Круглый стол

Математические модели в оценке информационной безопасности

(ПРИКЛАДНАЯ ДИСКРЕТНАЯ МАТЕМАТИКА)

21.05.2025

Актуальность

- Рост киберугроз: +125% целевых атак в 2023 году (по данным Verizon).
- Ущерб бизнесу: Средние потери от утечки данных — $4.45 млн (IBM).
- Регуляторные требования: ГОСТ Р 57580.1, NIST CSF 2.0, ФСТЭК №239.

- **Цель доклада**: Систематизировать математические модели ИБ, их применение и эффективность.

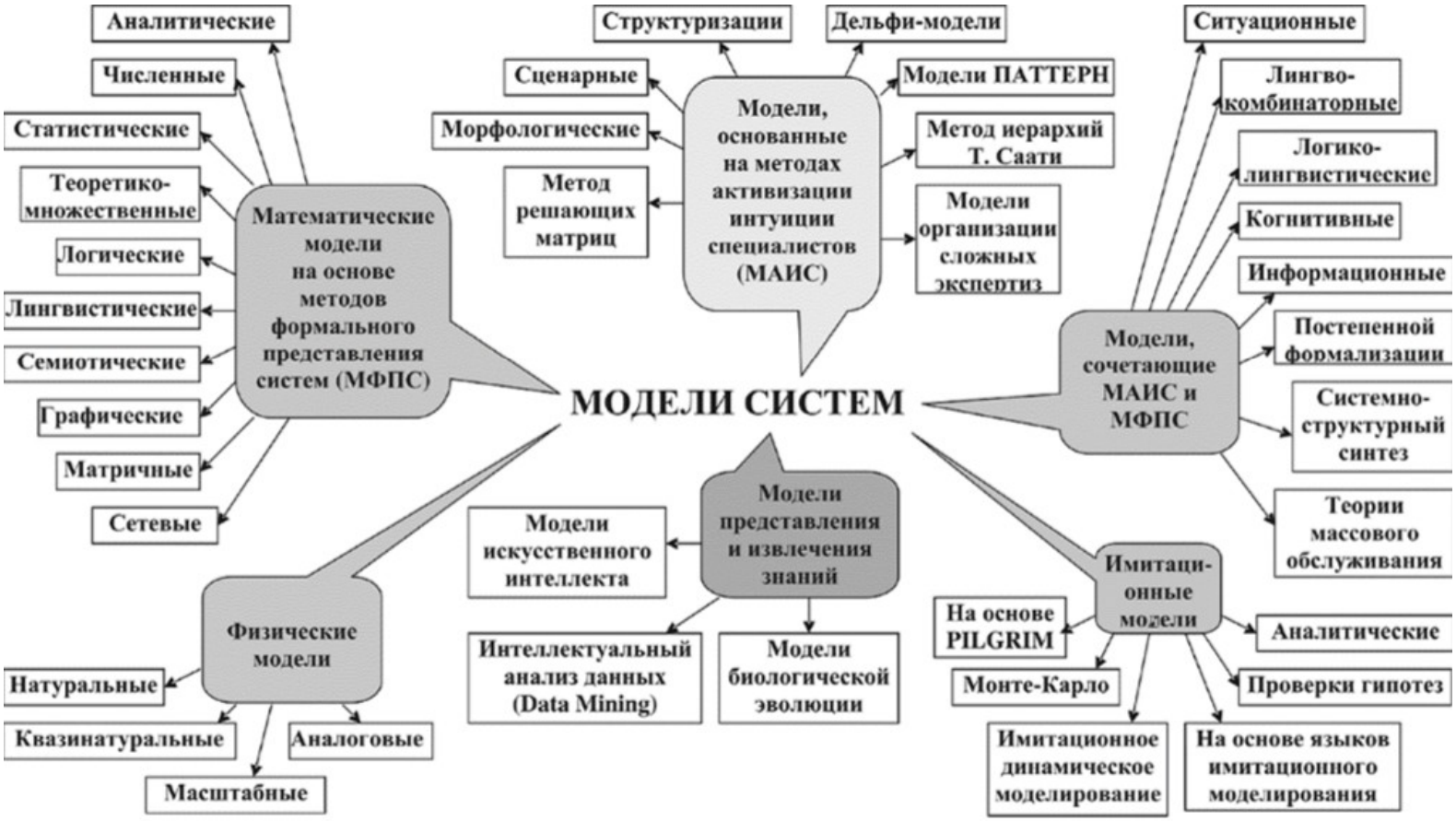
МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

- Математическая модель в информационной безопасности — это описание сценариев в виде последовательности действий нарушителей и соответствующих ответных мер. Приближения таких моделей описывают процессы взаимодействия нарушителя с системой защиты и возможные результаты действий

В процессе проектирования сложных систем, таких как комплексные и интегрированные СЗИ информационных систем (ИС), в большинстве случаев прибегают к моделированию основных процессов, происходящих внутри системы и на стыке среда-система. Кроме того, **модели могут использоваться для проведения мониторинга и аудита безопасности на этапах эксплуатации и сопровождения ИС**.

Под моделированием здесь понимается математическое моделирование, позволяющее получить формальное описание системы и производить в дальнейшем количественные и качественные оценки ее показателей.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ
классификация моделей по Волковой (методы системного анализа)



МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Таблица 1 – Комплекс методов моделирования

Классы методов и моделей	**Назначение методов и моделей**
Теоретические методы	Предназначены для обоснования требований к адекватности реакции на угрозы нарушения ИБ ОЗ, его системы информационной безопасности (СИБ). **Моделирование** ИБ базируется на внедрении методов системного анализа возможных исходов дуэли между сторонами А и В в заданном контексте, аспектах и условиях с позиции ER концепции (сущности исследуемых событий, отношения между ними, влияющая на них атрибутика) на основе логико-вероятностно-информационного подхода и ветвлении главной цели ОЗ, его СИБ на частные. Результаты такого системного анализа представляют в виде синтаксической модели ИБ ОЗ в виде скобочной конструкции, разработанной по правилу составления формулы Бэкуса-Наура.
Эвентологические методы	Предназначены для учёта влияния человеческого (промахи и ошибки), природного, других объективных и субъективных факторов на ситуацию и результаты (исходы дуэли). **Моделирование** базируется на методах теории нечётких множеств и нечёткой логики, интеллектуальных систем, возможностей и риска, прогнозирования и принятия решений, оптимального управления.
Эмпирические методы	Предназначены для оценки достоверности и полезности разработанных теоретических основ системного математического моделирования ИБ ОЗ, его СИБ. **Моделирование** базируется на векторной статистике и численных методах.
Синтаксические модели	Предназначены для выявления *сущности* реально складывающейся и прогнозируемой обстановки во внешней и внутренней среде объекта защиты в условиях XXI века. Исследования базируются на методах системного анализа, а его *результаты* оформляются *по правилу составления формулы Бэкуса-Наура.*
Семантические модели	Предназначены для выявления отношений между сущностями на основе *логико- вероятностно-информационного* подхода и *ветвления* главной цели защиты от рассматриваемых угроз (устойчивость развития) на локальные цели, адекватно принятой скобочной конструкции таких отношений
Математические модели	Разрабатываются по методу вложений критериев, методов и систем защиты информации адекватно формуле Бэкуса-Наура с учётом разработанной системы отношений между сущностями условно взаимосвязанного развития внешней и внутренней среды объекта защиты по ситуации и результатам в условиях XXI века.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

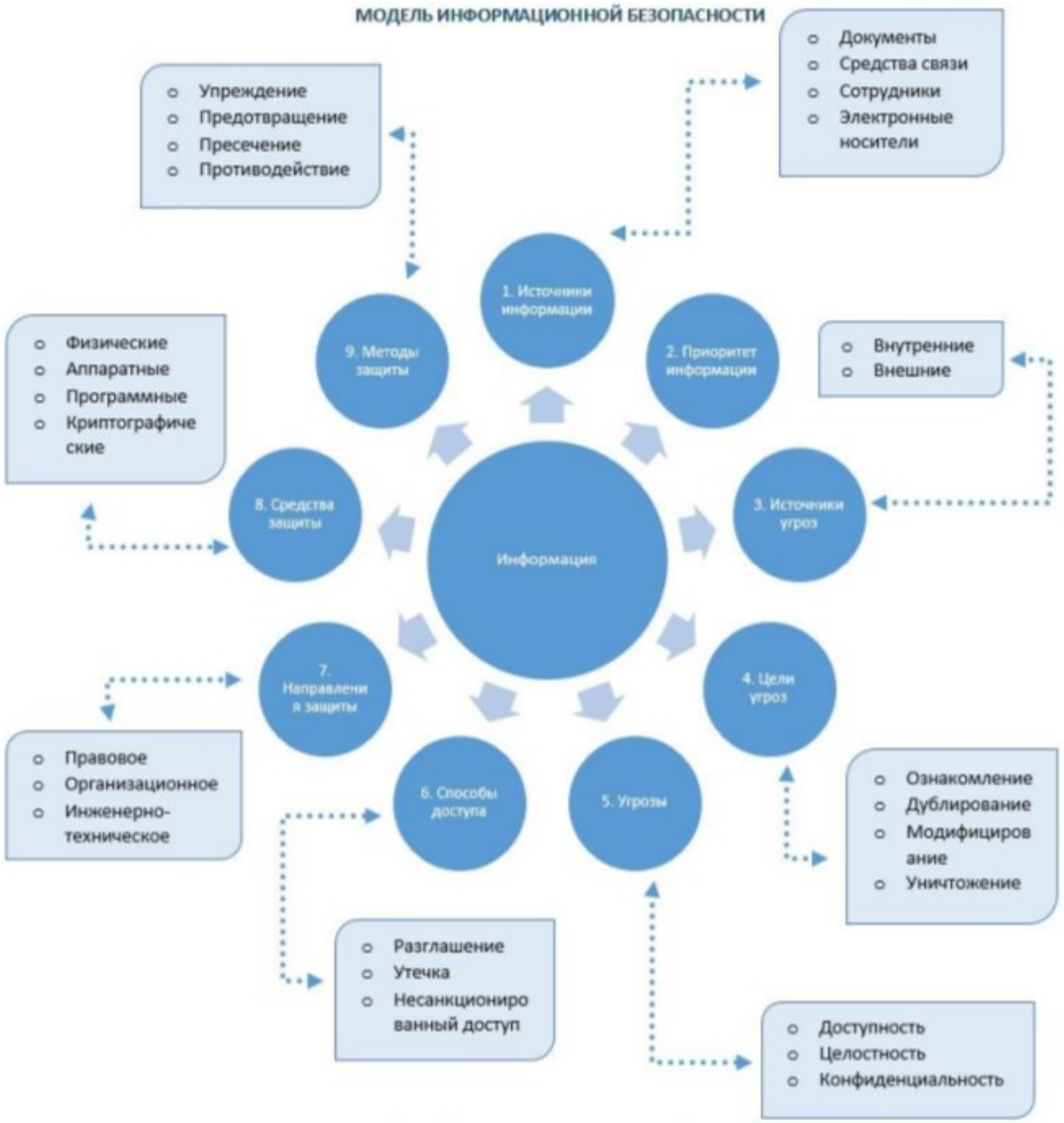


Рис. 2. Концептуальная модель ИБ
Составлено авторами по материалам [8]

