

Article

A Web-Based Honeypot in IPv6 to Enhance Security

Keyong Wang ¹, Mengyao Tong ¹, Dequan Yang ^{2,*} and Yuhang Liu ³

¹ School of Continuing Education, Beijing Institute of Technology, Beijing 100081, China; wangkeyong@bit.edu.cn (K.W.); tongmengyao@bit.edu.cn (M.T.)

² Network Information Technology Center, Beijing Institute of Technology, Beijing 100081, China

³ School of Computer, Beijing Institute of Technology, Beijing 100081, China; liuyh@bit.edu.cn

* Correspondence: yangdequan@bit.edu.cn

Received: 5 August 2020; Accepted: 9 September 2020; Published: 12 September 2020



Abstract: IPv6 is a next-generation IP protocol that replaces IPv4. It not only expands the number of network address resources but also solves the problem of multiple access devices connected to the Internet. While IPv6 has brought excellent convenience to the public, related security issues have gradually emerged, and an assessment of the security situation in IPv6 has also become more important. Unlike passive defense, the honeypot is a security device for active defense. The real network application and the fake network application, disguised by the honeypot, are located on a similar subnet, and provide a network application service; but, in both cases, behavior logs from unauthorized users are caught. In this manner, and to protect web-based applications from attacks, this article introduces the design and implementation of a web-based honeypot that includes a weak password module and an SQL inject module, which supports the IPv6 network to capture unauthorized access behavior. We also propose the Security Situation Index (SSI), which can measure the security situation of the network application environment. The value of SSI is established according to the different parameters that are based on honeypots. There is a firewall outside the test system environment, so the obtained data should be used as the real invasion data, and the captured behavior is not a false positive. Threats can be spotted smartly by deploying honeypots; this paper demonstrates that the honeypot is an excellent method of capturing malicious requests and can be measured with the SSI of the whole system. According to the information, the administrator can modify the current security policy, which can improve the security level of a whole IPv6 network system.

Keywords: IPv6; web-based honeypot; situation awareness; network security; Security Situation Index (SSI)

1. Introduction

We are entering the all-digital society of the Internet of Everything, where the number of communication entities is exponentially increasing; but, except for computers and mobile phones, all smart devices are assigned an IP [1]. The total of about 4 billion IP addresses that IPv4 can provide in the “Internet of Everything” is not enough. IPv6 is the sixth version of the IP address protocol. It is designed to have an address length of 128 bits and can provide 2^{128} -bit IP addresses. Nearly infinite IP addresses allow IPv6 to accommodate a large number of devices [2]. Together with technologies such as 5G, it will support the rapid development of emerging technologies such as mobile Internet, Internet of Things, industrial Internet, cloud computing, big data, and artificial intelligence to effectively meet new demands for future business.

While IPv6 has brought great convenience to our lives, to address this there are developments designed to protect servers like those outlined by the authors of [3]. For example, the problem of exhaustion of NAT device address pools leads to Distributed deny of service (DDoS) attacks. Secondly, the Neighbor Discovery Protocol (NDP) used by IPv6 can discover other nodes on the

same link in time. Through these nodes, the address can be effectively parsed, the link router can be clarified, and the neighbor reachable information can be maintained. Through the neighbor discovery protocol, the intruder can send the wrong router announcement, which will cause the IP packet to be transmitted to the unspecified location to implement packet modification, reject the packet service or data interception. In addition, the existence of the IPv6 protocol mobility feature provides convenience for node communication. Due to the unfixed features of the mobile node, it also gives the criminals a chance. Therefore, it is imperative to strengthen the research on defense mechanisms. Among them, security situation awareness is an important part that cannot be ignored. IPv6 moves to be a large environment in the network world [4]. Some security issues in IPv6 have been focused recently [5], are using multi-address generation and duplicate address detection to prevent deny of service (DoS) in IPv6. Fernández et al. [6] did some security works in Vehicular IPv6 communications. In Internet of Things (IoT) field, there are also new security issues [7].

Honeypots are deployed as an active network security protection technology that can trick an attacker into attacking a noncritical environment. A honeypot is a safe or noncritical environment specially set up to trap attackers and allowing researchers to observe how real attackers behave. It is a deceptive system containing vulnerabilities. Network services or information can be spoofed to entice attackers to attack them, so as to capture and analyze the attack behavior, understand the tools and methods used by the attacker, infer the attack intention and motivation, and enable the defender to understand the security threats they face, and the technical and management methods to enhance the security protection capabilities of the actual system. Security researchers and analysts usually use honeypots for the purpose of observing the actions of attackers, focusing on defense research. CISOs and defensive teams can use this information to improve the security status of the service system by studying new attack methods and implementing new defenses to better resist security threats, reduce network security risks, and protect server security. As a phishing tool, honeypots have no production value. Every request to honeypots is considered as a suspicious behavior. Its value can be measured by the information it captures. It can obtain valuable data by analyzing various types of information captured [8]. Therefore, many applications with IPv6 address are vulnerable against attacks, IPv6 also can be a protocol in IoT. In the IoT world, W. and B. et al. [9] used a honeypot to capture malicious requests. N. and D. et al. give an analysis of honeypot architectures and presents a small, low-cost, configurable platform for use as a “lightweight” honeypot [10].

