Switch’s CAM Table Poisoning Attack: Hands-on Lab Exercises for Network Security Education

Zouheir Trabelsi
Faculty of Information Technology
UAE University
Al-Ain, UAE

trabelsi@uaeu.ac.ae

Abstract
Teaching offensive techniques is a necessary component of a computer security education and yields better security professionals than teaching defensive techniques alone. In this paper, we describe a case study of the implementation of comprehensive hands-on lab exercises that are essential to security education. The first hands-on lab exercise is about how to perform a Denial of Service (DoS) attack based on the poisoning of the CAM tables (Content Access Memory) of Local Area Network (LAN) switches. The second exercise is about how to prevent CAM table poisoning attack. The hands-on labs confirmed further the ethical and legal concerns regarding the teaching of offensive techniques in the academic environment. In fact, the number of injected malicious traffic targeting the university switches’ CAM tables, increased considerably each time the students experiment the DoS attack. That is why every course in IT security should be accompanied by a basic discussion of legal implications and ethics.

Keywords: Switch CAM table poisoning, DoS attack, Security port.

1 Introduction
Network security courses are often taught as concepts, at relatively abstract levels. A curriculum that covers the concepts of network security without giving suitable coverage to practical implementation deprives the student of the opportunity to experience the technologies and techniques required to ensure security. A hands-on approach to disseminating knowledge of network security will prepare the student for the complexities of conducting research and development in this field. Such an approach is rarely seen in most graduate and undergraduate courses. Even when the hands-on approach is advocated, by some, it is usually dominated by exercises using defensive techniques.

Recently, offensive techniques, originally developed by hackers, are gaining widespread approval and interest. Offensive techniques yield better security professionals than those taught only defensive techniques (Mink and Freiling 2006, Arce and McGraw 2004, Arnett and Schmidt 2005, Dornseif, Holz, and Mink 2005, Vigna 2003, Yuan, Matthews, Wright, Xu, and Yu 2010, Livermore 2007). It is important to note that the corporate businesses employ experts that use offensive techniques for penetration testing, to ensure their security. The use of offensive techniques to provide secure environments for large corporate entities has created the new genre of “ethical hackers”. Many educators in this field feel that offensive methods should not be taught to students as this only increases the population of “malicious hackers”. Many educators in this field feel that hands-on courses that teach security attacks in detail are unethical, and create the potential for some to use the tools and techniques in an irresponsible manner (Harris 2004, Caltagirone, Ortman, Melton, Manz, King, and Oman 2006, Livermore 2007). The social implication is to restrict the injection of new hackers into society.

However, others claim that teaching offensive techniques yields better security professionals than those that are taught only defensive techniques (Mink and Freiling 2006, Arce and McGraw 2004, Arnett and Schmidt 2005, Dornseif, Holz, and Mink 2005, Vigna 2003, Yuan, Matthews, Wright, Xu, and Yu 2010, Livermore 2007). It is important to note that the corporate businesses employ experts that use offensive techniques for penetration testing, to ensure their security. The use of offensive techniques to provide secure environments for large corporate entities has created the new genre of hackers, the “ethical hacker”!

We believe that offensive techniques are central, to better understand security breaches and system failures. Teaching network attacks with hands-on experiments is a necessary component of education in network security. Moreover, we believe that security students need to experiment attack techniques to be able to implement appropriate and efficient security solutions. This approach to education will enable the student to provide confidentiality, integrity, and availability for computer systems, networks, resources, and data. One cannot perfectly design or build defenses for attacks that one has not truly experienced, first-hand. However, we agree that offensive techniques must not be taught as the primary focus of a course. Every course in IT security must be accompanied by discussion of legal implications and cover the ethical responsibilities of the student towards their community and society at large.

Network security lacks sufficient and contemporary textbooks and technical papers that describe in detail hands-on exercises that include both offensive and defensive implementation within an isolated network laboratory environment. To reduce the learning of this void in security education, this paper proposes comprehensive hands-on lab exercises that are essential to security education. The first lab exercise is about how to perform...
a DoS attack based on the poisoning of the CAM table of a LAN switch. The second lab exercise is about how to prevent the poisoning of the switch’s CAM table. The lab exercises allow students to better analyze and improve the discussed offensive and defensive techniques. The lab exercises can be offered to students during security courses related to intrusion detection and prevention techniques, particularly to DoS attacks. It is designed to accompany and complement any existing trade or academic press text.

