Shopping Cart Weak Links

By Brian Krebs

Authentication Comes Of Age

By Robert Vamosi

Recommendations for a Safer Facebook

By Eugene Kaspersky

SECUREVIEW

2nd quarter 2011

Inside Stuxnet's Stolen Certificates

An up-close look at some mistakes made when the Stuxnet creators stole and signed the malware's digital certificates.

SECURELIST

your link to our lab

Analysis

May 03 2010

Monthly Malware Statistics: April 2010

Malicious programs detected on users' computers

Apr 29 2010

Crimeware: A new round of confrontation begins...

This article provides an analysis of recent developments regarding attacks launched by

Apr 20 2010

Spam evolution: March 2010

The amount of spam detected in mail traffic averaged 82.9% in March 2010. A low of 78% was

Natalia Zablotskaya »

Virus Watch | "I want your clothes, shoes, and motorbike"

04 May, 13:29 GMT

VitalyK »

Research | Gumblar: Farewell Japan

03 May, 18:11 GMT

Georg 'oxff' Wicherski »

Incidents | Heloag has rather no friends, just a master

→ Detailed and prompt analyses of Internet threats
→ Expert opinions from the Company's professionals
→ The world's largest security encyclopedia
→ Real-time malware statistics

KASPERSKY lab

www.securelist.com

INSIDE THIS ISSUE

A Word From The Editor

This issue of the SECUREVIEW Magazine comes at a crucial period in the security industry. Not a day goes by that we're not hearing new reports of a data breach or targeted attack against a high-profile company. Zero-day vulnerabilities and the software patching treadmill dominates the headlines. We are inundated with new malware attacks on social media platforms and there are signs that mobile devices and cloud networks are also becoming a ripe target for cybercriminals.

In this issue, we cover the IT security industry from all angles -- from state-sponsored cyber-espionage to the latest hack against Research in Motion's BlackBerry smart phones. We look closely at the security and privacy risks from social networks, especially Facebook and offer important advice on password management and blunting threats against enterprise databases.

I'd like to also call special attention to an interview with Alexey Polyakov, head of Kaspersky Lab's Global Emergency Response Team (GERT), where we dive deeper into the common configuration mistakes that expose businesses to malware infestation. This is the kind of information that goes a long way to helping businesses reduce their exposure to hacker attacks. I hope our readers find the information and advice useful and actionable.

Stay secure!

- Ryan Naraine
Editor-in-Chief

Editor-in-Chief: Ryan Naraine
News Editor: Darya Skilyazhneva
Copy Editor: Christian Perry
Design: Shiek Mohamed

Editorial matters:
editorial@secureviewmag.com
http://www.secureviewmag.com

Anatomy Of The RSA Targeted Zero-Day Attack

BY URI RIVNER

Uri is head of new technologies, identity protection and verification at RSA, the security division of EMC. He is responsible for moving new technologies and innovations from concept to reality and has been involved in the research of online fraud and the development of mitigation strategies and technologies to prevent it.

Advanced Persistent Threat (APT) attacks typically have three main phases. The first is the social engineering attack; that's one of the key elements that differentiates an APT from good old hacking. From the very first mention of APTs, it's been clear that these attacks will be difficult to defend against, as they use a combination of social engineering with vulnerabilities in the end-point to access users' PCs. Once inside, you're already in the network; you just have to find your way to the right users and systems, and carry on with "regular" hacking activities.

End-point security struggles with protecting against more simple form attacks such as data-stealing Trojans, which is why you can find so many examples of banking Trojans, or employees compromised with a Trojan that grabs the corporate data and sends it to a Trojan mothership halfway across the world. If Trojans available for sale from every digital thug on the cyber block are getting through the perimeter, what should we expect when it comes to the more devious attacks that are currently launched against private-sector companies?

The social engineering part is equally simple. In the early 1980s, you would have guys like Matthew Broderick in "War Games," searching for modems connected to sensitive networks. Broderick mapped networks and found weak spots. His attacks had nothing to do with the users; he used weaknesses in the infrastructure. But if Matthew was staging an APT hack today, the first thing he'd do is visit social media sites. He'd collect intelligence on the organizations' people, not infrastructure. Then he'd send a spear-phishing email to the employees of interest.

In our case, the attacker sent two different phishing emails over a two-day period. These emails were sent to two small groups of employees. When you look at the list of users who were targeted, you don't see any glaring insights; nothing that spells high profile or high-value targets.

The email subject line read "2011 Recruitment Plan". This was intriguing enough for one of the employees to actually pull the email out of their Junk Box and double-click on the email attachment, which was an excel spreadsheet titled "2011 Recruitment plan.xls".

The spreadsheet contained a zero-day exploit that installs a backdoor through Adobe Flash vulnerability (CVE-2011-0609). Adobe has since released an emergency patch for the zero-day. The exploit injects malicious code into the employee's PC, allowing full access into the machine. The attacker in this case installed a customized remote administration tool known as Poison Ivy RAT variant; if you are familiar with APTs, you will recognize Poison Ivy, as it has been used extensively in many other attacks, including GhostNet. Often these remote administration tools the purpose of which is simply to allow external control of the PC or server are set up in a reverse-connect mode: this means they pull commands from the central command and control servers, then execute the commands, rather than getting commands remotely. This connectivity method makes them more difficult to detect, as the PC reaches out to the command and control rather than the other way around.

The next phase of an APT is moving laterally inside the network once it's compromised some of the employee PCs. The thing is, the initial entry points are not strategic enough for the attack-

ers; they need users with more access, more admin rights to relevant services and servers.

This is one of the key reasons why, having failed to prevent the initial social engineering phase, detecting it quickly is so important. In many of the APTs publicized in the last 18 months the attackers had months to do digital "shoulder surfing" on the attacked users, map the network and the resources, and start looking for a path to the coveted assets they desired. Then they use the compromised accounts, coupled with various other tactics, to gain access to more "strategic" users. In the RSA attack, the timeline was shorter, but still there was time for the attacker to identify and gain access to more strategic users.

The attacker first harvested access credentials from the compromised users (user, domain admin, and service accounts). They performed privilege escalation on non-administrative users in the targeted systems, and then moved on to gain access to key high value targets, which included process experts and IT and Non-IT specific server administrators.

If the attacker thinks they can exist in the environment without being detected, they may continue in a stealth mode for a long while. If they think they run the risk of being detected, however, they move much faster and complete the third, and most "noisy", stage of the attack. Since RSA detected this attack in progress, it is likely the attacker had to move very quickly to accomplish anything in this phase.

In the third stage of an APT, the goal is to extract what you can. The attacker in the RSA case established access to staging servers at key aggregation points; this was done to get ready for extraction. Then they went into the servers of interest, removed data and moved it to internal staging servers, where the data was aggregated, compressed and encrypted for extraction.

The attacker then used FTP to transfer many password-protected RAR files from the RSA file server to an outside staging server at an external, compromised machine at a hosting provider. The files were subsequently pulled by the attacker and removed from the external compromised host to remove any traces of the attack.

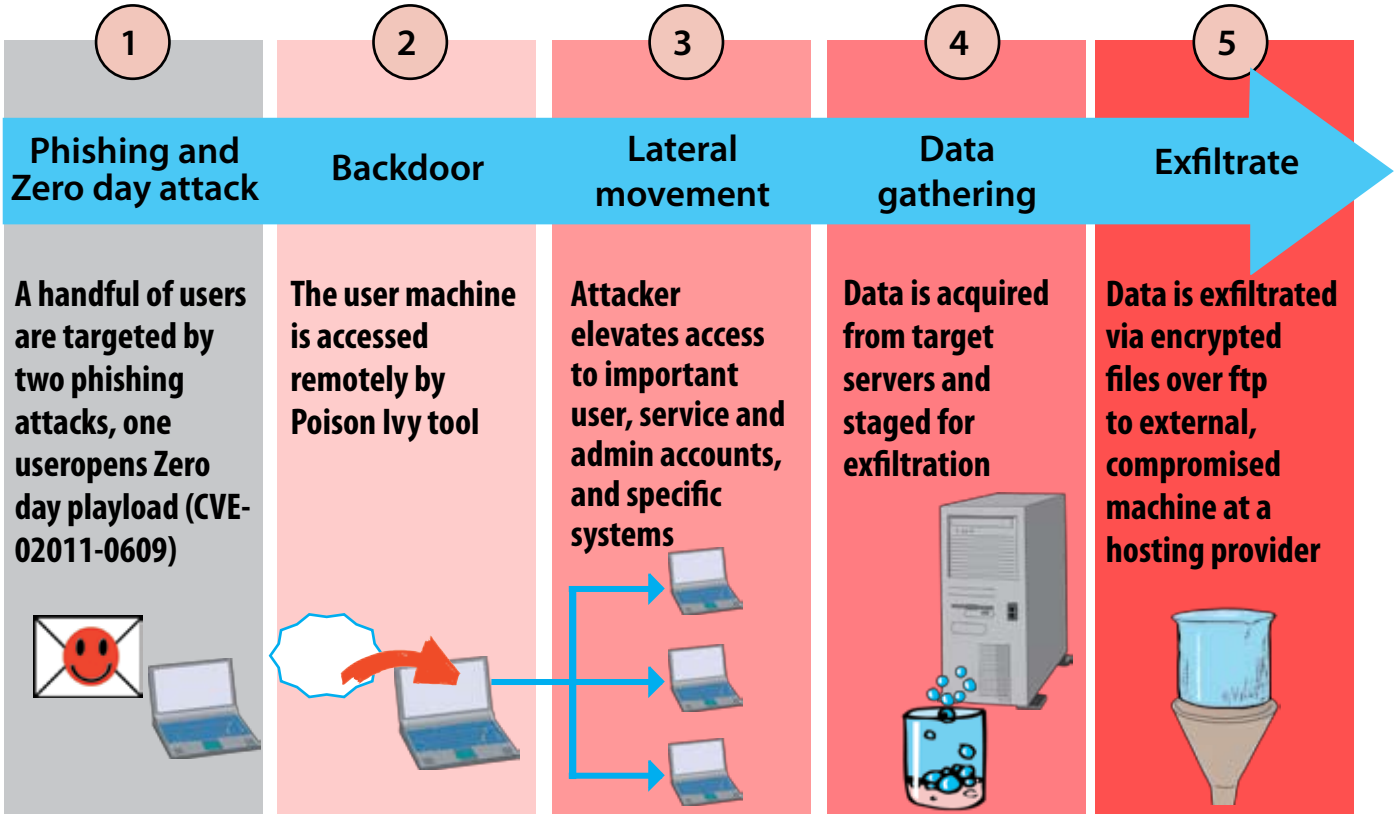
I hope this description provides information that can be used to understand what has happened and correlate with other APTs.† In addition, three URLs associated with this attacker are:

Good[DOT]mincesur[DOT]com | up82673[DOT]hopto[DOT]org | www[DOT]cz88[DOT]net

Perhaps this incident can be used as an exercise when you look at your own infrastructure and wonder what mitigation options you have against similar attacks. There's a reason why APTs are so dangerous, and it has to tell us something. As an industry, we have to act fast and develop a new defense doctrine; the happy days of good old hacking are gone, and gone, too are the old defense paradigms. New threats call for new strategies.

At RSA, we're already learning fast, making both small-term hardening moves and giant strides towards establishing a whole new defense doctrine. We're implementing techniques that just a couple of weeks ago I thought were in the realm of long-term roadmaps.

There are so many historic examples of campaigns that seemed hopeless at the time but were then turned through sheer will, creativity and leadership. I'm sure that in a few years, Advanced Persistent Threats will become a familiar, almost mainstream form of attack and that we'll be able to deploy effective defenses against those who want to spy and control on our intellectual property, digital assets and critical infrastructure.



A Tale of Stolen Certificates

BY COSTIN G. RAIU

Director, Global Research and Analysis Team, Kaspersky Lab.

