Dynamic Network Topology Through Malware Attack Detection on Software Defined Network

Ms. R. Mahalakshmi[1]
Mr. S. Jeevanandham[2], Mr. S. V. Gokulnath[3]
UG Scholars
Department of Computer Science and Engineering,
Muthayammal Engineering College (Autonomous),
Rasipuram, Namakkal.

Mrs. G. Sumathi[4]
Assistant Professor
Department of Computer Science and Engineering,
Muthayammal Engineering College (Autonomous),
Rasipuram, Namakkal.

***Abstract*- Malware detection has long become a challenge in research. The present methods depend upon malware signature which is proved to not be effective nowadays. The recent technologies specialize in using probabilistic model like machine learning to detect the existence of malware. However, don't achieve such a decent performance. Particularly, machine learning techniques still have a difficulty of high feature engineering overhead. During this work, a propose machine learning method to detect malware supported their malicious behavior. This work has the potential to enable novel security applications that support flexible, on-demand deployment of system elements. It can give targeted forensic evidence collection and investigation of network attacks. Such unique capabilities are instrumental to network intrusion detection that's challenged by large volumes of information and complicated network topologies. The pliability of Software-Defined Networking (SDN) provides a chance to develop a malware analysis architecture integrating different systems and networks profile configuration. The proposed work is design architecture specialized in malware analysis using SDN to dynamic reconfigure the network environment supported malware actions. We use well known machine learning scheme (Support Vector Machine) as a core system for detecting malware by using only traffic features that can be extracted using an SDN controller. As result, we demonstrate that our solution can trigger more malware's events than traditional solutions that don't consider sandbox surround environment a malware analysis.**

Keywords*:- *Malware detection, Machine Learning, Support vector Machine, SDN (Software Defined Network) and Network Topology

I INTRODUCTION

Malware attackers try and infiltrate layers of protection and defensive solutions, leading to threats on a network and its assets. Anti-malware software's are widely employed in enterprises for a protracted time since they will provide some level of security on computer networks and systems to detect and mitigate malware attacks. However, many anti-malware solutions typically utilize static string-matching approaches, hashing schemes, or network communication white listing. These solutions are too simple to resolve sophisticate malware attacks, which may hide communication channels to bypass most detection schemes by purposely integrating evasive techniques. The problem has posed a heavy threat to the protection of an enterprise and it's also a grand challenge that has to be addressed. A number of the sophisticate malware attackers use either a static or dynamic method to speak with a centralized server to service a Command and Control (C2). In an exceedingly static method, everything is fixed. As an example, the malware has both a set IP address and a set name permanently (i.e., its name won't change throughout its lifespan). Thus, as long this malware has been identified as a threat, an easy rule will be applied to resolve this malware threat issue. in an exceedingly dynamic method, Domain Generation Algorithm (DGA) has been commonly accustomed communicate back to a spread of servers. The DGA could be a sequencing algorithm that's accustomed periodically generate an oversized number of domain names, which are often employed by malware to evade domain-based firewall controls. The generated domain names give malicious actors the chance to cover their C2 servers so it's hard for the enterprise to spot the DGA. The domains generated by DGAs are short-lived registered domains and that they are easier for human to spot but harder for machines to detect automatically. The given input sort of a timestamp, a deterministic output will follow as pre-defined by the DGA.

The challenge behind deterring a DGA approach is that an administrator would wish to spot the malware, the DGA, and so the seed value to separate past malicious networks and future servers within the sequence. The DGA increases the problem to manage malicious communications as an advanced threat actor has the ability to change the server or location periodically the malware communicates back (callback) to the C2 in an automatic fashion. There is a grant challenge within the detection of a DGA.

The Content-Centric Mobile Network (CCMN), discusses a cloud CCMN architecture that's aimed to serve the foreseeing multimedia traffic. They also stated that the 2 classical problems of content placement and request routing are yet to be addressed under the scenario of elasticity nature of cloud computing. The work presented in presents a generic framework to support the information-centric Internet-of-Things (IoT) services by complementing global cloud and ICN altogether.

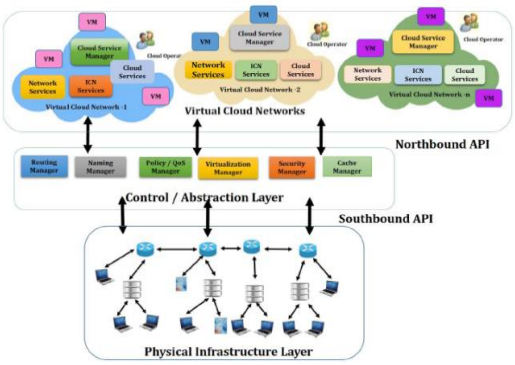

Fig 1. Content-Centric Mobile Network System

This helped them in developing a multilayer, content- and service-centric approach to IoT data management. However, the framework has incorporated the techniques from ICN in a very loose manner which is hardly clear within the paper. In reference to cloud resource management, has designed virtual network abstractions subject to QoS requirements in data center networks and address bandwidth allocation for access networks.

