КАЗАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ (ФИЛИАЛ)
АВТОНОМНОЙ НЕКОММЕРЧЕСКОЙ ОБРАЗОВАТЕЛЬНОЙ
ОРГАНИЗАЦИИ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

АНАЛИЗ УЯЗВИМОСТЕЙ КОМПЬЮТЕРНЫХ СИСТЕМ И СЕТЕЙ

Специальность: 10.05.01 Компьютерная безопасность
Специализация: «Разработка защищенного программного обеспечения»
Формы обучения: очная
Объем дисциплины:
в зачетных единицах: 2 з.е.
в академических часах: 72 ак.ч.

Форма промежуточной аттестации: зачет

2025

Оценочные материалы по дисциплине Анализ уязвимостей компьютерных систем и сетей

обсуждены и рекомендованы к утверждению решением кафедры гуманитарных дисциплин и иностранных языков от 21 марта 2025 г., протокол № 7

одобрены Научно-методическим советом Казанского кооперативного института (филиала) от «2» апреля 2025 г., протокол № 3

СОДЕРЖАНИЕ

Целью оценочных материалов по дисциплине (модулю) (далее –ОМ) является комплексная оценка результатов обучения по дисциплине Анализ уязвимостей компьютерных систем и сетей и установление уровня сформированности компетенций обучающегося.

ОМ предназначены для проведения текущего контроля успеваемости и промежуточной аттестации.

ОМ включают оценочные средства для проведения текущего контроля успеваемости и промежуточной аттестации.

Оценочные материалы разработаны в соответствии с рабочей программой дисциплины Анализ уязвимостей компьютерных систем и сетей.

1. Паспорт оценочных материалов по дисциплине «Анализ уязвимостей компьютерных систем и сетей»

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения	Контролируемые разделы, темы дисциплины	Наименование оценочного средства[1]	
				Текущий контроль	Промежуточная аттестация
1	2	3	4	5	6
ПК-3.4 Способен анализировать уязвимости и угрозы информационной безопасности в компьютерных системах и сетях	ПК-3.4.1 Анализирует компьютерную систему с целью определения уровня защищённости и доверияx	Знать: основные концепции и стандарты информационной безопасности, типы угроз и уязвимостей компьютерных систем, а также методы их выявления и классификации, методологии и инструменты для оценки уровня защищённости и доверия компьютерных систем. Уметь: идентифицировать компоненты компьютерной системы, подлежащие анализу, проводить анализ рисков для компьютерной системы, выявлять уязвимости и определять потенциальные угрозы, использовать специализированные инструменты и программное обеспечение для сканирования, тестирования и анализа безопасности компьютерных систем. Владеть: навыками применения методов и инструментов анализа защищённости компьютерных систем, навыками выявления, оценки и классификации угроз и уязвимостей компьютерных систем, навыками разработки и реализации рекомендаций по повышению	Тема 1. Понятие и классификация уязвимостей программного обеспечения Тема 2. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 3. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 4. Предотвращение уязвимостей на этапе реализации Тема 5. Системы обнаружения и предупреждения компьютерных атак	Реферат (Тема 1-5) Тест (тема 1-5) Домашние задания по практическим занятиям (тема 1-5)	Тест (П 1-10 тестового задания)

		уровня защищённости и доверия компьютерных систем.			
	ПК-3.4.2 Демонстрирует умение проводить анализ безопасности компоненты программныхи программно-аппаратных средств защиты информации в компьютерных системах	Знать: основные подходы и методики анализа безопасности компонентов программных и программно-аппаратных средств защиты информации, типовые виды атак и сценарии компрометации компонент защитных механизмов, методы верификации и аттестации средств защиты информации, современные средства и технологии мониторинга и аудита безопасности компонент защиты информации. Уметь: проводить детальное исследование функциональности и архитектуры компонент программных и аппаратных средств защиты информации, выявлять возможные уязвимости и слабые места в компонентах защитной инфраструктуры, оценивать эффективность используемых средств защиты и степень соответствия установленным требованиям и стандартам. Владеть: навыками практической работы с современными инструментами и методами анализа безопасности программных и аппаратных решений, способностью разрабатывать рекомендации по устранению выявленных недостатков и оптимизации настроек существующих средств защиты, методиками документального	Тема 1. Понятие и классификация уязвимостей программного обеспечения Тема 2. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 3. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 4. Предотвращение уязвимостей на этапе реализации Тема 5. Системы обнаружения и предупреждения компьютерных атак	Реферат (Тема 1-5) Тест (тема 1-5) Домашние задания по практическим занятиям (тема 1-5)	Тест (П 11-20 тестового задания)

		оформления результатов анализа и оценивания степени риска для автоматизированных систем обработки информации.			
	ПК-3.4.3. Формирует предложения по устранению выявленных уязвимостей	Знать: основные подходы и принципы устранения уязвимостей программного и аппаратного обеспечения, рекомендации по выбору оптимальных способов нейтрализации выявленных слабых мест, законодательные требования и регламенты, регламентирующие процесс исправления дефектов и повышение уровня защищенности информационных ресурсов. Уметь: анализировать выявленные уязвимости и оценивать их критичность для функционирования системы, определять наиболее эффективные пути ликвидации выявленных проблем и предотвращения возможных инцидентов, составлять аргументированные предложения по модернизации существующей системы защиты. Владеть: навыками разработки конкретных мероприятий по устранению выявленных уязвимостей, умениями формулировать конкретные рекомендации и технические задания на устранение найденных дефектов, наличием опыта управления проектами по улучшению уровня защиты ИТ-инфраструктуры организаций.	Тема 1. Понятие и классификация уязвимостей программного обеспечения Тема 2. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 3. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности. Тема 4. Предотвращение уязвимостей на этапе реализации Тема 5. Системы обнаружения и предупреждения компьютерных атак	Реферат (Тема 1-5) Тест (тема 1-5) Домашние задания по практическим занятиям (тема 1-5)	Тест (П 21-30 тестового задания)

2. Оценочные материалы по дисциплине «Анализ уязвимостей компьютерных систем и сетей»

2.1 Оценочные материалы для проведения текущего контроля успеваемости

Тематика рефератов

Тема 1. Понятие и классификация уязвимостей программного обеспечения

1. Эволюция понятий «угроза», «уязвимость» и «атака» в контексте кибербезопасности.
2. Сравнительный анализ классификаций уязвимостей информационных систем.
3. Роль и значение базы данных CVE в современном управлении уязвимостями.
4. Банк данных угроз безопасности информации ФСТЭК России: структура, назначение и практическое применение.
5. Стандарт CVSS: анализ алгоритма оценки и его роль в приоритизации устранения уязвимостей.
6. Национальная стандартизация в области анализа уязвимостей: требования ГОСТ Р 56546-2015 и ГОСТ Р 58142-2018.
7. Методология оценки защищенности в соответствии с ГОСТ Р ИСО/МЭК 15408 и ГОСТ Р ИСО/МЭК 18045.
8. Феномен уязвимостей нулевого дня (Zero-Day): методы обнаружения и противодействия.
9. Классификация и жизненный цикл эксплойтов.
10. Анализ наиболее распространенных и критических уязвимостей программного обеспечения за последнее десятилетие.

Тема 2. Актуальные уязвимости современного программного обеспечения. Программные и аппаратные уязвимости

1. Аппаратные уязвимости класса Side-Channel: детальный анализ Meltdown и Spectre.
2. Уязвимости микропрограммного обеспечения: от БИОС/UEFI до контроллеров современных устройств.
3. Оценка рисков, связанных с уязвимостями в программируемых логических интегральных схемах (ПЛИС).
4. Уязвимости в портативных технических средствах (смартфоны, IoT-устройства) и методы защиты.
5. Анализ ландшафта угроз для сетевого и телекоммуникационного оборудования.
6. Уязвимости в средствах защиты информации: парадокс и методы минимизации рисков.
7. Уязвимости в операционных системах: сравнительный анализ Windows,

Linux и macOS.
8. Уязвимости в прикладном программном обеспечении: офисные пакеты, браузеры, системы управления контентом.
9. Специфика выявления и устранения уязвимостей в специальном программном обеспечении.
10. Комплексный подход к управлению аппаратными и программными уязвимостями в корпоративной среде.

Тема 3. Уязвимости этапа проектирования ПО. Методы и инструменты анализа

1. Сравнительный анализ методов выявления уязвимостей на этапе проектирования ПО.
2. Методология тестирования на проникновение (Penetration Testing): этапы, инструменты и правовые аспекты.
3. Современные методы и инструменты разведки (Reconnaissance) для сбора информации об цели.
4. Методы сканирования сетей и систем: от классического сканирования портов до современных техник уклонения.
5. Автоматизированное сканирование уязвимостей: возможности и ограничения.
6. Роль экспертного анализа документации и кода в выявлении архитектурных уязвимостей.
7. Статистические методы в анализе уязвимостей: прогнозирование и оценка рисков.
8. Моделирование атак (Attack Modeling) как инструмент проактивной безопасности.
9. Методы визуализации данных для анализа безопасности сложных информационных систем.
10. Формирование итогового отчета по тестированию на проникновение: структура и рекомендации по устранению уязвимостей.

Тема 4. Предотвращение уязвимостей на этапе реализации

1. Методологии безопасной разработки программного обеспечения (SDLC).
2. Тестирование веб-приложений на безопасность: этапы, методы и инструменты.
3. Межсайтовый скриптинг (XSS): классификация, механизмы атак и методы защиты.
4. SQL-инъекция: принципы работы, современные векторы атак и способы предотвращения.
5. Анализ и предотвращение уязвимостей, связанных с аутентификацией и авторизацией.
6. Уязвимости, связанные с неправильной обработкой входных данных (Input Validation).
7. Роль статического и динамического анализа кода в предотвращении уязвимостей на этапе реализации.

8. Безопасная настройка веб-серверов и сред выполнения как метод предотвращения уязвимостей.
9. Уязвимости компонентов веб-приложений (CMS, фреймворки) и управление зависимостями.
10. Формирование культуры безопасности среди разработчиков для предотвращения уязвимостей.

Тема 5. Системы обнаружения и предупреждения компьютерных атак

1. Принципы построения и классификация систем обнаружения и предупреждения атак (IDS/IPS).
2. Методика тестирования на проникновение в беспроводные сети Wi-Fi.
3. Современные инструменты и техники анализа трафика беспроводных сетей.
4. Уязвимости и методы защиты мобильных устройств на базе iOS и Android.
5. Платформа Kali NetHunter для мобильного тестирования на проникновение: возможности и сценарии применения.
6. Арсенал инструментов анализа уязвимостей в дистрибутиве Kali Linux.
7. Обнаружение и анализ аномалий в работе компьютерных систем и сетей.
8. Роль систем SIEM (Security Information and Event Management) в обнаружении сложных многоэтапных атак.
9. Современные тенденции в развитии средств противодействия сетевым атакам.
10. Построение комплексной системы мониторинга и реагирования на инциденты ИБ на основе результатов анализа уязвимостей.

Критерии оценки выполнения рефератов по учебной дисциплине

Оценка «отлично» выставляется студенту, если тема реферата раскрыта в полной мере и все требования по написанию реферата соблюдены.

Оценка «хорошо» выставляется студенту, если недостаточно раскрыта тема реферата, но все требования по написанию реферата соблюдены.

Оценка «удовлетворительно» выставляется студенту, если недостаточно раскрыта тема реферата, не все требования по написанию реферата соблюдены, присутствуют ошибки и неточности в работе.

Оценка «неудовлетворительно» выставляется студенту, если отсутствует реферат.

Рекомендации по написанию реферата:

Реферат — письменная работа, выполняемая обучающимся в течение определенного срока.

Реферат — краткое точное изложение сущности какого-либо вопроса, темы на основе одной или нескольких книг, монографий или других

первоисточников. Реферат должен содержать основные фактические сведения и выводы по рассматриваемому вопросу.

Реферат отвечает на вопрос — что содержится в данной публикации (публикациях). Реферат — не механический пересказ работы, а изложение ее существа.

Кроме реферирования прочитанной литературы, от обучающегося требуется аргументированное изложение собственных мыслей по рассматриваемому вопросу. Тему реферата предлагает преподаватель или сам обучающийся, в последнем случае она должна быть согласованна с преподавателем.

В реферате нужны развернутые аргументы, рассуждения, сравнения. Материал подается не столько в развитии, сколько в форме констатации или описания.

