

HAKING

PRACTICAL PROTECTION

IT SECURITY MAGAZINE

Vol.5 No.5
Issue 5/2010(30)
1733-7186

IS DDOS STILL A THREAT?

JAILBREAKING AND PENETRATING
WITH THE IPHONE 3G & 3GS

FLASH MEMORY FORENSIC TOOLS - PART TWO

BEGINNER'S GUIDE TO CYBERCRIME
UNDERSTANDING ATTACK METHODOLOGIES AND
A MORE PROACTIVE APPROACH TO DEFENSE

PULLING KERNEL FORENSIC WITH PYTHON

MORE SECURE PHP SERVER SIDE SOURCE ENCRYPTION

SECURING PUBLIC SERVICES USING TARIQ

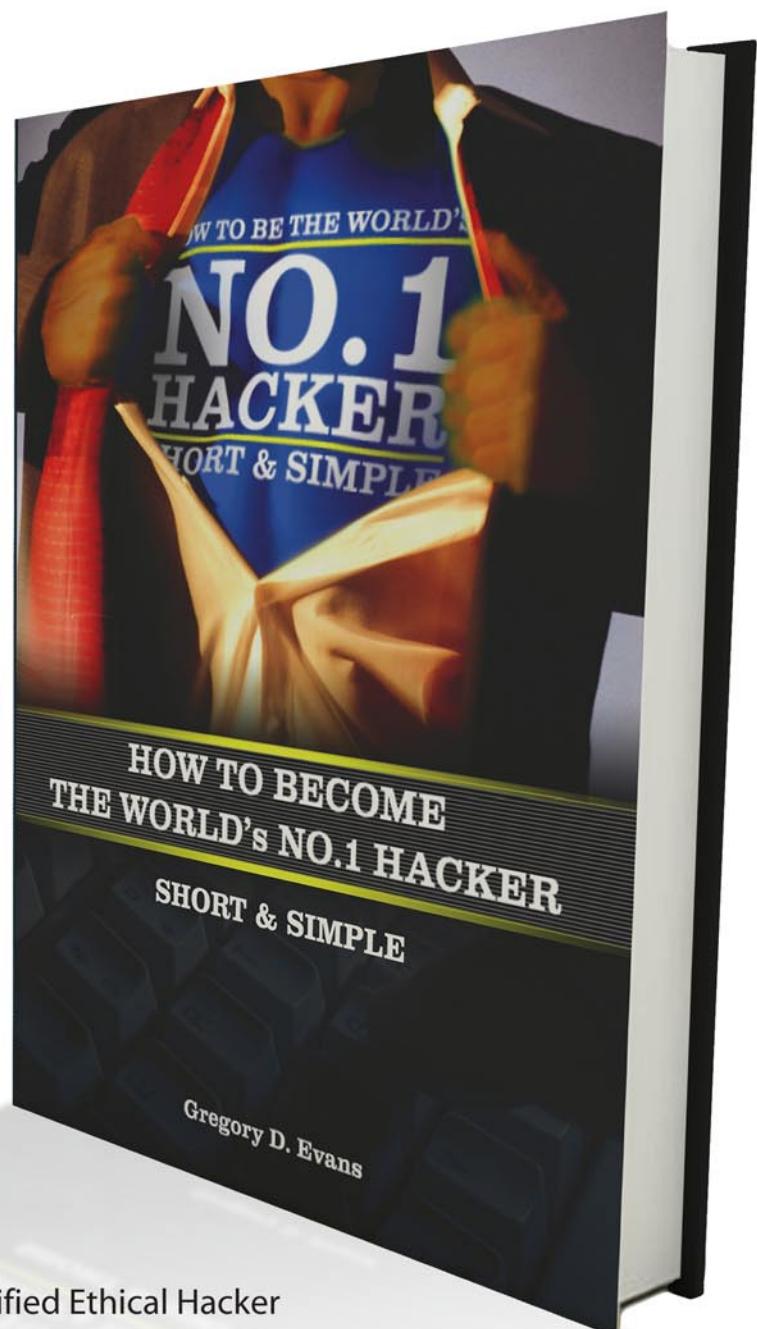
PLUS

EXPERT SAYS:
DON'T LET THE ZOMBIES
TAKE YOU DOWN! BY IAN KILPATRICK

Now Available at Amazon.com and All Book Stores

**THIS GUIDE COVERS A WIDE
RANGE OF TOPICS, INCLUDING:**

- Installing Spyware
- Cracking Wireless Network
- Hacking into a computer
- Hacking Web servers
- Tapping VOIP telephones
- Password Cracking
- Hacking Voicemail
- Cell Phone Spyware
- SQL Injection
- Email Spoofing
- Reseting Windows Passwords
- Social Engineering
- Denial of Service
- How to make money as a Certified Ethical Hacker
- And Much, Much More!



amazon.com

HOWTOBECOMETHEWORLDSNO1HACKER.COM

www.HACKERGEARONLINE.com

**ORDER YOUR T-SHIRT
TODAY!**




HACKERGEARONLINE

HAKIN9 team

Editor in Chief: Karolina Lesińska

karolina.lesinska@hakin9.org

Advisory Editor: Ewa Dudzic

ewa.dudzic@hakin9.org

Editorial Advisory Board: Matt Jonkman, Rebecca Wynn, Rishi Narang, Shyaam Sundhar, Terron Williams, Steve Lape, Aditya K Sood, Donald Iverson, Flemming Laugaard, Nick Baronian, Michael Munt

DTP: Ireneusz Pogroszewski

Art Director: Agnieszka Marchocka

agnieszka.marchocka@software.com.pl

Cover's graphic: Łukasz Pabian

Proofreaders: James Broad, Ed Werzyn, Neil Smith, Steve Lape, Michael Munt, Monroe Dowling, Kevin McDonald

Contributing editor: James Broad

Top Betatesters: Joshua Morin, Michele Orru, Shon Robinson, Brandon Dixon, Stephen Argent, Jason Carpenter, Rishi Narang, Graham Hill, Daniel Bright, Francisco Jesús Gómez Rodríguez, Julián Estévez, Michael Sconzo, Laszlo Acs, Bob Folden, Cloud Strife, Marc-Andre Meloche, Robert White, Bob Monroe,

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: Karolina Lesińska
karolina.lesinska@hakin9.org

Subscription: Iwona Brzezik
Email: iwona.brzezik@software.com.pl

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

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](http://www.smartdraw.com) program by  SmartDraw

The editors use automatic DTP system  AUPUS
Mathematical formulas created by Design Science MathType™

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.

CONTENTS

Dear readers,

This is the second time we meet digitally. This time I want to thank you for your support and involvement in promoting our magazine. In the last months we noticed a great growth of Hakin9 readers and I am sure you actively take part in it:) So, thank you!

In this issue we focus on several issues: Matt Jonkman gives us his thoughts on DDOS attacks, and in the expert section you will find an article on botnets – dangers and protection against them. In the attack section you will read a great work on jailbreaking and penetrating with the Iphone 3G & 3GS. In the defense section there is a beginner's guide to cybercrime focusing on understanding attack methodologies and a more proactive approach to defense.

As I have mentioned last time, you will be receiving a newsletter with new issue at the end of each month, so keep an eye on your emails! If you would like to help in creating Hakin9 magazine, become an author, proofreader or betatester – don't hesitate! Keep the mails coming in!

Enjoy your reading! And remember – go green, choose download!

best regards
Karolina Lesińska
Editor-in-Chief



REGULARS

06 in Brief

Latest news from the IT security world

Armando Romeo

ID Theft Protect

10 Tools

NTFS Mechanic

Active@ Undelete

Michael Munt

39 Emerging Threats

Is DDOS Still a Threat?

Matt Jonkman

46 Expert Says...

Don't let the zombies take you down!

Ian Kilpatrick

BASICS

12 Pulling Kernel Forensic with Python

Daniel Lohin

ATTACK

18 Jailbreaking and Penetrating with the Iphone 3G & 3GS

Wardell Motley

22 Flash Memory Forensic Tools - part two

Salvatore Fiorillo

DEFENSE

30 Securing Public Services Using Tariq

Ali Hussein

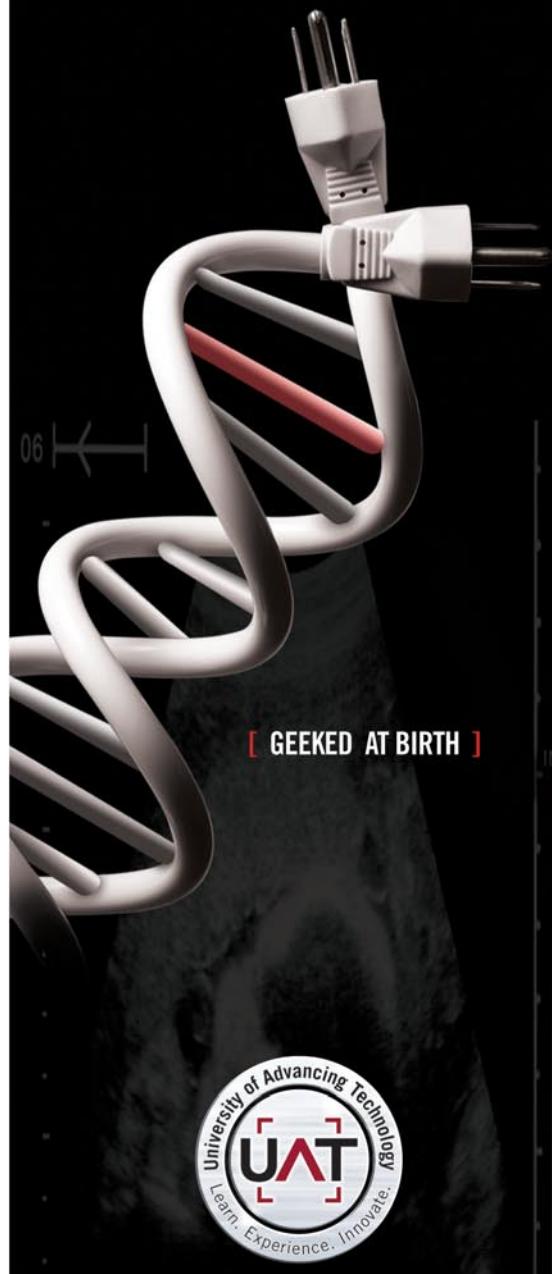
34 Beginner's Guide to Cybercrime – Understanding Attack

Methodologies and a More Proactive Approach to Defense

Gary Miliefsky

40 More Secure PHP Server Side Source Encryption

Israel Torres



[IT'S IN YOUR DNA]

LEARN:

- | | |
|---------------------------------|-----------------------------------|
| Advancing Computer Science | Network Engineering |
| Artificial Life Programming | Network Security |
| Digital Media | Open Source Technologies |
| Digital Video | Robotics And Embedded Systems |
| Enterprise Software Development | Serious Game And Simulation |
| Game Art And Animation | Technology Forensics |
| Game Design | Virtual Modeling And Design |
| Game Programming | Web And Social Media Technologies |

www.uat.edu > 877.UAT.GEEK

Beware the ID theft protectors

ID Theft is a true concern that is not going to stop.

Incidents increased by 11% from 2008 to 2009 affecting over 11 million Americans in 2009.

The most likely to fall victim are young adult and small business owners.

The first are not aware of the risks related to privacy loss through social networks.

The latters are subject to complete a large number of financial transactions online and offline that necessarily require the use of information such as SSN, Tax ID and email addresses.

When a market is growing so much and TV starts to consider this a real plague, it's not to wait too long before someone comes up with a solution. A fake solution in this case.

Lifelock claims itself "leader in ID theft protection". They try to avoid that your data falls into the wrong hands and even if it happens they help you find out where your data is.

The business model is similar to an insurance: you pay \$10 to \$15 every month and if you fall victim of an ID theft they will help you keep up with the costs of solving the issue up to 1 million dollar. Everything sounds fantastic, until you find out that Lifelock own CEO has been fallen victim of ID theft at least 13 times in the last 2 years. That New York times has uncovered, in a series of articles, how the whole business is based on deceptive advertising and no real value is brought to the user.

The Tempe company operations still go on even after a 12\$ Million penalty and will probably go on spending million dollar of TV commercials and deceptive message to address a market and problem for which a real solution is not yet available.

Khobe – malware bypassing all Windows AV's

The headlines of Matousec.com research sounded to Antivirus vendors hype and terrifying at the same time: *New malware bypasses virtually all Windows AV's*. Researchers, in early May 2010, said they were still able to have all the most common Antivirus tool protections bypassed: the method was known to Antivirus vendors and indeed not new. The devised malware affects all the protection mechanisms employing SSDT hooking on Windows. According to researchers *most of security software vendors implemented their kernel hooks very poorly and their applications were creating another holes into the operating system instead of protecting it*. A new tool, named BsodHook, has been devised to find this kind of vulnerabilities automatically. Vulnerable products includes a very wide range of well known tools including McAfee, TrendMicro, AVG and Symantec. The method used by researchers has demonstrated to be very reliable and with a high success rate on multi-processor systems.

The disaffection of the community towards anti-malware vendors and the objective hype in the headline made the research traverse Twitter and all the security web sites, that have all given massive coverage.

Responses from the Antivirus vendors, through their corporate blog, were limited to *we are not vulnerable or it is unjustified hype*.

Now Facebook Privacy is a concern

After years of blindness, Facebook users now realized their privacy is *at risk*. Google searches for *how to remove facebook account* is rising and all the printed and online magazines, after months of hype and tutorials on how to buy fake-gift

for your friends, now host articles on how to handle your privacy concern. Even programmers, now prefer to code online tools such as Openbook and Zesty.ca/facebook instead of pumping new facebook application into the funnel. These tools are now getting famous and very (mis)used as Facebook privacy is getting laxer and laxer.

Facebook lack of privacy is basically creating another grey market where your information is easily accessed and possibly sold.

The latest Facebook privacy policy is 5830 words, 1287 words longer than United States Constitution and it tends to be more and more permissive about what you must to share.

Doing something about it is now creating another market niche. Now we have services that will fine tune your account to avoid giving out too much information. Why? Because according to PcWorld there are *over 50 settings and 170 options to adjust. And even that won't completely safeguard your info*.

As long as having a Facebook account is felt as one of the universal individual's right, (a sort of cyber freedom of speech?), Zuckerberg and his multi-billion dollar new-con investors, will have the power and the arrogance to ask for *forgiveness and never for permissions*.

Do you trust Google?

United States is the only among western countries not having a federal law on Privacy. This doesn't entitle Google, a US corporation, to collect Europeans' data. This is the summarized statement given by German consumer protection ministry when the shocking news was disclosed: Google has for years carried out extensive wardriving collecting at least 600 gigabytes of illegal data through the use of special wireless equipment included in Google

Street cars. Google admitted this behavior in a blog post clarifying that the collected data regarded solely photos, 3D building imagery and WiFi network information. At first.

Among this information there is SSID of networks and MAC addresses, but not payload data according to big G. After this post, dated 27 April 2010, Alan Eustace, Google senior vice president of research, gave a completely different and clarifying version: *It's now clear that we have been mistakenly collecting samples of payload data from open WiFi networks* stated. The payload were collected by mistake. But it has been collected. A piece of software coded by a former Google engineer had been included in the firmware of the devices shipped in Google cars. This firmware was originally meant to only store SSID's and MAC addresses.

This mistake will cause Google a series of legal issues in Europe where Privacy is still something serious.

Metasploit Express released

Since the Metasploit project buyout by Rapid7, the Framework, led by HD Moore, has boosted its operations bringing an integration with Core Impact and now a commercial version of the open source exploitation framework named Metasploit Express. The project will now fork and both the open source framework, now released in its 3.4 version, and the commercial version will be supported in parallel.

Metasploit Express has been a great addition to the fast growing Rapid7 company: a penetration tester has now the power of Rapid7 vulnerability management solutions, namely Nmap, and the exploitation power, now even automated and extended of a commercial exploitation

framework supported by the open source community.

Metasploit Express features a GUI for automatic scanning and exploitation configuration, administration and advanced reporting management.

It also emphasizes the importance of security auditing and exploitation workflow, that is extremely important when testing the security of large enterprises.

All these features and an advertised ease of use, position this tool in the enterprise segment for in-house security auditing and for small-business security vendors and consultants in the penetration testing field.

Metasploit new release includes massive improvements to exploitation payloads, especially meterpreter and new brute forcing capabilities introduced in version 3.4.

Need SEO? Ask hackers

This is not to be confused with Blackhat SEO that has a completely different meaning.

But the habit of exploiting SEO techniques for malicious purposes is now consolidated among criminals. It has been named as SEO poisoning and we have had the most prominent example with the Chile earthquake: rogue pages, containing malware and other browser exploits, appeared on top of the google ranking for hot searches, in the hours of the tragedy.

Search terms like *chile earthquake find relatives* or *Chile quake 2010 tsunami* were heavily addressed with rogue blog posts appearing among more reputable news websites.

The technique is relatively simple. Everyone can get the list of the hottest search keywords using free to use google tools. Then a number of back-links pointing to the rogue page is required. A number of small websites are believed to

be owned by criminals just for this purpose. Usually criminals, use iframe injection attacks to have a number of vulnerable and unaware websites to link back to their rogue page.

Google favors websites with a greater number of backlinks or backlinks with some reputation. Yahoo and other search engines do not base their ranking on the number of backlinks rather on the so called on-page optimization, thus making it even more simple for a hacker to forge a well optimized web pages to show early in search results. However, Google is the most targeted search engine since it's by far the most used.

When such an attack is launched it takes just a few hours for results to appear.

Criminals are now very smart at picking the hottest topics: Miss USA Rima Fakih's past photos appearing on Google Images are the latest example.

Source: source: Armando Romeo

Destructive Malware Identified

A new computer virus that replaces all files in the C: drive with copies of itself has been identified by a leading UK internet security company. The malware, named W32/Scar-H, can lead to a cascade effect where, in the end, it takes down the entire computer system. Oddly, there seems to be no financial motive behind the virus since its function is purely destructive. ID Theft Protect says that this type of approach (hard drive destruction) is very unusual. Maybe someone has a grudge against a particular organisation or person?

Google Groups Delivering Malware

Cybercriminals are using Google Groups to distribute rogue anti-virus software and other malware, according to leading security

researchers. The attackers are sending e-mails to Google Groups members asking them to update their e-mail settings by following linked instructions.

The links take users to a fake Google Groups page that infects visitors' PCs with a Trojan that downloads malicious software, including rogue anti-virus program *Desktop Security 2010*. The rogue software runs a fake PC scan, notifies the user that the PC has been infected and then prompts the user to buy software to remove the threat. The malware is designed to trick users into handing over their credit card details and other personal information to purchase the bogus software.

Software Piracy is on the Increase

The overall rate of software piracy increased two percent compared to 2008, a spike that primarily can be attributed to the rapid growth of the consumer PC market in Brazil, India and China, a leading report by IDC. Overall, the commercial value of global software theft exceeded US\$51 billion in 2009.

In the study released earlier in May, IDC researchers analysed PC and software trends in 111 countries. Researchers found that some progress has been made in the fight against piracy. During 2009, unlicensed PC software use decreased in 49 percent of the nations studied.

The United States had a 20 percent software piracy rate, the lowest out of all countries studied. In addition, Japan and Luxembourg had piracy rates of 21 percent. Countries with the highest piracy rates included Georgia, Bangladesh, Zimbabwe and Moldova, each with a piracy rate above 90 percent.

Windows 7 Aero Flaw Identified

In May, a serious vulnerability was identified in Microsoft's new

operating system – Windows 7 and Windows Server 2008 RC2. The security flaw could expose users to code execution and denial-of-service (DDOS) attacks. The file responsible for the flaw was found in the Canonical Display Driver (cdd.dll), which is used by desktop composition to blend the *Windows Graphics Device Interface* (GDI) and DirectX drawing.

Microsoft has stated that it is much more likely that an attacker who successfully exploited this vulnerability could cause the affected system to stop responding and automatically restart. The company has activated its security response process and promises a security patch to follow very shortly.

Windows 7 Trojan Horse Threat

Cyber criminals have disguised Trojan horse malware under the guise of a Windows 7 compatibility checker. The malware comes as a zip-based attachment to email messages supposed offering help on upgrading Windows boxes. But this *Windows 7 Upgrade Advisor Setup* assistant offers only a Trojan, instead of the promised compatibility checking tool.

Windows users who open and run the application end up with systems compromised with a backdoor that allows hackers to insert other viruses and spyware. The hackers behind the attack get to pimp out these compromised systems to other miscreants, earning illicit affiliate income in the process.

Yahoo! Messenger Malware Threat

A new worm has materialised via Yahoo Instant Messenger. It appears that it is even more sophisticated in social engineering and payload than previous worm attacks on Yahoo Instant Messenger. This new worm installs via the backdoor of Windows

systems that use ONLY Yahoo Instant Messenger.

The malware arrives via an instant message through Yahoo or Skype with any one of a number of messages, including „Does my new hair style look good? bad? perfect?” or *My printer is about to be thrown through a window if this pic won't come out right. You see anything wrong with it?*

The message includes a link to a web page that looks like it leads to a JPEG image file. When the link is clicked, the browser displays an interface that looks like the RapidShare web hosting site and offers up a ZIP file for download. The extracted file is actually an executable file with a .com extension.

