Algorithms for Network Discovery and Detection of MAC and IP Spoofing Security Attacks

Paulo Lopes, Paulo Salvador, António Nogueira
DETI, University of Aveiro / Instituto de Telecomunicações
Aveiro, Portugal
{pjl90, salvador, nogueira}@ua.pt

***Abstract* — Data Link and Network layers of the OSI model use, respectively, MAC and IP addresses to provide communication between different network devices. Since this is a widely used model, it is frequently explored for various malicious activities. MAC and IP spoofing attacks are the origin of many security threats; so, preventing them is essential to obtain a protected and trustful network. This paper presents an efficient mechanism to detect and block these attacks based on the use of the SNMP protocol, which allows remote access to network devices in order to retrieve their MIB information and is supported by most of the existing network equipment. On a first stage, network discovery is used to identify the devices that are present on the network; then, by selecting and manipulating the MIB information retrieved from these devices, appropriate algorithms are proposed to detect both IP and MAC spoofing attacks. Many performance evaluation tests were conducted and the results obtained proved that these approaches are able to quickly and efficiently detect and block these network security attacks.**

Keywords-SNMP; Network Discovery; MAC Spoofing; IP Spoofing.

I. INTRODUCTION

Today, networks have a fundamental role in our lives, being used for business, communication, data exchange, entertainment, and so on. Due to this increasing importance, networks have been improved in order to become more resilient, secure and able to cope with the appearance of new technologies and applications. The seven layer Open Systems Interconnection (OSI) model was adopted by most of the systems to provide communication between devices. Layer 2, also called Data Link layer, uses a physical Media Access Control (MAC) address to provide communication between the different devices in a local network. This address is a serial number that uniquely identifies the device. Layer 3, also called Network Layer, is responsible for packet routing functions, using the Internet Protocol (IP) to deliver packets from source to destination based on their IP addresses.

Network security vulnerabilities have been intensively explored with the appearance of tools that are able to retrieve critical information, access to unauthorized networks or even overload servers and network connections. This paper focuses on two types of network security attacks: MAC and IP spoofing. MAC spoofing attacks take advantage of the fact that even though a MAC address is supposed to be permanent it can be changed in most of the devices. In this way, an attacker can easily impersonate any user on the network by changing the MAC address of his machine in order to match the MAC address of his target host. Spoofing MAC addresses is one of the most common network attacks and is mostly used to get access to an unauthorized network, using the identity of an authorized client. IP spoofing attacks are similar to MAC spoofing attacks but, in this case, the IP address must be configured to match the IP address of the victim, while the MAC address remains unchanged. Again, the intruder will impersonate an authorized client, getting access to the network.

The approaches proposed in this paper to prevent these types of network attacks are based on the Simple Network Management Protocol (SNMP) protocol [1]. SNMP is used to remotely manage network devices by using data stored on their Management Information Base (MIB) [2] and is supported by most of the network devices. A MIB is a virtual database with information about the network and the device itself; this information is hierarchically organized and each object is identified by the Object Identifier (OID). It is possible to detect and block MAC and IP spoofing attacks, as well as perform network discovery, simply by retrieving and managing the information contained on the MIB of each network device. As will be shown later, the developed algorithms proved to be reliable and efficient in the detection and blocking of both types of network security attacks.

In order to detect and prevent this type of security threat, it is crucial to have a complete knowledge of the network topology. So, this paper will also present a network discovery algorithm that is able to find and distinguish the different network devices, whether they are Layer 2 or Layer 3 equipments. For all network simulations that were carried out, the network discovery method was able to identify the different devices and retrieve network information from their forwarding and Address Resolution Protocol (ARP) tables. The deployment of these methodologies is very simple, so they can be applied on any network, assuming that all devices have been correctly configured.

The rest of this paper is organized as follows. Section II presents the related work on remote access tools and methodologies used to prevent MAC and IP spoofing security attacks; Section III describes a method to perform network discovery using SNMP; Sections IV and V describe the methodologies used to prevent MAC and IP spoofing security attacks using SNMP, respectively. Both sections are divided in two parts: part A presents a method to detect attacks, while part B discusses a solution to block them. Section VI describes the experimental tests that were carried out and the main results obtained and, finally, Section VII concludes the paper.

 ISBN: 978-1-61208-292-9

II. RELATED WORK

A. Remote Access Tools

Many protocols and tools have been developed to remotely manage network devices. Telnet, Secure Shell (SSH) and SNMP are some of the most commonly used protocols.

Telnet is a network protocol used to connect to remote machines located in the same LAN or in the Internet. A Transport Control Protocol (TCP) connection is established to log into a remote machine, using its IP address and port number [3]. The most relevant advantages of this functionality are the fact that it is supported by most operating systems and it provides access to several network services. However, Telnet has some security problems: by default, it doesn't support encryption and most implementations do not even have any authentication, so passwords and other secret information exchanged between devices can be easily intercepted and read. Due to this lack of security, Telnet has been discontinued and replaced by more secure tools.

SSH (Secure Shell) is another network management protocol developed to provide remote access, being primarily used in UNIX and Linux environments [4]. Unlike Telnet, SSH provides encryption and prevents attackers from accessing secret information included in the data packets. Nowadays, it is the most secure and used tool for remote access.

SNMP (Simple Network Management Protocol) is a management protocol that allows a client or manager to poll network devices (agents) running on a network for specific information [1]. This information is contained in a text file, called MIB, and is hierarchically organized. SNMP uses specific commands to access and manage this information. Unlike previous remote access tools, which operate by getting access to a remote machine and then executing commands as if we were working directly on the device, SNMP commands are sent from the local machine to retrieve information from the server. Thus, we only need to execute commands from the local machine in order to get information from any network device that supports SNMP. This allows the user to easily develop scripts that can automatically retrieve and manage information contained on the MIB of each network device; this is why we choose this protocol to implement the methodology for preventing IP and MAC spoofing attacks.

To correct the security deficiencies of SNMPv1 and v2 (the first two versions that were released), SNMPv3 defines an overall SNMP architecture and a set of security capabilities, including three important services: authentication, privacy, and access control. Using SNMPv3, users can securely collect management information from their SNMP agents without fear that the data has been tampered with. Also, confidential information, such as SNMP set packets that change a device's configuration, can be encrypted to prevent their contents from being exposed on the wire. Also, the group-based administrative model allows different users to access the same SNMP agent with varying access privileges.

