HTTPS Hacking Protection

Thawatchai Chomsiri
Faculty of Informatics, Mahasarakham University,
Mahasarakham 44150, Thailand.
thawatchai@msu.ac.th

Abstract

In general, E-Commerce sites utilize SSL to ward off the authorized detection and decoding of confidential data over a network. In most cases, the communication between Web Browser and E-Commerce Web Server is uses HTTPS protocol. However, the communication often induces some drawbacks, simply denoted by hole. This, in addition, furnishes an opportunity for a hacker to manipulate the data, i.e. decoding the data, using SSL-MITM (SSL Man in the Middle) technique. According to the trials in an experiment with Auditor Security Collection, the results illustrate a hacker and a victim who are on the same local area network; the hacker could be able to decode confidential data (password or credit card number) with the possibility of more than 50 %. This paper presents 3 different methodologies to prevent the decoding using SSL-MITM on the confidential data which normally traverses over e-commerce websites. In addition, the evaluation of 3 schemes is conducted to show the degrees of efficiency of the techniques. Furthermore, this information can be preliminarily utilized as a factor to increase the security of e-commerce website.

1. Introduction

Since the internet reach cover to the world trends of business has been gently changed. Clients have more alternative ways to purchase the products. With e-Commerce, clients could be able to buy some products via e-commerce websites. As a result, they might have to provide some confidential data to purchase an item. In addition, Internet is everywhere. It is easily accessed. Therefore, the security is needed to take into an account. There are some literatures trying to develop and study a mechanism to provide secured information on the Internet, for example, SSL[1], VPN[2] and HTTPS [3][4] etc.

HTTPS was first introduced to be used as a secured communication channel, rather than normally HTTP protocol. It, in addition, provides a reliability communication over the internet in term of security issue by protecting your information to be seized. As a result, a great number of e-commerce sites [5] run their business using the protocol. However, one major drawback found in HTTPS is that it cannot tolerate to unauthorized access using SSL Man in the Middle technique [6]. This leads to security issue when the confidential information of the customers is hacked [7]. According to the preliminary experiment in the lab with low degree of congestion and a proper manner of data capturing, it indicates that the information on HTTPS has as high as 100% [8] of possibility to be recorded and decoded. In addition the experiment was also conducted in normal environment. The results show that the confidential data can be accessed by the intruders (password or credit card number) with the possibility of more than 50 % [8].

There are many research about protecting secure data, such as [9] and [10] focused on Wi-Fi Networks and human factors, [11] secure the online retrieval of certificates by confirming a certificate's fingerprint out-of-band. Boneh D. and Franklin M. [12] use Identity-based cryptographic algorithms which do not require certificates. They use as a party's public key the party's identifier (e.g.,fully qualified domain name or IP address). Ackerman and Cranor [13] have proposed critics that help Web browser users understand and negotiate privacy issues that may be involved in visiting certain sites.

This paper addresses the problem of the drawback of HTTPS and proposes 3 techniques to recover the drawback of the protocol. In addition, the imperial data testing is provides to choose the efficiency of the techniques.

21st International Conference on
Advanced Information Networking and Applications Workshops (AINAW'07)
0-7695-2847-3/07 $20.00 © 2007 IEEE

2. HTTPS Background

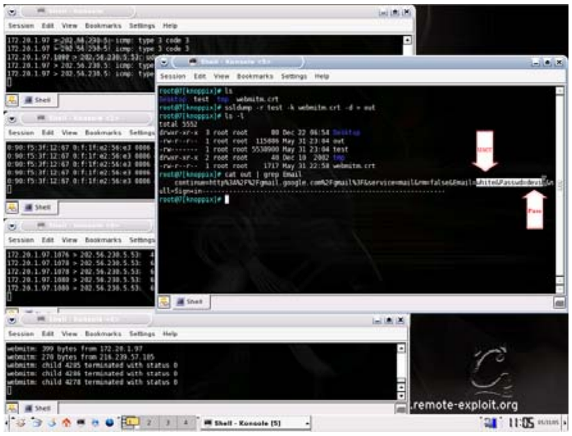
The communication is triggered when a client send a request to the server by specifying an URL on HTTPS protocol using port number 443 [3]. The web server, providing a service for HTTPS, responds the client by sending the certificate [3] to the client side. By this, web browser signifies a public key of the web server, which packed in the certificate. The key is used to encode the information that the client send consecutively to the web server. Technically, the initial information that the client sends to the web server is a session key, which would be utilized for further data transmission between the client and the web server. Consequently, web server uses its private key to decode the information (session key) transmitted by the client. As a consequence, only either the web server or the client understands the session key and that the further transmission is secured.

