Aiming at Higher Network Security Through Extensive Penetration Tests

A. Bechtsoudis and N. Sklavos, IEEE Members

***Abstract*— Modern enterprise infrastructures adopt multilayer network architectures and heterogeneous server environments in order to efficiently fulfill each organization's goals and objectives. These complex network architectures have resulted in increased demands of information security measures. Each organization needs to effectively deal with this major security concerns, forming a security policy according to its requirements and objectives. An efficient security policy must be proactive in order to provide sufficient defense layers against a variety of known and unknown attack classes and cases. This proactive approach is usually interpreted wrongly in only up-to-date software and hardware. Regular updates are necessary, although, not enough, because potential mis-configurations and design flaws cannot be located and patched, making the whole network vulnerable to attackers. In this paper we present how a comprehensive security level can be reached through extensive Penetration Tests (Ethical Hacking). We present a Penetration Test methodology and framework capable to expose possible exploitable vulnerabilities in every network layer. Additionally, we conducted an extensive analysis of a network penetration test case study against a network simulation lab setup, exposing common network mis-configurations and their security implications to the whole network and its users.**

***Keywords*— penetration testing, network security, ethical hacking, proactive security policy.**

EXTENDED ABSTRACT

TODAY'S leading enterprises utilize state of the art ICT integrated solutions and technologies into their business operational processes, in an attempt to obtain the largest market share, locally or internationally. On the other hand, trailing and middle scale organizations cannot afford such costs resulting into partially adopting a subset of these high end ICT features. Despite their different levels of ICT integration, every modern organization has to effectively deal with the security issues that arise from these technologies [1].

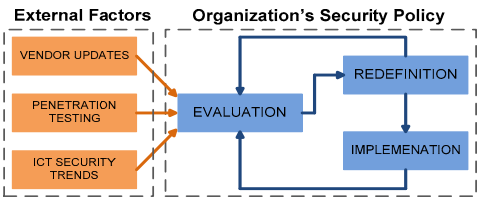
Multilayer network architectures, scalable web services, custom applications, distributed services and heterogeneous server platform environments, form a small sample of the infrastructure's complexity in modern organizations. These complex architectures in the core network infrastructure, result in large and more difficult than ever security demands in order to keep data and information assets secure. Additionally to this recently added system and network complexity, criminal organizations have formulated their hacking procedures in a try to break into corporate networks and harm the organization with every possible way [2].

Most companies and institutes work diligently to maintain an effective security policy, implementing the latest products and services to prevent fraud, sabotage, information leakage, vandalism and denial of service attacks. However this proactive up-to-date approach does not result in a successful security policy. The problem is that they still do not know whether and where they are vulnerable. They just take it on faith that the vendors' fixes will keep their network safe.

Unfortunately, the up-to-date security approach is not adequate because it does not detect mis-configured settings or network infrastructure design flaws that can put the network under great risk. An organization that truly wants to adopt a proactive approach, aggressively seeks out all types of vulnerabilities by using relevant methods with the actual hackers. This process of systematically and actively testing a deployed network to determine potential vulnerabilities is called Penetration Testing, and is also known as Ethical Hacking [4,5]. A network penetration test is conducted using specific tools and processes to scan the network for vulnerabilities and discover exploitation mechanisms taking advantage of the discovered security holes. These exhaustive tests can be conducted either by the organization's internal IT security department or by an external certified penetration testing and security auditing organization.

Each organization's management must continuously seek for the maximum information input and reevaluate their security policy in an endless loop, as shown in *Fig.1*. This approach will form a truly proactive security policy which is carefully redefined in a regular basis, taking into account every possible parameter (social, technical, environmental) might affect it [3].

The remainder of the paper is organized as follows. Section II discusses network attack taxonomy, by dividing the threats into classes according to their operational model. We present the proposed penetration testing methodology and working framework in Section III. In Section IV we analyze the case study scenario and the lab setup where the penetration test was conducted. Test results and its effects in contrast to a real network setup are shown in Section V. Finally, Section VI summarizes and concludes this paper.



A. Bechtsoudis, Computer Engineering & Informatics Dept, University of Patras, HELLAS, abechtsoudis@ieee.org
N. Sklavos, KNOSSOSnet Research Group, Informatics & MM Dept, Technological Educational Institute of Patras, HELLAS, nsklavos@ieee.org

Figure 1. Interaction between security policy and input from external factors.

