



Оценка защищенности Web-приложений

Сергей Гордейчик
Positive Technologies

Насколько уязвимы Web-приложения?

Что такое «безопасное приложение»?

Методики и подходы

Критерии качества

•Мировая статистика

- Mitre: более четверти уязвимостей, обнаруженных в 2006 году приходится на Web-приложения [1].
- Symantec «Internet Security Threat Report»: до 70% уязвимостей, используемых злоумышленниками, связаны с Web-приложениями [2].
- Web Application Security Consortiums: 70% приложений имеют проблемы с безопасностью [3].

•Российская действительность

- До 65% Web-приложения содержат уязвимости высокой степени риска [4].

[1] <http://cwe.mitre.org/documents/vuln-trends.html>

[2] http://www.symantec.com/specprog/threatreport/ent-whitepaper_symantec_internet_security_threat_report_x_09_2006.en-us.pdf

[3] <http://www.webappsec.org/projects/threat/>

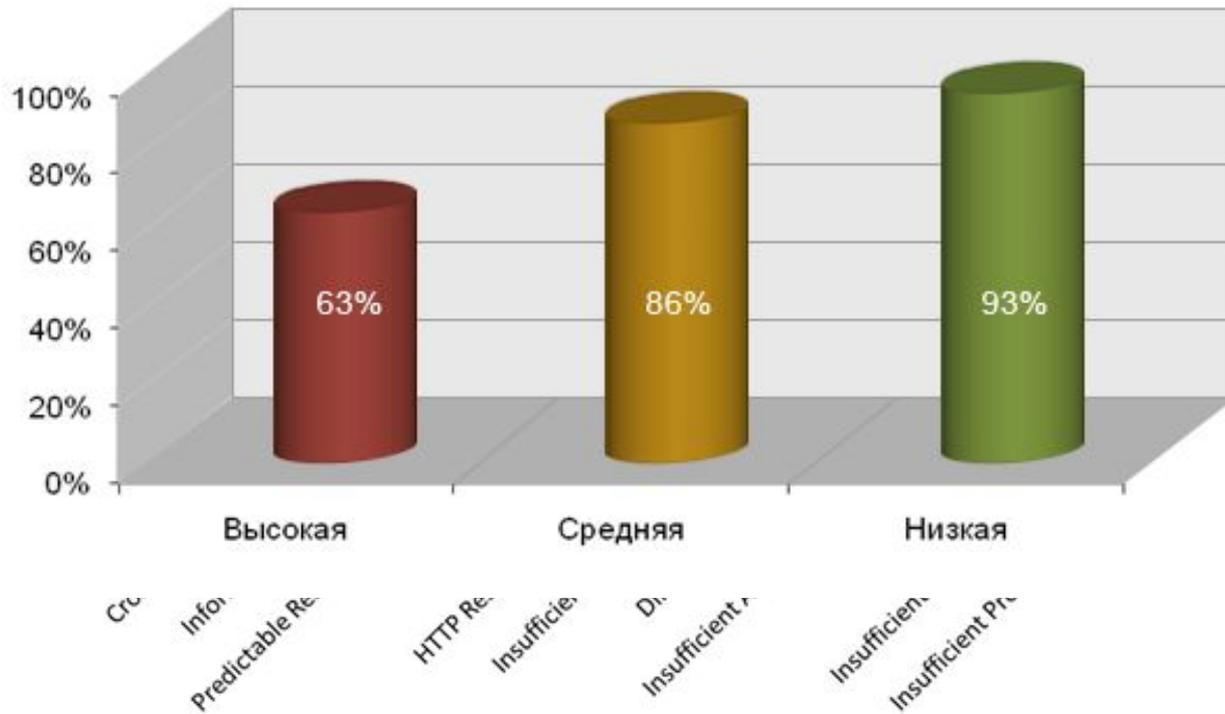
[4] <http://www.ptsecurity.ru/stat2007.asp>

Уязвимости Web-приложений



Scripting on Leakage
onse Splitting
ial

35.5



statistics/

Выработка и контроль требований по безопасности приложений учитывающих качество реализации

- Элемент Secure SDLC
- Крайне ресурсоемкое решение
- Тяжело интегрируется во многие модели разработки

«Оценка защищенности», «тесты на проникновение» и т.д.

- Уязвимости обнаруживаются после «сдачи» приложения
- Зачастую уязвимости не устраняются (15% уязвимостей были обнаружены повторно)
- Может быть весьма ресурсоемким решением (особенно в случае анализа кода)

Сканеры уязвимостей

- Достаточно бюджетное решение
- Позволяют обнаруживать до 70% уязвимостей (по отношению к Pentest)
- Многие типы уязвимостей (особенно связанные с бизнес-логикой) не могут быть найдены сканером

Web Application Firewall

- Бюджетное решение
- Весьма «капризно» в настройке
- Низкая эффективность
- Качество реализации защитных механизмов «непрозрачно»

Что такое безопасное Web-
приложение???

