Honeypots Aiding Network Forensics: Challenges and Notions

QassimNasir and Zahraa A. Al-Mousa
Electrical and Computer Engineering, University of Sharjah, UAE
Email: nasir@sharjah.ac.ae; u00028486@sharjah.ac.ae

***Abstract*—**Communications over the Internet are under serious risks as attacks are increasing day after day. Network forensics is the process of investigation such attacks through analyzing network data and events. Many challenges are facing investigators due to the rapid growing of network scale and intruders' skills. Honeypots are computer traps that are meant to be compromised to attract hackers and monitor their strategies and tools. Using honeypots provides a cost-effective solution to increase the security of an organization. Monitoring malicious traffic is useful for network forensics and intrusion detection systems. This paper focused on studying network forensics methodologies and tools in addition to developing a well understanding of honeypots terminologies and their value in network forensics. Honeypot tools differ in several aspects discussed here in an objective comparison. Moreover, Honeypots efficiency is evaluated versus network intrusion detection and prevention system (IDPS). Data received by traditional network tools can be correlated with honeypots captures to obtain more valuable evidence and clues. The study also provides a literature review of previous researches on honeypots aid to network forensics in addition to multiple recommendations to overcome honeypot limitations.

***Index Terms*—**network forensics, honeypots, intrusion detection and prevention system.

I. Introduction

Network security is becoming the keystone issue in modern societies. New threats are arising every day demanding advanced security solutions that are no more provided by traditional tools. Investigating such attacks is a challenging task that is strongly related to what is known as network forensics. Network forensics aim is to introduce investigation capabilities in current networks. It refers to the investigation process that is done through analyzing network data and events. Network forensics is essential for exploring attacks from both inside and outside the network to detect threats and improve the system [1]. Traditional tools used in investigations, such as firewalls and Intrusion detection and prevention systems (IDPSs), are not further enough to reach all required evidence.

The idea behind honeypots emerged in the early nineties; mainly to solve security problems unsolved by traditional protection systems. In 1991, two famous publications stood out that are considered the foundations of modern honeypots: "The Cuckoos Egg" by Clifford Stoll and "An Evening with Berferd" by Bill Cheswick. In 1997, The Deception Toolkit, one of the landmark Honeypots, was released by Fred Cohen. It was a designed for UNIX systems that imitate a number of known vulnerabilities. In 2002, the Honeynet Project was improved to involve the whole security community under the name of Honeynet Research Alliance. As a result, Honeypot became a popular concept and more beneficial and stimulating for both researchers and companies. In 2003, a number of significant Honeypot tools were presented such as advanced virtual honeynets. Up to this day, honeypots is a hot area of study and new developments are still emerging in both wired and wireless environments.

A Honeypot is a computer or a trap that seems to contain valuable information or resources and appears to be part of a network, but is actually isolated and monitored. By attracting hackers to the system, updated and valuable data can be collected that helps in understanding attackers' methodologies and tools. Honeypots can be classified depending on their level of interactions to low, medium and high interaction honeypots. The higher the level of interaction, the more significant the collected data with increasing risks accordingly. Moreover, honeypots can be classified according to their purposes to research honeypots, and production honeypots. Another possible distinction in the area of honeypots differentiates between physical and virtual honeypots. From an investigative perspective, a honeypot is an ideal tool to closely study attackers and monitor their movements. Several studies relate honeypots to the field of network forensics [1]-[4]. Generally they consider honeypots as a tool or a technique that can be deployed in the network forensics process. This study intention is to understand the concept of honeypots as a pioneering security tool in a network forensics based infrastructure. The focus will be on the facilities that a honeypot can provide for a network forensics investigator. A number of honeypot tools are available as open sources for developers and researchers and are evaluated according to predefined factors. Honeypots can be compared to useful network forensics techniques such as IDPSs to clarify its usefulness to the intended area.

Manuscript received June 27, 2013; revised October 30, 2013.
Corresponding author email: nasir@sharjah.ac.ae.
doi:10.12720/jcm.8.11.700-707

The rest of the paper is divided as follows: section I discuses network forensics principles and challenges followed by section II that defines honeypots and its types of interactions. Section III presents an objective comparison of available honeypot tools. In the last two sections related work in honeypots aid to network forensics is presented in addition to recommendations and future work intended to address some of honeypot limitation.