Physical cloud infrastructure layer consists of switches (legacy or open flow), cloud servers and terminals (in information-centric networking, terminals are either consumers (cloud customers) or producers (data store)). Consumers are sending their request for any data via the Request platform API, which is responsible for interpreting the use request in ICN enabled format. Physical network consists of L2/L3 legacy and Open Flow enabled switches (routers).

Control/Abstraction layer may be a logical entity which is controlled by an SDN controller. The controller consists of security manager, policy/QoS manager, routing manager, virtualization manager, naming manager and cache manager, and resource manager.

Cache Manager keeps track of content store of every network element (routers or switches) of the ICCN infrastructure. This manager is additionally accountable for caching and replacement policies. It decides whether information should be considered for caching or not. The default policy in ICCN is to cache everything. However, this could not be possible to cache everything during a large network like ICCN.

Security Manager provides security and integrity of knowledge in ICCN. this may be achieved by the utilization of public key digital signature for every data packets. Security manager is centralized and it can act because the public key infrastructure (PKI) for the key distribution in ICCN. Further, the safety manager can provide authentication of consumers and producers. Therefore, it's difficult for an attacker to falsely inject any data packets in ICCN.

Policy/QoS Manager enforces policies of an ICCN for managing the network (network resources like consumer, producer or switches). For instance, a consumer can access the network resources (e.g., Internet) during a specific period of your time. This kind of policy is managed by the policy manager for better utilization of network resources.

III RELATED WORK

1. A TWO-HASHING TABLES MULTIPLE STRING PATTERN MATCHING ALGORITHM

The K. Rieck [1] plays important roles in various fields including the operating system commands (Unix grep command using Comments-Walter and agrep using Wu-Manber), intrusion detection systems. Bit-parallel-based algorithms accommodate the patterns by the form of bits. Navarov [2] showed how to apply the single Shift-Or and the single Shift-And to the Multiple Shift-And, the Multiple-BNDM. The Bit parallel principle is restricted by the computer word; moreover, the inherited algorithms need to deal with the bit conversion thus taking more time.

2. AN SDN BASED FRAMEWORK FOR GUARANTEEING SECURITY AND PERFORMANCE IN INFORMATION CENTRIC CLOUD NETWORKS

To address the limitation of content-centric mobile network (CCMN), discusses a cloud CCMN architecture that is aimed to serve the foreseeing multimedia traffic. U. Ghosh [3] stated that the two classical problems of content placement and request routing are yet to be addressed under the scenario of elasticity nature of cloud computing. The work presented in C. Khancome [4] presents a generic framework to support the information-centric Internet-of-Things (IoT) services by complementing global cloud and ICN altogether. In regard to cloud resource management, has designed virtual network abstractions subject to QoS requirements in data center networks and addresses bandwidth allocation for access networks.

3. ENSURING CLOUD SERVICE GUARANTEES VIA SERVICE LEVEL AGREEMENT (SLA)-BASED RESOURCE ALLOCATION

An SLA-based resource management problem in high-performance cloud auditing where we are required to calculate the number of service resources required to ensure that QoS requirements including performance, availability, and security are met subject to a given fee. For example, F. Maggi [5] to response time of a service request meets the requirement of a predefined percentile response time under a given fee. The SLA-based resource optimization problem can be constructed by A. K. Sood [6] to minimizing the total cost of service providers while satisfying SLA guarantees.

4. LEARNING AND CLASSIFICATION OF MALWARE BEHAVIOR

The K. Xiong [7] to, detecting of the analysis environment is no general limitation of our approach: to mitigate this risk, we can easily substitute our analysis platform with a more resilient platform or even use several

different analysis platforms to generate the behavior-based report. A malware binary might try to mimic the behavior of a different malware family or even benign binaries, e.g. using methods proposed. The xiong [8] to consider analysis reports, however, differ from sequential representations such as system call traces in that multiple occurrences of identical activities are discarded.

5. MULTIPLE PRIORITY CUSTOMER SERVICE GUARANTEES IN CLUSTER COMPUTING

The probability distribution of low-priority response time does not have a closed form solution in a single queue. M. Estar [10] the priority-type resource management problem becomes very difficult to solve. We propose an approach to computing the probability distribution to evaluate percentile response time numerically. Then, we present an approach for optimal resource management that minimizes the total cost of computer resources required while preserving a given percentile of the response time for multiple customers.