The main contribution of this paper is to create a web-based honeypot in the IPv6 environment and provide its Security Situation Index (SSI) to measure the Situation Awareness of the network. We also do some experiments to test the value of web-based honeypot and SSI. This paper adopts honeypot technology to capture the behavior of intrusion detection and finally realizes IPv6 security situation awareness. A multi-honeypot system is deployed in the system for network security situation awareness, attracting and transferring attacks from illegal users, confusing the attacker, wasting their time and traffic attacks, and collecting data on the behavior and characteristics of the attacker. The system continuously captures the attack samples, obtains the SSI for measuring IPv6 and performs different types of emergency response levels according to the index. The defense system can formulate a defense scheme based on various information of the intruder that is fed back, and dynamically implement a defense strategy, such as blacklisting and current limiting, to restrict access to fixed-form features. Overall, it is possible to achieve a measure of security awareness and defensive implementation.

2. Related Works

2.1. Security Situation Awareness

Since the concept of security situation awareness has been proposed, its related research has been advancing. Liu and Yu et al. [11] proposed a fusion method which assign different weights to different data sources to improve the accuracy of fusion. They analyzed a cognitive adjustment mechanism

to solve the problem of automatic control. This is the first discussion to use Cognitive Awareness Control to address Network Security Situation Awareness (NSSA) regulatory issues. Zhao and Liu discuss the network security situational awareness in the big data environment, and established the network security situational awareness index system. The index factors are selected and quantified, and then the network security situational awareness system is constructed by calculating the situation value [12]. Zhang and Shi et al. introduce the concept and challenges of NSSA and discusses ways to solve them [13]. They provided an assessment of the state of cyber security and how to apply it to NSSA. At last, they proposed a multi-level analysis framework of NSSA. Liang, Y. and Wang, H.Q. et al. combine evolutionary strategies and neural networks to propose a quantitative method for network security situational awareness [14]. The evolutionary strategy was used to optimize the parameters of the neural network, and then the evolutionary neural network model was established to extract the network security status factors and quantify the network security status. Wang and Liang et al. propose a research method of network situational awareness based on stochastic game theory [15]. In their work, a network situation-aware quantification method based on network service state analysis was proposed and a network attack and defense game model including both attackers and defenders was also constructed.

2.2. Honeygot and IPv6

Honeygot technology is one of the ways to help to spot threats in the IPv6 network space. Typically used in the online server as a bait for the primary server to mitigate the attack to the bait instead of the primary server. Recently, honeygot research has also expanded to many fields, including camouflage deception, data capture, data control, and data analysis. There are several kinds of honeygot methods, some include blockchain enabled architecture, SDN-enabled architecture, social network-based architecture, service port-based architecture gameplay model based, etc. The details of research as follows: Shi, L. and Li, Y. et al. [16] think that applying the dynamics of honeygot to system services shows that real or fake services (honeygot) can be changed in different hosts. In addition, they use a blockchain platform to decentralize the system and store port access data by providing a private chain. Fan, W. and Du, Z. et al. [17] propose a novel honeygot architecture called HoneyDOC to support a full range of honeygot design and implementation. The HoneyDOC architecture clearly identifies three basic independent and collaborative modules: baits, traps, and coordinators. Based on an efficient architecture, a software-defined networked honeygot system is designed that provides a high degree of programmability and technically maintains the ability to capture high quality data. Paradise, A. and Shabtai, A. et al. [18] propose a framework for managing social network honeygot, and analyzes the deployment process of social network honeygot and their maintenance in actual social networks. The honeygot profile has been successfully absorbed into the organization's social network and received suspicious friend requests and emails showing the basic signs of a potential attack. La, Q.D and Quek, T.Q. et al. [19] addresses the issue of defending against attacks in honeygot-enabled networks by studying gameplay models that include the spoofing of attackers and defenders. Attackers may try to deceive defenders by suspicious activity to seemingly normal activities, and defenders can in turn use honeygot as a deception to trap the attacker. The problem is modeled as a Bayesian game with incomplete information, where both the one-game and the repeated game versions are determined to be equalized. The results show that there is a threshold for the frequency of active attackers, at which both players will take deceptive action. Below this threshold, defenders can mix their strategies while keeping the attacker's success rate low.

To deal with the security problem on an IP address level, Kishimoto, K.; Ohira, K.; Yamaguchi, Y, et al. [20] did some honeygot work in IP protocol layer. Scheffler T, Schindler S, Schnor B, et al. provide a honeygot named HoneydV6. Besides IPv6 packet processing, HoneydV6 implements necessary parts of the ICMPv6 and the Neighbor Discovery Protocol (NDP) [21,22]. Prof. Dr. Bettina Schnor proposes two different honeygot architectures and presents the corresponding prototype implementations, called Honeydv6 and Hyhoneydv6, to overcome the need for IPv6 honeygot. To catch the behavior of

attack on IPv6 Extension Headers/Fragmentation Mechanism/Flow Label, some honeypot support IPv6 network in http mode, but there is no weak password and SQL injection module [23]. Zuzcak M, Sochor T. [24] propose an FTP honeypot which based on port 22 in SSH mode. Zabal L, Kolar D, Fujdiak R. [25] survey honeypot problematics and deception-based defensive strategies in the cyberworld.

