Безопасность приложений

RCE

Injections?

- Под инъекциями в OWASP TOP-10 подразумевается все подряд
- Любое внедрение команд/операторов в любой путь исполнения программы
- Практически любую уязвимость можно представить как A01 ;)

Remote Code Execution

- Ошибочно отождествляют с OS commanding (OWASP, WASC)
- Code - это не только команды ОС, но и код приложения (PHP, NodeJS, Java, etc)
- OS commanding - термин относящийся только к ошибкам вызова шелла из приложений

OS commanding

- Проблема не нова - фильтрация пользовательских данных (input validation)
- Не стоит вызывать команды ОС прямо из приложений - используйте модули-обертки
- Типичный шаблон уязвимости:
  - exec(“unzip ”.$_POST[‘filename’])

Команды и аргументы

- Командная строка и ее особенности зависят от ОС и установленного шелла (sh!=bash)
- Эти особенности не всегда могут учитываться при фильтрации
- Юникодные символы, пайпы, переносы строк - никакой системы...

Выполнение кода из файлов - upload

- Самая частая проблема выполнения кода - ошибки при загрузке файлов
- Веб-приложения чаще всего представляют собой отдельные файлы с логикой
- Выполнение каждого такого файла напрямую через HTTP-запрос - ДИЗАЙН

Выполнение кода из файлов - including

- (require|include)(_once)? - PHP функции для подключения из одного сценария другого - типичные проблемные места
- Формат файла может быть произвольным, если в тексте есть макросы интерпретатора (<?php - PHP, <% - JSP)
- Шаблон уязвимости:
  - include(“/languages/”.$_COOKIE[‘lang’].”.tpl”);

Выполнение кода из файлов - что делать?

- Использовать application-server и настраивать права ФС + конфиги сервера
- Загружать пользовательские файлы в отдельную среду без возможности исполнения
- Проверять user input / переписывать имена файлов

Макросы

- Приложение по дизайну часто может/ должно обрабатывать некоторые собственные команды - макросы
- Обработка макросов реализуется чаще всего через сам интерпретатор (eval и аналоги)
- Фильтрация, снова фильтрация...

Пример ошибки парсера макросов

- Struts2 OGNL exploit:
  - nokia, qiwi и все-все-все

http://host/struts2-blank/example/X.action?action:%25{(new+java.lang.ProcessBuilder(new+java.lang.String[]{'command','goes','here'})).start()}

RCE при десериализации

- Сериализация объектов – это удобно
  - сериализуются данные, а код?
- При десериализации
  - pickle, yaml, …
  - http://pyyaml.org/wiki/PyYAMLDocumentation
  - pickle.loads("cos\nsystem\n(S'ls ~'\ntR.")
  - типичные CTF-таски
- Не десериализовывать user input!

RCE при XSLT

- XML, XSL, XSLT
- Многие парсеры (Xalan, libxml) поддерживают вызов методов языка во время трансформаций
- Правила трансформацией брать от пользователя? Будет больно!

RCE при XSLT

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:jv="http://xml.apache.org/xalan/java" exclude-
result-prefixes="jv" version="1.0">
<xsl:template match="/">
<root>
<xsl:variable name="osversion”
select="jv:java.lang.System.getProperty('os.name')"/>
<xsl:value-of select="$osversion" />
</root>
</xsl:template>
</xsl:stylesheet>
```

Вопросы?