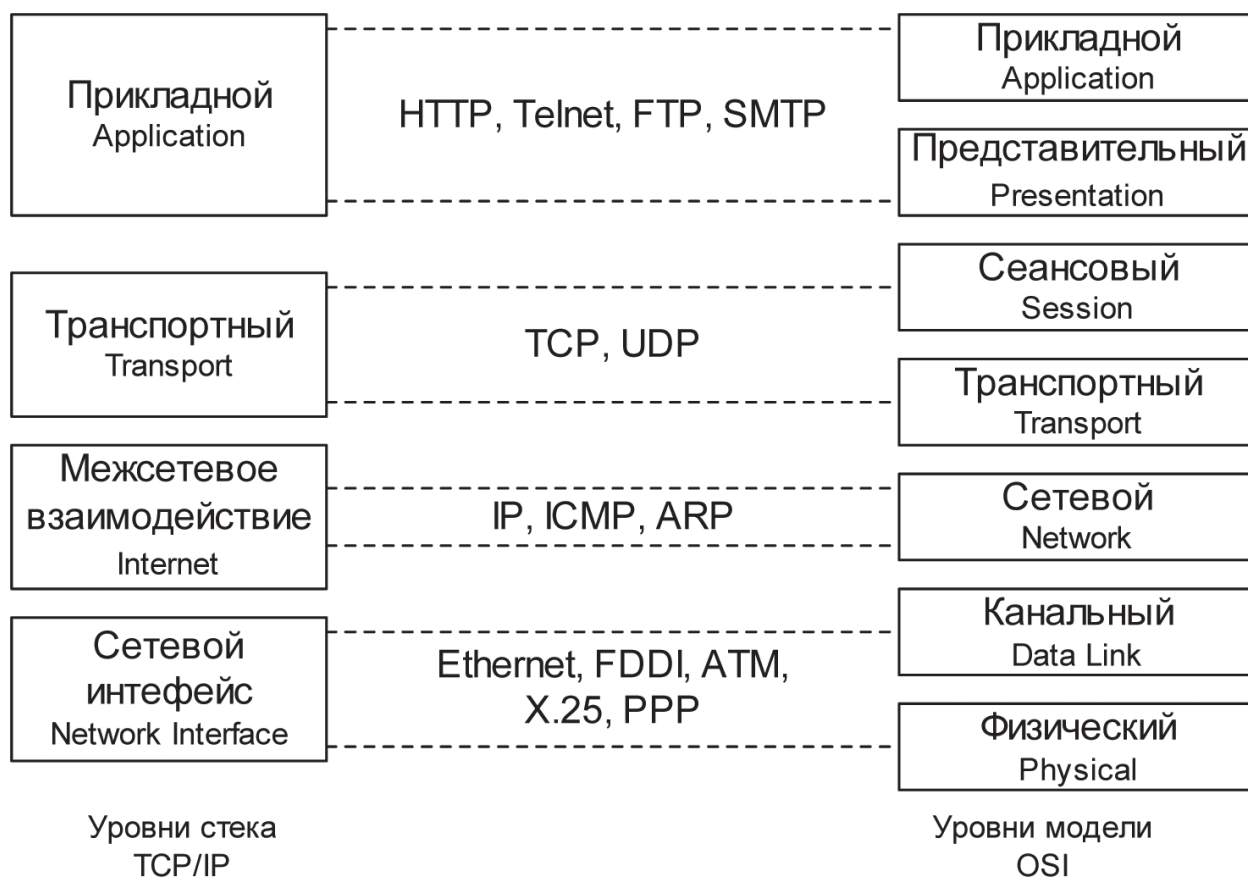
В реальных сетях некоторые из коммуникационных устройств поддерживают и протоколы верхних уровней. К примеру, коммутаторы часто поддерживают протоколы SNMP и Telnet, которые по факту не нужны для выполнения основных функций устройств, но позволяют конфигурировать их и управлять удаленно.

Стек протоколов TCP/IP.

В настоящее время основным используемым стеком протоколов является TCP/IP (Transmission Control Protocol / Internet Protocol), разработанный еще до принятия OSI в качестве эталонной модели и вне связи с ней. Этим объясняется неполное соответствие стека TCP/IP уровням модели OSI и некоторые противоречия между этими стандартами. Некоторые специалисты называют стек протоколов TCP/IP упрощенной моделью OSI-стандарта. Но соответствие уровней стека TCP/IP уровням модели OSI достаточно условно: уровни TCP/IP можно поставить в соответствие четырем верхним уровням модели OSI. Данный стек изначально был разработан по инициативе Министерства обороны США в конце 1970-х гг. и предназначался для связи экспериментальной сети ARPAnet с другими спутниковыми сетями. Он представляет набор общих протоколов для разрозненных вычислительных сред.

Большой вклад в развитие стека TCP/IP внес Калифорнийский университет в Беркли (University of California, Berkeley), реализовав данный набор протоколов в своей версии ОС – UNIX, которая оказалась весьма востребованной на развивающемся рынке ИТ, и ее широкое распространение привело к господству стека протоколов TCP/IP в области межсетевого взаимодействия.

Стек TCP/IP объединяет в себе целый набор взаимодействующих между собой протоколов. Структура протоколов TCP/IP приведена на рисунке.



Стек протоколов имеет 4 уровня: прикладной, транспортный, уровень межсетевого взаимодействия и уровень сетевых интерфейсов. Для сравнения на рисунке показано соответствие семиуровневой модели OSI. Следует заметить, что нижний уровень модели – уровень сетевых интерфейсов – строго говоря, не выполняет функции канального и физического уровней, а лишь обеспечивает связь (интерфейс) верхних уровней DARPA с технологиями сетей, входящих в составную сеть (например, Ethernet, FDDI, ATM).

Самыми важными из протоколов являются:

1) **протокол IP**, отвечающий за поиск маршрута (или маршрутов) в Интернете от одного компьютера к другому через множество промежуточных сетей, шлюзов и маршрутизаторов и передачу блоков данных по этим маршрутам. Суть состоит в том что у каждого пользователя всемирной сети Интернет должен быть свой уникальный IP-адрес. Структура IP-адреса организована таким образом, что каждый компьютер, через который проходит какой-либо TCP-пакет, может по четырем битам IP-адреса определить, кому из ближайших соседей надо переслать пакет, чтобы он оказался ближе к получателю. В результате конечно числа перебросок пакет достигает адресата. В данном случае оценивается не географическая близость, пропускная способность линий связи. Два компьютера, находящихся на разных континентах, но связанные высокопроизводительной линией космической связи, считаются более близкими друг другу, чем два компьютера из соседних городов, связанных телефонной связью. Решением вопросов близости узлов сети занимаются маршрутизаторы.

2) **протокол TCP**, обеспечивающий надежную доставку, безошибочность и правильный порядок приема передаваемых данных. Этот протокол называют протоколом «с установлением соединения». Два узла, связывающиеся при помощи этого протокола, договариваются о том, что они будут обмениваться данными и принимают соглашения об управлении потоком данных. Данные нарезаются в маркированные пакеты, чтобы обеспечить правильность сборки на компьютере получателя.

3) **протокол UDP**, обеспечивает передачу пакетов дейтаграммным способом, т.е. каждый блок информации (пакет) обрабатывается и распространяется как независимая единица – дейтаграмма. При этом не гарантируется доставка этих пакетов, из-за чего не может гарантироваться правильность сборки отправленных данных.

Когда данные передаются от прикладного уровня к транспортному, затем к уровню межсетевого взаимодействия и далее через уровень сетевого интерфейса в сеть, каждый протокол выполняет соответствующую обработку и инкапсулирует результат этой обработки, присоединяя спереди свой заголовок. Схема процесса инкапсуляции передаваемых данных и формирования заголовков пакетов в стеке TCP/IP показана на следующем рисунке.

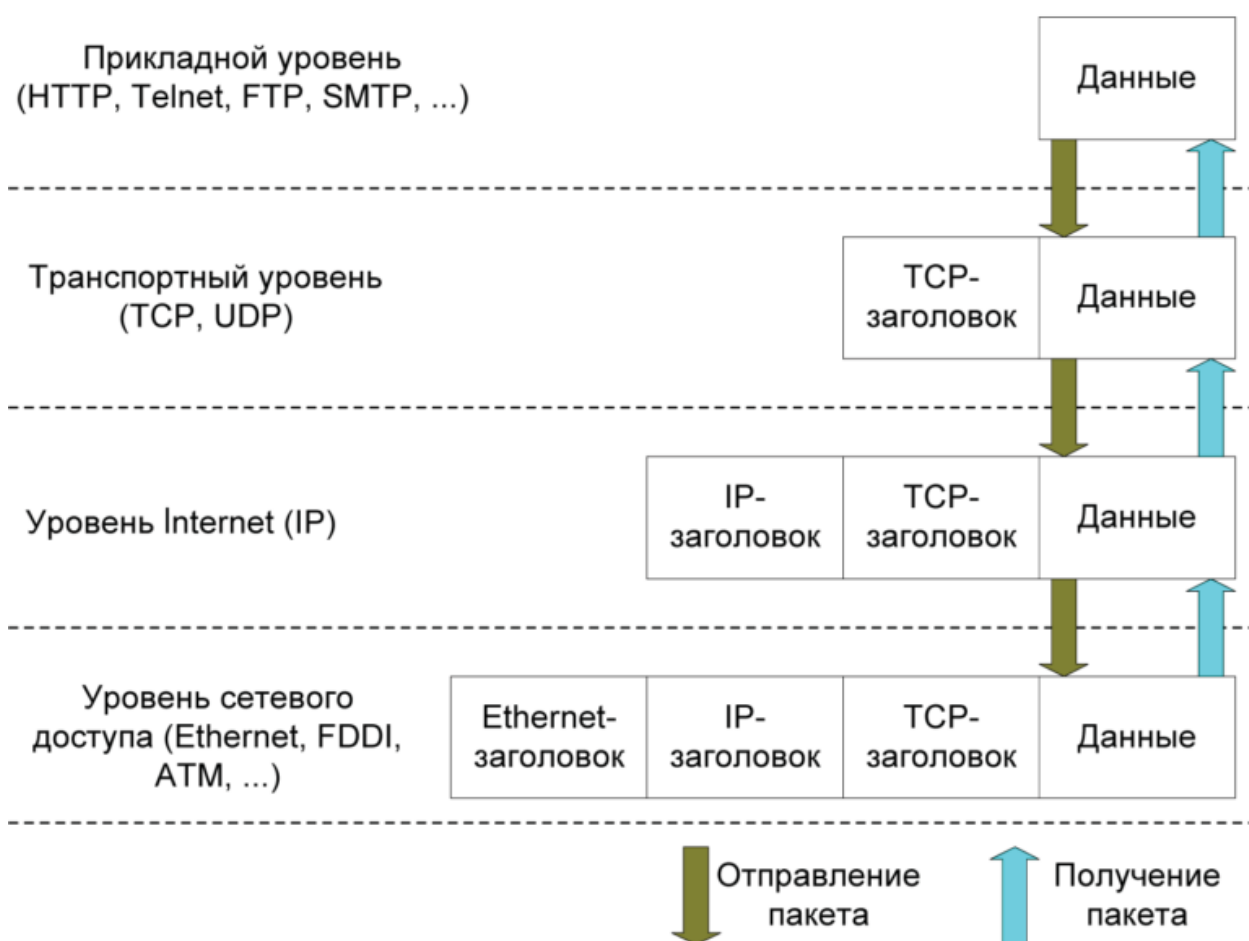


Рис. схема инкапсуляции

При приеме потока информации эти заголовки последовательно удаляются по мере обработки данных и передачи их вверх по стеку. Такой

подход обеспечивает необходимую гибкость в обработке передаваемых данных, поскольку верхним уровням вовсе не нужно касаться технологии, используемой на нижних уровнях.

Благодаря своей популярности TCP/IP стал стандартом де-факто для межсетевого взаимодействия и именно на основе данной технологии заработала развивающаяся в то время глобальная компьютерная сеть Internet. Однако повсеместное распространение стека протоколов TCP/IP обнажило и его слабые стороны: создавая свое детище, архитекторы стека не видели причин особенно беспокоиться о защите сетей, строящихся на его основе. Поэтому в ранних версиях протокола IP вообще отсутствовали требования безопасности, что привело к изначальной уязвимости его реализации. Надо понимать, что и сейчас, если не принято специальных мер, то все данные передаются протоколами TCP/IP в открытом виде. Соответственно любой узел сети и оператор узла, находящийся на пути следования данных от отправителя к получателю, может скопировать себе все передаваемые данные и использовать их в дальнейшем в своих целях.

Следует привести недавний пример. В июне 2020 года обнаружили 19 уязвимостей – в совокупности именуемых *Ripple20* – в небольшой библиотеке, выпущенной в 1997 году софтверной фирмой Treck из Цинциннати (штат Огайо). За прошедшие с тех пор более двух десятков лет эта библиотека, содержащая облегченный стек TCP/IP для подключения маломощной техники к Интернету, была интегрирована в «сотни миллионов» промышленных и потребительских продуктов. Список уязвимого оборудования включает устройства умного дома, медицинские системы жизнеобеспечения, производственную технику, принтеры, маршрутизаторы, авионику, различные решения уровня предприятия и многое другое. Затронутые «жертвы» варьируются от бутиков для одного человека до многонациональных корпораций из списка Fortune 500, включая HP, Schneider Electric, Intel, Rockwell Automation, Caterpillar, Baxter, а также многих других крупных международных поставщиков.

Некоторые из этих уязвимостей – критические и делают возможным удаленное выполнение кода. Самое печальное в том, что часть устройств никогда не получит обновлений.

Сетевые уязвимости, угрозы и атаки.

Компьютерная сеть основана на передаче пакетов данных между ее конечными узлами – серверами и клиентскими компьютерами. Для того чтобы такая передача стала возможна, все конечные узлы сети соединены друг с другом линиями связи и промежуточными узлами – маршрутизаторами и коммутаторами – транспортной инфраструктурой.

Рисунок. Фрагмент транспортной инфраструктуры сети (см презентацию).

Естественно, что транспортная инфраструктура сети является заманчивой мишенью для злоумышленников, поскольку фактически все

сетевые уровни содержат уязвимости и злоумышленники имеют изобилие возможностей для осуществления различных атак. Без создания надлежащей защиты любая часть любой сети может оказаться уязвимой для атак или другой несанкционированной деятельности. Угроза может исходить из широкого круга источников, включая профессиональных хакеров, конкурентов или даже собственных работников. Для определения наилучшего способа нейтрализации этих угроз и защиты сетей от осуществляемых атак, «безопасники» должны знать множество типов возможных атак и тот вред, который эти атаки могут нанести сетевой инфраструктуре вашей организации.

Основные причины уязвимости компьютерных сетей

В свою очередь, развитие глобальной сети Интернет способствовало использованию для построения глобальных корпоративных связей более дешевого и более доступного (по сравнению с выделенными каналами) транспорта Интернета. Сеть Интернет предлагает разнообразные методы коммуникации и способы доступа к информации, поэтому для многих компаний она быстро становится неотъемлемой частью их информационной системы. Отметим основные возможности, предоставляемые сетью Интернет для построения корпоративных сетей:

- ***дешевые и доступные коммуникационные каналы Интернета.*** В связи с бурным развитием Интернета и сетей коллективного доступа в мире произошел качественный скачок в распространении и доступности информации. Выход в интернет стоит не дорого и доступен практически всем. Стремясь к экономии средств, предприятия начали активно использовать эти каналы для передачи критической коммерческой и управленческой информации;

- ***универсальность.*** Развитие интернет-технологий привело к возникновению популярной глобальной службы World Wide Web (WWW), что позволило пользователям работать с информацией в режиме прямого подключения с использованием веб-браузеров для просмотра интересующей информации. Стандартизация интерфейсов обмена данными между утилитами просмотра информации и информационными серверами позволила организовать одинаковый интерфейс с пользователем для различных платформ;

- ***доступ к разнообразной информации и услугам в Интернете.*** Кроме транспортных услуг по транзитной передаче данных для абонентов любых типов, сеть Интернет обеспечивает также достаточно широкий набор высокоуровневых интернет-сервисов: всемирная паутина WWW; сервис имен доменов DNS; доступ к файловым архивам FTP и облачным сервисам; электронная почта e-mail; телеконференции; сервисы общения; сервисы поиска информации прочее. Связав свои сети с внешними ресурсами, компании могут реализовать постоянные коммуникации и организовать эффективный поток информации между людьми. Соединение внутренних сетей с внешними организациями и ресурсами позволяет компаниям

воспользоваться преимуществами этих сетей – снижением затрат и повышением эффективности;

- **простота использования.** При использовании интернет-технологий в большинстве случаев не требуется специального обучения персонала.

Построение корпоративных компьютерных сетей с применением технологии Интернета означает, прежде всего, использование стека TCP/IP для транспортировки данных и технологии Web для их представления. Как следствие такого влияния, основными причинами уязвимости компьютерных сетей стали:

1. Проектирование сети INTERNET как открытой и децентрализованной сети с изначальным отсутствием политики безопасности. Сеть Интернет создавалась для связи между государственными учреждениями и университетами с целью оказания помощи учебному процессу. На начальном этапе никто не мог предположить дальнейший масштаб его развития и интеграции в жизнь современного общества, в связи с чем вопросам безопасности не уделялось должного внимания.

2. Множественные уязвимости служб протокола TCP/IP. Уязвимости протоколов, входящих в стек TCP/IP обусловлены, как правило, слабой аутентификацией, ограничением размера буфера, отсутствием проверки корректности служебной информации и т.п.

3. Множество уязвимостей в программном и аппаратном обеспечении, работающем на ПК, подключенном к компьютерной сети.

4. Возможность оставаться анонимным при работе в INTERNET

5. Распространение простых в использовании ОС и средств разработки. Этот фактор снижает требования к уровню знаний злоумышленника (к примеру, во времена Митника требовалось знать стеки протоколов и архитектуры сети) и навыкам программирования. Теперь, для того чтобы получить несанкционированный доступ достаточно знать IP-адрес жертвы и щелкнуть мышкой в специальной программе для проведения атаки (например, Kali Linux и Metasploit, вы удивитесь как это просто).

Нарушитель, осуществляя атаку, обычно ставит перед собой следующие цели:

- нарушение конфиденциальности передаваемой информации;
- нарушение целостности и достоверности передаваемой информации;
- нарушение работоспособности системы в целом или отдельных ее частей.

Примеры уязвимостей по уровням модели OSI

Получив базовое представление о том, как структурированы сети и как происходит обмен данными, посмотрим на некоторые конкретные уязвимости сетей и возможные виды атак. Существует множество способов классификации уязвимостей систем безопасности и атак. Будет полезным сделать их небольшой обзор с точки зрения уровней OSI.

Краткая характеристика уязвимостей нескольких протоколов по уровням модели OSI (на примере стека TCP/IP) приведена в таблице.

Наименование протокола	Соответствие уровню OSI	Характеристика уязвимости	Содержание нарушения безопасности информации
FTP (File Transfer Protocol) – протокол передачи файлов по сети	Прикладной, представительный, сеансовый	1. Аутентификация на базе открытого текста (пароли пересылаются в незашифрованном виде). 2. Доступ по умолчанию. 3. Наличие двух открытых портов	Возможность перехвата данных учетной записи (имен зарегистрированных пользователей, паролей). Получение удаленного доступа к хостам
Telnet – протокол управления удаленным терминалом	Прикладной, представительный, сеансовый	Аутентификация на базе открытого текста (пароли пересылаются в незашифрованном виде)	Возможность перехвата данных учетной записи пользователя. Получение удаленного доступа к хостам
UDP – протокол передачи данных без установления соединения	Транспортный	Отсутствие механизма предотвращения перегрузок буфера. Отсутствие проверки доставки пакетов адресату	Возможность реализации UDP-шторма. В результате обмена пакетами происходит существенное снижение производительности сервера. Вероятность потери информации в процессе передачи
ARP – протокол преобразования IP-адреса в физический адрес	Сетевой	Аутентификация на базе открытого текста (информация пересылается в незашифрованном виде)	Возможность перехвата трафика злоумышленником
RIP – протокол маршрутной информации	Транспортный	Отсутствие аутентификации управляющих сообщений об изменении маршрута	Возможность перенаправления трафика через хост злоумышленника
TCP – протокол управления передачей	Транспортный	Отсутствие механизма проверки корректности заполнения служебных заголовков пакета	Существенное снижение скорости обмена и даже полный разрыв произвольных соединений по протоколу TCP
DNS – протокол установления соответствия mnemonic-имен и сетевых адресов	Прикладной, представительный, сеансовый	Отсутствие средств проверки аутентификации полученных данных от источника	Фальсификация ответа DNS-сервера
IGMP – протокол передачи сообщений о маршрутизации	Сетевой	Отсутствие аутентификации сообщений об изменении параметров маршрута	Возможность подделки маршрута. Приводит к остановке операционных систем Win9x / WinNT
SMTP – протокол обеспечения сервиса доставки сообщений по электронной почте	Прикладной, представительный, сеансовый	Отсутствие поддержки аутентификации заголовков сообщений	Возможность подделки сообщений электронной почты, а также адреса отправителя сообщения
SNMP – протокол управления маршрутизаторами в сетях	Прикладной, представительный, сеансовый	Отсутствие поддержки аутентификации заголовков сообщений	Возможность достижения максимальной пропускной способности сети

В таблице перечислены уязвимости отдельных протоколов стека TCP/IP, обусловленные факторами «на уровне идеи». Необходимо отметить, что у этих же протоколов имеются и другие уязвимости, которые обусловлены ошибками и недоработками в их реализации, потенциальной возможностью наличия в них «закладок», вредоносных программ и т. д. Кроме того, с течением времени ПО, в котором реализована поддержка этих протоколов, обновляется, и соответственно каждая новая версия ПО может содержать в себе новые ошибки или «закладки».

Львиная доля уязвимостей приходится на **уровень приложения (он же – прикладной)**, расположенный в непосредственной близости к самому пользовательскому приложению. Яркими примерами этого являются такие протоколы как Telnet и FTP. Эти приложения пересылают пользовательские пароли таким образом, что любой, кто может "прослушивать" сетевой трафик, получит пользовательское регистрационное имя и пароль, а, следовательно, неправомерный доступ. На **уровне представления данных** существует множество способов атак против зашифрованных данных. На **уровне сеанса** в Remote Procedure Call (RPC) существует одна из самых серьезных уязвимостей компьютерных систем по версии института SANS. На **уровне транспорта** выполняются атаки с помощью SYN flood или подмены одного из участников TCP-соединения (TCP-hijacking). Сканирование портов является распространенной техникой, используемой хакерами для выявления уязвимых систем. Подмена IP-адреса - обычная атака для **уровня сети**. Частое прослушивание трафика и перехват сообщений - атаки, осуществляемые на **Уровне 1 и 2**. В свою очередь, появление беспроводных сетей открыло новые возможности для хакеров из-за возможности доступа к **физическому уровню сети**.

Уязвимости протоколов сетевого взаимодействия связаны с особенностями их программной реализации и обусловлены ограничениями на размеры применяемого буфера, недостатками процедуры аутентификации, отсутствием проверок правильности служебной информации и др.

Отметим также, что существуют протоколы с поддержкой шифрования процедуры авторизации, установленной сессии и передаваемых данных (например, SFTP, SSH). При этом и такие протоколы нельзя назвать полностью безопасными в силу неизбежных ошибок в их реализации, а также неидеальной криптографической составляющей.

Для систематизации описания множества уязвимостей программных сетевых протоколов и ПО используется единая база данных (БД) уязвимостей CVE (Common Vulnerabilities and Exposures), в разработке которой принимали участие специалисты многих известных компаний и организаций, таких как MITRE, ISS, Cisco, BindView, Axent, NFR, L-3, CyberSafe, CERT, Carnegie Mellon University, Институт SANS и т. д. Эта БД постоянно пополняется и используется при разработке многочисленных программных средств анализа защищенности и, прежде всего, средств мониторинга сетей.

Основные проблемы стека протоколов TCP/IP

Рассмотрим основные проблемы стека протоколов TCP/IP.

Протокол IP.

Основой архитектуры TCP/IP является протокол **IP** (Internet Protocol), работающий на сетевом уровне модели OSI. Он обеспечивает перемещение пакетов между узлами составной сети, образованной объединением множества сетей. Протокол IP соединяет воедино в общем случае разнородные сети, каждая из которых может работать на основе самых разных транспортных технологий нижележащих уровней: Ethernet, SDH, MPLS и др.

Протокол IP работает как на конечных узлах сети, так и на маршрутизаторах, основным назначением которых является чтение адреса назначения пакета и передача пакета следующему маршрутизатору, который находится на пути следования пакета.

Кроме того, протокол IP это дейтаграммный протокол, который работает без установления соединения. Это означает, что любой узел Интернета может направлять пакеты любому другому узлу без предварительной процедуры получения разрешения «поговорить». Можно сказать, что протокол IP является краеугольным камнем «демократии» Интернета – все узлы в сети равны, каждый может общаться с каждым, при этом отказаться от такого общения с помощью средств самого протокола IP нельзя: если уж кто-то отправил пакет предназначенный для вас, то он скорее всего дойдет, если не было предпринято специальных мер безопасности. И когда клиентский компьютер, имеющий, например, IP-адрес 173.194.67.94, направляет пакет серверу с адресом 8.8.8.8, то сеть однозначно знает, какому серверу из миллионов серверов, подключенных к Интернету, нужно доставить этот пакет и какому клиенту вернуть ответ.

Данное свойство IP – обеспечение взаимодействия каждого узла с каждым, без предварительного установления соединения, и представляют одну из главных уязвимостей IP-сетей. Действительно, в IP-сети любой злоумышленник в общем случае имеет доступ к любому узлу сети, а значит, имеет возможность организовать атаку.

Все методы предотвращения атак на транспортную инфраструктуру сети основаны на ограничении указанного свойства, в частности, именно этим и занимаются фаерволлы.

Нужно отметить, что не все транспортные протоколы являются такими же демократичными, например, существуют такие протоколы как X.25 (уже практически не используется), frame relay, ATM, MPLS, которые включают процедуру предварительного установления соединения между источником и узлом. При этом такое соединение устанавливается администратором сети, а не пользователем, поэтому риск атаки здесь существенно меньше.

Протоколы TCP и UDP.

Протоколы транспортного уровня TCP и UDP работают «над» протоколом IP, используя его как инструмент для решения своих задач – передачи данных между приложениями ~~двух~~ взаимодействующих по сети

конечных узлов. Можно сказать, что протоколы TCP и UDP – это протоколы конечных узлов, т.к. даже в том случае, когда они работают на маршрутизаторе (для удаленного администрирования и конфигурирования), последний выступает как конечный узел в паре «компьютер администратора» - «конфигуратор маршрутизатора».

Рисунок с работой протокола TCP (см. презентацию).

Рассмотрим упрощенно работу протокола TCP. Приложение-отправитель направляет протоколу TCP поток байтов, которые необходимо передать приложению-получателю, работающему на другом конечном узле сети. Протокол TCP делит поступающий поток данных на сегменты и передает вниз протоколу IP. Протокол IP узла отправителя упаковывает сегменты в пакеты и отправляет на ближайший маршрутизатор, протокол IP которого передает их дальше. Пакеты перемещаются по сети от одного маршрутизатора к другому. Когда пакеты достигают узел назначения, протокол IP конечного узла передает пакеты вверх протоколу TCP, который организует передачу данных приложению-получателю.

Протокол TCP обеспечивает гарантированную доставку данных, для этого в нем предусмотрена процедура установления логического соединения (сессия) между протоколами TCP, работающими на узлах отправителя и получателя. В рамках соединения протокол TCP нумерует пакеты, отправляет квитанции подтверждения их приема, в случае потери данных организует повторные передачи, распознает и уничтожает дубликаты, и после получения всех сегментов передает приложению-получателю данные в том порядке, в котором они были отправлены.

UDP является дейтаграммным протоколом и, следовательно, не может гарантировать доставку данных. Он используется в тех случаях, когда задача надежного обмена данными либо вообще не ставится, либо решается средствами более высокого уровня – прикладным уровнем или пользовательскими приложениями.

Для того, чтобы данные попали именно тому приложению, которому они предназначаются, в заголовках сегментов TCP и UDP имеются 2 поля: **порт источника** и **порт назначения**. Эти порты являются программными точками входа-выхода ОС; с каждым портом связана какая-то прикладная программа, которая обменивается данными по сети. Порты идентифицируются номерами, при этом за первыми 1023 номерами централизованно закреплены определенные сетевые приложения (например, порт 21 присвоен сервису передачи файлов ftp, а порт 22 – протоколу удаленного доступа ssh).

Хотя сами протоколы TCP и UDP относятся к транспортному уровню сети, их заголовки несут информацию о приложениях, которые этими протоколами используются, поэтому порты TCP и UDP всегда фигурируют в описаниях конфигураций файерволов для защиты от атак на прикладном уровне. В более поздних лекциях будут рассмотрены такие конфигурации.

Протокол TCP является более уязвимым для атак, чем протокол UDP, из-за того, что он включает **процедуру установления логического соединения**

между конечными узлами. Именно эту процедуру пытаются использовать злоумышленники для организации DoS и DDoS-атак, направленных на истощение ресурсов транспортной подсистемы конечного узла.

Протоколы маршрутизации.

Для того чтобы знать, какому следующему маршрутизатору передать IP пакет, чтобы он дошел до адресата, маршрутизаторы строят и используют **таблицы маршрутизации**. Таблица маршрутизации состоит из записей, в которых IP-адресу назначения (чаще диапазону) ставится в соответствие выходной порт маршрутизатора. Таблицы маршрутизации могут быть созданы вручную, либо построены автоматически с помощью **протоколов маршрутизации**.

Провайдеры Интернета используют два типа протоколов маршрутизации:

- внутренние протоколы маршрутизации – Interior Gateway Protocol, IGP, к которым относятся протоколы OSPF и IS-IS

- внешние протоколы маршрутизации – Exterior Gateway Protocol, EGP, представленные сегодня только одним протоколом BGP.

Протоколы IGP применяются для построения таблиц маршрутизации, отражающих пути пакетов внутри сети провайдера, а протоколы EGP используются для построения таблиц, отражающих маршруты пакетов между сетями провайдеров.

Протоколы маршрутизации являются важным элементом транспортной инфраструктуры сети и одним из ее уязвимых мест – ведь нарушив нормальный процесс построения таблицы маршрутизации, можно полностью нарушить транспортировку пакетов через сеть.

История Интернета знает несколько печальных случаев, когда из-за ошибки в конфигурировании протокола BGP одного маршрутизатора таблицы маршрутизации большого числа провайдеров оказались искажены и нормальная работа Интернета для многих пользователей оказалась прерванной на несколько часов.

Протокол ICMP.

Протокол ICMP (Internet Control Message Protocol) играет роль протокола обратной связи для узлов сети при передаче пакетов, работает на сетевом уровне модели OSI. При этом он не предназначен для исправления возникших при передаче пакета проблем: если пакет потерян, ICMP не может послать его заново. Задача ICMP другая – он является средством оповещения отправителя о «несчастных случаях», произошедших с его пакетами на пути следования или по прибытии на узел назначения. Если, например, маршрутизатор не знает, как передать пакет в сеть назначения из-за того, что адрес этой сети не содержится в его таблице маршрутизации, то он отправляет узлу, пославшему пакет, сообщение «узел назначения недостижим» с кодом «сеть назначения неизвестна». Протокол ICMP может также корректировать поведение конечного узла или маршрутизатора, отправив сообщение

«перенаправление маршрута», считая что выбранный маршрут не является рациональным. Очевидно, что злоумышленник может использовать сообщения протокола ICMP, чтобы «скорректировать» маршрутизацию пакетов в сети выгодным для себя образом.

Протокол DNS.

Протокол DNS (Domain Name System) – протокол прикладного уровня, выполняющий важную для пользователей Интернета функцию – он отображает понятные пользователю символьные (называемые также доменными) имена узлов сети, такие как www.kai.ru, в IP-адреса этих узлов.

Дело в том, что маршрутизаторы передают пакеты на основании IP-адресов, а символьные имена они не понимают. В IP-пакетах использовать доменные имена в качестве адреса назначения также нельзя. Служба DNS нужна для того, чтобы преобразовать символьные имена, вводимые в окно браузера или в поле адреса электронной почты, в IP-адреса, понятные маршрутизаторам. Это преобразование инициирует узел-отправитель, точнее – компонента его ОС, называемая *резольвером*. Запрос резольвера идет к одному из известных конечному узлу *DNS-серверов*, который сам или же с помощью других DNS-серверов, работающих в Интернете, находит искомое соответствие и возвращает IP-адрес конечному узлу. После этого конечный узел формирует IP-пакет с найденным IP-адресом назначения и отправляет его в сеть.

Служба DNS представляет собой распределенную иерархическую базу данных, при этом иерархия серверов DNS отражает иерархию доменных имен. Серверы нижнего уровня иерархии хранят адреса имен соответствующего домена, например сервер домена kai.ru хранит адреса имен вида xyz.kai.ru. Серверы следующего уровня иерархии хранят уже не адреса конечных узлов, а адреса серверов DNS доменов нижнего уровня: например, DNS-сервер домена ru хранит адрес DNS-сервера домена kai.ru, а также всех DNS-серверов других доменов, входящих в домен ru.

Очевидно, что служба DNS является весьма уязвимым элементом транспортной инфраструктуры. Если она не работает, то клиенты оказываются отрезанными от сайтов Интернета, т.к. резольверы их компьютеров не могут сделать первый шаг в отправке запроса на сайты, поскольку символьное имя сайт не отображается на его IP-адрес. Другой вариант атаки – это подмена действительного IP-адреса на ложный, тогда сеть вроде бы работает и сайт отвечает, но это не тот сайт, с которым пользователь хотел бы взаимодействовать.

За время существования Интернета было осуществлено очень много атак на его транспортную инфраструктуру, и в последние годы их количество и качество постоянно растут. Рост числа и интенсивности атак отражает растущую значимость сетевых сервисов для бизнеса и политики, а также растущие мощности компьютеров, большинство из которых сегодня

оснащены интерфейсами 1 GE, а некоторые – интерфейсами 10 GE, так что сгенерировать мощный поток данных для них не является проблемой.

При изучении основных типов атак необходимо уделять внимание следующим обстоятельствам:

- какой элемент сети атакуется: компьютер, маршрутизатор или сервер DNS;
- какой протокол используется для атаки;
- происходит ли взлом используемого в атаке протокола, т.е. нарушается его нормальная работа, или же протокол только является инструментом для порождения трафика DoS/DDoS-атаки;
- на исчерпание какого типа ресурсов направлена атака: количество соединений с конечным узлом или пропускной способности интерфейса.

Классификация атак и их основные виды.

Характерные особенности сетевых атак.

С точки зрения безопасности *распределенные корпоративные системы* характеризуются наличием *удаленных атак*, поскольку компоненты распределенных систем обычно используют открытые каналы передачи данных и нарушитель может не только проводить пассивное прослушивание передаваемой информации, но и модифицировать передаваемый трафик, т.н. активное воздействие. И если активное воздействие на трафик может быть зафиксировано, то пассивное воздействие практически не поддается обнаружению. Трудность выявления факта проведения удаленной атаки выводит этот вид неправомерных действий на первое место по степени опасности, поскольку необнаруживаемость препятствует своевременному реагированию на угрозу, в результате чего у злоумышленника растут шансы на успешность атаки.

Для безопасности *локальной сети* на первое по значимости место выходят *нарушения зарегистрированных пользователей*, поскольку в основном каналы передачи данных локальной сети находятся на контролируемой территории и защита от несанкционированного подключения к ним реализуется административными методами.

По мере развития компьютерных и сетевых технологий (например, с появлением мобильных приложений и элементов ActiveX и Flash) список возможных сетевых атак на IP-сети постоянно расширяется.

Классификация атак.

Рассмотрим наиболее общую классификацию сетевых атак.

По характеру сетевые атаки могут быть:

1. Активными – атака, которая воздействуют на компоненты информационной системы, при реализации которой оказывается непосредственное влияние на работу системы. В результате активных действий в системе происходят определенные изменения. Поэтому активные воздействия легче обнаружить, чем пассивные. Примерами активных атак могут быть: зловредный программный код-вирус, внедренный в исполняемую

программу, искажение данных на страницах веб-сайта, блокировка сетевого сервиса путем бомбардировки его ложными запросами (DDOS-атака в виде шторма TCP-запросами) или внедренное в коммуникационный протокол ложное сообщение.

Многие активные кибер-атаки относят к типу «*взламывание*», проводя аналогию с бытовыми ограблениями со взломом, когда видны поврежденные замки, опустошенные ящики и разбросанные вещи. В компьютерной системе после активного проникновения тоже остаются следы «взлома», например поступают странные диагностические сообщения, приложения начинают работать медленно или зависать, а в трафике появляются необъяснимые всплески активности. Однако очевидно, что в жизни следы тоже можно «замести», а значит и подготовленная активная атака может пройти незамеченной, особенно если «безопасники» плохо представляют возможные последствия такого рода атак.

2. Пассивными – атака, которая не оказывает влияния на работу информационной системы, но может нарушить правила доступа к защищаемой информации. Такие атаки связаны со сбором информации о системе, например прослушиванием внутрисетевого трафика или перехватом сообщений, передаваемым по каналам связи. В большинстве случаев пассивные атаки не оставляют следов, их сложно выявить и они остаются незамеченными. Если использовать общепринятую терминологию, пассивные атаки можно назвать разведкой.

Представленная классификация является «идеальным случаем». На практике редко встречаются пассивные или активные атаки в чистом виде. Чаще атаки включают подготовительный этап сбора информации об атакуемой системе, а затем на основе этих данных осуществляется активное вмешательство в ее работу. Часто сбор информации помогает не только эффективнее внедриться, но и скрыть следы этого проникновения. Наиболее полезной информацией для злоумышленника являются сведения о типе операционной системы, приложениях и сервисах, запущенных в сети, IP-адресах, номерах портов клиентских сервисов, имена пользователей и пароли. Эта информация может быть получена в т.ч. из открытых источников или с применением методов социальной инженерии.

Коротко рассмотрим несколько типов популярных атак. Более подробно они будут рассмотрены в следующих лекциях.

- подслушивание (сниффинг, sniffing);
- посредничество;
- перехват сеанса (session hijacking);
- отказ в обслуживании;
- парольные атаки;
- атаки на уровне приложений;
- сетевая разведка;
- вредоносные программы;
- спуфинг (подмена доверенного субъекта);
- кража личности;

- фишинг;
- сетевая разведка.

Отказ в обслуживании.

К числу активных атак относят две весьма распространенные атаки:

- отказ в обслуживании, Denial of Service, DoS

и

- распределенная атака отказа в обслуживании, Distributed Denial of Service, DDoS.

Смысл DoS-атак прямо следует из названия. Система, предназначенная для выполнения запросов легальных пользователей, вдруг перестает это делать или делает это с большими задержками, что эквивалентно отказу. Эта атака не нацелена на получение доступа к вашей сети или на извлечение из этой сети какой-либо информации. По существу, эта атака лишает обычных пользователей доступа к ресурсам или сети организации в целом.

Рис. Схема DDoS-атаки (см. презентацию)

Отказ в обслуживании может наступить

- в результате редкой флуктуации интенсивности запросов, например при одновременном заходе на сайт bb.kai.ru сотен студентов возникает перегрузка сервера ввиду всплеска запросов,

- а также в результате злонамеренных действий, когда перегрузка создается искусственно, например на атакуемый компьютер посылается интенсивный поток запросов, сгенерированных средствами злоумышленника. Этот поток затопляет атакуемый компьютер, вызывая его перегрузку и часто делая его недоступным.

Блокировка происходит в результате исчерпания ресурсов процессора, либо ОС, либо канала связи (полосы пропускания). DoS-атака использует простой факт: компьютер подключен к сети, а значит уязвим. Большинство атак DoS опирается на общие слабости системной архитектуры. В случае использования некоторых серверных приложений (web, ftp и др.) атаки DoS могут заключаться в том, чтобы занять все соединения, доступные для этих приложений, и держать их в занятом состоянии, не допуская обслуживания обычных пользователей.

Рис. Схема DDoS-атаки (см. презентацию)

Злоумышленник может многократно усилить эффект от проведения атаки «отказ в обслуживании» путем кражи чужой вычислительной мощности. Для этого вредоносное программное обеспечение загружается на атакованный ранее и контролируемый компьютер, заставляя компьютер работать на себя, используя часть его вычислительной мощности. При этом такому компьютеру-зомби не наносится вреда, кроме снижения производительности. Комбинация из согласованных между собой «зомби» многократно усиливает поток запросов на компьютер-жертву. В таких случаях атаку называют распределенной атакой или DDoS-атакой.

DoS-атаки трудно предотвратить, т.к. для этого требуется координация действий с провайдером. Если трафик, предназначенный для переполнения

вашей сети, не остановить у провайдера, то на входе в сеть вы это сделать уже не сможете, потому что вся полоса пропускания будет занята.