II. Network Forensics Principles and Challenges

Network forensics is a sub-branch of digital forensics that is related to monitoring and analyzing computer network traffic for the purposes of collecting evidence. Network forensics is not considered a protection product; yet, it is not meant to replace firewalls and IDPS systems [1]. However, it is a complex procedure in which approaches, tools and professional intelligence are gathered to serve the investigation process.

TABLE I: SELECTED NETWORK FORENSICS TOOLS AND VALUE

Tool Name	Value to Network Forensics
Switches	Contain Content Addressable Memory, CAM tables, which determine the physical port corresponding to a predetermined MAC address.
Routers	Store routing table that maps ports to connected networks allowing tracing the path of network traffic through several networks. It can also function as a packet filters and logs denied traffic or maintain statistics on allowed traffic.
IDPSs	Monitor traffic in real time to detect and alert about threats and malicious activities
Firewalls	Analyze and log the source and destination addresses, packet payloads, port numbers, and encapsulated protocols to make a decision about certain traffic help the forensics process looking for evidence.
Web Proxies	Can be configured to maintain special logs for extended period of time. Such logs store the web surfing for an entire organization rather than a limited number of users. Forensic analysts can then examine data to find evidence such as sending emails, inappropriate web browsing habits or web-based malware.
Special Type Servers	The main types are DHCP Servers, Name Servers, Authentication Servers, Application Servers and Central Log Servers. The data stored in each server can because as clues to aid forensics investigations. The details of each server value for network forensics are out of this research scope.

Network forensic investigations can be complicated according to the case and available clues. The reason is that network forensics investigations need to deal with volatile and dynamic information in addition to unidentified people in different places [5]. Levels of Network forensics sources vary according to the network environment. For example, large financial institutions have different equipment, staff, and network infrastructure than other small offices of government agencies. Yet, similarities always exist in tools, equipment and methodologies. Switches, routers, web proxies, firewall and IDPS are common technologies used in network forensics. Each of them has certain values to the forensics process. The data stored in each server or in log files can beused as clues to aid forensics investigations. Table I. clarifiesthe assessment of some traditional tools used to aid network forensics investigations. The value of each tool is measured by the quality and accuracy of data provided in a certain investigation event.Switches and routers are meant to serve network architecture rather than serving security explicitly. However, their critical positionsin the network infrastructure lead designers to add some methodologies to aid security requirements and investigations. Firewalls and Web proxies consider network security as a central constraint; although not intended explicitly to aid network forensics. Among traditional forensics tools, IDPSs are vital in any network forensics based infrastructure.

Network intrusion detection and prevention systems are designed as security tools that facilitate security analysts and forensic investigations. IDPSs monitor traffic in real time to detect and alert about threats and malicious activities [5]. The performance of these two tools depends on many factors such as the location, number and capacity of sensors placed in the network topology. IDPS may be configured to focus on certain events or destinations during an ongoing investigation to serve forensic investigations. Despite the valuable services IDPS offers to the network environment, it is generally wise to keep it somewhat separate from the production environment so that it does not adversely affect it performance by increase latency or attracting intruders to exploit its' vulnerabilities [6].

Network environment contains numerous possible sources of evidence such as IDSs, wireless access points, web proxies or central log servers; making it critical to pinpoint the correct location of an evidence or even gain access to it for political or technical reasons. Furthermore, Network devices may not store all content of files and their metadata due to some design issues such as the storage capacity. Instead, only selected data is kept that may or may not contain the full details of evidence.Furthermore, data may be so volatile due to the absence of a secondary or persistent storage. Any reset or failure may result in losing significant data.Due to several issues related to network forensics as a new area in digital investigations, there exist conflicting or nonexistent of legal guides for admission of various network-based digital evidence. All these challenges and moreleads researchers to engage more inventive tools to aid network forensics investigations process.

III. Honeypots Basics and Types of Interaction

Honeypot concept emerged in the early nineties to overcome the cumulative security risk that wasn't handled by common protection and detection systems such as firewalls. In [7] honeypot was defined as an information system resource whose value lies in unauthorized or illicit use of that resource. This definition highlights all the different indices of honeypots. Honeypots generally has no authorized activity, which means any interaction with a honeypot is most likely to be unauthorized or malicious activity.The value of honeypots is weighted by the data that can be gained from them. Generally a honeypot monitors every action an attacker makes including access attempts, keystrokes, resources accessed and modified, and processes executed. Analyzing these data provide significance information that is not available for IDPS or other protection systems.

