International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 24 (2017) pp. 14944-14947
© Research India Publications. http://www.ripublication.com

IP-in-IP Encapsulation Methods using Proxy Server for Medical Information Exchange

Hak Joon Lee, Yong Kyu Park and Yong Gyu Jung*

Innogs Korea, inc., Innoplex 203, Yangsan-ro 57-5, YoungDeoungPo-gu, Seoul, 07271, Korea.

Korea Internet & Security Agency, IT Venture Tower, 135 Jungdae-ro, Songpa-gu, Seoul 05717, Korea.

**Eulji University, Department of Medical IT, 553 Sanseong-Daero, Sujeong-gu, Seongnam-si, Gyeonggi-do, 13135, Korea.*
*(*Corresponding Author)*

Abstract

Due to the development of wireless internet and IoT environment, healthcare services have also started to be provided based on IoT environment. Service in IoT environment has the same attribute as Internet communication based on general TCP / IP. Service server attacks due to client hacking, which is one of the weaknesses of internet communication, also appear in communication in IoT environment. Especially, IoT devices that are clients of IoT environment are often attacked by service servers because they often have insufficient system resources to apply security. In this paper, we propose a method to protect the service server in the network infrastructure.

Keywords: IP-in-IP encapsulation, IoT, Security, Hidden of Server IP address

INTRODUCTION

After the era of smartphones, smart phones such as smart watch, smart band, smart scale and smart shoe have been improved to connect to the Internet. Qualcomm and other telecommunication module manufacturers have made it possible to miniaturize Bluetooth transmitter and receiver module and 3G / LTE transmitter and receiver module and operate with low power, and at the same time provide mass production at low cost. So, not only traditional smart devices like smartphones and tablets but also products like watch, scale, shoe, clothes, necklace, and bracelet, which were generally thought to be difficult to connect to the network, became available. And services that make meaningful use of the information collected through these products have started to emerge in earnest.

The IoT device and the service server communicate over the Internet. These services need to prepare for the hacking of personal information and the DDoS attack of the service server, which is a problem as much as sending and receiving data through the Internet. In this paper, we propose a server security infrastructure system that protects a service server against hacker attacks by preventing the IP of the service server from being exposed even if the IoT device as a client is hacked in the data transmission.

PROBLEMS AND SOLUTIONS

A typical Internet communication is a structure for connecting and transmitting data between a client and a service server through IP. The client can directly access the service server by inputting the IP of the service server to access the service server, and can confirm the IP of the service server through the DNS using the address composed of the domain name and access the service server. DNS is a service that connects domain and IP. After purchasing domain, it registers IP to connect to domain through DNS. Client obtains IP address of domain from DNS. After that, it connects to the service server through the corresponding IP.

Typical Internet communication is accessed via IP, and the connecting IP is exposed to the client. Therefore, when a hacker hacks a client and acquires the IP of the server, the server is exposed to various threats including a hacker's DDoS attack. Therefore, instead of accessing the service server directly when connecting to the service server, it often uses a method to block the attack which is a threat through a security device such as a firewall, IPS (or IDS) in front of the service server.

And since the attack of the service server starts usually from the hacking of the client, the security of the client becomes a lot of care. Install a variety of security solutions on the client to prevent the risk of hacking by installing antivirus or preventing infection by malicious code. Devices such as smart phones, tablets, and PCs are usually used as such. But general healthcare products are IoT concepts and there are many small and light products, so system resources are not enough. Because it often goes to a minimum resource, it is faithful to the function of the healthcare and there are many cases that it is poor in security. Even if it is not IoT product, it can hack any kind of products such as PC and smart phone. Even if the device itself is not hacked, it is possible to secure the IP of the service server to intercept the communication packet leaving the device. Among the communication methods in the Internet network, the communication method through the SSL is also very likely to expose the service IP because the data can be encrypted but not encrypted to the IP header. That is, there is a problem that the hacker attacks the service server by the IP exposure of the service server and disables the service server, thereby disabling the Healthcare System.

In addition, hackers can acquire the IP of the service server by hacking the client, and the hacker can unauthorized the healthcare-related data to be transmitted to the service server. Since healthcare data are mainly health related data of users who use clients, they belong to sensitive information as personal information and hackers can sufficiently exploit them. In addition, since the service server changes the data and sends it to the service server, the service server collects and judges the erroneous information, so that the service server can deliver the result different from the original result to the client.