ОБЗОР ПРОЦЕССА МЕНЕДЖМЕНТА РИСКА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ ГОСТ Р ИСО/МЭК 27005-2010

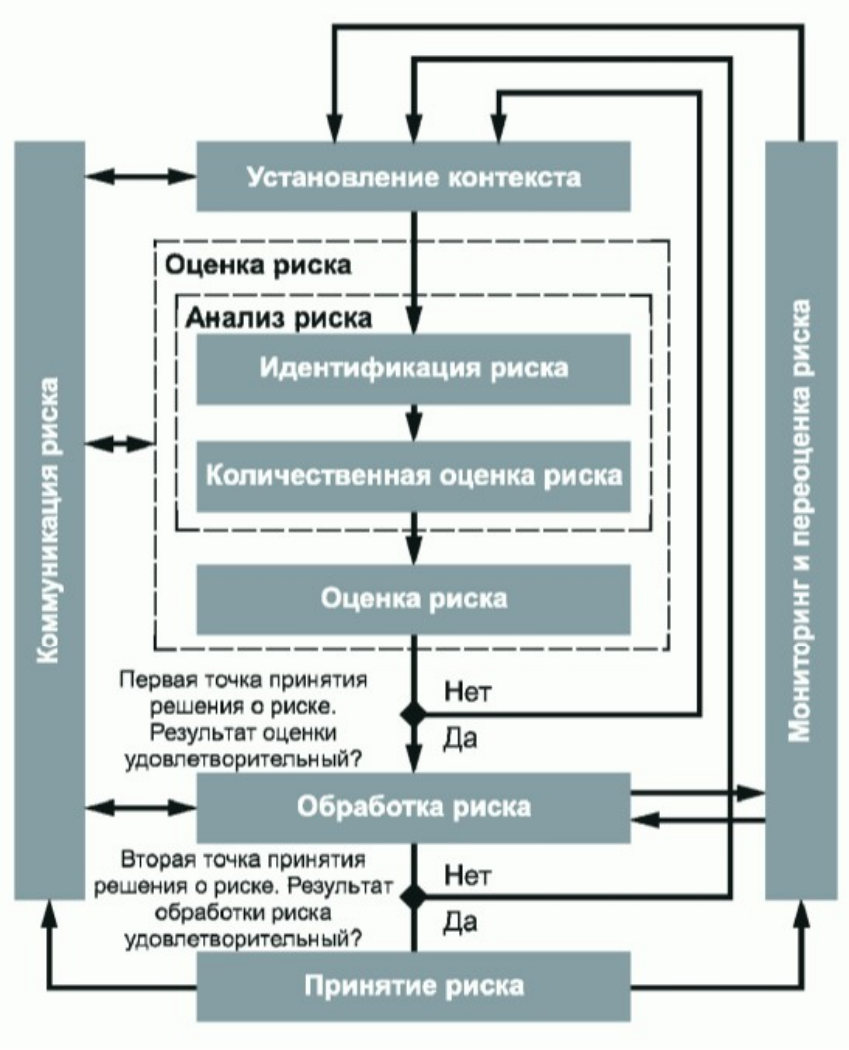


Рисунок 1. Процесс менеджмента риска информационной безопасности

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Отличия большинства моделей заключаются в том, какие параметры они используют в качестве входных, а какие — представляют в виде выходных после проведения расчетов.

Кроме того, **в последнее время широкое распространение получают методы моделирования, основанные на неформальной теории систем: методы структурирования, методы оценивания и методы поиска оптимальных решений.**

Методы структурирования являются развитием формального описания, распространяющимся на организационно-технические системы.

Использование этих методов позволяет представить архитектуру и процессы функционирования сложной системы в виде, удовлетворяющем следующим условиям:

1. полнота отражения основных элементов и их взаимосвязей;
2. простота организации элементов и их взаимосвязей;
3. гибкость — простота внесения изменений в структуру и т. д.

Методы оценивания позволяют определить значения характеристик системы, которые не могут быть измерены или получены с использованием аналитических выражений, либо в процессе статистического анализа, — вероятности реализации угроз, эффективность элемента системы защиты и др.

В основу таких методов положено экспертное оценивание — подход, заключающийся в привлечении специалистов в соответствующих областях знаний для получения значений некоторых характеристик.

Методы поиска оптимальных решений представляют собой обобщение большого количества самостоятельных, в большинстве своем математических теорий с целью решения задач оптимизации. В общем случае к этой группе можно также отнести методы неформального сведения сложной задачи к формальному описанию с последующим применением формальных подходов. Комбинирование методов этих трех групп позволяет расширить возможности применения формальных теорий для проведения полноценного моделирования систем защиты.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

- Основными классическими моделями, ориентированными на защиту от угроз конфиденциальности и целостности информации, а также от угрозы раскрытия параметров КС, являются модели дискреционного управления доступом, изолированной программной среды (ИПС), мандатного управления доступом, безопасности информационных потоков, ролевого управления доступом. Кроме того, в настоящее время активно развивается семейство моделей безопасности логического управления доступом и информационными потоками (сокращенно ДП-моделей).

- Классические модели и ДП-модели построены с применением следующих основных понятий: субъект (subject), объект (object), контейнер (container), сущность (entity), доступ (access), право доступа (access right), информационный поток по памяти (information flow by memory) или по времени (information flow by time). При этом в современной теории компьютерной безопасности наибольшее развитие в области формального моделирования безопасности КС получил подход, заключающийся в представлении исследуемой КС в виде абстрактной системы (автомата), каждое состояние которой описывается доступами, реализуемыми субъектами к сущностям, а переходы КС из состояния в состояние описываются командами или правилами преобразования состояний, выполнение которых, как правило, инициируется субъектами.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

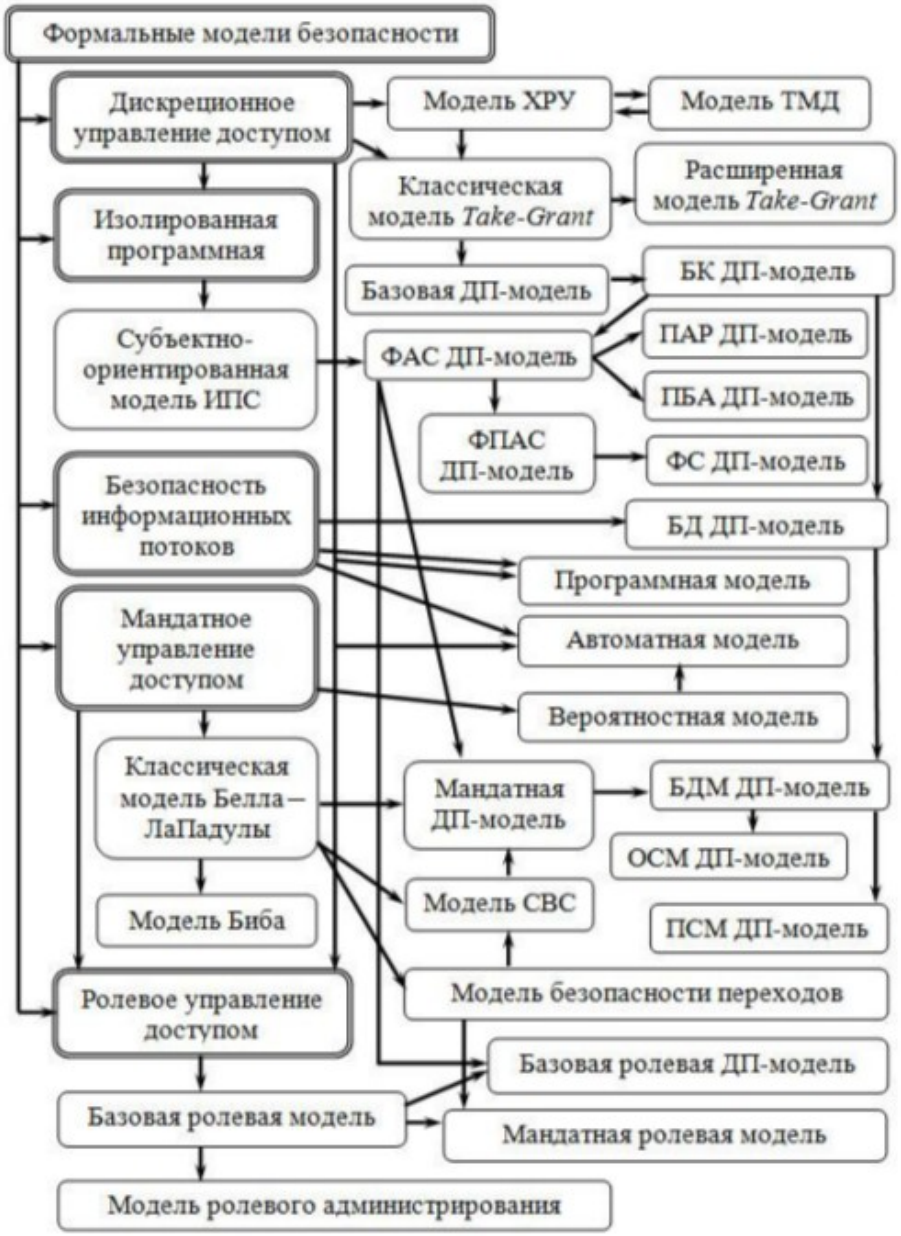


Рис. 1. Классификация и взаимосвязь положений моделей безопасности КС

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

4.1. Классификация математических моделей информационной безопасности по основным видам угроз

