#webappsec #yowwest15 https://joind.in/14594

Web Application Security

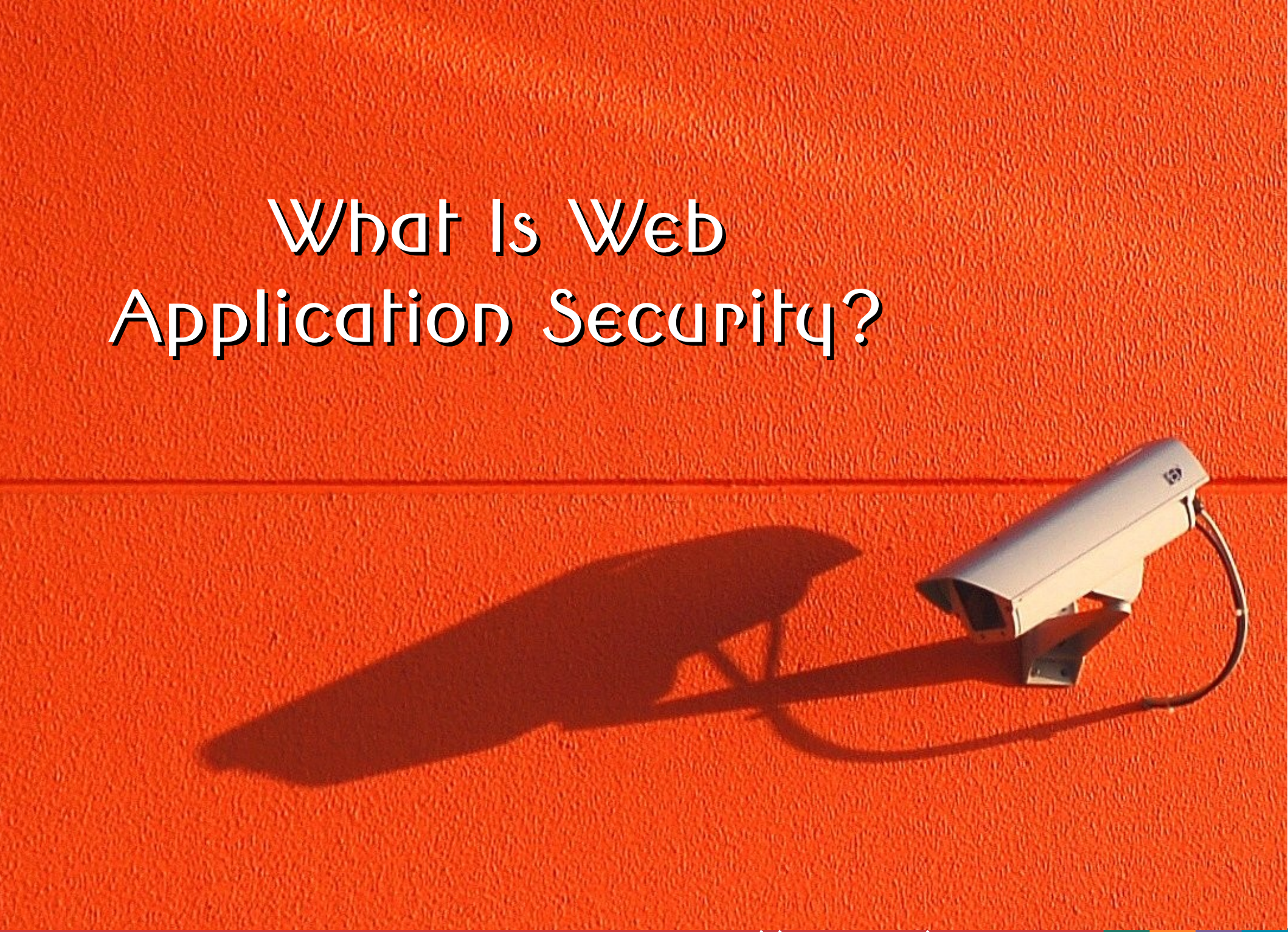
Winning When The Odds Are Against You

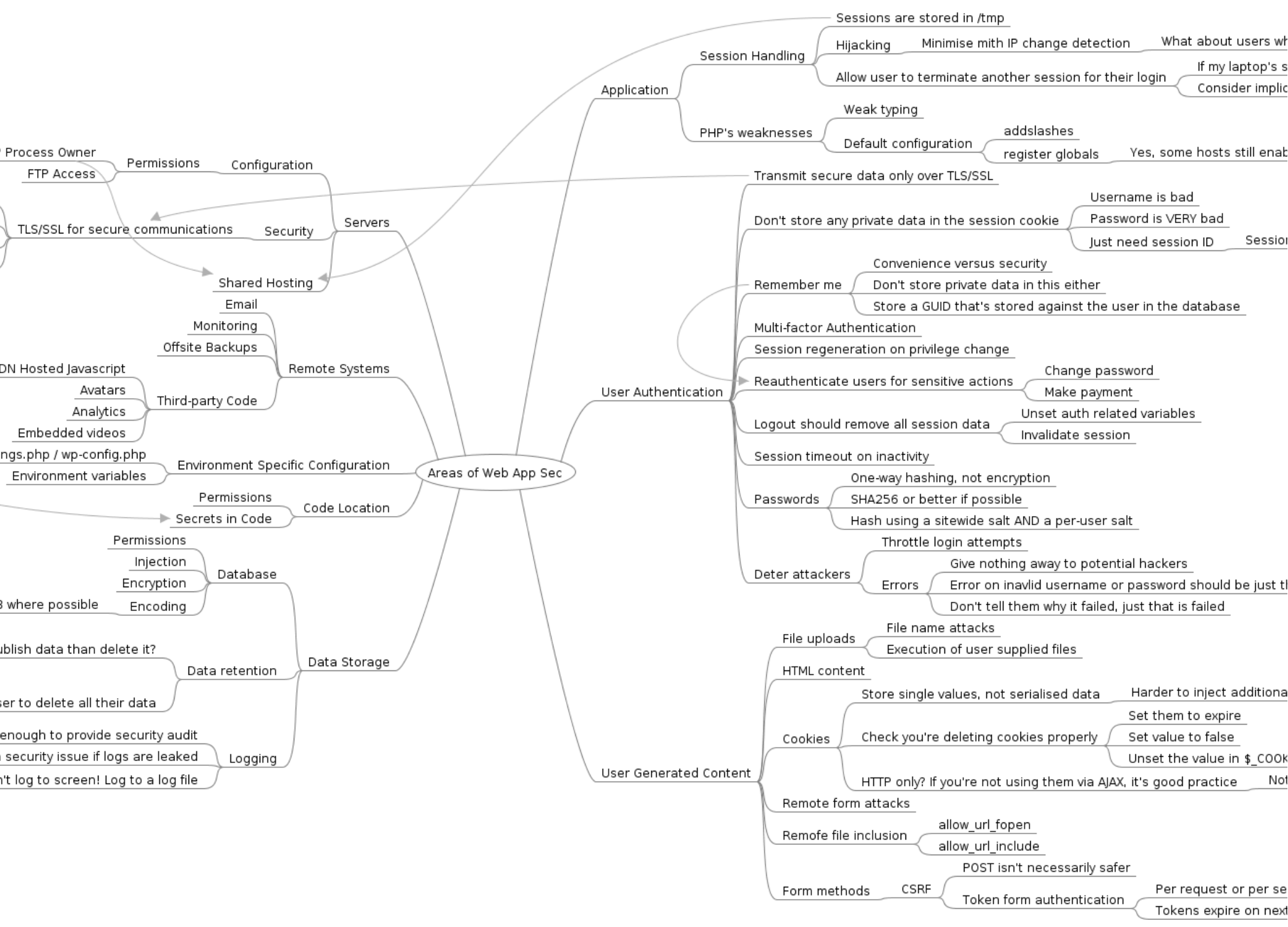
Ben Dechrai
@bendechrai

CTO FOR HIRE

YOW! West, Perth, 2015

What Is Web Application Security?




Areas of Web App Sec
Application
Session Handling
Sessions are stored in /tmp
Hijacking
Minimise mith IP change detection
What about users wh
Allow user to terminate another session for their login
If my laptop's s
Consider implic
PHP's weaknesses
Weak typing
Default configuration
addslashes
register globals
Yes, some hosts still enab
Transmit secure data only over TLS/SSL
User Authentication
Don't store any private data in the session cookie
Username is bad
Password is VERY bad
Just need session ID
Session
Remember me
Convenience versus security
Don't store private data in this either
Store a GUID that's stored against the user in the database
Multi-factor Authentication
Session regeneration on privilege change
Reauthenticate users for sensitive actions
Change password
Make payment
Logout should remove all session data
Unset auth related variables
Invalidate session
Session timeout on inactivity
Passwords
One-way hashing, not encryption
SHA256 or better if possible
Hash using a sitewide salt AND a per-user salt
Deter attackers
Throttle login attempts
Errors
Give nothing away to potential hackers
Error on inavlid username or password should be just t
Don't tell them why it failed, just that is failed
User Generated Content
File uploads
File name attacks
Execution of user supplied files
HTML content
Cookies
Store single values, not serialised data
Harder to inject additiona
Check you're deleting cookies properly
Set them to expire
Set value to false
Unset the value in $_COOK
HTTP only? If you're not using them via AJAX, it's good practice
Not
Remote form attacks
Remofe file inclusion
allow_url_fopen
allow_url_include
Form methods
CSRF
POST isn't necessarily safer
Token form authentication
Per request or per se
Tokens expire on next
Servers
Configuration
Permissions
Process Owner
FTP Access
Security
TLS/SSL for secure communications
Shared Hosting
Remote Systems
Email
Monitoring
Offsite Backups
Third-party Code
DN Hosted Javascript
Avatars
Analytics
Embedded videos
Environment Specific Configuration
ngs.php / wp-config.php
Environment variables
Code Location
Permissions
Secrets in Code
Data Storage
Database
Permissions
Injection
Encryption
Encoding
3 where possible
Data retention
ublish data than delete it?
ser to delete all their data
Logging
enough to provide security audit
security issue if logs are leaked
n't log to screen! Log to a log file

Where to Start?

owasp.org

OWASP

Open Web Application
Security Project

OWASP
Open Web Application Security Project

Top Ten Cheat Sheet

Injection	Cross Site Scripting
Weak authentication & session management	Insecure Direct Object Reference
Cross Site Request Forgery	Security Misconfiguration
Insufficient Cryptographic Storage	Failure to Restrict URL access
Insufficient Transport Layer Protection	Unvalidated Redirects and Forwards

OWASP
Open Web Application Security Project

Top Ten Cheat Sheet

Injection	Cross Site Scripting
Weak authentication & session management	Insecure Direct Object Reference
Cross Site Request Forgery	Security Misconfiguration
Insufficient Cryptographic Storage	Failure to Restrict URL access
Insufficient Transport Layer Protection	Unvalidated Redirects and Forwards

Demo

Solutions?

Think like your favourite programming language...

Not in your favourite programming language...

Think LIKE your favourite programming language...

