ISSN 2395-1621

A Survey Paper on Intrusion Detection System with Deceptive Virtual Hosts for Industrial Control Networks

[#1]Ms. Smita Boharpi, [#2]Prof.Sonal Fatangare

[1]smita.boharpi@gmail.com
[2]sonal_fatangare@rediffmail.com

[#12]Department of Computer Engineering, RMD Sinhgad School of Engineering, Pune, India.

ABSTRACT

Industrial control systems are a distributed network most commonly used for operating the networks which are tightly coupled with physical process. In this way challenging issue is handling the security issues on a large distributed network. To handle this system various tools were evolved, in which effective tool is honeypot which is used for monitoring and focusing the activities of intruders. Honey pot tracks the unauthorized users accessing the information on a control system. These honey pots are self configured that examine control system network traffic and to observed environment. Ettercap is used for host identification. In this scenario, an anomaly detection system monitored the network activity of the honey pots. The most significant drawbacks in the detection systems are traffic overload, unknown attacks, false positives and false negatives. The intrusion detection using honeypot for reducing drawbacks of the existing system. So component of honeypot cooperates with IDS which increase flexibility, configurability and security of intrusion detection system.

ARTICLE INFO

Article History

Received:28th December 2015

Received in revised form :

30th December 2015

Accepted:1st December , 2015

Published online :

2nd December 2015

Keywords— Intrusion Detection, Industrial Control, Honeypot, Network Security

I. INTRODUCTION

The security of network is a big issue for security administrators because network is growing day by day. Security on the Internet and on Local Area Networks is now at the fore front of computer network related issues. Network security consists of the provisions and policies adopted by a network administrator to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources. Network security involves the authorization of access to data in a network, which is controlled by the network administrator. Each and every client who is working on the internet wants security of information but sometimes he or she do not know that someone else may be a intruder is collecting the information. Information is an asset that must be protected. Network security is the process by which digital information assets are protected, the goals of security are to protect confidentiality, maintain integrity, and assure availability. To secure the information and the entire network system, one specific methodology is required which can be capable of providing the complete security solutions [3]. So honey pot systems are decoy server or systems setup to gather information regarding an attacker or intruder into system. It is important to remember that honey pots do not replace other traditional internet security system; they are additional level or system. In that system Ettercap XML output is used for autonomous creation and update of honeyed configuration. Automatically created virtual host were deployed in concert with an anomaly behavior (AB) system in an attack scenario [1]. Deceptive system is called honey pots that emulate critical network entities have been deployed with monitoring solutions to improve detection accuracy and precision rates [1]. The honeypot can be classified into two categories: high interaction and low interaction honey pot. High interaction honey pot system typically hardware replicas of existing operational components that include the appropriate software. Low interaction virtual honey pots are used to gather information. So high or low interaction honeypot can only detect attacks which is directed at them [9].The honey pots accurate reconstruction of a host network presence. It has automation key capability. It is usually cheaper to build better tools than manually manage the configurations of individual devices in a large system. They are self-configurable dynamic virtual hosts by adapting to an operational network environment. A honeypot is a trap set to detect, deflect, or in some manner counteract attempts at unauthorized use of information systems. It consists of a computer, data, or a network site

that appears to be part of a network, but is actually isolated and monitored, and which seems to contain information or a resource of value to attackers. Honeypot can be classified based on their deployment and based on their level of involvement. Based on deployment, honeypot may be classified as, Production Honeypot and Research Honeypot.
High-interaction honey-pots imitate the activities of the production systems that host a variety of services and, therefore, an attacker may be allowed a lot of services to waste his time. By employing virtual machines, multiple honey-pots can be hosted on a single physical machine. Therefore, even if the honey-pot is compromised, it can be restored more quickly. In general, high-interaction honey-pots provide more security by being difficult to detect, but they are highly expensive to maintain. Example: Honey net
Low-interaction honey-pots simulate only the services frequently requested by attackers. Since they consume relatively few resources, multiple virtual machines can easily be hosted on one physical system, the virtual systems have a short response time, and less code is required, reducing the complexity of the virtual system's security. Example: Honey-d

II. EXISTING SYSTEM

Dynamic virtual honey pots are used to gain more information about the attacker using Honeyd and intrusion detection system. DHP solutions that gather network information, process that information into a configuration, and deployed appropriately. This paper proposes active, passive scanning method to gather network information.