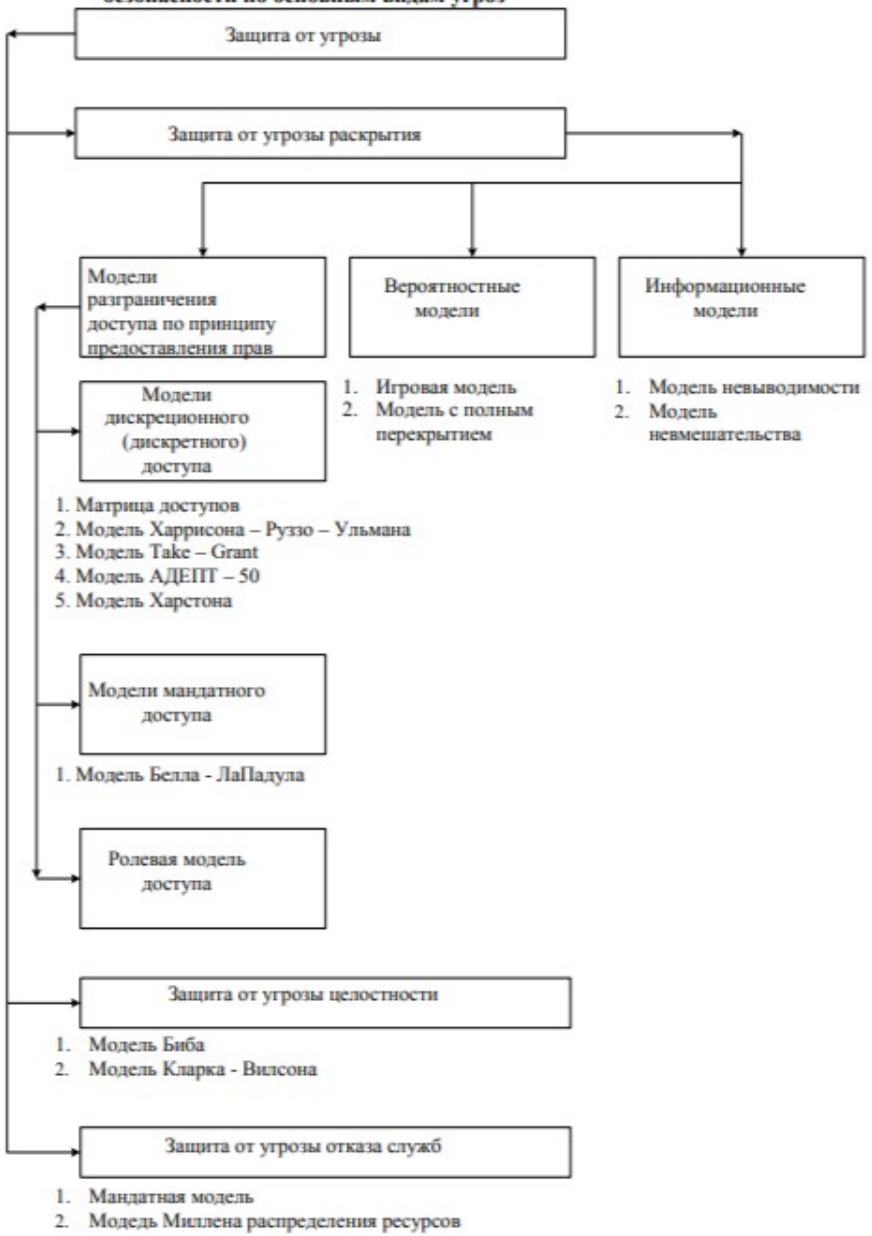


Рис. 4.1. – Классификация моделей информационной безопасности

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

- Основу моделей обеспечения безопасности информации составляют следующие теории:
- – формально-эвристический подход;
- – теория вероятностей и случайных процессов;
- – эволюционное моделирование;
- – теория графов, автоматов и сетей Петри;
- – теории игр и конфликтов;
- – теория катастроф;
- – теория нечетких множеств;
- – энтропийный подход.

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

- **Математическая модель должна обладать следующими свойствами:**
- – масштабируемость;
- – наглядность;
- – практическая направленность;
- – универсальность;
- – комплексность;
- – простота использования;
- – возможность функционирования в условиях высокой неопределенности исходной информации;
- – учет различных факторов воздействия на информационные ресурсы.

- **Как привило, результатами моделирования являются:**
- – оценка возможности реализации различных угроз на информационные
- системы и проведения атак на них;
- – количественная оценка качества функционирования системы защиты;
- – оценка экономической эффективности применения средств защиты информации;
- — структура построения системы защиты информационной системы.

- «У человека должен быть здравый смысл.
- Для всего остального есть ГОСТ»
- (М.С. Кейно)

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

- Оценки параметров СЗИ в условиях высокой степени неопределенности условий ее функционирования должны вычисляться с использованием не одной математической модели, а согласованного семейства моделей, адаптивно конструирующихся одна из другой и, таким образом, непрерывно совершенствующихся на основе оптимального выбора исходных данных.

- При синтезе оптимальных систем защиты исходными должны явиться следующие два положения:
- • выбор математически продуктивного критерия оптимальности в соответствии с архитектурой системы защиты и технологией обработки информации на объекте;
- • четкая математическая формулировка задачи, учитывающая все априорные сведения и позволяющая решить ее в соответствии с принятым критерием.

- Итогом решения задачи синтеза оптимальной системы защиты и его конечной целью должны быть четыре содержательных результата:
- • архитектура системы защиты;
- • количественная оценка качества ее функционирования;
- • оценка практической чувствительности разработанных моделей к отклонениям от априорных данных;
- • физическая реализуемость синтезируемых систем защиты в современных системах обмена данными (соответствие технологии обработки информации уровню ее защиты).

Графовые модели систем защиты информации

Двудольный граф или биграф — это математический термин теории графов, обозначающий граф, множество вершин которого можно разбить на две части таким образом, что каждое ребро графа соединяет какую-то вершину из одной части с какой-то вершиной другой части, то есть не существует ребра, соединяющего две вершины из одной и той же части.

Полный двудольный граф

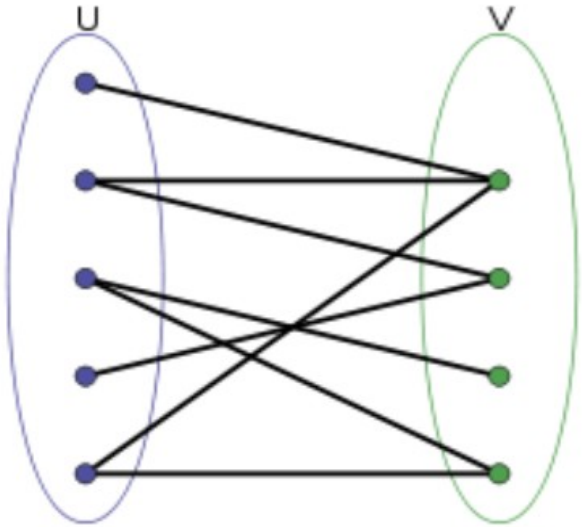
Неориентированный граф $G=(W,E)$ называется двудольным, если множество его вершин можно разбить на две части $U \cup V = W$, так, что ни одна вершина в U не соединена с вершинами в U и ни одна вершина в V не соединена с вершинами в V.

В этом случае, подмножества вершин U и V называются долями двудольного графа G.

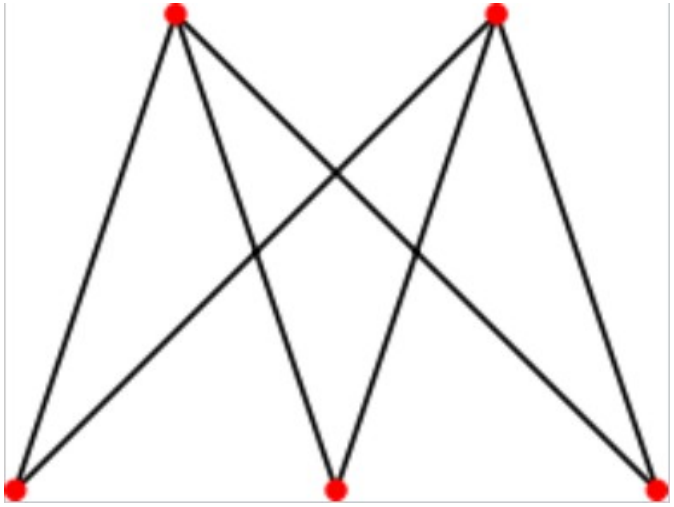
Двудольный граф называется **полным двудольным** (это понятие отлично от полного графа; то есть, такого, в котором каждая пара вершин соединена ребром), если для каждой пары вершин $u \in U$, $v \in V$, существует ребро $u, v \in E$.

Для $|U|=i, |V|=j$ такой граф обозначается символом $K_{i,j}$.

Двудольные графы естественно возникают при моделировании отношений между двумя различными классами объектов. К примеру граф футболистов и клубов, ребро соединяет соответствующего игрока и клуб, если игрок играл в этом клубе.



Двудольный граф



Полный двудольный граф $K_{3,2}$

Графовые модели компьютерных атак

Рассмотрим далее несколько приложений математической теории графов к моделированию СЗИ, первым из которых являются графы компьютерных атак*. Неформально, граф атаки — это граф, представляющий все возможные последовательности действий нарушителя для реализации угрозы. Такие последовательности действий называются путями атак (рис. 7).

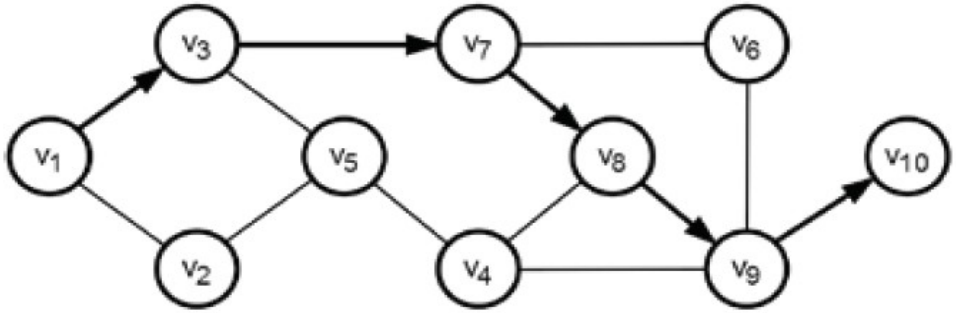


Рис. 7. Граф атаки

Выделяют следующие виды графов атак:

− **state enumeration graph** (граф перечисления состояний) — в таких графах вершинам соответствуют тройки *(s, d, a)*, где *s* — источник атаки, *d* — цель атаки, *a* — элементарная атака (или использование уязвимости); дуги обозначают переходы из одного состояния в другое;

− **condition-oriented dependency graph** (граф зависимостей, ориентированных на условия) — вершинам соответствуют результаты атак, а дугам — элементарные атаки, приводящие к таким результатам;

− **exploit dependency graph** (граф зависимостей эксплойтов или **г**раф условий реализации возможностей эксплойтов**) — вершины соответствуют результатам атак или элементарным атакам, дуги отображают зависимости между вершинами — условия, необходимые для выполнения атаки и следствие атаки.

Такие модели применяются в основном на этапе аудита безопасности сетей для выявления слабых мест системы защиты и прогнозирования действий нарушителя.

*компьютерная атака: Целенаправленное несанкционированное воздействие на информацию, на ресурс автоматизированной информационной системы или получение несанкционированного доступа к ним с применением программных или программно-аппаратных средств (ГОСТ Р 51275-2006 Защита информации. Объект информатизации. Факторы, воздействующие на информацию. Общие положения)

**Эксплойты — это подвид вредоносных программ. Они содержат данные или исполняемый код, способный воспользоваться одной или несколькими уязвимостями в программном обеспечении на локальном или удаленном компьютере.

