

Hack the Clock:

Automating CVE Exploit Searches

Save time, money, ***and not get bored.***

JORDAN BONAGURA

DEFCON

SecureIdeas

JORDAN BONAGURA

whoami

Over 20y of industry experience

Senior Security Consultant at Secure Ideas

Penetration Tester ←→ I love it!

Security Researcher ☺ ☺ ☺

College Professor and course coordinator

Worked on encryption of satellite communication

Worked with police crime-solving unit



JORDAN.BONAGURA@SECUREIDEAS.COM

www.secureideas.com

SecureIdeas

TABLE OF CONTENTS

01

THE BASIC

02

THE PROJECT

03

HANDS ON,
DEMO & CODE

04

AND NOW?

01

THE BASIC

. . . . .

PENTEST METHODOLOGY

Reconnaissance

- Search public sources for information on the target organization.
  - Users, accounts, config information, keys, known vulns, etc.

Mapping

- Study application data and behavior.
  - Architecture, fields, client/server data flow, data types, headers, APIs, sensitive input, etc.

Discovery

- Leverage the accumulated information to find vulnerabilities using both automated tools and manual testing.

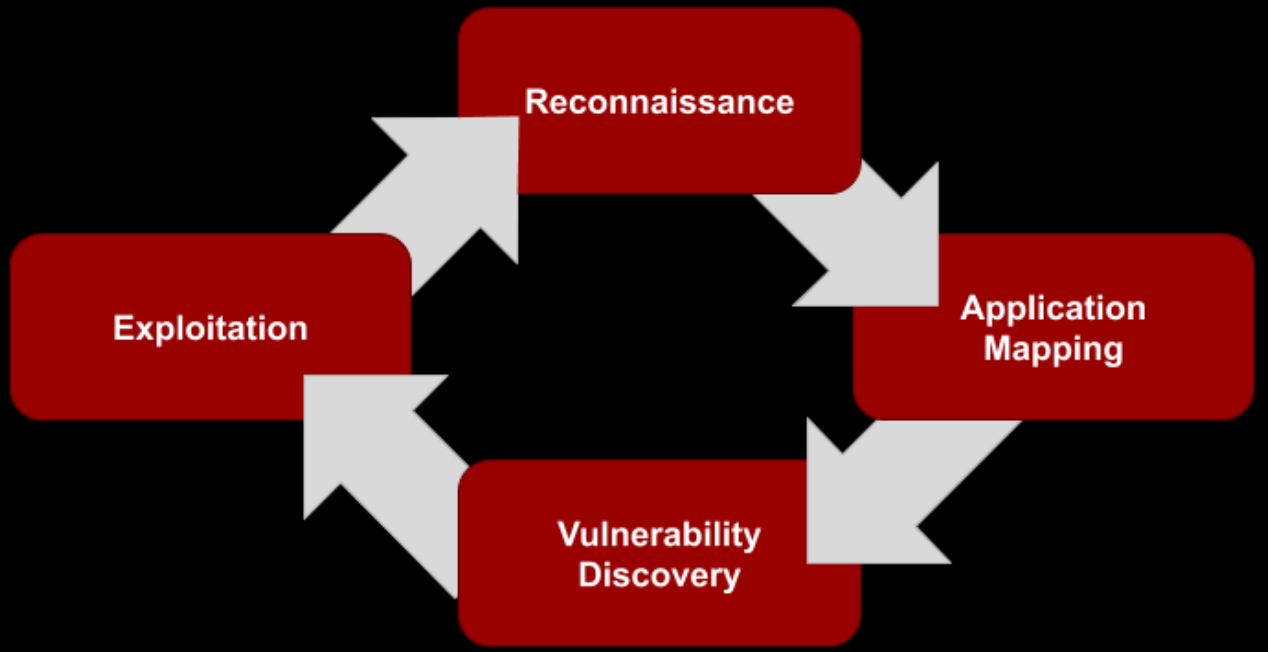


professionally evil®

PENTEST METHODOLOGY

Exploitation

- Exploiting the vulnerability gives a realistic understanding of the business exposure and is often limited to the minimum PoC needed to understand and reproduce the issue.



OK, but WHY do we exploit?







OK, but WHY do we exploit?



Validates the Finding

- Build Proof of Concept
- Steps to reproduce

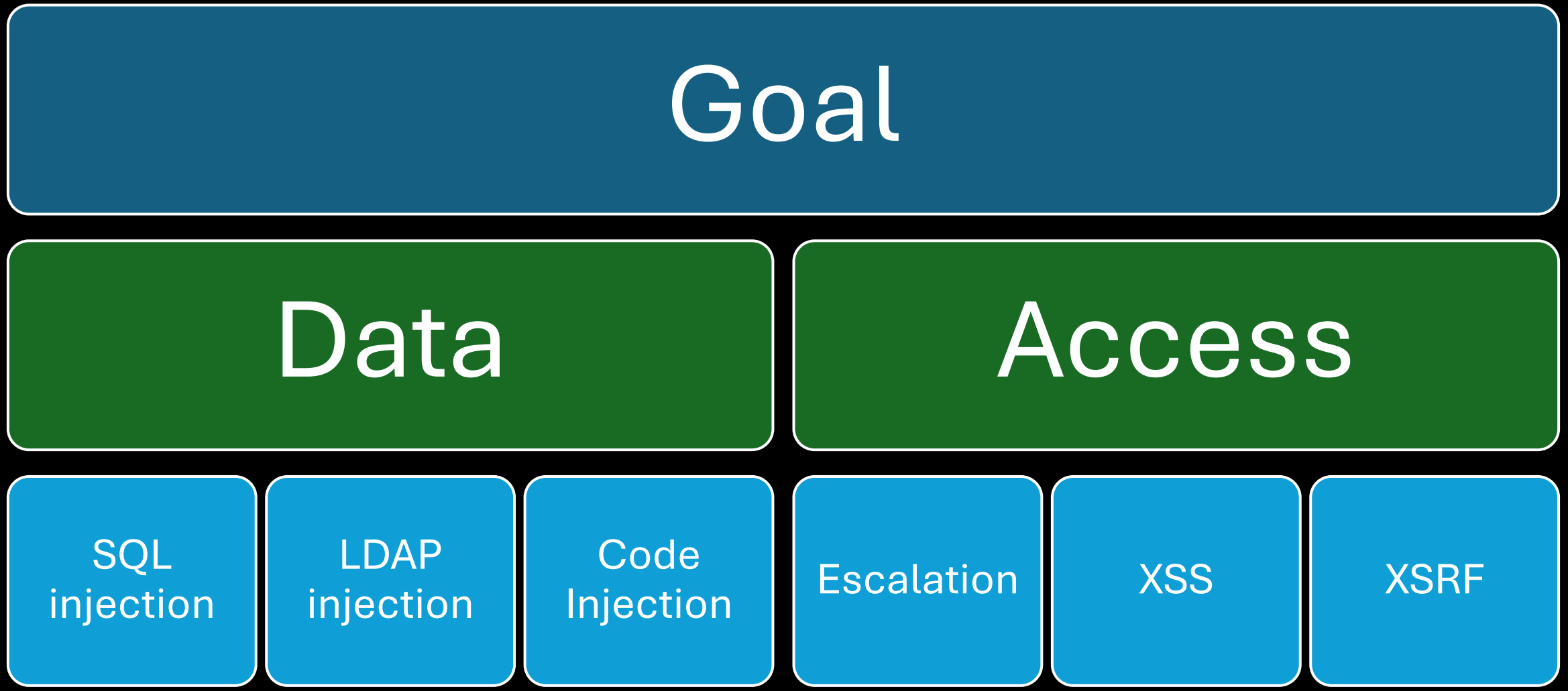
Evaluates the Risk

- Better sense of context
- Are there other (hidden) controls?

