Russian Software Testing
Qualifications Board

Сертифицированный тестировщик

Программа обучения Продвинутого уровня
Тестировщик безопасности

Версия 2016

International Software Testing Qualifications Board

Уведомление об авторских правах

История изменений

Версия	Дата	Комментарии
0.1	24 Апреля 2015	Базовая версия, созданная на основе существующего проекта программы Expert Security Tester версии 3.9.
0.2	15 Июня 2015	Консолидированный вклад авторов после встречи авторов в Осло
1.0 - Beta	20 Сентября 2015	Бета-версия - включены комментарии к альфа-версии
1.0 – GA Candidate	4 Марта 2016	После проверки Экзаменационной рабочей группы LO 4.1.2 изменен с K2 и K3 и переформулирован соответствующим образом. Текст уже адекватно поддерживает K3 LO.
1.0 - GA	18 Марта 2016	Выпуск GA - включены комментарии к бета-версии
1.0 - GA	19 Апреля 2019	Обновленное уведомление об авторских правах
RSTQB 2022	7 Июня 2022	Перевод на русский язык

Оглавление

Благодарности

Перевод версии документа 2022 года выполнен рабочей группой Продвинутого Уровня АНО «Коллегия экспертов по качеству программного обеспечения» (Russian Software Testing Qualifications Board): Павел Шариков (руководитель группы), Александр Александров (редактор).

Этот документ был подготовлен основной командой рабочей группы Продвинутого Уровня из International Software Testing Qualifications Board.

Основная команда благодарит группу проверки и все национальные коллегии за их предложения и вклад.

На момент завершения программы обучения «Тестировщик безопасности Продвинутого уровня» рабочая группа имела следующий состав:

Авторы основной группы для программы обучения «Тестировщик безопасности Продвинутого уровня»: Randall Rice (председатель), Hugh Tazwell Daughtrey (вице-председатель), Frans Dijkman, Joel Oliveira, Alain Ribault.

В редактировании и предоставлении замечаний принимали участие (в алфавитном порядке): Tarun Banga, Clive Bates, Hugh Tazwell Daughtrey (вице-председатель), Frans Dijkman (автор), Christian Alexander Graf, Wenda Hu, Matthias Hamburg, Prof. Dr. Stefan Karsch, Sebastian Malyska, Satoshi Masuda, Gary Mogyorodi, Raine Moilanen, Joel Oliveira, Meile Posthuma, Alain Ribault, Randall Rice (председатель), Ian Ross, Kwangik Seo, Dave van Stein, Ernst von Düring, Attila Toth, Wei Xue, Dr. Nor Adnan Yahaya, Xiaofeng Yang, Wenqiang Zheng, Ping Zuo.

Кроме того, мы выражаем признательность и благодарность лидерам и членам рабочей группы на уровне экспертов за их раннее и постоянное руководство: Graham Bath (председатель, рабочая группа Экспертного Уровня), Judy McKay (вице-председатель, рабочая группа Экспертного Уровня).

Документ был официально выпущен Генеральной Ассамблеей ISTQB 18 Марта 2016 года.

0. Предисловие к программе обучения

0.1 Цель этого документа

Программа обучения представляет собой основу международной сертификации на квалификацию «Тестировщик безопасности Продвинутого уровня» в области тестирования программного обеспечения. ISTQB® распространяет эту программу:

1. Национальным коллегиям для перевода на национальный язык и аккредитации организаторов обучения. Национальные коллегии могут приспособить программу обучения к особенностям конкретных языков и определить ссылки для адаптации к местным публикациям.
2. Экзаменационным комиссиям для формирования экзаменационных вопросов на национальном языке, адаптированные к целям обучения каждого курса.
3. Организаторам обучения для разработки программы обучения и определения соответствующих методов обучения.
4. Кандидатам на получение сертификатов для подготовки к экзамену (в рамках программы обучения или независимо).
5. Международному сообществу разработки ПО и систем для повышения уровня профессионализма при тестировании ПО и систем, и использования как основы для книг и статей.

ISTQB® может разрешить другим лицам использовать эту программу обучения в своих целях при условии, что они обратятся за письменным разрешением.

0.2 Обзор

Квалификация тестировщика безопасности Продвинутого уровня предназначена для людей, которые уже достигли Продвинутого уровня в своей карьере в области тестирования программного обеспечения и хотят развивать свой опыт в области тестирования безопасности. Модули, предлагаемые на Продвинутом уровне, охватывают широкий спектр тем тестирования.

Для получения сертификата Продвинутого уровня модуля «Тестировщик безопасности» кандидаты должны иметь сертификат Сертифицированный тестировщик ПО Базового уровня и подтвердить экзаменационной комиссии, что они имеют достаточный практический опыт для сертификации на Продвинутый уровень, который должен быть не менее трех лет соответствующего академического, практического или консультационного опыта. Обратитесь в соответствующую экзаменационную комиссию, чтобы определить их конкретные критерии практического опыта.

0.3 Экзамен

Все экзамены, проводимые на Продвинутом уровне для этого модуля, основаны на данной программе «Тестировщик безопасности Продвинутого уровня».

Формат экзамена определяется руководством по экзамену Продвинутого ISTQB.

Экзамены можно сдавать как часть аккредитованного курса обучения или независимо (например, в экзаменационном центре или на открытом экзамене). Экзамены можно сдавать в бумажном или электронном виде, но все экзамены должны проводиться под наблюдением (под контролем лица, уполномоченного национальной или экзаменационной коллегией).

0.4 Как организована данная учебная программа

Есть десять глав. Заголовок верхнего уровня показывает время необходимой для изучения главы. Например:

1. Основы тестирования безопасности 105 мин.

показывает, что на изучение материала 1 главы отводится 105 минут.

Конкретные цели обучения перечислены в начале каждой главы.

0.5 Определения

Многие термины, используемые в литературе по программному обеспечению, используются взаимозаменяемо. В то время как кандидатам на экзаменах Базового и Продвинутого уровня могут задаваться вопросы, основанные только на Стандартном глоссарии терминов ISTQB, дополнительно ожидается, что кандидаты на этом уровне должны знать и уметь работать с различными определениями.

Примечание: “Обеспечение информации” (IA) упоминается только в разделе 2.4. За цитатой в 2.4.3 следует утверждение, что IA следует рассматривать шире, чем “тестирование безопасности”, точно так же, как QA следует рассматривать шире, чем тестирование программного обеспечения.

«Информационная безопасность» используется в разделах 2.2, 2.3.1, 2.7.2, 6 (Основные сведения), 6.1.3 и на протяжении всей главы 9.

Термин “кибербезопасность”, который в некоторых кругах теперь называют IA, не используется.

Ключевые слова, перечисленные в начале каждой главы этой программы Продвинутого уровня, определены либо в стандартном глоссарии терминов, используемых в тестировании программного обеспечения, опубликованном ISTQB, либо приведены в упомянутой литературе.

0.6 Уровень детализации

Уровень детализации этой программы обучения позволяет проводить обучение и экзамены на международном уровне. Для достижения этой цели учебный план включает:

- Общие учебные цели, описывающие замысел Продвинутого уровня
- Цели обучения для каждой области знаний с описанием когнитивных результатов обучения и мышления, которые необходимо достичь
- Список информации для обучения, включая описание и ссылки на дополнительные источники, если это необходимо
- Описание ключевых концепций для обучения, включая такие источники, как общепринятая литература или стандарты

- В этой программе обучения могут упоминаться определенные инструменты, методы и товарные знаки. Эта учебная программа не предназначена для продвижения или рекомендации какого-либо конкретного решения по обеспечению безопасности.

Содержание программы обучения не является описанием всей области знаний для Продвинутых тестировщиков безопасности; оно отражает уровень детализации, который должен быть охвачен в учебном курсе для продвинутых тестировщиков безопасности.

0.7 Цели обучения / уровень знаний

Содержание этой программы обучения, термины и основные элементы (цели) всех перечисленных стандартов должны быть, по крайней мере, запомнены (K1) и поняты (K2), даже если они не упомянуты явно в целях обучения.

Следующие цели обучения определены как применимые к данной программе обучения. Каждая тема в программе обучения будет рассмотрена в соответствии с поставленной перед ней целью обучения.

Уровень 1: Запомнить (K1)
Кандидат узнает, запоминает и вспоминает термин или понятие.
Ключевые слова: Помнить, вспоминать, узнавать, знать.
Пример
Может распознать определение «риска» как:

- фактор, который может привести к негативным последствиям в будущем; обычно выражается как влияние и вероятность.

Уровень 2: Понимать (K2)
Кандидат может выбирать причины или пояснения к утверждениям, относящимся к теме, а также может обобщать, дифференцировать, классифицировать и приводить примеры фактов (например, сравнивать термины), концепции тестирования, процедуры тестирования (объясняя последовательность заданий).
Ключевые слова: Обобщать, классифицировать, сравнивать, сопоставлять, противопоставлять, приводить примеры, интерпретировать, переводить, представлять, делать выводы, заключать, классифицировать.
Примеры
Объяснить причину, по которой тесты безопасности должны быть разработаны как можно раньше:

- Для обнаружения дефектов и уязвимостей в системе безопасности, при условии, что их устранение обойдется дешевле
- Чтобы избежать создания системы или приложения, подверженных постоянному исправлению уязвимостей безопасности

Уровень 3: Применять (K3)
Кандидат может выбрать правильное применение концепции или метода и применить его к данному контексту. K3 обычно применим к процедурным знаниям. Здесь нет никакого творческого акта, такого как оценка программного приложения или создание модели для данной программы. Когда предоставляется модель и в программе объясняются процедурные шаги, необходимые для создания тестовых примеров на основе этой модели, то это K3.
Ключевые слова: Внедрять, выполнять, использовать, следовать процедуре, применять процедуру.

Пример

- Использовать общую процедуру создания тестового сценария безопасности, и выбрать тестовые сценарии из заданной диаграммы таблицы переходов, чтобы охватить все переходы.

Уровень 4: Анализировать (K4)
Кандидат может разделить информацию, относящуюся к процедуре или методу, на составные части для лучшего понимания, а также может различать факты и выводы. Типичное применение — проанализировать документ, программное обеспечение, проектную ситуацию и предложить соответствующие действия для решения проблемы или задачи.
Ключевые слова: Анализировать, дифференцировать, выбирать, структурировать, фокусировать, приписывать, деконструировать, оценивать, судить, контролировать, координировать, создавать, синтезировать, генерировать, выдвигать гипотезы, планировать, проектировать, конструировать, производить.
Пример

- Проанализировать риски безопасности продукта и предложить превентивные и корректирующие действия по снижению рисков.
- Выбрать средства тестирования безопасности, которые были бы наиболее подходящими в данной ситуации с учетом прошлых неудач в области безопасности.

Ссылки (Для целей обучения)
Bloom, B. S. (1956). *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain,* David McKay, Co. Inc.
Anderson, L. W. and Krathwohl, D. R. (eds) (2001). *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives*, Allyn & Bacon.

1 Основы тестирования безопасности - 105 мин.

Ключевые слова

конфиденциальность данных, этичный взломщик, информационная безопасность, тестирование на проникновение, оценка риска, подверженность риску, смягчение рисков, атака на систему безопасности, аудит безопасности, политика безопасности, процедура безопасности, риск безопасности

Цели обучения для основ тестирования безопасности

1.1 Риски безопасности

AS-1.1.1 (K2) Понимать роль оценки рисков в предоставлении информации для планирования и разработки тестов безопасности и приведения тестирования безопасности в соответствие с потребностями бизнеса

AS-1.1.2 (K4) Уметь определять значительные активы, подлежащие защите, стоимость каждого актива и данные, необходимые для оценки уровня безопасности, должного для каждого актива

AS-1.1.3 (K4) Анализировать эффективное использование методов оценки рисков в конкретной ситуации для выявления текущих и будущих угроз безопасности

1.2 Политики и процедуры информационной безопасности

AS-1.2.1 (K2) Понимать концепцию политик и процедур безопасности и то, как они применяются в информационных системах

AS-1.2.2 (K4) Уметь анализировать заданный набор политик и процедур безопасности вместе с результатами тестов безопасности, чтобы определить их эффективность

1.3 Аудит безопасности и его роль в тестировании безопасности

AS-1.3.1 (K2) Понимать цель аудита безопасности

Функциональное тестирование основано на множестве элементов, таких как риски, требования, сценарии использования и модели. Тестирование безопасности основано на аспектах множества элементов спецификаций, но также направлено на проверку и подтверждение рисков безопасности, процедур и политик безопасности, поведения злоумышленников и известных уязвимостей безопасности.

1.1 Риски безопасности

1.1.1 Роль оценки рисков в тестировании безопасности

Цели тестирования безопасности основаны на рисках безопасности. Эти риски выявляются путём проведения оценки рисков безопасности. Общие методы управления рисками описаны в [ISTQB_FL_SYL] и [ISTQB_ATM_SYL].

Риск характеризует меру, в которой предприятию угрожает потенциальное обстоятельство или событие, и обычно зависит от:

- Неблагоприятных воздействий, которые могут возникнуть в случае наступления обстоятельства или события, и
- Вероятности возникновения.

Риски информационной безопасности - это те риски, которые возникают в результате нарушения конфиденциальности, целостности или доступности информации или информационных систем и отражают потенциальные неблагоприятные воздействия на деятельность организации (т. е. цель, функции, имидж или репутацию организации), активы организации, отдельных лиц, другие организации и страну. [NIST 800-30]

Роль оценки рисков безопасности состоит в том, чтобы позволить организации понять, какие области и активы могут быть подвержены риску, и определить величину каждого риска. Для тестировщиков безопасности оценка рисков безопасности может быть богатым источником информации, на основе которой можно планировать и разрабатывать тесты безопасности. Кроме того, оценка рисков безопасности может использоваться для определения приоритетов тестов безопасности, чтобы максимальный уровень тщательности тестирования и покрытия можно было сосредоточить на областях с наибольшим риском.

Приоритезация тестов безопасности на основе оценки рисков безопасности позволяет согласовать тесты в соответствии с целями безопасности бизнеса. Однако для того, чтобы это согласование произошло, оценка рисков безопасности должна точно отражать угрозы безопасности организации, затронутые заинтересованные стороны и активы, которые необходимо защитить.

Важно понимать, что любая оценка риска (безопасности или иного) является лишь моментальным снимком в определенный момент времени, основанным на ограниченной информации, которая может привести к неверным предположениям и выводам. Риски безопасности постоянно меняются в организации и проектах, например, поскольку новые угрозы появляются ежедневно. Поэтому оценка рисков безопасности должна проводиться регулярно. Точный интервал времени для проведения оценки рисков безопасности зависит от организации и степени изменений, с которыми она сталкивается (происходящих в ней). Некоторые организации проводят оценку рисков безопасности раз в три-шесть месяцев, в то время как другие проводят оценку еженедельно или ежегодно.

Еще одна проблема, связанная с оценкой рисков, - это уровень знаний участников. Риски можно упустить из-за отсутствия подробной информации. Кроме того, риски можно упустить, если люди не понимают угрозы и риски безопасности. По этой причине полезно запрашивать сведения у самых разных людей и уделять особое внимание уровню детализации информации, которую они предоставляют.

Вполне реально ожидать, что могут быть сделаны неверные предположения, приводящие к тому, что важные риски безопасности будут упущены при оценке. Способы борьбы с отсутствием или неполнотой информации о рисках включают использование установленной методологии оценки рисков безопасности в качестве контрольного списка и получение информации от нескольких людей. Один из таких методов можно найти в [NIST 800-30].

1.1.2 Определение активов

Не вся информация, подлежащая защите, находится в цифровом формате, например, копии документов (контракты, планы, письменные заметки, логины и пароли в письменной форме). Хотя эта информация и не в цифровом формате, она может иметь большую ценность. Поэтому возникает вопрос, какая информация является цифровой, а какая нет? Возможно, защищаемый актив существует как в цифровом, так и в физическом формате. При определении активов, подлежащих защите, следует задать следующие вопросы:

Какие активы являются ценными для организации?
Примеры конфиденциальной информации высокой ценности включают:

- Данные о клиентах
- Бизнес-планы
- Собственное программное обеспечение, разработанное компанией
- Системную документацию
- Рисунки и диаграммы, являющиеся собственностью компании
- Интеллектуальная собственность (например, процессы, коммерческие секреты)
- Финансовые электронные таблицы
- Презентации и учебные курсы
- Документы
- Электронную почту
- Документы сотрудников
- Налоговые декларации

Хотя многие активы основаны на информации, возможно, что некоторые активы в организации имеют физический или нематериальный характер (природу). Примеры этих активов могут включать:

- Физические прототипы новых устройств в стадии разработки
- Возможность предоставлять услуги
- Репутацию и доверие к компании

Насколько ценен актив?
Многие чувствительные активы имеют материальную ценность. Другие оцениваются больше по затратам и последствиям их потери. Например, что конкурент сделал бы с вашим бизнес-планом?

Ценность бывает трудно оценить с уверенностью; однако некоторые методы определения стоимости цифровых активов включают:

- Будущий доход от актива

- Ценность для конкурента, который может получить информацию
- Время и усилия, необходимые для воссоздания актива
- Штрафы и санкции за невозможность предоставить необходимую информацию, например, для аудита или судебного процесса
- Штрафы и санкции за потерю данных клиента

Где расположены цифровые активы?
В прошлом цифровые активы находились на серверах, компьютерах или периферийных устройствах, таких как жесткие диски или компакт-диски. Хотя это устаревший и неорганизованный подход, конфиденциальные данные все еще могут быть на старых компакт-дисках, DVD-дисках и USB-накопителях. Более надежным способом хранения цифровых активов является использование защищенных корпоративных серверов с надежным шифрованием всех конфиденциальных данных. Для доступа к конфиденциальным данным, хранящимся на защищенных серверах, требуются аутентификация и авторизация. Кроме того, могут потребоваться другие средства защиты, такие как цифровые сертификаты для доступа к конфиденциальной информации через Интернет.

Хранение данных меняется. Теперь большие объемы бизнес-данных могут храниться на мобильных устройствах, таких как смартфоны, планшеты и флэш-накопители. Когда цифровая информация переносится в облачное хранилище, возникает новый набор проблем безопасности, связанных с доступом к данным.

Важность вопроса хранения данных обусловлена случаями в прошлом, когда люди, которым были доверены конфиденциальные данные, просто выходили из здания компании с жестким диском, заполненным частными данными клиентов и бизнеса. Один из таких случаев в США был связан с жестким диском, украденным из охраняемого помещения в правительственном агентстве безопасности, на котором хранилась платежная ведомость и банковская информация более чем 100 000 нынешних и бывших работников. [Washington Post, 2007].

Как осуществляется доступ к цифровым активам?
Общие методы доступа к цифровым активам включают:
- Доступ к компьютеру через локальную сеть или сети Wi-Fi
- Удалённый доступ через виртуальную частную сеть (VPN) или облачный диск
- Передача физических хранилищ данных (компакт-диски, DVD- и USB-накопители) от человека к человеку, что является низко технологичной, но очень распространенной практикой
- Отправка файлов по электронной почте

Как обеспечивается сохранность цифровых активов?
Существует несколько способов защиты цифровых активов, включая:
- Шифрование (какой тип и степень защиты, у кого есть ключи?)
- Аутентификация и токены (Требуются ли цифровые сертификаты? Являются ли политики паролей адекватными и соблюдаются ли они?)
- Авторизация (Какие уровни привилегий были предоставлены пользователям, работающим с цифровыми активами?)

1.1.3 Анализ методов оценки рисков

Процесс оценки рисков безопасности очень похож на стандартную оценку рисков, с основным отличием в том, что основное внимание уделяется областям, связанным с безопасностью.

Оценка риска безопасности должна включать точку зрения внешних заинтересованных сторон, участвующих в тестировании безопасности (т.е. людей или сторон, вовлеченных в проект или продукт, которые находятся за пределами компании и имеют четкую заинтересованность в безопасности проекта/продукта). К таким заинтересованным сторонам относятся:

- Клиенты и пользователи. Полезно для понимания перспектив, получения информации для тестирования безопасности и установления хорошего общения
- Граждане и общество. Важно донести, что информационная безопасность - это усилия и ответственность всего сообщества
- Регулирующие органы. Необходимы для обеспечения соответствия действующему законодательству в области информационной безопасности

Подготовка к оценке риска включает следующие задачи [NIST 800-30]:

- Определить цель оценки
- Определить объём оценки
- Определить допущения и ограничения, связанные с оценкой
- Определить источники информации, которые будут использоваться в качестве исходных данных для оценки
- Определить модель риска и аналитические подходы (т.е. подходы к оценке и анализу), которые будут использоваться во время оценки

Проведение оценки рисков включает следующие задачи [NIST 800-30]:

- Определить источники угроз, которые имеют отношение к организации
- Определить события угрозы, которые могут быть порождены этими источниками
- Определить уязвимости в организации, которые могут быть использованы источниками угроз посредством конкретных угрожающих событий, и предрасполагающие условия, которые могут повлиять на успешную эксплуатацию
- Определить вероятность того, что идентифицированные источники угроз инициируют конкретные угрожающие события, и вероятность того, что эти события будут успешными
- Определить негативные последствия для операций и активов организации, отдельных лиц, других организаций и страны в результате использования уязвимостей источниками угроз

Общение и обмен информацией состоит из следующих задач [NIST 800-30]:

- Передача результатов оценки риска
- Обмен информацией, разработанной в ходе выполнения оценки риска, для поддержки других мероприятий по управлению рисками

1.2 Политики и процедуры информационной безопасности

1.2.1 Понимание политик и процедур безопасности

Обычно политики информационной безопасности в разных организациях различаются в зависимости от бизнес-модели, специфики отрасли и уникальных рисков безопасности, с которыми сталкивается организация. Цели политик безопасности схожи даже при большом диапазоне различий. Основой всех политик безопасности должна быть оценка рисков безопасности, в ходе которой изучаются конкретные угрозы безопасности и то, как они влияют на организацию. [Jackson, 2010].

Примеры политик безопасности включают, но не ограничиваются следующими: [Jackson, 2010]:

Допустимое использование – Эта политика определяет правила, которых должен придерживаться пользователь компьютерной системы, чтобы соответствовать политике и процедурам безопасности организации. Эта политика охватывает как приемлемое, так и неприемлемое поведение при использовании цифровых ресурсов, таких как сети, веб-сайты и данные. Кроме того, политика может применяться как к внутренним, так и к внешним пользователям систем организации. Важно, чтобы пользователи системы всегда понимали эту политику и следовали ей. Чтобы предотвратить путаницу и случайные нарушения политики, в ней должны быть определены конкретные правила, касающиеся приемлемого поведения, неприемлемого поведения и требуемого поведения.

Минимальный доступ – Эта политика определяет минимальные уровни доступа, необходимые для выполнения определённых задач. Цель этой политики - предотвратить предоставление пользователям прав доступа, превышающих те, которые необходимы для выполнения их задач. Наличие прав доступа выше, чем необходимо, может создать возможности непреднамеренного или преднамеренного злоупотребления привилегиями пользователя.

Доступ к сети – Эта политика определяет критерии доступа к различным типам сетей, таким как локальные сети (LAN) и беспроводные сети. Кроме того, эта политика может определять допустимые и недопустимые действия в сети. Эта политика часто запрещает пользователям добавлять в сеть несанкционированные устройства, такие как маршрутизаторы и точки доступа.

Удалённый доступ – Эта политика требует того, что необходимо для предоставления удаленного доступа к сети как внутренним сотрудникам, так и внешним (не являющимся сотрудниками) пользователям. Использование VPN часто рассматривается в этой политике.

Доступ к Интернету – Эта политика определяет допустимое использование Интернета сотрудниками и гостями организации. В сферу действия этой политики входят типы веб-сайтов, которые можно и нельзя посещать, например, сайты азартных игр или порнографии, а также вопросы о том, разрешено ли нерабочее использование Интернета. Хотя некоторые из пунктов этой политики могут быть также рассмотрены в политике приемлемого использования, некоторые организации предпочитают определять эту политику отдельно из-за большого количества людей, ведущих бизнес в Интернете.

Управление учетными записями пользователей – Эта политика определяет создание, ведение и удаление учётных записей пользователей. Регулярный аудит учетных записей пользователей также рассматривается для обеспечения соответствия политике.

Классификация данных – Существует множество способов классификации данных с точки зрения безопасности. В этой программе обучения термин «конфиденциальные данные» используется как общий термин для любых данных, которые должны быть защищены во избежание потери. Политика классификации данных определяет различные типы данных, которые считаются конфиденциальными и должны быть защищены. Имея политику классификации данных, организация может создать средства контроля для защиты данных на основе их ценности для организации и ее клиентов. Как правило, область бизнеса, которая создает данные, отвечает за их классификацию на основе стандартной структуры классификации.

Ниже приведен пример структуры классификации данных (из бизнес-контекста):

- Публичные: Любой человек внутри или вне организации может просматривать эти данные (например, документы и веб-страницы, обращенные наружу).

- Конфиденциальные: Обычно это классификация по умолчанию для любых документов, созданных внутри организации. Эти документы могут включать электронную почту, отчеты и презентации, которые используются внутри организации. Примером может быть отчет о продажах. С информацией такого уровня должны иметь возможность работать только авторизованные пользователи этих данных. Перед передачей такого рода информации третьим лицам, например, консультантам, часто требуется заключение соглашений о неразглашении.

- Конфиденциальные высокого уровня: Это более высокий уровень конфиденциальности для конфиденциальной информации, которая должна быть доступна только определенным людям в организации. Сюда относится такая информация, как коммерческая тайна, стратегические планы, дизайн продукции и непубличные финансовые данные. Обмен данными такого типа не допускается, кроме как с явного разрешения владельца данных.

- Частные: Это информация, доступ к которой часто ограничивается должностными лицами организации, которые должны иметь специальное разрешение на доступ к ней. В случае разглашения эта информация может иметь серьезные негативные последствия, например, финансовый ущерб, для организации. Из-за высокого риска, связанного с потерей, частная информация должна быть защищена с особой тщательностью. Эти данные могут включать информацию об исследованиях и разработках, планы слияний и поглощений, а также информацию о клиентах, например, данные кредитных карт и счетов.

- Секретные: В корпоративном контексте это информация, которую организация получает от внешней стороны для внесения изменений, но которая не должна стать известной внутри или за пределами организации. Примером в корпоративном контексте может быть проектный документ, созданный консультантом, работающим над новым типом технологии, которая предполагает сотрудничество с другими компаниями, каждая из которых должна хранить информацию на секретном уровне до тех пор, пока технология не будет готова к раскрытию. Она сравнима с конфиденциального высокого уровня с той разницей, что может не иметь ощутимой ценности для самой организации. В этом отношении она отличается от коммерческой тайны. Однако раскрытие секретной информации может нанести вред организации, другим организациям или стране. В военном и правительственном контексте это информация, которая может быть разработана или получена, но должна быть известна только людям с определенным уровнем допуска. В военном контексте это включает детали научных или исследовательских проектов, содержат в себе новые технологические разработки или методы, имеющие прямое военное применение, жизненно важное для национальной обороны.

