Team 4 attack findings

Target Team Details:
URL: https://neu-csye6225-spring2017-team-4.me/

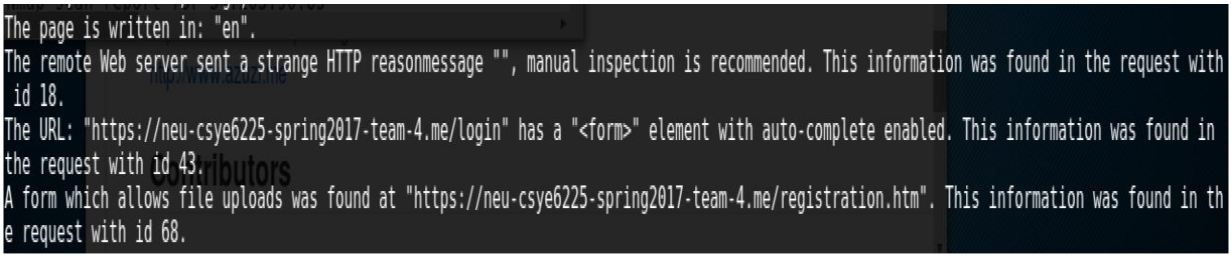
1. The victim’s web application accepts all types of files in their file upload section.

   Uploaded files represent a significant risk to applications. The first step in many attacks is to get some code to the system to be attacked. Then the attack only needs to find a way to get the code executed. Using a file upload helps the attacker accomplish the first step.

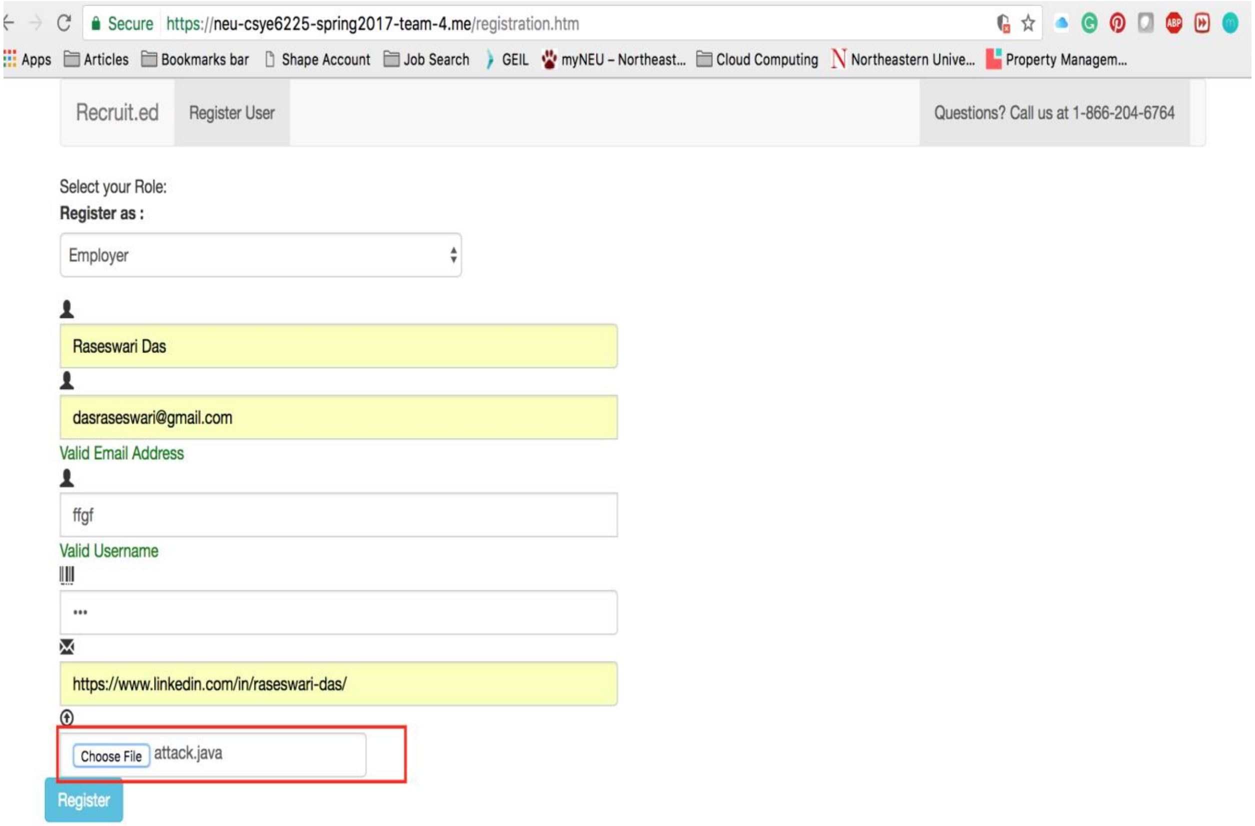
   The consequences of unrestricted file upload can vary, including complete system takeover, an overloaded file system or database, forwarding attacks to back-end systems, client-side attacks, or simple defacement. It depends on what the application does with the uploaded file and especially where it is stored.