2.3. Access Control and Security Issues in IPv6

Access control is a traditional basic issue in security field. Masood, A.; Ghafoor, A. et al. [26] propose an approach for conformance testing of implementations required to enforce access control policies specified using the Temporal Role-Based Access Control (TRBAC) model. Uddin, M.; Islam, S. et al. [27] defined an Authorising Workflow Task Role Based Access Control using the existing task and workflow concepts. It integrates the dynamic Segregation of Duties (SoD) considering the task instance restriction to ensure overall access governance and accountability. It enhances the existing access control models such as Role Based Access Control (RBAC) by dynamically granting users access right and providing Access governance. Liu, Q.; Zhang, H. et al. [28] present an access control model for resource sharing based on the role-based access control intended for multi-domain Manufacturing Internet of Things (MIoT). In multi-domain systems, to respond to the assigning request for permission for the certain role from the certain user, an authority action sequence named the authorization route is employed to determine an appropriate authorization state. Traditional trust models are based on reputation which is only a numerical value. Therefore, it is not fit for fine-grained access control which is needed in many online applications. Towards this problem, Xu, C.; Wang, Y. et al. [29] proposed a novel trust model based on temporal historical data for access control. In addition, there are community-based security access models [30], trust-based fuzzy access control methods [31], etc., which provide an attempted approach to IPv6 security system access control. There are some Network-based IPv6 attack types, as follows: Address Spoofing, Duplicate Address Detection Attack, Prefix spoofing attack [32]. Redirection Attack, Attacks on Cryptographically Generated Address (CGA), Attacks against IPv6 Service Applications, Security risks under IPv6 and IPv4 dual stack.

To the best of our knowledge, there are many IPv4 based honeypot, but still, few honeypots support the IPv6 mode. In comparison with other honeypots, we focus on the web application types of honeypot that gives a virtual machine configurable. The novel web-based honeypot in IPv6 described in this paper enables us to deployed easily in different network areas includes core areas, Demilitarized Zone (DMZ) areas, client access areas, etc., which can test the granularity defense capability of access control in access control field. Our environment is capable of investigating the attack techniques include weak password and SQL injection attacks. The case of brute force cracking in SSH or RDP protocol will not be considered according to the character of this kind honeypot.

3. Design of a Honeypot-Based IPv6 System

There are different kinds of attacks in each layer, even each protocol. For different types of protocol, there should be a different kind of honeypot. Therefore, we propose a web-based HTTP service honeypot in port 80, which includes the weak password module and the SQL injection module. And then, we present a group of honeypots to get the index of network security situation and detect the whole security level of the network.

We developed a honeypot-based IPv6 security situation awareness system that is a trap to catch attackers. Since honeypots do not provide truly valuable services to the outside world, all attempts at honeypots are very suspicious; another use of honeypots is to delay the attack action of the attacker on the true target and disperse the attacker's attention driving the attacker to waste time on the honeypot and make them to abandon the unsuccessful attack. This platform can monitor the IPv6 system security situation in real time, capture the behavior of illegal users, and use the honeypot system to transfer attacks from the main system to the trap system. Through analysis of behavior logs Process and analyze the behavior of the attacker.

The honeypot system is created using a docker container based virtual machine with the following specifications: 2 CPU/4GB/1Gb Ethernet IPv6. The honeypot can be configured using a different based on CPU/memory/network arguments. The honeypot should deploy in the same network as the web server with the firewall and connect to the internet.

The goal of the honeypot is to store every request from the attacker to the web application; so, in this honeypot, several low-interaction web applications are installed: Weak password module and SQL injection websites module. The attacker makes a malicious request, this honeypot will process the request, and meantime get the log of malicious behavior, and replies to the administrator. Therefore, we use the docker container to provide several web applications as the main body of the honeypot which include: Section 3.1 Basic log system, Section 3.2 Weak password Website and Section 3.3 SQL injection Website. To measure the security level of the test network, we describe the Index of Security Situation Awareness in Section 3.4.

3.1. Fake Website Logs

Logs are the most primitive information records of system operation, and they are the most basic means for detecting network security. The server generates a large number of logs every day: system logs, access logs, database logs, etc. The logs record a large amount of information about system operation, including program execution, data operations, operating errors, system crashes, and IP information and operation information accessed by users and many more. Analyzing the logs can mine the system operation status to ensure the normal operation of the system can quickly locate operating errors and handle optimizations, analyze server traffic and user behavior characteristics to help provide better services, quickly find security issues and make a timely response.

The purpose of the virtual website is to generate a login log. This article creates a virtual website. The website implements a simple login function page and content display page. It generates logs by simulating login and records related log information such as time, username, password, and IP, and supports IPv6 access. Enter the username and password through the login page, the system is matched, and if it is correct, the display page is entered. In this system we used the ELK. We designed a log management system using a technology solution combining Docker container and ELK (Elasticsearch + Logstash + Kibana) [33]. The business process design is shown in Figure 1.

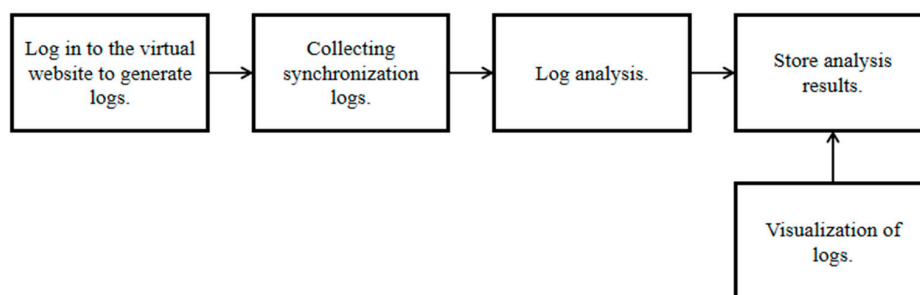


Figure 1. The framework of flow chart.