Source: ID Theft Protect

Foxit Readers adds 'Safe Mode'

Foxit Corp (US) has added new security features to its alternative PDF reader software to help thwart recent malware attacks that exploit the */launch* feature. With Foxit PDF Reader Version 3.3, the company has added a Safe Mode that blocks external commands from being executed by the software. The Safe Mode is a key part of a new Trust Manager in the Foxit PDF Reader.

Earlier this month, Foxit Reader adopted a warning message before running any executable command embedded in a PDF document. The changes follow the discovery by a leading researcher, that dangerous executables can be embedded into PDF files (and executed) without exploiting any vulnerabilities.

Source: ID Theft Protect/Foxit Corp (US)

>

Let's face it. Checking every web page for cross-site scripting is fun for about four minutes. Then it gets really dull.

Outsource the boredom with Burp Scanner.

<http://portswigger.net>

NTFS Mechanic

Disk & Data Recovery for NTFS Drives

Items Tested:

40GB External USB HDD that has had an extensive amount of files written to it, and then randomly deleted, approximately 16GB in total and has intermittent connection issues to the point that the local machine doesn't actually register the drive is there.

Once I had the software installed it was time to see how it performs. I plugged the external drive in and then powered up the software. It saw my drive straight away, but it didn't actually state what disk format the drive actually was. This might be due to the fact that the operating system didn't actually find the drive itself, so it was a pleasant surprise that this program did indeed find it.

You are able to configure what types of files you actually want the program to be searching for during the recovery process, for this test I just left everything as default which means everything was selected.

I selected my external USB Drive and it scanned the partitions first to ensure that it can actually see the drive correctly. Once this part of the process has been completed it then requests that you allow it to scan the whole partition that you have selected, this appears to be a very cpu intensive program so I would suggest to just leave it running on its own if

Pricing

Standard \$99.95
Business \$199.95
Professional \$299.95
Prices are in US Dollars

possible. It took just over an hour to scan through a 40GB hard drive. Once it was finished NTFS Mechanic provides all the data that's on the drive, deleted and non-deleted files. You can select in the right hand menu to only see the recovered files, which makes it a lot easier to see what the program has actually found.

If you look at the properties of the files and folders that have been listed as being recovered, you can actually see the prognosis of each file if you decided to proceed and recover the file completely.

The process for recovery couldn't be much easier, it's simply a case of going through the folder list and selecting the files you want to recover and then just say where you want them to be stored.

The program performs really well and managed to recover data from a disk that hasn't been seen by any of my machines for a little while now which quite impressed me.

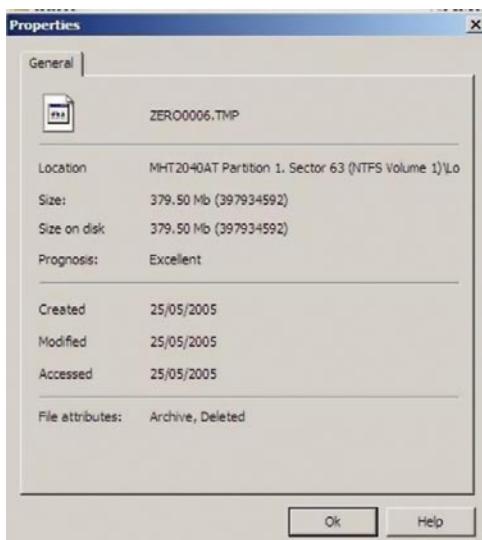
I noticed that there were a few areas within the program that could do with some QA work as there were non English characters in use and some screens weren't actually needed in my opinion but they aren't detrimental to the product.

I would gladly have this tool in my toolbox.

http://recoverymechanic.com/ntfs_recovery/ntfs_mechanic.php

Partition Recovery
Hard Drive Recovery
Recover deleted files

by Michael Munt



Active@ LiveCD Disk Suite Edition

Windows Based

Active@ LiveCD provides a bootable CD that gives you a lightweight Windows (WinPE 2.0) environment or a DOS based environment with a powerful suite of tools. You have the option to add additional files, drivers and even scripts to aid you at the time of disk creation.

You're able to create and restore images of the disks, explore the images and recover specific files and folders from these images. You're also able to create a complete raw image which can be used for forensic purposes, finally you can completely clone a disk which is useful for when creating a system image for rollouts of new equipment. The file recovery recognises file types by their actual headers so even if the files have been renamed by a virus etc, you can still recover them, the ability to rebuild RAID arrays and recover data from them is an excellent feature and something that is usually forgotten about by other recovery systems.

A full partition management system is included allowing you to have full control of the partitions on the local machine (FAT12, FAT16, FAT32, NTFS, NTFS5 are supported). You are able to perform partition recovery on the fly with no reboot being required. You have the ability to create multiple partitions on USB/Flash drive devices, and also create partitions using the FAT32 format up to 1TB in size. You can assign or even change partition settings on any drive that is connected to the system whilst using the LiveCD.

For secure deletion of data, KillDisk is provided and this excellent tool securely overwrites and destroys all data on the disk or selected partition. For the ultra paranoid you can manually select up to 99 passes when erasing to ensure there is nothing left on there at all. Remember you can always double check this, by booting back up with the disk and try to recover any data from the disk.

Also included is a password manager that gives you complete control over all accounts that are local to the machine you are using. It detects all known Microsoft Security Databases (SAM). You're able to reset or change any of the flags that are currently set on any of the accounts that you have identified.

Product Details	Personal	Corporate
Active@ Boot Disk (Win Edition)	\$79.95	\$99.95
Active@ Boot Disk (DOS Edition)	\$69.95	\$89.90
Active@ Boot Disk Suite (Win + DOS)	\$109.95	\$129.95
Active@ Boot Disk (DOS Edition) Enterprise	not applicable	\$3499.00

Full hard disk performance monitoring and control is also included, you can set the system to send out email notifications once certain criteria has been met. You can create full detailed reports concerning the performance of the hard drives in question, which is invaluable when trying to track down errors on a intermittent faulty drive. There is a full suite of other applications included that will allow you to perform a multitude of tasks from taking screenshots to editing the local registry. Full control of the network settings and once online you're able to connect to FTP, Telnet and even surf the internet using the inbuilt browser (I found this browser to be a lot quicker than the Internet Explorer or Firefox on my normal machine)

DOS Based

Even on the dos based side of the suite you are given an excellent range of tools. Uneraser will allow you to undelete files from FAT16, FAT32 and NTFS partitions. Supporting long filenames, creating disk images and even Master Boot Record backups. Using the disk viewer you can view any hard disk drive sectors no matter the version of Windows OS installed. Killdisk (DOS version) is included as is a full partition recovery solution. The password changer performs exactly as the windows based one, giving you full control over all the local accounts on the system. Finally the NTFS reader allows you read access to the NTFS drive and you can preview all files (even long filenames) and transfer them across to NTFS or FAT volumes, even to network based drives.

Once again Active@ have produced an excellent piece of software and this one is also go straight into my dvd case and will have a permanent home there. I can't sing its praises highly enough.

by Michael Munt

Pulling Kernel Forensic Data

with Python

How to proceed with gathering forensic information of Linux machines when a user-level rootkit is suspected to be installed by utilizing Python to automate the process of pulling data.

What you will learn...

- A basic understanding of /proc and how it can be used to collect information about the Linux kernel
- Using Python to collect information from /proc in an automated fashion

What you should know...

- A basic understanding of Linux and Operating Systems
- Experience with high level programming languages

When dealing with a machine that may be potentially compromised it is critical that an incident analyst use as little tools as possible that are on the operating system itself. Many tools on a Linux or Unix system like ps, netstat, arp, etc could have been compromised by the attacker to prevent the user from finding traces of the malicious actor in an incident. If an attacker is running a process on a box called *virus* it is a common technique to replace the ps command which normally lists running processes with a version that will not display any executable with the name *virus*. This presents an analyst trying to perform live analysis a unique problem. This technique would be classified as a user level rootkit. How do you get information about what is running on the machine without trusting the machine itself. In many instances an analyst will carry around many common tools on a disk which are statically linked, or contain no

dependencies of the system itself. Another method is to communicate with the */proc* filesystem itself to pull this information. Linux and many other forms of UNIX contain a */proc* pseudo-filesystem which contains what appears to be a filesystem, but actually is a method of communicating with the underlying kernel. By opening many of these files an analyst is able to get a lot of information about processes the kernel is running, network connections, open file handles and more. In addition, a root user can actually manipulate kernel variables on a live system.

To view the contents of this filesystem simply list the contents of */proc* as if it were a regular directory with the command *ls /proc* (see Figure 1).

In this directory is a wealth of information. To view information about the current processor on the system list the contents of the */proc/cpuinfo* as if you were outputting a file with the command *cat /proc/cpuinfo*. It is possible to get a lot of useful information about what is running in the kernel by using this mechanism. This article looks at

```
dlohin@dlohin-laptop:~$ ls /proc
1 1294 2 2134 29 7  cgroups      mtrr
10 13 20 2137 3 70  cmdline     net
1001 1385 2006 2139 30 71  cpufreq     pagetypeinfo
1092 1386 2048 2140 31 717  crypto      partitions
1095 1397 2051 2148 315 72  devices     sched_debug
11 14 2052 22 32 73  diskstats   schedstat
1101 15 2056 2209 33 737  dma        scsi
1105 1543 2059 2222 34 74  driver      self
1108 1576 2061 2233 35 75  execdomains slabinfo
1109 1580 2072 2234 358 757  fb         softirqs
1124 1584 2078 2311 37 824  filesystems stat
1125 1588 2078 2323 38 825  interrupts  sys
1128 1592 2083 2323 38 825  iomem      sysrq-trigger
1140 1596 2088 2338 4 869  iports     sysvipc
1142 16 21 24 4261 882  irq       timer_list
1143 1650 2100 2434 436 885  kallsyms   timer_stats
1147 1682 2102 2478 437 896  kcore      tty
```

Figure 1. Contents of /proc

```
dlohin@dlohin-laptop:~$ ls -la /proc/10/
attr/          fd/          mountstats    sessionid
auxv/          fdinfo/      net/          smaps
cgroup/        io/          numa_maps    stack
clear_refs/    latency/    oom_adj      stat
coredump_filter limits/    oom_score    status
cmdline/       loginuid/   pagemap      syscall
cwd/          maps/        personality  task/
environ/       mem/        root/        wchan
exe/          mountinfo/  sched/       -
fd/           mounts/     schedstat/ 
mountstats/   sessionid/
```

Figure 2. Contents of pid information in /proc

how to get information from the proc psuedo-filesystem for forensic purposes to get information directly from the kernel, which will bypass potentially compromised tools like ps, netstat, etc.

Process information

In the `/proc` directory should be a series of what appears to be random numbers. These are actually directories that correspond to each Process ID currently running on the system (see Figure 2). In this directory we see several files that are of interest to us.

cmdline file: Displays the command that was run to execute the particular command.

cwd: The current working directory of the process
exe: A symlink that points to the executable to the application running (useful if you expect that malicious software to make sure a process isn't running from a strange location).

fd: Currently open file descriptors, which will be discussed further.

net: Information on the network connections which will be discussed further.

maps: contains open shared libraries for information

There is an excellent Python Package which allows you to easily pull information from proc easily in a very python manner. <http://pypi.python.org/pypi/enumprocess/0.1>

Listing 1. Creating a simple Python script to pull open libraries by processes from `/proc`

```
#!/usr/bin/env python

import enumprocess
class processtest:
    def processCheck(self):
        """This will get all the running processes running on the system"""
        processinfo = {}
        for i in enumprocess.getPidNames():
            try:
                processinfo = enumprocess.getPidDetails(i)
                print "PID %d: %s" % (i,processinfo['name'])
            except:
                print("can't read the process %s, possible permissions issue? " % i)

    def getLibs(self):
        """Print the process and all shared libraries that are currently open WARNING THIS WILL PRINT A LOT"""
        #http://linux.die.net/man/5/proc
        for i in enumprocess.getPidNames():
            try:
                processinfo = enumprocess.getPidDetails(i)
                print("PID: %s NAME: %s" % (i,processinfo['name']))
                path = "/proc/" + str(i) + "/maps"
                maps = open(path)
                maps.readline()
                for i in maps:
                    print("      %s" % i)
            except:
                print("can't read the process %s, possible permissions issue?" % i)

process = processtest()

print("=====Process Checks=====\\n")
process.processCheck()
print("=====Library Dump=====\\n")
process.getLibs()
```

Listing 2. Pulling open file handles of processes in /proc

```
#!/usr/bin/env python

import re
import os
import enumprocess

class fdFunctions:
    def getPIDByFD(self, lookFor):
        """Put the fh to look for, and will suck out the process that currently has it open, you do not need the
           whole thing, just a bit to find it"""
        fileHandles = self.getOpenFDs()
        for fd in fileHandles:
            processNumber = fd[0]
            fdNumber= fd[1]
            match = re.match("/proc/[0-9]+/fd/([s\w:\[\]\_!#\$\%&'\(\)\-\@\^\'{}\~\+\,\.\.;\=\[\]\]+)",fileHandles[fd])
            if match != None:
                pass
            if(match != None and match.group(1) == lookFor):
                return processNumber

    def getOpenFDs(self):
        """Finds a process and what open file handles they currently have, returns a multidimensional dictionary
           of process number, the file descriptor number"""
        contentsInProc = os.listdir("/proc")
        processMap = {}
        for i in contentsInProc:
            process = re.match(r"([0-9]+)",i)
            if process:
                try:
                    fds = "/proc/" + process.group(0) + "/fd"
                    fileDescriptors = os.listdir(fds)
                    for j in fileDescriptors:
                        #real path gets me the path of the symlink
                        path = os.path.realpath(fds+"/"+j)
                        processMap[(i,j)] = path
                except OSError:
                    print "Can't open, permission denied?"
        return processMap

    def printOpenFDs(self):
        """Finds a process and what open file handles they currently have, returns a multidimensional dictionary
           of process number, the file descriptor number"""
        contentsInProc = os.listdir("/proc")
        for i in contentsInProc:
            process = re.match(r"([0-9]+)",i)
            if process:
                try:
                    fds = "/proc/" + process.group(0) + "/fd"
                    fileDescriptors = os.listdir(fds)
                    for j in fileDescriptors:
                        #real path gets me the path of the symlink
```

Listing 2. Pulling open file handles of processes in /proc

```

        path = os.path.realpath(fds+"/"+j)
        print "PID: %s FD: %s Filename: %s" % (i,j,path)
    except OSError:
        print "Can't open, permission denied?"


def getFDsByPID(self,pidToLookFor):
    """Pass in the pid and it will return a list of all the file descriptors"""
    fileHandles = self.getOpenFDs()
    fdReturn = []
    for fd in fileHandles:
        processNumber = fd[0]
        fdNumber= fd[1]
        if processNumber == pidToLookFor:
            #Create an array of fd Number
            fdReturn.append(fileHandles[fd])
    return fdReturn

fd = fdFunctions()
fd.printOpenFDs()

```

Enumprocess works on both Windows and Linux, but we will only be focusing on Linux for this process. If you look over the Enumprocess source code you will note that enumprocess is basically pulling information from `/proc` to get process number and other information. We will be expanding on this by pulling network information, file handles and shared libraries.

It is possible to install the enumprocess library on your machine, but normally when you are working on a victim's machine they prefer that you do not install anything on their machine. If you download the `.tar.gz` file one this site you can pull just the library itself. If you then place the directory to the library in the same folder as your python script you will be able to use this library without installing the library on the machine, which is preferred. You are also trusting the libraries on the computer less which is preferred in investigations. We will be putting all files in `~/pidenum` (~ is a short cut for your home directory). To do this:

```

mkdir ~/pidenum
tar xvzf enumprocess-0.1.tar.gz
cd enumprocess-0.1/src/
cp -rpf enumprocess ~/pidenum
cd ~/pidenum/

```

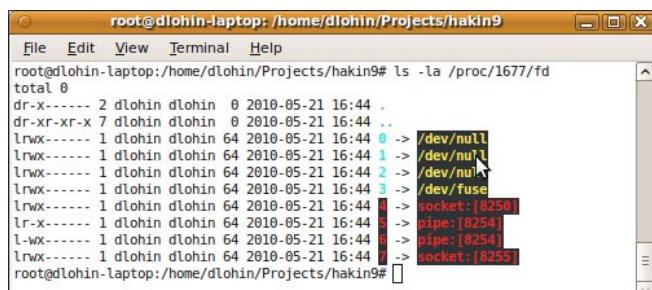


Figure 3. View open filehandles in a process

Place your following scripts that will be covered in this article in a seperate file in this `~/pidenum` directory. This will allow you to use the library without installing anything. When you want to run these scripts on a customer's machine, just ensure you copy this folder with your script.

Note that all of these scripts must be run as root. In many cases if you run these as a regular user, it will work, but you won't be able to see information on processes other than your own.

First Python PID script

For using Python we will write a simple Python object that will use enumprocess to output all processes as well as print out the open shared libraries by all processes in the system. `/proc/<pid>/maps` is a simple file in `/proc` that shows all the shared libraries open by a process. You can view this by simply running the command `cat /proc/<pid>/maps`. All the scripts in this article have been tested on both Ubuntu and Fedora (see Listing 1).

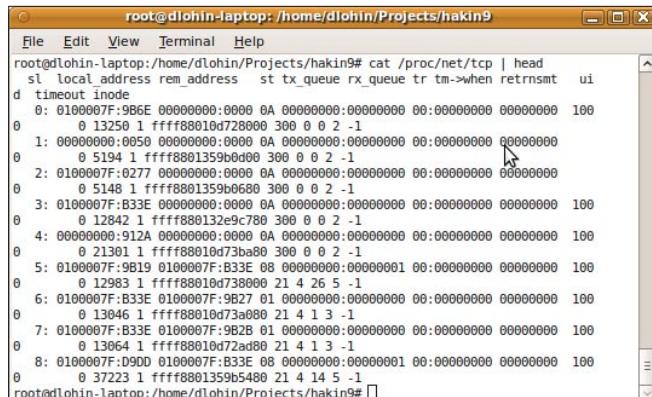


Figure 4. Viewing process network information

Listing 3. Accessing network information to view active connections of a process

```
#####Add the entire FDFunctions() class above this!#####

class networkConnstest(object):
    """This will look at all established TCP connections as reported by /proc/net/tcp and report the information
       as well as what process is using them"""
    def getOpenPorts(self):
        tcp = open("/proc/net/tcp")
        #Throw away the header
        tcp.readline()

        ip = IPFunctions()
        fh = fdFunctions()
        #loop through each, pulling the necessary information
        for i in tcp:
            #nasty regex... match all of the information for the network connections.
            info = re.match("\s+[0-9]+:\s+(\w+):(\w+)\s+(\w+):(\w+)\s+\w+\s+\w+:\w+\s\w+:\w+\s\w+\s+(\w+)\ \s+\w+\s+(\w+)",i)
            #All of the addresses are in HEX need to convert them.
            localAddress = ip.convertHexIPToString(info.group(1))
            localPort = ip.convertHexToString(info.group(2))
            remoteAddress = ip.convertHexIPToString(info.group(3))
            remotePort = ip.convertHexToString(info.group(4))
            uid = info.group(5)
            #Inode is the socket
            inode = info.group(6)
            #The socket the file descriptor
            socket = "socket:[+inode+]"
            # a socket is just a file, so it can be retrieved the same a file descriptor
            pid = fh.getPIDByFD(socket)
            #We have all the necessary info for the ports open, now lets get the app
            processDetails = enumprocess.getPidDetails(pid)
            try:
                print("Pid: %s Name: %s" % (pid, processDetails['name']))
                print("      Pid for socket is %s, name is %s" % (pid,processDetails['name']))
                print "      local address, port: %s, %s" % (localAddress,localPort)
                print "      remote address, port: %s, %s" % (remoteAddress,remotePort)
            except:
                print "Can't open, permission denied?"

        network = networkConnstest()

    class IPFunctions(object):
        """This is needed because the IPs are all in hex and we want them to be easily readable"""
        def convertHexIPToString(self,ipHex):
            """Take an IP in Hex and make it look like a string with periods"""
            count = 0
            octet=""
            ip = ""
            for i in ipHex:
                count += 1
                #print "%s\n" %i
                octet = octet+i
```

Listing 3. Accessing network information to view active connections of a process

```
if count == 2:
    count = 0
    ipOct = str(int(octet,16))
    ip = ipOct+"."+ip
    octet = ""
ip = ip.rstrip(".")
return ip

def convertHexToString(self,hex):
    """Simple function that will be used in order to convert the HEX of port numbers"""
    return str(int(hex,16))

print("=====Network connections=====\\n")
network = networkConnstest()
network.getOpenPorts()
```

File handle information

Often in investigations, it is desired to understand what files are currently open, and what network connections are currently being made. `/proc/<pid>/fd/<file descriptor number>`. Each of these is a symlink to the file that is opened by that particular process.

By running the `ls -la` command on each of these file descriptors and you will be able to view. Because in Unix everything is a file, network connections or sockets will also show up in the file descriptors category, showing a symlink to `socket:[socket number]` (see Figure 3).

