Actionable Intelligence-Oriented Cyber Threat Modeling Framework

Bongsik Shin
SDSU
bshin@sdsu.edu

Aaron Elkins
SDSU
aelkins@sdsu.edu

Lance Larson
SDSU
llarson@sdsu.edu

Marc Perez
SDSU
marcperez@gmail

Lance Cameron
SDSU
lancecmrn237@gmail

Abstract

Amid the growing challenges of cybersecurity, the new paradigm of cyber threat intelligence (or CTI) has gained momentum to better respond to cyber threats. there has been one fundamental and very practical problem of information overload. Organizations face this problem in constructing an effective CTI program. We developed a cyber threat intelligence prototype that automatically and dynamically performs the correlation of business assets, vulnerabilities, and cyber threat information in a scoped setting to remediate the challenge of information overload. The tool that embraces automatization functions are frequently termed security orchestration, automation, and response (SOAR). TIME (Threat Intelligence Modeling Environment) conducts SOAR functionality by automatically repeating the full cycle of determining internal assets that are particularly vulnerable to external threats.

1. Introduction

The age of hyper-connectivity has arrived. As its negative consequence, there is no shortage of alarming stories on severe security breaches. The recent crypto-ransomware attacks on numerous business and government organizations for financial extortions highlight just how serious cybercrimes have become. As the Information Technology (IT) horizon continues to expand, especially with the spread of mobile, IoT (Internet of Things), and industrial control technologies generally called SCADA (Supervisory Control and Data Acquisition), the attack surface continues to swell This poses great challenges to organizations. To make the matter worse, threats propagate faster than threat information or threat feeds. This makes it difficult for defenders to keep up with their proliferation in a timely manner. Not surprisingly, cybercrimes have become a large underground economy sector with complex supply chains operating in the shadow of anonymity and obscurity. Amid the growing challenges, the new paradigm of cyber threat intelligence (or CTI) has gained momentum to better deal with cyber threats. Through CTI, an organization can better understand potential adversaries and predict threat/attack methods they will likely use. In addition,, they can build actionable defense models against imminent or looming threats.

Amid the rise of CTI, we developed a cyber threat intelligence prototype that automatically and dynamically performs the correlation of business assets, vulnerabilities, and cyber threat information in a scoped setting called TIME (for Threat Intelligence Modeling Environment). TIME repeats the cycle of: collect internal asset data; gather vulnerability and threat data; correlate vulnerabilities with assets; and derive CTI and alerts of significant internal asset-related vulnerabilities in a timely manner. Corporate assets (especially data/information assets), software and hardware vulnerabilities, and cyber threats are highly dynamic. That is, an organization's own data and other assets, software and hardware vulnerabilities, and cyberspace threats and exploits, continuously evolve, which poses a real challenge to its defenders.

For automatic derivation of significant CTI, the TIME framework takes advantage of several standards from the *National Institute of Standards and Technology* (NIST) intended for the formalization of vulnerability and threat management. TIME is tightly coupled with the NIST Framework designed to improve Critical Infrastructure Cybersecurity. TIME was prompted by the dearth of modeling methods that effectively and efficiently embrace threat intelligence to find countermeasures against threats *in a timely manner* (i.e., prepare defense before a threat actor strikes). TIME is an effort to augment CTI analysts' ability in battling cyber threats by providing *actionable* intelligence that includes such attributes as expected impact (e.g., benign, critical), confidence (e.g., low, high), and operational priority. The modeling method is oriented to facilitate: forward-looking and proactive (rather than reactive) problem solving; timely decision making and deployment of countermeasures; prioritization in risk remediation to face a large number of attack vectors and agile adversaries; active remediation of significant risks rather than passive regulatory compliance; and collaborative problem solving through threat information and intelligence sharing.

URI: https://hdl.handle.net/10125/80160
978-0-9981331-5-7
(CC BY-NC-ND 4.0)

The key contribution of this project is that traditional risk assessments, based on the asset-threat-vulnerability triangulation, can be automated. If necessary, the automation can be scaled vertically and horizontally along with an organization's defense line. Although the automation of threat intelligence discovery is scoped in this project, we underscore that it can be scaled to the enterprise-level. TIME's approach has a potential to reduce attack surfaces, fortify defense-in-depth through improved CTI capabilities, decrease the cycle time of decision making, relieve CTI analysts from information overload, and increase agility of cyber defense capabilities among others.

2. Background: The Rise of CTI

Threat intelligence represents information *actionable* by a particular organization (not just any threat information) as it is particularly relevant to its context [1] [2]. There is a great deal of cyber threat information out there. An organization only needs a relevant subset of it. If adversaries are taking advantage of a particular vulnerability in the Linux OS to penetrate the organizational defense, this threat is not actionable for a business if it purely relies on Windows OS. Threat intelligence should be contextually relevant and be driven by evidence-based knowledge to quickly and accurately address dangers to an organization in an anticipatory manner.

