Введение в тему безопасности веб-приложений

Дмитрий Евтеев (Positive Technologies)

POSITIVE TECHNOLOGIES

Немного статистики

По данным компании Positive Technologies

- более 80% сайтов содержат критические уязвимости
- вероятность автоматизированного заражения страниц уязвимого веб-приложения вредоносным кодом составляет сегодня приблизительно 15-20%

http://ptsecurity.ru/analytics.asp

POSITIVE TECHNOLOGIES

Последние громкие события

- **Массовый взлом сайтов через внедрение операторов SQL**

  http://www.xakep.ru/post/57405/

  http://www.xakep.ru/post/55242/

- **Уязвимость в плагине WordPress привела к массовому заражению веб-сайтов (загрузка серверных сценариев)**

  http://www.anti-malware.ru/news/2011-10-31/4838

  Спустя три месяца с момента обнаружения уязвимости динамика поражения блогов WordPress сохраняется - http://www.opennet.ru/opennews/art.shtml?num=32219

- **Уязвимость в плагинах Joomla привела к массовому взлому веб-сайтов (LFI)**

  http://tacticalwebappsec.blogspot.com/2011/11/mass-joomla-component-lfi-attacks.html

Из чего складывается безопасность

- **Компьютер и браузер пользователя**
- **Канал связи между всеми компонентами**
- **Межсетевой экран, IPS, WAF**
- **Балансировщик нагрузки**
- **Веб-сервер**
- **Приложение**
- **Back end (СУБД, XML, SOAP, etc)**

Из чего складывается безопасность #1

Компьютер и браузер пользователя

- Обеспечить полноценную безопасность с не доверенного источника невозможно (!)
- Безопасность браузера

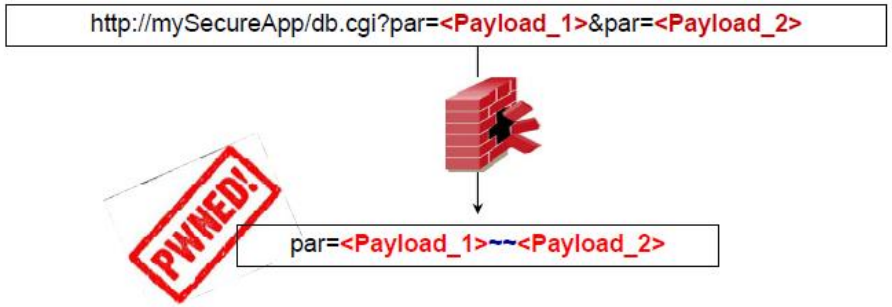
- Same origin policy
- Sandbox
- механизмы противодействующие атакам с использованием межсайтового скриптинга
- блокировка вредоносных сайтов
- механизмы аутентификации
- изоляция вкладок
- компоненты (Flash, Java, etc)
- дополнения (eq NoScript)

Из чего складывается безопасность #2

- **Канал связи между всеми компонентами**
- **Браузер <-> веб-сервер**
  - SSL/TLS
  - Basic/Digest/NTLM
  - Cookie Secure
- **Веб-сервер <-> Back end**
  - Встроенные механизмы криптографической защиты
  - IPSEC

Из чего складывается безопасность #3

- **Межсетевой экран, IPS, WAF**
- **Межсетевой экран**
  - deny all & profit1
- **IPS/WAF**
  - Ошибки первого и второго рода
  - Не должны мешать «правильной» работе приложения
  - Не обеспечивают 100% защиты (!)