Network security threats have been a problem since the birth of small networks with only a few hosts communicating over it [7]. Till 1998 reported incidents did not exceed the limit of 4000 cases per year. Since 1999 there has been a marked increase in the number of incidents reported from the Computer Emergency Response Team Coordination Center (CERT/CC) [6]. In addition to the increase in the amount of the incidents, the sophistication of the attacks has also increased. High skilled hackers every day evaluate large amounts of source codes and operational frameworks to discover new vulnerabilities that may lead to a potential break-in or some sort of security bridge. New attack vectors and malicious payloads are arising and spread quickly day by day, forcing organizations to regularly revise their security policy and apply the relative vendor patches.

Additionally to the sophisticated network attacks carried out by skilled professionals, the spread of the Internet has led to a special class of "user-friendly" attacks, where the user does not need advanced technical knowledge and skills to launch such an attack. This has led to the rise of various groups of hackers, known as "script-kiddies", who while ignorant of how their attack works, can cause great damage.

Each attacker, either belonging to a professional group or being a "script-kiddy", has certain goals that security researchers and ICT administrators must be familiar with. Attacker goals can be divided into four main classes: interruption, interception, modification and fabrication. An interruption attack aims to make network or system resources unavailable by carrying out large or special crafted amounts of information packets. It is an attack on availability mainly expressed by denial of service (DoS) attacks [8,9]. The second class is the interception attacks, where the attacker tries to gain unauthorized access to a network or system. A major example is a simple eavesdropping [10] on a communication channel where sensitive data are transmitted through it. Modification attacks aim to modify information that is transferred during a communication session of two or more parties. This class mainly include network spoofing attack [11] where the information source and data fields are altered pretending to originative from another source. Finally, the fourth class contains fabrication attacks which aim to bypass authenticity checks by mimicking or impersonating information.

Keeping the above attacker goals in mind, there are two main types of attacks whose aim is to compromise the security of a network – passive and active attacks. During a passive attack the attacker simply monitors the transmission between two parties and captures information that is sent and received. The attacker does not intend to interrupt the service, or cause an effect, but to only read the information. If information is encrypted or obfuscated, it will be more difficult to interpret it. Although, the attacker simply observes the data flow and tries extract useful information about the evolved parties. Passive attacks are usually harder to detect as there is little or no impact. On the other hand, an active attack aims to cause disruption, and is usually easily recognized. Unlike a passive attack, active attacks modify information, interrupt services and aim to gain unauthorized access to the network systems.

The process of penetration testing as shown in Fig. 2, can be broadly divided into four phases: planning, discovery, exploitation and reporting. Initially at the planning phase, the scope for the assignment is defined. Management approvals, documents and agreements like NDA (Non Disclosure Agreements) are signed under the guidance of responsible legal departments and lawyers. After the management consent, the penetration testing team gathers crucial input about the organization operational procedures and security policies, towards defining the scope for the test.

Following the initial planning, the actual penetration test starts with the discovery phase, also known as information gathering phase. During the information gathering process the penetration testing team launches scanning and enumeration procedures to gain as much information as possible about the target network and the participating systems and services. The gathering phase can be further divided into non-intrusive (public repositories, documents, mailing lists, web profiles etc) and intrusive (port scanning, firewall rules, matching OS fingerprints etc) inspection processes. Having adequate amount of information the testing team can profile the target network and enumerate possible exploitable vulnerabilities using relative public or personal security knowledge bases.

The third and most import phase of a penetration test is the exploitation phase. Using as input the discovered vulnerabilities arriving from the previous phase, the penetration testing team revises matching proof-of-concept exploits that may lead to a network or service security bridge. Depending on the agreement with the management and the exploitation implication level, the attacks can be launched either in an identical network simulation lab or in the actual network using adequate security prerequisites. While exploiting network vulnerabilities and mis-configurations, the testing team might discover additional information that can feedback the discovery phase, resulting in new attack scenarios and exploits. This interaction between the discovery and exploitation phases is continuous throughout the actual test.

The last phase that completes a penetration test process is the reporting phase. The report writing can begin in parallel to the other three stages, although must finish after exploitation phase has been completed. A successful report details all the findings and their impacts to the organization by taking into account both the technical and management aspects in its format. It is very important to conduct a fully detailed and well documented report in order to inform the management about the security risks and provide technical details and high level recommendations to the ICT department.