Графовые модели атак

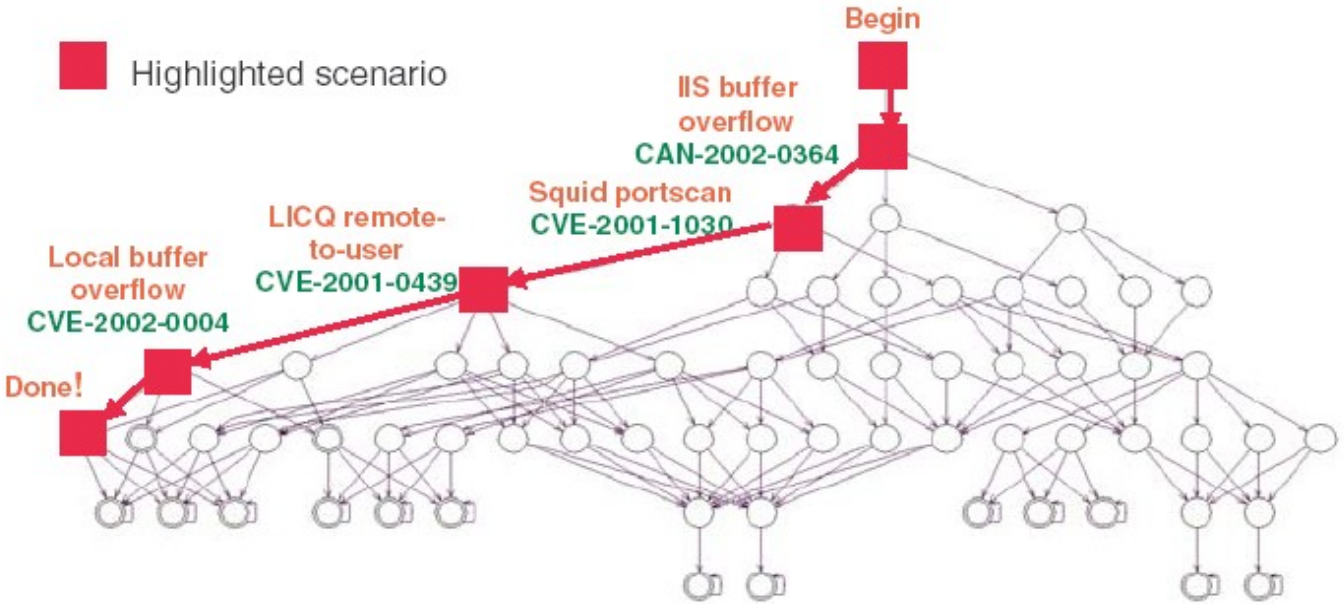
Исследования, связанные с построением, анализом и применением графов атак, ведутся приблизительно с 1994 года. В отечественной литературе данной тематике уделяется незначительное внимание, несмотря на то, что в зарубежных публикациях приводятся примеры эффективно работающих систем, в том числе и коммерческих.

Неформально, граф атак – это граф, представляющий всевозможные последовательности действий нарушителя для достижения угроз (целей). Такие последовательности действий называются трассами (путями) атак.

На рис. 8 – 10 изображены примеры графов атак.

State enumeration graph (граф перечисления состояний) (рис. 8) – в таких графах вершинам соответствуют тройки (s, d, a), где s – источник атаки, d – цель атаки, a – элементарная атака*; дуги обозначают переходы из одного состояни

Рис. 8



*Под элементарной атакой (atomic attack) понимают использование нарушителем уязвимости, например, переполнение буфера службы SSH, позволяющее удаленно получить права администратора системы.

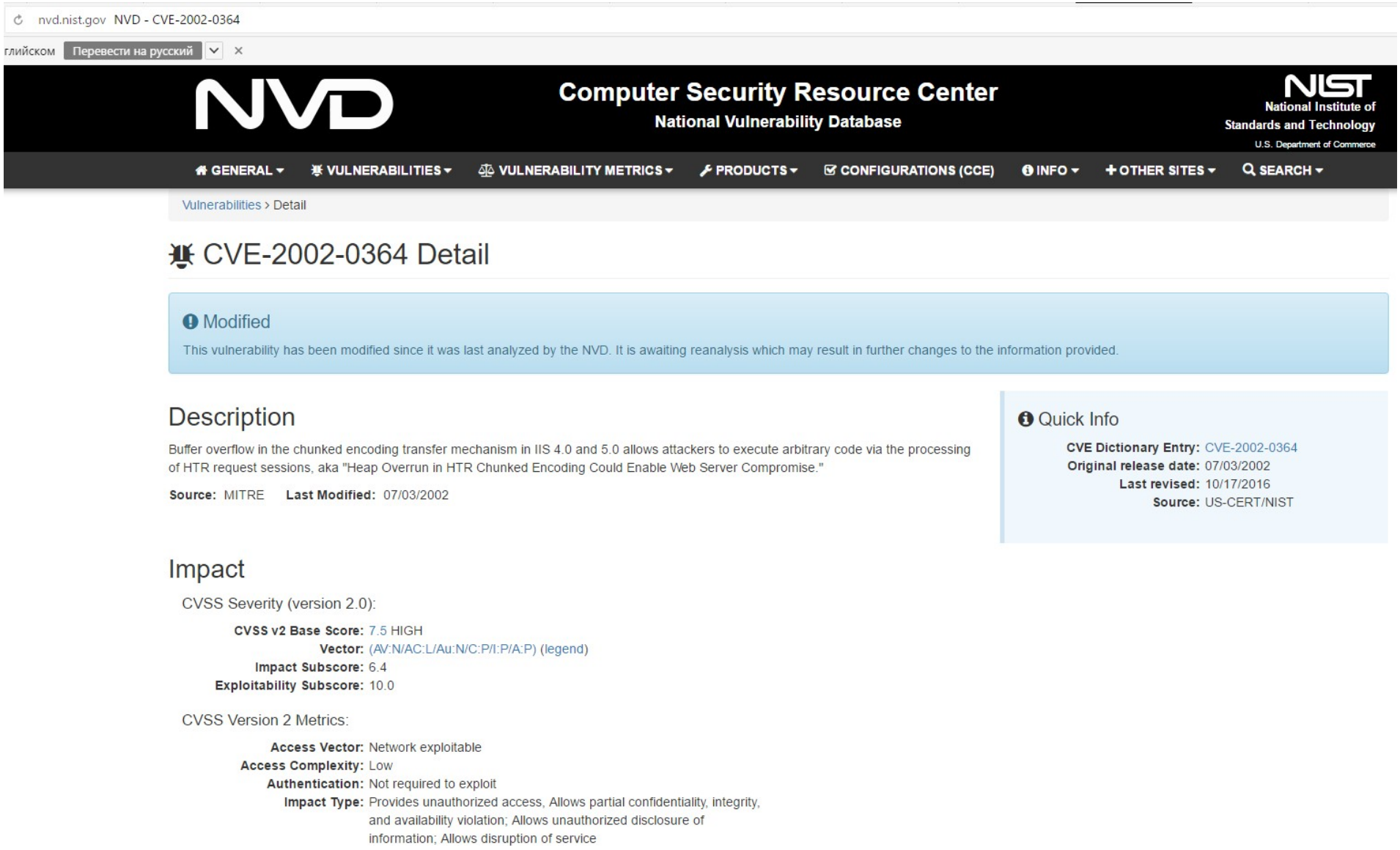
CVE — базы уязвимостей NIST США (CVE-2002-0364 – идентификатор уязвимости);

IIS (Internet Information Services, до версии 5.1 — Internet Information Server) —набор серверов для нескольких служб Интернета от компании Майкрософт (распространяется с операционными системами семейства Windows NT).

Основным компонентом IIS является веб-сервер, который позволяет размещать в Интернете сайты. IIS поддерживает протоколы HTTP, HTTPS, FTP, POP3, SMTP, NNTP. По данным компании Netcraft на июнь 2015 года почти 22 млн сайтов обслуживаются веб-сервером IIS, что составляет 12,32 % от общего числа веб-сайтов.

Национальная база данных уязвимостей (NIST США)

Исследования,



Графовые модели атак

Condition-oriented dependency graph (граф зависимостей, ориентированных на условия) (рис. 9) – вершинам соответствуют результаты атак, а дугам – элементарные атаки, приводящие к таким результатам;

Exploit dependency graph (граф условий реализации возможностей эксплойтов*) (рис. 10) – вершины соответствуют результатам атак или элементарным атакам, дуги отображают зависимости между вершинами – условия, необходимые для выполнения атаки и следствие атаки. Например, атака RSH*** возможна, если нарушитель имеет привилегии суперпользователя на хосте** 1 и хост 3 доверяет хосту 1. В результате атаки нарушитель получает привилегии пользователя на хосте 3.

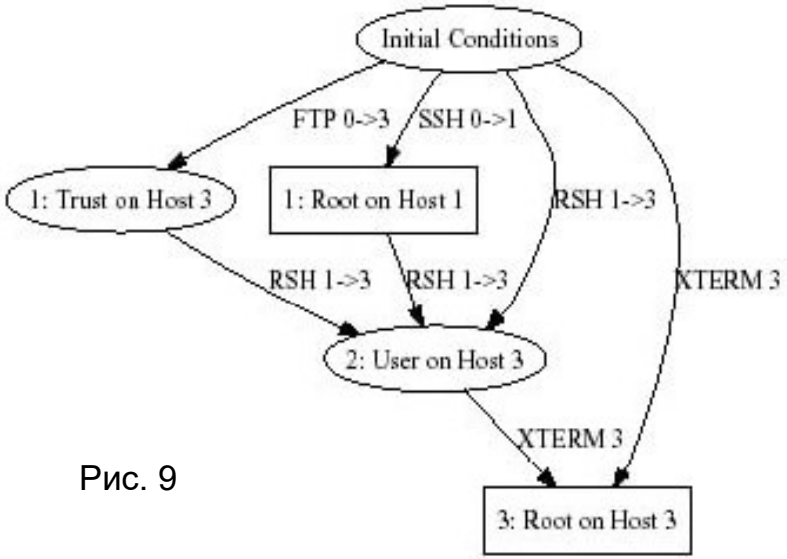


Рис. 9

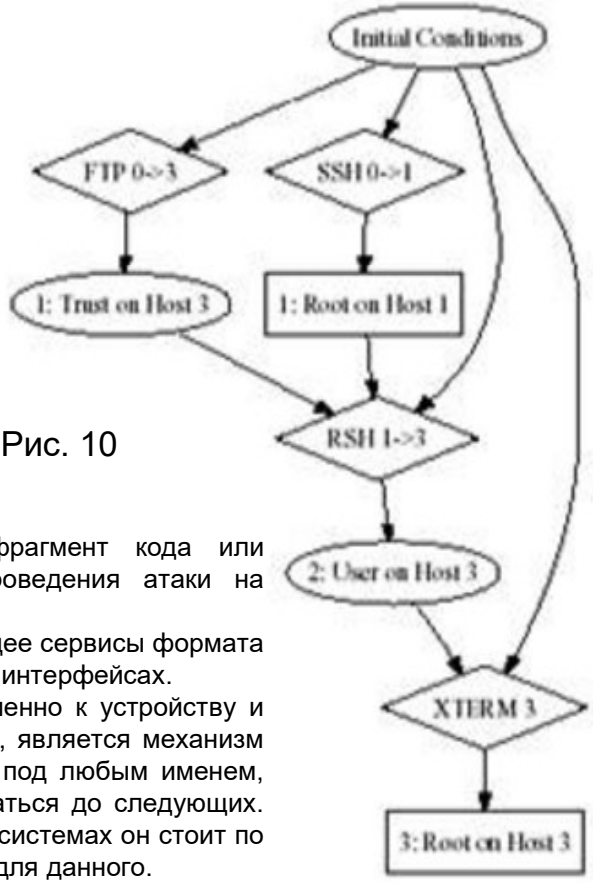


Рис. 10

*Эксплойт (от англ. exploit - эксплуатировать) - это компьютерная программа, фрагмент кода или последовательность команд, использующих уязвимости в ПО и предназначенные для проведения атаки на вычислительную систему.

