An Analysis of Local Area Network ARP Spoofing

Jyotinder Kaur
Computer Science & Engineering Department
BBSBEC, Fatehgarh sahib(Punjab) India

Sandeep Kaur Dhanda
Computer Science & Engineering Department
BBSBEC, Fatehgarh sahib(Punjab) India

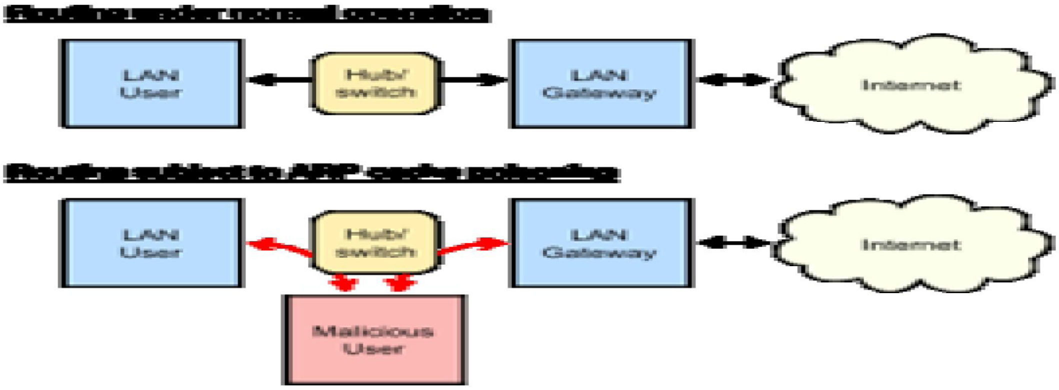
Abstract: -ARP spoofing is a technique whereby an attacker sends fake Address Resolution Protocol (ARP) messages onto a Network. The aim is to associate the attacker's MAC Address with the IP address of another host causing any traffic meant for that IP address to be sent to the attacker instead. ARP spoofing may allow an attacker to intercept data frames on a LAN, modify the traffic, or stop the traffic altogether. Often the attack is used as an opening for other attacks, such as denial of service. The main aim of in this paper Identify various LAN attacks including eavesdropping, denial-of-service etc. and determine relevant security controls required for specific LAN, Wi-Fi attacks. Our topic is basically the based on the Network Sniffing. Sometimes it is possible observe/record traffic traveling on a network. Network traffic may contain valuable information like Usernames and passwords-mail, web requests (and replies), data files Etc.Most sniffers include a protocol analysis component, which organizes and displays the contents of the traffic. The main objectives are in client service architecture, any information that is traveling should be known; this information is provided by valid MAC address. If in any case, we don't have client-server architecture then how this problem can be solved, but the basis should be spoofing. Fetching important credential information from the user side. The main task considered in this paper work is the analysis of network based on ARP Spoofing. Sometimes it is possible observe/record traffic traveling on a network. Network traffic may contain valuable information like Usernames and passwords, E-mail, web requests (and replies), data files. In this ARP we will try to have access to a server which will have access to all the client systems connected to it. Server will have the information of all the systems connected and will contain the MAC address of each. Our preference to this topic is basically to make the administrator or the server vigilant about the network usages by the clients and to keep a track of the activities performed over a network. Further there will be a comparison between the various packet analyzer tools which serve our purpose and will be in accordance with our parameters like traffic monitoring, LAN Detective Internet Monitor.