3.2. Weak Password Website Module in the Honeypot

In the Internet environment, simple password is the biggest risk faced by the server. Although everyone knows that setting a longer, more complex password will be more secure, there are always some users who use simple, easy-to-remember password strings for convenience. Weak passwords are not strictly and accurately defined. Generally, passwords that are easily guessed by others (they may know you well) or cracked by cracking tools are weak passwords. A weak password is a password that contains only simple numbers and letters, such as “123”, “abc”, etc. In the experiment, when an intruder tries to invade the honeypot network, each user needs to enter the login information, including the username and password, which will be displayed in the log and track the intruder’s IP address.

3.3. SQL Injection Website Module in the Honeypot

SQL injection attacks are one of the common methods used by hackers to attack databases. The user can submit a piece of database query code and get some data he wants to know based on the results returned by the program. This is SQL injection. SQL injection is accessed from a normal port, and it looks no different from ordinary web page access. Therefore, firewalls on the market will not warn about SQL injection. If the administrator does not have the habit of viewing logs, they may not be aware of the intrusion for a long time.

The system consists of virtual websites and set up a group which based on “Docker plus ELK” [33] architecture to form a honeypot to fool attackers and record log information. Docker can easily create a lightweight, portable container for any application. The role of the virtual website is to induce logins and record information. The main function of ELK is log management and analysis. ELK is deployed in Docker containers. The system architecture design is shown in Figure 2.

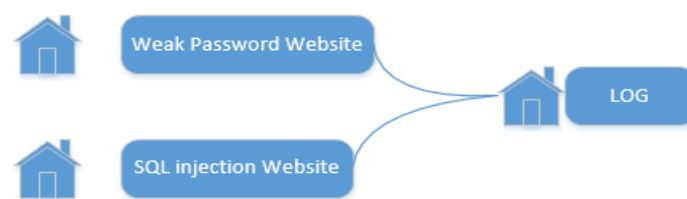


Figure 2. The framework of system architecture.

3.4. Index of Security Situation Awareness (SSI)

As we all know, in view of websites, the computer network environment can be divided into several levels of security. There are core security areas, medium-security areas, and general security areas. We deploy different numbers of honeypots in different levels of the application service network area. Each honeypot has two fake services to test security. According to the data from the network environment, we can compute the SSI. Definitions can be defined as follows:

Definition 1. Suppose there are three kinds of areas sets: high-security area (H), middle-security area (M) and low-security area (L).

Definition 2. Suppose that the honeypots (HH_i) located in high-security area include weak password style (HH_iW) and SQL injection style (HH_iI). i is the number of honeypots.

$$\begin{aligned} HH_i &= HH_iW + HH_iI \\ HH_iW &\in \{0, 1\}, \\ HH_iI &\in \{0, 1\}. \end{aligned} \quad (1)$$

If the honeypot catches the attack attempt of weak password behavior, the value of HH_iW is 1. Else the value is 0.

If the honeypot catches the attack attempt of SQL injection behavior, the value of HH_iI is 1. Else the value is 0.

Definition 3. Suppose that the honeypots (MH_j) located in middle-security area include weak password style (MH_jW) and SQL injection style (MH_jI). j is the number of honeypots.

$$\begin{aligned} MH_j &= MH_jW + MH_jI \\ MH_jW &\in \{0, 1\}, \\ MH_jI &\in \{0, 1\}. \end{aligned} \quad (2)$$

If the honeypot catches the attack attempt of weak password behavior, the value of MH_jW is 1. Else the value is 0.

If the honeypot catches the attack attempt of SQL injection behavior, the value of MH_jI is 1. Else the value is 0.

Definition 4. Suppose that the honeypots (LH_k) located in low-security area includes weak password style (LH_kW) and SQL injection style (LH_kI). k is the number of honeypots.

$$\begin{aligned} LH_k &= LH_kW + LH_kI \\ LH_kW &\in \{0, 1\}, \\ LH_kI &\in \{0, 1\}. \end{aligned} \quad (3)$$

If the honeypot catches the attack attempt of weak password behavior, the value of LH_kW is 1. Else the value is 0.

If the honeypot catches the attack attempt of SQL injection behavior, the value of LH_kI is 1. Else the value is 0.

Definition 5. SSI is supposed as the situation security index.

$$SSI = x \sum_0^m HH_i + y \sum_0^n MH_j + z \sum_0^p LH_k \quad (4)$$

$i \in (0, m); j \in (0, n); k \in (0, p)$ which present the number of honeypots in High-security area (m), Middle-security area (n) and Low-security area (p). The value of x, y, z depend on the Weighting Factor in the location of level of security. Different area has different weight factor. Weight is a parameter, shown in Table 1; it depends on the importance of the detection service application environment deprived from the Model for Measuring Value of an Asset Based on Confidentiality Integrity Availability (CIA) [34].

Table 1. Weight Factor based CIA.

	Impact on Business				
	Very Low	Low	Medium	High	Very High
Factor	1	2	3	4	5

Figure 3 displays the base unit of SSI.