Figure 2. Penetration testing methodology diagram.

We have created a virtual lab capable to simulate enterprise networks. Using the lab setup as a real case study, we conducted a penetration test against the setup, in order to expose mis-configurations and security design flaws. The scope of the test was to simulate an internal attack, in order to expose the potential risks of a compromised host in the internal network.

Before proceeding to the attack scenario we will briefly present the lab setup.

A. Lab Setup

As shown in Fig. 3, the lab is mainly consisted of two border routers chained together under a common public channel, forming two internal networks. Each internal network has a sufficient amount of nodes, simulating backbone servers and simple host machines. In our case study we used only three nodes at the victim network, and one node at the attacker's network. Additionally, we used an extra node attached to the common channel as the attacker's operational terminal.

The routing network devices have been simulated in software using the GNS3 network simulator [12] with the latest CISCO IOS [13]. The simulated router interfaces has been bridged to the hosts' physical NICs using operating system's bridge utilities. To simulate the hosts that participate in the network topology, we used VMWare's virtualization solutions, to create virtual machines with different operating systems and functionalities. Finally, all the host machines that run the simulation and virtualization software are connected to a gigabit switch to form the common channel.

B. Scenario

During the planning phase we have concluded that we need to launch an internal penetration test to expose the network security design flaws arriving from a compromised source within the network. Consequently, we assume that we have administration level access to the "Victim PC" with the IP address *192.168.8.22*. Our goal is to intercept the server's communication sessions passing its network traffic through an external network capture host in the attacking network. It is an external man-in-the-middle attack (MITM) [18], where the server's network traffic, before routed to the common channel, is passing through a host outside the internal network using a dedicated tunneling protocol [14]. Of course we have administration access level to the attacker's network including both the router and the sniffer host.

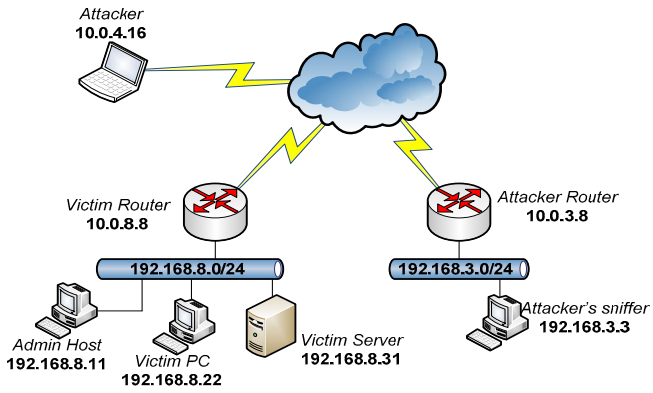


Figure 3. Simulated lab topology and initial configuration.

During our penetration test we will focus on the network setup and configuration by mainly targeting the victim router device. The vulnerability assessment of the services and applications running on the main server and hosts is beyond the scope of our penetration test case study.

C. Penetration Test Implementation

Having high privilege access to the internal victim host, we launched the actual test starting with the discovery stage where we gathered information about the internal network. The first step was to map the network using a network mapping tool to discover the IP addresses of the active nodes. Except the IP address of the host that we have access to; we discovered two more addresses, the server's and an additional host, used from the network administrator to manage the network.

Knowing the hosts that participate in the internal network, we proceeded to the establishment of a network capture utility, in a try to obtain the transferred data over the network. The internal hosts communicate under a network switch, so we were not able to obtain the network traffic except the packets with source or destination the IP address of our host. Although, we noticed that every 5 min the host was sending monitor statistics to the server using the SNMP (Simple Network Management Protocol) [15]. The SNMP community string that was used for the communication was named "public". Digging into the host's SNMP service configuration file we discovered that despite the "public" read only community string, there exist another community string named "private" with write permissions. Both community strings are protected by an ACL (Access List), allowing only the server's and administrator's host IP addresses to access them.

Routing network devices can execute configuration and administration commands derived from SNMP request packets. A very useful feature that Cisco IOS have is the ability to copy their running configuration to a remote host using TFTP (Trivial File Transfer Protocol). Of course like the host, router's SNMP community strings were protected with relevant ACLs, limiting the allowed IP addresses.