The paper is organized as follows: Section 2 includes a brief understanding of switch’s CAM table, to form a base for subsequent sections. Section 3 discusses the first hands-on lab. Section 4 discusses the second hands-on lab. Section 5 discusses some ethical concerns related to teaching offensive techniques. Section 6 discusses the student satisfaction and the effect of offering the hands-on lab exercises on the student performance. Finally, Section 7 concludes the paper.

2 Background: Switch’s CAM Table

To form the base for subsequent sections, this section includes a brief understanding of the switch’s CAM table. LAN’s switches maintain a table called the CAM table, and maps individual MAC addresses on the network to the physical ports on the switch. This allows the switch to direct data out of the physical port where the recipient is located, as opposed to indiscriminately broadcasting the data out of all ports as a hub does. The advantage of this method is that data is bridged exclusively to the network segment containing the computer that the data is specifically destined for.

Figure 1 shows an example of entries in the CAM table of a switch. Four hosts are connected to the switch. For example, the first host (whose MAC address is 00:0F:1F:C0:EB:49) is connected to Port #1 (Interface: FastEthernet0/1) on the switch.

![Figure 1: The entries of a CAM table](image)

When the switch receives a packet from a host, it extracts first the destination MAC address from the header of the Ethernet frame. Using this MAC address, the switch gets the corresponding port number from the CAM table. Then, the packet is sent only to the host connected to that port. Therefore, even by setting a computer’s network interface card (NIC) into the promiscuous mode, sniffing traffic in a switched LAN network is not possible. However, hackers use the Man-in-the-Middle (MiM) attack technique to intercept and sniff traffic in switched LAN network (SwitchSniffer 2011, Winarp 2011, and WinArpAttacker 2011).

3 Lab exercise: DoS attack based on CAM Table Poisoning

This hands-on lab exercise is about DoS attack using CAM table poisoning technique. The learning objective of this lab exercise is for students to learn how to poison the CAM table of a LAN’s switch in order to perform DoS attack on target LAN’s hosts.

3.1 Attack Description

This attack intends to corrupt the entries in the switch’s CAM table, so that the network traffic will be redirected. That is, a malicious host (connected to Port #a in a switch), sends a fake packet, with the source MAC address in the packet’s Ethernet header set to the MAC address of a target host (connected to Port #b). The destination MAC address in the packet’s Ethernet header can be any address. Once the switch receives the packet, it updates its CAM table. Therefore, the CAM table’s entry for that target host’s MAC address will be corrupted. Hence, the target host will be considered as a host connected to Port #a. Any packet sent to the target host (destination MAC address in the packet’s Ethernet header is equal to the target host’s MAC address) will be forwarded to Port #a; that is, to the malicious host.

As example of CAM table poisoning attack, Figure 1 shows that in the CAM table of a switch, there are four hosts connected to the switch. Host #1, the malicious host, attacks the switch’s CAM table using 3 fake packets. The three packets are almost the same, but they have different source MAC addresses in the Ethernet headers. The information of the packets is as follows:

1. **First fake packet**: Source MAC address in the Ethernet header = 00:08:74:04:BC:4A (Host #2).
2. **Second fake packet**: Source MAC address in the Ethernet header = 00:08:74:05:AD:20 (Host #3).
3. **Third fake packet**: Source MAC address in the Ethernet header = 00:03:0D:38:79:57 (Host #4).

After this attack, the switch’s CAM table becomes corrupted, as shown in Figure 2. The CAM table shows that all four hosts are connected to the switch’s Port#1 (FastEthernet 0/1). However, physically only Host#1 is connected to Port#1.

![Figure 2: The content of the CAM table after the CAM table poisoning attack](image)

Once a packet is sent to one of these three hosts (Host#2, Host#3 and Host#4), the switch will forward it to Port#1; that is, to Host#1. This situation may create a DoS situation, since the switch is not forwarding the packets, issued from these three hosts, to their destinations (Figure 3).
3.2 Experiment

The following experiment describes how to poison the CAM table of a target switch. A simple network is used in the experiment. Three Windows XP based hosts are connected to a switch and each host is assigned a static IP address, as shown in Figure 4. The experiments discussed here use a switch device from a leader in the market namely Cisco, but the knowledge can be easily adapted to any other available switches with similar security features, such as Juniper switches.