The legendary strategist and philosopher Sun Tzu, in his transcendent book -- *The Art of War*, states that if one knows herself and also her enemies, she will always win; but, if she does not know her enemies, the chance of winning drops considerably. The importance of *knowing enemies*, as an example of situational awareness [3] [1], is also paramount in cybersecurity when the battlefield is cyberspace and organizations play defense against invisible aggressors. Traditionally, the situational awareness aspect of *knowing your enemies* has received little attention in organizational risk management. Sun Tzu's 2400-year-old lesson seems to offer a clue as to why traditional cybersecurity efforts have not been fully successful against increasingly sophisticated adversaries. The logic is straightforward. As threats directly cause damages to organizations, they should drive much of a firm's efforts (e.g., resource allocations, prioritization of countermeasures, and etc.) in forming the defense strategy. There is evidence that the orientations of organizational cybersecurity efforts have not been balanced.

First, traditional approaches and defense solutions have been primarily reactive (e.g., fixing vulnerabilities when breaches occur). They have focused on covering a broad spectrum of threats in a generalized manner, which typically includes setting up general defense/packet filtering rules on host/network firewalls, implementing intrusion detection systems, and offering general user security training. All of which are largely aligned with the traditional defense paradigm [1]. The generalized measures are extremely important. Numerous publicized incidents underscore this approach. The '*prepare ourselves'* against unknown/less-known enemies through general defense measures and then '*wait and see'* hoping for nothing to happen, is not enough. This approach is able to fight off serious attacks. The more *proactive, anticipatory*, and *targeted* counter moves are, the more likely they are able to preempt threats and to lower security failure rates [4]. It has been repeatedly pointed out that the traditional defense approach is prone to expose organizations to vulnerabilities in dealing with more sustained, serious forms of threats/attacks, such as *advanced persistent threats* [5] [6] [7].

Second, organizations have been placing much more emphasis on regulatory compliance. This naturally causes the threat aspect of risk management to receive much less attention [8]. In scholarly research, compliance is one of the most frequented themes. Current studies examine compliance from such perspectives as security standards [9]; Sarbanes-Oxley Act [10]; and organizational policies [11] [12]. Complying with the laws, regulations, standards, directives, and policies is so fundamental that failure can result in crippling consequences. It also puts accountability on non-complying organizations or employees. This forces them to do a better job in protecting key assets (e.g., health records). However, compliance is a necessary condition and hardly sufficient for safety from threats. In fact, compliance may engender a false sense of security when organizations need to do much more than simply fulfilling the compliance requirements.

Third, popular risk modeling frameworks/methods are designed to facilitate the *know yourself* process (i.e., understand internal weaknesses and risks). This largely fails to include threat intelligence elements in guiding organizational efforts. Traditional threat modeling is done in terms of asset-based, software-based, or attacker-based, and it is not difficult to observe their orientations toward internal risk assessment/discovery [13]. Asset-based threat modeling (e.g., OCTAVE [Operationally Critical Threat, Asset, and Vulnerability EvaluationSM]) is internal asset focused as well. Software-based threat modeling (e.g., OWASP [Open Web Application Security Project[) aims to address security issues associated with software applications deployed. The attacker-based threat modeling that

factors in the attackers' motivation, such as STRIDE, which is an acronym for six threat categories: Spoofing identity, Tampering with data, Repudiation threats, Information disclosure, Denial of service and Elevation of privileges, places much focus on software applications.

CTI is a movement to strike a balance between 'know your invisible enemies' and traditional cybersecurity management. It tends to put more weight in understanding internal weaknesses or vulnerabilities [1]. CTI goes beyond conventional risk-management orientations designed to improve 'general readiness' against known or unknown threats. It does this by remedying internal weaknesses or vulnerabilities [14]. Knowing an organization's enemies demands anticipatory preparations based on the acquired knowledge of adversaries (e.g., motivations) and TTPs (techniques, tactics, and procedures) they use. With the realization that the general security readiness approaches are not enough to effectively mitigate threats, organizations have increasingly embraced CTI [15] [16].

3. Related Works: SOAR

A special challenge of building a CTI program at an organization is information overload. More organizations are increasingly adopting CTI technologies and commercial CTI services [7] [17]. The adoptions are particularly in sectors such as finance, aerospace, defense, government, and IT. They have a lot to lose from cyber breaches. There has been one fundamental and very practical problem of information overload organizations face in constructing an effective CTI program. There is the inundation of threat (e.g., indicators of compromise such as malware hashes, IPs, domains, DNS) and vulnerability (e.g., defects in software design) information from open-source sites (called OSINT) and subscription-based services (e.g., IBM's x-Force) [18].

The information overload situation gets worse as CTI practitioners also need to examine (e.g., correlate) the log and alert data produced by internal network nodes and defense systems such as the firewall and intrusion detection system (IDS). There are practical challenges for the CTI analyst to effectively process this much internal and external data. For this reason, large companies maintain a CTI team [1][16]. Further, practically the only way of handling the information overload situation is the smart automation of CTI activities [19] [20]. There are however significant challenges in automating CTI. This includes the feeding of unstructured data from different CTI data sources and heterogeneity of data attributes gathered from various systems deployed.

