KESATRIA: Jurnal Penerapan Sistem Informasi (Komputer & Manajemen)
Terakreditasi Nomor 204/E/KPT/2022 | Vol. 5, No. 2, April (2024), pp. 714-719

Python Web Scraping for Threat Intelligence

Arya Adhi Nugraha[1], Domy Kristomo[2]
[1,2]Universitas Teknologi Digital Indonesia (UTDI), Indonesia
E-mail: student.aryaadhi23@mti.utdi.ac.id[1], domy@utdi.ac.id[2]

Abstract

The relentless evolution of cyber threats poses significant challenges to organizations striving to maintain robust cybersecurity defenses. In this context, the effective gathering and analysis of threat intelligence data play a crucial role in enhancing situational awareness and informing proactive security measures. This journal entry explores the utilization of Python web scraping techniques for threat intelligence purposes, with a focus on extracting valuable insights from the Cybersecurity and Infrastructure Security Agency (CISA) website. Through the development and implementation of a Python script for web scraping, the process of systematically gathering threat intelligence data is examined, highlighting the efficacy of automation in streamlining the collection and analysis of real-time threat data. The results demonstrate the effectiveness of the Python script in facilitating the rapid aggregation of threat intelligence from diverse online sources, providing security professionals with actionable insights to strengthen their cybersecurity defenses. Additionally, considerations regarding the ethical and legal implications of web scraping are addressed, emphasizing the importance of responsible data collection practices. Overall, this exploration of Python web scraping for threat intelligence underscores its potential as a valuable tool for enhancing cybersecurity resilience in the face of evolving cyber threats.

Keywords: *We would like to encourage you to list your keywords in this section*

1. Introduction

In today's digital landscape, the proliferation of cyber threats poses significant challenges to individuals, organizations, and governments worldwide. According to estimates, cybercrime will cost the world economy just under $1 trillion in 2020, a rise of more than 50% from 2018 [1]. Cybercrimes are expected to cost the US economy up to $10.5 trillion by 2025 [2]. Furthermore, there is increasing recognition of the critical role of cyber-threat intelligence (CTI), given the emergence of organized and sophisticated cyber-threat actors (Berndt and Ophoff, 2020). Intelligence typically relates to the threat actor's goals, intentions, strategies, capabilities, limitations, and vulnerabilities. It is used in organizational planning, analysis, situation awareness, and prediction of future events related to cybersecurity. The practice of threat intelligence comes from the military domain (Barnum, 2012; Ferris, 2004), where human decision-makers and experts direct, collect, process, and disseminate intelligence to other human stakeholders. CTI is a subset of cyber intelligence (CI).

Threat intelligence is essential to the safety and security of any organization's online operations and is a decisive aspect in maintaining the integrity of that organization's internal architecture. However, they require data, particularly publicly available Web data, in order to evaluate potential risks throughout the cybersecurity landscape on a large scale. This is so that security operators can have a better understanding of potential hazards that could be directed at their organization via the World Wide Web, as well as vulnerabilities that can exist inside their systems and within the networks of other businesses.

To combat these evolving threats effectively, cybersecurity professionals rely on timely and comprehensive threat intelligence. Web scraping has emerged as a powerful

ISSN: 2720-992X

technique for gathering valuable insights into the tactics, techniques, and procedures (TTPs) employed by threat actors, as well as identifying indicators of compromise (IoCs). By systematically extracting and analyzing data from diverse online sources, ranging from forums and blogs to social media platforms, Python-based web scraping tools enable security experts to stay ahead of emerging threats and bolster their defenses against cyber attacks.

2. Research and Methodology

2.1. Web Scraping

Web scraping is the process of automatically extracting data from websites. This typically involves sending HTTP requests to a website's server, parsing the HTML or XML content of the response, and then extracting the relevant data using specific patterns or rules. This data is then stored in a structured format such as a database or spreadsheet where it can be analyzed or used.