To pull this information I will build a Python class that allows information to be easily pulled (see Listing 2).

Network Information

Information on individual network connections for each process is stored in `/proc/<pid>/net/tcp` and `/proc/<pid>/net/tcp6` for all IPV6 connections. This is a file that you can simply run the `cat` command on to dump the contents, but it is a little complicated to read. The local and remote address is written in hex along with the port. Each two hex values correspond to one octet in an IP address. C09C0334:0050 corresponds to 192.168.156.52 port 80. You can use the Windows calculator to perform these calculation, but the Python script will automatically convert these for you as well. This requires the `fdFunctions` class to work which was included in the section above as we are able to treat the network connections as files in Unix (see Figure 4).

There are two classes contained here, the first class is responsible for pulling the information out of the `/proc/net` file. Then we will use the `getPIDbyFD` function in the `fhFunctions` class to pull the PID out for the open socket. The `IPfunctions` class is responsible for converting the HEX address to standard IP address as well as the port number from HEX to base 10 (see Listing 3).

Conclusion

It needs to be understood that these python scripts do have some limitations, for one it relies on the integrity of Python on the victim's box. If the hacker was able to change the various userland binaries, then they may have changed parts of Python. With that said, Python is usually not a high priority target to cover their tracks and probably will be safe in these instances. These python scripts also do not help with kernel level rootkits. A kernel level rootkit will modify the system calls to the kernel and no user-land tool will be able to overcome this.

By understanding the `/proc` filesystem it is possible to view information about a computer system without relying on user level tools like netstat, lsof and so forth. This script is useful for quickly collecting information on a system when it is suspected of compromise. These scripts can be greatly expanded to pull a lot more information out of a system with a little bit of work. The enumprocess contains a lot more information. Understanding the `/proc` filesystem is useful for any security professional that wants to further understand their linux based system and what functions it is currently performing at any given moment.

To see the full script go to: <http://dremspider.net/scripts/hakin9.py>

DANIEL LOHIN

Daniel Lohin currently works as a Information Security consultant at Booz Allen Hamilton. Daniel Lohin is focused on incident detection as well as response. He is currently finishing up his Master's in Information Security at George Mason University. When he is not studying, working or breaking his computer he is bike riding with his girlfriend, Meagan.

Jailbreaking and Penetrating

with the Iphone 3G & 3GS

Today Smart phones are getting smarter and smarter. They are a far cry away from the Walkie-Talkie like devices from the the early 90's.

What you will learn...

- Jail Breaking Iphone 3G & 3GS
- Penetrating Networks with the Iphone Platform

What you should know...

- How to run command line tools like Nmap, Metasploit
- Basic Networking and Security

Now a smart phone in the hands of skilled attacker can be used to help penetrate networks on the fly. No longer do you need to walk around with a bulky laptop to get the job done. By taking an IPHONE and making a few software adjustments and installing the right tools you can be well on your way to finding vulnerabilities in your network before the rest of the world does.

Setting up

Before we get started there are a few things that we will need to download beforehand to make things a bit easier as we progress. First back up all files on your IPHONE! Pictures, phones numbers and anything else that you deem valuable. Jailbreaking an IPHONE can be a simple straight forward process, however, I have heard horror stories of people bricking there IPHONE's after attempting a jailbreak the wrong way. Its better to be safe than sorry so backup. Next I will need you to download the following software packages.

- *iTunes 9.0* – This can be downloaded from oldapps.com,
- *WinSCP* – This can be downloaded from winscp.net.

Iphone Jailbreaking

First off if you are running version OS 3.1.3 on your Iphone then this should work for you (this has not been tested on any later versions). First install iTunes 9.1 on your PC and allow it to sync with your Iphone. Then close iTunes and place your Iphone in DFU mode by doing the following.

Step 0

Backup your IPHONE. Save all of your pictures and contacts and everything else. Take your IPhone and put into DFU Mode.

Step 1

Open Itunes and connect the iPhone to your PC.

Step 2

Press and hold the Home button and the Sleep/Wake button at the same time. After exactly 10 seconds release the Sleep/Wake button (Figure 1).

Continue holding the home button until iTunes pops up telling you that it has detected an iPhone in recovery mode (Figure 2).

Step 3

Next place your mouse over the restore button and hold down the shift key. Browse for the `sn0wbreeze_iphone 3G.ipsw` supplied. A snowflake will flash briefly and the process will begin. It will take about 10 to 15 minutes to restore. After the process completes you should have your Jail Broken device with Cydia installed and ready to go.



Figure 1. Placing the IPHONE into DFU Mode

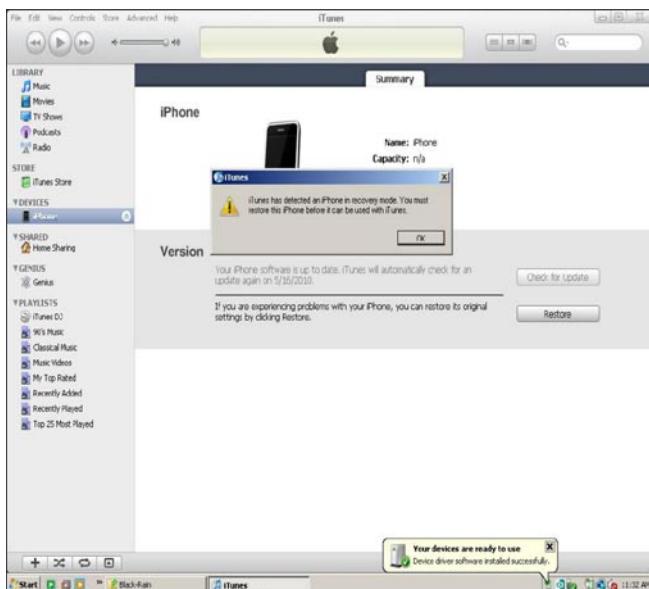


Figure 2. Restoring Custom IPSW

Iphone Software Installation

First we will start out by installing some basic utilities that will allow you to move around your IPHONE easier and allow you access to information that you will find useful as we progress. Before you begin installing any software for your IPHONE I highly recommend connecting to a local wireless access point that's close to you. If you try to download these installs over an Edge network like AT&T's for example it will go painfully slow. The Installation is quite simple let's open up Cydia and do a search for it. You should find Cydia by scrolling to the right of your screen. Tap the Cydia icon and it should open up for you. You may receive a refresh error just hit the okay button and continue. We will start out downloading MobileTerminal. This will allow you access to the command line on the IPHONE. You will be able to use MobileTerminal to change the default password on the iphone from alpine to something more secure and to your liking. Install Tap Mobile Terminal and then select Install and Confirm (Figure 3).

IPhone Password change and cont software Installation

After you have installed mobile terminal find the icon on springboard and tap it. It should bring up a terminal window where you will be able to log in as root and change the password from the default.

```
iPhone:~ mobile$ su
Password: alpine
iPhone:/var/mobile root# passw
Changing password for root.
New password:
Retype new password:
iPhone:/var/mobile root#
```

Next we will install OpenSSH. It will allow us to move files back and forth from your PC to your Iphone. Open

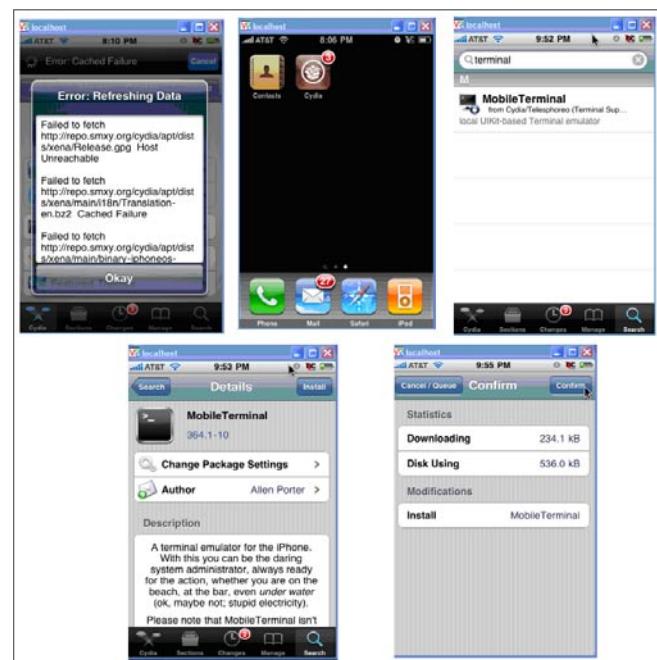


Figure 3. Mobile Terminal Installation

up Cydia and do a search for OpenSSH. Once you have located it run the install and confirm. After the installation it should make SSH available immediately on your Iphone (Figure 4).

Next we will install SBSettings. The purpose of SBSettings is to allow a quick view of your IP address once you connect to a wireless AP. This will come in handy later on. SBSettings also allows you to disable and enable certain services on the fly instead of having to resort to the command line or browsing through a ton of menus. Just as we did with Mobile Terminal above reopen Cydia and do a search for SBSettings. Install and Confirm the installation. It will install and it will then restart springboard. After springboard comes back up give the SBSettings a try by placing your finger at the top of your screen close to where your signal icon is and slide your finger from left to right. It should bring down a drop down menu that allows you see to quite a bit of useful information. Here you have the ability of enabling and disabling your wifi or killing processes. You will also notice that you now can view your IP address if you are connected to a local wireless lan. The Wi-Fi Address is the address the Wireless AP gives you while the Data IP address will be the

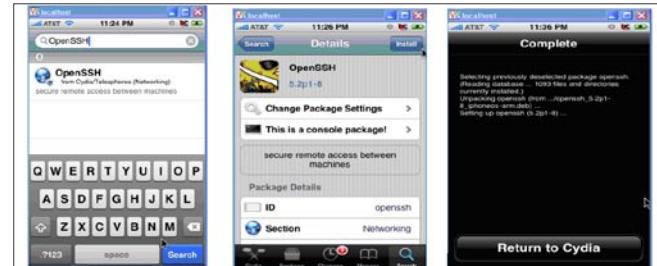


Figure 4. OPENSSH Installation

ATTACK



Figure 5. SBSettings and Installation

IP given to you by your service provider. In order to enable or disable a service simply tap its icon. As you can see SSH and wifi are enabled and indicated by the green icon color while Bluetooth has been disabled and indicated by its red icon color (Figure 5).

Next let's go out and grab Nmap and Metasploit. Just as we have done with previous installations. After both of those are installed some wireless reconnaissance software in this case Stumbler Plus for the IPHONE. Stumbler plus will allow you to scan for wireless access points that are close by and will give you some idea as to what type of encryption they are running and some other useful information. After installing Stumbler plus go to your desktop and install WinSCP that we downloaded earlier and download stumbler plus again from (<http://www.iphone.mysticwall.com/download/stumblerplus-1.2rev1.tar.gz>).

You should now be able to access the OpenSSH which we installed earlier on your Iphone. Login with the username root and the password that you chose earlier. Unzip the files you downloaded and then use WinSCP to browse for them. In WinSCP on your phone go to the root then go into applications. You should see a list of all your previously installed Iphone apps. In WinSCP on your PC located the stumblerplus.app you extracted earlier and select all the files within that directory and copy and paste them into the stumblerplus.app on the Iphone. A warning message will pop up telling you that you are overwriting files which is fine let it overwrite them all. Close WinSCP and you should now be able to run Stumblerplus.



Figure 6. Stumbler Plus & Nmap Scan

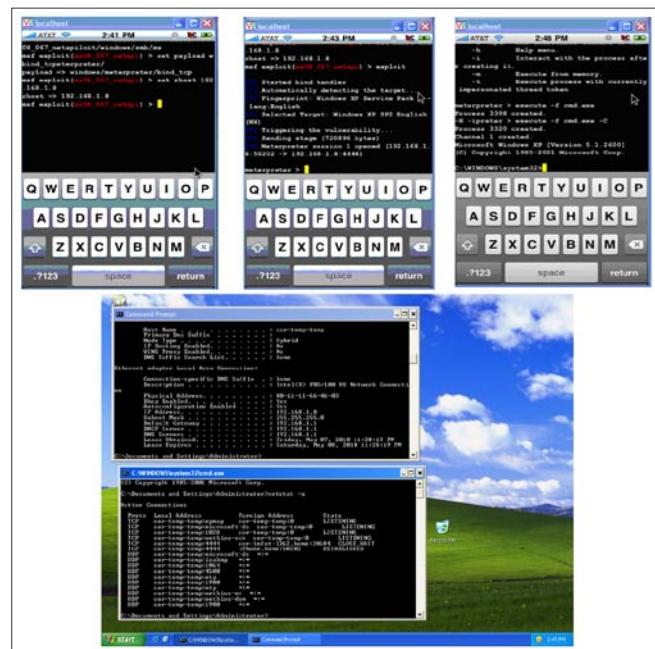


Figure 7. Metasploit & Windows Command Shell

iPhone Network Penetration

Now that we have everything installed successfully lets get to business open up Stumlerplus and do a search for wireless AP's by tapping the Scan button. In this case we will connect to the New Caprica AP shown here as it doesn't have any encryption enabled. Next we will Open nmap and see if there are any live hosts on our AP and what if any ports are available (Figure 6).

Next we will close down Stumbler Plus and Open Nmap and run a quick search for live hosts.

```
iPhone:~ mobile$ Nmap -vvv -P0 -sV 192.168.1.2-255.
```

As you can see we have several ports open here all are of the windows variety. Next we can open up Metasploit and try out a common exploit to see if we can pop a shell on this host. Here we will use the ms08_067_netapi with bind_tcp as our shell push back (Figure 7).

Conclusion

As we have demonstrated today with a little skill and the right tools a sophisticated attacker can take advantage of the right tools on the Iphone platform. Although the technology has not fully matured what we have looked at today proves beyond the shadow of a doubt that in the future attackers will be even more mobile and inconspicuous than your normal run of the meal hacker.

WARDELL MOTLEY JR.

Wardell Motley is a Systems Administrator for a Large clothing Manufactures in Dallas Texas. He is a member of the ISSA and in his spare time works as freelance IT security researcher.

EXCLUSIVE&PRO CLUB

NETIKUS.NET Ltd

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

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



Heorot.net

Heorot.net provides training for penetration testers of all skill levels. Developer of the 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



VINTEGRIS S.L

VINTEGRIS S.L is a company dedicated to IT security in Spain. We focus on development of authentications, web access control, password management and synchronization, and digital signature systems, to integrate into the IT of our customers. We also perform integration of third-party recognized security products. Most of our consultants are CISA and CISSP certified and our company is ISO/27001 certified.
<http://www.vintegris.com>
e-mail: info@vintegris.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



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/>

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

Testing

Flash Memory Forensic Tools – part two

This second part is focused on advanced tests done on flash memory embedded in a Nokia mobile phone. Tests presented in this article are not for all as they require a well furnished lab; even that what we try to demonstrate here is that – when flash mobile forensic will leave its infancy – there are some issues forensic officers should take in consideration.

What you will learn...

- This article will present some underestimated issues on flash memories forensic.
- Reader will also understand how some techniques already seen with hard drive forensic can be reused with success to avoid detection in flash memories too.

What you should know...

- For this second part, too, a basic introduction to digital forensic issues will be helpful (it is not a requirement).

First of all: is it possible to hide data in flash memory using techniques as seen in hard disk forensic? Unfortunately the answer is yes and for unexpected reasons, too. Outcomes presented in this article were updated in December 2009: we are working for a new and wider release of such tests and results, when ready, will be presented to public using same channel.

At the end of this article there are references mentioned in first and second part of paper.

Keywords

Mobile forensic, OneNAND, NAND, NOR, bad blocks, wear levelling, ECC, FTL

A brief digression on evidence metrics

Considering a digital device as body of evidence, it is possible to define some statements:

- E as the full set of evidences Existing on the device
- A as the set of evidences Acquired by forensic tools (i.e. dd)
- O as the set of evidences Observed (found) by the analysts

so that:

- Y is the ratio between Acquired evidences and Existing evidences [$A/E=Y$] and represents the quality of forensic tools used (1=better, 0=worse);
- K is the ratio between Observed evidences and Acquired evidences [$O/A=K$] and represents the analyst's skill (1=better, 0=worse);
- Z is the ratio between Observed evidences and Existing evidences [$O/E=Z$] and represents the overall quality of analysis (1=better, 0=worse) see Table 1.

Table 1. Quantitative relation between evidences, analyst's skill, and quality of tools

Units of evidences			Y (A/E) (tool quality)	K (O/A) (analyst skill)	Z (O/E) (overall qual- ity of analysis)
Existing (E)	Acquired (A)	Observed (O)			
100	100	100	1	1	1
100	80	80	0,8	1	0,8
100	80	60	0,8	0,75	0,6

Thus, a good tool with a good analyst gives an overall good analysis (case 1), a mediocre tool (case 2) or a mediocre analyst (case 3) will limit the overall value of examination. Of course this is just a quantitative and not qualitative measurement: the importance of each evidence is set aside see Figure 1.

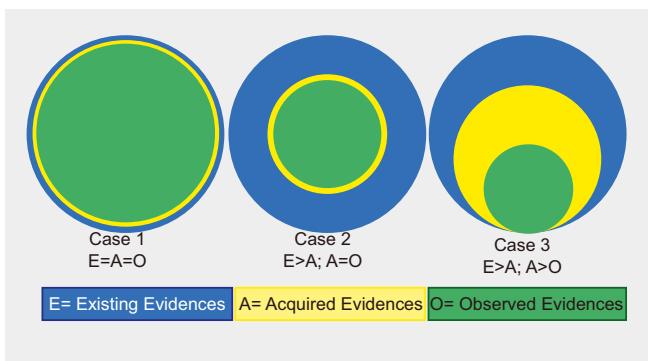


Figure 1. Quantitative relation between evidences, analyst's skill, and quality of tools

Logical vs Physical acquisition

Logical and physical acquisitions are already well defined in the NIST Special Publication 800-101 Guidelines on Cell Phone Forensics (Jansen and Ayers, 2007):

Forensic tools acquire data from a device in one of two ways: physical acquisition or logical acquisition. Physical acquisition implies a bit-by-bit copy of an entire physical store (e.g., a memory chip), while logical acquisition implies a bit-by-bit copy of logical storage objects (e.g., directories and files) that reside on a logical store (e.g., a file system partition). The difference lies in the distinction between memory as seen by a process through the operating system facilities (i.e., a logical view), versus memory as seen in raw form by the processor and other related hardware components (i.e., a physical view).

Physical acquisition has advantages over logical acquisition, since it allows deleted files and any data remnants present (e.g., in unallocated memory or file system space) to be examined, which otherwise would go unaccounted.

In the image below is given a representation of both methods, in case of memory not physically extracted from hosting device, that is, left on the phone and accessed with traditional means see Figure 2.

Proprietary cables with USB interface are used for both techniques, while JTAG or FBUS interfaces (where present) are mainly used for physical

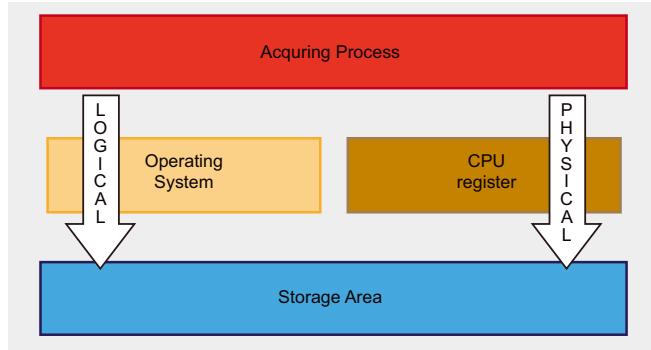


Figure 2. Logical vs. Physical acquisition for flash memory on the hosting device (not extracted)

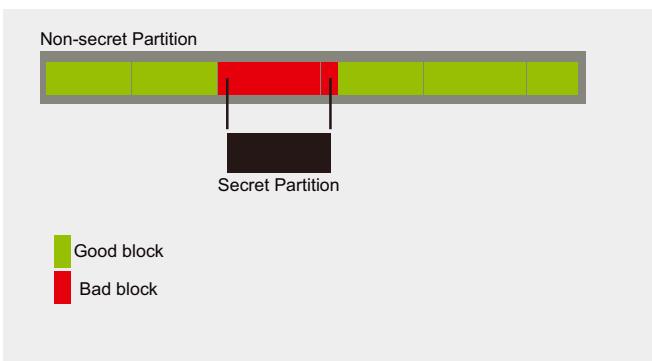


Figure 3. Hiding data in bad blocks (David, 2009)

acquisition; it is also possible get data via infrared and Bluetooth interface using OBEX protocol, but this is a method that poses some limitation and is generally less used (McCarthy, 2005). Some Nokia phones are now explored: registry addresses are blurred for confidentiality.