Despite the challenges, the practical need and urgency for automating cybersecurity functions drives the rise of the security orchestration, automation and response system (SOAR). Although the SOAR system's architecture continues to evolve, it includes various system features. This includes the integration of heterogeneous security systems, orchestration of workflows, event management, automation, and case management [19]. This way SOAR can improve security analysts' or SOC's performance in various dimensions such as time to detect, time to respond, time to qualify and time to investigate [19] [20]. Amid the constant influx of large data to analyze, organizations are increasingly adopting AI/ML for automating cybersecurity functions [21]. The automation can be implemented in many ways. This includes anomaly detection and predictive intelligence [22] [23], alert triage [24], threat intelligence collection [25], event classification [26], event correlation [23], activity and event prioritization [26] [27], and incidence response [28] [21]. [20] summarized various SOAR solutions commercially available and their platform capabilities in automating *CTI, identification, containment, eradication,* and *recovery* functions.

4. High-Level Systems Architecture

We develop knowledge and build an artifact to tackle a problem the cybersecurity industry faces -- the inundation of data/information. This data needs to be processed in time to uncover actionable CTI particularly germane to an organization. An obvious approach to the information overload problem is to automate the CTI discovery process [18]. Despite its side effects noted, our research aims to develop TIME to "automate" the correlation between vulnerability, threat information and assets of a firm. The development is intended to implement a function that could become an essential element of SOAR.

To facilitate the efforts, TIME is tightly coupled with Security Content Automation Protocol (or SCAP) standards from NIST. Various open standards have been announced under the umbrella concept of SCAP, a multi-purpose framework aiming to automate security controls. This includes vulnerability, threat management, and compliance checking. There are several well-known SCAP standards. CVSS (Common Vulnerability Scoring System) and CWSS (Common Weakness Scoring System) rate the severity of software vulnerabilities and weaknesses discovered. CVE (Common Vulnerabilities Enumeration) lists publicly known software vulnerabilities. CAPEC (Common Attack Pattern Enumeration and Classification) is a comprehensive dictionary of known attack patterns cyber adversaries

use to exploit chosen targets. CCE (Common Configuration Enumeration) recommends secure configuration of software products and helps identify system misconfigurations. CWE (Common Weakness Enumeration) is a dictionary of software weakness types. CPE (Common Platform Enumeration) provides a structured naming scheme for software packages and hardware devices. There are other standardization efforts such as the STIX language for CTI sharing. While they are intended to facilitate automation of the CTI discovery and sharing processes, cybersecurity practitioners still heavily rely on the manual processing of threat and vulnerability data to uncover relevant CTI [20]. This work is an effort to partly fill the void.

Figure 1 summarizes a high-level view of TIME's system components and external data sources that continuously supply CTI-derivable data to TIME. The TIME framework can be adapted to integrate with multiple CTI sources, whether they are open-source or not. The current implementation was developed and tested using AT&T's AlienVault Open Threat Exchange repository and IBM's X-Force Exchange. The AT&T and IBM sites provide highly recent cyber threat and vulnerability information that can be pulled by the TIME system. Integration with other CTI sources comes down to understanding what the repository provides in their CTI and the structure of the information returned when CTI from the source repository is retrieved.

TIME also scans the local network and its end points (i.e., client and server computers) to obtain software and hardware assets deployed on each device (i.e., endpoint). TIME has logic to use a three-way triangulation and correlation logic among threats and vulnerabilities. These are reported by external sources. Software and hardware assets installed on endpoints have a logic to automatically derive internal assets that are pronounced to have vulnerabilities by AT&T's AlienVault and IBM's x-Force. For the automation, TIME also downloads data (e.g., CVE, CPE, CAPEC) from SCAP database sources and stores them in the local database. Additionally, TIME has a GUI module (Grafana-based) that displays the dashboard-style summary of alert data if there is a significant vulnerability found from an existing asset.

Data from AT&T's AlienVault and IBM's x-Force Exchange are much more dynamic than SCAP reference sources. Their threat (e.g., IP, malware hash) and vulnerability (e.g., newly found Windows vulnerability) information is more frequently updated whenever there is new information. In contrast, changes in SCAP reference data (similar to dictionaries) remain relatively more static.

5. Research Method

At the high-level, the objective of this research is to empirically prove the **technical feasibility of automating** the following process through a proof of concept: (1) periodically downloads vulnerability and threat data from two external sources (AT&T's AlienVault and IBM x-Force), (2) identify records related to a software vulnerability or vulnerabilities and extract essential information, (3) pull software and hardware asset information from the end-points of a local organization, and (4) correlate the results of (2), (3), and SCAP standards to determine local assets particularly vulnerable to cyber-attacks. The information of SW and HW vulnerabilities of internal assets are tied to the CVE database. To develop the TIME prototype, a number of artifacts have been produced to facilitate the prototype design and implementation while undertaking the project. They include: (1) data flow diagram, (2) entity relationship diagram, (3) a Postgres database, (4) metadata of database tables and attributes, (5) pseudo codes, (6)