HTTP

GET /index.html →

← <html>..</html>

GET /css/styles.css →

GET /js/script.js →

GET /images/logo.jpg →

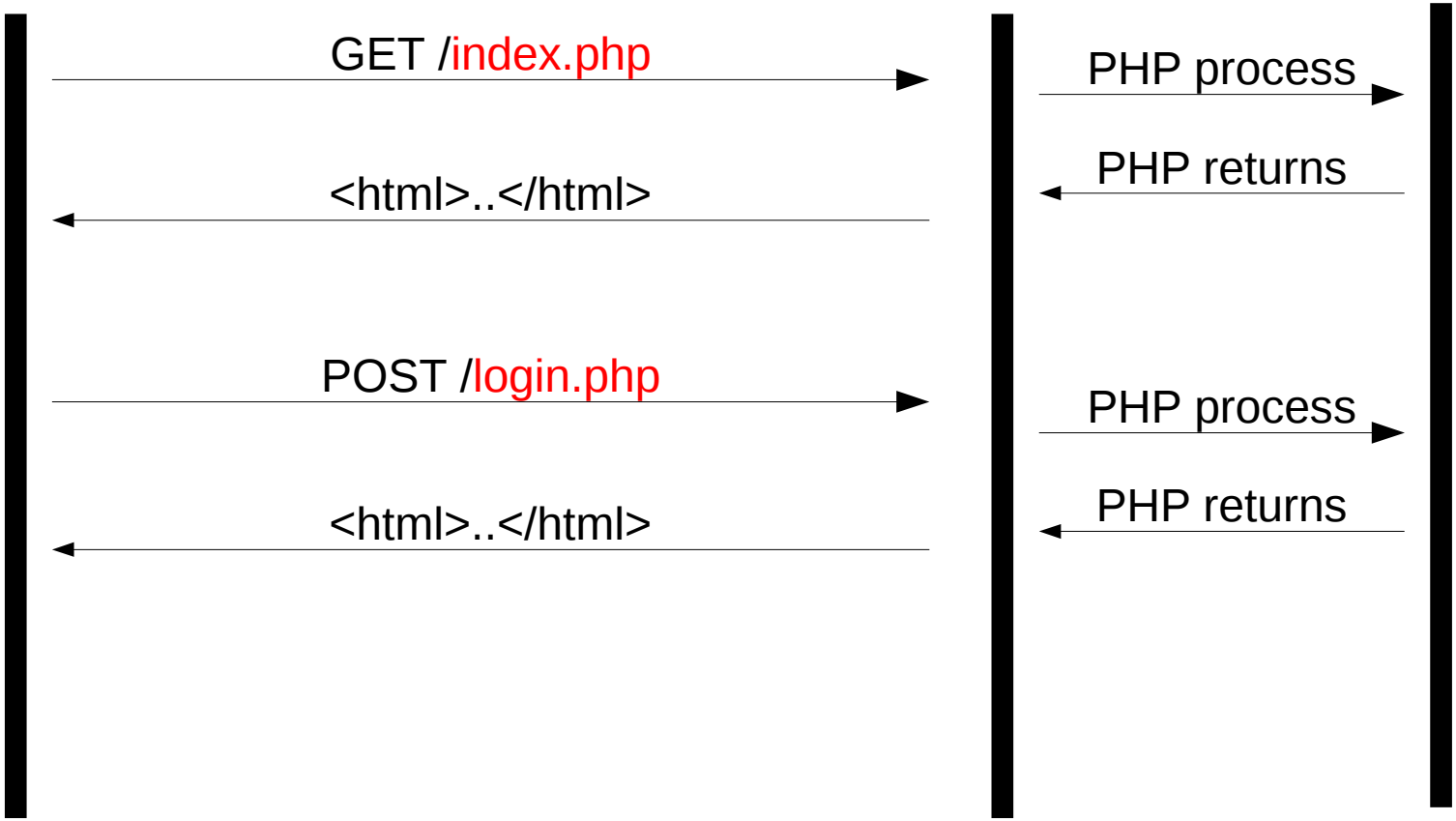
← body { ... }

← $(document).ready(...)

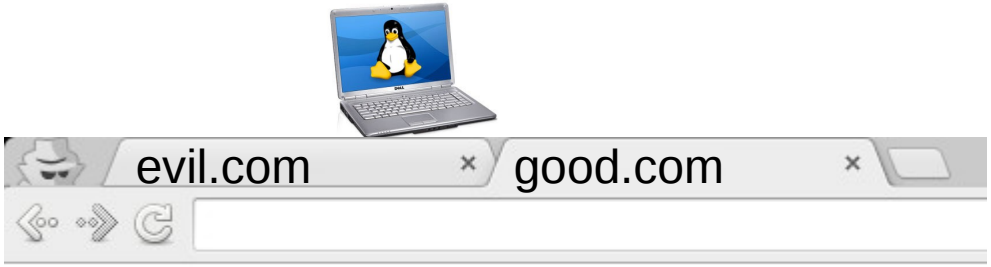
← ...