Keyword: - Data, Network, packets analyzer tools, sniffers

I. INTRODUCTION

Software that detects ARP spoofing generally relies on some form of certification or crosschecking of ARP responses. Uncertified ARP responses are then blocked. These techniques may be integrated with the DHCP server so that both dynamic and static IP addresses are certified. This capability may be implemented in individual hosts or may be integrated into Ethernet switches or other network equipment. The existence of multiple IP addresses associated with a single MAC address may indicate an ARP spoof attack, although there are legitimate uses of such a configuration. In a more passive approach a device listens for ARP replies on a network, and sends a notification via email when an ARP entry changes. In Microsoft Windows, the behavior of the ARP cache can be configured through several registry entries under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters,ArpCacheLife,ArpCacheMinReferenceLife, ArpUseEtherSNAP, ArpTRSingleRoute, ArpAlwaysSourceRoute, ArpRetryCount. AntiARP also provides Windows-based spoofing prevention at the kernel level. ArpStar is a Linux module for kernel 2.6 and Linksys routers that drops invalid packets that violate mapping, and contains an option to repoison/heal. The simplest form of certification is the use of static, read-only entries for critical services in the ARP cache of a host. This prevents only simple attacks and does not scale on a large network, since the mapping has to be set for each pair of machines resulting in approximately n^2 ARP entries that have to be configured when n machines are present:

On every machine there must be an ARP entry for every other machine on the network, which are *n* ARP entries on every of the *n* machines.



Generally, the goal of the attack is to associate the attacker's MAC address with the IP address of a target host, so that any traffic meant for the target host will be sent to the attacker's MAC instead. The attacker could then choose to Inspect the packets, and forward the traffic to the actual default gateway (interception), Modify the data before forwarding it (man-in-the-middle attack) and Launch a denial-of-service attack by causing some or all of the packets on the network to be dropped.

Figure No 1 A successful ARP spoofing (poisoning) attack allows an attacker to alter routing on a network, effectively allowing for a man-in-the-middle attack.

II. PROBLEM FORMULATION

Packets contain a lot of useful information about password activity that can be used as a description of the general password behavior network administrator to capture such kind of network information. Packets are units of data traveling in these computer networks and they can carry all the important information from its source to final destination. Packets also contain a wealth of information about the network infrastructure, topologies and provide the information of network traffic. Password packet analyzer will be very useful to people who have the intention to look into more details of what is actually going on inside the network. This analyzer provides additional information about sniffing which users may find helpful. Packet Analyzer also detects network misuse by internal and external users and also handles the documenting regulatory compliance through logging all perimeter and endpoint traffic. ARP Cache is the technique by which an attacker maliciously modifies the mapping of an IP address to its corresponding MAC address in the ARP cache of another host. This is a man-in-the-middle (MiM) attack by which the attacker can divert the traffic passing between two machines to pass via him. The main task considered in this paper work is the analysis of network based on ARP Spoofing. Sometimes it is possible observe/record traffic traveling on a network. Network traffic may contain valuable information like Usernames and passwords, E-mail, web requests (and replies), data files. So ARP poisoning can allow an attacker to capture and modify network traffic as a man-in-the-middle. The packet information which is traveling on the network is to be watched carefully this the main aim of in this paper.

III. WORKING

ARP poisoning is a method used for manipulating the flow of traffic between arbitrary hosts on a local area network. Exploiting a network with an ARP poisoning attack allows an attacker to reroute traffic passing between workstations and servers on the LAN through a malicious node, where the traffic can be monitored, modified, or DoSed by the attacker. At the highest level, ARP poisoning works by modifying the ARP tables – small databases linking MAC hardware addresses to IP addresses – in target machines by exploiting fundamental weaknesses in the way network drivers handle ARP traffic. Because local area networks are the smallest unit of network infrastructure, the rules for passing data between computers vary from the commonly known TCP/IP and DNS structure used on the Internet. On the LAN, packets are exchanged using physical MAC addresses as a base network identifier rather then IP addresses. MAC addresses of all hosts in the subnet are mapped to IP addresses by the ARP protocol, which relies on a decentralized infrastructure in which each network node maintains its own table of MAC and IP addresses. No central server maintains an authoritative list, a role played by DNS servers in the domain name system. Because each node maintains its own mapping table of MAC and IP addresses, network drivers must be very proactive in requesting and extracting routing information from the network in order to maintain an accurate

ARP table. Large volumes of ARP request packets are sent across the wire via broadcast, each requesting that the owner of a particular IP address inform the requester of its existence and MAC hardware address. When a node sees a MAC request targeted at it's IP address, it responds with an ARP reply packet containing its current MAC. The requesting machine will then cache the IP address, and the MAC sent in the reply packet, in its ARP table.