B. Prevention of MAC and IP spoofing attacks

Several works have addressed the same subject that is studied in this paper, the prevention of MAC and IP spoofing network attacks. Sasu et al. [5] proposes a method to detect MAC spoofing attacks based on the Destination Traffic Fingerprint (DTF). The general idea of this method is to generate constant traffic, which is used as a reference fingerprint, from an end device to a set of IP destinations. The IP address and the traffic percentage are recorded for each fingerprint and, based on this reference, the method establishes an Overall Degree of Recognition that will be used to determine if a MAC address is being spoofed or not.

In Puangpronpitag et al. [6], an egress Network Access Controller NAC () is used to authenticate internal users before they access the external network. Since MAC spoofing attacks can bypass the egress NAC by spoofing the MAC address of an authenticated client in order to get access to the network, the proposed solution is based on an authentication visa checking mechanism. This solution is mostly used on Wi-Fi hotspots, although it can also be used on wired connections using Ethernet ports.

The approach proposed in Wang et al. [7] to prevent IP spoofing attacks is based on the fact that even though an attacker can forge any field of the IP header, he cannot fake the number of hops an IP packet travels to reach its destination. Then, it is possible to create a map of the IP addresses corresponding to the different hops in order to detect spoofed IP packets. The filtering technique is called Hop-Count Filtering (HCF) and detects IP spoofing attacks using an IP-To-Hop-Count (IP2HC) mapping table.

Yao et al. [8] proposes a method to perform IP spoofing filtering that presents resource consumption proportional to the size of the attack. The filtering mechanism, called Virtual Anti-Spoofing Edge (VASE), uses sampling and on-demand filter configuration to detect IP spoofing attacks. Due to the intermittent nature of the attacks, unnecessary overhead is reduced.

In Gonzalez et al. [9], the authors propose a method, based on a Bayesian inference model, to detect attacks that are triggered by access routers. The model evaluates the trustworthiness of a router based on the packets it forwards: a judge router samples all traffic forwarded by each access router and computes the corresponding trust values.

Finally, the approach proposed by Ma [10] provides a defense against IP spoofing attacks using the traceroute utility and relying on the cooperation between trusted adjacent nodes in order to detect and block intruders from external networks. From the obtained results, it is possible to conclude that this approach provides an easy way to effectively detect and prevent IP spoofing attack.

Mopari et al. [11] provides a framework for detecting the DDoS attack and dropping the spoofed packets. By analyzing the number of hops that packets travelled before reaching the destination, the legitimacy of a packet can be found out. In fact, an attacker can forge any field in the IP packet but he cannot control hop count. So, by generating an IP to Hop-Count

 ISBN: 978-1-61208-292-9

mapping table and inspecting it, spoofed packets can be identified.

III. NETWORK DISCOVERY

As previously stated, the main objective of this work is the development of an integrated management tool, based on the SNMP protocol, which can be used to discover the different network elements, detect and prevent MAC and IP spoofing network security attacks. The next three sections will consecutively present the network discovery approach that was devised, as well as the network security attack detection and prevention methodologies that were developed for both types of security flaws.

When an algorithm is used to prevent network security attacks, every device present on the network should be individually analyzed. These devices operate at layers 2 and 3 of OSI model and, in order to find them, network discovery mechanisms should be deployed.

A. Basic Principle

The mechanism illustrated in Fig. 1 is able to perform equipment discovery on the whole network. It starts by accessing an already known router in the network. Then, it retrieves information from the MIB of this router using the "snmpwalk" SNMP command, putting it in an array that can be easily accessed later. This data is retrieved from the MIB objects shown in Table I, which contain information about destination networks, network masks, next-hop IP addresses, used interfaces and route types.

After this step, information from the objects represented in Table II is also retrieved. These objects contain information about the IP addresses corresponding to the media-dependent physical addresses, as well as the associated address types (static or dynamic), MAC addresses and interfaces [12][13][14]. These objects contain all information that it is necessary to perform network discovery, so they must be retrieved every time a router is analyzed.

TABLE I. SOME MIB OBJECTS FROM CISCO IP-FORWARD-MIB

MIB Object	OID	Description
ipCidrRouteDest	.1.3.6.1.2.1.4.24.4.1.1	Destination networks
ipCidrRouteMask	.1.3.6.1.2.1.4.24.4.1.2	Masks of the destination networks
ipCidrRouteNextHop	.1.3.6.1.2.1.4.24.4.1.4	Next hop IP addresses
ipCidrRouteIfIndex	.1.3.6.1.2.1.4.24.4.1.5	Used interfaces
ipCidrRouteType	.1.3.6.1.2.1.4.24.4.1.6	Route types
ipCidrRouteMetric1	.1.3.6.1.2.1.4.24.4.1.11	Route metrics

Each router can have several IP addresses associated to each interface. When performing network discovery, each device only needs to be analyzed once; however, since it can have more than one IP address, the algorithm can analyze the same router more than once. This is why the next step records all IP addresses associated to each interface in order to assure that the device is analyzed only once. This information is found in the router MIB object *ipAdEntAddr* (OID .1.3.6.1.2.1.4.20.1.1).

In order to move to the next network device, destination networks are retrieved from the router MIB. For each destination network, the algorithm must check the route type. If the route type to that network is indirect, the value of the next-hop IP address is read and the algorithm moves to the router with this IP address, following all the previous steps. Since this is the first router, we can move to the next device without checking if it was already analyzed. However, from now on it is necessary to compare the next-hop IP address with the list of IP addresses corresponding to the devices where we have already been.

TABLE II. SOME MIB OBJECTS FROM CISCO IP-MIB AND RFC1213-MIB

MIB Object	OID	Description
ipNetToMediaNetAddress	.1.3.6.1.2.1.4.22.1.3	IP address of media-dependent physical interfaces
ipNetToMediaType	.1.3.6.1.2.1.4.22.1.4	Address type
atPhysAddress	.1.3.6.1.2.1.3.1.1.2	MAC address
atIfIndex	.1.3.6.1.2.1.3.1.1.1	Interface

If the route type to a destination network is of the direct type, the IP addresses of all Layer 2 devices present on that network must be read. Whenever the algorithm finds in the list an IP address corresponding to a network device that was not already analyzed and whose address type is defined as dynamic (because static IP addresses usually belong to the interfaces of the device that it is being analyzed), then the algorithm moves to this new network device. After all Layer 2 devices present on a given network have been analyzed, the next destination network from the array is read and the route type is checked again. Since this is a recursive algorithm, when there are no more destination networks to reach, we must go back to the previous router that was being analyzed. When the first router that was analyzed is finally reached and there is no more destination networks to move to or Layer 2 devices to analyze in a given network, then it means that all network devices have been discovered.