Содержание реферируемого произведения излагается объективно от имени автора. Если в первичном документе главная мысль сформулирована недостаточно четко, в реферате она должна быть конкретизирована и выделена.

Структура реферата:

1. Титульный лист.

2. После титульного листа на отдельной странице следует оглавление (план, содержание), в котором указаны названия всех разделов (пунктов плана) реферата и номера страниц, указывающие начало этих разделов в тексте реферата.

3. После оглавления следует введение. Объем введения составляет 1,5-2 страницы.

4. Основная часть реферата может иметь одну или несколько глав, состоящих из 2-3 параграфов (подпунктов, разделов) и предполагает осмысленное и логичное изложение главных положений и идей, содержащихся в изученной литературе. В тексте обязательны ссылки на первоисточники. В том случае если цитируется или используется чья-либо неординарная мысль, идея, вывод, приводится какой-либо цифрой материал, таблицу - обязательно сделайте ссылку на того автора у кого вы взяли данный материал.

5. Заключение содержит главные выводы, и итоги из текста основной части, в нем отмечается, как выполнены задачи и достигнуты ли цели, сформулированные во введении.

6. Приложение может включать графики, таблицы, расчеты.

7. Библиография (список литературы) здесь указывается реально использованная для написания реферата литература. Список составляется согласно правилам библиографического описания.

Этапы работы над рефератом.

Работу над рефератом можно условно подразделить на три этапа:

1. Подготовительный этап, включающий изучение предмета исследования;

2. Изложение результатов изучения в виде связного текста;

3. Устное сообщение по теме реферата.

1. Подготовительный этап работы.

Формулировка темы. Подготовительная работа над рефератом начинается с формулировки темы. Тема в концентрированном виде выражает содержание будущего текста, фиксируя как предмет исследования, так и его ожидаемый результат. Для того чтобы работа над рефератом была успешной, необходимо, чтобы тема заключала в себе проблему, скрытый вопрос (даже если наука уже давно дала ответ на этот вопрос, студент, только знакомящийся с соответствующей областью знаний, будет вынужден искать ответ заново, что даст толчок к развитию проблемного, исследовательского мышления).

Поиск источников. Грамотно сформулированная тема зафиксировала предмет изучения; задача обучающегося — найти информацию, относящуюся к данному предмету и разрешить поставленную проблему. Выполнение этой задачи начинается с поиска источников. На этом этапе необходимо вспомнить, как работать с энциклопедиями и энциклопедическими словарями.

Работа с источниками. Работу с источниками надо начинать с ознакомительного чтения, т. е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения. В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Если в книге нет подробного оглавления, следует обратить внимание на предметные и именные указатели.

Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет. Необходимо также проанализировать, какие из утверждений автора носят проблематичный, гипотетический характер и уловить скрытые вопросы.

Понятно, что умение таким образом работать с текстом приходит далеко не сразу. Наилучший способ научиться выделять главное в тексте, улавливать проблематичный характер утверждений, давать оценку авторской позиции — это сравнительное чтение, в ходе которого студент знакомится с различными мнениями по одному и тому же вопросу, сравнивает весомость и доказательность аргументов сторон и делает вывод о наибольшей убедительности той или иной позиции.

Создание конспектов для написания реферата.

Подготовительный этап работы завершается созданием конспектов, фиксирующих основные тезисы и аргументы. Здесь важно вспомнить, что конспекты пишутся на одной стороне листа, с полями и достаточным для исправления и ремарок межстрочным расстоянием (эти правила соблюдаются для удобства редактирования). Если в конспектах приводятся цитаты, то непременно должно быть дано указание на источник (автор, название, выходные данные, № страницы).

По завершении предварительного этапа можно переходить

непосредственно к созданию текста реферата.

2. Создание текста.

Общие требования к тексту.

Текст реферата должен подчиняться определенным требованиям: он должен раскрывать тему, обладать связностью и цельностью.

Раскрытие темы предполагает, что в тексте реферата излагается относящийся к теме материал и предлагаются пути решения содержащейся в теме проблемы; связность текста предполагает смысловую соотносительность отдельных компонентов, а цельность - смысловую законченность текста.

С точки зрения связности все тексты делятся на тексты-констатации и тексты-рассуждения. Тексты-констатации содержат результаты ознакомления с предметом и фиксируют устойчивые и несомненные суждения. В текстах-рассуждениях одни мысли извлекаются из других, некоторые ставятся под сомнение, дается им оценка, выдвигаются различные предположения.

План реферата.

Универсальный план реферата – введение, основной текст и заключение.

Требования к введению.

Во введении аргументируется актуальность исследования, - т. е. выявляется практическое и теоретическое значение данного исследования. Далее констатируется, что сделано в данной области предшественниками; перечисляются положения, которые должны быть обоснованы. Введение может также содержать обзор источников или экспериментальных данных, уточнение исходных понятий и терминов, сведения о методах исследования. Во введении обязательно формулируются цель и задачи реферата.

Объем введения - в среднем около 10% от общего объема реферата.

Основная часть реферата.

Основная часть реферата раскрывает содержание темы. Она наиболее значительна по объему, наиболее значима и ответственна. В ней обосновываются основные тезисы реферата, приводятся развернутые аргументы, предполагаются гипотезы, касающиеся существа обсуждаемого вопроса. Важно проследить, чтобы основная часть не имела форму монолога. Аргументируя собственную позицию, можно и должно анализировать, и оценивать позиции различных исследователей, с чем-то соглашаться, чему-то возражать, кого-то опровергать. Текст основной части делится на главы, параграфы, пункты. План основной части может быть составлен с использованием различных методов группировки материала: классификации (эмпирические исследования), типологии (теоретические исследования), периодизации (исторические исследования).

Заключение.

Заключение — последняя часть научного текста. В ней краткой и сжатой форме излагаются полученные результаты, представляющие собой ответ на главный вопрос исследования. Здесь же могут намечаться и дальнейшие перспективы развития темы. Небольшое по объему сообщение также не может обойтись без заключительной части - пусть это будут две-три фразы. Но в них должен подводиться итог проделанной работы.

Список использованной литературы.

Реферат любого уровня сложности обязательно сопровождается списком используемой литературы. Названия книг в списке располагают по алфавиту с указанием выходных данных использованных книг.

Требования, предъявляемыек оформлению реферата

Объемы рефератов – от 10-18 машинописных страниц. Работа выполняется на одной стороне листа стандартного формата. По обеим сторонам листа оставляются поля размером 35 мм. слева и 15 мм. справа, рекомендуется шрифт 12-14, интервал - 1,5. Все листы реферата должны быть пронумерованы. Каждый вопрос в тексте должен иметь заголовок в точном соответствии с наименованием в плане-оглавлении. При написании и оформлении реферата следует избегать типичных ошибок, например, таких:

– поверхностное изложение основных теоретических вопросов выбранной темы, когда автор не понимает, какие проблемы в тексте являются главными, а какие второстепенными,

– в некоторых случаях проблемы, рассматриваемые в разделах, не раскрывают основных аспектов выбранной для реферата темы,

– дословное переписывание книг, статей, заимствования рефератов из интернета и т. д.

При проверке реферата преподавателем оцениваются:

1. Знания и умения на уровне требований стандарта конкретной дисциплины: знание фактического материала, усвоение общих представлений, понятий, идей.

2. Характеристика реализации цели и задач исследования (новизна и актуальность поставленных в реферате проблем, правильность формулирования цели, определения задач исследования, правильность выбора методов решения задач и реализации цели; соответствие выводов решаемым задачам, поставленной цели, убедительность выводов).

3. Степень обоснованности аргументов и обобщений (полнота, глубина, всесторонность раскрытия темы, логичность и последовательность изложения материала, корректность аргументации и системы доказательств, характер и достоверность примеров, иллюстративного материала, широта кругозора автора, наличие знаний интегрированного характера, способность к обобщению).

4. Качество и ценность полученных результатов (степень завершенности реферативного исследования, спорность или однозначность выводов).

5. Использование литературных источников.

6. Культура письменного изложения материала.

7. Культура оформления материалов работы.

Комплект типовых домашних заданий по практическим занятиям

Тема 1. Понятие и классификация уязвимостей программного обеспечения

Общая инструкция для студента:
Вам необходимо выполнить **один** вариант задания, указанный преподавателем. Используйте открытые источники: базы данных уязвимостей (NVD, CVE Details), официальные сайты производителей ПО, стандарты (CVSS, ГОСТ). Результат оформите в виде отчета объемом **2-3 страницы** с четкой структурой, соответствующей пунктам задания.

Вариант 1

Тема: Анализ уязвимости веб-сервиса через базу CVE.

Задание:

Найдите в базе данных CVE уязвимость **CVE-2021-44228** (Log4Shell). Проведите ее комплексный анализ по следующему плану:

1. **Краткое описание:** Дайте собственную формулировку сути уязвимости, не копируя текст из CVE.
2. **Классификация:** Определите тип уязвимости по следующим критериям:
   - Уровень проявления: код, конфигурация, архитектура?
   - Причина возникновения: входные данные, контроль доступа, логическая ошибка?
3. **Оценка по CVSS v3.1:** Заполните таблицу с обоснованием каждого показателя:

Вектор	Показатель	Выбранное значение	Обоснование
AV	Вектор атаки		
AC	Сложность атаки		
PR	Уровень привилегий		
UI	Вовлечение пользователя		
S	Область воздействия		
C	Влияние на конфиденциальность		
I	Влияние на целостность		
A	Влияние на доступность		

   - Рассчитайте итоговый базовый балл.
4. **Сравнение с базой ФСТЭК:** Найдите данную уязвимость (или ее аналог) в банке данных угроз ФСТЭК России. Сравните приведенную там информацию с данными из CVE (описание, критичность, производитель).
5. **Рекомендации:** Сформулируйте 3-4 конкретных действия для системного администратора по устранению или компенсации данной уязвимости.

Критерии оценки: Полнота анализа, правильность классификации и расчета CVSS, глубина сравнения, практическая ценность рекомендаций.

Вариант 2

Тема: Сравнительный анализ уязвимостей в операционных системах.

Задание:

Сравните две уязвимости: **CVE-2020-0601** (CurveBall) для Windows и **CVE-2021-4034** (PwnKit) для Linux. Проведите анализ по пунктам:

1. **Сводная таблица:** Заполните таблицу для наглядного сравнения:

Параметр	CVE-2020-0601	CVE-2021-4034
Краткое описание		
Ключевой компонент ОС		
Тип уязвимости (уровень проявления)		
Базовый балл CVSS v3.x		
Вектор атаки (Network/Adjacent/Local)		

2. **Глубокий анализ различий:** Проанализируйте, какая из уязвимостей представляет большую угрозу в корпоративной среде. Обоснуйте свой ответ, учитывая:
    - Сложность эксплуатации.
    - Уровень привилегий, необходимый для атаки.
    - Потенциальный ущерб (масштабируемость атаки).
3. **Классификация по ГОСТ Р ИСО/МЭК 15408-2:** Определите класс и семейство уязвимостей в соответствии с данным стандартом (например, "Security Audit", "Protection of the TSF" и т.д.).
4. **Вывод:** Сформулируйте вывод о том, являются ли рассмотренные уязвимости типичными для своих классов ОС, и какие общие меры защиты могут их предотвратить.

Критерии оценки: Качество сравнительного анализа, глубина аргументации, корректность классификации по ГОСТ, логичность вывода.

Вариант 3

Тема: Исследование "уязвимости нулевого дня".

Задание:

На примере уязвимости **CVE-2022-41080** (OMIGOD) или любой другой актуальной "zero-day" на ваш выбор, исследуйте ее жизненный цикл.

1. **Хронология событий:** Восстановите хронологию по следующему плану:
    - Дата первого обнаружения/эксплуатации "в дикой природе" (in the wild).
    - Дата публикации CVE и выпуска патча производителем.
    - Дата включения информации в банк данных ФСТЭК России (если есть).
2. **Анализ информационного освещения:** Сравните, как информация об уязвимости подавалась в CVE, в новостных лентах (например, The Hacker News) и в технических блогах (например, MSRC). Выделите 3 ключевых различия в подаче информации для разных аудиторий.
3. **Классификация эксплойта:** К какому типу относится эксплойт, использующий эту уязвимость (Remote Code Execution, Privilege Escalation, DoS)? Является ли он публично доступным (например, в Metasploit)?