**Хост (от англ. host — «хозяин, принимающий гостей») — любое устройство, предоставляющее сервисы формата «клиент-сервер» в режиме сервера по каким-либо интерфейсам и уникально определённое на этих интерфейсах.

***RSH расшифровывается как Remote SHell - протокол, позволяющий подключаться удаленно к устройству и выполнять на нем команды. Типичным (и самым опасным) механизмом, использующим доверие, является механизм доверенных хостов, при подключении с которых с помощью r-служб можно зарегистрироваться под любым именем, кроме root, без указания пароля. Вскрыв один из хостов, кракер сможет легко по цепочке добраться до следующих. Опаснее другое: если система, которую собираются атаковать, содержит шаблон "+" (в некоторых системах он стоит по умолчанию) в файлах, где описываются доверенные хосты, то все хосты становятся доверенными для данного.

****xterm является стандартным эмулятором терминала в Unix. Пользователь имеет возможность работать с несколькими xterm терминалами, запущенными в одно и то же время на одном и том же дисплее. Каждый из виртуальных терминалов предоставляет независимый ввод-вывод для процессов, запущенных в каждом из них.

Графовые модели атак

Использование недостатков идентификации tcp-пакетов для атаки на rsh-сервер

В ОС UNIX существует понятие: доверенный хост. Доверенным по отношению к данному хосту называется хост, доступ на который пользователю с данного хоста возможен без его аутентификации и идентификации с помощью r-службы. Обычно, в ОС UNIX существует файл rhost, в котором находится список имен и IP-адресов доверенных хостов. Для получения к ним удаленного доступа пользователю необходимо воспользоваться программами, входящими в r-службу (например, rlogin, rsh и.т.д.). В этом случае при использовании r-программ пользователю для получения удаленного доступа не требуется проходить стандартную процедуру идентификации и аутентификации, заключающуюся во вводе его логического имени и пароля. Аутентифицирующей информацией для r-службы является IP-адрес хоста, с которого пользователь осуществляет r-доступ. Отметим, что все программы из r-службы реализованы на базе протокола TCP. Одной из программ, входящих в r-службу, является rsh,с помощью которой возможно осуществление данной атаки. Программа rsh (remote shell) позволяет отдавать команды shell удаленному хосту. При этом, что является чрезвычайно важным в данном случае, для того, чтобы отдать команду, достаточно послать запрос, но необязательно получать на него ответ. При атаке на r-службы вся сложность для атакующего заключается в том, что ему необходимо послать пакет от имени доверенного хоста, то есть, в качестве адреса отправителя необходимо указать IP-адрес доверенного хоста. Следовательно, ответный пакет будет отправлен именно на доверенный хост, а не на хост атакующего.

Схема удаленной атаки на rsh-сервер была впервые описана небезызвестным P.T. Моррисом в Belt Labs computer Science Technical Report #117, February 25, 1985. Она заключается в следующем:

Пусть хост A доверяет хосту B. Хост X - это станция атакующего.

Вначале атакующий X открывает настоящее TCP-соединение с хостом B на любой TCP-порт (mail, echo и т.д.). В результате X получит текущее значение на данный момент времени ISSb. Далее, X от имени A посылает на B TCP-запрос на открытие соединения:

1. X->B: SYN.ISSx

Получив этот запрос, B анализирует IP—адрес отправителя и решает, что пакет пришел с хоста A. Следовательно, B в ответ посылает на A новое значение ISSb':

2. B ->A: SYN,ACK, ISSb', ACK(ISSx+1)

X никогда не получит это сообщение от B, но, используя предыдущее значение ISSb и схему для получения ISSb‘ (см. выше),может послать на B:

3. X ->B: ACK, ISSx+1, ACK(ISSb'+1)

Отметим, что для того, чтобы послать этот пакет потребуется перебрать некоторое количество возможных значений ACK(ISSb'+1), но не потребуется подбор ISSx+1,так как этот параметр TCP-соединения был послан с X на B в первом пакете.

В итоге rsh-сервер на хосте B считает, что к нему подключился пользователь с доверенного хоста A, а на самом деле это атакующий с хоста X. И хотя X никогда не увидит пакеты с хоста B, но он сможет выполнять на нем команды.

Графовые модели атак

При синтезе графа атак возникают следующие задачи: формализация понятия атаки, разработка формального языка моделирования атак и компьютерной системы (включающей нарушителя, его цели, сеть, средства защиты, отношение достижимости хостов и т.д.), выбор или разработка средств построения графа атаки и его визуализации, разработка средств автоматизации построения и анализа графа.

В основном графы атак рассматриваются в контексте анализа защищенности сетей. Обычно такой анализ сводится к последовательному сканированию всех хостов сети на наличие известных уязвимостей. Результатом является отчет, содержащий перечень найденных уязвимостей и рекомендации по их устранению.

В настоящее время постепенно внедряется другая парадигма анализа защищенности, учитывающая “топологию” компьютерной системы – взаимосвязь объектов компьютерной системы, их свойств и характеристик. Такой анализ защищенности называется топологическим, а средства, его выполняющие, топологическими сканерами безопасности.

Топологический анализ защищенности предполагает построение графа атак на основе результатов сканирования сети, модели нарушителя и данных о конфигурации сети (фильтрации МЭ, маршрутизации, обнаружения атак, достижимости хостов и т.д.) **и его анализ** (вероятностный, минимизационный и т.д.).

Построенный граф содержит все известные сценарии атак для достижения нарушителем угроз.

Результатом его анализа может являться:

- перечень успешных атак, не обнаруживаемых IDS*;
- соотношение реализуемых мер безопасности и уровня защищенности сети;
- перечень наиболее критичных уязвимостей;
- перечень мер, позволяющих предотвратить использование уязвимостей в ПО, для которого отсутствуют обновления;
- наименьшее множество мер, реализация которых сделает сеть защищенной.

Графы атак также используются при расследовании компьютерных инцидентов, для анализа рисков и корреляций предупреждений систем обнаружения атак.

Первоначально графы атак строили вручную, затем были предложены различные подходы к автоматизации данного процесса.

Ключевой проблемой построения графа атак является масштабируемость – возможность построения графа атаки для сети с большим числом хостов и уязвимостей.

*Система обнаружения вторжений (СОВ), английский термин — Intrusion Detection System (IDS).— программное или аппаратное средство, предназначенное для выявления фактов неавторизованного доступа в компьютерную систему или сеть либо несанкционированного управления ими в основном через Интернет. Системы обнаружения вторжений обеспечивают дополнительный уровень защиты компьютерных систем.

Пример сложного графа модели информационной безопасности

- **Сценарий**: Модель атаки на корпоративную сеть банка с использованием тактик MITRE ATT&CK.
- **1. Описание графа**
- **Тип**: Направленный ациклический граф (DAG) с вероятностными метками.
  Узлы: Этапы атаки, уязвимости, контрмеры.
  Рёбра: Возможные переходы между этапами (с вероятностями).

- **2. Ключевые элементы**
- **Вероятности**:
  - P=0.3: Вероятность, что сотрудник откроет фишинговое письмо.
  - P=0.6: Шанс обхода антивируса (если вложение содержит 0-day).
- **Критические узлы**:
  - **Эскалация привилегий**: Точка невозврата для атакующего.
  - **Доступ к БД**: Максимальный ущерб (штрафы + репутация).
- **Контрмеры**:
  - **EDR**: Снижает вероятность установки бекдора с 80% до 30%.
  - **PAM (Privileged Access Management)**: Блокирует 90% попыток эскалации.

- **3. Математическая модель**
- **Расчет общего риска**:
- Риск=Рфишинг×Робход AV×⋯×Ущерб
- Для данного графа:
- Риск=0.3×0.6×0.5×0.5×$10M=$450K
- **Оптимизация**:
  Внедрение EDR + обучение снижает риск до:
- 0.1×0.3×0.2×0.1×$10M=$6K

- **4. Анализ результатов**
- **Слабые места**:
  - Этапы с высокой вероятностью успеха (фишинг, обход AV).
- **Оптимальные контрмеры**:
  - Внедрение EDR + обучение дает ROI 150% за год.
- **Рекомендации**:
- Приоритет: EDR + PAM.
- Мониторинг: Focus на узлы C (обход AV) и H (эскалация).

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Динамика функционирования сети Петри обеспечивается при помощи маркеров (фишек), которые обозначим как M, графически представляются точками внутри позиций. Неформально динамику функционирования сети Петри можно представить как совокупность локальных действий пользователя, работающего с СЗИ от НСД в АИС, которые называются срабатываниями переходов (рисунок 1).

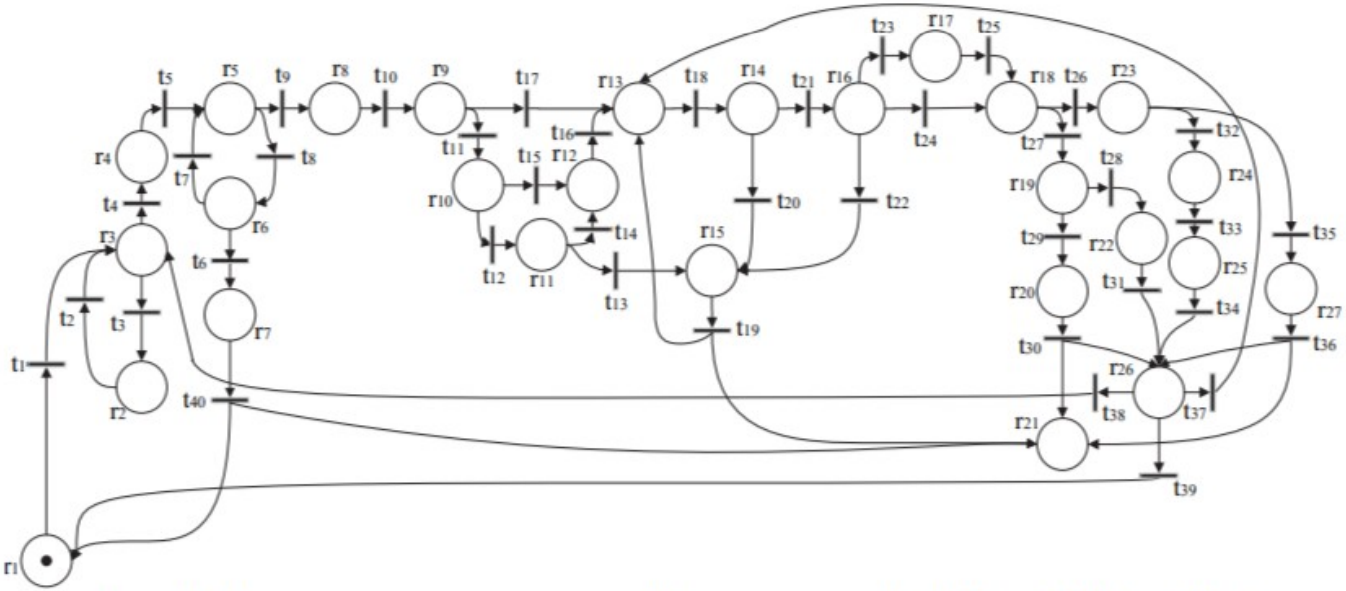
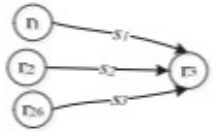


