Firewall for Next Generation with Dynamic Networks

Savita Panghal[1], VK Srivastva[2]

1.Baba Mastnath University, AsthalBohar, Rohtak
2. Professor, Head of Dept. of Computer Science&Application, BMU, Asthalbohar, Rohtak

Abstract| As we know a firewall has delicate security building blocks for networks to control and monitor incoming and outgoing traffic that is completely based on induction security rules which introduce as firewall rules. Firewall are regularly developed to enlarge network security from being a simple filtering device firewall has been developed to operating in a combination of intrusion detection methods and a prevention systems. In this paper we introduce Firewall policies and define their application in highly dynamic networks and we describe the needs for the next generation firewall policies and how maximum advantage of generative policy models will be taken.

Index terms:-firewall, intrusion detecton, prevention systems ,generative policy , coalitions networks .

I. INTRODUCTION

Firewall system have developed broadly over the years with various types of firewall that execute variety of purposes. Generally, firewall system are designed to protect networks with well define limits, and border like in enterprise networks.therefore to present firewall capabilities solutions need a high level of estimated power to inspect network data and to rapidly reduce threats. Firewall also have extra additional security measures, such as access control to network resources as part of a broader security architecture mainly on the current firewalls, the rules are statically and manually are well defined. Therefore, the rules increase according to network and for that network managing is a major issue for example leading to believable vulnerabilities and threats. Although firewall systems are effective hurdle for threats in networks. they are not sufficient for dynamic coalition networks they may involve autonomous mobile devices and Internet of things (IOT) systems. In coalition network firewalls are expected to operate in environments that are represented by unstable, ineffectively, difficulty and vagueness. The protection systems may also be highly mobile, such as a system that applies moving target defence technique, which result in various components of the systems to change in response to dissimilarity in its environment. Therefore new generation firewalls are required to adopt changes without interference of human beings. However, this is important that adversary could not use the autonomous behaviour of the firewall to create new attack vectors based on their behaviour and response under threats out of which some issues and attacks are discussed in [1] applies to firewall avoidance. Furthermore we considered the challenges faced during coalition process for getting effective distribution of information from diverse and distributed physical sensors and decision makers across coalition partners which varied on various level of trust and uncertainty[2],[3],[4].

II. OUTLINE OF FIREWALL CONCEPTS

In this we try to focus on key concepts of a firewall and review a well known open source firewall that includes various packages that can be used beside. The primary role of firewall is to inspect and filter data between network section. It allows filtering packets based on their attribute and perform actions on the packets that matches some specific policy and rules of firewalls. Most of the common application which protect data between an internal and the internet is based rules specify by the match condition for data and the action that will be taken when conditions are matched satisfied. The data can be matched on including IP addresses and port fields, number of characteristics, protocol, and ICMP type. In this a ruleset is followed which means a series of firewall rules, that rules are

executed in numerical sequence, according to the rule number, from lowest to highest. If the data matches the characteristics specified by a rule, the action of the rule is perform; if not the next rule is perform or processed and so on.

Firewall rules are defined as ECA(Event-Condition-Actions). In this rules are activate automatic, validation of stated conditions and actions, if the condition hold. However within a system, their interaction can be difficult to examine, therefore the process of execution one rule may cause an event which activate another rule or rule set, these rules may in turn to activate further rules and there is indeed the believable for an infinite cascade of rule firings to arise.

We define basically two type of firewall stateless and stateful. In a stateless firewall we considers every packet in discontinuity. Packets can be receive or reject according to only basic Access Control List (ACL) criteria such as the source and destination fields in the IP or TCP/UDP headers. It does not store connection information and has no requirement to look up every packets relation to previous flows, both of which consume small amounts of memory and CPU. On other hand a stateful firewall keeps a state table of previously seen flows and packets can be receive or reject according to their relation with previous packets. As a general rule, stateful firewalls are generally preferred where application data is popular.