4. **Оценка угрозы по CVSS:** Рассчитайте не только базовый, но и временной (Temporal) балл CVSS для двух моментов:
    - На момент обнаружения (патча нет, эксплойты активны).
    - На текущий момент (патч выпущен).
    - Объясните, почему баллы изменились (или не изменились).
5. **Рекомендации для CISO:** Составьте памятку для руководителя службы информационной безопасности на период между обнаружением "zero-day" и выпуском патча. Что необходимо сделать в первую очередь?

Критерии оценки: Точность хронологии, качество сравнительного анализа источников, понимание жизненного цикла уязвимости, практическая ценность рекомендаций.

Вариант 4: Анализ уязвимости **CVE-2017-0144** (EternalBlue). Сфокусируйтесь на ее историческом значении, классификации как многофакторной уязвимости (код ОС + сетевая служба) и анализу причин ее массовой эксплуатации.

Вариант 5: Исследование уязвимости в браузере (например, **CVE-2023-4863** в Chrome). Проанализируйте ее через призму OWASP Top 10, оцените риск для рядового пользователя и сформулируйте рекомендации по безопасной настройке браузера.

Вариант 6: Сравнение описания одной и той же уязвимости (например, **CVE-2019-19781** в Citrix) в базе CVE и в банке данных ФСТЭК. Проанализируйте расхождения в терминологии, оценке критичности и рекомендуемых действиях. Объясните возможные причины этих расхождений.

Вариант 7: Анализ организационной уязвимости. На примере инцидента, связанного с утечкой данных из-за неправильно настроенного S3-бакета Amazon, докажите, что уязвимость носит организационно-технический характер. Предложите регламент по предотвращению подобных инцидентов.

Вариант 8: Детальный разбор уязвимости типа "Insecure Deserialization" (**CVE-2017-9805** в Struts2). Объясните механизм ее работы на понятном языке, классифицируйте и постройте подробный сценарий атаки.

Вариант 9: Анализ уязвимости в системе виртуализации (**CVE-2021-21972** в vSphere). Оцените ее критичность для ЦОД, определите вектор атаки и область воздействия (S) в CVSS. Разработайте план экстренных действий для администратора.

Вариант 10: Комплексный анализ "цепочки уязвимостей". На примере любого известного инцидента (например, атака на SolarWinds) выделите 2-3 ключевые уязвимости, которые были использованы на разных этапах. Постройте схему атаки и для каждой уязвимости укажите ее тип и классификацию по CVSS.

Тема 2. Актуальные уязвимости современного программного обеспечения. Программные и аппаратные уязвимости.

Общая инструкция для студента:
Вам необходимо выполнить **один** вариант задания, указанный преподавателем.

Используйте открытые источники: базы данных уязвимостей (NVD, CVE Details), технические отчеты производителей, официальные сайты. Результат оформите в виде отчета объемом **2-3 страницы** с четкой структурой, соответствующей пунктам задания.

Вариант 1

Тема: Исследование аппаратных уязвимостей класса Side-Channel.

Задание:

Проведите сравнительный анализ уязвимостей **Meltdown (CVE-2017-5754)** и **Spectre (CVE-2017-5753, CVE-2017-5715)**.

1. **Принцип работы:** Опишите на концептуальном уровне (без углубления в архитектуру CPU), в чем заключается фундаментальное различие в механизме эксплуатации уязвимостей Meltdown и Spectre.
2. **Сравнительная таблица:** Заполните таблицу для наглядного сравнения:

Критерий	Meltdown	Spectre
Подверженные архитектуры процессоров		
Уровень изоляции, который нарушается		
Тип данных, к которым возможен доступ		
Сложность создания рабочего эксплойта		

3. **Анализ исправлений:** Изучите, какие основные методы устранения (микрокод, патчи ОС) были предложены для каждой уязвимости. Как эти исправления повлияли на производительность систем?
4. **Практические последствия:** Сформулируйте вывод: какая из этих уязвимостей представляет большую долгосрочную угрозу и почему? Аргументируйте свою позицию.

Критерии оценки: Глубина понимания принципов работы уязвимостей, точность сравнительного анализа, качество анализа исправлений и их последствий, обоснованность вывода.

Вариант 2

Тема: Анализ уязвимостей микропрограммного обеспечения.

Задание:

Исследуйте уязвимость **CVE-2023-1011/CVE-2023-1012** (уязвимости в микрокоде Trusted Platform Module - TPM).

1. **Краткое описание:** Дайте собственную формулировку сути уязвимости, объяснив, что такое TPM и почему уязвимость на этом уровне критична.
2. **Классификация и воздействие:** Определите:
    - Тип уязвимости: аппаратная, программная или микропрограммная?
    - К каким последствиям может привести эксплуатация данной уязвимости с точки зрения цепочки доверия (Chain of Trust)?
3. **Сценарий атаки:** Опишите гипотетический многоэтапный сценарий атаки, где эксплуатация данной уязвимости в TPM является ключевым звеном для обхода системы безопасной загрузки (Secure Boot).
4. **Рекомендации по устранению:** Разработайте инструкцию для системного администратора по проверке и обновлению микрокода TPM

на компьютерах в корпоративной сети. Укажите, с какими сложностями можно столкнуться.

Критерии оценки: Понимание роли TPM, точность классификации, реалистичность сценария атаки, практическая применимость и полнота рекомендаций.

Вариант 3

Тема: Исследование уязвимостей в сетевом оборудовании.

Задание:

Проанализируйте критическую уязвимость в сетевом оборудовании, например, **CVE-2023-20198** в веб-интерфейсе маршрутизаторов Cisco.

1. **Технический анализ:** Опишите механизм уязвимости. Какие компоненты веб-интерфейса задействованы? Требуются ли аутентификация и специальные привилегии для эксплуатации?
2. **Оценка риска:** Рассчитайте базовый балл CVSS v3.1 для этой уязвимости. Обоснуйте значения самых критичных показателей (например, Vector Attack, Impact).
3. **Потенциальный ущерб:** Опишите три различных сценария ущерба для организации, которая не устранила данную уязвимость вовремя (например, перехват трафика, создание ботнета, шпионаж).
4. **Практические шаги:** Составьте план экстренных действий для сетевого администратора, включающий:
    - Этап идентификации (как найти уязвимые устройства).
    - Этап устранения (где найти и как применить патч).
    - Этап компенсации (что делать, если патч нельзя установить немедленно).

Критерии оценки: Техническая глубина анализа, корректность расчета CVSS, разнообразие и реалистичность сценариев ущерба, полнота и структурированность плана действий.

Вариант 4

Тема: Анализ уязвимостей в средствах защиты информации.

Задание:

Исследуйте уязвимость в антивирусном программном обеспечении, например, **CVE-2019-1367** в Microsoft Defender.

1. **Парадокс уязвимости:** Объясните, почему уязвимости в средствах защиты информации (СЗИ) представляют особую опасность по сравнению с уязвимостями в обычном ПО.
2. **Механизм уязвимости:** Опишите, в каком компоненте антивирусного ядра (сканирование, эвристический анализ, обработка архивов) была обнаружена данная уязвимость и в чем она заключалась.
3. **Сценарий атаки "День нуля":** Разработайте гипотетический сценарий, при котором злоумышленник использует данную уязвимость в антивирусе для отключения защиты на целевой системе до момента выхода патча.

4. **Рекомендации по выбору СЗИ:** На основе анализа данной уязвимости сформулируйте 3-4 критерия, которые должна учитывать компания при выборе антивирусного решения, чтобы минимизировать риски от подобных уязвимостей.

Критерии оценки: Понимание специфики СЗИ, точность технического анализа, оригинальность и реалистичность сценария, практическая ценность рекомендаций.

Вариант 5

Тема: Исследование уязвимостей в IoT-устройствах.

Задание:

Проанализируйте типичные уязвимости для IP-камер или домашних маршрутизаторов (например, уязвимость **CVE-2021-35394** в веб-интерфейсе роутеров).

1. **Анализ причин:** Выявите и опишите три основные причины, по которым устройства IoT особенно уязвимы (например, использование стандартных учетных данных, отсутствие OTA-обновлений).
2. **Классификация:** Для выбранной уязвимости определите ее тип (например, уязвимость конфигурации, бэкдор, RCE) и оцените ее по CVSS.
3. **Модель угроз:** Постройте простую модель угроз для умного дома, где уязвимый роутер является точкой входа. Опишите, как атака на роутер может привести к компрометации других устройств (камер, умных ламп).
4. **Производителю на заметку:** Составьте памятку для производителя IoT-устройств из 5 пунктов, содержащую ключевые меры безопасности, которые необходимо внедрить на этапе проектирования и производства.

Критерии оценки: Глубина анализа коренных причин, точность классификации, логичность построения модели угроз, практическая ценность памятки для производителя.

Вариант 6

Тема: Сравнительный анализ уязвимостей в облачной инфраструктуре.

Задание:

Сравните уязвимость конфигурации (**CVE-2022-28376** - уязвимость в облачном сервисе) и уязвимость кода в гипервизоре (**CVE-2018-6111** в KVM).

1. **Сравнительная таблица:** Заполните таблицу:

Параметр	Уязвимость конфигурации (CVE-2022-28376)	Уязвимость гипервизора (CVE-2018-6111)
Уровень воздействия (изоляция арендаторов)		
Ответственность за устранение (клиент/провайдер)		

Сложность эксплуатации в многопользовательской среде		
Потенциальный масштаб инцидента		

2. **Модель ответственности:** Объясните, в соответствии с моделью ответственности cloud (IaaS, PaaS, SaaS), кто отвечает за устранение каждой из этих уязвимостей – провайдер облачных услуг или клиент?
3. **Вывод:** Сформулируйте вывод о том, уязвимости какого типа (конфигурации или кода) представляют на сегодня большую угрозу для облачной безопасности и почему.

Критерии оценки: Качество сравнительного анализа, понимание модели ответственности в cloud, обоснованность и глубина вывода.

Вариант 7

Тема: Анализ уязвимостей в системах управления промышленными процессами (ICS/SCADA).

Задание:

Исследуйте известную уязвимость в SCADA-системе, например, **CVE-2015-5374** в Siemens SIMATIC S7.

1. **Специфика ICS/SCADA:** Объясните, чем уязвимости в SCADA-системах fundamentally отличаются от уязвимостей в офисном ПО с точки зрения потенциального ущерба и требований к доступности.
2. **Технический разбор:** Опишите протокол, в котором была найдена уязвимость, и механизм ее эксплуатации. Почему такие системы часто не могут быть быстро пропатчены?
3. **Сценарий кибератаки на АСУ ТП:** Опишите гипотетический сценарий, где эксплуатация данной уязвимости приводит к физическим последствиям (остановка производства, повреждение оборудования).
4. **Компенсирующие меры:** Предложите 3-4 компенсирующие меры защиты для такой системы, которые можно применить, если установка патча невозможна по производственным причинам.

Критерии оценки: Понимание специфики ICS/SCADA, точность технического разбора, реалистичность и драматизм сценария, практическая ценность компенсирующих мер.

Вариант 8

Тема: Исследование уязвимостей в протоколах беспроводной связи.

Задание:

Проанализируйте уязвимость **CVE-2017-13080** (атака KRACK - Key Reinstallation Attacks) против протокола WPA2.

1. **Принцип атаки:** Объясните на концептуальном уровне, в чем заключается уязвимость четырехстороннего handshake в WPA2, которую эксплуатирует KRACK.

2. **Классификация:** К какому типу уязвимостей относится KRACK (уязвимость реализации, протокола, конфигурации)? Свой ответ обоснуйте.
3. **Практическая демонстрация:** Опишите, что может увидеть пользователь (или администратор) во время успешной атаки KRACK? Меняются ли индикаторы подключения к сети?
4. **Эволюция защиты:** Сравните уязвимость WPA2 (KRACK) с безопасностью нового протокола WPA3. Как WPA3 устраняет данную проблему?

Критерии оценки: Четкость объяснения принципа атаки, точность классификации и ее обоснование, понимание практических аспектов, глубина сравнения с WPA3.

Вариант 9

Тема: Анализ уязвимостей в контейнеризации и оркестрации.

Задание:

Исследуйте уязвимость **CVE-2019-5736** в runc (контейнерной среде выполнения).

1. **Контекст:** Объясните, что такое runc и какую роль он играет в экосистеме Docker и Kubernetes.
2. **Механизм побега (Escape):** Опишите механизм уязвимости, который позволяет контейнеру с правами root скомпрометировать хост-систему.
3. **Оценка риска для кластера Kubernetes:** Спрогнозируйте, как эксплуатация этой уязвимости в одном поде (pod) может привести к компрометации всего кластера.
4. **Рекомендации по харденингу:** Составьте список из 5 рекомендаций по безопасной настройке контейнеров и оркестратора, которые минимизируют риск эксплуатации подобных уязвимостей даже до выхода патчей.

