

Содержание

Введение.....	3
1. Проблема сетевых атак	6
1.1. Уязвимости	6
1.1.1 Базы уязвимостей.....	7
1.1.2 Классификация уязвимостей по ГОСТ Р 56546-2015	8
1.1.3 Уязвимости приводящие к проникновению на узел	8
1.2. Эксплойты.....	8
1.2.1 Базы эксплойтов	9
1.3. Сценарий сетевой атаки	10
1.3.1 Сбор информации	11
2. Оценка защищенности и выбор мер защиты в современных компьютерных сетях	13
2.1. Нормативная база задачи оценки защищенности сетевой инфраструктуры	13
2.1.1 Стандарты в области оценки безопасности компонентов сетевой инфраструктуры	13
2.1.2 Методики оценки защищенности компьютерной сети	13
2.1.1 Качественные методики оценки защищенности	13
2.1.2 Количественные методики оценки защищенности	17
2.1.3 Смешанные методики оценки защищенности	20
2.2. Показатели защищенности и выбора контрмер и способы их вычисления.....	25
2.2.1 Базовые показатели.....	25
2.2.2 Концепция графов атак	26
2.2.3 Показатели на основе графов атак	29
2.2.4 Методики выбора защитных мер на основе графов атак	31
2.3. Способы построения графов атак	35
2.3.1 Ручное построение графов атак	35
2.3.2 Автоматизированное построение графов атак	35
3. Метод оценки защищенности и выбора защитных мер на основе максимизации параметра уязвимости сети	35
3.1. Показатели защищенности узлов сетевой инфраструктуры	35

3.2. Метод оценки защищенности сетевой инфраструктуры	35
3.3. Алгоритм вычисления показателя защищенности сетевой инфраструктуры	35
3.4. Метод выбора защитных мер	35
4. Реализация системы оценки защищенности и выбора защитных мер	35
4.1. Архитектура системы оценки защищенности и выбора защитных мер....	35
5. Тестирование эффективности разработанного метода.....	35
5.1. Анализ существующих аналогов	36
6. Разработка системы автоматизированного построения и анализа графа потенциальных атак.....	37
7. Источники информации.....	38
8. Краткое описание разрабатываемой системы	39

ВВЕДЕНИЕ

Внутренняя сетевая инфраструктура практически любой организации представляет собой сложную структуру, состоящую из множества различных сервисов, направленных на поддержание функционирования компании. Эта структура очень динамична: добавляются новые сервисы, меняются конфигурации существующих, создаются новые связи между сервисами. В процессе роста (жизни) системы процесс обеспечения её информационной безопасности и защита критически важных объектов становятся нетривиальной задачей (все более сложной задачей).

Причиной нарушения информационной безопасности чаще всего становятся:

- уязвимости в операционных системах;
- уязвимости приложений, осуществляющих сетевое взаимодействие с пользователем или друг с другом;
- неправильная конфигурация программного обеспечения;
- ошибки контроля доступа

Используя имеющиеся уязвимости и недостатки системы внешние и внутренние нарушители проводят сетевые атаки, приводящие к компрометации различных узлов и реализации различных угроз информационной безопасности сети.

Для выявления недостатков компонентов системы, а также поиска уязвимостей и потенциальных векторов атак на информационные ресурсы, проводится анализ защищенности сети. Одним из наиболее эффективных методов анализа является тестирование на проникновение, в ходе которого осуществляется моделирование атак реальных злоумышленников. Такой подход позволяет в полной мере провести оценку защищенности сетевой инфраструктуры, оценить существующие и предложить новые способы защиты.

При этом все возрастающая сложность компьютерных систем: количество узлов в сети, множество различных версий сервисов и потенциальных уязвимостей и наличие средств защиты обуславливает необходимость в разработке автоматизированных систем анализа защищенности сети. Такие системы должны не только проводить анализ сети на наличие уязвимостей, построение трасс атак и их **реализацию** (моделирование), но также предлагать оптимальные пути их исправления.

В больших компаниях существует практика закрытия только тех уязвимостей, что приводят к проникновению в сеть и компрометации наиболее критических узлов. Из-за сложности сетевой инфраструктуры часто закрывают глаза на остальные цепочки уязвимостей, которые все ещё остаются в локальной сети организации после проведения анализа защищенности, что приводит к повторной компрометации сети при нахождении новой точки входа.

Необходимо не только защитить сеть от проникновения извне, но и обеспечить должный уровень защищенности внутренней сети. **Так, по данным Positive Technologies за 2019 год [https://www.ptsecurity.com/ru-ru/about/news/pentestery-preodoleli-setevoy-perimetr-92-procenta-kompanij/] при проведении внешнего тестирования на проникновение экспертам удалось преодолеть сетевой периметр 92% организаций, тогда как от лица внутреннего нарушителя был получен полный контроль над инфраструктурой во всех исследуемых системах.**

Для оценки уровня защищенности системы в данной работе предлагается использовать подход, основанный на анализе графа потенциальных атак с **составлением** метрик защищенности узлов и сети в целом для определения наиболее эффективных защитных мер, **которые приведут к максимальному увеличению уровня защищенности сети с наименьшими усилиями.** Для построения графа атак предлагается использовать различные базы уязвимостей и средства автоматизации этапов сканирования узлов и эксплуатации уязвимостей.

Добавлено примечание ([z1]): Может впахнуть сюда итерированность метода. То есть получение максимального профита за 1 итерацию – закрытие 1 уязвимости.

Цель данной работы – разработать автоматизированное средство оценки уровня защищенности сетевой инфраструктуры.

Цель данной работы – разработать автоматизированное средство выработки рекомендаций по повышению уровня защищенности сетевой инфраструктуры.

Для достижения поставленной цели определены следующие задачи:

- Описать основные классы уязвимостей, приводящие к компрометации узлов и методы их поиска; / Описать концепцию тестирования на проникновение и основные методы его автоматизации;
- Проанализировать применимость графов атак в задачах оценки защищенности сети;
- Разработать метод поиска оптимальных защитных мер на основе анализа графа атак;
- Разработать автоматизированную систему построения графа атак и анализа защищенности сети;
- Провести экспериментальную оценку эффективности разработанного метода.

1. ПРОБЛЕМА СЕТЕВЫХ АТАК

Под сетевой атакой можно понимать любой процесс, нацеленный на компрометацию безопасности сети. Результатом выполнения сетевой атаки может стать: захват контроля над удаленной/локальной вычислительной системой, либо её дестабилизация, повышение прав в системе, получение данных пользователей, пользующихся данной вычислительной системой.

Видов сетевых атак огромное множество и часть из них в состоянии реализовать человек, не владеющий большим запасом знаний в области информационной безопасности (далее ИБ).

(ТУТ КАКАЯ-НИБУДЬ СТАТИСТИКА ПО СЕТЕВЫМ АТАКАМ)

1.1. Уязвимости

В области информационной безопасности уязвимостью считается любой недостаток системы, используя который злоумышленник может провести атаку на систему. Уязвимость может быть результатом ошибок, допущенных на этапе программирования или проектирования отдельной программы, протокола или запроса, слабых паролей или ненадежных политик доступа.

Наиболее частой причиной возникновения уязвимостей становятся:

- ошибки проектирования, разработки программного продукта, протокола или запроса;
- слабые пароли;
- намеренно оставленные лазейки;
- неправильные настройки оборудования;
- отсутствие надежных политик доступа;
- несанкционированные неумышленные действия пользователей;

В наиболее широком смысле уязвимости можно разделить на две группы: известные и 0-day эксплойты. Известные уязвимости хорошо задокументированы исследователями, а соответствующие программные продукты имеют патчи, устраняющие возможность эксплуатации данных уязвимостей.