Управление конфигурацией и изменениями – Эта политика может иметь обычный операционный контекст, например, описывать, как управляются и конфигурируются изменения в системах, чтобы предотвратить перебои в работе из-за неожиданного воздействия. С точки зрения безопасности, управление конфигурацией контролирует, как параметры безопасности применяются к защищенным устройствам и приложениям. Риск заключается в том, что несанкционированное изменение в защищенном устройстве может привести к уязвимости безопасности, которая может остаться незамеченной.

Другой риск заключается в том, что несанкционированное изменение кода или конфигурации приложения может создать уязвимость безопасности. Эта политика включает стандартные конфигурации, которые должны использоваться, процесс утверждения всех изменений и процесс

отката в случае возникновения проблем. Эта политика может применяться ко всем ИТ-услугам, приложениям и устройствам в организации.

Безопасность серверов – Эта политика налагает ответственность на владельца (владельцев) сервера за соблюдение корпоративных методов обеспечения безопасности, а также передовых отраслевых методов установки, конфигурирования и эксплуатации серверов и систем. Кроме того, необходимо определить и поддерживать базовые конфигурации. Примеры практик, описанных в этой политике, включают требования безопасности, резервное копирование и восстановление, а также ограничение активных служб только теми, которые необходимы для работы приложений. Также в эту политику могут быть включены требования мониторинга и аудита для обеспечения правильной конфигурации и обновления сервера.

Мобильные устройства – Мобильные устройства имеют уникальный набор проблем безопасности, поэтому может потребоваться отдельная политика именно для мобильных устройств. Например, ноутбуки и смартфоны можно легко потерять или украсть, что может привести к потере корпоративных и частных данных. Эти устройства также имеют высокий риск контакта с вредоносным ПО. Указанные риски требуют определённых правил и мер предосторожности, которые необходимо соблюдать для снижения рисков и ограничения подверженности организации угрозам безопасности. Такая политика может включать требования к тому, какие данные должны быть зашифрованы, к установке и поддержке актуальных версий антивирусного программного обеспечения, а также к тому, когда для доступа к устройству необходимо использовать пароли. Кроме того, в этой политике определяются типы организационной информации, которая может находиться на мобильных устройствах. Физическая безопасность также может быть рассмотрена, например, наличие кабельных замков для портативных компьютеров и процедур для сообщения о потерянных или украденных устройствах.

Гостевой доступ – Эта политика определяет методы, которые должны применяться для защиты организации, при этом позволяя компании принимать гостей и других лиц в организационных сетях. Одним из аспектов этой политики является требование к гостям ознакомиться с политикой приемлемого использования и согласиться с ней, прежде чем предоставлять им доступ к сети. Эта политика может быть реализована различными способами, например, гости должны подписать политику приемлемого использования, а затем ввести предоставленный им код временного доступа. Основная цель этой политики заключается в том, чтобы обеспечить соблюдение стандартов безопасности организации и при этом предусмотреть процедуры предоставления гостям доступа к сети или Интернету.

Физическая безопасность – Эта политика определяет средства контроля, необходимые для физических объектов, поскольку нахождение в физической близости от защищённых устройств может увеличить риск нарушения безопасности. Эта политика может также охватывать другие риски, такие как нарушение энергоснабжения, кража, пожар и стихийные бедствия. Здесь также рассматривается вопрос о том, какие устройства можно вносить или выносить из компании, особенно в тех местах, где хранится конфиденциальная информация.

Политика паролей – Эта политика определяет минимальные требования к надежным паролям и другим методам безопасного использования паролей, таким, как продолжительность времени между обязательными сменами паролей, способы защиты конфиденциальности паролей (например, отказ от использования функции «запомнить пароль» в браузерах, запрет на совместное использование паролей и запрет на передачу паролей по электронной почте). Эта политика может применяться к приложениям, учетным записям пользователей и любым другим местам, где требуются пароли.

Защита от вредоносного ПО – Эта политика определяет систему защиты и поведения для предотвращения, обнаружения и удаления вредоносного ПО. Поскольку вредоносные и

шпионские программы могут быть получены из различных источников, это важная политика, которую должен понимать и соблюдать каждый сотрудник организации. Например, эта политика может ограничивать использование USB-накопителей.

Реагирование на инциденты – Эта политика описывает, как реагировать на инциденты, связанные с безопасностью. Эти инциденты могут варьироваться от обнаружения вредоносных программ и нарушений политики приемлемого использования до несанкционированного доступа к конфиденциальным данным. Важно разработать эту политику до того, как произойдет инцидент, чтобы избежать необходимости определять соответствующие ответные меры в каждом конкретном случае. В этой политике также рассматриваются вопросы коммуникации, включая реакцию СМИ и уведомление правоохранительных органов.

Политика аудита – Эта политика разрешает аудиторам запрашивать доступ к системам с целью проведения аудита. Аудиторской группе может понадобиться доступ к данным журналов, записям сетевого трафика и другим криминалистическим данным.

Лицензирование программного обеспечения – Эта политика касается того, как организация получает и лицензирует используемое ею программное обеспечение. Если лицензии на коммерческое программное обеспечение нарушаются, организация рискует получить штрафы и судебные иски. В связи с этим важно, чтобы лицензии были идентифицированы и контролировались. Загрузка и установка не одобренного программного обеспечения - ключевой запрет, часто встречающийся в этой политике.

Электронный мониторинг и конфиденциальность – Организации имеют право и несут ответственность за мониторинг электронных коммуникаций с использованием оборудования и ресурсов компании. Это включает переписку по электронной почте и социальные сети. Данная политика описывает, какой мониторинг осуществляется организацией и какие данные подлежат сбору. Законы разных стран отличаются, поэтому перед составлением такой политики необходимо проконсультироваться с юристом. [Jackson, 2010].

Процедуры безопасности
Процедуры безопасности определяют шаги, которые необходимо предпринять для реализации конкретной политики или контроля, а также шаги, которые необходимо предпринять в ответ на конкретный инцидент безопасности. Официальные, документированные процедуры способствуют реализации политики безопасности и обязательных мер контроля.

Политики, стандарты и руководства описывают средства контроля безопасности, которые должны быть установлены, в то время как процедура описывает специфику, объясняя, как реализовать средства контроля безопасности пошагово. Например, процедура может быть написана для объяснения того, как предоставлять уровни доступа пользователям, подробно описывая каждый шаг, который необходимо предпринять для обеспечения правильного уровня доступа, чтобы права пользователя соответствовали применимой политике, стандартам и руководствам.

1.2.2 Анализ политик и процедур безопасности

Перед оценкой набора политик и процедур безопасности важно определить цель (цели) оценки и набор критериев, по которым можно судить об адекватности политик и процедур. В некоторых случаях критерии могут быть определены стандартами, такими как COBIT [COBIT], ISO27001 [ISO27001] или PCI [PCI].

Кроме того, необходимо определить:

- Какие ресурсы необходимы с точки зрения навыков и знаний в конкретных оцениваемых областях

- Как измерить адекватность политик и процедур
- Что измерять и оценивать (например, эффективность, действенность, пригодность для использования, принятие)
- Доступность политик и процедур в организации
- Контрольный список для руководства оценкой и обеспечения последовательности

Контрольный список действует как руководство, которое указывает аудитору, куда смотреть и чего ожидать. Инструменты, такие как средства аудита паролей, могут быть полезны при тестировании определенных средств контроля, чтобы определить, достигают ли они своих целей, и для получения данных, которые могут быть использованы в дальнейшем при оценке рисков. Аудитор ищет «доказательства» соответствия политикам, средствам контроля и стандартам. Некоторые из перечисленных ниже задач являются статичными по своей природе, в то время как другие, такие как наблюдение за процессами в действии, являются динамичными. Аудитор делает следующее:

- Изучает системную документацию
- Опрашивает людей на предмет их восприятия эффективности политик и процедур
- Опрашивает ключевой персонал, вовлеченный в контролируемые процессы
- Наблюдает за работой систем и процессов
- Анализирует результаты предыдущих аудитов для выявления тенденций
- Анализирует журналы и отчеты
- Анализирует конфигурацию технических средств контроля, например, конфигурацию межсетевого экрана и системы обнаружения вторжений
- Анализирует образцы транзакций данных для выявления любых аномалий или подозрительных операций [Jackson, 2010]

Управление

Средства контроля безопасности — это технические или административные средства защиты или противодействия, позволяющие избежать, противостоять или минимизировать потери, или недоступность из-за угроз, действующих на соответствующую уязвимость, т.е. риск безопасности [Northcutt, 2014]. Например, контроль безопасности в системе расчёта заработной платы может заключаться в том, что два человека должны независимо друг от друга утвердить изменение информации о размере заработной платы сотрудника. Тестировщики безопасности должны знать о конкретных средствах контроля в своей организации и включать тесты данных средств в тесты безопасности.

Основные категории контроля безопасности - административный, технический и физический. В рамках каждой категории могут быть реализованы следующие виды контроля: превентивный, детективный, корректирующий или восстановительный. Эти типы контроля работают вместе, и в целом для эффективной защиты актива необходимо обеспечить контроль из каждой категории. [Jackson, 2010].

Список 20 важнейших критических элементов контроля безопасности можно найти на сайте www.sans.org. [Web-1].

Тесты безопасности

Основное отличие тестирования безопасности от статического анализа политик и процедур безопасности заключается в использовании результатов тестов, разработанных специально для проверки или подтверждения эффективности политик и процедур безопасности. Эти тесты сосредоточены на риске того, что политика безопасности может быть внедрена, может соблюдаться, но не является эффективной защитой активов.

Также при проведении оценки политики и процедур безопасности можно получить информацию о выполнении определённых задач. Тестирование безопасности этих задач может помочь

определить, насколько эффективны политики и процедуры безопасности на практике. Например, политика и процедура паролей может казаться разумной и эффективной на бумаге, но при использовании инструмента для взлома паролей процедура может не соответствовать поставленным целям.

Политики и процедуры безопасности могут быть источником тестов безопасности; однако тестировщик безопасности должен помнить, что атаки постоянно развиваются. Появляются новые атаки, и, как и в любом программном приложении, могут появиться новые дефекты - все это является причиной проведения тестов безопасности с точки зрения злоумышленника.

1.3 Аудит безопасности и его роль в тестировании безопасности

Аудит безопасности - это ручная проверка и оценка, которая выявляет слабые места в процессах и инфраструктуре безопасности организации. Аудит безопасности на процедурном уровне (например, для проверки внутреннего контроля) может проводиться вручную. Аудит безопасности на архитектурном уровне часто проводится с помощью инструментов аудита безопасности, которые могут быть согласованы с конкретным решением поставщика сетей, серверной архитектуры и рабочих станций.

Как и тестирование безопасности, аудит безопасности не гарантирует, что все уязвимости будут найдены. Однако аудит - это еще одно мероприятие обеспечения безопасности, позволяющее выявить проблемные области и указать, где требуется исправление.

В некоторых подходах к аудиту безопасности тестирование проводится как часть аудита. Однако объём аудита безопасности гораздо шире, чем тестирование безопасности. Аудит безопасности часто исследует такие области, как процедуры, политики и средства контроля, которые трудно протестировать прямым способом. Тестирование безопасности больше связано с технологиями поддержки безопасности, такими как конфигурация брандмауэра, правильное применение аутентификации и шифрования, а также применение прав пользователей.

Существует пять столпов аудита безопасности [Jackson, 2010]:

Оценка – Оценка документирует и определяет потенциальные угрозы, ключевые активы, политику и процедуры, а также терпимость руководства к риску. Оценки не являются одноразовыми мероприятиями. Поскольку окружающая среда и бизнес постоянно меняются, оценки должны проводиться на регулярной основе. Это также дает возможность узнать, насколько актуальны и эффективны политики безопасности.

Предотвращение – Выходит за рамки технологии и содержит административные, операционные и технические средства контроля. Предотвращение осуществляется не только с помощью технологий, но и с помощью политик, процедур и осведомленности. Хотя предотвратить все атаки нереально, сочетание защитных мер может значительно затруднить успех злоумышленника.

Обнаружение – Это способ выявления нарушения безопасности или вторжения. Без адекватных механизмов обнаружения существует риск не знать, была ли сеть взломана. Средства контроля обнаружения помогают выявлять инциденты безопасности и обеспечивают видимость действий в сети. Раннее обнаружение инцидента позволяет принять соответствующую реакцию для быстрого восстановления услуг.

Реакция – Время реакции значительно сокращается при наличии хороших средств защиты и механизмов обнаружения. Хотя нарушение безопасности - это плохая новость, важно знать, если

оно произошло. Быстрое время реакции имеет решающее значение для минимизации последствий инцидента. Быстрая реакция требует хорошей превентивной защиты и механизмов обнаружения угроз безопасности для предоставления данных и контекста, необходимых для реагирования. Скорость и эффективность реагирования на инциденты является ключевым показателем эффективности усилий организации по обеспечению безопасности.

Восстановление – Восстановление начинается с определения того, что произошло, чтобы системы можно было восстановить без воссоздания той же уязвимости или условия, которые в первую очередь вызвали инцидент. Фаза восстановления не заканчивается восстановлением системы. Существует также анализ первопричин, который определяет, какие изменения необходимо внести в процессы, процедуры и технологии, чтобы снизить вероятность возникновения уязвимостей такого же типа в будущем. Аудитор должен убедиться, что у организации есть план восстановления, включающий способы предотвращения подобных инцидентов в будущем.

1.3.1 Цель аудита безопасности

Ниже приведен список элементов, которые могут быть обнаружены в ходе аудита безопасности:

- Недостаточная физическая безопасность. Политика безопасности может требовать шифрования всех данных о клиентах, как при хранении, так и при передаче. Например, в ходе аудита выясняется, что раз в неделю всем менеджерам отправляется файл с информацией о клиентах в виде физического отчёта. Каждый экземпляр такого отчета в конце концов выбрасывается, но выясняется, что некоторые менеджеры небрежно выбрасывают физические отчёты в мусорное ведро, где их может найти любой, кто захочет порыться в мусоре.
- Неадекватное обслуживание паролей. Политика безопасности может требовать, чтобы каждый пользователь менял свой пароль каждые 30 дней. Аудит безопасности показывает, что пароли меняются, но многие пользователи просто чередуют «ПарольА» и «ПарольБ» каждый месяц. (История паролей - обычная функция в инструментах аудита паролей).
- Неадекватный контроль прав пользователей и разделения привилегий. Примером негативного результата, являющегося следствием неадекватного контроля, может быть ситуация, когда пользователям было предоставлено больше прав доступа к объектам, чем необходимо для выполнения их работы. Другим примером может быть совместное использование файлов отдельного пользователя в сети, когда они должны быть частными. Это особенно актуально для пользователей с ноутбуками и особенно для тех, кто выходит в интернет через Wi-Fi дома или в общественных местах.
- Неадекватная безопасность на уровне сервера. Конкретные области аудита включают:
  - Распределение портов и безопасность
  - Защиту данных
  - Защиту учетных записей пользователей (логины и другую конфиденциальную информацию)
- Неадекватное применение обновлений безопасности поставщика (вендора)
- Неадекватные механизмы обнаружения вторжений
- Неадекватные планы реагирования в случае нарушения безопасности

1.3.2 Выявление, оценка и снижение рисков

После того, как в ходе аудита были выявлены проблемные области, необходимо оценить риск и разработать план улучшения безопасности таких областей. Отчёт аудитора может включать рекомендации, а также другие области риска. С этого момента можно планировать мероприятия по выявлению, оценке и снижению риска.

Идентификация риска - это процесс документирования риска или области риска. В контексте ИТ-безопасности риски связаны с безопасностью. Оценка рисков - это деятельность, которая присваивает значение идентифицированным рискам. Важно понимать, что традиционные модели оценки рисков ИТ не являются достаточными для рассмотрения рисков безопасности ИТ. Любая модель или подход к оценке рисков безопасности должны быть специально ориентированы на профили рисков ИТ-безопасности.

Атрибутами риска являются потенциальное воздействие и вероятность возникновения риска. Величина риска рассчитывается путём умножения потенциального воздействия или убытков на вероятность их возникновения. Например, если информация о счёте одного клиента будет скомпрометирована, каковы будут последствия? А если у этого клиента на депозите 100 миллионов долларов?

Вероятность возникновения может быть определена путем применения модели оценки рисков безопасности, например, в публикации NIST 800-30 «Руководство по проведению оценки рисков» [NIST 800-30]. Еще одним отличным руководством по проведению оценки рисков безопасности является методология оценки рисков OWASP [OWASP2]. Следующая информация взята из [NIST 800-30].

Модели риска определяют факторы риска, подлежащие оценке, и взаимосвязи между этими факторами. Факторы риска - это характеристики, используемые в моделях риска в качестве исходных данных для определения уровней риска при оценке риска. Факторы риска также широко используются в сообщениях о рисках, чтобы подчеркнуть, что именно сильно влияет на уровни риска в конкретных ситуациях, обстоятельствах или контекстах.

Типичные факторы риска включают угрозу, уязвимость, воздействие, вероятность и предрасполагающее условие. Факторы риска могут быть разложены на более подробные характеристики (например, угрозы разложены на источники угроз и события угроз). Эти определения важны для организаций для документирования перед проведением оценки рисков, поскольку оценки полагаются на чётко определённые атрибуты угроз, уязвимостей, воздействия и других факторов риска для эффективного определения риска.

Угрозы

Угроза - это любое обстоятельство или событие, способное оказать негативное воздействие на деятельность и активы организации, отдельных лиц, других организаций или страны через информационную систему посредством несанкционированного доступа, уничтожения, раскрытия или модификации информации и/или отказа в обслуживании.

Угрожающие события вызываются источниками угроз. Источник угрозы характеризуется как:

- намерение и метод, направленные на использование уязвимости; или
- ситуация и метод, которые могут случайно использовать уязвимость.

В целом, типы источников угроз включают:

- Враждебные кибер- или физические атаки
- Человеческие ошибки действия или бездействия
- Структурные сбои контролируемых организацией ресурсов (например, оборудования, программного обеспечения, средств контроля окружающей среды)
- Природные и техногенные катастрофы, аварии и сбои, не находящиеся под контролем организации.

Были разработаны различные классификации источников угроз. В некоторых классификациях источников угроз в качестве принципа организации используется тип неблагоприятного

воздействия. Несколько источников угроз могут инициировать или вызвать одно и то же событие - например, сервер обеспечения может быть выведен из строя в результате атаки типа «отказ в обслуживании», преднамеренного действия злонамеренного системного администратора, административной ошибки, неисправности оборудования или сбоя питания.

Уязвимости и предрасполагающие условия
Уязвимость - это слабое место в информационной системе, процедурах системной безопасности, внутреннем контроле или реализации, которое может быть использовано источником угрозы.

Большинство уязвимостей информационной системы могут быть связаны с элементами контроля безопасности, которые либо не были применены (намеренно или непреднамеренно), либо были применены, но сохранили некоторые недостатки. Однако важно также учитывать возможность появления новых уязвимостей, которые могут возникнуть естественным образом с течением времени по мере развития миссий/бизнес-функций организации, изменения среды функционирования, распространения новых технологий и появления новых угроз. В контексте таких изменений существующие средства контроля безопасности могут стать неадекватными, и может потребоваться их переоценка на предмет эффективности. Тенденция к потенциальному снижению эффективности средств контроля безопасности с течением времени усиливает необходимость проведения оценки рисков на протяжении всего жизненного цикла программного обеспечения, а также важность программ непрерывного мониторинга для получения постоянной ситуационной осведомленности о состоянии безопасности организации.

Уязвимости выявляются не только в рамках информационных систем. Если рассматривать информационные системы в более широком контексте, то уязвимости можно обнаружить в организационных структурах управления (например, отсутствие эффективных стратегий управления рисками и адекватного определения рисков, плохие внутриведомственные связи, непоследовательные решения об относительных приоритетах миссий/деловых функций или несоответствие архитектуры предприятия поддержке миссий/бизнес деятельности).

Уязвимости также могут быть обнаружены во внешних отношениях (например, зависимость от конкретных источников энергии, цепочек поставок, информационных технологий и телекоммуникационных провайдеров), миссиях/бизнес-процессах (например, плохо определенные процессы или процессы, не учитывающие риски) и архитектурах безопасности предприятия/информационной безопасности (например, плохие архитектурные решения, приводящие к отсутствию разнообразия или устойчивости в информационных системах организации).

Влияние
Уровень воздействия события угрозы - это величина ущерба, который можно ожидать в результате последствий несанкционированного раскрытия информации, несанкционированной модификации информации, несанкционированного уничтожения информации или потери доступности информации или информационной системы. Такой ущерб может быть нанесен различным организационным и не организационным заинтересованным сторонам, включая:

- Руководителей агентств
- Владельцев миссий и бизнеса
- Владельцев информации/руководителей
- Владельцев миссий/бизнес-процессов
- Владельцев информационных систем
- Отдельных лиц/групп в государственном или частном секторе, полагающихся на организацию - по сути, всех, кто заинтересован в деятельности организации, ее активах или лицах, включая другие организации, сотрудничающие с организацией, или страной

Следующая информация должна быть явно документирована организацией:

- Процесс, используемый для проведения определения воздействия
- Допущения, связанные с определением воздействия
- Источники и методы получения информации о воздействии
- Обоснование выводов, сделанных в отношении определения воздействия

Организации могут чётко определить, как установленные приоритеты и ценности определяют идентификацию активов высокой ценности и потенциальное негативное воздействие на заинтересованные стороны организации. Если такая информация не определена, приоритеты и ценности, связанные с определением целей источников угроз и соответствующих организационных воздействий, обычно могут быть получены из стратегического планирования и политики. Например, уровни категоризации безопасности указывают на организационные последствия компрометации различных типов информации.

Вероятность

Вероятность наступления рассматривает вероятность (или возможность) того, что событие угрозы приведет к негативному воздействию, независимо от величины ожидаемого ущерба. Это взвешенный коэффициент риска, основанный на анализе вероятности того, что данная угроза способна использовать данную уязвимость (или набор уязвимостей). Коэффициент риска вероятности объединяет оценку вероятности того, что угрожающее событие будет инициировано, с оценкой вероятности воздействия (т.е. вероятности того, что угрожающее событие приведет к неблагоприятным последствиям).

Для враждебных угроз оценка вероятности возникновения обычно основывается на:

- Намерениях противника
- Возможностях противника
- Целях противника

Для других, не состязательных, угрожающих событий вероятность наступления оценивается на основе исторических фактов и данных, эмпирических данных или других факторов. Обратите внимание, что вероятность того, что угрожающее событие будет инициировано или произойдет, оценивается в отношении конкретного временного интервала (например, ближайшие шесть месяцев, ближайший год или период до достижения определённого рубежа).

Если событие угрозы почти наверняка будет инициировано или произойдет в (заданном или подразумеваемом) временном интервале, то при оценке риска может учитываться предполагаемая частота этого события. Вероятность возникновения угрозы также может быть основана на состоянии организации, включая, например, её основную миссию/бизнес-процессы, архитектуру предприятия, архитектуру информационной безопасности, информационные системы и среду, в которой эти системы работают. Также следует учитывать предрасполагающие условия, наличие и эффективность развернутых средств контроля безопасности для защиты от несанкционированного/нежелательного поведения, обнаружения и ограничения ущерба и/или поддержания или восстановления возможностей миссии/бизнеса.

Определение уровня риска безопасности

Оценка вероятности возникновения и оценка воздействия могут быть объединены для расчета общей степени тяжести риска. Конкретные баллы оценки могут быть использованы в качестве основы для заполнения матрицы рисков. В других случаях могут использоваться нечисловые оценки (низкая, средняя или высокая).

Оценка для матрицы рисков может быть основана на шкале от 0 до 9, где числовые значения определяются конкретными критериями. Например, критерии вероятности риска для конфиденциальности данных могут быть оценены как:

0 - <3 (Низкий) Частные данные не хранятся на локальных устройствах и шифруются при хранении на защищенных носителях.
3 - <6 (Средний) Частные данные могут находиться на таких устройствах, как ноутбуки, но они зашифрованы.
6 - 9 (Высокий) Точно неизвестно, находятся ли частные данные на локальных устройствах. Шифрование не может быть гарантировано.

Аналогичным образом, критерии воздействия риска могут быть оценены по той же шкале 0 - 9 на основе конкретных критериев. Например:

0 - <3 (Низкий) Компрометация частных данных затронет менее 200 человек.
3 - <6 (Средний) Компрометация частных данных затронет от 200 до 1 000 человек.
6 - 9 (Высокий) Компрометация частных данных затронет более 1 000 человек.

Как бы тестировщик ни пришёл к оценкам вероятности и воздействия, эти оценки могут быть объединены в окончательный рейтинг серьёзности для элемента риска. Если есть хорошая информация о влиянии на бизнес, ее следует использовать вместо информации о техническом воздействии. Если информации о бизнесе нет, то вместо нее следует использовать информацию о техническом воздействии.

Ниже приведен примерный вид матрицы рисков, которую можно использовать для определения серьезности отдельных рисков.

Общая степень риска				
Влияние риска	Высокая	Средняя	Высокая	Критическая
	Средняя	Средняя	Средняя	Высокая
	Низкая	Низкая	Низкая	Средняя
		Низкая	Средняя	Высокая
		Вероятность риска		

В приведенной выше примерном виде матрицы рисков, если вероятность средняя, а воздействие высокое, то общая степень серьезности высокая.

Кроме того, в отчёте об оценке риска необходимо определить, является ли риск постоянным. Постоянные риски указывают на повышенную вероятность возникновения убытков.

Серьезность риска определяет относительную важность снижения риска. Чем выше серьезность риска, тем более незамедлительным должно быть реагирование. Уровень детализации, представленный в любой конкретной оценке риска, соответствует цели оценки риска и типу исходных данных, необходимых для поддержки последующих определений вероятности и воздействия.

1.3.3 Люди, процессы и технологии

В ИТ-практике организации также есть три компонента: люди, процессы и технологии. Все они оказывают влияние на безопасность. Как пишет Крис Джексон в книге «*Аудит сетевой безопасности*» [Jackson, 2010], «все инциденты безопасности, от взлома до потери записей клиентов, обычно можно отследить по недостаткам, которые можно отнести к людям, процессам или технологиям».

Люди: Люди могут включать конечных пользователей, системных администраторов, владельцев данных и руководителей организации. Каждый человек имеет различные уровни навыков, отношения и планы, что влияет на то, как безопасность влияет на них, и как они влияют на эффективность средств контроля безопасности. Независимо от наличия политики безопасности, процедур и средств контроля, они будут неэффективны, если люди не будут следовать им. Если люди не следуют политикам безопасности, может возникнуть необходимость в корректирующих мероприятиях, например, в проведении тренингов по повышению осведомленности о безопасности или наказаниях за несоблюдение. Организационные структуры и политики безопасности часто определяются людьми, как внутренними, так и внешними по отношению к организации.

Процесс: Процессы определяют, как предоставляются ИТ-услуги, включая услуги, связанные с безопасностью. В контексте безопасности процессы включают процедуры и стандарты, которые вводятся в действие для защиты ценных активов. Чтобы быть эффективными, процессы должны быть определенными, актуальными, последовательными и соответствовать лучшим практикам безопасности. Процессы определяют роли и обязанности, средства контроля, инструменты и конкретные шаги, связанные с выполнением задачи.