3. SSL MITM Background

Decoding HTTPS using SSL Man in the middle [6] has following step:

1. Notifying a gateway-router that hacker-machine is victim-machine.
2. Notifying the victim-machine that the hacker-machine is gateway-router.
3. Enable packet routing feature on hacker-machine.
4. Running DNS Spoof [14] to enforce the victim to connects to HTTP/HTTPS port at hacker machine.
5. Distributing fake certificate to the victim.
6. Communicating with the victim using fake certificate
7. Communicating with the HTTPS web-site using genuine certificate obtained from the HTTPS web-site.
8. Transmitting the parameters and data between the victim and the HTTPS web-site.
9. Recording data transferred between two end-hosts.
10. Decoding data.

In practice, the above steps are not necessary in order. It can be processed in arbitrarily sequence; however, some is required to be processed in earlier step. For example, once the hacker precedes step 1 and 2 consequently, packet could not be delivered unless step 3 is started. Considering step 2 and 3, it is generally possible that delay could be found between these two steps. This delay, in turn, can be determined as a crucial indicator to the victim noticing an irregular behavior of the system. Therefore, the sequence of the step is needed to take into a consideration, a proper order is provided in the next section. Given a scenario that we are trying to capture and decode Gmail's password of a user, we first initial IP Address for victim is 192.168.1.9, for gateway-router is 192.168.1.1 and for hacker is 192.168.1.55. A MAC Address for hacker is AA-BB-CC-DD-EE-FF (a hacker using single Network Interface Card). Auditor Security Collection, hacking tool, is utilized. The tool is downloadable at www.remote-exploit.org, a bootable CD running on Linux platform.



```
172.20.1.97 > 202.56.230.5: icmp: type 3 code 3
172.20.1.97 > 202.56.230.5: icmp: type 3 code 3
172.20.1.97.1099 > 202.56.230.5.53:
172.20.1.97 > 202.56.230.5: icmp: type
172.20.1.97 > 202.56.230.5: icmp: type
```

```
root@7[knoppix]# ls
Desktop  test  tmp  webmitm.crt
root@7[knoppix]# ssldump -r test -k webmitm.crt -d > out
root@7[knoppix]# ls -l
total 5552
drwxr-xr-x  3 root root      80 Dec 22 06:54 Desktop
-rw-r--r--  1 root root  115606 May 31 23:04 out
-rw-------  1 root root 5530900 May 31 23:04 test
drwxr-xr-x  2 root root      40 Dec 10  2002 tmp
-rw-r--r--  1 root root    1717 May 31 22:58 webmitm.crt
root@7[knoppix]# cat out | grep Email
    continue=http%3A%2F%2Fgmail.google.com%2Fgmail%3F&service=mail&rm=false&Email=whited&Passwd=devil&n
ull=Sign+in--------------------------------------------------------------
root@7[knoppix]#
```

```
0:90:f5:3f:12:67 0:f:1f:e2:56:e3 0806
0:90:f5:3f:12:67 0:f:1f:e2:56:e3 0806
0:90:f5:3f:12:67 0:f:1f:e2:56:e3 0806
0:90:f5:3f:12:67 0:f:1f:e2:56:e3 0806
0:90:f5:3f:12:67 0:f:1f:e2:56:e3 0806
```

```
172.20.1.97.1076 > 202.56.230.5.53:
172.20.1.97.1078 > 202.56.230.5.53:
172.20.1.97.1078 > 202.56.230.5.53:
172.20.1.97.1080 > 202.56.230.5.53:
172.20.1.97.1080 > 202.56.230.5.53:
```

```
webmitm: 399 bytes from 172.20.1.97
webmitm: 270 bytes from 216.239.57.105
webmitm: child 4285 terminated with status 0
webmitm: child 4286 terminated with status 0
webmitm: child 4278 terminated with status 0
```