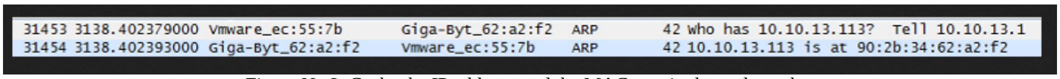


Figure No 2. Cache the IP address, and the MAC sent in the reply packet

Now, we can discuss an inherent weakness in the ARP protocol that allow a malicious attacker to modify the ARP table in any node on the LAN. Most mainstream operating systems, as revealed by our research, extract and use information received from unsolicited ARP replies. Unsolicited ARP replies are ARP reply packets received by a machine that the machine never asked for – AKA, an ARP response was never sent to the node the ARP reply is coming from. This allows a hacker to forge an ARP reply in which the IP address and MAC address fields can be set to any values. The victim receiving this forged packet will accept the reply, and load the MAC/IP pair contained in the packet into the victim's ARP table. If a legitimate MAC address entry exists in the ARP table for that IP address, it will be overwritten by the MAC address from the attacker's forged ARP reply. After the attacker's MAC address is injected into a poisoned ARP table, any traffic sent to that IP address will actually be routed to the attacker's hardware instead of the real owner of the IP.

Figure No 3. Unpoisoned ARP Table

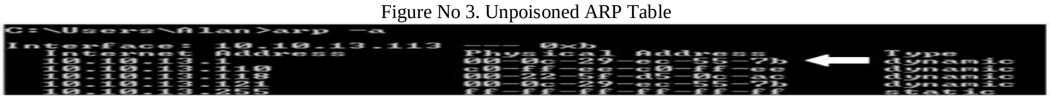


Figure No 4. Poisoned ARP Table.

By modifying the MAC address associated with an IP address in the target computer's ARP table, an attacker can trick them into sending data that should be routed to the targeted IP address to the MAC address of the hacker's machine. The attacker can then read, and even modify, the data before seamlessly forwarding it on to the intended destination. Using this method, a transparent Man In The Middle (MITM) attack can be carried out, with no apparent symptoms to the victim.

The Spoof Detection Engine

The Spoof Detection Engine is the heart of the whole system. The three different ARP Cycle packets as discussed in Section 2.2 are treated in slightly different ways by the Spoof Detection Engine to detect an attempted spoofing. The Spoof Detection Engine works based on the following Rules: Rule A: "The network interface card of a host will accept packets sent to its MAC address, Broadcast address and subscribed multicast addresses. It will pass on these packets to the IP layer. The IP layer will only accept IP packets addressed to its IP address(s) and will silently discard the rest of the packets. If the accepted packet is a TCP packet it is passed on to the TCP layer. If a TCP SYN packet is received then the host will either respond back with a TCP SYN/ACK packet if the destination port is open or with a TCP RST packet if the port is closed". Rule B: "The attacker can spoof ARP packets impersonating a host but he can never stop the real host from replying to ARP requests (or any other packet) sent to it. The valid assumption here is that the real host is up on the network." It should be noted that these rules have been derived from the correct behavior that a host's network stack should exhibit when it receives a packet. To exemplify Rule A, let a host have MAC address = X and IP address = Y. If this host receives a packet with destination MAC address = X and destination IP address = Z then even though the network interface card would accept the packet as the destination MAC address matches, the host's network stack will silently discard this packet as the destination IP address does not match, without sending any error messages back to the source of the packet. Based on Rule A, we can conceive of two types of probe packets from a host's network stack point of view which we will use to detect ARP spoofing. Right MAC – Wrong IP packet: The destination MAC address in the packet is of the host but the IP address is invalid and does not correspond to any of the host's addresses. The destination host will silently drop this packet. Right MAC – Right IP packet: The destination MAC address and IP addresses pairs are of the host's and its network stack accepts it.