Step 1: View the CAM table contents

To view the CAM table contents of a switch, simply perform the following steps:

- Connect a LAN’s host to the console port on the switch.
- Run the Terminal Application program (For example: HyperTerminal) in the host.
- Under “Connect Using:” option, select one of the appropriate communication port (COM1, COM2, etc.) that the console cable is attached.
- Select OK and a “Port Settings” window will pop-up prompting you to define the data rate and communication setting as defined by the vendor. (Most vendors have the following settings: 9600-Bits per second, 8-Data Bits, None-Parity, 1-Stop bits and None-Flow control.)
- Select OK. This will place you in the Terminal Window.
- Depress the “Enter” key a few times until a menu from the switch appears in the Terminal Window.
- If the menu appears, then you are ready to configure the switch as needed.
- In case of a Cisco switch (Cisco 2011), type the following command to view the contents of the CAM table:

```
Switch>enable /enter the enable command to access privileged EXEC mode.
Switch# show mac-address-table
```

- The CAM table content is:

This screen shows that three hosts, whose MAC addresses are displayed, are connected on Port #2, Port #4, and Port #6, respectively.

3.2.2 Step 2: Poison the CAM table contents

We assume that Host A wants to poison the switch’s CAM table, by inserting the invalid entry: MAC address of Host B  Switch’s Port 0/2 (Fa0/2). This invalid entry will tell the switch that Host B is now located at Port 0/2 (Fa0/2). However, physically, Host B is still located at Port 0/4 (Fa0/4). Hence to perform this attack, Host A should send to any destination host in the LAN network a fake packet (IP or ARP packet) whose Ethernet source MAC address is equal to the MAC address of Host B:

<table>
<thead>
<tr>
<th>IP or ARP packet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet header:</td>
</tr>
<tr>
<td>- Source MAC address</td>
</tr>
<tr>
<td>- Destination MAC address</td>
</tr>
</tbody>
</table>

Using any packet builder tool, such as CommView Packet Builder or Engage Packet Builder, the above fake packet can be easily built. In this lab exercise, we use CommView Packet Builder, since it provides a very friendly GUI interface to build IP, TCP, ICMP, UDP and ARP packets. For example, the following screenshot shows that a fake ICMP echo packet, whose MAC source is equal to the MAC address of Host B, is built at Host A:
After sending the fake ICMP echo packet, the CAM table becomes corrupted, as follows:

The above screenshot shows that Host B is connected on Port #2. However, physically, Host B is still connected on Port #4. Consequently, when a host in the LAN network sends packets to Host B, the switch will not forward them to Host B; in contrast they will be forward to Host A. This is a DoS attack, since the LAN network’s hosts are not able to communicate properly with Host B.

3.3 MAC Flood Attack

An old attack technique for sniffing traffic in a switched LAN network is based on MAC flooding. MAC flooding is a technique employed to compromise the security of network switches. In a typical MAC flooding attack, a switch is flooded with many Ethernet frames, each containing different source MAC addresses, by the attacker. The intention is to consume the limited memory set aside in the switch to store the MAC address table. That is, some CAM tables of old switch models may be overflowed and revert to broadcast mode (hub mode known also as the ‘fail open mode’) as a consequence after which sniffing can be easily performed. After launching a successful MAC flooding attack, a malicious user could then use a packet analyser (a sniffer) to capture sensitive data being transmitted between other computers, which would not be accessible when the switch operates normally.

4 Lab exercise: Prevention of CAM Table Poisoning

This lab exercise is about preventing the poisoning of the switch’s CAM table. The learning objective of this lab exercise is for students to learn how to protect switches from CAM table poisoning attack.

To prevent CAM table poisoning, security administrators usually rely on the presence of one or more features in their switches. With a feature often called "port security" by vendors, many advanced switches can be configured to limit the number of MAC addresses that can be learned on ports connected to end stations. A smaller table of "secure" MAC addresses is maintained in addition to (and as a subset to) the traditional CAM table.

For example, Cisco Catalyst 3560 Series switches (Cisco 2011) allow to restrict the number of legitimate MAC addresses on a port (or an interface) using the port security feature. When that number is exceeded, a security violation would be triggered and a violation action would be performed based on the mode configured on that port. Therefore, any unauthorized MAC addresses would be prevented from accessing and corrupting the CAM table.

A switch’s port can be configured for one of three violation modes, based on the action to be taken if a violation occurs:

- **Protect**—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until the switch administrator removes a sufficient number of secure MAC addresses. The switch administrator is not notified that a security violation has occurred.
- **Restrict**—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until the switch administrator removes a sufficient number of secure MAC addresses. In this mode, the switch administrator is notified that a security violation has occurred.
- **Shutdown**—A port security violation causes the interface to shut down immediately. When a secure port is in the error-disabled state, the switch administrator can bring it out of this state by entering the **err disable recovery cause psecure_violation** global configuration command or he can manually re-enable it by entering the shutdown and no shutdown interface configuration commands. This is the default mode. The switch administrator can also customize the time to recover from this state.