Flash peculiarities in the acquisition process

During this research it comes out the high level of confidentiality surrounding the flash technologies and market, so that nobody seems to be able to set a definitive point on how others can use or implement flash technologies: a problem reported since the begin of mobile forensic (Willassen, 2003). In an attempt to understand better what really happen inside a flash there were several meetings with highly skilled people from the flash manufacturing field and the focus was set on how to preserve integrity of evidence and grant completeness of acquisition. This is what came out:

Real effect of reclaim:

- garbage collection is a known activity but not so well documented for seized devices
- garbage collection is a background activity, this means that when a mobile phone is powered on, even in service mode, such activity could be autonomously triggered with the effect of destroying useful data in invalid blocks

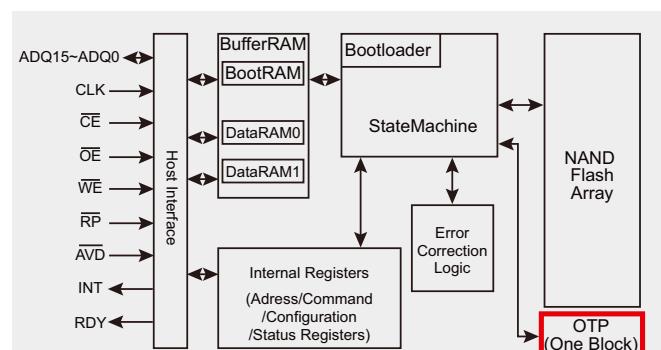


Figure 4. Block Diagram on a multiplexed OneNAND™

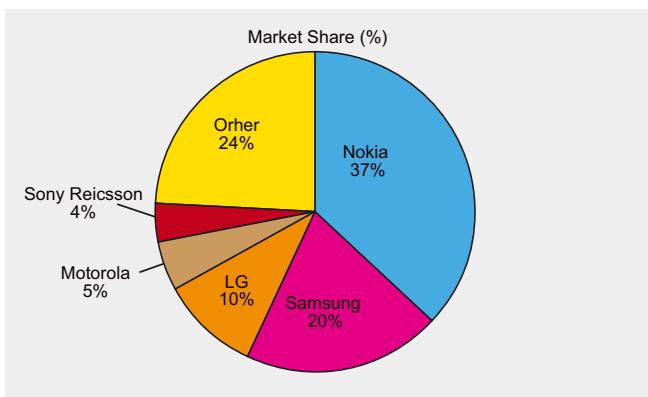


Figure 5. Worldwide Mobile Terminal Sales to End Users in 2Q09 (Gartner, 2009)

Effective management of bad blocks:

- if the FTL is embedded in the flash memory (like in case of managed flash) then it will be difficult to access and manage bad blocks because they will be hidden to the host file system;
- if the FTL is supplied from the host (like in case of raw flash) then there are chances to manage bad blocks properly and have direct access to them. Analogous experiences are reported with modern hard disks managed with GNU ddrescue (There is still an open debate on hard disk bad block management. Some interesting links are: <http://tech.groups.Yahoo.com/group/ForensicAnalysis/message/82>, <http://www.forensicfocus.com/index.php?name=Forums&file=viewtopic&t=2557>) (Carrier, 2005, Lyle and Wozar, 2007, Mukasey et al., 2008).

Security through obscurity

Even knowing the memory specs, manufacturers can apply autonomous decisions on how manage the chip: it can happen that a managed flash will be used with disabled features, or that a flash raw memory be customized as for manufacturer needs. Furthermore, due to high competition and Intellectual Property protection, generally, there are not public information on the chip used. At begin of the research some manufacturers were contacted to get some info: it was even difficult to know the destination of some branded components.

Bad management of good blocks

A block is considered bad when there are multiple bit errors that are not recoverable (Numonyx, 2008a). Like hard disks, NAND flash generally ships with a list of existing bad blocks set in a location defined by the manufacturer. Additionally, to this list will be added all future blocks that will fail to operate during device lifecycle. Forensic investigators are already aware of the possibility to manipulate Bad Block List to hide information (David, 2009) this aspect should not be underestimated in flash memories as they are able to store even larger quantity

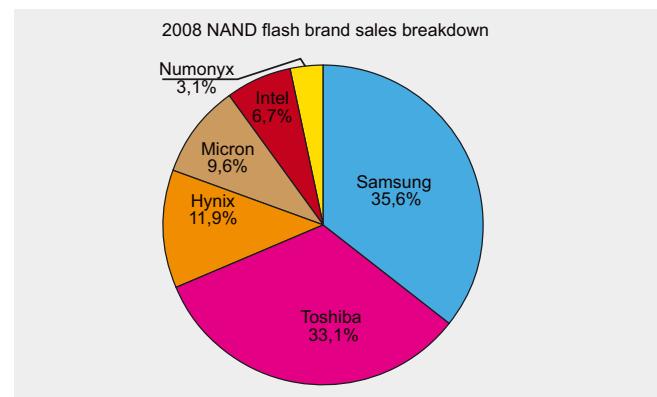


Figure 6. 4Q08 NAND flash brand sales breakdown (DRAMeXchange, 2009)

of data: a working OS could be as small as 50 MB (www.damnsmalllinux.org) or much less with Embedded distro (www.emdebian.org) see Figure 3.

Misuse of Hidden Protected Area

It could be possible for an hacker to store data even in the Hidden Protected Area also referred as One Time Programming (Samsung, 2007a). The size of this area is generally equal to one block but variants are allowed (Samsung, 2005c, Micron, 2006c); it can be blocked, but usually this task is left under hosting manufacturer care (ibid) see Figure 4.

Computer analysts already know the issue related to Host Protected Areas (HPA) and Device Configuration Overlays (DCO) in hard drives (Gupta et al., 2006, Carrier, 2005): with flash memories we have similar issues. In future works we plan to test the possibility to change (doubling) the dimension of such area and then to store and hide data in it.

How the choice of the flash memory and mobile phone was driven and the team was set

Simply, the choice of mobile phone and flash memory to use was made by statistics. Nokia is the best seller in the mobile phone market and Samsung is the leader in the NAND flash market see Figure 5 and 6.

Then the choice to use an OneNAND was made for its advanced characteristics and the Nokia model was chosen on the basis of a block of ten OneNAND available at moment. Numonyx has licensing agreement with Samsung to produce OneNAND™, so it was decided to call Numonyx for support and the folks there were happy to help. Then, was asked support to an advanced Nokia service repair centre that was willing to help, too: in few days a virtual team with high skilled people was s and ready to start. As this market is so hard-hitting, a low profile participation has been adopted.

How NOR and NAND are accessed on a Nokia N70

The implementation layout of NOR and NAND chips in a Nokia mobile phone (N70 model), is presented in the picture below (left). The combo

memory (NAND+SDRAM) flash is managed by a TI microcontroller unit (mcu) OMAP 1710. OMAP stands for Open Multimedia Application Platform and it is the application processor running with Symbian operating system (EPOC). The NOR flash is managed by the microprocessor RAP3G (3G Radio Application Processor). Evidences on mobile phone are stored in NAND flash: whatever means are used, to access the NAND storage area it is required to move through the OMAP processor (right) see Figure 7.

How OneNAND™ is accessed on a Nokia 6650F

The Nokia 6650F phone has been introduced on the market on 2008. The application memory of the device consists of NAND/DDR combo memory. The stacked DDR/NAND application memory has 512 Mbit of DDR memory and 1024 Mbit of flash memory (1024 Mb are equal to 128 MB). This is the phone we have chosen to be used for tests presented later: on the left the phone schematic, then two picture of the internal side (with indication of the OneNAND™), the relation between processor and flash memory and flash memory pins layout. Larger images are available in appendices see Figure 8.

How data on NAND are accessed via USB or JTAG on a Nokia 6120c

To perform a memory dump of the flash memory via physical acquisition on a Nokia 6120c, either with

a USB cable or a FBUS/JTAG interface, it is required processor involvement (in this case it is a RapidoYawe (The chip with HSDPA logic (Yawe) stacked on the RAP3G processor unit (RAPIDO) forms the RapidoYawe CPU)). In the tables below are presented schematics of connections between two devices (memory and processor). This phone will replace the Nokia 6650F in our tests, as explained later: the layout is very similar. Larger images are available in appendices see Figure 9.

Test Phase 1: preparing the phone

On a new flash memory (identical to the one on the testing Nokia mobile phone) were stored some data in four good blocks; such blocks were then marked as bad, by opportunely manipulating the relative spare area. Next, the original flash device embedded in the phone was replaced with the one with four bad blocks and the phone refurbished with original software: now, there is a working phone with data hided in bad blocks. The detailed procedure is in the appendices.

Test Phase 2. Feeding forensic tools with our phones: results and feedbacks

At beginning, when decision on which type of phone to use was made, it was considered an advantage to use a Nokia phone, due to its popularity. Not too much attention was paid on the specific model we were using: all in all there was an OneNAND™ inside and this was considered an advantage for the research. As

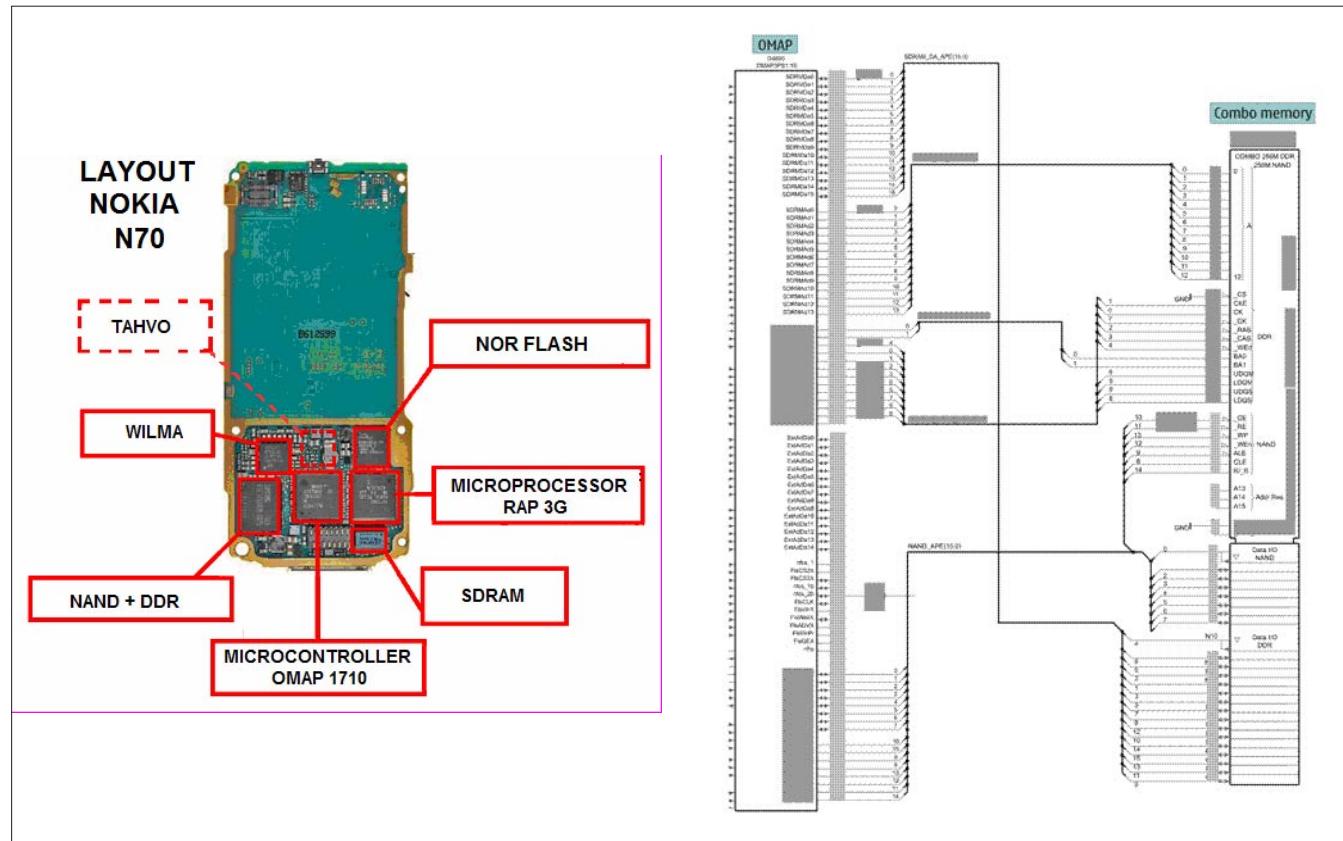


Figure 7. Layout of a Nokia N70 (left), and OMAP and NAND flash relation on Nokia N70 (right)

ATTACK

the testing memory was a raw NAND, we were optimistic forensic software would be able to acquire bad blocks because there were not embedded FTL layer could interfere with the imaging process.

Then, we used some of the best forensic software to test the acquisition of bad blocks from our phones, and this is what we got (in alphabetical order).

- CelleBrite UFED – This solution was not able to perform the physical acquisition.
- Logicube CellDEK – We were not able to perform any acquisition with CellDEK because the required module, even already ordered, was not available at time of examination.
- Micro Systemation XACT – This solution was not able to perform the physical acquisition.
- Paraben Device Seizure 3.1 – This solution was not able to perform the physical acquisition.

At this stage, we decided to speak directly with technical support of these companies and tell them the problem we faced. An email was sent either to companies aforementioned and to others that have been tested their products with NIST (as reported in the CFTT web page http://www.cftt.nist.gov/mobile_devices.htm). The test of the emails is reported in appendices. So far, these are the replies we got:

CelleBrite, Micro Systemation and Paraben confirmed the inability of their solution to get physical acquisition of our phone (even they can do with others); Guidance Software, Logicube, and Susteen did not reply.

For what we tested and understood, with these solutions and the phone we used, if sensitive data are hidden in bad blocks they will go undetected. Furthermore, with this software, good blocks with wrong ECC (i.e. due to power failure) could hide valid data to forensic analyst.

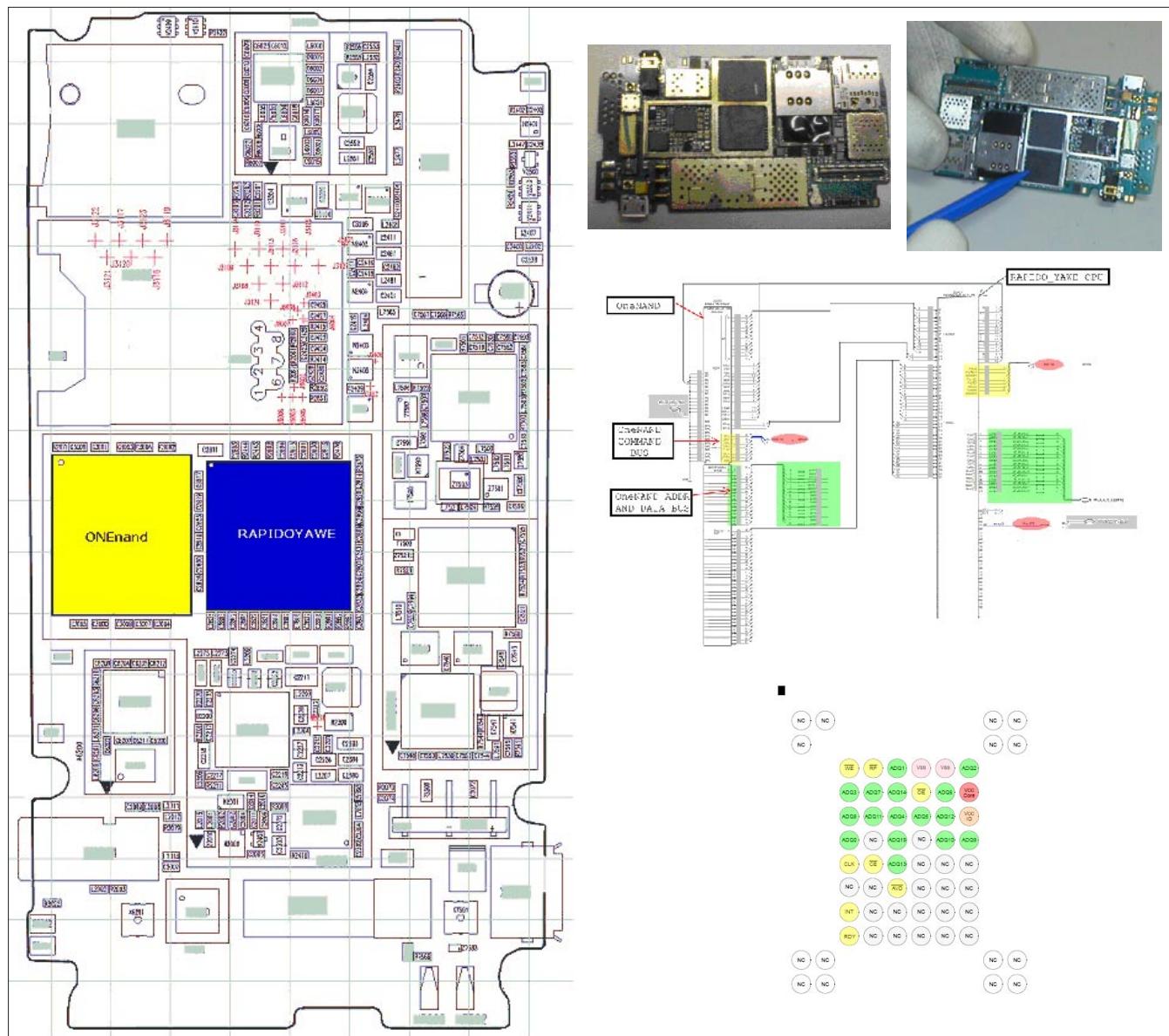


Figure 8. From left to right (clockwise): Nokia 6650F layout; the internal hardware, stencil pointing at the OneNAND™ flash; schematic showing connections between CPU and OneNAND™; and generic OneNAND™ pins layout

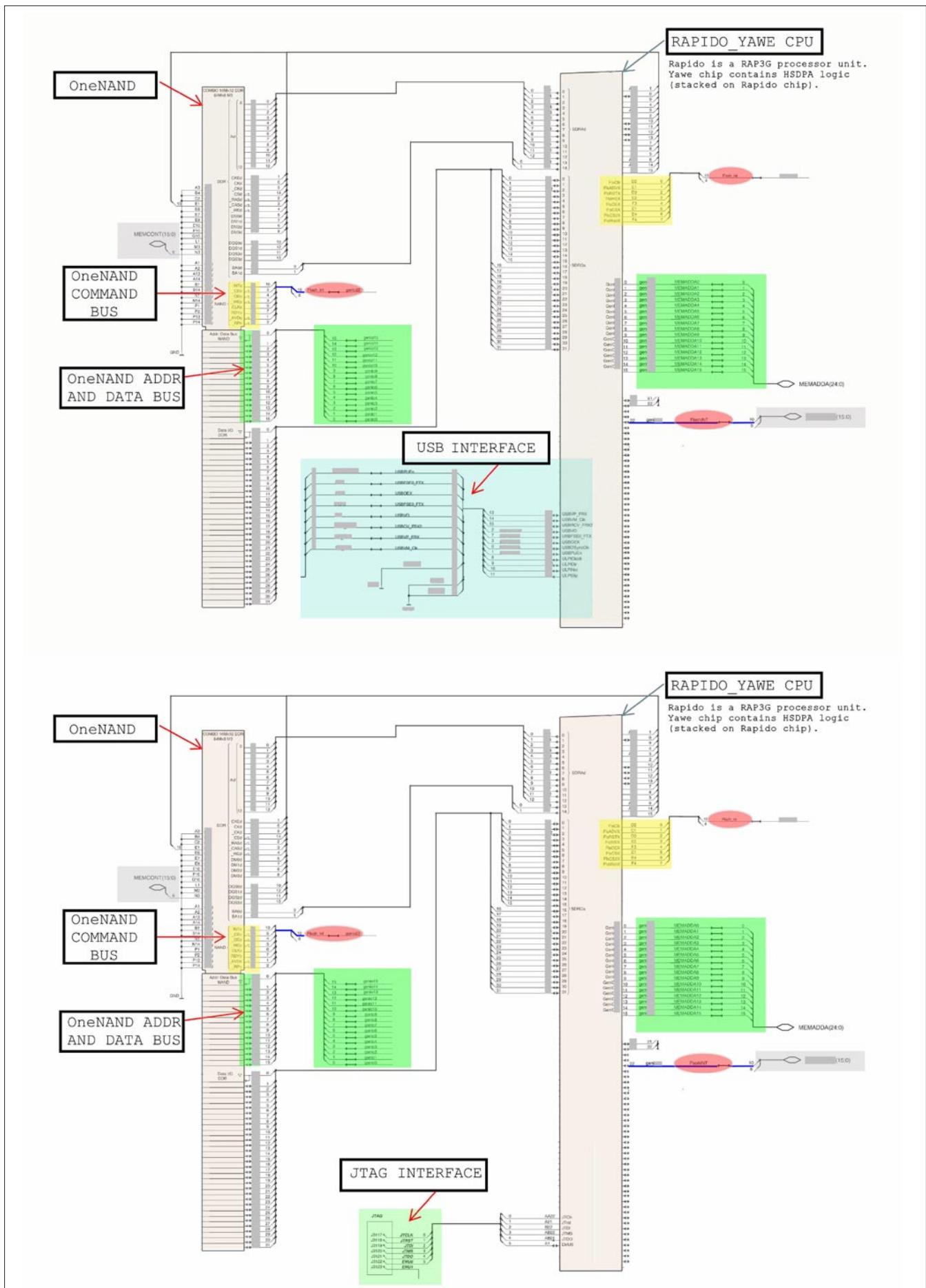


Figure 9. Adapted layout of access to NAND memory via USB (top) or JTAG (bottom)

Reporting to forensic metrics

Our test take a lot of time to be set and only few minutes to be waived: we were a little disappointed. Going back to evidences metrics seen before, we should say that any forensic tool not able to deal with bad blocks (completeness of evidence) should fall at least in the case number two. This without considering underground Reclaim activities, yet (the effect of Reclaim on integrity of evidence need further analysis).