For Layer 2 devices the task is much simpler. A list of devices belonging to a given network is analyzed using the *ipNetToMediaNetAddress* MIB object and when a Switch or Access Point that was not analyzed is found, the algorithm has simply to move to there, retrieve the necessary information and read the next IP address from the list. So, the first thing to do with Layer 2 devices is to record its IP address and then retrieve information about its forwarding table. This can be done by retrieving information from the following MIB objects: *dot1dTpFdbAddress* (OID .1.3.6.1.2.1.17.4.3.1.1) and *dot1dTpFdbPort* (OID .1.3.6.1.2.1.17.4.3.1.2). These objects represent, respectively, the MAC addresses and the corresponding bridge ports from the forwarding table of the device.

 ISBN: 978-1-61208-292-9

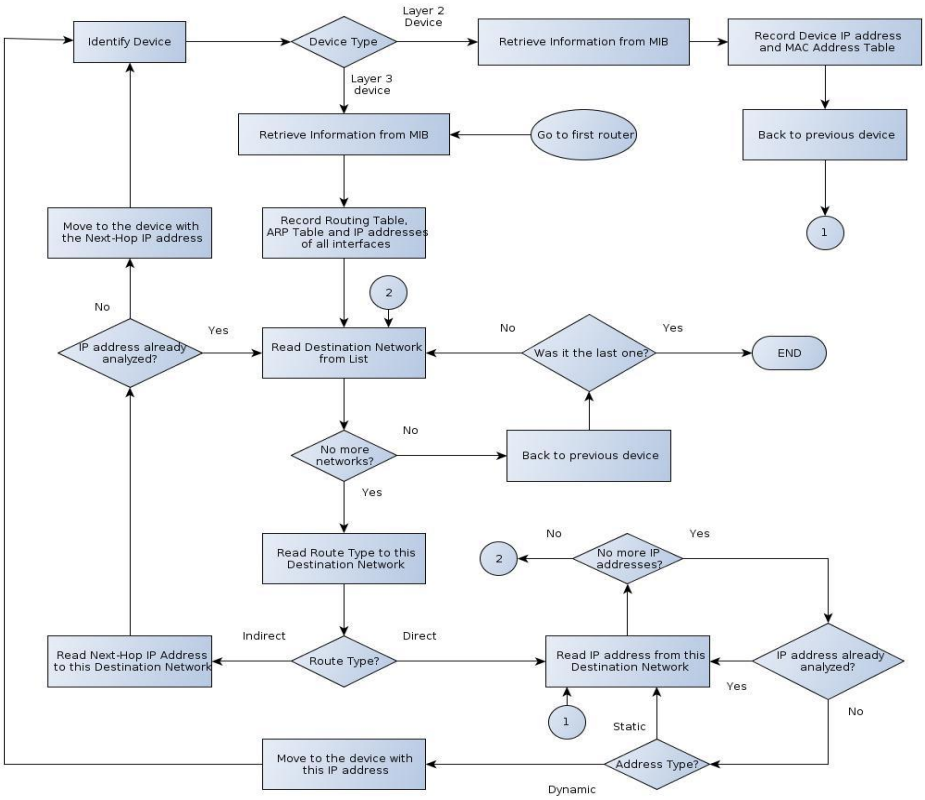

Figure 1. Network discovery algorithm

To convert the bridge port into the actual device interface, two MIB objects should be used: *dot1dBasePortIfIndex* (OID .1.3.6.1.2.1.17.1.4.1.2), to get the interface index, and *ifDescr* (OID .1.3.6.1.2.1.2.2.1.2), to get the interface name [15]. This process allows retrieving the same information that is obtained when the "show mac-address-table" command is executed in Cisco devices.

B. Some considerations

Although most of the network devices support the SNMP protocol, there are still some exceptions: we can consider that all Layer 3 devices support SNMP, but Layer 2 devices can be managed (devices that support SNMP) or unmanaged. If a network has any unmanaged switch, it won't be detected by the network discovery mechanism. To solve this problem, a counter can be created to check how many Layer 2 devices the algorithm analyzes in a certain Local Area Network (LAN). This counter will obviously count the number of managed devices. On a managed switch, the *atPhysAddress* MIB object can be used to count the number of Layer 2 devices present in the LAN (even the unmanaged ones). The difference between the two counters corresponds to the number of unmanaged devices. These unmanaged switches must be manually checked every time a MAC spoofing or IP spoofing attack can not be blocked.

Another point that has to be taken into account is the fact that Layer 2 devices include switches and access points. They have different characteristics and consequently they must be treated differently. This paper will describe in detail the steps to detect any network attack and block it in case the attacker is accessing the network from a switch. In this case, the port it is connected to must be blocked. On the other hand, if we are dealing with an attack triggered from an access point, then the attack can only be detected when it belongs to the IP spoofing attack type. This is due to the fact that, using this method, MAC spoofing attacks are detected based on the MAC address and interface that the intruder is using to access the network. In case the attacker is accessing the network from the same access point of the authorized client, there is no way to distinguish between them, because they are using the same MAC address and the same interface. That situation does not happen on IP spoofing attacks because in this case IP addresses and MAC addresses are compared and, once the MAC address of the intruder is found, the task is simply to find it on the network and block it. If the attacker is accessing the network from an access point, the procedure is similar to the case of switches but, instead of blocking the interface that the attacker is using (the wireless interface), the MAC address of the device that is being used to perform the attack is blocked; otherwise, the other devices that are using the interface could not access the network anymore. Blocking the MAC address of an end host must be done manually via SSH, for example, through the MAC Access Control List (ACL) of the access point. When

 ISBN: 978-1-61208-292-9

performing MAC spoofing detection, access points are considered unmanaged devices.

Finally, it is important to refer the case of routers that are working with a switch module. Although they are routers by default, they can work like switches and have exactly the same behavior. They can also be accessed via SNMP and its information can be retrieved, similarly to any other network device. But during this work we have seen that most of these devices have a lack of information on their MIBs, which do not allowed us to retrieve the necessary information from this type of devices. For this reason, any router working with a switch module will be considered as an unmanaged switch.