BY ALEX GOSTEV

Chief Security Expert, Global Research and Analysis Team, Kaspersky Lab.

Introduction

The Stuxnet virus was first discovered in June 2010 (*1) by the Belarusian company Virus Blokada. In the beginning, it garnered plenty of attention from the security community for two important reasons: Its LNK-based, 0-day Autorun-like spreading mechanism and its signed system drivers. Weeks later, as analysis progressed, Stuxnet became even more famous for other traits, notably its complex and highly specialized subroutines designed to sabotage industrial processes through reprogramming of Siemens PLCs.

Currently, there are four known Stuxnet driver files. Three of these show a very interesting particularity, as they are digitally signed with the private keys from two digital certificates belonging to Realtek and JMicron — both well-known companies. The fourth known file is not signed and seems to be a memory dump of one of the two Realtek-signed drivers.

The first set of signed drivers was detected in the wild and is dropped by the so-called "original" Stuxnet sample. The third driver, which was signed by JMicron Technology of Taiwan, was found on July 17, 2010, by the Slovakia-based security company ESET (*9). Except for the digital signatures, it is similar in behavior to the previous version.

Many questions have been asked in relation to these three signed driver files (*3). For instance, how did the attackers manage to obtain the private keys required to sign them? Were Realtek and JMicron involved in the operation and willingly sign the files? Since both companies have development offices in China, are the Chinese involved?

In this article, we will not try to answer all these questions, but we will try to answer perhaps the most interesting question: How did the attackers sign the Stuxnet drivers with the RealTek and JMicron certificates?

STUXNET'S SYSTEM DRIVERS

The original version of Stuxnet drops two Windows driver files. These files are named "mrxcls.sys" and "mrxnet.sys" and get written in the %SYSTEM%\drivers folder (for example, C:\Windows\System32\drivers\mrxcls.sys and C:\Windows\System32\drivers\mrxnet.sys).

Filename	Size	MD5
mrxnet.sys	17,400 bytes	cc1db5360109de3b857654297d262ca1
mrxcls.sys	26,616 bytes	f8153747bae8b4ae48837ee17172151e

Table 1 — Stuxnet.a drivers

Each of these two driver files has a specific purpose. Mrxnet.sys is the rootkit component used to hide the presence of the worm on removable drives. Mrxcls.sys has a different purpose: It loads the malicious code each time the operating system starts by injecting the worm code into "services.exe" and two processes specific to Siemens software (Step7/S7 and WinCC).

On July 17, 2010, (*9) a third Stuxnet driver file appeared, under the name "jmidebs.sys":

Filename	Size	MD5
jmidebs.sys	25,552 bytes	1e17d81979271cfa44d471430fe123a5

Table 2 — The "unknown" Stuxnet variant driver

According to the program header, jmidebs.sys was compiled July 14, 2010, at 12:05:36 (GMT+2).

The timing of the third driver file's appearance is interesting because on July 16, 2010, VeriSign revoked the certificate used to sign the first two drivers, belonging to Realtek Semiconductor, although there was a discussion about this going on for a few days already on various security forums. However, it's important to note that the Realtek certificate used to sign the two drivers expired on June 12, 2010, so it couldn't have been used again to sign potentially new variants of the worm. This could explain why the attackers experimented with a second certificate that had a much longer validity time (until July 26, 2012).

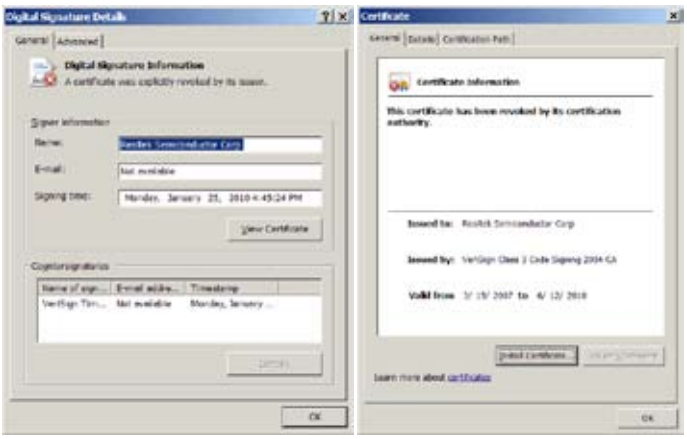


Figure 1 — Stuxnet Realtek signed driver

The software digital signing process has an interesting particularity: It may include the date and time when the signing took place. The signing times of the two Stuxnet.a drivers are the following:

File	Signing time
mrxcls.sys	Monday, January 25, 2010 4:45:14 p.m.
mrxnet.sys	Monday, January 25, 2010 4:45:24 p.m.

We can observe that it took the attacker roughly 10 seconds to sign the first file and then move to the second. In the JMicron-signed driver, the timestamp field seems to have been corrupted, though it's not known whether this was intentional.

Following is the compilation time:

File	Compilation/linking time
mrxcls.sys	Thursday, January 01, 2009 8:53:25 p.m.
mrxnet.sys	Monday, January 25, 2010 4:39:24 p.m.

The compilation time of mrxcls.sys could indicate some kind of error, because it's unlikely that somebody would be working on the driver on January 1. In addition, the year (2009) is potentially odd, although it's possible the driver had been created that early in the project's development by some hardworking contributors.

On the other hand, the compilation time of mrxnet.sys is a lot more interesting. The driver seems to have been compiled January 25, 2010 4:39:24 p.m. (GMT+2). According to the information from the digital signature, the very same driver had been signed January 25, 2010 4:45:24 p.m.(GMT+2). If the information is correct, it indicates a very important trait: The attacker compiled the file and signed it only 6 minutes later.

If this is the case, the attacker most likely had direct access to the secret key from the certificate and used it to sign the just-compiled driver.

BROKEN URL CONNECTION

Inside the digital signature attached to the drivers is a field called the description URL. This field can be used to specify additional information about the signature or about the publisher of the program.

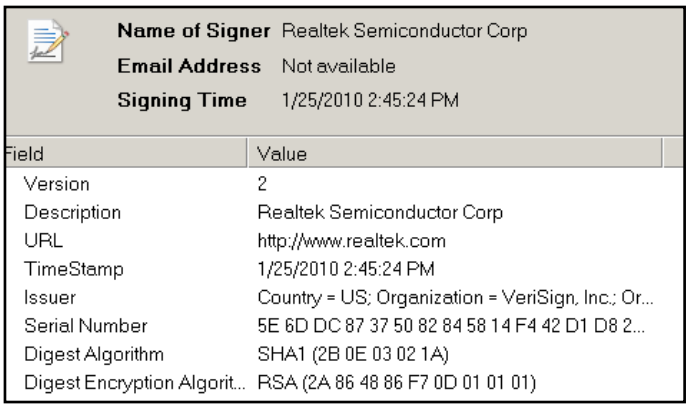


Figure 2 - Digital signature information — Stuxnet.a mrxnet.sys

For the two drivers dropped by Stuxnet.a, the URL was set to "http://www.realtek.com," as pictured in the previous image.

Here's how a valid, genuine digital signature from Realtek for a system driver looks:

Figure 3 - Genuine Realtek signature — rtlh86.sys

Comparing the Stuxnet Realtek-signed drivers and a genuine Realtek-signed driver reveals a number of important differences:

- The Stuxnet driver signature includes a URL (http://www.realtek.com)
- Genuine Realtek drivers do not have URLs (however, we should note that some regular Realtek software, such as driver installers, do have digital signatures with URLs)
- The Stuxnet driver signature has an additional "TimeStamp" field in the signature block

Here's how the digital signature for the JMicron-signed driver looks:

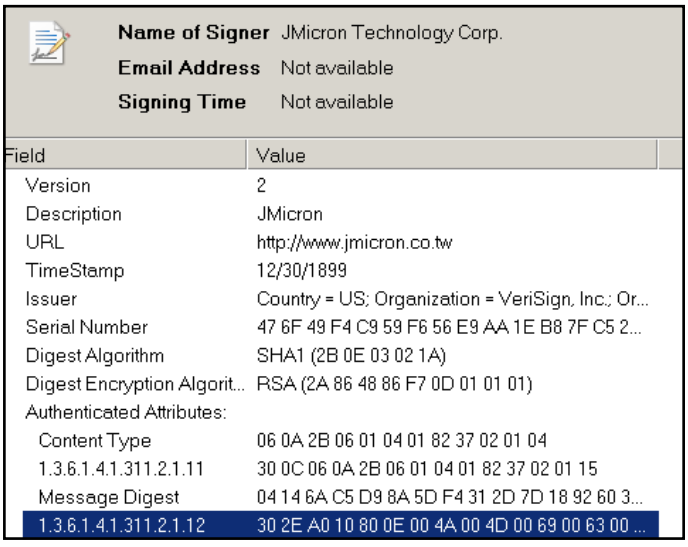


Figure 4 — Unknown Stuxnet driver digital signature information

While working with our colleagues, Mike Pavluschik and Sergey Golovanov, on this topic, we discovered an interesting element in the Unknown Stuxnet driver. Inside the digital signature block, the description URL has been set to "http://www.jmicron.co.tw". However, this URL returns a "Server not found" error.

Figure 5 - Invalid description URL — www.jmicron.co.tw

This is because the domain "jmicron.co.tw" doesn't exist. Looking at other JMicron-signed drivers reveals the problem. For legitimately signed JMicron drivers, the description URL is actually missing. Perhaps our attacker was in a hurry and executed a mental typo: "www.jmicron.co.tw" instead of the valid Web site, which is "www.jmicron.com.tw"

There is another reason to believe our attacker was in a big hurry when using the JMicron certificate. If we look at the resource section of the JMicron-signed Stuxnet driver, we find the following interesting fields:

Child Type:	**StringFileInfo**
Language/Code Page:	1033/1200
Comments:	change me
CompanyName:	change me
FileDescription:	change me
FileVersion:	3.00
InternalName:	change me
LegalCopyright:	change me
LegalTrademarks:	change me
OriginalFilename:	change me
ProductName:	change me
ProductVersion:	3.00

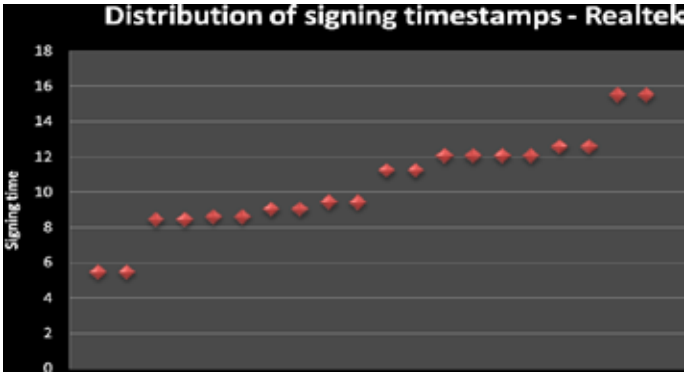
Note that almost all the fields have a default "change me" value. This was not the case with the Realtek-signed drivers, where the fields have been carefully set as follows:

Child Type:	**StringFileInfo**
Language/Code Page:	1033/1200
CompanyName:	Microsoft Corporation
FileDescription:	Windows NT NET Minirdr
FileVersion:	5.1.2600.2902 (xpsp_sp2_gdr.060505-0036)
InternalName:	MRxCls.sys
LegalCopyright:	? Microsoft Corporation. All rights reserved.
OriginalFilename:	MRXNET.Sys
ProductName:	Microsoft« Windows« Operating System
ProductVersion:	5.1.2600.2902

EARLY BIRD CATCHES THE WORM

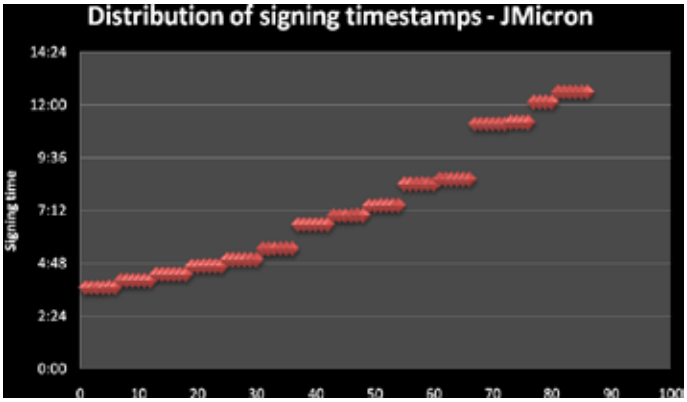
Assuming the signature timestamp (as well as compiling time) in the Stuxnet drivers (and binaries, for a fact) are correct, we had the idea to compare them to the normal timestamps we would find in the legit Realtek- (and JMicron-) signed drivers. Since both Realtek and JMicron are based in Asia (and are potentially doing driver development in the Hsinchu Science Park, which is GMT+8), we expect their sig-

natures to be in this time zone.
We examined 20 Windows Vista drivers downloaded from Realtek's FTP site (*5).



