Main Screen Turn On

Hands On Workshop

Introductions

Jack Skinner `@developerjack`

Katie McLaughlin `@glasnt`

And what about you?

(not your employer)

What’s your flavour?

PHP, Ruby, Python?
Wordpress, Drupal, Magento?

What the CVE?

CVE

Common **V**ulnerabilities and **E**xposures

- Unique database of vulnerabilities
- Identified by ID `CVE-YYYY-NNNNNN`
- https://cve.mitre.org

OWASP

Open **W**eb **A**pplication **S**ecurity **P**roject

- https://www.owasp.org

OWASP Top 10

- Most common attack vectors for Web

OWASP Top 10

Injection

Broken Authentication and Session Management

Cross-Site Scripting (XSS)

Insecure Direct Object References

Security Misconfiguration

Sensitive Data Exposure

Missing Function Level Access Control

Cross-Site Request Forgery (CSRF)

Using Components with Known Vulnerabilities

Unvalidated Redirects and Forwards

OWASP Top 10

The Big Three Covered in Hands On

Injection

Broken Authentication and Session Management

Cross-Site Scripting (XSS)

Insecure Direct Object References

Security Misconfiguration

Sensitive Data Exposure

Missing Function Level Access Control

Cross-Site Request Forgery (CSRF)

Using Components with Known Vulnerabilities

Unvalidated Redirects and Forwards

OWASP Top 10

Poor Authentication

Injection

Broken Authentication and Session Management

Cross-Site Scripting (XSS)

Insecure Direct Object References

Security Misconfiguration

Sensitive Data Exposure

Missing Function Level Access Control

Cross-Site Request Forgery (CSRF)

Using Components with Known Vulnerabilities

Unvalidated Redirects and Forwards

OWASP Top 10

Poor Authorisation

Injection

Broken Authentication and Session Management

Cross-Site Scripting (XSS)

Insecure Direct Object References

Security Misconfiguration

Sensitive Data Exposure

Missing Function Level Access Control

Cross-Site Request Forgery (CSRF)

Using Components with Known Vulnerabilities

Unvalidated Redirects and Forwards

OWASP Top 10

The cause of much sadness

Injection

Broken Authentication and Session Management

Cross-Site Scripting (XSS)

Insecure Direct Object References

Security Misconfiguration

Sensitive Data Exposure

Missing Function Level Access Control

Cross-Site Request Forgery (CSRF)

Using Components with Known Vulnerabilities

Unvalidated Redirects and Forwards

With CVEs come patches

Free fixes!

How long has it been since you patched?

In the last 6 months…

Multiple Content Management System* CVEs

- Wordpress 64
- Drupal 63
- Magento 7
- Joomla 1

*includes plugins

cve.mitre.com

In the last 6 months…

Multiple Framework CVEs

- Django 9
- Ruby on Rails 4
- Symfony 1

cve.mitre.com

In the last 6 months…

Multiple Language CVEs

- PHP 41
- Perl 6
- Python 4
- Ruby 2

cve.mitre.com

It’s not just your framework

In the last 12 months…

Multiple Major Operating System CVEs

CVE-2014-6271
CVE-2015-0204
CVE-2015-3456

In the last 12 months…

… also known as …

Logos bring recognition

Don’t forget your web servers!

Web Servers

CVEs recorded against:

- Nginx
- Apache
- IIS

cve.mitre.com

It’s not just your external stack

CRMs and Ticket Systems

CVEs recorded against:

- SugarCRM
- JIRA
- Confluence
- Request Tracker

and many more...

cve.mitre.com

But...

There’s probably even more issues

Responsible Disclosure is optional

Deprecated CMS Versions

Are you running unpatched applications?

Example: Wordpress

Wordpress 2.x?

Wordpress 2.x?

wat?
Upgrade!!

Wordpress 3.x?

Wordpress 3.x?

Upgrade!

Wordpress 4.0?

Upgrade

Wordpress 4.1.1?

Upgrade

Wordpress 4.2?

Upgrade

Wordpress 4.2.2?

Keep patches up to date

Deprecated Language Versions

Are you running unpatched languages?

PHP

What version are you running?

PHP 5.0, 5.1, 5.2?

PHP 5.0, 5.1, 5.2?

Upgrade
Unmaintained

php.net/eol.php

PHP 5.3?

PHP 5.3?

Upgrade
Unmaintained

php.net/eol.php

PHP 5.4, 5.5, 5.6?

PHP 5.4, 5.5, 5.6?

Keep patches up to date

Ruby

What version are you running?

Ruby 1.8?

Ruby 1.8?

Upgrade
Unmaintained

Ruby 1.9.3?

Ruby 1.9.3?

Upgrade
Unmaintained

Ruby 2.x?

Ruby 2.x?

Keep patches up to date

```
gem install bundle-audit
```

So just how much DON’T we patch?

Surely it’s not *that* bad... right?

Estimated Insecure Installations

PHP …%
Python …%
Perl …%

Drupal …%
Wordpress …%

Take a wild guess.

Estimated Insecure Installations

PHP	74%
Python	22%
Perl	18%
Drupal	55%
Wordpress	40%

Data from W3Techs,
correlated by known linux
distribution supposed numbers

blog.ircmaxell.com/2014/12/php-install-statistics.html

But but but...

“It’s too hard!”

“Upgrades are hard!”
“Patching is boring!”

No it’s not.

It’s easier than losing your business

“That won’t happen to me”

In the last 18 months…

“I’m not big enough to be a target”

‘**Security through Obscurity**’ in the real world of professional IT is over

Eben Moglen, January 13, 2015

youtube.com/watch?v=OAKRdZLU7AU

So what can you do?

Patch your shit.

… and …

Know the enemy

Let’s do some hacking!

Repeat after me…

I _____ do solemnly and sincerely and truly declare and affirm...

Repeat after me…

… to use the following knowledge for good and not evil…

Repeat after me…

… for learning and not profit.

And agree that if I break my oath
I will…

Repeat after me…

… rebuild wordpress.com using nothing but Fortran, HTML tables and transparent GIFs.

Let’s do some hacking!

joesbaconemporium.com

canhazwifi
yesyoucan