III. METHODOLOGY

SDN could be a newly emerging computer networking architecture. Its main distinguishing factor is that the separation of the information plane from the control plane in routers and switches. Software Defined Network is an approach to using open protocols, like open flow, to use globally aware software control at the perimeters of the network to access network switches and routers that typically would use closed and proprietary firmware.

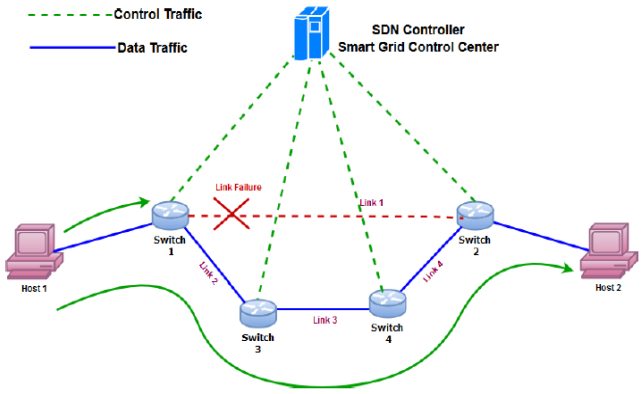

Fig 2. Proposed Work Architecture

SDN main components are controllers, the forwarding devices and therefore the communication protocols between them. The SDN technology is an approach to network management that allows dynamic, programmatically efficient network configuration so as to boost network performance and monitoring making it more like cloud computing than traditional network management. The SDN layers essentially acts a virtual software switch or router in situ of (or in conjunction with) the physical network devices. So rather than software embedded in routers and switches managing the traffic, software from outside the devices takes over the work.

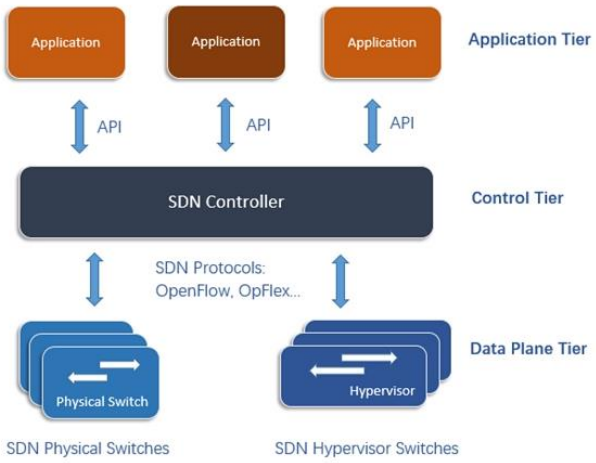

Fig 3. SDN Controller System

An SDN controller is that the applying that acts as a strategic control point during software defined network. An SDN manages flow control to the switches and routers. SDN technology detaches network control from networking hardware devices, making SDN networking directly programmable. SDN networking deploys a centralized intelligent controller, which programs devices like SDN data switch by software, communication between data devices and applications and displays the network during a virtual switch.

A. THREAT MODELS

Threat actors need some way to manage and maintain the malware in a very C2 environment while operating in an unnoticeable manner from network security systems. The successfulness of the malware doesn't require an internet site to be registered or valid and a DGA may iterate a sequence that winds up in an NXDOMAIN situation (unregistered). Blacklisting, establishing a DNS sinkhole, and implementing a firewall rule are standard techniques to forestall a malicious network activity from malware and so the signatures to implement these mitigation techniques are often provided by threat intelligence feeds.

B. THE QUALITY OF SERVICE

As a complementary technology with security, trust solves the matter of providing corresponding access control supported judging the quality of services, and it makes the quality security services more robust and reliable by ensuring that each one the communicating nodes are trusted during authentication, authorization, or key management. The security-related behavior includes four trust indicators: the authentication type, the authorization type, thyself-security competence and also the amount of malicious access. The QoS-related behavior consists of this CPU utilization rate, memory utilization rate, memory device utilization rate, average latent period and average task success ratio.

C. THEART INTELLIGENCE FEED

The VN request, the components of the cloud infrastructure (the physical servers and so the WDM

network equipment) should consume a specific amount of power. This power consumption are going to be divided into two parts: the workload-dependent power and so the workload-independent power. During this paper, we seek advice from the workload-independent power consumption as "idle power." The idle power that's consumed by the cloud infrastructure is going to be reduced by a power-efficient VN provisioning scheme. The essential ideal power-efficient VN provisioning is to indicate off the lightly loaded equipment of the cloud infrastructure by consolidating the VMs onto fewer physical servers and by routing the communication demands on fewer fiber links.