Технология: Технология охватывает средства, оборудование, компьютерную технику и программное обеспечение, которые автоматизируют или поддерживают бизнес. Технологии позволяют людям выполнять повторяющуюся работу быстрее и с меньшим количеством ошибок, чем если бы она выполнялась вручную. Фактически, некоторые задачи, такие как введение паролей, были бы невозможны без соответствующих инструментов. Риск заключается в том, что неправильное использование технологии может помочь людям быстрее совершать ошибки.

Эти три области можно представить как стороны треугольника, известного из управления проектами , которые вместе образуют полное ИТ-решение. Если игнорировать любую из этих трех областей, пострадает вся работа по обеспечению ИТ и безопасности.

При оценке средств контроля безопасности аудитор должен посмотреть на систему с точки зрения злоумышленника и предугадать, как люди, процессы или технологии могут быть использованы для получения несанкционированного доступа к ценным активам. Руководство организаций часто удивляется тому, что механизмы безопасности, которые они считали надежными, таковыми не являются. Единственный способ узнать наверняка, работает ли та или иная защита безопасности и эффективна ли она, - это протестировать систему с точки зрения злоумышленника. Это часто называют этическим взломом или тестированием на проникновение.

Именно здесь связь между аудитом и тестированием становится наиболее прямой. Аудит выявляет недостатки и важные области для тестирования. Тестирование безопасности - это средство, с помощью которого можно доказать или опровергнуть, что средства контроля безопасности действительно существуют и эффективно работают.

Пример сценария:

Налоговое агентство одной из стран является объектом аудита безопасности. Один из выводов аудита заключается в том, что преступники могут подать поддельную налоговую декларацию и получить возврат налогов, причитающихся обманутому налогоплательщику. Этот вывод аудита подтверждается тестированием системы безопасности, и риск оценивается как «критический». Налоговое агентство признает возможность такого мошеннического риска, но решает не предпринимать никаких действий в связи с этим риском до следующего года.

Обманутые налогоплательщики, соблюдающие все предписанные процедуры безопасности, могут подать иск к налоговому агентству, которое знало о дефекте в процессе подачи налоговой декларации. В этом случае налоговое агентство будет нести ответственность за мошенничество.

2. Цели, задачи и стратегии тестирования безопасности - 130 мин.

Ключевые слова
межсайтовый сценарий, запутывание данных, отказ в обслуживании, обеспечение информации, политика безопасности, тестирование безопасности, уязвимость безопасности, жизненный цикл программного обеспечения, стратегия тестирования

Цели обучения для целей, задач и стратегий тестирования безопасности

2.1 Введение
В этом разделе нет целей обучения.

2.2 Назначение тестирования безопасности
AS-2.2.1 (K2) Понять, для чего в организации необходимо тестирование безопасности, включая преимущества для организации, такие как снижение рисков и повышение уровня доверия и уверенности

2.3 Организационный контекст
AS-2.3.1 (K2) Понять, как реалии проекта, бизнес-ограничения, жизненный цикл разработки программного обеспечения и другие факторы влияют на миссию команды тестирования безопасности

2.4 Цели тестирования безопасности
AS-2.4.1 (K2) Объясните, почему цели и задачи тестирования безопасности должны соответствовать политике безопасности организации и другим целям тестирования в организации
AS-2.4.2 (K3) Для конкретного проекта продемонстрировать способность определять цели тестирования безопасности на основе функциональных возможностей, технологических атрибутов и известных уязвимостей
AS-2.4.3 (K2) Понять взаимосвязь между обеспечением безопасности информации и тестированием безопасности

2.5 Объем и покрытие целей тестирования безопасности
AS-2.5.1 (K3) Для конкретного проекта продемонстрировать способность определить взаимосвязь между целями тестирования безопасности и необходимостью обеспечения целостности конфиденциальных цифровых и физических активов

2.6 Подходы к тестированию безопасности
AS-2.6.1 (K4) Проанализировать конкретную ситуацию и определить, какие подходы к тестированию безопасности наиболее вероятны для достижения успеха
AS-2.6.2 (K4) Проанализировать ситуацию, в которой данный подход к тестированию безопасности дал сбой, и определите вероятные причины сбоя
AS-2.6.3 (K3) Для заданного сценария продемонстрировать способность идентифицировать различные заинтересованные стороны и проиллюстрировать преимущества тестирования безопасности для каждой группы заинтересованных сторон

2.7 Совершенствование практик тестирования безопасности

AS-2.7.1 (K4) Проанализировать KPI (ключевые показатели эффективности), чтобы определить практики тестирования безопасности, которые нуждаются в улучшении, и элементы, которые не нуждаются в улучшении

2.1 Введение

Прежде чем применять специализированные методы тестирования безопасности, важно понять более широкий контекст тестирования безопасности и его роль в конкретной организации. Это понимание отвечает на следующие вопросы:

- Зачем нужно тестирование безопасности?
- Какова цель тестирования безопасности?
- Как тестирование безопасности вписывается в организацию?

Тестирование безопасности отличается от других форм функционального тестирования в двух важных областях [ISTQB_ATTA_SYL]:

1. Общепринятые методы выбора тестовых входных данных могут не учитывать важные проблемы безопасности
2. Симптомы дефектов безопасности сильно отличаются от тех, которые обнаруживаются при других типах функционального тестирования

Тестирование безопасности оценивает уязвимость системы к угрозам путем попытки компрометации системы. Ниже приведен список потенциальных угроз, которые должны быть изучены в ходе тестирования безопасности [ISTQB_ATTA_SYL]:

- Несанкционированное копирование приложений или данных
- Несанкционированный доступ (например, возможность выполнения задач, на которые у пользователя нет прав). Права пользователей, доступ и привилегии находятся в центре внимания этого тестирования. Данная информация должна быть доступна в спецификациях системы.
- Программное обеспечение, проявляющее непредвиденные побочные эффекты при выполнении своей предназначенной функции. Например, медиаплеер, который корректно воспроизводит звук, но при этом записывает файлы в незашифрованное временное хранилище, демонстрирует побочный эффект, которым могут воспользоваться пираты программного обеспечения.
- Код, вставленный в веб-страницу, который может быть использован последующими пользователями (межсайтовый скриптинг или XSS) и может быть вредоносным.
- Переполнение буфера, которое может быть вызвано вводом данных в поле ввода пользовательского интерфейса, которые допускает количество символов больше, чем может правильно обработать код. Уязвимость переполнения буфера предоставляет возможность выполнения инструкций вредоносного кода.
- Отказ в обслуживании, который не позволяет пользователям взаимодействовать с приложением (например, путем перегрузки веб-сервера «мешающими» запросами)
- Перехват, имитация и/или изменение и последующая ретрансляция сообщений (например, транзакций по кредитным картам) третьей стороной таким образом, чтобы пользователь оставался в неведении о присутствии этой третьей стороны (атака “Человек посередине”)
- Взлом кодов шифрования, используемых для защиты конфиденциальных данных
- Логические «бомбы» (иногда называемые пасхальными яйцами), которые могут быть злонамеренно вставлены в код и которые активируются только при определенных условиях (например, в конкретную дату). Логические бомбы могут выполнять вредоносные действия, такие как удаление файлов или форматирование дисков.

Тестирование безопасности должно быть интегрировано со всеми другими мероприятиями по разработке и тестированию. Это требует учета уникальных потребностей организации, любых

существующих политик безопасности, текущих наборов навыков тестирования безопасности и любых существующих стратегий тестирования.

2.2 Назначение тестирования безопасности

Как и тестирование программного обеспечения в целом, тестирование безопасности не может гарантировать, что система или организация будет защищена от атак. Однако тестирование безопасности может помочь выявить риски и оценить эффективность существующих средств защиты. Существуют и другие мероприятия, дополняющие тестирование безопасности, такие как аудиты и рецензирование методов обеспечения безопасности.

Тестирование безопасности также показывает, что при защите цифровых активов была проведена надлежащая проверка. В случае нарушения безопасности может последовать судебное разбирательство. Если компания может доказать, что она предприняла разумные шаги для защиты цифровых активов, такие как тестирование на наличие уязвимостей, это может быть защитой в суде. Тестирование безопасности также может быть гарантией для клиентов и заказчиков того, что организация предпринимает надлежащие шаги для защиты конфиденциальной информации.

2.3 Организационный контекст

Безопасность часто является одним из видов функционального тестирования, выполняемого наряду с другими типами тестирования. Имея в распоряжении ограниченное время для тестирования, руководитель тестирования должен решить, какой объем тестирования может быть выполнен, включая тестирование безопасности. Нередко тестирование безопасности рассматривается как активность, для которой необходима роль специалиста и поэтому передается на аутсорсинг организации, специализирующейся на тестировании безопасности. Степень тестирования безопасности в конечном итоге, определяется бизнесом или организационными рисками, основанными на безопасности. Когда в организации существует множество угроз безопасности, требуется более тщательное тестирование безопасности.

Как и тестирование программного обеспечения, обеспечение информационной безопасности - это деятельность на протяжении всего жизненного цикла. Потребности в безопасности должны быть определены в требованиях, выражены в дизайне и реализованы в коде. Затем тестирование безопасности может проверить и подтвердить правильность и эффективность реализации безопасности. Безопасность не может быть эффективно исправлена только в коде или протестирована в нем. Программное обеспечение может быть защищено только тогда, когда безопасность встроена в программное обеспечение с использованием безопасных методов программирования и проектирования.

Реалии ограниченного времени, ресурсов и объема, наряду с уровнями риска, наборами навыков тестирования безопасности и подходами к жизненному циклу, сильно влияют на успех группы тестирования безопасности в организации.

2.4 Цели тестирования безопасности

2.4.1 Согласование целей тестирования безопасности

Политика тестирования безопасности может быть разработана после утверждения политики безопасности организации высшим руководством. Важно, чтобы цели и задачи тестирования безопасности, изложенные в политике тестирования безопасности, соответствовали общей политике безопасности организации. В противном случае либо будут проведены несанкционированные тесты безопасности, либо тесты безопасности могут не достичь желаемых целей.

2.4.2 Определение целей тестирования безопасности

Цели тестирования безопасности можно рассматривать в том же свете, что и цели функционального тестирования, но они сосредоточены на целях безопасности. Для каждой функции безопасности системы или приложения должна быть одна или несколько целей тестирования безопасности.

Цели тестирования безопасности также должны основываться на атрибутах технологии (например, сеть, мобильная связь, облако, локальная сеть) и известных уязвимостях как в приложении, так и на общих уязвимостях. Например, целями тестирования безопасности могут быть следующие:

- Убедиться, что при проверке аутентификации с помощью пароля применяется правильное правило проверки надежности пароля
- Убедиться, что все поля ввода данных проверены на возможность атаки с использованием SQL-инъекций
- Убедиться, что файлы данных клиентов зашифрованы с правильной надежностью

2.4.3 Разница между обеспечением информационной безопасности и тестированием безопасности

Обеспечение информации (IA) определяется как «Меры по защите и охране информации и информационных систем обеспечивая их доступность, целостность, аутентификацию, конфиденциальность и безотказность. Эти меры включают в себя обеспечение восстановления информационных систем путем включения возможностей защиты, обнаружения и реагирования.» [NISTIR 7298]

Тестирование безопасности - это «Процесс, используемый для определения того, что функции безопасности системы реализованы так, как задумано, и соответствуют предлагаемой среде приложения». [MDA1]

При сравнении терминов «обеспечение информации» и «тестирование безопасности» IA является более широким, всеобъемлющим термином. Эта взаимосвязь аналогична взаимосвязи между обеспечением качества (QA) и тестированием программного обеспечения.

2.5 Объем и охват целей тестирования безопасности

Чем больше потребность в целостности конфиденциальных цифровых и физических активов, тем больше потребность в достижении целей тестирования безопасности. Цели тестирования безопасности, по сути, описывают объем тестирования безопасности. Если объем слишком мал, уверенность в адекватности безопасности не будет достигнута. Если объем слишком велик, ресурсы могут быть исчерпаны до завершения теста.

Цели тестирования безопасности должны описывать ожидаемые результаты тестирования безопасности в отношении проверки и подтверждения имеющихся средств защиты для конфиденциальных цифровых и физических активов. Цели тестирования безопасности должны быть непосредственно связаны с конкретными активами, защитными мерами, рисками и выявлением уязвимостей в системе безопасности.

2.6 Подходы к тестированию безопасности

Стратегия тестирования безопасности определяется для формализации и передачи общего направления организации в области тестирования безопасности. Затем определяются подходы, реализующие стратегию тестирования безопасности.

2.6.1 Анализ подходов к тестированию безопасности

У каждой организации есть уникальные бизнес-задачи и миссия, которые, в свою очередь, требуют уникальных стратегий и подходов к тестированию безопасности для выявления и снижения рисков безопасности. Однако существуют также некоторые проблемы безопасности, которые являются общими для многих организаций.

Подход к тестированию безопасности определяется на уровне проекта и должен соответствовать политике и стратегии тестирования организации. Подход к тестированию безопасности проекта будет представлять собой специфическое для конкретного проекта сочетание методов, инструментов и навыков для решения задач тестирования безопасности этого проекта.

При анализе ситуации с целью определения подхода к тестированию безопасности учитывайте следующее:

- Источник поставки систем или приложений
- Результаты любого предыдущего тестирования безопасности
- Политика безопасности
- Политика тестирования безопасности
- Любые оценки рисков безопасности, уже выполненные в организации
- Используемая техническая среда (например, тип и версия программного обеспечения, фреймворки, языки программирования, операционные системы)
- Навыки тестирования безопасности в команде тестирования
- Общеизвестные риски безопасности
- Организацию стратегии тестирования безопасности
- Структура команды проекта
- Опыт работы команды тестирования с различными инструментами тестирования безопасности
- Ограничения (например, ограниченные ресурсы, ограниченное время, отсутствие доступа к средам)
- Допущения (например, допущения относительно других ранее выполненных форм тестирования безопасности)

Различные технические среды и типы приложений (например, клиент/сервер, веб, высокопроизводительные отказоустойчивые сервера, или мейнфреймы) часто требуют различных подходов и стратегий тестирования безопасности. Например, при разработке программного обеспечения может потребоваться проверка кода для обнаружения уязвимостей, в то время как тестирование программного обеспечения может потребовать запутывания тестовых

данных. Веб-приложения имеют другие уязвимости, чем мейнфреймы, и поэтому требуют других видов тестов безопасности.

Некоторые уязвимости являются общими для нескольких технологий. Например, уязвимости переполнения буфера могут возникать в клиент-серверных, веб- и мобильных приложениях с различиями в зависимости от того, как в каждой технологии обрабатывается управление памятью. Результат одинаков во всех средах - непредсказуемое поведение программного обеспечения, которое может позволить злоумышленнику получить доступ к приложению и выполнять задачи, которые обычно запрещены.

Недостаточная защита данных может иметь место в любой технологии или среде. Однако шифрование данных в веб- и мобильных средах отличается от шифрования в среде мейнфреймов. Алгоритмы шифрования могут быть одинаковыми (или похожими), но разница в том, что в случае веб- и мобильных приложений данные должны быть защищены при передаче через Интернет. Во всех технологиях конфиденциальные данные должны храниться в зашифрованном формате. Имели место инциденты, когда конфиденциальные данные высокопроизводительного отказоустойчивого сервера физически отправлялись (с использованием ленты) другой стороне в незашифрованном формате. «Компания Cattles Group, специализирующаяся на кредитовании физических лиц и взыскании долгов, признала потерю двух резервных лент, содержащих информацию примерно о 1,4 миллионах клиентов.» [Computer Weekly]

2.6.2 Анализ отказов в подходах к тестированию безопасности

Необходимо понимать, что существуют различные степени отказа. Тот факт, что уязвимость безопасности не обнаружена и не устранена, не обязательно означает, что попытка тестирования безопасности неудачна. Существует слишком много возможных уязвимостей безопасности, и ежедневно обнаруживаются новые. Однако есть и другие случаи, когда подходы к тестированию безопасности оказались недостаточными для эффективного выявления рисков безопасности, что привело к компрометации конфиденциальных данных и других цифровых активов.

Анализ первопричин может помочь определить, почему подход к тестированию безопасности мог оказаться неудачным. Возможные причины включают:

- Отсутствие руководящей роли в организации тестирования безопасности
- Отсутствие у руководства ресурсов, необходимых для реализации стратегии тестирования безопасности (например, нехватка финансирования, времени, других ресурсов)
- Отсутствие эффективной реализации подхода к тестированию безопасности (например, отсутствие навыков, необходимых для выполнения требуемых задач)
- Отсутствие организационного понимания и поддержки подхода к тестированию безопасности
- Отсутствие понимания и поддержки подхода к тестированию безопасности со стороны заинтересованных сторон
- Отсутствие понимания рисков безопасности
- Отсутствие согласованности между подходом к тестированию и политикой и стратегией тестирования безопасности организации
- Отсутствие понимания цели системы
- Отсутствие технической информации о системе (что приводит к неверным предположениям)
- Отсутствие эффективных инструментов тестирования безопасности
- Отсутствие навыков тестирования безопасности

2.6.3 Идентификация заинтересованных сторон

Чтобы усилия по тестированию безопасности были эффективными, необходимо предоставить руководству бизнес-обоснование. В нем должны быть четко определены риски, связанные с нарушением безопасности, и преимущества эффективного подхода к тестированию безопасности для конкретного проекта.

Разные заинтересованные стороны будут видеть разные преимущества подхода к тестированию безопасности:

- Высшее руководство может увидеть защиту бизнеса как преимущество
- Руководство среднего звена может увидеть должную осмотрительность
- Бизнес-клиенты могут увидеть защиту от мошенничества
- Сотрудники по соблюдению нормативно-правового соответствия (для внутренних корпоративных политик безопасности) могут увидеть уверенность в том, что организация соответствует юридическим обязательствам
- Сотрудники регулирующих органов (для внешних законов о безопасности) могут увидеть преимущество в том, что правила безопасности соблюдаются
- Специалисты по конфиденциальности могут увидеть преимущество в том, что личные данные хранятся в безопасности, а при защите цифровых активов была проявлена должная осмотрительность.

2.7 Улучшение практик тестирования безопасности

Чтобы улучшить практики тестирования безопасности, сначала необходимо оценить существующие практики. Для этого должен существовать объективный способ оценки практик тестирования безопасности. Они основаны на ключевых показателях целей тестирования безопасности, по которым можно определить степень успеха ключевых элементов стратегии.

Эти практики должны быть оценены следующим образом:

- В краткосрочной и долгосрочной перспективе
- Учитывая процесс и организацию
- Учитывая людей, инструменты, системы и методы

Ключевые показатели включают, но не ограничиваются:

- Уровни покрытия тестами рисков безопасности
- Уровни покрытия тестами политик и практик безопасности
- Уровни покрытия тестами требований безопасности
- Уровни эффективности уже затраченных усилий по тестированию безопасности, основанные на том, когда и где были выявлены уязвимости в системе безопасности. Это включает в себя уязвимости безопасности как до, так и после выпуска.

3. Процессы тестирования безопасности - 140 мин.

Ключевые слова

хищение учетных записей, взлом пароля, социальная инженерия, подход к тестированию, план тестирования, процесс тестирования

Цели обучения для процессов тестирования безопасности

3.1 Определение процесса тестирования безопасности

AS-3.1.1 (K3) Для заданного проекта продемонстрировать способность определять элементы эффективного процесса тестирования безопасности

3.2 Планирование тестирования безопасности

AS-3.2.1 (K4) Провести анализ заданного плана тестирования безопасности, предоставив отзывы о сильных и слабых сторонах плана

3.3 Проектирование тестов безопасности

AS-3.3.1 (K3) Для заданного проекта внедрить концептуальные (абстрактные) тесты безопасности, основанные на выбранном подходе к тестированию безопасности, наряду с выявленными функциональными и структурными рисками безопасности

AS-3.3.2 (K3) Реализовать тестовые сценарии для проверки политик и процедур безопасности

3.4 Выполнение тестов безопасности

AS-3.4.1 (K2) Понимать ключевые элементы и характеристики эффективной среды тестирования безопасности

AS-3.4.2 (K2) Понимать важность планирования и получения разрешений перед выполнением любого теста безопасности

3.5 Оценка тестирования безопасности

AS-3.5.1 (K4) Провести анализ результатов тестирования безопасности, чтобы определить следующее:

- Природу уязвимости системы безопасности
- Степень уязвимости системы безопасности
- Потенциальное влияние уязвимости системы безопасности
- Предлагаемое исправление
- Оптимальные методы отчетности о тестировании

3.6 Обслуживание тестов безопасности

AS-3.6.1 (K2) Понимать важность поддержания процессов тестирования безопасности с учетом меняющегося характера технологий и угроз

3.1 Определение процесса тестирования безопасности

Как и тестирование программного обеспечения в целом, тестирование безопасности является деятельностью жизненного цикла. Неспособность внедрить и протестировать средства безопасности на протяжении всего проекта может привести к серьезным дефектам безопасности, которые, возможно, никогда не будут полностью устранены. Процесс тестирования безопасности должен быть согласован с процессом разработки, чтобы при необходимости выполнялись соответствующие действия по тестированию.

Риски и потребности каждой организации, связанные с тестированием безопасности, будут уникальными из-за характера организации, технической среды, процесса разработки программного обеспечения и бизнес-рисков. Следовательно, процесс тестирования безопасности должен определяться в контексте этих факторов.

3.1.1 Процесс тестирования безопасности ISTQB

В Таблице 3.1 показана взаимосвязь между общим процессом тестирования ISTQB, описанным в программах Базового и Продвинутого уровней ISTQB, и процессом тестирования безопасности ISTQB. Примеры задач тестирования безопасности показаны для каждого шага процесса.

Таблица 3.1 – Процесс тестирования безопасности ISTQB

Процесс тестирования ISTQB	Процесс тестирования безопасности ISTQB	Пример задач тестирования безопасности
Планирование и контроль тестирования	Планирование и контроль тестирования безопасности, цель которого состоит в определении соответствующего объема тестирования, который соответствует рискам безопасности.	• Определить цели тестирования безопасности • Определить объем тестирования безопасности • Определить ресурсы тестирования безопасности • Определить оценки и графики тестирования безопасности • Определить метрики тестирования безопасности, критерии начала и завершения тестирования безопасности • Провести мониторинг хода и результатов тестирования безопасности • Принять необходимые меры в ответ на информацию, полученную в ходе других мероприятий по тестированию безопасности.
Анализ и проектирование тестов	Анализ и проектирование тестирования безопасности, цель которого состоит в том, чтобы получить представление о конкретных угрозах и рисках безопасности	• Провести обзор элементов, которые служат основой для тестирования безопасности, таких как оценки рисков безопасности, требования

Процесс тестирования ISTQB	Процесс тестирования безопасности ISTQB	Пример задач тестирования безопасности
	на основе оценок безопасности, аудитов и стандартных источников известных уязвимостей.	безопасности и политики безопасности • Определить условия тестирования безопасности на основе: • Целей тестирования • Рисков безопасности • Стандартов безопасности и известных уязвимостей • Средств защиты, применяемых для обеспечения безопасности системы и ее данных • Объема тестирования безопасности • Применимости инструментов тестирования безопасности
Реализация и выполнение тестов	Реализация и выполнение тестов безопасности, цель которого состоит в том, чтобы перевести концептуальные тесты в тесты, которые могут быть выполнены вручную или с помощью инструментов. Кроме того, цель состоит в том, чтобы выполнить эти тесты с использованием различных точек зрения тестирования безопасности – внутреннего пользователя, внешнего пользователя, злоумышленника и т.д.	• Создать тестовые сценарии безопасности, сценарии тестирования или другие тестовые спецификации • Выполнить функциональные тесты безопасности на основе конкретных спецификаций тестов безопасности • Выполнить функциональное тестирование безопасности и проникновения на основе знаний и интуиции тестировщика • Провести тестирование безопасности на основе модели системы • Настроить или подготовить тестовую среду для проведения тестирования безопасности
Оценка критериев завершения тестирования и отчетность	Оценка и отчетность результатов тестирования безопасности - часто выполняется параллельно с выполнением тестов, чтобы оценить отдельные тесты и	• Определить конкретные уязвимости безопасности на основе результатов тестирования • Оценить уровни риска безопасности на основе

Процесс тестирования ISTQB	Процесс тестирования безопасности ISTQB	Пример задач тестирования безопасности
	как можно скорее сообщить о новых угрозах.	результатов проведенных тестов безопасности • Сообщить промежуточные и окончательные результаты тестирования безопасности
Завершение тестирования	Завершение тестирования - цель состоит в том, чтобы довести деятельность по тестированию безопасности до завершения, чтобы тесты можно было сопровождать и проводить на регулярной основе для поддержки любых новых требований безопасности и/или обнаружения любых новых угроз. Кроме того, все оборудование для тестирования безопасности и результаты хранятся в защищенном виде, но при этом доступны для использования в случае необходимости в будущих тестах безопасности.	• Обеспечить проведение всех запланированных тестов безопасности • Определить, были ли предоставлены результаты тестирования безопасности (отчеты) • Архивировать результаты тестов, тестовые данные и другую конфиденциальную информацию в безопасных местах • Провести анализ результатов тестов безопасности для улучшения разработки систем и приложений с точки зрения безопасности

Важно понимать, что процесс тестирования безопасности ISTQB не обязательно носит последовательный характер. Процесс тестирования безопасности должен соответствовать процессу жизненного цикла разработки программного обеспечения организации. Основным следствием процесса, описанного в этом разделе, является то, что действия по тестированию безопасности выполняются параллельно и во время других действий и тестов жизненного цикла проекта.

Кроме того, задачи тестирования безопасности, показанные в таблице 3.1, приведены в качестве примеров, а не обязательных требований к задачам тестирования безопасности. Точные задачи тестирования безопасности для организации зависят от стратегии тестирования безопасности и подхода, принятых организацией, как показано на рисунке 3.1 ниже.

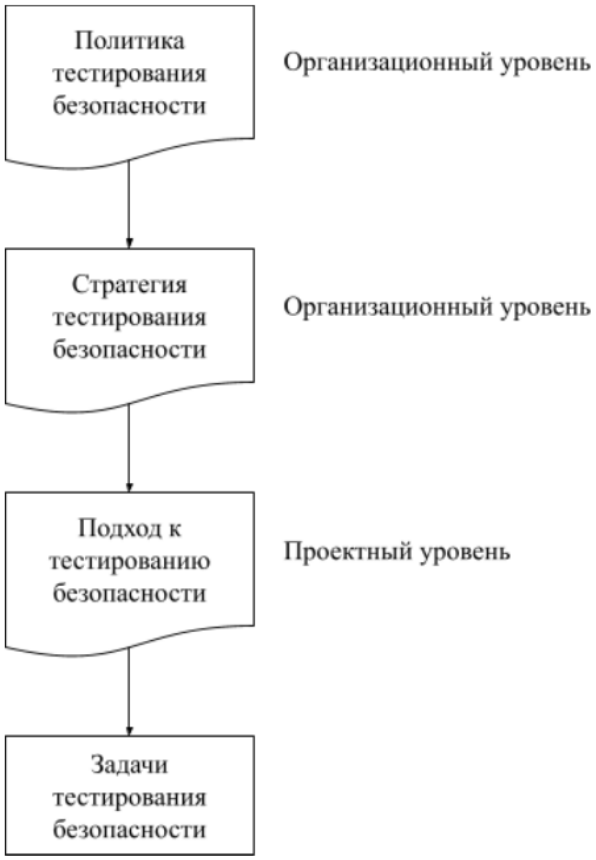

Рисунок 3.1 – Иерархия планирования тестирования безопасности

Приведение процесса тестирования безопасности в соответствие с конкретной моделью жизненного цикла приложения

Каждый из следующих типов процессов жизненного цикла связан с проблемами тестирования безопасности. Важно согласовать тестирование безопасности с жизненным циклом.