Из чего складывается безопасность #4

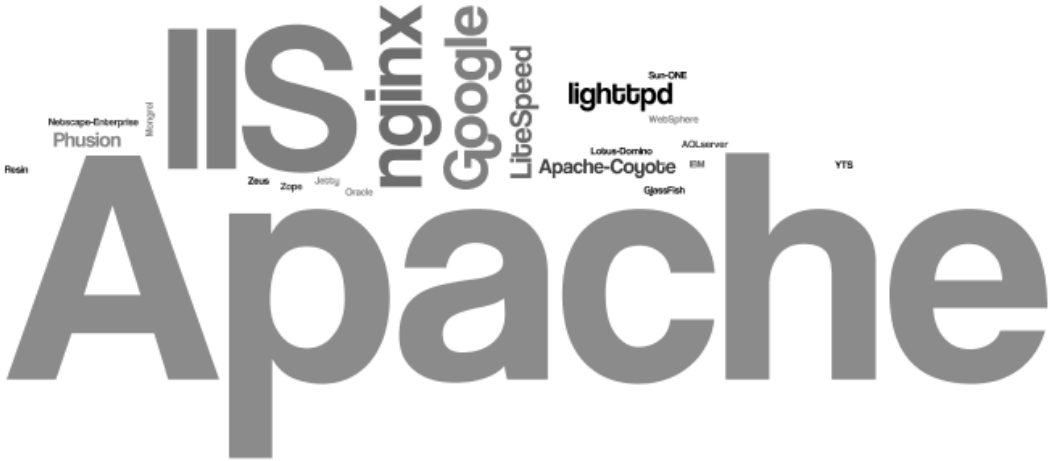
Балансировщик нагрузки

- должен обеспечивать защиту от DoS-атак
- вводит в заблуждение веб-хакера, действующего удаленно :)

Из чего складывается безопасность #5

Веб-сервер

- управление обновлениями (для всех компонентов)
- уязвимые конфигурации (листинг каталогов, Verb Tampering, стандартные сообщения об ошибках и т.п.)
- разграничение доступа (процесс и файловая система)
- превентивные механизмы защиты (systrace, apparmor, etc)



Из чего складывается безопасность #6

- **Веб-приложение - логика, реализованная на некотором языке разработки**

- **Наиболее часто ~~встречаемые~~ используемые уязвимости**
  - **на стороне сервера (внедрение операторов SQL, подбор паролей, выполнение команд операционной системы)**
  - **на стороне клиента (межсайтовое выполнение сценариев, перенаправление HTTP-запроса, открытое перенаправление)**

Какой код более кошерный?

Распределение уязвимостей по данным VERACODE

Java		ColdFusion		C/C++		.NET		PHP	
Cross-site Scripting (XSS)	50%	Cross-site Scripting (XSS)	89%	Buffer Overflow	27%	Cross-site Scripting (XSS)	44%	Cross-site Scripting (XSS)	80%
CRLF Injection	17%	SQL Injection	9%	Error Handling	23%	Information Leakage	23%	Directory Traversal	8%
Information Leakage	14%	OS Command Injection	<1%	Potential Backdoor	22%	Cryptographic Issues	11%	SQL Injection	6%
Cryptographic Issues	5%	Information Leakage	<1%	Numeric Orders	11%	Directory Traversal	8%	Information Leakage	3%
Directory Traversal	4%	Directory Traversal	<1%	Buffer Mgmt Errors	9%	Insufficient Input Validation	6%	Code Injection	1%

