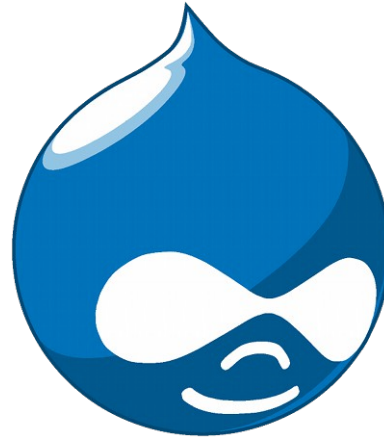


Keeping your Drupal Site Secure



Stéphane Corlosquet -
scorlosquet@gmail.com
Drupal Nights @ BioRAFT
Aug 15th, 2013

ACQUIA™

General tips

- Use HTTPS, SSH, SFTP
- Strong password policy
- Server – LAMP stack
- Require SSH keys
- Keep your site settings secure
 - Permissions
 - Text formats
 - PHP filter

Drupal 7

- Stronger password hashing / salt
- Login flood control
 - prevents brute-force credential guessing
- Protected cron
 - prevents Denial of Service attacks
- Update manager
 - Update module from the web UI

Modules enhancing security

- Secure login
- Password policy
- Paranoia
- Hacked!
- Permissions Lock

Drupal specific hosting

- Can your hosting provider help you improve your security process?
 - [Insight](#) (part of Acquia Cloud hosting)
 - [Pantheon](#) (self-service security updates)
- Tuned for Drupal security (and performance)
 - Code, DB, uploaded files, config
 - Managed security updates:
 - Remote administration (Acquia hosting)

Security process

- Ongoing maintenance
- Cost
- Managed hosting
- Drupal.org packaging infrastructure

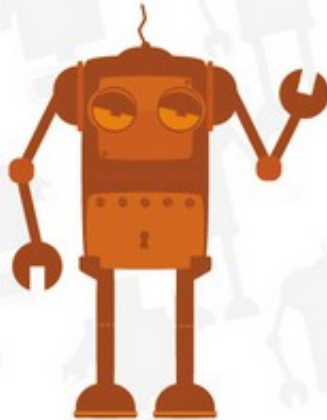
Security process

- **Drupal Security Team**
 - Keep Drupal code secure in core and contrib
 - Educate the community on security best practices
 - Developers
 - Site builders
 - Site administrators and users
 - Decision makers
 - **Security Advisory** for every security release

Security process

Security Team

A global group of some of the world's leading web security experts, always on-call to assess, evaluate, and address issues affecting Drupal's security.



Project maintainers

Drupal's active developer community is more than 15,000 strong and includes experts in all areas of today's web and its technologies. Different maintainers are responsible for different plug-in modules and Drupal's core.



Drupal users

More than 700,000 people, running more than 1 million websites, use, test, and improve Drupal on a daily basis. New vulnerabilities are quickly identified and confidentially reported to the Drupal security team.