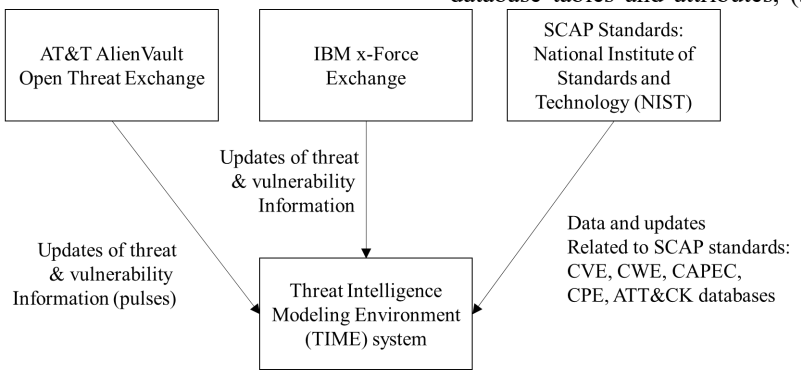

Figure 1. Relationship between the TIME framework and external data sources

various scripts including APIs and database queries, and (7) graphical user interface for alerting threats and vulnerabilities in a dashboard style.

6. TIME System Modules

To enable the automated alert of vulnerable assets (e.g., applications, operating system) and its possible attack vectors (as possible threats), the TIME system consists of a number of input components and they are summarized in Figure 2. These components are conveniently categorized into layers and described in this section.

6.1 SCAP Database Layer

The SCAP layer represents external databases of different SCAP standards. These include CVE, CWE, CPE, CAPEC, and ATT&CK. The ATT&CK framework is a knowledge base of tactics, techniques, and procedures (TTPs) used by cyber attackers. Although not a SCAP standard, ATT&CK offers rich perspectives in understanding how a particular vulnerability could be exploited by adversaries. The CVE standard includes CVSS ratings. They are reference databases that enable the extraction of CTI based on the correlation of CVE identifier(s) from online sources (e.g., AlienVault) and CPE identifier of an internal asset.

6.2 Threat/Vulnerability/SCAP DB Update Layer

The threat/vulnerability/SCAP update layer includes functions that routinely poll online sources (i.e., AT&T AlienVault, IBM x-Force, and NIST) and downloads new threat and vulnerability updates. They are in the form of CTI reports. It also updates the SCAP databases. The TIME framework obtains batches of CTI reports every 2 hours (configurable) from the AT&T AlienVault and IBM x-Force repository. In AT&T AlienVault, the CTI report is provided in the form of 'pulses.' The pulse represents a collection of Indicators of Compromise (IOCs) related to potentially malicious activities. Pulses are created by the AlienVault's research team and other community members of Open Threat Exchange (or OTX). OTX is a threat data sharing platform. Information obtained from NIST's SCAP sources is populated to the backend CVE, CVSS, CPE, and ATT&CK to assist in the production of alerts powered by TIME's internal correlation logic. The backend also stores CWE and CAPEC html views for offline viewing to provide reference information that supplements a particular alert.

The CTI report from online sources (e.g., AT&T AlienVault) includes information pertaining to threats discovered in the wild (i.e., world wide web) up to a few months before the report was released. This information contains (1) the name of the malware tool(s) and/or the threat actors and (2) the general

