(не)сигурност
в Web приложенията

Максим Крижановски

Софтуерен инженер

Intro

- Защо всеки сайт е жертва?
  - Кражба на e-mail адреси
  - Разпространение на спам
  - Разпространение на вируси
  - Автоматизирани атаки

XSS (cross-site scripting)

- Позволява изпълнение на произволен JavaScript
- Кражба на сесия/cookie, подмяна на съдържание, пренасочване на потребителя
- Изпълнение на ajax заявка и кражба на резултата
- Осъществява се чрез нефилтриран HTML output, най-често в input, в резултати от търсене, и др.

XSS (cross-site scripting)

- Когато се извежда текст, да се използва htmlspecialchars
- Whitelist на тагове
- XSS е възможен и в атрибути!

```
$var = "javascript:alert('xss')";
<img src="<?php echo $var;">
```

- HTML Purifier for the rescue!

SQL Injection

- Данните в заявката трябва да се escape-ват!

```
$_GET['id'] = 1 UNION SELECT * from users

SELECT * FROM products WHERE id = $_GET['id']
```

- Дори и ако идват от базата данни!

```
$result = mysqli_query('...');

$row = mysqli_fetch_array($result);

$text = strip_tags($row['text']);

$sql = "UPDATE mytable SET myfield =

"$text";
```

- Prepared statement е универсалното решение

Context

$value = “some word\r\n
a: b”;

V Безопасно за SQL

V Безопасно за HTML

X SMTP header injection!

Не може да има универсално филтриране

CSRF (cross-site request forgery)

- Операции, променящи състоянието, не трябва да са GET
- GET заявка може да се осъществи от името на потребителя чрез прост HTML
- HTML-а може да е inject-нат в друг сайт посредством XSS
- POST заявка
- Security token (one time only)

Storing passwords

- Plain text does not work
- Simple hashing may seem better, but does not work either
- RAINBOW
- Salt passwords / use HMAC
- Use mcrypt extension

File uploads

- Не се доверявайте на $_FILES[‘name’]

Качването на файл е HTTP заявка и като такава, всички данни могат да бъдат манипулирани. Използвайте basename(), когато извличате името

- $_FILES[‘tmp_name’] е единсвения безопасен елемент
- Не позволявайте качването на произволни по тип файлове, или ги съхранявайте извън web root
- Внимавайте с презаписването на файлове!

Information disclosure

Всяка информация за системата може да помогне за проникването в нея

- expose_php = Off
- Turn off error reporting

mod_security

- Автоматично филтриране на заявки
- Работи чрез blacklist
- Може да се наложи да се настрои за конкретно приложение
- Няма универсално лекарство – може да помогне, но по-добре заложете на сигурния код

CORS (cross-origin resource sharing)

- AJAX не позволява достъп до друг домейн
- Нито JS манипулация на frame в друг домейн
- Освен ако няма Access-Control-Allow-Origin
- IE още не го поддържа
- Алтернативи (за IE): jsonp / proxy
- Content-security policy

(see http://people.mozilla.org/~bsterne/content-security-policy/details.html for details)

mod_rewrite

- Позволява един URL адрес да изглежда като друг
- Полезно за скриване на параметри
- Използва се за SEO
- Позволява забрана за достъп, базирано на условия
- Позволява redirect (също полезно при SEO)
- Базирано на регулярни изрази

mod_rewrite

```
RewriteRule (.*) index.php [L]
RewriteRule old.html new.html [R=301,L]
RewriteRule /news/([0-9]+)/(.*) news.php?id=$1

RewriteCond %{HTTP_REFERER} !domain.com
RewriteRule .* - [F]
```

Максим Крижановски
darhazer@gmail.com
http://linkedin.com/in/darhazer
@darhazer