Спуфинг (подмена доверенного субъекта).

Одним из основных приемов замещения следов является подмена содержимого пакетов – спуфинг, spoofing. В частности, для сокрытия места нахождения источника вредительских пакетов злоумышленник изменяет значение поля адреса отправителя в заголовках пакетов. Поскольку адрес отправителя генерируется автоматически системным ПО, злоумышленник вносит изменения в соответствующие программные модули так, чтобы они давали ему возможность отправлять со своего ПК пакеты с любыми IP-адресами.

Другой целью спуфинга является *подмена доверенного субъекта*. Большая часть сетей и ОС использует IP-адрес для определения нужного адресата. В некоторых случаях возможно некорректное присвоение IP-адреса (подмена IP-адреса отправителя другим адресом) – такой способ атаки называют *фальсификация адреса* или *IP-spoofing*.

Злоумышленник может воспользоваться IP-адресом, находящимся в пределах диапазона санкционированных IP-адресов, или авторизованным внешним адресом, которому разрешается доступ к определенным сетевым ресурсам.

Угроза спуфинга может быть ослаблена с помощью различных мер: правильная настройка управления доступом из внешней сети; пресечение попыток спуфинга чужих сетей пользователями своей сети. Следует иметь ввиду, что IP-спуфинг может быть осуществлен при условии проведения аутентификации пользователей на базе IP-адресов, поэтому введение дополнительных методов аутентификации пользователей позволяет предотвратить атаки IP-спуфинга.

Внедрение вредоносных программ.

Многочисленная группа активных атак связана с внедрением в компьютеры вредоносных программ (malware, malicious software). К этому типу атак относятся троянские и шпионские программы, руткиты, черви, вирусы, спам, логические бомбы и др.

Рис. Вредоносные программы (см. презентацию)

Эти программы проникают на атакуемые компьютеры разными путями. Самый простой – *самодоставка*, когда пользователь загружает файлы из непроверенных источников или открывает подозрительные файлы, пришедшие, например, по электронной почте. Другой способ проникновения – *размножение*, когда копии таких программ, используя собственные механизмы, распространяются по компьютерам сети без участия пользователей. Разновидностью таких программ являются *сетевые черви*, которые распространяются по сетям используя механизмы определения узлов, которые могут быть поражены.

Одним из ярких примеров ВПО являются *шпионские программы* (*spyware*), которые тайно (как правило удаленно) устанавливаются на компьютеры пользователей, чтобы отслеживать их действия. Например, ввод логина и пароля, посещение сайтов, обмен информацией в сети и прочее. Собранная информация пересылается злоумышленнику, который применяет ее в преступных целях. Интересные примеры легальных spyware-программ: бесплатные и популярные браузеры, где пользователи вынуждены давать согласие на сбор и использование этих данных, чтобы воспользоваться удобным приложением.

Ransomware. Разновидность вредоносного ПО, которое ограничивает доступ к вашим файлам до тех пор, пока вы не заплатите выкуп. Ограничение происходит за счёт установки специального шифрования на жестком диске или сервере, снять которое самостоятельно без специальных ключей практически невозможно.

Потери, вызванные ВПО, могут заключаться не только в уничтожении, искажении или похищении информации, приведении в нерабочее состояние ПО, но и в значительных затратах времени администраторов на обнаружение и распознавание атак, фильтрацию сообщений, тестирование и перезагрузку систем. ВПО в начале 2010-х годов были одной из основных причин нарушения безопасности компьютерных сетей. Однако с улучшением качества антивирусных средств, статистика последних лет показывает резкое снижение ущерба от подобного ПО.

Для защиты от ВПО необходимо применение ряда мер:

- исключение несанкционированного доступа к исполняемым файлам;
- тестирование приобретаемых программных средств;
- контроль целостности исполняемых файлов и системных областей;
- создание замкнутой среды исполнения программ.

Борьба с ВПО ведется с помощью эффективного антивирусного ПО.

Кража личности.

Пассивные атаки, хоть и не выглядят опасными, на самом деле могут оказаться серьезнее активных. Для проведения практически любой активной атаки требуются данные о системе, на которую готовится нападение. Если такой системой является отдельный человек, то сбор данных называют *кражей личности*.

Аферы, когда один человек выдает себя за другого, известны уже тысячи лет. А вот в случае использования Интернета уже не требуется личное присутствие в офисе и человек доказывает свою идентичность, передавая свои персональные данные используя интерактивную систему веб-сайта. Для того чтобы злоумышленнику выдать себя за другого человека достаточно узнать о нем паспортные данные и банковские реквизиты. После этого можно взять кредит на чужое имя, получить доступ к чужому счету и многое другое.

Фишинг.

Фишинг, phishing (искаженное fishing) – это один из приемов, используемых мошенниками для «выуживания» персональных и идентификационных данных пользователей. Сюда относятся кражи паролей, номеров кредитных карт, банковских счетов, пин-кодов и другой конфиденциальной информации.

Фишинг использует не технические недостатки, как правило, а легковерность пользователей Интернета. Тут могут использоваться приемы социальной инженерии, например в виде подставных звонков от легальных организаций с целью выяснения критичных данных, либо поддельных писем, где необходимо перейти по ссылке и заполнить интерактивную форму, содержащую поля с персональными данными, ИНН, девичьей фамилией матери, кличкой любимой собаки и др.

Сейчас чаще используются поддельные сайты, выглядящие как легальные и использующие доменные имена, похожие на настоящие. Например, www.paypal.com и www.peypal.com, здесь можно и не заметить подмены.

В настоящее время мошенники часто используют троянские программы. Задача фишера в этом случае сильно упрощается – достаточно заставить пользователя перейти на фишерский сайт и «подцепить» программу, которая самостоятельно найдет на диске жертвы все что нужно. Наравне с троянами также используются и кейлоггеры, которые отслеживают нажатия клавиш.

Основной защитой от фишинга являются спам-фильтры и антивирусы. Также активно используются плагины для браузеров, блокирующие сайты, попавшие в черные списки мошеннических ресурсов. К сожалению, программный инструмент для защиты от фишинга обладает ограниченной эффективностью, поскольку злоумышленники эксплуатируют в первую очередь не бреши в ПО, а человеческую психологию.

Сетевая разведка.

К этому типу относятся атаки, направленные на сбор информации о компьютерной системе, которая позже, на основании собранных данных, может быть подвергнута другой атаке – взлому. Поскольку не все виды атак применимы к одним и тем же операционным системам и сетевым приложениям, злоумышленникам важно провести разведку и узнать адреса компьютеров, версии ОС, порты приложений.

Для этих целей злоумышленники выполняют **сканирование** системы, т.е. с помощью специальных программ направляют в исследуемую систему запросы, формат которых может оказаться совместимым с форматом сообщений, используемым тестируемой системой. Получив такой запрос, атакуемая система может принять его за обычное протокольное сообщение и отправить ответ, из которого злоумышленник извлечет нужную ему информацию.

Другой популярной формой разведки является атака **«сканирование портов»**, когда на компьютер направляется последовательность TCP- и UDP-запросов для того, чтобы по ответам системы выяснить, какие порты ОС

открыты и тем самым узнать, какие приложения в системе установлены и активны. Т.к. многие вирусы для взаимодействия с внешним миром также используют определенные порты, то сканирование портов позволяет добыть информацию о заражении компьютера вирусами.

Для сканирования сетей хакеры используют как собственные утилиты, так и легальные программные средства, предназначенные для работы администраторов сетей. Такие средства часто оснащены удобным интерфейсом и позволяют преступникам воссоздать полную конфигурацию связей, адреса подсетей и узлов исследуемой сети, т.е. выполнить *network-mapping*.

Злоумышленники чаще сканируют тысячи сетей, пока не найдут компьютер с уязвимостями ОС и приложений или ранее внедренными вредоносными программами. Благодаря высокой степени стандартизации программного и аппаратного обеспечения и наличию большого числа однотипных вредительских программ, вероятность успеха такого поиска очень высока.

Полностью избавиться от сетевой разведки практически невозможно. Если к примеру, запретить пакеты ICMP-протокола на периферийных маршрутизаторах, вы избавитесь от нескольких видов тестирования, но потеряете возможность диагностики сетевых сбоев. И в любом случае это не решит проблему сканирования портов. Поэтому используют системы обнаружения вторжений, которые уведомляют администратора о ведущейся сетевой разведке.

Подслушивание (sniffing).

По большей части данные в компьютерных сетях передаются в открытом виде (незащищенной форме, открытым текстом), что позволяет злоумышленнику, получившему доступ к линиям передачи данных в вашей сети, подслушивать или считывать трафик. Для подслушивания в КС используют сниффер. *Сниффер пакетов* представляет собой прикладную программу, которая перехватывает все сетевые пакеты, передаваемые через определенный сетевой домен.

В настоящее время снифферы работают в сетях на вполне легальной основе: они используются для диагностики неисправностей и анализа трафика. Например, всем известная программа Wireshark. Однако, ввиду того, что некоторые сетевые приложения передают данные в текстовом формате (Telnet, FTP, SMTP, POP3 и т.д.), с помощью сниффера можно узнать полезную, а иногда и конфиденциальную информацию, например, логины и пароли пользователей.

Предотвратить угрозу сниффинга пакетов можно с помощью следующих мер и средств: применение для аутентификации однократных паролей; установка аппаратных или программных средств, распознающих снифферы; применение криптографической защиты каналов связи.

Посредничество.

Атака типа «посредничество» подразумевает активное подслушивание, перехват и управление передаваемыми данными невидимым промежуточным узлом. Когда компьютеры взаимодействуют на низких сетевых уровнях, они не всегда могут определить, с кем именно они обмениваются данными.

Для проведения известной атаки Man-in-the-Middle (человек посередине) злоумышленнику нужен доступ к пакетам, передаваемым по сети. Для атак этого типа часто используются снифферы пакетов, транспортные протоколы и протоколы маршрутизации.

В более общем случае атаки MiM проводятся с целью кражи информации, перехвата текущей сессии и получения доступа к частным сетевым ресурсам, для анализа трафика и получения информации о сети и ее пользователях, для проведения атак типа DoS, искажения передаваемых данных и ввода несанкционированной информации в сетевые сессии.

Одним из видов атак «посредничество» является *перехват сеанса (session hijacking)*. По окончании начальной процедуры аутентификации соединение, установленное законным пользователем, например, с почтовым сервером, переключается злоумышленником на новый хост, а исходному серверу выдается команда разорвать соединение. В результате «собеседник» законного пользователя оказывается незаметно подмененным.

Эффективно бороться с атаками типа MiM можно только с помощью криптографии и использования инфраструктуры управления открытыми ключами (PKI, Public Key Infrastructure).

Парольные атаки.

Целью этих атак является завладение паролем и логином законного пользователя. Злоумышленники могут проводить парольные атаки используя методы спуфинга, сниффинга или простого перебора. IP-спуфинг и сниффинг были рассмотрены ранее. Эти методы позволяют завладеть паролем и логином, если они передаются открытым текстом по незащищенному каналу.

Однако часто хакеры пытаются подобрать пароль и логин, используя для этого многочисленные попытки доступа. Такой подход носит название атака полного перебора (brute force attack). Для этой атаки используется специальная программа, которая пытается получить доступ к ресурсу путем подбора пароля на правах обычного пользователя.

Средства перехвата, подбора и взлома паролей в настоящее время считаются практически легальными и официально выпускаются достаточно большим числом компаний. Они позиционируются как программы для аудита безопасности и восстановления забытых паролей (например, Cain and Abel).

Пароль или ключ, к которому получает доступ атакующий, называется скомпрометированным.

Парольные атаки можно избежать путем использования одноразовых паролей и криптографической аутентификации, сведя практически на нет угрозу таких атак. К сожалению не все приложения, хосты и устройства поддерживают указанные методы аутентификации.

Атаки на уровне приложений.

Самый распространенный способ подобных атак состоит в использовании известных слабостей и уязвимостей серверного ПО и ОС. Сведения об уязвимостях широко публикуются, чтобы дать возможность администраторам вовремя исправить проблему, например, с помощью патчей и обновлений. Однако и злоумышленникам доступна эта информация, поэтому тут происходит извечная битва: кто быстрее?

Невозможно полностью исключить атаки на уровне приложений. Хакеры и эксперты в кибербезопасности постоянно открывают и публикуют все новые уязвимые места прикладных программ и ОС.

Для снижения угроз от атак этого типа важно осуществлять хорошее системное администрирование: анализировать журналы и log'и, использовать самое свежее ПО; вовремя устанавливать обновления; внедрять системы распознавания атак.

Рассмотренные типы атак на компьютерные сети возможны в силу ряда причин:

- использование общедоступных каналов передачи данных, а важнейшие данные передаются по сети в незашифрованном виде;
- уязвимости в процедурах идентификации, реализованных в стеке ТСП/IP. Идентифицирующая информация на уровне IP передается в открытом виде;
- отсутствие в базовой версии стека протоколов ТСП/IP механизмов, обеспечивающих конфиденциальность и целостность передаваемых сообщений;
- аутентификация отправителя в IP-сетях часто осуществляется по его IP-адресу. Процедура аутентификации выполняется только на стадии установления соединения, а в дальнейшем подлинность принимаемых пакетов не проверяется;
- отсутствие возможности контроля за маршрутом прохождения сообщений в сети Интернет, что делает удаленные сетевые атаки практически безнаказанными.

Примеры сетевых атак.

Рассмотрим наиболее распространенные примеры сетевых атак.

1. Прослушивание каналов связи:

- Сниффинг
- Спуфинг

2. Посредничество

- Подделка ТСП-сегмента
- Повторение ТСП-сегмента
- Сброс ТСП-соединения

3. Отказ в доступе:

- DoS
- DDoS

4. Проникновение в компьютерные сети:

- Парольные атаки
- Вирусы
- Эксплойты
- Инъекции

5. Сканирование компьютерных сетей

Прослушивание сетевого трафика.

Сниффинг – неавторизованный перехват сетевого трафика, использует перевод сетевой платы в смешанный режим работы. Возможность прослушивания канала связи (sniffing) в локальной сети организации. Для прослушивания трафика необходимо перевести сетевой адаптер в «беспорядочный» (promiscuous) режим. В данном режиме адаптер перехватывает все сетевые пакеты, проходящие через него, а не только предназначенные данному адресу, как в нормальном режиме функционирования. Если локальная сеть построена на концентраторах, то для злоумышленника оказывается доступным весь сетевой трафик в пределах сегмента локальной сети.

Пассивный перехват – скрытый несанкционированный мониторинг или прослушивание сети (снифферы общего вида).

В сети построенной на коммутаторах, трафик направляется только тому компьютеру, которому он предназначен. То есть если компьютер "А" обменивается пакетами с компьютером "В", то компьютер "С" не способен перехватывать этот трафик. Однако, существует ряд технологий позволяющих обойти ограничения, накладываемые коммутаторами.

Это спуфинг – ситуация, в которой один человек или программа успешно маскируется под другую путем фальсификации данных и позволяет получить незаконные преимущества. ARP Spoofing (ARP-poisoning), MAC Flooding и MAC Duplicating.

Они относятся к Активному перехвату – перехват, использующий создание скрытого канала связи.

Можно выделить следующие **методы прослушивания сети:**

- Снифферы общего вида
- ARP spoofing
- MAC Flooding
- MAC Duplicating
- Перенаправление трафика (ICMP-атака)

Для прослушивания сетевого трафика в сети, построенной на концентраторах злоумышленнику достаточно запустить на своем компьютере программу-сниффер и анализировать проходящие пакеты. Поскольку данная атака носит пассивный характер (нет непосредственного воздействия), то обнаружить ее достаточно тяжело.

Способы защиты от снифферов и их выявление

- Метод пинга
- Метод ARP
- Антиснифферы (Promiscan, Antisniff)
- Коммутируемая инфраструктура
- Криптография

Рассмотрим некоторые существующие методы определения наличия запущенного sniffера в локальной сети - это метод пинга, метод ARP.

Метод пинга (Ping method) использует уловку, заключающуюся в отсылке «ICMP Echo request» (Ping запроса) не на MAC-адрес машины, а на ее IP-адрес.

1. Допустим, хост, который мы подозреваем на использование sniffера, имеет IP-адрес 10.1.1.1 и MAC-адрес 00-40-05-A4-79-32.

2. Ваш компьютер должен находиться в том же сегменте ЛВС, что и подозреваемый компьютер.

3. Вы посылаете «ICMP Echo request», указав в запросе IP-адрес подозреваемого хоста и его слегка измененный MAC-адрес, например, 00-40-05-A4-79-33.

4. Каждый хост, получив данный запрос, сравнивает указанный в запросе MAC-адрес со своим MAC-адресом. В случае совпадения MAC-адресов, хост отвечает источнику запроса с помощью «ICMP Echo Reply», иначе пакет игнорируется. В данном случае, ни один из хостов в ЛВС не должен увидеть данный пакет.

5. Если же получен ответ от какого-либо хоста, это значит что у него не используется фильтр MAC-адресов, т.е. его сетевой адаптер находится в «беспорядочном режиме». Следовательно на данном хосте используется sniffer.

Метод пинга может быть перенесен на другие протоколы, которые генерируют ответы на запросы, например, запрос на установление TCP-соединения или запрос по протоколу UDP на порт 7 (эхо).

Метод ARP (ARP method) использует уловку, заключающуюся в посылке ложного широковещательного запроса.

1. Вы посылаете широковещательный ARP-запрос, но где вместо широковещательного адреса «FF:FF:FF:FF:FF:FF» указан адрес «FF:FF:FF:FF:FF:FE» (ложный широковещательный адрес, из которого вычли один бит). Поскольку адрес не является широковещательным, теоретически ни один из хостов не должен ответить на такой запрос. Однако практические эксперименты, что Windows 2000/XP/2003 при условии, что сетевой адаптер, работает в беспорядочном режиме, посчитает такой запрос широковещательным. Соответственно хост (А), на котором запущен sniffer, сравнив IP-адрес в запросе со своим IP-адресом, пошлет ответ ARP-reply. Таким образом, хост (А) выдаст, что он прослушивает весь сетевой трафик.

Отметим только, что данные методы в большинстве случаев позволяют лишь с некоторой вероятностью определить наличие sniffера.

Для более точного определения sniffеров используются антиснифферы.

Антиснифферы — аппаратные или программные средства, которые помогают распознать [снифферы](#).

С помощью sniffера в руках злоумышленника может оказаться конфиденциальная информация, например, имена пользователей и пароли. Одним из средств снижения угрозы sniffинга являются антиснифферы. Принцип работы антисниффера заключается в измерении времени реагирования [хостов](#) на сетевые запросы и определении, не приходится ли хостам обрабатывать «лишний» трафик.

Антиснифферы не могут полностью ликвидировать угрозу sniffинга, но они очень важны при построении комплексной системы защиты. Однако наиболее эффективной мерой, по мнению ряда специалистов, будет просто сделать работу sniffеров бессмысленной. Для этого достаточно защитить передаваемые по каналу связи данные современными методами [криптографии](#). В результате [хакер](#) перехватит не сообщение, а зашифрованный текст, то есть непонятную для него последовательность бит. Однако это не поможет скрыть сам факт передачи информации по сети и наличие канала связи между хостами.

AntiSniff — данная программа определяет машину в сети, которая собирает и анализирует данные (пакеты), для неё не предназначенные. Машина может находиться в таком состоянии по двум причинам — либо она собирает пароли и логины, либо её оператор по каким-то причинам решил проанализировать сетевой трафик. Как правило, в больших сетях, где требуется обеспечение безопасности данных, отслеживание таких машин — в режиме sniff'a — является крайне важным делом.

PromiScan — является инновационным прикладным программным обеспечением для того, чтобы удаленно контролировать компьютеры в локальных сетях и определить местонахождение интерфейсов сети, которые незаконно принимают все пакеты. PromiScan обнаруживает sniffеры и предупреждает администратора.

Спуфинг.

Метод **ARP-Spoofing** основан на подмене MAC адреса, а также атаке «человек посередине» (man-in-the-middle).

Данная атака возможна из-за уязвимости в реализации протокола ARP. Протокол разрешения адресов ARP (Address Resolution Protocol) предназначен для выяснения MAC-адреса хоста по его IP-адресу.

ARP - сетевой протокол, использующийся для связи IP-адреса устройства с его MAC адресом, тем самым делая возможным связь между устройствами сетевого уровня локальной и глобальной сети.

Для обмена информацией двум хостам в сети Ethernet, каждому из них необходимо получить MAC-адрес другого. Эта процедура осуществляется с использованием протокола ARP:

1. Хост «А», желающий установить соединение с хостом «В», сначала проверяет наличие MAC-адреса хоста «В» в своем ARP-кэше.
2. В случае его отсутствия в кэше, осуществляется рассылка широковещательного запроса с целью выявить MAC-адрес, соответствующий IP-адресу хоста «В».
3. Хост «В», сравнив IP-адрес в запросе со своим IP-адресом, посылает ответ (ARP-reply), в который помещает свой MAC-адрес.
4. Оба хоста «А» и «В» помещают полученные MAC-адреса в свои ARP-кэши, чтобы минимизировать количество широковещательных запросов.

Теперь хосты могут обмениваться данными, используя MAC-адреса.

Атаку на данный информационный обмен возможно произвести, потому что протокол ARP не требует аутентификации.

Для реализации атаки злоумышленнику с хоста «С» необходимо послать обоим хостам сгенерированные ARP-reply пакеты:

- для хоста «А», в котором прописано, что IP-адресу хоста «В» соответствует MAC-адрес хоста «С»;
- для хоста «В», в котором прописано, что IP-адресу хоста «А» соответствует MAC-адрес хоста «С».

Хосты «А» и «В» в соответствии со спецификацией протокола ARP, получив подобные reply-пакеты, обновят свои ARP-кэши.

Теперь, пакеты, отправляемые хостом «А» хосту «В» будут фактически отсылаться хосту «С», поскольку в ARP-кэше хоста «А» IP-адресу хоста «В» соответствует MAC-адрес хоста «С». Поэтому данная атака получила также название ARP-poisoning (отравление ARP-кэша). Для нормальной передачи пакетов между хостами «А» и «В» хосту «С» необходимо выполнять функции роутера для данных хостов, т.е. организовать их передачу по маршрутам А-С-В и В-С-А.

Отметим некоторые особенности реализации данной атаки:

- так как протокол ARP функционирует только рамках одной широковещательной подсети, атаку ARP-spoofing нельзя провести для хостов в разных подсетях или виртуальных локальных вычислительных сетях (VLAN);
- поскольку операционная система хостов периодически обновляет ARP-кэш, хосту «С» необходимо периодически выполнять процедуру «отравления кэша» для хостов «А» и «В»;
- в случае прослушивания трафика между некоторым хостом и роутером сети, в результате получается, что злоумышленник сможет прослушивать трафик между данным хостом и любым хостом в Интернет.

Следует отметить, что если на коммутаторе не включена функция Port-Security (данная функция будет рассмотрена позже), то можно в качестве MAC-адреса сниффера использовать любой MAC-адрес.

Защита от ARP-спуффинга

1. Отслеживание изменения ARP-таблицы.

2. Использовать статическую ARP таблицу. Можно избежать атаки ARP-spoofing путем настраивания ARP-таблицы вручную. Тогда злоумышленник не сможет обновлять ARP-таблицы путем отправки ARP-ответов на интерфейсы компьютеров.
3. Использование маршрутизаторов. Тут таблица маршрутизации основанная на IP адресах. И прописана статистически.
4. Использование VLAN. Если в локальной сети есть разделение на несколько VLAN, то атака ARP-spoofing может быть применена только к компьютерам, находящимся в одном VLAN. Идеальной ситуацией, с точки зрения безопасности, является наличие только одного компьютера и интерфейса маршрутизатора в одном VLAN. Атака ARP-spoofing для такого сегмента невозможна.
5. Использование VPN. Это каналы в которых трафик шифруется.
6. Использование систем обнаружения вторжений. Использование сетевых систем обнаружения вторжений, например ISS RealSecure, позволяет выявить ARP-атаку путем обнаружения в сети двух одинаковых IP-адресов.

Поскольку механизм атаки ARP-Spoofing основан на уязвимости в протоколе ARP, имеет смысл доработать данный протокол. Для ОС Linux есть утилита `Arp_antidote`, изменяющая реализацию протокола ARP в ОС таким образом, чтобы сделать данную атаку бессмысленной. Механизм обновленного протокола работает следующим образом. При приеме ARP-reply пакета производится сравнение старого и нового MAC-адреса, и при обнаружении его изменения запускается процедура верификации. Посылается ARP-запрос, требующий всем хозяевам IP-адреса сообщить свои MAC-адреса. В случае атаки ARP-Spoofing "настоящая" система, имеющая этот IP-адрес, ответит на запрос, и, таким образом, атака будет распознана. Если же изменение MAC-адреса было связано не с атакой, а со стандартными ситуациями, ответа, содержащего "старый" MAC-адрес, не будет, и по прошествии определенного таймаута система обновит запись в кэше. При обнаружении подозрительной ситуации ("двойника") ядро выводит сообщение: "ARP_ANTIDOTE: Possible MITM attempt!" и не обновляет запись ARP-кэша, а наоборот, прописывает старую запись как статическую.

Основным средством противостояния спуфингу IP-адресов источника является применение на маршрутизаторах техники «Проверки обратного пути» (Reverse path check, RPC). Идея этой проверки достаточно проста – пакет должен передаваться маршрутизатором в соответствии с его адресом назначения только в том случае, если его адрес источника имеется в таблице маршрутизации для интерфейса, с которого этот пакет получен. Действительно, если компьютер злоумышленника подключен к сети 212.100.100.0/24, но генерирует пакеты с адресом источника 25.0.30.18, то маршрутизатор провайдера, к которому подключена сеть 212.100.100.0/24 легко может проверить, что через интерфейс, на который был получен пакет с подделанным адресом, достичь сеть 25.0.30.18 нельзя, а значит пакет нужно

отбросить. Однако RPC может приводить к отбрасыванию пакетов легального пользователя, если его сеть имеет несколько подключений к сетям разных провайдеров.

Можно выделить следующие методы посредничества (MitM):

- Подделка TCP-сегмента
- Повторение TCP-сегментов
- Сброс TCP-соединения

Подделка TCP-сегмента.

Для защиты сегментов некоторого TCP-соединения от смешения с сегментами других соединений для каждого соединения случайным образом выбирается номер первого байта передаваемого потока данных. Затем каждый сегмент данных идентифицируется сдвигом относительно начала потока. В ходе переговорного процесса модули TCP обоих участвующих в обмене сторон договариваются между собой о параметрах процедуры обмена данными. Одним из таких параметров является *начальный номер байта*, с которого будет вестись отсчет в течение времени существования данного соединения. При приеме очередного сегмента протокол TCP проверяет, находится ли его порядковый номер в разрешенном для данного соединения диапазоне, и только в случае положительного результата такой проверки добавляет принятые данные к ранее принятым в ходе данного соединения TCP байтам.

Однако этот механизм защиты не так уж надежен, чем и пользуются злоумышленники. Атака «подделка TCP-сегмента» состоит в генерации сегментов TCP, все атрибуты которых имеют значения, легитимные для некоторого существующего TCP-соединения атакуемого компьютера, т.е. IP-адреса, номера TCP-портов источника и назначения, а также порядковые номера из текущего диапазона. Принимающая сторона не может отличить такие «поддельные» сегменты от настоящих и помещает информацию злоумышленника в поток пользовательских данных, а значит, злоумышленник может добиться желаемого эффекта, например поместить ложную информацию в базу данных, заразить атакуемый компьютер вирусом и т.п.

Для того чтобы «поддельный» сегмент выглядел как настоящий, атакующий может либо прослушивать трафик, либо просто перебирать все возможные значения адресов, портов и порядковых номеров сегментов.

Как правило, уязвимыми являются длительные TCP-соединения, например соединения, установленные для загрузки больших видеофайлов.

Повторение TCP-сегментов

Если злоумышленник смог каким-то образом перехватить трафик между двумя участниками TCP-соединения, то впоследствии он может просто повторно использовать перехваченные сегменты, пересылая их участникам соединения. Злоумышленник может использовать эту технику для разных целей, например он может вызвать таким образом нарушение работы

некоторого приложения, пользующегося TCP как транспортом, за счет представления устаревшей информации (перехваченной) как новой.

Сброс TCP-соединения

Эта атака является разновидностью предыдущей. Она использует флаг RST (ReSeT) в заголовке сегмента TCP. Этот флаг предназначен для аварийного прекращения TCP-соединения, при его приеме узел должен немедленно завершить сессию, к которой сегмент, несущий флаг RST, относится, и удалить все данные, полученные в ходе этого соединения. Разработчики протокола TCP ввели этот флаг для отработки аварийных ситуаций, например если в одном из узлов произошел сбой во время TCP-соединения, то после восстановления сбоя он может воспользоваться этим признаком и уведомить узел-собеседник, что сессия не может быть продолжена и все принятые ранее данные недействительны.

Для того чтобы атака удалась, злоумышленник должен «подделать» заголовок сегмента TCP, как и в предыдущем случае.

Очевидно, что данная атака, при реализации, может серьезно нарушить взаимодействие узлов компьютерной сети.

Интересно также, что техника «сброс соединения» используется не только злоумышленниками, но и разработчиками средств защиты, например, некоторые фаерволы используют ее для прекращения атаки. Известен также случай, когда ее применил провайдер (Comcast) для того, чтобы бороться с программами обмена файлами, работающими на пользовательских компьютерах и нарушающими авторские права владельцев аудио и звукозаписей. Данная практика была осуждена и признана незаконной Федеральной комиссией по связи США.

Борьба с данной атакой может вестись по двум направлениям:

- недопущение вспомогательной атаки – прослушивание трафика
- внесение изменений в протокол TCP, например, включением в него аутентификации каждого сегмента с использованием цифровой подписи. Для этого существует два стандарта, описывающие механизм цифровой подписи сегментов TCP: TCP MD5 (RFC 2386) и TCP AO (RFC 5925).

Атаки типа «отказ в доступе».

DDos (Distributed Denial of Service, распределённая атака типа «отказ в обслуживании») и **DoS** (Denial of Service, отказ в обслуживании) — [сетевая атака](#) на вычислительную систему с целью довести её до отказа, то есть создание таких условий, при которых легальные пользователи системы не могут получить доступ к предоставляемым системным ресурсам (серверам), либо этот доступ затруднён.

Если атака выполняется одновременно с большого числа компьютеров, говорят о DDoS-атаке.

Причины использования DDoS-атак:

- Личная неприязнь
- Развлечение

- Политический протест
- Недобросовестная конкуренция
- Вымогательство или шантаж

Классификация атак типа отказ в обслуживании.

- Атаки, основанные на насыщении полосы пропускания
- Атаки, основанные на недостатке ресурсов жертвы
- Атаки, основанные на ошибках в программном коде
- Атаки, основанные на маршрутизации и атаки на DNS-серверы

Атаки, основанные на насыщении полосы пропускания - атаки, связанные с большим количеством обычно бессмысленных или сформированных в неправильном формате запросов к компьютерной системе или сетевому оборудованию (flood-атаки).

HTTP-флуд и ping-флуд

Это самый примитивный вид DoS-атаки. ***ping-флуд*** Насыщение полосы пропускания можно осуществить с помощью обычных ping-запросов только в том случае, если канал атакующего намного шире канала компьютера-жертвы, скорость в котором. Но такая атака бесполезна против сервера, так как тот, в свою очередь, обладает довольно широкой полосой пропускания. Для атаки на сервер обычно применяется HTTP-флуд.

HTTP-флуд - атакующий шлёт маленький по объёму ***HTTP-пакет***, но такой, чтобы сервер ответил на него пакетом, размер которого в сотни раз больше. Даже если канал сервера в десять раз шире канала атакующего, то все равно есть большой шанс насытить полосу пропускания жертвы. А для того, чтобы ответные HTTP-пакеты не вызвали отказ в обслуживании у злоумышленника, он каждый раз подменяет свой ip-адрес на ip-адреса узлов в сети.

Smurf-атака (ICMP-флуд)

Smurf атака использует IP спуфинг для перенасыщения сети плохим ICMP трафиком и снижения пропускной способности сети. Это DDoS-атака, использующая функцию эхо-запроса протокола ICMP. Название атаки произошло от имени файла smurf.c, содержащего код атаки и получившего распространение в 1998 году.

Одна из самых опасных видов DoS-атак, так как у компьютера-жертвы после такой атаки произойдет отказ в обслуживании практически со 100 % гарантией. Злоумышленник использует широковебательную рассылку для проверки работающих узлов в системе, отправляя [ping-запрос](#). Очевидно, атакующий в одиночку не сможет вывести из строя компьютер-жертву, поэтому требуется ещё один участник — это усиливающая сеть. В ней по широковебательному адресу злоумышленник отправляет поддельный [ICMP пакет](#). Затем адрес атакующего меняется на адрес жертвы. Согласно спецификации протокола ICMP, получил «эхо-запрос» (тип сообщения номер 8) все узлы пришлют ей ответ с кодом 0 – т.н. «эхо-ответ». Поэтому ICMP-

пакет, отправленный злоумышленником через усиливающую сеть, содержащую 200 узлов, будет усилен в 200 раз. Поэтому для такой атаки обычно выбирается большая сеть, чтобы у компьютера-жертвы не было никаких шансов.

Атака Smurf использует тот факт, что эхо-запрос может быть послан не только по индивидуальному, но и широковещательному (broadcast) адресу некоторой сети. Например, если адрес сети 10.116.1.0/24, то ее широковещательным адресом будет 10.116.1.255, и пакет с таким адресом назначения должен быть доставлен всем узлам этой сети. В свою очередь, атака использует характерный прием – усиление атаки за счет отражения посланного пакета большим количеством компьютеров за счет подмены в заголовке широковещательного пакета адреса отправителя. Этот прием превращает DoS-атаку в DDoS без использования сети ботов, т.к. все компьютеры, отвечающие на эхо-запросы, работают в обычном режиме протокола ICMP.

Атака ICMP-smurf представляет сегодня скорее исторический интерес, т.к. до 1999 года широковещательная передача была обязательной для маршрутизаторов Интернета, но из-за атак, подобных ICMP-Smurf, в стандарты было внесено изменение и сейчас режимом по-умолчанию является фильтрация пакетов с широковещательными адресами. Кроме того, промежуточная сеть, узлы которой используются для отражения эхо-запроса, может быть экранирована с помощью файерволла от эхо-запросов, поступающих со стороны внешних сетей.

В том случае, когда атака начинается из сети, к которой принадлежит атакуемый компьютер, экранирование запросов не может быть выполнено. В этом случае атаку можно предотвратить, запретив компьютерам сети реагировать на широковещательные эхо-запросы.

UDP-затопление

Протокол UDP работает без установления соединений, это свойство используется при организации DoS-атака UDP-Flood, направленной на исчерпание пропускной способности интерфейса атакуемого компьютера, т.к. атакуемый компьютер обязан принимать все направленные ему UDP-дейтаграммы и не может заставить передающий компьютер ограничить скорость направленных ему пакетов, как это можно сделать в протоколе TCP, уменьшив размер окна приема. В этом отношении UDP DoS-атака похожа на атаку ICMP-флуд, т.к. заключается просто в направлении интенсивного потока UDP-дейтаграмм на атакуемый компьютер.

Злоумышленник может использовать аппаратный генератор трафика для того, чтобы генерировать UDP-трафик с максимально возможной скоростью выходного интерфейса, игнорируя ответные ICMP-сообщения в тех случаях, когда программный порт, указанный в пакетах UDP, у атакуемого компьютера не открыт.

Слабостью этого вида атак является принципиальное ограничение интенсивности атаки интерфейсом атакующего компьютера. Однако

злоумышленник может преодолеть это ограничение, заполучив в свое распоряжение сеть ботов.

Кроме прямой UDP-атаки, существуют также атаки отражения, когда UDP-трафик используется для инициирования большого количества ответных пакетов, которые и атакуют компьютер жертвы («переотражение»).

ICMP/UDP-затопление.

DoS-атака ICMP/UDP-flood имеет двойное имя, т.к. использует два протокола. Злоумышленник направляет UDP-пакеты, в которых в поле адреса источника указан адрес компьютера-жертвы, на программные порты компьютеров вспомогательной сети. В пакетах UDP указываются номера портов, находящихся в *пассивном состоянии*, т.е. с указанными портами не связаны приложения, слушающие сеть. При получении такого пакета компьютеры вспомогательной сети в соответствии с логикой работы стека TCP/IP отвечают подмененному источнику UDP-пакетов диагностическими сообщениями протокола ICMP “порт назначения недостижим”.

Обратите внимание, что если на прошлом слайде ICMP-ответы уходили в никуда, то здесь атака построена на отражении трафика от компьютеров вспомогательной сети на компьютер-жертву. В случае использования широковещательного адреса она становится DDoS-атакой.

В числе мер предотвращения входят те же меры, что и для ICMP Smurf-атаки, плюс пропуск файрволом только тех пакетов, порты которых соответствуют активным приложениям. Кроме того, можно ввести ограничение на интенсивность сообщений «*Порт назначения недостижим*».

Атака Fraggle (UDP/echo/chargen-затопление)

Атака Fraggle является полным аналогом Smurf-атаки, где вместо ICMP пакетов используются отраженные пакеты [UDP](#), направленные на определенные порты.

Принцип действия этой атаки простой: на 7 (*echo*) и 19 порт (*chargen*) жертвы отправляются [echo-команды](#) по широковещательному запросу. Согласно спецификации протоколов, сервис *chargen* при обращении к нему генерирует в ответ строку случайных символов случайной длины (от 0 до 512 байт) и посылает ее обратно обратившемуся хосту. Аналогичное назначение имеет сервис *echo*, он просто возвращает строку любого запроса по адресу обратившегося хоста.

Очевидно, что подмена ip-адреса злоумышленника на ip-адрес жертвы приводит бомбардировке атакуемого хоста множеством ответных сообщений. Их количество зависит от числа узлов в сети. Эта атака приводит к насыщению полосы пропускания и полному отказу в обслуживании жертвы. Если все же служба *echo* отключена, то будут сгенерированы ICMP-сообщения, что также приведёт к насыщению полосы.

Также, любопытной выглядит атака, когда атакующий посылает промежуточному хосту пакет, в котором в качестве порта назначения указывает номер 19, а в качестве порта-отправителя – 7. В этом случае единственный пакет атакующего вызывает бесконечный обмен пакетами

между сервисом chargen промежуточного хоста и сервисов echo атакуемого хоста.

Атака с помощью переполнения пакетами SYN (SYN-флуд)

Этот тип DoS-атаки активно применяется злоумышленниками на протяжении многих лет; впервые он был подробно описан в 1996 году, и уже в том же году началось его практическое применение, которое продолжается и по сей день.

Атака [SYN-флуд](#) использует уязвимость процедуры установления логического соединения протокола TCP. Соответственно атакующий использует некорректное установление TCP соединения.

Рисунок. Проведение DoS-атаки, в которой используются особенности протокола TCP (см. презентацию)

Для описания её действия можно остановиться на рассмотрении двух систем А и В, которые хотят [установить](#) между собой [TCP соединение](#), после которого они смогут обмениваться между собой данными. На установку соединения выделяется некоторое количество ресурсов, этим и пользуются DoS-атаки. Отправив несколько ложных запросов, можно израсходовать все ресурсы системы, отведённые на установление соединения. Рассмотрим подробнее, как это происходит. Хакер с системы А отправляет пакет SYN системе В, но предварительно поменяв свой IP-адрес на несуществующий. Затем, ничего не подозревая, компьютер В отправляет ответ SYN/ACK на несуществующий IP-адрес и переходит в состояние SYN-RECEIVED. Так как сообщение SYN/ACK не дойдет до системы А, то компьютер В никогда не получит пакет с флагом ACK. Данное потенциальное соединение будет помещено в очередь. Из очереди оно выйдет только по истечении 75 секунд. Этим пользуются злоумышленники и отправляют сразу несколько пакетов SYN на компьютер жертвы с интервалом в 10 секунд, чтобы полностью исчерпать ресурсы системы. Определить источник нападения очень непросто, так как злоумышленник постоянно меняет исходный IP-адрес.

Обычно атака SYN-flood обнаруживает себя несоответствием в трафике количества SYN-сегментов количеству ACK-сегментов, идущих от того же источника. Метод борьбы основан на фильтрации трафика от источника SYN flood пакетов, для чего нужно определить адрес атакующего узла. Очевидно, что при использовании спуфинга (подмены адреса отправителя) это сделать сложнее.

Атаки, основанные на недостатке ресурсов жертвы - атаки, связанные с перегрузкой системных ресурсов, таких как [оперативная](#) и физическая память, [процессорное время](#) и другие.

Отправка «тяжёлых» пакетов

Атакующий посылает серверу пакеты, требующие выполнения сложных вычислений со стороны сервера (канал обычно довольно широкий). Процессор сервера, когда будет их обрабатывать, может не справиться со сложными вычислениями. Из-за этого произойдёт сбой, и пользователи не смогут получить доступ к необходимым ресурсам.