Под угрозой [<https://bdu.fstec.ru/ubi/terms>] в ИБ понимается любая потенциальная возможность тем или иным образом нарушить информационную безопасность. **Попытка реализации такой угрозы посредством уязвимости и называется атакой.**

В начале 90-х годов стало очевидно, что для хранения всего объема записей о найденных уязвимостях необходимо провести их классификацию и систематизацию. Актуальность данной задачи обуславливалась появлением большого числа нового программного обеспечения: операционных систем, программ, платформ разработки; увеличением количества версий программных продуктов, а также частотой нахождения уязвимостей.

Каждой обнаруженной уязвимости требовалось присвоить некий идентификатор и дать ей краткое описание. Также важно было определить критичность данной уязвимости, например, по таким критериям как простота эксплуатации и последствия эксплуатации. Данная информации впоследствии могла быть дополнена рекомендациями по устранению уязвимости, а также информацией об уязвимых версиях продукта.

В наше время данные об уязвимости различного программного обеспечения вовсю используются злоумышленниками для совершения атак.

1.1.1 Базы уязвимостей

На текущий момент существует несколько широко распространенных баз уязвимостей[<https://safe-surf.ru/specialists/article/5228/607311/>]:

1. **Реестр уязвимостей БДУ ФСТЭК России;**
2. **MITRE CVE и база данных NVD;**
3. **OSVDB;**
4. **Secunia Advisory and Vulnerability Database;**
5. **VND от CERT/CC;**
6. **Exploit Database;**

Добавлено примечание ([z2]): Расширить описанием каждой из баз? Например потом можно будет сделать ссылку на CVE, как на основу в metasploit и т.п.

1.1.2 Классификация уязвимостей по ГОСТ Р 56546-2015

<https://files.stroyinf.ru/Data2/1/4293759/4293759791.pdf>

1.1.3 Уязвимости приводящие к проникновению на узел

Для проникновения во внутреннюю сеть организации, злоумышленнику необходимо найти узел данной сети, доступный из внешней сети, после чего он должен получить доступ к этому узлу и иметь возможность выполнить на нем произвольные команды. Уязвимости, способствующие проникновению на узел можно поделить на следующие типы [<https://xakep.ru/2017/04/10/hacking-attack-types/>]:

1. недостатки управления учетными записями и паролями;
2. уязвимости веб-приложений;
3. недостатки фильтрации трафика;
4. недостатки управления уязвимостями и обновлениями;
5. плохая осведомленность пользователей в вопросах информационной безопасности;
6. недостатки конфигурации и разграничения доступа;

Любая уязвимость каждого из этих типов способствует проникновению на удаленное устройство. Иногда для проникновения используются комбинации данных методов, однако это лишь повышает сложность атаки, но не увеличивает вероятность проникновения.

1.2. Эксплойты

Эксплойт [<https://www.gizmosphere.org/network-security-vulnerabilities-vs-exploits/>] – скрипт или программа, разработанная исключительно для эксплуатации определенной уязвимости. В процессе проведения сетевой атаки могут задействоваться одновременно несколько эксплойтов для достижения поставленных целей. При этом необходимо успешное выполнение каждого из эксплойтов. Закрытие любой из уязвимостей подобной цепочки приведет к несостоятельности исходной атаки.

Добавлено примечание ([z3]): Сюда же входят существующие эксплойты

Эксплойтом может также являться обычный текст или словестное описание того, как проэксплуатировать уязвимость. Чаще всего для написания эксплойтов используются следующие языки программирования: Python, Ruby, PHP, Perl, HTML, Javascript, C/C++.

В общем случае определяют 2 вида эксплойтов:

- удаленный – эксплуатирует уязвимость удаленной системы без предварительного доступа к ней;

- локальный – эксплуатирует уязвимость локальной системы. Требуется наличие предварительного доступа к целевой системе. Чаще всего используется для повышения привилегий.

Так как эксплойты разрабатываются для выполнения различных действий на атакуемой системе, то они могут классифицироваться по объекту назначения [<https://www.anti-malware.ru/threats/exploits>]:

- для браузеров и дополнений к ним;
- для операционных систем;
- для офисных программ, проигрывателей и другого прикладного программного обеспечения;
- для серверного программного обеспечения;
- для веб-сервисов, например, WordPress, Joomla, Drupal и др.
- для аппаратных компонентов.

Большинство существующих эксплойтов входят в ту или иную базу эксплойтов, которые в большинстве случаев создаются для научных целей.

1.2.1 Базы эксплойтов

Наиболее обширные базы эксплойтов на данный момент [<https://null-byte.wonderhowto.com/how-to/top-10-exploit-databases-for-finding-vulnerabilities-0189314/>]:

Добавлено примечание ([u4]): Объект назначения зависит от уязвимости => имеет ли смысл классифицировать эксплойты таким образом, если принадлежность классу целиком зависит от уязвимости?

1.3. Сценарий сетевой атаки

Реализация угрозы, как было описано выше, может быть направлена на нарушение целостности, доступности и конфиденциальности информации. Процесс реализации угрозы в общем случае состоит из 4 этапов [<https://cyberleninka.ru/article/v/klassifikatsiya-ugroz-i-uyazvimostey-informatsionnoy-bezopasnosti-v-korporativnyh-sistemah>]:

- Сбор информации о цели;
- Проникновение в целевую систему;
- Осуществление несанкционированных действий;
- Скрытие следов несанкционированного доступа.

На каждом из этих этапов может быть задействовано неограниченное число дополнительных операций, представленных на рисунке X.

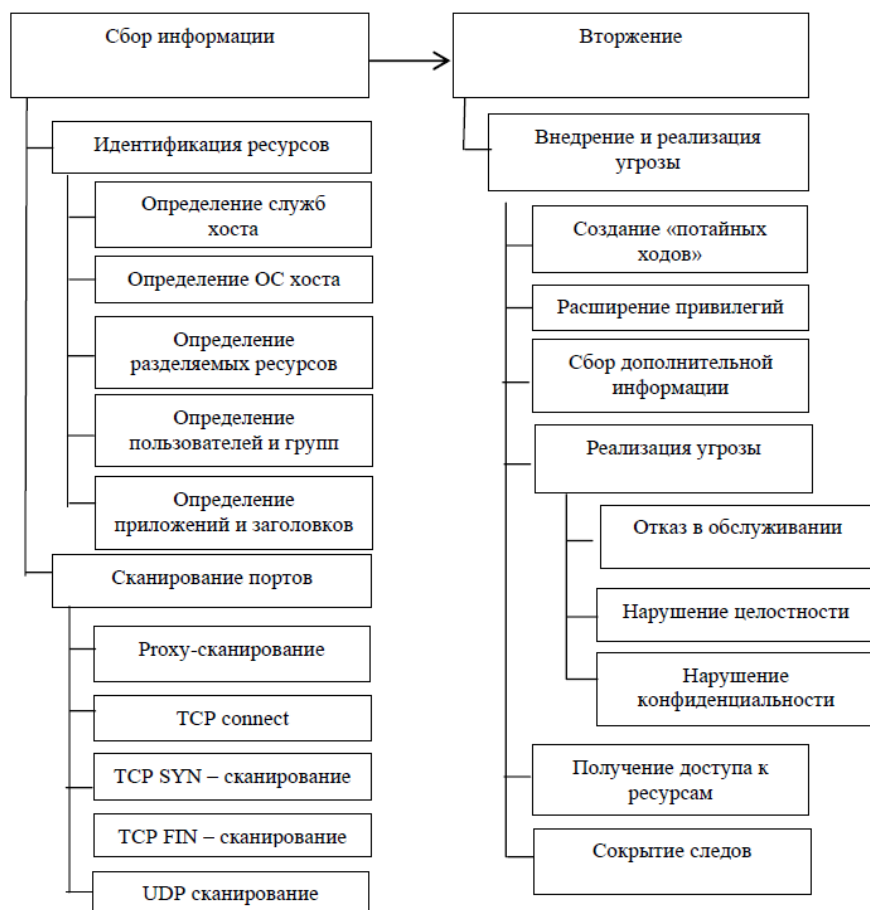


Рисунок X – Процесс реализации угрозы

1.3.1 Сбор информации

2. ОЦЕНКА ЗАЩИЩЕННОСТИ И ВЫБОР МЕР ЗАЩИТЫ В СОВРЕМЕННЫХ КОМПЬЮТЕРНЫХ СЕТЯХ

Задача оценки защищенности и выбора защитных мер в компьютерных сетях является одной из основных задач информационной безопасности. Согласно

