Министерство науки и высшего образования РФ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Национальный исследовательский университет «МЭИ»

Направление подготовки/специальность: 10.03.01 Информационная безопасность

Наименование образовательной программы: Безопасность компьютерных систем (продвинутый уровень)

Уровень образования: высшее образование - бакалавриат

Форма обучения: Очная

Оценочные материалы
по дисциплине
Безопасность Web-приложений

Москва
2022

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ РАЗРАБОТАЛ:

Преподаватель	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ» Сведения о владельце ЦЭП МЭИ Владелец: Баронов О.Р. Идентификатор: R90d76356-BaronovOR-7bf8fd7e	О.Р. Баронов
(должность)	(подпись)	(расшифровка подписи)

СОГЛАСОВАНО:

Руководитель образовательной программы	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ» Сведения о владельце ЦЭП МЭИ Владелец: Баронов О.Р. Идентификатор: R90d76356-BaronovOR-7bf8fd7e	О.Р. Баронов
(должность, ученая степень, ученое звание)	(подпись)	(расшифровка подписи)
Заведующий выпускающей кафедры	Подписано электронной подписью ФГБОУ ВО «НИУ «МЭИ» Сведения о владельце ЦЭП МЭИ Владелец: Невский А.Ю. Идентификатор: R4bc65573-NevskyAY-0b6e493d	А.Ю. Невский
(должность, ученая степень, ученое звание)	(подпись)	(расшифровка подписи)

ОБЩАЯ ЧАСТЬ

Оценочные материалы по дисциплине предназначены для оценки: достижения обучающимися запланированных результатов обучения по дисциплине, этапа формирования запланированных компетенций и уровня освоения дисциплины.

Оценочные материалы по дисциплине включают оценочные средства для проведения мероприятий текущего контроля успеваемости и промежуточной аттестации.

Формируемые у обучающегося компетенции:

1. ПК-3 Способен администрировать средства защиты информации в компьютерных системах и сетях

ПК-3.3 Администрирует средства защиты информации прикладного и системного программного обеспечения

и включает:

для текущего контроля успеваемости:

Форма реализации: Выполнение задания

1. Администрирование механизмов защиты от киберугроз в системных и прикладных программных продуктах, а также web-приложениях (Домашнее задание)
2. Анализ типовых механизмов защиты от кибератак на web-приложения (Отчет)
3. Анализ требований нормативных документов регуляторов по обеспечению защиты web-приложений (Отчет)
4. Формирование рекомендаций по разработке программы безопасности web-приложения кампании (Домашнее задание)

БРС дисциплины

7 семестр

Раздел дисциплины	Веса контрольных мероприятий, %				
	Индекс КМ:	КМ-1	КМ-2	КМ-3	КМ-4
	Срок КМ:	4	8	12	15
Требования к безопасности web-приложений					
Тема 1. Введение в безопасность приложений		+			
Тема 2. Жизненный цикл защиты web-приложения		+			
Тема 3. Построение программы безопасности web-приложения			+		
Тема 4. Сфера действия безопасности приложений				+	
Тема 5. Требования ГОСТ Р 56939-2016				+	
Защита Web-париложений					
Тема 6.. Выявление и эксплуатация SQL-инъекций в приложениях					+

Тема 7. Защита веб-приложений от атак типа XSS				+
Тема 8. Применение подхода DevSecOps в современных системах разработки программного обеспечения				+
Вес КМ:	15	35	15	35

$Общая часть/Для промежуточной аттестации$

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ТЕКУЩЕГО КОНТРОЛЯ

I. Оценочные средства для оценки запланированных результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Индекс компетенции	Индикатор	Запланированные результаты обучения по дисциплине	Контрольная точка
ПК-3	ПК-3.3$_{ПК-3}$ Администрирует средства защиты информации прикладного и системного программного обеспечения	Знать: требования нормативных документов регуляторов по обеспечению защиты web-приложений типовые механизмы защиты от кибератак на web-приложения Уметь: разрабатывать рекомендации по применению мер защиты web-приложений при их разработке, развертыванию и использованию администрировать механизмы защиты от киберугроз в системных и прикладных программных продуктах, а также web-приложениях	Анализ типовых механизмов защиты от кибератак на web-приложения (Отчет) Формирование рекомендаций по разработке программы безопасности web-приложения кампании (Домашнее задание) Анализ требований нормативных документов регуляторов по обеспечению защиты web-приложений (Отчет) Администрирование механизмов защиты от киберугроз в системных и прикладных программных продуктах, а также web-приложениях (Домашнее задание)