The Realtek signatures were mostly performed between 8 a.m. and 1 p.m. (GMT+2), which corresponds to 2 p.m. and 7 p.m. in Hsinchu. This is a trend that's typical for software developers. There are four cases (two groups of two) that are special, files signed very early in the morning (May 31 05:45:24 2010 GMT+2 and May 31 05:45:23 2010, GMT+2). There are also two cases of files that were signed late in the evening (Mar 04 15:50:17 2010, Mar 04 15:50:15 2010 GMT+2). Even these last two files, signed very late in the evening, are not as late as the Stuxnet.a drivers, which were signed on January 25, 2010 at 4:45 p.m., GMT+2. Should the Stuxnet.a drivers have been signed in Hsinchu, then it was probably very late in the evening— around 10:45 p.m.

The same situation can be found in the case of JMicron-signed drivers:



We examined 86 driver files from JMicron (*6) that contain digital signatures. According to the timestamps, they have been signed between 3:30 a.m. and 12:30 a.m. (GMT+2), again indicating an Asian time zone. The difference here seems to be that the JMicron developers sign files earlier in the morning than Realtek developers, or that the JMicron drivers are signed within an even later time zone than GMT+8.

Based on these facts, the most likely conclusion is that the attacker must have operated on a different time zone, such as somewhere in Europe, Africa or Middle East, and probably not in China or the Americas.

PRELIMINARY CONCLUSIONS

At his point in our research, we can draw a number of preliminary conclusions. First, it appears the attacker tried to imitate the signatures from Realtek and JMicron but was not sufficiently careful and slipped in a few inaccuracies. The attacker set the description URL to "www.realtek.com" for the first version of the drivers. In the case of JMicron, the attacker set the URL to the non-resolving "www.jmicron.co.tw."

Another observation regards the compiling/signing times. It seems that the attacker compiled the driver and a few minutes later signed it. This seems to indicate that the attacker probably had access to the certificates on the compiling machine, which makes it less likely to believe it was done on the spot. Finally, the signing time seems to indicate that our attacker was based somewhere in a European time zone rather than in Asia (or the United States, for that matter).

Together with the fact that the Stuxnet.a mrxnet.sys component signing was only 6 minutes after the compile time, this seems to be a strong indication that the attacker obtained access to the certificates and used them on the attacker's own machine at the attacker's convenience.

STOLEN OR SOLD CERTIFICATES?

Obviously, one of the most important questions is how the attacker got possession of the Realtek and JMicron certificates and secret keys. One possibility is that a disgruntled employee sold them on the black market. Yet, considering these are two separate companies, this prospect seems a bit unrealistic. Another possibility is that somebody physically broke into these two companies and stole the certificates for these specific purposes.

Finally, it's possible that the certificates were stolen with a specialized piece of software, a Trojan horse. This theory is supported by the presence of Trojan horses that steal digital certificates (*7). One such example is the infamous ZeuS Trojan, also known as Zbot, PRG, Wsnpoem, Gorhax and Kneber.

ZeuS steals certificates primarily for the purpose of compromising banking accounts. Some banks around the world do issue digital certificates to their users who need to access their online banking systems. Even if an attacker steals their login/username pair, they can't access the bank's online system without the associated certificate.

ZEUS STOLEN CERTIFICATES

The way ZeuS steals digital certificates from a computer is by exporting them from the Windows certificate store.

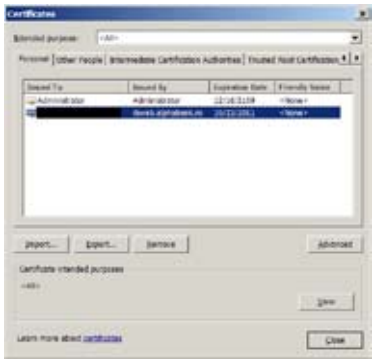


Figure 6 — Certificate store view in IE

This can be accomplished rather easily, thanks to a Windows-provided API function called PFXExportCertStoreEx() that does exactly that. The function's export will be saved into a "PFX" file, which ZeuS usually uploads to a dropzone.

We have successfully tested this procedure on a system by infecting it with a ZeuS variant and then watching it extract and upload a PFX file that contained the certificate and the private key required to sign software. To sign a program using a PFX (personal exchange format) container, we first need to export the data from the PFX file to a PEM (privacy enhanced mail) file using OpenSSL.

C:\> openssl.exe pkcs12 -in mycert.pfx -nocerts -nodes -out mykey.pem

To extract the private key from the PEM file, we can use the "pvk.exe" tool created by Dr. Steven N. Henson (*8).

C:\> pvk.exe -in mykey.pem -topvk -strong -out mykey.pvk

To sign files, we need the software publishing certificate as well, which can be extracted from the PEM file using OpenSSL:

C:\> openssl.exe crl2pkcs7 -nocrl -certfile mycert.pem -outform DER -out mycert.spc

Once we have the .PVK and .SPC files, we can proceed to signing program files using the SignCode.exe tool from Microsoft.

```
Usage:  SignCode [options] [FileName]
Options:
 -spc  <file>        Spc file containing software publishing certificates
 -v    <pvkFile>     Pvk file name containing the private key
 -k    <KeyName>     Key container name
 -n    <name>        Text name representing content of the file to be signed
 -i    <info>        Place to get more info on content (usually a URL)
 -p    <provider>    Name of the cryptographic provider on the system
 -y    <type>        Cryptographic provider type to use
 -ky   <keytype>     Key type
                       <signature|exchange|<integer>>
 -$    <authority>   Signing authority of the certificate
                       <individual|commercial>
                       Default to using certificate's highest capability
 -a    <algorithm>   Hashing algorithm for signing
                       <md5|sha1>. Default to md5
 -t    <URL>         TimeStamp server's http address
 -tr   <number>      The # of timestamp trial until succeeds. Default to 1
 -tw   <number>      The # of seconds delay between each timestamp. Default to 0
 -j    <dllName>     Name of the dll that provides attributes of the signature
 -jp   <param>       Parameter to be passed to the dll
 -c    <file>        file containing encoded software publishing certificate
 -s    <store>       Cert store containing certs. Default to my store
 -r    <location>    Location of the cert store in the registry
                       <localMachine|currentUser>. Default to currentUser
 -sp   <policy>      Add the certification path (chain) or add the certification
                     path excluding the root certificate (spcstore).
                       <chain|spcstore>. Default to spcstore
 -cn   <name>        The common name of the certificate
 -sha1 <thumbPrint>  The sha1 hash of the certificate
 -x                  Do not sign the file.  Only Timestamp the file

Note:   To sign with a SPC file, the required options are -spc and -v if
        your private key is in a PVK file. If your private key is in a
        registry key container, then -spc and -k are the required options.
```

Figure 7 — SignCode usage

We would like to note here that if the person signing the file wants to add a timestamp, this requires an Internet connection and a timestamp server. The URL to the timestamp server can be specified to SignCode using the "-t" switch.

One such service, for instance, is http://timestamp.verisign.com/scripts/timstamp.dll. In the case of Stuxnet, if this URL was used by its authors, then it's probable that VeriSign has the IP address from which the stamping request was received.

CONCLUSIONS

Although much has been discovered about the Stuxnet virus, there are still a lot of mysteries and unanswered questions that remain. Given the complexity of the worm, Kaspersky Lab believes it couldn't have been written without support from a state nation.

It can be assumed that such an entity would have considerable financial resources and could have used any certificates, including ones purchased directly with fake credit cards. The reason why the Realtek and JMicron certificates have been used remains a mystery; however, we can assume this was done to make it harder to track. In addition, signed drivers from well-known hardware companies such as these two are harder to spot.

Through the research in this paper, two obvious conclusions have emerged:

1. The certificates have been used by the attacker at times that indicate an EMEA time zone.
2. Certificates could have been stolen by the ZeuS (or similar) Trojan horse and then sold on the black market to the attacker.

We might never discover who was behind Stuxnet and how the certificates were used to sign the malware. However, if that information becomes known, it will be very interesting to find if our conclusions were right.

1. "Trojan-Spy.0485 and Malware-Cryptor.Win32.Inject.gen.2 Review" http://www.secureblog.info/files/new_rootkit.pdf

2
. Win32/Stuxnet Signed Binaries
https://blog.eset.com/2010/07/19/win32stuxnet-signed-binaries

3. Stuxnet Signed Certificates Frequently Asked Questions
http://www.securelist.com/en/blog/2236/Stuxnet_signed_certificates_frequently_asked_questions

4. W32.Stuxnet Dossier
http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf

5. Realtek Driver FTP Site ftp://210.51.181.211/cn/nic/

6. JMicron Driver FTP Site ftp://driver.jmicron.com.tw/

7. Stolen Digital Certificates Becoming Standard Malware Components http://threatpost.com/en_us/blogs/stolen-digital-certificates-becoming-standard-malware-components-093010

8. PVK File Information
http://www.drh-consultancy.demon.co.uk/pvk.html

9. Win32/Stuxnet Signed Binaries
https://blog.eset.com/2010/07/19/win32stuxnet-signed-binaries

Seven recommendations for a

Safer Facebook

BY EUGENE KASPERSKY

Eugene is chief executive officer of Kaspersky Lab. He is a laureate of the State Prize of the Russian Federation and a member of the Civic Chamber of the Russian Federation.

This year, the world's most popular social network is celebrating seven years of existence. For most of you out there, Facebook has provided a valuable platform for keeping in touch with friends in a totally new way, but for security researchers it's been seven years of new challenges that Web 2.0 has brought to the security area.

The main challenge with social networking security is that social networks are, well, social! Whenever the human mind gets involved, it opens the door for vulnerabilities to be exploited. I'm talking here about human vulnerabilities, the ones that are near impossible to defend. Why – you might ask – and the answer is simple: because the human mind is the hardest thing to patch.

It's been seven years in which malicious social engineering rose to a whole new level. Reconnaissance was made much easier, enabling targeted attacks to soar and a new breed of malicious code appeared, driven by the evolution of social networks: Web 2.0 worms, likejacking scams, clickjacking attacks and even 411 scams via Facebook's chat feature.

Is this likely to stop in the future, and if so, how exactly? The short answer is no, the threats are not going to disappear unless people would stop using social networks or social networks cease to exist. As both cases are highly unlikely to happen anytime soon, I think we should enhance the security and privacy features of our social networks to stop the problems from escalating out of control.

Therefore, here are my seven recommendations for making Facebook a much more private, secure playground:

1. Full SSL browsing enforced and mandatory for everyone. This is already available in Facebook via the privacy settings. This way, all users can make sure nobody is snooping in on their conversations, even if they're browsing Facebook through an untrusted internet connection, and render attack tools such as Firesheep completely useless.