Honeypots are novel technique to attract attack and fool them; however, there exist some legal issues that should be taken into consideration. Countries have diverse laws regarding honeypot usage, information capturing and data security. In general, honeypots laws are dependent on the captured data quality and the person who employ it and his/her goals. In general, it is wise to ask your company, network administrator, or a lawyer about the validity of your implementation. Privacy issue is related to the installer right to collect information from intruders. For example collecting information from employees in an organization by one of their colleges is a doubt. Privacy is also related to how much data are captured by a honeypot in relevant to the laws. High interaction honeypots involve more risks and contain more significant information. In addition, honeypots typically do not provide public accounts which indicate that they are not a service provider and not bounded by any privacy requirements intended for service.Different classifications demonstrate honeypots usage, structure and levels of interactions.

Honeypots can be differentiated into physical and virtual honeypot. Physical honeypot implies honeypot running on a physical Hardware machine. It is commonly associated with high-interaction, where the intruder is dealing with a full computer and an operating system. The downside of such a type is the scalability and flexibility of the system. Physical honeypot are typically expensive to install and maintain. For large address spaces, it is impractical or impossible to install a physical honeypot for each IP address. The disadvantages of Physical honeypot lead to the virtual honeypots. Virtual honeypots deploy one physical computer that hosts multiple virtual machines acting as honeypots. The main advantages are scalability and ease of maintenance. Thousands of honeypots can run on one machine and most of them are commercial, inexpensive and available to almost everyone. Some existing honeypot tools run several operating systems and their applications simultaneously on a single physical machine, making it much easier to collect data.

Honeypots can be classified according to their purposes to research and production honeypots. Research honeypots aim to discover new threats and malicious attacks and study the hackers' intentions and techniques. This type is mostly related to military area, researches and government organizations. Production honeypots aim to protect organizations and companies from threats. Generally Production honeypots are installed inside the production network to improve the overall security. In contrast to research honeypots, production honeypots capture limited amount of information, employing low interaction honeypots. By monitoring the network and capturing malicious traffic, network administrators can analyze the attacker attitude and conclude with some results. These results help lower the risks that may come from it towards the company. The admin working on such a project should be careful for his system not to be exploited to attack other systems using honeypot features.

Moreover, honeypots can be categorized according to the level of interaction permitted to attackers. The More amounts of data the higher level of interaction with increasing the risks accordingly. Low, medium and high levels of interaction are considered the three recognized types implemented currently.

A. Low Interaction Honeypots

Low interaction honeypotis the easiest one to configure and install. The attacker in this case will not deal with a real operating system but he can leave some fingerprints about his strategy and tools. The amount of data collected is considered small compared to other honeypot systems which indicate fewer risks. The reason is that Low interaction honeypots is not designed to represent a totally featured operating system and accordingly, it cannot be completely exploited. The network services simulated, such as TCP and IP, are just enough to deceive an intermediate adversary and make him believe that he/she is dealing with a real system. As a result, Low interaction honeypots can be used to identify new worms or tools and to analyze the network traffic in the purpose of measuring the security threats and weaknesses or the frequency of available attacks. Some well-known examples of low-interaction honeypots are: The Deception Toolkit, Nepenthes, HoneyD and Dionaea.

B. Medium Interaction Honeypots

Medium interaction honeypots involve additional risks but collect more data than low interaction honeypots. Similarly, there is no real operating system to interact with. Yet, more security holes exist from which a hacker can access the system. Consequently, the received attacks are more complicated and serious and lead to deeper analysis. Some of well-known medium interaction honeypots that are used today are: Mwcollect, honeytrap, Multipot and Nepenthes.

C. High Interaction Honeypots

High interaction honeypots are considered the most advanced type which involves high risks. Unlike low and medium interaction honeypots, there is a real operating system rather than simulated tasks. The hackers can obtain full control on a machine while he/she is monitored. Thus, more significant information can be gained about the attackers' tools, tactics, and motives. This type of interaction is cost-efficient method and it can add to intrusion detection systems (IDS) some advantages.