In 2005 SCADA Honey net project by Matthew Franz and Venkat Pothamsetty [4], the design utilizes honeyd for simulating a set of services for a PLC. The major contributions of this project are service scripts, which include functionality for file transfer protocol (FTP), Modbus, Telnet, and a web server. However, the SCADA Honey net does not consider automatic provisioning of the virtual hosts and is manually configured project.

In 2006 Digital Bond, Inc. by Dale Peterson [5], it utilizes two virtual machines instead of honeyd. One virtual machine includes network monitoring tools such as Snort with Digital Bond Quick draw IDS signatures to detect activity. The other virtual machine simulates a PLC with several exposed services. There is no dynamic provisioning of hosts or services, although it is possible to replace the virtual machine PLC with an actual hardware component.

The Gaia Maselli, Luca Deri, Stefano Suin [11], in that anomaly Detection System [11] for detection network anomalies on the basis of network traffic parameters. Every network has some global variable and relation of this variable is fixed, these variables are used to detect anomaly attack in network system. Anomaly Detection System use network traffic parameter to draw specific network traffic curve for each network that does not change over the time. This system uses Ntop [10] tool for monitoring network traffic. The Ntop is capturing and analyzing the network packet and store information in database. The Ntop measure all network parameter to detect network anomalies and Store information into the database. This system gives advantage to detect both signature attacks and anomaly attacks in network, but main disadvantage of this system is that it does not describe what an attack is and gives high false positive rate.

III. PRAPOSED SYSTEM

The proposed new system as dynamic honey pot with deceptive virtual hosts and intrusion detection system to provide security against cyber devices. The collaborative use of dynamic virtual honey pot in a control system. Utilizing Ettercap XML Output is developed for autonomous creation and update of a Honeyd configuration. Ettercap is an extensible network manipulation and reconnaissance tool.

IV. SYSTEM ARCHITECTURE

These act in a continuous cycle of processing and updating information represented by the dotted line box. It contains relationship of three key functional areas: 1) network entity identification (NEI); 2) DVH configuration; and 3) virtual host instantiation (VHI).

