Cybersecurity

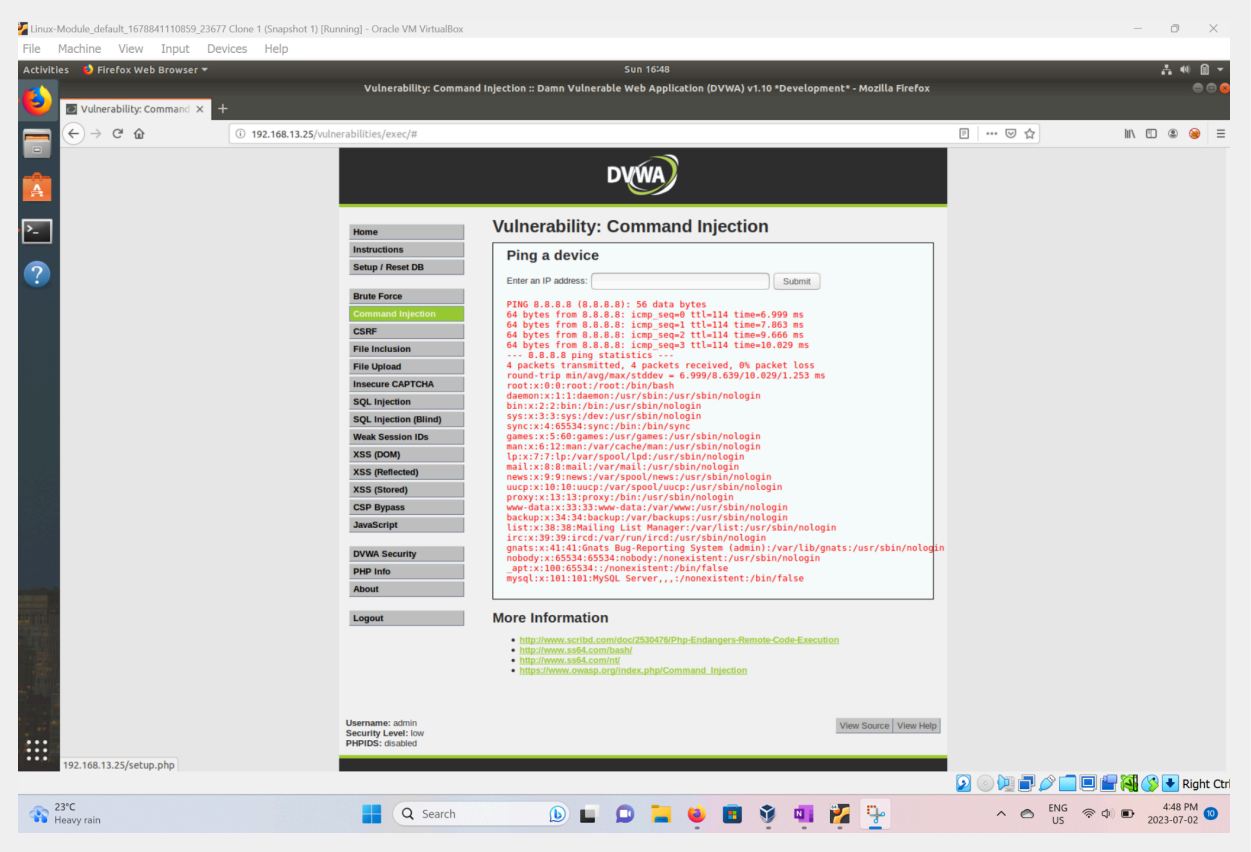
Module 15 Challenge Submission File

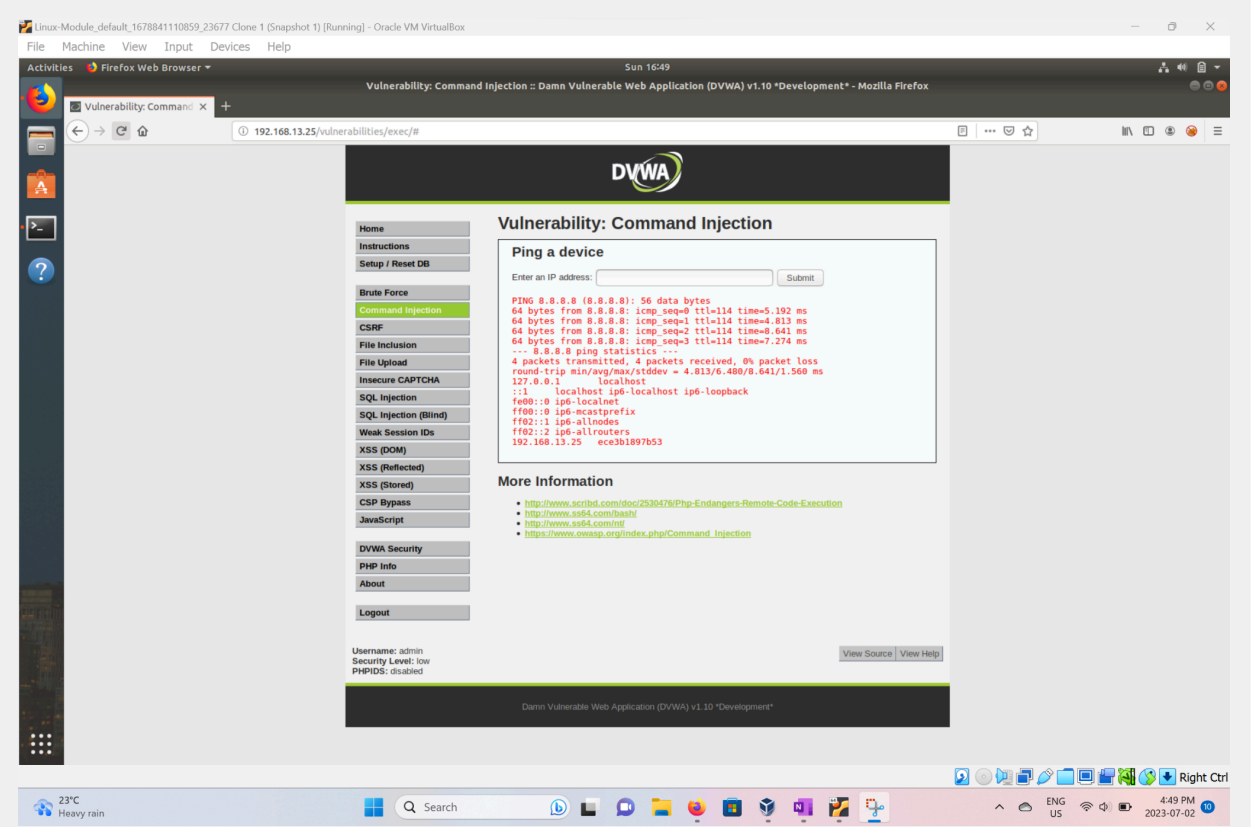
Testing Web Applications for Vulnerabilities

Make a copy of this document to work in, and then respond to each question below the prompt. Save and submit this completed file as your Challenge deliverable.

Web Application 1: *Your Wish is My Command Injection*

Provide a screenshot confirming that you successfully completed this exploit:



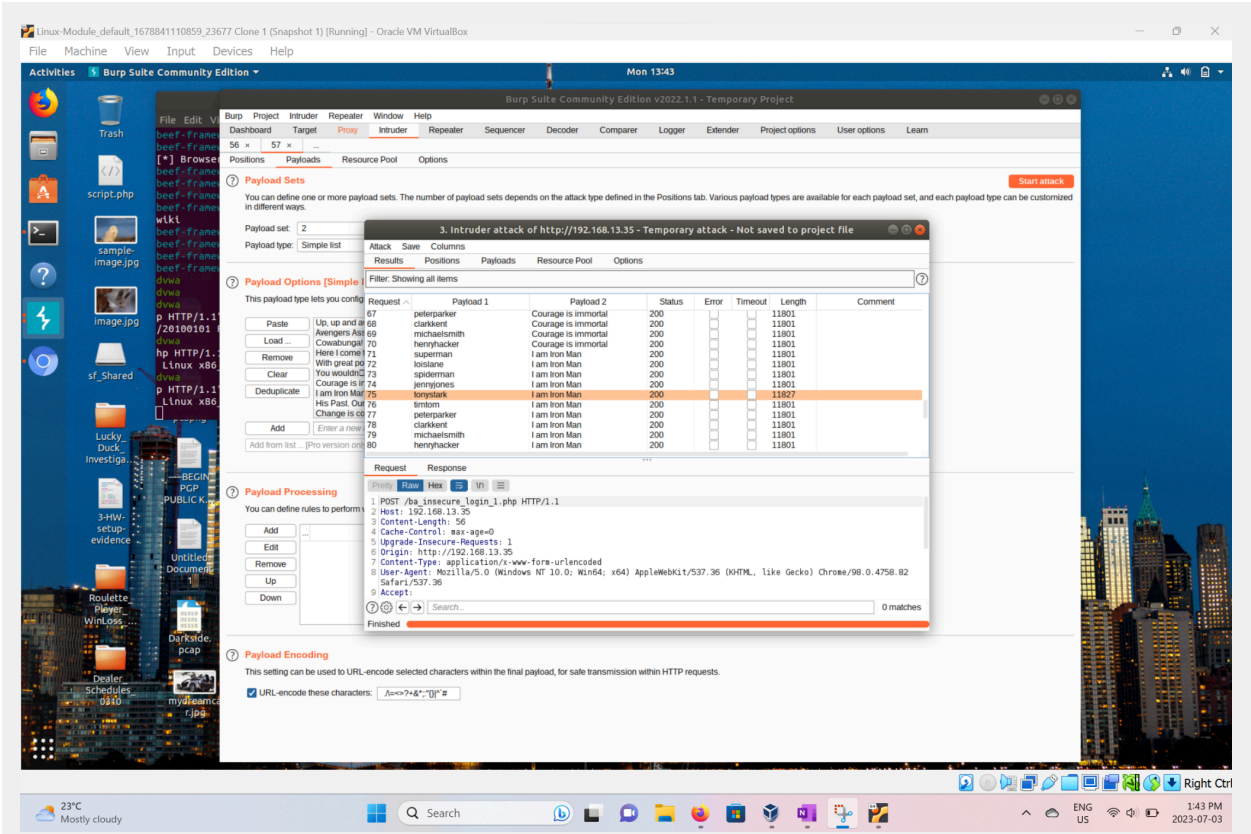


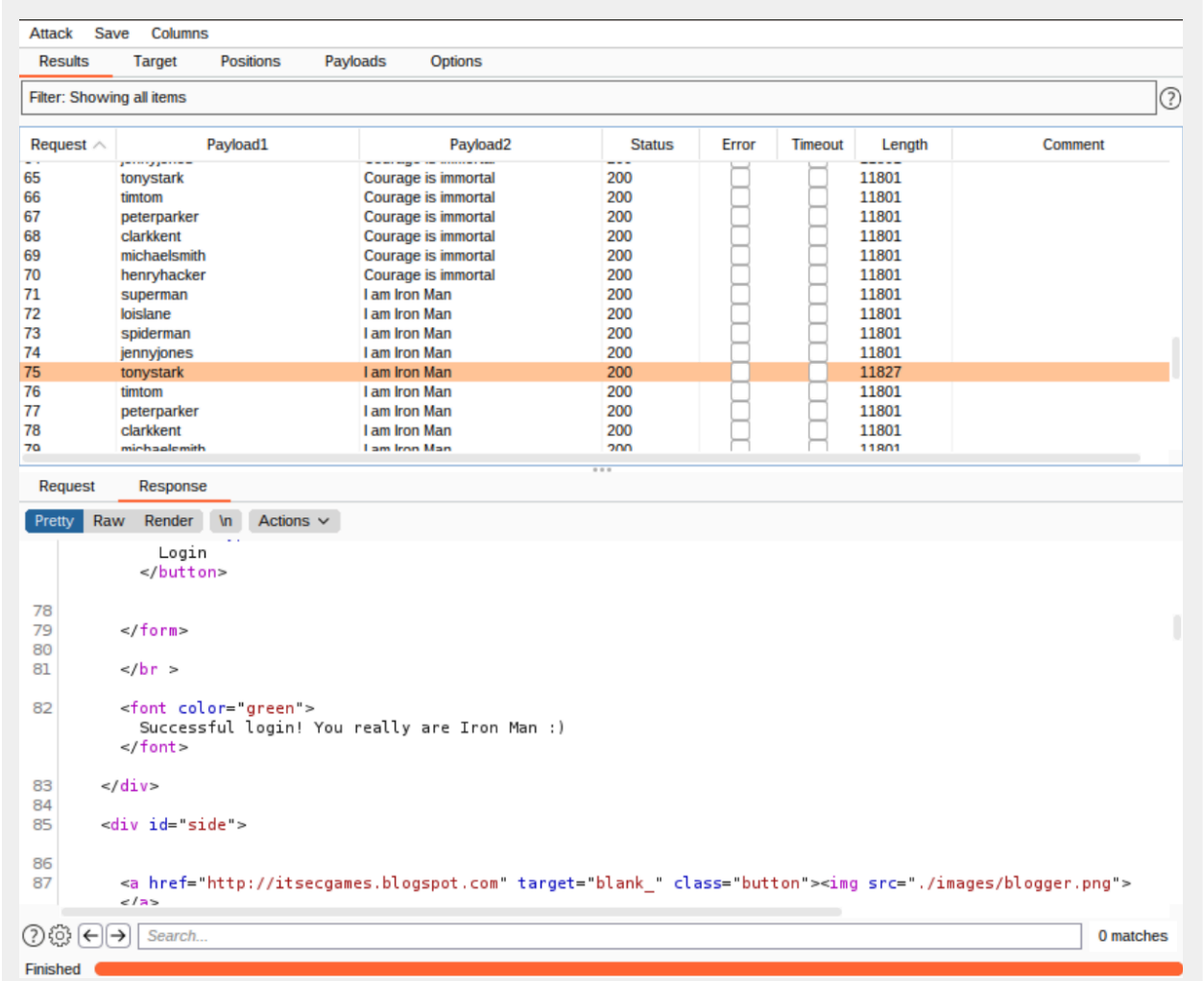
Write two or three sentences outlining mitigation strategies for this vulnerability:

```
A few ways to make sure that this does not happen is to separate
confidential files from the web server and accessible directories, you could
also set permissions to restrict the web server account to just a few select
individuals, and lastly you can add server-side validation so that people
cannot access unintended files.
```

Web Application 2: *A Brute Force to Be Reckoned With*

Provide a screenshot confirming that you successfully completed this exploit:


Linux-Module_default_1678841110859_23677 Clone 1 (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Burp Suite Community Edition
Mon 13:43
Burp Suite Community Edition v2022.1.1 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn
Positions Payloads Resource Pool Options
Payload Sets
Start attack
You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
Payload set:
Payload type: Simple list
Payload Options [Simple
This payload type lets you config
Paste
Load ...
Remove
Clear
Deduplicate
Add
Payload Processing
You can define rules to perform
Add
Edit
Remove
Up
Down
Payload Encoding
This setting can be used to URL-encode selected characters within the final payload, for safe transmission within HTTP requests.