Physical acquisition as option: what says the NIST

Many companies are proud to say their products have been successfully tested with NIST, but what exactly say a NIST report on mobile physical acquisition and completeness of evidences acquired?

A first answer can be found either in the version 1.1 (NIST, 2008) or 1.2 (NIST, 2009) of *GSM Mobile Device and Associated Media Tool Specification and Test Plan*, where is reported in the section CFT-IMO-05/06 and CFT-IMO-04, respectively, that physical acquisition is an optional feature. For analyst with hard disk forensic background, it could seem a little strange considering physical acquisition an option.

Furthermore, the word *completeness* is reported in the *2004 Digital Data Acquisition Tool Specification*, in the *2005 Digital Data Acquisition Tool Test Assertions and Test Plan Draft 1 for public comment Version 1.0*, in the *2008 GSM Mobile Device and Associated Media Tool Specification and Test Plan (ver 1.1)* but not in the *GSM Mobile Device and Associated Media Tool 3 Specification and Test Plan (ver 1.2)*: the question is why completeness of evidences is then shifted to be an optional feature. The NIST were contacted either at institutional and authors' addresses (email in appendices). This is the synthesis of answers got – the source asked not to be cited, but to refer to CFTF site

- Optional test cases are treated as Core test cases IF the tool provides the capability defined by the test case. Unfortunately, all mobile forensic tools do not have the ability to perform a physical acquisition at this time. The CFTT formal testing methodology validates that tools perform as they are designed not as one might wish them to.

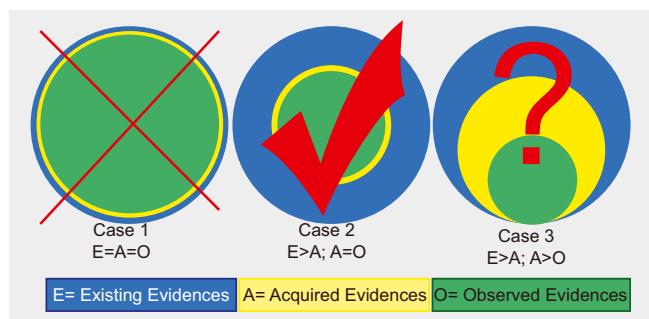


Figure 10. Quantitative relation between existing evidences, quality of tools, and analyst's skill

- Physical Acquisition is not an unreachable limit, but some tools are designed only for logical acquisitions. The specification and test plan state that if the tool provides the functionality optional cases and assertions are tested as if they are core. By following the CFTT formal testing methodology it allows all tools that have the ability to acquire data from mobile devices to receive a fair validation.

The aim of this paper is not to argue with NIST, but for what is written in the second sentence above, test on tools designed either for logical and physical acquisition, like Cellebrite UFED 1.1.05, should report physical acquisition in the core features: but by reading *Test Results for Mobile Device Acquisition Tool: Cellebrite UFED 1.1.05* it is possible to see that physical acquisitions is reported in the CFT-IMO-05 section, as an optional feature.

In the email sent to NIST, author suggests to shift this feature from optional to core section, because a document released from so regarded source, should not allow a workaround of an important point like this.

A confidential answer

We asked to forensic software houses cited above, why it is so difficult to perform a physical acquisition of non-volatile memory (We should not forget that on OneNAND we have both volatile memory (DDR) and non-volatile memory (NAND)) embedded in phones made by different manufacturers but using the same raw flash memory and the same I/O interface. This is the answer got from a source asked not to be disclosed:

- IP protection: many phone manufacturers need to protect their know-how, so they encrypt some area of the memory and use proprietary bootloading solutions. This means that a forensic software house should be able to decrypt, without altering, the content of the evidence and also it need do this for any mobile phone on the market: a very onerous task that in the lack of a collaboration between chip manufacturer and software developers is too uneconomical. When a flasher is used to change IMEI or unlock a phone it exactly circumvents this protection (for this, the source states further that in future mobile phones, JTAG interface will be disabled to prevent illegal activities).
- Market alliance: for reasons seen above, forensic solution providers could not have interest to release something harmful for phone manufacturers because otherwise the latter will not be anymore cooperative with them.

The ONFI project

The resolve the problem of disorder in the flash market, some manufacturers decided to setup a consortium

to define some standards: it is the *Open NAND Flash Interface* (ONFI) consortium. *The ONFI is an industry Workgroup made up of more than 80 companies that build, design-in, or enable NAND Flash memory, dedicated to simplifying NAND Flash integration into consumer electronic products, computing platforms, and any other application that requires solid state mass storage.* We define standardized component-level interface specifications as well as connector and module form factor specifications for NAND Flash (<http://onfi.org>).

Future works AND CALL FOR HELP

We plan to do some feature works especially to test the effect of reclaim in a controlled environment (like a mobile phone left in standby), and capture (by sniffing) and analysis of data travelling on the bus to/from mcu and NAND. As this tests will require financial as well as technical support, everybody interested to support this research can express her/his availability via email directly to me.

Credits

Author wish thanks Numonyx Flash Group, Nokia Lab Southern Italy, Polizia Postale e delle Comunicazioni for their help and support.

Conclusion

In this paper has been attempted to offer a wide overview of forensic analysis of non-volatile flash memory. Starting from academic and industrial literature, we ended with a practical and documented test in which some data were hidden in memory blocks (then marked as bad) to verify if it was possible to foul the acquisition process of nowadays forensic solutions. It was demonstrated that hiding data in such blocks is achievable: none of the software tested was able to get a physical acquisition of the flash memory. Furthermore a suggestion to considerer physical acquisition a core feature was sent to the NIST to make them more aware of the problem of data hiding in flash memories and the need to grant the completeness of evidence.

Author is available via email for any enquiry on the topic.

SALVATORE FIORILLO

Author is a security consultant and researcher focused on weaknesses in the logic of physical and digital systems. He holds a Master of Computer Security accomplished in Western Australia and the ISO 27001 certification, and have trained hundreds of security officer either of public and private organizations. As consultant he works only for few, interesting and selected customers. sfiorillo@theosecurity.com

A D V E R T I S E M E N T



**DATA RECOVERY SOFTWARE
FOR NTFS DRIVES**

<http://recoverymechanic.com>



Securing public

services using Tariq

When I first read about the port-knocking concept was really amazed how such service can help us secure other less secure services such as telnet, rsh, etc. But after a while I realized that it was a great solution even to the ground built up secure services such as SSH (Secure Shell)!

What you will learn...

- What port-knocking is, and the benefit of using it,
- Howto secure a public service such as SSH using Tariq.

What you should know...

- Howto configure a Linux iptables firewall,
- Difference between iptables firewalls policies.

Yes, even the most secure services whom was built from the scratch with security in mind fell to its knees when a 0day vulnerability was exposed CVE-2008-0166 [1][2], enabling attackers to conduct brute force guessing attacks against cryptographic keys, leading to a remote compromise. From here imagine how much a port-knocking solution can be helpful to us.

I think after reading the intro, some are starting to ask questions:

- What is this port-knocking?,
- Is port-knocking Security Through Obscurity?,
- What's new?.

What is this port-knocking?

Well first lets define the concept port-knocking. Simply, its a technique used to open port(s) on a remote firewall by generating a connection attempt on a pre-specified set of closed ports. Once the correct sequence of connection attempts is received, the firewall dynamically modifies its rules to allow the host which sent the connection attempts to connect over to specific port(s).

Is port-knocking

Security Through Obscurity?

Researchers are still arguing about the port-knocking technique and accuse that its "Security Through Obscurity"! This is a long going argue going out there

about this technique, but the true answer for me is: Port-knocking is a concealment in the same spirit as passwords and encryption keys [3].

What's new?

What's new in the port-knocking arena, is [Tariq](#) :)

Tariq Overview

[Tariq](#) is a new hybrid port-knocking technique, that uses Cryptography, Steganography, and Mutual Authentication to develop another security layer in front of any service that needs to be accessed from different locations in the globe.

[Tariq](#) was developed using python and scapy by me to fulfil my Ph.D. Research. We had to use a new methodology that can communicate in an unseen manner, making TCP Replay Attacks hard to be issued against [Tariq](#). We also wanted the implementation to listen to no ports, or bind itself to no socket for packets exchange, so that [Tariq](#) won't be exposed himself to a remote exploit.

What does Tariq mean?

In English, it means knocking, hammering or coming at night :)

How does Tariq Work?

[Tariq](#) works by first running the python application [TariqServer](#), the server shall be running in sniffing/packet

capturing mode, and the clients shall be using the python application `TariqClient` to open ports or executes remote commands on those server(s). The whole scenario can be summerized as following:

- Servers run the python app `TariqServer`, and clients open ports or executes remote commands on those servers by running the python app `TariqCleint`,
- `TariqClient` adds the action (open port/execute command) to a picture using Steganography,
- `TariqClient` uses the Steganography picture as a packet payload,
- `TariqClient` adds the payload to TCP SYN packet(s) to be sent on pre-specified ports (configured on the `TariqServer`),
- `TariqServer` captures the packets and makes sure it contains a picture,
- `TariqServer` extracts the commands from the Steganography picture. This is to make sure that the packet really holds a clients request,
- `TariqServer` selects a random number and encrypts it using the client's GnuPG public key,
- `TariqServer` uses the encrypted random number as a packet payload,
- `TariqServer` crafts a packet holding the payload and sends it to the client as if it is a reply to the clients SYN Packets. This is to complete the mutual authentication process,
- `TariqClient` receives the packet and extracts the payload,
- `TariqClient` decrypts the payload using its GnuPG private key,
- `TariqClient` uses the random number received as a packet payload to be sent to server after encrypting it using the `TariqServer`'s GnuPG public key. This is to ensure that he is who he claims to be (completing the mutaul authentication process, from the clients side),
- `TariqServer` receives the packet, extracts the payload, and decrypts it to make sure that he received the random number he sent to the client,
- `TariqServer` after verifying that the client is ligitmate executes the commands extracted from the picture sent in the first place.

And thats how `Tariq` works, no listening, no sockets, and no ports open, just pure packet crafting!

Why Is Tariq Needed?

Any host connected to the Internet needs to be secured against unauthorized intrusion and other attacks. Unfortunately, the only secure system is one that is completely inaccessible, but, to be useful, many

hosts need to make services accessible to other hosts. While some services need to be accessible to anyone from any location, others should only be accessed by a limited number of people, or from a limited set of locations. The most obvious way to limit access is to require users to authenticate themselves before granting them access. This is were `Tariq` comes in place. `Tariq` can be used to open ports on a firewall to authorized users, and blocking all other traffic users. `Tariq` can also be used to execute a remotely requested task, and finally for sure `Tariq` can close the open ports that have been opened by a previous `TariqClient` request.

`Tariq` runs as a port authentication service on the iptables firewall, which validates the identity of remote users and modifies firewall rules (plus other tasks) according to a mutual authentication process done between `TariqServer` and a `Tariq` client. `Tariq` could be used for a number of purposes, including:

- Making services invisible to port scans,
- Providing an extra layer of security that attackers must penetrate before accessing or breaking anything important,
- Acting as a stop-gap security measure for services with known unpatched vulnerabilities,
- Providing a wrapper for a legacy or proprietary services with insufficient integrated security.

Why Is Tariq Secure?

- `Tariq` Server's code is very simple, and is written completely using scapy (python),
- The code is concise enough to be easily audited,
- `Tariq` needs root privileges to adjust iptables rules, and perform remote tasks,
- `Tariq` does not listen on any TCP/UDP port, which means no sockets is used. `Tariq` uses scapy's capabilities to sniff the incoming traffic and uses Packet Crafting techniques to reply back to an legitimate client,
- The communication protocol is a simple secure encryption scheme that uses GnuPG keys with Steganography constructions. An observer watching packets is not given any indication that the SYN packet transmitted by `Tariq` is a port knocking request, but even if they knew, there would be no way for them to determine which port was requested to open, or what task was requested to be done as all of that is inserted into a png picture using Steganography and then encrypted using GnuPG keys,
- Replaying the knock request later does them no good, and in fact does not provide any information that might be useful in determining

the contents of future request. The mechanism works using a single packet for the mutual authentication.

Installation

Requirements:

- Python >= 2.6
- python-imaging – *Python Imaging Library (PIL)*
- GnuGP
- Scapy
- A recent Linux kernel with iptables (eg. 2.6)

Preparing the Client

Preparing GnuPG

You need to create a directory for gnupg and generate a pair of keys using the following commands:

```
mkdir /etc/tariq/.client-gpg  
chmod 600 /etc/tariq/.client-gpg  
gpg --homedir /etc/tariq/.client-gpg -gen-key
```

You need to export client's public key:

```
gpg --homedir /etc/tariq/.client-gpg -a --export  
tariq@arabnix.com > key.pub.txt
```

Configuring the client

Edit the `client.conf` file to specify the client gpg directory and the default gpg user:

```
client_gpg_dir=/etc/tariq/.client-gpg  
user=tariq@arabnix.com
```

And specify the image directory used for steganography, containing at least 1 reasonable png image file, just like the one included as a sample `sample.png`:

```
img_dir=/usr/share/TariqClient/img
```

Now specify the default secret knock sequence to match the sequence configured on the Tariq server.

```
secret_ports=10000,7456,22022,12121,10001
```

Note: you may pass the gpg user and knock sequence as arguments to `TariqClient` (see howto use section).

Installing The Server

After installing the requirements, the first step is to download, unpack, and install Tariq. Tariq can be downloaded from: <http://code.google.com/p/tariq/>. Once this is done, we need to configure the server.

Preparing GnuPG

You need to create a directory for gnupg using the following commands:

```
mkdir /etc/tariq/.server-gpg  
chmod 600 /etc/tariq/.server-gpg
```

You need to import and trust the client(s) public key(s):

```
gpg --homedir /etc/tariq/.server-gpg --import <  
client.pub.txt  
gpg --homedir /etc/tariq/.server-gpg --edit-key  
tariq@arabnix.com
```

Then select trust (5)

Preparing iptables

Create an iptables chain to be used by tariq server:

```
iptables -P INPUT DROP  
iptables -N tariq  
iptables -A INPUT -j tariq  
iptables -A INPUT -m state --state ESTABLISHED,RELATED  
-j ACCEPT
```

Optional: you may specify a range of ports to be filtered (dropped) in case you are running normal services on the same box:

```
iptables -A INPUT -p tcp -m tcp --dport 1000,65535 -j  
DROP  
iptables -A INPUT -p udp -m udp --dport 1000,65535 -j  
DROP  
iptables -A INPUT -p tcp -m tcp --dport 80 -m state --  
state NEW -j ACCEPT
```

IMPORTANT NOTE: Do not use the REJECT target with tariq.

Configuring the server

Edit `server.conf` and specify the correct sequence of ports, by using the `secret_ports` variable. Example:

```
secret_ports=10000,7456,22022,12121,10001
```

Now specify the server's gpg path:

```
server_gpg_dir=/etc/tariq/.server-gpg
```

Specify the iptables chain name you have created for tariq:

```
iptables_chain=tariq
```

Now please adjust the iptables chain name used to open ports for a successful knock:

On the 'Net

- <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-0166> – Mitre's CVE dictionary CVE-2008-0166,
- <http://www.debian.org/security/2008/dsa-1571> – DSA-1571-1 openssl – predictable random number generator,
- <http://www.cipherdyne.org/fwknop/docs/SPA.html> – Michael Rash, Developer of the SPA technique.
- <http://code.google.com/p/tariq/> – Current Tariq project home page.

```
open_tcp_port=-A tariq -s {ip} -p tcp -m state --  
state NEW -m tcp --dport {dport} -j  
ACCEPT  
  
open_udp_port=-A tariq -s {ip} -p udp -m state --state  
NEW -m udp --dport {dport} -j ACCEPT
```

```
./TariqCleint -u tariq@arabnix.com example.com E service  
httpd restart
```

Another example, here I'm sending an echo message to the box:

```
./TariqCleint -u tariq@arabnix.com example.com E echo  
"Hello, It's me tariq"
```

Finally to close the port you requested to open, all you need to do is either initiate a close port command or the `TariqServer` shall check after a prespecified period of time if there is some activity or not on that port, if there is, `Tariq` shall leave the port open, if not `Tariq` shall request the close of that port. The command to close the port is as simple as this:

```
./TariqCleint -u tariq@arabnix.com example.com C 22
```

As we saw, `Tariq` enabled us to create another layer of security which needs to be penetrated in order to reach or penetrate any of the services we are using on our Linux box (for example: SSH server). This security layer that `Tariq` added shall make it very difficult for attackers to gain remote access to our servers, and shall really make them think twice before spending lots of time trying to figure out how shall they reach the box, because how can they discover a vulnerability in something that isn't seen? :)

Advanced Configuration

Sniffing Specific Ports Only – Sometimes you might need to run `Tariq` on a box running different services for example webserver (port 80). This can be done by adjusting the `*sniff_range*` variable in the servers configuration file*.*

This shall make `Tariq` sniff or capture packets destined to that port range only, without interfering with packets destined to our webserver (port 80), so no packets shall be dropped.

Random number (blob) Size – you can also adjust the random number's size sent by `TariqServer` to the `TariqClient` as the challenge by the variable `*min_random_blob_size*` and `*max_random_blob_size*`.

Working Threads – You can also increase the number of working threads of the `TariqServer` in case you have a wide number of users to serve and running on a heavy traffic box using the variable `*threads_n*`. Also found in the server's configuration file.

Howto use tariq

To start running `tariq` server, just run the following command using user root:

```
./TariqServer
```

Now that you have `tariq` server running, the firewall rules configured on the server, and your profile installed on the client, you're ready to run some commands remotely or open some ports. Using user root, to open, for instance, ssh (22) on the remote server (`example.com`), all you simply need to do on the client, is run:

```
./TariqCleint -u tariq@arabnix.com example.com O 22
```

If you don't want to open a port but perform a remote command for instance restarting the `httpd` service on the box, you don't need to login remotely and do it yourself and still working with the default drop firewall. All you simply need to do on the client is run the following command:

ALI HUSSEIN

The author has been working as a network security officer for different large companies for more than five years. His day to day activity is related to firewall auditing, IDS/IPS, and policy enforcement. He is currently a Ph.D. student, holding an MS.c. degree in Computer Information Systems, and a BS.c. degree in Computer Science. Throughout his working career he managed to gain a couple of well known technical certificates such as: CNI, CLP10, CLA10, CLDA, IBM Certified Specialist – System p Administration, Novell Linux Specialist, and RHCE.

Beginner's Guide to Cybercrime

Understanding Attack Methodologies and a More Proactive approach to Defense

If you are a regular reader of Hakin9 Magazine, you probably already know a great deal about hacking. But do you know the difference between traditional crime and cybercrime? Do you know where are the cybercrime magnets?

What you will learn...

- Types of Cybercrime Attacks
- CyberCrime Magnets
- The 4D's and The Risk Formula
- Proactive Countermeasures

What you should know...

- Basic „Hacking“ Knowledge
- Different Types of Crime
- Finding Vulnerabilities
- Testing Security Tools

How about why nothing with an IP address is secure and why traditional countermeasures such as firewalls, anti-virus and intrusion detection fail? Would you like to learn new methods to proactively defend against attacks? If so, you've come to the right place.

First, let's start with a basic understanding of traditional crime vs. cybercrime. There are parallel crime methodologies between crime in the *real* world and the *digital* paradigm enabled by the internet protocols including the world wide web.

Traditional criminal techniques involve burglary, deceptive callers, extortion, fraud, identity theft and child exploitation, to name a few. In Cybercrime we experience the same end results using from hacking, phishing, Internet extortion, Internet fraud, identity theft and child exploitation (sources: uscert.gov, cybercrimes.gov and privacyrights.org see Figure 1)

If you take a few moments to visit PrivacyRights.org and click on the Chronology of Data Breaches, you'll notice over 350 million *personally identifiable information* (PII) records have been lost, stolen and hacked. This information is about breaches in the United States of America, alone. So do you still think you are secure or believe your anti-virus and firewall can truly secure your network or personal computer?



The Prevalence of New Malware

Most of the breaches happen because of new malware and more innovative malware. So let's start our journey with the basics of malware. What is it? Is it a virus, Trojan, worm, rootkit, botnet, zombie, keylogger, adware or spyware? It is all of these things and some are combined into what is known as *blended threats*.

Is your computer infected with malware? It is highly possible, as one study claims that 30,000 computers are becoming infected every day with new malware, known as zero-day (this means the day it was released and before an anti-virus vendor has a signature test for it), while still running firewalls and anti-virus software.

