GeekBrains

Актуальность использования поиска уязвимостей в WEB

Теоритический анализ ручного и автоматического поиска уязвимостей.

Дипломная работа посвящена исследованию актуальных систем безопасности, анализ их эффективности и оценка перспектив данного направления.

Анастасия Гаркавенко

Программист/ разработчик Python, 2023год

Немного о себе. Краткое описание
в несколько строк:

- Я проживаю в городе Саянск
- Увлекаюсь настольным теннисом
- Заняла 1 место среди подразделений на АО «Саянскхимпласт»
- В будущем планирую поступить в институт на факультет: автоматизации машиностроительного производства.

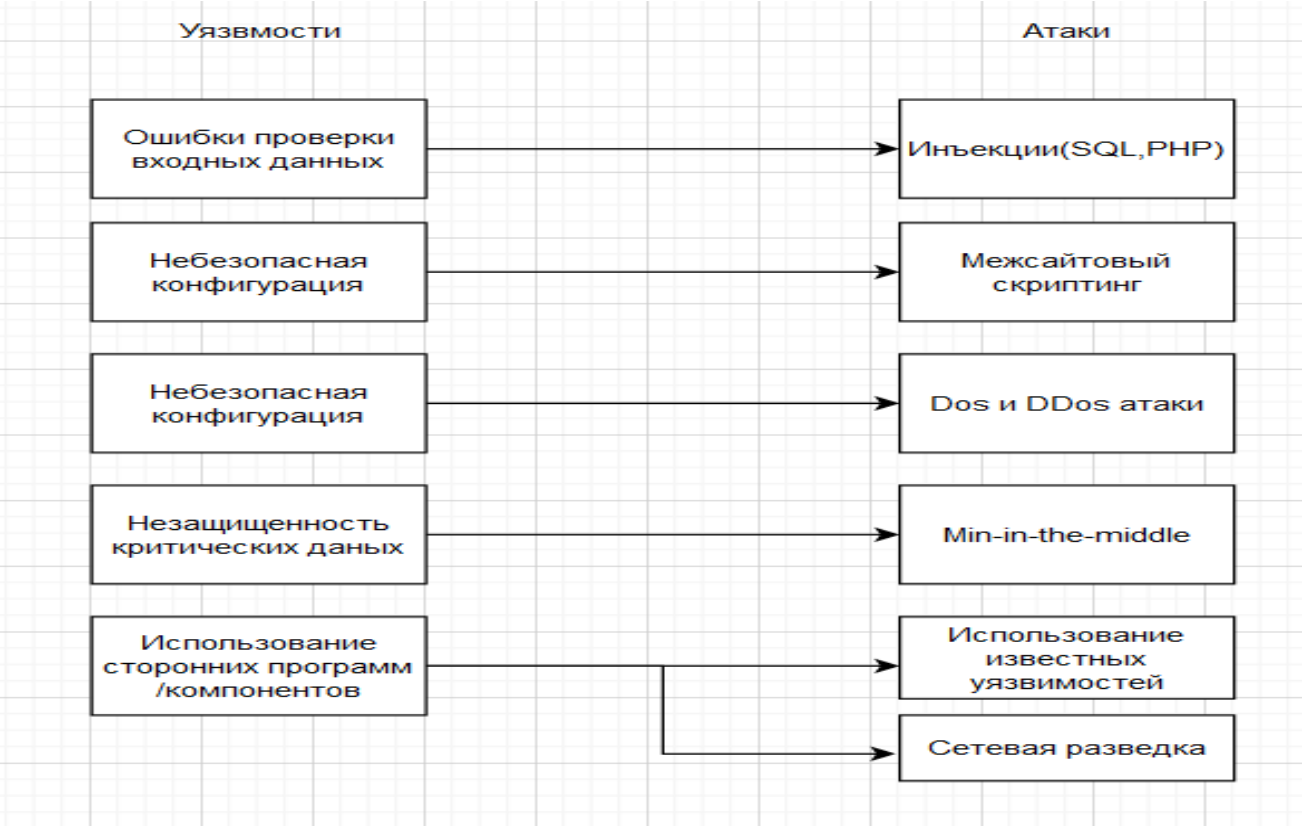
Анализ и перспективы обеспечения безопасности в WEB

При разработке проекта, на любой стадии работы, нужно задумываться о том как его защитить. Каждый разработчик должен помнить о том, что есть люди, которые занимаются нелегальной деятельностью. Поэтому каждый в команде должен заботиться о своевременном поиске дыр, через которые может пролезть мошенник. Анализ уязвимостей помогает обеспечить безопасность, выявить ошибки и утечки, предотвратить финансовые потери и сохранить репутацию. Он включает идентификацию уязвимостей, предотвращение кражи данных, защиту от кибератак, соответствие нормативам и стандартам, а также улучшает общую безопасность.