IV. MAC SPOOFING

As previously said, Layer 2 devices use MAC addresses as their LAN identifiers. This address is assigned by the manufacturer to each interface of the device and is controlled by the Organizationally Unique Identifiers (OUI) to be globally unique for all LAN-based devices. However, MAC addresses can easily be changed in most devices without any consequences on their performance. This means that faking MAC addresses is a simple way for an attacker to perform network security attacks. There are several reasons to perform this kind of attacks [16], but one of the most common is to impersonate an already authenticated user. In this case, the attacker just needs to know the client MAC address and change its own address accordingly. In this way, and since the user is already authenticated on the network, the attacker can send and receive traffic disguised by the MAC address of the user.

In the next sub-section, we will present a procedure, based on SNMP protocol, to detect these Layer 2 attacks and block the access of the intruder to the network.

A. Attack Detection

Fig. 2 describes a method to detect and block MAC spoofing attacks. This mechanism will basically create a record of the MAC addresses of all interfaces of the different network end devices. If someone tries to fake a MAC address, then the port or even the switch will change because that MAC address will appear on another location. This algorithm is able to detect such situation and figure out if it is really a MAC spoofing attack or if the client has simply changed the physical location of the device.

The algorithm starts by performing the network discovery procedures described in the previous section in order to find and identify all network devices. When dealing with MAC spoofing attacks, we just have to analyze switches. After selecting these devices, each one is analyzed individually. Then, useful information is retrieved from the MIB of the switches. Information that it is needed to detect MAC spoofing attacks should be selected, retrieved using the SNMP "*snmpwalk*" command and put in an array in order be easily accessible. The necessary MIB objects are *dot1dTpFdbAddress*, *dot1dTpFdbPort* and *atPhysAddress*, as already mentioned in the previous section. Below, we will show why this information is so important and we will mention other MIB objects that are used in this detection approach.

Switch access ports are needed because end hosts are connected there. Since all ports are already known, access ports can be selected using the MIB object *vlanPortIslOperStatus* (OID .1.3.6.1.4.1.9.5.1.9.3.1.8), which returns value '1' for Trunking and '2' for Not Trunking. However, an access port can also be connected to another network device instead of an end host. In this case, the MIB object *atPhysAddress* should be used. If any of the MAC addresses associated to an access port belongs to the list of MAC addresses of the *atPhysAddress* object, it means that the access port is not connected to an end device and should be excluded from the list of ports to analyze.

The first stage is completed and we now have all the necessary information. The next step consists of reading each MAC address associated to the selected access ports. When a MAC address is analyzed, the method should check if it was already recorded. We choose to maintain a record of all MAC addresses of the end hosts that are found on the network. If the MAC address that it is being analyzed does not exist yet in this historic, then a record must be added, containing the MAC address, the corresponding network device and the port where it is connected to. The access port is already known and the information about the device can be retrieved through the MIB object *hostName* (OID .1.3.6.1.4.1.9.2.1.3). The registration time is also recorded, as well as a counter whose value is 0. This is all the information that is needed regarding each MAC address that is detected in the network. Then, the next MAC address in the array should be read. When there are no more MAC addresses to read, the algorithm moves to the next Layer 2 device.

When a MAC address is already registered, its location in the network should be checked to verify if it is in the same place or if it has moved to another location. The historic already contains the switch and port associated to this MAC address. So, the recorded information is compared to the switch and port that the MAC address is using now: if they are equal, it means that the end host is in the same place; otherwise, we can be sure that the end host has changed its physical location or someone is faking this MAC address and is using it to connect to the network from another location.

TABLE III. SOME MIB OBJECTS FROM CISCO BRIDGE-MIB AND CISCO STACK-MIB

MIB Object	OID	Description
dot1dTdbAddress	.1.3.6.1.2.1.17.4.3.1.1	MAC addresses from the MAC address table
dot1dTpFdbPort	.1.3.6.1.2.1.17.4.3.1.2	Bridge ports from the MAC address table
dpt1dBasePortIfIndex	.1.3.6.1.2.1.17.1.4.1.2	Interface index
vlanPortlslOperStatus	.1.3.6.1.4.1.9.5.1.9.3.1.8	Trunk or access port

 ISBN: 978-1-61208-292-9

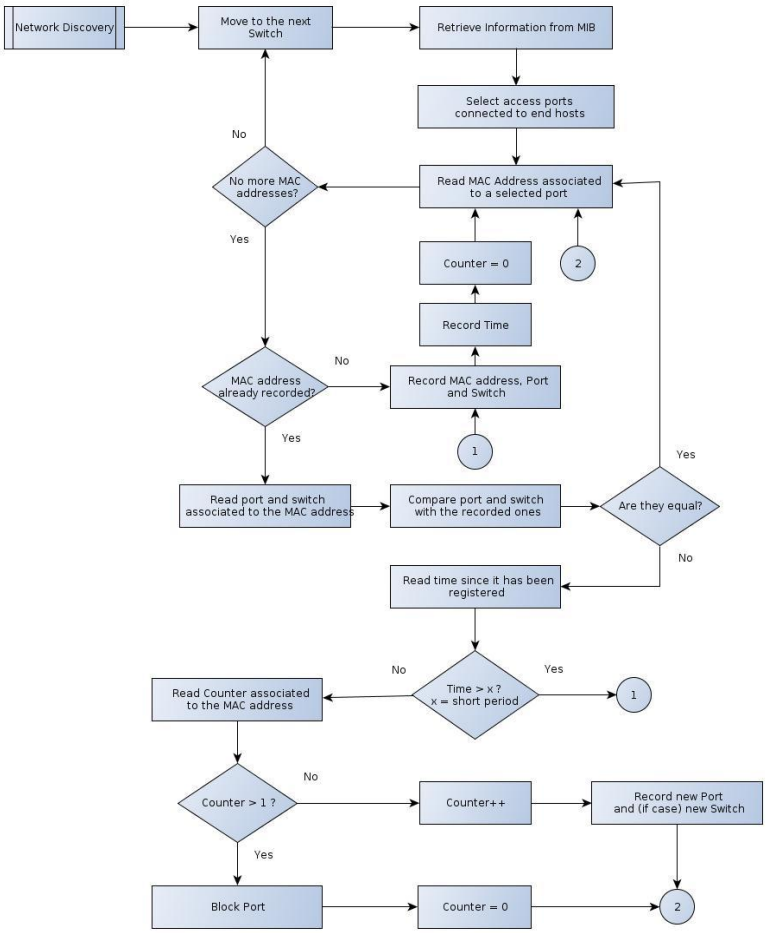

Figure 2. MAC Spoofing detection and blocking algorithm

B. Attack Blocking

Once a possible attack is detected, it is important to verify if it is a real attack or if the user has just moved the end device to another location on the network. The first question that should be answered in order to understand the reason for this change is: how much time has passed since the MAC address has been registered? When the MAC address was recorded for the first time, many parameters were saved, including the registration time. In this way, it is possible to check how much time has elapsed since that instant. When there is a MAC spoofing attack, a client is communicating and the attacker is using the same MAC address to send and receive traffic from the network, but from another location. This means that in a real MAC spoofing attack changes will be detected in the port (and possibly in the switch) associated to the MAC address in a short period of time (few seconds). So, if the time elapsed since the MAC address has been registered is greater than this short time interval, it means that probably the client has just changed his location and the network is not under attack. In this case, the new port has to be recorded and, if it is the case, the new switch. The registration time is also updated and the counter is set to value 0 (if it was not 0 already).