   There are really two classes of problems here. The first is with the file metadata, like the path and file name. These are generally provided by the transport, such as HTTP multi-part encoding. This data may trick the application into overwriting a critical file or storing the file in a bad location. They must validate the metadata extremely carefully before using it.

   Tool used: w3af (Kali Linux)



```
The page is written in: "en".
The remote Web server sent a strange HTTP reasonmessage "", manual inspection is recommended. This information was found in the request with
 id 18.
The URL: "https://neu-csye6225-spring2017-team-4.me/login" has a "<form>" element with auto-complete enabled. This information was found in
the request with id 43.
A form which allows file uploads was found at "https://neu-csye6225-spring2017-team-4.me/registration.htm". This information was found in th
e request with id 68.
```



Questions? Call us at 1-866-204-6764

Select your Role:

Register as :

Employer

Raseswari Das

dasraseswari@gmail.com

Valid Email Address

ffgf

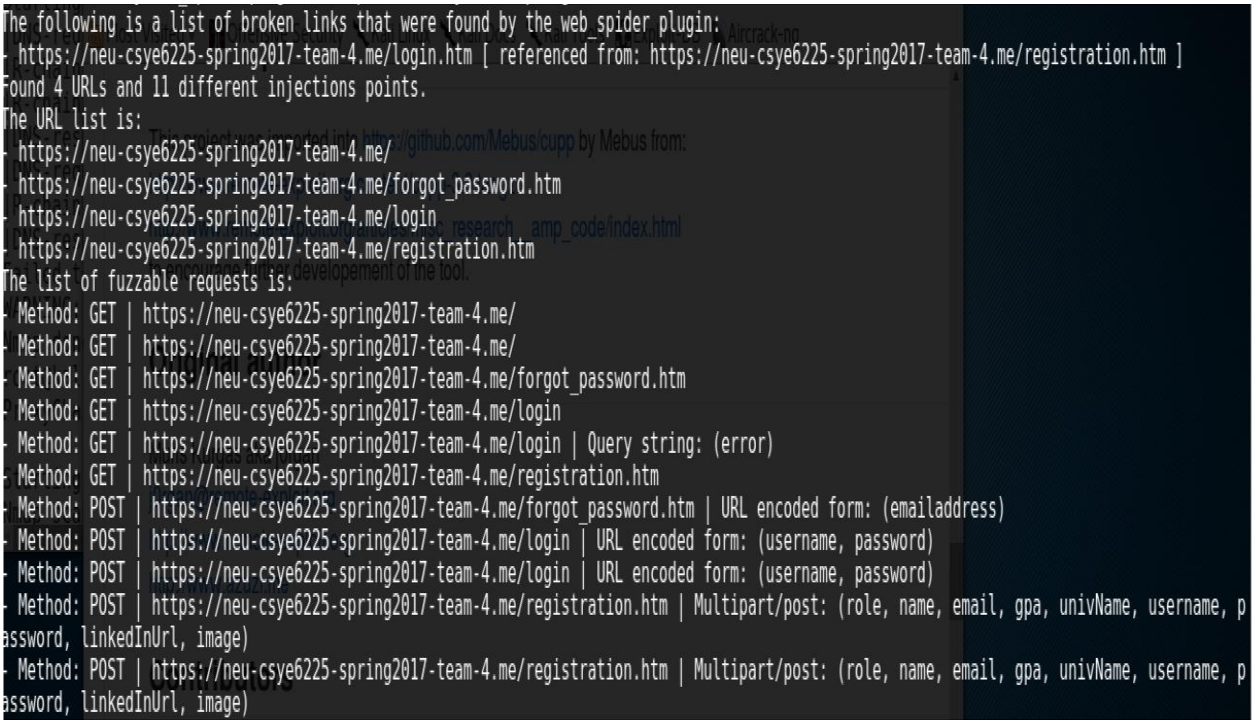
Valid Username

•••

https://www.linkedin.com/in/raseswari-das/

attack.java

2. Using W3af, we found several injection points, where a SQL injection attack was launched using "injection" of a SQL query via the input data from the client to the application. A successful SQL injection exploit can read sensitive data from the database, modify database data (Insert/Update/Delete), execute administration operations on the database (such as shutdown the DBMS), recover the content of a given file present on the DBMS file system and in some cases issue commands to the operating system.