Do you think some of the web sites you visit could be infected with malware? At least $\frac{1}{2}$ of the Top 100 sites, particularly social-networking sites such as Facebook or YouTube, support user-generated content, which is becoming a significant way to disseminate malware and conduct fraud. On Facebook and MySpace and other social-networking sites, there's an explicit sense of trust.

Do you pay your bills online? Criminals seized control of the CheckFree Web site and attempted to re-direct users to a Web site hosted in Ukraine that tried to install malware on victims' computers. CheckFree has more than 24 million customers and controls 70% to 80% of the online bill-payment market.

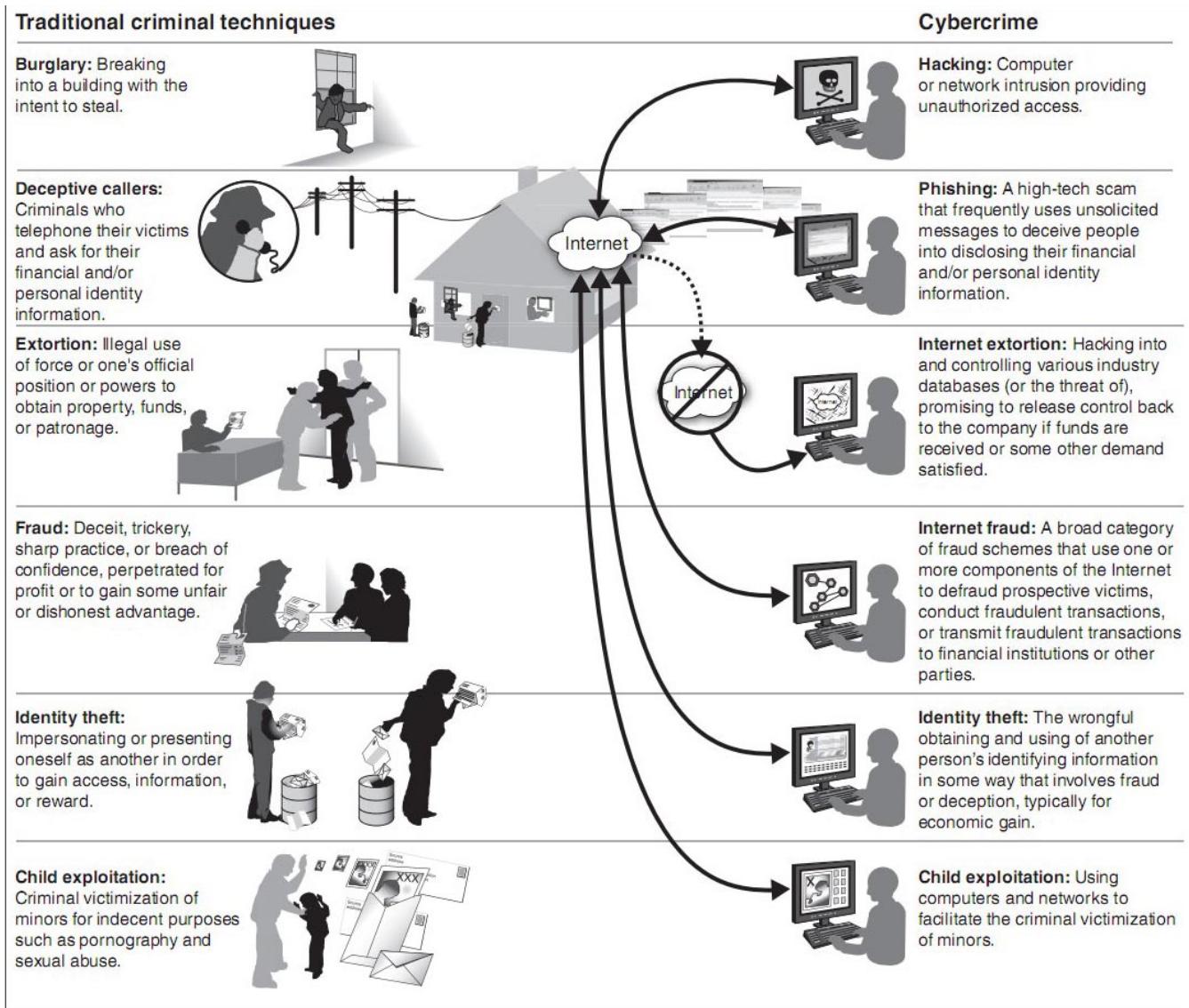


Figure 1. Traditional Crime vs Cybercrime

Much of the new malware is specifically designed to propagate across USB sticks. For example, the picture frame you just bought at Walmart using a USB connection might have come with zero-day malware from China. In addition, they work their way onto file servers using the Structured Message Block (SMB) protocol – that includes Linux and Windows file servers and network-attached storage devices. Some of this malware is so sophisticated, it finds data files such as .doc, .xls, .wav, .mp3, .pdf and others to infect so when someone else opens them, they too become infected.

Don't think you are safe at home, either. Cable networks are loaded with peer attackers. Most likely, a trusted telecommuter is using an insecure, hacked laptop with a key logger coming in securely into your network through an encrypted VPN tunnel.

Cloud Computing – A Malware Magnet

My next article will delve more deeply into Cloud computing and related security risks but for now, let's

just say the *Cloud* is also a cyber crime magnet. Why? Because cloud computing has shifted the paradigm for risk. The cloud offers low overhead in return for powerful remote business functionality. In return, you face the risk of data leakage, cloud attacks and cloud infections. You most likely will not know if and when it happens because of the remote aspects and the pervasive nature of the Cloud.



Secure Wireless Networking – Easily Hacked

Wired Equivalent Privacy (WEP) was the first commercial algorithm and attempt to secure wireless networks using the IEEE 802.11 standard. Because wireless networks broadcast messages using radio waves, they can more easily be eavesdropped than traditional wired local area networks. It was released in 1997 as an attempt

to provide confidentiality that would be comparable to that of wired networks. However, in less than four years, various weaknesses were uncovered in WEP and today, it can be cracked in minutes.

Then, just a few years later in 2003, along came *Wi-Fi Protected Access* (WPA) and later updated to WPA2 in 2004. Today, both WEP and WPA are widely deployed, yet with new tools such as BackTrack v4.0, anyone can gain access to a secure wireless network in a matter of minutes. In addition, most wireless routers have critical flaws known as *Common Vulnerabilities and Exposures* (CVEs). Now, you can break into the admin interface of a wireless router by sending malformed packets from your laptop without worrying about cracking the encryption. Just visit the *National Vulnerability Database* (NVD) located at <http://nvd.nist.gov> and type in *wireless* to see where the holes are located.

Is VoIP More Secure than Wireless?

So if wireless networks are not secure, would *Voice over IP* (VoIP) be better off, as they are usually, physically wired? The answer is no. There are dozens of VoIP holes, also found under the NVD. Some of these can be exploited by freely available tools online. These tools will allow you to take over the administrative console of the VoIP server by exploiting just one CVE – remember, all it takes is one hole and you can find many exploits. VoIP is

also easily susceptible to a man in the middle attack. A sample exploit known as *Voice over Misconfigured IP Telephony* (aka VOMIT) allows you to playback conversations that occurred earlier. Hackers simply use a TCP/IP ethertrace utility such as wireshark, save a ‘dump’ file of network traffic and then run the file through VOMIT to get a WAVE file of prior conversations.

What about other wireless communication devices such as a Blackberry, an iPhone, an iTouch or an iPad? My first question is – do they really belong on the ‘corporate’ network? If so, how do you know when they come and go, along with other portable devices and laptops? How do you stop them from bringing malware into the network? How do you stop them from being used to steal or leak confidential data? If you can’t control, track and manage assets, how can you claim that your network and your data is secure? You cannot. In fact, nothing with an IP address is secure. No device is safe. All IP-based devices are exposed to exploitation. Why? Because they are all targets – they can be spoofed, infected, remotely controlled and probably already are infected with some form of zero-day malware.

Traditional Countermeasures All Fail!

Anti-virus utilities are usually one to seven days BEHIND the current malware threat. With today’s malware, they are usually infected without knowing it. Just try *AVKILLER* as one of 400,000 sample pieces of zero-day malware to find out for yourself how serious this problem has become. Firewalls are easily circumvented or used as part of an exploit because of their exploitable holes (CVEs). Finally, *Intrusion Detection System* (IDS) detects odd or mal-behaving traffic AFTER the infected system or hacker system has breached the gates. To understand why these security countermeasures all fail, you need to understand the root cause of exploitation. CVEs are holes and are exploited daily. Let me give you a simple example: although there might be 9,000,000 signatures in your McAfee or Symantec Anti-virus scanner database (and growing exponentially), there are only about 43,000 CVEs.

If you close just one CVE, for example, you can block over 110,000 variants of W32 malware. If you aren’t visiting <http://nvd.nist.gov> to see what kind of exploitable holes you have in your network, cybercriminals CERTAINLY are... because everything with an IP address has a CVE, so, you need to figure out which ones are critical holes and how to patch, reconfigure and remove them. This is also known as *system hardening* and most folks seem too busy to find the time to go after the root cause analysis



and stay in reactive mode.... cleaning old viruses, patching one hole while opening another. You might think you are defending your castle with traditional countermeasures like bows, arrows and spears, however, today's cybercriminal is flying into your castle, behind the moat, using an apache helicopter, night goggles and a silencer.

Proactive Defense

- Learn and use the secret formulas

I've actually come up with a few simple formulas to help you understand how to reduce risk, comply with regulations and harden your systems. The first formula is based on US Military basic war tactics and is called the four D's. They are:

- Detect – awareness of a threat
- Deter – preempting exploitation
- Defend – fighting in real-time
- Defeat – winning the battle!

The second formula is well known in the network security circles and is called the *Risk Formula*, as follows:

$$R = T + V + A$$
$$(R)isk = (T)hreats + (V)ulnerabilities + (A)ssets$$

So, to fully understand your risks, you need to deal with:

Threats = Cybercriminals, Malware, Malicious Insiders

Vulnerabilities = Weaknesses that Threats exploit

Assets = People, Property, Your Network, Devices, etc.

Now, let's put these two formulas together – the 4Ds and the Risk Formula to build a more proactive, next generation defense:

$$4Ds \times R = [4Ds \times T] + [4Ds \times V] + [4Ds \times A]$$

You'll never be 100% secure but you can dramatically reduce your risk and proactively defend your organization by proactively containing and controlling threats, vulnerabilities and assets. Using the 4Ds with the Risk Formula:

- Threats need to be detected, deterred, defended against and defeated in real-time or expect DOWN-TIME.
- Vulnerabilities need to be detected, deterred, defended against and defeated (ie removed – system hardening, reconfiguration, patching, etc.) as quickly as possible or expect to be EXPLOITED.
- Assets need to be controlled – which ones gain access to your network/infrastructure and those

that are trusted but weak or infected need to be quarantined in real-time or expect MALWARE PROPOGATION.

Proactive Defense

- Employee Awareness and Training

With these two formulas in place, you'll still need to account for the most important challenge to network security – untrained and easily exploited employees. You'll need to begin to invite employees to a quarterly 'lunch and learn' training session, give them 'bite-sized' nuggets of best practice information. Maybe even consider giving them an award once per year to the best INFOSEC compliant employee who has shown an initiative to be proactive with your security policies, the 4Ds and the Risk Formula.

Remember, if you can keep them interested, they will take some of the knowledge you are imparting into their daily routines. That's the real goal. Launch a 4D and Risk Formula educational campaign so all employees in your organization to join your mission to protect corporate information. Create your own 'security broadcast channel' via email or really-simple syndication (RSS) and get the message out to your corporate work force. You can also give them 'security smart' tips or alert them to a new phishing scam or that the corporate had to let go of an individual who was attempting to steal corporate information. It's important to understand that keeping the entire team in the loop will help bolster the corporate security posture.

There are other tools available such as INFOSEC awareness posters, which you can get from one of the security awareness training companies. If you are creative and have the time, create post-cards with do's and don'ts of best practices for the employees that they can pin-up in their offices as reminders. The bottom line: knowledge is power so start empowering your fellow employees to gain a basic toehold in what they should and shouldn't do to help you in your mission of more uptime and less compliance headaches.

There are also some great corporate security policy tools available for free such as the powerful COBIT model at <http://www.isaca.org>, the e-tail/retail oriented PCI model from the PCI Security Standards Council found at <https://www.pcisecuritystandards.org/> and the extremely comprehensive international model called ISO27001/17799 from <http://www.iso.org/>. Any of these models will be a great starting point.

Proactive Defense – Strong Encryption

There's an old saying *loose lips sink ships*. The best practice is to look at all aspects of electronic communication and data manipulation throughout

your enterprise. That should include all instant messaging, file transfer, chat, e-mail, online meetings and webinars plus all data creation, change, storage, deletion and retrieval. For example, how are customer records stored? How are electronic versions of other confidential information protected? Backing up the data is not enough.

You should setup a VPN for external network access. Make sure the systems that access your network through the encrypted tunnel are also not the weakest links in your infrastructure so deploy HIPS on endpoints. You can encrypt everything from your hard drives to your email sessions to your file transfers. There are numerous free tools out there like <http://www.truecrypt.org> for hard drives and <http://www.openssl.org> for web, email and instant messaging, plus the grand-daddy of free encryption at <http://www.openpgp.org> PGP (Pretty Good Privacy).

You'll need policies in place for key storage and password access so if ever the keys and passwords are lost by the end-users, you'll have a way back in to decrypt the information, reset the keys or change the passwords. You might find out that some of the servers and services you are running already offer encryption if you simply check the box and turn this feature on.

Proactive Defense – Physical Access Control

Piggybacking and tailgating are a major physical security risk. Hence the need for more intelligence *Physical Access Control* (PAC), so, you'll need to make sure your PAC solution shares data over the network to you and (potentially) to your NAC solution. You should make sure your PAC solution uses two factor authentication and if your TCP/IP connections go down, the PAC system still functions mechanically with accessible local logs.

Proactive Defense – Network Access Control

Because so many exploits happen behind firewalls, you need to consider deploying *Network Access Control* (NAC). Simply put NAC determines who belongs on your network and who does not, so you should make sure your NAC solution doesn't telegraph to exploiters (ie *welcome to NAC portal... please wait, installing XYZ corp trust agent v3.1*). Also, you'll need to make sure it has a way to deal with non-Windows systems (hubs, switches, routers, blackberries, iphones, etc...) – it needs to be holistic. Try to find a non-inline or *out of band* appliance solution and avoid costly, hard to manage hacked agents.

Proactive Defense

– Host-based Intrusion Prevention System

Because so many Windows® systems are compromised – especially laptops, you need to consider *Host-based Intrusion Prevention Systems* (HIPS). Simply put HIPS blocks malicious software from functioning. The evolution of anti-virus will always be a newer, faster signature testing engine (even if they try to add HIPS) that's one step behind the latest malware attack. Look for a purely HIPS solution that blocks zero-day malware without signature updates (heuristically). It should help mitigate malware propagation, quarantine malware in real-time and not be a CPU or memory hog, making the end-user PC unusable.

Summary

Crime and Cybercrime are really the same concept, with the same end-results, only using different *vehicles* or mediums (ie physical vs logical). Web sites, e-mails, instant messaging, soft phones, and portable devices are all malware magnets. If you have an IP address, you are NOT secure and traditional countermeasures all fail! You can begin to take a more proactive approach to cyber defense by using and understanding the 4D's and the Risk Formula. You will never be 100% secure and you can NEVER block or prevent all intrusions so focus on INTRUSION DEFENSE and RISK MANAGEMENT – in other words, expect it to happen – use the 4D's and the Risk formula to contain the damage, if any. Don't forget to educate your fellow employees – the weakest link and to document your security policies. Stay vigilant and proactive so you will get one step ahead of the next threat.

Crime and Cybercrime are really the same.....Stay vigilant and proactive so you will get one step ahead of the next threat.

GARY S. MILIEFSKY, FMDHS, CISSP®

Gary S. Miliefsky is a 20+ year information security veteran and computer scientist. He is a member of ISC2.org and a CISSP®. Miliefsky is a Founding Member of the US Department of Homeland Security (DHS), serves on the advisory board of MITRE on the CVE Program (CVE.mitre.org) and is a founding board member of the National Information Security Group (NAISG.org).

Is DDOS Still a Threat?

Is DDOS, or Distributed Denial of Service, still a credible threat? Do we lay awake at night scared of when the next one might hit us?

An obvious question perhaps, they are still a threat to most online enterprises. But they're not the top of the news issues they once were. No one's taken a shot at Google, or Yahoo, or the other major sites that'll make the top of the mainstream news. Usually with a headline like *The Internet is Under Attack!!!!* If only the mainstream media and public really understood what we all know is actually going on in the undercurrents of the Internet, they'd be in a panic.

The obvious reason there hasn't really been a high profile DoS of late is that most of the larger sites are now using services like Akami, distributing their content over hundreds or thousands of nodes and geographically routing users to the closest node with the least load. This makes them arguably a near impossible DoS target. An attacker may slow down access in limited areas, but completely interrupting service is just not feasible without crippling the backend of these sites, or interrupting the DNS used to route users.

More importantly though, no one wants to be the target of the investigation behind a high profile attack. The bad guys realize (the smart ones at least) that there is so much crime, so many groups doing so many things, that as long as you stay under the radar your odds of being caught (or even investigated) are very VERY low.

We are still seeing DoS attacks, every day. It's become a tool for groups to attack and extort money from sites that can't afford the infrastructure to globally distribute their content. Online gambling sites are a particular target, and have been for some time. Many of these sites aren't legal in many countries so they can't get much in the way of law enforcement. The bad guys know this of course.

The largest threat from DoS attacks is yet to be fully realized I believe. We've seen previews of it in Georgia and Estonia. Nation states using DoS attacks as a disruption tactic in conjunction with a conventional attack. In these two very high profile attacks the effect was significant. All modern societies are very reliant on the Internet to conduct daily business, communicate orders and supply needs, manage public infrastructure, bank, and even track where vehicles are in transit.

I've written other articles in this magazine on the effects, that in a modern conflict an attacker can rely on the society of their enemy to tear itself apart

if the attacker can disrupt enough critical services. I won't rehash the details, but in summary if you make it impossible for people just to access their money electronically society as we operate now breaks down very quickly. Hoarding, looting, conflicts for basic resources. A week or two of mass hysteria and an attacking conventional force would easily be able to waltz right in and plant their own flag. Most of the society might not even notice!

Where will this go next? If I were a militia, a terrorist group, or even just a disgruntled teenager with a laptop, I'd be thinking DoS. Why risk agents or sleeper cells, finance them, sneak them into countries or secure areas to blow themselves up and perhaps 20 or 30 people? High risk, highly expensive, and minimal impact. Rather invest the money into training the same people to build and control large botnets. Build them out, make some money spamming penis enlargement pills while you've got it set up, and wait.

When the time is right, when your enemy does something particularly offensive, or you just feel like making it a bad day for a lot of people, launch. Hit the enemy in their weak spots. Disrupt banking, infrastructure controls (water, gas, oil distribution), and most importantly go after the supply chain for major food items. When a society suddenly can't get tomatoes in the grocery store they'll freak out. Seriously, it's all about the tomatoes.

Well, and a few other staples. Milk, rice, flour, etc. Most modern societies work with less than a week's supply in city to keep items fresh and minimize warehousing space in expensive retail locations. If you target the major food providers (most regions of a country have only two or three) and disrupt their ordering and dispatching capabilities things grind to a halt.

So I'm not saying I hope a terrorist group gets a clue and figures out how to truly strike at an electronic world. (hint, it's not vest bombs) I hope we as the vulnerable societies wake the freak up and do a much better job protecting our exposed underbellies.

As always please send me your thoughts,
jonkman@emergingthreats.net.

MATTHEW JONKMAN

More Secure

PHP Server Side Source Encryption

The Internet as we know it is full of mystery, intrigue and obfuscation. One of my favorite curiosities is finding ways to undo things that have been done then automating the process programmatically and retooling the concept entirely. Some may call this building a better mouse trap.

What you will learn...

- You will learn various methods to obfuscate and encrypt source code.

What you should know...