Проектирование («Фичесет»)

ГОСТ Р ИСО/МЭК 15408 (Common Criteria)

набор функций безопасности (аутентификация, аудит и т.д.)

Реализация и поддержка

OWASP top 10

http://www.owasp.org/index.php/OWASP_Top_Ten_Project

Web Security Threat Classification

<http://www.webappsec.org/projects/threat/>

OWASP top 10

http://www.owasp.org/index.php/OWASP_Top_Ten_Project

Поддерживается (последняя версия 2007 года)

Только 10 уязвимостей

Web Security Threat Classification

<http://www.webappsec.org/projects/threat/>

Текущая версия 1 – 2004/2004 год

Готовится к выходу 2 версия

WSTC v 1.0

6 классов уязвимостей
24 типа атак/уязвимостей

- 1 Authentication
- 2 Authorization
- 3 Client-side Attacks
- 4 Command Execution
- 5 Information Disclosure
- 6 Logical Attacks

WSTC v 2.0

9 классов уязвимостей
37 типа атак/уязвимостей

- 1 Authentication
- 2 Authorization
- 3 Client-Side
- 4 Command Execution
- 5 Information Disclosure
- 6 Logical Flaws
- 7 Misconfiguration
- 8 Protocol Abuse
- 9 XML Attacks

Анализ спецификации/проекта

Тестирование функций

Фаззинг (fuzzing)

Анализ исходного кода

Экспертная оценка архитектуры с точки зрения безопасности

Что хорошо:

Выявляются фундаментальные проблемы

Что плохо:

«Сплошное экспертное мнение», нет понимания реализации

Инструменты:

Office, Adobe Reader, диктофон

Проверка качества реализации механизмов безопасности

Что хорошо:

Authentication

Authorization

Logical Flaws

Что плохо:

Много ручной работы + нет четких критериев

Инструменты:

Браузер + расширения

Proxu

Инструменты:

Plugins

Selenium

TamperData

FireBug

Chickenfoot

Proxy

WebSaccarab

Praos

Sniffers

IE Inspector HTTP Analyzer

Передача «хорошо известных» плохих параметров на вход приложению

Что хорошо:

Client Side (Cross-Site Scripting, и т.д)

Code Execution (SQL Injection, и т.д)

Information Disclosure

XML

Protocol Abuse

Что плохо:

Все остальное

Проблемы с сессиями

Хорошее добавление – журналы аудита + grep
(СУБД, Web-сервер, сервер приложений, ОС)

Инструменты

Сканеры уязвимостей Web-приложений:

XSpider

WebInspect (HP)

Watchfire AppScan (IBM)

OWASP WebScarab

...

Дополнительно

Требуется понимание структуры приложения

Проверка исходного кода на предмет наличия уязвимостей

Что хорошо:

Облегчение поиска любых уязвимостей

Что плохо:

Большой объем ручной работы

Статический анализ

Широко распространен

Большое количество инструментов

Большое количество ложных срабатываний

Отсутствие анализа DataFlow

Динамический (гибридный) анализ

Позволяет отсеивать ложные срабатывания

Слабо распространен для Web-приложений

Инструменты

Coverity

Valgrind

Insure++

Checkmarx

CUTE

Fortify PTA

Open

FindBugs (Java)

LAPSE: Web Application Security Scanner for Java

Microsoft FxCop (.NET)

Вопрос об эффективности тех или иных подходов остается открытым.

Taking the Blinders off Black Box Security Testing, Fortify Software

Four Representative Applications

Application	Size of the Application # of Classes	Coverage Entire Application		Coverage Web Facing Points	
		Automated Test	Manual Test	Automated Test	Manual Test
Manufacturing Fulfillment Application	32	22%	36%	62%	96%
Online Commerce Application	6,278	24%	30%	49%	62%
AJAX-Based Commerce Application	737	8%	27%	23%	70%
Online Banking Application	52	13%	51%	77%	77%
Avg of Above 4		17%	36%	53%	76%
Avg Across all Tests Conducted		18%	26%	55%	68%

Большинство Web-приложений содержит серьезные уязвимости

- XSS, CSRF

- Разнообразные утечки информации

- SQL Injection

- Ошибки авторизации/аутентификации

Оптимальным является «Gray box» тестирование

- Сканеры

- Ручной анализ

- Анализ исходного кода

Степень покрытия функций приложения 20-50%

Вопросы?

Гордейчик Сергей
Positive Technologies
gordey@ptsecurity.ru