Переполнение сервера лог-файлами

Лог-файлы сервера – это файлы, в которых записываются действия пользователей сети или программы. Неквалифицированный администратор может неправильно настроить систему на своём сервере, не установив определённый лимит. Злоумышленник отправляет большие по объёму пакеты, которые вскоре займут всё свободное место на жёстком диске сервера. Но эта атака сработает только в случае с неопытным администратором, квалифицированные хранят лог-файлы на отдельном системном диске.

Недостаточная фильтрация и проверка данных пользователя

Пользователь может загружать данные любого размера, например. Недостаточная проверка данных пользователя также приводит к бесконечному либо длительному циклу или повышенному длительному потреблению процессорных ресурсов (вплоть до исчерпания процессорных ресурсов) либо выделению большого объёма оперативной памяти (вплоть до исчерпания доступной памяти).

Атака второго рода

Это атака, которая стремится вызвать ложное срабатывание системы защиты и таким образом привести к недоступности ресурса. Обычно системы защиты настраиваются так, что если она вышла из строя, то все запрещает или наоборот все пропускает.

IP-атаки

Протокол IP сам по себе не предоставляет злоумышленникам много шансов для атак, т.к. он работает без установления соединений и достаточно просто в обработке как маршрутизаторами, так и конечными узлами. Тем не менее, некоторые атаки существуют.

IP-атаки: Атака IP-опций.

Эта атака представляет собой DoS-атаку на маршрутизаторы и использует поле дополнительных опций протокола IP.

В соответствии со стандартом RFC 791 заголовок IP-пакета версии 4 может включать *поле опций*, которые задают некоторую нестандартную обработку пакета маршрутизатором. Например, существует опция «строгая маршрутизация от источника», которая позволяет отправителю IP-пакета задать точный список адресов промежуточных маршрутизаторов, через которые должен проходить маршрут доставки пакета, в то время как опция «свободная маршрутизация от источника» задает только некоторые из промежуточных маршрутизаторов маршрута. Опция «фиксация маршрута» требует от маршрутизаторов фиксации в пакете адресов промежуточных маршрутизаторов, которые передавали пакет. Существует также возможность для производителей маршрутизаторов определять свои типы опций.

Атака основана на том факте, что у большинства IP-пакетов поле опций отсутствует, поэтому для продвижения таких пакетов использует специализированные процессоры портов, которые очень быстро и экономно выполняют свою операцию. А вот если встречается пакет с полем опций, то специализированный процессор его обрабатывать не может и передает пакет ЦП маршрутизатора, и обработка пакета существенно замедляется. В

результате поток пакетов, у которых присутствует одна или несколько опций, может привести к серьезному замедлению работы маршрутизатора, в предельном случае – к отказам в обслуживании нормальных пакетов. Усугубляет ситуацию присутствие в пакете двух взаимоисключающих опций, например «строгая маршрутизация от источника» и «свободная маршрутизация от источника» с разными промежуточными адресами.

Обычная практика борьбы с этой атакой – полная фильтрация пакетов, в заголовке которых имеются опции. Возможно также полное игнорирование поля опций. Существует также промежуточная тактика, когда маршрутизатор реагирует только на некоторые типы опций, а остальные игнорирует.

Атаки, основанные на ошибках в программном коде - атаки, связанные с ошибками в программном коде, использование которых приводит к аварийному завершению серверного приложения. Например, при передаче определенной последовательности.

Недостатки в программном коде

Обработка исключительных ситуаций всегда была головной болью для создателей операционных систем. Злоумышленники ищут ошибки в программном коде какой-либо программы либо операционной системы, заставляют её обрабатывать такие исключительные ситуации, которые она обрабатывать не умеет. За счёт этого возникают ошибки. Простым примером может служить частая передача пакетов, в которой не учитываются спецификации и стандарты [RFC-документов](#). Злоумышленники наблюдают за тем, справляется ли [сетевой стек](#) с обработкой исключительных ситуаций. Если нет, то передача таких пакетов приведёт к панике ядра ([kernel panic](#)) или даже к краху всей системы в целом.

Ping-смерти

К этому классу относится ошибка [Ping of death](#), распространённая в 1990-е годы. Эта атака с несколько драматическим названием «Пинг смерти» состоит в отправке на атакуемый компьютер эхо-запроса с длиной IP-пакета, превышающей его максимально возможный размер.

Длина пакета IPv4 по стандарту [RFC 791 IPv4](#) не может превышать 65 535 байт; компьютеру-жертве посылается [ICMP](#)-пакет большей длины, предварительно разбитый на части; Если соответствующий буфер ядра ОС не рассчитан на такой размер (из-за отсутствия соответствующих проверок в программном коде), то у жертвы от такого пакета [переполняется буфер](#), а ОС терпит крах, отчего и пошло название атаки.

Атака пинг смерти уже давно имеет только исторический интерес, т.к. разработчики уязвимых ОС в середине 90-х годов устранили недостаток в программном коде и ввели в стек IP необходимую проверку длины собираемого фрагментированного IP-пакета и тем самым ликвидировали саму основу атаки.

Другая ошибка тех времён – [WinNuke](#) ([Windows 95](#) неправильно обрабатывала редкий бит TCP-пакета URG).

Переполнение буфера

Переполнение буфера возникает в том случае, если программа из-за ошибки программиста записывает данные за пределами буфера. Допустим, программист написал приложение для обмена данными по сети, которое работает по какому-либо протоколу. В этом протоколе строго указано, что определённое поле пакета максимум может содержать 65536 байт данных. Но после тестирования приложения оказалось, что в её клиентской части в это поле нет необходимости помещать данные, размер которых больше 255 байт. Поэтому и серверная часть примет не более 255 байт. Далее злоумышленник изменяет код приложения так, что теперь клиентская часть отправляет все допустимые по протоколу 65536 байт, но сервер к их приёму не готов. Из-за этого возникает переполнение буфера, и пользователи не могут получить доступ к приложению.

IP-атаки: Атака IP-фрагментация.

Эта атака направлена на конечные узлы IP-сетей, в обязанность которых входит сборка фрагментированного IP-пакета в единое целое. Как показала практика, операция сборки имеет несколько уязвимостей, одна из которых уже была упомянута при описании атаки «пинг смерти», когда ОС терпит крах из-за превышения длины собранного пакета размера буфера.

Злоумышленник может использовать механизм фрагментации для атаки на конечный узел.

Превышение максимальной длины пакета (переполнение буфера сборки). Максимальное значение смещения фрагмента равно 65528. Т.к. максимальная длина IP-пакета 65535 байт, то очевидно, что последний фрагмент не должен иметь длину более 7 байтов. Задавая фрагмент с максимальным смещением и размером в 8 и более байтов, злоумышленник переполняет буфер ядра ОС, что может привести к краху ОС.

Перекрытие сегментов за счет специального подбора смещений и длин фрагментов. Некоторые ОС не справляются со сборкой таких пакетов и терпят крах. Атака Teardrop использует эту уязвимость.

Замещение фрагментов. Используется для обмана таких защитных средств, как файерволлы и системы обнаружения вторжений (IDS). Пакеты атаки фрагментируются и посылаются вместе с фрагментами-дубликатами, в которых содержится безобидная информация. Первым посылается безобидный фрагмент, а потом – фрагмент, содержащий код атаки, но с такими же смещением и длиной. В результате пришедший позже фрагмент атаки замещает безобидный фрагмент. Не все файерволлы и системы IDS распознают организованную таким образом атаку.

Незавершенные фрагменты. Эта DoS-атака направлена на исчерпание буферов сборки фрагментов. Атакующий посылает на атакуемый компьютер поток пакетов небольшого размера, при этом каждый пакет разбит на два фрагмента. Первый фрагмент из пары посылается с нулевым смещением, а второй – с максимально возможным, поэтому при сборке они занимают максимальную память, отводимую под буфер. Если количество фрагментированных таким образом пакетов достаточно велико, то за время

тайм-аута вся память ядра ОС, отводимая под сборку пакетов, оказывается исчерпанной и наступает отказ в обслуживании фрагментированных пакетов.

Способы защиты от DoS-атак

1. Функции антиспуфинга.

Это меры по борьбе с вторжениями, основанными на подделке исходящего IP-адреса. Смысл антиспуфинга заключается в запрете приёма пакетов с адресов, на которые запрос не отправлялся. Для этого могут применяться как специальные программные средства, так и точное конфигурирование сети (четкое указание диапазонов IP-адресов и исключение возможности отправки пакетов в Интернет от пользователей с несуществующими IP-адресами).

Правильная конфигурация функций анти-спуфинга на ваших маршрутизаторах и межсетевых экранах поможет снизить риск DoS. Эти функции, как минимум, должны включать фильтрацию RFC 2827. Если хакер не сможет замаскировать свою истинную личность, он вряд ли решится провести атаку.

Вы можете пресечь попытки спуфинга чужих сетей пользователями вашей сети (и стать добропорядочным "сетевым гражданином"). Для этого необходимо отбраковывать любой исходящий трафик, исходный адрес которого не является одним из IP-адресов вашей организации. Этот тип фильтрации, известный под названием "RFC 2827", может выполнять и ваш провайдер (ISP). В результате отбраковывается весь трафик, который не имеет исходного адреса, ожидаемого на определенном интерфейсе. К примеру, если ISP предоставляет соединение с IP-адресом 15.1.1.0/24, он может настроить фильтр таким образом, чтобы с данного интерфейса на маршрутизатор ISP допускался только трафик, поступающий с адреса 15.1.1.0/24. Заметим, что до тех пор, пока все провайдеры не внедрят этот тип фильтрации, его эффективность будет намного ниже возможной. Кроме того, чем дальше от фильтруемых устройств, тем труднее проводить точную фильтрацию. Так, например, фильтрация RFC 2827 на уровне маршрутизатора доступа требует пропуска всего трафика с главного сетевого адреса (10.0.0.0/8), тогда как на уровне распределения (в данной архитектуре) можно ограничить трафик более точно (адрес - 10.1.5.0/24).

2. Функции анти-DoS.

Правильная конфигурация функций анти-DoS на маршрутизаторах и межсетевых экранах может ограничить эффективность атак. Эти функции часто ограничивают число полуоткрытых каналов в любой момент времени. Время сессий.

3. Ограничение объема трафика.

Организация может попросить провайдера ограничить объем трафика. Этот тип фильтрации позволяет ограничить объем некритического трафика, проходящего по вашей сети. Обычным примером является ограничение объемов трафика ICMP, который используется только для диагностических целей. Атаки (D)DoS часто используют ICMP.

4. Устранение уязвимостей системы.
5. Наращивание ресурсов.
6. Рассредоточение. Построение распределённых и продублированных систем, которые не прекратят обслуживать пользователей даже если некоторые их элементы станут недоступны из-за атаки.
7. Использование систем обнаружения и предотвращения вторжений (IDS/IPS).

Проникновение в компьютерные сети

I. Парольные атаки возможны с помощью перебора, вирусов "троянский конь", IP-спуфинг и sniffing пакетов.

Методы, снижающие риск угрозы парольных атак.

1. Требования к сложности паролей.

1.1. Пароль не может содержать какие-либо части пользовательского имени;

1.2. Длина пароля должна быть не менее 6 символов;

1.3. Пароль обязательно должен быть составлен из следующих символов:

- символы латинского алфавита в верхнем регистре;
- символы латинского алфавита в нижнем регистре;
- цифры от 0 до 9;
- специальные символы, например, !, \$, #, %.

2. Периодическая смена паролей.

Причем, периодичность должна варьироваться для разных групп пользователей. Частая смена паролей, особенно к которым предъявлены требования повышенной сложности и большой длины, приведет к росту недовольства пользователей и к придумыванию ими различных схем упрощения данной процедуры. Например, пароли будут записываться на бумажки и прятаться в «надежных» местах, или будет придумана предсказуемая схема создания нового пароля из старого и т.д. Поэтому периодичность смены паролей рекомендуется устанавливать в 3-6 месяцев.

3. Надежное хранение паролей.

Это конечно организационная мера, а не техническая, однако важность ее в данном разделе безусловна. Действительно, даже самый сложный и длинный пароль легко скомпрометировать, если пользователь хранит его в легкодоступном месте.

II. Использование уязвимостей программных кодов. Нарушение защиты оперативной памяти, уязвимости контроля вводимых данных, скрытые коммуникации и скрытые каналы.

Программная система, состоящая из десятков тысяч строк кода, всегда имеет уязвимости, которые может использовать злоумышленник. Эти уязвимости, как правило, являются результатом ошибок программистов. Уязвимости могут быть и результатом плохого проектного решения или плохого управления проектом.

Существуют известные типичные уязвимости программного кода, которые полезно знать разработчикам для того, чтобы не допускать их появления в своих программах.

Уязвимости, связанные с нарушением защиты оперативной памяти.

Области оперативной памяти отдельных процессов защищены друг от друга, а также от области памяти ядра за счет архитектурных решения ОС. Однако, несмотря на различные меры, некорректное использование областей памяти все же может происходить в пределах адресного пространства одного процесса или в пределах ядра ОС. В последнем случае это особенно опасно, т.к. при этом крах может потерпеть вся система, а не отдельное приложение.

Переполнение буфера памяти является, наверно, наиболее часто используемой при атаках уязвимостью, связанной с нарушением защиты оперативной памяти. Например, пинг смерти. Механизм атаки типичен – он использует отсутствие контроля над вводимой из внешнего мира информацией, а именно не контролируется длина помещаемого в буфер пакета.

Переполнение стека является частным случаем переполнения буфера. Этот вид уязвимости часто используется для выполнения кода злоумышленника. Переполнение стека может произойти в случае, когда в области локальной памяти функции помещаются данные, длина которых оказывается больше длины этой области. В таком случае эти данные могут наложиться на адрес возврата функции и после завершения вызванной функции произойдет переход на некоторый адрес, который может быть случайным или специально сформированным злоумышленником, например вредоносный код.

Уязвимости контроля вводимых данных

Переполнение буфера является частным случаем уязвимостей, являющихся следствием слабого контроля вводимых данных. В более общем случае специальный вид вводимых данных может вызвать совершенно непредвиденные разработчиком последствия и этот факт может использовать злоумышленник. Обобщенно такой тип атаки называют *внедрением кода (code infection)*.

Внедряемый код может представлять собой программный код в классическом смысле этого термина, а может представлять и некоторую последовательность символов, неверно обрабатываемую программой. Это происходит в тех случаях, когда разработчик программы рассчитывает, что пользователи всегда будут вводить только «правильные» данные, т.е. данные только такого типа и из такого диапазона, который разработчик имел ввиду.

Специальные символы, встречающиеся в строке ввода, могут вызывать непредвиденные эффекты в поведении программы, если разработчик не учел такой возможности ввода. Например, пусть некоторый веб-сервис принимает запросы на загрузку файлов удаленных пользователей из домашних каталогов.

В случае, если разработчик при обработке имени файла не учел наличие точек и спец.символов, то он может посчитать имя /home/bob/files/../../../../etc/passwd легальным, принимая во внимание только его начальную часть /home/bob/files и передать его операционной системе. В свою очередь парсер ОС работает правильно, и в результате файл паролей систем будет передан злоумышленнику.

Существует большое количество различных типов атак внедрения кода в зависимости от атакуемого приложения и применяемого трюка. Очень распространены атаки на веб-сервисы и базы данных, что естественно, т.к. это наиболее популярные на сегодняшний день приложения.

III. Эксплойты это атаки основанные на эксплуатации различных дефектов в программном обеспечении (потому и получил такое название - от англ. эксплуатировать, использовать).

Эксплойты представляют собой вредоносные программы, реализующие известную уязвимость в ОС или прикладном ПО, для получения несанкционированного доступа к уязвимому хосту или нарушение его работоспособности.

Многие современные сетевые черви также используют эксплойты в составе своих функциональных компонентов для распространения в компьютерных сетях.

Современные программные продукты из-за конкуренции попадают в продажу с ошибками и недоработками. Разработчики, включая в свои изделия всевозможные функции, не успевают выполнить качественную отладку создаваемых программных систем. Ошибки и недоработки, оставшиеся в этих системах, приводят к случайным и преднамеренным нарушениям информационной безопасности. Например, причинами большинства случайных потерь информации являются отказы в работе программно-аппаратных средств, а большинство атак на компьютерные системы основаны на найденных ошибках и недоработках в программном обеспечении.

Таким образом, перед администраторами стоит проблема слежения за периодическим установлением обновлений и заплаток ПО, устраняющих известные уязвимости.

Пример нашумевшего в начале 2000 годов эксплойта KaHt2, реализующего одну из самых серьезных уязвимостей, найденную в то время в семействе операционных систем компании MS Windows. Данный эксплойт организывает атаку типа «отказ-в-обслуживании» (Denial-Of-Service, DoS) на службу «Удаленного вызова процедур» (Remote Procedure Call, RPC). Атаке уязвимы системы MS Windows NT/2000/XP/2003. Эксплойт KaHt2, реализует атаку на службу RPC, в результате которой осуществляется ошибка переполнения буфера, позволяющая злоумышленнику выполнить любой код на удаленной системе.

Другой, более современный пример эксплойта - Wannacry. Уязвимость ETERNALBLUE, которую использует WannaCry, заключается в реализации Windows протокола SMB. Он помогает различным узлам сети

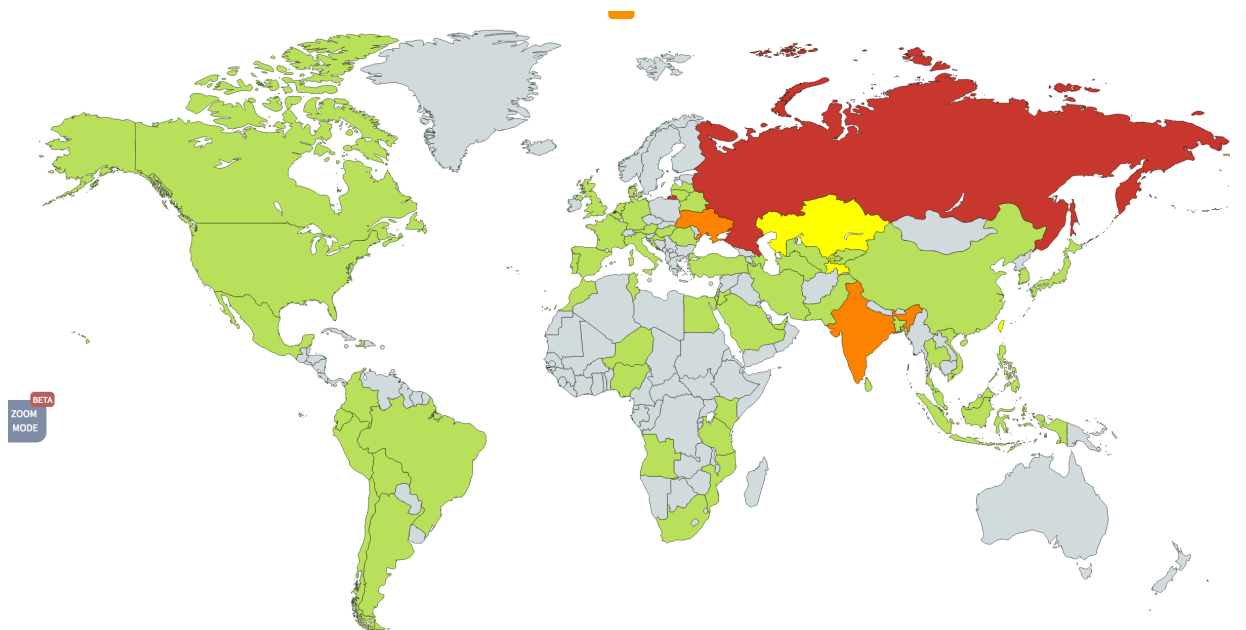
взаимодействовать, а реализацию Microsoft можно обмануть специально созданными пакетами для выполнения любого кода. Вредоносная программа WannaCry ищет уязвимые компьютеры, путем сканирования открытого извне TCP-порта 445 (Server Message Block/SMB).



Уязвимы были все версии Windows до Windows 10 уязвимы. WannaCry быстро распространился по компьютерным сетям мае 2017 года. Он заражал компьютеры, шифровал файлы на жестком диске ПК, а затем требовал выкуп в биткоинах за засшифровку. По иронии судьбы, патч, который защищает от WannaCry, был уже доступен до начала атаки. Microsoft в обновлении MS17-010, что вышло 14 марта 2017 года, исправил реализацию протокола SMB для Windows. Несмотря на критическое обновление, многие системы все еще не обновились до мая 2017 года. Эта атака заставила Microsoft выпустить патч даже для XP, что поддержка прекращена.

Считается, что Агентство национальной безопасности США обнаружило эту уязвимость уже давно. Вместо того, чтобы сообщить общественности, оно разработало код под названием EternalBlue. Этот эксплойт, в свою очередь, был похищен группой хакеров Shadow Brokers. После массового распространения WannaCry по всему миру Microsoft обвинила правительство США, что оно не поделилось информацией об этой уязвимости раньше.

Больше всего атак пришлось на Россию, но также от WannaCry серьезно пострадали Украина, Индия, Тайвань, всего же мы обнаружили WannaCry в 74 странах. И это за один только первый день атаки.



Действие всех эксплойтов сводится либо к получению удаленного доступа к атакуемой системе в виде командной оболочки, т.н. шелла (shell или rootshell), либо в удаленном выполнении какой-либо системной команды (например, добавление нового пользователя командой net user add), либо к вынужденной перезагрузке удаленной системы.

Последствия применения эксплойтов могут быть самыми критическими. В случае получения злоумышленником удаленного доступа к системе, он имеет практически полный (системный) доступ к компьютеру. Последующие действия злоумышленника и ущерб от них могут быть следующими:

- внедрение троянской программы. Блокируя работу антивируса, можно установить на скомпрометированной системе программу удаленного администрирования - так называемого троянского коня. Последствия использования данных программ будет рассмотрено в следующем параграфе;
- внедрение набора утилит для сокрытия факта компрометации системы, так называемых Rootkits;
- несанкционированное копирование злоумышленником данных с жестким и съемных носителей информации скомпрометированной системы;
- заведение на удаленном компьютере новых учетных записей с любыми правами в системе для последующего доступа как удаленно, так и локально;
- кража файла с хэшами паролей пользователей компьютера для их последующего подбора. В случае если скомпрометированной системой является доменный контроллер, то под угрозой оказываются все пользователи данного домена.
- уничтожение или модификация информации на удаленном хосте. Может привести к значительным финансовым или материальным потерям.
- осуществление действий от имени пользователя скомпрометированной системы.

IV. Внедрение вредоносных программ. Троянские программы, сетевые черви, вирусы.

Многочисленная группа атак связана с внедрением в компьютеры вредоносных программ (malware), к числу которых относятся троянские и шпионские программы, черви, вирусы, спам, логические бомбы и некоторые другие типы программ, нацеленные на нарушение безопасности. Вредоносный код чаще всего классифицируют по способу проникновения кода в чужой компьютер, а также по целевому назначению, так что один и тот же код может иметь по крайней мере два названия, например, червь и шпионская программа.

На практике злоумышленники часто сочетают в одной и той же программе различные типы угроз. Например, некоторые черви способны маскироваться как троянские программы или подобно вирусам заражать исполняемые файлы на локальном диске, а некоторые вирусы наделены способностями червей самокопироваться на другие компьютеры.

Ущерб, наносимый ВПО, может выражаться не только в уничтожении, искажении или похищении информации, приведении в нерабочее состояние ПО, а значит, и компьютера в целом, но и в значительных затратах времени и сил администраторов на обнаружение и распознавание атак, фильтрацию внешних сообщений, тестирование и перезагрузку систем. ВПО в начале этого десятилетия были одной из основных причин нарушения безопасности компьютерных сетей.

Троянские программы (Trojans) - вредоносные программы, основное предназначение которых незаметно проникнуть на компьютер под видом законной программы и выполнить вредоносные действия. Троянские программы могут применять в качестве прикрытия знакомые пользователю приложения, с которыми он работал и раньше, или принимать вид нового приложения, которые пытается заинтересовать жертву якобы полезными функциями и даже имитировать «бурную» деятельность. Однако суть троянской программы в любом случае остается вредительской: она может уничтожать или искажать информацию на диске, передавать данные (например, пароли) с зараженного компьютера на удаленный компьютер злоумышленника, приводить в неработоспособное состояние атакованный компьютер и т.д.

Троянские программы (также называемые троянцами или троянскими конями) состоят из двух частей: серверной (server) и клиентской (client).

Обычно троянские программы выполняют одну или несколько задач:

- предоставление удаленного доступа злоумышленнику (remote access).
- перехват и пересылка паролей.
- запись всех нажатий клавиатурных клавиш (keyloggers).
- уничтожение файлов, либо шифрование.
- создание платформы для распределенной DoS-атаки.

Основными способами проникновения троянских программ в настоящее время являются:

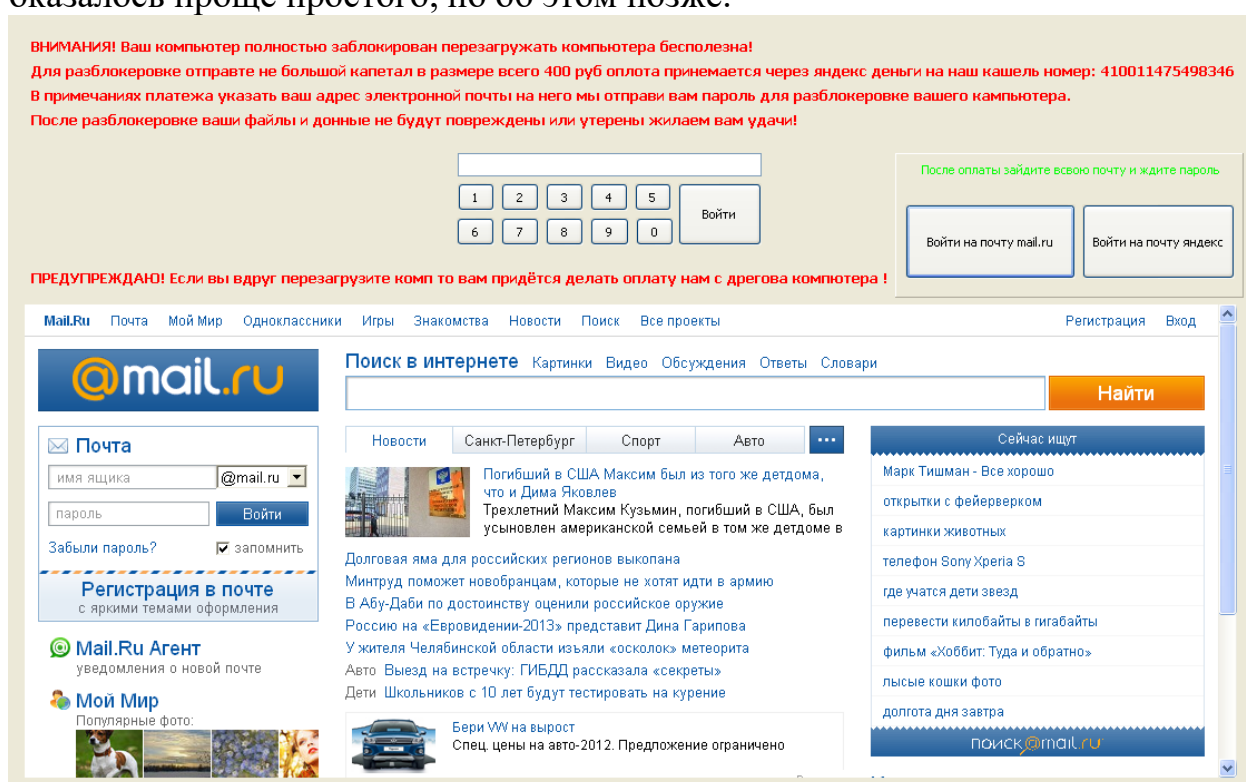
- Запуск вложений в письмах электронной почты;

- Запуск активного содержимого web-страниц неблагонадежных web-сайтов;
- Запуск непроверенных антивирусным ПО программ из внешних источников.

Методы защиты: антивирусы, персональные межсетевые экраны

При попытке программы - троянского коня осуществить выход в сеть, МЭ в соответствии с настроенными правилами его работы, либо блокирует данное обращение, либо выведет уведомление для текущего пользователя.

В качестве интересного примера можно рассмотреть троян 2013 года под названием Trojan.Winlock.8026, который вызывал у пользователей смех вместо тревоги. Этот троянец-блокировщик демонстрировал своим жертвам требования об оплате. Однако разблокировать компьютер от этого вируса оказалось проще простого, но об этом позже.



Троянец представляет собой примитивную форму, созданную с использованием среды разработки Delphi, код которой содержит не меньше нелепых ошибок, чем демонстрируемый на экране текст. Форма разработана с помощью стандартного конструктора Delphi, ничем не упакована, исполняемый файл вредоносной программы занимает более 7 Мб, а все ресурсы (включая код разблокировки) хранятся в приложении в открытом виде. По всей видимости, коварный злоумышленник создавал эту грозную вредоносную программу второпях, пока родители не вернулись с работы и не заставили его делать домашнее задание по русскому языку.

Впрочем, этот троян не представляет серьезной опасности и сам по себе: если по ошибке или в силу роковой случайности вы запустили данную вредоносную программу на своем компьютере, то, хорошенько отсмеявшись, воспользуйтесь кодом 141989081989 для его разблокировки.

Сетевые черви (worm) – это программы, способные к самостоятельному распространению своих копий среди узлов в пределах локальной сети, а также по глобальным связям, перемещаясь от одного компьютера к другому без всякого участия в этом процессе пользователей сети.

Поскольку большинство сетевых червей передаются в виде файлов, основным механизмом их распространения являются сетевые службы, основанные на файловом обмене. Так, червь может рассылать свои копии по сети в виде вложений в сообщения электронной почты или с помощью размещения ссылок на зараженный файл на каком-либо веб-сайте. Однако существуют и другие разновидности червей, которые для своей экспансии используют более сложные приемы, например, связанные с ошибками в ПО. Этот вид ВПО использует для распространения рассмотренные ранее эксплойты.

Главная цель и результат деятельности червя состоит в том, чтобы передать свою копию на максимально возможное число компьютеров. При этом для поиска компьютеров – новых потенциальных жертв – черви задействуют встроенные в них средства. Типичная программа-червь не удаляет и не искажает пользовательские и системные файлы, не перехватывает электронную почту пользователей, не портит содержимое баз данных, а наносит вред атакованным компьютерам потреблением их ресурсов, например для рассылки спама или проведения массовой атаки в составе ботнета.

При создании типичного сетевого червя хакер прежде всего определяет перечень сетевых уязвимостей, которые он собирается использовать для проведения атак. Такими уязвимостями могут быть известные, но неисправленные на некоторых компьютерах ошибки в ПО. Чем шире перечень уязвимостей, тем больше узлов может быть поражено данным червем.

Червь состоит из *двух основных функциональных компонентов* – атакующего блока и блока поиска целей.

Атакующий блок состоит из нескольких модулей (векторов атаки), каждый из которых рассчитан на поражение конкретного типа уязвимости. Этот блок передает свою копию на атакуемый хост.

Блок поиска целей (локатор) собирает информацию об узлах сети, а затем на основании этой информации определяет, какие из исследованных узлов обладают теми уязвимостями, для которых хакер имеет средства атаки.

Некоторые черви нагружены их создателями и другими вспомогательными функциями.

Упрощенно жизненный цикл червя может быть описан рекурсивной процедурой, состоящей из циклического запуска локатора и атакующего блока на каждом из последующих заражаемых компьютеров.

Рисунок Экспансия червя в сети (см. презентацию).

В начале каждого нового цикла червь запускает локатор для поиска и формирования списка узлов-целей, пригодных для проведения атак, а затем, используя средства атакующего блока, пытается эксплуатировать уязвимости узлов из этого списка. В результате успешной атаки червь копирует все свои

программы на «новую территорию» и активирует локатор. После этого начинается новый цикл. На рисунке показано как червь лавинообразно распространяется по сети.

Локатор идентифицирует цели по адресам электронной почты, IP-адресам, характеристикам установленных на хостах операционных систем, номерам портов, типам и версиям приложений. Для сбора информации локатор может предпринимать действия, связанные как с поисками интересующих данных на захваченном им хосте, так и зондированием сетевого окружения.

Компьютерные пользователи старшего возраста должны помнить w32.Blaster.worm - червя, который заражал Windows от 2000 до XP и не пускал в интернет, перезагружая компьютер.

Blaster, также известный как Lovsan, Lovesan или MSBlast – компьютерный червь, распространявшийся на машинах, работавших с операционными системами Windows 2000 и Windows XP. Эпидемия этого червя наблюдалась в августе 2003 года. Во второй половине 2003 года этот червь заразил не менее 100 000 персональных компьютеров по всей планете.

История червя Blaster начинается с того момента, когда коллектив разработчиков Xfocus нашёл уязвимость в операционных системах Windows, связанную с переполнением буфера. Уязвимость в сервисе RPC DCOM, присутствовала в то время во всех операционных системах семейств Windows 2000, Windows XP и Windows 2003. Эта уязвимость вызывается соответствующим образом составленным TCP/IP пакетом, пришедшим на порт 135, 139 или 445 атакуемого компьютера. Она позволяет как минимум провести DoS-атаку (в данном случае - атакуемый компьютер перезагружается), а как максимум - выполнить в памяти атакуемого компьютера любой код. Эта уязвимость послужила поводом к созданию множества вредоносных программ, из которых наиболее известным стал червь Blaster.

Первая волна заражения данным червём прошла 11 августа по компьютерным сетям США. В отличие от других червей заражение Blaster проходило не при контакте с заражённым файлом, а случайным образом. Попадая в компьютер, вирус начинал генерировать случайные IP-адреса, и, сгенерировав адрес, искал уязвимости в системе жертвы, а найдя — заражал компьютер. Далее цикл повторялся.

От такого способа распространения вредоносной программы пострадало множество компьютеров. По отчётам из Лаборатории Касперского — по всему миру было заражено порядка 300 тысяч компьютеров, из которых 30 тысяч в России. Для пользователя данный червь был сравнительно безопасен, если не считать побочного эффекта в виде регулярной перезагрузки компьютера. В результате своей деятельности червь приводит к нестабильной работе службы RPC. После сообщения об ошибке в работе службы компьютер начинал перезагружаться через произвольные интервалы времени.

Целью этого червя являлась атака на серверы Microsoft 16 августа 2003 года в полночь. Однако Microsoft временно закрыла свои серверы, что

позволило сократить ущерб от вируса к минимуму. Blaster в своём коде содержал скрытое послание, адресованное Биллу Гейтсу: «Билли Гейтс, зачем вы делаете это возможным? Хватит делать деньги, исправьте ваше программное обеспечение!»

Вирусами называются вредоносные программы, которые внедряются в другие программы для выполнения определенной нежелательной функции на рабочей станции конечного пользователя.

В отличие от червей вирусы не содержат в себе встроенного механизма активного распространения по сети, они способны размножаться своими силами только в пределах одного компьютера. В свою очередь, передача вируса на другой компьютер происходит с участием пользователя. Например, пользователь может записать свой файл, зараженный вирусом, на сетевой файловый сервер, откуда тот может быть скопирован всеми пользователями, имеющими доступ к данному серверу. Пользователь может также передать другому пользователю съемный носитель с зараженным файлом или послать такой файл по электронной почте. То есть именно пользователь является главным звеном в цепочке распространения вируса за пределы своего компьютера.

Тяжесть последствий вирусного заражения зависит от того, какие вредоносные действия были запрограммированы в вирусе злоумышленником.

Как правило, применение вирусов позволяет достигнуть следующих целей:

- Уничтожение или непоправимое изменение текстовых документов, исполняемых файлов, баз данных;
- Нарушение работоспособности всей корпоративной сети и отдельных элементов: серверов, рабочих станций.

«Ярлыки на флешке вместо папок» достаточно частое явление, возникшее в результате халатности многих пользователей, не использующих антивирусное ПО на своих компьютерах.

Этот тип вируса дублирует ваши файлы и папки, затем прячет и заменяет их. Вирус представляет комбинацию вирусов трояна и червя. Опасность заключается в том, что вы запускаете вирус каждый раз, когда хотите открыть ваш файл или папку. После запуска вирус распространяет себя заражая все большее количество файлов и часто устанавливает дополнительно вредоносное ПО которое может украсть данные о паролях и кредитных картах, сохраненных у вас на компьютере.

V. Скрытые коммуникации и скрытые каналы

Скрытый канал (covert channel) – это коммуникационный канал, пересылающий информацию методом, который изначально для этого не был предназначен.

Техника скрытых каналов основывается на том простом факте, что любое изменение состояния какого-то объекта несет информацию. Обнаружить закономерность в изменениях состояния объекта, которое

изначально никак не было предназначено для передачи информации, бывает очень сложно. За высокую степень скрытности многие каналы такого типа расплачиваются низкой скоростью передачи информации. Например, наличие или отсутствие цветка на подоконнике несет в себе 1 бит информации.

В рассматриваемом контексте скрытый канал – это механизм для передачи информации, не предусмотренный разработчиком информационной системы. Очевидно, что передача данных по скрытому каналу не контролируется обычными механизмами безопасности ОС, такими как аутентификация и авторизация, именно поэтому наличие скрытых каналов очень опасно. Недаром все стандарты безопасности ИС уделяют скрытым каналам большое внимание и требуют анализа системы на наличие таких каналов при сертификации.

Кроме скрытых каналов, существуют также **скрытые коммуникации**. Они используют для передачи сообщений легальные каналы, однако действуют таким образом, что эти сообщения незаметны для легальных пользователей, т.к. они скрыты внутри других сообщений, используемых как контейнеры.

Общей особенностью скрытых каналов и скрытых коммуникаций является то, что здесь скрывается не только содержание сообщения, но и сам факт коммуникации.

Скрытыми коммуникациями занимается *стеганография*, поэтому такой способ передачи секретной информации иногда называется *стегоканалом*.

Примеров скрытых каналов бесчисленное множество. Стоит изучить многочисленные исследования кибер-специалистов из Университета имени Бен-Гуриона (Израиль), занимающиеся вопросами защиты информации на, казалось бы, неуязвимых системах - физически изолированных компьютеров.

Например, разработанная ими технология aIR-Jumper позволяет обнаруживать невидимый для невооруженного глаза инфракрасный свет и создавать с его помощью «скрытые двунаправленные беспроводные каналы доступа к изолированным сетям, не имеющим выхода в Интернет». Ученые доказали, что посредством вредоносного программного обеспечения злоумышленники могут удаленно контролировать интенсивность ИК-света, незаметно передавать и записывать в такой способ сигнал через самую обычную камеру и затем расшифровывать эти данные, получая в свое распоряжение различную, в том числе и конфиденциальную информацию: коды, пароли и закодированные ключи.

Другой пример скрытого канала основан на возможностях использования сетевых средств и протоколов. Например, *скрытый временной канал*, который может быть построен на основе синхронизации обращений к некоторому системному ресурсу, например, к сокету TCP или UDP. Процесс-получатель считывает периоды занятости определенного сокета и получает таким образом информацию. В другом случае процесс может кодировать информацию, посылая пакеты на удаленный хост в определенные моменты времени.

Наиболее популярным примером скрытых коммуникаций является помещение секретного сообщения в биты цифровой фотографии, внешний вид которой от такой операции практически не отличается от исходного. Существует большое количество программ, в т.ч. и бесплатных (например, EZStego, Xiao Steganography и др.), которые умеют делать это для различных форматов цифровых изображений. Для того чтобы такой канал стал действительно тайным, сама по себе передача фотографий не должна вызывать подозрений. Так, отправка кошек настолько популярна в соц.сетях, что их можно использовать как очень хороший контейнер. Предлагаю лишний раз задуматься перед тем, как выкладывать очередную фотографию в инстаграмм... а не могла ли какая-либо программа скрыть в этой передаче критическую для вас информацию?

Еще один пример скрытых коммуникаций – параметр ISN в пакете TCP SYN, указывающий начальный номер последовательности, используя который, а именно кодируя его определенным образом, можно передавать скрытую информацию по установленному легальному соединению.

VI. Программные закладки.

Программные закладки – это внесенные в ПО функциональные объекты, которые при определенных условиях инициируют выполнение не описанных в документации функций, позволяющих осуществлять несанкционированные воздействия на информацию.

Функции, описание которых отсутствует в документации на программу, называют недекларированными возможностями (НДВ) программ.

Подразумевается, что закладка наносит какой-то ущерб системе, на которой она установлена, т.е. является по сути вредоносным кодом, замаскированным в глубине полезного программного продукта.

В то же время, НДВ программы не обязательно являются вредоносными, они могут быть просто дополнительными функциями, которые разработчик программы решил включить для отладки, но не стал их описывать для рядовых пользователей. Это могут быть и просто забытые функции. Существует также класс НДВ программы, внедряемых в нее для развлечения пользователя и самих программистов – это т.н. «пасхальные яйца», Easter eggs. Пасхалки прерывают нормальную работу пользователя и приветствуют его интересной картинкой или сообщением, а то и приглашением поиграть в игру.