There are several draw backs of using such type of interaction. It is a time-consuming process and it requires a dedicated machine to be installed in; since it may threat personal private data. Using virtuality is significant in this case. Virtual high-interaction honeypots are implemented by deploying one physical machine that hosts several virtual machines acting as honeypots. The benefits of using such methodology are in three key points: ease of configuration, ease of maintenance and less risk. First, the virtual machine should be downloaded, deployed at a physical machine, and executed. After that, the deployment of virtual high interaction honeypots is simple because of the available solutions that offer an already preconfigured honeypots. Virtual high interaction honeypots are also easy to maintain; since it can be restored to the original state after being done of watching a previous attack. Moreover, virtuality in honeypots poses fewer risks as it is less likely to compromise or corrupt the actual data by an intruder. Common examples of high inter action honeypots used today are: Honeywall CDROM, Sebek, High Interaction Honeypot Analysis Toolkit (HIHAT), and HoneyBow.

IV. Low Interaction Honeypot Tools: An Objective Comparison

Different commercial honeypot tools were released to serve security demands and requirements. Few recent studies conducted a proper comparison between existence honeypot tools. One of the significant studies was done by the ENISA, the European network and information security agency [8]. With contribution of a number of expert authors, the study compared tools according to specific detection scopes: quality of data collected, accuracy of emulation, reliability, scalability and performance, extensibility, ease of use, embedability, support and cost. Level of rating provided ranges from excellent to poor while an overall evaluation is rated by its usefulness to Computer Emergency Response Teams (CERT). The results show that the most useful general purpose honeypots is Dionaea followed by HoneyD and HoneyBOT.

Table II: Selected Network Forensics Tools and Value

	Dio-naea	Hon-eyD	Glastopf	Hone-yBOT	Nepe-ntheses	Decep-tionToolk-it
Ease of Config.		✓	✓	✓		✓
Free Version	✓	✓	✓	✓	✓	✓
Open Source	✓	✓	✓	✓	✓	✓
Web Simulation			✓	✓		
Hard Det. by Attackers			✓			
Support of ipv6	✓		✓	✓	✓	
VoIP Module	✓					
Capture Malware	✓	✓		✓	✓	
Det. New Attacks	✓	✓	✓	✓		
OS Emulation.		✓			✓	

The central motivation to conduct our own comparison is to choose the best tool according to our perspectives and objectives. Our target was low interaction honeypots since they are the best choice for beginners to understand honeypot functionality and value.The comparison focused on different factors that fit our research scope and goals. Table II. presents the evaluation results of available low interaction honeypot tools based on literature review of books, publications and official websites. The comparison factors were chosen according to features that affect network forensics area. The main factors are: ease of configuration, open source availability, free version availability, web emulation capability, operating system emulation capability, detection of malware and hard detectability by attackers.Sex tools were compared, the main attention of each tool is as follows: Dionaea to capture malware, HoneyD to simulate various operating system services, Glastopfto simulate web application, HoneyBOT to mimic vulnerable services, Nepentheses to emulate common Windows services and Deception toolkit to emulatesingle host services.

Hard detectability by attacks is a factor measured by the ways obtainable for attackers to detect they are dealing with a honeypot. This is a major limitation in any honeypot tool since it affect its significance and amount of data captured; because an attacker will disconnect as soon as he/she discovered the fool system.Tools such as

Dionaea, HoneyD and HoneyBOT can be discoverable by attackers with few NMAP (Network Mapper is a free and open source to perform network discovery and security auditing) scripts. Deception Toolkit and HoneyDare among the first tools presented in the market as free and open source products. Deception Toolkit had the least features while HoneyD, Dionaea and Glastopf had the highest. Some categories where obtain based on reviewers and experiences of users such as hard detectability by attackers.

UNIX-Like Systems are superior in security features compare to the other OSs on the market. Linux ecosystem is evolving fast in terms of variety, number and quality of applications in addition to other complementary services such as availability of support. Compared to Windows, security problems are less disturbing Linux such that an independent study has shown that Linux kernel has 0.17 security flaws per 1,000 code, lines while this average is about 10-20 flaws of proprietary software [12]. Among the tools compared, HoneyD, Dionaea, Deception Toolkit and Nepetheses can be installed in UNIX platform. Linux network scripts and management should be studied carefully to serve the research requirement if any were deployed.

Compared to the study evaluation in [8], our outcomeswere similarin a way or another. Dionaea is the highest recommended because of the variety of services it supports and good quality of emulation. It is a new tool that embedsPython as scripting language and able to detect shellcodes using Libemu and supports IPv6Glastopf is the best for Web simulation to detect web attacks. HoneyD is a stable solution for beginners to understand honeypots and for administrators to add more complexity to virtual networks due to the large number of IP addresses it can occupy. HoneyBOTaccording to our results comes in the same level with HoneyD but for windows platform. It is not sensiblefor network forensics investigations to deployNepentheses and Deception Toolkit due to their limitations and poor supports.