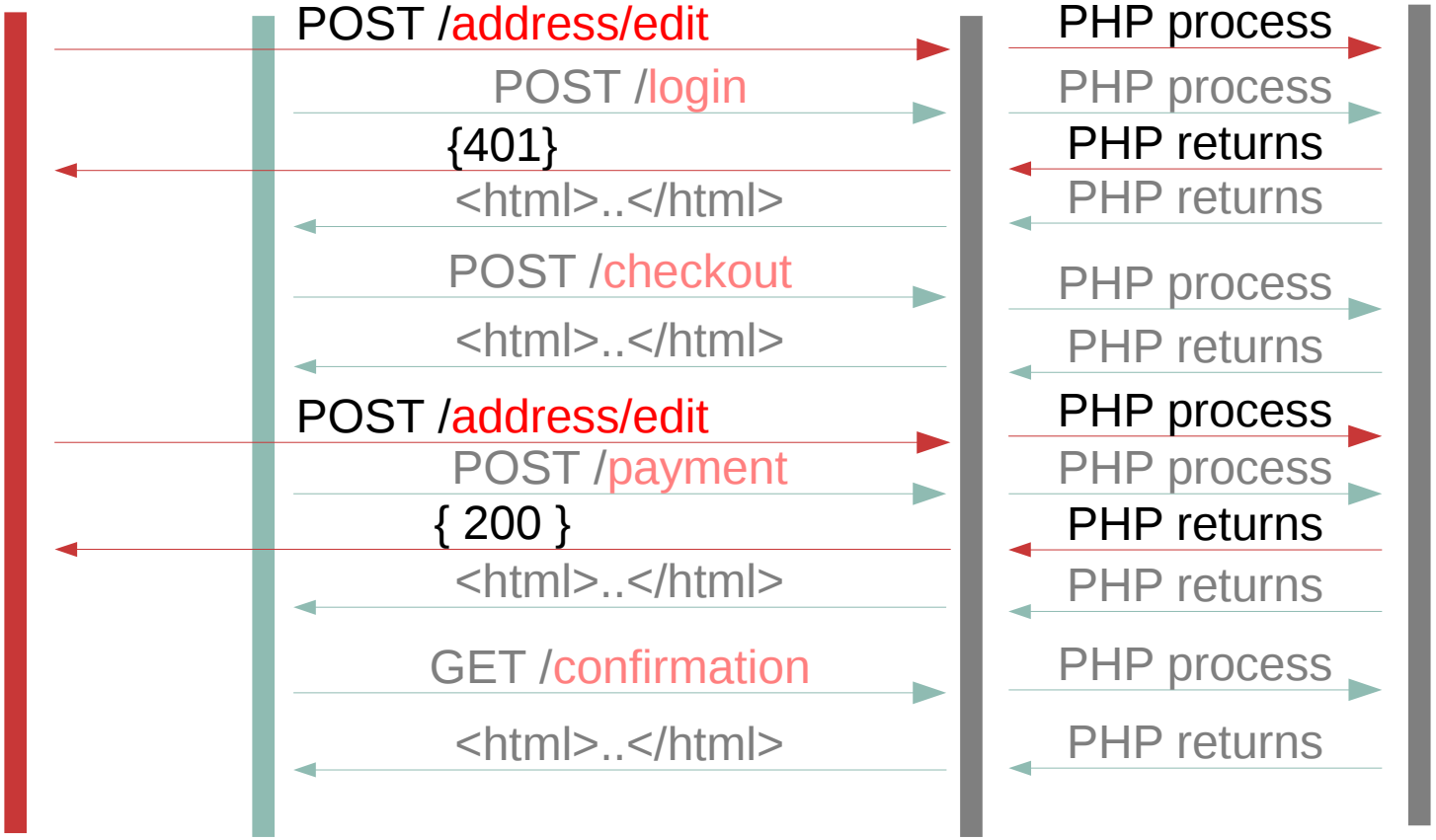
HTTP

php


GET /index.php
PHP process
PHP returns
<html>..</html>
POST /login.php
PHP process
PHP returns
<html>..</html>