Критерии оценки: Понимание архитектуры контейнеризации, точность описания механизма escape, глубина анализа рисков для кластера, практическая ценность рекомендаций по харденингу.

Вариант 10

Тема: Комплексный анализ цепочки эксплуатации уязвимостей.

Задание:

На примере любого известного вредоносного ПО (например, **WannaCry** или **NotPetya**) проанализируйте цепочку уязвимостей, которые оно использовало.

1. **Карта атаки:** Визуализируйте цепочку атаки в виде блок-схемы, где каждый шаг – это эксплуатация конкретной уязвимости (CVE).
2. **Анализ звеньев цепи:** Для каждой уязвимости в цепочке укажите:
    - Ее тип (аппаратная, ПО, сетевая, организационная).
    - Базовый балл CVSS.

- Какую роль она играла в атаке (первоначальный доступ, распространение, выполнение).

3. **Ключевое звено:** Определите, устранение какой уязвимости в цепочке было бы наиболее эффективным для срыва всей атаки. Свой выбор обоснуйте.
4. **Уроки для защиты:** Сформулируйте 3 основных вывода для построения системы защиты, которые следуют из анализа данной цепочки эксплуатации.

Критерии оценки: Полнота и наглядность карты атаки, точность анализа каждого звена, обоснованность выбора ключевого звена, глубина и практическая ценность выводов.

Тема 3. Уязвимости этапа проектирования программного обеспечения. Методы и инструменты анализа уязвимостей в сфере информационной безопасности.

Общая инструкция для студенту:
Вам необходимо выполнить **один** вариант задания. Используйте специализированный дистрибутив Kali Linux, документацию к инструментам и учебные стенды (например, Metasploitable, DVWA). Отчет должен содержать скриншоты, команды и их вывод, а также развернутые выводы по каждому этапу.

Вариант 1

Тема: Проведение пассивной и активной разведки против целевого домена.

Задание:

Для целевого домена (например, example.com, по согласованию с преподавателем) проведите комплексную разведку.

1. **Пассивная разведка (OSINT):**
   - Используя whois, определите регистратора домена, дату создания и контактные данные (замаскируйте их в отчете).
   - С помощью theHarvester соберите email-адреса и поддомены, связанные с целевым доменом.
2. **Активная разведка:**
   - Используя nmap, выполните SYN-сканирование для определения открытых портов и версий служб на целевом хосте.
   - С помощью dnsrecon или dig проведите перебор поддоменов (brute-force).
3. **Анализ и выводы:** На основе собранных данных составьте профиль цели: предположите тип организации, используемые технологии (веб-сервер, ОС) и потенциальные векторы для дальнейшей атаки. Какая информация, собранная на этом этапе, оказалась наиболее ценной?

Критерии оценки: Полнота собранной информации, корректное использование инструментов, глубина анализа и обоснованность выводов о потенциальных векторах атаки.

Вариант 2

Тема: Сканирование сети и уязвимостей с помощью автоматизированных инструментов.

Задание:

Для целевой сети (например, 192.168.1.0/24, по согласованию с преподавателем) проведите сканирование.

1. **Сканирование сети:**
   - Используя nmap с различными типами сканирования (-sS, -sT), определите "живые" хосты в сети и составьте их список.
2. **Сканирование уязвимостей:**
   - Выберите один "живой" хост и проведите его углубленное сканирование с помощью nmap с использованием скриптов NSE (-sC).
   - Запустите сканирование этого хоста с помощью OpenVAS или Nessus.
3. **Сравнительный анализ:** Сравните результаты, полученные с помощью Nmap NSE и OpenVAS/Nessus. Какой инструмент предоставил более полную информацию об уязвимостях? Составьте таблицу с 3-5 наиболее критичными уязвимостями, обнаруженными каждым инструментом.
4. **Рекомендации:** Сформулируйте первоочередные рекомендации по устранению найденных уязвимостей для системного администратора.

Критерии оценки: Корректность использования инструментов, точность сравнения результатов, глубина анализа критичности уязвимостей, практическая ценность рекомендаций.

Вариант 3

Тема: Моделирование атаки с использованием фреймворка Metasploit.

Задание:

На учебном стенде (Metasploitable2/3) смоделируйте атаку с получением доступа к системе.

1. **Выбор и настройка эксплойта:**
   - На основе данных сканирования выберите уязвимую службу на целевом хосте.
   - В msfconsole подберите и настройте соответствующий эксплойт (укажите RHOSTS, RPORT и т.д.).
2. **Выбор и настройка полезной нагрузки (Payload):**
   - Подберите полезную нагрузку, которая обеспечит обратное соединение (reverse shell) с атакующей машиной.
   - Настройте параметры полезной нагрузки (LHOST, LPORT).
3. **Проведение атаки:**

- Запустите эксплойт. Предоставьте скриншоты, подтверждающие успешное получение доступа (сессия Meterpreter или командной оболочки).

4. **Анализ результатов:** Опишите, какие именно действия выполняет данный эксплойт "под капотом". Какие последствия для целевой системы повлекла бы успешная атака в реальных условиях?

Критерии оценки: Правильность выбора и настройки эксплойта и полезной нагрузки, успешное получение доступа, глубина анализа механизма работы эксплойта и его последствий.

Вариант 4

Тема: Анализ веб-приложения на уязвимости с помощью OWASP ZAP.

Задание:

Для уязвимого веб-приложения (DVWA или иного, по согласованию с преподавателем) проведите анализ безопасности.

1. **Автоматизированное сканирование:**
   - Настройте OWASP ZAP в качестве прокси.
   - Проведите "Атаку" (Attack) на целевое приложение после его автоматического сканирования (Spider).
2. **Ручное тестирование:**
   - Вручную исследуйте параметры приложения (например, форму входа), пытаясь найти уязвимости, которые не обнаружило автоматическое сканирование (например, логические уязвимости).
3. **Классификация и оценка:**
   - Составьте отчет в OWASP ZAP. Сгруппируйте найденные уязвимости по степени риска (High, Medium, Low).
   - Для каждой уязвимости уровня High предложите конкретное исправление на уровне кода или конфигурации.
4. **Сравнение методов:** Сделайте вывод о том, какие уязвимости эффективнее обнаруживаются автоматическим сканированием, а какие требуют обязательного ручного тестирования.

Критерии оценки: Умение работать с OWASP ZAP, качество ручного тестирования, точность классификации и оценки уязвимостей, глубина сравнения методов.

Вариант 5

Тема: Тестирование на проникновение в беспроводную сеть (в контролируемой среде).

Задание:

В специально созданной для тестирования беспроводной сети проведите аудит безопасности.

1. **Мониторинг эфира:**
   - Используя airodump-ng, выполните мониторинг беспроводных сетей. Определите BSSID, канал и метод шифрования целевой сети.
2. **Перехват handshake:**

- Зафиксируйте процесс аутентификации клиента в сети (перехватите handshake).

3. **Взлом ключа:**
   - Используя aircrack-ng и словарь, попытайтесь подобрать ключ WPA/WPA2 на основе перехваченного handshake.
4. **Анализ и рекомендации:** Опишите, от чего зависит успех атаки брутфорсом? Сформулируйте рекомендации по настройке беспроводной сети для защиты от подобных атак (минимум 4 пункта, выходящие за рамки "использовать сложный пароль").

Критерии оценки: Корректность использования инструментов, понимание процесса аутентификации в Wi-Fi, осознание факторов, влияющих на взлом, практическая ценность рекомендаций.

Вариант 6

Тема: Анализ защищенности с помощью статического анализа кода (SAST).

Задание:

Проанализируйте предоставленный фрагмент кода на Python/Java с помощью инструментов статического анализа (например, Bandit для Python, SpotBugs для Java).

1. **Настройка и запуск:**
   - Установите и настройте выбранный инструмент.
   - Запустите анализ предоставленного файла с кодом.
2. **Интерпретация результатов:**
   - Составьте таблицу с найденными предупреждениями (issues), указанием их типа (например, "SQL injection", "Hardcoded password") и степени серьезности.
3. **Исправление кода:**
   - Выберите 3 предупреждения с высоким уровнем серьезности и исправьте соответствующие участки кода. Предоставьте код "до" и "после".
4. **Оценка инструмента:** Каковы сильные и слабые стороны использованного вами SAST-инструмента? Какие типы уязвимостей он может пропустить?

Критерии оценки: Умение работать с SAST-инструментом, точность интерпретации результатов, качество исправлений, критическая оценка возможностей инструмента.

Вариант 7

Тема: Социальная инженерия и создание фишинговой кампании.

Задание:

В изолированной лабораторной среде смоделируйте фишинговую атаку с использованием The Social-Engineer Toolkit (SET).

1. **Создание фишинговой страницы:**

- С помощью SET создайте клон страницы входа в популярную социальную сеть или корпоративный портал.

2. **Выбор метода доставки:**
   - Настройте метод доставки фишинговой ссылки (например, сгенерируйте ссылку и QR-код).
3. **Анализ механизма:**
   - Опишите, как SET перехватывает введенные учетные данные. Какие технические методы для этого используются?
4. **Разработка контрмер:** На основе проведенного моделирования разработайте памятку для сотрудников компании из 5 пунктов по распознаванию фишинговых писем и веб-страниц.

Критерии оценки: Корректность работы с SET, понимание технических механизмов фишинга, практическая ценность и четкость разработанной памятки.

Вариант 8

Тема: Повышение привилегий в операционной системе Windows.

Задание:

Получив первоначальный доступ к Windows-системе (на учебном стенде) с правами пользователя, добейтесь повышения привилегий до уровня NT AUTHORITY\SYSTEM.

1. **Сбор информации:**
   - Используя встроенные средства Windows и утилиты типа winpeas или windows-exploit-suggester, соберите информацию о системе: версия ОС, установленные патчи, запущенные службы, небезопасные права доступа к файлам/реестру.
2. **Выбор вектора атаки:**
   - На основе собранной информации выберите и обоснуйте метод повышения привилегий (например, эксплуатация уязвимости в ядре, злоупотребление службой, неправильные права на исполняемый файл).
3. **Эксплуатация:**
   - Реализуйте выбранный метод, получив права SYSTEM. Предоставьте скриншоты, подтверждающие успех.
4. **Рекомендации по харденингу:** Сформулируйте 3 рекомендации по настройке политик безопасности Windows, которые предотвратили бы использованный вами метод эскалации привилегий.

Критерии оценки: Полнота сбора информации, обоснованность выбора вектора атаки, успешность эксплуатации, практическая ценность рекомендаций по харденингу.

Вариант 9

Тема: Анализ вредоносного ПО с помощью песочницы.

Задание:

Используя общедоступную песочницу (например, Any.run, Hybrid Analysis), проанализируйте предоставленный подозрительный файл.

1. **Статический анализ:**
    - Загрузите файл в песочницу. Изучите результаты статического анализа: хэши, импортируемые библиотеки, строки, потенциальные IOC (Indicators of Compromise).
2. **Динамический анализ:**
    - Проанализируйте отчет о поведении файла в песочнице: какие процессы были созданы, какие изменения внесены в файловую систему и реестр, с какими сетевыми адресами осуществлялось соединение?
3. **Определение угрозы:** На основе анализа классифицируйте угрозу (троян, ransomware, майнер и т.д.). Каковы были бы последствия его запуска в реальной системе?
4. **Составление IoC:** Составьте список индикаторов компрометации (IoC) для этого образца, которые можно было бы использовать для его обнаружения в корпоративной сети.

Критерии оценки: Умение работать с отчетом песочницы, точность классификации угрозы, глубина анализа поведения, корректность составления списка IoC.

Вариант 10

Тема: Создание отчета о тестировании на проникновение.

Задание:

На основе результатов, полученных при выполнении любого из предыдущих заданий (например, Варианта 2 или 3), составьте профессиональный отчет о тестировании на проникновение.