A. PfSense

pfSense [6] is a freeopen source operating system for network firewall distribution which includes additional features that are not available in some commercial solutions. pfSense is a stateful firewall with packet inspection, meaning the state table maintains information of the open network connections. It doesn't require a devoted device for its extend, however it is a software package. therefore, it is suitable for virtualized environments or in dynamic networks. While providing the basic role of a firewall, i.e., filtering data by sender and receiver IP fields (e.g., addresses and ports) and protocol, pfSense comes up with additional features. For example, it uses p0f utility for advanced passive data and operating system fingerprinting, e.g., allows to filter the OS behind the commence connections. Packet normalization is authorize by default with pfSense. It offers various tool for manipulate and development of the state table. Maintaining the state table is crucial for high availability (HA).Through replicated to all backup configured firewalls, the network connections are not disrupted during a failure. pfSense can be install and configured with several other open source security software and packages such as Snort and Suricata [7]. These packages can be used alongside with pfSense to improve security management in the network

III PREVENTION SYSTEM AND INTRUSION DETECTION

An Intrusion Detection System (IDS) is a security and threat prevention technology that examines network traffic flow to detect and prevent vulnerability achievements. Or IDS is a device that monitors a network or systems for policy breach or malicious activities. Through these policy breach are basically reported to an administrator or collected centre to an event management system for further study. When an IDS focuses on monitoring and analyzing the network traffic, it is called a network IDS (NIDS). An NIDS (network Intrusion Detection System) is a non-resistant system that scans data and reports back on threats. On the other side, an Intrusion Prevention System (IPS) is a network security and threat prevention technology that inspect network data flows to detect and prevent vulnerability make use of. Vulnerability utilize usually come in the form of destructive inputs to a target application or service that attackers use to interrupt and achieve control over an application or machines. An IPS often sits directly behind the firewall and provides a supporting layer of investigation and security. An IPS is expand in the direct communication path between source and destination (inline mode), while actively examine and taking self operating actions on all data flows that enterinto the network segments. Therefore we need to discuss about the details Network IDS/IPS called Bro and Suricata.

i) BRO NIDS

This is a platform with build upfeatures that supported in typical IDS tools[8]. It supports both signature and anomaly-based IDS and its extensible architecture provides the ability to write custom policy an reviewer. Bro is based on three-tier layered architecture [9]: Tap, Platform, and Applications. The tap network link sends up a copy of the traffic to the packet processing module which filters down the high-volume stream via standard libpcap (promiscuous capture library) [10] packet capture library.defined as a common standard format for files in which captured frames are stored, also known as the tcpdump format, currently a de facto standard used widely in public network traffic achives. The platform layer is component of two main modules: the Event Engine converts the captured data00 to a series of high-level events reflecting underlying networks activities in policy-neutral terms while the Policy Script Interpreter executes a set of event handlers written in Bro's custom scripting language. The script can consolidate the policies and context from the past and takes actions (e.g., generate alerts, record to disk, executes response programs, etc.). Bro can install the standalone or cluster mode. The scripting language being an event-driven, can be used to express arbitrary analysis tasks and customize policies or define actions to be taken given an event. Bro's scripting language facilitates a much broader range of very different approaches to finding mischievous activity, including semantic misuse detection, anomaly detection, and behavioral study. Bro can act as a dynamic and intelligent

firewall when used in combination with blocking gateway (e.g., firewall). For example, it can block the access from offending IP addresses, known hostile activity, terminate connections and/or sends alarms, locates site policy violations. Furthermore, the dynamic application detection feature allows port selection rather than specifying which protocol analyzer to use for a given port. However , using the NetControl framework, Bro can connect with various network devices and equipments such as switches, firewalls, and routers through their specified API. The NetControl framework provides a flexible, unified interface for active response and hides the complexity of heterogeneous network equipment behind a simple task-oriented API, which is easily usable via Bro scripts [9]. Hence variour type application and various use cases, also called Bro Frameworks, to be supported with Bro are very broad, ranging from intrusion detection, vulnerabilities management, file analysis, traffic analysis and measurement, compliance monitoring, etc. Bro comes with many pre-written scripts and analyzers for many protocols that are highly customizable to support traffic analysis for specific environment and needs. Virtually with Bro scripting language, various type of policies can be defined and Bro can interface with other network equipments and applications for real-time exchange of information. [9].