- Basic HTML/PHP/Javascript and general programming knowledge

Scenario 1: A common technique used today to obfuscate code

This scenario begins as follows: I recently had a conversation with a hacking buddy of mine (Kyle Price) in regards to hiding information but still

being able to use the information; namely in a web environment using PHP. I explained that most attempts to hide server-side PHP code were simple to *decrypt* because they needed to be in a usable state at one time or another. It is at this moment in time that it unravels and shows its true self. With such a blinding

```
eval($000000000000(base64_decode('LdK3jqRYA
EDRnxlpukWAN6XVBvgCCl/YZIS3DyigH+brd4JNb6k
W8Fs/Ppl2xiG/Z+vPNsqhvpTVsVcVl+/X61B959d43k
BTT9RKiffyqQortvNypg3NQcjCbSuDcMdEggArY3EyGM
NEGf2zRWvCyVjtQn4IMp0CSZg7amsDajsTggHr+cr3U
3Hfc55rewMA3nxggqVNEU01X2FdjHND0QRbvjZ1U5kbN
z+R3yql3arHYsnBU1HL0PVctd3M9CtKy/BKGyW0EY7T
qspIhnoL07WyihbZLmq0r09QLCdrMgUG2ncC25vkj0W
s05UP0xS7K58rjffendmiPCV2/2k0Rp/prmKBEtKH1v
PQXDEMfWAqd8yBbmpJAbJ6a+hVrWVbaslgxPURdf0j
1sZW0lqkK0cU7xwSez8iIz/AFVx5X6Mvx8iQhXERTo/
6xGL1E0ZkWZaa1q5Iqb0BeqT9cwXhUwb9CwguCtp4U
wdrHBsm417io2An10FfTPcKghN33MGCz6wywfv1on3aX0
HQjdz2LYW3gzuRykalx7hIwcdJ0Tfx4IM2jwFMhpC9i
VScm604Y955YXVmBtSqGwB1/qo7YSyqmRho91Wu9D4
guJsEod1w3KensNEUBdrsRC3yVLwIppmxh59ocqp2rt
wo4wzz1BNfJ94g0s+jsNxodiwRteEKzc0JU7sPkHFTX
KfcHd4Wjs7mmmmguU3uioAC13MCNuNY0GXHEMa19wL7
5o715F8b1aq1EfL6Gfn65S6eTgxrbw4GWQPIkUP8s38
W3hPvxf01kBxVXoeyxkUbXHdpoC36kZ5kHq6XJNJaQI
SFhIGZjzVqlvQL7u1RIRUVwt0X8a0I1LJHf1M6M2NTh
a8mhtZHmz+EBsyWl3eRsomCv38yN/l5aSwuUKH0szb
NnRKJ2p0zfUM79KwhgrYpBKl+8c03hZkqtvIP4C8zkQ
G+rZitrKeA6U+13D9gXG/Hc14vGSBmAwFn6ehzV1BSF
EbjW80dSaW+/f39/f3P/8B'')));
```

Figure 1. Obfuscated code

```
eval($000000000000(base64_decode('LdK3jqRYA
EDRnxlpukWAN6XVBvgCCl/YZIS3DyigH+brd4JNb6k
W8Fs/Ppl2xiG/Z+vPNsqhvpTVsVcVl+/X61B959d43k
BTT9RKiffyqQortvNypg3NQcjCbSuDcMdEggArY3EyGM
NEGf2zRWvCyVjtQn4IMp0CSZg7amsDajsTggHr+cr3U
3Hfc55rewMA3nxggqVNEU01X2FdjHND0QRbvjZ1U5kbN
z+R3yql3arHYsnBU1HL0PVctd3M9CtKy/BKGyW0EY7T
qspIhnoL07WyihbZLmq0r09QLCdrMgUG2ncC25vkj0W
s05UP0xS7K58rjffendmiPCV2/2k0Rp/prmKBEtKH1v
PQXDEMfWAqd8yBbmpJAbJ6a+hVrWVbaslgxPURdf0j
1sZW0lqkK0cU7xwSez8iIz/AFVx5X6Mvx8iQhXERTo/
6xGL1E0ZkWZaa1q5Iqb0BeqT9cwXhUwb9CwguCtp4U
wdrHBsm417io2An10FfTPcKghN33MGCz6wywfv1on3aX0
HQjdz2LYW3gzuRykalx7hIwcdJ0Tfx4IM2jwFMhpC9i
VScm604Y955YXVmBtSqGwB1/qo7YSyqmRho91Wu9D4
guJsEod1w3KensNEUBdrsRC3yVLwIppmxh59ocqp2rt
wo4wzz1BNfJ94g0s+jsNxodiwRteEKzc0JU7sPkHFTX
KfcHd4Wjs7mmmmguU3uioAC13MCNuNY0GXHEMa19wL7
5o715F8b1aq1EfL6Gfn65S6eTgxrbw4GWQPIkUP8s38
W3hPvxf01kBxVXoeyxkUbXHdpoC36kZ5kHq6XJNJaQI
SFhIGZjzVqlvQL7u1RIRUVwt0X8a0I1LJHf1M6M2NTh
a8mhtZHmz+EBsyWl3eRsomCv38yN/l5aSwuUKH0szb
NnRKJ2p0zfUM79KwhgrYpBKl+8c03hZkqtvIP4C8zkQ
G+rZitrKeA6U+13D9gXG/Hc14vGSBmAwFn6ehzV1BSF
EbjW80dSaW+/f39/f3P/8B'')));
```

Figure 2. Eval function

```

eval($000000000000(base64_decode('LZK5jqtIA
AB/ZqWZEYExGBr0tAEY0xgw2EBz0FKBzX2Y/+r6N8Em
IVYF1Sxh/f2PaZIk+T++o3BM2Mt/0IK/0Pn+0mNHLbt
xFQTp1bItgmm1gqvjdJ42geny/WK4louJ55w3cTs7nG6z/
MqnSiFxJiGcp0EBwe4pwjj8epzgyc9c3NN5dvVY9lvLW
rJEFa+RF+jhoM/cItDIPJGnnjBY/goUK3hiB7hMxuDP
rbvEwCSCiWboeRASQufP5R6VQsf7MC14jykvnD2hVNH
yJLvvVSmNF61Amagpd2+0A4njr0h7?DsaXDDBC9Wi9j
XIhDih+cd0Z1iP30T92DAnmyUybjf/LPKa+p6Q+8JW
jYN7EVvb3dddIIhr+dJQPemfx7RML4+nsvRNmBQr3N1
6fUrvnGaI8HCKB3xievSMvsek1PQkNvHaeMmz9hHOiY
P5gEij700hUQo3py3ZKxtDmM13MQM1w9+K9k1ZBV5y2
rGn6UizX1RH0ff4mXm4rJKm/PD2hiB1ceo2k57Ldr5t
k+1QYUV0g88qj40IMCB5MiCIWRG1cUno60kZ2eb4fPy
bKdXa0R2x1rnAK22zq9cNQWoPNC+3rvvMwPXZ0jU+lsF
QPdtPAOBwi08HZUI0gbQCM4myHnm06PN140XvxFRLm
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));

```

Figure 3. Decoded second time

vulnerability hiding something from someone that knows what to look for is just a game it will eventually lose. Kyle wasn't exactly sure how it all worked so I asked him to send me an obfuscated piece of code and I would show him how to decode it. He searched the 'net and eventually sent me an email that looked something like this (Figure 1).

It only took me a quick second to spot the infamous PHP eval function call (Figure 2) eval (\$000000000000(base64_decode('). For those that don't have experience with this allow me to explain by breaking this down.

eval: eval evaluates a string as PHP code (\$000000000000: this is a bogus function call but should be `gzinflate()` to inflate deflated string (base64_decode('): this decodes data encoded with MIME base64.

Put all three functions together and you are running a routine to *unwrap* a string that's been deflated and base64 encoded. To undo this you only need to reverse the process and this is exactly what the code in Figure 1 is doing already for you. The only tricky part here is that the programmer is trying to

```

echo "Congratulations: ";
echo str_rot13('gur frperg zrffntr');
echo " found! ";

```

Figure 4. Decoded tenth time

```

'::::;
-->[wrop : 7]
eval($000000000000(base64_decode('LcTJcoIwADQn+1MdTyIQlCnp5010EHA1ubSYQlYdgIE80t7
6Ts8wqJg8+asHmf9t4mjg9yjCT0gSN1Wbdzvx7eRFM4Kt/K+bvhY9U2js57hncpK3M289x12UrQ1Ytqx
wNC0HAIS+0F9NdygJc9d/fMzSp3ed1B73z1Shpk/V29pSm7Tqs6LU-092j8V7gydhlNU1d1Id06g
Nu0l0v4ew9JrbneAvNw0t0Ev5N9MsNdb6vEM0L+5Xx3z01cS1+o0Fn/r/zC50IPexP6e102Sz0tR0nY
uS9E57h1uwlw1u/Xe1N9DcJ2c2zX1M1N9PTX4kFPYveRCYQAgeluhupr06urw2lkH73ny1kH3vPip
Y1XXNaUlmBu/uko/38Y0uh0q73q8pVbzGx65sPkmlTeJDHtUxsl+sG0z8r7dbj/+AA=')));
-->[wrop : 8]
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
ltfBaer1k2GjxWBS51msN0+1Q1ZKXK1TJ3hryjH6u1mHr5/p40VJrc7Z1Ad/oQfeiUqmzpgul.jgmQTDezt
D6193rf5x0gzb1q8VtdbsrsmwCCPlyIp5yrRJY3jvQu6z0rCwXq1+vWd'')));
-->[wrop : 9]
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
ltfBaer1k2GjxWBS51msN0+1Q1ZKXK1TJ3hryjH6u1mHr5/p40VJrc7Z1Ad/oQfeiUqmzpgul.jgmQTDezt
D6193rf5x0gzb1q8VtdbsrsmwCCPlyIp5yrRJY3jvQu6z0rCwXq1+vWd'')));
-->[wrop : 10]
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
ltfBaer1k2GjxWBS51msN0+1Q1ZKXK1TJ3hryjH6u1mHr5/p40VJrc7Z1Ad/oQfeiUqmzpgul.jgmQTDezt
D6193rf5x0gzb1q8VtdbsrsmwCCPlyIp5yrRJY3jvQu6z0rCwXq1+vWd'')));
-->[wrop : 11]
eval($000000000000(base64_decode('40pNzshXUEr0zsv5iwpzUkscyzPK7zULLmAssUlxTFF+WX
GPrqkXeFmfpWmpqJqBr/Bw/peVFKlxQjUpkGt1l+oIK1K0A0=')));
-->[wrapped 11 times]>>>
----- snip -----
echo "Congratulations: ";
echo str_rot13('gur frperg zrffntr');
echo " found! ";
----- snip -----
> |

```

Figure 5. Full decloaking results

deter scanners and attackers from figuring out they are using `gzinflate`. They have done this by using a combination of zeros and upper case letter Os as a variable name replacement. By simple replacing `$000000000000` with `gzinflate` you've broken the first step of the deobfuscation.

Doing the replacement and running eval then decodes to another mystery. The code we decoded

```

7sGWhkBoRmFRAYFzWwZaSfxfZ+IYrdWFNjexyEX1J9KJwUeZtGR637svLnEILBTYrh1utAUImqPzbrsQo
wv8375lql.8j1qwdqE8KT0d1njbPd1jT0nhdHtYm0t0NmcmhgMfse15M7bw54m1d+Hwuhn0ckxT5r8t
5rCodHgCo6Hfgx15xD1hxFBLssX9F54V6n9rwmvNvRhjy1b5hxyt1N2ak1U90k30G7zUfbRG
oG5sJvd/Np2Dp50dP4PrntJhNb5p61YuvGLIw+0KMuveCap4t7VX6f1n0fNs5p6q77dar0c12067B7
k7x7HgCo6Hfgx15xD1hxFBLssX9F54V6n9rwmvNvRhjy1b5hxyt1N2ak1U90k30G7zUfbRG
JecsGd0y1G08F9qfZgp6nk1Rk180u0zTAZpVb9t02085HeX0TcEwvcZyB8EW7b0w8tBjy8a5
f7x7HgCo6Hfgx15xD1hxFBLssX9F54V6n9rwmvNvRhjy1b5hxyt1N2ak1U90k30G7zUfbRG
R6193rf5x0gzb1q8VtdbsrsmwCCPlyIp5yrRJY3jvQu6z0rCwXq1+vWd''));
-->[wrop : 10]
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+9mbNT
tsQldnSys4hGfKVAJxWSJyQpRJ2krimw6+rUv83kXut
qpARCzpRSIV7CGjJvEI8lIsrvrtDk5N592x3bVryDC5IU
2Vci04aS0gGaYjp1RtWeace+8zy2uFC4ptWSgCPX6b4
cSGYKijWsSkVKvapLRy3EM/m6z/6z3qLI7/V4Di/56T
DKJ4J2fmeofovzdKnkem03FW2tqMKTnv6ezKbqm3yxY
EmJspDOFrxxk4GhrVruvXjswglwLEB2rnKF1onkhbQBZ
/0/Xz8/Pn78=')));
-->[wrapped 11 times]>>>
----- snip -----
eval($000000000000(base64_decode('LcTJcoIwADQy3RGHRZERJHpkH2+EvkZqbLpAEKRCCKSTH6
bvl0j442W34EAQdg2wExDSn/hBad1QfFejAx3nvmox01gZdVdf3aFk5Gm6NKYQy5KCuZtL0N+
```

```
<?php
include "./itcloaker.php";

// decloak("v", "encoded.txt", "\$000000000000"); // verbose
// decloak("n", "encoded.txt", "\$000000000000"); // non-verbose
// encloak("v", "decoded.txt", "gzinflate"); // verbose
// encloak("n", "decoded.txt", "gzinflate"); // non-verbose

// samples

//encloak("n", "use_source.txt", "gzinflate");
//decloak("n", "use_crypt.txt", "\$000000000000");

?>
```

Figure 7. ITCloaker function calls

```
<?php

// test1.php a simple example
include "./xorlib.php";

$secretkey = "random data";

echo XOREncrypt("echo \"this begins this test\n";
\$myvar=3*2; echo \$myvar; echo "\n"; echo \
\$myvar+1 . "\n"; echo "this ends this test\n";
", $secretkey);

?>
```

Figure 8. Encrypting using XORlib.php

almost looks identical to what we just decoded – but shorter in length (Figure 3) and we are back to the `$zero+0` `gzinflate` variable label. In fact this process is repeated for a total of 10 times before we finally get to the true encapsulated source (Figure 4)...congratulations indeed.

After running through the process manually I quickly built up a script that would programmatically `decloak` obfuscated code (Figure 5) created by the PHP obfuscator Kyle used as well as mimic the obfuscator (Figure 6) itself by creating the same type of result with arbitrary code and aptly named it `itcloaker.php` (as it cloaks and decloaks) ([source code: itcloaker.php](#)) here I've created a few functions that you can include and call from your own PHP code (Figure 7).

Now this whole episode happened in a matter of minutes before I sent the resultant original source

```
<?php

// test1.php a simple example
include "./xorlib.php";

$secretkey = "random data";

eval(XORDecrypt
("FwIGC09PVAwIB0EQBAkNAR4AEAkdlIIVCxcbZwJfQVAMCxcPF
lJeClZaVAQRQCQFESwBZEgAGWlIEDQwATQJuQ09BFwIGC09JTR0X
FRNZUE5KT08qRlpUBBEJAURNGUgNElQEHAUdRBsFSRdBAAQBFWRGVA==",
$secretkey));

?>
```

Figure 10. Setting up for decrypt function

code back to Kyle. He wasn't as happy as I thought he was going to be. I felt like I just told him Santa Claus wasn't real (and proved it). We then conversed further and drew pictures on the white board about a more secure form of obfuscation and I brought up the notion of using something more complex and using something more like a one-time-pad using XOR with a keyed passphrase; then to using remote passphrase keys via SSL to a remote server with more control, port knocking, random key generation... I then went on my way to create such a creature (*ultimately named itarmor*).

Scenario 2: A more secure technique using XOR encryption

This next scenario involves developing a more secure technique I've named `itarmor` as its purpose it to armor the code from simple attacks as described in Scenario 1.

I found a nice pre-fabricated free PHP xor snippet authored by Jonas John created in 2007 and licensed as public domain; the main function is `XOREncryption()` with two complimentary helper functions `XOREncrypt()` and `XORDecrypt()`. I originally planned to roll my own but this function fit perfectly for my needs in a very short amount of time. Saving time by not reinventing the wheel is good! I saved the source and labeled it `xorlib.php` for all intents and purposes.

```
> php test1.php
FwIGC09PVAwIB0EQBAkNAR4AEAkdlIIVCxcbZwJfQVAMCxcPF
lJeClZaVAQRQCQFESwBZEgAGWlIEDQwATQJuQ09BFwIGC09JTR0X
FRNZUE5KT08qRlpUBBEJAURNGUgNElQEHAUdRBsFSRdBAAQBFWRGVA==>
```

Figure 9. XORed and base64 encoded

```
> php test1.php
FwIGC09PVAwIB0EQBAkNAR4AEAkdlIIVCxcbZwJfQVAMCxcPF
lJeClZaVAQRQCQFESwBZEgAGWlIEDQwATQJuQ09BFwIGC09JTR0X
FRNZUE5KT08qRlpUBBEJAURNGUgNElQEHAUdRBsFSRdBAAQBFWRGVA==>
> php test2.php
this begins this test
6
7
this ends this test
>
```

Figure 11. Decrypted using XORlib.php

```
<?php

// test1.php a simple example
include "./xorlib.php";

include "./secret.php";

eval(XORDecrypt(
"FwIGC09PVAwIB0EQBAkNAR
4AEAkdlEIVCxcbZwJfQVAM
CxcPF1JeCLZaVAQRCQFESw
BZEgAGWLIEDQwATQJuQ09B
FwIGC09JTR0XFRNZUE5KT0
8qR1pUBBEJAURNGUgNE1QE
HAUdRBsFSRdBAAQBFWRGVA
=="
,$secretkey));

?>
```

Figure 12. Abstracting the secret key

I created a very simple PHP test that prints to the screen and does math with variables. In the first test I call `xorlib.php` from another PHP file named `test1.php` (Figure 8) using `include "./xorlib.php";` Using `XOREncrypt()` I get the resulting base64 encoding (Figure 9) To run this in PHP I now use `eval()` and `XORDecrypt()` to decode using the secret key *random data*. (Figure 10) and when we execute it using PHP `test2.php` we get the expected calculated results! (Figure 11) this is a step in the right way but aside from `xorlib.php` being local the `$secretkey` value is also right there in the code plain as day.

For the third test (Figure 12) I removed `$secretkey` to another file like `xorlib.php` using `include()` and reformatted the code for a more uniform look and received the expected successful results.

Scenario 3: A more secure technique using XOR encryption via remote https

This scenario evolves Scenario 2 by removing the `$secretkey` from the local environment to a remote environment using a few more barriers for someone trying to reverse or backtrace what the secret key is.

The concept behind this remote secretkey is that this secret key could be changing every few minutes via a cron job or perhaps when a client doesn't pay the monthly invoice, or used as some type of license,

```
<?php

// force ssl
if ( $_SERVER['SERVER_PORT'] == 80 ) {
header('Location:https://'.$_SERVER['HTTP_HOST'].dirname($_SERVER['PHP_SELF']).basename($_SERVER['PHP_SELF']));
die();
}

// get incoming IP
function getRealIpAddr(){
if (!empty($_SERVER['HTTP_CLIENT_IP'])) {
$ip=$_SERVER['HTTP_CLIENT_IP'];
}
elseif (!empty($_SERVER['HTTP_X_FORWARDED_FOR'])) {
$ip=$_SERVER['HTTP_X_FORWARDED_FOR'];
}
else {
$ip=$_SERVER['REMOTE_ADDR'];
}
return $ip;
}

// testing for a user agent - in this case if there is one then fail.
$ userAgent = $_SERVER['HTTP_USER_AGENT'];
if (empty($userAgent)) {
$accessKey = getRealIpAddr();
}

// if the incoming IP is not the IP expected/allowed then fail.
$clientID = "xxx.xxx.xxx.xxx";
if (strcmp($accessKey, $clientID) == 0 ) {

    // if the IP is expected then give secret key.
    echo "<?php\n";
    echo '$secretKey = \"random data\";\n'; // access granted here is
the key
    echo "?>';
} else {
    echo "access denied!";
}
}
else {
echo "access denied!";
}

?>
```

Figure 13. PHP defensive methods to hide key

deployment tracking, to stop the average user from copying your code and frankly the possibilities are limitless.

I wrote a new version of `secret.php` so that it just did not contain the variable and value but now had multiple

```
<?php

// test1.php a simple example
include "./xorlib.php";

include "https://israeltorres.org/secret.php";

eval(XORDecrypt(
"FwIGC09PVAwIB0EQBAkNAR
4AEAkdlEIVCxcbZwJfQVAM
CxcPF1JeCLZaVAQRCQFESw
BZEgAGWLIEDQwATQJuQ09B
FwIGC09JTR0XFRNZUE5KT0
8qR1pUBBEJAURNGUgNE1QE
HAUdRBsFSRdBAAQBFWRGVA
=="
,$secretKey));
```

Figure 14. Remote placement of secret key



Figure 15. Enable php.ini for remote access

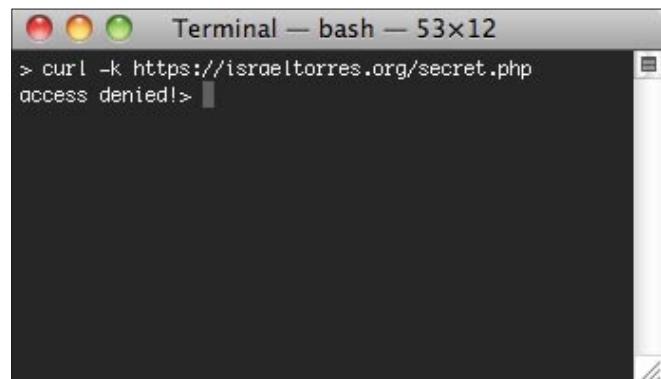


Figure 18. Attempt to get key using plain curl

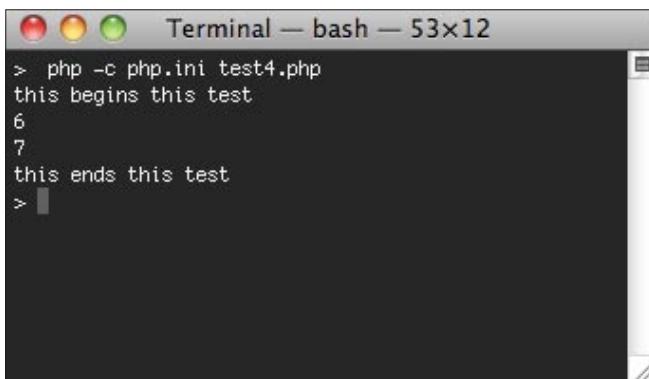


Figure 16. Successful remote decoding result

barriers to thwart a common *script kiddie* from running a simple attack. These barriers are as follows in this order (Figure 13)*:

- Barrier #1: Forcing SSL makes sniffing the secretkey via wireshark more difficult.
- Barrier #2: Checking the requestor's IP address to make sure it's the correct server making the request.
- Barrier #3: Checking to see if the requestor is using a specific type of User Agent.

