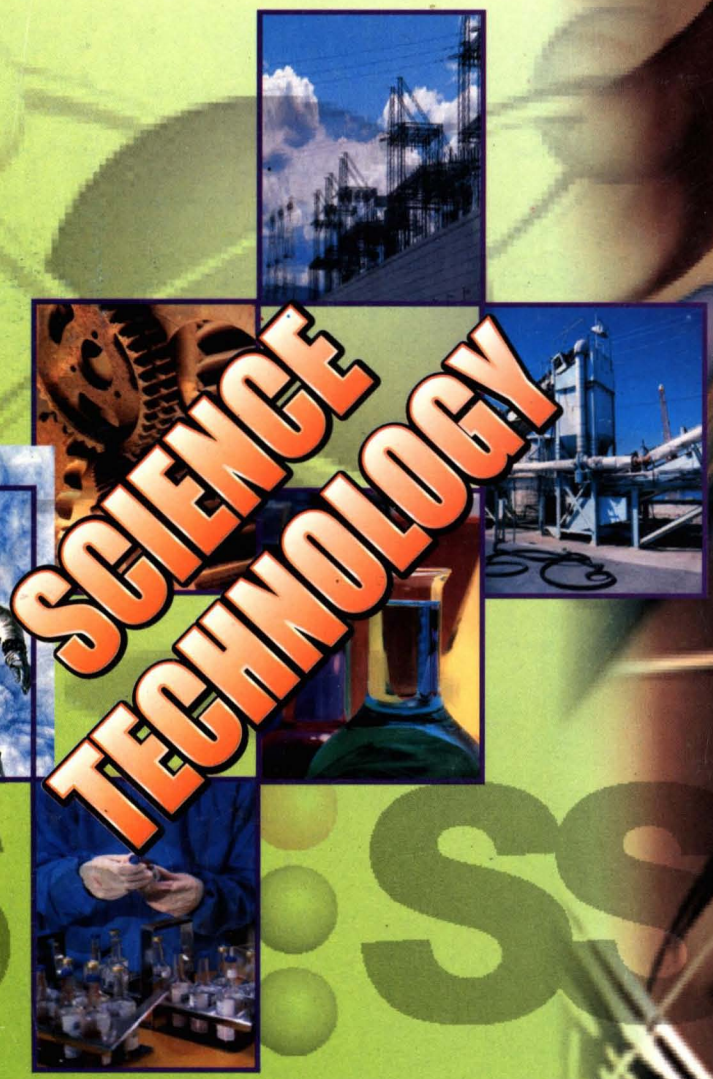
UNIVERSITI TEKNOLOGI MARA

ISO 9001:2000 No Sijil :0500132

Globalising Knowledge and Information



NATIONAL SEMINAR ON SCIENCE TECHNOLOGY & SOCIAL SCIENCES 2006

30-31 May 2006

Swiss Garden Resort & Spa

Kuantan, Pahang

Port Knocking

Saadiah Yahya
Mohamed Sulaiman Sultan Suhaibuddeen

ABSTRACT

Around the globe, network administrators are challenged to balance flexibility and security elements when designing and maintaining their network infrastructure. Firewalls are a long-standing basic security measure that organizations use to isolate networks from the Internet. Whether it's a stand-alone appliance firewall like CheckPoint, one of the various host-based systems such as ZoneAlarm, or the Windows Firewall system included with Windows XP Service Pack 2, these devices go a long way toward protecting networks from unwanted traffic, including viruses, Trojans, and hackers. A firewall should provide some form of shield against malfeasant motives by adding an extra layer of network security allowing trusted and authorized users to connect through. Unfortunately, it is not as easy as it sounds; it is a tough task to come up with a mechanism to distinguish the bad guys, because filtering on the basis of IP addresses and ports does not differentiate connecting users. Bad guys possible and do come from trusted IP addresses. On the other hand, open ports remain a consciously known vulnerability. Building up a very secure rule sets and policies alone appear insufficient. Port Knocking a method of establishing a connection to a secured network or computer within a network that does not have an open port is the answer. A remote device sends a series of series of connection attempts, in the form of packets, to the computer's closed ports, and the attempts are silently ignored but logged by the firewall. When the remote device has established the predetermined sequence of port connection attempts, a daemon triggers a port to open, and the network connection is established. An advantage of using a port knocking technique is that a malicious hacker cannot detect if a device is listening for port knocks. Having port knocking alone to handle a high secure site is not sufficient, therefore, combining it with another technology called "Wake-On-LAN" (WOL) definitely will synergize the outcome. This research is investigating the Port Knocking technology on the protected system and determines whether interaction between Wake-On-LAN and Port Knocking can offer a better-synergized security system. The effectiveness and practicality of having another layer of protection for the server in DMZ area using this method on the firewall, and more broadly measures the performance and resources usage impact of the server involved in the study is exploited. This include investigating the way how the firewall have the intelligence to avoid log rotate issues, how the firewall have the intelligence to tell whether the server is already alive or idle, and also the firewall know when is the correct time to drop all the remote connection to the server and later suspend the server.