V. Honeypots Aiding Network Forensics: Challenges and Related Work

Honeypots can be very substantial in network forensics. Network forensics is meant to investigate and gather evidence to analyze or answer questions in a security issue. Honeypots offer several log files that a forensic party can analyze. It can also imitate the original victim, to repeat the analysis process from the beginning without losing any important data [5]. Using honeypots provides a cost-effective approach to assist the network forensics investigations. Instead of monitoring and sniffing the whole network traffic, honeypots receives only malicious traffic. Focusing on the target traffic leads to much easier investigation and analysis. Forensic analysis of a honeypot data generally differ from a similar analysis of data collected by a compromised hosts in two crucial points: all traffic from the honeypot itself is malicious, and a full capture of data sent and received by honeypot is usually available [3].

Honeypots are powerful to understand attacker behavior, how attacks are conducted and techniques used by attackers. This is addressed by the fact that honeypots capture all the activity received by intruders including connection requests, port scans, malware transfer and others. All such activities that enters or leaves the honeypots must be logged to be analyzed later either to look for evidence or to support other forensics tools such as IDPSs and the network forensic analysis tools (NFAT) (Network forensic analysis tools (NFAT) focus primarily on collecting and analyzing wired network traffic. An NFAT stores most or all of the traffic that it sees, and then performs analysis, unlike a network-based IDPS, that performs exhaustive analysis and stores only the necessary network traffic).

Honeypots are also available as open source codes and are easy to download and configure. In case downloaded within an organization, honeypots can capture significant data without any significant costs or maintenances' work. It is also effective in decreasing the analyzing time; since it receive only malicious traffic rather than logging all ongoing connections such common sniffing tools like Wireshark.

Honeypots can imitate the services of an existence victim [2]. This may be influential on the forensics process. For example, if an old victim was a router standing in a critical part of an organization, a honeypot has the ability to attend to be this router, simulating its operating system, IP address and open ports. This will attract similar traffic to that of the previous victim and may lead to the real suspicious.

Almulhem in [1] choose honeypots to be one of the three main models in network forensics, along with IDPS and computer forensics. He claimed that from an investigative perspective, honeypot is a powerful tool to study attackers and capture their tools. However, certain limitations exist such as legal issues. A honeypot is exclusively setup to be compromised and has no value; compromising it does not incur any damages legally to be used as evidence. Moreover, honeypots can be regarded as a margin between keeping attackers out of a network and inviting them and may be judged as unfair entrapment.

F. Raynal *et al*. in [3] and [4] presented the term Honeypot Forensics to describe the forensics investigation aided by honeypot functionality. The main claim was that honeypot goal is to improve our understanding of blackhats from two points of views: technical and ethnological. By increasing the verbosity of honeypots logs and traces, every action the intruder makes is discovered. Choosing honeypot type, way of setup and its environment are very significant since they affect analysis. The author illustrated that system analysis is an important part of honeypot forensics. System analysis includes information obtained from logs of basic security services such as a firewall, IDPS, kernel, and

others. In the study honeypots where deployed and traffic captured to and from a honeypot was logged. Forensics were applied on honeypot, and correlated with system and network events. The analytical results showed that Common forensics differ from forensics applied to the honeypot in amount of information, quality of information and tools. A honeypot has almost too many pieces of information not available in a compromised host. Honeypot enclose high quality of significant information while a compromised host usually has a lot of useless traffic. Regarding tools, honeypot still suffer from sufficient tools to analyze data.

Meghanathan *et al*. in [11] discussed the use of honeypots and honeynets in network forensics and more precisely in gathering intelligence about intruders, there tools and strategies. Similar to honeypots, a honeynet is a network explicitly designed to be compromised. By attracting intruders, there tools can be identified in addition to vulnerabilities exploited. A honeywall, employed by honeynets, is a protection system that acts as firewalls to protect non-honeypot systems from attacks originating from a compromised honeypot. A honeywall functions by being setup with several data control to limit outbound connections allowed per hour, bandwidth available to the attacker's traffic and can also modify some malicious data packets on-the-fly. By capturing all traffic leaving and entering a honeynet, useful information about attackers' strategies can be gained to improve the efficiency and performance of a honeynet. Ther role of network forensics stated by [11] was based on the fact that any traffic directed to any honeynet is considered an attack or intrusion. Consequently, forensic analysis of its activities is less likely to lead to false negatives and false positives compared to IDPS. When sufficient evidence is collected within a honeynet, an administrator can stop running the honeynet and analyze the malicious traffic collected.