URL-encode these characters:
3. Intruder attack of http://192.168.13.35 - Temporary attack - Not saved to project file
Attack Save Columns
Results Positions Payloads Resource Pool Options
Filter: Showing all items
Request Payload 1 Payload 2 Status Error Timeout Length Comment
67 peterparker Courage is immortal 200 11801
68 clarkkent Courage is immortal 200 11801
69 michaelsmith Courage is immortal 200 11801
70 hennyhacker Courage is immortal 200 11801
71 superman I am Iron Man 200 11801
72 loislane I am Iron Man 200 11801
73 spiderman I am Iron Man 200 11801
74 jennyjones I am Iron Man 200 11801
75 tonystark I am Iron Man 200 11827
76 timtom I am Iron Man 200 11801
77 peterparker I am Iron Man 200 11801
78 clarkkent I am Iron Man 200 11801
79 michaelsmith I am Iron Man 200 11801
80 hennyhacker I am Iron Man 200 11801
Request Response
Pretty Raw Hex \n
1 POST /ba_insecure_login_1.php HTTP/1.1
2 Host: 192.168.13.35
3 Content-Length: 56
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.13.35
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/98.0.4758.82 Safari/537.36
9 Accept:
Search...
0 matches
Finished
Trash
script.php
sample-image.jpg
image.jpg
sf_Shared
Lucky_Duck_Investiga...
3-HW-setup-evidence
Roulette_Player_WinLoss...
Dealer_Schedules_0310
Darkside.pcap
Right Ctrl
23°C
Mostly cloudy
Search
ENG
US
1:43 PM
2023-07-03