Veracode, 2011, http://info.veracode.com/rs/veracode/images/soss-v3.pdf

Из чего складывается безопасность #7

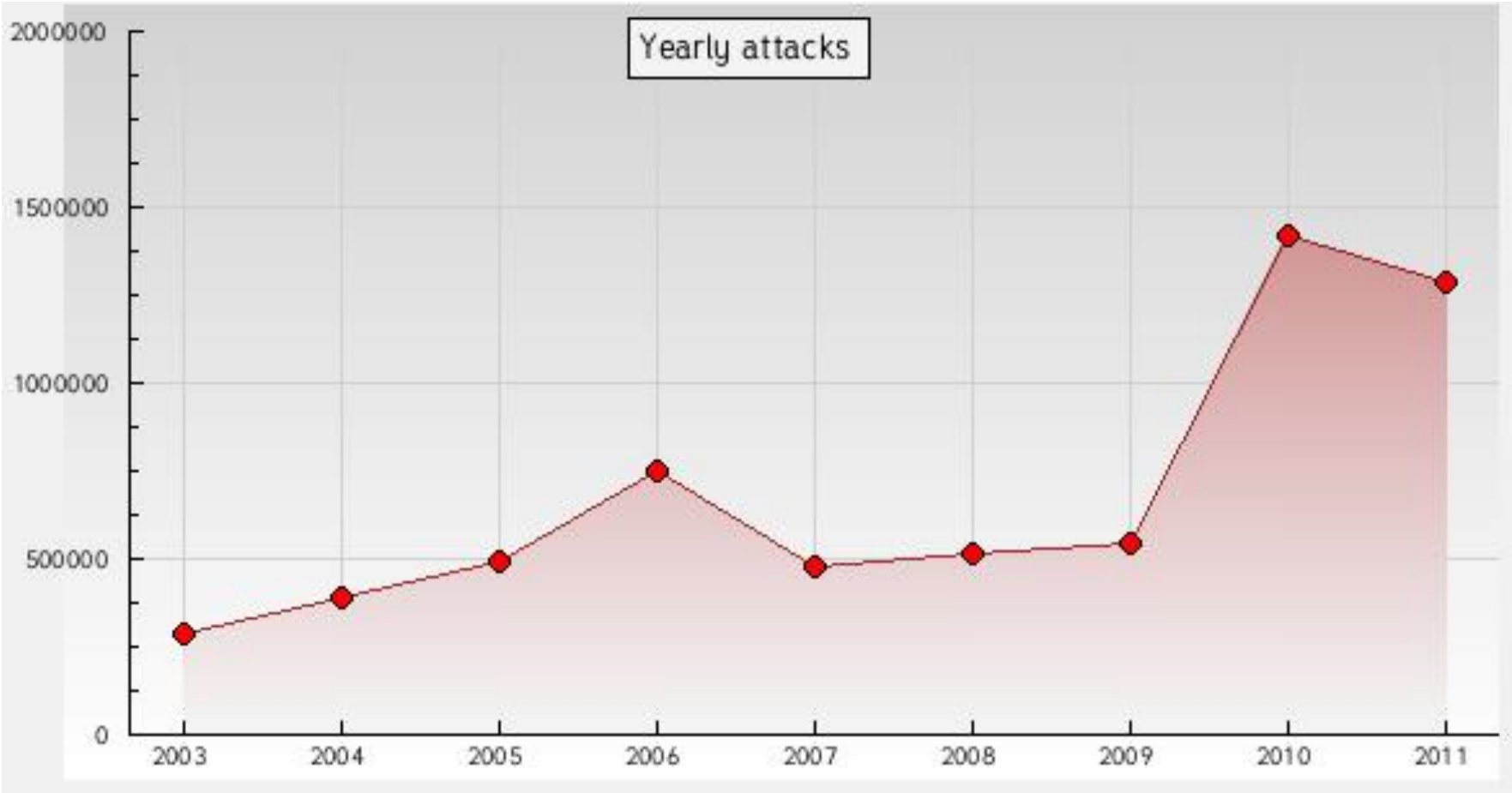
- **Back end (СУБД, XML, SOAP, etc)**
- **СУБД**
  - управление обновлениями (для всех компонентов)
  - уязвимые конфигурации (PostgreSQL, LOAD DATA LOCAL…)
  - разграничение доступа (процесс, файловая система, базы…)
  - превентивные механизмы защиты (systrace, apparmor, etc)
  - криптографическая защита данных (!)

Угрозы веб-приложений

Какие цели преследует злоумышленник взламывая веб-сайты?

- Black SEO
- Распространение спама
- Распространение вредоносного кода
- Кража и искажение данных, предоставленных на сайте
- Удаление содержимого сайта (DoS-атака)
- Использование сайта в качестве плацдарма для проведения атак на другие ресурсы
- Ради шутки ☺

Взломы ради шуток ☺



Статистика дефейсов zone-h.org (http://zone-h.org/stats/ymd)