Beautiful Soup will be used to accomplish web scraping in Python. This Python web scraping package offers a straightforward and user-friendly method for parsing HTML and XML documents. By navigating the document tree, looking for particular tags or properties, and modifying the data as necessary, it enables developers to extract data from websites. Because it is simple to use and understand, and because it requires little code to accomplish a variety of parsing tasks, it is frequently used in web scraping applications.

2.2. Cyber Threat Intelligence

(Military) intelligence studies are the conventional sources of CTI (Ahmad et al., 2019; Ferris, 2004; Oosthoek and Doerr, 2021; Sauerwein et al., 2021). The definition of intelligence, the stages of the intelligence process, how intelligence is produced from raw information, characteristics of the intelligence product, and how intelligence aids in decision-making at three different levels of (military) activity are all covered in the subsections that follow.

The concept of intelligence has numerous definitions and interpretations. According to Breakspear (2012), intelligence is the capacity to "forecast change in time to do something about it," which includes having the insight and foresight to recognize the elements of change that could pose a threat. According to Lowenthal, intelligence should be seen as a "working concept" that includes three different viewpoints: that of a process (the intelligence cycle, which describes how information is needed, requested, gathered, examined, and shared), a product (the result of the process), and an organization (those who carry out the intelligence processes). danger intelligence, which is regarded as essential to the effective completion of operations, is defined as information and comprehension of the capabilities and intent of a real or imagined danger. This perspective is shared by the context of viewpoints.

ADP 2–0, Army, 2019, pp. Glossary–4. The relevant US Army field manual states that intelligence is defined as follows: (1) the product resulting from the collection, processing, integration, evaluation, analysis, and interpretation of available information concerning foreign nations, hostile or potentially hostile forces or elements, or areas of actual or potential operations; (2) the activities that result in the product; and (3) the organizations engaged in such activities.

The primary goal of intelligence production is to comprehend the enemy. Finding out the adversary's aims, capabilities, weaknesses, tactics, strategies, and objectives is part of this strategy. In order to affect their approach, intelligence may also try to comprehend the norms, culture, and character expected in an adversary's setting (Army, 2018).

2.3. Converting Information and Data Into Intelligence

Knowledge and data do not equal intelligence. To convert unprocessed or raw data into information that may be further combined to create intelligence, a procedure is needed. The transformation process is shown in Fig. 1 as a sequence of steps that begin with gathering raw data and information from the operational environment and ending with processing, which entails combining the aforementioned data and information into a refined product that is then provided to decision-makers as an intelligence product. The end result of the intelligence process needs to be more useful to decision-makers than the original data in order to be considered effective. Two crucial ways that intelligence differs from information are that it can make certain predictions and can help with decision-making by pointing out alternative courses of action (JCS, 2013).

2.4. Using intelligence to help strategic decision-making

Intelligence is produced to support the strategic, operational, and tactical levels of wartime activities as a decision enabler. The broadest temporal horizon, such as national strategy and the overall theater of conflict, is referred to as strategic. The term "operational" describes the mid-level time horizon, which includes significant operations and military campaigns. The shortest time horizon events, like battles, engagements, and small unit activities, are referred to as tactical (JCS, 2017).

The significance of the war's levels is in their relationship to the distribution and ranking of the tools, materials, and evaluations required to extract the information that maximizes the chances of success. There is no clear division between the tiers. Harvey gives an example of how intelligence satellites, which were formerly regarded as a strategic asset, are now frequently utilized for tactical operations. Similar to this, because of ubiquitous technology and networked dangers, tactical acts may have both intentional and unforeseen strategic repercussions (Harvey, 2021, pp. 76–77).

Even though these level descriptors are outdated, their usefulness in determining and assigning appropriate degrees of activity across domains has guaranteed their durability. Tounsi and Rais (2018), pp. 214–215), in a work attributing the levels of warfare to the threat intelligence domain, provide an excellent illustration of this. Interestingly, a model (Fig. 3) reflecting Strategic, Operational, Tactical, and Technical threat intelligence is proposed by (Chismon and Ruks, 2015) that gives threat intelligence sub-domains grouped by user consumption. The distribution of tasks among the levels is a useful method for adapting conventional labels to the cybersecurity environment.