A. Evaluating Intrusion Detection and Prevention Systems Versus Honeypots

Intrusion Detection and Prevention Systems (IDPSs) were discussed briefly in section II as a network forensics tool. In fact, IDPSs are vital in any network environment that seeks high security levels. It is wise to compare such a powerful technology with honeypots to outline how both systems serve network forensics investigations. The main areas in which they differ are information gathering, detectability and protection ability.

In information gathering, IDPSs monitor the whole production level traffic, including internal and external packets. On the other hand, honeypots are resources that are expected to be accessed only by an adversary. Moreover, not all attacks are received to honeypots; yet, only those who targeted the honeypot are received and logged. Although this seems to be an advantage for IDPSs over, it may add to honeypots that they are bandwidth and time efficient. Honeypots needs less storage and bandwidth requirement and it requires less time for a forensics investigator to look for his/her target evidences.

Detectability is another distinguish factor between the two systems. To define suspicious activities, IDPSs use set of rules to match packets. Generally, a rule consists of four main subjects: type of packet to search, a string of content to compare, a location where that string is to be searched for, and a related action to take if all the conditions are met. Although there are many rules in various forums, usually the core of any rule is in the matching process between strings and anyplace in the packet payload. The challenge is in the computational overhead that requires searching every byte in each packet for a potential match from a large library of strings. In contrast, honeypots can be a detection source without any sort of computations of matching. It receives the connection request and discovers it is an attempt by a certain adversary. This is a simple and powerful way but not complete. Moreover, honeypots are inherently less susceptible to false positives but on the cost of greater administration overhead. In addition, they will not be able to detect attacks directed at a production resource, not to the honeypot itself. This indicates that IDPSs systems cover larger range of attacks and attack types against a network, but at a price of higher false positives.

Providing protection capabilities to the system is another important factor. Honeypot by itself doesn't provide protection services. However, it can be designed to do so with an overhead administration work. Honeypot information can be correlated with other system logs to prevent attacks. IDPSs can provide a significant level of protection; however it may not be able to achieve full analysis under high loads or congested networks [6].

Despite all factors,Honeypots has its limitations as a security system in any production network; it can capture attacks targeting it only, some experienced hackers will easily understand that he/she is attacking a honeypot system and If not secured enough, honeypot may be used as a zombie to attack other systems.In general, Honeypots and IDPS can be realized as complementary technologies. Honeypots can detect attacks that are missed by IDPS; While IDPSs can be used as part of a system to redirect attackers away from production resources to a honeypot to be analyzed.

VI. Recommendations and Future Work

There are several recommendations that should be considered while employing honeypots to serve Network forensics. A network administrator should be aware of all honeypots limitation and consequent risk. He/she should know where and why to deploy honeypots in network forensics infrastructure to reach the highest possible compensations. One of the most vital limitations of honeypots is that it is easily detected by attackers. Soon or after a period of time, an intruder will know that he/she was fooled by a honeypot machine and stop sending messages and maybe succeed to change his/her

location and information to escape from suspicion. A possible solution might be to enhance honeypot view by intruders by adding security services such as firewalls. Those security services can be weak such that they are easy to be compromised.

Honeypots can also be automated if it is employed in each node of the system without affecting its security concerns. Nyre in [9] proposed a scheme that is capable of tolerating attacks and talented to preserve the integrity, confidentiality and availability of the system in the same time. A honeypot state is issued on every legitimate machine whenever an attack is detected and the system lets the attacker retain some control. By not notifying the intruder of the detection, he is left exploiting the honeypot and providing beneficial information to system administrators. This strategy has its limitations in the sense it issue honeypot state after an attack takes place, knowing that some novel attacks cannot be detected easily by common detection tools.