DEVELOPMENT, RESULTS AND DISCUSSION

If the service server's security infrastructure system only hides the IP of the service server, a proxy server between the client and the service server may be sufficient by storing the information of the service server in the proxy server. However, from the perspective of the service server, it is necessary to check client IP which client has sent the data. For the service server, the information of the client that generates and sends the actual data is more important than the proxy server. It is possible to provide the client's information to the data in the sending and receiving packets, but if it is possible to verify the client's information on the Internet communication without doing so, there is no need to include a process of separately checking the client's information between the client and the service server.

To implement this function, the proxy server used in the relay system is set to 2. In addition to the proxy server provided through DNS, there is a proxy server that performs another function. A proxy server provided through DNS is referred to as a 1st proxy server and a proxy server performing another function is referred to as a 2nd proxy server. The role of the 1st proxy server is to create an IP packet (Inner IP) that secures the client's IP when receiving the packet from the client and sets the secured client IP as the source IP and the destination IP as the service server IP and to add IP packet of the IP-in-IP method is added by adding an IP header (Outer IP) in which the IP of the 1st proxy server is set as the source IP and the IP of the 2nd proxy server is set as the target IP in front of the corresponding IP packet. Since the proxy server can control and generate IP packets, IP generation and control functions are used to create IP packet headers such as those sent by the client to the service server, and the source IP, which is an original function of proxy server, to generate an IP header of the IP-in-IP method.

If the 1st proxy server transmits an IP packet in the IP-in-IP format to the 2nd proxy server, the 2nd proxy server removes the Outer IP header from the received IP-in-IP format IP packet, and transmits Inner IP packet to the destination IP in the network. The IP header of the packet transmitted to the service server through the 2nd proxy server has a client IP address in the source IP address, so that it has the same effect as if the client directly transmitted to the service server. In addition, since the 2nd proxy server need not know the information of the service server, unnecessary sharing of the service server information can be prevented.

As shown from Figure 1, the packets in the communication between the 1st proxy server and the 2nd proxy server are IP-in-IP type IP packets, and the IP header of the Outer IP contains the IPs of the proxy servers. In the IP header, you can see the IP of the client and the IP of the service server. In communication between the 2nd proxy server and the service server, the Outer IP header is removed and the Inner IP is transmitted as it is. Thus, it can be seen that the IP of the client is set in the source IP and the IP of the service server is set in the target IP. For a service server, it would appear that the client is sending packets directly to the service server. Using this method, the service server can check the information of the client, so that the administrator can take action on the client, which is a problem when a problematic client occurs.

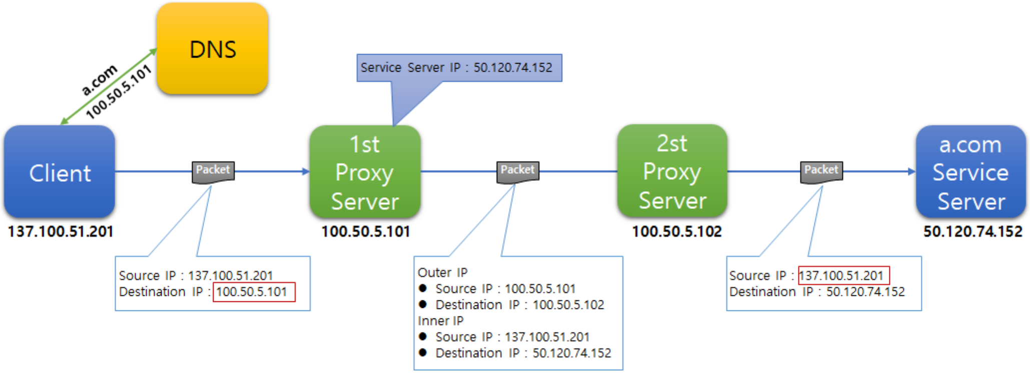
DNS
a.com
100.50.5.101
Service Server IP : 50.120.74.152
Client
137.100.51.201
Packet
1st Proxy Server
100.50.5.101
Packet
2st Proxy Server
100.50.5.102
Packet
a.com Service Server
50.120.74.152
Source IP : 137.100.51.201
Destination IP : 100.50.5.101
Outer IP
● Source IP : 100.50.5.101
● Destination IP : 100.50.5.102
Inner IP
● Source IP : 137.100.51.201
● Destination IP : 50.120.74.152
Source IP : 137.100.51.201
Destination IP : 50.120.74.152

