



Securing Drupal

Defense Against the Dark Arts

Presented By

André Van Klaveren / @opratr



André Van Klaveren, CISSP, GSSP

- 20+ Years in Professional IT
- Building Drupal sites since 2005
- Senior Solution Architect
- Application Security Specialist
- AppSec Lead @ USDA Rural Development



“Why Should We Care?”

“I only run a blog site, I’ve got nothing a hacker would want.”

“We don’t collect sensitive information on our site so we shouldn’t attract hackers.”

“We’re not big enough to worry about being hacked.”

“Security is not our responsibility, that’s what our security team is for.”



Why You Should Care

“There are two types of companies: those who have been hacked, and those who don’t yet know they have been hacked.”

- John Chambers, Chairman and CEO of Cisco

Cybersecurity incidents involving U.S. government agencies jumped 35% between 2010 and 2013.

- GAO-14-354, a report to congressional requesters

100% of business networks have traffic going to websites that host malware

- Cisco 2015 Annual Security Report

“[T]here are only two types of companies: those that have been hacked and those that will be. And even they are converging into one category: companies that have been hacked and will be hacked again.”

- Robert S. Mueller III, Director, FBI, RSA Cyber Security Conference San Francisco, CA (Mar. 1, 2012)



Criminal Motivations

Thrill / Challenge

Politics

Vigilantism

Idealism

Terrorism

Financial Gain

Religion

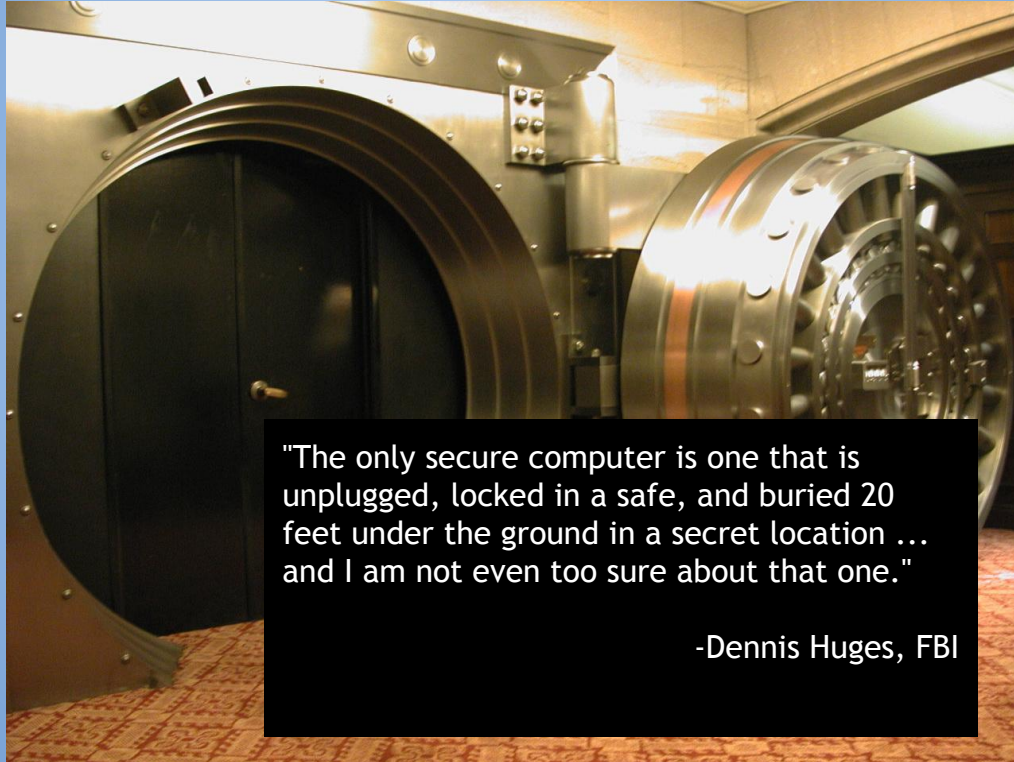
Cyber Warfare

Espionage

There are as many motivations for cyber-crime as there are cyber-criminals!