Having finished with the information gathering phase we have enough knowledge to attack the victim network. The first goal was to obtain the victim's router configuration for further analysis and modification. Knowing that SNMP is implemented in the victim network, we have exploited its weaknesses to hit our first goal.

SNMP is a connectionless UDP based protocol. All UDP based protocols extensively suffer from source address spoofing attacks arrived from UDP design principles. Knowing the IP addresses that are allowed from victim router's SNMP ACLs, we can generate and send specially crafted SNMP request packets to bypass them and force the

router to copy its configuration to a TFTP service that we have installed in the "Victim PC" that we have access.

To generate these requests we used SNMP client software, a network capture tool, a packet editing tool and a packet generator. Initially a valid SNMP request is generated asking from router to copy its configuration to our TFTP service. Of course the request was ignored by the router, because 192.168.8.22 is not allowed from the SNMP ACL. To bypass this restriction we captured the request packet and changed its IP source address from 192.168.8.22 to 192.168.8.11, impersonating the administrator's IP address. Sending this new modified packet we bypassed the ACL protection layer and forced the router to transfer its configuration file to our TFTP service directory.

Having the full configuration file, we were able to modify it adding a new GRE (Generic Routing Encapsulation) tunnel interface and alter the routing rules [16]. The tunnel's source point is at the victim router and the end point at the attacker's router. After uploading the new configuration back to the router using the same source address spoofing technique, server's network traffic will be transmitted over the GRE tunnel to the attacker's router. The router at the other point will decapsulate the traffic and forwards it to the sniffer machine before sending it back to tunnel's source. The new server's network data flow is illustrated in Fig. 4.

Our penetration test against the lab setup revealed great security design flaws in the core network. Having access only to an internal host with no administration privileges in the network, we managed to bypass router's defense layers and compromise the network. When we started the test, no prior knowledge about the internal network setup has been taken as input, simulating the behavior and characteristics of a real network attack.

After successfully crafting the initially planned MITM attack through the GRE tunnel, every user's interaction with the backbone server is captured by the sniffing machine and can be further analyzed and used for malicious purposes. Network administrators must fully reevaluate the network security policy, by implementing anti-spoofing shields [11] and find ways to overcome protocol native security weaknesses [17].

The penetration test case study that we have implemented effectively pivot through discovered network configuration

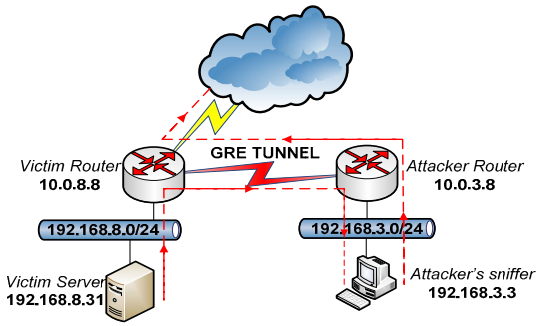


Figure 4. Network configuration and data flow after exploitation phase.

parameters to achieve its goal. An effective security policy must limit at the lowest possible this information leakage. Additionally, system and service configurations must be carefully revised in order to implement only the necessary features, preventing critical information exposures.

We have analyzed a penetration testing case study against a simulated network setup. Network devices and hosts participating in the network were updated with the latest vendor releases. Despite this up-to-date security policy, we have managed to compromise the internal network, taking advantage of mis-configurations and security design flaws.

ICT security issues, concerns and trends are rapidly evolving, posing a major challenge to the organizations' business operations. Our results provide great evidence that regular penetration tests must be conducted to the organization's network. Vendor updates are necessary although not enough for a proactive and efficient security policy.