D. DOMAIN GENERATIION ALOGORITHM

A Domain Generation Algorithm may be a program that's designed to come up with domain names in a very particular fashion. Even then, taking down sites that malware employing a DGA will be a challenge as defenders must bear the method of working with ISPs to require down these malicious domains one by one. Using real-time active malicious domains derived from DGAs on the public Internet measures the accuracy of the proposed approach. Specifically, threat intelligence feeds collected from Consulting over a period of one year were obtained through daily manual querying demonstrated trends of ongoing threats.

E. MACHINE LEARING FRAMEWORK

DNS queries with the payload as the input. Then, the DNS queries will be passed to our process step, which consists of 4 important components:

(1) We first use a domain-request packet filter to get domain names and then store them in a dynamic blacklist. If the input is a known domain, we will skip

(2) Directly go to the output; otherwise, we will proceed to the next component. Then, a feature extractor is used to extract domain features

(3) Next, we apply the first-level classification to distinguish DGA domains from non-DGA domains and the second-level clustering to group similar DGA domains

(4) Finally, we use a time-series model to predict the features of a domain. After the domain name goes through the process step, we will append this domain to the dynamic blacklist

IV CONLUSION

We have presented the implementation of a collaborative detection mechanism of network attacks. Our approach is exclusive to use a synergistic monitoring, detection, and mitigation strategy to comprehend the total capabilities of SDN. During this work, we propose a machine learning method to detect malware supported their malicious behavior. The detection and mitigation in system is proven robust through experimentation. Furthermore, the overall time required for this collaborative system to detect an attack is low. Thus, this solution can potentially be deployed in a very real system where such an attack is detected and mitigated in time before legitimate users start to suffer. We are working to use this collaborative approach to other security applications, including detection and mitigation of other attacks. Our goal is to develop a scientific methodology along this line of work

V REFERENCES

[1] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, "Learning and classification of malware behavior," in International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Springer, 2008, pp. 108–125

[2] T. Chin, K. Xiong, and M. Rahouti, "SDN-based kernel modular countermeasure for intrusion detection," in Proceedings of 13rd EAI International Conference on Security and Privacy in Communication Networks. Springer, 2017

[3] U. Ghosh, P. Chatterjee, D. Tosh, S. Shetty, K. Xiong, and C. Kamhoua, "An SDN based framework for guaranteeing security and performance in information-centric cloud networks," in Proceedings of the 11th IEEE International Conference on Cloud Computing (IEEE Cloud), 2017

[4] C. Khancome, V. Boonjing, and P. Chanvarasuth, "A two-hashing table multiple string pattern matching algorithm," in Tenth International Conference on Information Technology: New Generations (ITNG). IEEE, 2013, pp. 696–701

[5] S. Schiavoni, F. Maggi, L. Cavallaro, and S. Zanero, "Phoenix: DGA-based botnet tracking and intelligence," in International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Springer, 2014, pp. 192–211

[6] A. K. Sood and S. Zeadally, "A taxonomy of domain-generation algorithms," IEEE Security & Privacy, vol. 14, no. 4, pp. 46–53, 2016

[7] K. Xiong, "Multiple priority customer service guarantees in cluster computing," in Proceedings of the IEEE International Symposium on Parallel & Distributed Processing (IPDPS). IEEE, 2009, pp. 1–12

[8] Xiong, "Resource optimization and security for cloud services." Wiley- ISTE, 2014

[9] K. Xiong, "Resource optimization and security for distributed computing, https://repository.lib.ncsu.edu/handle/1840.16/3581," 2008

[10] B. Mark et al., "GENI: A federated testbed for innovative network experiments," Computer Networks, 2014

[11] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise," in Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, ser. KDD. AAAI Press, 1996,pp.226–231.[Online].Available: http://dl.acm.org/citation.cfm?id=3001460.3001507

[12] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN revisited: Why and how you should (still) use DBSCAN," ACM Trans. Database Syst., vol. 42, no. 3, pp. 19:1–19:21, Jul. 2017. [Online]. Available: http://doi.acm.org/10.1145/3068335

[13] T. Chin, K. Xiong, C. Hu, and Y. Li, "A machine learning framework for studying domain generation algorithm (DGA)-based malware," in SecureComm, 2018

[14] K. Xiong and X. Chen, "Ensuring cloud service guarantees via service level agreement (SLA)-based resource allocation," in Proceedings of the IEEE 35th International Conference on Distributed Computing Systems Workshops, ICDCS Workshops. IEEE, 2015, pp. 35–41

[15] T. Chin and K. Xiong, "Dynamic generation containment systems (DGCS): A moving target defense approach," in Proceedings of the 3rd International Workshop on Emerging Ideas and Trends in Engineering of Cyber- Physical Systems (EITEC), vol. 00, April 2016, pp. 11–16. [Online]. Available: doi.ieeecomputersociety.org/10.1109/EITEC.2016.7503690