Последовательные жизненные циклы

В этих проектах тестировщик безопасности должен знать следующее:

- Потребности и риски безопасности определяются на ранних этапах проекта и должны быть задокументированы в спецификациях требований к программному обеспечению.
- Потребности в безопасности могут меняться в ходе проекта, но могут не отражаться в обновленных требованиях к программному обеспечению. Тесты безопасности могут показаться очень конкретными и полными, но могут быть неполными или неактуальными из-за рисков, связанных с поздних изменений в проекте.
- Тесты безопасности можно выполнять в любое время, но обычно эти тесты выполняются на поздних стадиях проекта.
- Может быть трудно рассмотреть результаты тестирования безопасности в конце последовательного жизненного цикла проекта.

Итеративные/инкрементные жизненные циклы
Инкрементные проекты предусматривают выпуск небольших и частых версий приложения. Примером такого подхода являются методы Agile. В таких проектах тестировщик безопасности должен знать следующее:

- Потребности и риски в области безопасности возникают на протяжении всего проекта (обычно в контексте итерации или спринта) и могут быть определены в спецификациях требований, пользовательских историях, моделях, критериях приемки и/или прототипах.
- Потребности и риски безопасности могут меняться в ходе проекта и могут (должны) быть рассмотрены в той итерации, в которой они были выявлены.
- Тесты безопасности могут проводиться непрерывно на протяжении всего проекта.
- В зависимости от характера риска безопасности, может оказаться невозможным полностью смягчить и протестировать его в течение одного короткого цикла выпуска.

Готовый коммерческий продукт (COTS)
Эти проекты часто носят характер черного ящика и могут позволять или не позволять настройку. Они часто содержат уязвимости безопасности и требуют частого обновления безопасности и исправления. Доступ к коду отсутствует, поэтому структурный анализ и структурное тестирование невозможны.

Программное обеспечение с открытым исходным кодом
Это вариант COTS, но с одним важным отличием — код доступен для просмотра всему миру. Эти продукты также имеют уязвимости в безопасности, поэтому крайне важно, чтобы исправления безопасности постоянно обновлялись. Как только уязвимость безопасности становится достоянием общественности, пользователи данной конкретной версии программного обеспечения (и более ранних) подвергаются риску атаки.

Пример - процесс тестирования безопасности в последовательном жизненном цикле
Важно отметить, что тестирование безопасности не обязательно должно ограничиваться одним этапом или действием в проекте. особенно важно избегать ситуаций, когда тестирование безопасности (и другие тесты) не выполняется до этапа приемки проекта. В конце проекта особенно дорого и рискованно устранять любые обнаруженные дефекты. Ниже показаны необходимые задачи тестирования безопасности, которые должны выполняться на каждом этапе последовательного жизненного цикла:

- **Требования** – Требования безопасности определяются и пересматриваются как часть общей работы по требованиям, чтобы выразить потребности организации. Здесь также могут быть написаны сценарии использования. Именно на этом этапе следует разработать подход к тестированию безопасности.

- **Анализ и проектирование** – Обычно кто-то в роли бизнес-аналитика изучает первоначальное изложение требований и уточняет их, чтобы заполнить пробелы. Затем системный аналитик и/или архитектор анализирует требования, чтобы предложить оптимальный способ предоставления решения для удовлетворения потребностей пользователей. В этом случае безопасность будет одной из функциональных и нефункциональных потребностей наряду с другими, такими как удобство использования и эффективность. На данном этапе разработчики тестов безопасности могут получить представление об архитектуре и о том, что необходимо протестировать с точки зрения как структурной, так и функциональной безопасности. Также должны быть определены основные цели тестирования безопасности.

- **Детальное проектирование** – На этом этапе проектируются пользовательские интерфейсы и базы данных. Функциональные правила уточняются, а проектирование

тестов безопасности становится более детальным. Первые тесты безопасности могут быть выполнены на основе моделей.

- **Кодирование/реализация** – Это происходит, когда проектные спецификации реализуются. Это первая возможность протестировать структуру приложения, включая тестирование на уязвимости безопасности, такие как дефекты переполнения буфера и правки на уровне полей, которые могут позволить осуществить SQL-инъекции. Статический анализ и обзор кода очень ценны на этом этапе и должны включать изучение кода с точки зрения безопасности. Тестирование компонентов также является ключевым действием для проверки правильности работы кода. Интеграционное тестирование между компонентами также может начинаться по мере того, как компоненты, взаимодействующие друг с другом, становятся доступными для тестирования в небольших сборках.

- **Системное тестирование** – Это тестирование систем и подсистем. Системный тест включает в себя программное обеспечение, оборудование, данные, процедуры и то, как люди взаимодействуют с системой. Эти тесты часто носят транзакционный характер для проверки бизнес-процессов. Основой для системного тестирования могут быть требования, модели проектирования, сценарии использования и любые другие спецификации, отражающие перспективу системы. Кроме того, может потребоваться системное интеграционное тестирование, чтобы проверить, как различные (под) системы взаимодействуют и обмениваются данными. Тестирование безопасности на этом этапе приобретает более широкий вид, поскольку в нем участвуют аппаратные средства и обмен данными. Можно протестировать безопасность транзакций, которая включает в себя аутентификацию, хранение данных, реализацию брандмауэра, а также процедурные меры безопасности.

- **Пользовательское приемочное тестирование** – На этом этапе тестирования проверяется, что система поддерживает реальные бизнес-процессы и может охватывать несколько систем в нескольких организациях. Целью этого этапа является не столько поиск дефектов, сколько подтверждение того, что система отвечает потребностям пользователей в реальных условиях. Это включает в себя обеспечение того, чтобы требования безопасности были реализованы и соблюдены должным образом. На этом этапе тестирование безопасности уже должно быть в основном выполнено, но все еще есть возможности для тестирования сценариев безопасности, которые возникают на уровне бизнес-процессов.

- **Развертывание** – Это происходит, когда завершенная и протестированная система развертывается для пользователей. Это может произойти многими способами, например, при пилотном развертывании для отдельных групп или массовом развертывании для всех пользователей. Другой подход заключается в параллельном развертывании, при котором старая система и новая система работают одновременно в течение некоторого времени. Большая часть решения о реализации прямого переключения зависит от риска развертывания для всех пользователей и уверенности, полученной во время приемочного тестирования. Безопасность является проблемой во время развертывания системы, поскольку все компоненты системы должны быть развернуты таким образом, чтобы не возникало новых уязвимостей. Это может произойти, если настройки безопасности в целевой среде неверны. Примером этого может быть, если права доступа к базе данных неверны в реальной среде.

- **Сопровождение** – По мере появления новых потребностей или обнаружения дефектов после развертывания выполняется техническое обслуживание. Тестирование приобретает другое измерение, поскольку основное внимание уделяется тестированию

изменений и выполнению регрессионного тестирования. Также следует проводить тестирование безопасности, чтобы убедиться, что новые уязвимости не появляются во время изменений. Частью процесса технического обслуживания является поддержание брандмауэров и других технологий безопасности в актуальном состоянии. Непрерывный мониторинг системы может обнаружить подозрительную активность, которую необходимо немедленно устранить.

Пример – процесс тестирования безопасности в итеративном/инкрементном жизненном цикле
Существует множество методологий, которые были реализованы за последние 20 лет для определения построения программного обеспечения с небольшими приращениями или итерациями. В этом примере выпуски программного обеспечения поставляются каждые четыре недели. Основой разработки (и тестирования) являются пользовательские истории, каждая из которых имеет определенные критерии приемки.

Выбор функций для создания и предоставления основывается на приоритетном списке задач. Выбранные функции должны отражать те элементы, которые обеспечивают наибольшую ценность и достижимы в рамках спринта. Тестировщик безопасности работает с бизнесом и/или владельцем продукта, чтобы получить надлежащие и корректные требования безопасности.

В этом примере для первой итерации выбраны четыре основные функции безопасности, поскольку они будут необходимы для разработки многих других функций. К этим функциям относятся:

- Вход пользователя в систему
- Включение SSL (Secure Socket Layer)
- Восстановление утерянного пароля
- Блокировка учетной записи после трех неудачных попыток ввода неправильного пароля

Каждая из этих функций написана в виде пользовательских историй и преобразована в более подробные требования, каждая из которых имеет критерии приемки.

Тестировщик безопасности нахоится в тесном контакте с разработчиком, чтобы:

- обеспечить отражение правильных политик и протоколов в коде в части обеспечения безопасности кода
- тестировать функции по мере их разработки.

В этом примере первый релиз может состоять только из страницы входа в систему и связанных с ней функций, таких как сброс утерянного пароля и управление блокировкой. В следующей итерации будут разработаны другие функции, исходя из приоритета заинтересованных сторон. В каждой итерации тестировщик безопасности будет проверять, правильно ли работают средства контроля безопасности и не появились ли новые уязвимости безопасности. Итерации будут продолжаться до тех пор, пока не будут выполнены все задачи бэклога.

В каждом примере (итеративном/инкрементальном и последовательном) этапы процесса тестирования безопасности можно рассматривать как неотъемлемые задачи обеспечения безопасности приложения.

3.2 Планирование тестирования безопасности

3.2.1 Цели планирования тестирования безопасности

Тестирование безопасности в целом должно быть сосредоточено на двух аспектах:

- Проверка того, что разработанные средства защиты безопасности реализованы и функционируют должным образом
- Проверка появления новых уязвимостей во время разработки приложения

Как упоминалось ранее в этой программе обучения, все меры безопасности, которые необходимо внедрить, должны основываться на анализе рисков. Это обеспечивает отправную точку при планировании тестирования безопасности для проекта.

Многих из непреднамеренно разработанных уязвимостей можно избежать, используя действия по обеспечению качества и лучшие практики во время работ по архитектуре, проектированию и кодированию. Проверка наличия уязвимостей начинается с оценки методов, используемых командой разработчиков. На основании полученных результатов может возникнуть необходимость в выборе и реализации дополнительных тестов безопасности.

3.2.2 Ключевые элементы плана тестирования безопасности

Ключевые элементы плана тестирования безопасности перечислены ниже. Каждый из них можно определить, задав определенные вопросы для данного проекта.

- Определение объема тестирования безопасности
  - Что входит и выходит за рамки объема проекта?
  - Что достижимо с учетом ресурсов проекта, рисков безопасности и временных ограничений?

- Определение того, кто должен проводить тестирование безопасности
  - Есть ли в организации сотрудники, обладающие соответствующими навыками тестирования безопасности?
  - Чувствует ли организация себя комфортно при передаче тестирования безопасности на аутсорсинг?
  - В случае коммерческого программного обеспечения и программного обеспечения, разработанного поставщиком, за какие тесты безопасности отвечает поставщик, а за какие — заказчик?
  - Нуждаются ли тестировщики безопасности в обучении использованию конкретных инструментов тестирования безопасности?

- Выделение соответствующего графика для тестирования безопасности с учетом других требований к планированию тестирования проекта
  - Какие элементы, связанные с безопасностью, должны быть реализованы и протестированы до проведения других испытаний? (например, права доступа, имена пользователей и пароли)
  - Когда функции безопасности будут доступны для тестирования?
  - Сколько времени потребуется для проведения тестирования безопасности с учетом запланированных ресурсов и объема?

- Определение задач, которые необходимо выполнить, и времени, необходимого для каждой из них
  - Сколько времени требуется для разработки соответствующих тестов безопасности на основе запланированных ресурсов и объема?
  - Сколько времени необходимо для оценки и отчета о результатах выполнения тестов безопасности?
  - Сколько времени требуется для выполнения регрессионных тестов, поскольку они связаны с безопасностью?

- Сколько времени требуется для создания окружения тестирования безопасности?

- Определение окружения тестирования безопасности
  - Какова степень окружения? (платформа, технология, размер, местоположение)
  - Это новое окружение?
  - Какие инструменты тестирования безопасности и другие инструменты тестирования необходимо установить в окружении?

- Получение разрешений и согласований для тестирования безопасности
  - Кто должен рецензировать и утверждать тесты безопасности?
  - Когда это разрешение необходимо?
  - Достаточно ли бюджета и финансирования?

Как и любой результат проекта, план тестирования безопасности должен быть прорецензирован для оценки полноты и правильности. Поскольку тесты безопасности часто носят технический характер, наиболее подходящим методом может быть сессия рецензирования. Однако также могут быть уместны разборы и инспекции.

Типовой контрольный список может помочь сформировать основу того, что будет рассмотрено на обзорной сессии. Как и любой другой обзор, обратная связь должна быть конструктивной и не направлена на производителя плана тестирования безопасности. В группу проверки должны входить квалифицированные специалисты из всех областей, на которые влияют аспекты безопасности, обсуждаемые в плане тестирования безопасности. Члены группы проверки не обязательно должны быть специалистами по тестированию безопасности или обладать опытом в области безопасности. Например, у руководителя бизнес-подразделения может быть информация о рисках безопасности, которая должна быть отражена в плане тестирования безопасности. ИТ-аудиторы и администраторы безопасности особенно полезны при анализе планов тестирования безопасности благодаря своему знанию политик и процедур безопасности.

3.3 Проектирование тестов безопасности

Существует несколько способов начать проектирование тестов безопасности. Например, его можно начать:
- На основании проведенного анализа рисков
- На основе имеющейся модели угроз
- На основе специальной классификации рисков безопасности по происхождению (см. Программу обучения [ISTQB_ATTA_SYL])

Любой из них может стать работоспособной основой.

В зависимости от типа проектаважно обеспечить наличие тестов безопасности на каждом соответствующем этапе разработки.

3.3.1 Проектирование тестов безопасности

Подробные тесты безопасности основаны на рисках безопасности, стратегии тестирования безопасности и других источниках, таких как модели угроз. Тесты безопасности также можно рассматривать как функциональные и структурные по своей природе. Например, в случае тестирования безопасности веб-сайта электронной коммерции, функциональными рисками безопасности могут быть SQL-инъекции, хищение учетных записей и взлом паролей. Примером структурного риска безопасности может быть состояние переполнения буфера, которое позволит злоумышленнику получить доступ через сбой в памяти.

Ниже перечислены необходимые атрибуты подробных тестов безопасности:

- Приоритетность определяется выявленными рисками безопасности и моделями угроз
- Трассируемость в соответствии с определенными требованиями безопасности
- Определенность, исходя из целевой аудитории (разработчики, тестировщики функционала, тестировщики безопасности)
- Определенность на основе профилей дефектов безопасности
- Проектируемость для автоматизации, если применимо

В модель разработки тестов безопасности рекомендуется включать:

- Подход к тестированию безопасности (уровень проекта)
- Риски, модели угроз и требования к тестированию безопасности (уровень проекта)
- Методы разработки тестов безопасности (на основе рисков, требований и приложения)
- Тестовые сценарии и сценарии безопасности

В оставшейся части этой главы представлены общие риски и уязвимости безопасности, а также связанные с ними методы разработки тестов безопасности. Новые угрозы безопасности и уязвимости возникают быстро, поэтому рекомендуется, чтобы планировщики тестов безопасности были в курсе стандартов безопасности и списков угроз, как указано в главе 9.

Ключевой принцип заключается в том, что процесс разработки тестов безопасности должен быть способен создавать и внедрять тесты на основе любого выявленного риска безопасности, требования или угрозы.

Функциональные средства контроля безопасности (например, средства контроля транзакций)
Эти тесты предназначены для проверки и подтверждения наличия средств контроля, их правильной работы и эффективности в обнаружении и предотвращении несанкционированных действий.

Пример: Банковский кассир не может санкционировать снятие наличных свыше определенной суммы без разрешения главного кассира, также зарегистрированного в системе.

Функциональные средства контроля доступа (например, имена пользователей, пароли, токены)
Эти тесты, возможно, являются тем, о чем большинство людей сразу же думают с точки зрения тестирования безопасности. Тесты включают:

- Политики имени пользователя и пароля применяются правильно
- Уровень контроля доступа соответствует риску
- Средства контроля доступа устойчивы к программному обеспечению для взлома паролей

Пример: Хищение учетных записей - это практика определения имени пользователя. После того как имя пользователя угадано или идентифицировано, пароль является оставшейся частью, необходимой для получения доступа к системе. Распространенный тест заключается в проверке того, что при вводе правильного имени пользователя с неправильным паролем в сообщении об ошибке не указывается, какой из элементов неверен.

Структурный контроль доступа (например, права доступа пользователей, уровни шифрования, аутентификация)

Тесты для этих элементов управления основаны на том, как были установлены права пользователя для доступа к данным, функционального доступа и уровней конфиденциальности. Структурный контроль доступа обычно применяется системным администратором, администратором безопасности или администратором базы данных. В некоторых случаях права доступа являются опцией конфигурации в приложении. В других случаях права доступа применяются на уровне системной инфраструктуры.

Тесты структурного контроля доступа включают создание тестовых учетных записей пользователей для каждого уровня доступа безопасности и проверку того, что каждый уровень доступа не имеет прав доступа, ограниченных для этого уровня. Например, учетные записи пользователей могут быть созданы для минимального уровня доступа, доступа уровня менеджера и доступа администратора. Необходимо провести тестирование, чтобы убедиться, что пользователь с минимальным доступом не может выполнять действия с доступом уровня менеджера.

Практики безопасного кодирования
Это в первую очередь метод статического тестирования, позволяющий определить, следуют ли разработчики программного обеспечения и систем установленным методам обеспечения безопасности при создании приложений.

Ключевой принцип заключается в том, что многие атаки на систему безопасности осуществляются за счет использования программных дефектов, заставляющих систему вести себя непредвиденным образом.

Очень короткий список методов безопасного кодирования включает:

- Для создания случайных идентификаторов сеанса используются проверенные алгоритмы и элементы управления сеансом.
- Решения по авторизации принимаются только доверенными объектами системы, находящимися под управлением организации, осуществляющей авторизацию (например, авторизация должна происходить на стороне сервера).
- Защищенная информация не должна появляться в сообщениях об ошибках. Эта информация может включать сведения о системе, идентификаторы сеансов и информацию об учетной записи.
- Ошибки приложения должны обрабатываться внутри приложения, а не зависеть от конфигурации сервера.
- Запросы HTTP GET не должны содержать конфиденциальную информацию.
- Обработчики ошибок не должны отображать трассировку стека или другую отладочную информацию.
- Все сбои при проверке ввода данных должны запротоколированы.
- Любая конфиденциальная информация, которая может временно храниться на сервере, должна быть защищена (например, следует использовать шифрование). Эта временно защищенная информация должна быть удалена, когда в ней больше нет необходимости.
- Приложение не должно иметь возможности отдавать команды операционной системе напрямую. Вместо этого для выполнения задач операционной системы следует использовать встроенные API.
- Пароли, строки подключения или другая конфиденциальная информация не должны храниться в виде открытого текста на клиентских машинах (например, в файлах cookie). Вложение такой информации в незащищенные форматы, такие как Adobe flash, скомпилированный код и механизм управления состоянием HTML-страницы Microsoft (ASP.NET), должно быть запрещено.
- Для передачи всей конфиденциальной информации следует использовать шифрование. Безопасность транспортного уровня (TLS) - это способ защиты передаваемых данных при

использовании HTTP-соединений. Для соединений, отличных от HTTP, для передачи конфиденциальной информации следует использовать шифрование.

- Данные, предоставляемые пользователем, не должны передаваться непосредственно в любую динамическую функцию «include».
- Все предоставленные пользователем данные должны быть должным образом очищены и проверены перед использованием приложением.
- Переменные должны быть строго типизированы в языках, поддерживающих проверку типов. То есть переменные должны иметь определенный тип ввода. Например, числовое поле не должно принимать буквенные символы. Это ограничение будет задано в определении типа переменной, а также в базе данных. Можно написать безопасный код на JavaScript (Node JS) или других языках, которые не поддерживают принудительную проверку типов компилятором.
- Вместо использования нового неуправляемого кода для общих задач используйте проверенный, надежный и утвержденный код, находящийся под управлением конфигурации.
- Запускайте службы с минимально возможными привилегиями (никогда под управлением root), и каждая служба должна иметь свою собственную учетную запись пользователя в операционной системе.

Список методов безопасного кодирования можно найти в кратком справочном руководстве owasp по методам безопасного кодирования [OWASP1] и в 10 лучших методах безопасного кодирования [CERT1]. Кроме того, SANS составляет список 25 самых опасных ошибок в программном обеспечении [SANS1].

Динамические тесты могут быть выполнены, чтобы определить, соблюдались ли разработчиками такие методы, как проверка данных и обмен сообщениями об ошибках. Кроме того, с помощью инструментов динамического тестирования памяти можно выявить одну из наиболее распространенных уязвимостей безопасности - переполнение буфера памяти.

Доступ к операционной системе

Получив доступ к операционной системе, злоумышленник может контролировать данные, доступ к сети и внедрять вредоносные программы. Тесты для этого могут включать тестирование на возможность внедрения вредоносных программ, скрытно действующих в зараженной системе и обладающих специальными средствами, затрудняющими их обнаружение системами безопасности (руткиты) и другого вредоносного кода в систему.

Языковые уязвимости (например, Java)

По словам исследователей безопасности из WhiteHat Security, поставщика средств защиты приложений, по всем направлениям не было существенных различий между языками, когда речь шла об уязвимостях безопасности. [WhiteHat Security, 2014] В апреле 2014 года WhiteHat security опубликовала отчет о статистике безопасности веб-сайтов, основанный на оценках уязвимостей, проведенных на 30 000 веб-сайтах клиентов с использованием фирменного сканера, и результаты показали незначительные различия в относительной безопасности таких языков, как .NET, Java, PHP, ASP, ColdFusion и Perl. Эти шесть языков имели относительно одинаковое среднее количество уязвимостей, а такие проблемы, как SQL-инъекции и уязвимости в межсайтовых сценариях, оставались широко распространенными. [WhiteHat Security, 2014] Важно понимать, что безопасный код может быть реализован на многих языках, так же как и небезопасный код. Ключевым фактором является то, как приложение кодируется (реализуется) на любом используемом языке.

Отдел cert института программной инженерии предоставляет публикации [CERT2] и инструменты [CERT3], в которых рассматриваются вопросы безопасности, специфичные для конкретного

языка. Кроме того, база данных заметок об уязвимостях [CERT4] предоставляет своевременную информацию об уязвимостях программного обеспечения. Заметки об уязвимостях включают резюме, технические подробности, информацию об исправлении и списки затронутых поставщиков.

Уязвимости платформы (например, Windows, Linux, Mac OS, iOS, Android)
Каждая вычислительная платформа имеет свой собственный набор уязвимостей безопасности. Задача тестировщика безопасности состоит в том, чтобы обеспечить своевременное применение обновлений безопасности платформы на всех устройствах, на которых работает затронутая платформа.

Внешние угрозы
Внешние угрозы безопасности — это те, о которых большинство людей думают, когда думают о кибератаках. Некоторые внешние угрозы, такие как использование уязвимостей приложений или языков, можно обнаружить, протестировать и предотвратить.

Отказ в обслуживании (DoS) — еще один тип внешней угрозы. Как правило, эти атаки основаны на перегрузке ресурсов системы или приложения, так что система или приложение становятся недоступными для законных пользователей. DoS-атаки могут быть нацелены на пропускную способность сети, подключение системы или приложения, а также на определенные службы или функции.

Атака распределенного отказа в обслуживании (DDoS) — это тип DoS, при котором атака запускается косвенно с использованием других ресурсов компьютера. Возможными методами являются усиление или использование ботнетов, представляющих собой большое количество ранее скомпрометированных компьютеров, находящихся под контролем или командой злоумышленника. Злоумышленник может получить контроль, просто запустив вирусную инфекцию или используя троянские программы. Зараженные компьютеры могут использоваться в качестве агентов, каждый из которых отправляет трафик определенной жертве (сети), выбранной злоумышленником.

При использовании атак с усилением или отражением злоумышленник использует уязвимость (или даже желаемую функциональность) в конкретных протоколах (например, DNS или NTP). Злоумышленник отправляет большой объем трафика на широковещательные IP-адреса (несколько хостов), содержащие поддельный исходный адрес жертвы. Это приводит к тому, что широковещательная служба повторяет этот трафик по адресу жертвы и умножает первоначальный объем трафика на количество хостов. Когда злоумышленник отправляет такого рода запросы несколько раз в секунду, жертва внезапно сталкивается с большим количеством ответов, которые ей приходится отправлять.

Пример: Злоумышленник А отправляет запрос в систему Б для получения полного списка всех известных записей DNS, выдавая себя за жертву В, часто с поддельным IP-адресом. Затем система Б отправит полный список Жертве В, которая заполнит сервер жертвы В увеличенным объемом данных.

Другой формой DoS-атак являются атаки на истощение ресурсов. Такого рода атаки злоупотребляют желаемой функциональностью, потребляя вычислительные ресурсы (процессор, память, дисковое хранилище и т.д.), необходимые для обеспечения функциональности.

Пример: Одной из функций протокола SSL является возможность генерировать новые ключи в существующем сеансе, если клиент или сервер подозревает, что сеанс скомпрометирован. Генерация ключей — ресурсоемкий процесс. Когда злоумышленник посылает запрос на генерацию новых ключей несколько раз в секунду, плохо настроенная или незащищенная система

может оказаться в ситуации, когда она только генерирует новые ключи и у нее не остается ресурсов для выполнения других действий.

Наконец, существуют так называемые логические DoS-атаки, когда злоумышленник может злоупотребить предусмотренной функциональностью, чтобы помешать другим пользователям получить доступ к системе.

Пример: Приложение использует предсказуемые имена пользователей и блокирует пользователя навсегда после трех неудачных попыток входа. Злоумышленник может угадать имена пользователей и заблокировать множество учетных записей в системе, в результате чего многие пользователи не смогут получить к ней доступ (и косвенно совершить DoS-атаку на службу поддержки).

Существует четыре уровня тестирования для DDoS-атак.

1. Проверка, чтобы убедиться, что компьютеры не заражены известными вредоносными программами
2. Проверка способности системы обнаружения вторжений быстро идентифицировать несколько запросов с одного компьютера за короткий промежуток времени
3. Определение конфигурации, позволяющий использовать функции, которые могут быть использованы злоумышленником (например, SSL, веб-сервер, DNS)
4. Выявление логических дефектов, которые могут привести к DoS

Вторжения - это еще одна форма внешней атаки. Существует множество способов осуществить внешнее вторжение в систему. Эти атаки основаны на том, что кто-то «взламывает» систему, чтобы получить информацию. Некоторые из этих методов перечисленны в приведенном ниже списке:

- Социальная инженерия
- Инъекционные атаки (SQL, вредоносный код)
- Компрометация учетной записи (хищение, сброс пароля)
- Эксплуатация известных уязвимостей (брандмауэр, операционная система, фреймворк, приложение)
- Атаки вредоносного ПО
- Атаки на небезопасную конфигурацию
- Дефекты авторизации
- Атаки на логику приложения (использование дефектов приложения, особенно в веб-приложениях, для злоупотребления функциями - например, выполнение шагов не по порядку для покупок в приложении для электронной коммерции с целью получения скидки или кредита)

Перехват сетевой передачи, отправленной изнутри организации кому-либо в другой организации, считается не атакой вторжения, а скорее внутренним нарушением.