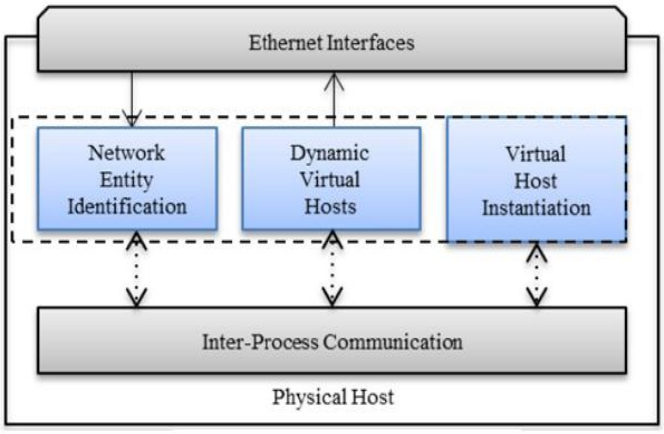

Fig. System Architecture

4.1.1 Network Entity Identification

The network entity identification (NEI) component monitors network traffic from which it extracts the source, destination, and port activity. Information from the NEI is delivered to an implementation of the logic tasked with creating a DHP configuration. To identifying network entities, NEI needs to provide the information necessary to create a representative virtual network presence. The critically required capabilities examined were OS identification, port or service identification per host, and the capture of media access control (MAC) addresses with a resolution to the appropriate vendor [1].
Ettercap is chosen for its support of XML output. Ettercap is an extensible network manipulation and reconnaissance tool. It is an established and popular tool in the hacking Community. Its used as a source of information for DHP creation. It was run as a daemon process with unified sniffing. In this mode, it maintains internal network host records and updates them as new information is found. A binary log file is continuously updated as well. An Ettercap companion executable Etterlog is then run on the log files with a -x option to produce an XML file. This data file is the source for communication of the network entity information to the DVH configuration process. The Ettercap tool was selected for identifying network entities. It provides information on host Internet protocol (IP) addresses, MAC

values, and port usage. Ettercap formatted XML output that can easily be integrated into other systems. Finally Ettercap is capable of performing more advanced operations that could be useful for future functional enhancements.

4.1.2 Dynamic Virtual Hosts

These hosts emulate the network signature of actual systems on a physical network. Honeyd is a popular open source solution for virtual honey pots that provides a flexible and feature rich configuration capability. As autonomous configuration is a desired aspect for minimization of expensive manual configuration, Honeyd configuration flexibility is an advantage. The goal is the automatic configuration and dynamic update of a variable length list of virtual hosts based on information gathered from actual hosts using Ettercap. It has four functional areas in DVH: OS selection, OS name mapping, MAC creation and Service (port) emulation.

4.1.2.1 OS Selection

For any given host on a network, Ettercap may not be able to identify the operating system. If this occurs, for an emulation target, then an OS must be chosen. It is desirable to provide an exact match in network behavior. This does not necessarily require an exact match with the OS name in the database If a candidate OS is not identified by examining ports, then the MAC address is examined. Find closest compares the vendor identification section of the candidate MAC address of with the MAC addresses for each host in o. If a match is found that has an identified OS, then this value is placed on a candidate list. After examining the largest matching value, if one exists, from the candidate list is chosen as the OS. The assumption is that any hosts on the network that have the same NIC vendor may be performing similar functions and thereby have a similar OS. Several control system vendors have an organizationally unique identifier for their network devices.

4.1.2 .2 OS Name Mapping

The Honeyd configuration value for an OS makes use of the Nmap version 1 database defined named values. Similarly, Ettercap utilizes its own defined name values that do not directly match Nmap.Each OS name combination is written to a file for reference during creation of the configuration. In the Create MAC function, the last three octets are randomly generated and appended to the end of the captured candidate vendor portion. This new MAC is then compared with all other MACs noted in the Ettercap host list.

4.1.2 .3 MAC Creation

Honeyd provides two options for specifying the MAC address, either by vendor name or the six-octet string. Because Honeyd has hard coded vendor strings, the six-octet representation was chosen for use in the algorithm. Ettercap captures this MAC octet address for all hosts. The MAC protocol specifies that the first three octets are organizationally unique and should not overlap with any other vendor. Thus, in order to create a new MAC address that appears to come from a specific vendor, these first three octets were used. The vendor typically assigns the remaining three octets.

4.1.2 .4 Network Service Emulation

The host entries in contain network ports that are active during the capture session. Along with the port number, a port service name is available. This service name is a human readable text value that is defined in an Ettercap configuration file called etter.services. Utilizing the service names contained in this file, a new configuration file called serv.conf was created. This file maps the service name to a service emulation script path.

4.1.3 Virtual Host Instantiation and Update

The candidate emulation hosts are provided at startup as a list of IP addresses. It is assumed that if a host in the list disappears from passive sensing, then the user still desires to have an emulated version of it. The overhead to maintain the missing host's records is minimal. The actual system has to have appeared in the passive analysis during the monitoring period to create an initial virtual host configuration. An initial configuration file is created by Create Host Conf .Changes to the configuration of the virtual hosts running under honeyd is performed while the system is running. In network host activity are noted and stored on a list for possible action. These actions include adding network services, updating OS configuration, and changing MAC addresses.