2. Two-factor authentication for all users with compatible mobile devices. Banks are offering e-tokens for their customers to safely accessing their online banking accounts, but in a world where social networking sites are more important than ever, users should have the same technology available for protecting their Facebook accounts as well. This was enabled by Google not so long ago with a relatively simple mobile application. This way, an attacker would have to compromise two devices to get access to a Facebook account.

3. A clear line between trusted and untrusted Facebook apps. Malicious Facebook apps are being analyzed and reported by researchers on a daily basis – so it would be terrific if Facebook would manually check and approve all incoming applications to make sure no malicious app gets on to an user's profile. As this task would probably be impossible, an idea would be to have an ever increasing list of trusted/approved applications that a regular user can add to his profile. If the user wants to use an application that is not trusted, he should be able to run it in some sort of "profile sandbox", so that any malicious activity would not affect other users.

4. Tighten up the "recommended" privacy controls. Currently, the Facebook recommended privacy settings allow "everyone" to access your status, photos, and posts, your bio and favorite quotations and see your family and relationships, while your "friends of friends" only have access to the photos and videos you're tagged in, religious and political views plus your birthday. It is too easy for an attacker to become the friend of a friend of someone and get all the data they need to reset a password for a webmail account.

5. Permanently deleting your account should permanently delete your account -- but it doesn't. "Copies of some material (photos, notes, etc.) may remain in our servers for technical reasons, but this material is disassociated from any personal identifiers and completely inaccessible to other people using Facebook". This needs to be fixed as it is a major privacy and security risk even for people who have removed their Facebook identity.

6. Commit to keeping children safe by taking parental control to a whole new level. Parents should be able to set-up limited access accounts for their children, as sub-accounts under their main Facebook presence. The limited sub-accounts could automatically be turned into full accounts once the child reaches the age of consent.

7. Educate your users. Yes, the page at facebook.com/security is a good

Continued on page 35

Online Shopping Carts: Web's Weak Link

BY BRIAN KREBS

Brian is a freelance journalist based in the U.S. He previously worked as a reporter for The Washington Post from 1995 to 2009, authoring more than 1,300 blog posts for the Security Fix blog, as well as hundreds of stories for washingtonpost.com and The Washington Post newspaper.

The old truism "You get what you pay for" often applies when shopping for a website design company. Bargain website designers are giving some clients more than they expected — vulnerabilities that allow hackers and phishers access to customers' sites.

A Texas-based Web design company promised to help customers set up custom Web sites at a competitive price, but that wasn't all it was giving them. Hackers had broken into the company's site and planted a tool that turned its server into a point-and-click weapon for hijacking other Web sites and seeding them with malicious code and phishing kits.

The planted hacktool exploited a security hole in client sites that were running outdated versions of osCommerce, a popular open-source shopping cart program that is rapidly becoming a vector for malware, spam and phishing scams.

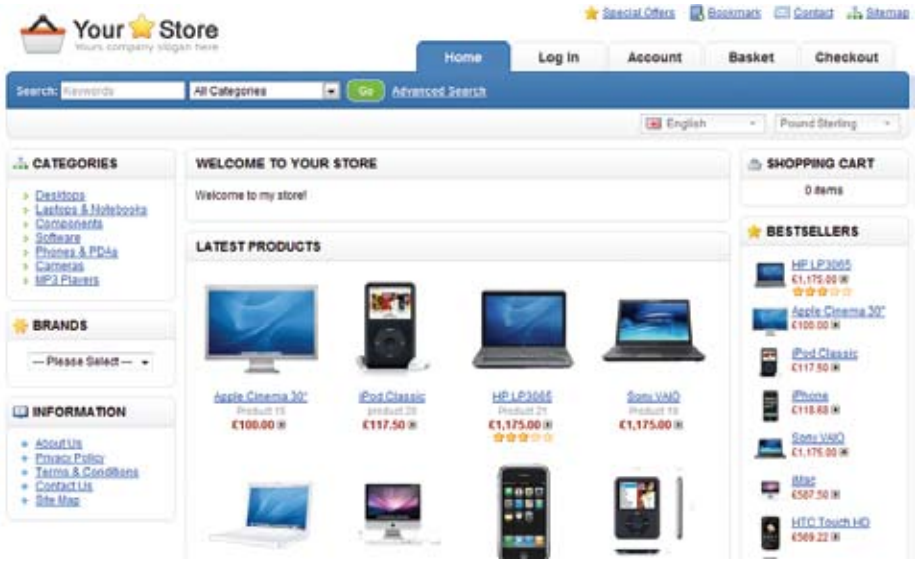
Security experts say an alarmingly large number of phishing and scam pages are being enabled by online stores using severely outdated shopping cart software that is trivial to compromise. It may be difficult to imagine a software suite, whose sole purpose is to enable online credit card transactions, being so widely insecure that someone would write a hacktool to attack it, but experts say shopping cart vulnerabilities are extreme examples of a pervasive and growing problem: Far too many Web site owners update their underlying operating systems and Web server software but neglect security of the applications that run on those systems.

"In the Web app security industry, we know that everyone gets a penetration test; it's just that some you pay for and some you don't," said Jeremiah Grossman, chief technology officer for WhiteHat Security, a Web site risk management company based in Santa Clara, Calif.

"Shopping carts are notoriously vulnerable because they are pretty complex pieces of software, and as a result online retailers usually choose not to write them in-house" and instead get them off-the-shelf, Grossman said. "It's the responsibility of the shopping cart software vendor to supply patches to address known vulnerabilities, and it's the retailer's responsibility to install them. Both parties have had a poor history in doing their job in a timely fashion, usually opting to focus on new functionality rather than fixing security loopholes."

John LaCour, president of PhishLabs, of Charleston, S.C., said his organization's latest statistics show that roughly 5 percent of all new phishing sites appear to be legitimate e-commerce sites that may have been hacked via insecure osCommerce shopping cart installations.

"Anecdotally, it's rather easy to tell when you see a phishing site that's been hacked via osCommerce, based on the path of the phish," LaCour said.



"In a typical osCommerce installation, the 'catalog' or 'images' directory is writable so that the site administrator can upload pictures of items that are for sale, and we're seeing more and more phishing sites in those directories."

Attackers seem to be having great success with two different remote exploits released last year that allow miscreants to upload arbitrary files to osCommerce servers. The osCommerce team shipped a giant set of updates in November 2010 to plug the holes, but many site owners never update their cart software if it is in place and working correctly, said Ben Maynard, an Australian software developer based in Toronto.

Jeremiah Grossman

"Many of the shopping carts out there have serious security issues, and that's mainly because clients rarely ask you to come back and tweak things once they have them up and running the way they want," Maynard said. "The last thing you're doing for most clients is updating their Web site software, unless they have agreed to some kind of maintenance plan, but most don't."

RESISTING CHANGE

When he worked for his previous employer in Australia, Maynard and a co-worker built their own shopping cart suite from the ground up. That custom code would have come in handy at Maynard's new job in Canada, where his first assignment was to implement a shopping cart for a customer Web site. Unfortunately, his former employer claimed that software as intellectual property.

After researching the available open-source shopping cart packages, Maynard settled on one called OpenCart, in part because it appeared to be the least insecure of the free offerings. But Maynard quickly discovered a glaring cross-site request forgery vulnerability in OpenCart that could be used by attackers to hijack Web sites using the cart software just by tricking the administrator into clicking a poisoned link or visiting a specially crafted Web page while logged in to the OpenCart administration page. Maynard fired off an email to the developer to warn him about the vulnerability and offered to help him fix it.

Maynard said he was taken by surprise when the developer dismissed the vulnerability as trivial and told Maynard to stop wasting his time. Maynard wrote about the experience on his blog, and the OpenCart developer proceeded to leave a series of comments on the blog post, calling Maynard an "idiot" and a "prat."

A month later, Maynard released his own version of OpenCart that fixed the security flaw he had blogged about, as well as two other vulnerabilities. Maynard said the OpenCart developer — apparently still angry over their public argument — promptly reversed all of the security changes and rereleased the vulnerable version. Fortunately, Maynard said, another developer has since taken over the OpenCart project and appears to have incorporated those security fixes.

"Some of these shopping carts have a decent support community, but most of the sites I see using the cart software don't seem to be taking advantage of that or keeping up with the latest versions," Maynard said.

SCRIPT KIDDIES AND SPAMMERS UNITE

The automated exploit tool left behind at the Texas website design company was planted by a Portugese hacking team that specializes in hacking osCommerce sites and uploading backdoor "shells" that can be used to surreptitiously maintain complete control over the sites.

Often, the hackers will deface a site's home page, leaving behind a digital "tag" for bragging rights. And in many cases, the backdoored sites are sold to spammers or converted by the hackers themselves into sites that earn them commissions for pimping rogue online pharmacies, said Peter Bennett, a longtime antispam activist from New Zealand.

Bennett was part of a motley crew of antispam activists, or "antis," that helped to bring down infamous pill spammer Shane Atkinson and other junk email purveyors. Bennett has been operating several "honeypot" machines, Web servers intentionally left vulnerable to entice hackers and allow the study of their methods. Bennett outfitted each honeypot with slightly out-of-date osCommerce packages, and late last year he watched as an Indonesian hackers broke in and began using the servers to relay junk messages advertising pharmacy sites.

Bennett soon discovered that his Web site was part of a giant botnet of at least 1,200 compromised sites that was being used to relay approximately 25 million junk email messages each day, and that a large percentage of sites in the botnet also appeared to have been compromised via osCommerce shopping cart vulnerabilities.

Bennett said the defacement tags left by the hackers implicate two Indonesian miscreants. One uses the online handle "kaMtiEz" and is part of an Indonesian defacement team called Magelangcyber Team; the other goes by the nickname "Hmei7" and leaves his mark by inserting a page labeled "indonesia.htm" on all defaced sites. Both hackers are listed among the Top 10 most active website defacers, according to Zone-h.org, a website that tracks defacement activity.

The New Zealand antispam activist isn't the only one noticing malicious activity emanating from sites tagged by kaMtiEz and Hmei7. A source at a major Canadian bank, who spoke only on condition of anonymity, said websites defaced by these two hackers are showing up time and again as hosts for phishing pages targeting Canadian financial institutions.

SHOPPING FOR A BRIGHTER FUTURE

Unfortunately, the shopping cart vulnerability is only one of many website flaws that allow phishers, spammers and defacers to ply their trade with ease. Steven Burn, a volunteer incident handler with Malwaredomainlist.com, flags hundreds of new sites each day that have been hacked through a variety of Web application vulnerabilities and are used to host malicious software and exploit kits.

Burn said one of the biggest causes of compromised sites he's seeing now are content management systems like

Continued on page 35

Hacking the BlackBerry in 10 seconds

BY WILLEM PINCKAERS

Willem is a senior security consultant at Matasano Security. He specializes in vulnerability research, reverse engineering and penetration testing.

I picked up a BlackBerry on Thursday and cleared the entire weekend to work on the phone. I got up at 10 a.m, hacked until 5 a.m., slept a few hours, then got back to the BlackBerry.

It's four weeks until CanSecWest, a security research conference in Vancouver that hosts the Pwn2Own competition. My team is planning on attacking several targets, and I'm responsible for the BlackBerry. The BlackBerry OS is patched slowly, so there's a window of opportunity to use a bug that is public and fixed in WebKit but still present and unpatched in BlackBerry's WebKit-based browser. That would spare one of the bugs in my personal collection while still demonstrating enough to win the contest.

There's no debugger on BlackBerry, no crashdumps and no documentation. This is a true black box pen test. Writing an exploit for a completely unknown device is far harder than doing the same on a platform where there is a lot of existing information, but it's more fun to explore the unknown.

Since we do not have a debugger or crashdumps on the device, an information leak is needed. Normally I would use a debugger or crashdumps to figure out why the attempt doesn't work, but on a BlackBerry, the only thing we can see is whether the device crashes, doesn't crash or hangs for a long time. That's not much to go on. Using an information leak, it's possible to dump parts of memory that will help me figure out what is happening. The first output from the information leak are memory addresses and a dump of the WebKit code as it is running on the BlackBerry.