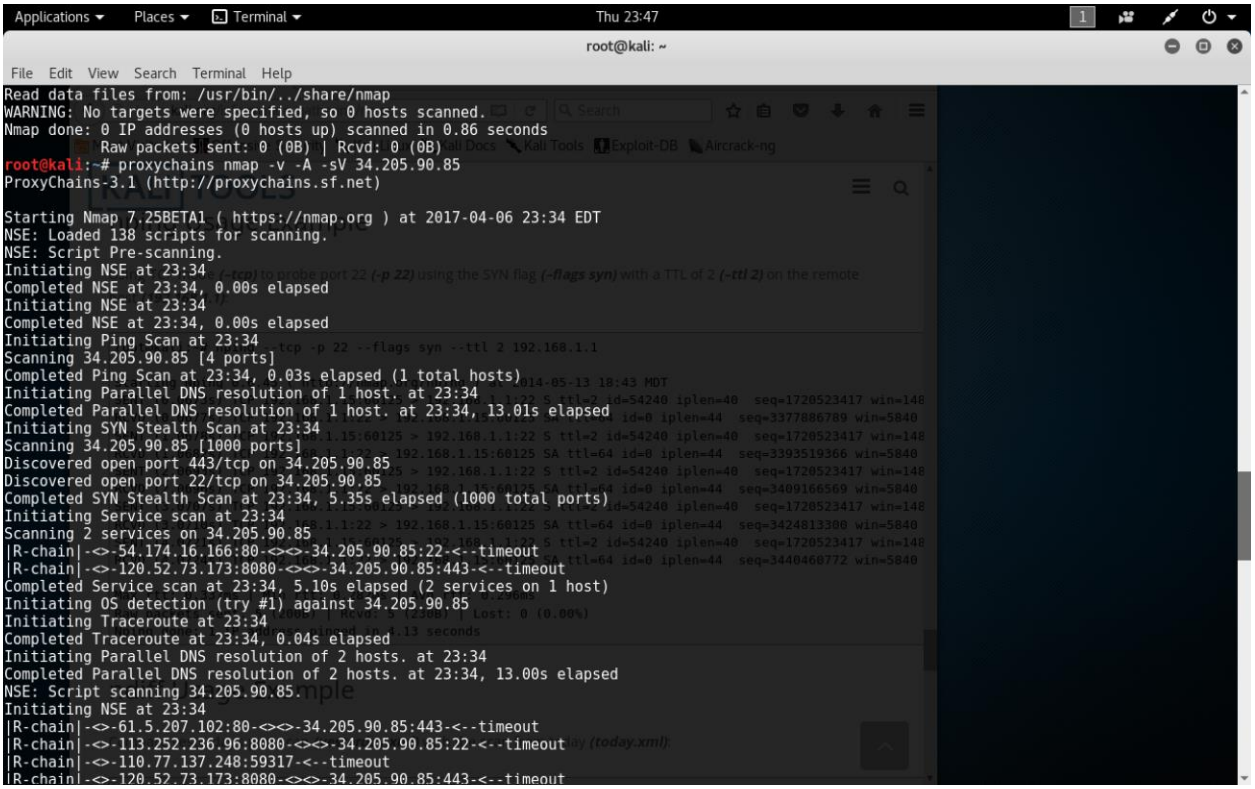


```
The following is a list of broken links that were found by the web spider plugin:
- https://neu-csye6225-spring2017-team-4.me/login.htm [ referenced from: https://neu-csye6225-spring2017-team-4.me/registration.htm ]
Found 4 URLs and 11 different injections points.
The URL list is:
- https://neu-csye6225-spring2017-team-4.me/
- https://neu-csye6225-spring2017-team-4.me/forgot_password.htm
- https://neu-csye6225-spring2017-team-4.me/login
- https://neu-csye6225-spring2017-team-4.me/registration.htm
The list of fuzzable requests is:
- Method: GET | https://neu-csye6225-spring2017-team-4.me/
- Method: GET | https://neu-csye6225-spring2017-team-4.me/
- Method: GET | https://neu-csye6225-spring2017-team-4.me/forgot_password.htm
- Method: GET | https://neu-csye6225-spring2017-team-4.me/login
- Method: GET | https://neu-csye6225-spring2017-team-4.me/login | Query string: (error)
- Method: GET | https://neu-csye6225-spring2017-team-4.me/registration.htm
- Method: POST | https://neu-csye6225-spring2017-team-4.me/forgot_password.htm | URL encoded form: (emailaddress)
- Method: POST | https://neu-csye6225-spring2017-team-4.me/login | URL encoded form: (username, password)
- Method: POST | https://neu-csye6225-spring2017-team-4.me/login | URL encoded form: (username, password)
- Method: POST | https://neu-csye6225-spring2017-team-4.me/registration.htm | Multipart/post: (role, name, email, gpa, univName, username, p
assword, linkedInUrl, image)
- Method: POST | https://neu-csye6225-spring2017-team-4.me/registration.htm | Multipart/post: (role, name, email, gpa, univName, username, p
assword, linkedInUrl, image)
```