On the other hand, if the time since the MAC address registration is shorter than the time period that is considered normal when the network is under attack (in our tests this value was considered as equal to 30 seconds), then another question arises: how many times this MAC address has changed its location during the short time period we are considering? The counter parameter can be used to answer this question. If a change was detected in the last seconds, then the counter associated to the MAC address must be checked. If the counter has a value of 0 or 1, then it means that in the last seconds that MAC address has not changed its location or has changed it only once, which can be considered as normal. In this case, the counter is incremented and the new port is updated. The time parameter is not updated because it is necessary to check if there will be more changes in the next few seconds. If the counter reaches a value greater than 1, it means that a change of location was detected more than once in a short period of a few seconds, which can be considered as an unusual behavior and consequently there is a high probability that the network is under a MAC spoofing attack.

When a MAC spoofing attack is detected, it must be blocked. Using this method, this operation is really easy to accomplish because a record of the previous ports and switches is maintained and compared to the port and switch that a given MAC address is using now to access the network. So, if a MAC spoofing attack is detected and the attacker is using a switch to perform the attack, the port where the MAC address

 ISBN: 978-1-61208-292-9

is connected to at the moment will be blocked. Using information corresponding to the bridge ports associated to the different MAC address (available from the *dot1dTpFdbPort* MIB object), the interface index of the device that has to be blocked can be retrieved using the SNMP "snmpget" command over the *dot1dBasePortIfIndex* MIB object (OID .1.3.6.1.2.1.17.1.4.1.2). Finally, we can block the port using the SNMP "snmpset" command over the MIB object *ifAdminStatus* (OID .1.3.6.1.2.1.2.2.1.7), which will shut down the interface and block the attack. In case the attacker is accessing the network from an unmanaged device, the device must be checked manually, as previously said.

V. IP SPOOFING

After the analysis of Layer 2 network attacks, it is time deal with Layer 3 attacks or IP spoofing attacks. Unlike MAC addresses, IP addresses must be configured whenever new equipment is connected to the network; otherwise, communication will fail. But, when IP addresses are not assigned automatically through Dynamic Host Control Protocol (DHCP) and the user does not know all IP addresses of the network, there is always the risk to configure a device with an IP address that is already in use. IP spoofing attacks are based on the principle that if the intruder impersonates an authorized client by using its IP address, then he can get access to the network because all devices will believe that those packets come from a trusted host [17].

There are several tools to prevent this kind of network attacks. Here, we will present a simple methodology based on the SNMP protocol. Like we did in the previous section, the approach will be divided in two parts: detecting the IP spoofing attack and blocking it.

A. Attack Detection

Fig. 3 illustrates a method to detect IP spoofing attacks. For each detected end host a record is created containing its IP and MAC addresses. If an attacker tries to use an IP address that is already in use, that occurrence will be detected by the simple reason that the MAC address of his device is different from the MAC address of the victim. This is the basic principle of this method. As shown in Fig. 3, the first thing to do is a network discovery to find all routers, switches and access points of the network. Since we are talking about Layer 3 attacks, all routers must be analyzed until an IP spoofing attack is detected. When that happens, the attacker access to the network must be blocked. To do so, all Layer 2 devices have to be checked until the intruder is found. First of all, after having a complete list of all Layer 2 and Layer 3 devices, each router of the network is analyzed separately. Then, it is necessary to retrieve and select information from its MIB in order to detect the attack. The MIB objects that should be retrieved and put in an array are: *ipNetToMediaNetAddress*, *ipNetToMediaType* and *atPhysAddress*. All of them were already mentioned in previous sections.

With this information, it is possible to have access to all IP addresses of the router forwarding table, as well as the correspondent MAC addresses and address types. A new cycle must be initiated in order to analyze all these IP addresses until there are no more addresses to read, and then move to another router and perform the same steps. When an IP address is analyzed, the first thing to do is to check for the address type. An IP address can be selected to be static or dynamic, but in this case we are only interested on dynamic addresses because we are looking for IP addresses of end devices and these are always dynamic. If an IP address is static, then the next IP address from the array must be read. If that IP address is dynamic, we have to check if it was already recorded. Like happened for MAC spoofing attacks, a record including some different parameters is kept in order to have a comparison base for the future. For each end host IP address, the corresponding MAC address and registration time are saved. If a given IP address was already registered, then recorded information must be checked. First, the MAC address that was recorded should be read and compared to the MAC address of the device that is using the same IP address at this moment. If they are equal, then it means that the IP address is being used by the same equipment and nothing wrong is happening, so the next IP address from the array can be read. If the MAC address is different, two possible things could have happened: the user simply started using a new device and configured it with the same IP address in order to have access to the network or someone is trying to perform a network attack by using the IP address of an authorized client.

In order to distinguish between these two situations, the registration time parameter is used. It is not common that an IP address is associated to different end devices in a short period of time. It can happen occasionally, for example when an end host leaves the network and the IP address that was associated to it is available to be assigned to another device. It is expected that once an end host is configured with an IP address, no one else will get the same IP address for a period of time of at least some minutes. Based on this principle, if different MAC addresses are detected for the same IP address, it must be verified how many time has passed since it was registered.

If this time is greater than the time period that is considered as normal, then a new record for this new MAC address must be created, besides updating the new registration time. On the other hand, if only a short period of time has elapsed since it was registered, then there is a great probability that some user is using the IP address of someone else to perform an IP spoofing attack against the network. In this case, we have to move on to the next stage in order to find the location of this new MAC address in the network and block the port of the switch or access point where it is connected to.

B. Attack Blocking

At this point, the IP spoofing attack was detected and the MAC address of the device that it is being used to perform the attack is already known. So, each Layer 2 device on the network should be analyzed in order to localize this MAC address. Fig. 4 describes the approach that was devised to block IP spoofing attacks.