The next step is to pick a vulnerability to exploit. There is a buffer overflow vulnerability that was released in November 2010 but is still present on the BlackBerry. Perfect. I use the information leak to find the address of the heap, which is where the browser stores temporary data like the current Web page and images on the page. Most modern operating systems randomize the address of the heap, which would make exploiting a little bit harder; however, the BlackBerry has no such protections.

To exploit the vulnerability I have to set up the heap in a specific way so I can overflow a specific structure on the heap. This structure is the internal representation for a piece of text on a website. The vulnerability is in the handling of the text nodes, so this is a good target to overflow. The first field of the text node structure contains a pointer to a list of functions associated with the text node. If I change this pointer I can redirect execution to our code and get complete control of the BlackBerry.

Once I have a stable way to organize the heap and reliably overflow the pointer to the functions, we can start testing. The first test attempts to redirect execution to code that already exists on the BlackBerry. Instead of the JavaScript nodeType call returning the value 3, I redirect it to existing code elsewhere that returns 0. Now I can control the execution flow in the browser.

Modern operating systems have protection that prevents executing code from the heap. We do not know if BlackBerry is a modern operating system, but we will soon find out. I write machine code to the heap that returns 17 (I just like 17). Using the information leak, I find the location of our machine code and use the exploit to redirect execution to my code. It returns 17! The browser crashes, but we can worry about that later. Now we know that the heap is executable—another protection that isn't present on the BlackBerry. Having an executable heap makes my life a lot easier. Instead of having to use pieces of the machine code that are already present in the browser, I can just add new on the heap and execute it from there.

Now to write new code to execute on the BlackBerry. Since I do not know

anything about the BlackBerry operating system, I cannot easily write code that does something useful. I don't know how the BlackBerry opens and reads a file, or how it makes a network connection. By reading the error messages embedded in the code (for example, Error: cannot open file %), I make an educated guess that the code just before the error message is likely used to open files. By reading the assembly code, I slowly managed to piece together most of the needed functionality. Now I can open, read, write and delete files.

I still don't know how the BlackBerry makes network connections. Without making network connections, I can't exfiltrate the data from the BlackBerry. I decide to save myself some time and just use JavaScript. The hard part is making the exploit so stable that it first executes our machine code and then returns to JavaScript without crashing the browser. I like my exploits to be graceful.

After several hours of crashes, reading memory and reading the binary assembly code, the exploit is stable enough. I can now execute my code, return to JavaScript and let JavaScript send the stolen photo to my remote server. The exploit is even stable enough to perform this round trip multiple times. The browser doesn't crash in the end, so an attacker could steal data from the BlackBerry and the user wouldn't notice anything. Graceful.

Now I need to convert the binary data to a text format that JavaScript can send to us. I could use JavaScript to do the conversion, but it's risky since the encoding in JavaScript frequently crashes the browser. I end up writing a little bit of machine code to do the hex encoding and use JavaScript to deliver the hex-encoded string.

With a few more tweaks, I have a fully working and reliable exploit that will grab a photo and the BlackBerry Messenger contact list and also creates a small file on the BlackBerry. Two weeks before CanSecWest, we get the lottery results for position in the competition. It went badly for my team. We ended up last (or close to last) on most of the targets we wanted. That means the targets the other team members worked on are

Continued on page 34

Willem Pinckaers (sitting center) gets ready to hack into a BlackBerry device to win the 2011 Pwn2Own challenge.

RIM BlackBerry Security Checklist

BlackBerry smartphones include many easy-to-use security features that go a long way toward protecting your private information from prying eyes. To get that protection, follow this checklist:

Use a strong password

Setting a strong password is the single easiest and most effective way to lock down your private data. Without a password, much of your data is accessible to prying eyes. With a password, you are far more secure.

Passwords:

- Must be 4–14 characters in length
- Cannot be identical characters (1111) or sequences (1234)

Set the number of password attempts

If a password is typed incorrectly 10 consecutive times, all of the information on the BlackBerry smartphone is automatically deleted. This is a security feature.

Encrypt data on your BlackBerry smartphone and media card

It is also smart to encrypt all data on your BlackBerry smartphone and your media card. Encryption mixes everything up so no one but you can read anything without the correct password.

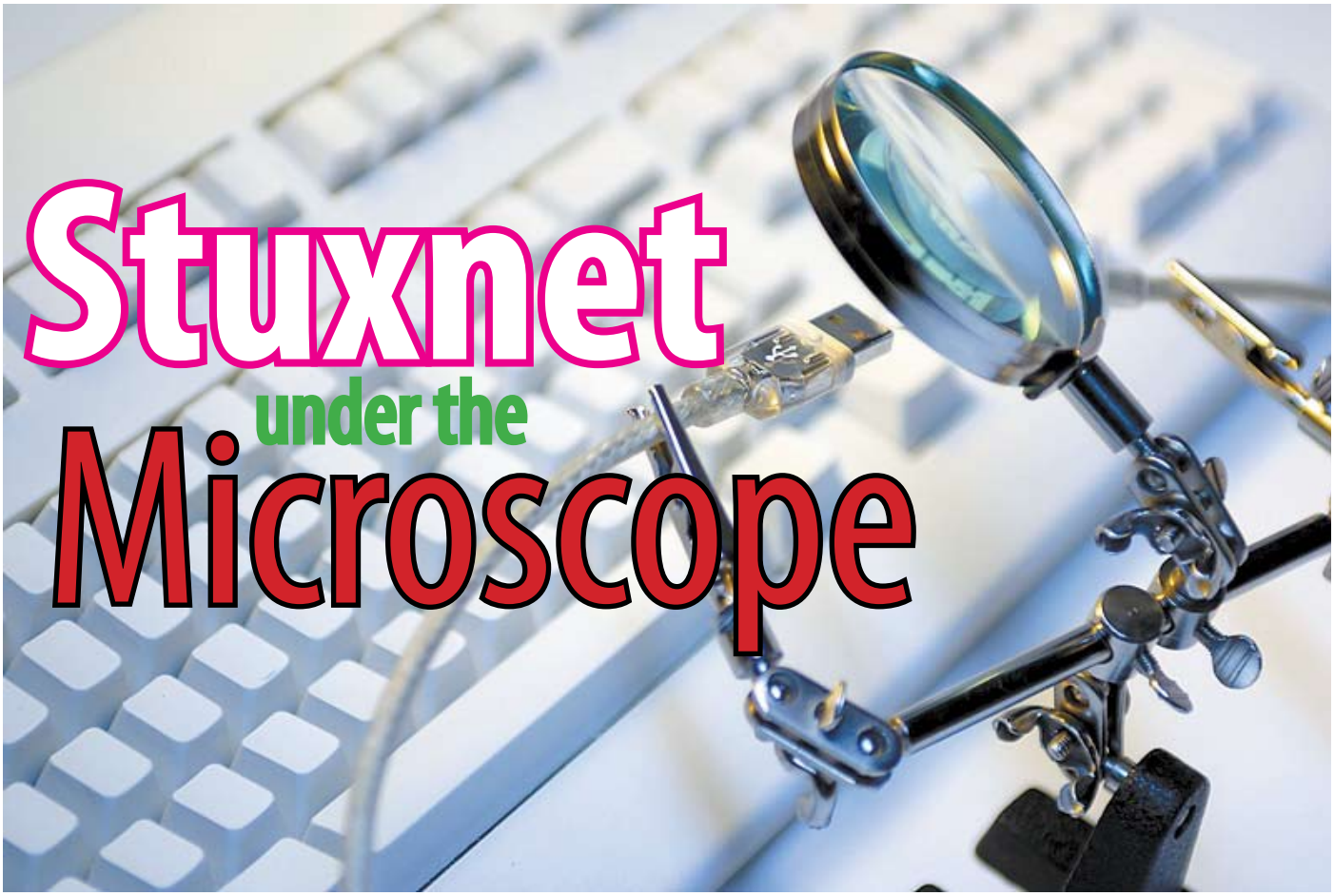
Lock your phone automatically after a certain amount of time

You should set the Security Timeout feature to automatically lock your BlackBerry smartphone after a set time of inactivity (maximum is one hour). For lost or stolen smartphones, this is a critical security block.

Lockdown Bluetooth

Bluetooth is a terrific way to connect to other devices for hands-free talking or even to play music wirelessly. But Bluetooth also adds a potential entry for malicious hackers. Luckily, changing a simple setting can block most threats.

Instructions for implementing this checklist available at: http://www.blackberry.com/newsletters/connection/personal/i410/checklist.shtml



Stuxnet under the Microscope

BY ROBERT LEMOS

Robert is a freelance journalist based in the United States. He writes about computer security, technology and space science.

Researchers interested in peering into Stuxnet's effectiveness discovered a valuable tool last month: Stuxnet keeps a diary. As the worm spreads among computers, it documents its activities, including the date of the infection, the name of the compromised system, the domain name, and the internal and external IP addresses, according to research performed by security company Symantec.

"Every time Stuxnet infects a computer, it appends a little bit of information about that infected computer onto itself," said Liam O'Murchu, a researcher with the Cupertino, Calif., company. "So when Stuxnet spreads to the next computer, there is a trace of where it came from."

The diary of Stuxnet reveals its cyberattack history consisted of 10 separate assaults starting in June 2009, targeting five organizations with offices in Iran and supporting the widely held theory that Stuxnet targeted Iran's nuclear processing capabilities. The infection logs also reveal that more than 1,800 organizations, as identified by specific domains, have been infected by the program. Symantec collected 3,280 samples of Stuxnet, which allowed researchers to track the worm's path of infection through more than 12,000 computers.

Symantec's research and efforts by other security experts — including German researcher Ralph Langner and antivirus companies Kaspersky and ESET — have peeled back more layers of Stuxnet's functionality and given security experts a glimpse into the in-

ner workings of the worm. Researchers concluded that Stuxnet is a prime example of a sophisticated cyber-weapon and will undoubtedly impact the future development of malicious software. The ability of the worm to rapidly infect process-control and other sensitive systems, resist efforts to extract it from networks and limit its spread to a small subset of computers offers an example that will only increase the sophistication of future cyber-attacks.

While the Aurora attacks against Google and other technology companies were likely at the behest of a nation-state, Stuxnet shows a level of professional attacks that dwarfs others, said Frank Heidt, CEO of security company Leviathan Security. "With Aurora, we said this is a professional operation in the sense that the people behind it wore uniforms. [Stuxnet] was professional in the sense of the craftsmanship. This was a very well-crafted, well-planned and very well-executed attack. It is sort of a night-and-day difference of professionalism."

The glimpse at a professionally constructed threat may be the biggest contribution of Stuxnet to the world of attack code. Stuxnet's main lesson may not be the attack it conducted but the possibilities the code poses to less-professional attackers.

TARGET: EMBEDDED SYSTEMS

Take embedded and nontraditional computer systems. While Stuxnet can spread to nonspecialized desktop systems, it specifically targets computers running Siemens SIMATIC Process Control System 7.

While security researchers have often maintained that such systems could easily be targeted by malware, in turn bridging the division between the digital and the physical world, naysayers have labeled such warnings as fearmongering. Stuxnet demonstrates that such attacks are not only possible, but effective. U.S. and Israeli officials, for example, estimate that Iran's nuclear program may have been set back several years by the Stuxnet attack.

Eric Byres

In a recent paper, three researchers concluded that the worm's multitude of infection vectors and control systems' need to have some connectivity make it nearly impossible to defend against a well-constructed, multipronged attack such as Stuxnet. Eric Byres, chief technology officer with Tofino Security and one of the paper's authors, discovered the worm could find multiple pathways to infect control systems.

"Stuxnet has given the whole world a crash course in writing PLC code for dummies," Byres said. "They have given [other attackers] a path to impact control systems."