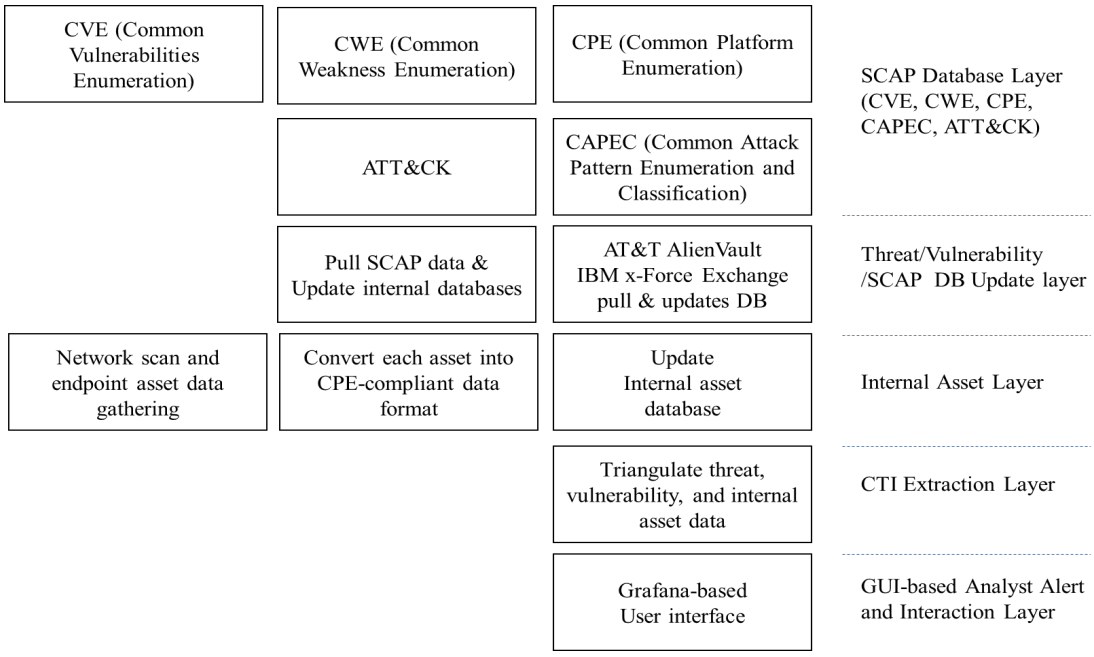

Figure 2. TIME system components and function modules

indicators of compromise associated with the threat. These indicators of compromise include CVE identifiers, file hashes, IP addresses, hostnames,

domain names, URLs, and other signature-based components. These assist security analysts in the detection, mitigation, and removal of threats in their local environments.

6.3 Internal Asset Layer

The internal asset layer scans the endpoints (clients and servers) over the network and gathers asset data (i.e., applications, operating systems, and hardware) per device. The only requirement is that each installed asset is captured and parsed to obtain product information, version information, manufacturer/vendor information, and system architecture. TIME includes a functional module to 'convert' the obtained data to their Common Platform Enumeration (CPE)-compliant, well-formed named strings, and store them into a database. The current implementation of the asset information collection module was developed for and tested on MS Windows 10 endpoints using a PowerShell script that collected system information, formatted it according to the CPE specification, saved the information locally, and finally sent the information to the TIME backend server for further processing. The files created for sending to the backend server included the asset device name. This is a unique identifier provided within the system information for the device. To continuously monitor for new installations, a schedule can be established to either (a) remind for a manual run of the script necessary to collect the information or (b) automatically run a script that will collect the information and subsequently send it to the TIME backend.

6.4 CTI Extraction Layer

The CTI extraction layer derives actionable CTI by correlating threat, vulnerability, and internal asset data, and by providing related reference information (if requested by CTI analysts) through SCAP databases. The TIME framework focuses on using the CPE and CVE standards for associating vulnerability information from X-Force and AlienVault with applications, operating systems, and hardware assets installed on the internal client or server device. In producing CTI alerts based on the TIME's correlation

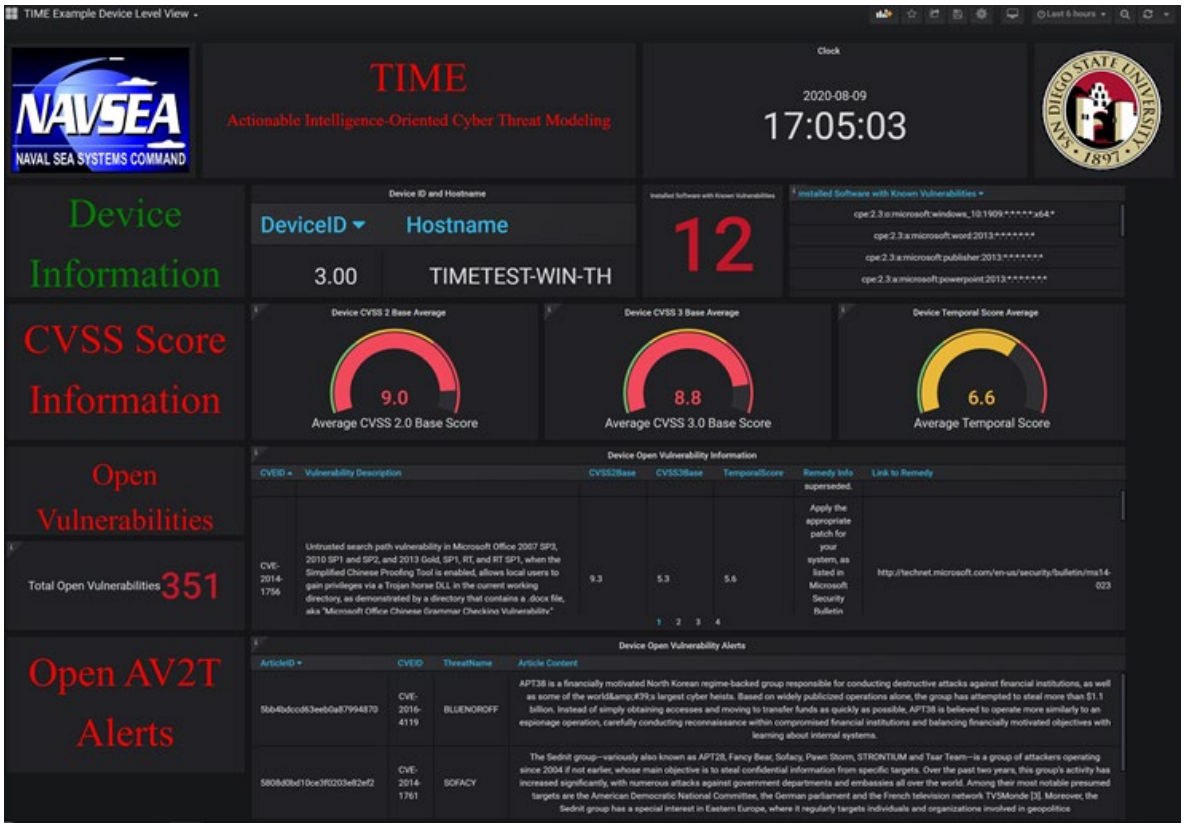

Figure 3: Dashboard views of an asset vulnerability for threat alert

logic, the CVE identifier (e.g., CVE-2021-3462: A privilege escalation vulnerability in Lenovo Power Management Driver) obtained from online sources plays a triggering role.