II. Содержание оценочных средств. Шкала и критерии оценивания

КМ-1. Анализ типовых механизмов защиты от кибератак на web-приложения

Формы реализации: Выполнение задания
Тип контрольного мероприятия: Отчет
Вес контрольного мероприятия в БРС: 15
Процедура проведения контрольного мероприятия: Самостоятельное выполнение практического занятия

Краткое содержание задания:
Используя информацию тем 1 и 2, а также общедоступных интернет - ресурсов провести анализ типовых механизмов защиты от кибератак на web-приложения

Контрольные вопросы/задания:

Знать: типовые механизмы защиты от кибератак на web-приложения	1.Проблема безопасности web-приложений 2.Отличия защиты web-приложений от защиты сетей и хостов 3.Мероприятия безопасной разработки web-приложений 4.Мероприятия безопасного развертывания web-приложений 5.Мероприятия безопасного использования web-приложений

Описание шкалы оценивания:

Оценка: зачтено
Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено
Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-2. Формирование рекомендаций по разработке программы безопасности web-приложения кампании

Формы реализации: Выполнение задания
Тип контрольного мероприятия: Домашнее задание
Вес контрольного мероприятия в БРС: 35
Процедура проведения контрольного мероприятия: Самостоятельное выполнение практического занятия

Краткое содержание задания:
Используя материалы лекции по теме 3 и интернет - сформировать рекомендаций по разработке программы безопасности web-приложения в интересах типовой организации

Контрольные вопросы/задания:

Уметь: разрабатывать рекомендации по применению	1.Порядок использования статических и динамических инструментов для первоначальной

мер защиты web-приложений при их разработке, развертыванию и использованию	проверки кода 2.Основы сканирования web-приложений для оценки уровня защищенности, проверки установки обновлений и недостатков в конфигурации

Описание шкалы оценивания:

Оценка: зачтено
Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено
Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-3. Анализ требований нормативных документов регуляторов по обеспечению защиты web-приложений

Формы реализации: Выполнение задания
Тип контрольного мероприятия: Отчет
Вес контрольного мероприятия в БРС: 15
Процедура проведения контрольного мероприятия: Самостоятельное выполнение практического занятия

Краткое содержание задания:
Используя информацию тем 4 и 5, а также общедоступных интернет - ресурсов провести анализ требований нормативных документов регуляторов по обеспечению защиты web-приложений

Контрольные вопросы/задания:

Знать: требования нормативных документов регуляторов по обеспечению защиты web-приложений	1.Перечислите сферы действия безопасности приложений 2.Что является источником требований безопасности приложений

Описание шкалы оценивания:

Оценка: зачтено
Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено
Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

КМ-4. Администрирование механизмов защиты от киберугроз в системных и прикладных программных продуктах, а также web-приложениях

Формы реализации: Выполнение задания
Тип контрольного мероприятия: Домашнее задание
Вес контрольного мероприятия в БРС: 35
Процедура проведения контрольного мероприятия: Самостоятельное выполнение практического занятия

Краткое содержание задания:

Используя материалы лекции по темам 6, 7, 8 и интернет - решить задание по предотвращению атак на web-приложение, связанных с инъекциями команд, с XSS с CSRF, Path/Directory Traversal и Open Redirect

Контрольные вопросы/задания:

Уметь: администрировать механизмы защиты от киберугроз в системных и прикладных программных продуктах, а также web-приложениях	1.Какие бывают техники атак при SQL-инъекциях 2.Понятие DevSecOps

Описание шкалы оценивания:

Оценка: зачтено
Описание характеристики выполнения знания: Оценка "зачтено" выставляется если задание выполнено правильно или с незначительными недочетами