ii) SNORT

Snort [11] is based on rules in NIDS and IPS capable of performing traffic analysis (e.g., protocol analysis, content searching and matching), detecting various attacks and probes, and packet logging in real-time. Moreover Snort combines the benefits of protocol, signature, and anomaly-based inspection methods to perform flexible and efficient protection against security threats. Snort's rule is collection of the rule header and options. Through which when rule match according to standard then only Snort come to know what to do on a particular request. Through this Data Acquisition (DAQ) concept has been introduced in Snort 2.9 to replace the direct calls to libpcap functions. Snort can operate in passive (tap) and inline modes. In passive mode, Snort acts as an IDS. therefore with the inline mode, Snort acts as an IPS allowing drop rules to trigger. In inline mode, Snort creates a transparent bridge between two network segments, and is responsible for passing traffic between the two segments. Snort inspects the traffic based on the specified rules, then either drop the suspicious traffic or pass it out to the other interface without any tempering. [12].

iii) SURICATA

Suricata [13] is another open source able be for real-time network IDS, IPS and network security monitoring (NSM). It examine the network traffic using a powerful

and extensive rules and signature language, and has a scripting support for detection of complex threats, policy violations and malicious behavior. Suricata can also detect many anomalies in the traffic it inspects. however it has similar working as Snort, its modularity and automatic protocols recognition are the key advantages. Suricata is based on rule/signature which consist of action, header and rule-options.

IV ISSUES OF IDS/IPS AND THEIR CHALLENGES

However we have descried IDS and IPS present higher advantages and benefits, after that they have limitations. In fact, they are mostly static and required manual configuration. For e.g when large volume of alerts and notifications, a network administrator should manually sort out the issues using visualization or reporting tools to identify the alerts that pose effectual risks. Now a days network are fairly dynamic and keeping machine up to date with humans in the loop can be challenging and prone to errors. Therefore IDS/IPS usually are unaware to the context and hinder the full potential of network security automation . therefore , without context aware capability, fast threats evaluation and extinguishing as well as accomplished and credible automation become challenging. Even the rule are not automatically updated. Further depend on external exchequer that updates and write rules at a given frequency to encounter new and develop threats might not be systematic to protect against unseen attacks. e.g Emerging threats [14] is changing it self daily, Talos [15] is changing weekly or multiply times a week.

With these advancement in machine learning, IDS and IPS will be extended with efficient predictive methods for writing rules for new traffic based on insights from history. Through the predictive model can help to detect unseen data or abnormal behaviors. This falls in the big umbrella of generative policy model [5], therefore each IDSor IPS device has capability to generate in real time new policies and rules based on the context. ECA policies are then created in real time to adopt the data and the dynamicity of the networks. It is impotant for automated and reliable policy based management systems for e.g in distributed and coalitions networks.

Firewalls are designed to block or accept different types of traffic based on the 5-tuple(that is destination IP addressesand source, destination and source ports, and protocol.) Instead of detecting or blocking attacks. firewalls aim is not to inspect intrusion inside a network. Therefore efficient and dynamic defense system such as IDS/IPS are deployed to detect attacks and improve security management capability. IDS/IPS able to find and catch the attacks that the firewall didn’t see or allowed traffic and detect mis-configured firewall.

V. INFERENCE SYSTEM OF FIREWALL MANAGEMENT

According to various research activities into firewall design for context aware information masking led to the initial realization of an inference management firewall (IMF) capability [3],[16],[4].The focus of this work is based around the notion of inferences that can be learned from shared information. However each possible inference is classified and assigned to either a whitelist (which represents inferences that are permitted by policy) or a blacklist (of those inferences that are not permissible). These lists are then used as factors in determining appropriate access control policies (and mechanisms) over a shareable data set.