Figure 1. HTTPS Decoding Tools from www.remote-exploit.org

The procedure starts by enabling packet routing at hacker side (step 3). In practice, there are many techniques to enable packet routing service, for instance, using command echo 1 > /proc/sys/net/ipv4/ip_forward or using fragrouter [6] program following parameter B1 (i.e. #fragrouter –B1). As a result, the hacker notifies the victim that the hacker is a Gateway Router using ARP Spoof [15] technique:

```
# arpspoof –t  192.168.1.9  192.168.1
```

This command signals ARP information to the victim side (192.168.1.9) that the Gateway Router (192.168.1.1) has specific MAC Address which is AA-BB-CC-DD-EE-FF. All frames of data are, therefore, transferred from the victim to the hacker machine. On the other hand, the hacker has to notify a Gateway Router as it is a victim by using the following command:

```
# arpspoof  –t  192.168.1.1  192.168.9
```

At this point, all data frames sending from the victim to the Gateway Router is also delivered to the hacker machine.

The webmitm [6] is software distributing an imitated certification to a victim and opening HTTP/HTTPS port. However, the connection between a victim and a hacker has to be preliminarily established. The hacker, as a result, uses dnsspoof (software) to alter the direction of primary connection:

From Victim → www.gmail.com

To Victim → Hacker → www.gmail.com

Usage

```
# dnsspoof
# webmitm   -d
```

By this, all HTTP/HTTPS packet transmitting between the victim and the HTTPS web-site (www.gmail.com) are accordingly pass through the hacker machine. Once the victim requests to https://www.gmail.com in order to send the information on secure connection, Gmail generate a certificate. The hacker, consequently, captures the certificate and sends to another with new fake certificate (which include hacker's public key). Every time the victim logins using user name and password, the information is actually stopped by at the hacker machine, according to Man in the Middle analogy. The hacker is able to decode the information by using his private key. In addition, the hacker encodes the information before transmitting it to Gmail on HTTPS using the original certificate.

Hacker uses Ethereal [16] (or uses Sniffer) to capture the information, especially the information sent by the victim. The application is run certainly once it is command by webmitm –d. In order to abate missed-capturing, the Ethereal will take a certain of time so as to it is convinced that the victim has completely sent username and password. The capturing is halted and the captured information is stored, and saves as a filename "test". Decoding the data is, as a consequence, proceeded by ssldump with some parameter as follows:

```
# ssldump   –r   test   –k   webmitm.crt   >   out
```

The decoded data is, by the above command, save in filename "out". That is, hacker is able to read the data (pain text) sent by the victim.

4. Protection Scheme

This paper propose 3 defender-techniques to protect your confidential data to be possibly decoded by an unauthorized intruder (hacker) according the previous scenario. The first technique uses Static ARP on the switching. Whilst the second technique; relies on using ARP Watch software to monitor the matching between IP Address and MAC Address. Blocking is trigged when the software notices (some irregular behaviors occur). The last technique utilizes Anti-Sniff scanning Network Interface Card that is currently running promiscuous mode, i.e. running a program capturing the data. If it is found, the connection is blocked. There are more than 3 techniques to protect the data in such the circumstance; however, those techniques are difficult to implement in practice. For example, setting a policy to the clients (user) to view a certificate every time if their connecting to HTTPS web-site to monitor the abnormality on the certificate. Another technique is using DNSSEC [17] and VPN at the client site. However, this technique is complicated and high-priced. Therefore, the techniques proposed in this paper are the simple technique that can be implemented in practice for. A general detail is provided in the next section.

4.1. Use Static ARP

According to the decoding technique by Man in The Middle, ARP Spoof is encouraged to process in order to hoodwink both victim and gateway-router, previously explain. Therefore, it is another solution to interrupt ARP Spoof so as to it could not finish its task or halts immediately. "Port Secure" is a feature provided on the switch. Generally, Port Secure is used to configure and set ARP Table to static item. As a result, the values which are in the table could not be modified.

The advantage of this technique is that the administrator is configuring switch only, no necessary to configure client unless new client added to network. But disadvantage is that some model of switch no have "Port Secure" feature.