V. ALGORITHM

Using Ettercap XML output, a novel four-step algorithm is developed for autonomous creation and update of a Honeyd configuration. An Ettercap companion executable Etterlog is then run on the log file with a -x option to produce an XML file. This data file is the source for communication of the network entity information to the DVH configuration process. Create and update virtual hosts with the following: Network Entity Identification.

Write entities to XML
1. Read data from input files and Ettercap
2. For each IP create a Dynamic Virtual Host
3. Find closest representative OS
4. Map OS values to honeyed names
5. Create MAC address for new hosts
6. Create Features for device specific behaviors
7. Create Config for virtual hosts

VI. CONCLUSION

Honey pots are gaining importance as a useful security tool alongside firewalls, IDS and antivirus software. Honey pots are intended for the early detection of attacking activity. The primary enabling technologies include host monitoring, reconfigurable deceptive virtual hosts and a network AB monitor. An anomaly detection system monitored the network activity of the honey pots. The role of the automatically deployed honey pots was to attract and possibly delay an intruder on the network. It reduces operational interaction and increase awareness of the security situation. In that system Ettercap is chosen for host identification. The honey pots can be a valuable tool in both research and production environments, they are

typically not easy to configure, particularly if the goal is to mimic a real network environment. Using intrusion detection system and using virtual honeypot
that enables hiding of the honey pots and thereby creating efficient attackers signature and detecting intruder characteristics. A honeypot is a host that has no real purpose other than to capture unauthorized activity. So honeypot reduces this problem by not having any true production traffic. It provides an environment where intruders can be trapped or vulnerabilities accessed before an attack is made an real assets.

REFRENCES

[1] Todd Vollmer, Senior Member, IEEE, and Milos Manic, Senior Member, IEEE, "Cyber-Physical System Security with Deceptive Virtual Hosts for Industrial Control Networks", IEEE Transactions on Industrial Informatics, VOL.10, No.2, MAY2014.
[2] J.Hieb and J.H.Graham,"Anomaly-based intrusion detection for network monitoring using a dynamic honey pot," Intell Syst.Res.Lab, UnivLouisville, Louisville, KY, TR-ISRL-04-03, Dec.2004.
[3] C.Hecker, K.L.Nance, and B.Hay,"dynamic honey pot construction," in proc. 10 the Coll.Inf. Syst. Secure.Edue.Adelphi, MD, USA, 2006.pp.4880-4889.
[4] V.Pothamsetty and M.Franz. SCADA Honeynet Project [online] Available: http://scadahoneynet.sourceforge.net/
[5] Digital Bond Incorporated. SCADA Honeynet [Online]. Available: http:// www.digitalbond.com/tools/scada-honeynet/
[6] M.A.McQueen and W.F.Boyer,"Deception used for Cyber Defense of Control Systems," in Proc. 2nd IEEE Conf. on Human System Interaction, May 2009.
[7] D.A.Shea"Critical infrastucture: control system and the terrorist threat, "Library of Congress RL315334, Jan.2004.
Intrusion Detection System with Deceptive Virtual Host For Industrial Control NW
[8] Y.Huang, et al,"Understanding the physical and economic consequences of attacks on control system,"Int. J. Critical Infrastructure Protection, pp.73-83, 2009.
[9] Jammi Ashok, Y.Raju, S.Munisankaraiah"Intrusion detection through honeypot," Jammi Ashok et. Al/ International Journal of Engineering Science and Technology, Vol.2 (10) 2010, 5689-5696.
[10] C.Rieger, D. Gertman, and M.McQueen,"Resilient control systems: next generation design research", in Proc.2nd IEEE Conf.HumanSystem Interactions, May 2009, pp.632-636
[11] Gaia Maselli, Luca Deri, Stefano Suin,"Design and Implementation of an Anomaly Detection System: an Empirical Approach", University of Pisa