In practical exploration, an architecture was defined that classifies the IMF into three general components: the first component comprises a network policy enforcement and a decision-making system that operates at the core of the information network; the second component is the end-point policy evaluation and enforcement system that enforces policy on low-capacity mobile devices operating at the edge of the information network (typically at source and sink points); the final component and third is the communication and associated systems that integrate the different inference management tiers into a logically single firewall. A demonstrable prototype was developed as part of the study, consisting of the three components described above. This prototype applies access control policy over a publishsubscribe messaging pattern and is based on the ITA Information Fabric [17]. The prototype supports practical exploration of the inference management principle on data traversing a network of participating nodes. At the edges of network are mobile devices that are capable of sensing their environment and publishing sensed data to the network. Inference management is sucked up at the edge by utilizing IPShield [2] running on Android devices. Although this prior work builds a Ground basis for further practical investigation, it also reinforced with realizing a number of initial challenges. Firstly, the data publisher is required to configure privacy policy in a trends that enables administration as a shared responsibility between network participants. This requires the definition and application of a suitable policy scheme. Secondly, bidirectional exchange of control data must be set up between participants in the network core and those operating at the network edge. For example the edge nodes are required to provide the network core with release policies in accordance to the preference to owner information must be private. Similarly, participants operating in the network core must provide policy applicable to information consumers at the network edge. The nature of this scheme for policy expression and management, which was beyond the scope of the original research, must operate in a distributed fashion and be robust to changes in operational dynamics, such as when a network node is removed, or when power availability or network capacity is unexpectedlyabolish. A further area of investigation

considers the challenges of achieving information security with the goal of mitigating appropriate vulnerabilities

VI. CONSULTATION OVER THE DIFFERENT ANOMALIES IN FIREWALL AND DISTRIBUTED FIREWALLS SYSTEMS

An important phase of firewalls that should be taken into opinion for designing the next-generation of firewalls systems are the anomalies [18] that are created between rules. These anomalies can be created because of an incorrect ordering or representation of firewall rules or redundancies and conflicts between rules of different firewalls. Some of the most common anomalies are when a packet matches different firewall rules, or when we are in a distributed firewalls environment and for the same packet different firewalls that are on the same path performs different actions. Some of the anomalies of centralized firewall system are due to the bad ordering of the firewall rules.

here, we present some of the firewall systems anomalies, where we denote by ri, rj the firewall rules, with < the relation of precedence between them, e.g., ri<rj means that rj has a higher ordering respect to rule ri, so if the rule ordering for ri is i and for rj is j, then j is smaller i.

• Shadowing anomaly: rule ri is shadowed by rule rj , when rj matches all the packets matched by ri, and because ri<rj , rule ri is never applied to these packets, instead rj applies. The shadowing problem is a crucial anomaly because the shadowed rules never applies, thus a packet that should be blocked is permitted and vice versa.

• Correlation anomaly: rule ri is correlated with a rule rj , if they perform different actions and ri matches some packets where rj can be applied, and rj matches some packets where ri can be applied. These rules can be seen as partially redundant for their spectrum of action but have different actions.

• Generalization anomaly: rule ri is a generalization of rule rj , if they have different actions and if rj is able to match all the packets matched by ri. In this case, we are dealing with a redundancy, where ri is included in rj , but these rules apply different actions.

• Redundancy anomaly: two rules are redundant if they match the same packets and they perform the same actions. In this case, one of them can be removed.

• Irrelevance anomaly: a rule ri that does not match any traffic is irrelevant. This rule can be removed from the firewall rules. Nowadays, we often find systems that use different firewalls. In this case, the anomalies created are not only the one of the

firewalls themselves, but also what can be created by the use of different ones. It is important for our future work to understand and analyze the created anomalies, especially dealing with coalitions, where every coalition can have their own sets of firewalls or rules, with their appropriate ordering, thus it is common that conflicts and anomalies arise between the various firewalls' rules. Below, we present some of the distributed firewalls systems anomalies, where the firewalls' rules are denote by ri, rj , in this case we use $\prec$ to denote that a firewall rule is more close to the destination of the packet than another one, thus ri$\prec$rj means that rule rj is part of a firewall that is more close to the destination then the firewall of rule ri.