Pivot Further

- Gain access to additional targets
- Can be a specific goal of the test

SOME EXPLOIT CHOICES



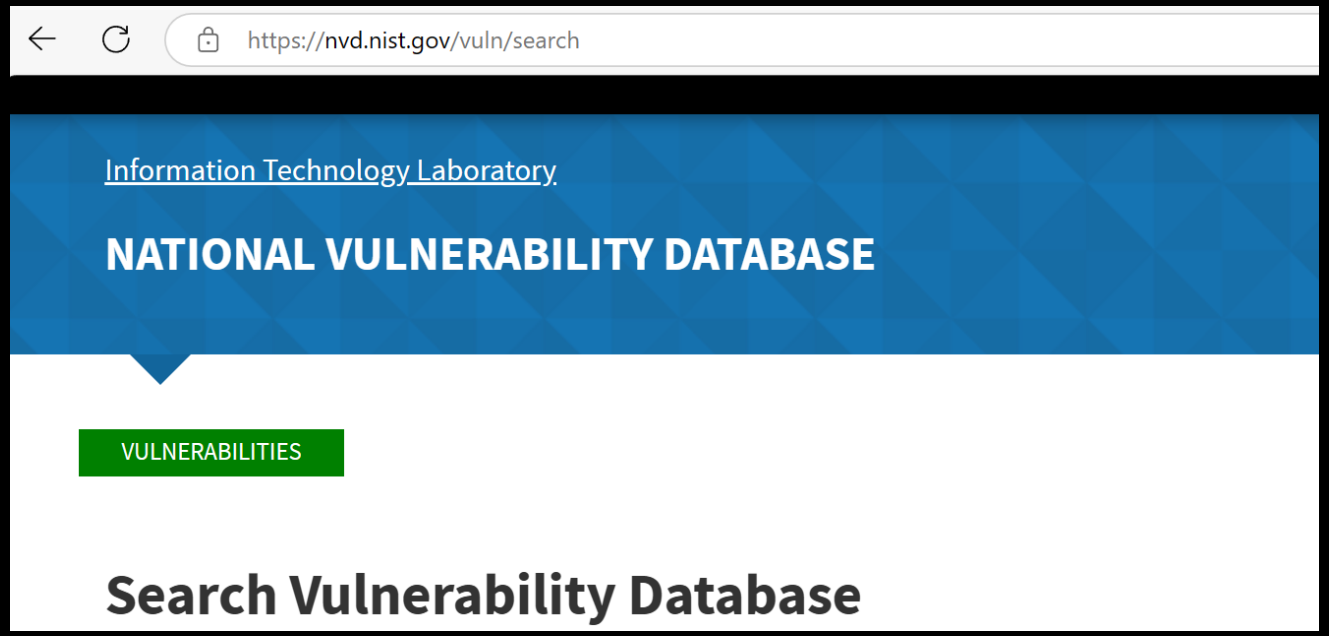
CVE

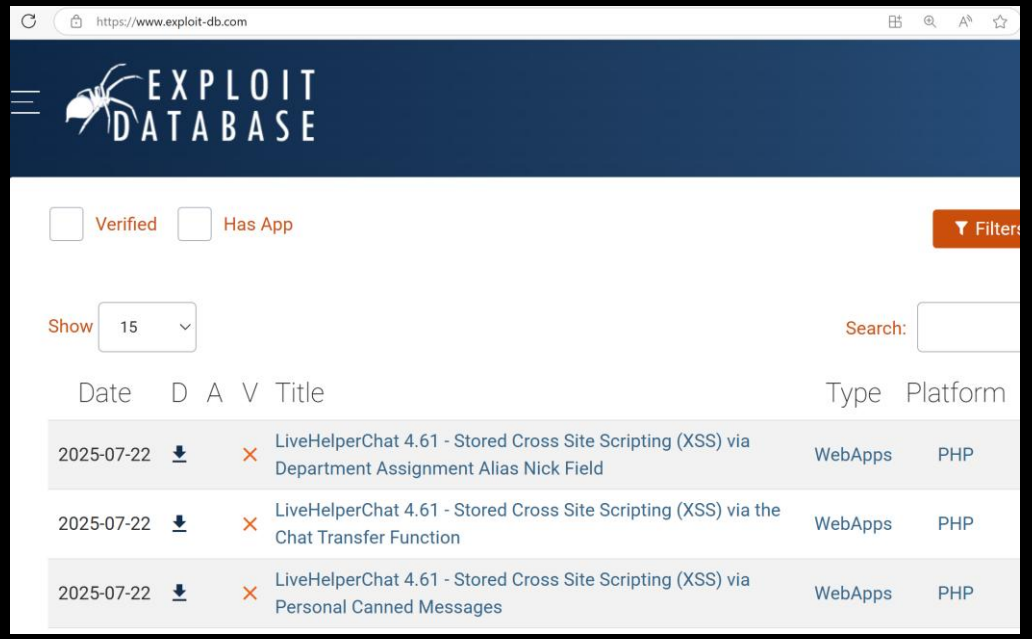
Common Vulnerabilities and Exposures

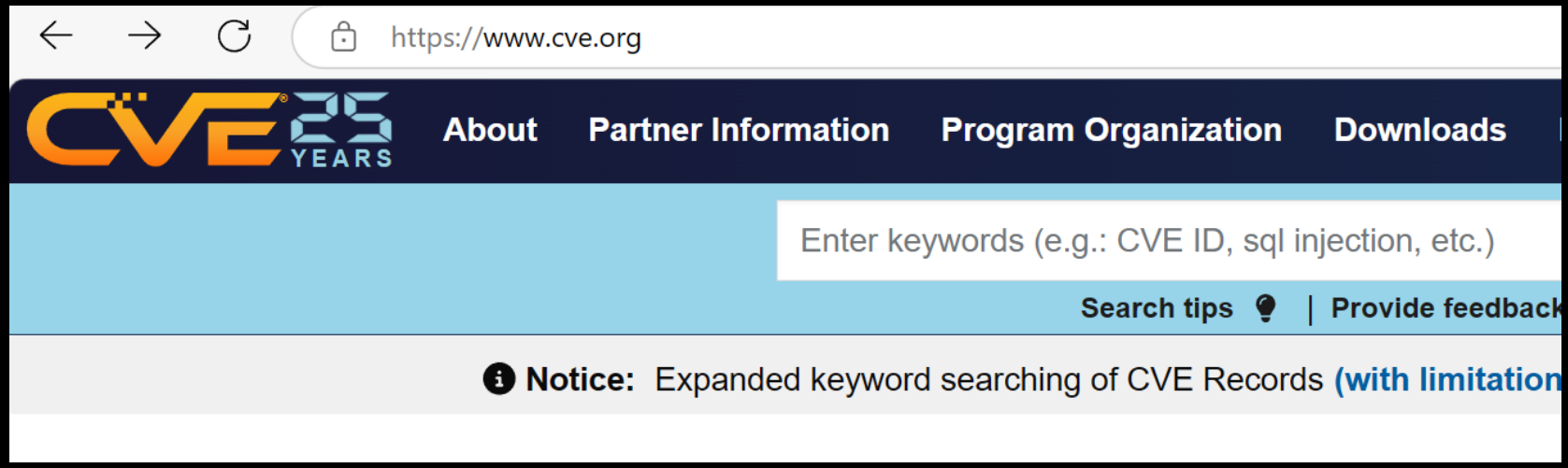
- Publicly disclosed vulnerabilities
  - Managed by MITRE in coordination with U.S. Department of Homeland Security
  - Provides a standardized way for known security vulnerabilities
  - Unique identifier = CVE ID = mask (CVE-YEAR-NUMBER)



CVE – WHERE?