Once tested I replaced the local secret.php file with the remote secret.php in test4.php (Figure 14) with: `include "https://israeltorres.org/secret.php";`

On my mac I needed to create local `php.ini` (Figure 15) file with one line `allow_url_include=1` to allow remote include files and use the following syntax in terminal (Figure 16): `php -c php.ini test4.php` and received the expected decoded and calculated results.

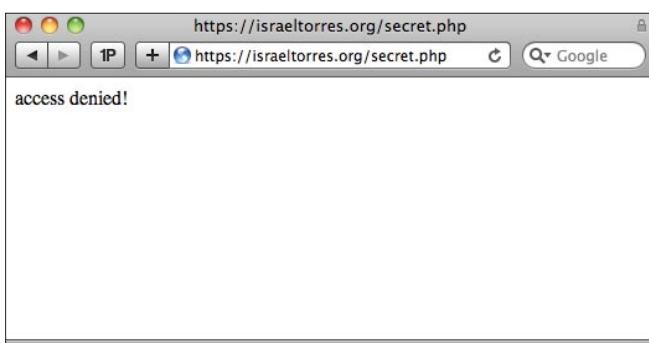


Figure 17. Attempt to get key using web browser



Figure 19. Bypassing User Agent using curl

I ran a browser test using Safari (Figure 17) and got the expected result as the User-Agent for Safari is: `Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_6_3; en-us) AppleWebKit/531.22.7 (KHTML, like Gecko) Version/4.0.5 Safari/531.22.7` and in the secret.php check I explicitly stated NO USER AGENT was permitted (you can change this to special strings; that's up to you to play with – as you'll want to change it after seeing the next example).

I further tested it using curl (Figure 18) and because the default curl request has a User Agent string of: `curl/7.19.7 (universal-apple-darwin10.0) libcurl/7.19.7 OpenSSL/0.9.8l zlib/1.2.3` this command also gets access denied. I was able to easily thwart this by using the `-A ""` null parameter to get the secret key (Figure 19).



Figure 20. Advanced „randomness generator”

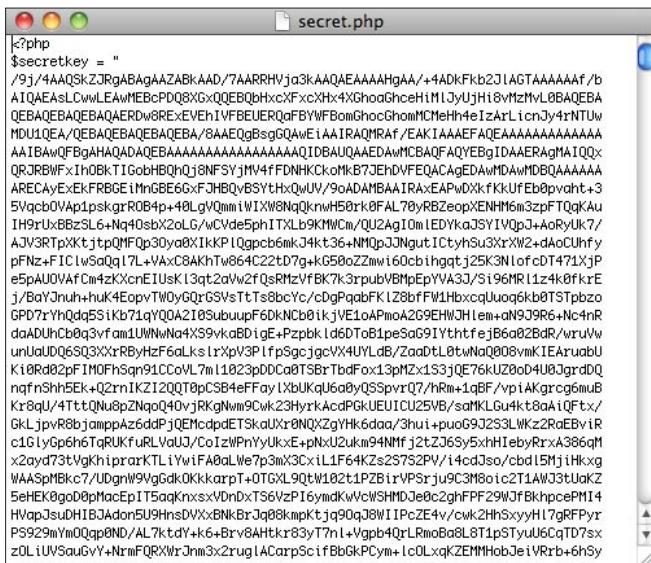


Figure 21. Larger and more random secret key

Again you can modify your User Agent and match it with the remote `secret.php` to further confuse the attackers.

Scenario 4:

A more secure technique using XOR encryption with more secret key randomness

At this point my example ending with `test4.php` makes it pretty obscure for an attacker to successfully reverse without the protected secret key over PHP, SSL, IP, and User Agent strings.

I went further on using base64 file encoder/decoder to encode a *random* image and then use that as the

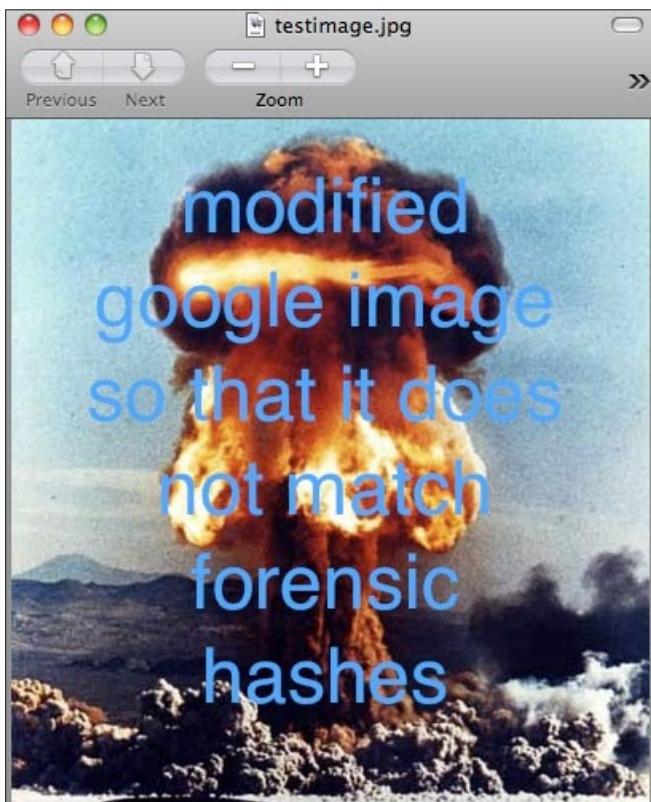


Figure 22. Sample modified Google image

Notes

all source code created and tested on my Apple Macbook Pro running

Mac OS X 10.6.3

PHP 5.3.1 (cli) (built: Feb 11 2010 02:32:22)

GNU bash, version 3.2.48(1)-release (x86_64-apple-darwin10.0)

Special Thanks to Kyle Price

Web Links and References

- <http://php.net/manual/en/function.eval.php>
 - <http://php.net/manual/en/function.gzinflate.php>
 - <http://php.net/manual/en/function.base64-decode.php>
 - <http://www.jonasjohn.de/snippets/php/xor-encryption.htm>
 - http://en.wikipedia.org/wiki/XOR_cipher
 - <http://php.net/manual/en/function.base64-encode.php>
- *various snippets the put together (ssl)secret.php
- <http://www.commandlinefu.com/>
 - <http://snipplr.com/>
 - <http://www.google.com/>

`$secretkey` (Figure 20) which significantly increases the random characters in the secret key as well as gives it an extremely large and generous key space for a great one-time pad using XOR (Figure 21).

Note: For better security be sure not to use images from google without some further modification as someone that is really skilled may be able to find the right image you used to create the secret key (highly unlikely, but not impossible) (Figure 22).

Conclusion

In my PHP code examples and scenarios above I've taken quite a few steps to further armor remote code protection from common basic obfuscation techniques that use `eval()` and can easily be decoded locally using simple scripts and also provided methods to do so easily (i.e. itcloaker). Some may require more modification but the basic process is there. Also note that this armoring technique didn't require any special server modifications or additional software modules installed that third party obfuscators/ encryptors may use. In the security universe nothing is entirely fool-proof but it certainly changes the game in the world of building a better mouse trap.

ISRAEL TORRES

Hacker at large with interests in the hacking realm.

hakin9@israeltorres.org

[hakin9 crypto challenge](http://hakin9.cryptochallenge)

<http://hakin9.israeltorres.org/>

Don't let the zombies take you down

Ian Kilpatrick, chairman of Wick Hill Group, specialists in secure infrastructure solutions

Over the last year, the incidence of botnet (or zombie) attacks has been growing rapidly. Some service providers around the world have already begun to take action against botnets [1] and there is increased interest from other service providers, and from companies, in dealing with this serious security threat.

Botnets are most closely associated with computers being taken over and used to send out spam emails. However the threat is much wider than that. At the other end of the scale, there are criminals renting out botnets to harvest personal banking and security information, mount serious commercial attacks, steal money or commit fraud.

Both individuals and businesses are being targeted. Web sites are being infected (so called *drive-by infections*) so that they deliver malicious code to the sites' visitors. Botnets are also being used to mount DDoS attacks on businesses, which can have serious consequences. Twitter was recently the victim of a DDoS attack and temporarily closed down [2].

These are not trivial threats. There is a significant amount of money to be made in harvesting banking information, launching blackmailing DDoS attacks, or in just renting out the Zombie army for someone else to use. So there is continual recruitment and development of these armies, as well as investment in the *command and control* infrastructures by *bot herders*, the individuals or organisations which control a group of botnets.

Botnets can be hugely sophisticated and very resilient, with their own forms of disaster recovery built in, so they can continue to function even when attacked.

Recent research by Trend

Micro [3], which gives some idea of the scale of the problem and the difficulties of disinfection, found that the industry underestimated the length of time PCs were infected with botnets. The company found that, in 100 million compromised machines, the average infection was 300 days, not the estimated six weeks.

The scale of individual boherds can also be very high. Recently a botnet of over 2 million pcs was discovered in the UK and US [4]. And a Dutch botnet had over 1.4 million in the herd [5].

How are you infected?

Botnets are multiple software robots (bots) that can run autonomously. They can be malign or benign, but we are just looking at the malign here. Bots are typically delivered by e-mail or from a web site.

Users are now well aware of email-based threats and many have protected themselves in this area, so web-based delivery of bots is increasing. This can be through going onto what appears to be an innocent web site and picking up a malicious download. This kind of threat can also evade traditional list-based web content security systems,

which rely on prepared lists of good and bad sites. Typically, infected *good* sites will not be identified on these lists.

Some *phishing* emails will take you to web sites where you may inadvertently download a bot. Your users could bring them in on laptops or USBs potentially infecting your whole network. You can even catch bots by taking part in MMORPGs (massive multiplayer online role playing games).

Trojans and worms are common methods of joining boherds. Conficker, which recently cost Manchester City Council over £1.5 million, is a sophisticated, self-replicating worm managed by a central command and control structure.

You are also a target if you fail to use the right anti-virus and fail to rapidly update vulnerability patches.

Dangers

Once you're part of a zombie army, you may not notice anything and be totally unaware that your machine is infected. But the bot is now secretly installed on your computer and can use it to send out large volumes of spam in the background, or harvest keystroke information, passwords, online banking details, log-on details, etc.

In the case of botnets being used to launch DDoS attacks, forensic tracking has led authorities to investigate innocent botnet members. It's also possible that you could find your company blacklisted as an organisation sending out spam.

Bots can penetrate the corporate network so they can potentially monitor everything going on, compromising your security by potentially passing on information on passwords or online banking.

And, once installed, significant spam activity, caused by the bot, might slow down your network, leaving your system sluggish, but leaving you unaware of the cause.

Protecting against bots

There are many things you can do to protect your organisation from becoming part of a boherd. Applying security patches to key applications, as soon as is practicable, is a major help. These vulnerabilities are high risk until patched.

In a cyber security report by Lumension, released in 2009, security and forensic analyst Paul Henry said: *Until the underlying patch management issue is dealt*

with, botnets will continue their explosive growth on the public internet [6].

The best way to prevent botnets, though, is by having proper security solutions in place to begin with.

For companies, the place to start is at the gateway. However gateway security will not be enough when mobile users and visitors are connecting inside the gateway. Proper access control and strong two factor authentication will help here.

If staff are using USBs, laptops, iPods, etc. inside the gateway, there is the risk that they are bypassing gateway security controls and infecting network connected devices – so your security policy should cover the safe use of mobile equipment.

Other high risk areas inside the network include infections picked up from staff visiting malicious web sites. A classic security method here is to deploy multi-layer protection. Alongside your gateway protection, you should also be installing protection on your PCs. This should ideally be from a different manufacturer than that used for your gateway protection.

There are many endpoint (PC/Laptop) solutions available that will provide protection. Solutions from companies such as Check Point and Kaspersky Lab will scan all incoming and outgoing data traffic on PCs for malicious content and give them protection against being hijacked for botnet activity.

Endpoint security solutions, such as those mentioned above, will protect against malicious code downloading from infected web sites, as well as Trojans from e-mail or mobile devices, including USBs.

At the gateway, companies such as M86 and Finjan provide web gateway protection that can identify and defend against malicious code loaded on rogue and infected web sites.



Ends

- Australian Internet Industry Association (government advisory) drafts code of conduct for fighting botnets – <http://www.itnews.com.au/News/155673,isps-asked-to-cut-off-malware-infected-pcs.aspx> [1]
- http://www.it-director.com/technology/news_release.php?rel=12725 [2]
- <http://www.infosecurity-magazine.com/view/4016/compromised-machines-stay-compromised-trend-micro/> [3]
- <http://www.itnews.com.au/News/143123,massive-uk-and-us-botnet-uncovered.aspx> [4]
- http://www.infopackets.com/news/technology/word_of_the_day/2009/20090519_botnet.htm [5]
- http://www.lumensionsecurity.com/nwr_pressReleasesDetails.jsp;jsessionid=12892CA71D631B12F401988967085B11?i-d=152123&metadataId=152123 [6]
- Dutch ISPs sign agreement for fighting botnets – <http://www.computerweekly.com/blogs/when-it-meets-politics/2009/09/learning-from-the-dutch---isp.html> [7]
- Messaging Anti-Abuse Working Group publishes best practices for fighting botnets – <http://finance.yahoo.com/news/MAAWG-Tackles-Bots-with-New-prnews-1561387349.html?x=0&.v=1> [8]
- IETF draft standard for fighting botnets – <http://www.scmagazineus.com/Standard-offers-best-practices-for-ISPs-to-fight-botnets/article/149162/> [9]
- <http://blogs.zdnet.com/security/?p=4404> [10]

If you want to protect your own web site from being infected and delivering malicious code to your customers, companies such as Check Point and Barracuda Networks have web application firewall capabilities to protect against this increasingly prevalent threat.

Other solutions, such as Barracuda Networks' anti-spam, virus and spyware firewall, can help protect traffic going in and out of your network. This would include attempts to send spam or return spyware data.

You can also detect bots by using traffic management solutions, such as those from Allot. They are able to identify traffic patterns, even masked traffic patterns, which could be bot activity.

Network intelligence systems, such as those from Loglogic or ArcSight, can also help. They can bring together and let you analyse, all log information on your network, down to a granular/PC level, highlighting any unusual behaviour.

Web sites such as Spamhaus.org explain how you can identify and remove botnets if you're worried you may have one. At a corporate level, some of the above solutions will also disinfect your existing estate. At a personal level, companies such as Kaspersky Lab and Webroot provide low cost protection.

Need for action

There are many ways for the unsuspecting or unprotected to be infected and some of this should be dealt with by service providers. Some ISPs are making strong efforts to manage the problem. For example, earlier this year Dutch ISPs banded together to deal with the threat [7].

However, they are the exception. Many service providers don't respond unless they find themselves blacklisted for sending out spam or they become victims of a DDoS attack themselves.

This is not a customer-friendly approach and is short sighted because there are solutions available for service providers, such as ServiceProtector from Allot, which can effectively neutralise botnets and stop spam being sent out from subscribers' computers, as well as preventing spam being received by them.

It will also, importantly, protect service providers and enterprises from DDoS attacks, leaving them little excuse to carry on doing nothing about this serious security threat.

A number of other initiatives are taking place, though, in the fight against botnets. The Messaging Anti-Abuse Working Group recently published best practises for fighting botnets [8] The IETF (Internet Engineering Task Force) has also published some best practises [9]. And many large organisations are becoming increasingly vocal in their requirements for botnets to be dealt with – witness Google's recent comments [10].

With pressure increasing, it is likely that there will be some significant moves against the botnet threat over the next few years.

IAN KILPATRICK

Ian Kilpatrick is chairman of value added distributor Wick Hill Group plc, specialists in secure infrastructure solutions. Kilpatrick has been involved with the Group for more than 30 years. Wick Hill is an international organisation supplying SMEs and most of the Times Top 1000 companies through a value-added network of accredited resellers.

Kilpatrick has an in-depth experience of computing with a strong vision of the future in IT. He looks at computing from a business point-of-view and his approach reflects his philosophy that business benefits and ease-of-use are the key factors in IT, rather than just technology. He has authored numerous articles and publications, as well as being a regular speaker at conferences, exhibitions and seminars.

Stop them before they stop you...



...with award-winning security solutions from Black Box.

**Protect your network,
your data, your infrastructure,
and your personnel.**



Network Access Control

Protect your network from unwanted access.



Internet Threat Protection

Protect against malware, Internet threats, and non-work-related use.



Physical Security

Physically secure your most sensitive assets with military-grade biometrics.

Call 1-800-355-7996 or visit www.blackbox.com/go/security



eLearnSecurity
Forging security professionals

Want to become the world's no.1 hacker?

Then you are on the wrong page.

We can only teach you how to stand out as a
professional penetration tester
and leave script kiddies behind



Online Penetration Testing Course

"What CEH should have been"

EthicalHacker.net

3 domains - 18 modules

Web Application Attacks

Network Attacks

Systems Attacks

From basic to advanced topics

Life-time access to course material

Get certified with our practical exam

All the most advanced and up to date attacks

Learn what your clients want from top pentesters

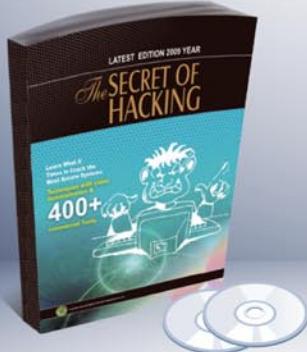
Thousands? No. Only \$569 with coupon: HAK9-ELS-5

www.elearnsecurity.com

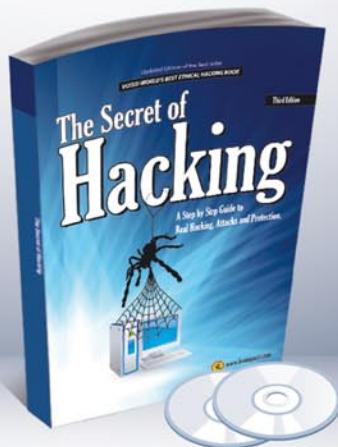


Want's to be the Best Ethical Hacker & Security Expert

GET "The Secret of Hacking" with 2 DVD (40,000 full ver tools)+ Videos.



2nd Edition
List Price: **USD 98**
Offer Price: **53 USD ONLY**



3rd Edition
List Price: **USD 99**
Offer Price: **54 USD ONLY**

Combo Offer (with 4 DVDs)
 3rd Edition + 2nd Edition + 1st edition in PDF
 List Price: **USD 399**
 Offer Price: **Rs. 99 USD ONLY**
 = Order Combo KIT (Save 53%)



SPECIAL COMPANY HIGHLIGHTS ...

- » We are the world's first company that released Exploit on Ms Office 2007
- » We also released first multi hop Exploit for PDF 8/9 (hide exe into PDF file)
- » Leo Impact Security, inc have more then 5 patent pending research



UNCOMMON FEATURE'S:

- 21 WAYS TO HACK & PROTECT EMAIL ID & PASSWORDS
- LEARN BASIC TO ADVANCED HACKING AND SECURITY
- LEARN REMOTE HACKING(WITHOUT ANY ATTACHMENTS)
- LEARN NETBANKING & CREDIT CARDS HACKING & SECURITY
- EASILY PASS CEH, CHFI, CISSP, CISA CERTIFICATIONS (Free Dumps)
- LEARN VIRUS RESEARCH & DEVELOPMENT.
- 30 DAYS MONEY BACK GUARANTEE IF YOU ARE NOT SATISFIED
- No shipping and Hidden cost + Works on all Operating system (Widnows, Linux, Mac OS)



Incredible Offer :: Order Now

www.theseceotf hacking.com

Now available on Amazon.com



:: Get Surprise Free Gift ::

www.theseceotf hacking.com



**LEO IMPACT
SECURITY**

Leo Impact Security, INC
616, Corporate Way, Suite 2
#4000, Valley Cottage, NY 10989
Phone: +1 818 252 9090 (USA)