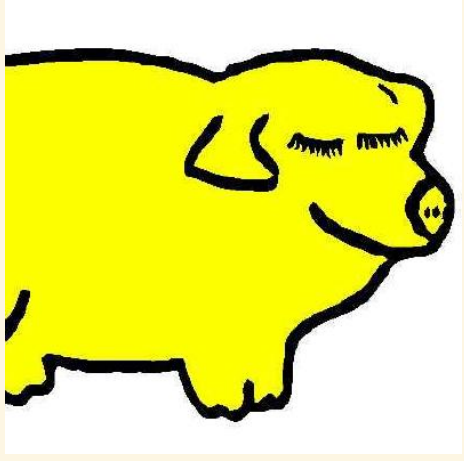
SECURITY IN DRUPAL: WHAT CAN GO WRONG?

Benji Fisher

March 15, 2024 - DrupalCamp NJ

INTRODUCTION

ABOUT ME



- Benji Fisher
- @benjifisher on d.o
- @benjifisher on GitHub
- @benjifisher on GitLab

Usability group, Migration subsystem, Security team

FOLLOW ALONG

- https://slides.benjifisher.info/ (GitLab Pages)

OUTLINE

- Introduction
- What is the OWASP Top Ten?
- What is Drupal?
- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- …

OUTLINE (CONTINUED)

- …
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures
- A10:2021-Server-Side Request Forgery
- Conclusion

ATTRIBUTION

These slides borrow from some of Peter Wolanin's "Cracking Drupal" presentations and from https://owasp.org/. According to the standard footer,

> *Unless otherwise specified, all content on the site is Creative Commons Attribution-ShareAlike v4.0 and provided without warranty of service or accuracy.*

All of my slide decks have a similar license.

WHAT IS THE OWASP TOP TEN?

OPEN WEB APPLICATION SECURITY PROJECT® (OWASP)

The Open Web Application Security Project® (OWASP) is a nonprofit foundation that works to improve the security of software.

source: https://owasp.org/

OWASP is not Drupal-specific. Let's "get off the island"!

OWASP TOP TEN

The OWASP Top 10 is a standard awareness document for developers and web application security. It represents a broad consensus about the most critical security risks to web applications.

source: https://owasp.org/www-project-top-ten/

The list is updated every few years. The most recent version is from 2021.

WHAT IS DRUPAL?

DRUPAL: A CONTENT MANAGEMENT SYSTEM

Drupal is a web-based content management system (CMS):

> *Enter data in my forms. I will save it to the database, then generate web pages.*

Hacker:

DRUPAL: EXPLOITS OF A MOM

Hacker:

Name: Robert'); DROP TABLE Stude

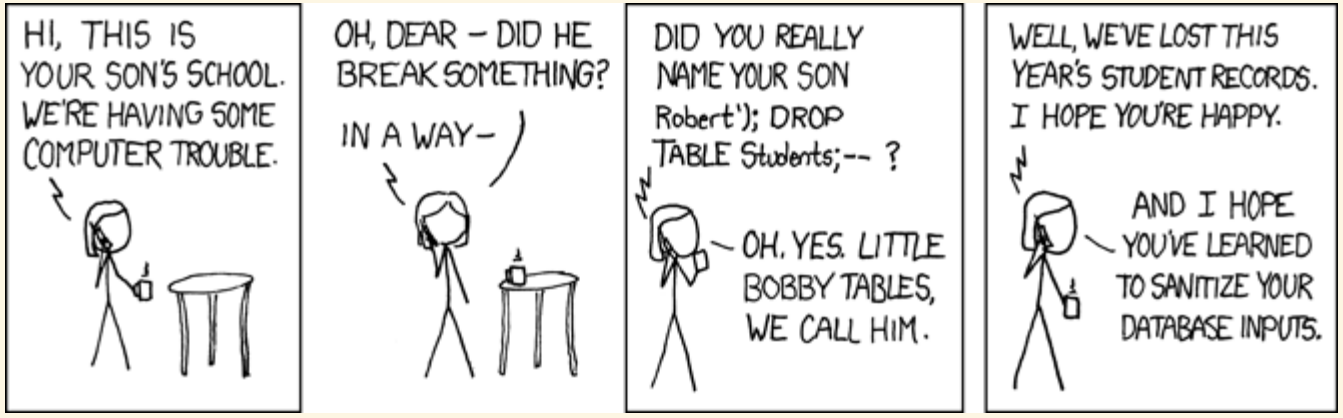
Then

```
$sql = "INSERT INTO Students (name) VALUES('$name')";
```

will become

```
$sql = "INSERT INTO Students (name) VALUES('Robert');
        DROP TABLE Students; --')";
```