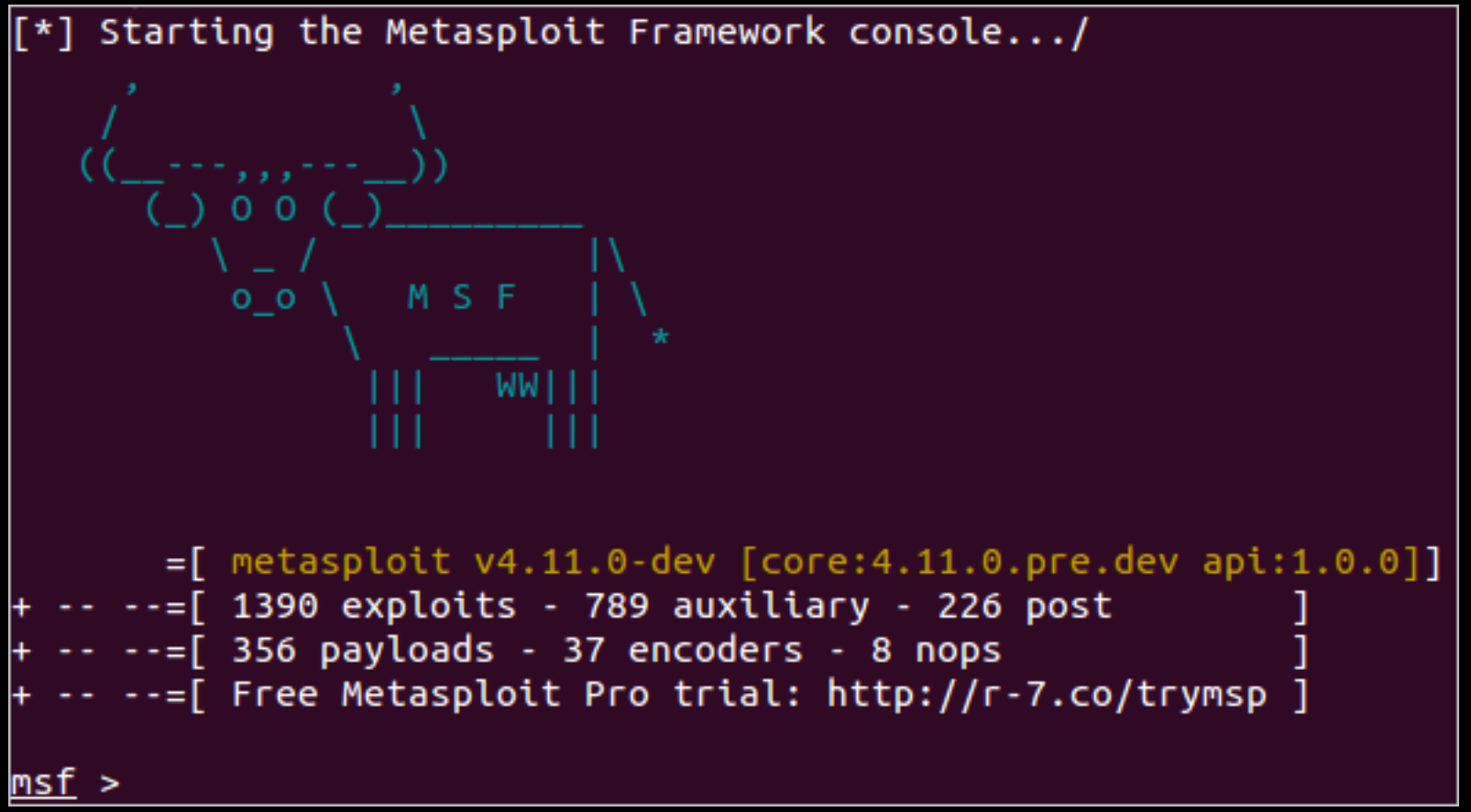

https://nvd.nist.gov/vuln/search
Information Technology Laboratory
NATIONAL VULNERABILITY DATABASE
VULNERABILITIES
Search Vulnerability Database


https://www.exploit-db.com
EXPLOIT DATABASE
Verified
Has App
Filter
Show 15
Search:
Date D A V Title Type Platform
2025-07-22 LiveHelperChat 4.61 - Stored Cross Site Scripting (XSS) via Department Assignment Alias Nick Field WebApps PHP
2025-07-22 LiveHelperChat 4.61 - Stored Cross Site Scripting (XSS) via the Chat Transfer Function WebApps PHP
2025-07-22 LiveHelperChat 4.61 - Stored Cross Site Scripting (XSS) via Personal Canned Messages WebApps PHP


https://www.cve.org
CVE 25 YEARS
About
Partner Information
Program Organization
Downloads
Enter keywords (e.g.: CVE ID, sql injection, etc.)
Search tips | Provide feedback
Notice: Expanded keyword searching of CVE Records (with limitation


GitHub

CVE – I DON'T NEED IT…



CVE – I DON’T NEED IT…

```
msf6 > search MySchool
[-] No results from search
msf6 >
```

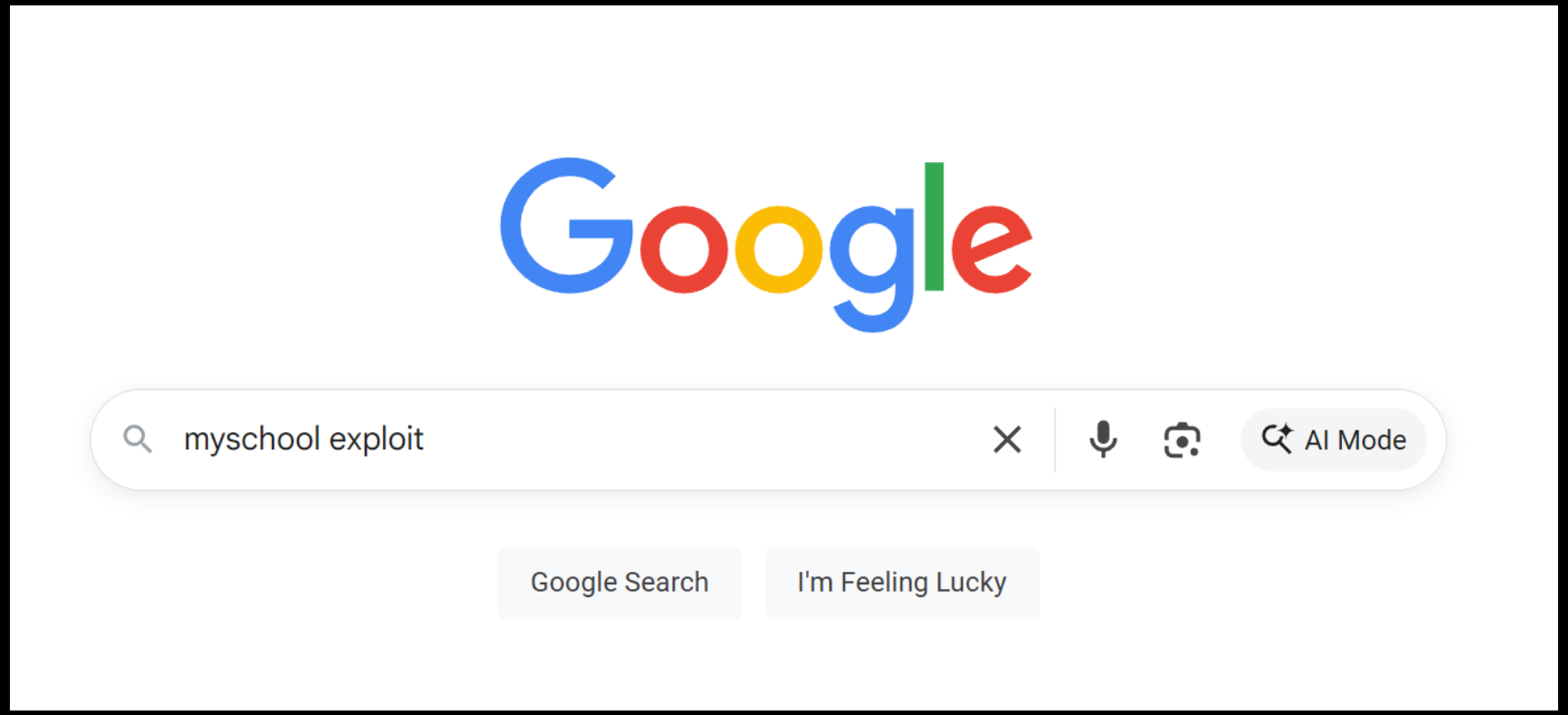




So boring…



So boring…


Google
myschool exploit
AI Mode
Google Search
I'm Feeling Lucky



Wait…



02

THE PROJECT

x x x x x

CVEpwn – CVE Exploit Finder

- **https://github.com/secureideasllc/CVEpwn**

- Security tool designed to **search** for exploits.