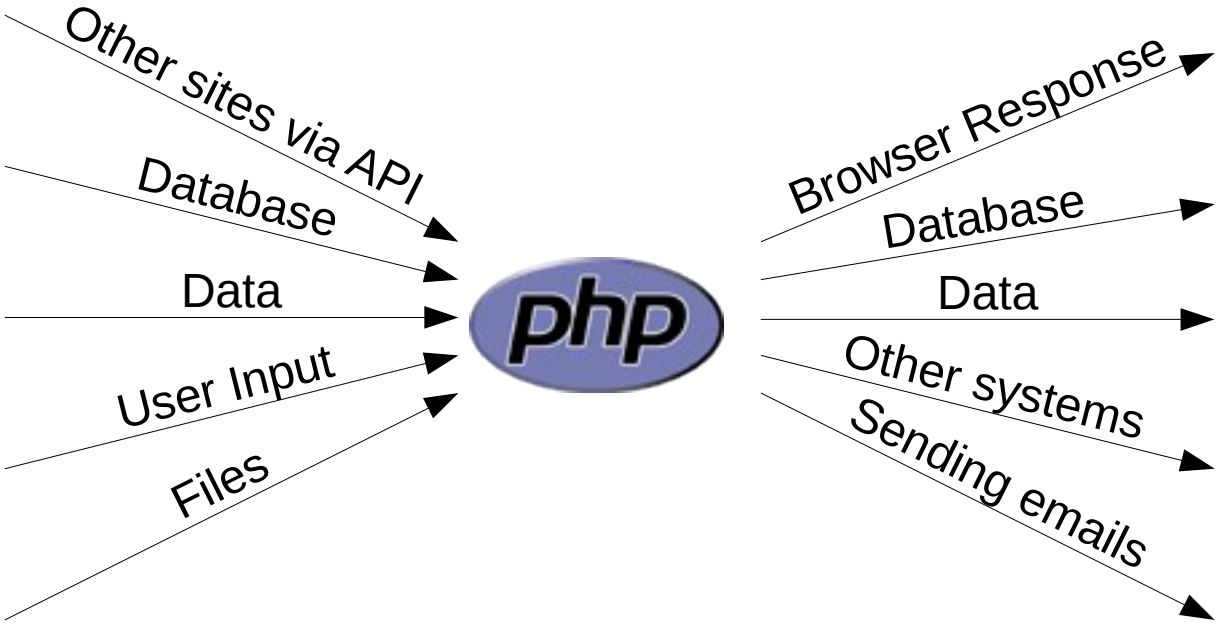
Cross-Site Request Forgeries



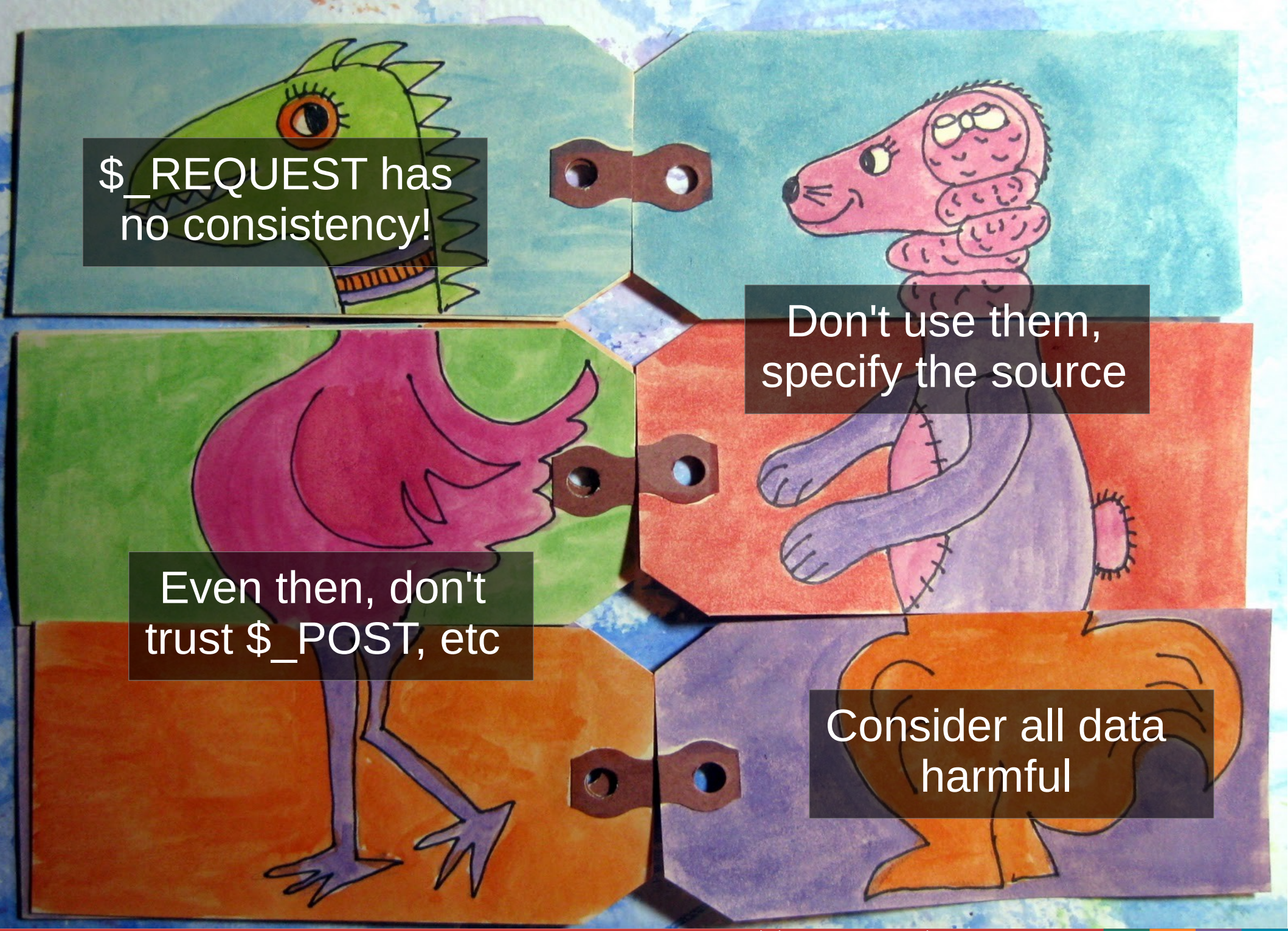


PHP Ain't Clever

(hint, not many programming languages are!)



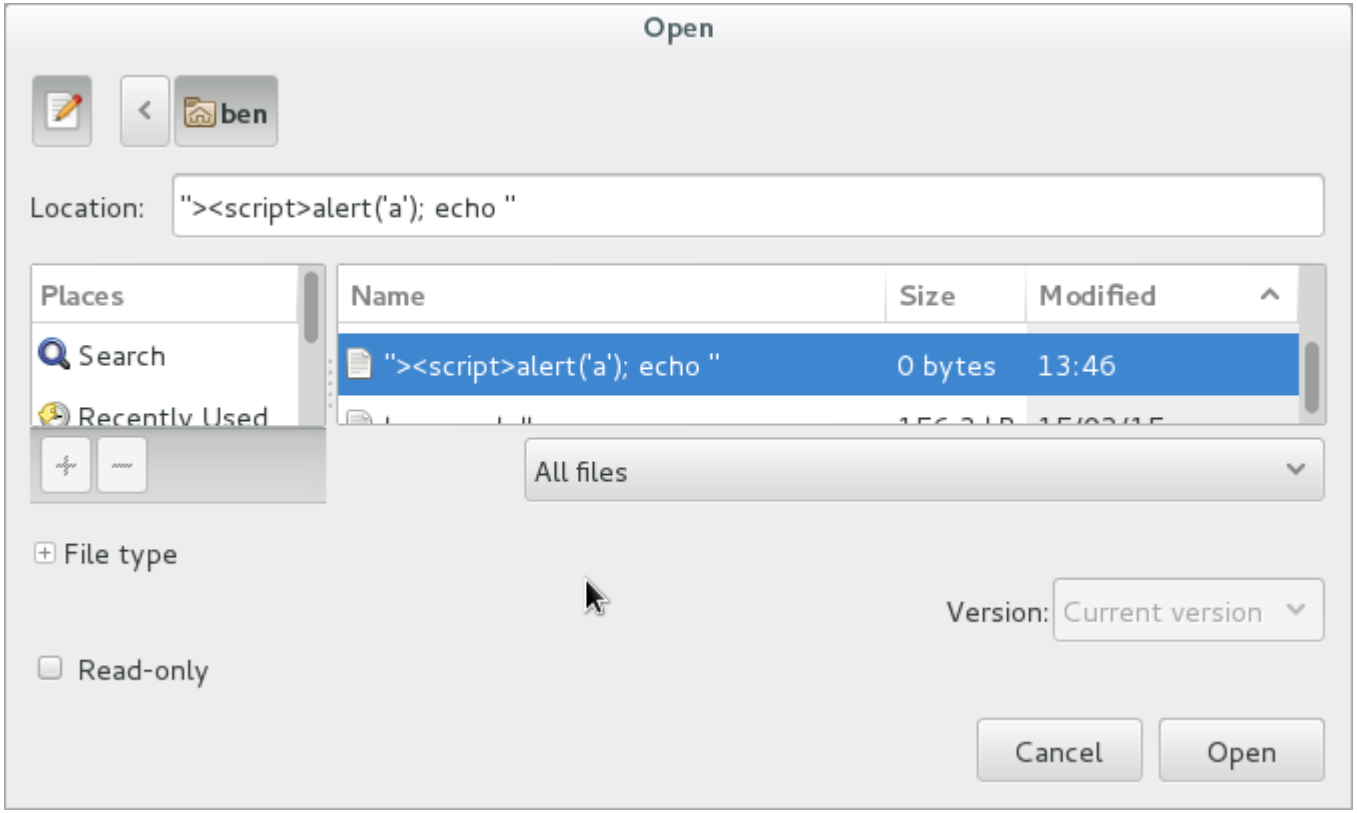
Piecing Data Together


$_REQUEST has no consistency!
Don't use them, specify the source
Even then, don't trust $_POST, etc
Consider all data harmful

Validate All Incoming Data

Look Out for Odd Entry Points

Look Out for Odd Entry Points



Email Addresses?

ben@dechrai.com

ben+yowwest15@dechrai.com

"Ben at YOW! West"@dechrai.com

Bḝn@169.254.141.29

Email Addresses?

```
(?:[a-z0-9!#$%&'*+/=?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*+/=?^_`{|}~-]
+)*
  |  "(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]
      |  \\[\x01-\x09\x0b\x0c\x0e-\x7f])*")
@ (?:(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-
9-]*[a-z0-9])?
  |  \[(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}
       (?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?|[a-z0-9-]*[a-z0-
9]:
          (?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21-\x5a\x53-\x7f]
          |  \\[\x01-\x09\x0b\x0c\x0e-\x7f])+)
     \])
```

Email Addresses?

```
(?:[a-z0-9!#$%&'*+/=?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*+/=?^_`{|}~-]
+)*
  |  "(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]
     |  \\[\x01-\x09\x0b\x0c\x0e-\x7f])*")
@ (?:(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-
9-]*[a-z0-9])?
  |  \[(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}
       (?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?|[a-z0-9-]*[a-z0-
9]:
          (?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21-\x5a\x53-\x7f]
          |  \\[\x01-\x09\x0b\x0c\x0e-\x7f])+)
     \])
```

Some people, when confronted with a problem, think, "I know, I'll use regular expressions." Now they have two problems.