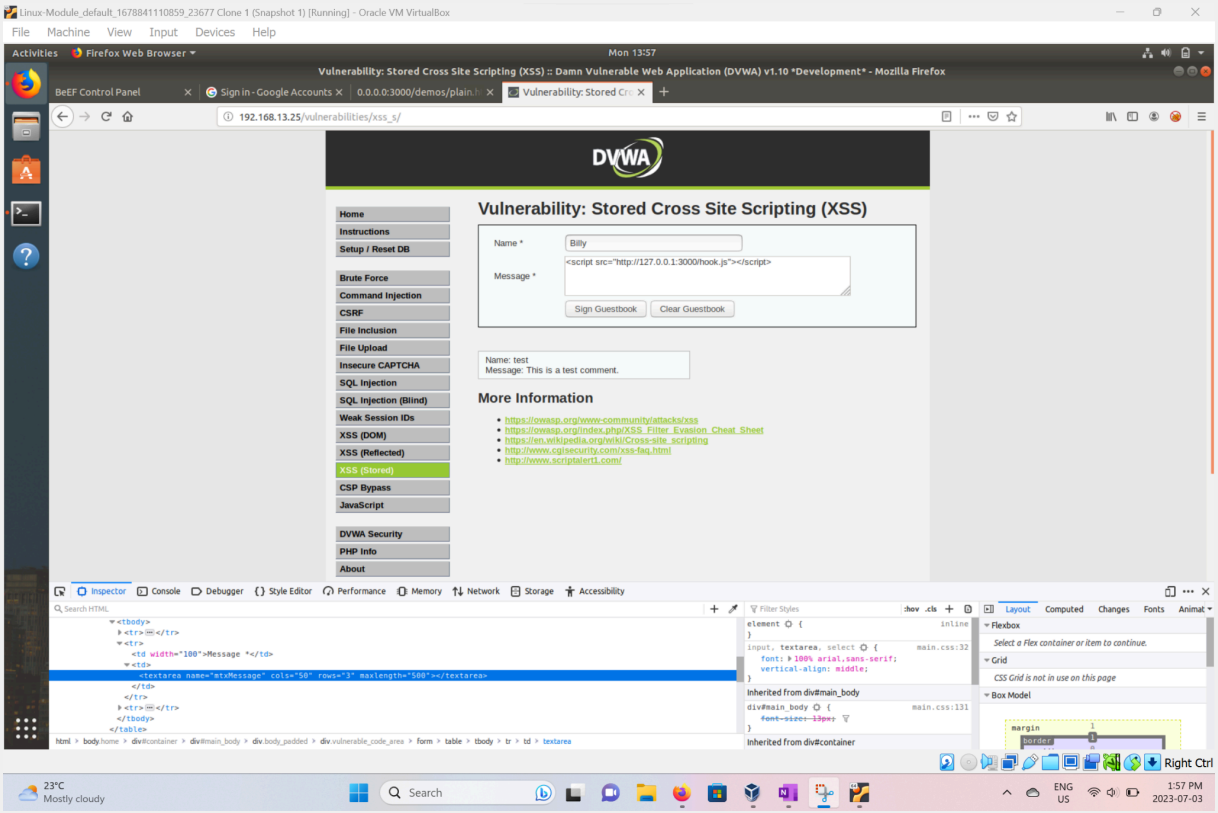
The Iron man line is the on that is different which means it could be the correct combination. After logging it a message displays stating you are iron man.