1. **Структура отчета:** Отчет должен содержать следующие разделы:
    - Резюме для руководства (не техническое, на 1/2 страницы).
    - Введение (цели, объем тестирования, методология).
    - Информация о целевой системе.
    - Результаты (таблица с уязвимостями, их критичностью и CVSS).
    - Детальное описание 2-3 наиболее критичных уязвимостей (включая шаги для воспроизведения, скриншоты, последствия).
    - Рекомендации по устранению для каждой уязвимости.
    - Приложения (использованные команды, инструменты).
2. **Язык и стиль:** Резюме должно быть написано для нетехнического руководства, а техническая часть – для системных администраторов.
3. **Визуализация:** Используйте таблицы, диаграммы и скриншоты для наглядности.

Критерии оценки: Соответствие профессиональной структуре, ясность и адресность изложения для разной аудитории, качество визуализации, практическая ценность рекомендаций.

Тема 4. Предотвращение уязвимостей на этапе реализации

Общая инструкция для студента:
Вам необходимо выполнить **один** вариант задания. Каждое задание направлено на анализ, обнаружение и исправление уязвимостей в исходном коде. Используйте знания о безопасном программировании, стандарты OWASP и инструменты статического/динамического анализа. Результат оформите в виде отчета, содержащего исходный код, описание уязвимости, исправленный код и обоснование внесенных изменений.

Вариант 1

Тема: Анализ и исправление уязвимости SQL-инъекции.

Задание:

Дан фрагмент кода на PHP, который обрабатывает аутентификацию пользователя.

```
php
$username = $_POST['username'];
$password = $_POST['password'];
$sql = "SELECT * FROM users WHERE username='$username' AND password='$password'";
$result = mysqli_query($conn, $sql);
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость SQL-инъекции в данном коде. Продемонстрируйте, какие данные можно ввести в поля username и password, чтобы обойти аутентификацию.
2. **Исправление кода:** Перепишите код, используя подготовленные выражения (Prepared Statements) с параметризованными запросами.
3. **Обоснование:** Объясните, почему использование подготовленных выражений является эффективной мерой защиты против SQL-инъекций.
4. **Дополнительные меры:** Предложите еще 2 дополнительных меры (например, на уровне конфигурации СУБД или логики приложения), которые повысят безопасность данного функционала.

Критерии оценки: Корректное объяснение уязвимости, успешное исправление кода с использованием подготовленных выражений, глубина обоснования и практическая ценность дополнительных мер.

Вариант 2

Тема: Анализ и исправление уязвимости Межсайтового скриптинга (XSS).

Задание:

Дан фрагмент кода на JavaScript (Node.js с использованием Express), который отображает комментарии пользователя.

```
javascript
app.get('/comment', (req, res) => {
    const userComment = req.query.comment;
    res.send('<div>' + userComment + '</div>');
```

});

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость XSS (Reflected). Продемонстрируйте пример вредоносной строки, которая может быть передана в параметре comment для выполнения скрипта в браузере другого пользователя.
2. **Исправление кода:** Перепишите код, чтобы обезопасить вывод данных. Используйте экранирование (escaping) специальных символов HTML.
3. **Обоснование:** Объясните, как экранирование предотвращает исполнение вредоносного скрипта.
4. **Контекст вывода:** Объясните, почему методы защиты от XSS могут различаться в зависимости от контекста вывода данных (HTML, атрибут, JavaScript).

Критерии оценки: Корректное объяснение уязвимости, успешное исправление кода с использованием экранирования, понимание различий в контекстах вывода.

Вариант 3

Тема: Анализ и исправление уязвимости Хранимого XSS.

Задание:

Дан фрагмент кода на Python (с использованием Flask), который сохраняет и отображает сообщения в гостевой книге.

python

```
@app.route('/add_message', methods=['POST'])
def add_message():
    message = request.form['message']
    # Сохранение сообщения в базу данных
    save_message_to_db(message)
    return redirect('/guestbook')

@app.route('/guestbook')
def guestbook():
    messages = get_messages_from_db()
    return render_template('guestbook.html', messages=messages)
```

Шаблон guestbook.html*:*

html

```
{% for msg in messages %}
    <p>{{ msg.text | safe }}</p>
{% endfor %}
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость Хранимого XSS. Почему использование фильтра safe в шаблоне опасно?
2. **Исправление кода:** Устраните уязвимость, убрав фильтр safe и обеспечив экранирование выводимых данных на уровне шаблонизатора.
3. **Сравнение уязвимостей:** Сравните Reflected XSS и Stored XSS по следующим параметрам: механизм заражения, потенциальный масштаб ущерба, сложность обнаружения для жертвы.

4. **Санитизация vs Экранирование:** В каком случае для защиты от XSS предпочтительнее использовать санитизацию (очистку) входных данных, а в каком — экранирование на выходе?

Критерии оценки: Корректное объяснение уязвимости, успешное исправление кода, глубина сравнительного анализа, понимание различий между санитизацией и экранированием.

Вариант 4

Тема: Анализ и исправление уязвимости Небезопасной десериализации.

Задание:

Дан фрагмент кода на Java, который десериализует объект, полученный из ненадежного источника.

java

```
FileInputStream fileIn = new FileInputStream("data.ser");
ObjectInputStream in = new ObjectInputStream(fileIn);
Object obj = in.readObject();
in.close();
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается риск небезопасной десериализации. Какие типы атак возможны при эксплуатации этой уязвимости (например, RCE, DoS)?
2. **Исправление кода:** Предложите и реализуйте безопасную альтернативу для обмена данными (например, использование JSON, XML с валидацией схемы или безопасных библиотек сериализации).
3. **Контрмеры:** Если бы десериализация была необходима, какие контрмеры можно было бы применить для снижения риска (например, белые списки классов, валидация данных)?
4. **Обоснование выбора:** Объясните, почему выбранный вами формат (JSON/XML) является более безопасным по сравнению с нативной бинарной десериализацией.

Критерии оценки: Понимание рисков небезопасной десериализации, корректное предложение и реализация безопасной альтернативы, глубина анализа контрмер.

Вариант 5

Тема: Анализ и исправление уязвимости Межсайтовой подделки запроса (CSRF).

Задание:

Дан фрагмент кода HTML-формы, которая изменяет email-адрес пользователя (без защиты от CSRF).

html

```
<form action="/change_email" method="POST">
   <input type="email" name="new_email">
   <input type="submit" value="Change Email">
</form>
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость CSRF. Опишите сценарий, при котором злоумышленник может заставить пользователя unintentionally отправить эту форму.
2. **Исправление кода:** Модифицируйте форму и серверную логику, реализовав защиту с использованием CSRF-токенов. Покажите код для генерации токена на сервере, его подстановки в форму и проверки на стороне сервера.
3. **Альтернативные методы:** Назовите и кратко охарактеризуйте еще два метода защиты от CSRF (помимо CSRF-токенов).
4. **SameSite Cookies:** Объясните, как атрибут SameSite у cookies может помочь в защите от CSRF-атак.

Критерии оценки: Корректное объяснение уязвимости и сценария атаки, успешная реализация защиты с использованием CSRF-токенов, знание альтернативных методов защиты.

Вариант 6

Тема: Анализ и исправление уязвимости Небезопасных прямых ссылок на объекты (IDOR).

Задание:

Дан фрагмент кода на Python (Flask), который предоставляет доступ к документам пользователя.

```
python
@app.route('/documents/<int:document_id>')
def get_document(document_id):
    document = get_document_by_id(document_id)
    return send_file(document.path)
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость IDOR. Как злоумышленник может получить доступ к чужим документам?
2. **Исправление кода:** Перепишите код, добавив проверку прав доступа. Предполагается, что у функции get_document_by_id нет встроенной проверки, а у аутентифицированного пользователя есть метод user.has_access_to(document).
3. **Принцип минимальных привилегий:** Как принцип минимальных привилегий (Least Privilege) связан с предотвращением IDOR? Дайте развернутый ответ.
4. **Глубокая защита:** Предложите архитектурное решение (на уровне проектирования системы), которое минимизировало бы риск появления IDOR-уязвимостей.

Критерии оценки: Корректное объяснение уязвимости, успешная реализация проверки прав доступа, глубина понимания принципа минимальных привилегий, практическая ценность архитектурного предложения.

Вариант 7

Тема: Анализ и исправление уязвимости Обхода путей (Path Traversal).

Задание:

Дан фрагмент кода на Node.js (Express), который позволяет пользователю скачивать файлы.

javascript

```
app.get('/download', (req, res) => {
   const filename = req.query.file;
   const filePath = `/uploads/${filename}`;
   res.download(filePath);
});
```

1. **Обнаружение уязвимости:** Объясните, в чем заключается уязвимость Path Traversal. Продемонстрируйте, как можно передать в параметре file значение, которое позволит скачать системный файл (например, /etc/passwd).
2. **Исправление кода:** Перепишите код, добавив валидацию параметра filename. Используйте метод, который нормализует путь и проверяет, находится ли итоговый путь внутри разрешенной директории (/uploads).
3. **Белый vs черный список:** Почему для валидации путей предпочтительнее использовать белый список разрешенных символов/паттернов, а не черный список запрещенных?
4. **Контекст выполнения:** Объясните, почему уязвимость Path Traversal может привести к RCE (Remote Code Execution) в определенных условиях?

Критерии оценки: Корректное объяснение уязвимости, успешная реализация валидации пути, понимание преимуществ белого списка, глубина анализа потенциального воздействия.

Вариант 8

Тема: Анализ и исправление уязвимостей в механизме аутентификации.

Задание:

Дан фрагмент кода на PHP, реализующий самодельную аутентификацию.

php