Решение задачи / План работы

1. Анализ предметной области	2. Проектирование средства автоматизированного поиска уязвимостей для Веб - приложений	3. Реализация и тестирование средства автоматизированного поиска уязвимостей для Веб - приложений
Обзор существующих методов и инструментов для поиска уязвимостей в Веб - приложениях	Описание архитектуры средства	Выбор языка программирования и фреймворка для реализации средства
Анализ типичных уязвимостей и их последствий для Веб- приложений	Разработка модулей и функциональности средства	Система управления базой данных
Обзор существующих методов защиты Веб - приложений от уязвимостей	Разработка основных алгоритмов работы средства	Тестирование средства на тестовых Веб - приложениях с известными уязвимостями
Описание основных требований к средству автоматизированного поиска уязвимостей для Веб - приложений	Описание методов тестирования и проверки работоспособности средства	Заключение

Классификация уязвимостей и атак



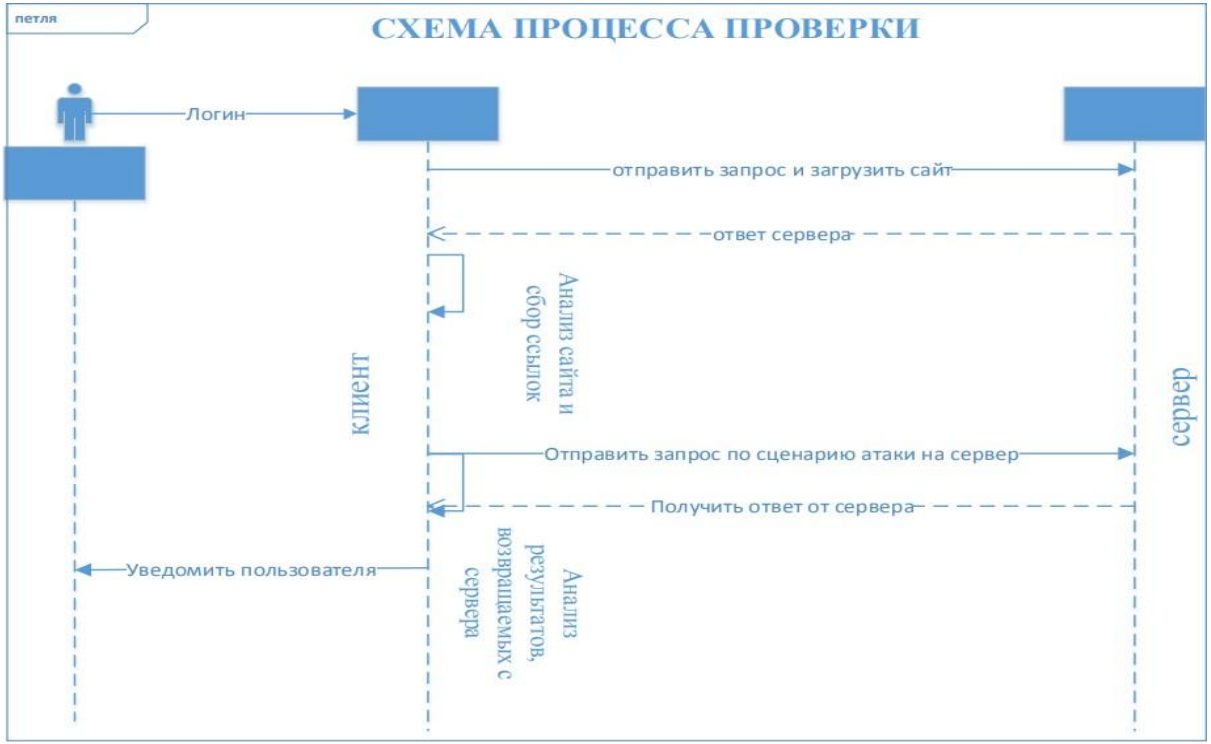
Сравнение ручного и автоматического тестирования

	Ручное тестированиеуязвимостей	Автоматическое тестирование уязвимостей
Определение	Процесс выявления уязвимостей и слабых мест в системе или приложении посредством ручной проверки и тестирования.	Процесс выявления уязвимостей и слабых мест в системе или приложении спомощью инструментов тестирования.
Плюсы	Позволяет применять более персонализированный подход и может обнаруживать сложные уязвимости, которые могутпропустить инструменты. Ручное тестированиеуязвимостей	Может быстро и эффективно обрабатывать большой объем кода, экономя время и ресурсы. Автоматическое тестирование уязвимостей
Минусы	Может отнимать много времени и требует квалифицированного персонала для эффективной работы.	Может привести к ложным срабатываниям или пропустить более тонкие уязвимости, требующие анализа человеком.
Подходит для	Небольших илиспециализированных приложений или ситуаций,когда необходим более индивидуальный подход.	крупномасштабных приложений или ситуаций, когда время и ресурсы ограничены.
Примеры	Тестирование на проникновение, проверкакода, моделирование угроз.	Статический анализ, Динамический анализ,нечеткое тестирование.