Рис. 1. Начальная разметка сети Петри типовой СЗИ от НСД в АИС

МАТЕМАТИЧЕСКИЕ МОДЕЛИ В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Таблица 3. Расчет вероятностей перехода

Графическое описание	Описание	Расчет вероятности
	s_1 — выключение компьютера после неоднократного ввода неправильного пароля; s_2 — завершение работы с СЗИ.	$P(r_1)=P(r_7)\times P(r_1\|r_7)+$ $+P(r_{26})\times P(r_1\|r_{26})$
	s_1 — предъявление идентификатора; s_2 — в случае, если идентификатор не зарегистрирован в системе предъявите другой; s_3 — повторное предъявление идентификатора.	$P(r_3)=P(r_1)\times P(r_3\|r_1)+$ $+P(r_2)\times P(r_3\|r_2)+$ $+P(r_{26})\times P(r_3\|r_{26})$
	s_1 — ввод пароля; s_2 — повторный ввод пароля.	$P(r_5)=P(r_4)\times P(r_5\|r_4)+$ $+P(r_6)\times P(r_5\|r_6)$
	s_1 — идентификационная информация внешнего носителя уже проверялась (механизм контроля устройств не срабатывает); s_2 — проверяется принадлежность внешнего носителя пользователю (срабатывает механизм контроля устройств).	$P(r_{12})=P(r_{10})\times P(r_{12}\|r_{10})+$ $+P(r_{11})\times P(r_{12}\|r_{11})$
	s_1 — обращение к объекту, находящемуся в АИС; s_2 — обращение к объекту, находящемуся во внешнем устройстве; s_3 — доступ к объекту был заблокирован, обращение к другому объекту.	$P(r_{13})=P(r_9)\times P(r_{13}\|r_9)+$ $+P(r_{12})\times P(r_{13}\|r_{12})+$ $+P(r_{26})\times P(r_{13}\|r_{26})$

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

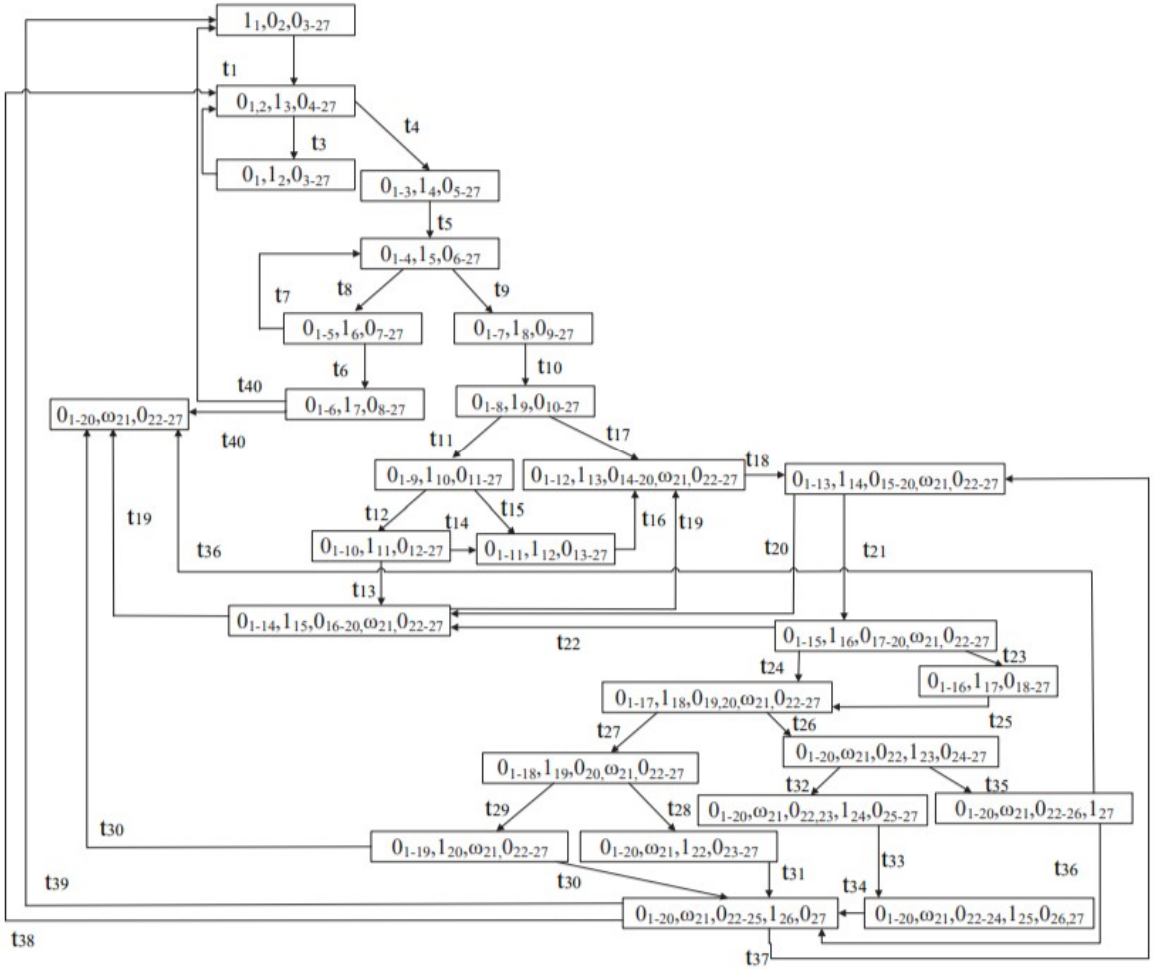


Рис. 2. Граф разметок сети Петри типовой СЗИ от НСД в АИС

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

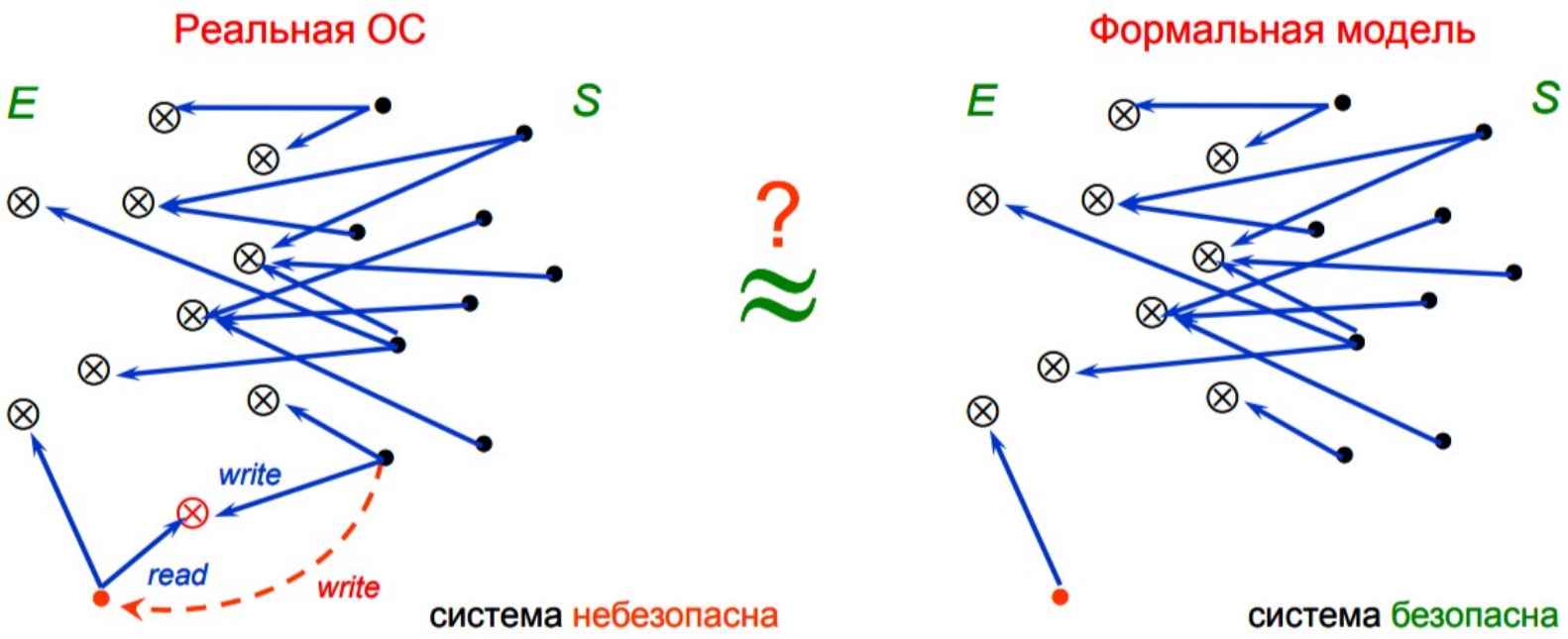
Система уравнений (4) представляет собой расчет переходных вероятностей согласно рисунку 1 и таблице 3:

$$
\begin{cases}
P(r_1), P(r_4), P(r_6), P(r_7), P(r_8), P(r_9), P(r_{10}), P(r_4), P(r_{11}), P(r_{14}), \\
P(r_{17}), P(r_{19}), P(r_{20}), P(r_{22}), P(r_{23}), P(r_{24}), P(r_{25}), P(r_{27}) = 1; \\
P(r_1) = P(r_7) \times P(r_1 | r_{26}) + P(r_{26}) \times P(r_1 | r_{26}); \\
P(r_3) = P(r_1) \times P(r_3 | r_1) + P(r_2) \times P(r_3 | r_2) + P(r_{26}) \times P(r_3 | r_{26}); \\
P(r_5) = P(r_4) \times P(r_5 | r_4) + P(r_6) \times P(r_5 | r_6); \\
P(r_{12}) = P(r_{10}) \times P(r_{12} | r_{10}) + P(r_{11}) \times P(r_{12} | r_{11}); \\
P(r_{13}) = P(r_9) \times P(r_{13} | r_9) + P(r_{12}) \times P(r_{13} | r_{12}) + P(r_{26}) \times P(r_{13} | r_{26}); \\
P(r_{15}) = P(r_{11}) \times P(r_{15} | r_{11}) + P(r_{14}) \times P(r_{15} | r_{14}) + P(r_{16}) \times P(r_{15} | r_{26}); \\
P(r_{18}) = P(r_{16}) \times P(r_{18} | r_{16}) + P(r_{17}) \times P(r_{18} | r_{17}); \\
P(r_{21}) = P(r_7) \times P(r_{21} | r_7) + P(r_{15}) \times P(r_{21} | r_{15}) + P(r_{20}) \times P(r_{21} | r_{20}) + P(r_{27}) \times P(r_{21} | r_{27}); \\
P(r_{26}) = P(r_{20}) \times P(r_{26} | r_{20}) + P(r_{22}) \times P(r_{26} | r_{22}) + P(r_{25}) \times P(r_{26} | r_{25}) + P(r_{27}) \times P(r_{26} | r_{27}).
\end{cases}
\qquad (4)
$$

МАТЕМАТИЧЕСКИЕ МОДЕЛИ
В ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

Применение классических методов теории моделирования (статистических, экспериментальных) для анализа адекватности моделей безопасности логического управления доступом и информационными потоками реальным КС затруднено.

