

AUTOMATED URL THREAT ANALYSIS USING VIRUS TOTAL API

Overview:

There are numerous types of cyber threats that we face daily, including malware, phishing, and backdoors. Professionals in the security field can utilize threat intelligence tools such as VirusTotal to determine whether URLs are potentially harmful. The purpose of this report is to present the results of a VirusTotal API analysis carried out on a sample URL (<http://example.com>).

Objective:

- Analyzing & Scanning URLs for possible security risks.
- To understand the threat detection capabilities of VirusTotal and to analyze its scan results.
- Aims to provide practical experience in the use of cyber threat intelligence tools.

Tools:

Kali Linux on Virtual Box, VirusTotal API, Python for API requests, Web browser for validation

Procedure:

- Selected <http://example.com> as the test URL.
- Used Python script to submit the URL to VirusTotal API.
- Execute the script and get the analysis output.
- Verified the scan result on the dashboard of VirusTotal.
- Evaluated the output based on different vendor reports which are displayed on the dashboard.

Python Script for VirusTotal URL Analysis:

```
[
import requests
import json

def scan_url(api_key, url):
    headers = {"x-apikey": api_key}
    data = {"url": url}
    response = requests.post("https://www.virustotal.com/api/v3/urls", headers=headers, data=data)

    if response.status_code == 200:
        scan_id = response.json()["data"]["id"]
        return scan_id
    else:
        print("Error submitting URL")
        return None

def get_report(api_key, scan_id):
    headers = {"x-apikey": api_key}
    response = requests.get(f"https://www.virustotal.com/api/v3/analyses/{scan_id}", headers=headers)

    if response.status_code == 200:
        return response.json()
```

```

else:
    print ("Error fetching report")
    return None

# Replace with your VirusTotal API key
API_KEY = "your_virustotal_api_key"
URL_TO_SCAN = "http://example.com"

scan_id = scan_url(API_KEY, URL_TO_SCAN)
if scan_id:
    report = get_report(API_KEY, scan_id)
    print (json.dumps(report, indent=4))
|

```

Note: add “your_virustotal_api_key” in this section

Scan Summary on VirusTotal:

- URL: http://example.com
- Detection Rate: 0/96 [0 Security Vendors Flagged that this URL is Malicious]
- Community score: -9 [No of Negative User Feedback]
- Status Code: 200 OK [This URL Is active]
- Content-Type: text/html [Standard webpage]
- Last Analysis Date: 8 minutes ago

Crowdsourced context:

- One "low" alert shows that is related to a backdoor via XFF [Cross-Forwarded-For attack].
- From ArcSight Threat Intelligence suggests past discussion of suspicious activity, but no confirmed threats.

Key Observation:

- VirusTotal does not identify this URL as malicious, but some community users have raised concerns.
- No security vendors have flagged the domain, meaning it is likely safe.
- The community score of -9 is negative suggesting past suspicious activity or potential misuse.

Conclusion:

- Although from community-tested URL has raised concerns it was not reported as malicious.
- To detect malware, phishing, and other threats VirusTotal is an effective cybersecurity tool.
- For high-risk investigation we can cross-check with different platforms for example- OpenPhish and URLHaus

Future Works:

- For bulk analysis, automate the retrieval of URLs from phishing feeds.

- Instead of using manual checks, adapt the Python scripts to retrieve results directly from VirusTotal.
- Combine the Google Safe Browsing API to expand this project and detect more threats.

Appendix:

VirusTotal Scan Result Images:

The first screenshot shows the VirusTotal scan results for the URL `http://example.com/`. The interface displays a green circle with the number 0, indicating a clean scan. The status is 'No security vendors flagged this URL as malicious'. The content type is 'text/html' and the last analysis date is '8 minutes ago'. The community score is 96. The 'DETECTION' tab is selected, showing a 'Crowdsourced context' section with a warning icon and text: 'Backdoor via XFF - Mysterious Threat Actor Under Radar - according to source ArcSight Threat Intelligence - 1 year ago'. Below this, it lists contextual indicators: 'The domain's Cisco Umbrella rank is 8898', 'The URL is known benign by Check Point's Threat Cloud', 'The domain is popular among websites with good reputation', and 'The domain is popular in the world'. It also provides a VirusTotal link and a description: 'Legitimate website which does not serve any malicious purpose.' The 'Security vendors' analysis' section is partially visible.

The second screenshot shows the 'Security vendors' analysis' section of the VirusTotal scan results. It displays a table of security vendors and their analysis results for the URL `http://example.com/`. All vendors listed show a 'Clean' result.

Security Vendor	Analysis Result
Abusix	Clean
ADMINUSLabs	Clean
AlienVault	Clean
Antiy-AVL	Clean
benkow.cc	Clean
BlockList	Clean
Certego	Clean
CINS Army	Clean
CRDF	Clean
Acronis	Clean
AILabs (MONITORAPP)	Clean
alphaMountain.ai	Clean
Artists Against 419	Clean
BitDefender	Clean
Blueliv	Clean
Chong Lua Dao	Clean
CMC Threat Intelligence	Clean
Criminal IP	Clean