3. Discovered open ports. An open port is an attack surface. The daemon that is listing on a port, could be vulnerable to a buffer overflow, or another remotely exploitable vulnerability.

   This report can assist analysts in identifying SSH server versions within the organization. Once analysts identify the different versions of SSH installed on hosts, they can update vulnerable SSH servers and clients.

   An important principle in security is reducing your attack surface, and ensure that servers have the minimum number of exposed services.



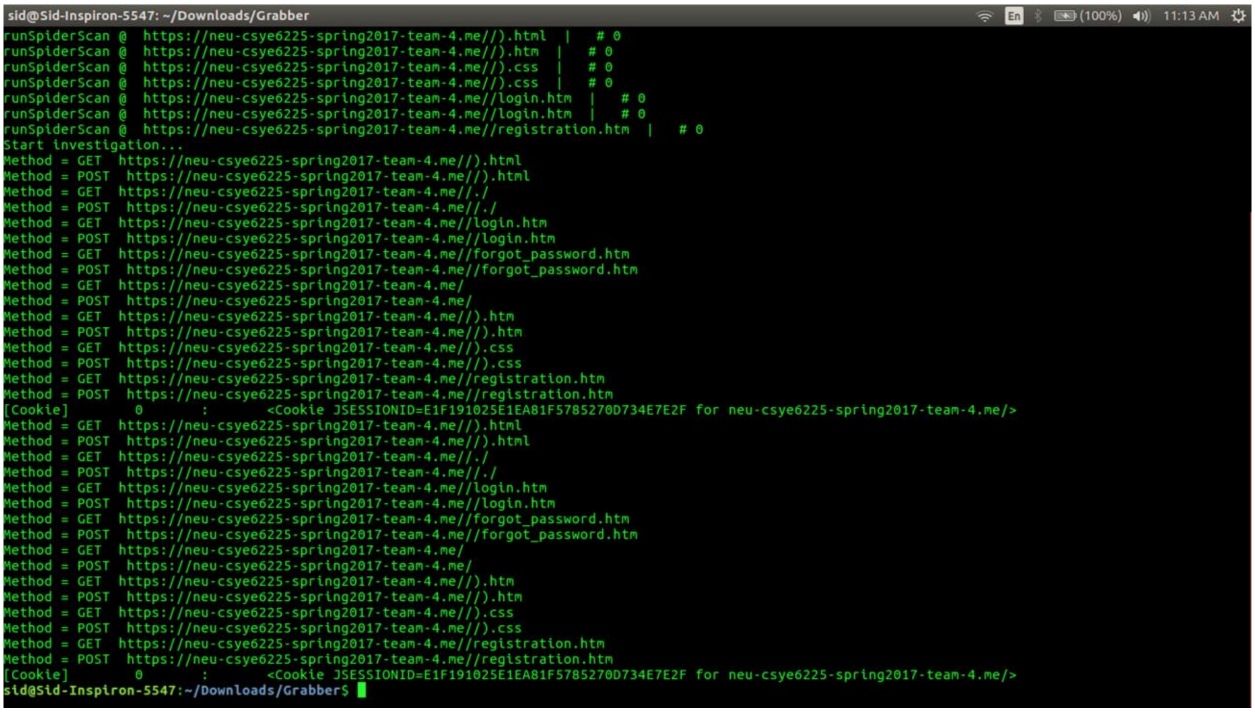
```
Read data files from: /usr/bin/../share/nmap
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.86 seconds
           Raw packets sent: 0 (0B) | Rcvd: 0 (0B)
root@kali:~# proxychains nmap -v -A -sV 34.205.90.85
ProxyChains-3.1 (http://proxychains.sf.net)

Starting Nmap 7.25BETA1 ( https://nmap.org ) at 2017-04-06 23:34 EDT
NSE: Loaded 138 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 23:34
Completed NSE at 23:34, 0.00s elapsed
Initiating NSE at 23:34
Completed NSE at 23:34, 0.00s elapsed
Initiating Ping Scan at 23:34
Scanning 34.205.90.85 [4 ports]
Completed Ping Scan at 23:34, 0.03s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 23:34
Completed Parallel DNS resolution of 1 host. at 23:34, 13.01s elapsed
Initiating SYN Stealth Scan at 23:34
Scanning 34.205.90.85 [1000 ports]
Discovered open port 443/tcp on 34.205.90.85
Discovered open port 22/tcp on 34.205.90.85
Completed SYN Stealth Scan at 23:34, 5.35s elapsed (1000 total ports)
Initiating Service scan at 23:34
Scanning 2 services on 34.205.90.85
|R-chain|-<>-54.174.16.166:80-<><>-34.205.90.85:22-<--timeout
|R-chain|-<>-120.52.73.173:8080-<><>-34.205.90.85:443-<--timeout
Completed Service scan at 23:34, 5.10s elapsed (2 services on 1 host)
Initiating OS detection (try #1) against 34.205.90.85
Initiating Traceroute at 23:34
Completed Traceroute at 23:34, 0.04s elapsed
Initiating Parallel DNS resolution of 2 hosts. at 23:34
Completed Parallel DNS resolution of 2 hosts. at 23:34, 13.00s elapsed
NSE: Script scanning 34.205.90.85.
Initiating NSE at 23:34
|R-chain|-<>-61.5.207.102:80-<><>-34.205.90.85:443-<--timeout
|R-chain|-<>-113.252.236.96:8080-<><>-34.205.90.85:22-<--timeout
|R-chain|-<>-110.77.137.248:59317-<--timeout
|R-chain|-<>-120.52.73.173:8080-<><>-34.205.90.85:443-<--timeout
```