4.2. Use ARP Watch

Once a hacker processes ARP Spoof, the values in the ARP Table on a gateway router and the victim-machine is modified. Therefore, monitoring the values over the period of time is another technique indicating if the information is hacked by SSL Man in The Middle.

ARP Watch is a program detecting the abnormality in ARP table. Setting up ARP Watch at victim side, it would certainly alert when ARP Spoof is running. It, in addition, gives a warning to the administrator once ARP Spoof is running after installing it on the gateway router. Specifically, it will tell the administrator an IP Address that is running ARP Spoof. However, ARP Watch would not well behave if a client changes a Network Interface Card.

The advantage of this technique is that the alert can be used to trace a hacker (e.g. behavior) and record it as an evident. However, there are some disadvantages of using this technique which are as fellows:

- There is incompatible support between the software and device, for example, Cisco Router and other box-shape router. Whilst, there are a small number of routers supporting the software, for instance, Windows NT/2000/2003 Router and Linux Router

- If the software could not be installed at gateway side, the software would be set up on many clients. This situation increases a load for user to monitor the connection, and

- It could possibly give unreliable information if the Network Interface Card is changes, but use the same IP Address still.

4.3. Use Anti Sniff

Decoding HTTPS can be accomplished by running ARP Spoof and data capturing. Thus, a hacker needs to use some specific software (for example Sniffer and Ethereal). We can, by this, detect a machine that is running software to capture the information by checking if a machine is working on Promiscuous mode.

Anti-Sniff is a generative application detecting a machine that is running in Promiscuous mode. The program can be installed on any one-machine in the network. The program, again, indicates some IP Address that is currently run data-capturing software. By this, the administrator can block the IP Address consequently. This technique is flexible since the administrator dose not need to be in one particular machine to scan the machine. However, the drawback is that scanning and monitoring must be run all time.

Remark1. Because of this hacking method use port number 443, thus firewall [18] cannot resolve this problem.

5. Evaluation

The evaluation is conducted to illustrate the performance of 3 techniques (which technique gives the best performance and once a number of clients are increased, how well these 3 techniques behave). In addition, the training on security issue is conducted with 20 students, who are the tester for the evaluation. Moreover the evaluation is conducted with 12 computer rooms, which room1 – room3 have 25 clients, room4 – room6 have 50 clients, room7 – room9 have 75 clients, and room10 – room12 have 100 clients. The administrators of each room use 3 techniques to protect a hacker (Static-ARP, ARP-Watch and Anti-Sniff). Each technique runs 10 days. The hackers, 20 students, try to hack using SSL-MITM method. Consequently, the result is record whether or not each hacker is able to decode the information, which is illustrated in Figure 2.

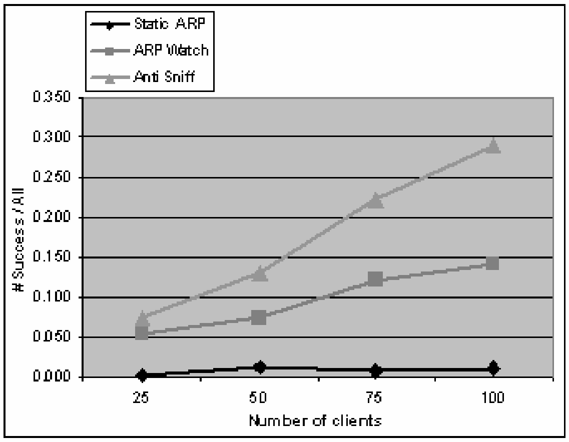


Figure 2. Experimental result.

According to the graph, x-axis demonstrates a number of clients in the internet-service room, which are 25, 50, 75 and 100 consequently. Whilst y-axis; is a ratio of hacked-success-count and hacked-count (Success/All). We use average values from rooms which number of clients are equals, and then plot to the graph. The result shows that detecting the information using static ARP in the same room gives the lowest rates, which imply that static ARP gives the best performance in the situation. On the other hand using Anti-Sniff shows the worst performance. What is behind the scene is that using the software the administrator has to watchfully monitor, which is cumbersome.

21st International Conference on
Advanced Information Networking and Applications Workshops (AINAW'07)
0-7695-2847-3/07 $20.00 © 2007 IEEE

According to the technique Static ARP is able to give 100% of protecting; however, the result insists the possibility that hacker can decode that data. This phenomenon is cautiously checked up and found that there are some machines substituted during the evaluation. The administrator needs to disable and enable (later) Port Security function in order to configure Static ARP. Therefore, Static ARP is not able to protect the connection at that time.