Оценка: не зачтено
Описание характеристики выполнения знания: Оценка "не зачтено" выставляется если задание не выполнено в отведенный срок или результат не соответствует заданию

СОДЕРЖАНИЕ ОЦЕНОЧНЫХ СРЕДСТВ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

7 семестр

Форма промежуточной аттестации: Зачет

Процедура проведения

Устный опрос

I. Перечень компетенций/индикаторов и контрольных вопросов проверки результатов освоения дисциплины

1. Компетенция/Индикатор: ПК-3.З$_{ПК-3}$ Администрирует средства защиты информации прикладного и системного программного обеспечения

Вопросы, задания

1.Проблемы безопасности приложений
2.Чем защита web-приложений отличается от защиты сетей и хостов
3.Цикла разработки и использования web-приложения
4.Основы безопасной разработки web-приложений
5.Инструменты статического анализ исходного кода web-приложения
6.Инструменты динамического анализ web-приложения
7.Основы безопасного развертывания web-приложений
8.Инструменты основанные на открытом коде для оценки уязвимости web-приложений с широким функционалом
9.Оценка уязвимостей на основе тестирования на проникновение
10.Интеграция оценки уязвимости и тестирования на проникновение в безопасное развертывание
11.Основы безопасного использования web-приложений
12.Задачи Web Application Firewall как средства мониторинга безопасности web-приложения
13.Рекомендации по разработке программы безопасности web-приложения крупной кампании
14.Рекомендации по разработке программы безопасности web-приложения кампании среднего размера в соответствие требованиям PCI
15.Рекомендации по разработке внутреннего web-приложения
16.Сфера действия безопасности приложений
17.Требования безопасности приложений согласно ГОСТ Р ИСО/МЭК 27034-1 – 2014. Менеджмент безопасности приложений
18.Процесс менеджмента информационной безопасности приложений
19.Нормативная структура организации (ONF)
20.ONF основа безопасности организации. Общая информация. Компоненты. Библиотека мер и средств контроля и управления безопасностью приложения (ASC) организации.
21.Процесс менеджмента нормативной структуры организации
22.Требования ГОСТ Р 56939-2016
23.Разработка безопасного программного обеспечения. Общие требования
24.Меры по разработке безопасного программного обеспечения
25.Выявление и эксплуатация SQL-инъекций в приложениях.
26.Причины возникновения SQL-инъекций
27.Техники, применяемые при эксплуатации SQLинъекций
28.Процесс обнаружения и эксплуатации SQL-инъекций

29.Защита веб-приложений от инъекций команд
30.Характеристика основ внедрения опасных команд
31.Методы обнаружения внедрения опасных команд
32.OWASP CheatSheet
33.Защита веб-приложений от атак типа XSS
34.Общее понятие XSS. Виды XSS. Контексты выполнения
35.Меры предотвращения stored и reflected XSS
36.Меры предотвращения stored и reflected XSS
37.Меры предотвращения DOM-based XSS.
38. Применение подхода DevSecOps в современных системах разработки программного обеспечения
39.Понятие DevSecOps. Организация фаззинга исходного кода

Материалы для проверки остаточных знаний

1.Проблема безопасности web-приложений

Ответы:

-

Верный ответ: Лишь некоторые приложения разработаны с учетом требований безопасности. • Приложение зависит от используемого web-браузера, который не является ни безопасной, ни доверенной средой. • Web-приложения развивались с целью обеспечения доступа к самым чувствительным системам бэк-енда из публичной сети без должного уровня безопасности. • Web не был разработан как безопасная платформа и, соответственно не соответствует тривиальным правилам безопасности – конфиденциальность, целостность, доступность. • Лишь небольшое количество нарушений безопасности обнаруживаются и позволяют проанализировать потери компании и, соответственно, привести к усилению внимания к проблеме со стороны руководства. • У нас есть огромный объем старого кода, который постоянно исправляется, но, в тоже время, постоянно дописывается все новый и новый код. • Web-приложение – это комплекс платформ, инструментов и сервисов от множества провайдеров, каждый со своими вызовами безопасности. • Многие web-приложения имеют критическое значение, но разработаны без учета этого факта. • Если Web-приложение разработано внутри компании, то именно внутри компании должны отслеживаться уязвимости и выпускаться исправления – никто другой эту работу не сделает. • Требования безопасности зачастую вступают в конфликт с требованиями, выдвигаемыми бизнесом, в частности, если это касается простоты использования и скорости работы. • Ни один инструмент безопасности web-приложений не обеспечит эффективную защиту в одиночку