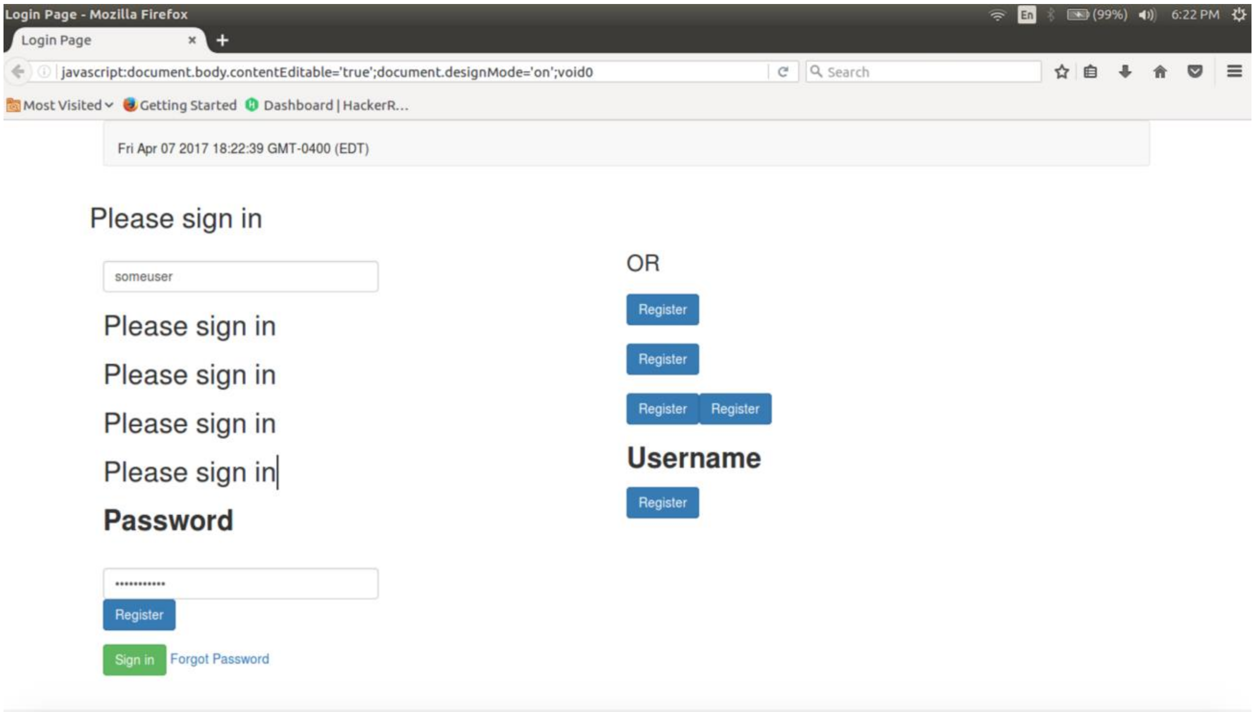
4. Victim's team is not checking the JS for XSS attacks. So a javascript code made the website editable on the client machine. This is one of the ways scammers create fake screenshots, fake Adsense & affiliate earnings, capture user data, and even fake Paypal transactions. All you need to do is visit the site you want to edit, paste the code below into your web browser address bar and hit enter.
   "javascript:document.body.contentEditable='true'; document.designMode='on'; void 0"