Внутренние угрозы

Самые большие угрозы могут быть внутренними. Рассмотрим следующие источники внутренней атаки:

- Корпоративный шпионаж, когда доверенный сотрудник может продать корпоративную информацию, включая информацию об учетной записи клиента, коммерческую тайну, информацию о доступе сотрудников и т. д.
- Информация, полученная аутсорсинговыми разработчиками, тестировщиками и другим персоналом (например, представителями службы поддержки клиентов). Иногда люди уходят из аутсорсинговой компании и уносят информацию с собой в голове.
- Кража жестких дисков и других физических устройств хранения данных

- Недовольные сотрудники, которые стремятся нанести ущерб компании путем утечки конфиденциальной информации или совершения хищений, выплачивая себе деньги под видом законных (но сфабрикованных) счетов.

Формат и структура теста безопасности
У каждой организации, проводящей тестирование безопасности, будет свой способ форматирования подробных тестов. Часто для разработки тестов безопасности можно использовать тот же формат, что и для других видов тестирования, с единственным различием в цели тестирования и тестовой среде.

Даже если организация следует таким стандартам, как IEEE 829-2008 и ISO 29119 [ISO/IEC/IEEE 29119-3], использование стандарта должно соответствовать потребностям организации. Однако эти стандарты формируют общее понимание того, что должно содержаться в различных документах по планированию тестирования. Во многих случаях тестовые сценарии и тестовые процедуры (автоматизированные сценарии) могут быть определены и реализованы в инструменте управления тестированием, который часто обеспечивает структуру форматирования.

Тестовые сценарии являются наиболее автономной формой описания тестов. Они не требуют последовательного выполнения. Если для достижения определенной цели тестирования необходимо последовательное выполнение, тестовые сценарии объединяются в последовательность, выраженную в тестовой процедуре или автоматизированном сценарии. Тест-кейсы обычно используются для тестирования отдельных условий. Например, при тестировании безопасности тестирование функции входа в систему может состоять из тест-кейсов, предназначенных для проверки правильности соблюдения требований к форматированию пароля.

Во время реализации теста тест-кейсы разрабатываются, приоритизируются и организуются в спецификации процедуры тестирования. Процедура тестирования определяет последовательность выполнения тест-кейсов. Если тесты выполняются с помощью инструмента выполнения тестов, последовательность действий указывается в сценарии тестирования (который представляет собой автоматизированную процедуру тестирования). Процедуры тестирования используются, когда важна последовательность действий. Например, процедура тестирования будет полезна при тестировании процесса «восстановления потерянного пароля».

Когда необходимы тесты, основанные на опыте, такие как исследовательские тесты, условия тестирования и ожидаемые результаты не определяются до начала тестирования, но протестированные условия и фактические результаты должны быть записаны тестировщиком безопасности для отчетности.

3.3.2 Проектирование тестов безопасности на основе политик и процедур

При разработке тестов для проверки политик и процедур безопасности эти элементы становятся базисом тестирования. С этой точки зрения, тесты безопасности являются средством аудита безопасности.

Политики и процедуры безопасности не должны быть единственным базисом тестирования, поскольку необходимы другие точки зрения на тестирование безопасности. Целями проектирования тестов для проверки политик и процедур безопасности являются:

- Понимание целей и областей действия политики/процедуры
- Оценка возможностей тестирования политик/процедур
- Создание тестов, непосредственно связанных с политикой/процедурой

Например, может существовать процедура, которая гласит: «*Все ИТ-системы компании XYZ ограничивают количество неудачных попыток входа в систему до трех. После трех неудачных попыток входа в систему наступает определенный период блокировки. Лица, не имеющие соответствующей информации о локальной учетной записи пользователя, не смогут получить доступ к нашей ИТ-системе и должны обратиться в службу ИТ-поддержки для подтверждения личности и получения временного пароля.*»

Это процедура, проверка которой потребует следующих шагов:

1. Трижды безуспешно попытайтесь войти в приложение. При третьей неудачной попытке должно появиться сообщение о блокировке. Все последующие попытки войти в учетную запись получают сообщение о блокировке.
2. Обратитесь в службу ИТ-поддержки и подтвердите личность. На известный адрес электронной почты будет выдан временный пароль.
3. Войдите в систему с временным паролем. Доступ должен быть предоставлен.
4. Создайте новый пароль, соответствующий политике паролей. Новый пароль должен быть принят.
5. Выйдите из системы.
6. Войдите с вновь созданным паролем. Доступ должен быть предоставлен.

Обратите внимание, что шаг 4 также дает возможность протестировать политику паролей.

Не все политики безопасности можно полностью протестировать. Например, «*Содержание записей аудита XYZ, Inc. содержит все проверенные события с отметкой даты/времени и прослеживается до конкретных лиц. Журналы, относящиеся к конкретному производителю, которые предоставляют достаточную информацию для выполнения этих требований, должны считаться достаточными для целей аудита.*»

Хотя тестирование не является невозможным, необходимо определить и провести тест, чтобы охватить все проверяемые события. Необходимо выполнить действия, чтобы вызвать выборочный набор событий, которые будут занесены в журналы аудита, и проверить точность занесенной в журнал информации, например, идентификатор пользователя и отметки даты и времени.

3.4 Выполнение тестов безопасности

3.4.1 Ключевые элементы и характеристики эффективного окружения тестирования безопасности

Хотя многие формы тестирования могут использовать тестовое окружение, расположенное на том же сервере и в той же сети, что и другие системы, тестирование безопасности сопряжено с уникальными рисками, которые требуют отдельного подхода к созданию тестового окружения. Это особенно верно при тестировании ненадежных приложений (например, от стороннего поставщика или поставщика с открытым исходным кодом).

Некоторые тесты безопасности, такие как тестирование функциональных элементов управления и управления сеансами, можно выполнять в типичном интегрированном тестовом окружении без высокого риска. Однако при тестировании неизвестного и ненадежного кода возможность повреждения сервера и/или сети вредоносной программой делает целесообразным тестирование в изолированном или виртуальном тестовом окружении.

Основными атрибутами окружения тестирования безопасности являются следующие:

1. Изолированность — от других систем (в зависимости от уровня риска вредоносного ПО)
2. Полнота — тестовое окружение должно отражать целевое (производственное) окружение с точки зрения:
   - Тестируемой системы или приложения
   - Операционной системы (точная версия и конфигурация)
   - Сети
   - Программное обеспечение промежуточного слоя
   - Настольных компьютеров (марка оборудования, процессор, память)
   - Мобильных устройств (производитель, процессор, память, управление питанием)
   - Баз данных
   - Прав доступа
   - Браузеров и плагинов
   - Взаимодействующих приложений
   - Данных (инженерные тестовые данные или производственные данные, которые были запутаны)
3. Восстанавливаемость — для повторения тестов по мере необходимости и восстановления после повреждения в случае его возникновения.

3.4.2 Важность планирования и согласований в тестировании безопасности

Существует несколько причин, по которым тестировщик безопасности должен получить одобрение перед выполнением тестов безопасности:

- Почти во всех странах противозаконно (пытаться) получить доступ к системам данных и их информации. В некоторых странах даже запрещено законом иметь доступ к инструментам тестирования безопасности. Это означает, что в большинстве тестов безопасности вы будете нарушать один или несколько законов. Единственный возможный способ провести тестирование — получить отказ от тестируемой системы или владельца данных и одобрение вашего руководства.
- Тесты безопасности могут вызывать предупреждения об обнаружении вторжений, и тестировщик может показаться злоумышленником. Тестирование на проникновение - это особый случай, когда такая авторизация особенно необходима.
- Тесты безопасности могут привести к серьезным системным сбоям и отключению электроэнергии. Риск должен быть известен, и должны быть приняты возможные меры предосторожности.

Без предварительного и специального разрешения на тестирование безопасности тестировщик может нарушить политику и процедуры безопасности. Это может привести к увольнению или судебному преследованию тестировщика.

Форма разрешения на проведение тестирования безопасности должна содержать следующую информацию:

- Наименование уполномоченного органа
- Имена тестирующего персонала и/или организации
- Техническое задание
- Даты выдачи разрешения (от/до)
- Другие важные сведения, такие как исходные IP-адреса, учетные записи пользователей и т. д.
- Аттестации:
  - Клиент является владельцем тестируемой системы
  - Клиент имеет право разрешить тестирование безопасности
  - Клиент выполнил резервное копирование всех систем и данных

  - Клиент проверил, что система может быть восстановлена из резервных копий в случае необходимости
  - Клиент понимает риски, связанные с тестированием безопасности
- Положение о страховании ответственности для проверяющей организации
- Подписи представителя клиента, уполномоченного заключать такие соглашения

Образец формы можно найти на [OWASP3].

3.5 Оценка теста безопасности

Как и большая часть тестирования, оценка тестов безопасности проводится по мере выполнения отдельных тестов. Оценка теста безопасности - это оценка результатов теста безопасности. При выявлении дефектов безопасности (уязвимостей) необходимо составить отчет об инциденте, в котором, как минимум, указывается следующее:

- Имя тестировщика, обнаружившего уязвимость
- Тестовое окружение, в котором была обнаружена уязвимость
- Выполненные шаги теста (для облегчения воссоздания результатов тестирования)
- Природа уязвимости системы безопасности
- Степень уязвимости системы безопасности
- Потенциальное влияние уязвимости системы безопасности
- Предлагаемый порядок действий по исправлению

Отчеты об инцидентах, связанных с испытаниями системы безопасности, могут быть зарегистрированы в той же системе управления инцидентами, что и отчеты о других видах тестирования. Отчеты о тестах безопасности должны быть отнесены к особой категории и могут нуждаться в ограничении доступа для предотвращения просмотра неуполномоченным персоналом. Такие ситуации могут возникнуть, когда:

- Тестирование безопасности проводится независимой организацией, а сообщения об инцидентах поступают в инструмент, который имеет мало ограничений на просмотр отчетов об инцидентах
- Уязвимости в системе безопасности могут быть выявлены, но не сразу устранены
- Внутренний персонал может рассматриваться как потенциальная угроза для использования уязвимостей системы безопасности

ИТ-аудитор должен быть в состоянии принять решение о том, следует ли ограничивать доступ к результатам тестирования безопасности.

По завершении основных мероприятий по тестированию безопасности, например по завершении системного тестирования, может быть выпущен окончательный отчет о тестировании безопасности. Этот отчет также может считаться конфиденциальным в зависимости от статуса устранения уязвимости.

3.6 Сопровождение тестов безопасности

Во многих случаях изменение процесса тестирования безопасности может заключаться только в добавлении новых типов тестов в ответ на новые типы угроз. Однако одно можно сказать наверняка - цели и угрозы тестирования безопасности меняются ежедневно, поэтому процесс тестирования безопасности должен быть спроектирован таким образом, чтобы его можно было легко изменить.

На рынке также появляются новые инструменты, помогающие выполнять тесты безопасности. Тестировщики безопасности должны следить за этими достижениями и оценивать, какие инструменты могут повысить мощность и гибкость тестирования безопасности.

4. Тестирование безопасности на протяжении всего жизненного цикла программного обеспечения - 225 мин.

Ключевые слова
сценарий злоупотребления, тестирование на случайных данных

Цели обучения для тестирования безопасности на протяжении всего жизненного цикла программного обеспечения

4.1 Роль тестирования безопасности в жизненном цикле программного обеспечения

AS-4.1.1 (K2) Объяснить, почему безопасность лучше всего достигается в рамках процесса жизненного цикла разработки программного обеспечения
AS-4.1.2 (K3) Осуществлять действия, связанные с безопасностью, для заданного жизненного цикла программного обеспечения (например, итеративный, последовательный)

4.2 Роль тестирования безопасности в требованиях

AS-4.2.1 (K4) Анализировать заданный набор требований с точки зрения безопасности для выявления недостатков

4.3 Роль тестирования безопасности при проектировании

AS-4.3.1 (K4) Анализировать данный проектный документ с точки зрения безопасности для выявления недостатков

4.4 Роль тестирования безопасности в деятельности по внедрению

AS-4.4.1 (K2) Понимать роль тестирования безопасности во время тестирования компонентов
AS-4.4.2 (K3) Реализовать тесты безопасности на уровне компонентов (абстрактные) с учетом заданной спецификации кодирования
AS-4.4.3 (K4) Анализировать результаты тестирования на определенном уровне компонентов для определения адекватности кода с точки зрения безопасности
AS-4.4.4 (K2) Понимать роль тестирования безопасности во время тестирования интеграции компонентов
AS-4.4.5 (K3) Реализовать тесты безопасности интеграции компонентов (абстрактные) с учетом определенной спецификации системы

4.5 Роль тестирования безопасности в деятельности по системному или приемочному тестированию

AS-4.5.1 (K3) Реализовать сквозной тестовый сценарий для тестирования безопасности, который проверяет одно или несколько заданных требований безопасности и тестирует описанный функциональный процесс
AS-4.5.2 (K3) Продемонстрировать способность определять набор критериев приемки для аспектов безопасности данного приемочного теста

4.6 Роль тестирования безопасности в техническом обслуживании

AS-4.6.1 (K3) Реализовать подход к сквозному повторному/регрессионному тестированию безопасности на основе заданного сценария

4.1 Роль тестирования безопасности в жизненном цикле программного обеспечения

Безопасность не тестируется и не вносится в уже созданное приложение. Как правило, она достигается за счет ориентированного на безопасность проектирования и тестирования на протяжении всего процесса создания приложения. Как и тестирование программного обеспечения в целом, тестирование безопасности является процессом, который должен происходить в рамках жизненного цикла разработки.

4.1.1 Взгляд на жизненный цикл тестирования безопасности

Процесс жизненного цикла программного обеспечения обеспечивает основу выполнения определенных действий в те моменты времени, которые связаны с другими действиями. Например, потребности пользователя должны быть получены до начала проектирования приложения. Выбор жизненного цикла программного обеспечения зависит от характера организации, проекта и подобных факторов [IEEE 12207]. Для целей этой программы обучения и целей тестирования безопасности концепции и методы могут быть применены к любому процессу жизненного цикла – последовательному или итеративному.

В главе 3 этой программы обучения был описан процесс тестирования безопасности, который соответствует образцу общего жизненного цикла программного обеспечения. Причины интеграции тестирования безопасности в жизненный цикл программного обеспечения обсуждаются в следующих разделах.

Выделение и обеспечение времени в жизненном цикле программного обеспечения, когда должны происходить действия, связанные с безопасностью
Например, при выявлении и определении потребностей пользователей бизнес-аналитик или системный аналитик должен выяснить:

- Какие уровни доступа к безопасности необходимы?
- Существуют ли какие-либо цифровые или физические активы, требующие специальной защиты?
- Насколько "открытым" должно быть приложение?
- Каковы риски безопасности?

Другой пример относится к времени кодирования. В это время разработчик имеет наилучшую возможность применить методы безопасного кодирования, чтобы избежать таких атак, как SQL-инъекции и атаки, направленные на переполнение буфера памяти. Обнаружение таких уязвимостей на более поздних стадиях проекта сложно и дорого, поскольку многие другие программные компоненты также могут нуждаться в аналогичном устранении и исправлении.

Обеспечение контрольных точек для проверки
Например, следует пересмотреть требования к безопасности или пользовательские истории, чтобы убедиться, что аспекты потребностей пользователя, связанные с безопасностью, были надлежащим образом исследованы и задокументированы. Изменения в коде также должны быть проверены на предмет выявления наличия вредоносного кода, внесенного со стороны внутренних сотрудников или подрядчиков.

Предоставление контрольных точек для тестирования
Например, при разработке следует документировать и проводить компонентные тесты, чтобы убедиться, что практика безопасного кодирования была соблюдена и успешно реализована.

Обеспечение критериев входа и выхода на протяжении всего проекта
Примером такой практики может служить то, что ни один компонент не может быть принят в интегрированное тестовое окружение до тех пор, пока результаты тестирования не продемонстрируют, что все действия, связанные с безопасностью (как разработка, так и тестирование), были успешно завершены. Это особенно важно на более поздних стадиях проекта, когда уязвимость безопасности может привести к возникновению риска безопасности, влияющего на всю систему или приложение.

4.1.2 Действия, связанные с безопасностью, в жизненном цикле программного обеспечения

Следующие действия, связанные с безопасностью, выполняются наряду с другими действиями проекта, а не в рамках отдельного жизненного цикла.

Требования: Требования собираются и определяются различными способами в зависимости от используемого жизненного цикла программного обеспечения. Следует признать, что требования могут инициироваться не только потребностями пользователей и заинтересованных сторон. Например, среди прочих могут быть нормативные требования, технические требования и бизнес-требования.

Цели требований включают:
- Понимание и определение потребностей в области безопасности со всех точек зрения как внутри организации, так и за ее пределами. Например, клиент бизнеса не входит в организацию, но ему необходимо, чтобы его личная информация оставалась в безопасности.
- Документирование потребностей в безопасности подробным и однозначным образом. Это позволяет проводить реализацию и тестирование, которые трассируются к требованиями, что позволяет проверять и подтверждать требования.

Мероприятия по выполнению требований включают:
- Определение всех заинтересованных и осведомленных лиц, которые могут внести свой вклад в разработку требований.
- Используя различные методы - интервью, семинары и т.д., собрать потребности в безопасности, выраженные каждой группой. Это также может быть выполнено во время выявления других требований.
- Документирование требований таким образом, чтобы их можно было просмотреть и отследить.
- Рецензирование требований на правильность, полноту, понятность и однозначность.

Проектирование: Система или приложение разрабатываются на основе потребностей, указанных в требованиях. Требования выражают потребности в безопасности, а проектирование воплощает эти потребности в работоспособный подход к решению.

К задачам проектирования относятся:
- Создание проекта системы или приложения, отвечающего заявленным требованиям безопасности

Проектная деятельность включает:
- Анализ документированных требований
- Достижение наиболее работоспособного подхода к разработке приложения безопасным способом

- Документирование проекта с использованием соответствующих методов в соответствии с жизненным циклом программного обеспечения. Например, при итеративном подходе сессии проектирования могут проводиться на доске, в то время как в других процессах проектирование может быть выражено в моделях.

Реализация: Это обычно известно как деятельность по кодированию.

Цели реализации включают:
- Перевод требований и дизайна в безопасный код, который отвечает функциональным потребностям, указанным в требованиях
- Внедрение любых других необходимых процедур или технологий (межсетевых экранов, токенов и т. д.) для обеспечения безопасности

Мероприятия по внедрению включают:
- Создание кода, отвечающего требованиям безопасности
- Выполнение компонентного тестирования для проверки правильности, эффективности и безопасности реализации
- Выполнение рецензирования компонентов для визуального контроля правильности, эффективности и безопасности внедрения

Системное тестирование:
Обратите внимание, что некоторые модели жизненного цикла программного обеспечения, такие как итерационные подходы к поставке, добавляют новые компоненты или улучшают существующие компоненты в течение более короткого промежутка времени и позволяют проводить системное тестирование гораздо чаще, чем другие, более последовательные подходы.

Цели системного тестирования включают:
- Выполнение сквозного тестирования для наблюдения за функционированием и производительностью системы в целом (аппаратное обеспечение, программное обеспечение, данные, люди и процедуры) после того, как различные компоненты системы были внедрены и интегрированы в полную систему
- Проверка правильности реализации требований безопасности с точки зрения системы

Мероприятия по системному тестированию включают:
- Выполнение тестов безопасности в некотором приближении к конечному окружению, что требует перехода от окружения разработки, в котором выполнялись предыдущие действия по внедрению и интеграции

Приемочное тестирование:
Это заключительный уровень тестирования, в ходе которого пользователи или представители пользователей системы убеждаются в том, что система предоставит необходимые возможности в целевом окружении.

Цели приемочного тестирования включают:
- Наличие пользователей или агентов, действующих от имени пользователей, для проведения тестирования безопасности в соответствии с критериями приемки, связанными с безопасностью, установленными для системы. Во многих случаях критерии приемки, связанные с безопасностью, сосредоточены на функциональных средствах контроля безопасности и процессах.

Мероприятия по приемочному тестированию включают:

- Установку системы в ее операционное окружение
- Выполнение тестов безопасности на основе критериев приемки
- Определение приемки по результатам испытаний

Следует отметить, что как системное, так и приемочное тестирование по существу представляют собой тесты «черного ящика» или тесты на стимулы-реакции без учета внутренней структуры или поведения компонентов в рамках всей системы. Предшествующее компонентное и интеграционное тестирование обеспечивают взаимодополняющие оценки, рассматривая и используя внутреннюю архитектуру компонентов и их взаимодействие в системе.

Сопровождение:
После ввода системы в эксплуатацию могут потребоваться дополнительные усилия по устранению дефектов в выпущенной версии (корректирующее обслуживание), адаптации к другим изменениям в операционном окружении (адаптивное обслуживание) или расширению или улучшению функций (профилактическое техническое обслуживание).

С точки зрения тестирования безопасности при сопровождении системы основное внимание уделяется тестированию изменений, вносимых для исправления дефектов (подтверждающее тестирование) и основной функциональности (регрессионное тестирование):

- Убедитесь, что в результате работ по техническому обслуживанию в систему не были внесены новые уязвимости
- Убедитесь, что существующие средства защиты по-прежнему эффективны после изменения

В этом контексте техническое обслуживание может включать обновления (например, операционной системы, баз данных), изменения кода, преобразование данных и миграцию платформы.

По сути, к любой деятельности по техническому обслуживанию следует относиться с такой же тщательностью и вниманием, как и к первоначальной разработке. В противном случае риск внедрения новых уязвимостей может серьезно поставить под угрозу безопасность операционной системы.

4.2 Роль тестирования безопасности в требованиях

В отношении требований в целом необходимо учитывать следующие соображения:

- Перед многими организациями стоит задача просто написать основные пользовательские требования, которые должны быть четкими, однозначными, полными, правильными и проверяемыми.
- Требования сильно подвержены изменениям на протяжении всего проекта, и поэтому поддержка требований может быть сложной задачей.
- Необходимы специальные навыки, чтобы понять потребности пользователя и другие потребности, такие как соответствие требованиям и технические потребности, прежде чем иметь возможность оформлять их в документы или вносить в инструменты управления требованиями.
- В требованиях могут быть пропуски и ошибки. Поэтому необходимы как верификация, так и валидация.
- Требования должны содержать потребности в характеристиках качества, таких как безопасность, производительность, удобство использования и т. д. Однако эти атрибуты часто ограничиваясь лишь требованиями к функциональности.

Задача состоит в том, чтобы добиться понимания и отражения перспективы безопасности в полном наборе требований к проекту. При оценке требований эффективным методом является использование контрольных списков в качестве руководства. Контрольный список может содержать множество пунктов, охватывающих различные темы. Для атрибутов, связанных с безопасностью, хорошей отправной точкой для оценки является следующее:

Потребности в конфиденциальности

- Были ли определены и задокументированы все группы пользователей и связанные с ними потребности в конфиденциальности данных?
- Определены ли все типы данных, на которые влияет требование конфиденциальности данных, и определены ли соответствующие потребности в конфиденциальности?
- Были ли определены и установлены права доступа пользователей?

Потребности в соответствии (политикам безопасности)

- Были ли определены и задокументированы все соответствующие политики безопасности?
- Были ли выявлены и задокументированы какие-либо исключения из политик безопасности?

Общие уязвимости – Они будут меняться со временем по мере изменения атак на систему безопасности, но их следует определять как риски в момент определения требований. Они также становятся основой для тестов безопасности.

- Все ли общие и известные уязвимости безопасности для документируемой функции были определены как известные риски?

Тестируемость

- Написано ли требование таким образом, что на его основе можно написать тесты безопасности и другие тесты?
- Определены ли и уточнены ли какие-либо неоднозначные термины, такие как «обработка должна быть безопасной» и «доступ предоставляется только уполномоченному персоналу», чтобы быть конкретными и проверяемыми?

Практичность – Существуют компромиссы между безопасностью и удобством использования. Например, вход пользователя на веб-сайт может быть настолько запутанным и сложным, что клиенты сдаются и переходят на другой сайт.

- Отражают ли требования соответствующий уровень процесса обеспечения безопасности по отношению к заданной функции?
- Являются ли процедуры безопасности четкими и понятными?
- Предусмотрены ли средства правовой защиты для авторизованных пользователей, у которых могут возникнуть проблемы с доступом к информации?

Производительность – Существует компромисс между безопасностью и производительностью. Например, высокий уровень шифрования может снизить производительность.

- Отражают ли требования соответствующий уровень эффективности безопасности по отношению к указанной функции?

4.3 Роль тестирования безопасности при проектировании

Следует выявлять и избегать методов проектирования, снижающих безопасность. Действия, связанные с тестированием, вносят свой вклад, распознавая конструкции программных систем,

которые могут быть уязвимы для компрометации, и направляя разработку программных систем с сильными, идентифицируемыми свойствами безопасности.

Центр безопасного проектирования компьютерного общества IEEE [IEEE1] рекомендует следующие ключевые подходы к проектированию:

- Зарабатывайте или отдавайте, но никогда не предполагайте, доверяйте. Убедитесь, что все данные от ненадежного клиента проверены. Предполагайте, что данные скомпрометированы. Не используйте авторизацию, контроль доступа, применение политик и использование конфиденциальных данных в клиентском коде
- Используйте механизм аутентификации, который невозможно обойти или подделать
- Авторизуйте после аутентифицирования
- Строго разделяйте инструкции по обработке данных и управлению и никогда не используйте инструкции по управлению процессом, полученные из ненадежных источников
- Определите подход, обеспечивающий явную валидацию всех данных
- Правильно используйте криптографию
- Определяйте конфиденциальные данные и способы их обработки
- Всегда учитывайте интересы пользователей
- Понимайте, как интеграция внешних компонентов может увеличить покрытие атаки
- Будьте гибкими при рассмотрении будущих изменений объектов и действующих лиц

4.4 Роль тестирования безопасности в деятельности по внедрению

Тестирование безопасности, как и другие виды тестирования, начинается с самого нижнего уровня реализации, используя отдельные программные компоненты, которые будут собраны в общую систему. После статической оценки этих компонентов тестирование обеспечивает дополнительный уровень оценки, исследуя динамическое поведение в ответ на допустимые и недопустимые входные данные.

4.4.1 Тестирование безопасности во время компонентного тестирования

4.4.1.1 Рекомендации по тестированию методом белого ящика/стекляного ящика

Уже отмечалось, что статическое тестирование включает весь спектр мероприятий по инспекции, разбору, аудиту и техническому рецензированию.

Так называемое тестирование методом белого ящика и/или стеклянного ящика (структурное) относится к тестам, спроектированным на основе доступности архитектуры или реализации программного обеспечения. Напротив, тестирование методом черного ящика (функциональное и нефункциональное) не основано на доступе к такой структурной информации и является просто тестированием «стимул-реакция».

Тестирование методом белого ящика может быть направлено на конкретные элементы управления, реализованные в модуле, и определять их эффективность. Наглядность структуры компонентов также позволяет измерить покрытие тестами, в виде процента выполняемых операторов, процента выполненных альтернатив или процента пройденных логических путей.