Однако, основная роль программных закладок, к сожалению, состоит в выполнении различной вредоносной работы:

- шпионить за действиями пользователя и передавать эту информацию;
- получать доступ к конфиденциальной информации;
- искажать и разрушать данные и т.д.

В России наиболее известным случаем использования программных закладок является афера программиста ОАО "Акционерный капитал", реестродержателя акций ОАО «Татнефть», похитившего в 2005 году с

помощью специально разработанного программного модуля 110 тысяч акций ОАО «Татнефть» на 5,7 млн руб и осужденного за это на пять с половиной лет.

Мошенники "одалживали" акции ОАО «Татнефть», снимая их со счетов, прокручивали на ММВБ, после чего возвращали, оставляя себе прибыль. Для этого в компьютерной программе Kapital, проводящей операции с ценными бумагами, программист-инсайдер создал специальную папку. В ней был файл-модуль с функцией Take CB. Она выбирала из базы данных владельцев акций "Татнефти" физических лиц, имевших не меньше 300 акций. Затем со счета каждого из них она переводила по 100 акций на свои счета. Затем ценные бумаги перемещались на ММВБ. Доходы, полученные от операций на бирже, перечислялись на карточные счета в казанском филиале банка "Зенит". Эти счета были также зарегистрированы на подставных лиц, которые снимали с них деньги и передавали мошенникам.

Основной задачей изобретения программиста-инсайдера было ввести в заблуждение компьютерную систему реестродержателя, чтобы она не могла заметить исчезновения акций со счетов законных владельцев. Его программа работала так, что после похищения ценных бумаг на счету акционера появлялся знак "минус". Когда же акции после операций на ММВБ возвращались на место, этот знак исчезал. А компьютерная система ОАО "Акционерный капитал" была создана таким образом, что реагировала только на положительные цифры.

Как установило следствие, господина Федотов и Белов совершали мошеннические операции с января по июнь 2005 года, похитив в общей сложности 110 тыс. акций ОАО "Татнефть" на 5,7 млн руб. Задержали мошенников в мае 2006 года. Суд назначил Александру Федотову семь лет лишения свободы с отбыванием наказания в колонии общего режима, а Евгению Белову — пять с половиной лет.

Чем можно защищаться? Принципиально можно выделить два направления защиты от программных закладок: *организационный* и *технический*. Рассмотрим их немного подробнее.

Организационный способ защиты. Привычный и всем понятный организационный подход к защите чаще всего применяется в крупных компаниях, ведущих собственные дополнительные разработки к крупным приложениям. Ключевым моментом этого подхода является разделение жизненного цикла продукта на три этапа: разработка, тестирование и непосредственное использование.

На первом этапе пишется код, на втором тестируется корректность реализации функционала и наличие ошибок, в том числе и с точки зрения безопасности (всевозможные тесты на проникновение). На обоих этапах хорошей практикой является привлечение в проектную команду консультантов по безопасности. После положительного заключения тестировщиков программное обеспечение передается во внедрение.

Таким образом, достигается разделение разработки, контроля качества и пользователей программного обеспечения. В результате вероятность успешной незаметной установки и последующего использования

программной закладки серьезно падает. Кроме того, для проведения успешной аферы нужно будет вовлечь в преступный сговор сотрудников на нескольких этапах, что может быть очень рискованно.

Технический способ защиты. Помимо организационного подхода существует и технический, когда разработанное самостоятельно или на заказ программное обеспечение отдается на аудит кода внешним экспертам, которые специализируются на подобных услугах. Это происходит чаще всего в тех случаях, когда разработку заказывают у третьей компании.

В качестве технических средств для поиска программных закладок могут использоваться сканеры кода, которые способны по определенным алгоритмам обнаруживать подозрительные куски кода.

Кроме этого, для обнаружения закладок могут использоваться тестовые среды и анализаторы потенциальных уязвимостей. Например, для SAP R/3 подобные технические средства и услуги предоставляют компания Onapsis и Virtual Forge, которые используют статический и динамический анализ кода.

До недавнего времени практически не было продуктов, которые бы доступны по цене и функционалу компаниям, разрабатывающим продукт для собственных нужд. С приходом в аудит кода технологий, прежде принятых в DLP-продуктах (лингвистический анализ и «цифровые отпечатки») стал возможен статический анализ кода на совпадение с шаблонами известных уязвимостей и закладок «на лету». Плюс этого метода в оперативности (код анализируется практически мгновенно и вне зависимости от логики программы) и отсутствия необходимости запуска программы на исполнение. Анализировать можно как готовые программы, так и произвольные куски кода, вплоть до одной строчки. Минусом является необходимость постоянного пополнения базы известных уязвимостей (с каждой уязвимости снимается уникальный отпечаток, который затем сравнивается с отпечатками нормализованного текста программы).

Такой анализ дает возможность сотрудникам, ответственным за безопасность и качество продуктов без привлечения сторонних аудиторов кода находить и обезвреживать часто встречающиеся закладки, типа обхода тестовой среды («если среда тестовая, то эту функцию не запускать»), вход в систему в обход авторизации, опасные манипуляции с критическими данными и другие. Одной из самых востребованных функций здесь является «поиск подобного», которая позволяет при обнаружении специфической закладки проверить наличие подобных операций во всем коде приложения.

VII. Аппаратные закладки.

Существует также класс *аппаратных закладок* (hardware backdoor) – устройство в электронной схеме, скрытно внедряемое к остальным элементам (или/и также аппаратной закладкой называется отдельная плата, микросхема, подключаемая к атакуемой системе или ее IT инфраструктуре). Фактически, аппаратной закладкой является скрытое техническое приспособление, своего рода жучок, который позволяет получить доступ к цели или сведения о ней.

Программные и аппаратные "закладки", в ряде случаев выполняют одинаковое предназначение, но аппаратная закладка может вообще не использовать ресурсы атакуемой системы, что делает ее практически "неуязвимой".

История с копировальным аппаратом ксерокс в посольстве СССР в США является наглядным примером использования аппаратных закладок.

Среди документов опубликованных Эдвардом Сноуденом, бывшим сотрудником ЦРУ и Агентства национальной безопасности США, были обнаружены материалы описывающие некоторые детали технологий шпионажа используемых АНБ. Список программных и аппаратных средств оформлен в виде небольшого каталога. Всего сорок восемь страниц отмеченных грифами «Секретно» и «Совершенно секретно», на которых дано краткое описание той или иной технологии для слежки. Данный список не является исчерпывающим. Представлены техники связанные с получением скрытого доступа к вычислительной технике и сетям, а также способы и устройства радиоэлектронной разведки связанные с мобильной связью и оборудование для наблюдения.

COTTONMOUTH-I аппаратная закладка на USB, предоставляющая беспроводной мост к целевой сети, а также загрузки эксплойтов на целевой системе. Может создавать скрытый канал связи для передачи команд и данных между аппаратными и программными закладками. При помощи встроенного радиопередатчика может взаимодействовать с другими COTTONMOUTH. В основе лежит элементная база TRINITY, в качестве радиопередатчика используется HOWLERMONKEY. Существует версия под названием MOCCASIN, представляющая собой закладку в коннекторе USB-клавиатуры.

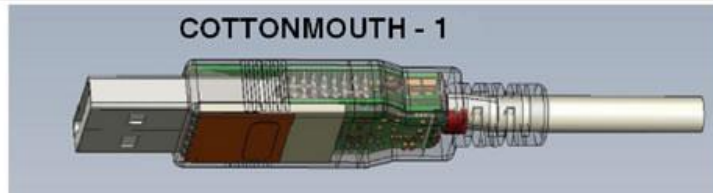


COTTONMOUTH-I

ANT Product Data

(TS//SI//REL) COTTONMOUTH-I (CM-I) is a Universal Serial Bus (USB) hardware implant which will provide a wireless bridge into a target network as well as the ability to load exploit software onto target PCs.

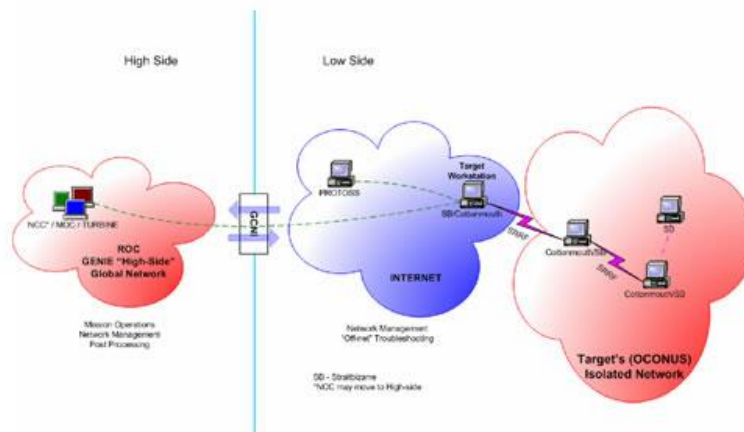
08/05/08



(TS//SI//REL) CM-I will provide air-gap bridging, software persistence capability, "in-field" re-programmability, and covert communications with a host software implant over the USB. The RF link will enable command and data infiltration and exfiltration. CM-I will also communicate with Data Network Technologies (DNT) software (STRAITBIZARRE) through a covert channel implemented on the USB, using this communication channel to pass commands and data between hardware and software implants. CM-I will be a GENIE-compliant implant based on CHIMNEYPOOL.

(TS//SI//REL) CM-I conceals digital components (TRINITY), USB 1.1 FS hub, switches, and HOWLERMONKEY (HM) RF Transceiver within the USB Series-A cable connector. MOCCASIN is the version permanently connected to a USB keyboard. Another version can be made with an unmodified USB connector at the other end. CM-I has the ability to communicate to other CM devices over the RF link using an over-the-air protocol called SPECULATION.

COTTONMOUTH CONOP
INTERNET Scenario



Status: Availability – January 2009

Unit Cost: 50 units: \$1,015K

POC: [REDACTED], S3223, [REDACTED], [REDACTED]@nsa.ic.gov
ALT POC: [REDACTED], S3223, [REDACTED], [REDACTED]@nsa.ic.gov

Derived From: NSA/CSSM 1-52
Dated: 20070108
Declassify On: 20320108

COTTONMOUTH-II аппаратная USB-закладка предоставляющая скрытый канал доступа к сети цели. Данная закладка предназначена для работы на шасси компьютера и представляет собой двухпортовый USB-коннектор на плату. Может создавать скрытый канал связи для передачи команд и данных между аппаратными и программными закладками.



RAGEMASTER — аппаратная закладка позволяющая перехватить сигнал от VGA монитора. Закладка прячется в обычный VGA-кабель соединяющий видеокарту и монитор, установлена, как правило, в феррит на видеокабеле. Реализован захват сигнала с красного цветового канала. Представляет собой пассивный отражатель, т.е. активируется при облучении радиосигналом от специализированного излучателя.





RAGEMASTER

ANT Product Data

(TS//SI//REL TO USA,FVEY) RF retro-reflector that provides an enhanced radar cross-section for VAGRANT collection. It's concealed in a standard computer video graphics array (VGA) cable between the video card and video monitor. It's typically installed in the ferrite on the video cable.

24 Jul 2008

(U) Capabilities

(TS//SI//REL TO USA,FVEY) RAGEMASTER provides a target for RF flooding and allows for easier collection of the VAGRANT video signal. The current RAGEMASTER unit taps the red video line on the VGA cable. It was found that, empirically, this provides the best video return and cleanest readout of the monitor contents.



(U) Concept of Operation

(TS//SI//REL TO USA,FVEY) The RAGEMASTER taps the red video line between the video card within the desktop unit and the computer monitor, typically an LCD. When the RAGEMASTER is illuminated by a radar unit, the illuminating signal is modulated with the red video information. This information is re-radiated, where it is picked up at the radar, demodulated, and passed onto the processing unit, such as a LFS-2 and an external monitor, NIGHTWATCH, GOTHAM, or (in the future) VIEWPLATE. The processor recreates the horizontal and vertical sync of the targeted monitor, thus allowing TAO personnel to see what is displayed on the targeted monitor.

Unit Cost: \$ 30

Status: Operational. Manufactured on an as-needed basis. Contact POC for availability information.

POC: [REDACTED], S32243, [REDACTED], [REDACTED]@nsa.ic.gov

Derived From: NSA/CSSM 1-52
Dated: 20070108
Declassify On: 20320108

VIII. SQL Инъекция это возможность выполнить произвольный запрос к базе данных (например, прочитать содержимое любых таблиц, удалить, изменить или добавить данные). Говоря простым языком — это атака на базу данных, которая позволит выполнить некоторое действие, которое не планировалось создателем скрипта.

Соответственно злоумышленник может получить возможность чтения и/или записи локальных файлов и выполнения произвольных команд на атакуемом сервере предоставив себе доступ изменением таблицы.

Атака типа внедрения SQL может быть возможна из-за некорректной обработки входных данных, используемых в SQL-запросах.

Принцип действия атаки путем внедрения кода SQL. Основная форма атаки SQL Injection состоит в прямой вставке кода в пользовательские входные переменные, которые объединяются с командами SQL и выполняются. Менее явная атака внедряет небезопасный код в строки, предназначенные для хранения в таблице или в виде метаданных. Когда впоследствии сохраненные строки объединяются с динамической командой SQL, происходит выполнение небезопасного кода.

Атака осуществляется посредством преждевременного завершения текстовой строки и присоединения к ней новой команды. Поскольку к вставленной команде перед выполнением могут быть добавлены дополнительные строки, злоумышленник заканчивает внедряемую строку меткой комментария «--». Весь последующий текст во время выполнения не учитывается.

Возможные SQL инъекции (SQL внедрения)

1) Наиболее простые — сворачивание условия WHERE к истинностному результату при любых значениях параметров.

2) Присоединение к запросу результатов другого запроса. Делается это через оператор UNION.

3) Закомментирование части запроса.

Существует **5 основных типов SQL инъекций**:

Классическая (In-Band или Union-based). Самая опасная и редко встречающаяся сегодня атака. Позволяет сразу получать любые данные из базы.

Error-based. Позволяет получать информацию о базе, таблицах и данных на основе выводимого текста ошибки СУБД.

Boolean-based. Вместо получения всех данных, атакующий может поштучно их перебирать, ориентируясь на простой ответ типа true/false.

Time-based. Похожа на предыдущую атаку принципом перебора, манипулируя временем отклика базы.

Out-of-Band. Очень редкие и специфические типы атак, основанные на индивидуальных особенностях баз данных.

Уязвимые точки для атаки находятся в местах, где формируется запрос к базе: форма аутентификации, поисковая строка, каталог, REST-запросы и непосредственно URL.

Для защиты от данного типа атак необходимо тщательно фильтровать входные параметры, значения которых будут использованы для построения SQL-запроса.

- Фильтрация строковых параметров (только буквы и числа)
- Фильтрация целочисленных параметров (только числа)

- Усечение входных параметров (количество символов)
- Функция по автоматическому определению зарезервированных SQL слов в тексте и занесению IP в бан лист
- Использование параметризованных запросов

Для каждого сервера и фреймворка есть свои тонкости и лучшие практики, но суть всегда одинакова: Нельзя вставлять данные в запрос напрямую. Всегда обрабатывайте ввод отдельно и формируйте запрос исключительно из безопасных значений. И естественно, не забывайте про ограничение прав доступа к базе.

Сканирование компьютерных сетей

Сетевая разведка или, по-другому, сканирование сети злоумышленником является этапом сбора необходимых сведений, предваряющим атаку.

Конкретный набор сведений зависит от типа атаки, но в общем случае «сетевого разведчика» интересуют следующие данные:

- IP-адреса активных хостов;
- номера активных и пассивных портов TCP и UDP;
- тип и версия ОС;
- тип и версия приложения.

Обнаружение IP-адресов активных хостов сети называют *сканированием сети* (network scanning), а активных и пассивных портов – *сканированием портов* (port scanning). Устройство или программа, выполняющие сканирование, называются *сканером*. Сам термин сканирование говорит о том, что злоумышленник перебирает все возможные IP-адреса некоторой подсети или номера портов.

На практике, внутренний нарушитель может собрать очень важную информацию, которая является недоступной для него в рамках служебных полномочий. Например, определить роли компьютеров в корпоративной сети, выделить файловые сервера и сервера баз данных, маршрутизаторы и интеллектуальные коммутаторы. И что особенно важно, именно результаты сканирования позволяют точно подобрать эксплойты для осуществления непосредственно несанкционированного доступа к узлам корпоративной сети. Следует отметить, что злоумышленник для этих целей, вероятнее всего, воспользуется одним из бесплатных сканеров. Однако применение в данном примере коммерческого сканера обусловлено тем, что это один из лучших инструментов для анализа защищенности сетей.

Злоумышленник, проанализировав службы, запущенные на хостах, может разделить их по функциональному признаку - доменные контроллеры, файловые, терминальные, принт-сервера, рабочие станции. По результатам сканирования можно выяснить, каким известным уязвимостям подвержены исследуемые хосты, и подобрать для последующей атаки соответствующие эксплойты.

Злоумышленник может получить следующую информацию:

- Карту сети
- Активные компоненты
- Открытые порты и сетевые сервисы
- Уязвимости компонентов

Реализация сканирования сети:

- Снифферы
- Сетевые сканеры
- Сканеры безопасности

Для сканирования сети и портов злоумышленники используют более изощренные методы, чем обычные пинги, т.к. обычной практикой в корпоративных сетях является запрет пинг-запросов. Например:

- процедура «*TCP SYN ping*», заключающаяся в попытке установления TCP-соединения с одним из публично доступных портов, например веб-сервиса (порт 80). Если в ответ на запрос установления соединения хост отвечает пакетом SYN/ACK, то можно с большой долей уверенности считать, что хост активен.

- процедура «*TCP ACK ping*», реализуемая путем отправки на сканируемый хост пакетов TCP со случайными значениями номеров портов. Если хост активен, то он отвечает пакетом TCP с признаком RST (сброс соединения), т.к. хост считает, что пакет пришел по ошибке – ведь соединение со сканирующим узлом установлено не было. Такой способ позволяет обойти файерволл, т.к. обычно межсетевые экраны блокируют установку соединений, но не контролируют установленные TCP-соединения, отличающиеся наличием признака ACK и отсутствием признака SYN. В таком случае, если TCP SYN ping не прошел, а TCP ACK ping был успешен, то сканер будет считать хост активным, но защищенным файерволлом.

- процедура «*UDP ping*» направляет UDP пакет с номером порта, который с большой вероятностью является пассивным. Например, это может быть порт, который не связан ни с одним из популярных приложений. В случае если компьютер активен, а порт пассивен, то сканер получает в ответ ICMP сообщение «порт недоступен», а если компьютер отключен, то «хост недоступен». Если же и компьютер и порт оказываются активными, то никакого ответа сканер не получает и определенного вывода о состоянии хоста он сделать не может. В этом причина выбора пассивного порта. Эффективность применения UDP объясняется тем, что многие файерволлы по умолчанию не блокируют трафик UDP.

- процедура «*ICMP timestamp ping*». Хост тестируется запросами синхронизации времени протокола ICMP (код 13). Т.к. администраторы сетей чаще блокируют только эхо-запросы ICMP, то этот тип запросов доходит до хоста, который обязательно инициирует ответ. Вместо запросов синхронизации времени можно также использовать запросы длины маски IP-адреса (код 17) или информационные запросы (код 15).

- процедура «*IP ping*» направляет на компьютер IP-пакет с кодом вложенного протокола, отличным от кодов протокола TCP, UDP или ICMP. Скорее всего такой тип протокола не поддерживается стеком TCP/IP данного компьютера, и активный хост ответит ICMP сообщением «протокол недостижим».

Очень похожие методы применяются и для **сканирования портов**. Здесь предпочтение отдается TCP SYN-сканированию, т.к. это самый быстрый способ. Производительность в данном случае очень важна, т.к. в отличие от сканирования хостов проверить нужно 65535 TCP-портов и столько же UDP-портов.

Сканирование портов злоумышленниками осуществляется с помощью специальных программных средств, многие из которых используются довольно легально – для инвентаризации сети и аудита ее защищенности. Среди наиболее популярных продуктов стоит отметить сканер с открытым кодом **NMAP**, позволяющую выполнить сканирование сети и портов на основе довольно большого количества различных процедур.

Сканирование сети и портов обычно не проходит незамеченным – очень вероятно, что средства протоколирования событий ОС и файрволлов зафиксируют этот процесс и администратор сканируемой сети начнёт расследовать инцидент, основываясь на адресе компьютера, который выполнял сканирование.

Современные МЭ имеют модули (plug-in) позволяющие обнаружить атаки и сканирование в режиме реального времени, также как это сделано в СОА и СОВ. Некоторые сканеры уязвимостей используют оригинальные методы, позволяющие производить сканирование максимально скрытно. Например, в сканере Nmap существуют возможности, позволяющие значительно затруднить обнаружение сканирования для СОА:

- возможность задавать временные параметры сканирования — интервалы между пакетами. Для выявления такого сканирования необходимо проанализировать пакеты за значительный промежуток времени;
- возможность задавать группу ложных хостов, с которых якобы производится сканирование, для скрытия реального IP-адреса злоумышленника. Данная функция особенно опасна, т.к. в качестве ложных хостов могут быть указаны хосты легальных сотрудников, что значительно затруднит обнаружение настоящего нарушителя.

Решением против подобных методов сканирования может быть использование сетевых СОА, либо периодическое изучение журналов регистрации МЭ.

Технологии защиты информации в сетях. Сеть как объект защиты.

Большинство современных автоматизированных систем обработки информации представляют собой распределенные системы, построенные на стандартных сетевых архитектурах и использующие типовые наборы сетевых сервисов и прикладного программного обеспечения. Корпоративные сети "наследуют" все "традиционные" для локальных вычислительных систем

способы несанкционированного вмешательства. Кроме того, для них характерны и специфические каналы проникновения и несанкционированного доступа к информации, обусловленные использованием сетевых технологий.

Перечислим основные особенности распределенных вычислительных систем:

- территориальная удаленность компонентов системы и наличие интенсивного обмена информацией между ними;
- широкий спектр используемых способов представления, хранения и передачи информации;
- интеграция данных различного назначения, принадлежащих различным субъектам, в рамках единых баз данных и, наоборот, размещение необходимых некоторым субъектам данных в различных удаленных узлах сети;
- абстрагирование владельцев данных от физических структур и места размещения данных;
- использование режимов распределенной обработки данных;
- участие в процессе автоматизированной обработки информации большого количества пользователей и персонала различных категорий;
- непосредственный и одновременный доступ к ресурсам большого числа пользователей;
- разнородность используемых средств вычислительной техники и программного обеспечения;

Способы безопасности информационных сетей. Системный подход к управлению безопасностью.

Обычно первое, что ассоциируется с ИБ, - это антивирусные программы, файерволлы, системы шифрования и другие технические средства защиты. Бесспорно, роль этих средств в обеспечении безопасности велика, однако не меньшее, а иногда и большее влияние на безопасность системы оказывают средства, построенные на качественно другой основе.

Существует два подхода к проблеме обеспечения безопасности компьютерных систем и сетей (КС): "фрагментарный" и комплексный.

Фрагментарный подход направлен на противодействие четко определенным угрозам в заданных условиях. В качестве примеров реализации такого подхода можно указать отдельные средства управления доступом, автономные средства шифрования, специализированные антивирусные программы и т.п.

Достоинством такого подхода является высокая избирательность к конкретной угрозе. Существенным недостатком данного подхода является отсутствие единой защищенной среды обработки информации. Фрагментарные меры защиты информации обеспечивают защиту конкретных объектов КС только от конкретной угрозы. Даже небольшое видоизменение угрозы ведет к потере эффективности защиты.

Комплексный подход ориентирован на создание защищенной среды обработки информации в КС, объединяющей в единый комплекс разнородные

меры противодействия угрозам. Организация защищенной среды обработки информации позволяет гарантировать определенный уровень безопасности КС, что является несомненным достоинством комплексного подхода. К недостаткам этого подхода относятся: ограничения на свободу действий пользователей КС, чувствительность к ошибкам установки и настройки средств защиты, сложность управления.

Комплексный подход применяют для защиты КС крупных организаций или небольших КС, выполняющих ответственные задачи либо обрабатывающих особо важную и секретную информацию ограниченного доступа. Нарушение безопасности информации в КС крупных организаций может нанести огромный материальный ущерб как самим организациям, так и их клиентам. Поэтому такие организации вынуждены уделять особое внимание гарантиям безопасности и реализовывать комплексную защиту. Комплексного подхода придерживается большинство государственных и крупных коммерческих предприятий и учреждений. Этот подход нашел свое отражение в различных стандартах и политиках безопасности. Последняя, как раз регламентирует эффективную работу средств защиты КС, охватывая все особенности процесса обработки информации и определяя поведение системы в различных ситуациях.

Для защиты интересов субъектов информационных отношений необходимо сочетать меры следующих уровней:

- законодательного (стандарты, законы, нормативные акты и т.п.);
- административно-организационного (действия общего характера, предпринимаемые руководством организации, и конкретные меры безопасности, имеющие отношение к людям);
- программно-технического (конкретные технические меры).

Меры законодательного уровня очень важны для обеспечения информационной безопасности.

Большинство людей не совершают противоправных действий не потому, что это технически невозможно, а потому, что это осуждается или наказывается обществом, что так поступать не принято. В рамках обеспечения информационной безопасности следует рассмотреть на законодательном уровне *две группы мер*:

- **направляющие и координирующие меры, способствующие повышению образованности общества** в области информационной безопасности, помогающие в разработке и распространении средств обеспечения информационной безопасности. В данном случае важно понять, что информационная безопасность это новая область деятельности и здесь важно не только запрещать и наказывать, но и научить, разъяснить и помочь. Общество должно осознать важность данной проблематики, понять основные пути решения соответствующих проблем. В свою очередь, государство может сделать это оптимальным образом. Здесь не нужно больших материальных затрат, требуются интеллектуальные вложения;

- меры, направленные на создание и поддержание в обществе негативного (в том числе карательного) отношения к нарушениям и нарушителям информационной безопасности.

Ко второй группе следует отнести основные законодательные акты по информационной безопасности, являющиеся частью правовой системы Российской Федерации.

В Конституции РФ содержится ряд правовых норм, определяющих основные права и свободы граждан России в области информатизации, в том числе ст. 23 определяет право на неприкосновенность частной жизни, личную и семейную тайну, тайну переписки, телефонных переговоров, почтовых, телеграфных и иных сообщений; ст. 42 обеспечивает право на получение достоверной информации о состоянии окружающей среды и др.

В Уголовном кодексе РФ имеются нормы, затрагивающие вопросы информационной безопасности граждан, организаций и государства. В числе таких статей ст. 137 «Нарушение неприкосновенности частной жизни», ст. 138 «Нарушение тайны переписки, телефонных переговоров, почтовых и телеграфных или иных сообщений», ст. 140 «Отказ в предоставлении гражданину информации», ст. 155 «Разглашение тайны усыновления (удочерения)», ст. 183 «Незаконное получение и разглашение сведений, составляющих коммерческую или банковскую тайну», ст. 272 «Неправомерный доступ к компьютерной информации», ст. 273 «Создание, использование или распространение вредоносных программ для ЭВМ», ст. 274 «Нарушение правил эксплуатации ЭВМ, системы ЭВМ или их сети» и др.

В Налоговом кодексе РФ имеется ст. 102 «налоговая тайна».

В Гражданском кодексе РФ вопросам обеспечения информационной безопасности посвящены ст. 139 «Служебная и коммерческая тайна», ст. 946 «Тайна страхования» и др.

Рост объемов киберпреступности привлекает внимание государства, и соответственно, является объектом законодательных инициатив. Однако теперь уже ясно, что преступность в сфере информационных технологий — явление глобальное. Следовательно, для эффективной борьбы с киберпреступностью необходимо сотрудничество на международном уровне, чтобы обеспечить преследование киберпреступников, невзирая на геополитические границы.

Европейская конвенция о киберпреступности

Одно из наиболее серьезных ограничений национального законодательства о компьютерных преступлениях состоит в том, что оно не позволяет эффективно бороться с глобальным явлением киберпреступности. Европейская конвенция о киберпреступности, разработанная с целью создания международной структуры для борьбы с киберпреступлениями, была принята Комитетом министров Совета Европы в ноябре 2001 года, а вступила в силу 1 июля 2004 года.

Конвенция охватывает широкий круг вопросов, в том числе все аспекты киберпреступности, включая незаконный доступ к компьютерным системам и перехват данных, воздействие на данные, воздействие на работу системы,

противозаконное использование устройств, подлог и мошенничество с использованием компьютерных технологий, правонарушения, связанные с детской порнографией, и правонарушения, связанные с авторским правом и смежными правами. При подготовке конвенции также преследовались цели формирования общей правоохранительной системы для борьбы с киберпреступностью и создания условий для обмена информацией между всеми странами, подписавшими конвенцию.

Меры административно-организационного уровня. Администрация организации должна сознавать необходимость поддержания режима информационной безопасности и выделения на эти цели соответствующих ресурсов. Основной мерой защиты административно-организационного уровня является политика безопасности и комплекс организационных мер. Под политикой безопасности понимается совокупность документированных управленческих решений, направленных на защиту информации и связанных с ней ресурсов организации. Политика безопасности определяет стратегические направления информационной защиты предприятия, очерчивает круг критически важных информационных ресурсов предприятия, защита которых представляет наивысший приоритет, предлагает меры, которые могут быть предприняты для устранения или уменьшения связанных с этими ресурсами рисков.

К комплексу организационных мер относятся меры безопасности, реализуемые людьми. Можно выделить следующие группы организационных мер:

- управление персоналом;
- физическая защита;
- поддержание работоспособности;
- реагирование на нарушения режима безопасности;
- планирование восстановительных работ.

Для каждой группы в каждой организации должен существовать набор регламентов, определяющих действия персонала.

Для поддержания режима информационной безопасности особенно важны меры программно-технического уровня, поскольку основная угроза компьютерным системам исходит от них самих: сбои оборудования, ошибки ПО, промахи пользователей и администраторов и т.п.

Меры и средства программно-технического уровня. В рамках современных информационных систем должны быть доступны по крайней мере следующие механизмы безопасности:

- идентификация и проверка подлинности пользователей;
- управление доступом;
- протоколирование и аудит;
- криптография;
- экранирование;
- обеспечение высокой доступности.

Именно этим видам средств защиты в основном посвящены лекции по данной дисциплине.

Технические средства принято разделять на программные, аппаратные и программно-аппаратные.

Программные средства включают защитные инструменты ОС (подсистемы аутентификации и авторизации пользователей, средства управления доступом, аудит и др.) и прикладные программы, предназначенные для решения задач безопасности (системы обнаружения и предотвращения вторжений, антивирусные средства, прокси-серверы).

Примером аппаратных средств, специализирующихся на информационной защите, являются ИБП, генераторы напряжения, видеонаблюдение, СКУД в помещения и др.

К аппаратно-программным средствам относятся, например, некоторые анализаторы сетевого трафика и МЭ.

К техническому уровню также относят и математические методы (методы криптографии), алгоритмы (эвристический алгоритм расчета времени оборота в протоколе TCP) и абстрактные модели (модели контроля доступа), которые будут также рассмотрены в последующих лекциях.

Необходимость применения стандартов. Информационные системы компаний почти всегда построены на основе программных и аппаратных продуктов различных производителей. Дело в том, что на данный момент нет ни одной компании-разработчика, которая предоставила бы потребителю полный перечень средств (от аппаратных до программных) для построения современной ИС. Чтобы обеспечить в разнородной ИС надежную защиту информации, требуются специалисты высокой квалификации, которые будут отвечать за безопасность каждого компонента ИС: правильно их настраивать, постоянно отслеживать происходящие изменения, контролировать работу пользователей. Очевидно, что чем разнороднее информационная система, тем сложнее обеспечить ее безопасность. Изобилие в корпоративных сетях и системах устройств защиты, МЭ, шлюзов и VPN, а также растущий спрос на доступ к корпоративным данным со стороны сотрудников, партнеров и заказчиков приводят к созданию сложной среды защиты, трудной для управления, а иногда и несовместимой.

Поэтому вполне очевидна потребность в применении единого набора стандартов поставщиками средств защиты, компаниями-системными интеграторами и организациями, выступающими в качестве заказчиков систем безопасности для своих корпоративных сетей и систем. Стандарты образуют понятийный базис, на котором строятся все работы по обеспечению информационной безопасности, и определяют критерии, которым должно следовать управление безопасностью.

Видеонаблюдение и надежный замок в офисе, продуманная процедура приема сотрудников на работу, закон, угрожающий хакеру уголовным преследованием, стандарт, помогающий провести анализ возможного ущерба из-за действия нарушителя, - все эти очень не похожие средства одинаково

важны для обеспечения безопасности. Но успех в области ИБ может принести только системный подход, при котором средства защиты разных типов применяются совместно и под централизованным управлением.

Таким образом, комплексный подход к решению проблемы обеспечения безопасности, рациональное сочетание законодательных, административно-организационных и программно-технических мер и обязательное следование промышленным, национальным и международным стандартам являются тем фундаментом, на котором строится вся система защиты корпоративных сетей.

Пути решения проблем защиты информации в сетях.

Для поиска решений проблем ИБ при работе в сети Интернет был создан независимый консорциум ISTF (Internet Security Task Force) – общественная организация, состоящая из представителей и экспертов компаний-поставщиков средств ИБ, электронных бизнесов и провайдеров интернет-инфраструктуры. Цель этого консорциума – разработка технических, организационных и операционных руководств по безопасности работы в Интернете.

Консорциум ISFT выделил 12 областей ИБ, на которых в первую очередь должны сконцентрировать свое внимание создатели электронного бизнеса, чтобы обеспечить его работоспособность:

- аутентификация (механизм объективного подтверждения идентифицирующей информации);
- право на частную, персональную информацию (обеспечение конфиденциальности информации);
- определение событий безопасности (Security Events);
- защита корпоративного периметра;
- определение атак;
- контроль за потенциально опасным содержанием (Malicious content);
- контроль доступа;
- администрирование;
- реакция на события (Incident Response).

Эти рекомендации помогают определить потенциальные бреши и дыры в их КС, которые, если не обратить на них должного внимания, могут использоваться взломщиками-хакерами. Реализация рекомендаций консорциума означает, что защита информации в системе электронного бизнеса должна быть комплексной.

По мнению членов консорциума, для комплексной защиты от угроз и гарантии экономически выгодного и безопасного использования коммуникационных ресурсов необходимо решить следующие задачи:

- проанализировать угрозы безопасности для ИС;
- разработать политику ИБ;
- защитить внешние каналы передачи информации, обеспечив конфиденциальность, целостность и подлинность передаваемой по ним информации;

- гарантировать возможность безопасного доступа к открытым ресурсам внешних сетей и Интернета, а также общения с пользователями этих сетей;
- защитить отдельные наиболее коммерчески значимые информационные системы независимо от используемых ими каналов передачи данных;
- предоставить защищенный удаленный доступ персонала к информационным ресурсам корпоративной сети;
- обеспечить надежное централизованное управление средствами сетевой защиты.

Согласно рекомендациям ISTF и классификации «рубежной обороны» Hurwitz Group первым и важнейшим этапом разработки системы ИБ являются механизмы управления доступом к сетям общего пользования и доступом из них, а также механизмы безопасных коммуникаций, реализуемые МЭ и продуктами частных защищенных виртуальных сетей (VPN). Сопровождая их средствами интеграции и управления всей ключевой информацией системы защиты (PKI – инфраструктура открытых ключей), можно получить целостную, централизованно управляемую систему информационной безопасности.

Следующий рубеж включает в себя интегрируемые в общую структуру средства контроля доступа пользователей в систему вместе с системой однократного входа и авторизации (Single Sign-On).

Антивирусная защита, средства, аудита и обнаружения атак, по существу, завершают создание интегрированной целостной системы безопасности, если речь не идет о работе с конфиденциальными данными. В этом случае потребуются также средства криптографической защиты данных и ЭЦП.

Для реализации основных функциональных компонентов системы безопасности применяются различные методы и средстваЗИ:

- защищенные коммуникационные протоколы;
- средства криптографии;
- механизмы аутентификации и авторизации;
- средства контроля доступа к рабочим местам сети и из сетей общего пользования;
- антивирусные комплексы;
- программы обнаружения атак и аудита;
- средства централизованного управления контролем доступа пользователей, а также безопасного обмена пакетами данных и сообщениями любых приложений по открытым IP-сетям.

Применение комплекса СЗИ на всех уровнях ИС позволяет построить эффективную и надежную систему обеспечений информационной безопасности.

Перечисленные выше методы и СЗИ будут подробно рассмотрены на следующих лекциях.

Модель OSI. Механизмы защиты информации, применяемые на разных уровнях.

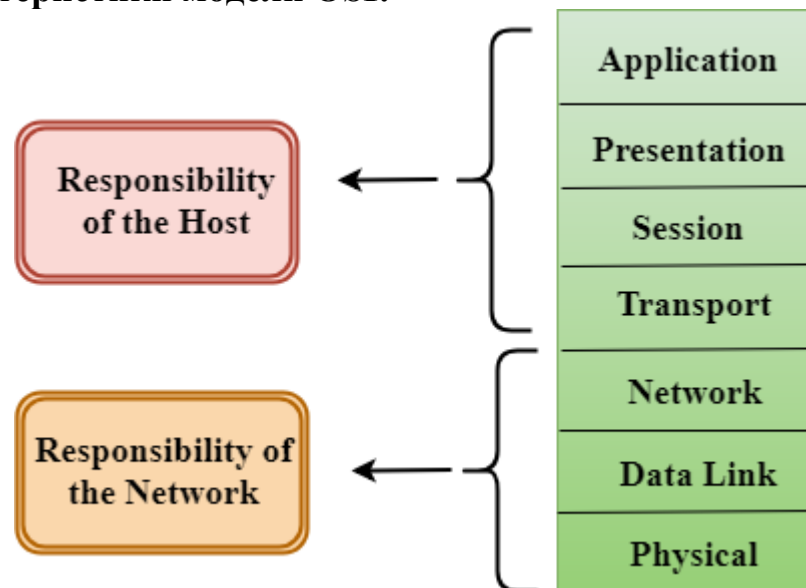
Изначально сеть Интернет рассматривалась как безопасная среда передачи данных между военными. Безопасность организовывалась на уровне физической изоляции объектов от посторонних лиц, и это было оправдано, когда к сети имело доступ ограниченное число машин. Однако, когда Интернет стал открытой информационной средой, не только в плане «свободы информации», но и с точки зрения несанкционированного доступа к этой информации, потребность в обеспечении безопасности передачи данных появилась.



Средством обеспечения информационной поддержки предприятия в подавляющем большинстве случаев является его компьютерная сеть. Особенности архитектуры компьютерных сетей описаны семиуровневой моделью взаимодействия открытых систем (Open Systems Interconnection, OSI) или просто «модель OSI». В соответствии с концептуальными положениями этой модели процесс информационного обмена в компьютерных сетях можно разделить на семь этапов в зависимости от того, каким образом, и между какими объектами происходит информационный обмен. Эти этапы называются уровнями модели взаимодействия открытых систем. Термин «открытая система» означает, то, что при построении этой системы были использованы доступные и открыто опубликованные стандарты и спецификации. Каждому уровню модели соответствует определенная группа стандартов и спецификаций.

Другими словами, модель OSI делит всю задачу на семь небольших и управляемых задач. Каждому слою назначается определенная задача. Каждый уровень является автономным, поэтому задача, назначенная каждому уровню, может выполняться независимо.

Характеристики модели OSI:



Модель OSI разделена на две базовых части: верхние и нижние слои.

- верхний слой модели OSI в основном связан с проблемами приложений, и они реализованы только в программном обеспечении. Уровень приложений наиболее близок к конечному пользователю. И конечный пользователь, и прикладной уровень взаимодействуют с программными приложениями. Верхний слой относится к слою чуть выше другого слоя.

- нижний слой модели OSI занимается проблемами передачи данных и ориентирован на выполнение коммуникационных функций в реальном масштабе времени. Канальный уровень и физический уровень реализованы в аппаратном и программном обеспечении. Физический уровень является самым низким уровнем модели OSI и наиболее близок к физической среде. Физический уровень в основном отвечает за размещение информации на физическом носителе.

Защита на различных уровнях модели OSI.

Что такое защита информации на каком-либо уровне модели OSI? Каждый последующий уровень сетевых пакетов инкапсулирован в предыдущем. То есть данные протокола прикладного уровня (например, HTTP) находится внутри пакета транспортного уровня (например, TCP), который находится внутри пакета сетевого уровня (например, IP), который находится внутри кадра канального уровня (например, кадра Ethernet).