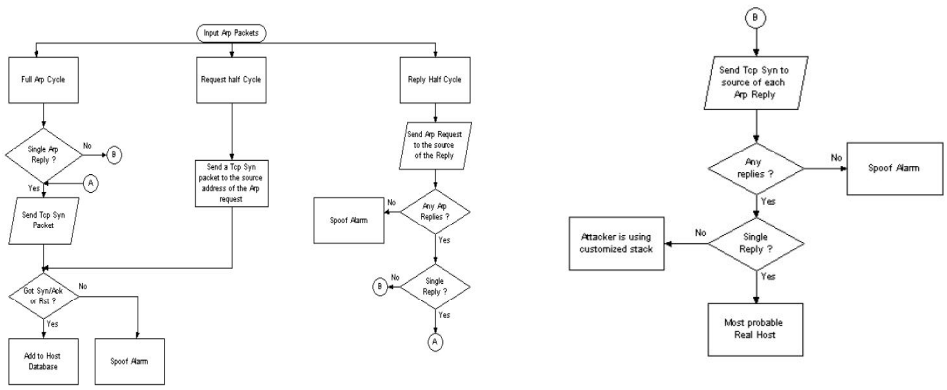


Figure No 5. Flow Chart Representation of the Spoof Detection Engine

Note that though we can detect ARP spoofing even in the presence of an attacker aware of our methods and using a customized stack we cannot predict the correct MAC to IP address mapping. This is the only limitation of our method in the presence of a customized stack.

IV. COMPARISON

The basic principle behind ARP spoofing is to exploit the above-mentioned vulnerabilities in the ARP protocol by sending spoofed ARP messages onto the LAN. ARP spoofing attacks can be run from a compromised host on the LAN, or from an attacker's machine that is connected directly to the target LAN. Generally, the goal of the attack is to associate the attacker's MAC address with the IP address of a target host, so that any traffic meant for the target host will be sent to the attacker's MAC instead. The attacker could then choose to: Inspect the packets, and forward the traffic to the actual default gateway (interception), Modify the data before forwarding it (man-in-the-middle attack).Launch a denial-of-service attack by causing some or all of the packets on the network to be dropped The techniques that are used in ARP spoofing can also be used to implement redundancy of network services. For example, some software allows a backup server to issue a gratuitous ARP request in order to take over for a defective server and transparently offer redundancy. ARP spoofing is often used by developers to debug IP traffic between two hosts when a switch is in use.

Defense	Spoofing
• DefendARP: A host-based ARP table monitoring and defense tool designed for use when connecting to public wifi. DefendARP detects ARP poisoning attacks, corrects the poisoned entry, and identifies the MAC and IP address of the attacker. • anti-arpspoof • Arpwatch • ArpON: Portable handler daemon for securing ARP against spoofing, cache poisoning or poison routing attacks in static, dynamic and hybrid networks. • Antidote: Linux daemon, monitors mappings, unusually large number of ARP packets. • Arp_Antidote: Linux Kernel Patch for 2.4.18 - 2.4.20, watches mappings, can define action to take when. • Arpalert: Predefined list of allowed MAC addresses, alert if MAC that is not in list. • Arpwatch/ArpwatchNG/Winarpwatch: Keep mappings of IP-	Some of the tools that can be used to carry out ARP spoofing attacks: • Arpspoof (part of the DSniff suite of tools) • Arpoison • Subterfuge • Ettercap • Seringe • ARP-FILLUP -V0.1 • arp-sk -v0.0.15 • ARPOc -v1.13 • arpalert -v0.3.2 • arping -v2.04 • arpmitm -v0.2 • arpoison -v0.5 • ArpSpyX -v1.1