What is Security?



"The only secure computer is one that is unplugged, locked in a safe, and buried 20 feet under the ground in a secret location ... and I am not even too sure about that one."

-Dennis Huges, FBI



What is Security?

- Security is a process
- Security is hard
- Involves technology and people
- Putting up barriers (Defense-in-depth)
- There is no “Silver Bullet”
- It’s all about Risk Management



Defense-in-Depth

- Multiple layers of security controls (defenses)
- Provides redundancy in the event a security control fails
- ...but, you must weigh the *cost* of the control against the *benefit*
 - Budget
 - System performance
 - User experience



Layers of a Drupal System

- Application (Drupal)
- Services (Apache, MySQL, Varnish, Redis,...)
- Server OS (Red Hat, Ubuntu, etc.)
- Network (Provider / Internet / CDN)
- Users (end users, admins, devs, ...)



Securing Drupal: Apply Updates!

- Drupal core
- Contributed modules (“contrib”)
- Libraries! (<https://libraries.io/>)
- Subscribe to project email lists
- Subscribe to Drupal Security email list

<https://www.drupal.org/security>



Securing Drupal: Authentication

- Enforce a strong password policy
(Passphrases)
- Single Sign On (SimpleSAMLPhp, LDAP,...)
- Strengthen login security
- Enforce session limits
- Enforce idle session logout
- Use Two-Factor Authentication (2FA)!



Multi-Factor Authentication

Factors of Authentication:

- Something you know
- Something you have
- Something you are
- Some place you are
- ...

```
login: andre  
password: xxxxxxxxx  
[andre@serenity ~]$
```



Two-Factor Authentication (2FA)

- Uses two factors for authentication
- Enabled by the TFA module
- Checks for something you *have*
- Pluggable
- TFA basic plugins module
 - Time-Based One-Time Password (TOTP)
 - FreeOTP
 - Google Authenticator
 - Authy
 - ...
 - SMS login codes via Twilio
 - Trusted device



<https://groups.drupal.org/node/439868>



Securing Drupal: Input Filters

- Don't use PHP filter
 - Removed in Drupal 8, for good reason!
- Be careful with Full HTML

<input checked="" type="checkbox"/>	Path	7.38	Allows users to rename URLs. Required by: Pathauto (enabled)
<input type="checkbox"/>	PHP Filter	7.38	Allows embedded PHP code/snippets to be evaluated.
<input type="checkbox"/>	Full	7.38	Allows your site to capture votes on different topics in the fo
<input checked="" type="checkbox"/>	RDF	7.38	Enriches your content with metadata to let other application understand its relationships and attributes.



Securing Drupal: Security Kit

- Cross-Site Scripting
 - Content-Security-Policy header
 - A policy framework that enables specifying trustworthy sources of content and to restrict its capabilities.
 - script-src, object-src, img-src, style-src, ...
 - X-XSS-PROTECTION header
 - Controls internal XSS filters in modern browsers



Securing Drupal: Security Kit (2)

- Clickjacking
 - X-Frame-Options
 - SameOrigin
 - Deny
 - Allow-From
- SSL/TLS
 - HTTP Strict Transport Security (HSTS)



Securing Drupal: Security Review

- Use Security Review to report on common Drupal security issues
- Review reports regularly

Untrusted roles do not have administrative or trusted Drupal permissions.
✘ Base URL is not set in settings.php.
✘ Errors are written to the screen.
✘ PHP files in the Drupal files directory can be executed.
Dangerous tags were not found in any submitted content (fields).
Drupal installation files and directories (except required) are not writable by the server.