4.1 Experiment

The following experiment describes how to configure and test the port security feature in Cisco Catalyst 3560 Series switches to prevent the poisoning of the CAM table. The experiment uses the same network architecture described in the previous lab, and consists of the following steps:

- **Step 1**: Configure the Restrict Mode Security Port in the switch.
- **Step 2**: Generate a malicious packet to poison the CAM table.
• Step 3: Configure the Shutdown Mode Security Port in the switch.

4.1.1 Step 1: Configure the Restrict Mode Security Port in the switch

The following steps allow configuring the Restrict Mode Security Port:

- Connect a host to the console port on the switch
- Run the Terminal Application program in the host
- Type the following commands:

  Switch>enable //enter the enable command to access privileged EXEC mode
  Switch# Configure terminal
  Switch(config)# interface fastethernet 0/2 // port security feature is applied on the host connected on Port #2
  Switch(config-if)# switchport mode access
  Switch(config-if)# switchport port-security
  Switch(config-if)# switchport port-security violation restrict
  Switch(config-if)# end
  Switch# copy running-config startup-config

- To display the port security mode, type the following command:

  Switch# show port-security

  The following results will appear:

  This screenshot shows clearly that there have been 27 packets that attempted to violate the security feature implemented on Port #2. These packets attempted to corrupt the CAM table; however, the switch has blocked them.

4.1.2 Step 2: Generate a malicious packet to poison the CAM table

Use any packet generator tool to generate a malicious packet whose MAC source in the Ethernet frame is equal to a fake MAC address. For example, we use the same fake ICMP echo packet generated in the previous hands-on lab exercise.

- Type the following command to view the CAM table contents after the poisoning attempt:

  Switch# show mac-address-table

  This screenshot shows clearly that the CAM table has not been corrupted.

- Display again the port security mode:

  This screen shows that there has been a packet that attempted to violate the security feature implemented on Port #2. The switch has blocked the malicious packet and shut down the port.

  The following screen shows clearly that Host A has lost its connection to the switch (Interface Fa0/2 has been shutdown), and a warning message appeared on Host A’s desktop, as follow:

5 Ethical Concern

The hands-on lab exercises have been used in our intrusion detection and response course in the last three years. A major ethical concern has been identified when
analysing the number of malicious IP and ARP packets injected in the university network.

We used the intrusion detection sensors installed in the network segments to collect malicious packets and detect potential attack traffic. Figure 5 shows that the total average number of malicious packets targeting the university switches’ CAM tables over the three years increased during the days following the hands-on lab exercises practice. This is a dilemma when offering hands-on lab exercises about offensive techniques.

Figure 5: Evolution of the number of detected malicious IP and ARP packets targeting the switches’ CAM tables

On the other hand, a survey showed that most of the students said that they have experiment the DoS attack using the CAM table poisoning technique, outside the university isolated network laboratory environment, particularly at their home’s networks. The victims were mainly their sisters and brothers’ computers. They used the DoS attack to prevent their victim computers from accessing the Internet. Table 1 shows the result of the survey conducted over the last three years on about 110 students enrolled in the intrusion detection and responses course.

<table>
<thead>
<tr>
<th>Questions</th>
<th>Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you experiment the DoS attack, outside the university isolated network laboratory environment, after the hands-on lab exercises practice?</td>
<td>• 82% of the students said “Yes”</td>
</tr>
<tr>
<td></td>
<td>• 13% of the students said “No”</td>
</tr>
<tr>
<td></td>
<td>• 5% abstained</td>
</tr>
<tr>
<td>If yes, where did you experiment the attacks?</td>
<td>• At the home’s network (72%)</td>
</tr>
<tr>
<td></td>
<td>• At the university’s network (25%)</td>
</tr>
<tr>
<td></td>
<td>• At other networks (3%)</td>
</tr>
<tr>
<td>What were your objectives of attacking your victims?</td>
<td>• For fun (76%)</td>
</tr>
<tr>
<td></td>
<td>• Deny the victim from accessing the Internet (24%)</td>
</tr>
</tbody>
</table>

Table 1: Student survey results  
(Number of students = 110)

It is often criticized that offensive methods should not be taught to students since this only increases the population of “malicious hackers”. We feel that this line of argument is flawed. Any security technique can be simultaneously used and abused. The trend towards penetration testing in corporate businesses shows that offensive techniques can be used to increase the level of security of an enterprise. So students trained in offensive techniques must not necessarily become black hats (malicious hackers), but rather can also become white hats (good security professionals). However, we agree that offensive techniques should not be taught in a standalone fashion. As with defensive techniques, every course in IT security should be accompanied by a basic discussion of legal implications and ethics. Students should be educated on their ethical responsibilities. Ethical behaviour is a mandatory part of information security curriculums.