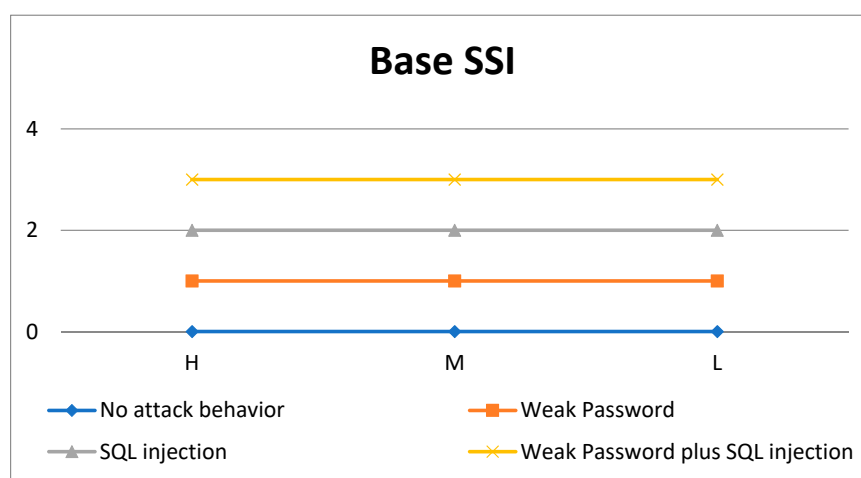


Figure 3. The base unit of SSI.

4. Experiment and Discussions

We need to test the proposed honeypot methods, so we developed two web applications: a weak password application port with 2022, [http://\[2408:400a:111:5600:a09d:3ab:d70d:648d\]:2022](http://[2408:400a:111:5600:a09d:3ab:d70d:648d]:2022) and shown in Figure 4; a SQL injection application port with 2023, [http://\[2408:400a:111:5600:a09d:3ab:d70d:648d\]:2023](http://[2408:400a:111:5600:a09d:3ab:d70d:648d]:2023) shown in Figure 5. As there is a firewall in front of the honeypot, a normal user cannot access these websites unless you attend the firewall in an inner network environment. The basic process is to arrange these two security traps, spoofs security vulnerabilities, deploys security traps to invade attackers, then collect attack information, because the honeypot system does not provide services to the outside, and there are no resources available, so any attempt to connect to the honeypot is suspicious. SSI is also computed in this section with different cases to measure the level of security.

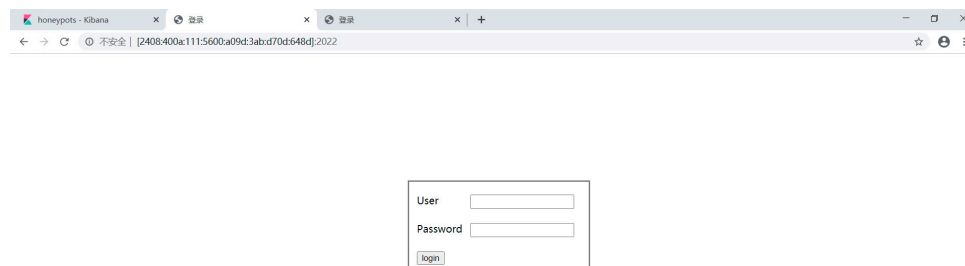


Figure 4. A weak password application website.

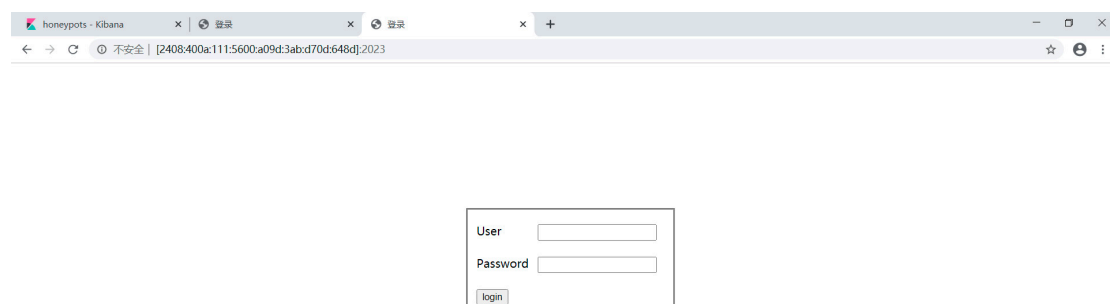


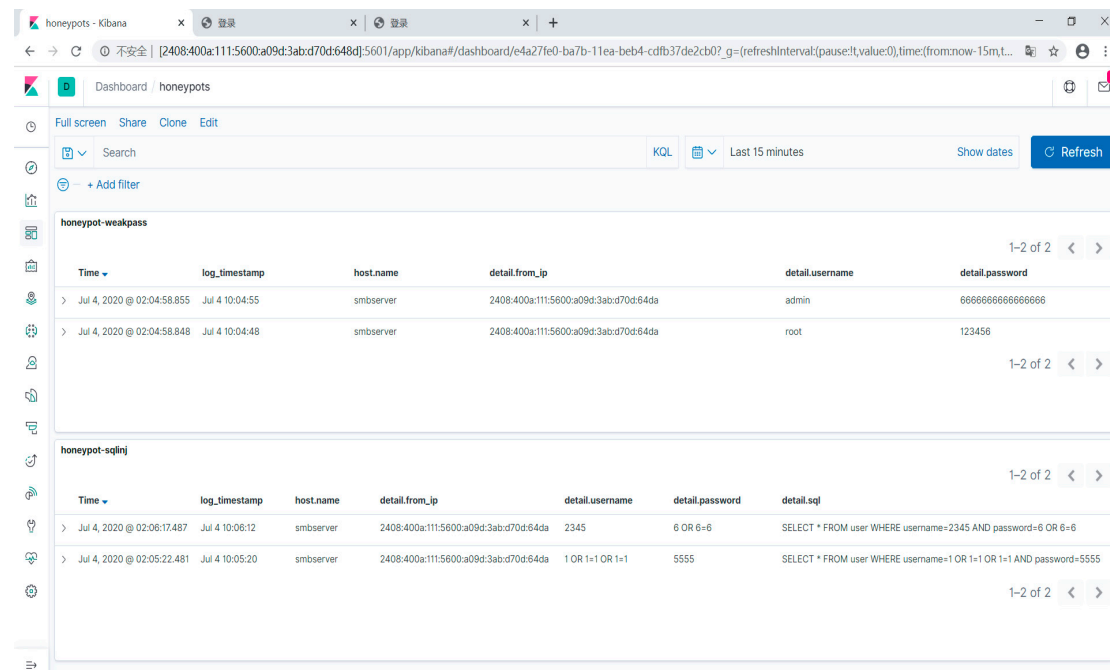
Figure 5. A SQL injection application.