Структурное тестирование безопасности может выполняться с помощью автоматизированных инструментов статического анализа и инструментов сканирования безопасности. Тестирование на

случайных данных (фаззинг) — это метод тестирования безопасности, используемый для обнаружения уязвимостей безопасности путем ввода огромных объемов случайных данных в тестируемый компонент или систему. Тестирование на случайных данных методом белого ящика (на небольших программных блоках, функциях, классах) может дать приемлемые результаты за гораздо меньшее время, чем тестирования на случайных данных методом черного ящика.

Инструменты тестирования на случайных данных методом белого ящика способны обнаружить утечки памяти, переполнение буфера и т.д., путем инструментального тестирования кода.

Во время структурного тестирования могут быть выявлены и устранены следующие уязвимости безопасности:

- Переполнения буфера памяти
- Вредоносный код, внедренный внутренним сотрудником или подрядчиком
- Доступ через «черный ход» (доступ через недокументированный интерфейс, известный только разработчику, который был намеренно реализован для обхода обычных средств контроля безопасности)

4.4.1.2 Рекомендации по функциональному тестированию безопасности

Адекватность тестирования безопасности на любом уровне должна определяться подтверждением соответствия заданным требованиям безопасности. Это в дополнение к наблюдению за реакциями на стрессы, которые явно не указаны в требованиях безопасности, оценках рисков безопасности и аналогичных документах. При тестировании на наличие слабых мест в системе безопасности требуется творческий подход, поскольку тестировщики исследуют то, что разработчики программного обеспечения упустили из виду.

4.4.2 Проектирование тестов безопасности на компонентном уровне

Один из примеров набора передовых методов кодирования высокого уровня можно найти в статье «10 Лучших практик безопасного кодирования» [CERT1], в которой говорится:

«Тесты для любого компонента должны включать поиск и оценку возможных нарушений этих практик:

- Валидация ввода.
- Обращение внимания на предупреждения компилятора.
- Архитектура и проектирование политик безопасности.
- Наибольшее упрощение.
- Запрет по умолчанию.
- Использование принципа наименьших привилегий.
- Очистка данных, отправляемых в другие системы.
- Практика глубокой защиты.
  Управление рисками с помощью нескольких защитных стратегий, чтобы, если один уровень защиты оказался неадекватным, другой уровень защиты мог предотвратить превращение недостатка безопасности в уязвимость, которую можно использовать, и/или ограничить последствия успешного использования.
- Использование эффективных методов обеспечения качества.
  Хорошие методы обеспечения качества могут быть эффективными при выявлении и устранении уязвимостей. Нечеткое тестирование, тестирование на проникновение и аудит исходного кода должны быть включены в эффективную программу обеспечения качества. Независимые проверки безопасности могут привести к созданию более безопасных систем. Внешние рецензенты привносят независимую точку зрения.
- Принятие стандарта безопасного кодирования.»

Тесты, проводимые по таким контрольным спискам лучших практик, должны включать оценку возможных нарушений этих практик на основе хорошо задокументированного анализа рисков, включающего реалистичное моделирование угроз. Другими словами, сосредоточьтесь на наиболее важных требованиях с точки зрения вероятности атаки и последствий компрометации.

4.4.3 Анализ тестов безопасности на компонентном уровне

Одним из ключевых показателей адекватности является оценка покрытия тестами. Различные меры охвата зависят от характера проводимого тестирования.

Тестирование на основе требований проверяет систему, чтобы обеспечить уверенность в том, что она удовлетворяет указанным требованиям. Без учета реализации (черный ящик) охват можно измерить одним из следующих способов:

- Процент общего количества протестированных требований
- Процент проверенных случаев использования/злоупотребления
- Процент протестированных критически важных функций, сценариев или потоков миссий

Тестирование на основе данных обеспечивает уверенность поведения системы в диапазоне и комбинации входных данных, пытаясь выбрать как можно меньше тестовых значений путем разделения пространства данных на классы эквивалентности и выбора одного представителя из каждого класса (с ожиданием, что элементы этого класса эквивалентны с точки зрения их способности обнаруживать отказы). Попарные и N-образные критерии охвата являются типичными формами критериев охвата данных.

Тестирование на основе моделей обеспечивает гарантию охвата с точки зрения выбранной нотации моделирования. Если в модели используется нотация «до-после», критерии могут включать охват причинно-следственных связей и охват всех дизъюнктов в постусловии.

Для моделей на основе переходов, которые используют явные графы, содержащие узлы и дуги, критерии покрытия графа включают процент узлов (состояний), процент переходов, процент пар переходов и процент циклов.

Структурное тестирование обеспечивает уверенность, основанную на видимости и анализе конкретной реализации. При простом перечислении охват тестами обычно определяется как процент пакетов, классов, методов, решений или строк исполняемого кода в приложении, которые выполняются тестами. Последнее называется покрытием операторов.

Цикломатическая сложность - это мера того, сколько различных независимых путей существует через элемент, котораяможет быть представлена в виде графа потока управления с узлами (точками принятия решений) и дугами (путями). Самым сильным из критериев, основанных на потоке управления, является покрытие пути, которое измеряет все пути входа-выхода в графе потока управления. Поскольку исчерпывающее тестирование пути, как правило, невозможно прежде всего из-за циклов, другие менее строгие критерии могут быть выражены в терминах выбранных логических путей, считающихся критическими (покрытие критического пути), или процентного соотношения результатов принятых решений (покрытия ветвлений).

4.4.4 Тестирование безопасности во время тестирования интеграции компонентов

Поскольку компоненты более низкого уровня интегрируются в подсистемы и, в конечном итоге, в целевую систему, возможности нарушений безопасности - это не просто совокупность уязвимостей в каждом из компонентов, рассматриваемая отдельно. Вместо этого становятся

возможными новые векторы атак на основе взаимодействия между компонентами и более крупными системными и организационными элементами.

С другой стороны, некоторые взаимодействия между компонентами могут смягчать или блокировать возможные последовательности, ведущие к нарушениям безопасности. Опять же, тестировщики безопасности должны проявить творческий подход к поиску того, на что разработчики не обратили внимания.

Интеграционное тестирование может продемонстрировать сложность конструкции системы и стабильность ее поведения. Подход к тестированию интеграции (например, сверху вниз или снизу вверх) может повлиять на сроки выявления проблем безопасности или необходимость дополнительных тестов, специфичных для безопасности.

4.4.5 Проектирование тестов безопасности на уровне интеграции компонентов

Как и компонентное тестирование, интеграционные тесты должны разрабатываться на основе хорошо документированного анализа рисков, включающего реалистичное моделирование угроз. Поскольку отдельные компоненты интегрируются вместе, обратите внимание на то, что во время интеграции может потребоваться создание «строительных лесов» (в виде заглушек и драйверов) для тестирования неполных путей через систему. По мере добавления в систему новых реализованных компонентов эти «строительные леса» постепенно удаляются, что позволяет более полно оценить функциональность, а также найти новые пути к уязвимостям, которые могут быть использованы.

4.5 Роль тестирования безопасности в системном и приемочном тестировании

4.5.1 Роль тестирования безопасности в системном тестировании

Системное тестирование — это первая сквозная проверка полностью интегрированных компонентов. Хотя обычно это делается в среде разработки, оно должно выявить возникающие свойства системы, которые не были замечены до завершения интеграции. Требования безопасности обычно рассматриваются в сочетании с одним или несколькими функциональными требованиями.

Например, «В процессе выполнения X система не должна допускать, чтобы происходило Y». По мере проведения функциональных тестов тестировщик должен искать способы нарушения ограничений безопасности.

Функциональные требования, включая требования к безопасности, обычно относятся к императивам. Другие спецификации, такие как сценарии использования, случаи злоупотреблений, модели процессов и модели перехода состояний, описывают процедуры, которые можно использовать для определения сквозных тестовых сценариев для тестирования безопасности.

4.5.2 Роль тестирования безопасности в приемочном тестировании

Приемочное тестирование отличается от системного тестирования тем, что оно проводится в реалистичном операционном окружении, если не в реальных условиях, в которых система будет введена в эксплуатацию. Такое тестирование позволяет разумно оценивать производительность и другие виды поведения на основе взаимодействий через внешние интерфейсы. Это также,

наконец, помещает систему в условия, в которых внешние агенты угроз будут пытаться найти слабые места на ежедневной основе.

В идеале приемочное тестирование должно подтверждать достижение первоначальных целей проекта. Это достигается путем проектирования и выполнения тестов для подтверждения соответствия критериям приемки. Потребности в безопасности должны быть документированы в критериях приемки.

Лучшее время для определения и документирования критериев приемки - до разработки или покупки системы. Таким образом, между поставщиком и покупателем может быть достигнуто первоначальное взаимопонимание, даже если оба они находятся в одной организации. Также часто критерии приемки изменяются или появляются в ходе проекта, поэтому эти критерии следует анализировать на предмет их влияния на тестирование безопасности.

В контексте тестирования безопасности критерии приемки могут носить глобальный характер. Например, могут быть пункты критериев приемки, которые определяют, что является приемлемым с точки зрения общей безопасности системы. Это может включать критерии, которые применяются ко всем системным функциям, таким как аутентификация пользователя, права пользователя, уровни шифрования, журналы аудита и т. д. В других случаях могут потребоваться специальные критерии приемки безопасности. Например, для некоторых функций, таких как выдача платежей, превышающих определенную сумму, может потребоваться два человека для утверждения платежа.

4.6 Роль тестирования безопасности в техническом обслуживании

Регрессионное тестирование предназначено для подтверждения того, что все ранее приемлемые варианты поведения системы остаются неизменными после внесения изменений. Что касается негативных аспектов тестирования безопасности, то такое подтверждение будет включать проверку того, что система продолжает успешно противостоять попыткам обойти установленные средства контроля безопасности. Повышение удобства использования или эффективности особенно часто приводит к снижению средств контроля безопасности.

Регрессионные тесты безопасности должны быть сосредоточены на подтверждении соответствия всем требованиям безопасности, а также на тестировании новых уязвимостей, которые могли появиться во время обслуживания.

Регрессионное тестирование часто применяется с набором тестовых сценариев, основанных на тестировании отдельных функций. Однако при тестирования безопасности этого часто бывает недостаточно для обнаружения регрессионных дефектов, влияющих на безопасность. Сценарии сквозного регрессионного тестирования являются более надежными и обеспечивают более высокий уровень уверенности в том, что все транзакции могут быть выполнены безопасным способом.

Для этого типа регрессионного тестирования необходимо определить набор сценариев тестирования безопасности и тестировать их каждый раз, когда в систему вносятся изменения. Имейте в виду, что системные изменения могут распространяться на аппаратное обеспечение, конфигурационные файлы, операционные системы, СУБД, сетевое и программное обеспечение - и любые другие компоненты системы. Регрессионные дефекты могут появиться в результате изменений в любом из них. Некоторые из регрессионных дефектов могут оказывать влияние на безопасность.

Примеры сценариев:
Пользователи могут войти на веб-сайт и совершить покупку безопасно, без ущерба для своей личной информации.

Пользователи могут выполнять только действия, определенные их правами и привилегиями. (Пользователь, работающий в отделе расчета заработной платы, может добавить нового сотрудника, но не иметь доступа к его банковской информации).

5. Тестирование механизмов безопасности - 240 мин.

Ключевые слова

антивирус, аутентификация, авторизация, демилитаризованная зона, шифрование, межсетевой экран, хеширование, внутренняя угроза, система обнаружения вторжений, вредоносная программа, сканирование вредоносной программы, сетевая зона, фарминг, фишинг, добавление случайных данных, усиление защиты системы, сканер уязвимостей

Цели обучения для тестирования механизмов безопасности

5.1 Усиление защиты системы

AS-5.1.1 (K2) Понимать концепцию усиления защиты системы и ее роль в повышении безопасности

AS-5.1.2 (K3) Продемонстрировать, как проверить эффективность общих механизмов безопасности системы

5.2 Аутентификация и авторизация

AS-5.2.1 (K2) Понимать взаимосвязь между аутентификацией и авторизацией и то, как они применяются для безопасности информационных систем

AS-5.2.2 (K3) Продемонстрировать, как проверить эффективность общих механизмов аутентификации и авторизации

5.3 Шифрование

AS-5.3.1 (K2) Понимать концепцию шифрования и то, как оно применяется для безопасности информационных систем

AS-5.3.2 (K3) Продемонстрировать, как проверить эффективность распространенных механизмов шифрования

5.4 Межсетевые экраны и сетевые зоны

AS-5.4.1 (K2) Понимать концепцию межсетевых экранов и использование сетевых зон и их применение для обеспечения безопасности информационных систем

AS-5.4.2 (K3) Продемонстрировать, как проверить эффективность существующих реализаций межсетевых экранов и сетевых зон

5.5 Обнаружение вторжений

AS-5.5.1 (K2) Понимать концепцию инструментов обнаружения вторжений и то, как они применяются для обеспечения безопасности информационных систем

AS-5.5.2 (K3) Продемонстрировать, как проверить эффективность существующих реализаций инструментов обнаружения вторжений

5.6 Сканирование вредоносной программы

AS-5.6.1 (K2) Понимать концепцию инструментов сканирования вредоносных программ и то, как они применяются для безопасности информационных систем

AS-5.6.2 (K3) Продемонстрировать, как проверить эффективность существующих реализаций инструментов сканирования вредоносных программ

5.7 Запутывание данных

AS-5.7.1 (K2) Понимать концепцию инструментов запутывания данных и то, как они применяются для обеспечения безопасности информационных систем

AS-5.7.2 (K3) Продемонстрировать, как проверить эффективность подходов к запутыванию данных

5.8 Повышение квалификации

AS-5.8.1 (K2) Понимать концепцию обучения безопасности как деятельности в рамках жизненного цикла программного обеспечения и почему оно необходимо для обеспечения безопасности информационных систем

As-5.8.2 (K3) Продемонстрировать, как проверить эффективность обучения по вопросам безопасности

5.1 Усиление защиты системы

За прошедшие годы в качестве ключевых методов защиты цифровых и физических активов появились различные механизмы безопасности. Каждый из этих механизмов можно применять по-разному — одни с помощью инструментов и инфраструктуры, другие вручную. В большинстве случаев ни один из этих механизмов сам по себе не является достаточным для защиты информации. Каждый механизм имеет свои преимущества и недостатки.

Тестировщики безопасности должны понимать нюансы каждой линии защиты, чтобы можно было спроектировать соответствующие тесты для верификации и валидации их эффективности. Тестировщики безопасности Продвинутого уровня должны понимать последствия каждого из механизмов, описанных в этой главе, чтобы спроектировать архитектуру тестирования, которая обеспечит основу для непрерывного тестирования безопасности.

5.1.1 Понимание усиления защиты системы

Современные системы становятся все более сложными, поэтому их поверхность атаки постоянно растет. Чем больше компонентов в системе, тем больше число потенциально уязвимых мест и, соответственно, поверхность атаки. Уязвимости возникают из-за ошибок проектирования (уязвимости проектирования), дефектов исходного кода (уязвимости конструкции) или недостаточной строгости конфигурации этих систем (уязвимости конфигурации).

Усиление защиты системы - это пошаговый процесс уменьшения поверхности атаки путем применения политики безопасности и различных уровней защиты. Основная цель состоит в том, чтобы обезопасить систему и снизить риски нарушения безопасности.

В зависимости от контекста, усиление защиты может применяться на разных уровнях:

- Усиление защиты программного или аппаратного компонента
- Усиление защиты продукта/приложения
- Усиление защиты системы
- Усиление защиты системы систем

Применяемые организационно-технические средства защиты должны включать:

- Удаление ненужного программного обеспечения (может содержать дефекты)
- Удаление ненужных библиотек и инструментов разработчика (могут содержать дефекты)
- Удаление ненужных учетных записей/логинов (векторы атаки)
- Удаление ненужных приложений (могут содержать дефекты) и сетевых служб (векторы атаки)
- Удаление ненужных периферийных устройств и аппаратных портов (например, USB-портов, устройств чтения карт памяти)
- Оперативное исправление систем и установка обновлений (например, активация автоматических обновлений)
- Обновление конфигураций
- Соблюдение правил кодирования (избегание уязвимостей «при проектировании»)
- Настройка удаленного сервера ведения журнала (например, rsyslog) таким образом, чтобы в случае взлома злоумышленник мог удалить файлы журнала только на скомпрометированном компьютере, но не на удаленном сервере ведения журналов

Следует использовать следующие механизмы безопасности:

- Строгая аутентификация и эффективное управление авторизацией (предоставление выделенным ролям только тех прав, которые необходимы для выполнения действий)
- Шифрование (связь и локальное хранилище)
- Межсетевые экраны (персональные, системные или веб-приложения) и определенные зоны безопасности (например, выполнение в песочнице)
- Система обнаружения вторжений
- Защита от вредоносных программ / антишпионских программ
- Запутывание данных и приложений (например, защита от обратной разработки)

Усиление защиты системы жизненно важно для защиты конфиденциальных активов организации, но правила безопасности должны применяться на правильном уровне и быть сбалансированы с удобством использования системы. В крайнем случае этого компромисса средства защиты отключаются, поскольку они блокируют работу компании.

5.1.2 Тестирование эффективности механизмов усиления защиты системы

Тестирование эффективности механизмов усиления защиты системы может быть выполнено различными способами. Тесты будут зависеть от характера защищаемой системы или приложения, чувствительности защищаемых активов и выявленных угроз. Усиление защиты системы ограничивает доступ к системе нужными ролями, открывает только необходимые службы и следит за обновлениями приложений. Поэтому для проверки эффективности усиления защиты системы необходимо проектировать тесты, чтобы было известно, работают ли активности по усилению защиты системы, применяются ли они в нужных местах и правильными способами. Также важно проверить систему на наличие избыточных мер защиты, которые могут быть чрезмерными с учетом рисков безопасности.

Некоторые тесты усиления защиты системы могут быть основаны на рецензировании или аудите, в то время как другие тесты могут основываться на способности определенных групп пользователей выполнять определенные действия или получать доступ к определенным данным.

Тесты могут включать:

- Аудит конфигурации серверов баз данных и приложений для проверки того, что пароли по умолчанию были изменены
- Аудит конфигурации системы для выявления ненужных служб и сетевых портов
- Проверку компонентов, библиотек и версий приложений на предмет того, что они не устарели и не уязвимы

Сканер уязвимостей может быть запущен для облегчения задач оценки уязвимостей, особенно если система сложная (например, многосайтовая среда). Инструменты статического анализа можно использовать для обнаружения нарушений правил кодирования, которые могут привести к уязвимостям конструкции. Анализаторы, ориентированные на безопасность, могут быть особенно полезны для обнаружения уязвимостей.

5.2 Аутентификация и авторизация

5.2.1 Связь между аутентификацией и авторизацией

Чувствительные активы организации (например, номера банковских счетов в списке клиентов, дизайн нового продукта) должны быть защищены и доступны только уполномоченному лицу.

Аутентификация основана на проверке идентификатора пользователя и токена для ответа на вопросы:

- Логин: кто пользователь?
- Пароль: действительно ли пользователь тот, за кого себя выдает?

В зависимости от необходимости защиты от атак с целью перехвата аутентификации или кражи пароля могут использоваться различные реализации механизмов аутентификации. К ним относятся обнаружение слабых паролей, использование одноразовых паролей (OTP), снятие отпечатков пальцев, сертификаты программного обеспечения, сертификаты в аппаратных токенах и другие средства аутентификации.

В зависимости от архитектуры системы, прикладного контекста и потребностей организации (простота управления логином/паролем), механизмы аутентификации могут включать локальную аутентификацию, аутентификацию на сервере, сетевую аутентификацию, единый вход (SSO) и подобные средства.

Авторизация используется для следующих целей:

- Чтобы проверить, имеет ли аутентифицированный пользователь права на выполнение действия (например, пользователь может осуществить вход на сервер, но не может изменять его данные, или пользователь имеет право использовать FTP-сервер, но только в своем выделенном пространстве)
- Чтобы определить, какой уровень доступа должен быть разрешен к ресурсам системы

Существует тесная связь между аутентификацией и авторизацией, основанная на принципе, что неавторизованный пользователь не имеет прав или имеет ограниченные права на систему (не уполномочен манипулировать конфиденциальными данными). Например, в контексте веб-сайта торгового предприятия неавторизованный пользователь может видеть список товаров, но перед покупкой выбранного товара он должен создать учетную запись. Аутентифицированный пользователь может приобрести товар, но не может выполнять административные функции.

5.2.2 Тестирование эффективности механизмов аутентификации и авторизации

Цель злоумышленников — хищение паролей или обход системы для выполнения несанкционированных действий. Обычно они используют различные типы уязвимостей: ошибки кодирования (отсутствие фильтрации ввода), старые уязвимые версии библиотек, ошибки конфигурации системы (сохранение паролей по умолчанию, прав по умолчанию) и слабые пароли (например, наиболее часто используемый пароль — «123456»).

Организация может иметь набор правил паролей, которые необходимо соблюдать, но, если пользователь не проявляет усердия в обеспечении безопасности пароля, правила паролей не будут иметь значения. Кроме того, правила паролей должны отражать текущую лучшую практику определения паролей. Такие методы можно найти в Руководстве по построению паролей от института SANS [SANS2].

Тесты механизмов аутентификации и авторизации могут включать:

- Атаки методом перебора и по словарю с целью обнаружения паролей пользователей. Первыми шагами могут быть попытки ввести «123456», «111111», дату рождения, имя домашнего животного и т.д.
- Использование отсутствия фильтрации ввода, например, для SQL-инъекции с целью аутентификации без известного логина/пароля.
- Ввод несанкционированного URI (../../ в учетной записи FTP) или URL-адреса (адрес сайта/администратора) для попытки получить доступ к конфиденциальным данным.

Другим примером может быть использование уязвимости в целевой системе (возможно, из-за того, что она не была обновлена), чтобы вызвать непреднамеренное поведение, что обычно приводит к получению контроля над системой и разрешению повышения привилегий.

5.3 Шифрование

5.3.1 Понимание шифрования

Чтобы избежать разглашения конфиденциальных данных, даже если к ним можно получить доступ при хранении или обмене между клиентом и сервером, можно использовать механизм шифрования. Хеширование и добавление случайных данных - это методы, используемые при шифровании.

Шифрование — это процесс кодирования данных (обычного текста) в зашифрованные данные (зашифрованный текст) с использованием криптографического алгоритма и секретов таким образом, что только авторизованные лица имеют право доступа с использованием механизма расшифровки. Секрет является единым и известен только авторизованным пользователям. Цель состоит в том, чтобы иметь достаточно надежное шифрование, и тем самым помешать злоумышленнику, которому, возможно, удалось украсть зашифрованные данные, восстановить обычный текст. Использование криптографических алгоритмов помогает обеспечить конфиденциальность, целостность, доступность конфиденциальных активов и отказ от манипулирования ими.

Криптографические протоколы могут использоваться для защиты информации:

- Хранящихся в системе, например, зашифрованных паролей в базе данных, логического зашифрованного диска, всего зашифрованного жесткого диска
- Во время коммуникаций, например, зашифрованной электронной почты, зашифрованного протокола связи (SSL, TLS)

Основными и хорошо известными используемыми криптографическими протоколами являются:

- Симметричное шифрование: использование общего секретного ключа
- Асимметричное шифрование: использование закрытого и открытого ключа

5.3.2 Тестирование эффективности основных механизмов шифрования

Известно, что некоторые криптографические механизмы слабы, особенно из-за короткого размера секретных ключей или статических ключей. Другие механизмы уязвимы, потому что либо они не реализованы с использованием лучших практик, либо в них присутствуют дефекты кодирования (например, переполнение буфера).

Тесты механизмов шифрования должны включать:

- Тесты уязвимостей проектирования:
  - Проверка того, что при симметричном шифровании используются правильные режимы
  - Проверка того, что размер криптографических ключей не слишком мал (например, с 2015 года ключ RSA менее 2048 бит считается небезопасным)
  - Проверка действительности сертификатов и возможность выдавать предупреждение, если сертификат является самоподписанным (SSL-strip можно использовать, чтобы избежать атак «человек посередине»)
  - Повторная атака (например, атака на протоколы Wired Equivalent Privacy (WEP))
  - Атаки на криптографические протоколы для проверки их уровня надежности [Bittau]
- Тесты уязвимостей конструкции:

- Проверка кода (например, чтобы убедиться, что функция random () по умолчанию не используется для генерации случайных чисел (начальных значений), поскольку алгоритм случайных чисел относительно легко взломать)
- Ввод большого количества случайных данных для использования неожиданного поведения
- Временные атаки (анализ времени, затраченного на выполнение криптографических алгоритмов)
- Анализ мощности (используется для зашифрованных аппаратных устройств)

- Тесты для уязвимостей конфигурации:
  - Оценка конфигурации криптографического протокола (например, конфигурация безопасности транспортного уровня (TLS) на стороне сервера, авторизованные протоколы на стороне клиента, на основе руководства по настройке TLS для администраторов)
  - Порядок шифров TLS на стороне сервера, чтобы узнать, существуют ли средства для понижения или пересмотра используемого шифра
- Тесты для проверки устаревших механизмов шифрования, которые могли стать слабыми и подверженными взлому

5.4 Межсетевые экраны и сетевые зоны

5.4.1 Понимание межсетевых экранов

Согласно [Chapman 2000], «сетевой экран — это компонент или набор компонентов, который ограничивает доступ между защищенной сетью и Интернетом или между другими наборами сетей». Сетевой экран реализует и обеспечивает соблюдение политики безопасности, основанной на определении разрешенных и запрещенных соединений. Сетевой экран может быть на основе хоста (программное обеспечение, работающее на одном хосте, которое отслеживает ввод/вывод приложений) или сетевым (программное обеспечение, которое отслеживает трафик между сетями).

Основная задача сетевого экрана — контролировать трафик между различными доверенными сетевыми зонами, фильтруя данные, проходящие по сети. Таким образом обнаруживается и блокируется вредоносный трафик, поступающий из ненадежной зоны.

Сетевая зона - это идентифицированная подсеть с определенным уровнем доверия:

- Интернет/общедоступная зона, которая считается ненадежной
- Несколько зон безопасности, называемых демилитаризованными зонами или DMZ, с разным уровнем доверия
- Одна или несколько частных/внутренних сетей считаются наиболее надежными

Сетевые зоны являются частью конфигурации сетевого экрана: они используются для определения разрешенных потоков между различными сетями. Весь запрещенный трафик блокируется.

Обычно сетевой экран фильтрует коммуникации на основе:

- Исходные и целевые адреса и протоколы (адреса Ethernet или IP, порты TCP/UDP и т.д.)
- Параметры протокола (фрагментация, TTL и т.д.)
- Размер данных

Сетевые экраны веб-приложений (WAF) также фильтруют коммуникации на основе:

- Пользовательские связи

- Фильтрация данных (например, с использованием описаний шаблонов)

5.4.2 Тестирование эффективности сетевого экрана

Из-за количества протоколов, их различных вариантов и сложности сетей, которые необходимо защитить, трудно эффективно сконфигурировать сетевой экран. Тесты на эффективность сетевого экрана должны включать:

- Сканирование портов для проверки правильности реализации политики безопасности
- Использование неправильно сформированных сетевых пакетов и сетевого ввода большого количества случайных данных для использования неожиданного поведения (например, отказ в обслуживании)
- Фрагментационные атаки для обхода функций фильтрации с целью продолжения атаки за сетевым экраном

Другим примером тестов, нацеленных на сетевые экраны веб-приложений (WAF), является кодирование и сжатие данных или их запутывание, чтобы скрыть вредоносную информацию, которая передает атаку.