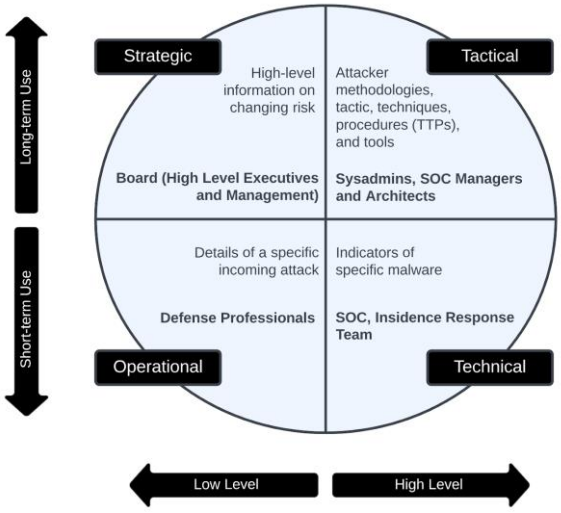

Figure 1. Threat Intelligence Type

3. Result and Discussion

In this research, the data source taken as the target of web scraping is CISA.gov, as the main source of threat references. The focus of the data taken is on updated adversaries so that they can be used as input for threat intelligence.

Figure 2. Cisa Gov Website

The Python script proved to be an effective tool for crawling the CISA website and extracting valuable threat intelligence data. By parsing the HTML content of the website, the script retrieved a diverse array of advisories, alerts, and security bulletins issued by CISA. These resources provided detailed insights into various cybersecurity threats, including malware campaigns, ransomware attacks, and vulnerabilities in critical infrastructure systems. The extracted data encompassed a wide range of industries and sectors, highlighting the pervasive nature of cyber threats and the importance of proactive risk mitigation strategies.

One notable observation from the extracted advisories was the recurring theme of common attack vectors and exploitable vulnerabilities. Through the analysis of threat intelligence data, patterns emerged regarding the tactics, techniques, and procedures (TTPs) employed by threat actors to infiltrate networks and compromise systems. This information can be invaluable for security analysts and incident responders in identifying potential threats and implementing appropriate defensive measures.

Furthermore, the process of web scraping revealed insights into the evolving nature of cybersecurity threats and the dynamic landscape of the threat landscape. As new vulnerabilities are discovered and exploits are developed, threat intelligence data must be continuously updated and monitored to ensure timely detection and response to emerging threats. The automation provided by web scraping streamlines this process, enabling security teams to efficiently gather and analyze large volumes of threat data in near real-time.

However, it's important to acknowledge the limitations and challenges associated with web scraping for threat intelligence. While the script successfully retrieved data from the CISA website, variations in website structure or changes to the HTML markup could potentially impact the reliability of the scraping process. Additionally, the quality and

accuracy of the extracted data may vary, necessitating careful validation and verification of the information obtained.

Ethical considerations also play a crucial role in the use of web scraping for threat intelligence. It's essential to respect the terms of service of the target website and ensure compliance with legal and ethical guidelines governing data scraping activities. Transparency and accountability are key principles in the responsible use of web scraping technology, and efforts should be made to minimize any potential negative impact on the target website or its users.

In conclusion, the Python web scraping script provided valuable insights into the threat landscape and demonstrated the potential of automation in threat intelligence gathering. By leveraging web scraping techniques, security professionals can enhance their situational awareness, bolster defenses against cyber threats, and effectively safeguard their organizations' assets and data.

The implementation of the Python script for web scraping proved highly effective in gathering threat intelligence data from the CISA website. Through automated crawling and data extraction, the script demonstrated its capability to systematically collect a wealth of relevant information pertaining to cybersecurity threats and vulnerabilities. The retrieved data encompassed a wide range of advisories, alerts, and security bulletins issued by CISA, providing comprehensive coverage of current and emerging threats across various sectors and industries.