REFERENCES

[1] Khidzir, N.Z., Mohamed, A. and Arshad, N.H.H., "Information Security Risk Management: An Empirical Study on the Difficulties and Practices in ICT Outsourcing", NETAPPS 2010.
[2] Kshetri, N., "The simple economics of cybercrimes", IEEE Security and Privacy (2006), Volume: 4, Issue: 1.
[3] Kotenko, I. and Bogdanov, V., "Proactive monitoring of security policy accomplishment in computer networks", IDAACS 2009.
[4] Hamisi, N.Y., Mvungi, N.H., Mfinanga, D.A. and Mwinyiwiwa, B.M.M., "Intrusion detection by penetration test in an organization network", ICAST 2009.
[5] Bishop, M., "About Penetration Testing", IEEE Security and Privacy (2007), Volume: 5, Issue: 6.
[6] CERT Coordination Center Statistics, "http://www.cert.org/stats".
[7] S. Hansman and R. Hunt, "A taxonomy of network and computer attacks", Computers Security (2005), Volume: 24, Issue: 1, Publisher: Elsevier, Pages: 31-43.
[8] Long, M., Chwan-Hwa Wu, Hung and J.Y., "Denial of service attacks on network-based control systems: impact and mitigation", IEEE Transactions on Industrial Informatics (2005), Volume: 1, Issue: 2.
[9] Meadows, C, "A formal framework and evaluation method for network denial of service", Computer Security Foundations Workshop, 1999.
[10] Ansari, S., Rajeev, S.G. and Chandrashekar, H.S., "Packet sniffing: a brief introduction", IEEE Potentials (2003), Volume: 21, Issue: 5.
[11] Ishibashi, H., Yamai, N., Abe, K. and Matsuura, T., "A protection method against unauthorized access and address spoofing for open network access systems", IEEE Pacific Rim Conference on Communication and Signal Processing, 2001.
[12] Graphical Network Simulator (GNS), "http://www.gns3.net".
[13] CISCO Internetwork Operating System (IOS), "http://www.cisco.com/warp/cpropub/45/tutorial.htm".
[14] Savochkin, A.A. and Gorokhovtsev, N.E., "Features of tunneling in IP/MPLS transport networks", CriMiCo 2009.
[15] Schonwalder, J., Pras, A., Harvan, M., Schippers, J. and Van de Meent, R., "SNMP Traffic Analysis: Approaches, Tools, and First Results", IEEE Integrated Network Management, 2007.
[16] Generic Routing Encapsulation (GRE) Protocol, "http://www.ietf.org/rfc/rfc2784.txt".
[17] F. Lan, W. Chunlei and M. Guoqing, "A framework for network security situation awareness based on knowledge discovery", ICCET 2010.
[18] Nath Nayak, G. and Ghosh Samaddar, S., "Different flavours of Man-In-The-Middle attack, consequences and feasible solutions", ICCSIT 2010.

Anestis Bechtsoudis is an undergraduate student at Computer Engineering & Informatics Dept., University of Patras, Hellas. Since 2009 Anestis is

working as Network & Security Administrator at Laboratory for Computing at the same university. His research interests include information & communications security, applied cryptography and computer networks. He is also serving as a reviewer at Computers & Electrical Engineering editorial-board and is an active IEEE member at Computer Society and student branches.

Nicolas Sklavos received the Ph.D. Degree in Electrical & Computer Engineering, and the Diploma in Electrical & Computer Engineering, in 2004 and in 2000 respectively, both from the Electrical & Computer Engineering Dept., University of Patras, Hellas. Since 2008, he is an Assistant Professor with the Informatics & MM Dept, Technological Educational Institute of Patras, Hellas. He is also adjunct faculty, Assistant Professor, with the Computer Engineering & Informatics Dept., University of Patras, Hellas from 2007. He holds an award for his PhD thesis on "VLSI Designs of Wireless Communications Security Systems", from IFIP VLSI SOC 2003. His research interests include Cryptographic Engineering, System on Chip Design, Computers Architecture, VLSI Design, Security of Computers and Networks. N. Sklavos has participated to a great number of European and National projects both research & development, in the areas of his research. He serves as evaluator of both European Commission Projects (FP7) and General Secretary of Research and Development, Hellas. He is director of KNOSSOSnet Research Group. Since 2007, he is the Chair of IEEE Hellas GOLD Affinity Group. He is the Editor-in-Chief for the Information Security Journal: A Global Perspective Journal, Taylor & Francis Group. He serves as Associate Editor for IEEE Latin America Transactions, IEEE Press, and Computers & Electrical Engineering Journal, Elsevier. He has been Guest Editor of Special Issues for Elsevier & Springer publishers. He was the General Co-Chair of ACM MobiMedia 2007 and General Chair of ATHENA 2011 Summer School. He has participated to the organization of more than 100 conferences organized by IEEE/ACM/IFIP, as Publicity, Publication Chair, Program Chair and Program Committee member. He has authored or co-authored more than 100 scientific articles, books, chapters, tutorials, in the areas of his research. His published works has received up to 750 non-self citations.