Review Roles and Permissions

- Principle of Least Privilege
- Consider blocking user 1 in Production
 - ...and any user with an ‘administer ...’ permission
- Regular audit of roles and permissions
 - Role Watchdog
 - Permission Watchdog



Contributed and Custom Modules

- Secure coding guidelines
- Look for well adopted and actively maintained modules

Project Information

Maintenance status: [Actively maintained](#)
Development status: [Under active development](#)
Module categories: Content , Spam Prevention
Reported installs: 74,371 sites currently report using this module. [View usage statistics.](#)
Downloads: 284,768
Automated tests: Enabled
Last modified: February 16, 2016

Maintainers for Honeygot

[geerlingguy](#) – 172 commits
last: 4 months ago, first: 4 years ago

All issues

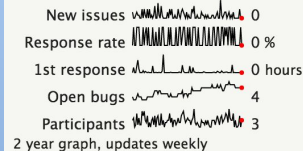
25 open, 233 total

Bug report

4 open, 100 total

[Subscribe via e-mail](#)

Statistics



Downloads

Recommended releases

Version	Download	Date
8.x-1.23	tar.gz (19.98 KB) zip (29.7 KB)	2016-Mar-11
7.x-1.22	tar.gz (18.38 KB) zip (22.07 KB)	2016-Mar-11

Development releases

Version	Download	Date
8.x-1.x-dev	tar.gz (19.92 KB) zip (29.58 KB)	2016-Jun-29
7.x-1.x-dev	tar.gz (18.39 KB) zip (22.08 KB)	2016-Mar-11



OWASP Top 10 (2013)




- A1 Injection
- A2 Broken Authentication and Session Management
- A3 Cross-Site Scripting (XSS)
- A4 Insecure Direct Object References
- A5 Security Misconfiguration
- A6 Sensitive Data Exposure
- A7 Missing Function Level Access Control
- A8 Cross-Site Request Forgery (CSRF)
- A9 Using Components with Known Vulnerabilities
- A10 Unvalidated Redirects and Forwards

<https://www.owasp.org>



Content Distribution Networks

- Akamai
 - Amazon Cloudfront
 - Cloudflare
 - Limelight
 - MaxCDN
 - ...
- 
- Geo-local content distribution
 - Content optimization
 - Analytics
 - IPv6
 - Distributed Denial of Service (DDoS) protection
 - Web Application Firewall
 - SQL Injection
 - SPAM
 - XSS
 - SSL/TLS
 - IP Based Traffic Blocking
 - Visitor reputation



Consider Managed Drupal Hosting

- Acquia
- Pantheon
- Blackmesh
- Platform.sh
- USDA EAS
- ...

ACQUIA™



BLACKMESH



platform 



Securing the User

“YOU are the weakest link!”

- Password Management (LastPass!)
- Phishing
- Online Hygiene (Bad Habits)
- Malware vector
- Change their behavior!



Security Awareness Training

- Awareness changes human behavior
- Topics:
 - Phishing
 - Poor password security/management
 - Sharing too much on Social Media
 - Data loss/exposure
 - Malware infection vectors*
 - ...



Security Awareness Impact

- First phish: 30-60% fall victim
- 6-12 months later: Low as 5%

The more often the training, the more effective the impact.

- Quarterly: 19%
- Every other month: 12%
- Monthly: 5%



Summary & Take-aways

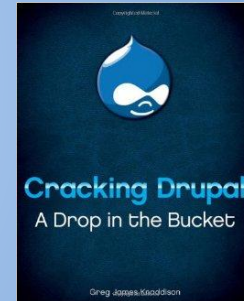
- Security needs to be a first-class requirement
- The user is often the weakest link in system security
- Practice the Principle of Least Privilege
- Monitor and review roles and permissions
- Patch and update quickly and often!
- Drupal core is secure, but can be made to be insecure
- Use well adopted and maintained contrib modules
- Limit custom module development as much as possible...
- ... and if you must write custom code, follow [Secure Coding Guidelines](#)

Defense-in-depth can mitigate the impact of a security incident.



References

- <https://www.drupal.org/security>
- <https://www.drupal.org/security/secure-configuration>
- <https://www.drupal.org/security-team>
- <http://www.drupalsecurityreport.org>
- <http://crackingdrupal.com/>
- <https://www.owasp.org>
- <http://www.securingthehuman.org/>
- <http://security-compass.myshopify.com/>
- <https://letsencrypt.org/>
- <https://www.drupal.org/project/seckit>
- https://www.drupal.org/project/security_review





Questions?





Thank You!

André Van Klaveren / @opratr