5.5 Обнаружение вторжений

5.5.1 Понимание инструментов обнаружения вторжений

С каждым годом количество атак увеличивается. Методы вторжения быстро развиваются, и ни одна система не является на 100% безопасной.

Система обнаружения вторжений (IDS) - это система (автономное устройство или приложение), которая отслеживает действия на разных уровнях (от сети до приложения, 7 уровней модели OSI) для обнаружения нарушений политики безопасности. При обнаружении отклонений от нормального поведения выдаются предупреждения, которые можно проанализировать для дальнейших действий (например, блокировки трафика, виртуального исправления).

Что касается стандартизации IDS, формат обмена данными об обнаружении вторжений инженерной рабочей группы интернета описывает подход к проектированию IDS, основанный на двух моделях безопасности:

- Негативная модель безопасности (обнаружение на основе сигнатур или черного списка): правило гласит: «разрешено все, что явно не запрещено». Обнаружение вторжений основано на списке известных атак или шаблонов.
- Позитивная модель безопасности (обнаружение на основе поведения или обнаружение «белого списка»): правило гласит: «все, что явно не разрешено, запрещено». Обнаружение вторжения основано на спецификации поведения системы, которую необходимо защитить, например, закодированная проверка входных данных в форму, с помощью регулярного выражения. Вторжение обнаруживается, если поведение отклоняется от нормального или ожидаемого поведения системы. Для создания спецификации может использоваться доверенный трафик.

Система обнаружения вторжений отличается от сетевого экрана тем, что сетевой экран смотрит на внешний трафик, чтобы остановить вторжения, тогда как система обнаружения вторжений анализирует подозрительные вторжения и выдает предупреждение, если они подтверждаются.

5.5.2 Тестирование эффективности средств обнаружения вторжений

Обнаружение на основе сценариев легко обойти, поскольку обнаруживаются только известные атаки. Тесты могут включать следующие методы обхода:

- Кодировку символов или изменение данных (например, добавление пробела, конца строки и т. д.)
- Фрагментацию IP, сегментацию TCP
- Шифрование, запутывание данных
- Шифрование URL-адреса

Обнаружение на основе поведения генерирует большое количество ложноположительных и ложноотрицательных результатов. Ложноотрицательный результат - это любое предупреждение, которое должно было сообщить о проблеме, но не сообщило. Ложноотрицательные результаты могут возникать, когда разрабатывается новая атака, о которой система обнаружения вторжений (IDS) не знает, или правило может быть написано таким образом, чтобы обнаружить одни атаки, но пропустить другие. Также следует учитывать точность данного метода обнаружения. Злоумышленник может провести атаку с целью изменения правил системы обнаружения вторжений о выдаче предупреждений при отклонении от нормального поведения, что приведет к изменению спецификации, которая спровоцирует избыточные реагирования и оповещения. Таким образом, этот новый трафик не считается аномальным. Дополнительные тесты должны использовать вредоносный трафик, чтобы добавить новые навязчивые спецификации, рассматриваемые как авторизованный трафик.

Некоторые исходные данные могут быть использованы для определения набора тестов для системы обнаружения вторжений (IDS), таких как «Система обнаружения вторжений для защиты профиля» [PP-IDS] и «Критерии оценки межсетевых экранов веб-приложений» [WAFEC].

5.6 Сканирование вредоносной программы

5.6.1 Понимание инструментов сканирования вредоносных программ

Вредоносный код может воздействовать на серверы и компьютеры конечных пользователей, предоставляя его создателям ожидаемые привилегии и целевые конфиденциальные данные. Вредоносный код помещается в цель с использованием различных средств, таких как электронная почта с вредоносными вложениями, поддельные URL-адреса, выполнение кода на стороне клиента и т. д.

Антивирусное приложение - это программное обеспечение, используемое для анализа, обнаружения и удаления вредоносного кода, полученного из разных источников, с различными целями обнаружения: вредоносное программное обеспечение, фишинг и фарминг.

Основная функция обнаружения, используемая антивирусными программами, - это стратегия, основанная на сигнатурах. Принцип заключается в поиске в базе данных известных шаблонов данных, описывающих подозрительный фрагмент кода. Однако новые вредоносные программы или вредоносные программы, сигнатура которых отсутствует в базе данных, не будут обнаружены и смогут заразить свою жертву. Для борьбы с этой проблемой в антивирусные программы часто встраивается эвристический механизм, позволяющий выявлять незначительные вариации известных вредоносных шаблонов.

5.6.2 Тестирование эффективности инструментов сканирования вредоносных программ

Разработчики вредоносных программ и доступов «с черного хода» используют различные техники защиты своего кода от обратной разработки и обнаружения антивирусными программами. Некоторые из этих методов включают:

- Использование функций системной библиотеки, вредоносными программами (например, FindWindow, который можно использовать для закрытия антивирусного приложения)
- Запутывание строк для затруднения понимания поведения вредоносного кода (например, с помощью шифрования). Примером может служить хранение JavaScript в PDF-документе. Другой способ заключается в использовании сжатия, такого как Ultimate Packer для исполняемых файлов (UPX).
- Динамическую загрузку функций и библиотек (например, для ограничения анализа вредоносного кода)
- Автоматическое обновление приложений (например, Skype Trojan)

Вредоносные программы также могут использовать другие аппаратные ресурсы, такие как графический процессор (GPU), для распаковки вредоносного кода и сохранения его в памяти для выполнения процессором. В этом случае вредоносная программа не может быть проанализирована перед выполнением.

С точки зрения функционального тестирования такой инструмент, как «Eicar» [EICAR] (тестовый файл для проверки работоспособности защиты от вредоносных программ), можно использовать для проверки эффективности защиты от вредоносных программ без разработки реальных вредоносных фрагментов кода.

Важным соображением при внедрении нового или обновления существующего приложения для защиты от вредоносных программ является тестирование реализации на репрезентативной платформе перед ее развертыванием во всей организации. Были случаи, когда антивирусное программное обеспечение ошибочно идентифицировало не зараженные файлы операционной системы как вредоносное ПО и помещало их в карантин, тем самым отключая все вычислительные возможности организации.

5.7 Запутывание данных

5.7.1 Понимание запутывания данных

Запутывание (иногда называемое маскированием) данных - это механизм, позволяющий сделать данные и исходный код непонятными для человека.

Эта техника используется в основном для защиты конфиденциальных данных от:

- Копирования, чтобы обойти механизмы защиты лицензии
- Обратногй разработки, для понимания кода с целью использования уязвимостей

Запутывание данных также может использоваться для того, чтобы сотрудники компании (вспомогательный персонал, функциональные тестировщики и т.д.) могли работать с не конфиденциальными данными, тем самым скрывая конфиденциальные данные от посторонних глаз. Некоторые могут называть запутывание данных «анонимизацией данных», поскольку оно сохраняет анонимность персональных данных человека.

Запутывание также может использоваться для защиты исходного кода от простого копирования и вставки (например, для защиты нового инновационного алгоритма) и повторного использования в будущем после того, как он был реконструирован для его понимания.

Иногда разработчикам необходимо оптимизировать свой код, чтобы сделать его более эффективным. Это может привести к запутыванию исходного кода (например, путем кодирования некоторых частей на языке ассемблера). Некоторые атаки на уровне веб-приложений состоят из внедрения. Чтобы добиться успеха, злоумышленникам необходимо знать структуру веб-сайта и HTML-страниц. Запутывание может помочь в защите конфиденциальных и важных HTML-страниц (например, страниц подключения и администрирования).

Можно использовать несколько методов запутывания, таких как кодирование base64, XORing, случайное переименование функций, переопределение методов, удаление пробелов с возвратом табуляции, перетасовка и т.д. Шифрование также является методом запутывания, но с проблемами, поскольку зашифрованные данные останутся доступными для просмотра тем, у кого есть действительные ключи. которые необходимо:

- Поддерживать в актуальном состоянии
- Следить за истечением срока годности
- Предотвращать несанкционированный доступ неавторизованных лиц.

Примечание: Запутывание данных часто используется злоумышленниками для сокрытия своего вредоносного кода и атак.

5.7.2 Тестирование эффективности подходов к запутыванию данных

Необходим жесткий контроль конфигурации трассируемости между запутанными данными и ключами, используемыми для запутывания, чтобы гарантировать использование правильных версий ключей. В противном случае данные не могут быть расшифрованы для использования.

Поскольку в некоторых тестах могут быть задействованы персональные данные, для целей тестирования может использоваться запутывание данных, чтобы сделать пользовательские данные, используемые в среде системного тестирования, анонимными. Конфиденциальные данные, такие как информация о пользователе, используемая информационной системой здравоохранения, не должны разглашаться тестировщикам. Тесты могут включать:

- Атаки методом перебора или по словарю с целью получения простых данных из запутанных данных

Тесты для проверки запутывания кода могут включать:

- Обратная разработка байт-кода Java (например, регенерация исходного кода Java с помощью Java Decompiler) или программ .NET (например, извлечение исходного кода .NET с помощью .NET Reflector),
- Атаки методом перебора, поскольку некоторые механизмы запутывания уязвимы (например, при использовании не исключающего ИЛИ [Chopitea]).

Теоретически код не может защитить себя от расшифровки запутывания, потому что всегда можно использовать отладку. Несмотря на то, что существуют инструменты защиты кода от декомпиляции, все еще существуют риски и ограничения в защите конфиденциальной информации, представленной кодом.

5.8 Повышение квалификации

5.8.1 Важность обучения по вопросам безопасности

Люди часто являются самым слабым звеном в общей картине безопасности. Следовательно, необходимо последовательное и непрерывное обучение, чтобы напоминать людям о важности соблюдения установленных политик безопасности и подчеркивать необходимость этих политик. Это обучение должно проводиться на протяжении всего жизненного цикла программного обеспечения и обновляться по мере добавления новых политик и возникновения новых угроз. Обучение должно охватывать выявление атак социальной инженерии и внутренних угроз.

5.8.2 Как проверить эффективность обучения безопасности

Например, в программе обучения безопасности может быть рассмотрена важность наличия надежных пользовательских паролей, которые хранятся в тайне.

Тесты могут включать:

- Социальную инженерию для попытки заставить пользователя раскрыть свой пароль во время телефонного разговора с подставным сотрудником технической поддержки
- Осматривание столов в поисках стикеров с паролями (особенно под клавиатурой)
- Запуск инструментов аудита паролей для выявления слабых паролей. Один из рисков этого типа инструментов заключается в том, что пароли могут быть видны человеку, проводящему тест.

Другой пример: разработчик не установил редактирование на уровне поля, чтобы предотвратить ввод SQL-команд в поле ввода данных. Из-за этой ошибки тестировщик безопасности может ввести SQL-команду и просмотреть содержимое базы данных клиента. Это указывает на то, что разработчику необходимо дополнительное обучение методам безопасного кодирования. Также было бы неплохо изучить практику кодирования других разработчиков, чтобы понять, является ли эта практика широко распространенной и нужна ли общая инициатива по улучшению процесса.

Третьим примером может быть попытка тестировщика получить несанкционированный физический доступ в офис и просмотреть документы, которые были оставлены открытыми.

6. Человеческий фактор в тестировании безопасности - 105 мин.

Ключевые слова
злоумышленник, ботнет, компьютерная криминалистика, взломщик, сбор информации, новички

Цели обучения для человеческого фактора в тестировании безопасности

6.1 Понимание злоумышленников

AS-6.1.1 (K2) Объяснить, как поведение человека может привести к рискам безопасности и как оно влияет на эффективность тестирования безопасности

AS-6.1.2 (K3) Для заданного сценария продемонстрировать способность определять способы, с помощью которых злоумышленник может получить ключевую информацию о цели, и применить меры для защиты окружения

AS-6.1.3 (K2) Объяснить общие мотивы и источники проведения атак на компьютерные системы

AS-6.1.4 (K4) Проанализировать сценарий атаки (выполненная и обнаруженная атака) и определить возможные источники и мотивы атаки

6.2 Социальная инженерия

AS-6.2.1 (K2) Объяснить, как средства защиты безопасности могут быть скомпрометированы с помощью социальной инженерии

6.3 Осведомленность о безопасности

AS-6.3.1 (K2) Понимать важность осведомленности о безопасности всей организации

AS-6.3.2 (K3) Учитывая определенные результаты тестирования, применить соответствующие действия для повышения осведомленности о безопасности

6.1 Понимание злоумышленников

В контексте информационной безопасности человек является одновременно самой большой угрозой и самым слабым звеном в защите.

Атаки на систему безопасности совершаются людьми с различными навыками и мотивациями. Кроме того, люди являются самыми большими факторами, способствующими большинству атак на безопасность. Простого понимания и внедрения технологии безопасности недостаточно для защиты от атак. Важно важно понимать образ мыслей, мотивы и методы злоумышленников и знать о человеческих слабостях в линии защиты.

6.1.1 Влияние человеческого поведения на риски безопасности

Ключевым этапом любой атаки является этап сбора информации (рекогносцировка), когда злоумышленник пытается найти и собрать информацию о цели. Вся информация, которая публикуется, иногда неосознанно, об организации, используемых системах и т.д. и хранится в Интернете, будет найдена и может/будет использована в атаке. Это не вопрос «если», а вопрос «когда». Помимо информации, официально публикуемой организацией, сотрудники также публикуют информацию о компании в своих социальных сетях. Количество и содержание этой информации постоянно меняется, часто представляя злоумышленникам некоторую ключевую информацию.

Злоумышленники не используют политику безопасности или предопределенные процедуры при атаке системы. На основе информации, которую они могут собрать, они определяют свою стратегию. Они будут обновлять свою базу знаний для каждой атаки, выполняя выборочный поиск и «посещая» уже известные IP-адреса.

Когда формулируется политика безопасности компании, она обычно основывается на ситуации и имеющихся фактах. Иногда это не включает всю общедоступную информацию, и, даже если это так, эта информация может измениться. Тесты безопасности, которые были актуальны на момент их создания, могут не обеспечить адекватного покрытия при изменении опубликованной информации.

6.1.2 Понимание мышления злоумышленника

Во время сбора информации или поиска следов злоумышленник пытается найти любую информацию о цели, используя пассивные и/или активные средства. Большинство ИТ-оборудования, работающего в сетях общего пользования, оставляет след в этих сетях. Эти следы могут и будут найдены. Google (включая Google Earth и Street View) или другие поисковые системы, Shodan [Web-5], Facebook, LinkedIn и другие социальные сети являются первыми источниками, используемыми для поиска информации о цели. IP-адреса, веб-страницы, номера телефонов, имена и структуры адресов электронной почты, операционная система и приложения - все это может дать полезную для злоумышленника информацию.

Возможное использование поисковой системы Google заключается в поиске конкретной информации о цели. Сотни запросов можно найти в базе данных взлома от Google [Web-4]. Shodan [Web-5] - это еще один инструмент, который используется для поиска конкретной информации, например, о том, какие компании в определенной области используют сервер Apache с уязвимой версией.

Большая часть этой информации может быть найдена пассивно, без фактического подключения к целевой системе. Другие используемые инструменты включают:

- Whois (сетевой протокол прикладного уровня, базирующийся на протоколе TCP) [Web-13]
- Базу данных Ripe (European IP Networks) [Web-12]
- DNS-поиск [Web-25]

С помощью методов активной разведки злоумышленник использует инструменты обнаружения хостов, открытых портов, операционной системы и приложений путем прикосновения к системе. Используемые здесь методы и инструменты включают:

- Тестовые опросы системы - Fping [Web-15], Hping [Web-19]
- Сканирование TCP/UDP – Nmap [Web-20], Zenmap [Web-21]
- Обнаружение операционной системы – Nmap [Web-20], Xprobe2 [Web-22]
- Служба «отпечатков пальцев» (Nmap также может определять тип и версию службы, работающей на обнаруженном открытом порту. Это делается путем сравнения «отпечатков пальцев» обнаруженной службы с собственной базой данных отпечатков пальцев Nmap)

Поскольку взлом системы запрещен законом в большинстве, если не во всех странах, взломщик постарается после этого уничтожить все доказательства взлома. Другими причинами уничтожения улик являются продление пребывания, продолжение использования системы в будущем и использование скомпрометированной системы или сети систем (ботнетов) для атаки на другие системы. Злоумышленник может использовать для этого такие инструменты, как NetCat [Web-14], или использовать веб-сайты, такие как IP TRacer [Web-7], а также туннелировать или изменять файлы журналов.

Другие методы и инструменты, используемые для сокрытия улик, включают средства сокрытия [Web-16], вредоносные программы, скрытно действующие в зараженной системе и обладающие специальными средствами, затрудняющими их обнаружение системами безопасности (руткиты) и потоковую передачу файлов. Все или большинство упомянутых здесь инструментов доступны через Интернет. Загрузка последней версии Kali Linux [Web-17] и поиск на сайте OWASP [OWASP1] дадут доступ ко многим из этих инструментов.

6.1.3 Общие мотивы и источники атак на компьютерные системы

Многие атаки и взломы информационных систем происходят изнутри организации. Злоумышленники (внутренние взломщики или внутренние угрозы) попытаются скомпрометировать систему, будучи авторизованными пользователями сети. В большинстве случаев мотивацией является месть, но последние тенденции показывают рост экономического шпионажа или воровства.

Внешние взломщики ответственны за меньшинство атак. Любопытство было одним из первых стимулов взлома информационных систем, и до сих пор остается таковым. Обладание некоторой информацией от крупных компаний или организаций и знание того, что другие этого не знают, является еще одним мотивом (престиж). Среди других мотиваторов - дурная слава или известность, вызов, скука и месть, причем последняя считается наиболее опасной формой (высшая мотивация).

Злоумышленников часто классифицируют по их мотивации и способностям. В нижней части спектра злоумышленников находятся "новички", которые просто выполняют атаки, созданные другими, в то время как в верхней части спектра находятся профессиональные (правительственные, ориентированные на взлом) организации и отдельные лица. Деятельность

по взлому - это атака на системы, основанная в основном на политических, а также экономических или демографических мотивах.

Мотивация может варьироваться от игры ради развлечения до полного краха системы или организации по любой причине (например, политической, идеологической, экономической, военной, коммерческой, террористической).

Способности к взлому варьируются от людей, имеющих некоторые системные и сетевые знания, работающих с простым домашним компьютером, до высококвалифицированных и образованных профессионалов, имеющих доступ к лабораториям, прокси-сетям и всему другому необходимому техническому оборудованию. Представление о потенциальных злоумышленниках поможет организации внедрить необходимую защиту и даст рекомендации по стратегии тестирования безопасности.

6.1.4 Понимание сценариев и мотивов атак

Инцидент безопасности определяется как связанное с безопасностью системное событие, при котором политика безопасности системы не соблюдается или иным образом нарушается. [RFC2828]

Выяснение того, что произошло и кто несет ответственность за инцидент, связанный с безопасностью, является целью дисциплины компьютерной криминалистики [Web-8], где основное внимание уделяется поиску цифровых доказательств атаки.

Процесс сбора доказательств включает три этапа:
1. Получение и аутентификация цифровых доказательств
2. Анализ
3. Отчет

6.1.4.1 Получение и аутентификация

Процесс управления инцидентами в организации должен восстановить систему в исходное состояние (до атаки) после сбора и хранения доказательств. Он начинается, когда системный администратор получает предупреждение от системы обнаружения вторжений (IDS) или других средств мониторинга. Другими типичными симптомами инцидентов безопасности являются:
- Подозрительные записи в журнале
- Необъяснимые учетные записи пользователей
- Измененные файлы/папки
- Запущенные необычные службы
- Необычное поведение системы
- Неудачные попытки входа в систему

После получения предупреждения процесс выполнения выглядит следующим образом:
1. Сделать снимок или копию исследуемой системы, чтобы собрать все необходимые доказательства.
2. После проверки подлинности доказательства (это подлинная и полная копия) создать копию и хранить ее в безопасном месте.
3. Провести анализ доказательств.
4. После завершения процесса судебной экспертизы устранить причину инцидента (искоренение).
5. Вернуть систему в нормальное состояние (восстановление).

На этих этапах любые уязвимости устраняются с помощью исправлений или установки нового программного обеспечения. В отчете о результатах следует описать процесс, которому следовали, а также инструменты, использованные в ходе этого процесса.

6.1.4.2 Анализ

После попыток взлома может оказаться возможным найти источник атак, изучив файлы системного журнала и активные сетевые подключения. Важно сделать копии всех файлов журнала и зафиксировать информацию о состоянии процесса. Во время активной атаки может иметь смысл собрать системную информацию, относящуюся к злоумышленнику (злоумышленникам), прежде чем блокировать их.

Любая атака через Интернет может быть прослежена до исходного IP-адреса, независимо от того, использовалась ли при этом электронная почта или Интернет-соединение. Это лишь вопрос времени, денег и усилий, а также оценки соответствующих затрат. Большинство злоумышленников используют прокси-серверы или цепочки прокси-серверов, сеть Tor [Web-9] или другие бесплатные анонимные варианты, чтобы скрыть свой реальный IP-адрес. Чем больше прокси-серверов используют злоумышленники, тем больше времени требуется для отслеживания исходного адреса. Местные законы, влияющие на физическое расположение прокси-серверов, также могут препятствовать этому расследованию.

Обнаружение злоумышленников и отслеживание IP-адреса до его источника можно выполнить с помощью таких инструментов, как Netstat (Windows) [Web-10], Tracert [Web-11] и веб-сайт IP Tracer [Web-7]. Netstat показывает подключения к компьютеру, порты и запущенные службы. Этот инструмент можно использовать для поиска любого странного или неизвестного IP-адреса или номера порта.
Примечание: В операционной системе Microsoft Windows также есть утилита tracert (в Linux и OS/X это «traceroute»), но упомянутые выше веб-службы независимы от этих утилит ввода-вывода.

В заголовке электронного письма, содержащего вирусы, может быть указан IP-адрес интернет-провайдера, отправившего письмо. Однако для большинства веб-почты (Gmail, Yahoo mail, Outlook.com) это IP-адрес провайдера веб-почты. Чтобы найти настоящий IP-адрес, необходимо посмотреть значение X-Originating-IP. Использование баз данных Whois [Web-13] приведет к получению подробной информации, которую можно использовать для связи с организацией-провайдером для продолжения расследования. Следует отметить, что электронная почта может отправляться с частных серверов и открытых ретрансляционных почтовых серверов. В этом случае может быть очень трудно определить фактический источник электронного сообщения.

Расследование атак с использованием ботнета затруднено. Для атакующего нет необходимости иметь онлайн-подключение к бот-серверу или бот-клиентам, поэтому отслеживание очень сложно или практически невозможно. В этом случае исследование клиентов может привести к серверу, но для того, чтобы установить истинный источник атаки, необходимо иметь доступ к серверу. Владельцы этих серверов могут не знать, что их машины являются частью ботнета.

6.1.4.3 Отчет

Отчетность об уязвимостях безопасности описана в главе 7.

6.2 Социальная инженерия

Мы можем внедрить все технические средства защиты, которые только можем придумать, чтобы защитить цифровые активы от внешнего мира, но в конечном итоге все сводится к тому, что сотрудникам (пользователям и администраторам) необходимо иметь доступ к этим активам для

выполнения своей работы. Им может потребоваться использовать аутентификацию для получения доступа со своих настольных компьютеров, ноутбуков, смартфонов, планшетов или других средств. Любая физическая защита безопасности для защиты доступа к офису и офисному оборудованию бессмысленна, если безопасность рабочего места ИТ-менеджера в его доме может быть легко скомпрометирована.

Именно человек и его (или ее) поведение представляют наибольшую угрозу безопасности. Если люди небрежно обращаются с конфиденциальной информацией, это оставляет слишком много следов в безопасных местах и слишком громко транслирует эту информацию (как устно, так и в электронном виде) в общественных местах.

Социальная инженерия — это искусство эксплуатации человека с использованием его общего поведения в качестве вектора атаки. Как социальные существа люди готовы доверять незнакомцам и помогать им. Это создает уязвимость для атаки. Манипулируя, влияя и убеждая полезных людей, злоумышленник попытается получить доступ, данные авторизации или другую конфиденциальную информацию.

Действия по использованию уязвимостей в программном обеспечении (эксплойты) могут быть выполнены путем прямого взаимодействия с человеком или с использованием компьютерного/сетевого оборудования.

Прямое взаимодействие с людьми может осуществляться в том числе и лично:

- Тайный вход или следование (человек, не имеющий надлежащей аутентификации, следует за сотрудником в запретную зону)
- Прослушивание (прослушивание чужих личных разговоров без их ведома)
- Подсматривание через плечо (заглядывание через чье-то плечо без его ведома, когда он выполняет задачи на компьютере или пишет)
- Использование телефона (например, получение пароля от ничего не подозревающего пользователя, выступая в роли другого человека, например, менеджера или сотрудника технической поддержки)

Компьютерная социальная инженерия может осуществляться с помощью:

- Отправки электронных писем, зараженных вредоносными программами.
- Использования чата или приложений для обмена мгновенными сообщениями. Используя чат и приложения для обмена мгновенными сообщениями, любой анонимный пользователь может общаться в чате с другим человеком в любой точке мира, не зная его подлинной личности. Кроме того, данные через мессенджеры можно легко перехватить.
- Использования всплывающих экранов. Например, на экране компьютера пользователя может появиться окно с сообщением пользователю о потере сетевого подключения. В этот момент пользователю предлагается повторно ввести имя пользователя и пароль. Затем программа, заранее установленная злоумышленником, может передать эту информацию на удаленный сайт.
- Отправки спама по электронной почте. Спам-письма изобилуют мошенническими предложениями и ссылками. Переход по этим ссылкам может привести к установке вредоносного ПО, которое может подвергнуть опасности всю сеть.
- Убеждения людей посетить зараженные (манипулируемые) веб-сайты. Эти попытки фишинга могут быть широко распространены или носить сугубо индивидуальный характер (целевой фишинг).

Единой защиты от социальной инженерии не существует. Можно применять средства защиты, чтобы контролировать ущерб (например, обеспечить наименьший уровень привилегий, который

все еще позволяет кому-то выполнять свою работу, разделение обязанностей, ротация обязанностей), но главная защита - это обучение и осведомленность на всех уровнях организации.

6.3 Осведомленность о безопасности

6.3.1 Важность осведомленности о безопасности

Модель угроз постоянно меняется, как упоминалось в других главах этой программы обучения. Сети развиваются, внедряются новые приложения, вводятся в эксплуатацию новые интерфейсы, внедряются и обнаруживаются новые уязвимости.

Помимо этих технических аспектов, существует человеческий фактор. Риски, которые когда-то были выявлены, но не стали проблемами, легко забываются, и меры предосторожности отменяются. Это открывает широкие возможности как для атак злоумышленников, так и для атак с применением социальной инженерии. Для того чтобы администраторы безопасности и все сотрудники были бдительны и информированы об изменениях модели угроз, необходимо регулярно проводить обучения по повышению осведомленности в вопросах безопасности. Это обучение может быть ориентировано на различные группы пользователей: разработчиков, операторов, руководство и рядовых пользователей.