Keywords: *Port Knocking, Wake-On-LAN, DMZ and Firewall*

Introduction

In couple of years back, Martin Krzywinski had been aggressively conducted proof-of-concept with a technique called Port Knocking. This method is an awesome idea and the implementation of this technology into the production environment is still in preliminary stage where currently it has many areas to be developed and enhanced.

Webopedia define Port Knocking as "A method of establishing a connection to a secured network or computer within a network that does not have an open port. A remote device sends a series of series of connection attempts, in the form of packets, to the computer's closed ports, and the attempts are silently ignored but logged by the firewall. When the remote device has established the predetermined sequence of port connection attempts, a daemon triggers a port to open, and the network connection is established. This security method is analogous to knowing a 'secret knock', and only people who know the proper knock sequence will be allowed access. An advantage of using a port knocking technique is that a malicious hacker cannot detect if a device is listening for port knocks."

Bruce Schneier an expert in cryptography endorsed the magnitude of Port Knocking in his article which says "It's a clever idea and one that could easily be built into VPN systems and the like. Network administrators could create unique knocks for their networks by issuing family keys, and only give them to authorized users. It's no substitute for good access control, but it's a nice addition. And it's an addition that's invisible to those who don't know about it"

Having port knocking alone to handle a high secure site is not sufficient, therefore combining it with another technology called "Wake-On-LAN" (WOL) definitely will synergize the outcome. WOL allows dormant computer in a network to be powered-on by utilizing a network packet. WOL is an Intel Wired for Management System product, as it was developed as part of the IBM and Intel Advanced Manageability Alliance. Wake-On-LAN is triggered by a so-called "Magic Packet" (trademark of Advanced Micro Devices), which is an Ethernet packet with a defined content - usually the bytes FFFFFFFFFFFF followed sixteen repeats of the target's MAC address, possibly followed by a four or six byte password. The content can be encapsulated in any type of packet (e.g. IP, IPX). Some studies

because the packet can easily constructed, therefore some of these implementation use only certain protocol such as ICMP or even crafting the packet.

Port Knocking With Finger Printing
It is good to narrow down on which client able to connect to the system by identifying their operating system, but again, this will also cap the ability of the authorized user to login into the system. The main problem for this is the maintenance issue where the security engineer need to actively add various trusted fingerprint from time to time and the idea of narrowing the scope will become redundant.

Port Knocking With Web Service
The idea to initiate knock using sequence of webpage is is not a wise idea, because, when running web service, we need to open 80. At the beginning stage, there are already a hole in the system, a hacker who can hack through the web service, can read the further instruction and get connected to the system that going to be protected.

Summary of Literature Review

By looking closely all the implementation mentioned, the term Port Knocking is only lingering about mending firewall rules, which is getting the client authenticated and allow a connection from the client to the firewall. It does not go a step forward than that; there is no implementation which actually have an artificial intelligence to reach the servers behind the firewall. The research on this paper had taken a step further which is to implant an artificial intelligence features to the implementation.

Typical Firewall Methodology

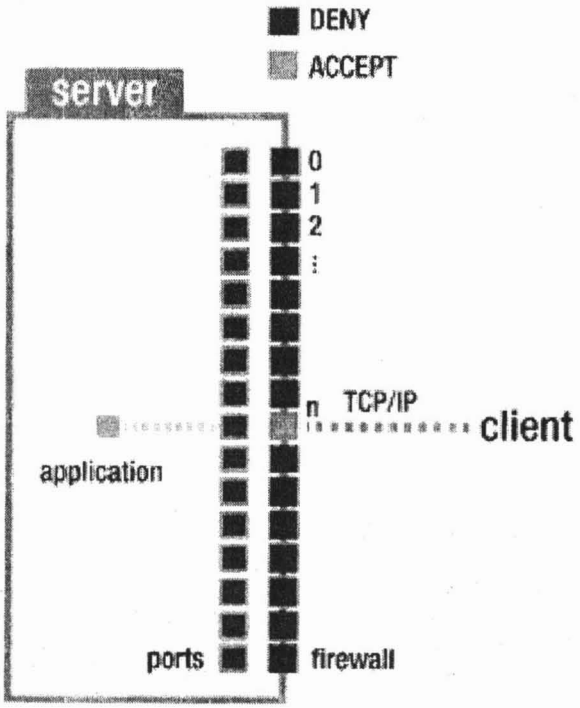


Fig. 1: A Typical Firewall Methodology

The firewall acts as the mediator between the servers in DMZ and clients. Only specific port is opened and the rest are closed. Normally all the DMZ servers shall have a private IP which will be readdressed using NAT (Network Address Translator) by the firewall.