—Jamie Zawinksi

Email Addresses?

Send them an email!

Names?

(see http://is.gd/validating_names)

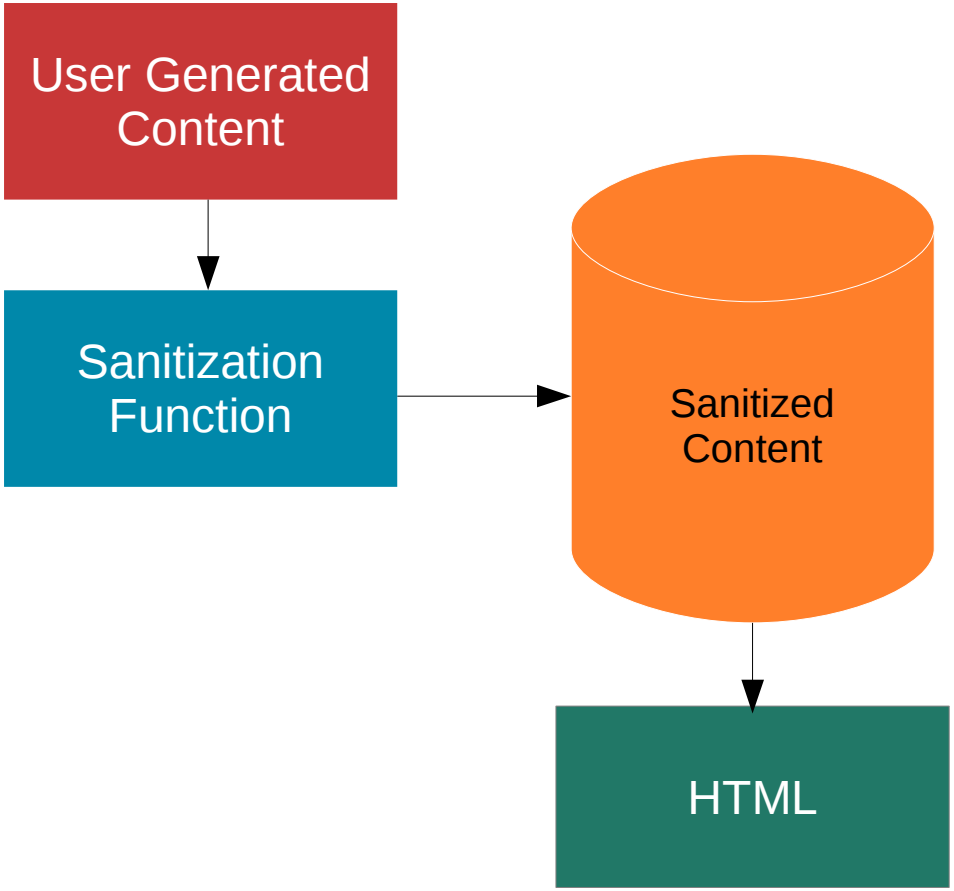
Falsehoods Programmers Believe About Names

1. People have exactly one canonical full name.
2. People have exactly one full name which they go by.
3. People have, at this point in time, exactly one canonical full name.
4. People have, at this point in time, one full name which they go by.
5. People have exactly N names, for any value of N.
6. People's names fit within a certain defined amount of space.
7. People's names do not change.
8. People's names change, but only at a certain enumerated set of events.
9. People's names are written in ASCII.
10. People's names are written in any single character set.
11. People's names are all mapped in Unicode code points.
12. People's names are case sensitive.
13. People's names are case insensitive.
14. People's names sometimes have prefixes or suffixes, but you can safely ignore those.
15. People's names do not contain numbers.

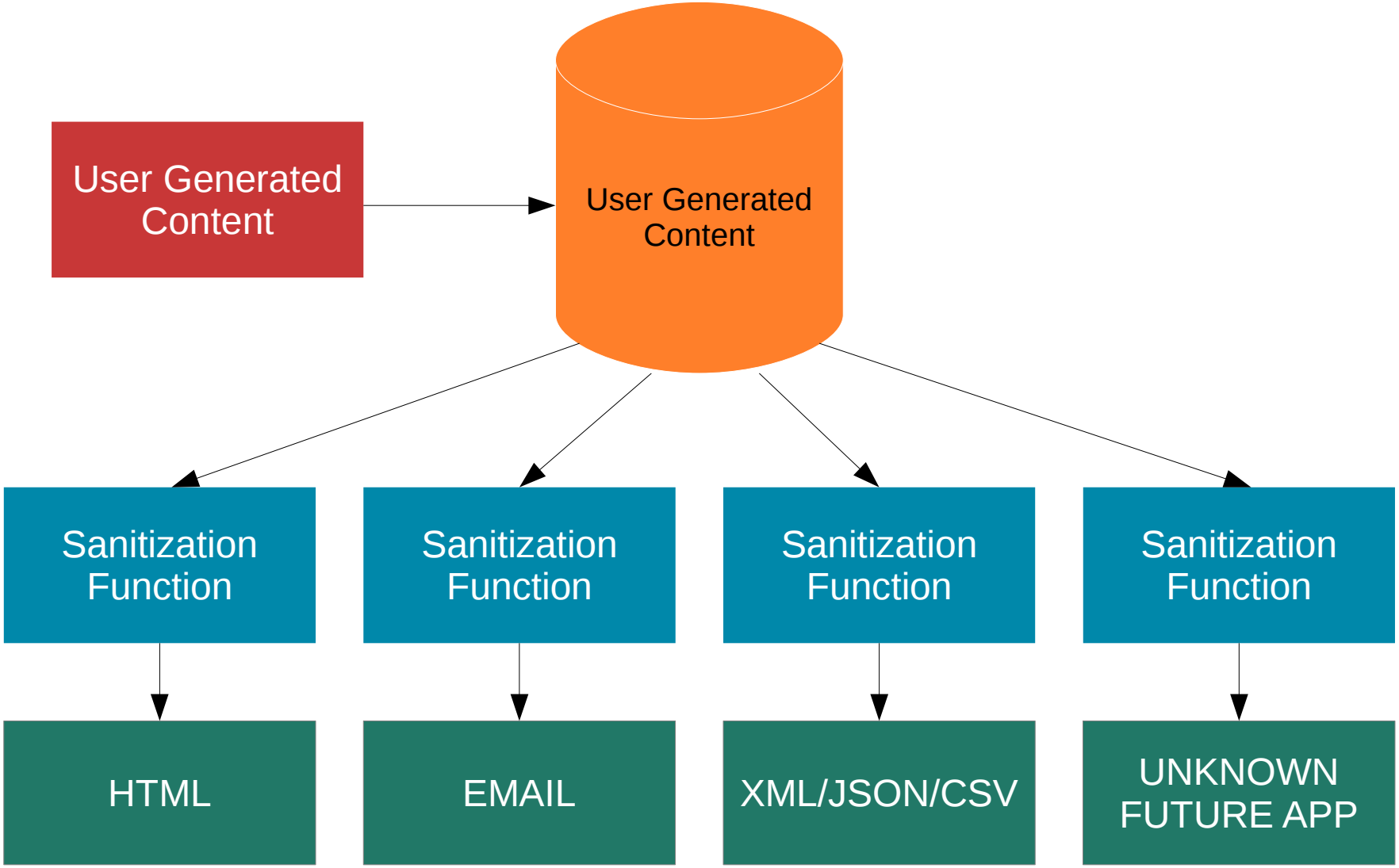
Names?