MAC pairs, report changes via Syslog, Email. • Prelude IDS: ArpSpoof plugin, basic checks on addresses. • Snort: Snort preprocessor Arpspoof, performs basic checks on addresses • XArp: Advanced ARP spoofing detection, active probing and passive checks. Two user interfaces: normal view with predefined security levels, pro view with per-interface configuration of detection modules and active validation. Windows and Linux, GUI-based.	• ArpToXin -v 1.0 • SwitchSniffer • APE - ARP Poisoning Engine • Simsang

Table No 1. Tools

Name	OS	GUI	Free	Protection	Per interface	Active/passive
Agnitum Outpost Firewall	Windows	Yes	No	Yes	No	Passive
AntiARP	Windows	Yes	No	Yes	No	Active+passive
Antidote	Linux	No	Yes	No	?	Passive
Arp_Antidote	Linux	No	Yes	No	?	Passive
Arpalert	Linux	No	Yes	No	Yes	Passive
ArpON	Linux/Mac/BSD	No	Yes	Yes	Yes	Active+passive
ArpGuard	Mac	Yes	No	Yes	Yes	Active+passive
ArpStar	Linux	No	Yes	Yes	?	Passive
ArpGuard	Mac	Yes	No	Yes	Yes	Active+passive
ArpStar	Linux	No	Yes	Yes	?	Passive
Arpwatch	Linux	No	Yes	No	Yes	Passive
ArpwatchNG	Linux	No	Yes	No	No	Passive
Colasoft Capsa	Windows	Yes	No	No	Yes	No detection, only analysis with manual inspection
Prelude IDS	?	?	?	?	?	?
remarp	Linux	No	Yes	No	No	Passive
Snort	Windows/Linux	No	Yes	No	Yes	Passive
Winarpwatch	Windows	No	Yes	No	No	Passive
XArp	Windows, Linux	Yes	Yes (+pro version)	Yes (Linux, pro)	Yes	Active + passive
Seconfig XP	Windows 2000/XP/2003 only	Yes	Yes	Yes	No	Only activates protection built-in some versions of Windows

Table No 2. Detail of Spoofing

The techniques an attacker would use on wireless networks. Regardless of the protocols, wireless networks will remain potentially insecure because an attacker can listen in without gaining physical access. In addition, the protocol designs were security-naïve. We have pointed out several existing tools that implement attack techniques that exploit the weaknesses in the protocol designs. The integration of wireless networks into existing networks also has been carelessly done. We pointed out several best practices that can mitigate the insecurities.

IV. IMPLEMENTATION

The attacker machine makes use of the stored ARP cache table to re-route or re-direct packets from a target, to an attacker machine, and then forward to the host, thus the attacker machine "sees" all traffic between target and host. First the target MAC address is established, and then the ARP Poison Routing feature "poisons" the cache of the target by forcing a cache update with the path re-routed so that the attacker machine forwards traffic to and from host and target. The attacker machine can also observe packets with a Sniffer such as Wireshark, Cain and Abel.

V. RESULT

We have also studied the impact on performance and we tested the round trip time (RTT), from when an ARP request is sent to when a corresponding ARP reply is received. The RTT was measured when the router is running as well as when the router is not running The performance test was done by sending 1000 pings from a host A to a host B. The ping causes the host A to issue an ARP request to B and receive an ARP reply from B. After each ping command the ARP cache of A was cleared. From an Ethereal trace running in A, we were able to measure the RTT of an ARP request-reply pair. So Network performance does decrease, Meaning if you try to poison all hosts to capture all the traffic between the switch and the hosts you will get a network performance decrease cause your comp has to reroute all the traffic and that depends on the computer speed and the upload speed you have .It seems easy when done with the right tools but the possibility to create a DOS on the network still exists. Taking the example that you poison all the host and supposing your computer can't handle all that traffic it causes a DOS on the network. The impact of this kind of attack is pretty big. Just imagine several cases where the network was designed where they thought only of immediate network risks and not about who is being next to who. That could create situations where normal employee/student is next to some one who sends sensitive information over the wire. This could cause severe loss of money or just pride. One more thing, we also tried using the switch as the start point of poisoning. And that didn't work out well. It seems it caused the network to fall apart. Because no one was able to ping or do anything on the network. So this caused in our test lab a DOS attack on the network.