In this scenario, the application server and the firewall are turned-on and running. This typical methodology had been proven as a very basic protection to the servers because the firewall role in this scenario is not protecting the server against untrusted client connection. If any client request for the service handled by the application server, the firewall will allow the connection if the running service is requested on the permitted port.

Port Knocking Firewall Methodology

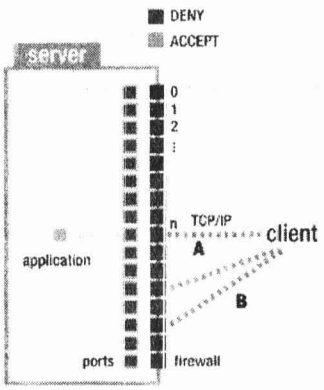


Fig. 2 : No Service Running Behind the Firewall

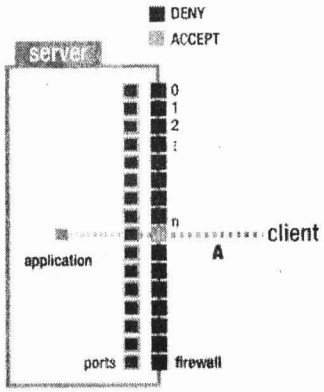


Fig. 3: Client is allowed to Talk to the Running Server

Initially at the view of the client, there is no service running behind the firewall, until the client is able to send the correct sequence of knocking, then only the firewall shall allow the client to talk to the running server.

Port Knocking With Wake on LAN Firewall Methodology

Alike the implementation of Typical Port Knocking, the client will not be able to see any server behind the firewall. Moreover in this new research, the application server is not running at all. Even though a hacker able to hack to the firewall, hacker would not know which machine running as the application server. Until the correct sequence being knock, then only the firewall shall do Wake-On-LAN for the application server to power on and allow the client connection.

Practical Solution

The synergy effect that we obtain other than the ultimate security that is provided by this research, are the implementation of Wake-On-LAN. We had done an impossible technology which initially done for LAN to be deployed on Internet with a secure way. Let us look on the benefit that comes-in indirectly form this integration.

System Access Challenge

A major challenge that IT personnel face in coordinating support tasks is how to access all of the individual servers on the network. Typically, computers are not left up and running 24 hours a day. Powering individual systems down when they are not in use definitely conserves energy. Unfortunately for network administrators, manually flipping a power switch has been the only way to gain access to networked systems. The implications for system administrators are significant. Personnel must be deployed to power on systems in order to access them. Not only is this costly, but gaining access to locked offices and rooms after hours can be difficult. Manually powering on systems is a barrier to performing routine maintenance and data backups. By utilizing knock-wol, we had proven that we can save the resource either on the electricity or the manpower to do this job because we had implant (Just In Time) JIT alike feature, where when there is an authorized request then only the computer will be powered and when there is no activity, it will self power-downed.

Extend Computer Life-Cycle

Every electronic equipment have some level of durability, in other words after exceeding those limit, this peripherals shall fail, this also implied for computer system, a hard disk for example is measured by cycle times, a monitor and most of other parts are determined by powered time. Turning a computer off and on also shall cause surge to the computer circuits and shall also contribute to reduce the period of operation. By running knock-wol, it will relatively prolong the life of the system because, this system only will be awake when used and when left idle it will hibernate.

Server Farm Resources Availability

If we were looking at a big implantation of our research which is in a server farm, here if we tweak a bit of our knock-wol agent to integrate with ipvsadm – a linux virtual server package, so that it will have some form of counter and have a limit burst, after getting concurrent connection and it burst the limit, the agent will awake the next server and load balance the traffic. This is also being done at the firewall level and does not need any changes on the server level. This will also make the resources in a server farm to be optimized and not having all the servers powered and running 24x7 even there is no traffic at all.

Critical Case upon Power Failure

When the electricity supply is discontinued, normally a server room shall have backup powers which run either on battery or generator, these batteries or generator also have limited resources and can last up to certain number of hours. If this backup is serving more servers therefore, more likely the supply will dry easily, if this server room implement knock-wol, it can relatively prolong the duration of this backup power because a server which is in hibernate mode consume very less power. As a proof, SBC2000-188 operating at 20 MHz with 128 Kbytes of RAM, 128 Kbytes of ROM, 4k bytes of EEPROM, and a Real Time Clock; report the usage of power in normal mode is 115mA and when it runs on hibernate mode it uses 0mA.