  - Multiple platforms (ExploitDB, GitHub, and CXSecurity)
  - Search based on CVEs or Keywords
  - Gathering potential exploits
  - Store results in .txt and .html files

- This tool does not test or execute exploits

CVEpwn - CVE Exploit Finder

- Requirements

  - Python 3.x
  - Required libraries:
    - `requests`
    - `beautifulsoup4`
    - `urllib`
    - `datetime`

CVEpwn - CVE Exploit Finder

Disclaimer

This tool is intended for ethical research and penetration testing purposes only. The author is not responsible for any misuse or illegal activities involving this tool.

0 3

HANDS ON



```
(.venv) PS C:\Users\JordanBonagura\PyCharmMiscProject> python .\cvepwn.py
Please enter the keyword to search for CVEs on NIST: Moveit
Folder 'Moveit_2025-07-16_11-47-57' created successfully.
Working on it...

**********************************************
This is totally an USER responsibility tool.
**********************************************

CVE-2019-18464
Published: 2019-10-31T17:15:10.400
VulnStatus: Modified
Description: In Progress MOVEit Transfer 10.2 before 10.2.6 (2018.3), 11.0 before 11.0.4 (2019.0.4), and 11.1 before 11.1.3 (2019.1.3), multiple SQL Ir
ttacker to gain unauthorized access to the database. Depending on the database engine being used (MySQL, Microsoft SQL Server, or Azure SQL), an attack
e able to alter the database.
Base Score: 9.8
Severity: CRITICAL
Attack Vector: NETWORK
----------------------------------------------
CVE-2019-18465
Published: 2019-10-31T17:15:10.477
VulnStatus: Modified
Description: In Progress MOVEit Transfer 11.1 before 11.1.3, a vulnerability has been found that could allow an attacker to sign in without full creder
rations, and is applicable only if the MySQL database is being used.
Base Score: 9.8
Severity: CRITICAL
Attack Vector: NETWORK
----------------------------------------------
CVE-2021-38159
Published: 2021-08-07T17:15:07.117
VulnStatus: Modified
```

NIST

CVEpwn – CVE Exploit Finder

- Summarize of NIST – CVEs Severity Classification

```
************************************
CVE Severity Classification:
Total:          39
Critical:       11
High:           13
Medium:         14
Low:            1
Not Available:  0
************************************

Do you want to save the results?: Yes(y) or No(n)
```

```
Do you want to save the results?: Yes(y) or No(n)

CVE data has been saved to 'Moveit_2025-07-16_11-47-57/CVEs_Moveit.html'

CVE data has been saved to 'Moveit_2025-07-16_11-47-57/CVEs_Moveit.txt'
```

CVEpwn - CVE Exploit Finder

- Search for **only critical** CVEs exploits on GitHub

```
Do you want to search for 11 CRITICAL CVEs exploits on GitHub? Yes(y) - No(n) - Keyword(k):
```

CVEpwn – CVE Exploit Finder

- It will try to search it respecting rate limits…

```
Do you want to search for 11 CRITICAL CVEs exploits on GitHub? Yes(y) - No(n) - Keyword(k): y
Working on: 1 of 11 - CVE-2019-18464
No repositories found for CVE-2019-18464.
-------------------------------------------------


Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 2 of 11 - CVE-2019-18465
No repositories found for CVE-2019-18465.
-------------------------------------------------


Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 3 of 11 - CVE-2021-38159
No repositories found for CVE-2021-38159.
-------------------------------------------------
```

CVEpwn - CVE Exploit Finder

```
Working on: 4 of 11 - CVE-2023-34362
Searching for repositories with query: CVE-2023-34362


Repository Name: CVE-2023-34362
Description: MOVEit CVE-2023-34362
URL: https://github.com/horizon3ai/CVE-2023-34362
--------------------------------------------------




Repository Name: CVE-2023-34362
Description: CVE-2023-34362: MOVEit Transfer Unauthenticated RCE
URL: https://github.com/sfewer-r7/CVE-2023-34362
--------------------------------------------------




Repository Name: CVE-2023-34362
Description: POC for CVE-2023-34362 affecting MOVEit Transfer
URL: https://github.com/Malwareman007/CVE-2023-34362
--------------------------------------------------




Repository Name: MOVEit-CVE-2023-34362
Description: Repository with everything I have tracking the impact of MOVEit CVE-2023-34362
URL: https://github.com/kenbuckler/MOVEit-CVE-2023-34362
--------------------------------------------------
```

CVEpwn - CVE Exploit Finder

```
Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 9 of 11 - CVE-2024-5805
No repositories found for CVE-2024-5805.
--------------------------------------------------


Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 10 of 11 - CVE-2024-5806
Searching for repositories with query: CVE-2024-5806


Repository Name: watchTowr-vs-progress-moveit_CVE-2024-5806
Description: Exploit for the CVE-2024-5806
URL: https://github.com/watchtowrlabs/watchTowr-vs-progress-moveit_CVE-2024-5806
--------------------------------------------------



Repository Name: CVE-2024-5806
Description: progress moveit cve-2024-5806
URL: https://github.com/sec13b/CVE-2024-5806
--------------------------------------------------
```

Github exploit data has been saved to files: Moveit_2025-07-16_11-47-57/Github_critical_exploits.txt and Moveit_2025-07-16_11-47-57/Github_critical_exploits.html

CVEpwn - CVE Exploit Finder

- Search for **only critical** CVEs exploits on ExploitDB

```
Do you want to search for 11 CRITICAL CVEs exploits on ExploitDB? Yes(y) - No(n) - Keyword(k):
```

CVEpwn – CVE Exploit Finder

- It will try to search it respecting rate limits…

```
Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 4 of 11 - CVE-2023-34362
No exploits found.
No repositories found for CVE-2023-34362.
------------------------------------------------


Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 5 of 11 - CVE-2023-35708
No exploits found.
No repositories found for CVE-2023-35708.
------------------------------------------------


Delaying between requests to avoid rate limits or CAPTCHA...
Working on: 6 of 11 - CVE-2019-16383
Searching for repositories with query: CVE-2019-16383

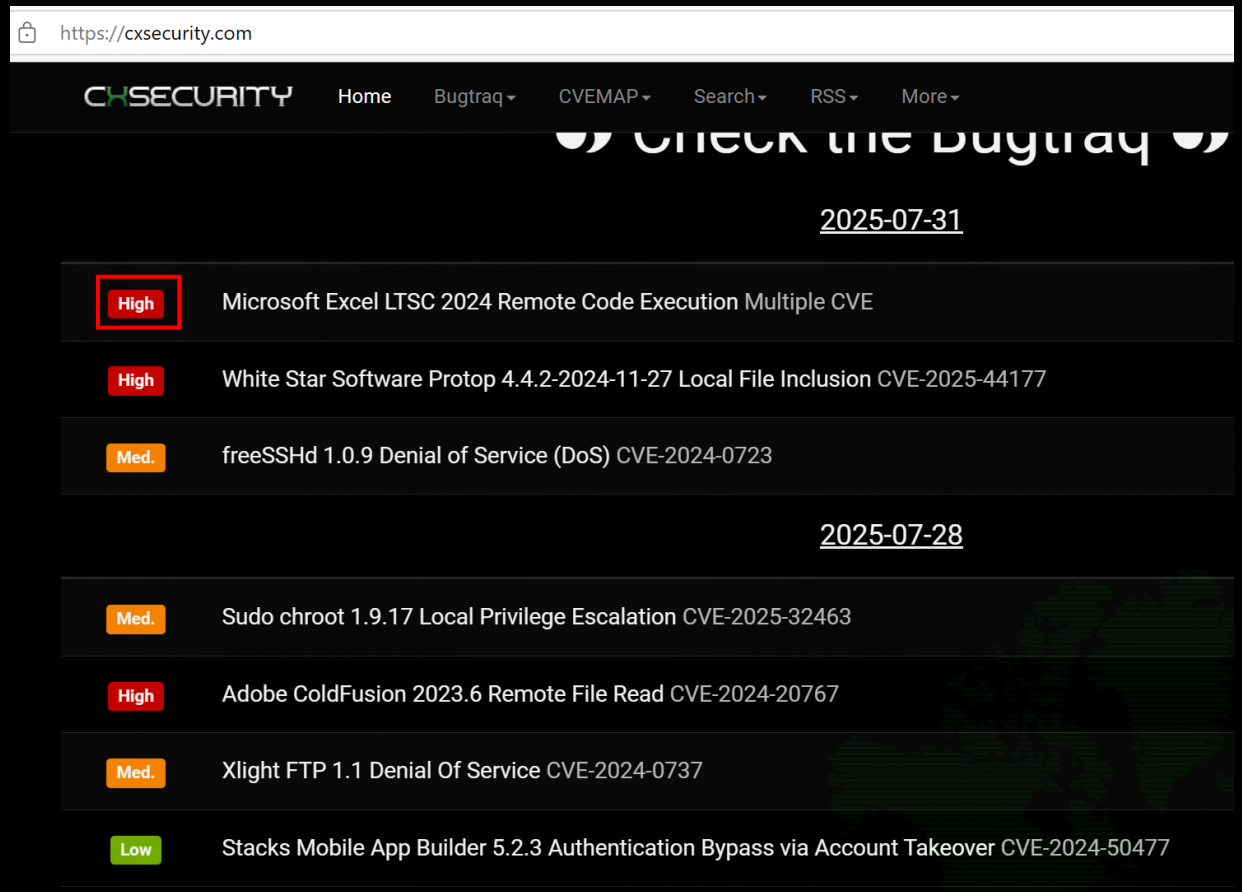
Description: 48316 - MOVEit Transfer 11.1.1 - &#039;token&#039; Unauthenticated SQL Injection
Download Link: https://www.exploit-db.com/download/48316

Repository Name: CVE-2019-16383
Description: 48316 - MOVEit Transfer 11.1.1 - &#039;token&#039; Unauthenticated SQL Injection
URL: https://www.exploit-db.com/download/48316
------------------------------------------------
```

```
ExploitDB data has been saved to files: Moveit_2025-07-16_11-47-57/ExploitDB_critical_exploits.txt and Moveit_2025-07-16_11-47-57/ExploitDB_critical_exploits.html
```