A honeypot based IDPS model is a possible optimization that can address some of both IDPS and honeypots limitations. Artail et al. in [10] proposed an adaptable honeypot-based approach that improves theIDSs and their protectionvalues. The main idea was to deploylow-interaction honeypots installed in the production network andhave them direct malicious traffic to high-interaction honeypots, where hackers deal with real services. After capturing traffic it detects attacks through making use of an IDS tool, using the results to take administrative actions toward protecting the network.The proposed design has its limitations since it depends on HoneyD tool specifically while more powerful tools emerged like Dionaea. Moreover, the focus in the later design was in capturing more traffic and analyzing it despite how this traffic was correlated with IDS database and forensics evidences. Our recommended approach is to design an intelligent system that adds the data captured by honeypots to IDPSs database storage. Fig. 1 shows a typical topology that illustrates how data are connected in a similar model. This topology is meant to be inside an organization network which has several LAN connected through a management network. This management network contains the basic security devices such as firewall and IDPS. Each separate LAN encloses one traffic sensors or a honeypot that receive malicious traffic from both inside and outside a network. Typically, these honeypot sensors log attacker's information such as IP and destination addresses and malware transfer. Logs are sent periodically to the management network to be correlated with IDPSs data base via an integrated intelligent system. Now, IDPSs library of suspicious behavior or baseline model is updated with valuable and very recent data that are not only used in forensics investigations, but also help in preventing the system from future attacks. The model can be generalized to work in cloud levels to provide more additions to the global network security. Each LAN networks and management network can be viewed as cloud in the large network. Communications in this case will be through cloud computing protocols and standards.

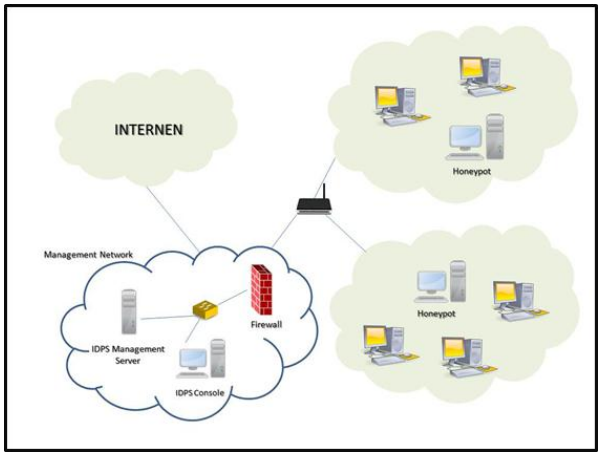


Figure 1. A distributed honeypot based IDPS model

A. *Future Work Plan in Honeypotsand Network Forensics*

Honeypots can be extremely vital in network forensics area. Several motives encourage us to go further and look for statistics and indications that prove the efficiency or even the necessary of deploying honeypots in any network forensics based infrastructure. Our future work will focus on studying and simulating honeypot basedIDPSmodel in clouds where several honeypots will be installed in different clouds.Initially, we will deal with low interaction honeypots since they are easy to configure and install. Afterward we will extend our study to deployhigh interaction honeypots to collect more significant data. The legal issue is a key consideration in the implementation step; since each country has its own regulations regarding honeypot usage and information capturing. The choice of which tool to use will be based on our prospective comparison. A suitable one is Dionaea, due to the variety of services it supports and good quality of simulation. It is also a new product that offers considerable support and bugs fixing. Data collected from each honeypot sensor will be sent in an appropriate format to a centralized IDPS; such that information will be added automatically to the IDPS library. Our work requires a proper knowledge of cloud computing strategies and standard. The efficiency of such a design will be evaluated according to the value of evidences it captures in addition to the detection and protection rates. Statistics with illustration graphs should be provided to aid any network forensics party seeking better performance and productivity.

VII. CONCLUSION

Attackers are getting more skilled as innovativesecurity technologies emerge. It is a two side war between attackers and researchers seeking robust security systems. Network forensics not only leads to identify the suspicious person, but it can also lead to further protect the network. Traditional tools such as

firewall and web proxies offer limited amount of data that needs the aid of novel practices such as honeypots. The research studies the concept of honeypots, level of interactions and limitations. Available tools can be compared to aid users and specially beginners to choose a suitable product among them. Our results were similar to previous conducted studies on the same area. Honeypots provide an updated source of information about attacks methodologies and tools in addition to several log files. It can imitate the original victim, to repeat the analysis process. It also provides a cost-effective approach; instead of sniffing the whole network traffic, honeypots receives only malicious traffic. However, honeypots are not enough to discover all clues and evidences that lead to the right target. Evaluating honeypots versus IDPSs shows that both technologies can be complementary. As a recommendation, honeypots can be automated or correlated with IDPSs management servers. Our future work willhelp to understand how honeypotscan help to improve the efficiency of a network forensics based infrastructure. By simulating a distributed honeypot based IDPS model, statistics can be provided to evaluate how honeypots can aid network forensics investigations and help to collect evidences.