2.Отличия защиты web-приложений от защиты сетей и хостов

Ответы:

-

Верный ответ: существенные отличия: 1. UI браузера: В большинстве приложений мы создаем пользовательский интерфейс. Но в случае с web-приложениями, мы полагаемся на сторонний просмотрщик (браузер), который не можем полностью контролировать и который может контактировать и с другими приложениями в один момент времени. 2. Собственный код порождает собственные уязвимости: В случае в web-приложениями вы обычно самостоятельно пишите код (используя при этом плагины и фреймворки). Это значит, что большая часть уязвимостей в вашем приложении будет уникальная. Если вы не будете контролировать свое приложение, то никто не сообщит вам о каких-либо имеющихся уязвимостях и никто не предложит патч, позволяющий их закрыть. 3. Вы разработчик: Когда появится уязвимость, у вас не будет разработчика, который подготовит патч (вы должны, конечно, установить патчи для всех инфраструктурных компонентов, фреймворков и

скриптов, которые вы используете). Если вы обслуживаете клиентов, то вам необходимо придерживаться пункта договора об уровне сервиса и обеспечивать необходимую доступность сервиса. 4. Межсетевые экраны в одиночку не могут защитить web-приложение: Когда мы обнаруживаем уязвимости в нашем корпоративном ПО, операционных системах, приложениях, базах данных и т.д., мы используем инструменты вроде межсетевых экранов или IPS для блокирования возможностей для потенциальной атаки на время ожидания выпуска патча для уязвимого программного обеспечения. Такая “блокировка до патча” имеет лишь ограниченную эффективность. А Web Application Firewall (WAF) не сможет защитить вас от логических ошибок. Хотя WAF может помочь с некоторыми классами атак, но “из коробки” они не знают или не понимают выше приложение, а потому не сможет “прикрыть” пользовательские уязвимости. WAF безусловно является важной частью защиты web-приложений, но лишь в том случае, если он является частью комплексной программы, о которой мы поговорим позже. 5. Вечная бета-версия: При разработке традиционного приложения проводится полный цикл проверки: контроль на стадии разработки, внутреннее тестирование, а также, зачастую, дополнительное тестирование с привлечением сторонних пользователей – бета-тестеров. И все это до того, как приложение перейдет в эксплуатацию. Было бы здорово, если бы и web-приложения проходили этот последовательный цикл, но, как уже было сказано, так бывает крайне редко. Большая часть web-приложений, даже рассматриваемых разработчиками как неокончательная, в действительности уже запущена в эксплуатацию и даже стали ключевыми решениями критически важными для бизнеса. Другие приложения и вовсе находятся в режиме постоянного изменения и их разработчики даже не придерживаются формального цикла разработки. Постоянно меняющиеся приложения – это непростая задача для существующих средств контроля безопасности, таких как WAF. 6. Опора на фреймворки/платформы: Web-приложения редко строятся с нуля, радуя “вылизанным” кодом на C. Чаще всего это смесь разнообразных фреймворков и средств разработки и платформ, которые не всегда разработаны для совместной работы. Задача разработчиков обеспечить взаимодействие всех этих частей, а также собственного кода. Такой подход, зачастую, создает серьезные проблемы безопасности из-за неправильного использования компонентов, их взаимодействия между собой, а также с собственным кодом. 7. Унаследованный код: Даже если новый код пишется с чистого листа и действительно безопасен, то не стоит забывать, что в большинстве случаев есть еще огромный массив старого кода, который также имеет огромное число уязвимостей, а потому подлежит анализу и исправлению. Если старый код находится в активном использовании, то он должен быть столь же безопасен, как и новый. Зачастую, именно устаревший код становится виновником атак на web-приложение. 8. Динамический контент: Большая часть web-приложений имеет чрезвычайно динамичный характер, создавая большую часть контента “на лету”, нередко с использованием данных (в том числе кода), предоставленных пользователем. А браузер пытается все это обработать, создавая, тем самым, новые классы проблем безопасности. 9. Новые классы уязвимостей: Как и в случае с классическими приложениями, исследователи и злоумышленники постоянно открывают новые классы уязвимости web-приложений. Таких примеров множество. А потому нужно помнить, что даже идеальный с точки зрения сегодняшнего дня код вовсе не гарантирует то, что он всегда будет безопасен.