2.1. Нормативная база задачи оценки защищенности сетевой инфраструктуры

2.1.1 *Стандарты в области оценки безопасности компонентов сетевой инфраструктуры*

2.1. Методики оценки защищенности компьютерной сети

Добавлено примечание ([05]): Разбавить

Оценка защищенности компьютерной сети – это процесс выявления уязвимостей, угроз и рисков, связанных с активами организации и мер защиты, которые могут смягчить эти риски [<https://www.tcdi.com/criteria-for-selecting-an-information-security-risk-assessment-methodology-qualitative-quantitative-or-mixed/>]. Существует два базовых подхода к оценке защищенности: качественный и количественный, а также смешанный подход, представляющий собой совокупность обоих подходов.

2.1.1 *Качественные методики оценки защищенности*

Качественные методики позволяют идентифицировать уязвимости и угрозы, описать причины их возникновения, возможные последствия и применяемые защитные меры и на их основе ранжировать риски, однако такие методики не позволяют определить численную величину риска.

Методика оценки риска называется качественной, если в процессе её выполнения формируется качественная оценка уровня риска. И.А. Педерсен и Н.Е. Брюковецкая определили несколько этапов качественной оценки риска (рисунок X) [<https://cyberleninka.ru/article/n/metodologiya-otsenki-riskov-predpriyatiya/viewer>]

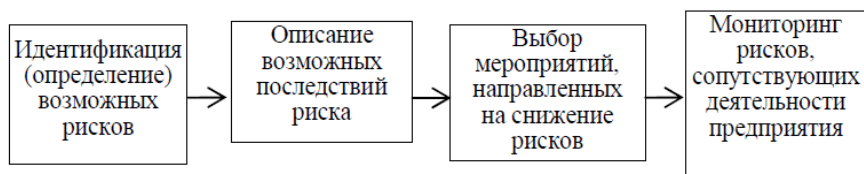


Рисунок X – Последовательность проведения качественного анализа рисков

Качественные методики используют опыт и суждения экспертов, а не математические формулы. Они также могут использовать опросы для определения уровня угрозы и ожидаемых рисков. Данные методики особенно эффективны, когда невозможно определить денежный эквивалент конкретного риска.

Качественные методики не требуют больших математических вычислений, но результаты, как правило, менее точны, чем полученные при количественной оценке.

Примеры качественных методик:

- COBRA [Visintine, V. Global information assurance certification paper];
- OCTAVE [Caralli, R. A. Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process];
- FRAP [Peltier, T. R. Information security risk analysis, Third Edition].

2.1.1.1 Методика COBRA

Методика COBRA разработана компанией Risk Associates [<https://www.securityauditor.net/>]. COBRA представляет собой процесс анализа рисков на основе экспертных систем, использующих обширную базу знаний по угрозам, уязвимостям и множеству опросников. Данная методика позволяет оценить соответствие оцениваемой системы стандарту ISO 17799 [ISO/IEC 17799:2005. Information technology. – Security techniques].

COBRA включает в себя 2 модуля: COBRA Policy Compliance Analyst и COBRA Data Protection Consultant. Первый позволяет определить, удовлетворяет ли оцениваемая система стандарту ISO 17799 и получить рекомендации. Второй содержит вопросы, позволяющие идентифицировать активы, угрозы, уязвимости и средства защиты.

Результат работы данной методики – отчёты, содержащие оценки рисков и рекомендации по их уменьшению, базирующиеся на общепринятых практиках.

2.1.1.2 Методика OCTAVE

Данная методика была создана для решения проблем информационной безопасности США. OCTAVE направлена на выявление, оценку и управление рисками информационной безопасности.

Эта методика помогает организации:

- определить наиболее важные для организации активы;
- выявить уязвимости и угрозы для этих активов (автоматизированное или ручное сканирование инфраструктуры);
- определить и оценить потенциальные последствия для организации в случае реализации угроз;
- инициировать действия по постоянному улучшению средств защиты для снижения рисков.

Данная методика основана на применении экспертных знаний.

2.1.1.3 Методика FRAP

Методика оценки рисков FRAP разработана Томасом Пелтиером. В методике обеспечение информационной безопасности предлагается рассматривать в рамках процесса управления рисками. Управление рисками в сфере информационной безопасности — процесс, позволяющий компаниям

найти баланс между затратами средств и сил на средства защиты и получаемым эффектом.

Основные этапы оценки рисков по FRAP:

1. Определение защищаемых активов производится с использованием опросных листов, изучения документации, использования инструментов автоматизированного анализа сетей;

2. Идентификация угроз. При составлении списка угроз могут использоваться разные подходы:

- заранее подготовленные экспертами перечни угроз, из которых выбираются актуальные для данной сети;
- анализ статистики происшествий в данной сети и в подобных ей — оценивается частота их возникновения;
- «мозговой штурм», проводимый сотрудниками компании.

3. Когда список угроз закончен, каждой из них сопоставляют вероятность возникновения. После чего оценивают ущерб, который может быть нанесен данной угрозой. Исходя из полученных значений, оценивается уровень угрозы.

При проведении анализа, как правило, принимают, что на начальном этапе отсутствуют средства и механизмы защиты. Таким образом оценивается уровень риска для незащищенной сети, что в последствии позволяет показать эффект от внедрения средств защиты информации.

Оценка производится для вероятности возникновения угрозы и ущерба от нее по следующим шкалам:

- Высокая – высокая вероятность того, что угроза реализуется в течение года;
- Средняя – угроза может быть реализована в течение следующего года;
- Низкая – низкая вероятность того, что угроза будет реализована в течение следующего года.

Ущерб активу определяется схожим образом (высокий/средний/низкий).

4. После определения вероятности возникновения ущерба и величины ущерба эксперты определяют средства, позволяющие снизить риски, ориентируясь на наиболее рентабельные. Данные о рисках и возможных защитных мерах документируются и передаются управляющему лицу.

5. Документирование. Результат работы по FRAP – полный набор документации по угрозам и рискам рассматриваемой системы, а также возможные средства минимизации уровней риска угроз.

2.1.2 Количественные методики оценки защищенности

Количественные методики описывают возможные риски в денежном или частотном эквиваленте. На основе полученных значений и стоимости реализации мер защиты риски сравниваются для принятия оптимальных мер защиты. При количественном анализе рисков выделяют несколько последовательных этапов (рисунок X) [<https://cyberleninka.ru/article/n/metodologiya-otsenki-riskov-predpriyatiya/viewer>].

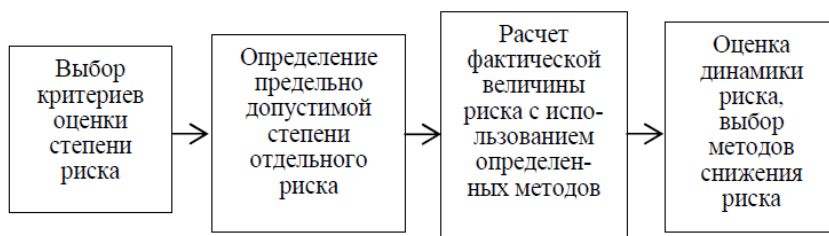


Рисунок X – Последовательность проведения количественного анализа рисков

К количественным методикам относят:

- RiskWatch;
- ГРИФ.