CVEpwn – CVE Exploit Finder

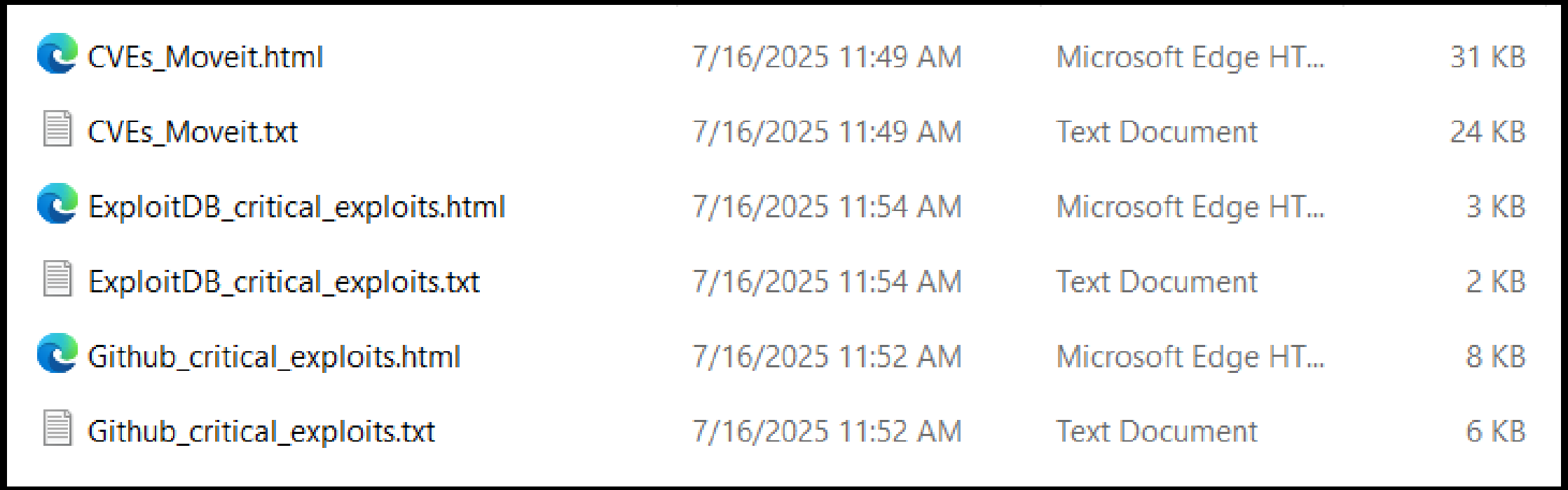
- Search for **High** exploits on CXSecurity




```
Do you want to search for High exploits on CXSecurity? Yes(y) - No(n)y
Please enter the keyword to search for: Moveit
Enter how many pages do you want to scrape on CXSecurity? 5
Proceeding to scrape 5 pages.
Scraping 5 pages now...
Scraping page 1 - URL: https://cxsecurity.com/exploit/1
Scraping page 2 - URL: https://cxsecurity.com/exploit/2
Scraping page 3 - URL: https://cxsecurity.com/exploit/3
Scraping page 4 - URL: https://cxsecurity.com/exploit/4
Scraping page 5 - URL: https://cxsecurity.com/exploit/5
Scraping complete. No results were found for the keyword: 'moveit'.
```

CVEpwn – CVE Exploit Finder

- Generated files:



CVEpwn – CVE Exploit Finder – NIST CVEs (html)

CVEpwn - CVE Exploit Finder

Author: jbonagura

Version: 1.0

Date: 3/2/2025

Description

CVEpwn is a tool designed to search for exploits related to CVE vulnerabilities across multiple platforms like GitHub, ExploitDB, and CXSecurity. It helps security professionals and researchers find potential exploits for discovered vulnerabilities.

Disclaimer

This tool is intended for ethical research and penetration testing purposes only. The author is not responsible for any misuse or illegal activities involving this tool.

CVE NIST Data Results:

CVE-2019-18464

Published: 2019-10-31T17:15:10.400

VulnStatus: Modified

Description: In Progress MOVEit Transfer 10.2 before 10.2.6 (2018.3), 11.0 before 11.0.4 (2019.0.4), and 11.1 before 11.1.3 (2019.1.3), multiple SQL Injection vulnerabilities have been found in the REST API that could allow an unauthenticated attacker to gain unauthorized access to the database. Depending on the database engine being used (MySQL, Microsoft SQL Server, or Azure SQL), an attacker may be able to infer information about the structure and contents of the database or may be able to alter the database.

Base Score: 9.8

Severity: CRITICAL

CVEpwn - CVE Exploit Finder - NIST CVEs (txt)

```
================================================================
                CVEpwn - CVE Exploit Finder
----------------------------------------------------------------
Author: jbonagura
Version: 1.0
Date: 3/2/2025

Description:
CVEpwn is a tool designed to search for exploits related to CVE
vulnerabilities across multiple platforms like GitHub, ExploitDB,
and CXSecurity. It helps security professionals and researchers
find potential exploits for discovered vulnerabilities.

Disclaimer:
This tool is intended for ethical research and penetration testing
purposes only. The author is not responsible for any misuse or
illegal activities involving this tool.

================================================================
CVE NIST Data Results:

CVE ID: CVE-2019-18464
Published: 2019-10-31T17:15:10.400
VulnStatus: Modified
Description: In Progress MOVEit Transfer 10.2 before 10.2.6 (2018.3), 11.0 before 11.0.4 (2019.0.4),
11.1.3 (2019.1.3), multiple SQL Injection vulnerabilities have been found in the REST API that could
unauthenticated attacker to gain unauthorized access to the database. Depending on the database engi
(MySQL, Microsoft SQL Server, or Azure SQL), an attacker may be able to infer information about the
contents of the database or may be able to alter the database.
Base Score: 9.8
Severity: CRITICAL
Attack Vector: NETWORK
```

CVEpwn - CVE Exploit Finder - GitHub (html)

Github Exploits Results:

Searching for repositories related to: CVE-2019-18464

No repositories found for CVE-2019-18464.