6. Conclusion

Although, e-commerce site has preliminary protection on the information using HTTPS, hacker is able to capture and decode the confidential information by using SSL Man in The Middle. The hacker has to hook up to the same Local Area Network with a victim. There are some tools that the hacker uses to capture/decode the information, for instance, ARP Spoof, DNS Spoof, Sniffing and SSL Dump. The solution brought to this situation is to interrupt SSL Man In The Middle while it is processing, especially interrupting ARP Spoof. The interruption can be encouraged by using Static ARP at the switch and use ARP Watch to alert the administrator. More to the point, Anti-Sniff provides a function to scan a machine that is capturing information. This paper, as a result, proposes these 3 techniques with the evaluation illustrating the performance of the techniques and we found that Static ARP on switch gives the best performance.

7. References

[1] A. Freier, P Karlton, and P. Kocher, “The SSL Protocol, version 3.0, Internet Draft”, March 1996.
< http://wp.netscape.com/eng/ssl3/ssl-toc.html >

[2] “A Framework for IP Based Virtual Private Networks”, RFC 2764, <http://www.apps.ietf.org/rfc/rfc2764.html>

[3] Keith W. Ross, James F. Kurrose., Addison Wesley Longman, “Computer Networking: a top-down approach featuring the Internet”, Inc., USA, 2004.

[4] Behrouz A. Forouzan, McGraw-Hill Companies, “Data Communications and Networking third edition”, Inc, New-York USA, 2004.

[5] Stuart McClure, Saumil Shah, and Shreeraj Shah, “Web Hacking: Attacks and Defense”, Addison-Wesley, September 2002.

[6] http://www.remote-exploit.org,
<http://www.remote-exploit.org/index.php/Auditor_mirrors>

[7] Stuart McClure, Joel Scambray, and George Kurtz, “Hacking Exposed 5th Edition”, ISBN: 0072260815, April 19, 2005.

[8] Thawatchai Chomsiri, and Preecha Noiumkar, “Decoding HTTPS secure data on LAN”, Draft Technical Report, Faculty of Informatics, Mahasarakham University, Mahasarakham Thailand, January 2006.

[9] Xia H., and Brustoloni J., “Detecting and Blocking Unauthorized Access in Wi-Fi Networks”, In Proc. Networking’2004, IFIP, Lecture Notes in Computer Science, 3042:795-806, Springer-Verlag, May 2004.
<http://www.cs.pitt.edu/˜jcb/papers/net2004.pdf>

[10] Xia H., and Brustoloni J., “Virtual Prepaid Tokens for Wi-Fi Hotspot Access”, In Proc. 29th Intl. Conf. Local Computer Networks (LCN), IEEE, Nov. 2004, pp. 232-239.

[11] Perrin T., “Public Key Distribution Through CryptoIDs”, In Proc. Workshop on New Security Paradigms, ACM, 2003, pp. 87-102.

[12] Boneh D., and Franklin M., “Identity Based Encryption from the Weil Pairing”, *Journal of Computing*, SIAM, 32(3):586-615, 2003.

[13] Ackerman M., and Cranor L., “Privacy Critics: UI Components to Safeguard Users’ Privacy”, In Proc. Conf. Human Factors in Computing Systems (CHI’99), Extended Abstracts, ACM, 1999, pp. 258-259.

[14] Song, D. dsniff.
<http://naughty.monkey.org/˜dugsong/dsniff/>

[15] ARP Spoof.
<http://www.oxid.it/downloads/apr-intro.swf>

[16] Angela D. Orebaugh, and Gilbert Ramirez, “Ethereal Packet Sniffing”, Syngress Publishing, February 2004.

[17] J. Galvin, “Public Key Distribution with Secure DNS”, In Proc. Sixth USENIX Security Symposium, 1996.

[18] Thawatchai Chomsiri, and Chotipat Pornavalai, “Firewall Rules Analysis”, Proc. of The 2006 International Conference on Security & Management (SAM'06), June 2006, Las Vegas, Nevada, USA.