 ISBN: 978-1-61208-292-9

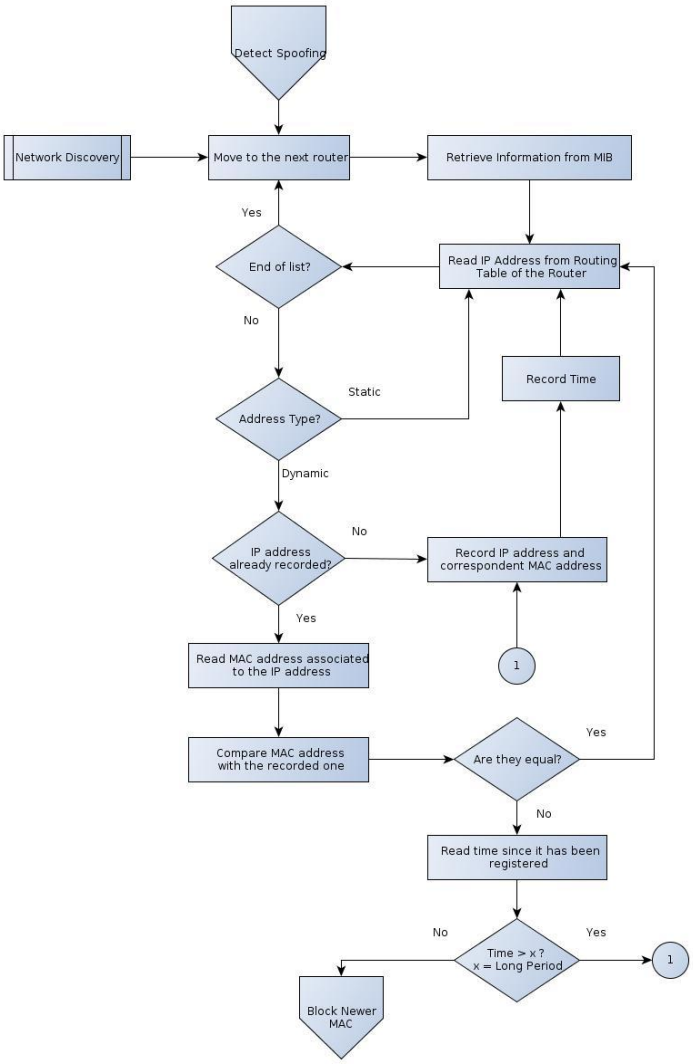

Figure 3. IP Spoofing Detection algorithm

The first thing to do is to retrieve the necessary information from the MIB of each Layer 2 device, as was previously done every time we needed to analyze any network device. In this case, the MIB information that it will be used is the same that was mentioned before to detect MAC spoofing attacks. So, the MIB objects retrieved from the switch or access point are the following: *dot1dTpFdbAddress*, *dot1dTpFdbPort* and *atPhysAddress*.

In the case of switches, ports that are being used exclusively by end devices should be identified. In order to do that, switch access ports are selected using the MIB object *vlanPortIslOperStatus* (OID .1.3.6.1.4.1.9.5.1.9.3.1.8). Then, the ports that are connected to other network devices must be excluded. If the MAC address associated to any of these ports is present in the list of MAC addresses retrieved from the *atPhysAddress* MIB object, it means that this port is not connected to an end host and can be excluded. After performing these steps, we only have the necessary switch ports.

The next step is to analyze each one of the selected ports until there are no more ports to read and, then, move to the next Layer 2 device. For each port, the associated MAC address in this particular instant is read; this address is compared with the MAC address that was previously identified as belonging to the intruder. If they are different, it means that the end device that is connected to the port is not the one we are looking for and we should move to the next port. When the right MAC address is finally found, the associated port is blocked. The interface index is necessary to block the port.

 ISBN: 978-1-61208-292-9

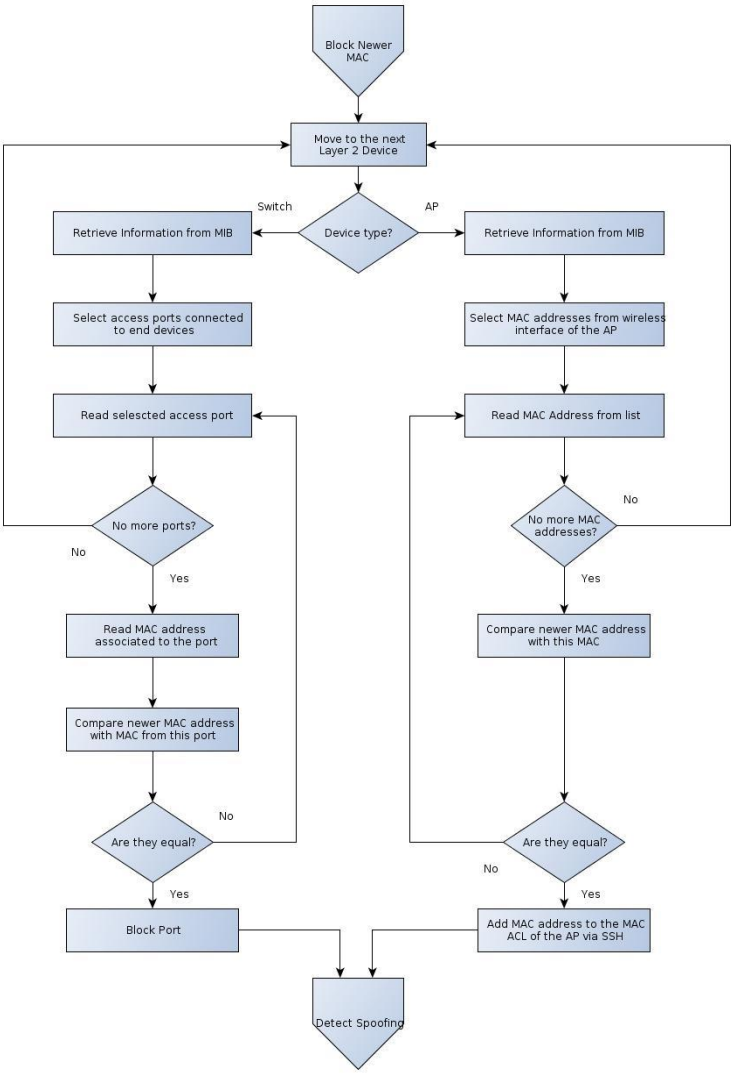

Figure 4. IP Spoofing Blocking algorithm

Using the bridge port retrieved from the *dot1dTpFdbPort* MIB object, it is possible to get the corresponding interface index using the *dot1dBasePortIfIndex* MIB object (OID .1.3.6.1.2.1.17.1.4.1.2) and executing the "snmpget" SNMP command. To turn the interface down, the "snmpset" command is executed over the *ifAdminStatus* MIB object (OID .1.3.6.1.2.1.2.2.1.7).