Write two or three sentences outlining mitigation strategies for this vulnerability:

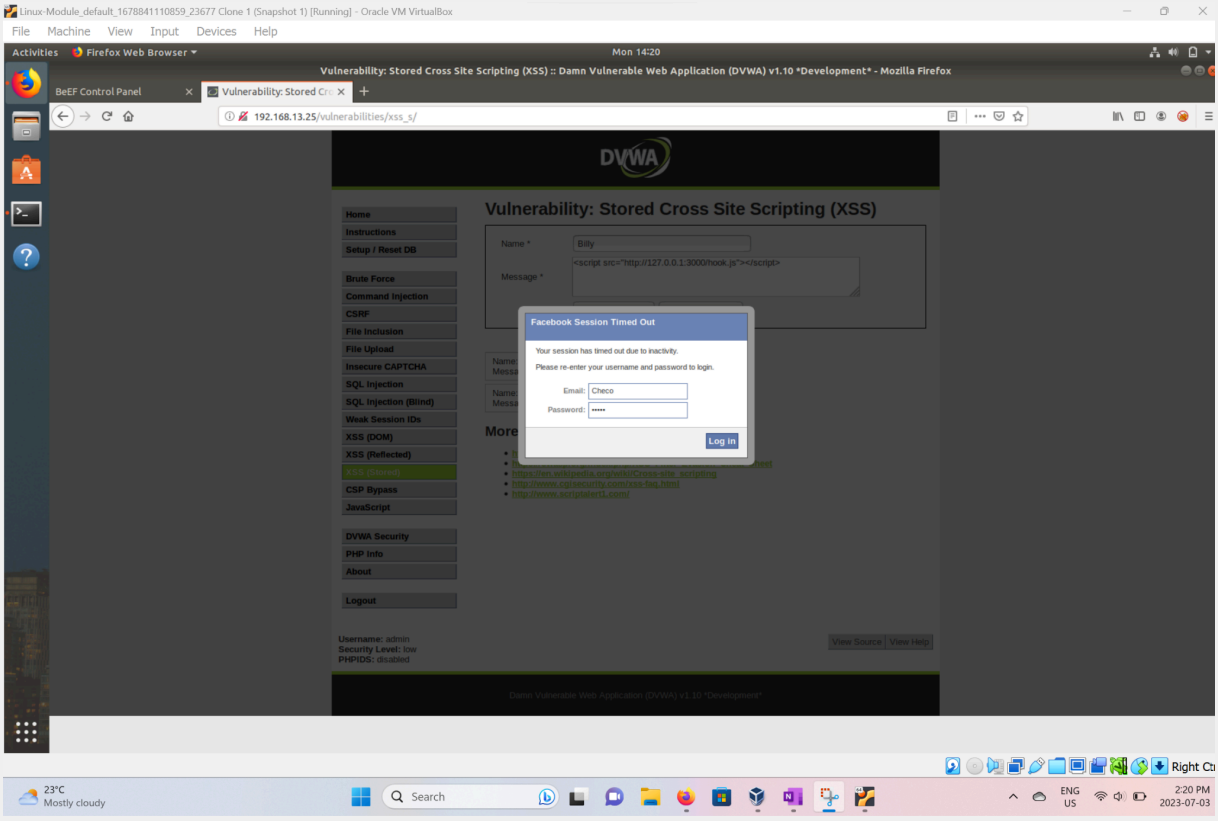
Mitigation Strategies could include requiring 2-factor authentication, a lockout period after a set number of password attempts and the use of complex password and usernames.