The module parses three key pieces of information from the online CTI report and temporarily stores them for analysis. First, the CVE identifier(s) from the CTI report is used to query the backend CVE/CPE association table. All CPEs associated with the CVE identifier are obtained and stored temporarily in a list data structure. Second, the module iterates through each CPE in the list, and compares them against CPEs associated with local device assets. If a match is found, the module will create an alert table that will include the CTI report, the CVE identifier from the CTI report, the threats and/or malware tools from the online CTI report, and the local device that contains the vulnerable installation. The alert allows CTI practitioners to become aware of local device assets with vulnerable installations as mentioned in the CTI report. It then provides them with relevant information to begin mitigating the risk immediately. Third, during the alert creation, the module also proceeds to obtain the relevant CWE, CAPEC, and ATT&CK information related to the CVE identifier(s) obtained from the CTI report (i.e., pulse). This process allows the system to automate the process of gathering known information about a specific vulnerability and constructing views related to that information on demand.

6.5 Graphical Interface Layer

Finally, the GUI-enabled interaction layer organizes the alert and other related information. It then presents it to the CTI analysts in a dashboard style. Figure 3 demonstrates sample alert screenshots produced in a dashboard style for CTI analysts.

7. TIME Development

The overview of TIME's development and the testing platform is provided in Figure 4. An overview of the technologies utilized for the TIME system development is provided below. The TIME framework backend was developed on a Linux machine, running Ubuntu Server 18.04.5, with 64 GB of RAM (as a Virtual Machine). PostgreSQL 10.0 for Linux was the database used to develop the backend framework for TIME. Python 3.8.2 and GCC 7.5.0 were used to develop the scripts and compile code on the server. PowerShell 5 Desktop Edition was used to create the script for gathering Windows endpoint-installed software and installed hardware product identifying information. Grafana open-source software was used to generate dashboard views for the TIME framework backend. Researchers used OpenVPN Connect, PuTTY, and WinSCP to remotely connect to and develop the TIME framework server

The testing environment was established on the same subnet as the development server. TIME was tested using MS Windows 10 Build 1909 Win32NT. Three virtual Windows 10 machines were created as endpoints to test framework capability. The same version of Windows was installed on all three virtual machines. Each Windows machine had various software products installed and varying configurations to make each device appear unique during testing,. and to evaluate the overall functionality of the various parts of the framework. All installed products were common installations found on Windows 10 devices. The computer systems used in validating TIME were all deployed via the VMWare hypervisor (see Figure 4).

The TIME server includes several components in a Linux environment. TIME functionality was coded in Python, an interpreted, object-oriented, high-level programming language. Postgres Database was used to store all the threat, vulnerability, and asset data within the TIME Asset Repository. The TIME Asset Repository is where all PC clients aggregate their asset information after the scripts are executed. Once the data from the scripts are uploaded, it is then imported into the TIME Postgres Database.

The PC Client 2 and Client 3 Virtual Machines are installed with Windows 10 and various applications for simulating asset data. This asset data will eventually be loaded into the database server. PC Client 1 Virtual Machine shares the same functionality as PC Clients 2 and Client 3, except for the added Grafana functionality that allows for displaying TIME solutions in a dashboard environment for easy visualization (See Figure 4). Grafana is accessed via web browsers and has direct connectivity to TIME data in the Postgres Database Server.

8. Discussion & Conclusion

8.1 TIME Performance

When there are practical difficulties in deriving advanced CTI by automatically correlating threats, vulnerabilities, and internal assets, the focus of the project was empirically showing its technical feasibility. In this proof-of-concept, the performance assessment evaluates whether TIME can successfully conduct the life cycle of all necessary activities in automation. This includes downloading CTI data from external sources, and identifying attributes with CVE information and other important items included in the

record. It also includes determining software platforms affected by CVE, gathering asset data, and converting them into well-formed CPE records. It also includes matching external data with CPE-compliant internal assets. Another step includes deriving additional references from SCAP databases when significant vulnerabilities of internal assets are found based on recursive DB queries. Lastly, it includes producing and displaying alerts of the highly vulnerable assets and supporting information (e.g., severity of vulnerability, related ATT&CK information). In the repeated empirical tests, when live external source data relevant to simulated internal assets are downloaded, TIME was able to automate the whole activity cycle without manual involvement.