In case the attacker is accessing the network from an access point, all MAC addresses connected to the wireless interface will be read. If the MAC address of the intruder is not present on this list of MAC addresses, it means that it is not connected to the access point and we can move to the next Layer 2 device. Otherwise, if the MAC address we are looking for is detected in a certain access point, it must be added to the MAC ACL of the access point via SSH in order to block the access of the host to the network.

This methodology is an efficient way to block IP spoofing attacks from intruders that are accessing the network using switches or access points.

VI. EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed methodologies, the simulation scenario illustrated in Fig. 5 was set up and several simulation tests were carried out. This network is composed by four Cisco C3640 routers connected to each other, in a mesh structure.

The first mechanism that should be tested is network discovery. As previously said, the network discovery algorithm has to be sure that all managed devices are analyzed once, which is assured by this network topology. The device identified as PC represents the local machine that will work as the monitoring station to manage the network. This machine is a common laptop running Linux Ubuntu 11.10. Router R1 and

 ISBN: 978-1-61208-292-9

PC are connected to a Cisco C3725 router using a switch module (SWR1), which is considered an unmanaged device. Finally, router R3 is connected to a Cisco C3750 Catalyst switch (SW1), which is also connected to two hosts. These hosts will be used to simulate MAC and IP spoofing attacks by simulating a user with authorized access to the network and an intruder that will impersonate the user to get access to the network. Before running the algorithms, some initial information has to be inserted. For the network discovery algorithm, it is necessary to provide the IP address of any one of the network routers; it is irrelevant which router is introduced because the algorithm was developed in order to discover all network devices, independently of the first router. Then, depending on the SNMP version that it is being used, the user has to insert the same community string (version 2) or authentication password (version 3) that was configured on the devices. This allows the correct execution of the SNMP commands at the local machine. When running the network discovery algorithm, an IP address of each router and the IP address of the switch were recorded for posterior use. This information will be useful for the attack detection algorithms. The router with the switch module was undetected, as supposed. It was also possible to arrange the information retrieved from the MIB of the devices in order to graphically consult the routing tables and ARP tables from each router and the forwarding table from each switch.

For the chosen network scenario, the network discovery algorithm took 3 minutes and 15.7 seconds from the beginning of its execution until it finished the whole discovery process. This time value was obtained using the “time” command, which returns the exact time that a process takes to be executed. When the algorithm execution finally stopped, it was possible to retrieve information from all network devices that support SNMP.

For detecting MAC spoofing attacks, the corresponding detection algorithm was executed in an infinite loop. Then, one of the hosts was assigned with an IP address. The MAC address of the other host was changed in order to match the one that was in use by the first host and the host was configured with a different IP address. As previously explained, this method detects MAC spoofing attacks based on the time that has elapsed since a MAC address is registered, which is approximately equivalent to the moment when the host executes a ping command for the first time. Thus, for simulation purposes, we defined a time period of 30 seconds to distinguish between an attack and a change on the device location.

For simulating a MAC spoofing attack, both hosts have to continuously send packets to the local machine. When the first host executes a ping command, the MAC address is registered, together with the corresponding information. Then, when the second host (the intruder) started sending packets, consecutive changes on the origin of the MAC address were detected and the attack was actually blocked. The switch interface where the attacker was connected to was shutdown and the real host kept accessing the network without its performance had been affected. To confirm the efficiency of this algorithm, 20 attack simulations were performed and the results obtained can be observed in Table IV. It was verified that the attacks were detected in 18 of the 20 simulations and once the attacks were detected they were always blocked. The time since the intrusion starts until the intruder's access is blocked was quite variable, with a mean value that falls, with 95% confidence, in interval [9.368; 12.429].

TABLE IV. MAC SPOOFING ATTACK RESULTS

Simulation	Detected	Blocked	Blocking Time (s)
1	✓	✓	12.181
2	✓	✓	7.271
3	✓	✓	8.544
4	×	×	-
5	✓	✓	9.732
6	✓	✓	18.548
7	✓	✓	15.572
8	✓	✓	10.348
9	×	×	-
10	✓	✓	12.835
11	✓	✓	7.649
12	✓	✓	8.640
13	✓	✓	12.248
14	✓	✓	13.800
15	✓	✓	11.561
16	✓	✓	9.825
17	✓	✓	12.216
18	✓	✓	7.459
19	✓	✓	6.836
20	✓	✓	10.909

To test if the algorithm is able to distinguish the situation of a simple change on the location of the device, the same hosts and the same configuration were used. The first host started sending packets to the local machine and, after some time, it stopped. The MAC address and its origin were registered by the algorithm. After a time period greater than 30 seconds, the second host executed a ping command. Since both hosts have the same MAC address, this procedure simulates a change on the location of the first host. As expected, the new MAC address information was registered and no attack was detected. So, this method is able to distinguish between an attack situation, where two computers with the same MAC address are accessing the network, and the situation where a device changes its physical location in the network.

Finally, in order to test the defense mechanism against IP spoofing attacks, the corresponding detection algorithm was executed in an infinite loop. The two hosts that were previously presented were used again. The host representing the victim was configured with an IP address and the same address was assigned to other host. Let us recall that this method detects IP spoofing attacks based on the time elapsed since an IP address is registered. In practice, this time period corresponds to some minutes, but for simulation purposes it was defined as 2 minutes.

To simulate the attack, the first host executed a ping command to the local machine. When the second host accessed the network and sent packets within a time period shorter than 2 minutes, the attack was immediately detected. The switch interface where the host was connected to was blocked and the performance of the first host was not affected.