Пример:



Полная классификация математических моделей оценки информационной безопасности

- **1. Вероятностные и статистические модели**
- **Цель:** Оценка рисков, прогнозирование атак на основе данных.
- **Модели анализа рисков:**
  - **CVSS (Common Vulnerability Scoring System)** – количественная оценка уязвимостей (NIST, 2023).
  - **Модели на основе теории вероятностей** (распределения Пуассона, Бернулли для частот атак).
- **Байесовские сети:**
  - Представление зависимостей между угрозами и уязвимостями (Frigault et al., 2008).
- **Примеры:**
- Оценка вероятности успешной атаки на систему с учетом исторических данных.
- **2. Детерминированные модели**
- **Цель:** Точное моделирование атак и защитных механизмов.
- **Графовые модели:**
  - **Attack Graphs** – представление цепочек атак (например, MulVAL).
  - **Деревья решений** для анализа инцидентов.
- **Теория игр:**
  - Модели типа **«Защитник vs. Атакующий»** (например, Stackelberg Games).
- **Примеры:**
- Оптимизация размещения средств защиты в корпоративной сети.
- **3. Формальные методы**
- **Цель:** Верификация корректности систем и протоколов.
- **Логические модели:**
  - **Модальная логика** (например, логика Бан-Логи для анализа протоколов (BAN Logic, 1989)).
- **Автоматные модели:**
  - Конечные автоматы для проверки состояний системы.
- **Примеры:**
- Доказательство безопасности криптопротокола (например, TLS).
- **4. Гибридные и динамические модели**
- **Цель:** Учет неопределенности и адаптивности.
- **Нечеткая логика (Fuzzy Logic):**
  - Оценка качественных параметров (например, модели на основе Fuzzy AHP).
- **Агент-ориентированные модели:**
  - Имитация поведения злоумышленников и защитников.
- **Примеры:**
- Адаптивная система реагирования на кибератаки.
- **5. Машинное обучение и AI-модели**
- **Цель:** Прогнозирование и автоматизация анализа.
- **Нейронные сети:**
  - Для обнаружения аномалий (например, LSTM для анализа логов).
- **Методы кластеризации:**
  - Выявление новых типов угроз (например, k-means).
- **Примеры:**
- Предсказание DDoS-атак на основе исторических данных.
- **6. Квантовые и постквантовые модели *(Перспективное направление)***
- **Цель:** Учет угроз квантовых вычислений.
- **Квантовые алгоритмы** для криптоанализа (например, Шора).
- **Модели устойчивости** к квантовым атакам (NIST Post-Quantum Cryptography, 2022).

Выбор модели зависит от:
- Точности требований (детерминизм vs. вероятности).
- Доступности данных (статистика vs. логика).
- Ресурсов (сложность вычислений).

Перечень видов математических моделей ИБ по назначению

- ## 1. Модели анализа угроз
- **Назначение**: Формализация и оценка потенциальных угроз
  Типы моделей:
- **Атаковые деревья (Attack Trees)** – иерархическое представление сценариев атак
- **Графы атак (Attack Graphs)** – моделирование цепочек эксплуатации уязвимостей
- **Модели STRIDE** – классификация угроз по 6 категориям (Spoofing, Tampering и др.)
- **Марковские модели** – оценка вероятностей перехода между состояниями системы при атаке
- **Пример**:
  Модель графа атак для Active Directory с расчетом вероятности компрометации домена.
- ## 2. Модели оценки рисков
- **Назначение**: Количественная оценка потенциального ущерба
  Типы моделей:
- **FAIR (Factor Analysis of Information Risk)** – факторный анализ рисков
- **Байесовские сети** – расчет условных вероятностей рисков
- **Матрицы рисков** – сопоставление вероятности и воздействия (5×5)
- **Монте-Карло** – симуляция сценариев методом случайных выборок
- **Пример**:
  Расчет финансового ущерба от утечки данных в банке с учетом:
  Риск = Вероятность × (Ущерб репутации + Штрафы регуляторов).
- ## 3. Модели защиты и контрмер
- **Назначение**: Оптимизация стратегий защиты
  Типы моделей:
- **Теория игр** (Stackelberg, Signaling Games) – моделирование взаимодействия атакующего и защитника
- **Деревья решений** – выбор оптимальных контрмер
- **Модели надежности** – расчет MTBF (Mean Time Between Failures) для систем
- **Пример**:
  Игра Stackelberg для распределения бюджета ИБ между защитой серверов и обучением сотрудников.
- ## 4. Модели криптографической стойкости
- **Назначение**: Оценка устойчивости шифрования
  Типы моделей:
- **Модели сложности** (O-нотация) – оценка времени взлома алгоритма
- **Решеточные модели** – для постквантовой криптографии (LWE)
- **Диффузионные/конфузионные модели** – анализ стойкости S-блоков
- **Пример**:
  Модель взлома RSA-2048 квантовым компьютером с 4096 кубитами.
- ## 5. Модели обнаружения аномалий
- **Назначение**: Выявление отклонений от нормального поведения
  Типы моделей:
- **Статистические** (z-score, метод 3-сигм)
- **Машинное обучение** (Isolation Forest, One-Class SVM)
- **Глубокое обучение** (LSTM, Autoencoders)
- **Пример**:

6. Модели управления доступом
Назначение: Формальная верификация политик доступа
Типы моделей:
Модель Белла-ЛаПадулы (многоуровневая секретность)
Модель Биба (целостность)
RBAC-модели (ролевой доступ)
Пример:
Доказательство отсутствия скрытых каналов в системе с мандатным управлением доступом.

7. Модели анализа уязвимостей
Назначение: Приоритезация уязвимостей
Типы моделей:
CVSS – количественная оценка уязвимостей
EPSS – прогнозирование эксплуатации уязвимостей
Модели распространения (SIR-модели) – анализ скорости заражения
Пример:
Прогноз вероятности эксплуатации CVE-2023-1234 через 30 дней после публикации.

8. Модели инцидент-менеджмента
Назначение: Оптимизация реагирования
Типы моделей:
Очереди массового обслуживания – расчет времени обработки инцидентов SOC
Petri Nets – моделирование workflows реагирования
Агентные модели – имитация действий киберпреступников
Пример:
Расчет оптимального числа аналитиков SOC при нагрузке 100 инцидентов/день.

9. Модели безопасности IoT/ICS
Назначение: Защита уязвимых устройств
Типы моделей:
Конечные автоматы – анализ состояний промышленных контроллеров
Нечеткая логика – оценка рисков при недостатке данных
Физические модели – анализ side-channel атак
Пример:
Модель атаки на PLC, приводящей к остановке конвейера.

10. Модели соответствия регуляторам
Назначение: Автоматизация аудита
Типы моделей:
Логические модели – проверка соответствия GDPR/152-ФЗ
Онтологии – сопоставление требований ISO 27001 и NIST CSF
Модели графов знаний – выявление противоречий в политиках
Пример:
Автоматическая генерация отчетов для ФСТЭК по ГОСТ Р 57580.1

Классификация по сложности

Уровень	Типы моделей	Требуемая экспертиза
Базовый	Матрицы рисков, CVSS	Junior Security Analyst
Средний	Байесовские сети, Attack Graphs	Security Engineer
Продвинутый	Теория игр, LSTM, решеточная криптография	PhD in Cybersecurity

Модели информационной безопасности с использованием машинного обучения и AI

- **1. Классификация угроз с помощью ML**

- **1.1. Обнаружение аномалий в сетевом трафике**
- **Алгоритмы**:
- **Изолирующий лес (Isolation Forest)** – для выявления редких событий (DDoS, сканирование портов).
- **One-Class SVM** – обучение только на "нормальном" трафике.
- **1.2. Детектирование вредоносного ПО**
- **Подходы**:
- **Статические признаки**: Хэши, строки, заголовки PE-файлов → Random Forest.
- **Динамический анализ**: Логи поведения (API-вызовы, доступ к реестру) → LSTM.
- **Датасет**: EMBER (1.1 млн образцов).
  Точность: До 99% (F1-score) для новых семейств malware.

- **2. Прогнозирование кибератак**
- **2.1. Предсказание атак на основе временных рядов**
- **Модель**: **Prophet** (Facebook) + **LSTM**.
  Данные: Логи IDS за 2 года.
- **Результат**: Прогноз всплесков атак с точностью 85%.
- **2.2. Генеративные модели для тестирования защиты**
- **Технология**: **GAN** (Generative Adversarial Networks).
- **Generator**: Создает фейковые атаки (например, SQL-инъекции).
- **Discriminator**: Пытается отличить их от реальных.
- **Применение**:
- Обучение WAF на синтетических данных → улучшение детектирования на 40%.

- **3. Анализ поведения пользователей (UEBA)**
- **3.1. Выявление инсайдеров**
- **Метрики**:
- Аномальный доступ к файлам.
- Необычное время работы.
- **Алгоритм**: **Autoencoders** (обучение на "нормальном" поведении).
- **Кейс**: В банке выявлен сотрудник, сливающий данные (точность 92%).

- **4. Обработка естественного языка (NLP) в ИБ**
- **4.1. Классификация фишинговых писем**
- **Модель**: **BERT** + **Logistic Regression**.
  Признаки:
- Текст письма.
- URL-адреса.
- Заголовки.
- **Датасет**: Enron Phishing.
- **Точность**: 98% (F1-score).

XAI (Explainable AI) и Federated Learning: Объяснение и применение в ИБ

- **1. XAI (Explainable AI) — Объяснимый искусственный интеллект**
- **Что это?**
  Методы и алгоритмы, которые делают предсказания ИИ-моделей **понятными для человека**. Позволяют ответить на вопросы:
- *Почему модель приняла такое решение?*
- *Какие факторы повлияли на результат?*
- **Примеры техник XAI:**
- **SHAP (Shapley Additive Explanations)** — оценивает вклад каждого признака в прогноз.
- **LIME (Local Interpretable Model-agnostic Explanations)** — аппроксимирует сложную модель простой для интерпретации.
- **Attention Maps** (в нейросетях) — выделяет значимые участки данных (например, в анализе вредоносного кода).
- **Применение в ИБ:**
- **Анализ решений SIEM-систем**: Почему атака классифицирована как APT, а не обычный сканирование портов?
- **Аудит моделей оценки рисков**: Какие параметры (IP, геолокация, поведение) повлияли на высокий балл угрозы?
- **Соответствие 152-ФЗ**: Обоснование автоматических блокировок для регуляторов.

- **2. Federated Learning (FL) — Федеративное обучение**
- **Что это?**
  Децентрализованный подход к обучению ИИ-моделей, при котором **данные не покидают устройств пользователей**. Обучение происходит так:
- Локальные модели тренируются на своих данных.
- Только **параметры моделей** (а не сырые данные) отправляются на сервер.
- Сервер агрегирует обновления (например, усредняет веса) и рассылает улучшенную модель всем участникам.
- **Преимущества для ИБ:**
- **Соблюдение конфиденциальности**: Персональные данные/логи атак не передаются в централизованное хранилище.
- **Устойчивость к атакам**: Даже если один узел скомпрометирован, злоумышленник не получит всех данных.
- **Эффективность**: Обучение на разнородных данных из разных источников (например, фишинг в банках vs. госсекторе).
- **Применение в ИБ:**
- **Коллективный анализ угроз**: Банки совместно улучшают модель детектирования мошенничества, не делясь транзакциями.
- **Распределенные SOC**: Анализ киберинцидентов в филиалах компании без передачи логов в центр.
- **IoT-безопасность**: Обучение на данных с датчиков без их централизованного сбора.