XKCD 327 (EXPLOITS OF A MOM)



source: https://xkcd.com/327/

DRUPAL: AN ACTIVE, INTERNATIONAL OSS PROJECT

The Drupal community is one of the largest open source communities in the world. We're more than 1,000,000 passionate developers, designers, trainers, strategists, coordinators, editors, and sponsors working together.

source: https://www.drupal.org/about

DRUPAL: TAKE SECURITY SERIOUSLY

The security team is an all-volunteer group of individuals who work to improve the security of the Drupal project. Members of the team come from countries across 4 continents … The team was formalized in 2005 with a mailing list and has had 3 team leads in that time period.

source: https://security.drupal.org/team-members

A01:2021-BROKEN ACCESS CONTROL

TYPES OF VULNERABILITY

1. Information disclosure
2. Edit/Delete by unauthorized user
3. Cross-Site Request Forgery (CSRF)
4. … and more

HORROR STORIES (CUSTOM MODULES)

One site had custom access control for `/user/1/edit`. The access function left off a “not” and granted access to anyone *except* User 1.

Q: How to protect yourself?

CUSTOM MODULES

How do you avoid horror stories?

- Code review
- Automated tests for every custom page/custom access
- Avoid custom code!

If customers knew the true cost of custom code, they would ask for less of it.

CROSS-SITE REQUEST FORGERY (CSRF)

- mysite.com: `<img src="https://example.com/node/123/delet`
- admin for example.com visits mysite.com

Questions:

- Why does this attack not work?
- What incorrect assumptions expose a similar vulnerability?

HOW TO AVOID CSRF

- Use confirmation forms.
- Expect users to tweak URLs: `.../edit`, `.../delete`
  and more.
- Do not assume form requests come from the form you
  created.
- Use CSRF tokens: CSRF access checking
- Remember: forms are hard! Web CMS is a terrible idea

CSRF IN DRUPAL

SA-CORE-2021-006:

- **Project**: Drupal core
- **Date**: 2021-September-15
- **Security risk**: Moderately critical 10⁄25 AC:Basic/A:User/CI:None/II:Some/E:Theoretical/TD:D
- **Vulnerability**: Cross Site Request Forgery
- **CVE IDs**: CVE-2020-13673

SA-CORE-2021-006 (CONTINUED)

The Drupal core Media module allows embedding internal and external media in content fields. In certain circumstances, the filter could allow an unprivileged user to inject HTML into a page when it is accessed by a trusted user with permission to embed media. In some cases, this could lead to cross-site scripting.

SA-CORE-2021-006 (FIX)

Solution: upgrade to Drupal 9.2.6, 9.1.13, or 8.9.19.

Commit b230624e5b:

- When editing a WYSIWYG field with a Media embed, add a CSRF token to the header of the jQuery request.
- Validate the token in the code that responds to that request.

A02:2021-CRYPTOGRAPHIC FAILURES

CRYPTOGRAPHY: WHAT GOES WRONG?

- data transmitted in clear text
- old or weak cryptographic algorithms or protocols
- encryption not enforced
- deprecated hash functions such as MD5 or SHA1

source: A02:2021 - Cryptographic Failures

KEEP IT SIMPLE

Unless you are a cryptography maven, do not try to do it yourself. Know when to call for an expert!

Q: What data need protection?

WHAT TO PROTECT?

- Passwords
- API keys
- Personally identifiable information (PII)
- Business secrets

PII includes Social Security numbers, credit cards, health information.