- Who decides if a name is valid?
  - Josè Smith
  - Laʒamon
  - Þórinn Eikinskjaldi
  - Πηληϊάδεω Ἀχιλῆος
  - Federico del Sagrado Corazón de Jesús García Lorca

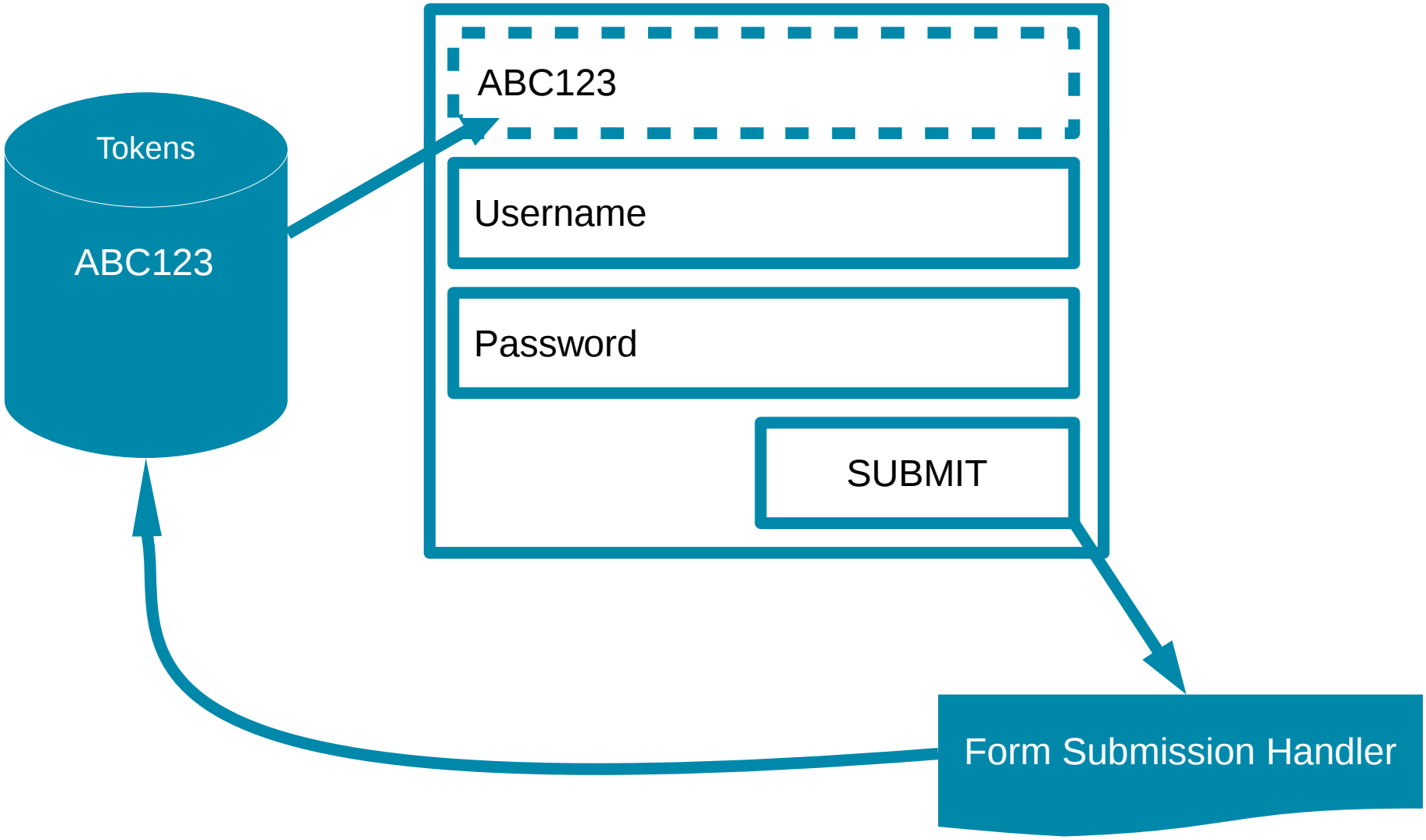
Sanitize Data



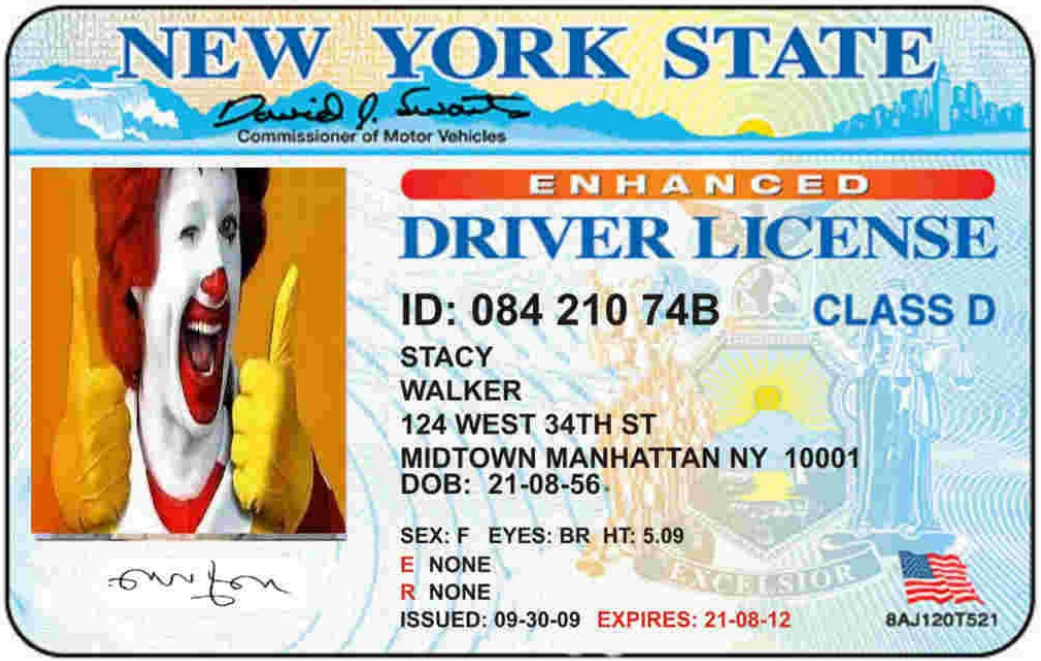
Sanitize on Output



Cross-Site Request Forgeries



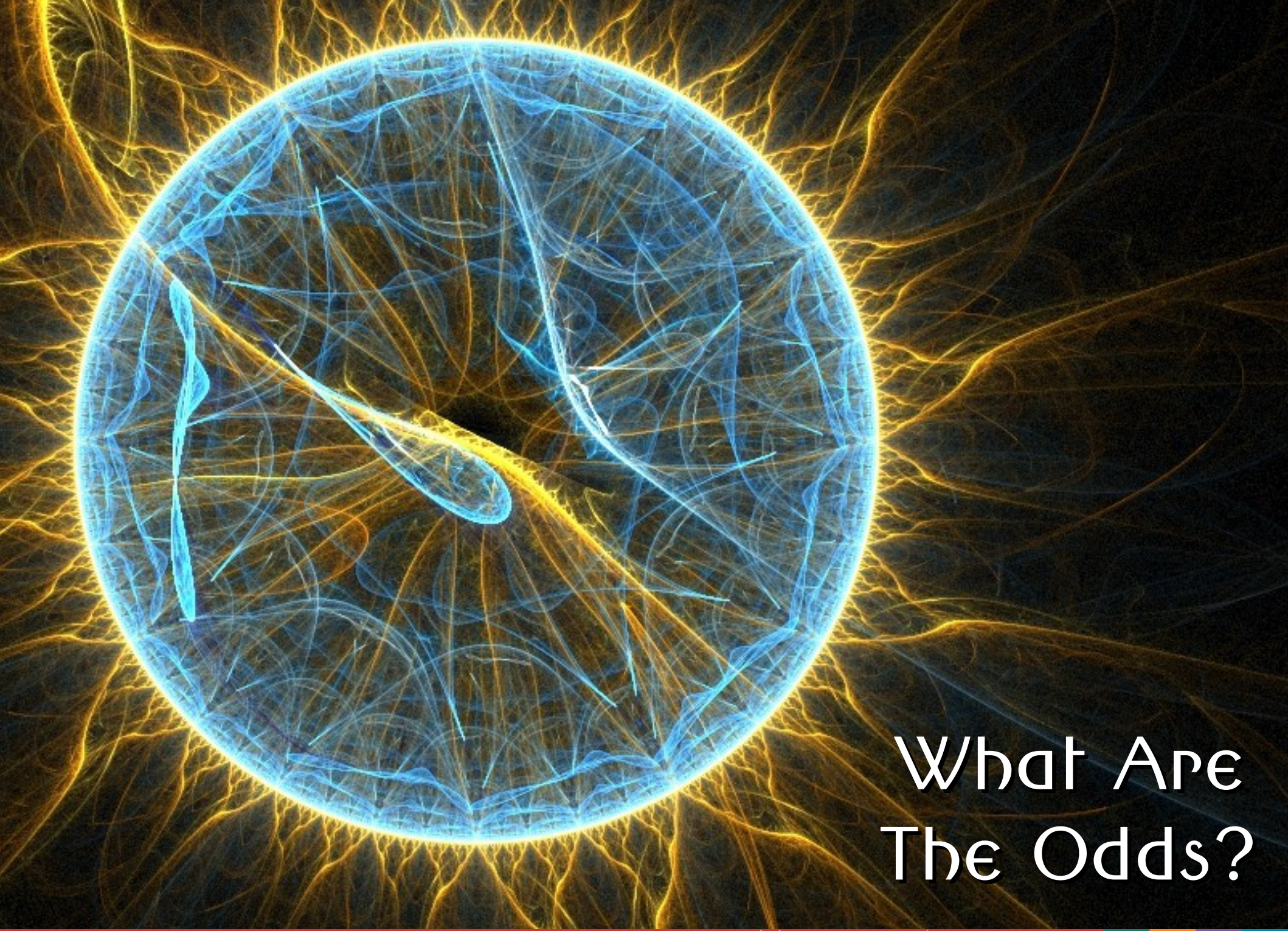
Cross-Site Request Forgeries



Referrers can be easily forged;

don't rely on them

```
curl
    --referer="http://good.com"
    --data "username=admin&password=complex"
    http://good.com/user/login/
```



What Are The Odds?

Making Prototypes Live

"That's great! Let's make it live!"

– Almost every CEO ever

Making Prototypes Live

References

- OWASP Top 10 Cheat Sheet

Credits

- Security Camera image by Henning Mühlinghaus
- Piecing Data by José Manuel Ríos Valiente
- Flip Book Animals by Jamie Fingal
- Customs and Immigation From learnz.org.nz
- Flight Tracks screenshot from flightaware.com
- Conception image by Lynn (Gracie's mom)
- Scooter Security by EpicFail.com

CTO FOR HIRE

Ben Dechrai
@bendechrai

Thank You!

Questions?

https://cto.to/webappsec

CC BY NC SA

#webappsec #yowwest15 https://joind.in/14594

YOW! West, Perth, 2015