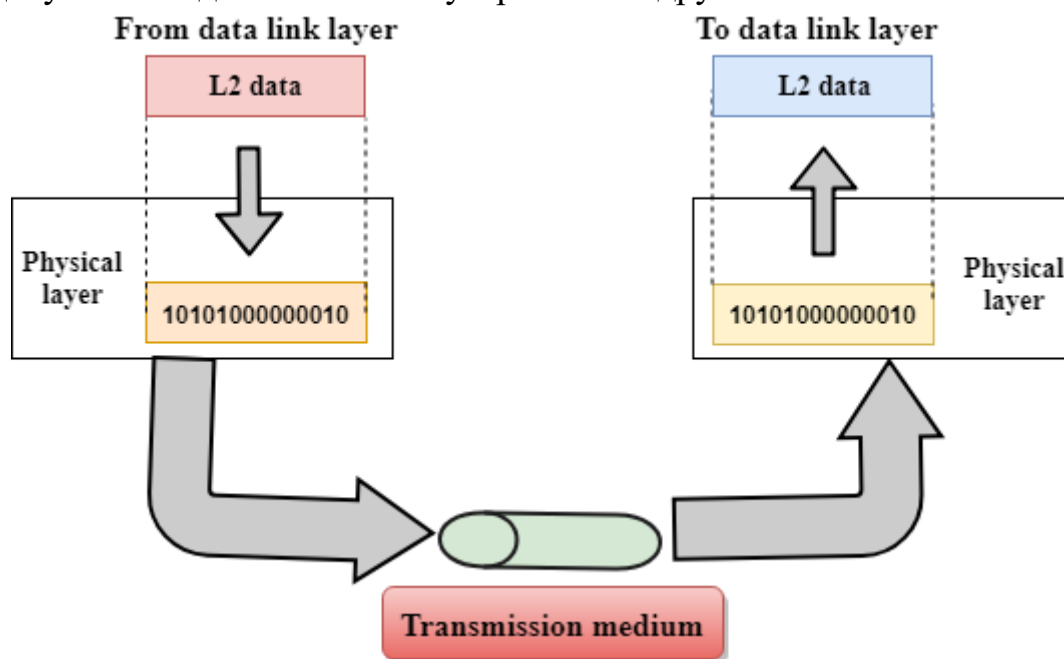
При защите информации на сетевом уровне шифруется содержимое пакета IP, то есть пакет TCP. В другом варианте защиты на сетевом уровне шифруется целый пакет IP и данный зашифрованный пакет в свою очередь инкапсулируется. Такая дополнительная инкапсуляция позволяет скрыть топологию сетей участников обмена.

Следует отметить, что защита, например, на канальном уровне обеспечивает абсолютно "прозрачное" использование сетевого уровня.

Физический уровень.

Основная функциональность физического уровня заключается в передаче отдельных битов от одного узла к другому узлу. Это самый низкий уровень модели OSI.

Протоколы физического уровня описывают электрические, механические, функциональные и процедурные средства для активации, поддержки и деактивации физического соединения, обеспечивающего передачу бит из одного сетевого устройства в другое.



Физический уровень получает пакеты данных от вышележащего канального уровня и преобразует их в оптические или электрические сигналы, соответствующие 0 и 1 бинарного потока. Эти сигналы посылаются через среду передачи на приемный узел.

Механические и электрические/оптические свойства среды передачи определяются на физическом уровне и включают в себя:

- тип кабелей и разъемов;
- разводку контактов в разъемах;
- схему кодирования сигналов для значений 0 и 1.

На этом же уровне определяются характеристики электрических сигналов, такие как:

- фронты импульсов;
- уровни напряжения или тока передаваемого сигнала;
- типы кодирования;
- скорости передачи сигналов.

Критерии безопасности уровня:

1. Физическая работоспособность.
2. Физическая целостность (целостность системы защиты).
3. Физическая доступность.

Набор показателей:

1. Напряжение.

2. Давление.
3. Скорость передачи.

Сигнатуры:

1. Давление в манометре (например, падение давления в трубе ниже определенного уровня является сигнатурой).
2. Использование каналов, не задействованных в защищаемой сети.
3. Перекрывающиеся каналы.
4. Внезапное изменение рабочего канала одним или несколькими устройствами, за которыми ведется наблюдение.
5. А вот события, фиксируемые СОВ на верхних уровнях стека протоколов, например, большое число фрагментированных пакетов или запросов TCP SYN, может указывать на сканирование портов или DoS-атаку. Но, если это просто результат плохой связи на физическом уровне, данная проблема не является сигнатурой.

Атаки, возможные на физическом уровне:

1. Обрыв кабеля.
2. Недопустимое повышение / понижение напряжения.
3. Среда теряет функциональность (утратила физические свойства).
4. Используемые характеристики находятся в неправильном диапазоне.
5. Изменились свойства канала, то есть передача, как таковая, совершаться будет, но сам канал передачи будет ненадежный (не защищен).

Обеспечить безопасность информационного обмена на физическом уровне модели можно за счет структуризации физических связей между узлами компьютерной сети. Защищенная физическая среда передачи данных является первым рубежом для злоумышленника или преградой для воздействия разрушительных факторов окружения.

Рассмотрим общие механизмы защиты физического уровня на примере *среды передачи, образованная медной витой парой*. Топология физических связей «звезда». Количество кабельных сегментов в данной сети соответствует количеству узлов. Нарушение целостности среды одного кабельного сегмента не влияет на работоспособность всей сети. Наиболее уязвимым элементом сети является центральное коммуникационное устройство (концентратор или коммутатор). Фактически устройства этого класса являются средством разделения среды передачи.

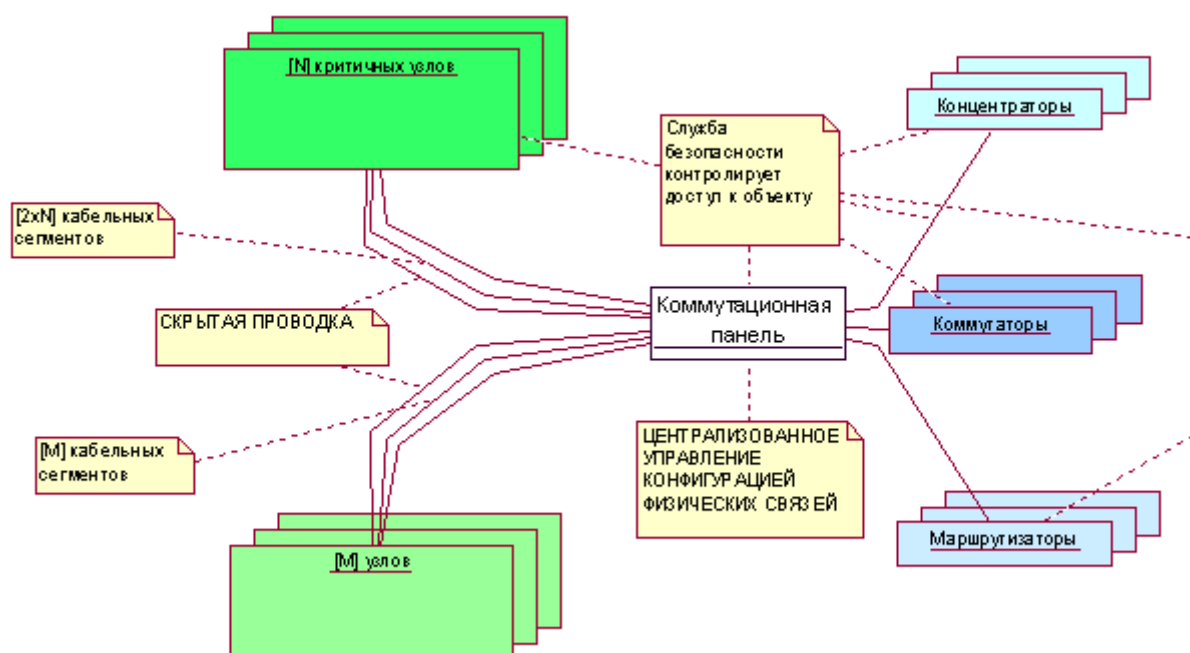
Среда передачи, образованная концентратором, позволяет злоумышленнику реализовать прослушивание трафика и атаку отказа в доступе, основанную на широковещательной рассылке сообщений. При этом злоумышленник может не иметь непосредственного физического доступа к самому концентратору.

Коммутаторы используются для осуществления попеременного доступа узлов к среде передачи. Разделение физической среды передачи между узлами во времени затрудняет прослушивание сети злоумышленником и создает дополнительную преграду для осуществления атак отказа в доступе, основанных на широковещательной рассылке сообщений в сети.

Использование тех и других устройств как средств образования среды передачи позволяет злоумышленнику вызвать отказ всей сети при наличии у него физического доступа к ним либо к системе их энергоснабжения.

Кроме того, для всех разновидностей медных кабельных систем, используемых в качестве среды передачи данных, имеет место наличие побочного электромагнитного излучения и наводок (ПЭМИН). Несмотря на свою вторичность, ПЭМИН является информативным для злоумышленника и позволяет ему анализировать пики сетевой активности, а при наличии анализатора спектра электромагнитного излучения, осуществить перехват передаваемых средой передачи сообщений.

Ниже приведены основные рекомендации, позволяющие снизить вероятность атак на физическом уровне компьютерной сети предприятия злоумышленником.



1. Рекомендуемая конфигурация физических связей в компьютерной сети предприятия — «звезда», при этом для подключения каждого узла выделен отдельный кабельный сегмент. В качестве среды передачи используется восьмижильный медный кабель типа «витая пара» либо оптоволокно.

2. Для подключения критически важных для предприятия серверов используют два кабельных сегмента — основной и резервный.

3. Прокладка сетевого кабеля осуществляется в скрытой проводке, либо в закрываемых кабель-каналах с возможностью опечатывания не срываемыми наклейками — «стикерами».

4. Кабельные сегменты, используемые для подключения всех узлов компьютерной сети, должны быть сконцентрированы на одной коммутационной панели.

5. В начальной конфигурации топологии физических связей должно быть исключено совместное использование среды передачи любой парой узлов сети. Исключение составляет связь с «узел-коммутатор».

6. Управление конфигурацией физических связей между узлами осуществляется только на коммутационной панели.

7. Коммутационная панель смонтирована в запираемом коммутационном шкафу. Доступ в помещение коммутационного шкафа строго ограничен и контролируется службой безопасности предприятия.

Особенности защиты физического уровня беспроводных сетей.

Особый класс сред передачи составляет беспроводная среда передачи или радиочастотный ресурс. При построении компьютерных сетей предприятий в настоящее время широко применяется технология WiFi. Топология физических связей сетей, построенных по этому принципу, — либо «точка-точка», либо «звезда». Особенность организации беспроводных сетей передачи данных, построенных с использованием технологии WiFi, предполагает наличие у злоумышленника полного доступа к среде передачи. Злоумышленник, обладающий WiFi-адаптером в состоянии без труда организовать прослушивание радиосети передачи данных. Парализовать работу такой сети можно при условии наличия у злоумышленника излучателя, работающего 2,4ГГц-овом в диапазоне частот и обладающего более высокими по сравнению с излучателями атакуемой радиосети мощностными характеристиками.

Рекомендации по защите радиосетей передачи данных.

1. Служба безопасности должна обеспечить строгое ограничение физического доступа персонала предприятия и полное исключение доступа посторонних на площадки монтажа приемного и излучающего оборудования радиосетей передачи данных. Доступ на площадки должен контролироваться службой безопасности предприятия.

2. Прокладка высокочастотного (антенного) кабеля должна быть выполнена скрытым способом либо в коробах с последующим опечатыванием коробов «стикерами».

3. Длина высокочастотного (антенного) кабельного сегмента должна быть минимальной.

4. Доступ в помещения с радиомодемами, радиомостами и станциями, оснащенными радиосетевыми адаптерами должен строго контролироваться службой безопасности предприятия.

5. Администратор сети должен детально документировать процедуры настройки беспроводных устройств, в т.ч. радиомодемов, мостов и станций.

6. Администратор сети должен регулярно менять реквизиты авторизации для удаленного управления этими устройствами.

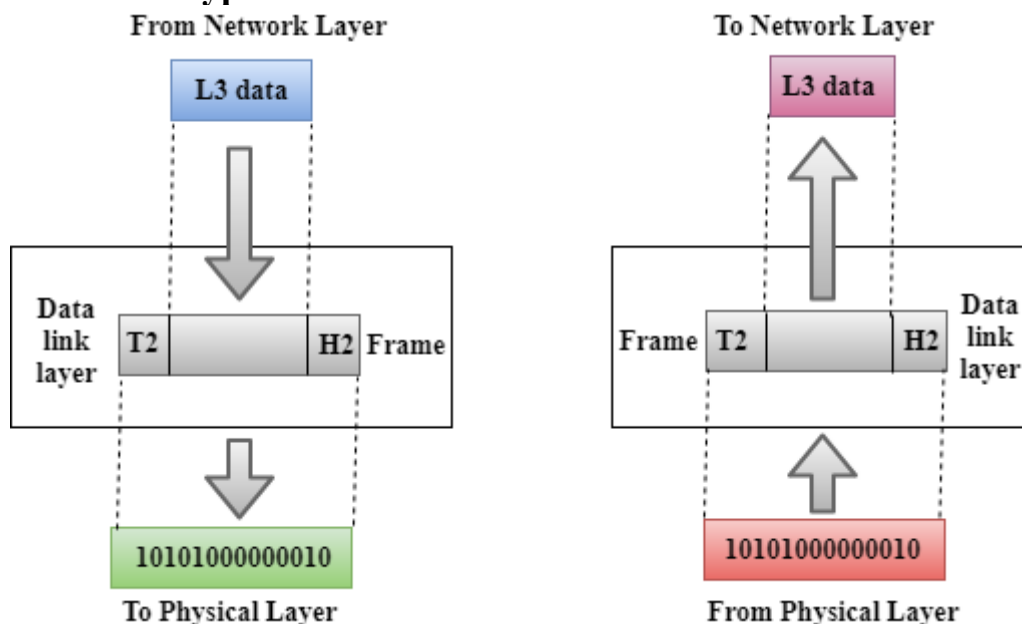
7. Администратор сети должен выделить отдельный адресный пул для администрирования этих устройств по сети.

8. Администратор сети должен отключить неиспользуемые функции радиомодемов, мостов и станций.

9. Администратор должен активировать функции радиомодема или радиомоста обеспечивающие «тунелирование» и криптографическую защиту передаваемых и принимаемых сообщений.

10. Администратор должен контролировать доступ к радиомодемам, радиомостам и станциям со стороны узлов компьютерной сети предприятия. Один из возможных способов контроля — использование межсетевого экрана.

Канальный уровень.



Этот слой отвечает за безошибочную передачу кадров данных. Канальный уровень подготавливает фреймы для передачи по локальной среде: он определяет формат данных в сети, обеспечивает надежную и эффективную связь между двумя или более устройствами, отвечает за уникальную идентификацию каждого устройства, которое находится в локальной сети.

Уровень содержит два подслоя:

1) Уровень управления логической связью (LLC):

Отвечает за передачу пакетов на сетевой уровень принимающего получателя.

Так же идентифицирует адрес протокола сетевого уровня из заголовка.

Это также обеспечивает управление потоком.

2) Уровень контроля доступа к среде (MAC):

Уровень управления доступом к среде является связующим звеном между уровнем управления логическим каналом и физическим уровнем сети.

Он используется для передачи пакетов по сети.

Функции канального уровня. Протоколы и стандарты этого уровня описывают процедуры проверки доступности среды передачи и корректности передачи данных.

- **кадрирование:** канальный уровень преобразует необработанный битовый поток физического объекта в пакеты, известные как кадры. Уровень передачи данных добавляет заголовок и трейлер к фрейму. Заголовок, который добавляется к фрейму, содержит аппаратный адрес назначения и адрес источника.

- **физическая адресация:** канальный уровень добавляет заголовок к фрейму, который содержит адрес назначения. Кадр передается по адресу назначения, указанному в заголовке.

- **управление потоком:** управление потоком является основной функциональностью уровня канала передачи данных. Это метод, с помощью которого поддерживается постоянная скорость передачи данных с обеих сторон, чтобы не повредить данные. Это гарантирует, что передающая станция, такая как сервер с более высокой скоростью обработки, не превышает принимающую станцию, с более низкой скоростью обработки.

- **контроль ошибок:** Контроль ошибок достигается путем добавления вычисленного значения CRC (англ. Cyclic redundancy check), которое помещается в трейлер уровня звена данных, который добавляется в кадр сообщения перед его отправкой на физический уровень. Принцип работы CRC таков: два устройства работают по стандарту - одно передающее другое принимающее, оба формируют поле для внесения контрольной суммы - отправитель пишет сумму получатель проверяет сумму, если сумма не совпадает - кадр отбрасывается, после чего получатель отправляет подтверждение для повторной передачи поврежденных кадров.

- **контроль доступа:** когда два или более устройств подключены к одному и тому же каналу связи, протоколы уровня канала передачи данных используются для определения того, какое устройство имеет контроль над каналом в данный момент времени. Осуществление контроля доступности среды необходимо т.к. спецификации физического уровня не учитывают то, что в некоторых сетях линии связи могут разделяться между несколькими взаимодействующими узлами и физическая среда передачи может быть занята.

Критерии безопасности уровня:

1. Корректность реализации соглашений протокола.
2. Статистическая устойчивость интенсивности широковещательных фреймов.
3. Статистическая устойчивость интенсивности фреймов с чужим MAC-адресом. MAC — адреса и идентификаторы виртуальных сегментов сети должны быть определены и перечислены.

Набор показателей:

1. Принадлежность текущего MAC-адреса или идентификатора таким спискам.
2. Частота поступлений широковещательных фреймов (например, должно быть 3 с / на определенный узел, а пришло 1000 фреймов).
3. Количество фреймов с чужим MAC-адресом.
4. Большое количество широковещательных фреймов.

Сигнатуры:

1. Несовпадение текущего MAC-адреса или идентификатора ни с одним элементом из элементов списка.
2. Превышение частоты некоторого заранее установленного порога с / на определенный узел.

3. Превышение количества пришедших на интерфейс коммутатора фреймов с чужим MAC-адресом.

4. Количество широковещательных фреймов превысило допустимые показатели.

5. Фреймы с другим VLAN ID (идентификатор VLAN, указывающий, какому VLAN'у принадлежит фрейм. Диапазон возможных значений VID от 0 до 4095).

Атаки, возможные на канальном уровне:

1. Перехват фреймов.

2. На интерфейс коммутатора пришел фрейм с чужим MAC-адресом.

3. Компрометация на ARP (1 хост подменяется другим).

4. Broadcast storm (считается, что приемлемая доля широковещательного трафика должна составлять 10 % от трафика всей сети. Значение в 20 % и выше должно классифицироваться как атака).

5. Попытка подмена VLAN.

Алгоритм определения доступности среды для всех технологий одинаков и основан на постоянном прослушивании среды передачи всеми подключенными к ней узлами. Эта особенность используется злоумышленниками для организации различных видов атак на компьютерные сети. Даже при условии соблюдения рекомендаций относительно исключения разделения среды передачи злоумышленник может осуществить прослушивание трафика между произвольно выбранной парой узлов компьютерной сети. Причем использование простых коммутаторов не является серьезной преградой для злоумышленника. Утверждение о полной защищенности сетей, построенных на основе топологии физических связей «звезда» и оснащенных простыми коммутаторами, является серьезным заблуждением.

Мы помним по прошлым лекциям про уязвимость системы разрешения сетевых адресов (ARP — Address Resolution Protocol), состоящей в том, что узлы доверяют содержимому кадра с ответом на запрос о разрешении сетевого адреса, который никак не проверяется и ничем не подтверждается. Этой уязвимостью и пользуются злоумышленники, реализуя различные техники подмены аппаратных адресов (ARP spoofing).

Защита данных в процессе передачи по открытым каналам основана на построении виртуальных защищенных каналов связи - **криптозащищенные туннели**. К протоколам построения защищённого канала передачи данных на канальном уровне относятся:

- протокол PPTP (Point-to-Point Tunneling Protocol), разработанный Microsoft совместно с компаниями Ascend Communications, 3Com/Primary Access, Ecl-Telematics и US Robotics;

- протокол L2TP (Layer-2 Tunneling Protocol), объединивший протокол L2F (Layer-2 Forwarding) и PPTP.

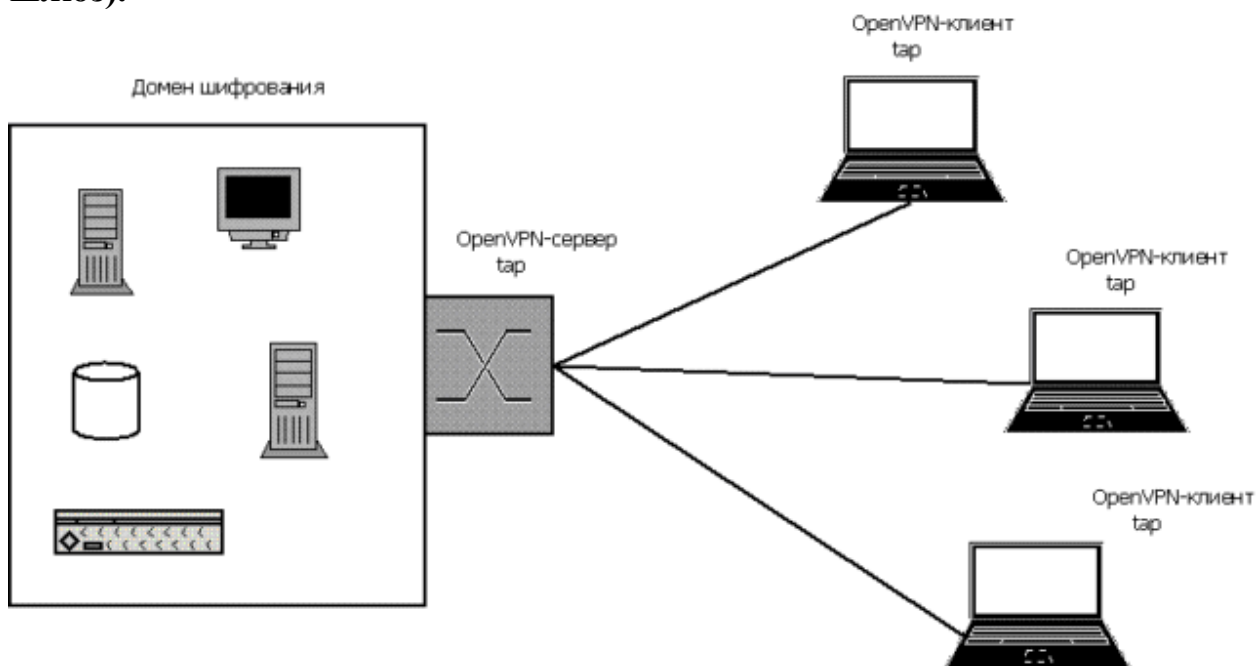
Вышеназванные протоколы объединяет то, что они являются протоколами туннелирования канального уровня. Определению защищенного канала соответствует лишь протокол PPTP, который обеспечивает

туннелирование и шифрование данных. Протокол L2TP, по сути, является только протоколом туннелирования, а функции защиты в них не поддерживаются. Существует вероятность использования данного протокола совместно с протоколом IPSec.

Подробнее про эти протоколы остановимся на одной из следующих лекций. Сейчас кратко рассмотрим **возможные варианты организации «закрытых» сетей** на базе сети Интернет и технологий туннелирования.

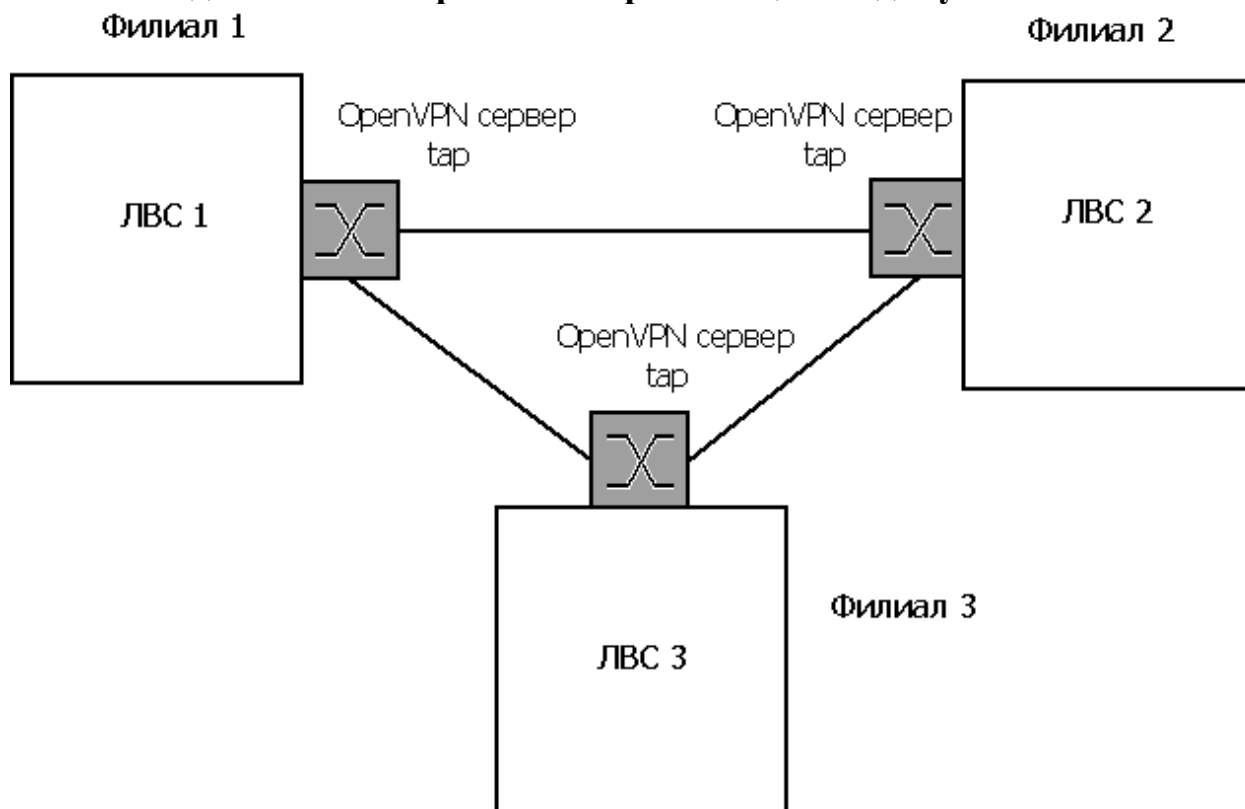
Одним из программных продуктов, реализующих защиту на канальном уровне, является OpenVPN. При подключении к такой сети клиент проходит процедуру строгой криптографической аутентификации по цифровому сертификату, что обеспечивает защиту от несанкционированного доступа к ресурсам сети. Кроме того, обеспечивается шифрование сетевого трафика при работе в сети. OpenVPN поддерживает режимы работы "мост" и "маршрутизатор". При работе в режиме "мост" происходит шифрование и инкапсуляция кадров Ethernet. Следует отметить, что если шифрование обеспечивает защиту от доступа к передаваемой информации, то из-за инкапсуляции злоумышленник не сможет выяснить адресата передаваемой информации.

Подключение удаленного сотрудника к корпоративной ЛВС (VPN-шлюз).



Решение данной задачи предполагает использование серверной части OpenVPN в качестве дополнительного шлюза в ЛВС организации. Сегмент ЛВС, доступный через VPN-шлюз, обычно называют доменом шифрования. Рабочие места и сервера, входящие в домен шифрования, не подключаются к VPN. При подключении к серверу OpenVPN клиент получает прозрачный доступ ко всем машинам, находящимся в домене шифрования. Получат ли подобный доступ к машине клиента машины из домена шифрования, зависит от настроек сервера OpenVPN.

Объединение ЛВС филиалов организации в единую сеть.



Другой задачей, которую можно решить с помощью OpenVPN, является объединение ЛВС филиалов организации в единую сеть через Интернет. В этом случае сервера OpenVPN устанавливается в качестве дополнительных шлюзов в свои ЛВС, а затем соединяются между собой.

Необходимо отметить, что в представленных схемах криптозащищенные туннели организуются только во внешних открытых и небезопасных сетях, например, сети Интернет. Т.е. внутри ЛВС данные на канальном уровне могут передаваться в открытом виде.

Далее приведены общие **рекомендации**, следование которым позволяет дополнительно защитить компьютерную сеть предприятия средствами канального уровня.

1. Администратор службы безопасности должен вести инвентаризационную ведомость соответствия аппаратных и сетевых адресов всех узлов сети предприятия.

2. Службой безопасности, совместно с отделом информационных технологий, должна быть разработана политика защиты компьютерной сети средствами канального уровня, определяющая допустимые маршруты передачи кадров канального уровня. Разработанная политика должна запрещать связи типа «один-ко-многим», не обоснованные требованиями информационной поддержки деятельности предприятия. Политикой также должны быть определены рабочие места, с которых разрешено конфигурирование средств коммутации канального уровня.

3. Средства коммутации канального уровня, используемые в компьютерной сети предприятия, должны быть настраиваемыми и обеспечивать разграничение доступа между узлами сети в соответствии с

разработанной политикой. Как правило, такие средства поддерживают технологию VLAN, позволяющую в рамках одного коммутатора выделить группы аппаратных адресов и сформировать для них правила трансляции кадров.

4. Администратор сети должен выполнить настройку подсистемы управления VLAN коммутатора, и других подсистем, необходимых для реализации разработанной политики защиты. В обязанности администратора входит также отключение неиспользуемых подсистем коммутатора.

5. Администратор сети должен регулярно контролировать соответствие конфигураций коммутаторов разработанной политике защиты.

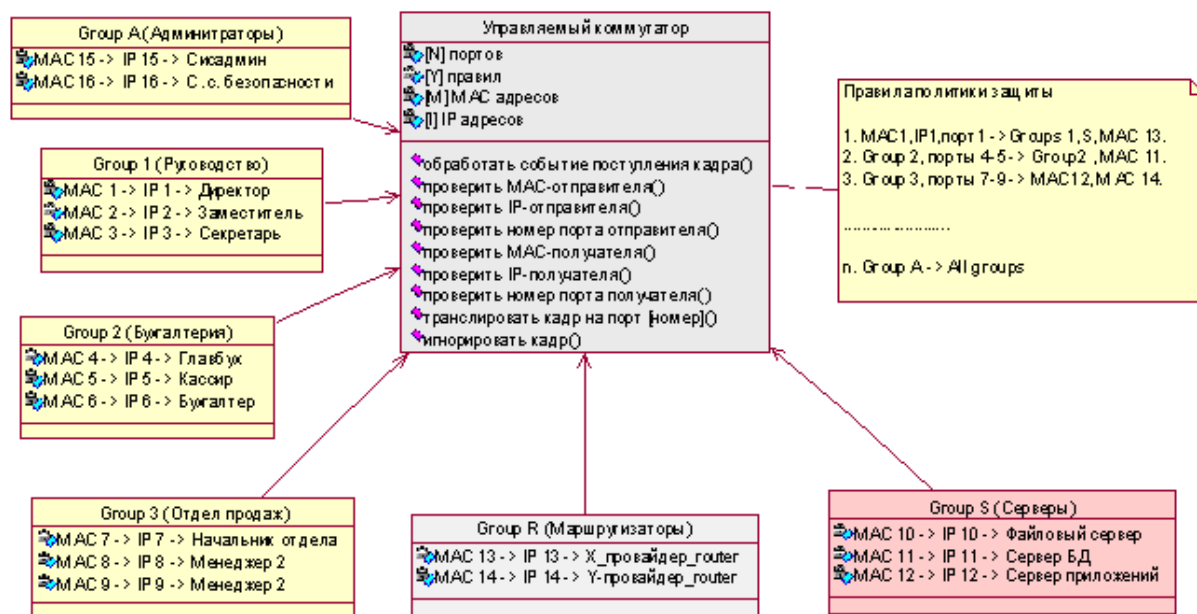
6. Администратор сети должен вести мониторинг сетевой активности пользователей с целью выявления источников аномально высокого количества широковещательных запросов.

7. Служба безопасности должна контролировать регулярность смены реквизитов авторизации администратора в подсистемах управления коммутаторами.

8. Служба безопасности должна контролировать регулярность выполнения администратором мероприятий, связанных с мониторингом сети, осуществлением профилактических работ по настройке коммутаторов, а также созданием резервных копий конфигураций коммутаторов.

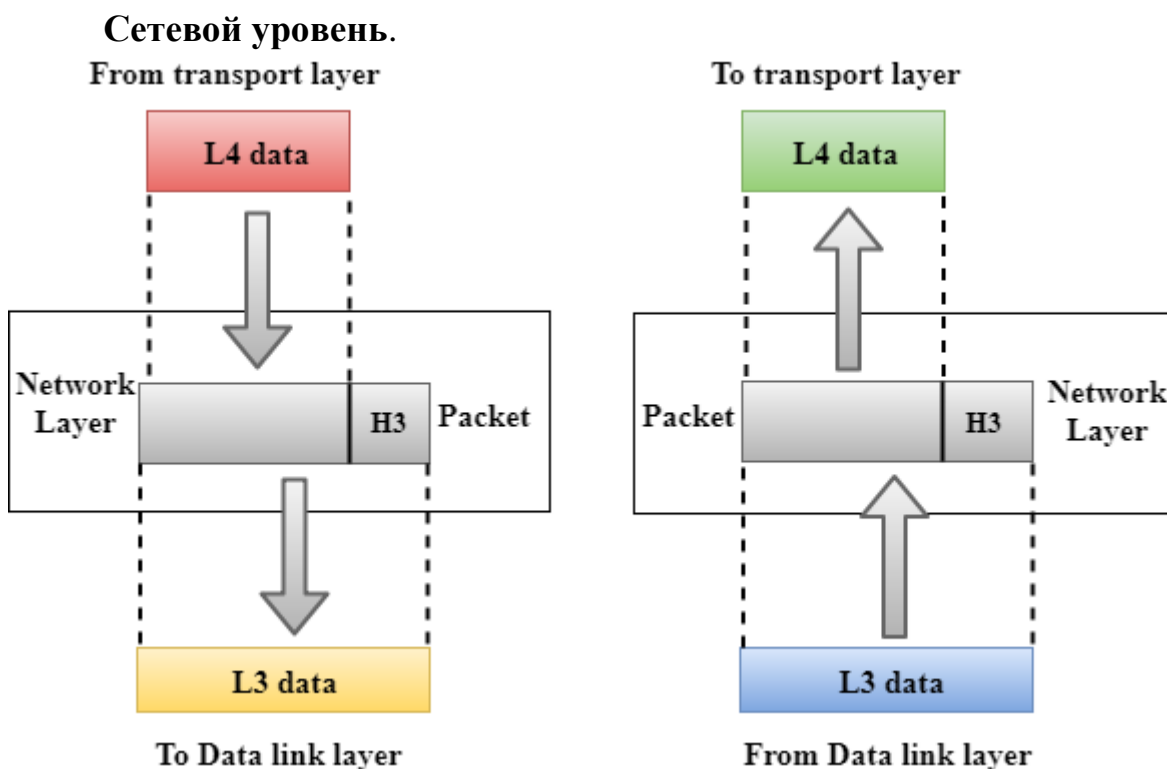
9. Служба безопасности должна обеспечить строгий контроль доступа в помещения, в которых расположены коммутаторы и рабочие станции, с которых разрешено управление коммутаторами.

На рисунке приведен пример формализованной политики защиты компьютерной сети средствами канального уровня.



В центре схемы находится управляемый коммутатор, обеспечивающий реализацию правил политики безопасности. Атрибуты коммутатора перечислены в верхней части блока, а его операции (функции) в нижней. Узлы сети сгруппированы по функциональному признаку. Пример записи правил

фильтрации трафика управляемым коммутатором приведен с права в соответствующей нотации.



Уровень 3 управляет адресацией устройств, отслеживает расположение устройств в сети, отвечает за маршрутизацию и пересылку пакетов.

Он определяет наилучший путь для перемещения данных из источника в место назначения в зависимости от состояния сети, приоритета обслуживания и других факторов.

Маршрутизаторы - это устройства уровня 3, они функционируют на этом уровне и используются для предоставления услуг маршрутизации в пределах межсетевого взаимодействия.

Основные задачи сетевого уровня:

- адресация пакетов;
- перевод логических имен в физические сетевые адреса (и обратно);
- выбор маршрута, по которому пакет доставляется по назначению (если в сети имеется несколько маршрутов).

Протоколы сетевого уровня модели OSI определяют адресацию и процессы, которые позволяют упаковывать и передавать данные транспортного уровня. Инкапсуляция сетевого уровня обеспечивает прохождение данных по сети к адресату (или другой сети) с минимальной нагрузкой. Наиболее часто на сетевом уровне используются протоколы IPv4 и IPv6.

Функции сетевого уровня:

- **межсетевое взаимодействие:** межсетевое взаимодействие является основной обязанностью сетевого уровня. Это обеспечивает логическую связь между различными устройствами.

- **адресация**: сетевой уровень добавляет адрес источника и назначения в заголовок кадра. Адресация используется для идентификации устройства в интернете.

- **маршрутизация**. Маршрутизация является основным компонентом сетевого уровня и определяет оптимальный путь из нескольких путей от источника к месту назначения.

- **пакетирование**: сетевой уровень получает пакеты от верхнего уровня и преобразует их в пакеты. Этот процесс известен как Пакетирование. Это достигается с помощью интернет-протокола (IP).

Критерии безопасности уровня:

1. Статистическая устойчивость интенсивности пропускной способности сети.
2. Допустимость уровня фрагментации пакетов.

Набор показателей:

1. Тип и размер пакетов (хостам рекомендуется отправлять пакеты размером более чем 576 байт, только если они уверены, что принимающий хост или промежуточная сеть готовы обслуживать пакеты такого размера).
2. Результаты трассировки (мы знаем, какой маршрут должен быть обычно, выполняем Tracert (Trace route), а он другой).
3. Уровень фрагментации пакетов.

Сигнатуры:

1. Отсутствие маршрутов в сети.
2. Пропажа пакетов (ошибки маршрутизации).
3. Превышение частоты некоторого заранее установленного порога с / на определенный узел.
4. Внезапное уменьшение пропускной способности сети.
5. Повышенный уровень фрагментации пакетов.
6. Частые повторные передачи пакетов.

Атаки, возможные на сетевом уровне:

1. Подмена default gateway (шлюза по умолчанию).
2. Нарушение процесса маршрутизации.
3. DDoS-атака.

Одной из задач администратора сети и сотрудников службы безопасности является защита адресного пространства сети от возможности его использования злоумышленником. Частично эту функцию выполняют механизмы маршрутизации, реализованные модулями протокола сетевого уровня. Т.е. осуществление обмена между узлами сетей с различными номерами невозможно без предварительной настройки локальных таблиц маршрутизации узлов этих сетей, либо без внесения изменений в конфигурацию маршрутизатора, осуществляющего обмен пакетами (пакетом называется блок данных, с которым работает протокол сетевого уровня).

Однако почти всегда в адресном пространстве сети остается часть адресов, не занятых в настоящий момент и поэтому доступных для эксплуатации злоумышленником. Это объясняется форматом представления

номера сети и номера узла IP-протокола. Количество узлов в сети — это всегда $2n$, т.е. 4,8,16,32,64 и т.д. Реальное же количество узлов не бывает таким. Кроме того, администратор всегда стремится зарезервировать адресное пространство для новых узлов. Именно этот резерв может и будет использован злоумышленником для осуществления атак на функционирующие узлы компьютерной сети.

Решение проблемы очевидно — нужно использовать все адресное пространство и не дать злоумышленнику возможности захватить адреса неиспользуемых узлов. Одним из способов является применение службы мониторинга сети и поддержки виртуальных узлов в резервном диапазоне адресов. Данная служба постоянно использует свободное адресное пространство сети, создавая собственные виртуальные хосты (новые виртуальные хосты создаются сразу после отключения от сети реально функционирующих доверенных узлов). Таким образом, служба подменяет собой отсутствующие в настоящий момент рабочие станции, серверы, маршрутизаторы и т.д.

Устранение уязвимостей компьютерных сетей возможно при создании системы защиты не для отдельных классов приложений, а для сети в целом. Применительно к IP-сетям это означает, что системы защиты должны также действовать на сетевом уровне модели OSI. Реализация защиты сети на третьем уровне гарантирует как минимум такую же степень защиты всех сетевых приложений, причем без какой-либо модификации последних.

Одним из эффективных инструментов реализации защиты на сетевом уровне является протокол IPSec - определенный IETF стандарт достоверной/конфиденциальной передачи данных по сетям IP. IPSec является неотъемлемой частью IPv6 - Интернет-протокола следующего поколения, и расширением существующие версии Интернет-протокола IPv4. IPSec определен в RFC с 2401 по 2412.