2.1.2.1 Методика RiskWatch

Методика RiskWatch [<https://riskwatch.com/2019/07/31/risk-scoring-methodology/>], разработанная при участии NIST, использует значение ожидаемых годовых потерь для оперирования значениями риска (ALE) и значение возврата инвестиций от реализации методов защиты (ROI). На основе данных параметров методика позволяет определить точное соотношение потерь от реализации угроз безопасности и стоимости реализации средств защиты.

В основе методики лежит обширная база знаний, содержащая информацию по активам, угрозам, уязвимостям, видам ущерба, защитным мерам а также опросным листам для оценки факторов риска.

Процесс анализа рисков по методике RiskWatch осуществляется в несколько этапов с использованием одноименного программного обеспечения:

1. Определение параметров исследования. На данном этапе задаются область оценки, категории ущерба, категории активов, рассматриваемые угрозы, уязвимости и применяемые контрмеры.
2. Ввод данных. В ходе данного этапа в систему оценки заносятся данные о ценности активов, вероятности угроз, степени уязвимости, частоте угроз и стоимости контрмер. Ценности активов, определяемой величиной ущерба в результате нарушения конфиденциальности, целостности и доступности активов, соответствуют определенные оценочные денежные величины.

Ожидаемая частота реализации угроз определяется в терминах среднегодовой оценочной частоты угрозы (AFE). База знаний RiskWatch определяет для каждой угрозы стандартную оценочную частоту (SAFE). Для вычисления рисков используется локальная оценочная частота угрозы (LAFE), которая определяется оператором, используя в качестве базового значения SAFE.

Для каждой контрмеры задается её стоимость, которая определяется стоимостью внедрения и сопровождения. Также учитывается, на какой стадии находится реализация контрмеры, продолжительность ее жизненного цикла и

насколько эта контрмера уменьшает оценочную частоту реализации угрозы (LAFE).

На данном этапе также осуществляется формирование опросных листов, используемых для получения информации об активах.

3. Оценка рисков. Для проведения оценки рисков производится связывание между собой данных о ценности активов, частоте угроз и степени уязвимости. В результате производится расчет ALE для каждой комбинации актив-угроза-уязвимость. На данном этапе также производится расчет ROI для контрмер и поиск сценария, при котором будет достигнуто максимальное уменьшение рисков при оптимальной комбинации контрмер.

4. Формирование отчётов. На данном этапе подготавливаются отчёты по проведенному анализу рисков. Генерируемые в процессе формирования отчётов графики иллюстрируют статистическую информацию о распределении уязвимостей информационной системы по областям контроля.

Рекомендуемые контрмеры могут быть отсортированы в порядке убывания значения ROI.

2.1.2.2 Методика ГРИФ

Система ГРИФ была разработана компанией Digital Security[<https://dsec.ru/press-release/digital-security-office-2005-kompleksnoe-reshenie-dlya-upravleniya-informatsionnoj-bezopasnostyu/>] в 2005 году, как система анализа и управления информационными рисками. Система ГРИФ может проводить несколько видов анализа: на основе модели информационных потоков, требующей от пользователя информации обо всех критически важных ресурсах, о пользователях, которые имеют к ним доступ и их правах доступа, и на основе модели угроз и уязвимостей. Для наших целей наиболее интересной представляется модель анализа угроз и уязвимостей, которая позволяет определить уязвимость каждого ресурса, содержащего информацию, подлежащую защите, и угрозы, которые могут быть реализованы, используя данные уязвимости. В результате работы данной системы создаётся полная

картина того, какие слабые места есть в системе и какой ущерб может быть ей нанесен.

Работа системы на основе модели уязвимостей и угроз выполняется в несколько этапов:

1. Пользователь определяет отделы, активы, угрозы и уязвимости информационной системы. Система ГРИФ содержит большую базу знаний об угрозах и уязвимостях, полнота которой достигается благодаря специально разработанной классификации угроз DSECCT, воплотившей в себе многолетний опыт в области ИБ (около 200 уязвимостей и 100 угроз);

2. Для вышеописанных ресурсов системы пользователь определяет связи (актив-отдел, уязвимость-угроза, актив-угроза);

3. Система ГРИФ анализирует полученные данные и генерирует отчёт, содержащий значение риска для каждого актива;

4. На основе полученных отчётов производится поиск оптимальных контрмер для снижения уровня остаточного риска;

5. На выходе система генерирует отчёт об уровне риска каждого актива, результат анализа причин наличия рисков и оценку эффективности различных мер защиты.

2.1.3 Смешанные методики оценки защищенности

На практике обособленное применение количественных или качественных методик. Смешанные методики сопоставляют качественному уровню определенный количественный диапазон [Астахов, А. Искусство управления информационными рисками].

К таким методикам относятся:

- CRAMM;
- Методика оценки рисков на основе графов атак.

2.1.3.1 Методика CRAMM

Методика CRAMM [<https://managementmania.com/en/cramm-ccta-risk-analysis-and-management-method>] объединяет в себе количественные и качественные методы для проведения комплексной оценки рисков. Разработана Central Computer and Telecommunications Agency (CCTA) в Великобритании. Данная методика включает в себя следующие этапы:

- Выявление и оценка активов. В качестве активов могут выступать: программное обеспечение, аппаратные ресурсы, либо данные;
- Выявление угроз и уязвимостей;
- Оценка рисков;
- Выбор защитных мер и их приоритезация.

Для оценки потенциального ущерба используется шкала от 1 до 10. Выявление угроз и уязвимостей осуществляется на основе экспертных знаний с помощью опросов. Значение уровня уязвимостей определяется как: высокий, средний и низкий, уровень угроз: очень высокий, высокий, средний, низкий и очень низкий. Риск оценивается в зависимости от годовых потерь по шкале от 1 до 7. Потери зависят от стоимости активов, уровня угрозы и уязвимости. Полученные уровни рисков используются для генерации вариантов защитных мер.

Таким образом, CRAMM комплексно охватывает все этапы управления рисками, начиная от фактического анализа рисков и заканчивая предложением контрмер, включая генерацию выходных данных для документации по безопасности (планирование действий в чрезвычайных ситуациях и обеспечение непрерывности). CRAMM одновременно поддерживается одноименным приложением, которое помогает в сборе данных, а также в расчете и обработке отчета по управлению рисками.

CRAMM также помогает доказать эффективность затрат на управление рисками, безопасность и планирование действий в чрезвычайных ситуациях. Он содержит обширную библиотеку контрмер безопасности. Также

применение методики CRAMM позволяет организациям подготовиться к сертификации в соответствии с ISO 27001.

2.1.3.2 Методика оценки рисков на основе графов атак

В работе [\[https://cyberleninka.ru/article/n/metodiki-i-programmnyy-komponent-otsenki-riskov-na-osnove-grafov-atak-dlya-sistem-upravleniya-informatsiey-i-sobytiyami-bezopasnosti/viewer\]](https://cyberleninka.ru/article/n/metodiki-i-programmnyy-komponent-otsenki-riskov-na-osnove-grafov-atak-dlya-sistem-upravleniya-informatsiey-i-sobytiyami-bezopasnosti/viewer) была предложена методика оценки рисков на основе построения и анализа графа атак для оценки защищенности сети. Данная методика использует модифицированную формулу расчета CVSS для определения уровня риска. Вместо исходного параметр CVSS SecurityRequirements вводится показатель Criticality, определяющий ценность актива для организации, который вычисляется с учетом финансовой стоимости актива и зависимостей свойств безопасности активов. Значения варьируются от 0 до 100. При этом выделяют следующие диапазоны критичности:

- [0;0,01) – ничтожно малая;
- [0,01;0,1) – малая;
- [0,1;1) – значительная;
- [1;10) – повреждающая;
- [10;100) – серьезная;
- 100 – смертельная.