HTTP AND HTTPS

- Do not manage your own servers unless that is your business.
- HTTPS provides encryption and authentication.
- Enforce HTTPS: Strict-Transport-Security header (HSTS).
  - CDN (Cloudflare, Fastly, …)
  - Server
  - Security Kit module
- SSL is insecure. Use TLS 1.2+

SSL LABS

For example, drupal.org SSL Labs report

https://www.ssllabs.com/ssltest/analyze.html?d=drupal.org&s=151.101.194.217

Qualys. SSL Labs

Home Projects Qualys Free Trial Contact

You are here: Home > Projects > SSL Server Test > drupal.org > 151.101.194.217

SSL Report: drupal.org (151.101.194.217)

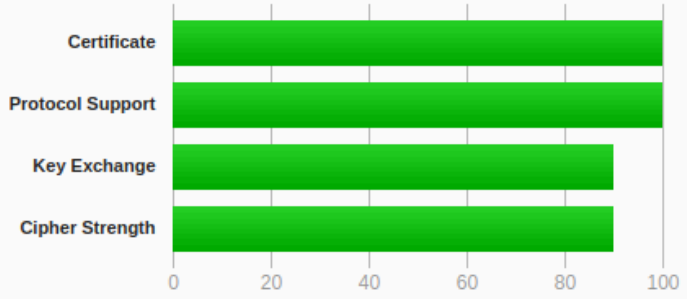
Assessed on: Sat, 19 Nov 2022 00:20:30 UTC | Hide | Clear cache

Scan Another »

Summary

Overall Rating

A+



Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.

This site works only in browsers with SNI support.

HTTP Strict Transport Security (HSTS) with long duration deployed on this server. MORE INFO »

DRUPAL UPDATE INFORMATION

… not to be confused with Automatic Updates/Project Browser initiative

In `Drupal\update\UpdateFetcher`:

```
/**
 * URL to check for updates, if a given project doesn't define
 *      its own.
 */
const UPDATE_DEFAULT_URL =
      'https://updates.drupal.org/release-history';
```

We did not fix that until Issue #1538118 2020-11-05.
Drupal 7 was fixed 2023-06-06.

API KEYS

- Use API keys for external services: SMTP, translation, ..
- Like long passwords
- Do not commit to your repository.
  - If you do, look up BFG Git (Big Friendly Giant).
- Securing Authentication Credentials in the "Security in Drupal" guide
  - Use environment variables.
  - Add `settings.php` value from external file.
  - Use contributed modules for key management.

PASSWORDS IN DRUPAL

- User management is one of Drupal's strengths. Do not "roll your own".
- Rule #1 (of many): do not store passwords in the database (nor the file system).
  - Store hashed (or encrypted) passwords.
- Rule #2: Make it secure even with the hashed passwords.
  - Add "salt" before hashing.
  - Use an expensive hash function.

PASSWORDS IN DRUPAL 10.0

In `core.services.yml`:

```yaml
# The argument to the hashing service defined in services.yml,
# to the constructor of PhpassHashedPassword is the log2
# number of iterations for password stretching.
# @todo increase by 1 every Drupal version in order to
# counteract increases in the speed and power of computers
# available to crack the hashes. The current password hashing
# method was introduced in Drupal 7 with a log2 count of 15.
password:
  class: Drupal\Core\Password\PhpassHashedPassword
  arguments: [16]
```

PASSWORDS IN DRUPAL 10.1+

In `core.services.yml`:

```
  password:
    class: Drupal\Core\Password\PhpPassword
```

- The comment is gone.
- The parameter is gone.
- See Password hashing is changed (change record).
- See Password Compatibility module (docs for core module)

A03:2021-INJECTION

INJECTION: WHAT GOES WRONG

- User-supplied data is not validated, filtered, or sanitized by the application.
- Dynamic queries … are used directly in the interpreter.
- Hostile data is used within … search parameters to extract additional, sensitive records.
- Hostile data is directly used or concatenated. …

source: A03:2021 - Injection

INJECTION IN DRUPAL: SA-CORE-2014-005

Drupal 7 includes a database abstraction API to ensure that queries executed against the database are sanitized to prevent SQL injection attacks.

A vulnerability in this API allows an attacker to send specially crafted requests resulting in arbitrary SQL execution. … this can lead to privilege escalation, arbitrary PHP execution, or other attacks.

