Owasp part 2

Сегодня мы продолжим тему безопасности в web’е, которую начали рассматривать в субботу. Поэтому хочу напомнить некоторые аспекты:

- Open Web Application Security Project – некоммерческая организация которая составляет топы ошибок безопасности

- И то что моя задача только ознакомить вас с тем что такое существует, и обратить на это ваше внимание. Более подробно с описанием каждой уязвимости, способов исправления можно ознакомиться по первой ссылкой в гугле по названию уязвимости. Это приведет вас на сайт owasp.org

Кратко вспомним почему появилась потребность в Web безопасности:

1. Люди ошибались, ошибаются, и будут ошибаться

2. Разработка так чтобы побыстрее, подешевле, и чтоб запускалось

3. Большое количество разработчиков не умеющих думать/программировать, а тем более разбираться в нюансах.

4. В вебе как и везде, есть некоторые достаточно специфичные темы, на которые никто не обращает внимание. А также, то что и сам разработчик может быть не полностью знаком с технологией которую он использует.

В прошлый раз мы рассмотрели 5 открывающих пункта рассматриваемого топа.

Сейчас продолжаем, и на очереди у нас 5й пункт - Неправильная конфигурация. Мы уже говорим про среду в которой наше ПО разрабатывается, тестируется, разварачивается.

Это связано с:

- Использование устаревшего, или уязвимого ПО

- Включены или установлены ненужные функции (например такие как, ненужные порты, службы, страницы, учетные записи или привилегии), на которые мы можем забить, которые мы можем забыть

- Отключены функции безопасности, настроены ненадежно. Думаю тут все ясно.

На 4м месте расположилась категория Небезопасный дизайн. Она касается рисков, связанных с недостатками проектирования. Те даже при совершенной реализации задумки, продукт будет небезопасен по определению, потому что необходимые средства контроля безопасности не создавались.

Тройку лидеров открывает семейство уязвимостей связанных с внедрением кода. Плохим способом взаимодействия пользователя с сервисом.

Последствия в зависимости от типа могут отличаться:

- Если sql-иньекции, то это утечка бд, внедрение вредоносных данных в бд

- Command Injection работает если данные пользователя прямо подставляются в команду. И от сюда уже полный/частичный контроль злоумышленником вашего сервера.

- File Inclusion - Включение локальных файлов. Когда недоброжелатели через браузер могут подключать файлы на сервере.

Второе место отходит криптографии. Использование старых или слабых алгоритмов, отсутствие шифрования необходимых данных, неправильное шифрование, или передача ключей – всё сюда.

Первое место занимают Нарушение уровня доступа. Здесь подразумеваются действия пользователей за пределами предполагаемых им разрешений.

Два вопроса

1. Это к нам относиться?

2. Где про это почитать?