В согласовании с идеологией модели OSI и основополагающим принципом IPSec, стандартизованными механизмами IP-безопасности должны пользоваться протоколы более высоких уровней и, в частности, управляющие протоколы, протоколы конфигурирования и маршрутизации. Протоколы IPSec обеспечивают управление доступом, целостность вне соединения, аутентификацию источника данных, защиту от воспроизведения, конфиденциальность и частичную защиту от анализа трафика.

Более подробно данный протокол будет рассмотрен в следующих лекциях.

OpenVPN в режиме "маршрутизатор" обеспечивает защиту информации на сетевом уровне. При этом также происходит строгая аутентификация участников обмена по цифровому сертификату, но шифруются и инкапсулируются IP-пакеты, а не кадры Ethernet. Спектр задач, которые можно решить таким способом, в общем, не отличается от спектра задач, решаемых с помощью OpenVPN в режиме "мост". Следует иметь в виду, что режим "маршрутизатор" является более производительным, чем режим "мост", но имеет и свои недостатки. В частности, не поддерживаются:

- сетевые протоколы, отличные от IP;
- широковещательные запросы.

Транспортный уровень.

Транспортный уровень



- Транспортный уровень обеспечивает сегментирование данных на узле-источнике и управляет сборкой сегментов на узле-назначения.
- Основные функции:
 - - Отслеживание индивидуальных коммуникаций между приложениями на узлах-источниках и узлах-назначения.
 - - Сегментирование данных
 - - Управление сегментами
 - - Сборка сегментов в на узлах-назначения
 - - Идентификация различных приложений

67

Транспортный уровень - это Уровень 4, гарантирующий, что сообщения передаются в том порядке, в котором они были отправлены, и нет дублирования данных.

Основная ответственность транспортного уровня заключается в полной передаче данных. Транспортный уровень является пограничным и связующим между верхними тремя, сильно зависящими от приложений, и тремя нижними уровнями, сильно привязанными к конкретной сети. Он получает данные из верхнего уровня и преобразует их в меньшие единицы, известные как сегменты.

Этот уровень можно назвать сквозным уровнем, поскольку он обеспечивает двухточечное соединение между источником и пунктом назначения для надежной доставки данных.

Два протокола, используемые на этом уровне:

1) **Протокол управления передачей TCP.** Это стандартный протокол, который позволяет системам общаться через Интернет. Он устанавливает и поддерживает связь между хостами.

Когда данные отправляются через соединение TCP, тогда протокол TCP делит данные на более мелкие единицы, известные как сегменты. Каждый сегмент проходит через Интернет, используя несколько маршрутов, и они

прибывают в пункт назначения в разных порядках. Протокол управления передачей переупорядочивает пакеты в правильном порядке на принимающей стороне.

2) **Протокол пользовательских датаграмм UDP.** Протокол пользовательских дейтаграмм - это протокол транспортного уровня.

Это ненадежный транспортный протокол, так как в этом случае получатель не отправляет подтверждение при получении пакета, отправитель не ожидает подтверждения. Следовательно, это делает протокол ненадежным.

Функции транспортного уровня:

- **адресация точки обслуживания:** компьютеры запускают несколько программ одновременно, по этой причине происходит передача данных из источника в место назначения не только с одного компьютера на другой компьютер, но и от одного процесса к другому процессу. Транспортный уровень добавляет заголовок, который содержит адрес, известный как адрес точки обслуживания или адрес порта. Ответственность сетевого уровня заключается в передаче данных с одного компьютера на другой компьютер, а ответственность транспортного уровня - в передаче сообщения правильному процессу.

- **сегментация и повторная сборка:** когда транспортный уровень получает сообщение от верхнего уровня, он разделяет сообщение на несколько сегментов, и каждому сегменту присваивается порядковый номер, который уникально идентифицирует каждый сегмент. Когда сообщение прибыло в пункт назначения, тогда транспортный уровень повторно собирает сообщение на основе их порядковых номеров.

- **управление соединением:** Транспортный уровень предоставляет две службы: служба, ориентированная на соединение, и служба без соединения.

1) **Служба без установления соединения** обрабатывает каждый сегмент как отдельный пакет, и все они перемещаются по разным маршрутам, чтобы достичь пункта назначения.

2) **Служба, ориентированная на установление соединения,** устанавливает соединение с транспортным уровнем на машине назначения - до доставки пакетов. В сервисе, ориентированном на соединение, все пакеты передаются по одному маршруту.

- **управление потоком:** транспортный уровень также отвечает за управление потоком.

- **контроль ошибок:** Транспортный уровень также отвечает за контроль ошибок. Контроль ошибок выполняется сквозным, а не по одной ссылке. Транспортный уровень отправителя гарантирует, что сообщение достигнет пункта назначения без каких-либо ошибок.

Критерии безопасности уровня:

1. Надежность соединения.
2. Способность к обнаружению и исправлению ошибок передачи.
3. Ложность порядка номеров сообщений.

Набор показателей:

1. Проходит / не проходит сегмент.

2. Совокупность количества и портов (существуют ACL блокирующие только по IP — адресам, а есть и другие, которые блокируют и по IP и по портам).

Сигнатуры:

1. Блокируются определенные службы.
2. Аномальное количество сегментов на определенный порт.
3. Аномальное количество запросов на порты.
4. Размер сегментов (размер сегмента представляет собой обычно небольшое число (от 500 байт до 5 килобайт)).
5. Большая или заведомо неполная последовательность формирования сегментов.

Атаки, возможные на транспортном уровне:

1. Подмена UDP пакетов.
2. Посылка / приемка «тяжелых» сегментов.
3. Атаки LAND

Итак, транспортные соединения используются для доступа к конкретному сетевому сервису, например web-сайту, терминальному серверу, почтовому серверу, серверу базы данных и т.п. Логические "концы" соединения называются портами. Использование свойств транспортных протоколов создает наиболее эффективную преграду деятельности злоумышленника. ***Здесь для защиты могут использоваться признаки, содержащиеся в заголовках сегментов (сегмент — блок данных с которыми работает транспортный протокол) транспортного протокола.*** Этими признаками являются тип транспортного протокола, номер порта и флаг синхронизации соединения.

Если средствами канального уровня можно защитить аппаратуру компьютерной сети, а протоколы сетевого уровня позволяют разграничить доступ к отдельным хостам и подсетям, то транспортный протокол используется как средство коммуникации сетевых приложений, функционирующих на платформе отдельных узлов (хостов). Любое сетевое приложение использует транспортный протокол для доставки обрабатываемых данных. Причем у каждого класса приложений имеется специфический номер транспортного порта. Это свойство может быть использовано злоумышленником для атаки на конкретный сетевой сервис или службу, либо администратором сети для защиты сетевых сервисов и служб.

Администратор формирует политику защиты сети средствами транспортного уровня в виде ведомости соответствия хостов, используемых ими сетевых адресов и доверенных приложений, функционирующих на платформах этих хостов. Формализованная запись этой ведомости представляет собой табличную структуру, содержащую:

- перечень узлов (хостов), их символьные имена;
- соответствующие этим узлам (хостам) сетевые адреса;
- перечень используемых каждым узлом (хостом) транспортных протоколов;

— перечень сетевых приложений, функционирующих в каждом узле и соответствующие этим приложениям порты транспортного протокола;

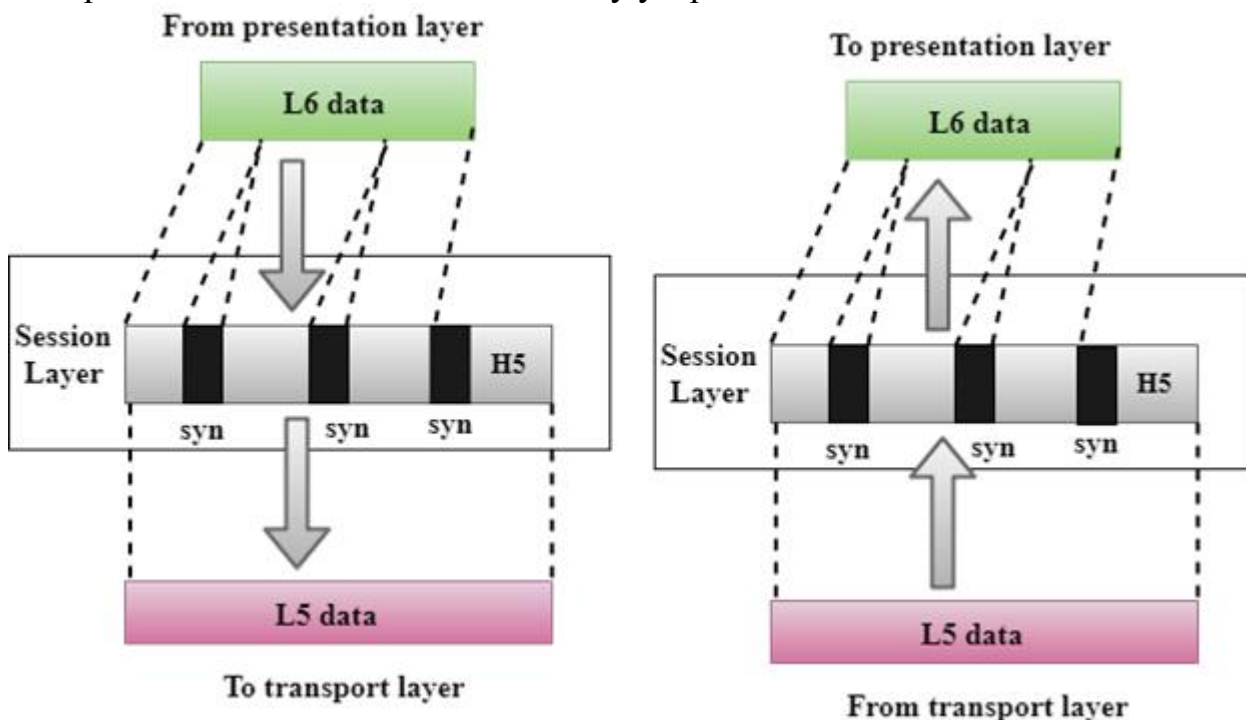
— по каждому сетевому приложению необходимо установить, является ли оно потребителем или поставщиком ресурса, т.е. разрешено ли ему инициировать исходящие соединения или принимать входящие.

Реализация политики защиты средствами транспортного уровня осуществляется с помощью межсетевых экранов (firewall).

Защита верхнего слоя модели OSI.

Кратко рассмотрим основные задачи уровней верхнего слоя модели OSI. Три верхних уровня модели OSI, составляющие верхний слой, по факту являются прикладным уровнем для стека TCP/IP, поэтому иногда сложно разграничить функции различных протоколов, используемых в локальных сетях и сети Интернет, построенных само собой на самом популярном стеке.

Сеансовый уровень используется для установления, поддержания и синхронизации взаимодействия между устройствами связи.

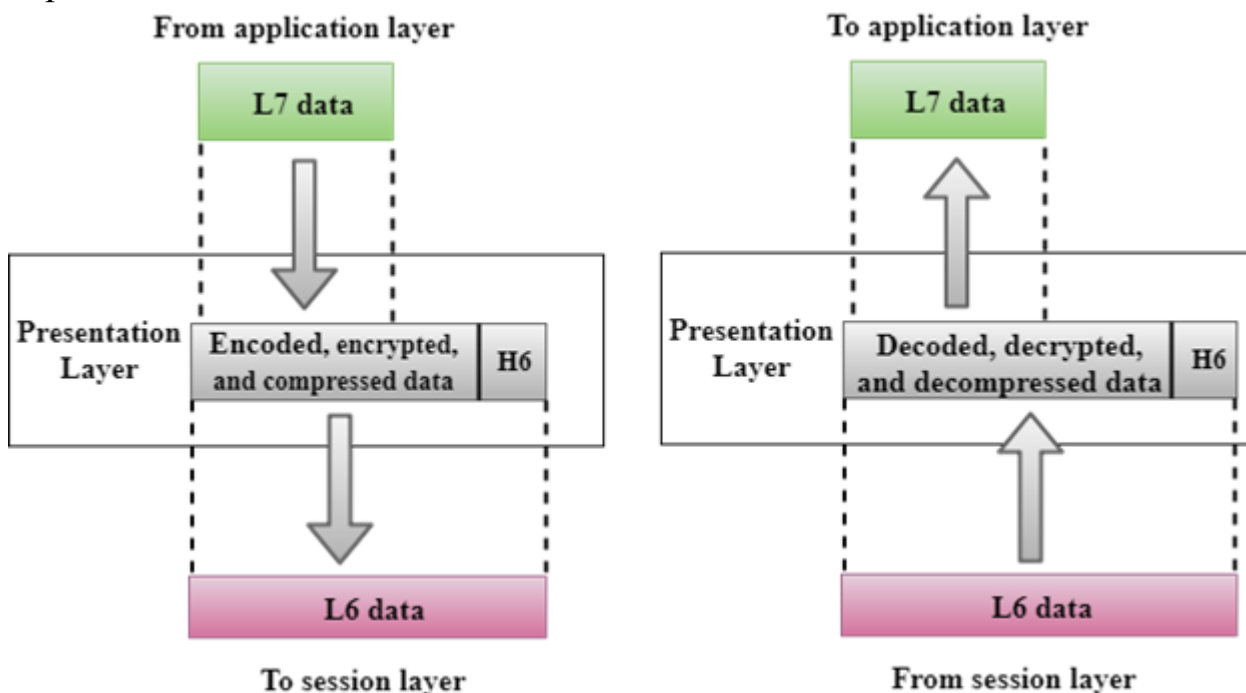


Функции сеансового уровня:

- **диалоговое управление:** сеансовый уровень действует как диалоговый контроллер, который создает диалог между двумя процессами, или мы можем сказать, что он обеспечивает связь между двумя процессами, которые могут быть либо полудуплексными, либо полнодуплексными.

- **синхронизация:** сеансовый уровень добавляет некоторые контрольные точки при передаче данных в последовательности. Если во время передачи данных произойдет какая-либо ошибка, то передача будет повторяться с контрольной точки. Этот процесс известен как Синхронизация и восстановление.

Уровень представления в основном касается синтаксиса и семантики информации, которой обмениваются две системы. Он действует как переводчик данных для сети.



Этот слой является частью операционной системы, которая преобразует данные из одного формата представления в другой формат. Уровень представления также известен как уровень синтаксиса.

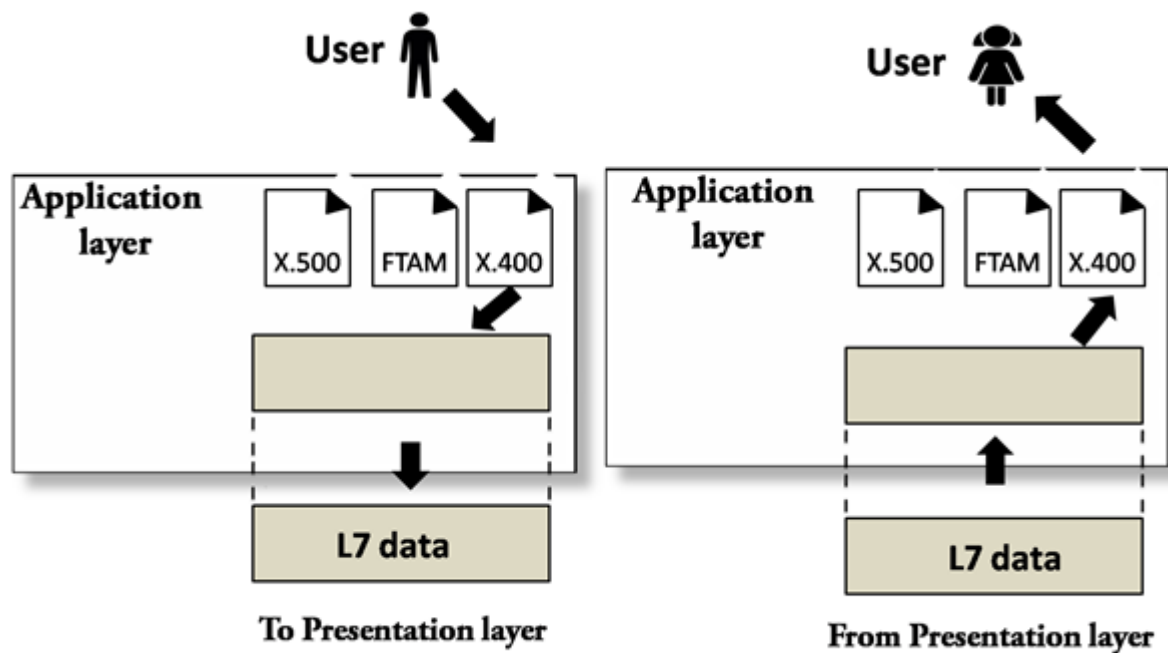
Функции уровня представления:

- **перевод**: процессы в двух системах обмениваются информацией в виде символьных строк, чисел и так далее. Разные компьютеры используют разные методы кодирования, уровень представления управляет взаимодействием между различными методами кодирования. Он преобразует данные из независимого от отправителя формата в общий формат и изменяет общий формат в зависимый от получателя формат на принимающей стороне.

- **шифрование**. Шифрование необходимо для обеспечения конфиденциальности. Именно эта функция верхнего слоя модели OSI позволяет обеспечить защиту данных. Чуть позже станет понятнее.

- **сжатие** - это процесс сжатия данных, т.е. сокращение числа передаваемых битов. Сжатие данных очень важно в мультимедиа, таких как текст, аудио, видео.

Прикладной уровень служит окном для пользователей и процессов приложений для доступа к сетевому сервису. Он решает такие вопросы, как прозрачность сети, распределение ресурсов и т.д.



Прикладной уровень не является приложением, но он выполняет функции прикладного уровня. Этот уровень предоставляет сетевые услуги конечным пользователям.

Функции прикладного уровня:

- **передача, доступ и управление файлами (FTAM):** прикладной уровень позволяет пользователю получать доступ к файлам на удаленном компьютере, извлекать файлы с компьютера и управлять файлами на удаленном компьютере.

- **почтовые службы:** прикладной уровень предоставляет средства для пересылки и хранения электронной почты.

- **службы каталогов:** приложение предоставляет источники распределенной базы данных и используется для предоставления этой глобальной информации о различных объектах.

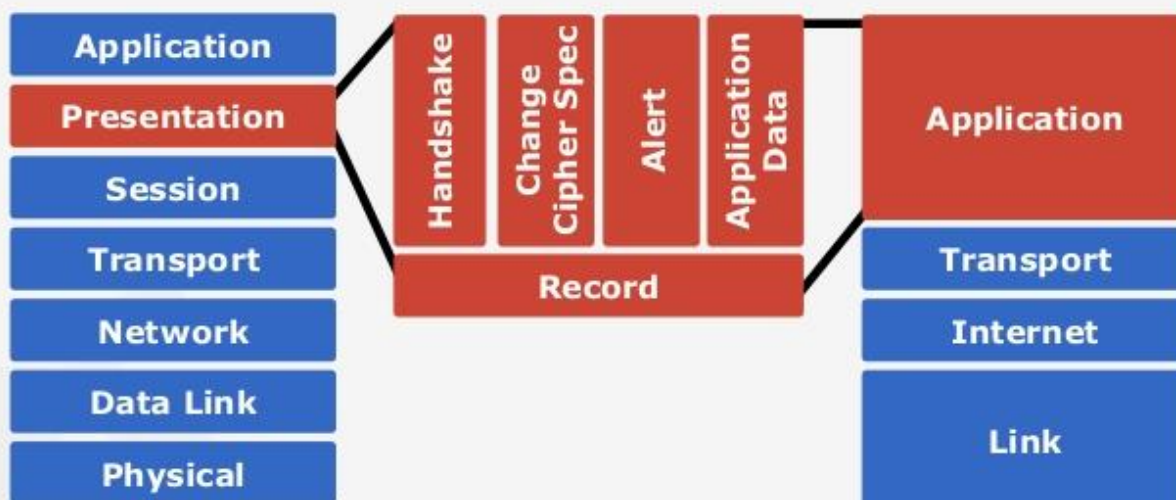
Здесь работает большинство известных и популярных протоколов: HTTP, SMTP, DNS и т.д.

Защита соединений на верхнем слое модели OSI может осуществляться по протоколам SSL/TLS прежде всего за счет шифрования данных.

SSL/TLS in Common Models

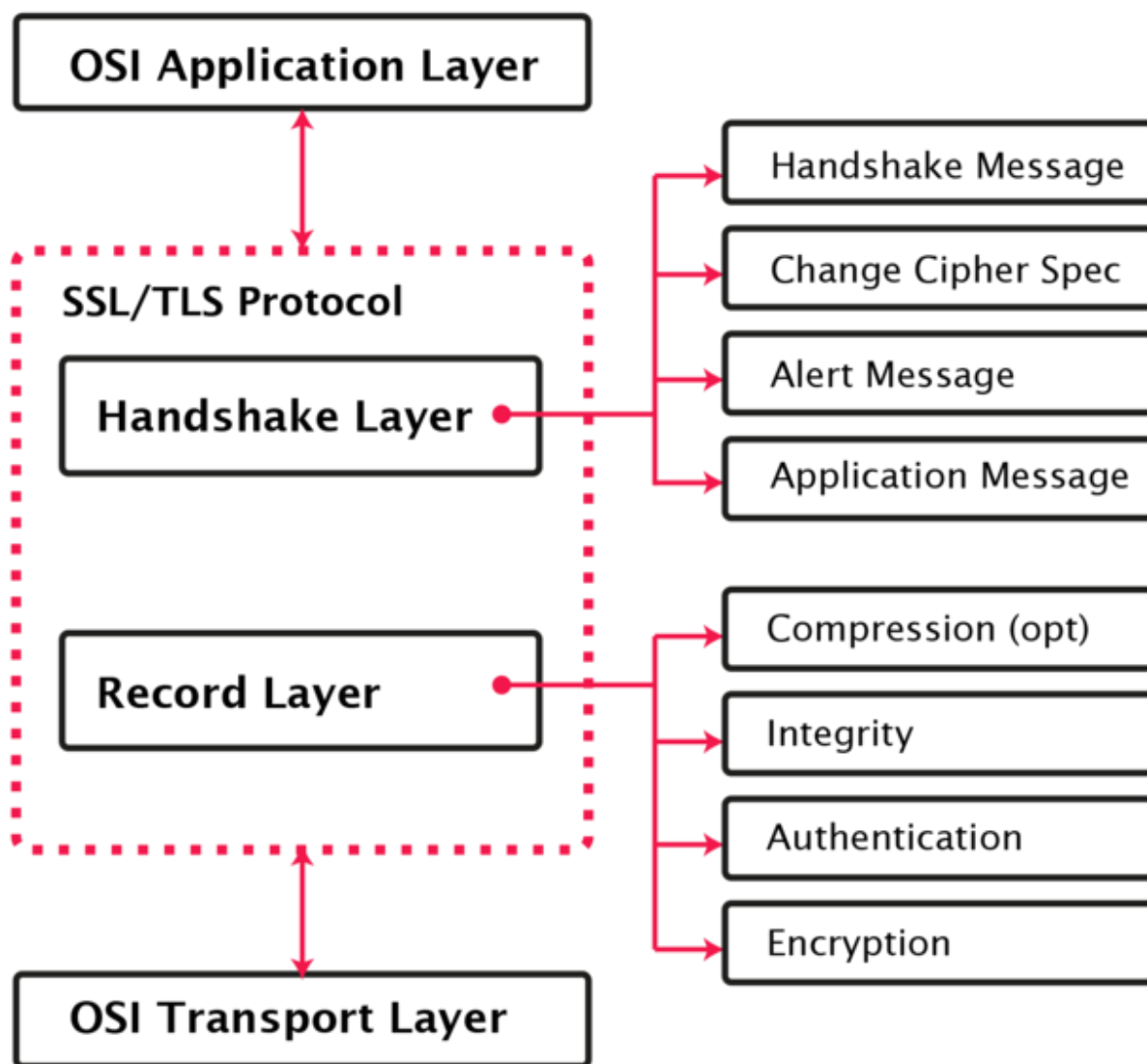
ISO/OSI model

TCP/IP model



Dan Luedtke <mail@danrl.de> • Wed Apr 18, 2012 • University of the German Federal Armed Forces, Munich • Slide 6

На самом деле, трудно отнести эти протоколы к какому-то одному из уровней модели OSI, ввиду того, что они разрабатывались под стек TCP/IP, таким образом ряд основных механизмов SSL/TLS функционирует на представительском уровне (шифрование), другая часть на сеансовом уровне, также есть пересечение с прикладным уровнем и транспортным. Более подробное описание протоколов SSL/TLS будет в лекции про технологию VPN.



Протокол SSL (Secure Socket Layer - уровень защищенных сокетов), разработанный Netscape Communications при участии RSA Data Security, предназначен для реализации защищенного обмена информацией в клиент/серверных приложениях. На практике SSL широко реализуется только совместно с протоколом прикладного уровня HTTP.

Функции безопасности, предоставляемые протоколом SSL:

- шифрование данных с целью предотвратить раскрытие конфиденциальных данных во время передачи;
- подписывание данных с целью предотвратить раскрытие конфиденциальных данных во время передачи;
- аутентификация клиента и сервера.

Протокол SSL использует криптографические методы защиты информации для обеспечения безопасности информационного обмена. Данный протокол выполняет взаимную аутентификацию, обеспечивает конфиденциальность и аутентичность передаваемых данных. Ядро протокола SSL - технология комплексного использования симметричных и асимметричных криптосистем. Взаимная аутентификация сторон выполняется при помощи обмена цифровыми сертификатами открытых ключей клиента и

сервера, заверенными цифровой подписью специальных сертификационных центров. Конфиденциальность обеспечивается шифрованием передаваемых данных с использованием симметричных сессионных ключей, которыми стороны обмениваются при установлении соединения. Подлинность и целостность информации обеспечиваются за счет формирования и проверки цифровой подписи. В качестве алгоритмов асимметричного шифрования применяются алгоритм RSA и алгоритм Диффи-Хеллмана.



Рисунок Криптотуннели, сформированные на основе протокола SSL

Согласно протоколу SSL криптозащищенные туннели создаются между конечными точками виртуальной сети. Клиент и сервер функционируют на компьютерах в конечных точках туннеля.

Протокол TLS (англ. Transport Layer Security) — это стандартный протокол, предназначенный для создания безопасных веб-соединений в Интернете или интрасетях. Он позволяет клиентам выполнять проверку подлинности серверов, а серверам — проверку подлинности клиентов (при необходимости). Этот протокол также обеспечивает защищенный канал путем шифрования передаваемых данных. TLS даёт возможность клиент-серверным приложениям осуществлять связь в сети таким образом, что нельзя производить прослушивание пакетов и осуществить несанкционированный доступ.

TLS-протокол основан на спецификации протокола SSL версии 3.0. Его спецификация определена рабочей группой IETF в документе RFC 2246, Протокол TLS. Последняя вышедшая спецификация протокола описана в

документе RFC 5246. Фактически, TLS 1.0 появился в 1999 году в виде обновления для SSL 3.0. Из-за подверженности SSL различным атакам его последняя третья версия была признана устаревшей в июне 2015 года.

TLS обеспечивает следующие меры безопасности:

- **Конфиденциальность.** Шифрует всю передаваемую информацию, что делает невозможным её прочтение при перехвате. Для шифрования используются симметричные алгоритмы.

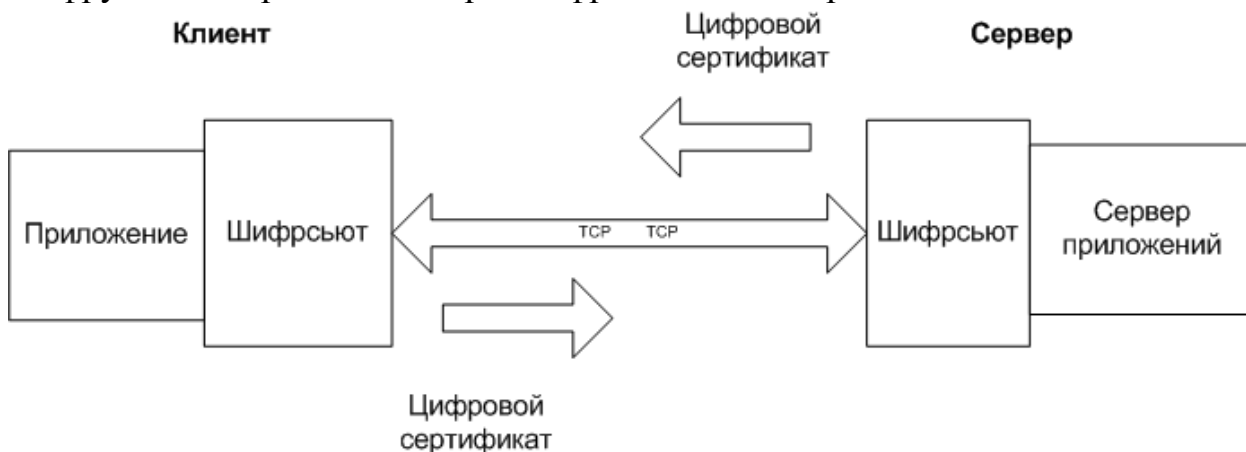
- **Аутентификация.** Подтверждает авторство информации и гарантирует, что обмен данными проходит между теми узлами, для которых изначально был организован защищенный канал.

- **Контроль целостности.** Проверяет получаемую информации на предмет возможной подмены или искажения. Для контроля целостности сообщений используются односторонние функции (хеш-функции).

Цель создания TLS - повышение защиты SSL и более точное и полное определение протокола. Поэтому, ввиду правопримественности, чаще говорят про связку протоколов SSL/TLS.

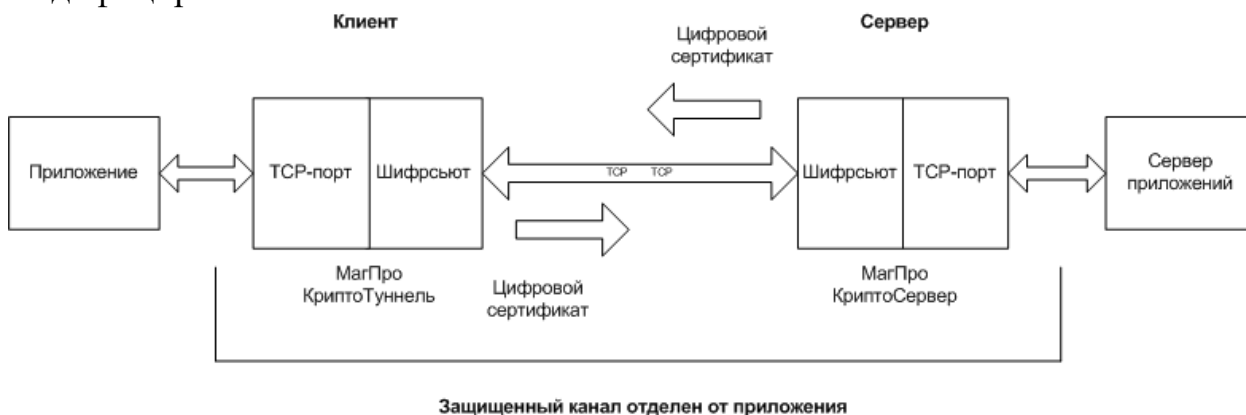
В верхнем слое модели OSI основном используются **2 схемы защиты**:

1) После установки транспортного соединения клиентское приложение инициирует процедуру "рукопожатия" с сервером. В результате этой процедуры происходит аутентификация сервера и клиента по цифровому сертификату, стороны договариваются об используемых алгоритмах шифрования – шифрсыютах. Шифрсыюты специальным образом "привязываются" к портам защищаемого соединения. Благодаря этому, все данные, попадающие в настроенный таким образом транспортный канал, шифруются отправителем и расшифровываются адресатом.



2) Для реализации защиты транспортного соединения приложение, осуществляющее обмен, должно использовать криптографическую библиотеку, реализующую SSL/TLS. Часто бывает, что приложение было написано без защиты, или же используемые им шифрсыюты не удовлетворяют требованиям к безопасности. Например, все web-браузеры и web-сервера используют для защиты шифрсыюты, реализованные по импортным алгоритмам, а в РФ ФСБ требует в ряде случаев использовать шифрсыюты, реализованные по ГОСТ.

Для решения указанной проблемы модуль обеспечения безопасности транспортного уровня следует отделить от приложения. Существенным плюсом данной схемы является то, что сами приложения не приходится модифицировать.



Подобную схему можно реализовать с помощью различных сертифицированных ФСБ и ФСТЭК программных продуктов, например, **КристоТуннель**.

Использование подобных продуктов позволяет обеспечить защиту практически любых «верхних» протоколов модели OSI и приложений, как-то:

- доступ к web-сайту по HTTP
- файловый обмен по WebDAV
- терминальный доступ по RDP (Remote Desktop)
- электронная почта (SMTP, POP3, IMAP)
- доступ к базе данных по SQL
- общие папки по NFS
- произвольный обмен по TCP-соединению (без динамического открытия портов).

Протокол HTTPS.

Любое действие в интернете — это обмен данными. Каждый раз, когда вы запускаете видеоролик, посылаете сообщение в социальной сети или открываете любимый сайт, ваш компьютер отправляет запрос к нужному серверу и получает от него ответ. Как правило, обмен данными происходит по протоколу HTTP. Этот протокол не только устанавливает правила обмена информацией, но и служит транспортом для передачи данных — с его помощью браузер загружает содержимое сайта на ваш компьютер или смартфон.

При всём удобстве и популярности HTTP у него есть один недостаток: данные передаются в открытом виде и никак не защищены. На пути из точки А в точку Б информация в интернете проходит через десятки промежуточных узлов, и, если хоть один из них находится под контролем злоумышленника, данные могут перехватить. То же самое может произойти, когда вы пользуетесь незащищённой сетью Wi-Fi, например, в кафе. Для установки безопасного соединения используется протокол HTTPS с поддержкой шифрования.

Почему HTTPS безопасен

Защиту данных в HTTPS обеспечивает криптографический протокол SSL/TLS, который шифрует передаваемую информацию. По сути этот протокол является обёрткой для HTTP. Он обеспечивает шифрование данных и делает их недоступными для просмотра посторонними. Протокол SSL/TLS хорош тем, что позволяет двум незнакомым между собой участникам сети установить защищённое соединение через незащищённый канал.

Используемая в протоколе HTTPS система TLS предполагает следующую защиту:

- **Все данные шифруются.** Так злоумышленники не смогут узнать, какая информация передается, и отследить действия пользователей на сайте. Для шифрования используется общий секретный ключ, который при установке безопасного соединения выбирают сервер и компьютер. Все ключи одноразовые, перехватить и подобрать очень сложно — длина ключа превышает 100 знаков.

- **Фиксация всех изменений** или искажений данных, даже если это было сделано случайно.

- **Аутентификация** гарантирует, что посетители попадут именно на тот сайт, который им нужен, и защищает от атаки посредника. Для этого у сайта должен быть специальный цифровой сертификат.

Применение HTTPS

В некоторых сервисах, например, в электронных платёжных системах, защита данных исключительно важна, поэтому в них используется только HTTPS. Этот протокол также очень часто применяется и в других сервисах, которые обрабатывают приватную информацию, в том числе любые персональные данные. Все современные браузеры поддерживают протокол HTTPS. Его не нужно специально настраивать — он автоматически включается в процесс, когда это необходимо и возможно.

Многие из вас уже привыкли к безопасности в Интернете, когда в браузере видите закрытый замочек или иную пиктограмму, обозначающую наличие безопасного соединения с сайтом. Это и является признаком использования HTTPS и наличие у сайта сертификата безопасности.

Первое, что делает браузер при установке безопасного соединения — проверяет наличие у сайта сертификата безопасности. Сертификат подтверждает, что организация или лицо, которому он выдан, действительно существует, и веб-адрес ему принадлежит. Выдачей сертификатов занимаются центры сертификации — что-то вроде паспортных столов. Как и в паспорте, в сертификате содержатся данные о его владельце, в том числе имя (или название организации), а также подпись, удостоверяющая подлинность сертификата.

Раздел 2. Средства защиты информации в компьютерных сетях

Базовые средства защиты в компьютерных сетях

1. межсетевые экраны (разграничение доступа),

2. виртуальные частные сети (защита информации от нарушения конфиденциальности и целостности),
3. стойкие протоколы аутентификации (защита от подмены доверенного субъекта),
4. системы обнаружения вторжений (активная идентификация атак),
5. анализ журналов безопасности (аудита) компьютерных систем (идентификация свершившихся атак),
6. сетевые сканеры безопасности.

Базовые технологии безопасности компьютерных сетей могут реализовываться самыми разными средствами обеспечения защиты. Все эти средства используются в различных компонентах КС: в ОС и приложениях, в транспортных протоколах и в сетевых устройствах. И хотя в конкретных продуктах могут быть применены различающиеся реализации технологий безопасности, все они построены в соответствии с едиными принципами, использующими схожие методы и приемы. Знание базовых технологий позволяет быстро «разобраться» с конкретными реализациями программных и аппаратных средств безопасности.

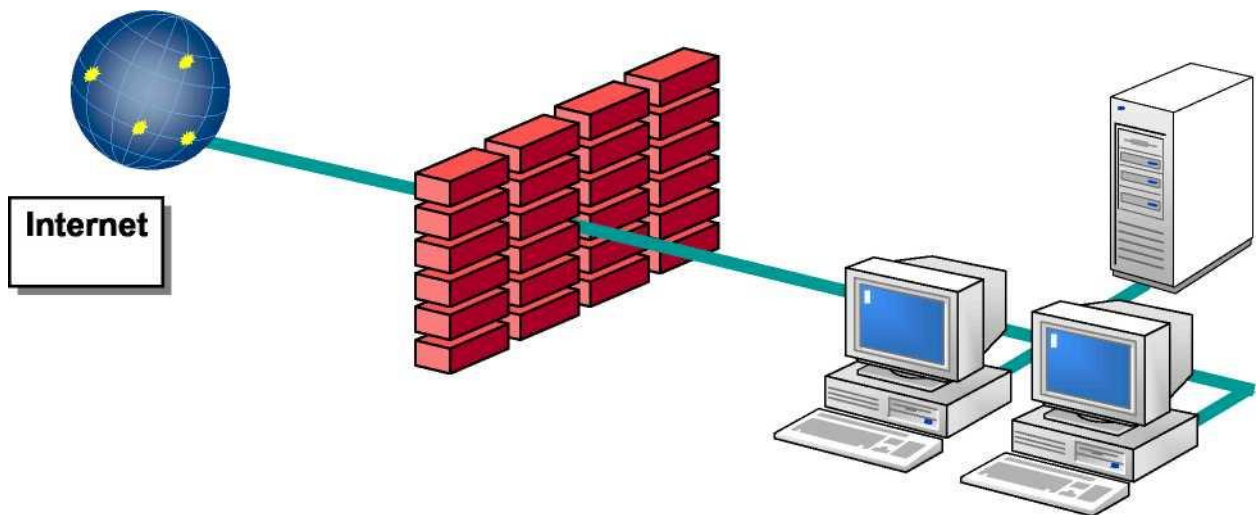
Тема 2.1. Межсетевые экраны, системы обнаружения атак

Межсетевые экраны

Межсетевой экран – это система межсетевой защиты, позволяющая разделить общую сеть на две или более частей и реализовать набор правил, определяющих условия прохождения данных через границу этих сетей.

МЭ называют локальное или функционально распределенное программное (программно-аппаратное) средство (комплекс), реализующее контроль за информацией, поступающей в автоматизированную систему и/или выходящей из автоматизированной системы. МЭ основное название, определенное в РД Гостехкомиссии РФ, для данного устройства. Также встречаются общепринятые названия брандмауэр и firewall (англ. огненная стена).

По определению МЭ служит контрольным пунктом на границе двух сетей. В самом распространенном случае эта граница лежит между внутренней сетью организации и внешней сетью, обычно сетью Интернет. Однако в общем случае, МЭ могут применяться для разграничения внутренних подсетей корпоративной сети организации.



Типовое размещение МЭ в корпоративной сети

Правила фильтрации МЭ

МЭ базируются по одному из двух принципов:

- запрещать все, что не разрешено в явной форме
- разрешать все, что не запрещено в явной форме

Контроль информационных потоков состоит в их фильтрации и преобразовании в соответствии с заданным набором правил. Каждый фильтр на основе анализа проходящих через него данных, принимает решение - пропустить дальше, перебросить за экран, блокировать или преобразовать данные.

Неотъемлемой функцией МЭ является **протоколирование** информационного обмена. Ведение журналов регистрации позволяет администратору провести аудит, выявить подозрительные действия, ошибки в конфигурации МЭ и принять решение об изменении правил МЭ.

Классификация МЭ

Выделяют следующие виды МЭ:

- МЭ нижнего слоя модели OSI (пакетные фильтры, пакетные фильтры с анализом состояния)
- межсетевые экраны сеансового уровня
- межсетевые экраны прикладного уровня
- гибридные технологии межсетевых экранов
- персональные МЭ
- МЭ с функцией NAT

Рассмотрим данные категории подробнее.