4.1. Honeypot Situation Awareness Parts

The log collection function is to synchronize the logs generated by the virtual website to Logstash in real time, Log analysis is completed by the Logstash filter plugin. This system mainly analyzes

and get the statistics the IP in the login information, using regular expression plugin (GROK) which supports IPv6 matching and resolution.

Converting the analyzed data into more intuitive charts to show the analysis results can make it easier for us to understand various analysis results and capture abnormal situations. We configure the visual display website to display the behavior or attackers. The interface with data is shown in Figure 6.



Time	log_timestamp	host_name	detail.from_ip	detail.username	detail.password
> Jul 4, 2020 @ 02:04:58.855	Jul 4 10:04:55	smbserver	2408:400a:111:5600:a09d:3abd:70d:64da	admin	6666666666666666
> Jul 4, 2020 @ 02:04:58.848	Jul 4 10:04:48	smbserver	2408:400a:111:5600:a09d:3abd:70d:64da	root	123456

Time	log_timestamp	host_name	detail.from_ip	detail.username	detail.password	detail.sql
> Jul 4, 2020 @ 02:06:17.487	Jul 4 10:06:12	smbserver	2408:400a:111:5600:a09d:3abd:70d:64da	2345	6 OR 6=6	SELECT * FROM user WHERE username=2345 AND password=6 OR 6=6
> Jul 4, 2020 @ 02:05:22.481	Jul 4 10:05:20	smbserver	2408:400a:111:5600:a09d:3abd:70d:64da	1 OR 1=1 OR 1=1	5555	SELECT * FROM user WHERE username=1 OR 1=1 OR 1=1 AND password=5555

Figure 6. Visual Display.

In the honeypot environment, defenders can systematically collect, analyze, and organize all digital evidence without the intruder's sense. Analyzing the log files based on the honeypot can help the defender determine the source and intent of the attack, and processing and combing the collected attack information can guide the configuration of defense device policies. The defender can cooperate with the honeypot system to implement linkage work, set strict access control policies on the network boundary area, core routers, and firewalls to form an active defense ecological chain and improve the level of network security defense. To evaluate the effectiveness of this method, we deployed the honeypot in the IPv6 based network environment. At the end, a fault visit is caught. The next step is that we need to locate the invade IP address, find the attacker, and improve the level of base line security policies.

4.2. The Index of Security Situation Awareness

We set some different cases to evaluate the value of Index of Security Situation in defined application environment. In the network application environment, three level of security protection are set in the experiment, different numbers of honeypot are set in Table 2 and Weighting Factor are set in Table 3. The argument 'a' is the number of honeypots in high-security area (H), 'b' is the number of honeypots in the middle-security area (M) and 'c' is the number of honeypots in the low-security area (L). The arguments 'x', 'y', 'z' are parameters which should be set in advance.

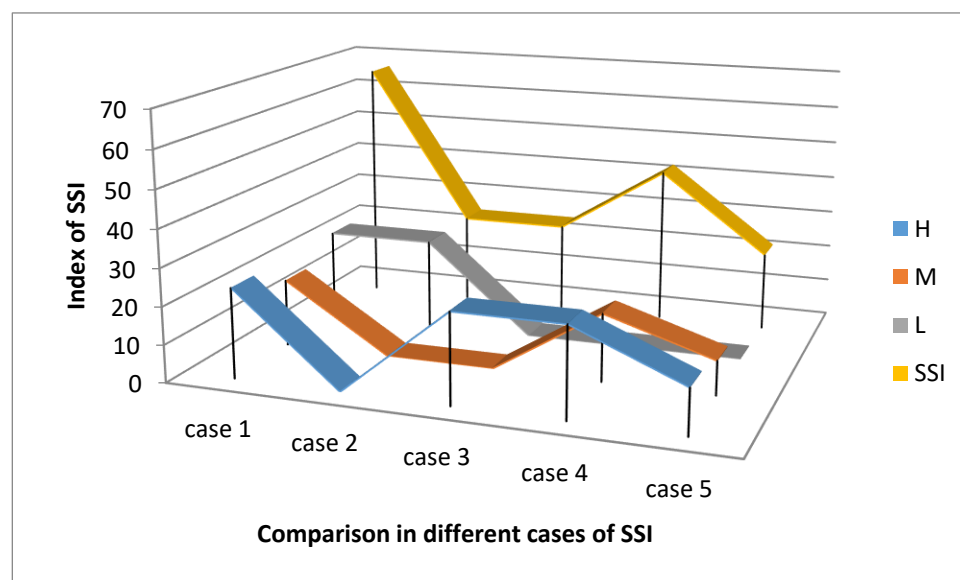
Table 2. Number of different honeypot locations.