This … can be exploited by anonymous users.

source: SA-CORE-2014-005

INJECTION: MY RESPONSE

Because of the severity of the vulnerability and the simplicity of the update, we tested … and updated the site today.

source: my e-mail to boss and site owner (paraphrase)

INJECTION: THE UPDATE

VULNERABLE CODE

```
foreach ($data as $i => $value) {
  $new_keys[$key . '_' . $i] = $value;
}
```

FIXED CODE

```
foreach (array_values($data) as $i => $value) {
  $new_keys[$key . '_' . $i] = $value;
}
```

(comment snipped from both)

INJECTION: THE NEXT STEP

```
// Update the query with the new placeholders.
// preg_replace is necessary to ensure the replacement
   does not affect
// placeholders that start with the same exact text. For
   example, if the
// query contains the placeholders :foo and :foobar, and
   :foo has an
// array of values, using str_replace would affect both
   placeholders,
// but using the following preg_replace would only affect
   :foo because
// it is followed by a non-word character.
$query = preg_replace(
  '#' . $key . '\b#',
  implode(', ', array_keys($new_keys)),
```

(line breaks added)

A04:2021-INSECURE DESIGN

A05:2021-SECURITY MISCONFIGURATION

A06:2021-VULNERABLE AND OUTDATED COMPONENTS

THE BEST KEPT SECRET IN WEB SECURITY

The secret:

The most important thing is to do all the boring stuff you *already know*.

It is a lot like …

CLICK BAIT?

How to live a longer, healthier life!

It takes just 4 minutes a day!

Does that seem too good to be true?

BRUSH YOUR TEETH!

- Two minutes, two times a day.
- Best advice you will get today.
- Also floss.
- You really will live a longer, healthier life.

WEB SECURITY HYGIENE

- Use good passwords. Have a policy.
- Keep your software up to date.
- Unless hosting is your core business, do not run your own servers.

DRUPAL: KNOW THE SCHEDULE

- Security release windows: Wednesdays 12-5 ET
- Drupal core updates (patch versions): third Wednesdays
- Drupal core updates (minor versions): June and December
- Minor versions are supported for one year.

DRUPAL: KNOW THE CHANNELS

- Web: Security advisories
- RSS: https://drupal.org/security/rss.xml, https://drupal.org/security/contrib/rss.xml, https://drupal.org/security/psa/rss.xml
- Email: https://www.drupal.org/user (Edit > My newsletters)
- Slack: `#security-team` channel in Drupal Slack

Unofficial: @drupalsecurity on X, Mastodon

DRUPAL: KNOW THE DIFFERENCE

- Major version (Drupal 9 to Drupal 10): disruptive
- Minor version (9.3 to 9.4): less disruptive, new features
- Patch version (9.3.6 to 9.3.7): should not be disruptive, bug fixes
- Security release (9.3.7 to 9.3.8): not disruptive (best effort)

DRUPAL: TRUST THE SECURITY TEAM

Two choices:

1. Read the SA, decide whether it impacts your site. If so, update.
2. Update your site.

Either way, you are trusting the security team:

1. They anticipated all the possible exploits.
2. The update is not disruptive.

DRUPAL AND SYMFONY

Q: Why was Drupal 9 EOL scheduled for Nov. 2023?

A: Drupal 9 used Symfony 4, which was EOL in Nov. 2023.

A07:2021-
IDENTIFICATION AND AUTHENTICATION FAILURES

A08:2021-SOFTWARE AND DATA INTEGRITY FAILURES

A09:2021-SECURITY LOGGING AND MONITORING FAILURES

A10:2021-SERVER-SIDE REQUEST FORGERY

CONCLUSION

SUMMARY

- Introduction
- What is the OWASP Top Ten?
- What is Drupal?
- A01:2021-Broken Access Control
- A02:2021-Cryptographic Failures
- A03:2021-Injection
- A04:2021-Insecure Design
- A05:2021-Security Misconfiguration
- …

SUMMARY (CONTINUED)

- …
- A06:2021-Vulnerable and Outdated Components
- A07:2021-Identification and Authentication Failures
- A08:2021-Software and Data Integrity Failures
- A09:2021-Security Logging and Monitoring Failures
- A10:2021-Server-Side Request Forgery
- Conclusion

REFERENCES

- Benji's slide decks and source files
- OWASP Top Ten and OWASP Top 10:2021
- Drupal Security Team
- Drupal core release cycle: major, minor, and patch releases
- Security advisories

CONTRIB MODULES

- Security Review: Check your site for misconfiguration
- Paranoia: No PHP `eval ( )` from the web interface
- Security Kit: Content Security Policy, Origin checks against CSRF, XSS
- Guardr: A Drupal distribution with security in mind
- Two-factor Authentication (TFA): Two-factor authentication for Drupal sites

QUESTIONS

COPYLEFT