3.Мероприятия безопасной разработки web-приложений

Ответы:

-

Верный ответ: • Безопасная разработка: Внедрение практик безопасной разработки в процессе создание web-приложения. • Статический анализ: Инструмент для

сканирования исходного кода приложения на ошибки безопасности. Также эти инструменты называют “white box”. • Динамический анализ: Инструменты исследующие не исходный код, а само запущенное приложение при попытках атаки. Эти инструменты часто называют “black box”.

4.Мероприятия безопасного развертывания web-приложений

Ответы:

-

Верный ответ: • Оценка уязвимости: удаленное сканирование web-приложения, как с учетной записью, так и без. Оценка уязвимости web-приложения концентрируется на самом приложении, в то время как стандартная оценка уязвимости сосредотачивается на исследовании хост-платформы. • Тестирование на проникновение: Тестирование на проникновение является фактически попыткой взлома, которая позволяет выявить уязвимости в системе безопасности и те риски, которые они несут. Тестирование на проникновение позволяет оценить недостатки приложения, классифицировать их и правильно расставить приоритеты.

5.Мероприятия безопасного использования web-приложений

Ответы:

-

Верный ответ: • Мониторинг активности приложения и базы данных: Инструмент, который контролирует активность приложения и базы данных (с помощью различных методов) для аудита и генерации предупреждений безопасности, основанных на нарушении заданных правил.

6.Основное содержание рекомендаций по разработке программы безопасности web-приложения крупной кампании

Ответы:

-

Верный ответ: Обучение и отладка процессов. Область, которая даст максимальный эффект в виде повышения уровня безопасности – это улучшение знаний и навыков команды разработчиков. Безопасный цикл разработки приложений. Должен стать одним из приоритетных требований при разработке приложения, включая в себя определенные требования к коду, проверке и тестированию на разных этапах разработки web-приложения. В противном случае добиться должного уровня будет невозможно, так как все силы будут брошены на доработку возможностей и функционала. Безопасность должны быть частью спецификации продукта и требований к нему, а каждая фаза разработки должны завершаться проверкой на соблюдение этих требований и соответствие спецификации. Наследуемый код приложения. Есть решение проблемы наследуемого кода. Одной из серьезных проблем – это наследуемый код, который, вполне вероятно, имеет проблемы с безопасностью. Есть несколько вариантов решения этой проблемы, но основными шагами в любом случае будут: 1) Выявление недостатков и проблем в коде (сканирование кода, оценка уязвимости и тестирование на проникновение); 2) Расстановка приоритетов по исправлению выявленных недостатков; 3) Планирование по устранению каждой найденной уязвимости. Общие методы исправления уязвимостей включают: 1) Переписывание сегментов кода; 2) методы инкапсуляции (например, интерфейс); 3) Дополнение существующего кода путем создания процедур проверки путей/методов во время исполнения; 4) временное защита при помощи настройки политик WAF; 5) Перемещение SQL-процессов и проверок в базу данных; 6) прекращение использования небезопасных функций. Периодические внешние проверки – оценка уязвимости, тестирование на проникновение и проверка исходного кода –настоятельно рекомендуются. Опытные непредвзятые специалисты, имеющие опыт в анализе угроз, могут выявить те

недостатки, которые были пропущены во время внутренних сканирований, а также могут помочь в обучении разработчиков иным векторам атак

7.Основное содержание рекомендаций по разработке программы безопасности web-приложения средней кампании

Ответы:

-

Верный ответ: Тренинги, образование и улучшение процессов. Опять же, максимальный упор мы делаем на образование и тренинги для персонала, в том числе для менеджмента проекта. Несмотря на то, что это требует достаточно солидных временных затрат, он позволяет ощутимо повысить уровень безопасности и является эффективным с экономической точки зрения (за счет улучшения качества кода, что снижает затраты на доработку в перспективе). Внешняя помощь. Подружитесь с аудитором или наймите его в качестве консультанта, который поможет подготовить и провести проверку на соответствие PCI-DSS. Аудитор не просто даст специфические рекомендации по отдельным требованиям PCI, он предоставит экспертную оценку, поможет в толковании некоторых неоднозначных моментов, окажет содействие в разработке стратегии и убережет от необдуманных и неэффективных трат. Раздел 11.3.2. 11.3 Тестирование на проникновение сети и web-приложения. В данном сценарии мы рекомендуем внешнее тестирование на проникновение – не только потому, что это требование стандарта DSS, но также потому, что независимый эксперт изучит ваше приложение позиции максимально близкой к хакерской.

8.Порядок использования статических и динамических инструментов для первоначальной проверки кода

Ответы:

-

Верный ответ: Средства статистического анализа в основном используются для сканирования необработанных условий ошибок, наличия объекта и/или определения размера, а также потенциально возможного переполнения буфера. Эти инструменты используются на стадии разработки для того чтобы выявить недостатки до перехода к более формализованным процедурам тестирования. Ведь чем раньше становится известно о проблеме, тем проще ее исправить. Статический анализ кода выполняется самими разработчиками, что значительно ускоряет поиск ошибок и делает это заметно дешевле, чем если бы анализом занимались отдельные люди. Эти инструменты могут быть интегрированы с управлением исходным кодом для автоматизированного выполнения анализа, опять же. Для того чтобы выявить недостатки в коде как можно раньше. Динамический анализ используется для выявления проблем и уязвимостей, которые не могут быть выявлены при анализе исходного кода или которые намного проще заметить в тот момент, когда приложение запущено. Один из типов анализа называется “fuzzers” которые направляет приложению заведомо вредные или фиктивные материалы для приложения и отслеживает результаты этих действий, а также сбои в работе приложения.

9.Основы сканирования web-приложений для оценки уровня защищенности, проверки установки обновлений и недостатков в конфигурации

Ответы:

-

Верный ответ: Имеется целый ряд коммерческих, бесплатных и основанных на открытом коде инструментов и решений для оценки уязвимости web-приложений с широким функционалом. Некоторые инструменты используют очень ограниченный набор эксплойтов, а потому опытные тестировщики используют не один, а целый набор инструментов, дополняя его ручными методами проверок. Например, есть

приложения сфокусированные на поиске и тестировании только уязвимостей связанных с SQL-инъекциями. Инструменты корпоративного класса должны быть более многофункциональными и включать ряд критически важных для web-приложения классов уязвимостей, таких как SQL-инъекции, межсайтовое исполнение скриптов и т.д. OWASP Top 10 является отличным базовым списком основных уязвимостей, но приложения корпоративного класса не должны ограничиваться проверкой только по одному списку или категории уязвимостей. Решение корпоративного класса также должно иметь возможность проверки нескольких приложений, поддерживать отслеживание в течение длительного времени и обеспечивать адекватной отчетностью (особенно в той части, что касается исполнения требований), а также быть настраиваемым для оценки выполнения локальных требований. Инструменты для оценки уязвимостей, как правило, являются программными, но могут иметь и аппаратную часть. Также они могут работать либо под контролем оператора, либо проводить сканирование в автоматическом режиме по расписанию. Так как web-приложения изменяются достаточно активно, то очень важно сканировать их после внесения любых изменений или новые версии перед развертыванием, а также контролировать действующие приложения на постоянной основе.