	<i>a</i>	<i>b</i>	<i>c</i>
case 1	2	2	4
case 2	0	0	4
case 3	2	0	0
case 4	2	2	0
case 5	1	1	0

Table 3. Weighting Factor.

	<i>x</i>	<i>y</i>	<i>z</i>
Weighting Factor	4	3	2

According to the Definition 1–5, we can compute the value of SSI, which shown in Figure 7. This result can help the administrator of defender understand the security situation of the protected network application. The smaller the amount of SSI, the safer the network application.

**Figure 7.** The Comparison in different cases of SSI.

According to the design of the above experiments, we have verified the honeypot-based web application in IPv6 and the SSI in the whole information system:

- (1) The model presented in this paper can run in an IPv6 network environment;
- (2) The security situation of the whole network can measure by deploying a group of honeypots in different security areas of the system.

5. Conclusions

A perfect honeypot deception can even make an attacker feel that they have not easily achieved the desired goal and convinced them that the intrusion was successful. From the perspective of network security protection, honeypot security technology, as an active security defense method, can effectively combat network attacks. This technology can play an important role in detection, protection, and response. It can find attacks, delay attacks, and play the role of defense against attacks.

In this paper, we proposed a web-based honeypot in IPv6 network environment with the main advantage of measuring the security situation awareness with SSI. In actual testing, the honeypot is quickly deployed using Docker technology. This method is easy to deploy, highly effective, and can effectively capture malicious attacks and record the attacker's information. Security administrators can update the policies to improve the ability of protecting the whole application system. The attacking IP captured by the honeypot can be rejected as a blacklist in firewall policies that essentially eliminates the possibility of an attack. In the meantime, these kinds of honeypot systems can measure the level of whole system security. In the future, we will create a low-interaction honeypot in more network modes.

Author Contributions: K.W. and D.Y. conceived and designed research; All authors wrote the initial paper; Y.L. and D.Y. conducted research; D.Y. and Y.L. revised the paper; All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by CERNET innovation Project grant number NGII20180407.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bouras, C.; Gkamas, A.; Primpas, D.; Stamos, K. IPv6 deployment: Real time applications and QoS aspects. *Comput. Commun.* **2006**, *29*, 1393–1401. [\[CrossRef\]](#)
2. Montavont, J.; Roth, D.; Noël, T. Mobile IPv6 in Internet of Things: Analysis, experimentations and optimizations. *Ad Hoc Netw.* **2014**, *14*, 15–25. [\[CrossRef\]](#)
3. Žagar, D.; Grgić, K.; Rimac-Drlje, S. Security aspects in IPv6 networks—Implementation and testing. *Comput. Electr. Eng.* **2007**, *33*, 425–437. [\[CrossRef\]](#)
4. Gomez, C.; Minaburo, A.; Toutain, L.; Barthel, D.; Zuniga, J.C. IPv6 over LPWANs: Connecting Low Power Wide Area Networks to the Internet (of Things). *IEEE Wirel. Commun.* **2020**, *27*, 206–213. [\[CrossRef\]](#)
5. Guangjia, S.; Hui, W.; Hangjun, W. Using multi-address generation and duplicate address detection to prevent DoS in IPv6. *IET Commun.* **2019**, *13*, 1390–1396. [\[CrossRef\]](#)
6. Fernandez, P.J.; Santa, J.; Bernal, F.; Skarmeta, A.F. Securing Vehicular IPv6 Communications. *IEEE Trans. Depend. Secure Comput.* **2016**, *13*, 46–58. [\[CrossRef\]](#)
7. García-Guerrero, E.E.; Inzunza-González, E.; López-Bonilla, O.R.; Cárdenas-Valdez, J.R.; Tlelo-Cuautle, E. Randomness improvement of chaotic maps for image encryption in a wireless communication scheme using PIC-microcontroller via Zigbee channels. *Chaos Solitons Fractals* **2020**, *133*, 109646. [\[CrossRef\]](#)
8. Sanders, C.; Smith, J. Chapter 12—Using Canary Honeypots for Detection, in *Applied Network Security Monitoring*; Sanders, C., Smith, J., Sanders, C., Smith, J., Eds.; Syngress: Boston, MA, USA, 2014; pp. 317–338.
9. Zhang, W.; Zhang, B.; Zhou, Y.; He, H.; Ding, Z. An IoT Honeynet based on Multi-port Honeypots for Capturing IoT attacks. *IEEE Internet Things J.* **2019**, *7*, 3991–3999. [\[CrossRef\]](#)
10. Eliot, N.; Kendall, D.; Brockway, M. A Flexible Laboratory Environment Supporting Honeypot Deployment for Teaching Real-World Cyber security Skills. *IEEE Access* **2018**, *6*, 34884–34895. [\[CrossRef\]](#)
11. Liu, X.; Yu, J.; Lv, W.; Yu, D.; Wang, Y.; Wu, Y. Network security situation: From awareness to awareness-control. *J. Netw. Comput. Appl.* **2019**, *139*, 15–30. [\[CrossRef\]](#)
12. Zhao, D.; Liu, J. Study on network security situation awareness based on particle swarm optimization algorithm. *Comput. Ind. Eng.* **2018**, *125*, 764–775. [\[CrossRef\]](#)
13. Zhang, H.; Shi, J.; Chen, X. A Multi-Level Analysis Framework in Network Security Situation Awareness. *Procedia Comput. Sci.* **2013**, *17*, 530–536. [\[CrossRef\]](#)
14. Liang, Y.; Wang, H.Q.; Lai, J.B. Quantification of Network Security Situational Awareness Based on Evolutionary Neural Network. In Proceedings of the 2007 International Conference on Machine Learning and Cybernetics, Hong Kong, China, 19–22 August 2007.
15. Wang, H.; Liang, Y.; Liu, X. Stochastic Game Theoretic Method of Quantification for Network Situational Awareness. In Proceedings of the 2008 International Conference on Internet Computing in Science and Engineering, Harbin, China, 28–29 January 2008.
16. Shi, L.; Li, Y.; Liu, T.; Liu, J.; Shan, B.; Chen, H. Dynamic Distributed Honeypot Based on Blockchain. *IEEE Access* **2019**, *7*, 72234–72246. [\[CrossRef\]](#)

