**Лекция 4.1. Уязвимости веб сервисов**

В данной лекции рассмотрены следующие вопросы:  
1. Инъекция;  
2. Недостатки аутентификации.  
3. Разглашение конфиденциальных данных.  
4. Внедрение внешних сущностей XML.  
5. Недостатки контроля доступа.  
6. Некорректная настройка параметров безопасности.  
7. Межсайтовое выполнение сценариев.  
8. Небезопасная десериализация.  
9.  Использование компонентов с известными уязвимостями.  
10. Недостатки журналирования и мониторинга.

Глава 1. Уязвимости веб сервисов

Уязвимости веб-приложений возникают тогда, когда разработчики добавляют небезопасный код в веб-приложение. Это может происходить как на этапе разработки, так и на этапе доработки или исправления найденных ранее уязвимостей.  
OWASP TOP 10  
А1. Инъекции. Уязвимости, связанные с внедрением SQL, NoSQL, OS и LDAP. Возникают, когда непроверенные данные отправляются интерпретатору в составе команды или запроса. Вредоносные данные могут заставить интерпретатор выполнить непредусмотренные команды или обратиться к данным без прохождения соответствующей авторизации.  
А2. Недостатки аутентификации. Функции приложений, связанные с аутентификацией и управлением сессиями, часто некорректно реализуются, позволяя злоумышленникам скомпрометировать пароли, ключи или сессионные токены, а также эксплуатировать другие ошибки реализации для временного или постоянного перехвата учётных записей пользователей.  
А3. Разглашение конфиденциальных данных. Многие веб-приложения и API имеют плохую защиту критичных финансовых, медицинских или персональных данных. Злоумышленники могут похитить или изменить эти данные, а затем осуществить мошеннические действия с кредитными картами или персональными данными. Конфиденциальные данные требуют дополнительных мер защиты, например их шифрования при хранении или передаче, а также специальных мер предосторожности при работе с браузером.  
А4. Внедрение внешних сущностей XML. Старые или плохо настроенные XML-процессоры обрабатывают ссылки на внешние сущности внутри документов. Эти сущности могут быть использованы для доступа к внутренним файлам через обработчиков URI файлов, общие папки, сканирование портов, удалённое выполнения кода и отказ в обслуживании.  
А5. Недостатки контроля доступа. Действия, разрешённые аутентифицированным пользователям, зачастую некорректно контролируются. Злоумышленники могут воспользоваться этими недостатками и получить несанкционированный доступ к учётным записям других пользователей или конфиденциальной информации, а также изменить пользовательские данные или права доступа.  
А6. Некорректная настройка параметров безопасности. Некорректная настройка безопасности является распространённой ошибкой. Это происходит из-за использования стандартных параметров безопасности, неполной или специфичной настройки, открытого облачного хранения, некорректных HTTP-заголовков и подробных сообщений об ошибках, содержащих критичные данные. Все ОС, фреймворки, библиотеки и приложения должны быть не только настроены должным образом, но и своевременно корректироваться и обновляться.  
А7. Межсайтовое выполнение сценариев. XSS имеет место, когда приложение добавляет непроверенные данные на новую веб-страницу без их соответствующей проверки или преобразования, или когда обновляет открытую страницу через API браузера, используя предоставленные пользователем данные, содержащие HTML- или JavaScript-код. С помощью XSS злоумышленники могут выполнять сценарии в браузере жертвы, позволяющие им перехватывать пользовательские сессии, подменять страницы сайта или перенаправлять пользователей на вредоносные сайты.  
А8. Небезопасная десериализация. Небезопасная десериализация часто приводит к удалённому выполнению кода. Ошибки десериализации, не приводящие к удалённому выполнению кода, могут быть использованы для атак с повторным воспроизведением, внедрением и повышением привилегий.  
А9. Использование компонентов с известными уязвимостями.  Компоненты, такие как библиотеки, фреймворки и программные модули, запускаются с привилегиями приложения. Эксплуатация уязвимого компонента может привести к потере данных или перехвату контроля над сервером. Использование приложениями и API компонентов с известными уязвимостями может нарушить защиту приложения и привести к серьёзным последствиям.  
Недостатки журналирования и мониторинга. Недостатки журналирования и мониторинга, а также отсутствие или неэффективное использование системы реагирования на инциденты, позволяет злоумышленникам развить атаку, скрыть своё присутствие и проникнуть в другие системы, а также изменить, извлечь или уничтожить данные. Проникновение в систему обычно обнаруживают только через 200 дней и, как правило, сторонние исследователи, а не в рамках внутренних проверок или мониторинга.