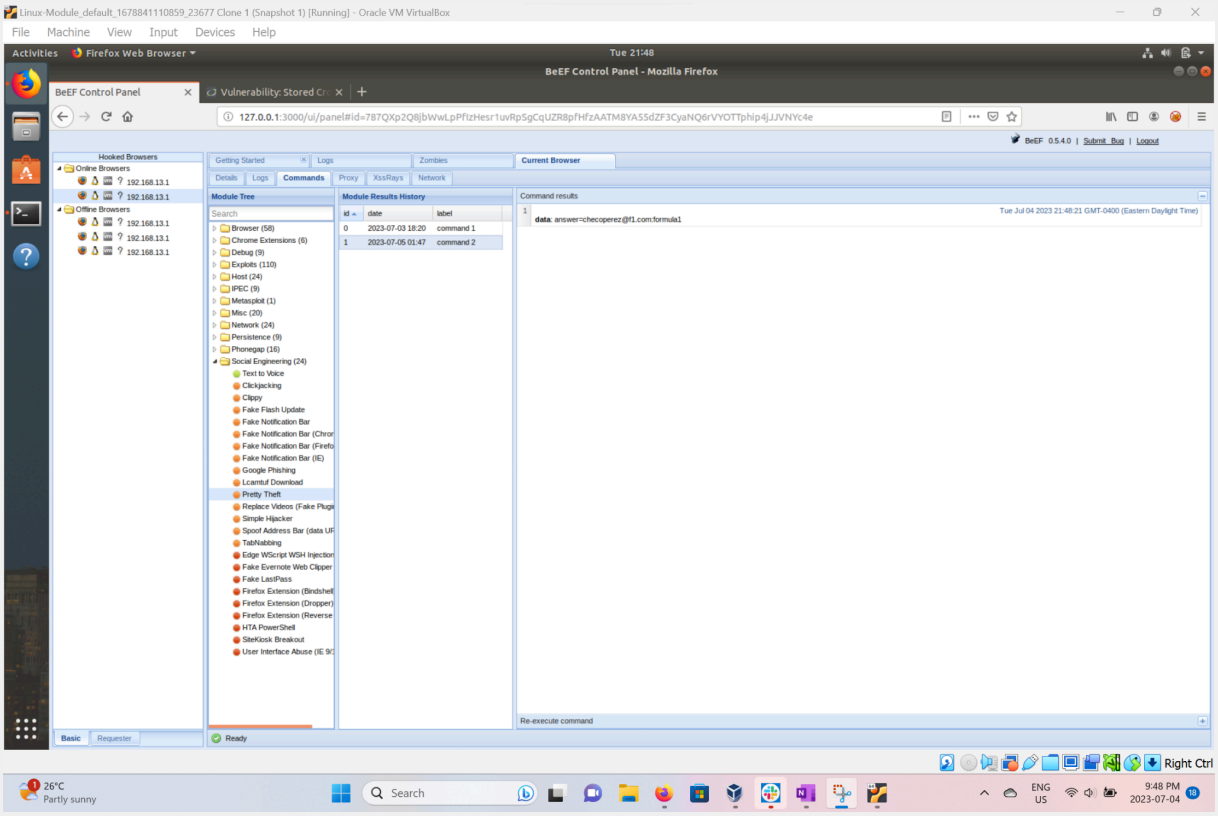
Web Application 3: *Where's the BeEF?*

Provide a screenshot confirming that you successfully completed this exploit:

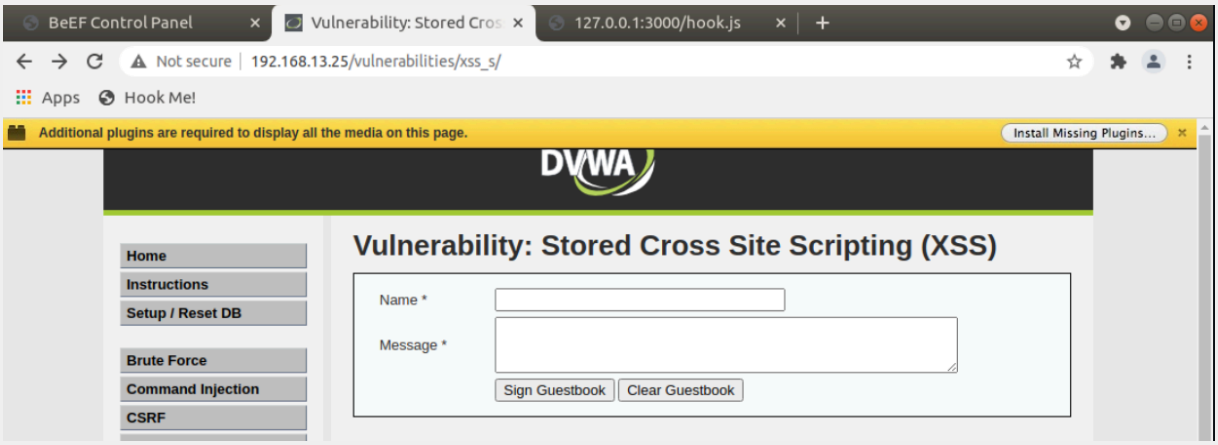


Had to increase the length to fit my script. I adjusted this by inspecting the element and increasing the length to 500.