В таблице X приведено преобразование шкалы для использования в уравнении CVSS.

Таблица X – Преобразование шкалы оценок критичности

Критичность	Значение
[0;0,01)	0
[0,01;0,1)	0,5
[0,1;1)	1

[1:10)	1,2
[10:100)	1,4
100	1,51

Уравнение CVSS:

$$Risk = round_to_1_decimal(AdjustedBase)$$

В раскрытом виде:

$$Risk = round_to_1_decimal(((0,6 * AdjustedImpact) + (0,4 * Exploitability) - 1,5) * f(AdjustedImpact))$$

где *Exploitability* – возможность использования уязвимости,

$$f(AdjustedImpact) = \begin{cases} 0, & \text{если } AdjustedImpact = 0 \\ 1,176, & \text{если } AdjustedImpact \neq 0 \end{cases}$$

$$AdjustedImpact = \min(10, 10,41 * (1 - (1 - ConfImpact * ConfReq) * (1 - IntegImpact * IntegReq) * (1 - AvailImpact * AvailReq))),$$

где *ConfImpact*, *IntegImpact*, *AvailImpact* – влияние на конфиденциальность, целостность и доступность в результате эксплуатации уязвимости; *ConfReq*, *IntegReq*, *AvailReq* – требования безопасности, а в данном контексте - критичность актива.

Тогда уравнение принимает вид:

$$AdjustedImpact = \min(10, 10,41 * (1 - (1 - ConfImpact * Criticality(c)) * (1 - IntegImpact * Criticality(i)) * (1 - AvailImpact * Criticality(a))))$$

где функция *Criticality* – критичность актива по нарушению конфиденциальности, целостности и доступности.

Исходя из вышеописанных формул риск может принимать значение от 0 до 10. После определения риска каждой уязвимости актива проводится определение риска программного обеспечения, как максимального риска его уязвимостей. Затем проводится оценка риска актива схожим образом. Уровень риска системы определяется как максимальной оценкой риска всех активов

системы, как высокий/средний/низкий в соответствии с CVSS. Данная методика позволяет выделить наиболее незащищенные участки системы.

Данная методика наиболее близка к применению в реальных условиях тестирования защищенности сетевой инфраструктуры, так как оценку уязвимостей по CVSS может провести и сторонний наблюдатель в отличие от методик, использующих опросы и требующих указать критичность активов с точки зрения бизнес-процессов организации. Однако, данная методика не учитывает важность самих узлов, но лишь возможность их компрометации путем эксплуатации наиболее критичных уязвимостей. Так, например, два узла, имеющие одинаковые максимальные по критичности уязвимости получают одинаковое значение риска, однако на одном из них может находиться несколько жизненно важных для работы инфраструктуры сервисов, важность которых никак не будет учтена. Также данная методика не описывает меры, принимаемые для снижения рисков.

Таким образом, необходимо разработать методику оценки защищенности сетевой инфраструктуры и выбора контрмер, которая соответствует следующим требованиям:

- Активы организации представляют собой сетевые узлы;
- Стороннее лицо, проводящее оценку защищенности сетевой инфраструктуры, должно быть способно провести самостоятельную оценку ценности активов для организации;
- Ценность узлов должна определяться на основании общедоступной информации и определяться целочисленным значением;
- Ценность узла должна зависеть от работающих на нем сервисов;
- Выбор защитных мер должен осуществляться с целью минимизации риска всей инфраструктуры.

2.2. Показатели защищенности и выбора контрмер и способы их вычисления

Методики оценки защищенности позволяют определить понятие показателя защищенности – параметра, определяющего качественную или количественную оценку защищенности анализируемой сети.

2.2.1 Базовые показатели

Базовыми называются показатели защищенности, непосредственно характеризующие элементы конфигурации и безопасности анализируемой системы, такие как: сервисы, установленные на узлах; уязвимости; источники угроз; атакующие действия; защитные меры. Определение их значений не требует проведения дополнительных вычислений.

В частности, в различных работах встречаются следующие виды показателей защищенности:

- В работе [Жижелев А. В., Панфилов А. П., Язов Ю. К., Батищев Р. В. К оценке эффективности защиты информации в телекоммуникационных системах посредством нечетких множеств // Изв. вузов. Приборостроение. 2003. т. 46, № 7. С. 22 – 29] показатель защищенности определяется исходя из ущерба сетевой инфраструктуры, связанного с реализацией угроз;

- В работе [Kanoun, W. Automated reaction based on risk analysis and attackers skills in intrusion detection systems] рассматривается показатель на основе уровня навыков атакующего. Уровень навыков варьируется в зависимости от уровня атакующего и его знаний о системе, а уровень атакующего определяется на основе максимальной сложности атакующих действий;

- В

[[https://cdn2.hubspot.net/hubfs/91381/2019%20Collateral/White%20Papers/Controlling%20Cyber%20Security/6%20Key%20Risk%20Management%20Metrics%20for%](https://cdn2.hubspot.net/hubfs/91381/2019%20Collateral/White%20Papers/Controlling%20Cyber%20Security/6%20Key%20Risk%20Management%20Metrics%20for%20)

20Controlling%20Cyber%20Security%20.pdf] используется целый ряд показателей защищенности:

- Суммарное количество обнаруженных уязвимостей;
- Актив с наибольшим текущим уровнем риска организации;
- Процент персонала, прошедшего обучение соблюдению политик и процедур информационной безопасности.

• В работе [The Center for Internet Security. The CIS Security Metrics [Текст]. – The Center for Internet Security, 2009. – 175 p.] выделяют такие показатели, как стоимость инцидентов, процент активов без уязвимостей, характеризующие количественные и стоимостные характеристики уязвимостей. В качестве показателей конфигурации системы используются метрики количества приложений и сервисов, процент критичных приложений и сервисов и другие.

Рассмотренные показатели защищенности позволяют охватить лишь часть критериев, требуемых для проведения полноценной проверки защищенности. Также большинство рассмотренных показателей требует вмешательства эксперта и дополнительных знаний о системе, что недоступно лицу, проводящему независимую оценку защищенности.

2.2.2 Концепция графов атак

Существует множество работ по применимости графов атак в задаче оценки защищенности сетевой инфраструктуры. Они применяются для анализа того, каким образом может развиваться атака внутри сети организации.

Граф атак можно представить как последовательность всех возможных действий злоумышленника для реализации угроз, так называемых трасс атак. Существует несколько общепринятых типов графов атак [Ingols, K. Practical attack graph generation for network defense [Текст] / K. Ingols, R. Lippmann, K. Piwowarski // Proceedings of 22nd Annual Conference on the Computer Security Applications (Miami Beach, FL, 2006). – IEEE, 2006. – P. 121 – 130.]:

Добавлено примечание ([z6]): Здесь можно привести ссылки на работы

- Полный граф атак – узлы такого графа представляют собой состояния, а ребра – уязвимости. Такие графы иллюстрируют каждую возможную трассу атак, которую может реализовать нарушитель. Они имеют сложность $O(n!)$, что негативно сказывается на их размере и, следовательно, на скорости вычислений. Часть полного графа атак представлена на рисунке X.

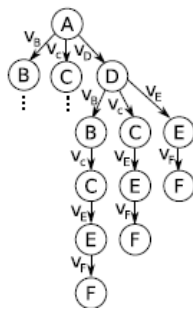


Рисунок X – Часть полного графа

- Граф предсказаний – узлы и ребра представляют собой те же сущности, что и в полном графе. Каждый узел попадает в граф предсказаний, если ни один его предок не использует ту же уязвимость для попадания в то же состояние. Данные графы не имеют недостатка полного графа по скорости построения и могут правильно прогнозировать влияние удаления любой из уязвимостей в сети. Как следствие такие графы строятся намного быстрее, чем полные, но тем не менее всё ещё содержат лишние структуры. Пример полного графа приведен на рисунке X.