Searching for repositories related to: CVE-2019-18465

No repositories found for CVE-2019-18465.

Searching for repositories related to: CVE-2021-38159

No repositories found for CVE-2021-38159.

Searching for repositories related to: CVE-2023-34362

CVE-2023-34362

Description: MOVEit CVE-2023-34362

URL: https://github.com/horizon3ai/CVE-2023-34362

CVE-2023-34362

Description: CVE-2023-34362: MOVEit Transfer Unauthenticated RCE

URL: https://github.com/sfewer-r7/CVE-2023-34362

CVEpwn - CVE Exploit Finder - GitHub (html)

Description: Repository with everything I have tracking the impact of MOVEit CVE-2023-34362

URL: https://github.com/kenbuckler/MOVEit-CVE-2023-34362

MOVEit_CVE-2023-34362_IOCs

Description: CVE-2023-34362-IOCs. More information on Deep Instinct's blog site.

URL: https://github.com/deepinstinct/MOVEit_CVE-2023-34362_IOCs

---

moveit-payload-decrypt-CVE-2023-34362

Description: This shellscript given the OrgKey 0 will parse the header of the base64 artifacts found in MOVEit Logs

URL: https://github.com/toorandom/moveit-payload-decrypt-CVE-2023-34362

MOVEit-Exploit

Description: an exploit of POC for CVE-2023-34362 affecting MOVEit Transfer

URL: https://github.com/errorfiathck/MOVEit-Exploit

---

cve-2023-34362-iocs

Description: CSV File Containing CVE-2023-34362 IOCs

URL: https://github.com/lithuanian-g/cve-2023-34362-iocs

CVEpwn – CVE Exploit Finder – GitHub (txt)

```
============================================================
Github Exploits Results:

Searching for repositories with query: CVE-2019-18464
No repositories found for CVE-2019-18464.
------------------------------------------------------

Searching for repositories with query: CVE-2019-18465
No repositories found for CVE-2019-18465.
------------------------------------------------------

Searching for repositories with query: CVE-2021-38159
No repositories found for CVE-2021-38159.
------------------------------------------------------

Searching for repositories with query: CVE-2023-34362

Repository Name: CVE-2023-34362
Description: MOVEit CVE-2023-34362
URL: https://github.com/horizon3ai/CVE-2023-34362
------------------------------------------------------


Repository Name: CVE-2023-34362
Description: CVE-2023-34362: MOVEit Transfer Unauthenticated RCE
URL: https://github.com/sfewer-r7/CVE-2023-34362
------------------------------------------------------


Repository Name: CVE-2023-34362
Description: POC for CVE-2023-34362 affecting MOVEit Transfer
URL: https://github.com/Malwareman007/CVE-2023-34362
```

CVEpwn - CVE Exploit Finder - ExploitDB (html)

ExploitDB Exploits Results:

No repositories found for CVE-2019-18464.

No repositories found for CVE-2019-18465.

No repositories found for CVE-2021-38159.

No repositories found for CVE-2023-34362.

No repositories found for CVE-2023-35708.

Searching for repositories related to: CVE-2019-16383

CVE-2019-16383

Description: 48316 - MOVEit Transfer 11.1.1 - 'token' Unauthenticated SQL Injection

URL: https://www.exploit-db.com/download/48316

No repositories found for CVE-2023-35036.

CVEpwn - CVE Exploit Finder - ExploitDB (txt)

```
============================================================
ExploitDB Exploits Results:

No repositories found for CVE-2019-18464.
------------------------------------------------

No repositories found for CVE-2019-18465.
------------------------------------------------

No repositories found for CVE-2021-38159.
------------------------------------------------

No repositories found for CVE-2023-34362.
------------------------------------------------

No repositories found for CVE-2023-35708.
------------------------------------------------

Searching for repositories with query: CVE-2019-16383

Repository Name: CVE-2019-16383
Description: 48316 - MOVEit Transfer 11.1.1 - &#039;token&#039; Unauthenticated SQL Injection
URL: https://www.exploit-db.com/download/48316
------------------------------------------------

No repositories found for CVE-2023-35036.
------------------------------------------------
```

CVEpwn – CVE Exploit Finder

- Case 2 – No **critical** CVEs on NIST

```
(.venv) PS C:\Users\JordanBonagura\PyCharmMiscProject> python .\cvepwn.py
Please enter the keyword to search for CVEs on NIST: MySchool
Folder 'MySchool_2025-07-16_12-27-10' created successfully.
Working on it...

**********************************************
This is totally an USER responsibility tool.
**********************************************



************************************
CVE Severity Classification:
Total:          0
Critical:       0
High:           0
Medium:         0
Low:            0
Not Available:  0
************************************
```

CVEpwn – CVE Exploit Finder

- Case 2 – No **critical** CVEs on NIST – GitHub?

```
Since no Critical vulnerabilities were found. Do you want to search for a keyword on GitHub? Yes(y) - No(n)
What's the keyword to search for? MySchool
Working on: 1 of 1 - myschool
Searching for repositories with query: myschool


Repository Name: sekoliko
Description: Sekoliko | MySchool | MonEcole : School management Software.
URL: https://github.com/julkwel/sekoliko
-------------------------------------------------




Repository Name: myschool
Description: Scraper, parser, backed database and data visualisation of the www.myschool.edu.au data.
URL: https://github.com/andrewharvey/myschool
-------------------------------------------------
```



CVEpwn – CVE Exploit Finder

- Case 2 – No **critical** CVEs on NIST – ExploitDB?

```
Since no Critical vulnerabilities were found. Do you want to search for a keyword on ExploitDB? Yes(y) - No(n)y
What's the keyword to search for? MySchool
Working on: 1 of 1 - myschool
Searching for repositories with query: myschool

Description: 17311 - E-Manage MySchool 7.02 - SQL Injection
Download Link: https://www.exploit-db.com/download/17311

Repository Name: myschool
Description: 17311 - E-Manage MySchool 7.02 - SQL Injection
URL: https://www.exploit-db.com/download/17311
-------------------------------------------------
```