```
if ($_POST['password'] == $user->password) {
   $_SESSION['loggedin'] = true;
   $_SESSION['username'] = $user->username;
}
```

1. **Обнаружение уязвимостей:** Найдите и опишите не менее трех уязвимостей или плохих практик в данном коде (например, хранение паролей в открытом виде, слабая проверка).
2. **Исправление кода:** Перепишите код, реализовав безопасное сравнение паролей с использованием функции password_verify() (предположите, что пароль хранится в виде хэша, созданного password_hash()).
3. **Хэширование паролей:** Объясните, почему нельзя использовать для хэширования паролей простые криптографические хэш-функции (например, MD5, SHA-1) без соли.

4. **Многофакторная аутентификация (MFA):** Предложите, как можно интегрировать MFA в данный механизм аутентификации для повышения безопасности.

Критерии оценки: Умение находить множественные уязвимости, корректное исправление кода с использованием безопасных функций, глубина понимания принципов хэширования паролей, практичность предложения по MFA.

Вариант 9

Тема: Анализ зависимостей проекта на наличие уязвимостей (SCA).

Задание:

Вам предоставлен файл зависимостей проекта (например, package.json для Node.js или requirements.txt для Python).

1. **Выбор инструмента:** Выберите и установите инструмент для статического анализа зависимостей (SCA), например, npm audit для Node.js, safety для Python или OWASP Dependency-Check для универсального использования.
2. **Проведение анализа:** Запустите анализ предоставленного файла зависимостей. Составьте таблицу с найденными уязвимостями, указав: название библиотеки, версия, CVE уязвимости, уровень критичности.
3. **План устранения:** Для 3 уязвимостей с высоким уровнем критичности предложите конкретный план действий: можно ли обновиться до безопасной версии, требуется ли изменение кода, есть ли компенсирующие контрмеры?
4. **Интеграция в процесс разработки:** Предложите, как можно интегрировать SCA-проверки в процесс CI/CD (Continuous Integration/Continuous Deployment), чтобы предотвратить попадание уязвимых зависимостей в продакшен.

Критерии оценки: Корректное использование SCA-инструмента, качество составления отчета, практическая ценность плана устранения, реалистичность предложений по интеграции в CI/CD.

Вариант 10

Тема: Комплексный анализ и рефакторинг небезопасного кода.

Задание:

Вам предоставлен небольшой, но функционально завершенный фрагмент кода веб-приложения (например, простой блог с комментариями), содержащий несколько уязвимостей, изученных в данной теме (например, SQLi, XSS, IDOR).

1. **Аудит кода:** Проведите полный аудит кода, составив список всех обнаруженных уязвимостей с указанием их типа и места в коде.
2. **Приоритизация:** Присвойте каждой уязвимости уровень критичности (Высокий, Средний, Низкий) на основе методологии CVSS или качественной оценки.

3. **Рефакторинг:** Проведите рефакторинг кода, устранив все обнаруженные уязвимости. Предоставьте исправленный код.
4. **Меморандум безопасности:** Напишите краткий меморандум (на 1 страницу) для разработчиков вашей команды, в котором сформулируйте 5 ключевых правил безопасного кодирования, которые помогут предотвратить появление подобных уязвимостей в будущем.

Критерии оценки: Полнота и точность аудита, обоснованность приоритизации, качество рефакторинга, ясность, лаконичность и практическая ценность меморандума.

Тема 5. Системы обнаружения и предупреждения компьютерных атак

Общая инструкция для студента:
Вам необходимо выполнить **один** вариант задания. Для работы используйте виртуальные лабораторные стенды, дистрибутивы безопасности (Kali Linux, Security Onion), дампы сетевого трафика и документацию к инструментам. Отчет должен содержать используемые команды, скриншоты, конфигурационные файлы и развернутые выводы.

Вариант 1

Тема: Настройка и использование сетевой системы обнаружения вторжений (NIDS) на примере Suricata.

Задание:

1. **Установка и базовая настройка:** Установите Suricata на виртуальную машину, настроив ее в режиме NIDS для прослушивания заданного сетевого интерфейса.
2. **Написание собственного правила:** Составьте и протестируйте собственное правило для Suricata, которое будет срабатывать при обнаружении в TCP-пакетах строки "malicious_pattern" в любом месте payload и отправлять alert.
3. **Генерация трафика и проверка:** Сгенерируйте тестовый трафик (например, с помощью curl или nc), содержащий и не содержащий данную строку. Убедитесь, что правило срабатывает корректно.
4. **Анализ логов:** Предоставьте скриншоты или выдержки из лог-файлов (eve.json), доказывающие работу правила. Проанализируйте информацию, которую предоставляет Suricata по сработавшему alert.

Критерии оценки: Успешная установка и настройка Suricata, корректный синтаксис собственного правила, умение генерировать тестовый трафик и анализировать логи.

Вариант 2

Тема: Обнаружение сканирования портов и сетевой разведки с помощью анализа сетевого трафика.

Задание:

Вам предоставлен файл дампа сетевого трафика (scan.pcap), содержащий сессию сканирования.

1. **Анализ в Wireshark:** Откройте файл в Wireshark. Используя статистику и фильтры, определите:
    - Целевой IP-адрес и диапазон сканируемых портов.
    - Тип сканирования (например, SYN, Connect, UDP).
    - Количество отправленных пакетов и примерное время проведения сканирования.
2. **Поиск аномалий:** Назовите ключевые признаки в трафике, которые указывают на сканирование, а не на легитимную активность.
3. **Написание правила для Suricata/Snort:** На основе выявленных признаков составьте правило для СОВ, которое бы детектировало подобное сканирование в реальном времени (например, по количеству SYN-пакетов на разные порты за короткий промежуток времени).
4. **Рекомендации:** Какие меры защиты (помимо детектирования) можно принять на сетевом периметре для усложнения задачи разведки для злоумышленника?

Критерии оценки: Глубина анализа трафика в Wireshark, точность определения типа сканирования, корректность составленного правила, практичность рекомендаций по защите.

Вариант 3

Тема: Расследование инцидента на основе логов веб-сервера.

Задание:

Вам предоставлен файл логов веб-сервера Apache/Nginx (access.log), в котором зафиксирована попытка взлома.

1. **Идентификация атаки:** Используя средства командной строки (grep, awk, sort, uniq) или специализированные утилиты (например, goaccess), найдите в логах:
    - IP-адрес атакующего.
    - Методы и URL, на которые направлялись запросы.
    - Признаки атаки (например, подбор параметров для SQLi, XSS, Path Traversal).
2. **Визуализация:** Постройте временную шкалу атаки, отметив ключевые этапы (разведка, попытки эксплуатации уязвимостей).
3. **Определение вектора атаки:** Классифицируйте тип атаки (например, Brute Force, SQL Injection, Local File Inclusion). Был ли злоумышленник успешен в получении доступа? Свой ответ обоснуйте.
4. **Рекомендации по реагированию:** Составьте план первоочередных действий для администратора: что заблокировать, что проверить на предмет компрометации, какие уроки извлечь.

Критерии оценки: Умение работать с логами, точность идентификации атаки и атакующего, качество визуализации, логичность и полнота плана реагирования.

Вариант 4

Тема: Анализ вредоносной активности с помощью SIEM-системы.

Задание:

В развернутой лабораторной среде с ELK Stack (Elasticsearch, Logstash, Kibana) или аналогом проанализируйте инъекцию данных, имитирующую активность ботнета.

1. **Обзор дашборда:** Опишите, какие данные представлены в Kibana (источники логов, типы событий).
2. **Поиск аномалий:** Используя фильтры и запросы KQL (Kibana Query Language), найдите подозрительную активность, например:
    - Исходящие DNS-запросы к необычным доменам.
    - Установление сетевых соединений на нестандартные порты.
    - Неудачные попытки входа в систему с большого количества хостов на один хост.
3. **Корреляция событий:** Свяжите разрозненные события из разных логов (веб-сервер, DNS, фаервол) в единую цепочку компрометации.
4. **Создание правила корреляции:** Предложите правило для SIEM (логику на естественном языке), которое бы автоматически детектировало подобную активность в будущем.

Критерии оценки: Умение ориентироваться в интерфейсе SIEM, навыки поиска и корреляции событий, практическая ценность предложенного правила.

Вариант 5

Тема: Тестирование и настройка системы предотвращения вторжений (IPS) на примере Suricata в режиме Inline.

Задание:

1. **Настройка IPS:** Настройте Suricata в режиме IPS (Network IPS) на шлюзе между двумя сегментами сети, используя nfqueue или af-packet в inline-режиме.
2. **Активация защитных правим:** Активируйте набор правим, которые не только детектируют, но и блокируют атаки (например, правила из набора Emerging Threats, помеченные как drop).
3. **Тестирование блокировки:** Смоделируйте атаку (например, попытку эксплуатации известной уязвимости с помощью metasploit или сканирование nmap с агрессивными опциями) с одной сети на другую. Убедитесь, что трафик блокируется.
4. **Анализ ложных срабатываний:** Проанализируйте логи и определите, не блокирует ли IPS легитимный трафик. Предложите, как можно настроить правила для минимизации ложных срабатываний, не жертвуя безопасностью.

Критерии оценки: Успешная настройка Suricata в режиме IPS, доказательство блокировки атакующего трафика, умение анализировать и предлагать решения для тонкой настройки правил.

Вариант 6

Тема: Расследование инцидента с ransomware по журналам Windows.

Задание:

Вам предоставлены журналы событий Windows (Security.evtx, System.evtx) с компьютера, предположительно инфицированного ransomware.

1. **Поиск IoC (Indicators of Compromise):** Используя утилиты для парсинга логов (например, ползунок времени в Просмотре событий или EvtxECmd), найдите:
   - Подозрительные процессы, которые были запущены.
   - Массовые изменения/удаления файлов в файловой системе.
   - Подозрительные сетевые подключения, если они залогированы.
   - Создание или изменение служб.
2. **Временная шкала:** Восстановите примерную хронологию событий, связанных с инцидентом.
3. **Определение точки входа:** Попробуйте определить, как ransomware попал в систему (фишинг, RDP, уязвимость).
4. **Рекомендации по восстановлению и предотвращению:** Составьте краткий план действий по очистке системы (или рекомендации по восстановлению из бэкапа) и перечень мер, которые предотвратили бы подобный инцидент в будущем.

Критерии оценки: Навыки работы с журналами Windows, умение выявлять IoC, точность восстановления хронологии, практичность рекомендаций.

Вариант 7

Тема: Анализ атаки на веб-приложение с помощью мод-безопасности (ModSecurity) в качестве WAF.

Задание:

1. **Настройка WAF:** Установите и настройте ModSecurity в качестве модуля для веб-сервера (Apache/Nginx) вместе с базовым набором правил OWASP Core Rule Set (CRS).
2. **Моделирование атаки:** Целенаправленно атакуйте собственное веб-приложение (например, DVWA), пытаясь провести SQL-инъекцию или XSS-атаку.
3. **Анализ срабатываний:** Найдите в логах ModSecurity (modsec_audit.log) записи, соответствующие вашим атакам. Определите, какое конкретно правило CRS сработало и почему.
4. **Настройка правил:** Предложите кастомизацию правил CRS (например, создание исключения для легитимного трафика вашего приложения, который ложно определяется как атака), не нарушая его защитных функций.

Критерии оценки: Успешная настройка ModSecurity, умение анализировать логи WAF, понимание логики работы CRS, грамотный подход к тонкой настройке правил.

Вариант 8

Тема: Использование анализатора сетевого трафика Zeek (ранее Bro) для детектирования аномалий.

Задание:

1. **Установка и запуск:** Установите Zeek и запустите его в режиме мониторинга на тестовом интерфейсе.
2. **Генерация трафика:** Сгенерируйте различный сетевой трафик: веб-серфинг, SSH-соединения, DNS-запросы.
3. **Анализ логов:** Изучите логи, которые Zeek создает по умолчанию (conn.log, http.log, dns.log). Опишите, какую информацию извлекает Zeek на каждом из уровней (сеть, приложение).
4. **Написание скрипта-политики:** Напишите простой скрипт на языке Zeek для детектирования подозрительного события (например, попытка подключения по SSH с несуществующего порта-источника) и генерации специального лога или уведомления.

Критерии оценки: Умение работать с Zeek, понимание формата и содержания его логов, корректность написанного скрипта-политики.

Вариант 9

Тема: Создание системы оповещения на основе триггеров СОВ.

Задание:

1. **Выбор сценария:** Выберите сценарий атаки (например, успешный brute-force по SSH, обнаружение известного вредоносного файла по хэшу, массовая рассылка спама с внутреннего хоста).
2. **Настройка детектирования:** Настройте Suricata, OSSEC HIDS или другую систему так, чтобы она детектировала выбранный вами сценарий и генерировала alert.
3. **Настройка оповещения:** Настройте механизм оповещения (например, отправку email через sendmail или скрипт на Python, который отправляет сообщение в Telegram/Slack) при срабатывании этого alert.