- **Перспективы в России**
- **XAI**:
  - Требуется для сертификации ИИ-моделей в госсекторе (проект Минцифры 2024).
  - Интеграция с отечественными SIEM («Рубин», «Киберпротект»).
- **Federated Learning**:
  - Пилотные проекты в Сбербанке и РЖД для анализа киберугроз.
  - Поддержка в российских фреймворках (например, OpenFL от Intel адаптирован под «Эльбрус»).
- **Пример российского кейса:**
  Компания «Ростелеком-Солар» использует XAI для объяснения срабатываний системы обнаружения атак на КИИ. Модель на основе CatBoost + SHAP показывает:
- 90% точности;
- Возможность аудита для ФСТЭК.

Квантово-безопасные алгоритмы

- **1. Угроза квантовых компьютеров**
- Квантовые компьютеры используют принципы квантовой механики (кубиты, суперпозиция, квантовая запутанность) для решения задач, непосильных классическим компьютерам.
- **Критические риски для ИБ:**
- **Алгоритм Шора** - взламывает RSA, ECC и другие асимметричные криптоалгоритмы за полиномиальное время
- **Алгоритм Гровера** - ускоряет перебор в 2^(n/2) раз, угрожая симметричному шифрованию
- **Сроки угрозы:**
- По оценкам NIST, RSA-2048 может быть взломан к 2030 году
- Банк России в "Стратегии развития финансовых технологий" указывает 2025-2027 гг. как сроки перехода на квантово-безопасные алгоритмы

- **2. Постквантовая криптография (PQC)**
- Алгоритмы, устойчивые к атакам как классических, так и квантовых компьютеров.
- **Основные подходы:**
- **Решеточная криптография (Lattice-based)**
  - Основывается на сложности решения задачи обучения с ошибками (LWE)
  - Примеры: Kyber (KEM), Dilithium (цифровые подписи)
  - Преимущества: высокая скорость, хорошие параметры безопасности
  - Недостатки: большие размеры ключей (~1-2 Кб)
- **Криптография на хеш-функциях**
  - Основана на стойкости хеш-функций
  - Пример: SPHINCS+ (подписи)
  - Плюсы: простая реализация
  - Минусы: большие размеры подписей (~40 Кб)
- **Кодовая криптография**
  - Использует исправление ошибок в кодах
  - Пример: Classic McEliece
  - Преимущества: проверенная стойкость
  - Недостатки: крайне большие ключи (1 Мб+)
- **Многомерные квадратичные системы**
  - Основаны на сложности решения систем нелинейных уравнений
  - Пример: Rainbow (подписи)
  - Проблемы: недавно были найдены уязвимости
- **3. Стандартизация NIST PQC**
- NIST ведет процесс стандартизации с 2016 года. Текущий статус (2024):
- **Уже стандартизированы:**
- CRYSTALS-Kyber (асимметричное шифрование)
- CRYSTALS-Dilithium (подписи)
- SPHINCS+ (подписи)
- Falcon (подписи)
- **На рассмотрении:**
- Classic McEliece
- BIKE
- HQC

Квантово-безопасные алгоритмы

- **4. Реализация в России**
- **Текущее состояние:**
- ГОСТ Р 34.10-2021 (подписи) и ГОСТ Р 34.11-2021 (хеши) пока не квантово-устойчивы
- Ведется разработка отечественных стандартов на базе решеточных методов
- **Рекомендации по переходу:**
- **Гибридные схемы**:
  - Совместное использование классических и PQC-алгоритмов
  - Пример: RSA + Kyber
- **Увеличение параметров**:
  - Переход на AES-256, SHA-384/512
- **Квантовое распределение ключей (QKD)**:
  - Физические методы защиты на основе квантовой криптографии
  - Ограничение: требует специальной инфраструктуры
- **5. Практические шаги для организаций**
- **Аудит криптографии**:
  - Выявление систем, использующих RSA, ECC, DSA
  - Оценка сроков жизненного цикла защищаемых данных
- **Пилотные проекты**:
  - Тестирование Open Quantum Safe (OQS) библиотеки
- **План перехода**:
  - 2024-2025: Тестирование и разработка стратегии
  - 2026-2028: Постепенный переход на гибридные схемы
  - После 2030: Полный переход на PQC
- **6. Ограничения и проблемы**
- **Производительность**:
  - PQC-алгоритмы в 10-100 раз медленнее классических
  - Требуют больше вычислительных ресурсов
- **Размеры ключей**:
  - Увеличение в 10-1000 раз по сравнению с RSA/ECC
  - Проблемы для IoT-устройств
- **Совместимость**:
  - Необходимость обновления протоколов (TLS 1.3+, IPsec)

Математические модели ИБ как задачи оптимизации

- Математические модели в информационной безопасности часто формулируются как задачи оптимизации, где требуется найти **наилучшее решение** (максимизация защиты или минимизация рисков) при заданных ограничениях (бюджет, ресурсы, требования регуляторов). Рассмотрим ключевые аспекты.
- **1. Типы оптимизационных задач в ИБ**
- **1.1. Минимизация рисков**
- **Формулировка**:
  Найти набор контрмер x, минимизирующий общий риск R:
- min$R(x)=i=1\sum n\ Pi(x)\cdot Ci(x)$ где:
- Pi(x) — вероятность успешной атаки по сценарию i*i*,
- Ci(x) — ущерб от атаки i.
- **Пример**:
  Выбор между установкой WAF (x1) или обновлением системы (x2) для защиты веб-приложения:
- $x1,x2\min(0.1\cdot106\cdot x1+0.3\cdot5\cdot105\cdot x2)$
- **Ограничения**:
- x1+x2≤бюджет,xi∈{0,1}$x1+x2\leq$бюджет,$xi\in\{0,1\}$
- **1.2. Оптимизация распределения ресурсов**
- **Формулировка**:
  Максимизировать уровень защиты S*S* при ограниченном бюджете B*B*:
- max$S(x)=j=1\sum mfj(xj)$где fj(xj)$fj(xj)$ — эффективность контрмеры j*j* (например, снижение вероятности атаки).
- **Пример**:
  Распределение бюджета между:
- Обучением сотрудников (x1),
- Покупкой антивируса (x2),
- Апгрейдом серверов (x3).
- **Ограничения**:
- $c1x1+c2x2+c3x3\leq B$ где ci — стоимость меры i.
- **1.3. Теория игр (оптимизация защиты против атакующего)**
- **Формулировка**:
  *И*гра между защитником (D*D*) и атакующим (A*A*) с платежной матрицей:
- Атака на сервер 1Атака на сервер 2**3**
- **ащита 1**-10K(D),+10$K(D)$,+10K (A)+20K(D),−20$K(D)$,−20K (A)**Защита 2**+15K(D),−15$K(D)$,−15K (A)-5K(D),+5$K(D)$,+5K (A)
- **Решение**:
  Найти оптимальную смешанную стратегию (Nash equilibrium):
- maxx miny (xTPy) где P — платежная матрица, x и y — стратегии игроков.

Методы решения

- **2.1. Линейное/целочисленное программирование**
- **Для задач с линейными ограничениями**:
- **Инструменты**: PuLP, SciPy, Gurobi.
- **Пример**: Оптимизация патчинга уязвимостей

- **2.2. Динамическое программирование**
- **Для многоэтапных решений**:
- **Пример**: Оптимальное обновление системы с учетом зависимостей уязвимостей.

- **2.3. Генетические алгоритмы**
- **Для нелинейных задач**:
- **Пример**: Выбор конфигурации фаервола с 100+ правилами.

- **2.4. Reinforcement Learning**
- **Для адаптивной защиты**:
- **Пример**: ИИ-агент, оптимизирующий правила IDS в реальном времени.

Заключение

- **1. Ключевые направления и методы**

Категория	Основные методы	Применение	Точность/Эффективность
Вероятностные модели	Байесовские сети, CVSS, Марковские цепи	Оценка рисков, прогнозирование атак	75–90%
Детерминированные	Графы атак, Теория игр, Деревья решений	Анализ уязвимостей, оптимизация защиты	Зависит от полноты данных
Формальные методы	BAN-логика, Конечные автоматы	Верификация протоколов, анализ корректности	90–99% (для спецификаций)
Гибридные модели	Нечеткая логика, Агентное моделирование	Оценка рисков IoT, сложных систем	80–85%
ML/AI	LSTM, Random Forest, GAN, BERT	Детектирование аномалий, анализ поведе	85–98%

- **2. Ключевые выводы**
- **Для количественной оценки рисков**:
  - **Лучший выбор**: Байесовские сети + графы атак.
  - **Пример**: Расчет вероятности успешной атаки на банковскую систему с учетом данных IDS.
- **Для защиты критической инфраструктуры (КИИ)**:
  - **Обязательно**: Соответствие ГОСТ Р 57580.1 и Приказу ФСТЭК №239.
  - **Инструменты**: AttackIQ, MaxPatrol для автоматического анализа уязвимостей.
- **Для обработки Big Data в ИБ**:
  - **ML-модели**: LSTM для сетевого трафика, BERT для анализа фишинга.
  - **Проблема**: Требуются XAI-методы (SHAP) для объяснения решений.
- **Тренды 2024–2025**:
  - **Federated Learning** для защиты приватности.
  - **Квантово-устойчивые алгоритмы** (NIST PQC).
  - **Интеграция ИИ в SOC** (например, ИИ-ассистенты для аналитиков).

Заключение

- **Ключевые выводы AI/ML в информационной безопасности**
- **1. Основные направления применения**
- ☑ **Обнаружение угроз**:
- **Аномалии в трафике**: Изолирующий лес, LSTM (точность до 95%).
- **Вредоносное ПО**: Статический (Random Forest) + динамический анализ (LSTM).
- ☑ **Прогнозирование атак**:
- Временные ряды (Prophet + LSTM) → прогноз с точностью 85%.
- Генеративные модели (GAN) → создание тестовых атак для тренировки WAF.
- ☑ **Анализ поведения (UEBA)**:
- Autoencoders для выявления инсайдеров (точность 92%).
- ☑ **NLP для ИБ**:
- BERT + логистическая регрессия → детектирование фишинга (F1-score 98%).
- **2. Критические технологии**
- ◇ **Explainable AI (XAI)**:
- SHAP/LIME для интерпретации решений ИИ (требуется по ГОСТ Р 57580.1).
- Пример: Объяснение блокировок SIEM для регуляторов.
- ◇ **Federated Learning**:
- Обучение моделей без передачи исходных данных → соответствие 152-ФЗ.
- Применение: Банки анализируют мошенничество, не обмениваясь транзакциями.
- ◇ **Квантово-безопасные алгоритмы**:
- Подготовка к постквантовой криптографии (NIST PQC стандарты).
- **3. Ограничения и решения**
- **Проблема** **Решение**
- Нехватка данных Синтетические данные (GAN)
- Санкционные ограничения OpenVINO, TensorFlow RS
- Высокая нагрузка на SOC Автоматизация с AutoML