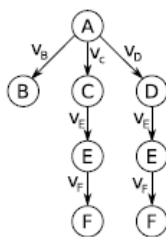


Рисунок X – Граф предсказаний

- МР-граф (граф со множеством предусловий) – содержит три типа узлов: уязвимости, состояния и предусловия. Для отображения связей с уже существующими узлами добавляются дополнительные циклические дуги. Данный граф строится быстро и может быть преобразован в полный граф или граф предсказаний. Пример такого графа приведен на рисунке X.

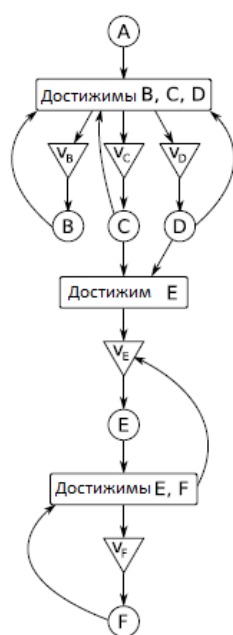


Рисунок X – МР-граф атак

При работе с графами атак возникает две основные проблемы:

- Обработка циклов;
- Значительное время обработки.

Для решения данных проблем применяются различные предположения, такие как: свойство монотонности графа и отказ от повторного посещения узлов; выделяются различные типы циклов графа и производятся расчеты вероятностей применения атакующих действий в зависимости от вида цикла.

2.2.3 Показатели на основе графов атак

Добавлено примечание ([z7]): Заменить перечисления схемой или таблицей?

В работе [<http://proceedings.spiiras.nw.ru/index.php/sp/article/view/1691>] рассматриваются релевантные исследования в области показателей защищенности, которые делятся на следующие группы:

- Топологические характеристики;
- Стоимостные характеристики;
- Характеристики атаки и реакции на атаку;
- Характеристики системы;
- Характеристики нарушителя;
- Показатели, применяемые при анализе уязвимостей нулевого дня.

Так, в работе [Mell P., Scarfone K., Romanosky S. A Complete Guide to the Common Vulnerability Scoring System Version 2.0, 2007] выделяются следующие топологические показатели:

- Критичность хоста – зависит от ущерба для бизнеса при потере данного хоста;
- Незащищенность – определяется простотой использования уязвимостей хоста и его достижимостью;
- Нисходящий риск – риск, полученный в результате прохождения через все хосты, атакуемые с данного хоста.

К стоимостным характеристикам относятся такие показатели, как:

- Ожидаемые годовые потери;
- Общий выигрыш – рассчитывается как разница между выигрышем от снижения годовых потерь и затратами на реализацию средств защиты;
- Возврат инвестиций от реагирования на атаку.

В качестве характеристики атаки рассматриваются:

- Потенциал атаки – определяет насколько близко нарушитель находится относительно своей цели;

- Уровень уверенности – определяет меру уверенности в том, что атака осуществляется в текущий момент;
- Показатель уверенности в компрометации – вероятностная величина, определяющая вероятность компрометации определенного узла;
- Ущерб от атаки – показатель, определяющийся на основе ценности активов;
- Эффективность реагирования;
- Выигрыш при реагировании;
- Побочные потери при реагировании.

В качестве характеристики нарушителя рассматривается уровень навыков нарушителя – помогает определить возможность проведения атаки нарушителем.

К характеристикам, используемым при анализе уязвимостей нулевого дня относятся:

- Вероятностная мера уязвимости – определяет насколько вероятно появление уязвимости нулевого дня определенной критичности за определенный срок;
- К-безопасность нулевого дня – определяет устойчивость сети к уязвимостям нулевого дня.

На основании вышеописанных уязвимостей авторы [Mell P., Scarfone K., Romanosky S. A Complete Guide to the Common Vulnerability Scoring System Version 2.0, 2007] проводят классификацию показателей защищенности, приведенную на рисунке X.



Рисунок X – Классификация показателей защищенности на основе графа атак

В данной работе за основу предлагается взять подход на основе оценки ниспадающего риска, рассчитанного на основе показателя ценности каждого узла, достижимого из текущего.

2.2.4 Методики выбора защитных мер на основе графов атак

В процессе оценки защищенности сетевой инфраструктуры, на этапе выбора защитных мер, необходимо провести приоритезацию и применение соответствующих защитных мер, способствующих понижению уровня риска системы.

Задача поиска оптимальных защитных мер на графе атак является нетривиальной, в виду проблем, описанных в п.2.2.2. В работе [\[https://deepai.org/publication/heuristic-approach-towards-countermeasure-selection-using-attack-graphs\]](https://deepai.org/publication/heuristic-approach-towards-countermeasure-selection-using-attack-graphs) описано несколько вариантов решения данной задачи.

2.2.5 Интегральные показатели

Интегральными называются показатели, непосредственно характеризующие безопасность всей сетевую инфраструктуры. Использование интегральных показателей – один из наиболее перспективных путей решения задачи оценки защищенности сети и выбора контрмер

[<https://cyberleninka.ru/article/n/osnovnye-teoreticheskie-polozeniya-strukturizatsii-chastnyh-pokazateley-zaschity-informatsii-s-tselyu-integralnoy-otsenki>].

Многие рассмотренных методики на финальной стадии выбора защитных мер используют именно интегральные показатели, потому что они позволяют определить такие характеристики системы как: уровень риска или поверхность атаки, который определяется на основе ресурсов, которые могут использоваться при проведении атаки.

2.2.5.1 Методика поддержки принятия решений, интегрированная в IDS

Одним из вариантов решения данной задачи является система поддержки принятия решений в выборе контрмер, которая реализуется как часть системы обнаружения вторжений (IDS) [<https://link.springer.com/article/10.1007%2F978-3-642-19974-4>]. Система содержит хранилище ранее проведенных атак и контрмер, представленных в LAMBDA – универсальном языке, используемом для описания атак в терминах условий, предшествующих эксплуатации (предусловий) и пост-эксплуатации (постусловий).

Контрмеры представлены в LAMBDA следующим образом: предварительное условие представляет собой состояние системы, которое требует применения контрмеры, а постусловие является состоянием системы после применения контрмеры (обычно отрицание предварительного условия). Авторы определили понятия корреляции и антикорреляции следующим образом: два состояния коррелируют, если постусловие одного поддерживает предварительные условия другого (т.е. эти экземпляры представляют собой этапы атаки в сценарии многоступенчатой атаки). Два состояния не коррелируют, если постусловие одного противоречит предварительным условиям второго (т.е. одно состояние представляет собой защитную меру, которая предотвращает появление второго состояния, представляющего атаку). Процесс рекомендаций по противодействию инициируется оповещениями IDS.

При получении предупреждения будущие шаги атаки определяются путем поиска корреляций в хранилище, а контрмеры - путем поиска антикорреляций.

Таким образом, для каждого узла графа-состояния строится набор антикорреляций, который не позволит совершить атаку. Метод хорош для противостояния известным атакам в режиме реального времени, но совершенно не подходит при статическом анализе графа с целью минимизации рисков.

2.2.5.2 Методики на основе теории игр

Данные методики характеризуются последовательностью состояний и действий атакующего и защитника (стратегии). Изменение состояния происходит при осуществлении действий в рамках определенной стратегии. Защищающая сторона стремится к выбору минимальной по стоимости стратегии, характеризующейся минимизацией затрат и времени, необходимых для реализации всех действий стратегии. На выходе данной методики – оптимальная стратегия, которую администратору необходимо применить для обеспечения защищенности системы.

Преимущество таких методик – учет временного аспекта.