REFERENCES

[1] A. Almulhem, "Network forensics: Notions and challenges," presented at the IEEE International Symposium on, Signal Processing and Information Technology, UAE, Ajman, Dec 14-17, 2009.

[2] D. Akkaya and F. Thalgott, *Honeypots in Network Security, Bachelor Project*, Linnaeuos University, Sweden, 2010.

[3] F. Raynal, D. Kaminsky, P. Biondi, and Y. Berthier, "Honeypot forensics, Part I: Analyzing the network," *IEEE Security & Privacy*, vol. 2, no. 4, pp. 72–78, July 2004.

[4] F. Raynal, D. Kaminsky, P. Biondi, and Y. Berthier, "Honeypot forensics, Part II: Analyzing the network," *IEEE Security & Privacy*, vol. 2, no. 5, pp. 77–80, July 2004.

[5] S. Davidoff and J. Ham. *Network Forensics Tracking Hackers Through Cyberspace*, USA: Pearson Education, 2012.

[6] K. Scarfone and P. Mell. "Guide to intrusion detection and prevention systems (IDPS)," *National Institute of Standards and Technology*, Gaithersburg, NIST Special Publication, 2007, pp. 800-94.

[7] N. Provosand and T. Holz, *Virtual Honeypots: From Botnet Tracking to Intrusion Detection*, 1st ed. Boston, USA: Addison Wesley Professional, 2007.

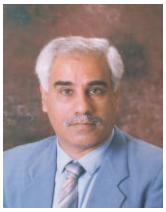
[8] T. Grudziecki *et al*., "Proactive detection of security incidents," Document Report, ENISA, 2012.

[9] A. Nyre, "Increasing survivability by dynamic deployment of honeypots," M.S. thesis, Dept. of Telematics, Univ. of Science and Technology, Norwegian, 2005.

[10] H. Artail, H. Safab, M. Sraj, I. Kuwatly, and Z. Al-Masri, "A hybrid honeypot framework for improving intrusion detection systems in protecting organizational networks," *Computer Security*, vol. 25, no. 4, pp. 274–288, 2006.

[11] N. Meghanathan, S. Allam, and L. Moore, "Tools and techniques for network forensics," *International Journal of Network Security & Its Applications*, vol.1, no.1, pp 14-25, April 2009.

[12] N. Economides and E. Katsamakas, "Linux vs. Windows: A comparison of application and platform innovation incentives for open source and proprietary software platform," in *Economics of Open Source Software Development*, Elsevier Publishers, 2006.



Qassim Nasir is currently an associate professor in University Of Sharjah since 2009, and assistant Professor in University Of Sharjah since 2001. He got the CISSP to add best practices in security to his academic background. In his current position Dr. Nasir involved in consulting industrial project in the area of telecommunication, communication and network security, and network design. Dr. Nasir taught courses in CISCO as trainer and offer many courses in the area of mobile computing, analogue and digital telecommunications, computer networks, network programming, web-based remote controls, web based instrumentation, and programmable logic controllers. Prior to joining the University of Sharjah, UAE in 2001 and for six years, Dr. Nasir was working with Nortel Networks, Canada, as a senior system designer, and project leader. Dr. Nasir was leading SONET OC192, OC-12, then he moved to ADSL group where he leads firmware design for NORTEL ADSL modems. Dr. Nasir was visiting professor at Helsinki University of Technology, Finland, during the summers of 2002 to 2009, and GIPSA lab, Grenoble France to work on a Joint research project. Dr. Nasir has contributes in the research in digital communications, Secure network design, Security/Power/Quality/Direction aware MAC and Network protocols, and haptic data transmission protocols. He also offers consultancy in IT security management, security penetration testing. Dr Nasir has published over 70 as conference publications and journal articles, two book chapters in the areas of Digital communications, computer Networks, and data security.

Zahraa Al-Mousa, received her B.S. in Computer Engineering (Honors) from University of Sharjah in 2010. Her main Interest is network security issues and solutions. She is currently finishing her thesis investigating in cooperative intrusion detection models in cloud computing. Zahraa has previous experiences in web pages integrity and network forensics.