Фильтрация трафика на нижнем слое модели OSI

Вообще говоря, разные типы сетевых устройств выполняют фильтрацию трафика с разными правилами, определяющими их функциональность.

Например, коммутаторы, выполняют фильтрацию на канальном уровне модели OSI, и значит анализироваться будут кадры данных, а в правилах

фильтрации будут МАК-адресов компьютеров и соответствующие им номера портов коммутатора. Этот подход позволяет изолировать одну часть сети от другой. МЭ экраны в этом случае называются файерволами канального уровня с фильтрацией на основе списков доступа (access-листами).

Стандартное функционирование МЭ маршрутизаторов определяется правилами фильтрации на основе адресных таблиц. Здесь уже анализируются пакеты данных и их заголовки, в частности IP-адреса отправителя и получателя. На практике условия фильтрации маршрутизаторов, не смотря на свое названия, существенно сложнее, и в них учитывается гораздо больше признаков, объединяющих также возможности более низкого канального уровня и «верхнего» - транспортного уровня. Это могут быть:

- IP-адрес источника и приемника;
- MAC-адрес источника и приемника;
- идентификатор интерфейса, с которого поступил пакет;
- тип протокола, сообщение которого несет IP-пакет;
- номер порта TCP/UDP.

Типу файерволов сетевого уровня соответствуют маршрутизаторы, поддерживающие пользовательские фильтры.

Рассмотрим пакетные фильтры подробнее.

Пакетные фильтры

Базовой возможностью межсетевого экрана является фильтрование пакетов. Первоначально межсетевые экраны были частью маршрутизаторов, обеспечивая управление доступом на основе адресов хостов и коммуникационных сессий. Эти устройства, также называемые межсетевыми экранами без анализа состояния, не поддерживали информацию о состоянии потока трафика, который проходит через межсетевой экран. Это означает, что они не могут определить, что несколько запросов принадлежат одной сессии. Фильтрование пакетов является основой большинства современных межсетевых экранов, хотя осталось немного пакетных фильтров, которые выполняют фильтрование без поддержки состояния. В отличие от более мощных фильтров, пакетные фильтры анализируют только заголовки сетевого и транспортного уровней, а не содержимое пакетов. Управление трафиком определяется набором директив, которые называются ruleset. Возможности фильтрования пакетов встроены в большинство ОС и устройств, выполняющих маршрутизацию. Самым типичным примером является маршрутизатор, в котором определены списки управления доступом.

Характеристики пакетной фильтрации:

- Наиболее поверхностная методология анализа пакетов, проходящих внутри сети.
- Каждый пакет рассматривается по отдельности
- Межсетевой экран читает и анализирует заголовки пакетов.

Управление трафиком осуществляется на основе анализа следующей информации, содержащейся в пакете:

- физический интерфейс, с которого принят пакет и направление (входящий или исходящий).
- IP-адрес источника в пакете – адрес хоста, с которого пришел пакет.
- IP-адрес получателя в пакете – адрес хоста, которому предназначен пакет.
- Транспортный протокол, используемый для взаимодействия хостов отправителя и получателя, такой как TCP, UDP или ICMP.
- Порт отправителя
- Порт получателя (например, TCP 80 для порта получателя и TCP 1320 для порта источника).

Фильтры пакетов без поддержки состояния уязвимы для атак, связанных с особенностями TCP/IP. Например, многие такие пакетные фильтры не могут определить, что информация в сетевом адресе подделана или каким-то образом изменена, или что присутствует комбинация параметров, разрешенная стандартами, но которая использует уязвимости в конкретном приложении или ОС. Атаки спуфинга, такие как использование некорректных адресов в заголовках пакетов, могут дать возможность атакующему обойти контроль, выполняемый межсетевым экраном. Межсетевые экраны, которые выполняются на более высоких уровнях, могут препятствовать некоторым атакам, связанным с подделкой адресов, проверяя, что сессия установлена, или аутентифицируя пользователей перед тем, как разрешить прохождение трафика. В силу этого большинство межсетевых экранов, которые реализуют фильтрацию пакетов, также поддерживают некоторую информацию о состоянии для пакетов, проходящих через межсетевой экран.

В некоторых случаях полезно фильтровать фрагментированные пакеты. Фрагментация пакетов допускается спецификациями TCP/IP и в некоторых ситуациях бывает необходима. Однако фрагментация пакетов делает определение некоторых атак более трудной, так как атака размещается во фрагментированных пакетах. Например, некоторые сетевые атаки используют пакеты, которые в нормальных ситуациях не могут появиться, например, посылая определенные фрагменты пакета, но не посылая первый фрагмент, или посылая фрагменты пакета, которые перекрывают друг друга. Чтобы предотвратить использование фрагментированных пакетов для выполнения атак, межсетевой экран можно сконфигурировать таким образом, чтобы блокировать фрагментированные пакеты.

Основным преимуществом пакетных фильтров является их скорость. Так как пакетные фильтры обычно проверяют данные только в заголовках сетевого и транспортного уровней, они могут выполнять это очень быстро. По этим причинам пакетные фильтры, встроенные в пограничные маршрутизаторы, идеальны для размещения на границе с сетью с невысокой степенью доверия. Пакетные фильтры, встроенные в пограничные маршрутизаторы, могут блокировать основные атаки, фильтруя нежелательные протоколы, выполняя простейший контроль доступа на уровне

сессий и затем передавая трафик другим межсетевым экранам для проверки данных на более высоких уровнях стека протоколов.

Достоинства и недостатки пакетных фильтров

Преимущества пакетных фильтров:

- Основным преимуществом пакетных фильтров является их скорость.

Недостатки пакетных фильтров:

- Так как пакетные фильтры не анализируют данные более высоких уровней, они не могут предотвратить атаки, которые используют уязвимости, специфичные для приложения. Например, пакетный фильтр не может блокировать конкретные команды приложения; если пакетный фильтр разрешает данный трафик для приложения, то все операции, определенные в данном приложении, будут разрешены.
- В логах пакетного фильтра содержится информация только о параметрах сетевого и транспортного уровней. Логи пакетного фильтра обычно содержат ту же информацию, которая использовалась при принятии решения о возможности доступа (адрес источника, адрес назначения, тип трафика и т.п.).
- Большинство пакетных фильтров не поддерживают возможность аутентификации пользователя. Данная возможность обеспечивается межсетевыми экранами, анализирующими более высокие уровни.
- Пакетные фильтры обычно уязвимы для атак, которые используют такие уязвимости TCP/IP, как подделка (spoofing) сетевого адреса. Многие пакетные фильтры не могут определить, что в сетевом пакете изменена адресная информация транспортного уровня. Spoofing-атаки обычно выполняются для обхода управления доступом, осуществляемого межсетевым экраном.
- Пакетные фильтры трудно конфигурировать. Можно случайно переконфигурировать пакетный фильтр для разрешения типов трафика, источников и назначений, которые должны быть запрещены.
- Необходимо открывать все порты с номерами больше 1023. Так как номер порта клиента может быть любым, так называемым "большим номером" (с 1023 до 65535), то на межсетевом экране приходится открывать все порты с номерами больше 1023.

Пакетные фильтры с анализом состояния

Анализ состояния добавляет возможность отслеживания состояния соединения и блокировку пакетов, которые не соответствуют ожидаемому состоянию. Для этого выполняется анализ данных транспортного уровня. Также как и при простом фильтровании пакетов межсетевой экран анализирует содержимое сетевого уровня на соответствие правилам. Но в

отличие от фильтрации пакетов, инспекция состояния отслеживает историю каждого соединения, используя для этого таблицу состояний. Хотя детали записей таблицы состояний во многом зависят от конкретной реализации межсетевого экрана, обычно они содержат IP-адрес источника, IP-адрес получателя и информацию о состоянии соединения.

В TCP-протоколе существуют три основных состояния – соединение устанавливается, используется и завершается. Причем в последнем случае любая из конечных точек может запросить завершение соединения. При анализе состояния межсетевой экран проверяет определенные значения в TCP-заголовках. Для каждого полученного пакета ищется запись в таблице состояний и определяется, что флаги в заголовках пакета соответствует ожидаемому состоянию. Например, атакующий может создать пакет, в заголовке которого указано, что он является частью установленного соединения, в надежде, что он пройдет через межсетевой экран. Если межсетевой экран использует анализ состояний, то он поймет, что пакет не является частью установленного соединения, так как в таблице отсутствует соответствующая запись, и отбросит такой пакет.

В простейшем случае межсетевой экран пропускает любой пакет, если он считает, что пакет является частью открытого соединения (или соединения, которое еще не полностью установлено). Хотя многие межсетевые экраны точно могут определить состояние таких протоколов, как TCP и UDP, и они могут блокировать пакеты, которые не соответствуют состоянию протокола. Например, часто межсетевой экран проверяет такие параметры, как последовательные номера TCP, и отбрасывает пакеты, номера которых вне ожидаемого диапазона.

Если хост из внутренней сети пытается соединиться с хостом за межсетевым экраном, то первым делом проверяется, разрешено ли это набором правил межсетевого экрана. Если это разрешено, то в таблицу состояний добавляется запись, которая указывает, что инициализируется новое соединение. После завершения трехшагового рукопожатия TCP состояние соединения будет изменено на "Установлено" ("Establish" или "TCP_OPEN", в зависимости от реализации), и всему последующему трафику, который соответствует данной записи, будет разрешено проходить через межсетевой экран.

Пакетные фильтры с анализом состояния

- Разбирают и анализируют все пакеты в каждом сетевом соединении.
- Анализируют как пакеты данных, так и их позицию в потоке данных.
- Отслеживает состояние сетевого соединения.
- Устраняет ограничения пакетной фильтрации, которая анализирует пакеты по отдельности.
- Анализирует пакеты как часть соединения.

- Могут анализировать IP-адреса, порты и номера пакетов.
- Могут контролировать установку и завершение соединения.

Преимущества межсетевых экранов с анализом состояния:

- Разрешают прохождение пакетов только для установленных соединений;

Недостатки межсетевых экранов с анализом состояния:

- Реально используются только в сетевой инфраструктуре TCP/IP. Хотя надо отметить, что межсетевые экраны с анализом состояния можно реализовать в других сетевых протоколах тем же способом, что и пакетные фильтры.

Так как некоторые протоколы, в частности UDP, не поддерживают состояния, и для них не существует инициализации, установления и завершения соединения, то для них невозможно определить состояние на транспортном уровне как для TCP. Для этих протоколов межсетевые экраны с поддержкой состояния имеют возможность только отслеживать IP-адреса и порты источника и получателя. Так например ответ DNS от внешнего источника будет пропускаться только в том случае, если межсетевой экран до этого видел соответствующий DNS-запрос от внутреннего хоста. Так как межсетевой экран не имеет возможности определить завершение сессии, запись удаляется из таблицы состояний после заранее сконфигурированного таймаута.

Межсетевые экраны сеансового уровня

МЭ сеансового уровня отслеживают состояние соединений за счет запоминания состояний сеансов протоколов, т.е. другими словами, выполняют операцию *stateful packet inspection* на уровнях, ниже прикладного. Прежде всего имеется ввиду состояние сеанса протокола TCP, его начальной трехшаговой процедуры установления соединения. Например, порядок получения сообщений с флагами SYN и ACK SYN.

МЭ сеансового уровня может защитить сеть от различных типов TCP-атак, в которых нарушается логика установления соединения – SYN Flood, RST, ACK Flood и др.

Для контроля процесса установления соединения МЭ фиксирует текущее состояние соединения, т.е. запоминает какое последнее сообщение отправил клиент и какое сообщение он ожидает получить.

Межсетевые экраны прикладного уровня

Сетевые экраны прикладного уровня способны интерпретировать, анализировать и контролировать содержимое сообщений, которыми обмениваются приложения. Они также работают на основе фильтрации с запоминанием состояния, но анализируют состояния протоколов не только нижних уровней, но и прикладного, таких как SSH, HTTP, FTP, SQL и др.

Такая технология анализа называется глубоким анализом пакета (deep packet inspection). Анализ состояния протокола добавляет в стандартный

анализ состояния базовую технологию обнаружения вторжения, которая анализирует протокол на сеансовом и прикладном уровне модели OSI, сравнивая поведение протокола с определенными производителем профилями и определяя отклонения в поведении. Это позволяет межсетевому экрану разрешать или запрещать доступ, основываясь на том, как выполняется приложение. Например, межсетевой экран прикладного уровня может определить, что почтовое сообщение содержит неразрешенный тип присоединенного файла (такой как выполняемый файл). Другая возможность состоит в том, что он может блокировать соединения, в которых выполняются определенные действия (например, присутствуют команды put в FTP). Данная возможность также позволяет разрешать или запрещать передавать веб-страницы в зависимости от конкретных типов содержимого, такого как Java или ActiveX, или проверять, что SSL сертификаты подписаны конкретным СА.

Межсетевые экраны прикладного уровня могут предоставлять возможность определять нежелательную последовательность команд, такую как некоторые повторяющиеся команды или команда, которой не предшествует другая команда, от которой зависит данная команда. Такие подозрительные команды часто означают атаки переполнения буфера, DoS-атаки или другие атаки, связанные с прикладными протоколами, таким как HTTP.

Другая возможность состоит в проверке входных данных для отдельных команд, такой как минимальная и максимальная длина аргументов. Например, аргумент имени пользователя длиной в 1000 символов является подозрительным или если он содержит бинарные данные. Межсетевые экраны прикладного уровня доступны для многих протоколов, включая HTTP, БД (SQL), почтовые (SMTP, POP, IMAP), VoIP и XML.

Другая возможность, которая встречается в некоторых межсетевых экранах прикладного уровня, состоит в отслеживании состояний приложения, при этом проверяется, что трафик соответствует шаблонам, определенным в спецификациях протокола.

Межсетевые экраны с возможностями анализа состояний и анализа состояний протокола не являются полной заменой IDPS, которые обычно имеют более обширные возможности определения проникновения. Например, IDPS используют сигнатурный и аномальный анализ для определения проблем, связанных с сетевым трафиком.

Возможности МЭ прикладного уровня:

- Идентификация и аутентификация пользователей при попытке установления соединения через МЭ;
- Фильтрация потока сообщений, например, динамический поиск вирусов и прозрачное шифрование информации; Когда пользователи вставляют данные, сервер базы данных прозрачно шифрует эти данные и сохраняет их в столбце. Точно так же, когда пользователи выбирают этот столбец, сервер базы данных автоматически расшифровывает его. Так как все это делается

прозрачно без какого-либо изменения кода приложения, эта функциональная возможность имеет соответствующее название: прозрачное шифрование данных

- Регистрация событий и реагирование на события;
- Кэширование данных, запрашиваемых из внешней сети.

Преимущества межсетевых экранов прикладного уровня:

- Межсетевой экран прикладного уровня имеет возможность выполнять аутентификацию пользователя. Часто существует возможность указывать тип аутентификации, который считается необходимым для данной инфраструктуры.
- Благодаря возможности аутентифицировать пользователя они считаются менее уязвимыми для атак подделки адреса.
- Межсетевые экраны прикладного уровня обычно имеют больше возможностей анализировать весь сетевой пакет, а не только сетевые адреса и номера портов. Например, они могут определять команды и данные, специфичные для каждого приложения.
- Как правило, межсетевые экраны прикладного уровня создают более подробные логи.

Недостатки межсетевых экранов прикладного уровня:

- Так как межсетевые экраны прикладного уровня "знают о пакете все", межсетевой экран вынужден тратить много времени на анализ каждого пакета. По этой причине они обычно не подходят для приложений, которым необходима высокая пропускная способность, или приложений реального времени. Чтобы уменьшить нагрузку на межсетевой экран, можно использовать выделенный прокси-сервер для обеспечения безопасности менее чувствительных ко времени сервисов, таких как e-mail и большинство веб-трафика.
- Другим недостатком является то, что они обрабатывают ограниченное количество сетевых приложений и протоколов и не могут автоматически поддерживать новые сетевые приложения и протоколы. Для каждого прикладного протокола, который должен проходить через межсетевой экран, необходим свой агент. Большинство производителей предоставляют общих агентов для поддержки неизвестных сетевых приложений или протоколов. Однако эти общие агенты не имеют большинства преимуществ межсетевых экранов прикладного уровня: как правило, они просто туннелируют трафик через межсетевой экран.

Гибридные технологии межсетевых экранов

Дальнейшее развитие сетевой инфраструктуры и информационной безопасности привели к стиранию границ между различными типами межсетевых экранов, которые обсуждались выше. Как результат, многие

межсетевые экраны соединяют функциональности нескольких различных типов межсетевых экранов. Например, многие производители прикладных прокси реализуют базовую функциональность пакетных фильтров.

Также многие разработчики пакетных фильтров как с анализом состояний, так и без, реализуют базовую функциональность прикладных прокси для ликвидации слабых мест, связанных с пакетными фильтрами. В большинстве случаев производители реализуют прикладные прокси для улучшения создания логов и аутентификации пользователя.

В результате этого не всегда просто решить, какой продукт наиболее подходит для данного приложения или данной инфраструктуры. Гибридные свойства платформ межсетевых экранов делают особенно важной фазу оценки межсетевого экрана. При выборе продукта важнее оценить поддерживаемые возможности, чем смотреть на формально заявленный тип межсетевого экрана.

Персональные МЭ

Хотя межсетевые экраны, установленные на границе сетевого периметра, обеспечивают определенную защиту внутренних хостов, во многих случаях требуется дополнительная защита сети. Межсетевые экраны, установленные на границы сети, не имеют возможности распознать все варианты и формы атак, позволяя некоторым атакам достигнуть внутренних хостов – после чего атака начинается с одного внутреннего хоста на другой возможно даже не проходя через межсетевой экран, установленный на границы сети. По этой причине разработчики сетевой архитектуры часто добавляют функциональность межсетевого экрана не только в сетевой периметр, что обеспечивает дополнительный уровень безопасности.

Межсетевые экраны для серверов и персональные межсетевые экраны для настольных компьютеров и ноутбуков обеспечивают дополнительный уровень безопасности от сетевых атак. Эти межсетевые экраны являются программными и устанавливаются на хостах, которые они защищают – каждый из них просматривает и управляет входящим и исходящим сетевым трафиком для отдельного хоста. Они обеспечивают более точную защиту, чем межсетевые экраны, расположенные в сети, учитывая конкретные потребности отдельных хостов.

Межсетевые экраны для хостов доступны как часть серверных ОС, таких как Linux, Windows, BSD, Mac OS X Server, они также могут быть установлены в качестве дополнительного компонента от третьих фирм. Политика безопасности межсетевого экрана для отдельного хоста разрешает только необходимый трафик, защищая сервер от вредоносной деятельности всех хостов, включая тех, которые расположены в той же самой подсети или в подсетях, которые не отделены межсетевым экраном. Может быть также полезно ограничение исходящего трафика для предотвращения распространения вредоносного ПО, которым может быть инфицирован хост. Межсетевые экраны для отдельного хоста обычно создают достаточно подробные логи и могут быть сконфигурированы для выполнения управления

доступом на основе адреса и на основе приложения. Многие межсетевые экраны для отдельного хоста также функционируют как системы предотвращения вторжения (IPS), т.е. после определения атаки предпринимают действия по остановке атакующего и предотвращению нанесения вреда защищаемому хосту.

В дополнение к традиционному фильтрованию с учетом состояния многие персональные межсетевые экраны могут быть сконфигурированы для разрешения взаимодействия на основе списка допустимых приложений – таких как веб-браузеры, соединяющиеся с веб-серверами, и почтовые клиенты, посылающие и получающие почтовые сообщения – и могут запрещать взаимодействие с использованием других приложений. Это часто называется межсетевым экраном на основе приложения. Управление доступом в этом случае основано на запуске приложений или сервисов, а не на доступе к портам или сервисам.

Управление персональными межсетевыми экранами должно быть централизовано, если это помогает эффективному созданию, распространению и внедрению политики для всех пользователей и групп. Это будет гарантировать выполнение политики безопасности при получении пользователем доступа к вычислительным ресурсам. Но не зависимо от способа управления любые уведомления, которые создаются межсетевым экраном, должны быть показаны пользователю персонального компьютера, чтобы помочь ему решить обнаруженные проблемы.

Основные функции:

- Блокирование на уровне приложений — позволяют лишь некоторым приложениям или библиотекам исполнять сетевые действия или принимать входящие подключения
- Блокирование на основе сигнатуры - постоянно контролировать сетевой трафик и блокировать все известные атаки.

Преимущества межсетевых экранов для отдельного хоста:

- Сервер защищен лучше, чем если бы он выполнялся на ОС, не имеющей межсетевого экрана, защищающего этот хост. Серверы должны иметь свою собственную защиту. Не следует предполагать, что они не могут быть атакованы только потому, что они расположены позади основного межсетевого экрана.
- Межсетевой экран, защищающий отдельный хост, достаточно хорошо выполняет функции обеспечения безопасности этого хоста.
- ПО, реализующее межсетевой экран для хоста, обычно обеспечивает возможности достаточно точного управления доступом и возможности ограничения трафика для серверов, выполняющихся на том же хосте. Обычно существуют достаточно хорошие возможности создания логов. Хотя межсетевые экраны, защищающие отдельный хост, менее предпочтительны в случае большого трафика и в окружениях с высокими требованиями к

безопасности, для внутренних сетей небольших офисов они обеспечивают адекватную безопасность при меньшей цене.

Недостаток межсетевых экранов для отдельного хоста:

- Каждый такой межсетевой экран необходимо администрировать самостоятельно, и после определенного количества серверов с межсетевыми экранами для отдельного хоста легче и дешевле просто разместить все серверы позади выделенного межсетевого экрана.

Файерволы с функцией NAT.

Одной из функций МЭ является *трансляция сетевых адресов (Network Address Translation, NAT)*. В этом случае фильтрация трафика заключается не в пропуске или отбрасывании пакетов, а в замене внешнего IP-адреса пакета, который использовался при маршрутизации пакета через Интернет, на внутренний, который используется для маршрутизации во внутренней сети, корпоративной или персональной.

Сегодня существует две причины использования технологии NAT: одна из них дефицит IPv4 адресов, а другая – скрытие адресов хостов для повышения безопасности сети. И в том и в другом случаях внутренняя сеть использует частные адреса, которые заменяются на один или несколько публичных адресов при отправке пакетов во внешние сети.

Применение NAT позволяет скрыть адреса узлов сети, чтобы не дать возможности злоумышленникам составить представление о структуре и масштабах корпоративной сети, а также о структуре и интенсивности трафика.

Технология NAT стояла у истоков зарождения МЭ как отдельного класса продуктов. В начале 90-х, когда дефицит адресов еще мало ощущался, несколько специалистов основали компанию Network Translation и разработали программный продукт PIX, который позволял транслировать сетевые адреса. Позднее эту компанию приобрела компания Cisco, а программный продукт стал знаменитым Cisco PIX Firewall, одним из флагманов этого класса средств защиты.

Традиционная технология NAT.

Технология трансляции сетевых адресов имеет несколько разновидностей, наиболее популярная из которых – традиционная технология трансляции сетевых адресов. Она позволяет узлам из частной сети прозрачным для пользователей образом получать доступ к узлам внешних сетей.

Рисунок. Схема действия традиционной технологии NAT (см. презентацию).

Идея технологии NAT состоит в следующем. Пусть сеть предприятия образует тупиковый домен, узлам которого присвоены частные адреса. На маршрутизаторе, связывающем сеть предприятия с внешней сетью, установлено ПО NAT. Это NAT-устройство динамически отображает набор частных адресов IP* на набор глобальных адресов IP, полученных предприятием от поставщика услуг и присвоенных внешнему интерфейсу маршрутизатора предприятия.

Традиционная технология NAT подразделяется на две технологии:

- базовая трансляция сетевых адресов (Basic Network Address Translation, Basic NAT), в которой для отображения используются только IP-адреса;
- трансляция сетевых адресов и портов (Network Address Port Translation, NAPT), в которой для отображения наряду с IP-адресами используются еще и так называемые *транспортные идентификаторы*, в качестве которых чаще всего выступают TCP- и UDP-порты.

Базовая трансляция сетевых адресов

Целью этой технологии является не столько решение проблемы дефицита адресов, сколько обеспечение безопасности.

Если количество локальных узлов, которым необходимо обеспечить выход во внешнюю сеть, меньше или равно имеющегося количества глобальных адресов, то для каждого частного адреса гарантировано однозначное отображение на глобальный адрес. В каждый момент времени кол-во внутренних узлов, которые получают возможность взаимодействовать с внешней сетью, ограничивается кол-ом адресов в глобальном наборе.

Соответствие внутренних адресов внешним задается таблицей, поддерживаемой маршрутизатором или другим NAT-серверным устройством.

Рис. Базовая трансляция сетевых адресов для исходящих сеансов (см. презентацию).

В данной технологии не требуется участия узлов отправителя и получателя, т.е. она прозрачна для пользователей.

Трансляция сетевых адресов и портов

Пусть имеется частная IP-сеть и глобальная связь с Интернетом. Внешнему интерфейсу пограничного маршрутизатора R2 назначен глобальный адрес, а остальным узлам сети назначены частные адреса. NAPT позволяет всем внутренней сети одновременно взаимодействовать с внешними сетями, используя единственный зарегистрированный IP-адрес.

Возникает законный вопрос, какие образом внешние пакеты, поступающие в ответ на запросы из частной сети, находят узел-отправитель, ведь в поле адреса источника всех пакетов, отправляющихся во внешнюю сеть, помещается один и тот же адрес – адрес внешнего интерфейса R2.

Решение состоит в том, что при прохождении пакета из внутренней во внешнюю сеть каждой паре {внутренний частный адрес; номер TCP- или UDP-порта отправителя} ставится в соответствие пара {глобальный IP-адрес внешнего интерфейса; назначенный номер TCP- или UDP-порта}. При этом назначенный номер порта выбирается произвольно и уникально в пределах всех узлов. Соответствие фиксируется в таблице.

Рис. Трансляция сетевых адресов и портов для исходящих TCP- и UDP-сеансов (см. презентацию).

Эта модель при наличии единственного зарегистрированного IP-адреса, удовлетворяет требованиям по доступу к внешним сетям большинства сетей средних размеров.

Схемы подключения МЭ

- Схема единой защиты локальной сети
- Схема с ДМЗ
- Схема с раздельной защитой закрытой и открытой подсетей.

Схема единой защиты локальной сети

Наиболее простым является решение, при котором межсетевой экран просто экранирует локальную сеть от глобальной. При этом WWW-сервер, FTP-сервер, почтовый сервер и другие сервера, оказываются также защищены межсетевым экраном. При этом требуется уделить много внимания на предотвращение проникновения на защищаемые станции локальной сети при помощи средств легкодоступных WWW-серверов.

Межсетевой экран – единственная точка связи между защищаемым и незащищаемым сегментами сети.

Межсетевой экран имеет две сетевых карты

- Одну – в защищаемую сеть
- Одну – в незащищаемую сеть

Относительно недороги и просты в использовании.

DMZ (ДМЗ) - фрагмент сети, не являющийся полностью доверенным. Представляет собой область в сети, системы в которой отделены от основной сети, к этим системам, как правило, предоставляется доступ извне и они не могут считаться доверенными. Смысл создания такого сегмента заключается в том, чтобы отделить системы, к которым осуществляют доступ пользователи интернета, от систем, с которыми работают только сотрудники организации.

Если сервисы предоставляются внешним клиентам по отношению к защищенной сети, то присутствует потенциальный риск. Например, если Port 80 остается открытым. Хакеры могут потенциально скомпрометировать сеть через данный порт и получить полный доступ к системе.

Схема с ДМЗ (DMZ)

Один межсетевой экран, три интерфейса

- Один – в защищаемую сеть
- Второй – в DMZ
- Третий – в незащищенную сеть

DMZ включает системы, которые реализуют сервисы для внешних пользователей (Web or SMTP servers etc.)

Если DMZ скомпрометирована, то доступ к остальной сети все еще ограничен.

Схема с раздельной защитой закрытой и открытой подсетей (Dual Firewalls)

Данная схема подключения обладает наивысшей защищенностью по сравнению с рассмотренными выше. Схема основана на применении двух МЭ, защищающих отдельно закрытую и открытую подсети. Участок сети между

МЭ также называется экранированной подсетью или демилитаризованной зоной (DMZ, demilitarized zone).

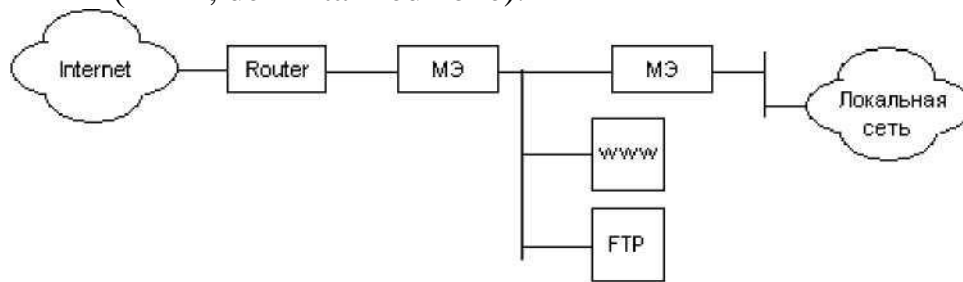


Схема с разделительной защитой закрытой и открытой подсетей

Использование двух МЭ, каждый из которых обладает двумя интерфейсами

- Один межсетевой экран соединяет незащищенную сеть и DMZ
- Второй межсетевой экран соединяет DMZ и защищенную сеть.

DMZ представляет собой буфер между защищенной и незащищенной сетью.

Лучше использовать два различных межсетевых экрана, для уменьшения возможности реализации различных уязвимостей.

Системы обнаружения атак

Система обнаружения вторжений (COB, Intrusion Detection System, IDS) - программное или аппаратное средство, предназначенное для предупреждения, выявления и протоколирования некоторых типов сетевых атак, например, фактов неавторизованного доступа в [компьютерную систему](#) или [сеть](#) либо несанкционированного управления ими в основном через [Интернет](#).

Наряду со стандартными средствами защиты, без которых невозможно нормальное функционирование АС (таких как МЭ, системы резервного копирования и антивирусные средства), существует необходимость использования COA (IDS, систем обнаружения атак или вторжений), которые являются основным средством борьбы с сетевыми атаками.

Необходимость COB:

- Для увеличения вероятности выявления лиц, атакующих сеть.
- Для выявления атак, которые не были предотвращены другими средствами защиты.
- Для выявления подготовительных действий к атакам.
- Для документирования существующих угроз.
- Для получения полезной информации о злоумышленниках.

Отличие COB И МЭ

Хотя и COB, и межсетевой экран относятся к средствам обеспечения информационной безопасности, межсетевой экран отличается тем, что ограничивает поступление на хост или подсеть определенных видов трафика для предотвращения вторжений и не отслеживает вторжения, происходящие

внутри сети. СОВ, напротив, пропускает трафик, анализируя его и сигнализируя при обнаружении подозрительной активности.

Другими словами, в отличие от файерволов, которые строят защиту сети исключительно на основе анализа сетевого трафика, СОВ учитывают в своей работе различные подозрительные события, происходящие в системе. Обнаружение нарушения безопасности проводится обычно с использованием эвристических правил и анализа сигнатур известных компьютерных атак.

Существуют ситуации, когда сетевой экран оказывается проницаемым для злоумышленника, например когда атака идет через туннель VPN из взломанной сети и т.п. И дело здесь не в плохой конфигурации МЭ, а в самом принципе его работы. Экран, несмотря на то что обладает памятью и анализирует последовательность событий, конфигурируется на блокирование трафика с заранее предсказуемыми признаками, например по IP-адресам или протоколам. Так что факт взлома из внешней сети, с которой у него был установлен защищенный канал и которая вела себя вполне корректно, в правилах экрана нельзя отразить, точно также, как и неожиданную попытку легального внутреннего пользователя скопировать файл с паролями или повысить уровень своих привилегий.

Подобные подозрительные действия может обнаружить только система со встроенными агентами во многих точках сети, причем она должна следить не только за трафиком, но и за обращениями к критически важным ресурсам ОС отдельных ПК, а также иметь информацию о перечне подозрительных действий (сигнатур атак) пользователей. Таковой и является СОВ. Она не дублирует действия МЭ, а дополняет их, производя, кроме того, автоматический анализ всех журналов событий, имеющихся у сетевых устройств и средств защиты, чтобы попытаться найти следы атаки, если ее не удалось зафиксировать в реальном времени.

Таким образом, главным отличием IDS от МЭ является то, что в обязанности IDS не входит блокировка подозрительного трафика. IDS только пытается выявить подозрительную активность и поднять тревогу – обычно путем предупреждения администратора сети электронным сообщением. Кроме поднятия тревоги IDS протоколирует подозрительные пакеты, помещая их в журнал.

Существует ряд проблем, с которыми неизбежно сталкиваются организации, развертывающие у себя систему выявления атак. Вот некоторые из них:

- Ложные срабатывания
- Ложные несрабатывания
- Квалификация экспертов по выявлению атак, требующаяся для внедрения и развертывания СОА

Обнаружение вторжений

Алгоритм работы СОВ достаточно прост и состоит из двух основных действий по обнаружению атак и вторжений.

- IDS осуществляет мониторинг системной активности определенным путем.
- Когда определяется несанкционированная активность, IDS принимает определенные решения.

Классификация IDS по типу идентификации атак:

Существует классификация IDS по типу определения атак на КС.

- Signature-based IDS
 - Использует базу данных сигнатур атак
 - Сравнивает текущую активность с БД
 - Для эффективной работы БД должна быть актуальной.

Правила, основанные на подписи атак используют характерную для атаки последовательность символов в данных пакета. Например, правило может диктовать поиск строки “user root” в полях пакета ftp – как известно, этот протокол передает пароли пользователей в открытом виде и использование его суперпользователем root считается грубым нарушением политики безопасности, так что IDS должна отслеживать такие случаи.

Чаще всего подписи атак относятся к прикладным протоколам, для обнаружения вторжения на транспортном уровне они менее пригодны. Для эффективной работы IDS должен иметь обширную постоянно пополняемую базу данных подписей атак.

- Knowledge-based IDS
 - Определяет атаки на основе анализа аномалий
 - Строит профиль «нормального» поведения системы, далее выявляет отклонения от «нормального» поведения.
 - Позволяет выявлять атаки с неизвестными сигнатурами, но обладает большим количеством ошибок.

Правила, основанные на статистических аномалиях трафика, проверяют такие характеристики трафика, как активность хостов, при превышении которой формируется вывод об отклонениях. В принципе любая статистика активности пользователей КС может использоваться для этой цели. Например, если 10% трафика пользователей отдела планирования всегда направлено к серверу базы данных финансового отдела, то появление пользователя, у которого 90% трафика идет на работу с этим сервером, может вызвать подозрение – возможно компьютер этого пользователя захвачен злоумышленником, удаленно пытающимся похитить чувствительные финансовые данные предприятия.

Классификация IDS

Также IDS различают по типу анализируемых протоколов. Правила, основанные на анализе протоколов контролируют логику работы протокола и фиксируют отклонения от него. Так как каждый протокол обладает специфической логикой, то IDS обычно имеет библиотеку программных модулей, каждый из которых может выполнять анализ поведения определенного протокола. Правила анализа протоколов написать существенно

сложнее, чем правила анализа подписи атаки, т.к. для этого нужно хорошо знать логику протокола и возможные попытки ее изменения. Реализация правил анализа протоколов требует большого быстродействия IDS, в противном случае процедура обнаружения вторжений может значительно замедлиться и IDS перестанет быть системой реального времени.

Существует несколько видов IDS по данной классификации.

- [Сетевая COB](#) Network-based IDS (NIDS) – анализирует трафик всего сетевого сегмента. Сетевая система обнаружения вторжений получает доступ к сетевому трафику, подключаясь к [хабу](#) или [свитчу](#), настроенному на [зеркалирование портов](#). В [сетевой COB](#), сенсоры расположены на важных для наблюдения точках сети, часто в [демитаризованной зоне](#), или на границе сети. Сенсор перехватывает весь сетевой трафик и анализирует содержимое каждого пакета на наличие вредоносных компонентов. Примером сетевой COB является [Snort](#).
- Основанное на протоколе COB (Protocol-based IDS, PIDS) представляет собой систему (либо агента), которая отслеживает и анализирует коммуникационные протоколы со связанными системами или пользователями. Для веб-сервера подобная COB обычно ведет наблюдение за HTTP и HTTPS протоколами. При использовании HTTPS COB должна располагаться на таком интерфейсе, чтобы просматривать HTTPS пакеты еще до их шифрования и отправки в сеть.
- Основанная на прикладных протоколах COB (Application Protocol-based IDS, APIDS) — это система (или агент), которая ведет наблюдение и анализ данных, передаваемых с использованием специфичных для определенных приложений протоколов. Например, на веб-сервере с SQL базой данных COB будет отслеживать содержимое SQL команд, передаваемых на сервер.
- Узловая COB [Host-based IDS \(HIDS\)](#) – [устанавливается на отдельном узле \(пользовательский компьютер или сервер\) и анализирует активность только этого узла](#). Это система (или агент), расположенная на хосте, отслеживающая вторжения, используя анализ системных вызовов, логов приложений, модификаций файлов (исполняемых, файлов паролей, системных баз данных), состояния хоста и прочих источников. Примером является [OSSEC](#).
- Гибридная COB совмещает два и более подходов к разработке COB. Данные от агентов на хостах комбинируются с сетевой информацией для создания наиболее полного представления о безопасности сети. В качестве примера гибридной COB можно привести [Prelude](#).

Типовая *архитектура системы выявления атак*, как правило, включает в себя следующие компоненты:

1. Источник данных
2. Датчик/Сенсор (средство сбора информации);
3. Анализатор (средство анализа информации);

4. Оператор IDS
5. Средства реагирования;
6. Средства управления.

Конечно, все эти компоненты могут функционировать и на одном компьютере и даже в рамках одного приложения, однако чаще всего они территориально и функционально распределены. Такие компоненты СОА, как анализаторы и средства управления, опасно размещать за МЭ во внешней сети, т. к. если они будут скомпрометированы, то злоумышленник может получить доступ к информации о структуре внутренней защищаемой сети на основе анализа базы правил, используемой СОА.

Источниками данных для сетевой IDS являются маршрутизаторы, коммутаторы и хосты локальной сети, словом все элементы сети, которые передают, генерируют и принимают трафик.

Датчики или Сетевые сенсоры осуществляют перехват сетевого трафика, циркулирующего в сети и передают их анализатору для выявления подозрительной активности. Датчик может представлять собой отдельный компьютер, подключенный к зеркализованному порту коммутатора, или это может быть программная компонента маршрутизатора, которая имеет доступ к пакетам, буферизуемым в его интерфейсах.

Датчик может осуществлять первичную фильтрацию пакетов, отбирая только те пакеты, которые удовлетворяют некоторым очевидным критериям, например те, которые направлены к публичным веб-сервисам, которые атакуются наиболее часто.

Также существуют хостовые сенсоры используют в качестве источников информации журналы регистрации событий ОС, СУБД и приложений. Информация о событиях также может быть получена хостовым сенсором непосредственно от ядра ОС, МЭ или приложения.

Анализатор, размещаемый на сервере безопасности, осуществляет централизованный сбор и анализ информации, полученной от сенсоров.

Анализатор является «мозгом» IDS, он получает данные от датчиков и проверяет их на наличие угроз и подозрительной активности в сети. Анализатор работает на основе правил, составленных администратором системы безопасности предприятия в соответствии с политикой безопасности. При выполнении условий одного из правил анализатор вырабатывает сообщение «тревога» и передает ее **средству реагирования IDS**.

Средства реагирования IDS – это программная компонента, которая хранит конфигурацию IDS и оповещает оператора IDS о тревоге в виде некоторого уведомления, привлекающего внимание.

Средства реагирования могут размещаться на станциях мониторинга сети, МЭ, серверах и рабочих станциях ЛВС. Типичный набор действий по реагированию на атаки включает в себя оповещение администратора безопасности (средствами электронной почты, вывода сообщения на консоль

или отправки смс), блокирование сетевых сессий и пользовательских регистрационных записей с целью немедленного прекращения атак, а также протоколирование действий атакующей стороны.

Средства управления предназначены для администрирования всех компонентов системы выявления атак, разработки алгоритмов выявления и реагирования на нарушения безопасности (политик безопасности), а также для просмотра информации о нарушениях и генерации отчетов.

Средствами управления пользуется **оператор IDS**, который на основе получаемых уведомлений принимает решение о реакции сети на подозрительную активность — это может быть отключение сетевого интерфейса, через который поступает подозрительный трафик, изменение правил МЭ для блокирования определенных пакетов или же игнорирование уведомления, если оператор считает, что вероятность вторжения мала. В любом случае все данные о потенциальном вторжении протоколируются в журнале и могут быть использованы в последствии для повторного анализа ситуации.

Описанная выше архитектура является функциональной, в реальной IDS эти функции не обязательно реализуются в отдельных блоках или модулях системы. В минимальном варианте все функции IDS могут быть сосредоточены в ПО единственного ПЭВМ, сетевой адаптер которого выполняет роль сенсора, за счет того, что он присоединен к зеркализованному порту коммутатора или маршрутизатора.