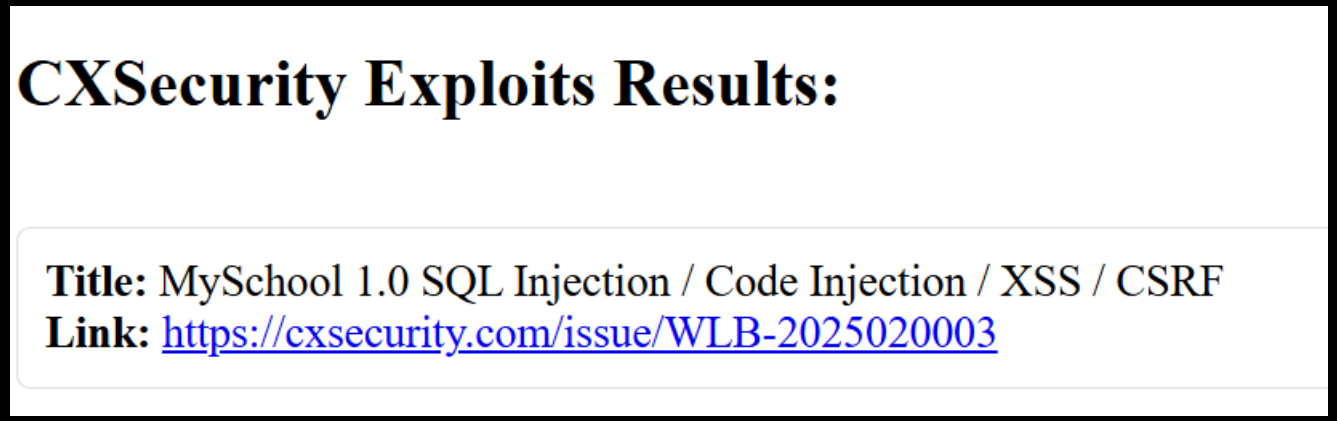
CVEpwn – CVE Exploit Finder

- Case 2 – No **critical** CVEs on NIST – CXSecurity?

```
Do you want to search for High exploits on CXSecurity? Yes(y) - No(n)y
Please enter the keyword to search for: MySchool
Enter how many pages do you want to scrape on CXSecurity? 15
Proceeding to scrape 15 pages.
Scraping 15 pages now...
Scraping page 1 - URL: https://cxsecurity.com/exploit/1
Scraping page 2 - URL: https://cxsecurity.com/exploit/2
Scraping page 3 - URL: https://cxsecurity.com/exploit/3
Scraping page 4 - URL: https://cxsecurity.com/exploit/4
Scraping page 5 - URL: https://cxsecurity.com/exploit/5
Scraping page 6 - URL: https://cxsecurity.com/exploit/6
Scraping page 7 - URL: https://cxsecurity.com/exploit/7
Scraping page 8 - URL: https://cxsecurity.com/exploit/8
Scraping page 9 - URL: https://cxsecurity.com/exploit/9
Scraping page 10 - URL: https://cxsecurity.com/exploit/10
Scraping page 11 - URL: https://cxsecurity.com/exploit/11
Scraping page 12 - URL: https://cxsecurity.com/exploit/12
Scraping page 13 - URL: https://cxsecurity.com/exploit/13
Scraping page 14 - URL: https://cxsecurity.com/exploit/14
Scraping page 15 - URL: https://cxsecurity.com/exploit/15
Results saved in MySchool_2025-07-16_12-27-10/CXSecurity_myschool.html and MySchool_2025-07-16_12-27-10/CXSecurity_myschool.txt
Scraping complete. Found 1 results for the keyword: myschool.
```

CVEpwn – CVE Exploit Finder

- Case 2 – No **critical** CVEs on NIST – CXSecurity Report



CVEpwn - CVE Exploit Finder - CXSecurity (txt)

```
============================================================
CVEpwn - CVE Exploit Finder
------------------------------------------------------------
Author: jbonagura
Version: 1.0
Date: 3/2/2025

Description:
CVEpwn is a tool designed to search for exploits related to CVE
vulnerabilities across multiple platforms like GitHub, ExploitDB,
and CXSecurity. It helps security professionals and researchers
find potential exploits for discovered vulnerabilities.

Disclaimer:
This tool is intended for ethical research and penetration testing
purposes only. The author is not responsible for any misuse or
illegal activities involving this tool.

============================================================
CXSecurity Exploits Results:

Title: MySchool 1.0 SQL Injection / Code Injection / XSS / CSRF - Link: https://cxsecurity.com/issue/WLB-2025020003
------------------------------------------------
```

0 3

DEMO



```
(.venv) PS C:\Users\JordanBonagura\PyCharmMiscProject> python .\cvepwn.py
```

```
(.venv) PS C:\Users\JordanBonagura\PyCharmMiscProject> python .\cvepwn.py
```

```
(.venv) PS C:\Users\JordanBonagura\PyCharmMiscProject> python .\cvepwn.py
```

0 3

CODE

< > < > < > < > < >

```python
import datetime
import requests
import time
import urllib.parse
from bs4 import BeautifulSoup
from pathlib import Path
import re

# Function to print CVE IDs with color based on severity
def print_cve_with_color(cve_id, severity, base_score):...

def save_to_txt(cve_list, keyword, folder):...

def save_to_html(cve_list, keyword, folder):...

def extract_cvss_data(cvss_metric_v2, cvss_metric_v3):...

def process_vulnerabilities(vulnerabilities):...

def display_cve_info(cve_list, severity_count):...

def get_cve_data(keyword, foldername):...

def write_to_file(file, content):...

def fetch_github_repositories(query_list, folder):...

def handle_github_search(severity_count, cve_list, folder):...

def handle_exploitdb_search(severity_count, cve_list, folder):...

def fetch_exploitdb_repositories(query_list, folder):...

def scrape_cxsecurity(folder):...

def main():...
```

CVEpwn – CVE Exploit Finder

- NIST – CVSS Metric (V3 and V2)

```python
def extract_cvss_data(cvss_metric_v2, cvss_metric_v3):  1 usage
    """ Extract CVSS data from v2 or v3 metrics. """
    # Default values
    base_score = None
    severity = "Not Available"
    attack_vector = "Not Available"

    # Use CVSS v3 if available, otherwise fallback to CVSS v2
    if cvss_metric_v3:
        cvss_data = cvss_metric_v3[0].get("cvssData", {})
        base_score = cvss_data.get("baseScore")
        severity = cvss_data.get("baseSeverity", "Not Available")
        attack_vector = cvss_data.get("attackVector", "Not Available")
    elif cvss_metric_v2:
        cvss_data = cvss_metric_v2[0].get("cvssData", {})
        base_score = cvss_data.get("baseScore")
        severity = cvss_data.get("baseSeverity", "Not Available")
        attack_vector = cvss_data.get("attackVector", "Not Available")

    return base_score, severity, attack_vector
```

CVEpwn – CVE Exploit Finder

- NIST – Search

```python
def get_cve_data(keyword, foldername):  1 usage
    print("Working on it...")
    # URL encode the keyword to handle spaces and special characters
    encoded_keyword = urllib.parse.quote_plus(keyword)

    # Build the URL with the user-provided keyword
    url = f"https://services.nvd.nist.gov/rest/json/cves/2.0?keywordSearch={encoded_keyword}"

    try:
        # Send GET request to the API
        response = requests.get(url)

        # Check if the response is successful
        if response.status_code == 200:
            data = response.json()

            # Extract the vulnerabilities list
            vulnerabilities = data.get("vulnerabilities", [])

            # Process vulnerabilities and severity counts
            cve_list, severity_count = process_vulnerabilities(vulnerabilities)

            # Sort the CVE list by base score in descending order
            cve_list.sort(key=lambda x: x['base_score'], reverse=True)

            # Display the CVE information and severity counts
            display_cve_info(cve_list, severity_count)
```

CVEpwn - CVE Exploit Finder

- GitHub - Search

```
        for i, query in enumerate(query_list, 1):
            # URL encode the query
            encoded_query = urllib.parse.quote_plus(query)
            github_url = f"https://api.github.com/search/repositories?q={encoded_query}"
            print(f"Working on: {i} of {total_queries} - {query}")  # Print the GitHub URL to the console
            response = requests.get(github_url)

            if response.status_code == 200:
                data = response.json()

                if data["total_count"] > 0:
                    # Write the query search information to the files
                    html_message = f"<h3>Searching for repositories related to: <strong>{query}</strong></h3>"
                    write_to_file(filehtml, html_message)
```

CVEpwn - CVE Exploit Finder

- ExploitDB - Search

```
for i, query in enumerate(query_list, 1):
    # URL encode the query
    encoded_query = urllib.parse.quote_plus(query)
    pattern_CVE =  r'^(CVE-\d{4}|\d{4})-\d+$'
    if re.match(pattern_CVE,encoded_query):
        exploitdb_url = f"https://www.exploit-db.com/search?cve={encoded_query}"
    else:
        exploitdb_url = f"https://www.exploit-db.com/search?q={encoded_query}"

    headers = {
        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36',
        'X-Requested-With': 'XMLHttpRequest',
        'Accept': 'application/json, text/javascript, */*; q=0.01'
    }
    print(f"Working on: {i} of {total_queries} - {query}")  # Print the URL to the console
    response = requests.get(exploitdb_url, headers=headers)

    if response.status_code == 200:
        data = response.json()

        # Check if there are any results
        if "data" in data and len(data["data"]) > 0:
            exploit = data["data"][0]  # Get the first exploit in the list
```

CVEpwn – CVE Exploit Finder

- CXSecurity – Search

```
def scrape_cxsecurity(folder):  1 usage
    base_url = 'https://cxsecurity.com/exploit/'
    headers = {
        'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36'
    }
    while True:
        # Ask the user to provide the keyword to search for CVEs on NIST
        keyword = input("Please enter the keyword to search for: ").strip()
        # Validate the keyword input
        if not keyword:
            print("Error: You must enter a valid keyword. Try again")
        else:
            break

    while True:
        user_input = input("Enter how many pages do you want to scrape on CXSecurity? ")

        # Check if the input is empty or not a valid integer
        if not user_input.isdigit() or user_input == "":
            print("Invalid input. Please enter a valid number of pages.")
        else:
            varpagecx = int(user_input)  # Assign the valid input to varpagecx
```