8.2 Merits

The resulting TIME system has several merits. *First, growing cyberattacks underscore that threat modeling should fortify its effectiveness and relevance by adding actionable intelligence components and TIME explores a potential solution*. Threat modeling is a process used by security planners to assist in preparing security measures for protecting systems, networks, and/or assets. There are three primary approaches of threat modeling: software-centric, asset-centric, and attacker-centric. Their limitations have been noted [13]. Today, attacks are coming from every angle with various motives and consequences. When embracing threat intelligence derived from various objects (e.g., critical assets, adversary's TTPs, vulnerabilities), subjects (e.g., threat actors and profiles), and processes (e.g., attack procedures) have become a crucial success condition of threat remediation. As related, there is a dearth of guidance in using threat intelligence to guard organizations by: uncovering intelligence from a multitude of large data sources; automating the intelligence gathering process and prioritizing threats; handling variances in data quality and relevancy of intelligence; and validating third-party intelligence and false positives. *TIME presents a potential solution path to the problems*.

Whereas traditional approaches in architecting a defense strategy have been primarily reactive (i.e., fixing vulnerabilities when breaches occur), *TIME promotes proactive and anticipatory defense*. Numerous publicized incidents repeatedly highlight the 'wait and see' approach after deploying defense measures (e.g., firewall, anti-virus, software patches). This approach is not sustainable anymore as it is unable to fend off serious attacks. A more proactive stance is necessary to preempt threats and to lower security failure rates. The traditional defense paradigm has become less effective, exposing an organization to more vulnerabilities in dealing with dangerous advanced persistent threats [5], composite/blended attacks [7], and multistage attacks [6]. The TIME methodology demonstrates the potential to contribute to the organization's cyber defense initiatives by suggesting a solution path to reduce attack surfaces; fortify defense in depth through improved intelligence capabilities; decrease cycle time of decision making;

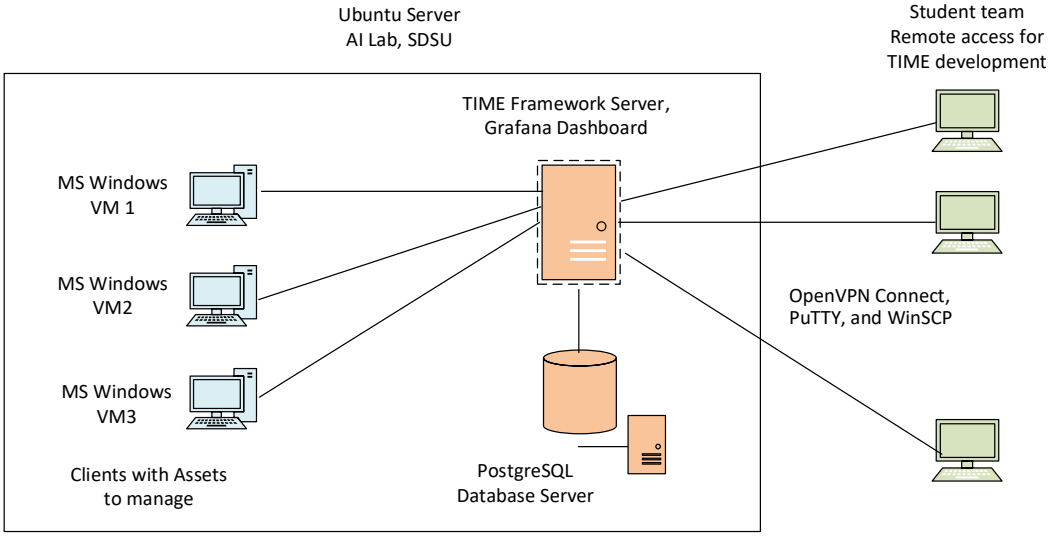

Figure 4. Architectural view of TIME development & test platform

and increase agility of cyberspace capabilities among others.

Acknowledgements

This research project was funded by the Naval Engineering Education Consortium (NEEC) program at Port Hueneme of the US Navy.

References

[1] Shin, B & Lowry, P (2020). A Review and Theoretical Explanation of the 'Cyberthreat-Intelligence (CTI) Capability' that Needs to be Fostered in Information Security Practitioners and How this Can be Accomplished. *Computers & Security*, vol 92, May, Article 101761

[2] Skopik, F. (2018). Introduction. In F. Skopik (Ed.), *Collaborative Cyber Threat Intelligence, Detecting and Responding to Advanced Cyber Attacks at the National Level* (pp. 1-18). Boca Raton, FL: CRC Press.

[3] Ahrend, J. M., Jirotka, M., and Jones, K. (2016). On the collaborative practices of cyber threat intelligence analysts to develop and utilize tacit threat and defence knowledge *Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)* (pp. 1-10).

[4] Kwon J, Johnson ME (2014). Proactive versus reactive security investments in the healthcare sector. *MIS Quarterly* 38(2):451-471.

[5] Ahmad A, Webb J, Desouza KC, Boorman J (2019). Strategically-motivated advanced persistent threat: Definition, process, tactics and a disinformation model of counterattack. *Computers & Security* 86(September):402-418.