4. **Тестирование:** Протестируйте всю цепочку: смоделируйте атаку и убедитесь, что оповещение приходит корректно и содержит всю необходимую для реагирования информацию.

Критерии оценки: Целостность подхода — от выбора сценария до настройки детектирования и оповещения, работоспособность всей цепочки, информативность оповещения.

Вариант 10

Тема: Сравнительный анализ эффективности различных СОВ.

Задание:

1. **Выбор инструментов:** Выберите две системы для сравнения (например, Suricata vs Snort, Wazuh HIDS vs OSSEC, YARA vs Loki).
2. **Создание тестового окружения:** Разверните обе системы в идентичных лабораторных условиях.

3. **Генерация тестового трафика/активности:** Создайте набор тестов, включающий как легитимный трафик/активность, так и известные атаки (например, из датасета DARPA, или смоделированные самостоятельно).
4. **Сравнительный анализ:** Проведите тестирование и заполните сравнительную таблицу по следующим критериям:
    - Процент обнаружения атак (True Positive Rate).
    - Количество ложных срабатываний (False Positives).
    - Простота установки и настройки.
    - Требования к ресурсам (CPU, RAM).
    - Функциональность и гибкость правил.
5. **Вывод:** Сформулируйте вывод, какая система и в каких условиях (малая компания, крупный ЦОД, точка на узле связи) предпочтительнее и почему.

Критерии оценки: Методичность подхода к сравнению, глубина анализа по заданным критериям, обоснованность и практическая ценность итогового вывода.

Критерии оценивания выполнения практического задания:

Оценка «отлично» выставляется, если: работа выполнена в полном объеме с соблюдением необходимой последовательности действий; работа проведена в условиях, обеспечивающих получение правильных результатов и выводов; соблюдены правила техники безопасности; в ответе правильно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления; правильно выполняет анализ ошибок.

Оценка «хорошо» выставляется, если: работа выполнена правильно с учетом 1-2 мелких погрешностей или 2-3 недочетов, исправленных самостоятельно по требованию преподавателя.

Оценка «удовлетворительно» выставляется, если: работа выполнена правильно не менее чем наполовину, допущены 1-2 погрешности или одна грубая ошибка.

Оценка «неудовлетворительно» выставляется, если: допущены две (и более) грубые ошибки в ходе работы, которые обучающийся не может исправить даже по требованию преподавателя или работа не выполнена полностью.

Типовые тесты

Тема 1. Понятие и классификация уязвимостей ПО

П1.1. Какая из перечисленных характеристик относится к понятию "уязвимость" в контексте ИБ?

а) Реализованная возможность нарушения безопасности системы

б) Слабость в системе, которая может быть использована для нарушения ее политики безопасности [✓]

в) Потенциальная причина инцидента, которая может нанести ущерб системе
г) Любая ошибка в программном коде

П1.2. Уязвимость CVE-2021-44228 (Log4Shell) относится primarily к классу:
а) Уязвимости конфигурации
б) Уязвимости архитектуры
в) Уязвимости кода [✓]
г) Организационной уязвимости

П1.3. Какой показатель в CVSS v3.1 характеризует возможность эксплуатации уязвимости без участия пользователя?
а) Attack Vector (AV)
б) User Interaction (UI) [✓]
в) Privileges Required (PR)
г) Scope (S)

П1.4. Уязвимость "нулевого дня" характеризуется тем, что:
а) Имеет нулевой балл по CVSS
б) Не описана в базе CVE
в) Эксплуатируется до момента публикации патча [✓]
г) Обнаружена в первый день месяца

П1.5. Установите соответствие между типом уязвимости и его описанием:

Тип уязвимости	Описание
1. Уязвимость кода	а) Ошибки в настройках ПО и сервисов
2. Уязвимость конфигурации	б) Фундаментальные просчеты в проектировании системы
3. Уязвимость архитектуры	в) Ошибки реализации в исходном коде

Правильный ответ: 1-в, 2-а, 3-б

П1.6. Основное назначение базы данных CVE заключается в:
а) Хранении эксплойтов для уязвимостей
б) Присвоении уникальных идентификаторов уязвимостям [✓]
в) Автоматическом исправлении уязвимостей
г) Оценке стоимости эксплуатации уязвимостей

П1.7. Уязвимость, связанная с неправильной настройкой прав доступа к файлам, классифицируется как:
а) Уязвимость кода
б) Уязвимость конфигурации [✓]
в) Уязвимость архитектуры
г) Многофакторная уязвимость

П1.8. При оценке уязвимости по CVSS v3.1 показатель Attack Complexity (AC) со значением "Low" означает:
а) Для эксплуатации требуются специальные условия
б) Для эксплуатации не требуются специальные условия [✓]

в) Уязвимость невозможно эксплуатировать удаленно
г) Для эксплуатации требуются повышенные привилегии

П1.9. Банк данных угроз безопасности информации ФСТЭК России предназначен для:
а) Хранения исходных кодов уязвимого ПО
б) Учета инцидентов информационной безопасности
в) Сбора и систематизации информации об уязвимостях [✓]
г) Проведения penetration-тестирования

П1.10. Уязвимость, возникающая из-за недостатков организационных процедур и политик безопасности, относится к классу:
а) Уязвимости кода
б) Уязвимости конфигурации
в) Организационной уязвимости [✓]
г) Уязвимости архитектуры

Тема 2. Актуальные уязвимости современного ПО

П2.1. Уязвимости Meltdown и Spectre относятся к классу:
а) Сетевых уязвимостей
б) Уязвимостей операционных систем
в) Аппаратных уязвимостей [✓]
г) Уязвимостей веб-приложений

П2.2. Основной механизм уязвимостей Spectre основан на:
а) Переполнении буфера в микрокоде процессора
б) Спекулятивном выполнении команд процессором [✓]
в) Ошибках кэширования данных в памяти
г) Неправильной обработке прерываний

П2.3. Уязвимости в микропрограммах БИОС/UEFI особенно опасны, потому что:
а) Они позволяют получить доступ к сетевым интерфейсам
б) Они затрагивают уровень загрузки и сохраняются после переустановки ОС [✓]
в) Их невозможно обнаружить сканерами уязвимостей
г) Они влияют только на производительность системы

П2.4. Уязвимости в IoT-устройствах часто связаны с:
а) Использованием устаревших версий ОС
б) Наличием hardcoded учетных данных [✓]
в) Отсутствием графического интерфейса
г) Малым объемом оперативной памяти

П2.5. Установите соответствие между классом уязвимостей и примером:

Класс уязвимостей	Пример
1. Аппаратные уязвимости	а) CVE-2021-44228 (Log4Shell)
2. Уязвимости микропрограмм	б) CVE-2017-5754 (Meltdown)
3. Уязвимости прикладного ПО	в) CVE-2023-1011 (TPM)

Правильный ответ: 1-б, 2-в, 3-а

П2.6. Уязвимости в сетевом оборудовании (маршрутизаторы, коммутаторы) особенно критичны, потому что:
а) Они требуют физического доступа для эксплуатации
б) Они являются точками входа в сетевую инфраструктуру [✓]
в) Они не поддерживают автоматическое обновление
г) Они используют специализированные ОС

П2.7. Уязвимости в средствах защиты информации (антивирусы, DLP) опасны тем, что:
а) Они замедляют работу системы
б) Они могут быть использованы для обхода механизмов защиты [✓]
в) Они требуют лицензирования
г) Они сложны в настройке

П2.8. Уязвимости в общесистемном ПО (ОС) typically имеют высокий балл CVSS из-за:
а) Широкой распространенности и уровня доступа [✓]
б) Сложности обнаружения
в) Невозможности автоматического исправления
г) Отсутствия документации

П2.9. Уязвимости в специальном ПО (SCADA, медицинское оборудование) характеризуются:
а) Высокой частотой обновлений
б) Длительным жизненным циклом и требованиями к доступности [✓]
в) Простотой эксплуатации
г) Отсутствием сетевых интерфейсов

П2.10. Уязвимости в портативных устройствах часто остаются неисправленными из-за:
а) Отсутствия механизмов обновления прошивок [✓]
б) Высокой стоимости обновления
в) Недостатка документации
г) Сложности эксплуатации уязвимостей

Тема 3. Методы и инструменты анализа уязвимостей

П3.1. Какой этап тестирования на проникновение предполагает сбор информации из открытых источников?
а) Сканирование
б) Разведка [✓]
в) Получение доступа
г) Повышение привилегий

П3.2. Какой инструмент используется primarily для сканирования портов?
а) Burp Suite
б) Metasploit
в) Nmap [✓]
г) Wireshark

П3.3. Какой метод разведки позволяет определить структуру сети и маршрутизацию?

а) DNS enumeration

б) Traceroute [✓]

в) Social engineering

г) Port scanning

П3.4. Основное назначение фреймворка Metasploit в тестировании на проникновение:

а) Анализ сетевого трафика

б) Автоматизация эксплуатации уязвимостей [✓]

в) Сканирование веб-приложений

г) Аудит политик безопасности

П3.5. Установите соответствие между инструментом и его назначением:

Инструмент	Назначение
1. OWASP ZAP	а) Сканирование портов и служб
2. Wireshark	б) Тестирование безопасности веб-приложений
3. Aircrack-ng	в) Анализ сетевого трафика
4. John the Ripper	г) Взлом паролей беспроводных сетей
	д) Взлом паролей

Правильный ответ: 1-б, 2-в, 3-г, 4-д

П3.6. Какой тип сканирования уязвимостей предполагает активное взаимодействие с целевой системой?

а) Пассивное сканирование

б) Активное сканирование [✓]

в) Black box тестирование

г) White box тестирование

П3.7. Основная цель этапа "Заметание следов" в тестировании на проникновение:

а) Ускорение процесса тестирования

б) Удаление следов активности тестировщика [✓]

в) Сокрытие уязвимостей от администратора

г) Улучшение производительности системы

П3.8. Какой инструмент используется для анализа беспроводных сетей?

а) Sqlmap

б) Aircrack-ng [✓]

в) Hydra

г) Nikto

П3.9. Основное преимущество автоматизированного сканирования уязвимостей перед ручным тестированием:

а) Полнота покрытия и скорость проверки [✓]

б) Отсутствие ложных срабатываний

в) Возможность обхода систем защиты

г) Низкая стоимость реализации

П3.10. Какой метод тестирования предполагает полное знание архитектуры тестируемой системы?
а) Black box
б) Gray box
в) White box [✓]
г) Red team testing

Тема 4. Предотвращение уязвимостей на этапе реализации

П4.1. Какой метод защиты является наиболее эффективным против SQL-инъекций?
а) Валидация входных данных
б) Экранирование специальных символов
в) Использование подготовленных выражений [✓]
г) Фильтрация HTML-тегов

П4.2. Основной механизм защиты от XSS атак заключается в:
а) Шифровании выходных данных
б) Валидации HTTP-заголовков
в) Экранировании пользовательского ввода [✓]
г) Ограничении длины вводимых данных

П4.3. Уязвимость IDOR (Insecure Direct Object Reference) возникает due to:
а) Недостаточной проверки прав доступа к объектам [✓]
б) Отсутствия валидации входных данных
в) Неправильной настройки сервера БД
г) Использования устаревших библиотек

П4.4. Какой атрибут cookie помогает защититься от CSRF атак?
а) HttpOnly
б) Secure
в) SameSite [✓]
г) Max-Age

П4.5. Установите соответствие между уязвимостью и методом защиты:

Уязвимость	Метод защиты
1. Path Traversal	а) Проверка прав доступа к объектам
2. CSRF	б) Нормализация и валидация путей
3. IDOR	в) Использование CSRF-токенов

Правильный ответ: 1-б, 2-в, 3-а

П4.6. Основная причина уязвимостей типа "Небезопасная десериализация":
а) Отсутствие проверки целостности данных
б) Выполнение произвольного кода при десериализации [✓]
в) Неправильная настройка сервера приложений
г) Использование устаревших форматов данных

П4.7. Какой принцип безопасного кодирования предполагает проверку всех входных данных?

а) Принцип минимальных привилегий
б) Принцип полного противодействия
в) Принцип недоверия к пользовательскому вводу [✓]
г) Принцип глубокой защиты

П4.8. Основное назначение SAST (Static Application Security Testing) инструментов:
а) Тестирование работающего приложения
б) Анализ исходного кода на наличие уязвимостей [✓]
в) Тестирование сетевой безопасности
г) Проверка конфигурации сервера

П4.9. Уязвимости в компонентах сторонних производителей typically устраняются с помощью:
а) Рефакторинга кода
б) Обновления зависимостей [✓]
в) Изменения архитектуры приложения
г) Настройки веб-сервера

П4.10. Какой метод защиты помогает предотвратить утечку информации через ошибки приложения?
а) Настройка корректных HTTP-заголовков
б) Обработка исключений без раскрытия деталей реализации [✓]
в) Использование сложных паролей
г) Шифрование базы данных

Тема 5. Системы обнаружения и предупреждения компьютерных атак

П5.1. Основное отличие IDS от IPS заключается в:
а) Количестве обнаруживаемых атак
б) Возможности блокировать подозрительную активность [✓]
в) Типе анализируемого трафика
г) Используемых методах обнаружения

П5.2. Какой тип системы обнаружения атак анализирует трафик на сетевом уровне?
а) HIDS
б) NIDS [✓]
в) DIDS
г) LIDS

П5.3. Основной принцип работы сигнатурных систем обнаружения:
а) Анализ аномального поведения системы
б) Сравнение с базой известных шаблонов атак [✓]
в) Машинное обучение для выявления угроз
г) Статистический анализ сетевого трафика

П5.4. Какой инструмент является de facto стандартом для анализа сетевого трафика?
а) Tcpdump

б) Wireshark [✓]
в) Netcat
г) Curl

П5.5. Установите соответствие между системой и ее типом:

Система	Тип
1. Snort	а) HIDS
2. OSSEC	б) NIDS
3. Suricata	в) SIEM

Правильный ответ: 1-б, 2-а, 3-б

П5.6. Основное назначение SIEM систем:
а) Блокировка сетевых атак
б) Сбор и корреляция событий безопасности [✓]