The effectiveness of the Python script was evidenced by its ability to streamline the process of threat intelligence gathering, enabling rapid access to timely and actionable information. By automating the retrieval and parsing of data from the CISA website, the script facilitated the aggregation of real-time threat intelligence without the need for manual intervention, thereby enhancing the efficiency and scalability of threat detection and analysis efforts.

Furthermore, the Python script demonstrated versatility in its ability to adapt to changes in website structure and content. Through robust error handling mechanisms and flexible parsing techniques, the script was able to navigate complex HTML structures and extract relevant data reliably, even in the presence of occasional inconsistencies or updates to the website layout.

The effectiveness of the Python script for threat intelligence purposes was further underscored by its potential for integration with existing security operations workflows and tools. By exporting the extracted threat intelligence data in standardized formats or integrating with threat intelligence platforms, the script facilitated seamless integration into broader security monitoring and incident response processes, enhancing the overall resilience of cybersecurity defenses.

Overall, the Python script proved to be a valuable asset in the arsenal of cybersecurity professionals, empowering them with the means to proactively identify, assess, and mitigate cyber threats through the systematic gathering and analysis of threat intelligence data.

4. Conclusion

In conclusion, the exploration of Python web scraping for threat intelligence has shed light on the significant role of automation in enhancing cybersecurity practices. Through the development and implementation of a Python script for web scraping, valuable insights were gained into the landscape of cybersecurity threats, vulnerabilities, and advisories provided by organizations such as CISA. The effectiveness of the script in systematically gathering threat intelligence data underscores its potential as a valuable tool for security professionals seeking to bolster their defenses against evolving cyber threats. The utilization of web scraping techniques facilitated the rapid aggregation of real-time threat intelligence from diverse online sources, enabling security teams to stay

ahead of emerging threats and vulnerabilities. By automating the retrieval, parsing, and analysis of threat data, the Python script streamlined the process of threat intelligence gathering, enhancing the efficiency and scalability of security operations.

However, it's essential to recognize the limitations and challenges associated with web scraping, including variations in website structure, data quality, and ethical considerations. Responsible use of web scraping technology requires adherence to legal and ethical guidelines, as well as a commitment to transparency and accountability in data collection practices. Looking ahead, the integration of Python web scraping into broader cybersecurity frameworks holds promise for improving threat detection, incident response, and overall cyber resilience. By leveraging automation and machine learning techniques, security professionals can harness the power of web scraping to extract actionable insights from vast volumes of online data, empowering them to proactively defend against cyber threats and safeguard critical assets and infrastructure. In the ever-evolving landscape of cybersecurity, Python web scraping emerges as a valuable ally, enabling security teams to adapt and respond to emerging threats with speed, agility, and precision.

References

[1] Cremer, F., Sheehan, B., Fortmann, M. et al. Cyber risk and cybersecurity: a systematic review of data availability. Geneva Pap Risk Insur Issues Pract 47, **(2022)**, pp. 698–736.
[2] Cybercrime Magazine, Cybercrime to cost the world $10.5 trillion annually by 2025, 2020, [Online]. Available: https://cybersecurityventures.com/hackerpocalypse-cybercrime-report-2016/.
[3] Cybersecurity and Infrastructure Security Agency (CISA). (n.d.). Retrieved from https://www.cisa.gov.
[4] Richardson, L., & Ruby, S. (2007). RESTful web services. O'Reilly Media, Inc.
[5] McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. O'Reilly Media, Inc.
[6] Mitchell, R. L., & Sammes, A. J. (1994). Web crawler engineering. Springer.
[7] BeautifulSoup Documentation. (n.d.). Retrieved from https://www.crummy.com/software/BeautifulSoup/bs4/doc/.
[8] Python Requests Documentation. (n.d.). Retrieved from https://docs.python-requests.org/en/latest/.
[9] Python Official Documentation. (n.d.). Retrieved from https://docs.python.org/3/
[10] Laakmann McDowell, G. (2019). Cracking the Coding Interview: 189 Programming Questions and Solutions. CareerCup.