[6] Navarro J, Deruyver A, Parrend P (2018). A systematic survey on multi-step attack detection. *Computers & Security* 76(July):214-249.

[7] Tounsi W, Rais H (2018). A survey on technical threat intelligence in the age of sophisticated cyber-attacks. *Computers & Security* 72(January):212-233.

[8] Muckin M, Fitch S (2015). A threat-driven approach to cyber security. Lockheed Martin Corporation. Retrieved from https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Threat-Driven-Approach.pdf

[9] Smith, S., Winchester, D., Bunker, D., and Jamieson, R. (2010). "Circuits of Power: A Study of Mandated Compliance to an Information Systems Security "De Jure" Standard in a Government Organization," *MIS Quarterly* (34:3), pp. 463-486.

[10] Spears, J. L., and Barki, H. (2010). "User Participation in Information Systems Security Risk Management," *MIS Quarterly* (34:3), pp. 503-522.

[11] Johnston, A. C., Warkentin, M., and Siponen, M. (2015). "An Enhanced Fear Appeal Rhetorical Framework: Leveraging Threats to the Human Asset through Sanctioning Rhetoric," *MIS Quarterly* (39:10), pp. 113-134.

[12] Puhakainen, P., and Siponen, M. (2010). "Improving Employees' Compliance through Information Systems Security Training: An Action Research Study," *MIS Quarterly* (34:4), pp. 757-778.

[13] Hardy, G. M. (2012). "Beyond Continuous Monitoring: Threat Modeling for Real-Time Response," Retrieved from https://www.sans.org/reading-room/whitepapers/analyst/membership/35185

[14] Khan, T., Alam, M., Akhunzada, A., Hur, A., Asif, M., and Khan, M. K. (2019). Towards augmented proactive cyberthreat intelligence. *Journal of Parallel and Distributed Computing, 124*, 47-59.

[15] Samtani S, Chinn R, Chen H, Jr. JFN (2017). Exploring emerging hacker assets and key hackers for proactive cyber threat intelligence. *Journal of Management Information Systems* 34(4):1023–1053.

[16] Shackleford, D. (2018). CTI in security operations: SANS 2018 cyber threat intelligence survey. Retrieved from https://www.sans.org/reading-room/whitepapers/analyst/membership/38285

[17] Wagner TD, Mahbub K, Palomar E, Abdallah AE (2019). Cyber threat intelligence sharing: Survey and research directions. *Computers & Security* 87(November): Article 101589.

[18] Brown, R. and Lee, R. M. (2019). The evolution of cyber threat intelligence (CTI): 2019 SANS CTI survey. Retrieved from https://www.sans.org/reading-room/whitepapers/threats/paper/38790

[19] Brewer, R. (2019). Could SOAR save skills-short SOCs? *Computer Fraud & Security*, *2019*(10), 8-11.

[20] Kinyua, J., & Awuah, L. (2020). AI/ML in Security Orchestration, Automation and Response: Future Research Directions. *Intelligent Automation & Soft Computing*. AI/ML in Security Orchestration, Automation and Response: Future Research Directions

[21] Oltsik, B. J. "SOAPA: Unifying SIEM and SOAR with IBM security QRadar and IBM security resilient," 2020. [Online]. Available: https://www.ibm.com/security/digital-assets/resilient/unifying-siem-and-soar-with-soapa/.

[22] Amthor, P., Fischer, D., Kühnhauser, W.E., and Stelzer, D. (2019) "Automated cyber threat sensing and responding: Integrating threat intelligence into security-policy-controlled systems," in Proc. of the 14th Int. Conf. on Availability, Reliability and Security (ARES 2019) (ARES '19), Canterbury, United Kingdom.

[23] ServiceNow (2020). "A new finish line for ai in organizations," Available: https://workflow.servicenow. com/it-transformation/a-new-finish-line-for-ai-in-organizations

[24] Palmer, T., Arcilla, A., and Amato, D. (2019). "ESG validation - threatconnect security operations and analytics platform," Available: https://threatconnect.com/wp-content/uploads/ESG-Lab-Validation-ThreatConnectPlatform-Jan-2019.pdf

[25] Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., et al. (2016) "Acing the IOC game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security, New York, NY, USA, pp. 755–766.

[26] Gupta, N., Traore, I., and Quinan, P. M. (2019). "Automated event prioritization for security operation

center using deep learning," IEEE Int. Conf. on Big Data, 2019, Los Angeles, CA, USA.
[27] Monahan, D. (2019). "How using security orchestration, automation, and response tools makes life easier… and more difficult," Available: https://www.ibm.com https://www.enterprisemanagement.com/research/asset.php/3823/How-Using-Security-Orchestration,-Automation,-and-Response-Tools-Makes-Life-Easier.andMore-Difficult
[28] AlSadhan, T. and Park, J. S. (2016). "Security automation for information security continuous monitoring: Research framework," in Proc. IEEE World Congress on Services, SERVICES, pp. 130–131.