• Inter-firewall shadowing: when ri blocks packets that are permitted by rj , where ri$\prec$rj . This anomaly is important as traffic that should arrive to the destination is blocked.

• Spurious traffic: when ri permits packets that are denied by rj , where ri$\prec$rj . This anomaly is critical as non wanted traffic is getting close to the destination.

• Redundant anomaly: when rj denies packets that are denied by ri, where ri$\prec$rj . This anomaly effects the efficiency of the firewalls system, as traffic that was already blocked by the firewalls that are more far from the destination is blocked again by firewalls more closer to the destination.

• Correlation anomaly: when ri and rj have different actions and part of the packets matched by ri, are matched by rj, and vice-versa. There are different techniques for solving the above anomalies. In [18], the authors take these anomalies by constructing policies trees. The latter represent the firewall rules, where every node represent a network fields and every branch a possible value associated to that field. The graphical representation present by the policies trees helps pinpoint the various anomalies. In [19], the authors introduce a dynamic ruleordering technique, that uses Internet data characteristics, for firewall filtering. Other techniques are introduced for dealing firewall anomalies and their rule ordering. An interesting technique used in [20] is argumentation, where an innovative firewall configuration management is introduced that performs the automatic firewall rules ordering, by avoiding the creation of anomalies.

VII. CHALLENGES FOR NEXT GENERATION FIREWALLS

Recent IoT-based botnets [21] have shown that many types of the device can be easily compromised and hire into a botnet. In dynamic environments where devices can move inout from networks, we cannot certainly exclude the possibility that imperiled devices could move into a system externally protected by a firewall. Such devices can then start

executing actions, such as sending requests to a target destination as part of distributed denial of serviceattack. Preventing such malicious use of devices requires that firewalls be able to filter not only the data incoming toward the protected system but also the data outgoing from the protected system in order to make sure that the data is directed towards effectual destination and according to the specific missions being carried out by the protected system. An initial approach to build filtering capabilities to prevent IoT devices from being used as bots by a botnet has been recently proposed by Habibi et al. [22]. Such a firewall takes advantage of the fact that communication patterns for several categories of IoT devices are quite expected as these devices have often very specialized functions and usually only communicate with specific applications located at a predefined set of IP addresses. Therefore Extensive testing has shown that such a simple approach is effective.

Although research is needed for developing techniques for profiling devices distinguish by more complex communication patterns and correlate such patterns with the input received by the devices and the current context of the devices. An approach along such lines has been developed on the condition of data protection from insider threat [23]. The specific approach aims at creating profiles of SQL application programs. Such profiles record the specific SQL queries executed by the application programs based on the input parameters. At run-time, queries issued by each application are matched against the query profile of the application and if there is mismatch the query is flagged as anomalous. Such an approach is quite complex as it uses concolic testing techniques(which is known as a hybrid software verification technique that performs symbolic execution, a classical technique that treats program variables as symbolic variables,along a concrete execution path) and also the use of a log system to capture application input and SQL queries issued by applications. However maybe a simpler approach along those lines could be developed for profiling communications of IoT devices. In the current firewall, the matching criteria is based on execution of regular expression against IP packet headers.furthermore, the network IDS and IPS have capability to extract insights from the monitored data. With next-generation firewall, the advances in machine learning can be used to examine packets and build predictive models to foresee abnormal behaviors of unseen data. Such technology can be designed as plug-in to IDS/IPS that can be used to enrich and update firewall rules dynamically. This would allow it to learn actual context in order to refine the policies for potential unseen attacks. With the increasing adoption of network function virtualization (NFV) and software-defined network (SDN), virtual firewalls or network security devices can be easily install and setup throughout distributed network and pointof-entry.