The worm also demonstrates the vast benefits to attackers of focusing vulnerability research on specific file formats that may not be widely used. Stuxnet appends itself to Step 7 project files, which allows it to jump from workstation to workstation in a manufacturing facility. File formats that are not widely used generally do not have the same level of scrutiny — in terms of vetting for security flaws — that, say, Adobe PDF files or Microsoft Office files might undergo. Even systems and networks secured according to the best practices set by Siemens (the manufacturer whose software was targeted by Stuxnet) are vulnerable to attack by the program, according to the report. Other attackers will likely take the lesson to heart, said Symantec's O'Murchu.

"I think Stuxnet has shown that we will see more of these types of attacks. People didn't believe that you could get onto these systems, but Stuxnet has shown that it is possible," O'Murchu said.

CONTROLLED SPREAD

Stuxnet also demonstrates that the impact of a self-propagating program can be controlled to some extent. While the program was not finely targeted — Symantec estimates that more than 100,000 systems were infected by Stuxnet — its creators blunted the impact of the attack on non-Iranian control systems. The worm will not execute its primary infection routines on systems that do not match the targeted programmable logic controllers.

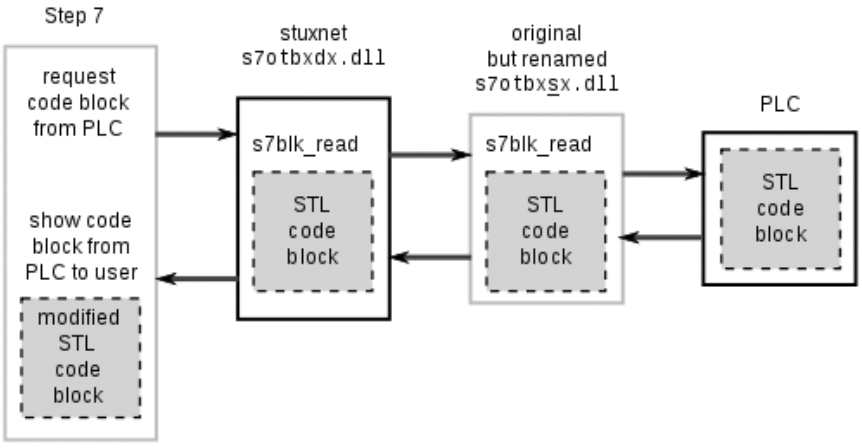
The ability to blunt its attack on nontargeted systems shows the level of professionalism inherent in Stuxnet, Leviathan's Heidt explained.

"There is very, very wise exception handling in it," he said. "There is very wise environmental reconnaissance."

The attackers who created the program released the original in June 2009, followed by updates in March 2010 and April 2010. Antivirus companies did not report detecting the worm until July 2010. Like the Nimda worm, which spread among Microsoft systems in September 2001, Stuxnet has multiple vectors attack. Nimda spread through email, network shares and vulnerabilities in Microsoft's Internet Information Services (IIS) Web server. Stuxnet has at least seven modes of propagation, including through network drives, WinCC database servers, Step 7 project files and print spoolers.

The March 2010 version added the

Continued on page 34



Overview of Stuxnet hijacking communication between Step 7 software and a Siemens PLC.

Apple iOS And The Enterprise

BY ANDREW STORMS

Andrew is director of security operations at nCircle Network Security.

Are you sick and tired of the IT guys telling you, "No iPads or iPhones"? Could your work benefit from the seamless calendar tools, messaging tools and social media access delivered by your personal iPhone?

I'm going to let you in on a little secret. Those hard-hearted IT guys would love to use iPhones and iPads, too, but they are worried about connecting iOS devices to enterprise networks. Your IT team also knows that even if it hasn't sanctioned your Apple device, you are using it for a few business tasks anyway, aren't you? Fess up: You know that right now there are a couple of work-related documents — and maybe even a sales contact draft or two — sitting on your iPad. Maybe you even keep all your passwords in a file stored on your iPad or iPhone for "easy access."

Cut all of those IT curmudgeons a little slack, though. I may be biased, but IT is a tough job. Information technology is one of the only places in the enterprise where users routinely get away with breaking corporate rules. In finance, for example, just because your accountant wants to drive a Ferrari doesn't mean he can just buy one and expense it. But, somehow, users feel that this approach is acceptable with IT. In users' minds, it's totally OK to connect your unsanctioned iPhone to your work computer to sync documents; after all, what can it really hurt, right?

I am an avid Apple user and responsible for an enterprise IT team. I completely understand why you want to use your Apple devices in the office; I'd like to use mine, too. But this doesn't change my answer when employees ask me about using their iPhones on our corporate network. It turns out there are some good reasons why IT teams have been trying to hold back an avalanche of iOS devices. The news is not all bad: It also turns out that there are some good reasons why IT should take a second look at some of the old iOS support issues, because Apple is actually making progress in several areas that might improve security in the long run.

There is an unusual amount of tension between enterprise IT and end users around the use of iOS devices, and both sides bear some responsibil-

ity. Apple's iPhones and iPads can make users more efficient, and IT does have legitimate issues with Apple's enterprise tools. Apple plays the biggest part in this dynamic, though, because it continues to prioritize development of end-user features ahead of solid security tools and architecture. The natural outcome of this development strategy is a dramatic increase in friction between users who want access to those cool new features at work and the people responsible for keeping your enterprise network compliant and operating securely.

ENTERPRISE UNFRIENDLY

From an IT perspective, iOS support is still fraught with issues that aren't clear-cut. First, despite the similar functionality and processing power shared by iPhones/iPads and laptops, users have a very different attitude about security when using these various devices. This fundamentally different mindset is not surprising given the evolution of Apple devices as "hip" consumer technology. It's only a problem when users bring their consumer-based, can't-live-without-it attitudes to the enterprise, where all users are (supposed to be) ruled by a very different set of requirements.

The lack of enterprise tools for the iPad and iPhone shouldn't be surprising. The operating system for iPhones and iPads (iOS) is a derivative of Apple's OSX, which also lacks many of the enterprise tools IT needs to meet management objectives as well as operational and compliance initiatives. Fortunately for everyone caught in this impasse, Apple is starting to address some of these deficiencies. Unfortunately for all concerned, the development of these tools is coming slowly, and Apple still has a long way to go to catch up with competitors.

When compared to other operating systems that were developed specifically for smartphones, like RIM's BlackBerry, iOS has a much broader base of potential functionality, yet it lacks many capabilities we would come to expect from a full-fledged operating system. Because iOS is built on OSX, it's not unreasonable to compare it to the management tools available for Windows 7.

Compare iOS with Windows 7 or RIM's operating system and it's clear that Apple is only now starting to receive some passing grades for enterprise management and security.

Historically, one major defect with iOS devices is that IT administrators lacked mature centralized management tools like those included in a Windows laptop or even RIM's BlackBerry devices. Recently, Apple has been taking security concerns more seriously and made significant headway last summer in supporting enterprise needs. The expanded controls provided by the Configuration Utility, such as adoption of certificate-based authentication, over-the-air provisioning and new data encryption methods, all made significant headway against basic IT requirements.

Another of IT's big gripes about Apple's configuration tool has been that it relied on the user to accept new configuration files by email or URL. We can all agree that most users would rather be playing Angry Birds than following some random new IT directions on how to apply a policy configuration file. Apple has addressed this by permitting the device to obtain configuration updates without user interaction. This still doesn't measure up to the BlackBerry Enterprise Server deployment, but it's nonetheless a solid step forward.

Configuration standards for Apple devices don't seem like a big deal to users. However, this problem is so big for IT that many enterprises have turned to third-party configuration and management solutions from companies such as MobileIron and BoxTone to regain some configuration controls missing from Apple's implementation. There appears to be a fair number of people betting that Apple is not going to deliver a competitive iOS configuration solution anytime soon.

Third-party solutions to the configuration issues are a positive development, but gaining control of configuration parameters is just one piece of the iOS configuration puzzle. Deciding exactly what the optimal configuration file should contain is another piece. The Center for Internet Security (CIS), a security standards organization, has produced an iPhone security configuration standard that should serve as the absolute minimum benchmark requirement for any business choosing to support iOS devices. How you make sure users don't deviate from this standard and prove it to your auditor is up to you. Be forewarned: Meeting these requirements may prove to be costly and cumbersome.

EYE ON SECURITY

Application configuration also proves to be a murky area, especially given that enterprise applications can be subject to a boatload of compliance requirements. Today, IT teams often depend on vendors to deliver applications configured to meet the regulatory compliance requirements appropriate for their intended use. For example, are those applications your doctor is using on his iPad HIPAA compliant?

If you are responsible for your company's compliance audit results, it is important to carefully vet the vendor's claims for each application. It also pays to ensure your vendors have a clear understanding of your regulatory requirements before deploying iOS applications. Vague vendor assurances will be cold comfort if your company fails an audit because one of these applications was incorrectly configured. After all, if we are talking about compliance for your business, it's your butt on the

line, not your vendor's.

With any computing device, IT is always trying balance end-user freedom against IT and security requirements. In this thankless task, the user is always the biggest problem. It's never easy to find a compromise between user access and security controls that pleases everyone, but in the case of iOS, the balance has been particularly hard to achieve.

Audit requirements are a great example of this problem. It's no easy task to provide the necessary audit data about iOS users. Think about the questions that auditors will ask: Can your iOS implementation provide your IT team with information about every user's historical actions? Can you provide a breakdown of what numbers were dialed, what text messages were sent, what company data is on the device, and the time, date and location the device was last unlocked? Your auditor may want to see all of this information.

Enterprise IT groups have become accustomed to the extensive logging capabilities of Windows and RIM tools for critical audit logs and to allow administrators to check on user compliance levels and actions. It's normal for IT to expect the same functionality from an iOS device, but in the past these tools did not exist. The good news is that Apple now provides a way for IT administrators to capture console logs. It's not perfect, but it's an improvement. The bad news is that both RIM and Microsoft have a big head start on Apple's enterprise credentials in all of these areas.

Even if your business isn't saddled with stringent compliance requirements, there are other iOS issues that give IT heartburn. The sad fact is that for all of Apple's "enterprise ready" hype, what users do on Apple devices can be an opaque conversation between the user and Apple. It's a shame, but Apple and its partners probably have more information about your user base than you do if recent Apple lawsuits regarding distribution of user data to advertising companies are any indication.

Another demonstration of Apple's hype being misaligned with corporate requirements is its security patching process. Both iOS and Mac OSX ship with dozens of publicly sourced applications. For example, the Safari browser is heavily dependent on WebKit, an open-source Web browser engine. There are many benefits that come with using open-source software, but it also means that any security bugs in the open-source engine could become public before Apple can distribute a patch. The classic example of this problem is the early iPhone bugs Charlie Miller found in WebKit. These kinds of problems could be mitigated if Apple opened up more of iOS for third-party security software development, but so far Apple's ecosystem remains closed.

Apple devices also fail to fit into normal enterprise IT security programs that include risk management procedures. Installing malware detection software or intrusion detection software on iOS devices would help mitigate ordinary security risks. Unfortunately, there is no endpoint software (such as that available from Symantec or McAfee) for the iPhone or iPad. Risk management teams need to have an action plan in place if significant security holes become public, and it's difficult to develop risk mitigation plans without any third-party tools.