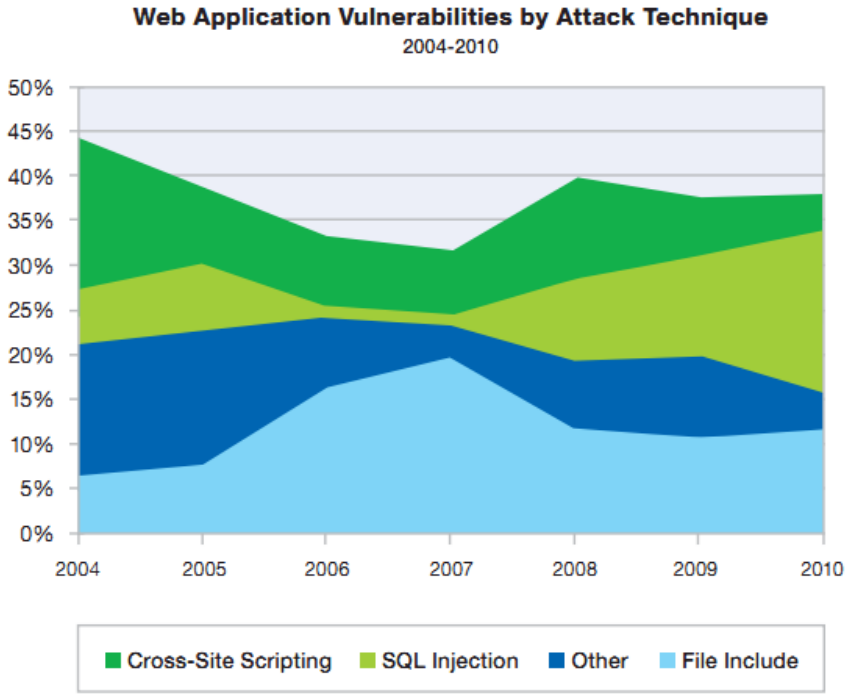
Методы реализации угроз

Какие методы использует злоумышленник при взломе веб-сайтов?

- Подбор паролей
- Уязвимости «нулевого» дня
- Тестирование функций (fuzzing)
- Поиск уязвимостей на основе анализа исходного кода, используемого приложения (включая его компоненты)
- Взлом сайта путем проведения атаки на другие сайты, расположенные на shared-хостинге
- Уязвимости в смежных компонентах системы
- Социальная инженерия / вредоносное ПО (угон адреса электронной почты, etc)

Зарубежная статистика

Распределение использования веб-уязвимостей по данным подразделения IBM X-Force



IBM X-Force, 2011,

http://public.dhe.ibm.com/common/ssi/ecm/en/wgl03007usen/WGL03007USEN.PDF

Типовые сценарии атак #1

Внедрение операторов SQL (MySQL) +

MD5(недостаточная строгость парольной политики) || отсутствие хеширования +

доступ к разделу администрирования сайтом +

возможность загрузки серверных расширений || редактирование файлов || ...

= profit1

Типовые сценарии атак #2

LFI + загрузка документов || доступ к логам || доступ к файлам сессии ... = profit11

open redirect + rn eq m || I eq l || vv eq w ... = фишинг

microsoft.com = rnicrosoft.com

mail.ru = MAIL.RU

Способы обнаружения уязвимостей в веб-приложениях

- **Тестирование функций**
  - Метод «черного-», «полосатого-» ящика
- **Фаззинг (fuzzing)**
- **Анализ исходного кода**
  - Статический/динамический/ручной анализ
- **Бинарный анализ приложения (binary analysis)**

А может WAF?

«Ложка дегтя в бочке меда»

- **За универсальность фильтров приходится расплачиваться ошибками первого и второго рода**
- **Не все фильтры одинаково полезны**
- **Ряд уязвимостей в веб-приложениях нельзя выявить сигнатурным путем**
- **Существуют универсальные методы обхода:**
  - **HTTP Parameter Pollution**
  - **HTTP Parameter Fragmentation**
  - **HTTP Parameter Contamination**
  - **коварные реплейсы (!)**
- **Появление уязвимостей 0day!**

Уязвимость уязвимости рознь!

http://seclists.org/fulldisclosure/2009/Aug/0113.html

WordPress is a state-of-the-art publishing platform with a focus on aesthetics, web stand
priceless at the same time. More simply, WordPress is what you use when you want to

III. DESCRIPTION

The way Wordpress handle a password reset looks like this: You submit your email adress or username via this form /wp-login.php?action=lostpassword ;
Wordpress send you a reset confirmation like that via email:

"
Someone has asked to reset the password for the following site and username. http://DOMAIN_NAME.TLD/wordpress
Username: admin
To reset your password visit the following address, otherwise just ignore this email and nothing will happen

http://DOMAIN_NAME.TLD/wordpress/wp-login.php?action=rp&key=o7naCKN3OoeU2KJMMsag "

You click on the link, and then Wordpress reset your admin password, and sends you over another email

Thursday, March 10, 2011

Gaining Administrative Privileges on any Blogger.com Account, 1337$ (Google Reward Program)

Неполный список администраторов такого приложения:

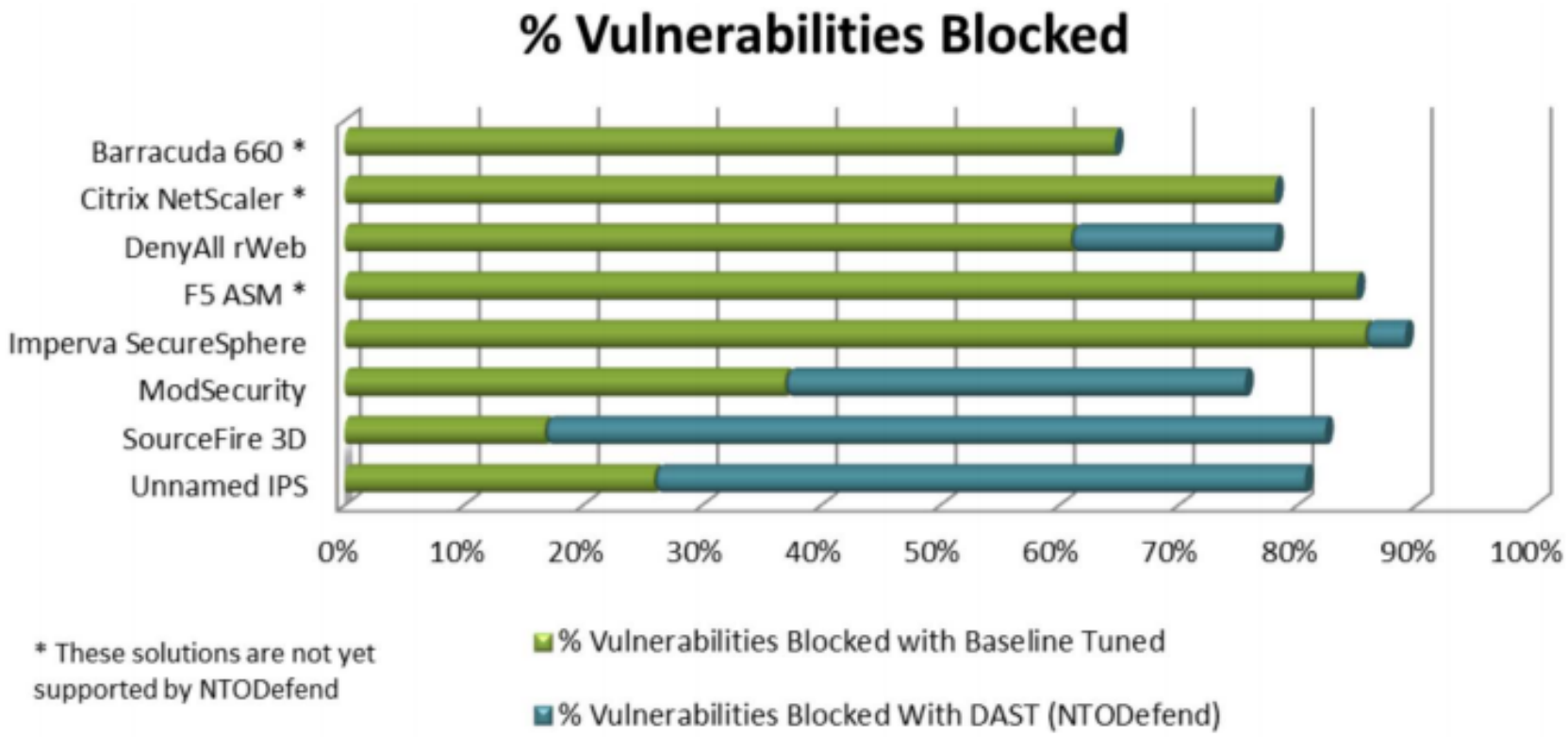
****admin, user**, r**t , …**

http://thedailywtf.com/Articles/Starring-The-Admin.aspx

```
if(strstr($username, '**')) {

    $admin = 1;
    $username = str_replace('**', '', $username);
    $_SESSION['admin'] = 1;

} else {

    $admin = 0;

}
```

Эффективность Web Application Firewall

- **Средняя эффективность по всем решениям составляет 79%**



Larry Suto, 2011, http://www.manvswebapp.com/wp-content/uploads/Analyzing_Effectiveness_of_Web_Application_Firewalls.pdf

Вместо заключения

Что нас ждет впереди?

- Перерождение известных уязвимостей
- Развитие комплексных и более сложных атак

Как подойти к вопросу безопасности веб-приложений?

Спасибо за внимание!
Вопросы?

devteev@ptsecurity.ru
http://devteev.blogspot.com/

POSITIVE TECHNOLOGIES