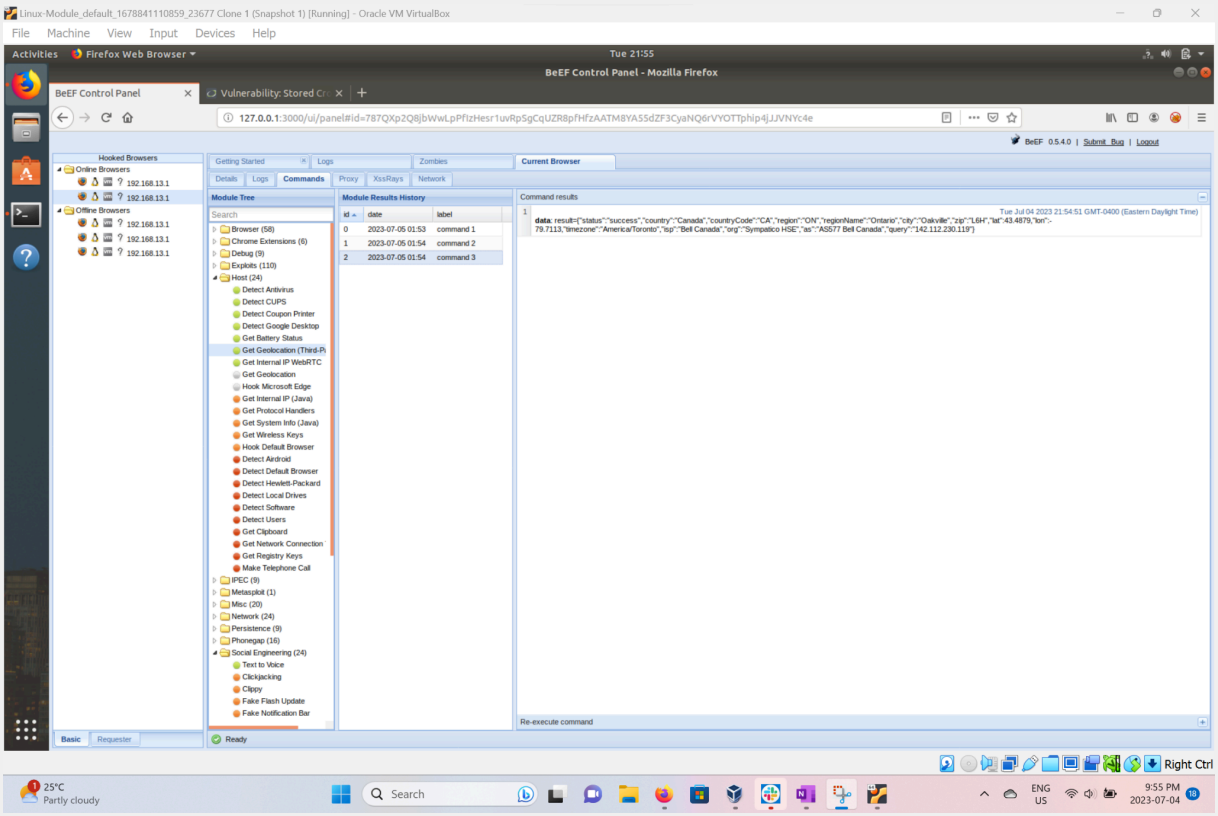




You can see the username and password that were used in the pop-up.



Here the plugin is able to be seen.



Here is the geolocation it was able to see from the option selected.

Write two or three sentences outlining mitigation strategies for this vulnerability:

Input validation is a good way to mitigate the risk of this as it helps prevent any cross site scripting. You can also use browser add-ons to block exploits.