VI. CONCLUSIONS AND FUTURE WORK

We have described the problem of ARP spoofing in networks. We have shown how the inherently secure wired clients are vulnerable to an attack from the wireless clients due to the nature of the setup of wireless-cum-wired networks within network devices like wireless access points or wireless routers. We have proposed a scheme to prevent the ARP cache poisoning attack from within these network devices. The proposed scheme is a feasible approach for preventing ARP cache poisoning, as it does not require any modification to the ARP protocol itself. Any modification to the ARP protocol would require the TCP/IP stack of all the hosts that need protection to be changed. The scheme also does not add any significant overhead while sending and while receiving an ARP packet. The checking of the ARP packet is only done once in the Access Point and not at both the sending and receiving sides. The proposed scheme opens up various possibilities for enhancement. The above setup does not support hosts that have static IP addresses. When a wireless client wants to join a wireless network, it has to first associate with the Access Point. The Access Point at this time can learn the static IP address to MAC address mapping of the wireless client and store it in the mapping table. In future, based on the outcome of this model, explore further to find ways to eliminate those identified multiple adversaries, from the wireless network. Thus way, wireless networks will be more robust and less prone to attack.

REFERENCES

[1] Amala Gracy, Chinnappan Jayakumar, " Identifying And Locating Multiple Spoofing Attackers Using Clustering In Wireless Network.", International Journal Of Wireless Communications And Mobile Computing. Vol. 1, No. 4, 2013, Pp. 82-90.

[2] Amala Gracy , Chinnappan Jayakumar ,"Identifying And Locating Multiple Spoofing Attackers Using Clustering In Wireless Network", International Journal Of Wireless Communications And Mobile Computing 2013; 1(4): 82-90

[3] Amit Kumar Tyagi, Surendra Kumar Tyagi, Prafull Kumar Singh," A Novel Approach to Detectand DefenceagainstAddress Resolution Protocol (ARP) Spoofing Attack", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 2, February 2014.

[4] Brustoloni, J. C. "Laboratory Experiments For Network Security Instruction," Acm Journal On Educational Resources In Computing, Vol. 6, No. 4, 2006.

[5] Brushi, D., Ornaghi, A., Rosti, E., " S-Arp: A Secure Address Resolution Protocol", Proceedings Of The 19th Annual Computer Security Applications Conference 2003.

[6] Bob Fleck And Jordan Dimov, "Wireless Access Points And Arp Poisoning: Wireless Vulnerabilities That Expose The Wired Network," October 2001.Retrieved On Jan 20, 2004.

[7] Bruce Potter And Bob Fleck, "802.11 Security, O'reilly & Associates",Isbn: 0-596-00290-4, 2002.

[8] Chris Hurley, Michael Puchol, Russ Rogers, And Frank Thornton, "Wardriving: Drive, Detect, Defend, A Guide To Wireless Security ", Isbn: 1931836035, Syngress, 2004.

[9] Gast, M. "802.11 Wireless Networks", The Definitive Guide. O'reilly Publishing, April 2002

[10] Hill, J. M. D., Carver, C. A., Humphries, J. W. And Pooch U. W. "Using An Isolated Network Laboratory To Teach Advanced Networks And Security," Proceedings Of Sigcse'01 - The 32th Technical Symposium On Computer Science Education, 2001, Pp. 36 – 40.

[11] Iyad Aldasouqi And Walid Salameh, "Detecting And Localizing Wireless Network Attacks Techniques", International Journal Of Computer Science And Security, Volume 4, No.1, 2010, Pp.82-97.

[12] John Bellardo And Stefan Savage, "802.11 Denial-Of-Service Attacks: Real Vulnerabilities And Practical Solutions", 2003, Usenix 2003 Proceedings. Retrieved Jan 20, 2004.