17. Fan, W.; Du, Z.; Smith-Creasey, M.; Fernández, D. HoneyDOC: An Efficient Honey-pot Architecture Enabling All-Round Design. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 683–697. [\[CrossRef\]](#)
18. Paradise, A.; Shabtai, A.; Puzis, R.; Elyashar, A.; Elovici, Y.; Roshandel, M.; Peylo, C. Creation and Management of Social Network Honey-pots for Detecting Targeted Cyber Attacks. *IEEE Trans. Comput. Soc. Syst.* **2017**, *4*, 65–79. [\[CrossRef\]](#)
19. La, Q.D.; Quek, T.Q.; Lee, J.; Jin, S.; Zhu, H. Deceptive Attack and Defense Game in Honey-pot-Enabled Networks for the Internet of Things. *IEEE Internet Things J.* **2016**, *3*, 1025–1035. [\[CrossRef\]](#)
20. Kishimoto, K.; Ohira, K.; Yamaguchi, Y.; Yamaki, H.; Takakura, H. An Adaptive Honey-pot System to Capture IPv6 Address Scans. In Proceedings of the 2012 International Conference on Cyber Security, Washington, DC, USA, 14–16 December 2012.
21. Schindler, S.; Schnor, B.; Kiertscher, S.; Scheffler, T.; Zack, E. HoneydV6: A low-interaction IPv6 honey-pot. In Proceedings of the 2013 International Conference on Security and Cryptography (SECRYPT), Reykjavik, Iceland, 29–31 July 2013.
22. Schindler, S.; Schnor, B.; Kiertscher, S.; Scheffler, T.; Zack, E. IPv6 Network Attack Detection with HoneydV6. *Commun. Comput. Inf.* **2013**, *456*, 252–269.
23. Honey-pot Architectures for IPv6 Networks. Available online: <https://www.cs.uni-potsdam.de/bs/research/docs/thesis/2016/schindler.pdf> (accessed on 2 September 2020).
24. Zuzcak, M.; Sochor, T. Application of Honey-pots in IPv6 Networks. In *AIP Conference Proceedings*; AIP Publishing LLC: Melville, NY, USA, 2015.
25. Zabal, L.; Kolar, D.; Fudjak, R. Current State of Honey-pots and Deception Strategies in Cybersecurity. In Proceedings of the 2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Dublin, Ireland, 28–30 October 2019; pp. 1–9.
26. Masood, A.; Ghafoor, A.; Mathur, A. Conformance Testing of Temporal Role-Based Access Control Systems. *IEEE Trans. Depend. Secure Comput.* **2010**, *7*, 144–158. [\[CrossRef\]](#)
27. Uddin, M.; Islam, S.; Al-Nemrat, A. A dynamic access control model using authorising workflow and task-role based access control. *IEEE Access* **2019**, *7*, 166676–166689. [\[CrossRef\]](#)
28. Liu, Q.; Zhang, H.; Wan, J.; Chen, X. An Access Control Model for Resource Sharing Based on the Role-Based Access Control Intended for Multi-Domain Manufacturing Internet of Things. *IEEE Access* **2017**, *5*, 7001–7011. [\[CrossRef\]](#)
29. Xu, C.; Wang, Y.; Wei, Q.; Wang, Q. A Novel Trust Model Based on Temporal Historical Data for Access Control. In Proceedings of the 2009 International Conference on Computational Intelligence and Security, Beijing, China, 11–14 December 2009.
30. Hussein, D.; Bertin, E.; Frey, V. A Community-Driven Access Control Approach in Distributed IoT Environments. *IEEE Commun. Mag.* **2017**, *55*, 146–153. [\[CrossRef\]](#)
31. Mahalle, P.N.; Thakre, P.A.; Prasad, N.R.; Prasad, R. A fuzzy approach to trust based access control in internet of things. In Proceedings of the Wireless VITAE 2013, Atlantic City, NJ, USA, 24–27 June 2013.
32. Gu, K.; Zhang, L.; Wang, Z.; Kong, Y. Comparative studies of IPv6 tunnel security. In Proceedings of the 2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Guilin, China, 29–31 July 2017.
33. Rochim, A.F.; Aziz, M.A.; Fauzi, A. Design Log Management System of Computer Network Devices Infrastructures Based on ELK Stack. In Proceedings of the 2019 International Conference on Electrical Engineering and Computer Science (ICECOS), Batam Island, Indonesia, 2–3 October 2019.
34. Shemlse Gebremedhin Kassa, CISA, CEH. 1 May 2017 ISACA JOURNAL. Available online: <https://www.isaca.org/resources/isaca-journal/issues/2017/volume-3/it-asset-valuation-risk-assessment-and-control-implementation-model> (accessed on 2 September 2020).