10. Перечислите сферы действия безопасности приложений

Ответы:

-

Верный ответ: Безопасность приложений обеспечивает защиту критических данных, вычисляемых, используемых, хранимых и передаваемых приложением, как требуется организации. Эта защита обеспечивает уверенность не только в доступности, целостности и конфиденциальности данных, но также в неотказуемости и аутентификации пользователей, имеющих к ним доступ. Критичность данных и иных активов должна определяться организацией посредством процесса оценки риска безопасности. Нуждающиеся в защите критические данные также могут представлять собой исходный код приложения, двоичный код и исполняемый код. На рисунке 2 показано графическое представление сферы действия безопасности приложений в виде области, ограниченной пунктирными линиями. Это представление не означает, что все элементы в показанной выше сфере действия являются частью приложения, а говорит о том, что все эти элементы требуют защиты для обеспечения безопасности приложения. Таким образом, сфера действия безопасности приложения является более широкой, чем сфера действия самого приложения. Приведенная ниже таблица иллюстрирует это отличие. Бизнес-контекст - Регулятивный контекст Процессы жизненного цикла приложений - Процессы, связанные с приложениями Технологический контекст - Спецификации приложений Прикладные данные - Данные организации и пользователей Роли и полномочия Сфера действия безопасности приложений

11. Что является источником требований безопасности приложений

Ответы:

-

Верный ответ: Согласно ИСО/МЭК 27005, требования безопасности приложений идентифицируются посредством оценки риска и обработки риска и диктуются такими факторами, как спецификации приложений, целевая среда приложений (бизнес-контекст, регулятивный и технологический контексты), критические данные и выбор, который делает владелец приложений. Функциональные требования безопасности диктуют, какие функциональные возможности безопасности будут реализованы в приложении. Нефункциональные требования безопасности направлены на качество безопасности, которое должно проявлять приложение. Все

эти меры и средства контроля и управления должны быть полностью определены и утверждены организацией.

12. Какие бывают техники атак при SQL-инъекциях

Ответы:

-

Верный ответ: Сообществом OWASP были описаны пять основных методов (техник) атак при SQL-инъекциях. Оператор Union: подход может быть использован при наличии уязвимости в запросе SELECT, позволяющей объединить два запроса в один результат или набор результатов. Логический метод: предполагает использование логического условия, либо условий, позволяющих достоверно определить истинность или ложность некоего предположения. На основании ошибок: этот метод предполагает намеренную передачу различных некорректных запросов, с целью вынудить web-приложение выдать информацию об ошибке, на основании которой, злоумышленник может составить и передать корректный инжектированный запрос. Метод с альтернативным каналом передачи данных: метод предполагает использование альтернативного канала передачи извлеченных данных (например через исходящее HTTP соединение с web-сервером) Time delay: метод использует команды базы данных, например sleep для того чтобы определить задержу по условным запросам. Метод эффективен, когда нет возможности получить ответ от web-приложения (результат, ошибка) Кроме того, может использоваться комбинированный подход, включающий в себя сочетание двух или более перечисленных техник. По способу извлечения данных выделяют три типа атак: Связанный: данные в результате инжектированного SQL запроса извлекаются тем же путем, которым был передан сам инжектированный запрос. Это самый прямолинейный вид атаки, в результате которого, запрошенные модифицированным запросом данные, отображаются непосредственно на странице web-приложения. Не связанный: данные в результате инжектированного SQL запроса извлекаются путем, отличным от способа передачи модифицированного запроса. (например данные передаются злоумышленнику в сообщении на электронную почту) Дедуктивный или слепой: В результате SQL инъекции фактического извлечения данных не происходит, но злоумышленник может получить информацию, наблюдая за поведением web-сервера, в результате отправки серии специфических инжектированных SQL запросов.

13. Понятие DevSecOps

Ответы:

-

Верный ответ: DevSecOps – это методология разработки ПО, включающая процессы, инструменты и методы защиты приложений от угроз на протяжении всего цикла. Это комплексный, автоматизированный, контролируемый на всех этапах процесс.

II. Описание шкалы оценивания

Оценка: зачтено
Описание характеристики выполнения знания: Работа выполнена верно или с несущественными недостатками

Оценка: не зачтено
Описание характеристики выполнения знания: Работа не выполнена или выполнена преимущественно неправильно

III. Правила выставления итоговой оценки по курсу