 ISBN: 978-1-61208-292-9

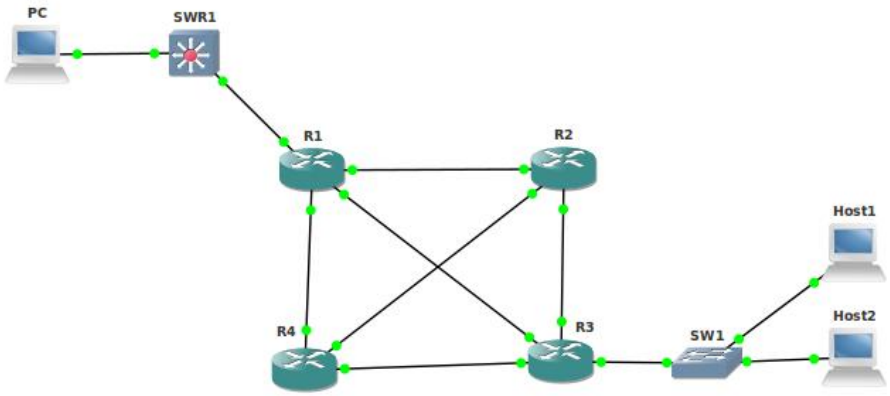

Figure 5. Testing Network

TABLE V. IP SPOOFING ATTACK RESULTS

Simulation	Detected	Blocked	Blocking Time (s)
1	✓	✓	8.079
2	✓	✓	8.352
3	✓	✓	8.469
4	✓	✓	9.042
5	✓	✓	9.040
6	✓	✓	10.848
7	✓	✓	8.612
8	✓	✓	9.292
9	✓	✓	9.076
10	✓	✓	7.536
11	✓	✓	9.071
12	✓	✓	7.863
13	✓	✓	8.795
14	✓	✓	8.920
15	✓	✓	8.613
16	✓	✓	8.424
17	✓	✓	8.560
18	✓	✓	9.264
19	✓	✓	8.452
20	✓	✓	8.517

To test the real efficiency of the algorithm, 20 attack simulations were performed. The simulation results are shown in Table V: 20 out of the 20 attacks were detected and all of them were also blocked. In terms of blocking time, it was quite regular, or at least more regular than in the MAC spoofing detection case, with a 95% confidence interval for the mean time equal to [8.426; 9.057].

On the other hand, in order to test if the algorithm is able to detect the situation of a second machine that is assigned with the same IP address but does not have any malicious purpose, the first host executed a ping command and stopped after some time. The second host has also executed a ping command but more than 2 minutes after the first one; in this case, the attack was not detected and information regarding the origin of the IP address was updated.

These experimental tests proved the efficiency of the proposed methodologies for detecting and blocking MAC and IP spoofing attacks by distinguishing between the situations corresponding to real network security attacks and to changes on the network layout. The proposed methodologies are easily deployed and work in any network, assuming that all devices are correctly configured.

VII. CONCLUSION

This paper presented several methodologies to perform network discovery and prevent MAC and IP spoofing attacks. The proposed tools are very simple to implement and can be deployed in any network that requires monitoring and has stringent security requirements. The proposed tools were developed for Cisco equipment but can be easily extended to devices from any other vendor by adapting the MIB objects that should be retrieved. There are several approaches in the literature for the detection of MAC and IP spoofing network security attacks. The great advantage of the proposed methodologies relies on the fact that they are based on the popular SNMP protocol, are very simple to use and have the potential to simultaneously perform other network monitoring tasks.

ACKNOWLEDGEMENTS

This work was supported by Fundação para a Ciência e a Tecnologia (FCT) of Portugal.

REFERENCES

[1] A. Clemm, Network Management Fundamentals, Cisco Press, 2006, pp 249-261.

[2] K. McCloghrie and M. Rose, RFC1213 - Management Information Base for Network Management of TCP/IP-based internets: MIB-II, March 1991.

[3] G. Sanjing and H. Lihui, “Research of the Telnet Remote Login”, Proceedings of the Third International Symposium on Electronic Commerce and Security Workshops (ISECS ’10), Guangzhou, P.R. China, 29-31 July, 2010, pp. 219-221.

[4] T. Ylönen, “SSH – Secure Login Connections over the Internet”, Sixth USENIX Security Symposium, San Jose, California, USA, 22-25 July, 1996, pp. 37-42.

[5] E. Sasu and O. Prostean, “Network simulation for MAC spoofing detection, using DTF method”, 7th IEEE Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 24-26 May, 2012, pp. 291-296.

[6] S. Puangpronpitag and A. Suwannasa, “A design of egress NAC using an authentication visa checking mechanism to protect against

 ISBN: 978-1-61208-292-9

MAC address spoofing attacks", 8th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Thailand, 17-19 May, 2011, pp. 300-303.

[7] H. Wang, C. Jin and Kang Shin, "Defense Against Spoofed IP Traffic Using Hop-Count Filtering", IEEE/ACM Transactions on Networking, vol. 15, Issue: 1, Feb. 2007, pp. 40-53.

[8] G. Yao, J. Bi and P. Xiao, "VASE: Filtering IP spoofing traffic with agility", International Journal of Computer Networks, vol. 57, Issue 1, Jan. 2013, pp. 243-257.

[9] J. Gonzalez, M. Anwar and J. Joshi, "A trust-based approach against IP-spoofing attacks", 9th Annual International Conference on Privacy, Security and Trust (PST), 19-21 July, 2011, pp- 63-70.

[10] Y. Ma, "An Effective Method for Defense against IP Spoofing Attack", 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), 23-25 Sept., 2010, pp. 1-4.

[11] I. Mopari, S. Pukale and M. Dhore, "Detection and defense against DDoS attack with IP spoofing", Proceedings of the International Conference on Advances in Computing, Communication and Control, 2008, pp. 489-493, http://dx.doi.org/10.1145/1523103.1523200

[12] Cisco Tools & Resources, "SNMP Object Navigator", URL: http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en, (20 Feb 2013).

[13] Y. Qiuxiang, "Algorithm Research of Topology Discovery on SNMP", International Conference on Computer Application and System Modeling (ICCASM), 22-24 October, 2010, vol. 12, pp. 496-497.

[14] K. Qin and C. Li, "Network Topologic Discovery Based On SNMP", Proceedings of the 5th International Conference on Ubiquitous Information Technologies and Applications (CUTE), 16-18 December, 2010, pp. 1-3.

[15] Cisco IP Application Services, "Using SNMP to Find a Port Number from a MAC Address on a Catalyst Switch", URL: http://www.cisco.com/en/US/tech/tk648/tk362/technologies_tech_note09186a00801c9199.shtml, (25 Feb 2013)

[16] A. Pandey and J. Saini, "Counter Measures to Combat Misuses of MAC Address Spoofing Techniques", Int. J. Advanced Networking and Applications, 2012, vol. 3, Issue 05, pp. 1358-1361.

[17] S. Rana and T. Bansod, "IP Spoofing Attack Detection using Route Based Information", International Journal of Advanced Research in Computer Engineering & Technology, vol. 1, Issue 4, June 2012.

 ISBN: 978-1-61208-292-9