Глава 2. Уязвимости веб сервисов

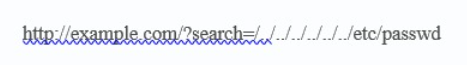
Рассмотрим типовые уязвимости, которым подвержены многие веб-приложения.  
Инъекции  
Как и полагается, атаки класса «Инъекции» занимают лидирующую строчку рейтинга  
OWASP Top 10, встречаясь практически повсеместно и являясь крайне разнообразными в реализации. Уязвимости подобного класса начинаются SQL-инъекциями, в различных его вариациях, и заканчивая RCE — удалённым выполнением кода:



XSS  
Межсайтовый скриптинг — уязвимость, встречающаяся на данный момент куда реже, чем раньше, если верить рейтингу OWASP Top 10, но несмотря на это не стала менее опасной для веб-приложений и пользователей:



LFI/RFI  
Уязвимости данного класса позволяют злоумышленникам через браузер включать локальные и удалённые файлы на сервере в ответ от веб-приложения:



Атаки через JSON и XML  
Веб-приложения и API, обрабатывающие запросы в формате JSON или XML, также подвержены атакам, поскольку такие форматы имеют свои недостатки.  
Инструменты поиска уязвимостей  
Есть коммерческие продукты: Acunetix, Nessus Scanner, Nexpose, но имеющие и бесплатные пробные версии с ограниченным функционалом. А есть и полностью бесплатные: Wapiti, Nikto, Vega, SQLmap.  
С коммерческими продуктами всё в основном просто и понятно. Они разрабатываются для максимальной функциональности и удобства пользователей. Как правило, при их использовании от пользователя практически никаких действий не требуется — всё происходит автоматически, стоит только указать цель сканирования.  
С полностью бесплатными решениями дело обстоит немного иначе. Они сами по себе требуют от пользователя участия на протяжении всего процесса сканирования и к тому же многие из них часто бывают узкоспециализированными, не позволяющими охватить весь спектр уязвимостей для их поиска.

1. Уязвимость, которая возникает, когда веб-приложение неправильно обрабатывает пользовательский ввод и позволяет злоумышленнику внедрить вредоносный код в запросы, выполняемые базой данных.
   1. XML
   2. SQL-инъекция
   3. XSS-атака
2. Уязвимость, которая позволяет злоумышленнику внедрить вредоносный код (обычно JavaScript) в веб-страницу. Злоумышленник может использовать эту уязвимость для кражи сессионных куки, перехвата личной информации пользователей или перенаправления на фальшивые веб-сайты с целью мошенничества.
   1. XML
   2. SQL-инъекция
   3. XSS-атака
3. Уязвимость, связанная с обработкой сериализованных объектов данных.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
4. Уязвимость, которая является одной из наиболее распространённых уязвимостей веб-приложений. Злоумышленники могут искать и эксплуатировать известные уязвимости в сторонних компонентах, таких как библиотеки, фреймворки и плагины.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
5. Уязвимость, которая затрудняет расследование инцидентов и выявление атак, а недостаточный мониторинг может привести к упущению предупреждающих сигналов и задержке в реагировании на инциденты.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
6. Уязвимость, которая возникает, когда механизмы аутентификации слабы или недостаточно защищены.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
7. Уязвимость, которая возникает, когда конфиденциальные данные, такие как пароли, персональная информация или финансовые данные, неправильно хранятся, передаются или обрабатываются.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
8. Уязвимость, которая возникает, когда система неправильно ограничивает доступ пользователей к ресурсам или функциональности. В этом случае злоумышленник получает несанкционированный доступ к конфиденциальным данным.
   1. Недостатки журналирования и мониторинга
   2. Использование компонентов с известными уязвимостями
   3. Разглашение конфиденциальных данных
   4. Небезопасная десериализация
   5. Недостатки аутентификации
   6. Недостатки контроля доступа
9. Какое сканирование портов позволяет собирать информацию о доступных портах и службах без отправки активных запросов?
   1. Пассивное сканирование
   2. Активное сканирование
   3. Нет верного ответа
10. Какое сканирование портов включает отправку запросов на открытые порты для определения, какие службы работают на удалённой системе?
    1. Пассивное сканирование
    2. Активное сканирование
    3. Нет верного ответа