Conclusion and Recommendations

This paper had initiated a synergy effect by providing solution on day to day problem that most IT Manager want to have in their network, which is a dynamic computing in a secure mode. What we managed to proof is that Port Knocking, a new technology will perform an awesome outcome when it is integrated with Wake-On-LAN. There are many avenues that can be explored more, for example, the encryption part, this paper really recognize the benefit of having encryption module planted, where when the client communicate with the firewall, the session is not easily understood by the sniffer.

References

Advance Micro Device, Magic Packet ® Technology. (2005). *Magic Packet*. [On-line] Available: http://www.amd.com/us-en/ConnectivitySolutions/TechnicalResources/0,,50_2334_2481,00.html

Boyce Michael. (2004). *BashPortKnocking*. Bash IPTables Portknocking. [On-line] Available: http://www.phantomcode.com/bashiptableportknocking/

Bruce Schneier & Crypto-Gram. (2004 March 15). *Port Knocking*. [On-line] Available: http://www.schneier.com/crypto-gram-0403.html#5

Cappella & Tan Chew Keong. (2004, May). *sig2knock*. Port Knocking Project by SIG^2 G-TEC Lab. [On-line] Available: http://www.security.org.sg/code/portknock1.html

Claes M Nyberg. (2003, December). *SA*. A Non Listening Remote Shell and Execution Server. [On-line] Available: http://cmn.listprojects.darklab.org/

D. Eastlake. (1994, December). *Randomness Recommendations for Security*. Network Working Group - Request for Comments. [On-line] Available: http://www.ietf.org/rfc/rfc1750.txt

David Worth. (2003). *COK*. Port Knock Knock. [On-line] Available: http://www.hexi-dump.org/bytes.html

Donald Becker (2005). *Etherwake*. Wake-On-LAN under Linux. [On-line] Available: http://www.scyld.com/wakeonlan.html

FX, Phenoelit. (2005). *cd00r*. Port Knocking. [On-line[Available: http://www.phenoelit.de/stuff/cd00rdescr.html

Geek.com LLC. (1996-2005). *Glossary Search Results*. [On-line] Available: http://www.geek.com/glossary/glossary_search.cgi?w

Ico Doornekamp. (2005). *How does WOL work?*. Port Knocking. [On-line] Available: http://www.tc.umn.edu/~olve0003/wakeonlanREADME.txt

James Meehan. (2004, February). *Pasmal.*SourceForge.net: Project Info – pasmal (packet auth. sniffer - mal). [On-line] Available: http://sourceforge.net/projects/pasmal/

JB Ward. (2005, September). *The Doorman - or - "Silent Running".*The Doorman. [On-line] Available: http://doorman.sourceforge.net/

Joe Walko. (2004, June). *CryptKnock.* Cryptknock - a simple encrypted port knocker. [On-line] Available: http://cryptknock.sourceforge.net/

Jon Snell. (2005). *Combo.* Nerd Info [On-line] Available: http://www.e-normous.com/nerd/combo/

Judd Vinet. (2004, April). *Knockd.* Port Knocking. [On-line] Available: http://www.zeroflux.org/cgi-bin/cvstrac/knock/wiki

Korotkov Eugeny. (2003). *Bash.* Port Knock. [On-line] Available: http://wwwbigfoot.com/~zhokuzma/

Krzywinski, M. (2003, December 12-17). *Port Knocking: Network Authentication across Closed Ports.* SysAdmin Magazine 12. [On-line] Available: http://www.portknocking.org/

Marcello Greco & Alessandro Barenghi. (2003). *Port Knocking Suite.* [On-line] Available: http://digilander.iol.it/grecom

Marilen Corciovei. (2005, January). *knockd.py.* Sample port knocking implementation on FreeBSD with python. [On-line] Available: http://len.is-a-geek.org/misc/portknock.html

Martin Krzywinski. (2003, June). *Port Knocking.* Linux Journal – Security. [On-line] Available: http://www.linuxjournal.com/article/6811

Mike Aiello. (2004). *Winportknocking.* SourceForge.net: Project Info - Windows Port Knocking. [On-line] Available: http://sourceforge.net/projects/winportknocking

OldW0lf. (2004). *TocToc.* Pork Knocking. [On-line] Available: http://www.atrix-team.org/toctoc/

Paul Gregoire. (2004). *jPortKnock.* What the hell is port knocking? [On-line] Available: http://www.gregoire.org/code/jportknock/

Shachar Shemesh. (2004, December) *temprules.* SourceForge.net: Project Info – Temprules. [On-line] Available: http://sourceforge.net/projects/temprules

Tony Smitch. (2004, August). *Port Key.* Smee.org!. [On-line] Available: http://www.smee.org/portkey/

SAADIAH YAHYA, MOHAMED SULAIMAN SULTAN SUHAIBUDDEEN, Faculty of Science Quantitative and Information Technology, Universiti Teknologi MARA. saadiah @tmsk.uitm.edu.my, mohamedsulaiman@gmail.com