MORE TROUBLE AHEAD

Assuming IT can find its way around the configuration, compliance and risk mitigation issues, it still has to navigate a bunch of basic hardware concerns unique to Apple devices. Fixing broken

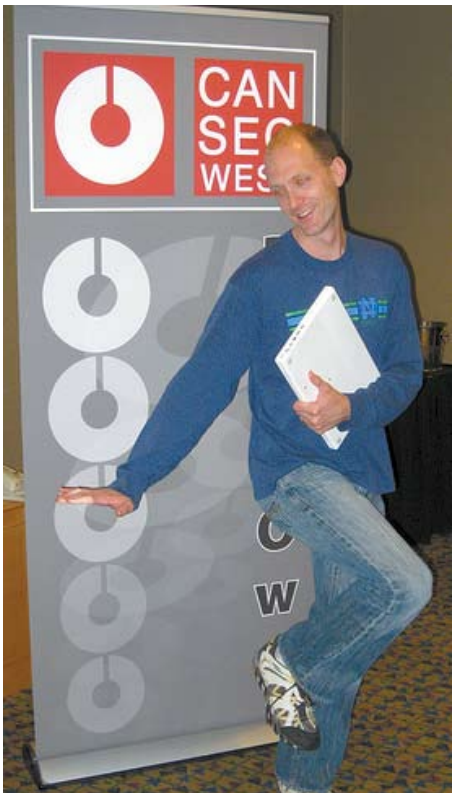


Photo: TippingPoint Zero Day Initiative

Charlie Miller

iOS devices is problematic because there are no in-field replaceable parts. Even a simple battery change requires returning the device to the factory. All of these day-to-day problems become very painful for users and IT teams. They also consume far more than their share of the already stretched-to-the-limit IT resources. Believe me — every IT team has a long, rapidly growing list of other tasks. In short, supporting Apple hardware is extremely expensive.

Another serious consideration centers around the user's relationship with iTunes. Apple carefully vets every app in its store, so users haven't had to think about security concerns. Users trust Apple apps, and this often gets in the way of the more critical decision-making process generally applied to

corporate applications.

iTunes is a finely tuned consumer technology marketplace. Combine users' implicit trust in Apple apps and the iTunes single-click buying process with the impulse-driven, consumer mindset of most iOS users, and you get an environment perfect for cybercrime. Add recent trends toward mobile payments and "malvertising" and you get a combination of security variables that gives security and compliance teams the worst sort of nightmares. And I haven't even touched on the data privacy concerns connected with apps that release users' GPS coordinates or usage data. The phenomenal success of iTunes is definitely a double-edged sword for the enterprise.

Apple recently provided IT the option of completely disabling access to the iTunes store. This sounds like a good idea to IT, but it's likely to be difficult in practice. There are a lot of reasons users love their iPhones, and many of those reasons are tied to iTunes. There are no good answers for these dilemmas.

You would think that all the security attention the iPhone has received would mean that iOS devices would be rapidly becoming more resilient to security breaches. Instead, serious bugs continue to haunt iOS with great regularity. Apple skeptics think there are indications that the underlying architecture of iOS lacks proper security design. If that is true, iOS will be plagued with security issues until the underlying issues are corrected — a slow process at best.

For the time being, IT security teams have to consider iOS support as one more security time drain. IT has to prepare for an increase in the number of regular risk discussions as more security bugs are found in iOS. Moving forward, everyone supporting iOS must keep a close eye on what security researchers are finding and how Apple is responding to manage risk levels.

Another disturbing sign of flawed iOS security is jailbreaking, which allows customization and the use of applications unauthorized by Apple. Every version up to iOS 4.2.1 has been jailbroken in some way. Given this track record, we should expect iOS 4.3 to be broken as well. While some may consider it a cool thing to do, jailbreaking is a form of a security breach. Apple has many valid (and maybe some invalid) reasons to stop this practice, but so far, it has been unable to stop it.

Physical theft of an iPhone combined with jailbreaking tools has been shown repeatedly to subvert device passcode and data encryption protection. This issue is probably the single biggest concern for all enterprise IT teams when they consider support for iOS devices. Unfortunately, many IT teams lack staff expertise to test and retest data device encryption exploits. For those who have to support iOS, I can only recommend diligent application of basic security practices. Make sure you require passcodes and data encryption on every device and keep your fingers crossed that Apple correctly implements its crypto.

Most of these problems are not unique to iOS, and they all can be overcome. Even with all these serious issues, the single biggest challenge for the enterprise has been Apple's attitude toward security and enterprise requirements. If Apple were to focus some of its famous innovation on the delivery of fast responses to corporate requirements and well-tested, well-documented bug fixes, corporate adoption would dramatically increase overnight.

iOS SUPPORT IN THE ENTERPRISE

• **Treat all devices with the same user mentality.** Train your users to understand that iPhones and laptops have the same potential for malware and can be breached in the same ways. Teach users to "think twice before you click" with iOS devices in exactly the same way they would on a laptop.

• **Don't neglect audit requirements.** If the device has confidential data or access to the network, it may be susceptible to the same audit requirements as a laptop or desktop. Figure out how you are going to address this and build it into your resource plan and business process before you allow iOS devices on the network — or as soon as possible, if they are already on the network.

• **Implement vulnerability and configuration management.** Mobile devices should not be immune to your existing vulnerability and configuration management policies. Ensure you have a way to test each device for known vulnerabilities. Make sure you can accurately account for the configuration of the device and be able to track and document changes over time.

• **Use asset tracking.** Treat these devices like you would with any other corporate asset. Ensure they are purchased, accounted for and tracked like a laptop. You should have a method to retrieve the device and its data and be able to securely delete everything from it when needed.

• **Ride the wave, but don't let it drown you.** Apple devices are here to stay. Take a proactive approach and engage your user community. Sometimes the best thing IT managers can do is get out of their comfort zone and talk with their user base. Creating a rapport with the users will go a long way when the hard decisions need to be made.

Authentication Comes Of Age

BY ROBERT VAMOSI

Robert is a security analyst and the author of When Gadgets Betray Us: The Dark Side of Our Infatuation with New Technology (Basic Books, April 2011).

In the wake of the WikiLeaks exposure of sensitive U.S. State Department communications, a little-noticed press release from the White House offered new guidance for government agencies on employee access to sensitive documents. Buried among the predictable suggestions was a call for government agencies to use risk and fraud analysis like that currently used within financial services for routine file and network access.

Going forward, passive screening similar to methods used to detect illegitimate bank and credit card account access might also one day be applied to the broader control of legitimate Internet access itself. If coupled with the U.S. government's proposal for a new framework for interoperable authentication, such behind-the-scenes screening might even allow individuals greater online control over what personal information about them is shared—and with whom.

BEHAVIORAL MODELS

When you access an online banking account, the data that's fed into a financial service risk and fraud analysis varies. The top level is interactive and includes username and password and perhaps a challenge response question (What model was your first car?). The next level is passive and includes many behavioral elements. Say you always check your bank balance online at 6 a.m. from the same computer. Your financial institution will build a history of these transactions and then use it to model your behavior. If one day you check your bank balance at 10 p.m. from a foreign location, using a foreign computer, you should expect to see additional requests for authentication before gaining access.

New network access controls for government employees therefore might work something like this: User Bob attempts to access a sensitive State Department document at 3 a.m. and transfer it to an external hard drive. The behavioral analysis engine will factor in a variety of information before letting

Bob gain access. Is Bob on the black list? If not, is there a historical record of Bob accessing such files past midnight? If so, is Bob using the same computer (such as an office computer)? Has Bob previously used an external hard drive? If so, is the velocity in Bob's keystrokes or navigation different (say, an automated script versus Bob's typical entry behavior)? And so on.

In the end, the system may prevent Bob from completing his task (maybe he doesn't have clearance) or he'll be asked for more proof he's really Bob. This provides government agencies with far more oversight than what currently exists. But while this is comforting to an employer, there's a danger that Bob's online behavior — especially when it's not collected through his employer but instead through a commercial third-party — may be used in ways he didn't intend.

LOYALTY CARDS

John Zurawski shops at grocery stores that don't require loyalty cards to get additional savings. "How often I buy rum to go with my coke is no one's business except my own, but would an employer be interested in that data? Possibly," said Zurawski, vice president of sales and marketing at Authentify, a company that sells out-of-band authentication technology. For example, employees working in a regulated environment, such as airline pilots, might wonder whether that grocery store information can or should be reported to the FAA when any pilot buys rum and coke in the same purchase.

Zurawski warns that our "digital persona" is still a new enough concept that actual protection under the law may not yet exist. "You can personally delete your favorites, delete your browser history and avoid downloading any files, but your ISP still has the log files for the IP address assigned to an account attached to your username and password at any given time," he said. This could lead to a robust market for such new, secondary PII (personally identifiable information) as our online behavior.

Alisdair Faulkner, chief products officer at ThreatMetrix, disagrees, saying such data collection is often contextual. For example, Google Checkout and Apple's App Store both allow consumers to keep their identity and purchasing details with just a single party, yet consumers "still benefit from being able to transact and purchase services from third-party suppliers without PII," he said. Amazon, for example, lets consumers buy through Amazon, a trusted brand, although the purchased product may actually come from a lesser-known third-party.

EVOLUTION OF PII

Both Authentify and ThreatMetrix are companies that provide authentication solutions without necessarily relying upon PII. Traditionally, PII has been defined as information that is considered unique to you, such as information found within your driver's license or passport. It includes name, address, driver's license number, date of birth and biometric information, such as height, weight and gender. Today, PII is digital and subject to misuse, with much of that information available on social networking sites or to someone half a world away with access to a data breach.

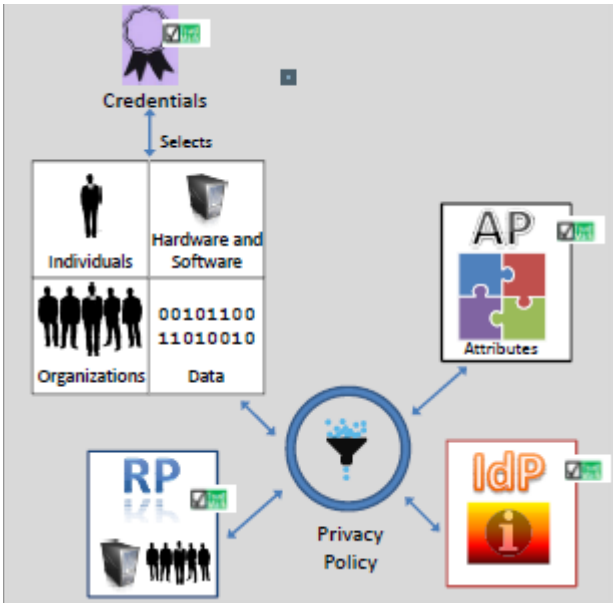
"Authentication without PII is not only possible, it will be demanded by consumers and society at large," Faulkner said. For example, ThreatMetrix favors device fingerprinting, a technology that collects hundreds of attributes about a device, such as its MAC address, operating system and browser version, to determine whether a person logging into an account is the legitimate owner. It does not collect the identity of the account owner, but instead verifies only that the hardware characteristics of the connecting device are the same as when the accountholder enrolled.

Thus, a Linux box running Windows XP in a virtual session from a Vietnam-based IP address might be flagged as suspect at a regional U.S. bank. To rule out false positives, ThreatMetrix says it collects hundreds of attributes about the devices accessing a site, so if an accountholder changes a browser or even the device itself, the person won't get automatically blocked.

Authentify uses out-of-band authentication to contact online accountholders by phone before login access is granted. This is particularly useful when someone attempts to change the address on an account or restore access to a dormant account. Zurawski said he believes that online behavior is "absolutely an identifiable trait" and wonders whether online behavior could indeed qualify as a new piece of PII.

LOCATION, LOCATION, LOCATION

A new piece of PII is location. The IP address logged in an online session already tells the site the location of the accountholder, but Faulkner said Visa Europe is experimenting with using a mobile device's built-in geolocation capabilities. For example, an ATM transaction might be denied or require additional authentication if your registered mobile device is not near the ATM. "As a consumer, the idea of using my phone location as an additional authentication factor may actually be welcomed if it cuts down on identity theft and the problems that causes," Faulkner said.



This chart shows what the government envisions as a trusted identity ecosystem. Credentials are assigned to individuals, hardware, organizations, and data.
Source: DRAFT National Strategy for Trusted Identities in Cyberspace

The mobile device may soon further authenticate its user if it has downloaded and installed a token from one or more third-party authentication providers. In the fall of 2010, the U.S. government proposed within its National Strategy for Trusted Identities in Cyber-

Continued on page 34

Cyberwar Takes Center Stage

At the largest IT security conference in North America, cyberwar, mobile security, cloud computing and secure software development drove the conversation.