2.2.5.3 Методики на основе количества достижимых узлов

В работе [Ingols, K. Practical attack graph generation for network defense] рассматривается методика выбора защитных мер на основе показателей, определяющих как много путей атаки будет заблокировано, введением конкретной защитной меры. Защитной мерой в данном случае выступает удаление узла-уязвимости МР-графа. Для упрощения поиска оптимального решения используется дополнительный тип узлов графа, все исходящие ребра которого направлены на узлы-уязвимости, которые требуют одинаковых условий для эксплуатации. Таким образом, остаётся определить, устранение путей к какому из таких узлов принесет наибольшую эффективность. Однако в работе не определяются конкретные показатели защищенности и методы их вычисления.

2.2.5.4 Методика на основе показателя процента компрометации сети (NCP)

Данная методика [Lippmann, R. P. Validating and restoring defense in depth using attack graphs [Текст] / R. P. Lippmann et al // Proceedings of MILCOM 2006 (Washington, DC).] опирается на показатель NCP, определяющий процент узлов, на которых атакующий может получить пользовательский или администраторский доступ. Показатель принимает значения в диапазоне от 0 до 100%. Выбор защитных мер осуществляется следующим образом:

- Производится обход графа в ширину, для каждого узла создаётся список входящих дуг. Дуги соответствуют уязвимостям, которые могут быть применены для компрометации узла;
 - Для полученного набора дуг подбираются защитные меры;
 - Узлы с одинаковым набором защитных мер объединяются в группы;
 - Для каждой защитной меры рассчитывается NCP (каждая защитная мера удаляет одну дугу графа);
 - Все защитные меры сортируются по увеличению значения NCP;
 - Выбирается мера с наименьшим NCP, то есть такая мера, разница предыдущего и текущего NCP которой максимальна.

Данная методика схожа с методикой, представленной в пункте 2.2.4.3, однако, работает куда медленней ввиду того, что достижимость пересчитывается для каждой отдельной контрмеры.

Большинство рассмотренных методик используют показатели, которые могут быть применены при оценке защищенности системы сторонним лицом, не имеющим доступа к описанию бизнес-процессов организации, реальной стоимости активов и их важности для организации, однако, они не решают проблему сложности графа и устранения циклов.

2.3. Способы построения графов атак

2.3.1 Ручное построение графов атак

2.3.2 Автоматизированное построение графов атак

3. МЕТОД ОЦЕНКИ ЗАЩИЩЕННОСТИ И ВЫБОРА ЗАЩИТНЫХ МЕР НА ОСНОВЕ МАКСИМИЗАЦИИ ПАРАМЕТРА УЯЗВИМОСТИ СЕТИ

3.1. Показатели защищенности узлов сетевой инфраструктуры

3.2. Метод оценки защищенности сетевой инфраструктуры

3.3. Алгоритм вычисления показателя защищенности сетевой инфраструктуры

3.4. Метод выбора защитных мер

4. РЕАЛИЗАЦИЯ СИСТЕМЫ ОЦЕНКИ ЗАЩИЩЕННОСТИ И ВЫБОРА ЗАЩИТНЫХ МЕР

4.1. Архитектура системы оценки защищенности и выбора защитных мер

5. ТЕСТИРОВАНИЕ ЭФФЕКТИВНОСТИ РАЗРАБОТАННОГО МЕТОДА

Графы атак являются ценнейшим инструментом, позволяющим проиллюстрировать, каким образом злоумышленник может получить доступ к целевой системе. Проведя анализ графа, специалисты по информационной безопасности могут сосредоточить свои усилия на устранении уязвимостей, предоставляющих злоумышленникам максимальный доступ.

Графы отражают все возможные пути атак, кроме того они могут иллюстрировать все состояния системы с переходами из состояния в состояние в соответствии с используемыми уязвимостями.

Существует несколько типов графов атак:

1.

Рассматриваемый граф, содержащий множество вершин-узлов и множество вершин-уязвимостей является двудольным графом, так как

множество вершин V разбито на два непересекающихся подмножества $V1$ и $V2$, причём всякое ребро E инцидентно вершине из $V1$ и вершине из $V2$ (то есть соединяет вершину из $V1$ с вершиной из $V2$).

5.1. Анализ существующих аналогов

В [Ingols, K. Practical attack graph generation for network defense] предлагается подход к генерации графа атак с множеством предусловий, позволяющий выявить наиболее критичные уязвимости системы (дающие нарушителю наибольший доступ в системе). Достижимость хоста для нарушителя определяется наличием администраторского или гостевого доступа. Авторы формируют рекомендации по реализации защитных мер на основе показателей, определяющих как много путей атаки будет заблокировано введением защитной меры. Недостаток: в работе не описаны конкретные показатели и не предложено методик их вычисления и методик определения уровня защищенности.

В итоге они анализируют, каким образом можно попасть на выбранный узел и, соответственно, какие нужно закрыть уязвимости, чтобы этого избежать. Причём узлы они группируют в мой аналог компонент сильной связности и называют его prerequisite node. Для выбора лучшего решения они "взвешивают" все рекомендации по числу закрытых для доступа хакера хостов (почему они тогда просто не обрывают начальные ребра от точки входа?).

Добавлено примечание ([u8]): Вариант похож на мой. Также используется поиск узких мест. Однако не учитывается целиком защищенность сети, лишь количество узлов, достижимых от точки входа. Если сравнить такой метод в рамках моей метрики – защищенность сети, то он проиграет.

6. РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗИРОВАННОГО ПОСТРОЕНИЯ И АНАЛИЗА ГРАФА ПОТЕНЦИАЛЬНЫХ АТАК

Для применения описанного выше метода анализа в первую очередь необходимо определить способ хранения графа. В данной работе предлагается использовать ...

Каждая точка принадлежит определенному подграфу графа потенциальных атак. Имея полный набор подграфов, достижимых из текущей точки легко посчитать уровень угрозы этой точки. Более того, уровень угроз данной точки должен совпасть с уровнем угрозы любого узла, входящего в тот же подграф, если точки находятся в одном цикле. Таким образом, анализ графа должен проходить в несколько этапов:

- Определение подграфов графа потенциальных атак, уровень подсчет уровня угрозы в которых может быть оптимизирован (должен быть критерий, согласно которому найденные подграфы нужно относить к «обособленным». Сюда точно будут входить компоненты связности (точно компоненты сильной связности), любые циклы)

- Подсчет угрозы каждого узла
- Подсчет текущего уровня угрозы сети
- Удаление одного из узлов-уязвимостей -> пересчет угроз узлов и сети

7. ИСТОЧНИКИ ИНФОРМАЦИИ

<https://bdu.fstec.ru/ubi/terms>

8. КРАТКОЕ ОПИСАНИЕ РАЗРАБАТЫВАЕМОЙ СИСТЕМЫ

Краткое изложение сути работы

В процессе проведения тестирования на проникновение производится построение графа атак.

$G = (V, A)$ – ориентированный граф, где V – непустое множество узлов, A – множество триплетов, называемых ребрами, вида (r_i, r_j, c_j) , где $r_i, r_j \in V, c_j \in CVE$ – идентификатор уязвимости, присутствующей на узле r_j . Направление дуги задаётся последовательностью следования узлов в записи.

$$t_{r_i} = \sum_{j \in V_i \subset V} Criticality(r_j) \quad (*)$$

Где t_i – угроза попадания злоумышленником на узел $r_i \in V$, r_j – узел графа, достижимый из r_i , $Criticality(r_j)$ – критичность захвата узла r_j , V_i – подмножество достижимых узлов из r_i .