Пример размещения COB.

В наиболее общем случае COB размещается внутри корпоративной КС, в месте циркуляции основных информационных потоков. Как видно из примера, IDS подключен к одному из портов коммутатора, который зеркалирует весь трафик сети на порт сенсора. При выявлении аномальной активности, такой как вторжение хакера, он извещает об этом сервер управления IDS, который действует по заданному сценарию.

Размещение сенсоров IDS

Существуют особенности размещения сенсоров IDS в компьютерных сетях. Как правило сенсоры размещаются в следующих точках:

- между маршрутизатором и межсетевым экраном;
- в "демилитаризованной зоне" (DMZ);
- за межсетевым экраном;
- у сервера удаленного доступа или у модемной стойки;
- в ключевых сегментах внутренней сети.

Более масштабируемой является реализация COB с несколькими датчиками-сенсорами, подключенными к различным сегментам сети, и посылающие захваченный трафик центральному анализатору. Такие сенсоры могут быть дополнительными ПО маршрутизатора или коммутатора, или же представлять собой отдельные аппаратные устройства.

Но для более полного понимания нюансов размещения элементов IDS рассмотрим три первые варианта установки сенсоров.

Размещение между маршрутизатором и межсетевым экраном.

Этот вариант позволит контролировать весь трафик, входящий в корпоративную сеть (в том числе и в демилитаризованную зону), а также весь исходящий трафик, который не блокируется межсетевым экраном. Данное решение также позволит защитить сам межсетевой экран, который часто является целью для атак злоумышленников.

Однако при таком положении сетевого сенсора он не сможет контролировать трафик, изолируемый межсетевым экраном и маршрутизатором, а также циркулирующий в локальной «демилитаризованной зоне», и исходящий из DMZ в локальную сеть. Кроме того, не стоит упускать из виду, что трафик, попадающий в сеть не через контролируемую сетевым сенсором точку (например, через резервное соединение или модем), не будет им проанализирован, и соответственно, атаки в неучтенном графике не будут обнаружены.

Размещение в "демилитаризованной зоне" (DMZ).

При размещении сенсоров в "демилитаризованной зоне" контролируется весь информационный поток, связанный с обращениями и ответами от ресурсов DMZ, например, веб-сервера, файловые и почтовые сервера и т.д.

При этом трафик, не проходящий через контролируемую зону, не анализируется сетевым сенсором системы обнаружения атак. Размещение сенсора в демилитаризованной зоне обычно практикуется компаниями, активно использующими внешне доступные ресурсы (электронные магазины, internet-порталы и т. п.).

Размещение за межсетевым экраном.

Размещение сенсоров за МЭ позволяет СОВ контролировать трафик проходящий как внутри КС организации, так и обращения пользователей во внешние сети и ДМЗ, но контроль трафика до МЭ, например между сетью Интернет и ДМЗ уже не ведется.

Этот подход обычно практикуется в дополнение к первому варианту. Такое расположение сенсора позволяет гарантировать, что межсетевой экран правильно настроен и никто не может через него проникнуть в корпоративную сеть, т.е. сетевой сенсор является средством контроля эффективности конфигурации межсетевого экрана. Одновременная регистрация одинаковых событий на обоих сенсорах (до и после МСЭ) позволит сравнить число атак, обнаруженных у межсетевого экрана и за ним, тем самым могут быть замечены "пробелы" в созданных администратором безопасности правилах.

Пассивные и активные системы обнаружения вторжений

- В пассивной СОВ при обнаружении нарушения безопасности, информация о нарушении записывается в лог приложения, а также сигналы опасности отправляются на консоль и/или администратору системы по определенному каналу связи.

- В активной системе, также известной как Система Предотвращения Вторжений (IPS), COB ведет ответные действия на нарушение, сбрасывая соединение или перенастраивая межсетевой экран для блокирования трафика от злоумышленника. Ответные действия могут проводиться автоматически либо по команде оператора.

Возможные действия пассивной COB

- Звуковое/ Визуальное оповещение
- Сообщение по email.
- SMS сообщение
- Запись в лог.
- Протоколирование пакета.
- Запуск программы.
- Переконфигурирование МЭ (для IPS)
- Обрыв соединения (для IPS)

Активные COB – Системы предотвращения вторжений.

Существуют также **Системы предотвращения/предупреждения вторжений** (Intrusion Prevention System, IPS), которые выполняют автоматические действия по прекращению атаки в случае ее обнаружения. IPS это программная или аппаратная система сетевой и компьютерной безопасности, обнаруживающая вторжения или нарушения безопасности и автоматически защищающая от них.

Системы IPS можно рассматривать как расширение IDS, так как задача отслеживания атак остается одинаковой. Однако, они отличаются в том, что IPS должна отслеживать активность в реальном времени и быстро реализовывать действия по предотвращению атак.

Часто такие системы перепоручают эту работу файерволу, передавая ему новое правило для блокировки подозрительного трафика, причем без участия оператора, как это происходит в IDS системах.

Классификация IPS

- Сетевые IPS (Network-based Intrusion Prevention, NIPS) - отслеживают трафик в компьютерной сети и блокируют подозрительные потоки данных. Их работа заключается в выявлении угроз на основе анализа трафика, проходящего через один или несколько локальных сегментов КС. Такая IDS анализирует все поля пакетов, в т.ч. и поле данных, которое переносит информацию приложений, поэтому ее возможности по обнаружению подозрительной активности гораздо больше, чем у систем, которым доступны только поля заголовков протоколов Ethernet и IP.
- IPS для беспроводных сетей (Wireless Intrusion Prevention Systems, WIPS) - проверяет активность в беспроводных сетях. В частности,

обнаруживает неверно сконфигурированные точки беспроводного доступа к сети, [атаки человек посередине](#), [спуфинг](#) mac-адресов.

- Анализатор поведения сети (Network Behavior Analysis, NBA) - анализирует сетевой трафик, идентифицирует нетипичные потоки, например DoS и [DDoS](#) атаки.
- [IPS для отдельных компьютеров](#) (Host-based Intrusion Prevention, HIPS) - резидентные программы, обнаруживающие подозрительную активность на компьютере. Они анализируют события, происходящие в ОС и приложениях.

Тема 2.2. Виртуальные частные сети (VPN)

Виртуальные частные сети (VPN). Концепция построения защищенных виртуальных частных сетей

В основе концепции построения защищенных виртуальных частных сетей VPN лежит достаточно простая идея: если в глобальной сети есть два узла, которые хотят обменяться информацией, то для обеспечения конфиденциальности и целостности передаваемой по открытым сетям информации между ними необходимо построить виртуальный туннель, доступ к которому должен быть чрезвычайно затруднен всем возможным активным и пассивным внешним наблюдателям. Термин «виртуальный» указывает на то, что соединение между двумя узлами сети не является постоянным (жестким) и существует только во время прохождения трафика по сети.

Преимущества, получаемые компанией при формировании таких виртуальных туннелей, заключаются, прежде всего, в значительной экономии финансовых средств, за счет использования общедоступной инфраструктуры Интернета и сетей провайдеров.

Функции и компоненты сети VPN

Виртуальной частной сетью (Virtual Private Network, VPN) называют объединение локальных сетей и отдельных компьютеров через открытую внешнюю среду передачи информации в единую виртуальную корпоративную сеть, обеспечивающую безопасность циркулирующих данных.

Есть еще несколько определений VPN. Одно из них называет ***виртуальной частной сетью*** инфраструктуру логических соединений в публичной сети, которая имитирует свойства частной сети, такие как безопасная передача данных, независимая адресация узлов и, возможно, гарантированная пропускная способность.

При подключении корпоративной локальной сети к открытой сети возникают угрозы безопасности двух основных типов:

- несанкционированный доступ к корпоративным данным в процессе их передачи по открытой сети;
- несанкционированный доступ к внутренним ресурсам корпоративной локальной сети, получаемый злоумышленником в результате несанкционированного входа в эту сеть.

Защита информации в процессе передачи по открытым каналам связи основана на выполнении следующих основных функций:

- аутентификации взаимодействующих сторон;
- криптографическом закрытии (шифровании) передаваемых данных;
- проверке подлинности и целостности доставленной информации.

Для этих функций характерна взаимосвязь друг с другом. Их реализация основана на использовании криптографических методов защиты информации.

Туннелирование

Защита информации в процессе ее передачи по открытым каналам основана на построении защищенных виртуальных каналов связи, называемых криптозащищенными туннелями. Каждый такой туннель представляет собой соединение, проведенное через открытую сеть, по которому передаются криптографически защищенные пакеты сообщений.

Основные задачи, решаемые VPN

1. Аутентификация взаимодействующих сторон.
2. Криптографическая защита передаваемой информации.
3. Подтверждение подлинности и целостности доставленной информации. (ЭЦП)
4. Защита от повтора, задержки и удаления сообщений. (Так как есть уведомления, подтверждение доставки, то повторы невозможны)
5. Защита от отрицания фактов отправления и приема сообщений. (Уведомления)

Создание защищенного туннеля выполняют компоненты виртуальной сети, функционирующие на узлах, между которыми формируется туннель. Эти компоненты принято называть инициатором и терминатором туннеля. **Инициатор туннеля** инкапсулирует (встраивает) пакеты в новый пакет, содержащий наряду с исходными данными новый заголовок с информацией об отправителе и получателе. Хотя все передаваемые по туннелю пакеты являются пакетами IP, инкапсулируемые пакеты могут принадлежать к протоколу любого типа, включая пакеты немаршрутизируемых протоколов, таких, как NetBEUI. Маршрут между инициатором и терминатором туннеля определяет обычная маршрутизируемая сеть IP, которая может быть и сетью отличной от Интернет. **Терминатор туннеля** выполняет процесс обратный инкапсуляции — он удаляет новые заголовки и направляет каждый исходный пакет в локальный стек протоколов или адресату в локальной сети.

Рис. Пример пакета, подготовленного для туннелирования (см. презентацию)

Сама по себе инкапсуляция никак не влияет на защищенность пакетов сообщений, передаваемых по туннелю. Но благодаря инкапсуляции появляется возможность полной криптографической защиты инкапсулируемых пакетов. Конфиденциальность инкапсулируемых пакетов обеспечивается путем их криптографического закрытия, то есть шифрования, а целостность и подлинность — путем формирования цифровой подписи. Поскольку существует большое множество методов криптозащиты данных, очень важно, чтобы инициатор и терминатор туннеля использовали одни и те же методы и могли согласовывать друг с другом эту

информацию. Кроме того, для возможности расшифровывания данных и проверки цифровой подписи при приеме инициатор и терминатор туннеля должны поддерживать функции безопасного обмена ключами. Ну и наконец, чтобы туннели создавались только между уполномоченными пользователями, конечные стороны взаимодействия требуется аутентифицировать.

Классификация виртуальных частных сетей VPN:

VPN различаются прежде всего тем, какие свойства частной сети они имитируют и в какой степени. Но общая классификация VPN обычно проводится по нескольким базовым принципам, например:

- По типу используемых каналов
- По архитектуре технической реализации
- По протоколам туннелирования и уровням эталонных моделей
- По способу технической реализации

Классификация по типу используемых каналов:

- VPN на основе защищенных каналов с шифрацией и аутентификацией, например IPSec. Это самый популярный класс, поскольку может реализовываться собственными силами, без привлечения ресурсов VPN сторонних организаций;

- VPN на основе логических (виртуальных) каналов транспортных технологий с установлением соединений, например технологии MPLS. Этот класс VPN в основном используется провайдерами Интернет и крупными организациями, имеющими собственную транспортную инфраструктуру для передачи данных.

Поскольку в основе работы большинства предприятий лежит использование открытых публичных сетей, то защищенный канал конечно же проще и дешевле образовать собственными силами. Таким образом, от провайдеров требуется только предоставление стандартного доступа в Интернет. В этом состоит преимущество VPN на основе шифрования, чем от технологии разделения трафика на логические каналы.

Классификация VPN по архитектуре технической реализации

- внутрикорпоративные VPN;
- межкорпоративные VPN;
- VPN с удаленным доступом.

Внутрикорпоративные сети VPN (intranet-VPN) предназначены для обеспечения защищенного взаимодействия между подразделениями внутри предприятия или между группой предприятий, объединенных корпоративными сетями связи, включая выделенные линии.

Конечные точки защищенного туннеля совпадают с конечными точками защищаемого потока сообщений.

Межкорпоративные сети VPN (extranet-VPN) обеспечивают сотрудникам предприятия защищенный обмен информацией со стратегическими партнерами по бизнесу, поставщиками, крупными заказчиками, пользователями, клиентами и т.д.

Extranet-VPN обеспечивает прямой доступ из сети одной компании к сети другой, тем самым способствуя повышению надежности связи, поддерживаемой в ходе делового сотрудничества. В межкорпоративных сетях большое значение придается контролю доступа посредством межсетевых экранов и аутентификации пользователей.

Конечные точки защищенного туннеля совпадают с МЭ или пограничным маршрутизатором локальной сети.

Виртуальные частные сети VPN с удаленным доступом (Remote Access) предназначены для обеспечения защищенного удаленного доступа к корпоративным информационным ресурсам мобильным и/или удаленным (home-office) сотрудникам компании.

Удаленный доступ через VPN клиентских компьютеров, которые подключаются к VPN-серверу.

Классификация VPN по протоколам туннелирования и уровням эталонных моделей

1. Канальный уровень - PPTP, L2F, L2TP;
2. Сетевой уровень - IPSec, SKIP;
3. Транспортный, **сеансовый** и представительский уровни (для TCP/IP – транспортный и прикладной) – SSL, TLS;
4. Прикладной уровень – S/MIME, HTTPS, SSH, PGP.

Для технологий безопасной передачи данных по общедоступной (незащищенной) сети применяют обобщенное название – защищенный канал (secure channel). Термин «канал» подчеркивает тот факт, что защита данных обеспечивается между двумя узлами сети вдоль некоторого виртуального пути, проложенного в сети с коммутацией пакетов. Защищенный канал можно построить с помощью системных средств, реализованных на разных уровнях модели взаимодействия открытых систем, известной нам как модели OSI.

От выбранного уровня OSI во многом зависит функциональность реализуемой VPN и ее совместимость с приложениями корпоративной ИС, а также с другими средствами защиты.

VPN строятся на достаточно низких уровнях модели OSI. Причина этого в том, что чем ниже в стеке реализованы средства защищенного канала, тем проще их сделать прозрачными для приложений и прикладных протоколов. Однако здесь возникает другая проблема - зависимость протокола защиты от конкретной сетевой технологии.

Рис. Уровни протоколов защищенного канала (см. презентацию)

Если для защиты данных используется протокол одного из верхних уровней (прикладного или представительского), то такой способ защиты не зависит от того, какие сети (IP или IPX, Ethernet или ATM) применяются для

транспортировки данных, что можно считать несомненным достоинством. С другой стороны, приложение при этом становится зависимым от конкретного протокола защиты, то есть для приложений подобный протокол не является прозрачным.

Защищенному каналу на самом высоком, прикладном уровне свойствен еще один недостаток = ограниченная область действия. Протокол защищает только вполне определенную сетевую службу - файловую, гипертекстовую или почтовую. Например, протокол S/MIME защищает исключительно сообщения электронной почты. Поэтому для каждой службы необходимо разрабатывать соответствующую защищенную версию протокола.

Таким образом, на верхних уровнях модели OSI существует жесткая связь между используемым стеком протоколов и приложением.

VPN канального уровня

Средства VPN, используемые на канальном уровне модели OSI, позволяют обеспечить инкапсуляцию различных видов трафика третьего уровня (и более высоких уровней) и построение виртуальных туннелей типа «точка-точка» (от маршрутизатора к маршрутизатору или от персонального компьютера к шлюзу ЛВС). К этой группе относятся VPN-продукты, которые используют протоколы L2F (Layer 2 Forwarding) и PPTP (Point-to-Point Tunneling Protocol), а также стандарт L2TP (Layer 2 Tunneling Protocol).

Напомню, что вышеназванные протоколы объединяет то, что они являются протоколами туннелирования канального уровня. Но определению защищенного канала соответствует лишь протокол PPTP, который обеспечивает туннелирование и шифрование данных. В свою очередь, протокол L2TP, по сути, является только протоколом туннелирования, а функции защиты в нем не поддерживаются. Защита обеспечивается при использовании данного протокола совместно с протоколом IPSec.

Важно отметить, что решения второго уровня модели OSI не приобретают высоко значения для взаимодействия ЛВС по причине недостаточной масштабируемости при необходимости иметь несколько туннелей с общими конечными точками.

VPN сетевого уровня

Реализация защиты сети на третьем уровне гарантирует как минимум достаточно высокую степень защиты всех сетевых приложений, причем без какой-либо модификации последних. VPN-продукты сетевого уровня выполняют инкапсуляцию IP в IP. Одними из широко известных протоколов на этом уровне является SKIP, который практически вытеснен другим протоколом VPN сетевого уровня – IPSec, предназначенным для аутентификации, туннелирования и шифрования IP-пакетов.

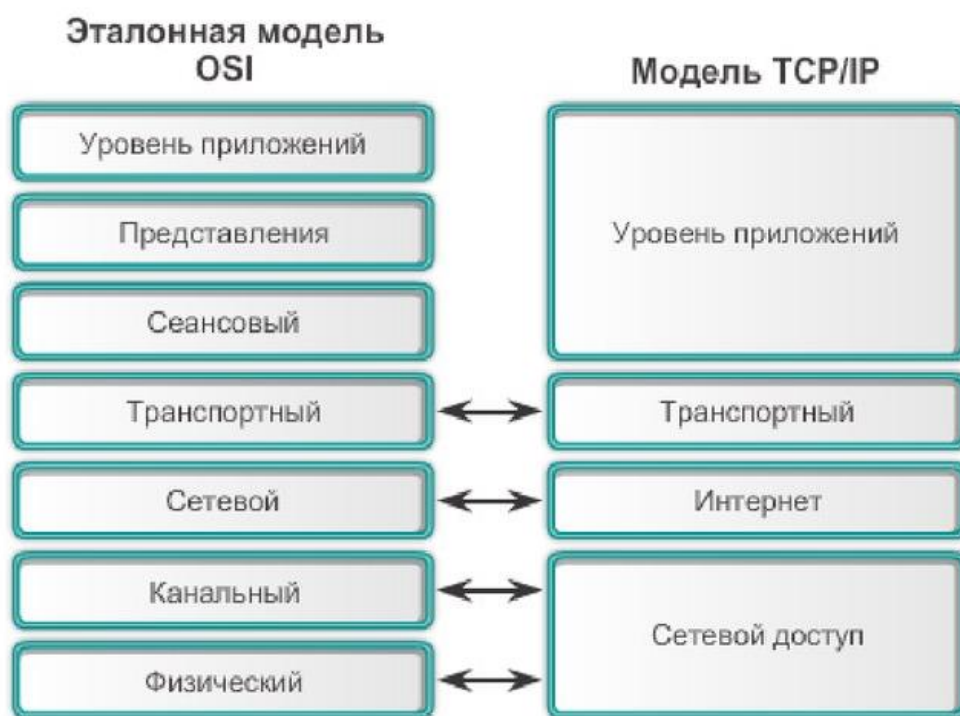
Протоколы IKE и IPSec обычно используются в паре: IKE обеспечивает обмен криптографическими ключами, IPSec - собственно защиту данных. IPSec считается наиболее перспективным протоколом. На базе IPSec в настоящее время разработан протокол сетевого уровня нового поколения - IP версии 6.

VPN верхнего слоя модели OSI.

Некоторые VPN используют другой подход под названием «посредники каналов» (circuit proxy). Этот метод функционирует над транспортным уровнем и ретранслирует трафик из защищенной сети в общедоступную сеть Internet для каждого сокета в отдельности. Напомню, что сокет IP идентифицируется комбинацией TCP-соединения и конкретного порта или заданным портом UDP. Получается, что VPN верхнего слоя характерны тем, что защищенные туннели строятся между каждым сокетом в отдельности, т.е. между двумя компьютерами может существовать несколько защищенных туннелей одновременно.

Мы помним, что стек протоколов TCP/IP не имеет сеансового и представительского уровней, однако ориентированные на сокеты операции часто называют операциями сеансового уровня.

Сравнение моделей OSI и TCP/IP



Таким образом, VPN верхнего слоя часто называют **VPN сеансового уровня**. Этот класс VPN осуществляет ретрансляцию трафика из защищаемой сети в Интернет и наоборот для каждого сокета (IP-адрес компьютера в совокупности с номером порта) в отдельности. При этом следует помнить, что механизм шифрования часто функционирует представительском уровне модели OSI, а инкапсуляция может производиться как транспортном, так и на сетевом уровнях.

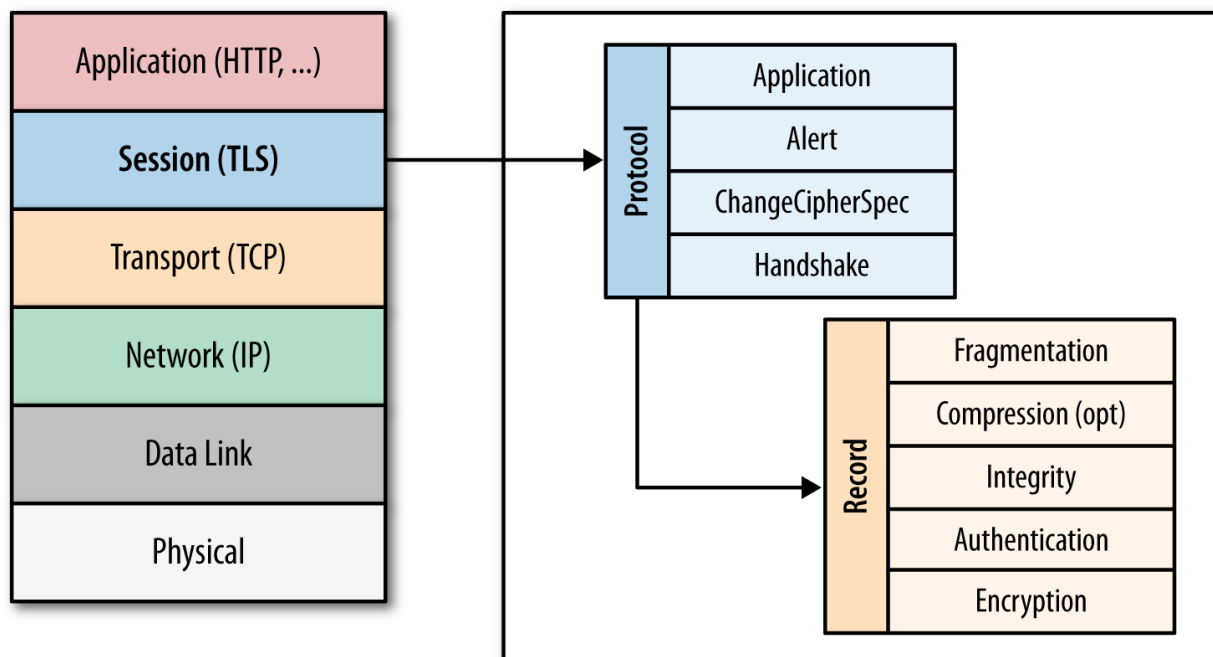
VPN сеансового уровня основаны на применении следующих протоколов:

- SOCKS;

- SSL/TLS - протоколы защиты транспортного уровня, используемые в паре с протоколом SOCKS.

- SSH.

К примеру, конкретное место ядра SSL/TLS в эталонной модели OSI представлено на рисунке.



Для создания защищенных туннелей на данных уровнях модели OSI используется протокол SSL/TLS или Secure Socket Layer/Transport Layer Security, реализующий шифрование и аутентификацию между транспортными уровнями приемника и передатчика.

По прошлым лекциям вы должны помнить, что TLS фактически заменил устаревший протокол SSL, но ввиду правопримественности обычно принято называть его современную версию связкой SSL/TLS.

Классификация VPN по способу технической реализации:

- VPN на основе сетевой операционной системы;
- VPN на основе межсетевых экранов;
- VPN на основе маршрутизаторов;
- VPN на основе программных решений;
- VPN на основе специализированных аппаратных средств со встроенными шифропроцессорами.

VPN на основе сетевой ОС

Реализация с помощью встроенных средств VPN в ОС.

Реализацию VPN на основе сетевой ОС можно рассмотреть на примере операционной системы Windows. Для создания VPN компания Microsoft предлагает протокол PPTP, интегрированный в сетевую операционную систему Windows.

VPN на основе маршрутизаторов

Данный способ построения VPN предполагает применение маршрутизаторов для создания защищенных каналов. Поскольку вся информация, исходящая из локальной сети, проходит через маршрутизатор, то вполне естественно возложить на него и задачи шифрования.

VPN на основе межсетевых экранов

Межсетевые экраны большинства производителей содержат функции туннелирования и шифрования данных. К программному обеспечению собственно межсетевого экрана добавляется модуль шифрования.

К недостаткам этого метода относятся высокая стоимость решения в пересчете на одно рабочее место и зависимость производительности от аппаратного обеспечения, на котором работает межсетевой экран. При использовании межсетевых экранов на базе ПК надо помнить, что подобный вариант подходит только для небольших сетей с ограниченным объемом передаваемой информации.

VPN на основе программного обеспечения

Для построения сетей VPN также применяются программные решения. При реализации подобных схем используется специализированное ПО, работающее на выделенном компьютере и в большинстве случаев выполняющее функции прокси-сервера. Компьютер с таким программным обеспечением может быть расположен за межсетевым экраном,

VPN на основе специализированных аппаратных средств со встроенными шифропроцессорами

Вариант построения VPN на специализированных аппаратных средствах может быть использован в сетях, требующих высокой производительности.

Раздел 3. Системы анализа защищенности компьютерных сетей

Тема 3.1. Сканеры защищенности компьютерных сетей

Сканеры уязвимостей.

Сканеры уязвимостей - это программные или аппаратные средства, служащие для осуществления диагностики и мониторинга сетевых компьютеров, позволяющее сканировать сети, компьютеры и приложения на предмет обнаружения возможных проблем в системе безопасности, оценивать и устранять уязвимости. Администраторы используют сканеры уязвимостей для оценки эффективности защиты компонентов их корпоративной сети. В результате анализа определяются уязвимые места системы, которые могут быть использованы злоумышленниками для осуществления несанкционированного доступа, и администратор принимает меры по их устранению. Таким образом, результатом работы сканера является достаточно подробная информация о корпоративной сети, включающая список сетевого оборудования, компьютеров, с запущенными на них службами, версиями сетевого программного обеспечения, уязвимостей присущих данному ПО, учетные записи пользователей системы.

Основной принцип их функционирования заключается в эмуляции действий потенциального злоумышленника по осуществлению сетевых атак.

Поиск уязвимостей путем имитации возможных атак является одним из наиболее эффективных способов анализа защищенности АС, который дополняет результаты анализа конфигурации по шаблонам, выполняемый локально с использованием шаблонов (списков проверки).

Современные сканеры способны обнаруживать сотни уязвимостей сетевых ресурсов, предоставляющих те или иные виды сетевых сервисов. Их предшественниками считаются сканеры телефонных номеров (war dialers), использовавшиеся с начала 80-х и не потерявшие актуальности по сей день. Первые сетевые сканеры представляли собой простейшие сценарии на языке Shell, сканировавшие различные TCP-порты. Сегодня они превратились в зрелые программные продукты, реализующие множество различных сценариев сканирования.

Современный сетевой сканер выполняет четыре основные задачи:

- Идентификацию доступных сетевых ресурсов и устройств;
- Идентификацию доступных сетевых сервисов, служб, установленного ПО и операционных систем;
- Идентификацию имеющихся уязвимостей сетевых сервисов;
- Формирование отчетов с детальным описанием проблемы и выдача рекомендаций по устранению уязвимостей.

В функциональность сетевого сканера не входит выдача рекомендаций по использованию найденных уязвимостей для реализации атак на сетевые ресурсы. Возможности сканера по анализу уязвимостей ограничены той информацией, которую могут предоставить ему доступные сетевые сервисы.

Механизмы работы сканеров безопасности

Принцип работы современного сканера заключается в моделировании действий злоумышленника, производящего анализ сети при помощи стандартных сетевых утилит. При этом используются известные уязвимости сетевых сервисов, сетевых протоколов и ОС для осуществления удаленных атак на системные ресурсы и осуществляется документирование удачных попыток.

Существует два основных механизма, при помощи которых сканер безопасности проверяет наличие уязвимости - сканирование (scan) и зондирование (probe)

Сканирование - механизм пассивного анализа, с помощью которого сканер пытается определить наличие уязвимости без фактического подтверждения ее наличия - по косвенным признакам. Этот метод является наиболее быстрым и простым для реализации. В терминах компании ISS данный метод получил название "логический вывод" (inference). Согласно компании Cisco этот процесс идентифицирует открытые порты, найденные на каждом сетевом устройстве, и собирает связанные с портами заголовки (banner), найденные при сканировании каждого порта. Каждый полученный заголовок сравнивается с таблицей правил определения сетевых устройств, операционных систем и потенциальных уязвимостей. На основе проведенного сравнения делается вывод о наличии или отсутствии уязвимости.

Зондирование - механизм активного анализа, который позволяет убедиться, присутствует или нет на анализируемом узле уязвимость. Зондирование выполняется путем имитации атаки, использующей проверяемую уязвимость. Этот метод более медленный, чем "сканирование", но почти всегда гораздо более точный, чем он. В терминах компании ISS данный метод получил название "подтверждение" (verification). Согласно компании Cisco этот процесс использует информацию, полученную в процессе сканирования ("логического вывода"), для детального анализа каждого сетевого устройства. Этот процесс также использует известные методы реализации атак для того, чтобы полностью подтвердить предполагаемые уязвимости и обнаружить другие уязвимости, которые не могут быть обнаружены пассивными методами, например подверженность атакам типа "отказ в обслуживании" ("denial of service").

В общем случае алгоритм работы сканеров следующий:

Проверка заголовков. Самый простой и быстрый способ на основе сканирования, однако имеющий ряд недостатков. Так, вывод о «провале» делается лишь по результатам анализа заголовков. К примеру, проверяя FTP-сервер, сканер узнает версию обеспечения и на основе этой информации сообщает о возможных уязвимостях. Естественно, специалисты по сетевой безопасности осведомлены о ненадежности этого метода, однако как первый шаг сканирования — это оптимальное решение, не приводящее к нарушению работы сети.

Активные зондирующие проверки (active probing check). Это сканирование, при котором не проверяется версия ПО, а сравнивается «цифровой слепок» фрагмента программы со «слепком» уязвимости. По тому же принципу действуют антивирусные программы, сравнивая ПО с имеющимися в базе сигнатурами вирусов. Тоже достаточно быстрый метод, хотя и медленнее первого, с большим коэффициентом надежности.

Имитация атак (exploit check). Это зондирование, которое эксплуатирует дефекты в программном обеспечении. Таким образом подается своеобразный импульс некоторым уязвимостям, которые не заметны до определенного момента. Эффективный метод, однако применить его можно не всегда. Так, вероятна ситуация, когда даже имитируемая атака просто отключит проверяемый узел сети.

Работу сканера уязвимостей можно разбить на 4 шага:

- Обнаружение активных IP-адресов, открытых портов, запущенной операционной системы и приложений.
- Составляется отчет о безопасности (необязательный шаг).
- Определяется уровень возможного вмешательства в операционную систему или приложения (может повлечь сбой).
- На заключительном этапе сканер может воспользоваться уязвимостью, вызвав сбой операционной системы или приложения, а также произвести автоматическое устранение уязвимостей. Не всегда данный этап

реализуется в сетевых сканерах безопасности, но часто встречается в сканерах системных.

Сканеры могут быть вредоносными или «дружественными». Последние обычно останавливаются в своих действиях на шаге 2 или 3, но никогда не доходят до шага 4.

Классы сканеров безопасности

1. Сканеры безопасности сетевых сервисов и протоколов (Nessus, XSpider, NMap).
2. Сканеры безопасности операционных систем (MBSA).
3. Сканеры безопасности приложений (HP WebInspect Agent, AppChecker, Appercut, PT Application Inspector).

Сканеры безопасности сетевых сервисов и протоколов

Они сканируют локальную или удаленную машину с целью обнаружения уязвимостей и начинают с получения предварительной информации о проверяемой системе: о разрешенных протоколах и открытых портах, версии ОС и т.д. Некоторые сканеры могут попытаться симитировать атаку на сетевой узел (реализацией моделей атак).

Сканеры безопасности операционных систем

Средства этого класса предназначены для проверки настроек ОС, влияющих на ее защищенность. К таким настройкам относятся: учетные записи пользователей, длина паролей и срок их действия, права пользователей на доступ к критичным системным файлам, уязвимые системные файлы и т.п. Данные сканеры могут проверить систему на наличие уязвимостей в прикладных программах и аппаратуре, уязвимостей, связанных с недостатками в конфигурировании системы (не согласующиеся с политиками безопасности).

Сканеры безопасности приложений

Несмотря на то, что особую популярность приобретают универсальные сканеры, качество проверок, определяемое их глубиной возможно обеспечить только специализированными сканерами, разработанными для конкретных прикладных программ, WEB-серверов и СУБД. Как правило их работа основана на специализированной методологии и использовании обширной базы знаний по уязвимостям конкретной прикладной системы.

Одним из главных требований к современным сетевым сканерам уязвимостей, помимо собственно безопасности, является наличие широкого функционала и поддержка различных операционных систем. Большинство популярных сканеров — кроссплатформенные (включая мобильные и виртуальные ОС).

Сканеры сети исследуют сразу несколько портов, что снижает время на проверку. И конечно, сканер должен проверить не только операционную систему, но и программное обеспечение, особое внимание уделяя популярным в хакерской среде продуктам (Adobe Flash Player, Outlook, различным браузерам).

К полезной функции сканеров нужно отнести и проверку раздробленной сети, что избавляет администратора от необходимости оценивать каждый узел в отдельности и несколько раз задавать параметры сканирования.

Современные сканеры просты в использовании, их работу можно настроить в соответствии с потребностями сети. Например, они позволяют задать перечень проверяемых устройств и типов уязвимостей, указать разрешенные для автоматического обновления приложения, установить периодичность проверки и предоставления отчетов. Получив подробный отчет об уязвимостях, одним нажатием кнопки можно задать их исправление.

Растущее количество угроз вынуждает разработчиков средств анализа защищенности постоянно усовершенствовать свои решения. Сейчас на рынке ИБ представлен широкий выбор сканеров безопасности от различных производителей, которые разнятся по своей эффективности.

Первый сканер уязвимостей сети появился более 20-ти лет назад, в 1992 году. Он носил имя SATAN и поначалу встретил огромное сопротивление специалистов, не понимающих его истинного предназначения. С тех пор технологии шагнули далеко вперед: сканеры, в отличие от «прародителя», стали кроссплатформенными, менее требовательными к системным ресурсам, более простыми в установке и использовании.

Сканеры безопасности:

- XSpider
- NMAP
- Nessus security scanner
- InsightVM
- Acunetix
- OpenVAS
- IBM Internet Scanner
- Shadow Security Scanner
- Retina Network Security Scanner

Компания Positive Technologies, выпустившая первую версию сканера XSpider в далеком 1998 году, как не сложно вычислить, работает на рынке более 20-ти лет и обладает одним из крупнейших исследовательских центров безопасности. Сканер анализа защищенности XSpider является признанным лидером среди средств сетевого аудита ИБ в России и Восточной Европе. Он находит широкое применение в подразделениях информационной безопасности государственных учреждений, в т.ч. ввиду сертификации Минобороны и ФСТЭК России. По заявлению разработчика, может выявить треть уязвимостей завтрашнего дня. Ключевой особенностью этого сканера является возможность обнаружения максимального количества «провалов» в сети еще до того, как их увидят хакеры.

Другие сканеры активно используются в коммерческих организациях по одной из очевидных причин – потому что существуют бесплатные версии

коммерческих программ, например, Nessus, функционал которого иногда даже интереснее чем у популярного в России XSpider'a.

Nessus является одним из популярных средств управления для поиска уязвимостями, используемым миллионами пользователей. Согласно статистике более 17% пользователей во всем мире предпочитают именно Nessus. Проект был запущен еще в 1998 году, а в 2003 разработчик Tenable Network Security сделал сетевой сканер безопасности коммерческим. Он охватывает большое количество типов активов: ОС, сетевые устройства, БД, Веб-серверы, Брандмауэры. Регулярно обновляемая база уязвимостей, простота в установке и использовании, высокий уровень точности – его преимущества перед конкурентами. А ключевой особенностью является использование плагинов. То есть любой тест на проникновение не зашивается наглухо внутрь программы, а оформляется в виде подключаемого плагина.

InsightVM – это система управления уязвимостями в реальном времени и аналитика конечных целей. С помощью InsightVM вы можете собирать, отслеживать и анализировать риски для новых и существующих сетей.

OpenVAS это одно из известных решений для сканирования и управления уязвимостями с открытым исходным кодом. OpenVAS – это платформа, которая включает в себя множество сервисов и инструментов, которая оптимально подходит для большинства задач тестирования уязвимостей в сети.

Сетевой сканер сети Acunetix для более чем 50 000 известных уязвимостей и анализа неправильной конфигурации. Acunetix использует сканер OpenVAS для обеспечения всесторонней проверки безопасности сети. Это онлайн-сканер, поэтому результаты сканирования доступны на панели управления, где вы можете развернуть отчет, риск, угрозы.

Symantec Security Check – бесплатный сканер одноименного производителя. Основные функции — обнаружение вирусов и троянов, интернет-червей, вредоносных программ, поиск уязвимостей в локальной сети. Это онлайн-продукт, состоящий из двух частей: Security Scan, которая проверяет безопасность системы, и Virus Detection, выполняющей полную проверку компьютера на вирусы. Устанавливается быстро и просто, работает через браузер. К сожалению, по итогам многих тестов уступает большинству конкурентов, хотя все свои основные функции выполняет. Согласно последним отзывам, этот сканер сети лучше использовать для дополнительной проверки.

Сканер Retina от BeyondTrust помогает находить уязвимости в сети, базе данных, в Интернете, в виртуальных и в инфраструктурных средах. Сканер Retina способен обнаруживать локальные и удаленные сетевые ресурсы. Работает, как и многие сканеры безопасности, без установки специальных агентов на хостах.

Методика использования сканеров

1. Инициализация (сбор предварительной информации и определение критичных зон)

2. Сбор информации (инвентаризация ресурсов)
3. Выявление уязвимостей (хостов и серверов; сетевых сервисов)
4. Сканирование сервисов баз данных.

Стадия инициализации

Получение

☐ Сетевой топологии со всеми подсетями, соединением с интернет, соединением с другими сетями.

☐ IP-адреса и маски

☐ Информация о шлюзах (IP addresses, accepted traffic).

☐ Информация о межсетевых экранах и IDS.

☐ Информация о критичных серверах

Инвентаризация сети (сбор информации)

☐ Идентификация хостов, серверов, сетевых принтеров, телекоммуникационных компонентов.

Выявление уязвимостей хостов и серверов

☐ Установка последних обновлений

☐ Проверка парольной политики

☐ Проверка на работу ненужных сервисов

☐ Проверка гостевых учетных записей

☐ Проверка на наличие расшаренных ресурсов

Выявление уязвимостей сетевых сервисов

☐ Конфигурация ОС

☐ Конфигурация приложений

☐ Проверка запущенных сервисов

☐ Возможность реализации DoS

☐ Проверка конфигурации WEB-браузеров и WEB-серверов

☐ Проверка на перебор паролей

Сканирование сервисов баз данных

☐ Резервное копирование

☐ Истекшие пароли

☐ Гостевые учетные записи

☐ Последние обновления

☐ Парольные атаки

Основным фактором, определяющим защищенность АС от угроз безопасности, является наличие в АС защиты от уязвимостей. Уязвимости могут быть обусловлены как ошибками в конфигурации компонентов АС, так и другими причинами, в число которых входят ошибки и программные закладки в коде ПО, отсутствие механизмов безопасности, их неправильное использование, либо их неадекватность существующим рискам, а также уязвимости, обусловленные человеческим фактором. Наличие уязвимостей в

системе защиты АС, в конечном счете, приводит к успешному осуществлению атак, использующих эти уязвимости.

Сетевые сканеры являются, пожалуй, наиболее доступными и широко используемыми средствами анализа защищенности. Сканер является необходимым инструментом в арсенале любого администратора либо аудитора безопасности АС.

В настоящее время существует большое количество как коммерческих, так и свободно распространяемых сканеров, как универсальных, так и специализированных, предназначенных для выявления только определенного класса уязвимостей. Но не забывайте соизмерять функционал продукта с потребностями своей организации и ее материальными возможностями. Основное, что нужно в работе сканера, — широкий охват сети проверяемых устройств и узлов, простота в использовании и точность настроек, возможность автоматической работы, которая не будет отвлекать специалистов от повседневных задач.