CVEpwn - CVE Exploit Finder

- CXSecurity - Search

```
        soup = BeautifulSoup(response.content, features: 'html.parser')

        # Find all rows where the span contains class 'label label-danger' (indicating High severity)
        rows = soup.find_all('tr')

        for row in rows:
            label = row.find('span', class_='label label-danger')
            if label:
                # Extract the necessary data
                link = row.find('div', class_='col-md-7').find('a', href=True)
                if link:
                    title = link['title']
                    # Only save if the title contains the keyword
                    if keyword in title.lower():
                        issue_link = link['href']

                        # Add formatted result to HTML content
                        html_entry = f"""
<div style="margin-bottom: 15px; padding: 10px; border: 1px solid #ddd; border-radius: 5px;">
    <strong>Title:</strong> {title}<br>
    <strong>Link:</strong> <a href="{issue_link}" target="_blank">{issue_link}</a>
```

0 4

AND NOW?

? ? ? ? ?

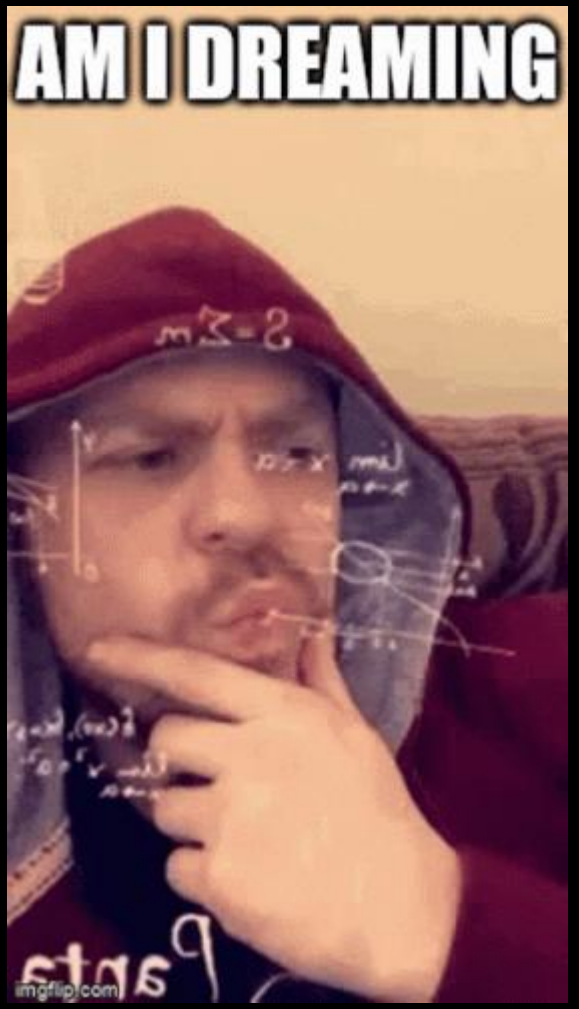
And now ? ? ?



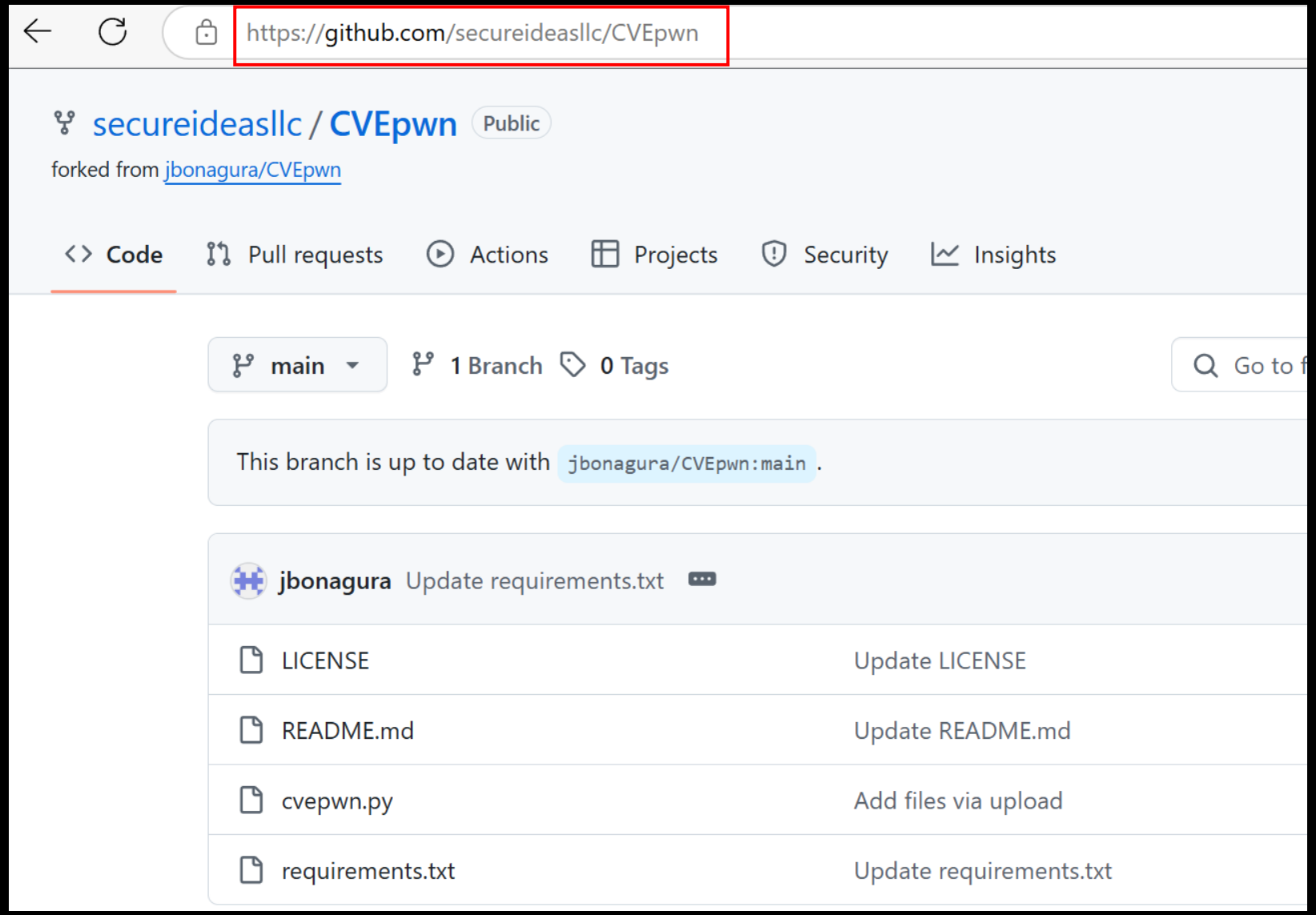
I need you… Why?

V2?

- Add more repo?
- Use command-line flags instead of asking for user input?
- Include the option to search for High severity in addition to Critical?
- Use Nessus’ output file as input for the scan?
- Improve the search to reduce false positive?
- Because **YOU rock** (Brainstorming)



GitHub


https://github.com/secureideasllc/CVEpwn
secureideasllc / CVEpwn
Public
forked from jbonagura/CVEpwn
Code
Pull requests
Actions
Projects
Security
Insights
main
1 Branch
0 Tags
Go to f
This branch is up to date with jbonagura/CVEpwn:main .
jbonagura Update requirements.txt
LICENSE
Update LICENSE
README.md
Update README.md
cvepwn.py
Add files via upload
requirements.txt
Update requirements.txt

QUESTIONS?



professionally evil®

jordan.bonagura@secureideas.com

www.secureideas.com

LinkedIn: Jordan Bonagura

SecureIdeas
professionally evil®