1. Vulnerability in code discovered.

2. Issue reported privately to Security Team.

3. Issue reviewed, potential

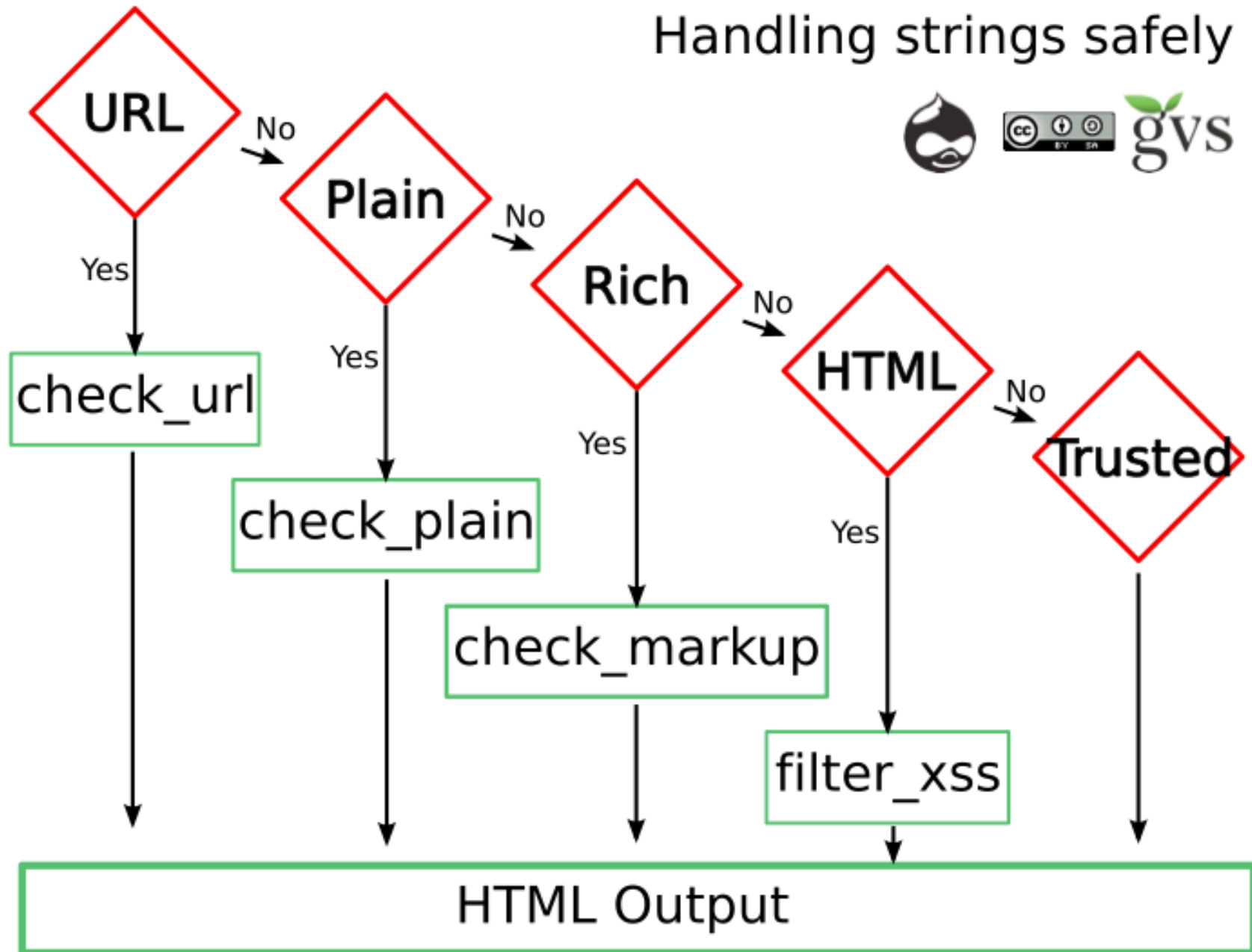


Developers & site maintainers

- Follow Drupal APIs and best practices
- Take & **verify** backups
- **Sanitize backups** for sharing

Cross Site Scripting

Handling strings safely



Drupal 8

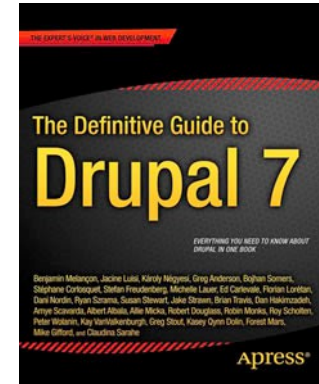
- Twig as templating language
 - Automatically sanitizes strings on output
 - No PHP in templates
- WYSIWYG in core
 - Streamlined filter mechanism (server and client side)
 - No more full HTML as last resort
- Local image input filter
 - Only allow images from same site

Book on Security in Drupal



References

- DGD7 chapter 6
- <http://drupal.org/security>
- <http://www.drupalscout.com/>
- <http://groups.drupal.org/best-practices-drupal-security>



Thanks!

- Stéphane Corlosquet:
 - scorlosquet@gmail.com
 - [@scorlosquet](#)
 - <http://openspring.net/>