в) Сканирование уязвимостей
г) Анализ производительности системы

П5.7. Какой метод обнаружения атак основан на анализе отклонений от нормального поведения?
а) Сигнатурный анализ
б) Анализ аномалий [✓]
в) Эвристический анализ
г) Статический анализ

П5.8. Основная проблема ложных срабатываний (false positives) в системах обнаружения атак:
а) Снижение производительности системы
б) Увеличение объема логов
в) Снижение доверия к системе и пропуск реальных угроз [✓]
г) Увеличение времени реакции на инциденты

П5.9. Какой компонент современных систем обнаружения атак использует машинное обучение?
а) База сигнатур
б) Модуль корреляции событий
в) Система поведенческого анализа [✓]
г) Механизм блокировки трафика

П5.10. Основное преимущество host-based систем обнаружения атак перед network-based:
а) Возможность анализа зашифрованного трафика [✓]
б) Более высокая производительность
в) Простота развертывания
г) Низкая стоимость реализации

Критерии оценивания тестовых заданий для текущего контроля

Максимальное количество баллов за решение одного теста составляет 5 баллов, которые переводятся в академические оценки следующим образом:

- оценка «отлично» (5 баллов) выставляется при условии правильного ответа студента не менее чем на 85% контрольных заданий

- оценка «хорошо» (4 балла) выставляется при условии правильного ответа студента не менее чем на 70% контрольных заданий

- оценка «удовлетворительно» (3 балла) выставляется при условии правильного ответа студента не менее чем на 51% контрольных заданий

- оценка «неудовлетворительно» (1-2 балла) выставляется при условии правильного ответа студента менее чем на 50% контрольных заданий

2.2 Оценочные материалы для проведения промежуточной аттестации

ПК-3.4.1 Анализирует компьютерную систему с целью определения уровня защищённости и доверия

П1. Какой показатель в CVSS v3.1 является определяющим для оценки уровня защищенности системы от удаленных атак?

а) User Interaction (UI);

б) Attack Vector (AV) с значением Network; [✓]

в) Privileges Required (PR);

г) Scope (S).

Ответ: б) Attack Vector (AV) с значением Network

П2. При анализе уровня доверия к операционной системе ключевым фактором является:

а) Количество установленных приложений;

б) Наличие и эффективность встроенной подсистемы безопасности; [✓]

в) Популярность ОС среди пользователей;

г) Стоимость лицензии.

Ответ: б) Наличие и эффективность встроенной подсистемы безопасности

П3. Уязвимости "нулевого дня" существенно снижают уровень доверия к системе, потому что:

а) Они имеют нулевой балл по CVSS;

б) Они не описаны в базе CVE;

в) Эксплуатируются до момента публикации патча и отсутствуют сигнатуры для обнаружения; [✓]

г) Обнаруживаются только в первый день месяца.

Ответ: в) Эксплуатируются до момента публикации патча и отсутствуют сигнатуры для обнаружения

П4. Установите соответствие между элементом анализа и оцениваемой характеристикой защищенности системы:

Элемент анализа	Характеристика защищенности
1. Аудит политик парольной сложности	а) Устойчивость к атакам на доступ
2. Сканирование открытых портов	б) Защищенность сетевого периметра
3. Анализ журналов аудита	в) Обеспечение подотчетности

Правильный ответ:

1 – а, 2 – б, 3 – в

П5. Установите правильную последовательность анализа уровня защищенности компьютерной системы:
а) Идентификация критических активов и данных.
б) Анализ архитектуры системы и точек входа.
в) Оценка механизмов контроля доступа.
г) Тестирование на уязвимости.
д) Формирование вывода об уровне доверия к системе.
Ответ: АБВГД

П6. Установите правильную последовательность оценки защищенности распределенной системы:
а) Анализ сетевых взаимодействий и протоколов.
б) Проверка аутентификации и авторизации.
в) Оценка шифрования передаваемых данных.
г) Тестирование устойчивости к атакам.
д) Определение общего уровня доверия к системе.
Ответ: АБВГД

П7. Как называется метод анализа безопасности, при котором система рассматривается как набор взаимодействующих компонентов?
Правильный ответ: Компонентный анализ безопасности (Component Security Analysis)

П8. Как называется характеристика системы, определяющая ее способность противостоять атакам?
Правильный ответ: Уровень защищенности (Security Level) или Устойчивость к атакам (Attack Resilience)

П9. Анализ уязвимости CVE-2021-44228 (Log4Shell) показал значение CVSS 10.0. О чем это свидетельствует?
а) Уязвимость невозможно эксплуатировать;
б) Уязвимость имеет низкий уровень опасности;
в) Система имеет критический уровень незащищенности от данной угрозы; [✓]
г) Система полностью защищена от данной угрозы.
Ответ: в) Система имеет критический уровень незащищенности от данной угрозы

П10. При анализе уровня доверия к микропрограмме БИОС/UEFI ключевым риском является:
а) Низкая скорость загрузки системы;
б) Отсутствие графического интерфейса;
в) Возможность сохранения вредоносного кода после переустановки ОС; [✓]
г) Сложность обновления прошивки.
Ответ: в) Возможность сохранения вредоносного кода после переустановки ОС

ПК-3.4.2 Демонстрирует умение проводить анализ безопасности

компонентов программных и программно-аппаратных средств защиты информации в компьютерных системах

П1. Какой инструмент является наиболее эффективным для анализа безопасности компонента веб-приложения на наличие XSS?

а) Nmap;

б) OWASP ZAP; [✓]

в) Aircrack-ng;

г) Wireshark.

Ответ: б) OWASP ZAP

П2. При анализе безопасности компонента системы, реализующего аутентификацию, необходимо проверить:

а) Скорость обработки запросов;

б) Устойчивость к перебору паролей и наличие механизмов блокировки; [✓]

в) Размер исполняемого файла;

г) Количество строк кода.

Ответ: б) Устойчивость к перебору паролей и наличие механизмов блокировки

П3. Анализ компонента антивирусного ПО выявил уязвимость CVE-2019-1367. Это означает, что:

а) Антивирусное ПО не может обнаруживать вирусы;

б) Компонент средства защиты информации сам содержит уязвимость; [✓]

в) Антивирусное ПО требует перелицензирования;

г) Антивирусное ПО замедляет работу системы.

Ответ: б) Компонент средства защиты информации сам содержит уязвимость

П4. Установите соответствие между компонентом системы и методом его анализа на безопасность:

Компонент системы	Метод анализа безопасности
1. Сетевой фаервол	а) Проверка правил фильтрации и тестирование на обход
2. Модуль шифрования	б) Анализ реализации криптоалгоритмов и стойкости ключей
3. Подсистема ввода-вывода	в) Проверка обработки исключений и целостности данных

Правильный ответ:

1 – а, 2 – б, 3 – в

П5. Установите правильную последовательность анализа компонента ПО на уязвимости:

а) Статический анализ исходного кода (SAST).

б) Анализ зависимостей (SCA) на наличие известных уязвимостей.

в) Динамический анализ (DAST) работающего компонента.

г) Анализ конфигурации развертывания.

д) Составление отчета о выявленных уязвимостях компонента.

Ответ: АБВГД

П6. Установите правильную последовательность тестирования

компонента межсетевого экрана:
а) Проверка корректности базовых правил фильтрации.
б) Тестирование на возможность обхода фильтрации.
в) Проверка устойчивости к DoS-атакам.
г) Анализ журналов работы компонента.
д) Оценка производительности под нагрузкой.
Ответ: АБВГД

П7. Как называется метод анализа, при котором проверяется поведение компонента при обработке специально сформированных некорректных входных данных?
Правильный ответ: Фаззинг-тестирование (Fuzzing Testing)

П8. Как называется анализ компонентов системы на соответствие требованиям стандартов безопасности?
Правильный ответ: Верификация безопасности (Security Verification) или Аудит безопасности (Security Audit)

П9. При анализе компонента, отвечающего за обработку пользовательских сессий, необходимо уделить особое внимание:
а) Размеру файлов cookie;
б) Скорости создания сессии;
в) Защите идентификаторов сессий от перехвата и подбора; [✓]
г) Цвету интерфейса пользователя.
Ответ: в) Защите идентификаторов сессий от перехвата и подбора

П10. Анализ компонента системы шифрования должен включать проверку:
а) Использования стандартных и проверенных криптографических библиотек; [✓]
б) Скорости работы алгоритма на устаревшем оборудовании;
в) Размера ключевых файлов на диске;
г) Возможности изменения алгоритма шифрования пользователем.
Ответ: а) Использования стандартных и проверенных криптографических библиотек

ПК-3.4.3 Формирует предложения по устранению выявленных уязвимостей

П1. Какое предложение по устранению является наиболее эффективным для уязвимости SQL-инъекции в компоненте веб-приложения?
а) Увеличить время ожидания ответа от БД;
б) Внедрить использование подготовленных выражений (Prepared Statements); [✓]
в) Увеличить длину входных параметров;
г) Добавить проверку на стороне клиента.
Ответ: б) Внедрить использование подготовленных выражений (Prepared Statements)

П2. При выявлении уязвимости XSS в компоненте отзывов сайта

необходимо предложить:
а) Отключить функционал отзывов;
б) Внедрить экранирование пользовательского ввода перед выводом в HTML; [✓]
в) Ограничить количество символов в отзыве;
г) Удалить все существующие отзывы.
Ответ: б) Внедрить экранирование пользовательского ввода перед выводом в HTML

П3. Для устранения уязвимости, связанной с хранением паролей в открытом виде, необходимо предложить:
а) Регулярно менять пароли;
б) Внедрить хэширование паролей с использованием адаптивных алгоритмов (bcrypt, Argon2); [✓]
в) Зашифровать файл с паролями;
г) Запретить пользователям изменять пароли.
Ответ: б) Внедрить хэширование паролей с использованием адаптивных алгоритмов (bcrypt, Argon2)

П4. Установите соответствие между выявленной уязвимостью и предлагаемой мерой по ее устранению:

Выявленная уязвимость	Предлагаемая мера устранения
1. Слабые пароли пользователей	а) Внедрение многофакторной аутентификации
2. Отсутствие защиты от CSRF	б) Внедрение политики сложных паролей и проверки утечек
3. Необновленное ПО с известными уязвимостями	в) Внедрение CSRF-токенов в формы
	г) Внедрение процесса регулярного обновления ПО и управления исправлениями

Правильный ответ:
1 – б, 2 – в, 3 – г

П5. Установите правильную последовательность действий по устранению критической уязвимости на проданной системе:
а) Немедленное применение временных компенсирующих контрмер (например, блокировка на уровне сети).
б) Тестирование патча в тестовой среде.
в) Установка официального патча от производителя.
г) Проведение повторного сканирования для подтверждения устранения уязвимости.
д) Внесение изменений в процессы для предотвращения повторного возникновения.
Ответ: АБВГД

П6. Установите правильную последовательность формирования предложений по устранению уязвимостей в отчете:

а) Приоритизация уязвимостей по уровню риска (CRITICAL, HIGH, MEDIUM, LOW).
б) Детальное описание каждой уязвимости и связанного с ней риска.
в) Формулировка конкретных технических рекомендаций по устранению.
г) Предложение компенсирующих контрмер, если немедленное устранение невозможно.
д) Оценка трудозатрат и рекомендации по срокам исправления.
Ответ: АБВГД

П7. Как называется временная мера защиты, применяемая до момента полного устранения уязвимости?
Правильный ответ: Компенсирующая контрмера (Compensating Control) или Временное решение (Workaround)

П8. Как называется процесс управления выявленными уязвимостями, включающий их отслеживание и контроль устранения?
Правильный ответ: Управление уязвимостями (Vulnerability Management)

П9. После выявления уязвимой версии библиотеки в компоненте ПО необходимо предложить:
а) Игнорировать риск, так как это сторонняя библиотека;
б) Обновить библиотеку до безопасной версии; [✓]
в) Удалить функционал, использующий эту библиотеку;
г) Увеличить логирование работы библиотеки.
Ответ: б) Обновить библиотеку до безопасной версии

П10. При формировании предложения по устранению уязвимости в компоненте сетевого оборудования необходимо:
а) Рекомендовать отключение оборудования;
б) Предложить конкретную команду или настройку для устранения уязвимости, ссылаясь на официальный источник; [✓]
в) Посоветовать заменить оборудование;
г) Предложить уменьшить нагрузку на оборудование.
Ответ: б) Предложить конкретную команду или настройку для устранения уязвимости, ссылаясь на официальный источник

Критерии оценки для проведения зачета по дисциплине (тестирование)

Шкала оценивания результатов промежуточной аттестации и критерии выставления оценок.

Оценка	Уровень сформированности компетенции	Критерии
«Отлично» («зачтено»)	Высокий	Выполнено более 80% заданий предложенного теста

«Хорошо» («зачтено»)	Хороший	Выполнено 60-80% заданий предложенного теста
«Удовлетворительно» («зачтено»)	Достаточный	Выполнено 35-60% заданий предложенного теста
«Неудовлетворительно» («не зачтено»)	Недостаточный	Выполнено менее 35% заданий предложенного теста

Обновление оценочных материалов дисциплины

обсуждено и рекомендовано к утверждению решением кафедры наименование кафедры
от ___ _____ 20_ г., протокол № ___

одобрено Научно-методическим советом _____ от ___ _____ 20__ г., протокол №____