Алгоритмы и методы обнаружения уязвимостей

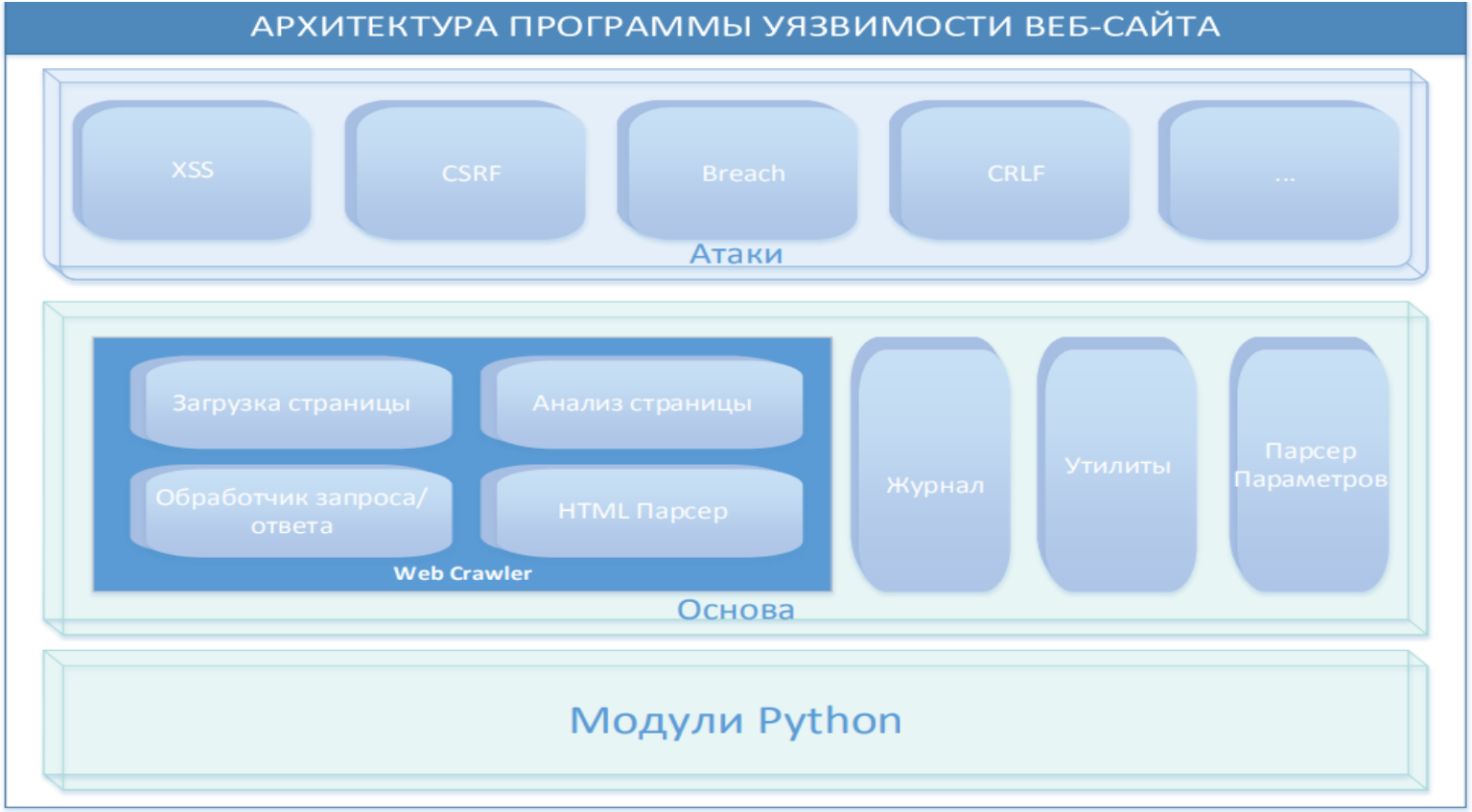
Алгоритм/метод	Описание	Примеры
Статическое тестирование безопасности приложений (SAST)	Анализ исходного кода для выявления уязвимостей	SQL-инъекция, XSS, переполнение буфера
Динамическое тестирование безопасности приложений (DAST)	Тестирование приложения в работающем состоянии на выявление уязвимостей	Инъекционные атаки, нарушенная аутентификацияи управление сессиями, небезопасная связь
Пушистое тестирование	Отправка больших объемов случайных данных в приложение для обнаружения уязвимостей	Переполнение буфера, уязвимости строки формата, утечки памяти
Ручное тестирование	Тестировщик вручную тестирует приложение для выявления уязвимостей	Логические ошибки,ошибки бизнес - логики,атаки социальной инженерии
Гибридное тестирование	Объединение нескольких методов тестирования для повышения точности и полноты	SAST, DAST, фаз - тестирование

Схема процесса проверки



Модель взаимодействия программы

Архитектура программы уязвимости Web-сайта



Идеи на будущее

Данная тема раскрывает проблему нашего века цифровизации. В новостях часто пишут, что в различных крупных сервисах происходит слив базы данных клиентов. Мошенники взламывают чужие приложения и получают доступ к документам и денежным средствам. Поэтому данный урок должен иметь одно из приоритетных мест в любом курсе обучения. Мы все работает на благо наших клиентов, партнеров и репутация разработчика который заботиться о безопасности клиента будет всегда на высоте. В будущем я буду придерживаться этих правил.