Figure 1: Structure of IP-in-IP for getting client IP

EXPERIMENT AND RESULTS

When the client analyzes the packet through the Wireshark in the communication of the 1st proxy server, it can be confirmed that the target IP is 192.168.100.10 which is the IP of the 1st proxy server, not the IP of the service server (192.168.100.51), and the protocol is TCP. In the communication between the 1st proxy server and the 2nd proxy server, the source IP is 192.168.100.10, which is the IP of the 1st proxy server, and the target IP is 192.168.100.11, which is the IP of the 2nd proxy server. It can be confirmed that the protocol is IPIP. The IP address of the client is 192.168.100.50, the IP of the service server is 192.168.100.51, and the protocol is TCP. 'IPIP' means the Outer IP header in the IP-in-IP format IP packet, and Inner IP protocol has the same protocol as TCP, which is the protocol of the packet that the client sends to the 1st proxy server. Finally, when the communication between the 2nd proxy server and the service server is checked, it can be confirmed that the Inner IP packet among the IP packets used in the communication between the 1st proxy server and the 2nd proxy server is used as it is. That is, it can be seen that the service server is supposed to recognize that it receives data from the client (192.168.100.50) rather than from the 2nd proxy server (192.168.100.10) or the 1st proxy server (192.168.100.10).

Table 1: Result of checking packet in path

Path	**Wireshark Data**
Client → The 1st Proxy Server	Internet Protocol Version 4, Src: 192.168.100.50, Dst: 192.168.100.10 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 55 Identification: 0x1173 (4407) Flags: 0x02 (Don't Fragment) Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum: 0x0000 [validation disabled] [Header checksum status: Unverified] Source: 192.168.100.50 Destination: 192.168.100.10 [Source GeoIP: Unknown] [Destination GeoIP: Unknown]
The 1st Proxy Server → The 2nd Proxy Server	Internet Protocol Version 4, Src: 192.168.100.10, Dst: 182.151.100.11 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 75 Identification: 0x1179 (4413) Flags: 0x02 (Don't Fragment) Fragment offset: 0 Time to live: 128 Protocol: IPIP (4) Header checksum: 0x0000 [validation disabled] [Header checksum status: Unverified] Source: 192.168.100.10 Destination: 192.168.100.11 [Source GeoIP: Unknown] [Destination GeoIP: Unknown] ~
The 2nd Proxy Server → Service Server	Internet Protocol Version 4, Src: 192.168.100.50, Dst: 192.168.100.51 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 55 Identification: 0x1512 (5394) Flags: 0x02 (Don't Fragment) Fragment offset: 0 Time to live: 128 Protocol: TCP (6) Header checksum: 0x0000 [validation disabled] [Header checksum status: Unverified] Source: 192.168.100.50 Destination: 192.168.100.51 [Source GeoIP: Unknown] [Destination GeoIP: Unknown]

CONCLUSION

Through this study, it is possible to prevent the adverse effect of the service of the entire Healthcare System by preventing the direct attack of the hacker against the service server by concealing information of the service server from the client. In addition, the service server normally obtains the information of the client. Therefore, it is possible to identify the problem client and to take measures for the client, so that the administrator can take effective measures against the hacking in the system as a whole. This information can be used as information to prepare. We can contribute to more diverse systems through the results of this study because it can be acquired the more various information, just hiding information and path control of service server through relay system

Financial support: This research was supported by Bio & Medical Technology Development Program (Research on demonstration of prognostic and diagnostic systems for aging related diseases), through the National Research Foundation of Korea (NRF), funded by the Ministry of Science and ICT (2016M3A9B6904246).

REFERENCES

[1] Da Xu, Li, Wu He, and Shancang Li. "Internet of things in industries: A survey." IEEE Transactions on industrial informatics 10.4 (2014): 2233-2243.

[2] Sajid, Anam, Haider Abbas, and Kashif Saleem. "Cloud-assisted IoT-based SCADA systems security: A review of the state of the art and future challenges." IEEE Access 4 (2016): 1375-1384.

[3] Lee, In, and Kyoochun Lee. "The Internet of Things (IoT): Applications, investments, and challenges for enterprises." Business Horizons 58.4 (2015): 431-440.

[4] Sicari, Sabrina, et al. "Security, privacy and trust in Internet of Things: The road ahead." Computer Networks 76 (2015): 146-164.

[5] System and Method for servicing domain name based on user information (Korean Patent Application Number: 10-1345372)

[6] Relay System for Transmitting IP Address of Client to Server and Method Therefor (PCT/KR2014/000404)

[7] Relaying System and Method for Transmitting IP Address of Client to Server Using Encapsulation Protocol (PCT/KR2014/005130)