VIII. CONCLUSION

we try addresses various existing firewall technologies and their policies. It also highlights the need for next generation firewall giving an indication of the challenges in the existing firewalls. Some of the key areas includes using machine learning, virtualization, autonomic systems, and knowledge base that would help to design these next-generation firewalls for highly dynamic networks such as coalitions environments.

IX REFERENCES

[1] A. Atlasis, "Attacking IPv6 implementation using fragmentation," BlackHat Europe, pp. 14–16, 2012. [2] S. Chakraborty, K. R. Raghavan, M. P. Johnson, and M. B. Srivastava, "A framework for context-aware privacy of sensor data on mobile systems," in Proc. of the 14th Workshop on Mobile Computing Systems and Applications. ACM, 2013, p. 11.

[3] S. Pipes, S. Chakraborty, and F. Cerutti. (2014) Inference management in the experimentation framework.[Online].Available: http://nisita.org/science-library/paper/doc-2775

[4] S. Pipes, B. Hardill, C. Gibson, M. Srivastava, and C. Bisdikian, "Exploitation of distributed, uncertain and obfuscated information."

[5] pfSense. https://www.pfsense.org/.

[6] https://doc.pfsense.org/index.php/package list.

[7] V. Paxson, "Bro: a System for Detecting Network Intruders in RealTime," Computer Networks, vol. 31, no. 23-24, pp. 2435–2463, 1999

[8] B. NIDS, "The bro network security monitor, https://www.bro.org/index.html."

[9] Libpcap. http://www.tcpdump.org/.

[10] Snort. https://www.snort.org/.

[11] SNORT Users Manual.

[12] Suricata. Suricata: Open source IDS/IPS/NSM engine, https://suricataids.org/.

[13]EmergingThreats.https://www.emergingthreats.net/.

[14] Snort Talos. https://www.snort.org/talos.

[15] S. Pipes and S. Chakraborty, "Multitiered inference management architecture for participatory sensing," in Pervasive Computing and Communications Workshop (PERCOM), 2014 IEEE International Conference on, 2014, pp. 74–79.

[16] J. Wright, C. Gibson, F. Bergamaschi, K. Marcus, R. Pressley, G. Verma, and G. Whipps, "A dynamic infrastructure for interconnecting disparate ISR/ISTAR assets (the ITA sensor fabric)," in Information Fusion, 2009. FUSION'09. 12th International Conference on, 2009, pp. 1393–1400.

[17] E. Al-Shaer, H. H. Hamed, R. Boutaba, and M. Hasan, "Conflict classification and analysis of distributed firewall policies," IEEE Journal on Selected Areas in Communications, vol. 23, no. 10, pp. 2069–2084, 2005.

[18] H. H. Hamed, A. El-Atawy, and E. Al-Shaer, "Adaptive statistical optimization techniques for firewall packet filtering," in INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 23-29 April 2006, Barcelona, Catalunya, Spain, 2006.

[19] A. K. Bandara, A. C. Kakas, E. C. Lupu, and A. Russo, "Using argumentation logic for firewall configuration management," in Integrated Network Management, IM 2009. 11th IFIP/IEEE International Symposium on Integrated Network Management, Hofstra University, Long Island, NY, USA, June 1-5, 2009, 2009, pp. 180–187.

[20] E. Bertino and N. Islam, “Botnets and internet of things security,” IEEE Computer., vol. 50, no. 2, pp. 76–79, 2017.

[21] J. Habibi, D. Midi, A. Mudgerikar, and E. Bertino, “Mitigating the internet of insecure things,” IEEE Internet of Things Journal.

[22] L. Bossi, E. Bertino, and S.-R. Hussain, “A system for profiling and monitoring database access patterns by application programs for anomaly detection,” IEEE Trans. Software Eng., vol. 43, no. 5, pp. 415– 431, 2017.