6 Student’s performance and satisfaction

From fall 2006 to spring 2008 (a two years period), students enrolled in the intrusion detection and response course were not offered hands-on lab exercises about CAM table poisoning attack technique. Only the conceptual part of the technique has been described in the class.

However, from fall 2008 to spring 2011 (a three years period), students were offered the hands-on lab exercises described in this paper. Over the last five years period, each semester the students were also given one quiz about switch’s CAM table poisoning attack technique. Figure 6 shows the students total average grades for the quiz, per semester. It is clear that from fall 2008, the students’ total average grade has started improving. This is mainly due to the fact that the hands-on lab exercises allowed students to better anatomize the attack technique and assimilate further the concepts learned from the lecture. The students have learned better with the hands-on lab exercises which had a positive effect on their grading performance.

Figure 6: Student total average grades in the quiz

On the other hand, the students were given a questionnaire survey to assess their overall satisfaction with the hands-on labs and get their feedback. The student survey results are listed in Table 2. Overall the students’ feedback was positive.
Proceedings of the Fourteenth Australasian Computing Education Conference (ACE2012), Melbourne, Australia

Hands-on lab exercises allow students to better anatomize and elaborate the attack in an isolated network laboratory environment. They are designed to be used as a part of an offensive approach to teaching information security: The right way. Proc. of the 3rd Annual Conference on Information Security Curriculum Development, Kennesaw, Georgia, USA, 44-48, ACM Press.


<table>
<thead>
<tr>
<th>Questions</th>
<th>Responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Did you envoy the labs?</td>
<td>• 87% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 10% agree</td>
</tr>
<tr>
<td></td>
<td>• 2% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 1% disagree</td>
</tr>
<tr>
<td>Do you think the labs are easy to follow and straightforward?</td>
<td>• 82% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 10% agree</td>
</tr>
<tr>
<td></td>
<td>• 5% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 3% disagree</td>
</tr>
<tr>
<td>Do you feel you understand the concepts better after performing the labs?</td>
<td>• 85% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 13% agree</td>
</tr>
<tr>
<td></td>
<td>• 1% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 1% disagree</td>
</tr>
<tr>
<td>How likely are you to recommend the labs to others?</td>
<td>• 86% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 11% agree</td>
</tr>
<tr>
<td></td>
<td>• 2% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 1% disagree</td>
</tr>
<tr>
<td>Would you like to see these labs (or similar labs) used in your network security classes?</td>
<td>• 87% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 8% agree</td>
</tr>
<tr>
<td></td>
<td>• 4% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 1% disagree</td>
</tr>
<tr>
<td>Laboratory exercises helped me to learn how to apply security principles and tools in practice.</td>
<td>• 85% strongly agree</td>
</tr>
<tr>
<td></td>
<td>• 8% agree</td>
</tr>
<tr>
<td></td>
<td>• 5% neither agree or disagree</td>
</tr>
<tr>
<td></td>
<td>• 2% disagree</td>
</tr>
</tbody>
</table>

Table 2: Student survey results (Number of students = 40)

7 Conclusion

This paper described in detail two hands-on lab exercises. The first hands-on lab exercise is about how to perform practically DoS attack using switch’s CAM table poisoning. The second hands-on lab exercise is about the implementation of “Security port” feature available in common switches for preventing the attack. The two hands-on lab exercises allow students to better understand and elaborate the attack in an isolated network laboratory environment. They are designed to be used as a part of an undergraduate-level course on network security and intrusion detection and prevention course.

However, a major ethical concern has been identified when analysing the alert logs generated by the intrusion detection sensors installed in the university networks. This is a dilemma when security students are exposed to offensive hands-on lab exercises. However, the ethical concerns of teaching students “hacking” are dwarfed by the need for knowledgeable, competent, and, above all, experienced computer security professionals in industry and government.

8 References