6.3.2 Повышение осведомленности о безопасности

Важно сохранять мышление, ориентированное на осведомленности о безопасности. Помимо общей информации о средствах защиты в компании, обучение должно содержать реальные примеры из практики, обнаруженные во время тестирования безопасности или в реальных инцидентах. Основываясь на этих примерах, должно быть легче обсуждать любые средства защиты или изменения, которые необходимо внедрить в организации.

План этого раздела ознакомительного обучения должен включать ответы на следующие вопросы:

- Как они (мы) это сделали?
- Каковы были последствия для бизнеса?
- Каковы были затраты на расследование и обработку инцидента?
- Каковы были затраты на устранение проблемы?
- Как можно было избежать инцидента?
- Какие изменения будут реализованы?

7. Оценка тестов безопасности и отчетность - 70 мин.

Ключевые слова
критерии приемки, вектор атаки, сводная таблица, критерии выхода

Цели обучения для оценки тестов безопасности и отчетности

7.1 Оценка теста безопасности

AS-7.1.1 (K2) Понимать необходимость пересмотра ожиданий безопасности и критериев приемки по мере изменения масштаба и целей проекта

7.2 Отчетность по тестированию безопасности

AS-7.2.1 (K2) Понимать важность обеспечения конфиденциальности и безопасности результатов тестирования безопасности

AS-7.2.2 (K2) Понимать необходимость создания надлежащих средств контроля и механизмов сбора данных для своевременного, точного и четкого представления исходных данных для отчетов о состоянии тестирования безопасности (например, панель мониторинга тестирования безопасности)

AS-7.2.3 (K4) Проанализировать полученный промежуточный отчет о состоянии тестирования безопасности для определения уровня точности, понятности и соответствия требованиям заинтересованных сторон

7.1 Оценка теста безопасности

Измерение результатов тестирования безопасности и оценка состояния в отношении ожиданий безопасности, критериев выхода и/или критериев приемки необходимы для определения выполнения критериев завершения тестирования.

Трудно знать все риски безопасности в начале проекта. Кроме того, ожидания заинтересованных сторон и пользователей иногда меняются в отношении необходимого уровня безопасности. Например, осведомленность о новой угрозе может привести к тому, что заинтересованным сторонам потребуется более высокий уровень безопасности, чем предполагалось изначально. Это одна из причин, по которой оценки рисков безопасности необходимо пересматривать на протяжении всего проекта, а результаты включать в планирование и выполнение тестов безопасности.

7.2 Отчетность по тестированию безопасности

7.2.1 Конфиденциальность результатов тестирования безопасности

Как правило, средний тестировщик знает больше о объекте тестирования после завершения тестирования по сравнению с большинством разработчиков или проектировщиков. При тщательном тестировании можно обнаружить наиболее важные слабые и сильные стороны системы. То же самое относится и к тестированию безопасности.

Тестируя реализацию безопасности, можно найти скрытые дыры и уязвимости системы безопасности. Их разница заключается в возможных негативных последствиях сообщения об этих уязвимостях людям, не являющимся непосредственными заинтересованными сторонами. Общепринятая практика заключается в том, что информация должна быть доступна только тем, кому она необходима. Это особенно относится к результатам проверки безопасности; осторожность при обмене информацией такого типа считается хорошей практикой.

7.2.2 Создание надлежащих механизмов контроля и сбора данных для отчетности о состоянии тестирования безопасности

Воздействие и последствия уязвимости в системе безопасности обычно оцениваются как более чувствительные по сравнению с обычными дефектами. Это приводит к необходимости более точного и четкого предоставления информации о характере дефекта и предполагаемых рисках. В большинстве проектов дефекты безопасности классифицируются с более высокой степенью серьезности, чем сопоставимые функциональные дефекты.

Последнее подразумевает, что руководство уделяет больше внимания дефектам безопасности, их рискам и возможным решениям. Отчеты о дефектах безопасности должны тщательно оценивать влияние обнаруженной проблемы, точность результатов тестирования и должны быть доступны четко определенным и своевременным образом. Хорошей практикой является обсуждение с руководством того, как и когда они хотели бы получить доступ к отчетам о дефектах безопасности.

7.2.3 Анализ промежуточных отчетов о состоянии тестирования безопасности

Отчеты о тестировании безопасности могут составляться в течение всего процесса тестирования безопасности или только в конце тестирования безопасности (например, в конце тестирования безопасности системы или в конце тестирования безопасности, выполняемого как часть приемочного тестирования). Раннее составление отчетов о тестировании безопасности приветствуется, поскольку это дает больше времени для устранения уязвимостей безопасности. Если процесс тестирования безопасности соответствует процессу, описанному в данной программе обучения, команда тестирования может обнаружить уязвимости и документировать наблюдения в ходе всех мероприятий по тестированию.

Структура отчета о тестировании безопасности должна содержать следующие разделы:

1. Идентификатор отчета
2. Краткое содержание
   a. Основные положения
   b. Основные результаты
3. Отклонения
   a. Соблюдение процесса тестирования
   b. Любые отклонения от запланированного процесса тестирования
   c. Используемые методы и инструменты (конфигурации, политики)
4. Всесторонняя оценка
   a. Оценка тестового покрытия на основе критериев, указанных в плане тестирования
   b. Объяснение любых элементов или функций, которые не были протестированы
5. Краткое изложение результатов
   a. Подведение итогов тестирования безопасности
   b. Список всех устраненных уязвимостей безопасности и способы их устранения
   c. Список всех не устраненных уязвимостей
6. Оценка
   a. Оценка наблюдаемых результатов тестирования и их статуса на основе критериев выхода
   b. Выявленные риски (классификации) и влияние не устраненных уязвимостей безопасности
7. Краткое описание деятельности
8. Утверждение отчета

Эффективность отчетов о тестировании безопасности зависит от:

- Сроков составления отчета
- Содержания отчета
- Получателей отчета
- Настроек содержания в соответствии с потребностями получателей в информации

Для удовлетворения потребностей различных заинтересованных сторон может потребоваться несколько отчетов. Например, содержание отчета для высшего руководства не будет совпадать с содержанием отчета для системного архитектора.

8. Инструменты тестирования безопасности - 55 мин.

Ключевые слова

нет

Цели обучения для инструментов тестирования безопасности

8.1 Типы и цели инструментов тестирования безопасности

AS-8.1.1 (K2) Объяснить роль инструментов статического и динамического анализа в тестировании безопасности

8.2 Выбор инструмента

AS-8.2.1 (K4) Анализировать и документировать потребности в тестировании безопасности, которые должны быть удовлетворены одним или несколькими инструментами

AS-8.2.2 (K2) Понимать проблемы, связанные с инструментами с открытым исходным кодом

AS-8.2.3 (K2) Понимать необходимость оценки возможностей поставщика по частому обновлению инструментов, чтобы быть в курсе угроз безопасности

8.1 Типы и цели инструментов тестирования безопасности

Действия по использованию уязвимостей в программном обеспечении (эксплойты), разработанные сообществом взломщиков, привели к разработке инструментов тестирования безопасности для защиты от этих угроз. Даже на ранних этапах деятельности по взлому (таких как взлом паролей) те, кто их использует, изобретали, создавали и улучшали простые инструменты. Инструменты, доказавшие свою эффективность, были распространены в сообществе взломщиков и в дальнейшем совершенствовались и совершенствовались. Сначала эти инструменты были разработаны для специальных задач и окружений. Удобство использования не было проблемой, поскольку почти все пользователи имели техническое образование. В конце концов, некоторые из инструментов взлома стали основой легальных инструментов тестирования безопасности, используемых администраторами и тестировщиками информационной безопасности.

Например, «Джон Потрошитель» был ранним инструментом взлома паролей с открытым исходным кодом, первоначально использовавшимся взломщиками для угадывания (взлома) паролей и получения доступа к сетям или приложениям Unix. Сегодня этот инструмент был усовершенствован и используется в законных целях для обнаружения слабых паролей Unix. [Web-26]

По мере того, как основные поставщики средств тестирования и разработки программного обеспечения, а также поставщики специализированных инструментов начали разрабатывать инструменты тестирования безопасности, многие из этих инструментов получили более широкие функциональные возможности и улучшенное удобство использования. Однако эта широкая функциональность привела к более сложным конфигурациям инструментов и проблемам реализации.

В то же время, когда появлялись ранние инструменты безопасности, первые версии интегрированных сред, таких как Nessus, Metasploit и другие, разрабатывались как инструменты с открытым исходным кодом, предлагающие улучшенную и расширенную функциональность, а в некоторых случаях также простой в освоении графический интерфейс.

Сегодня количество доступных инструментов тестирования безопасности огромно. Практически для любой среды или задачи можно найти специальный инструмент тестирования, как с открытым исходным кодом, так и с лицензией. Проблема со всеми этими инструментами заключается в том, что большинство из них представляют интеллектуальные системы, выполняющие нестандартизированные тесты. Все разработчики этих систем более или менее одинаково смотрят на то, как тестировать средства защиты или тестировать наличие уязвимостей. Однако эти инструменты могут использовать разные тестовые данные, разные реализации тестов и разные интерпретации результатов.

Инструменты тестирования безопасности могут использоваться для автоматизации оценки средств защиты. Инструменты тестирования безопасности также могут использоваться для обнаружения известных типов уязвимостей. Понимая, что один и тот же тип защиты или уязвимости может быть реализован по-разному, выбор и использование инструментов тестирования безопасности является сложной задачей для тестировщика безопасности, поскольку инструменты различаются по способу поиска уязвимостей и проверки защиты.

Консорциум по безопасности веб-приложений [Web-18] и веб-сайты OWASP [OWASP1] предлагают списки инструментов, разбитых по категориям. Инфраструктура тестирования на

проникновение Backtrack [Web-23] (или Kali Linux [Web-17]) представляет другие способы классификации инструментов тестирования безопасности.

Количество коммерческих инструментов безопасности довольно ограничено по сравнению с количеством инструментов с открытым исходным кодом. На момент разработки данной программы обучения (2016 год) нам удалось найти лишь ограниченное количество ресурсов, представляющих более или менее полный обзор надежных и заслуживающих доверия инструментов безопасности с открытым исходным кодом. Один из списков инструментов безопасности можно найти на сайте https://sectools.org [Web-24]. Ожидается, что продвинутый тестировщик безопасности будет вести свой собственный список доступных инструментов и обновлять его по мере изменения рынка инструментов.

Инструменты статического и динамического анализа полезны при тестировании безопасности. Преимущество статического тестирования заключается в том, что его можно проводить на самых ранних этапах жизненного цикла разработки. Инструменты статического анализа доступны для большинства языков программного обеспечения и обычно имеют возможность сообщать об аспектах безопасности.

Разница между инструментами динамического и статического тестирования в контексте тестирования безопасности иногда немного запутана по сравнению с другими видами тестирования. Определение статического тестирования связано с выполнением тестовых действий, когда тестируемая система или объект не находится в рабочем режиме. Нередко инструменты динамического тестирования безопасности исследуют систему, а не тестируемое приложение. С этой точки зрения такие инструменты динамического тестирования используются как разновидность инструментов статического тестирования. Например, инструмент динамического тестирования безопасности может выполнять статическое сканирование базы данных. Конечно, если в качестве объекта тестирования рассматривается вся система, то такие инструменты действительно являются инструментами динамического тестирования.

8.2 Выбор инструмента

8.2.1 Анализ и документирование потребностей в тестировании безопасности

Помимо прочего, основой тестирования безопасности могут стать следующие документы:

- Политика безопасности организации
- Политика организации в области тестирования
- Результаты анализа угроз и рисков для реальной системы/проекта
- Требования и другие спецификации системы
- Архитектура и проектирование системы
- Стратегия тестирования безопасности
- Тестируемая система или приложение
- Известные угрозы безопасности, эксплойты и уязвимости
- Профили пользователей

Все это и многое другое может предоставить информацию об угрозах и уязвимостях, которые могут быть использованы. В требованиях и проектной документации должно быть указано, как защищаются данные или информация. Это приведет к обзору:

- Интерфейсов, подлежащих тестированию (включая графический интерфейс)
- Протоколов и стандартов, подлежащих верификации

- Руководства по веб-кодированию, которое способствует использованию безопасных методов кодирования
- Конфигурации компонентов системы, подлежащих верификации (защите)

Необходимо определить, будет ли тестирование безопасности частью деятельности по разработке или деятельности по обслуживанию/эксплуатации. Вся эта информация приведет к требованиям к набору инструментов тестирования безопасности.

8.2.2 Проблемы с инструментами с открытым исходным кодом

Полное обсуждение проблем, которые могут возникнуть при использовании инструментов с открытым исходным кодом, см. в [ISTQB_ATM_SYL].

Как уже упоминалось, многие инструменты тестирования безопасности можно найти в открытых источниках. Эти инструменты распространяются и могут использоваться под различными лицензиями, которые разрешают бесплатное использование и модификацию исходного кода. Не все компании или проекты могут рассматривать использование инструментов с открытым исходным кодом в своих процессах разработки. Из-за проблем с соблюдением нормативных требований организации могут быть вынуждены использовать только коммерческие или иным образом сертифицированные инструменты.

Существует множество преимуществ и недостатков, связанных с инструментами под этими лицензиями. Во многих случаях инструменты с открытым исходным кодом можно получить бесплатно, но организации может потребоваться наличие технического потенциала для поддержки и конкретной настройки. Если таких возможностей нет, то их получение от разработчика программного обеспечения может обойтись недешево. Руководство администратора и руководство пользователя, если таковые имеются, в основном написаны с учетом специфической (технической) аудитории и чаще всего не описывают и не охватывают все функциональные возможности инструмента. Медиа-каналы, такие как YouTube, в последнее время являются дополнительным источником информации об использовании этих инструментов.

При расчете возврата инвестиций (ROI) для любого инструмента с открытым исходным кодом необходимо учитывать следующие аспекты:

- Ограниченную область применения инструмента (в большинстве случаев дополнительные или другие функциональные возможности не предлагаются)
- Время на обучение администрированию, настройке и использованию инструмента
- Время для инвестиций в форумы и группы пользователей в течение жизненного цикла
- Время, необходимое для обновления и модернизации (и внутренняя политика в отношении обновлений)
- Будущее направление развития инструмента (некоторые инструменты могут исчезнуть или стать коммерческими)
- Уровень отклика в сообществе поддержки для инструмента

Для большинства предприятий или проектов количество лицензий, необходимых для инструментов тестирования безопасности, ограничено одной или несколькими. Только крупные компании будут рассматривать возможность использования большего количества лицензий. Количество лицензий в основном будет основано на общей сумме функциональных областей, предоставляемых инструментом (например, веб-приложения, веб-сервисы, анализ кода, другие), и предполагаемой частоте, времени использования этих услуг и количестве тестировщиков безопасности, использующих инструмент.

8.2.3 Оценка возможностей поставщика инструментов

Если инструмент приобретается у поставщика, этот поставщик должен предложить ряд услуг, позволяющих запустить службу тестирования безопасности и расширить ее до необходимого уровня внутренней поддержки.

Для оценки возможностей поставщика можно использовать следующие характеристики:

- Типы предлагаемых лицензий (фиксированные/десктопные/плавающие/токеновые)
- Варианты масштабирования лицензии (по функциональной области, количество лицензий)
- Средства службы поддержки (часы работы службы поддержки)
- Форум/сообщество пользователей
- Частота обновления
- Руководства по администрированию и использованию
- Контракты на поддержку и техническое обслуживание

9. Стандарты и отраслевые тенденции - 40 мин.

Ключевые слова

стандарт, основанный на консенсусе

Цели обучения для стандартов и отраслевых тенденций

9.1 Понимание стандартов тестирования безопасности

9.1.1 (K2) Понимать преимущества использования стандартов тестирования безопасности и где их найти

9.1.2 (K2) Понимать разницу в применимости стандартов в нормативных и договорных ситуациях

9.2 Применение стандартов безопасности

9.2.1 (K2) Понимать разницу между обязательными (нормативными) и факультативными (информативными) положениями в любом стандарте

9.3 Тенденции отрасли

9.3.1 (K2) Понимать, где можно узнать об отраслевых тенденциях в области информационной безопасности

9.1 Понимание стандартов тестирования безопасности

Стандарты различных типов обеспечивают прозрачность профессионального консенсуса или нормативных обязательств. Стандарт, основанный на консенсусе, представляет взвешенное мнение компетентной группы экспертов и предоставляется для добровольного использования (полностью или частично) в договорных соглашениях между поставщиками и клиентами. Существуют другие типы так называемых стандартов, которые возникают на основе более неформальных или самоопределяющихся групп и могут быть специфичными для конкретного поставщика.

В регулируемых отраслях (включая медицину, финансовый сектор, транспорт и энергетику) государственные органы могут требовать соблюдения собственных нормативных актов или интерпретации других добровольных стандартов.

9.1.1 Преимущества использования стандартов тестирования безопасности

Стандарты, как правило, обеспечивают руководство и последовательность действий в выполнении задачи. Как правило, стандарты разрабатываются экспертами предметной области на основе консенсуса в отношении эффективных практик. Ниже приведены преимущества использования стандартов тестирования безопасности:

- Они определяют базис тестирования безопасности, устраняя необходимость начинать с «чистого листа».
- В них описываются эффективные средства защиты и способы проверки наиболее распространенных атак безопасности.
- Стандарты могут быть адаптированы для удовлетворения потребностей проекта или организации.
- Должная осмотрительность при тестировании безопасности может быть продемонстрирована соблюдением общепризнанных стандартов тестирования безопасности.

9.1.2 Применимость стандартов в нормативных и договорных ситуациях

В регулируемой деятельности все стороны должны быть осведомлены о своих обязательствах по соблюдению установленных стандартов, поскольку невыполнение этого требования может задержать или помешать утверждению разрабатываемого продукта и, в крайних случаях, привести к финансовым или уголовным санкциям.

В договорных ситуациях стандарты обеспечивают разумную и удобную основу согласования требований к проекту и продукту; они обеспечивают отправную точку вместо того, чтобы стороны начинали с нуля. Стандарты, основанные на консенсусе, позволяют распространять и внедрять или адаптировать передовую практику к конкретной ситуации.

Если стандарты не навязаны в одностороннем порядке регулирующим органом или не закреплены в договоре, не подлежащем обсуждению, они могут использоваться в качестве базиса для соглашения, достигаемого путем переговоров, или самостоятельно устанавливаться при выполнении собственной работы. Если контракт заключен на основании требования или соглашения о соблюдении конкретных стандартов, то организация обязана строго следовать этим стандартам и документировать любые отклонения.

9.1.3 Выбор стандартов безопасности

Конечно, не все стандарты безопасности применимы ко всем ситуациям. Организация несет

ответственность за выбор наиболее подходящего стандарта (стандартов) для своих систем, приложений, чувствительных цифровых активов, уровня риска и требований соответствия. Также важно понимать, что многие стандарты могут быть адаптированы для удовлетворения конкретных требований организации.

Список общих стандартов безопасности можно найти в главе 10.

9.2 Применение стандартов безопасности

Обратите внимание на точное использование формулировок в любом стандарте (применимо только к документам на английском языке): Слово shall определяет обязательные требования, которым необходимо следовать для соответствия стандарту, в то время как слова should и may указывают на необязательные задачи, которые не являются обязательными для утверждения соответствия стандарту. Одно из типичных злоупотреблений - перепутать эти термины, либо требуя необязательный пункт, либо рассматривая обязательный пункт как необязательный.

Ситуации, характерные для организации или проекта, могут диктовать отклонение от строгого смысла используемого стандарта. Обоснование упущений, изменений или дополнений к содержанию стандарта должно быть задокументировано и согласовано всеми сторонами.

9.3 Тенденции отрасли

9.3.1 Где узнать о тенденциях отрасли в области информационной безопасности

Как общие, так и отраслевые новостные службы (публикации, веб-сайты, рассылки по электронной почте), а также мероприятия (конференции, торговые выставки, встречи профессионального сообщества) содержат информацию и обсуждение новых или растущих проблем. Принадлежность к специализированному профессиональному сообществу или сообществу практиков, скорее всего, обеспечит своевременные и целенаправленные обновления. Учитывая скорость, с которой развиваются новые возможности по использованию уязвимостей в программном обеспечении (эксплойты), электронные оповещения могут предложить наиболее оперативные меры реагирования.

Периодическая публикация наиболее частых или вредоносных эксплойтов может выявить широко распространенные тенденции, но особое внимание следует уделять вопросам, более специфичным для отрасли, области применения или продуктов, с которыми приходится работать. Эти вопросы с большей вероятностью будут освещаться в специализированных изданиях и службах новостей, а также на технических конференциях и профессиональных мероприятиях.

9.3.2 Оценка методов тестирования безопасности на предмет улучшений

При внедрении новых технологий или новых видов использования существующих технологий часто возникает возможность неправильного использования или эксплуатации технологии, пока ее риски и ограничения не станут более понятными.

Например, рассмотрим мобильные устройства со службами определения местоположения. В обмен на удобство или другие стимулы люди, похоже, готовы разрешить ежеминутное отслеживание своих перемещений и действий.

У криминальных, ориентированных на взлом, экономических и политических агентов возникает более широкий спектр мотиваций и больше ресурсов. Схемы вымогательства и защиты изменились с физических угроз на цифровые.

Крупные специальные сети идеологически мотивированных людей могут быть в кратчайшие сроки направлены против объектов их гнева. Корпоративный шпионаж часто хорошо финансируется и мотивируется. Государства, стремящиеся к экономическим и военным преимуществам, особенно хорошо обеспечены ресурсами и могут считать себя неуязвимыми для санкций или ответных мер.

Поскольку угрозы постоянно меняются и совершенствуются, специалисты по тестированию безопасности должны быть всегда готовы к противодействию следующей угрозе. Осведомленность об отрасли, тщательное отслеживание тенденций в области безопасности и приобретение наиболее подходящих инструментов обеспечивают наилучшую защиту организации.

10. Ссылки

Документы ISTQB

[ISTQB_FL_SYL] ISTQB Foundation Syllabus, 2011

[ISTQB_ATM_SYL] ISTQB Advanced Test Manager Syllabus, 2012

[ISTQB_ATTA_SYL] ISTQB Advanced Technical Test Analyst Syllabus, 2012

Стандарты

[ISO/IEC/IEEE 29119-3] - Software and systems engineering -- Software testing -- Part 3: Test documentation

[IEEE 12207] - ISO/IEC/IEEE Standard for Systems and Software Engineering - Software Life Cycle Processes

COBIT - http://www.isaca.org

ISO27001 – Information Security Management - http://www.iso.org/iso/home/standards/management-standards/iso27001.htm

PCI – Payment Card Industry Standard - https://www.pcisecuritystandards.org/

Книги

[Chapman, 2000] Chapman, Cooper, Zwicky, Building Internet Firewalls, O'Reilly & Associates, 2000.

[Jackson, 2010] Jackson, Christopher; Network Security Auditing, 2010.

Статьи

[ComputerWeekly] http://www.computerweekly.com/news/2240113549/Cattles-lost-backup-tapes-highlight-risk-of-unencrypted-data-storage

[Northcutt, 2014] Northcutt, Stephen; Security Controls, SANS Institute.

[Washington Post, 2007] http://www.washingtonpost.com/wp-dyn/content/article/2007/05/04/AR2007050402152.html

Guides

[Bittau] Cryptographic protection of TCP Streams (tcpcrypt)
https://tools.ietf.org/html/draft-bittau-tcp-crypt-04

[CERT1] Top 10 Secure Coding Practices
https://www.securecoding.cert.org/confluence/display/seccode/Top+10+Secure+Coding+Practices

[CERT2] http://www.cert.org/secure-coding/publications/index.cfm

[CERT3] http://www.cert.org/secure-coding/tools/index.cfm

[IEEE1] Avoiding the Top 10 Security Flaws
http://cybersecurity.ieee.org/center-for-secure-design/avoiding-the-top-10-security-flaws.html

[MDA1] MDA Glossary, DoD Missile Defense Agency, www.mda.mil

[NIST 800-30] NIST Special Publication 800-30, Rev 1, *Guide for Conducting Risk Assessments* (2012)

[NISTIR 7298] Glossary of Key Information - Security Terms, Revision 2 (2013)

[OWASP1] OWASP Secure Coding Practices Quick Reference Guide
https://www.owasp.org/index.php/OWASP_Secure_Coding_Practices_-_Quick_Reference_Guide

[OWASP2] OWASP Risk Rating Methodology
https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology

[OWASP3] OWASP Sample Authorization Form
https://www.owasp.org/index.php?title=Authorization_form

[PP-IDS] US Government Protection Profile Intrustion Detection System for basic robustness environments, version 1.7, 25 July 2007.

[SANS1] 25 Most Dangerous Software Errors – http://www.sans.org

[SANS2] Password Construction Guidelines - https://www.sans.org/security-resources/policies/general/pdf/password-construction-guidelines

[WAFEC] Web Application Firewall Evaluation Criteria, wasc-wafec-v1.0.pdf, 2006.

Отчеты

[WhiteHat Security, 2014] https://www.whitehatsec.com

Прочие ссылки

Следующие ссылки указывают на информацию, доступную в интернете. Эти ссылки проверялись во время публикации настоящей программы обучения Продвинутого уровня.

[CERT4] Vulnerability Notes Database - http://www.kb.cert.org/vuls/

[Chopitea] tomchop.me/2012/12/yo-dawg-i-heard-you-like-xoring/

[EICAR] www.eicar.org

[RFC2828] Internet Security Glossary - http://www.rfc-archive.org/getrfc.php?rfc=2828

[Web-1] Top 20 Critical Security Controls - http://sans.org

[Web-2] National Vulnerability Database - https://web.nvd.nist.gov/view/ncp/repository

[Web-3] Website Security Statistics Report - https://www.whitehatsec.com/resource/stats.html

[Web-4] The Google Hacking Database – http://hackersforcharity.org/ghdb

[Web-5] Shodan - shodanhq.com

[Web-6] NetCat - http://sectools.org/tool/netcat/

[Web-7] IP Tracer - http://www.ip-adress.com/ip_tracer

[Web-8] Computer Forensics, Cybercrime and Steganography Resources – http://www.forensics.nl

[Web-9] Tor Project - https://www.torproject.org/

[Web-10] Netstat - https://technet.microsoft.com/en-us/library/Bb490947.aspx

[Web-11] Tracert – http://www.tracert.com

[Web-12] RIPE Scan - https://www.ripe.net

[Web-13] Whois - https://www.whois.net/

[Web-14] NetCat – http://netcat.sourceforge.net/

[Web-15] Fping – fping.org

[Web-16] Hidetools – http://hidetools.com/

[Web-17] Kali Linux – https://www.kali.org/

[Web-18] Web Application Security Consortium – http://www.webappsec.org/

[Web-19] Hping - http://www.hping.org/

[Web-20] Nmap - https://nmap.org/

[Web-21] Zenmap - https://nmap.org/zenmap/

[Web-22] Xprobe2 - http://null-byte.wonderhowto.com/how-to/hack-like-pro-conduct-os-fingerprinting-with-xprobe2-0148439/

[Web-23] BackTrack - http://www.backtrack-linux.org/

[Web-24] Top 125 Network Security Tools - at https://sectools.org

[Web-25] DNS Lookup - https://who.is/dns/

[Web-26] John the Ripper - http://www.openwall.com/john/