Узел r_j достижим из $r_i \iff \exists$ хотя бы 1 путь из r_i в $r_j, r_i \neq r_j$.

$$Criticality(r_j) = device_type_coef(r_j) \sum service_cost(service_name_{i_j})$$

Где $device_type_coef(r_j)$ – коэффициент зависящий от типа устройства. Можно использовать базу npar классификации устройств (28 штук) и сопоставить каждому коэффициент по личным соображениям. Например, от 1 до 2. $service_cost()$ – критичность одного сервиса. Каждому сервису или типу сервисов можно сопоставить конкретную критичность компрометации, $service_name_{i_j} - j_{\text{ый}}$ сервис узла r_i .

Угроза компрометации системы на k -ом шаге поиска контрмер рассчитывается следующим образом:

$$T_k = \sum_{i=0}^{|V|} t_{r_i}$$

Далее необходимо найти такую уязвимость c_i , при удалении которой максимально снизится угроза компрометации системы, то есть:

$$(T_{k-1} - T_k/\{c_i\}) \rightarrow \max$$

Для подсчета t_{r_i} необходимо в первую очередь определить множество V_i . Для этого обойти граф (можно использовать любой алгоритм обхода, пока что используется обход в ширину (DFS)), начиная от узла r_i , и пометить все достижимые узлы. Сложность алгоритма обхода $O(|V| + |A|)$. Таким образом, сложность подсчета T_k в худшем случае равняется

$$O((|V| + |A|) * (|V|)) = O(V^2 + A)$$

Данный алгоритм можно оптимизировать следующим образом. Представим, что каждый узел исходного графа принадлежит подграфу одного из двух типов – компоненте сильной связности или N-арному дереву.

Ориентированный граф называется сильно связным, если любые два его узла сильно связны. Два узла s и t любого графа сильно связны, если существует ориентированный путь из s в t и ориентированный путь из t в s . Компонентами сильной связности орграфа называются его максимальные по включению сильно связные подграфы. Областью сильной связности называется множество узлов компоненты сильной связности.

Осуществим поиск областей сильной связности (обозначим их S_i , например, с помощью алгоритма Косарайю, который использует двойной обход в глубину, следовательно его сложность $O(2|V| + 2|A|)$).

Ориентированное N-арное дерево — ациклический орграф (ориентированный граф, не содержащий циклов), в котором только один узел имеет нулевую степень захода (в него не ведут дуги), а все остальные узлы имеют степень захода не больше N. Узел с нулевой степенью захода называется корнем дерева, узлы с нулевой степенью исхода (из которых не исходит ни одна дуга) называются концевыми узлами или листьями. Обозначим их NT_i . Значение N может отличаться от дерева к дереву.

При этом в N-арное дерево включаются только те узлы, входящие дуги которых направлены от узлов того же дерева. Исключение может составлять только корневой узел. (доп условие)

В первую очередь производится поиск всех компонент сильной связности и соответствующих им областей сильной связности. Проиллюстрируем данный шаг на примере. Считаем, что все узлы имеют по 1 уязвимости, вследствие чего дуги не подписаны (рисунок X).

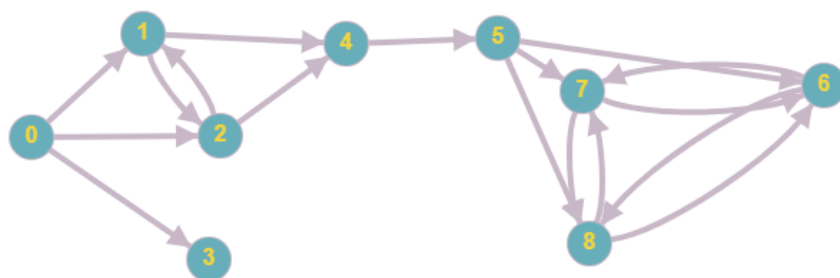


Рисунок X – Пример ориентированного графа

Обозначим на графе компоненты сильной связности и N-арные деревья (рисунок X). Компоненты сильной связности обозначены зеленым цветом, N-арные деревья – оранжевым.

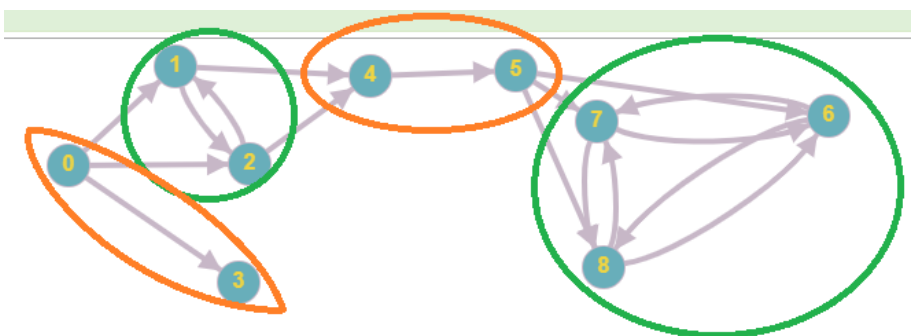


Рисунок X – Иллюстрация поиска подграфов двух типов

Рассмотрим некоторые свойства полученных подграфов:

- Из определения компоненты сильной связности следует, что в ней существуют маршруты из любого узла компоненты в любой другой. Отсюда:

$$\forall r_i, r_j \in S_k: t_{r_i} = t_{r_j}, r_i \neq r_j$$

S_k – k-ая область сильной связности

Следовательно, для каждой области сильной связности достаточно посчитать t_r единственный раз.

- Из определения компоненты сильной связности и N-арного дерева следует, что попав в любую из данных компонент гарантируется прохождение по всем узлам подграфа. При этом в N-арном дереве существует только один корневой узел, следовательно вход в данный подграф возможен исключительно через него, что позволит пройти по всем узлам данного дерева. Следовательно при достижении корневого узла такого дерева нет необходимости в пересчете критичности его узлов. Достаточно сделать это один раз:

$$S_{NT_i} = \sum_{r_j \in NT_i} Criticality(r_j)$$

S_{NT_i} – критичность захвата подграфа NT_i

- Ребра исходного графа распределяются на 3 категории:
 - Ребра, принадлежащие компонентам сильной связности;
 - Ребра, принадлежащие N-арным деревьям;
 - Ребра, связывающие различные подграфы.

В связи с этим при удалении ребер возможны 3 ситуации:

1. Если ребро находилось в компоненте сильной связности или N-арном дереве, необходимо осуществить перераспределение узлов данной компоненты по новым подграфам сильной связности и N-арным деревьям. Также необходимо осуществить новые расчёты параметров t_r и S_{NT}

2. Если ребро соединяло две компоненты, нет необходимости в пересчете параметров подграфов.

Исходя из вышеописанных свойств можно провести следующее преобразование: представим все компоненты сильной связности и N-арные деревья как узлы нового графа, которые включают в себя все входящие и все исходящие дуги исходного подграфа. При этом новое значение критичности узла высчитывается по формуле:

$$Criticality(r_i) = \sum_{r_j \in S_k} Criticality(r_j)$$

А новое значение угрозы попадания на узел r_i

$$t_{r_i} = |S_k| * t_{r_j}, \quad t_{r_j} \in S_k$$

Где S_k – k-ая область сильной связности

Полученный в результате граф, будет иметь следующий вид (рисунок X):

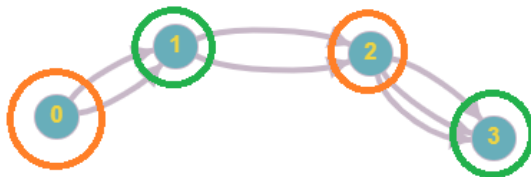


Рисунок X – Упрощенный вид графа

Таким образом, при определении значения угрозы компрометации системы T_k можно будет воспользоваться результатами расчетов, полученных в процессе вычисления угрозы компрометации системы T_{k-1} , что весьма критично, так рассматриваемая структура сетевой организации очень часто

выглядит так, что $|A| \gg |V|$. Следовательно перебор всех возможных решений займет продолжительное время.

(ОЦЕНИТЬ СЛОЖНОСТЬ)