   This javascript should be disabled for such cases.

   Tool used: Grabber (Kali Linux)



```
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//).html  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//).htm  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//).css  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//).css  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//login.htm  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//login.htm  |   # 0
runSpiderScan @  https://neu-csye6225-spring2017-team-4.me//registration.htm  |   # 0
Start investigation...
Method = GET  https://neu-csye6225-spring2017-team-4.me//).html
Method = POST  https://neu-csye6225-spring2017-team-4.me//).html
Method = GET  https://neu-csye6225-spring2017-team-4.me//./
Method = POST  https://neu-csye6225-spring2017-team-4.me//./
Method = GET  https://neu-csye6225-spring2017-team-4.me//login.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//login.htm
Method = GET  https://neu-csye6225-spring2017-team-4.me//forgot_password.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//forgot_password.htm
Method = GET  https://neu-csye6225-spring2017-team-4.me/
Method = POST  https://neu-csye6225-spring2017-team-4.me/
Method = GET  https://neu-csye6225-spring2017-team-4.me//).htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//).htm
Method = GET  https://neu-csye6225-spring2017-team-4.me//).css
Method = POST  https://neu-csye6225-spring2017-team-4.me//).css
Method = GET  https://neu-csye6225-spring2017-team-4.me//registration.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//registration.htm
[Cookie]        0       :       <Cookie JSESSIONID=E1F191025E1EA81F5785270D734E7E2F for neu-csye6225-spring2017-team-4.me/>
Method = GET  https://neu-csye6225-spring2017-team-4.me//).html
Method = POST  https://neu-csye6225-spring2017-team-4.me//).html
Method = GET  https://neu-csye6225-spring2017-team-4.me//./
Method = POST  https://neu-csye6225-spring2017-team-4.me//./
Method = GET  https://neu-csye6225-spring2017-team-4.me//login.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//login.htm
Method = GET  https://neu-csye6225-spring2017-team-4.me//forgot_password.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//forgot_password.htm
Method = GET  https://neu-csye6225-spring2017-team-4.me/
Method = POST  https://neu-csye6225-spring2017-team-4.me/
Method = GET  https://neu-csye6225-spring2017-team-4.me//).htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//).htm
Method = GET  https://neu-csye6225-spring2017-team-4.me//).css
Method = POST  https://neu-csye6225-spring2017-team-4.me//).css
Method = GET  https://neu-csye6225-spring2017-team-4.me//registration.htm
Method = POST  https://neu-csye6225-spring2017-team-4.me//registration.htm
[Cookie]        0       :       <Cookie JSESSIONID=E1F191025E1EA81F5785270D734E7E2F for neu-csye6225-spring2017-team-4.me/>
sid@Sid-Inspiron-5547:~/Downloads/Grabber$
```



```
javascript:document.body.contentEditable='true';document.designMode='on';void0
```

Fri Apr 07 2017 18:22:39 GMT-0400 (EDT)