[13] Jon Edney And William A. Arbaugh, "Real 802.11 Security: Wi-Fi Protected Access And 802.11i", 480 Pages, Addison Wesley, 2003, Isbn: 0-321-13620-9

[14] Jamil Farshchi, "Wireless Intrusion Detection Systems ", November 5, 2003, Retrieved Jan 20, 2004.

[15] Joshua Wright, "Detecting Wireless Lan Mac Address Spoofing", Retrieved On Jan 20, 2004.

[16] Jie Yang, Yingying Chen, Wade Trappe, "Detecting Spoofing Attacks In Mobile Wireless Environments", Proceedings Of 6th Annual Ieee Communications Society Conference On Sensor, Mesh And Ad Hoc Communications And Networks, 2009, Pp.107-189.

[17] Jayakumar C And Chellappan C, "A Qos Aware Energy Efficient Routing Protocol For Wireless Ad-Hoc Networks", Asian Journal Of Information Technology, Vol.4, No.6, 2005, Pp.578-582.
[18] Karmel A And C. Jayakumar, "Analysis Of Manet Routing Protocols Based On Traffic Type", Ijreat International Journal Of Research In Engineering & Advanced Technology, Vol.1, Issue 1, 2013, Pp.1-4.
[19] K. Tan, Guanling Chen, D. Kotz, A Campbell, "Detecting 802.11 Mac Layer Spoofing Using Received Signal Strength", Proceedings Of 27th Conference On Computer Communications, 2008, Pp.1768-1776.
[20] Lakshmi M. And Sankaranarayanan P.E., "Performance Analysis Of Three Routing Protocols In Wireless Ad Hoc Networks", Information Technology Journal 5 (1): 114-120, © 2006 Asian Network For Scientific Information.
[21] Robert Moskowitz, "Debunking The Myth Of Ssid Hiding", Retrieved On March 10, 2004.
[22] Roney Philip," Securing Wireless Networks from ARP Cache Poisoning", Department of Computer Science San Jose State University, May 2007.
[23] Sundaram Rajagopalan And Chein-Chung Shen, "What Does Using Tcp As An Evaluation Tool Reveal About Manet Routing Protocols?", Iwcmc'06 © 2006 Acm Journal.
[24] Silky Manwani," ARP Cache Poisoning Detection and Prevention", Department of Computer Science San Jose State University ,Dec 2003.
[25] Tom Karygiannis And Les Owens, Wireless Network Security: 802.11, Bluetooth And Handheld Devices, National Institute Of Standards And Technology Special Publication 800-48, November 2002. Retrieved Jan 20, 2004
[26] T. Mahesh, "Middleware Approach To Asynchronous And Backward Compatible Detection And Prevention Of Arp Cache Poisoning".
[27] Tripunithara, M.V., Dutta, P. (1999). A Middleware Approach To Asynchronous And Backward Compatible Detection And Prevention Of Arp Cache Poisoning, 15th Annual Computer Security Applications Conference (Acsac '99), 303.
[28] Vikram Gupta, Srikanth Krishnamurthy, And Michalis Faloutsos, "Denial Of Service Attacks At The Mac Layer In Wireless Ad Hoc Networks", Proceedings Of 2002 Milcom Conference, Anaheim, Ca, October 2002.
[29] William Stallings, "Wireless Communications & Networks", Prentice Hall, 2001, Isbn: 0130408646
[30] Wagner, P. J. And Wudi, J. M. "Designing And Implementing A Cyberwar Laboratory Exercise For A Computer Security Course," Proceedings Of Sigcse'04 - The 35th Technical Symposium On Computer Science Education, 2004, Pp. 402 – 406.
[31] Yuan, X. Wright, O. T., Yu, H., Williams, K. "Laboratory Design For Wireless Network Attacks", Proceedings Of The 2008 Information Security Curriculum Development Co