BY GEORGE HULME

An award winning writer and journalist, for more than 20 years George has written about business, technology, and IT security topics. He currently freelances for a wide range of publications, and is security blogger at InformationWeek.com.

For 20 years, the RSA Conference has been the annual focal point of the IT security industry. If attendees to the RSA Conference 2011 arrived in San Francisco feeling confident they could keep their networks and data secured from attack, they probably didn't fly home quite as confident.

First the good news: Attendance was strong this year. Vendors appeared more upbeat about their business than in the previous few years, and it was apparent on the show floor. Vendors didn't hold back when it came to vying for attention, as a man on stilts, a magician and even a sumo wrestler paraded about their respective vendors' booths to lure in prospects. The first RSA Conference, in 1991, had 50 attendees and ran for six hours. The attendance this year was an impressive 18,494.

The thirst for security information is much greater than it was 20 years ago. So much so, in fact, that a shadow IT security conference has emerged where presenters provide talks that didn't make the cut for the main show but remained worthwhile. Called Security B-Sides, it ran through Monday and Tuesday of RSA week and drew a full house itself.

Now for the bad news: The vibe at this year's RSA Conference was unlike any other of the past decade. Many of the security professionals at the show felt they are facing adversaries they can't stop with practical security defenses, such as threats like the Stuxnet worm that allegedly struck an Iranian uranium enrichment facility, and the successful attacks on security consultancy HBGary Federal by the pro-WikiLeaks hacktivist group Anonymous. In the backdrop of all of that was the rising number of impressive and successful attacks, such as those of Operation Aurora and Night Dragon, against large corporations.

The scent of fear wasn't lost on the vendors attending the show. Their overall message tapped into the unease, with booth presentations talking of the difficulty of stopping advanced, motivated adversaries, the malicious omnipotence of Stuxnet-like worms, and the weaknesses of cloud computing, virtualization and other rising technologies. Fortunately, many of the sessions at both Security B-Sides and the RSA Conference focused on the basics (or what should be the basics): Protecting sites from attack, improving application development, hardening databases, mobile security and improving end-user security awareness.

CYBERWAR IMMINENT?

Throughout the keynotes, talk of cyber-espionage, cyberwar and sophisticated threats resonated onstage. National Security Agency and Cyber Command Director General Keith B. Alexander cited the growth of mobile devices and Web-based computing, along with the increased reliance on digitally stored information, for creating new attack points against nation-states. Alexander said the rate of technological change is moving remarkably fast, in turn creating both tremendous opportunities for productivity gains and tremendous vulnerabilities.

When it comes to IT security, Alexander made it clear that now is a critical time. He said that while we've seen advanced attacks, and many advanced digital attack tools have been created, the world has yet to see them all in action. "Most of the destructive tools being developed haven't been used; we need to use this window of opportunity to develop defenses," he told the standing-room-only audience.

Alexander argued that protecting the Internet and private and public networks requires a team effort between governments and industry. The general also made it clear that, as far as the

Photo: RSA Security

HIGH-LEVEL DISCUSSIONS: James Lewis, Michael Chertoff, Michael McConnell and Bruce Schneier on the RSA Conference stage.

United States is concerned, "cyberspace" will be considered a defensible domain just like air, land and sea. He also outlined how the U.S. military is working to create defenses that are agile enough to change as technology changes and provide for an early-warning system of attacks.

That message echoed the points made in the Tuesday keynote by Deputy Secretary of Defense William Lynn III, who issued warnings about the abilities of attack software to damage critical infrastructures, such as water supplies and power plants. That kind of capability means that nation-states can target critical infrastructure and industrial systems. Lynn announced in his speech that the Department of Defense is readying completion of its cybersecurity strategy. The strategy, Cyber 3.0, puts the military on the front line of defending U.S. networks. Lynn acknowledged that cyber-attackers have noticeably stepped up their game in recent years, and the military is concerned not just about government networks, but industrial espionage and commercial theft. "These attacks blunt our edge and saps our competitiveness in the global economy," Lynn said. The protection of critical infrastructure, he said, requires effort and cooperation from both the public and the private sectors.

"Even if we execute it flawlessly, the fact is that the government cannot protect our nation alone," Lynn added. "Cyber defense is not a military mission, like defending our airspace, where the sole responsibility lies with the military."

BEWARE THE CYBERWAR BOOGIEMAN

Vendors used a hard sell on fear of cyberwar on the show floor, and government officials hyped it onstage, but many industry observers and show attendees were skeptical. This was made clear during the keynote panel "Cyberwar, Cybersecurity, and the Challenges Ahead," moderated by James Lewis, director and senior fellow at the Center for Strategic and International Studies. Panel participants included Michael Chertoff, former secretary of Homeland Security; Bruce Schneier, chief technology security officer at BT; and former intelligence chief Mike McConnell.

Lewis initiated the discussion by asking the audience if a number of recent high-profile attacks, such as Stuxnet and Operation Aurora, actually constituted cyberwar. Only a small number of hands in the audience rose in agreement. McConnell and Chertoff agreed that standard digital espionage and information theft don't rise to the level of cyberwar.

"I tend to look at security as a spectrum of challenges, and I draw a bright line between theft and espionage and then the destruction of systems," Chertoff said. "It depends upon the scale [of the destruction] and its genesis as to whether it is war."

To crystallize his point, Chertoff said that nations tolerate state-level spying and the stealing of national secrets, and these physical-world incidents aren't considered acts of war. "However, stealing and espionage are much different things than a sustained attack on the power grid," he said.

threatpost

The Kaspersky Lab Security News Service

Threatpost, Kaspersky Lab's Security News Service, is dedicated to helping security and business professionals succeed by delivering the most important and immediate security news and analysis available.

Threatpost offers a fresh approach to providing up-to-the-minute news and information for IT security and business professionals. Threatpost editors cover today's most relevant security news and the most pressing security issues of the day. They break important original stories, offer expert commentary on high-priority news aggregated from other sources, and engage with readers to discuss how and why these events matter.

Visit threatpost.com today!

IT security author and cryptography expert Schneier made the case that cyberwar is little more than a glamorized term used to sell unnecessary security gear and increase government defense budgets. "There's a lot of push for budget and power, and overstating the threat is a good way to get people scared," he said.

Regardless of how it's labeled, Lewis said the Internet is treacherous and likely to stay that way for everyone for some time. As for potential solutions, the panel put forth little more than increasing regulatory demands on companies to secure their networks and increasing the liability responsibilities for those who fail to protect their systems. "We are not in a state of cyberwar, but we are in something that is dangerous," he said.

Photo: RSA Security

The RSA Conference show floor was packed with security vendors.

MOBILE THREAT RISING

Security vendors have long warned about threats to mobile devices, but the last decade has seen little momentum behind the dissemination of mobile malware — especially when compared with the flood of Windows- and Web-based viruses, worms and attacks. That may be changing.

Mobile phones, tablets and other new technologies are poised to take over the workplace, but organizations that hope to secure them before that happens face an uphill battle, according to a symposium on mobile security. Experts at a half-day mobile security event warned that security, management and data protection are likely to be pressing problems for organizations of all sizes, particularly as consumer-driven adoption of multifunction mobile devices outstrips the ability of IT organizations to manage and monitor the devices within the workplace.

If anyone doubted the veracity of mobile device security, the doubts didn't linger for long. Before the RSA Conference was over, several new variants of the Zeus malware surfaced, aimed directly at common mobile phone devices running on Symbian, Windows and BlackBerry platforms. Similar to mobile Zeus variants that surfaced late last year, the new variants — after making their way onto a target device — ask users for details about their phone and phone number. With that information, the attackers then install additional code that can capture SMS messages. The attacks, according to researchers at Kaspersky Lab, targeted specific banking customers.

The main message from the show was that organizations had better prepare for more mobile attacks, but the security industry, telecommunication carriers and device makers all are working to make devices more secure. The question is: Will defenses arrive soon enough and be smart enough before the threat becomes pervasive?

GET OVER CLOUD SECURITY

Just like mobile security, much has been made of cloud security this year. And, argued David Mortman, contributing analyst at research company Securosis, much of what has been said has generated fear and has been misguided. "It's time to get past all of the fear surrounding cloud computing and ask what is specifically different about securing the cloud," Mortman said to an engaged audience at his Security B-Sides talk, "Cloud Security Realities."

Mortman urged attendees to ignore the superfluous debates raging around the security of private clouds versus public clouds, or if cloud computing is more (or less) secure than on-premises technologies. According to Mortman, the answer to only one question matters: Does cloud computing enable your business to be more cost-effective with an acceptable risk level?

Still, when it comes to securing these systems, Mortman said the effort should be treated much like any other outsourced arrangement. "That means educate yourself on the operational environment of the cloud provider and make solid recommendations to the business on how to move forward with a reasonable level of risk," he said.

That requires asking tough questions about how the provider manages IT risk. For example, how are employees checked? What are their vulnerability and change management processes? How is the infrastructure secured from physical attacks, among others? According to Mortman, the tradeoffs can be worth the effort if a customer obtains the required functionality with reasonable security levels.

Mortman used an infrastructure-as-a-service environment as an example. In such a situation, an enterprise will get a flat network with a firewall, no network segmentation and limited Web application firewall options. Missing from this lights-out data center are advanced security elements such as deep-packet inspection, patch management and intrusion-detection systems. Those responsibilities are up to the customer.

While the technology to secure cloud environments still lacks in maturity compared to on-premises environments, the situation is constantly improving, Mortman said. "Security people who are fighting the move to cloud need to start focusing more on how they can help the business to adopt cloud computing initiatives securely, and stop being a roadblock," he said.

HARNESSING PRIVILEGED Database Users

BY ESTEBAN MARTÍNEZ FAYÓ

Esteban is a security researcher for Application Security Inc.'s Team SHATTER. Esteban has discovered and helped to fix hundreds of security vulnerabilities in major vendor software products, including software from Oracle, IBM and Microsoft. He specializes in application security and is recognized as the discoverer of most of the vulnerabilities in Oracle server software.

An important requirement for ensuring database security compliance is the ability to keep privileged users under control. This process is often called separation of duty and is directly related to minimizing insider threats.

Most security regulations, including PCI-DSS (Payment Card Industry Data Security Standard), SOX (Sarbanes-Oxley) and HIPAA (Health Insurance Portability and Accountability Act), require the implementation of strict separation of duty practices to tackle the increasing presence of insider threats.

Principle of least privilege is a well-known best practice in the security industry in which users should always be granted the minimum privileges needed to perform their tasks. But what about users who are in charge of administering a database? These users frequently require extensive administrative and system privileges that allow them access to sensitive data, including PII (personally identifiable information). Today's enterprises face a difficult task trying to keep these privileged users under control through the implementation of strict separation of duty policies that allow users to perform their jobs and simultaneously secure the sensitive application data stored in databases.

PRIVILEGE CHALLENGES

DBMS (database management system) software, like most other commercial software, has not been designed to provide the strict separation of duty controls required by today's security and compliance regulations. The administrative privileges these applications provide for management and administration can lead, either directly or indirectly, to a complete compromise of the data contained in these systems. Further, the separation of duty controls between the differing administrative functions is very poor.

This software does not have a strong separation of duty between administrative tasks primarily because it was not an initial design goal. DBMS software usually has an administrative account that provides complete control of all the resources, data and configuration changes, typically including privileges over the native auditing facility. Even from accounts that contain limited administrative privileges, it is often possible (though not documented) to escalate privileges to super-admin.

DBMS software has two primary separation-of-duty issues. First, there are administrative privileges and accounts that grant full control over the software, including access to all confidential data. Second, certain restricted administrative privileges that give control over a portion of the complete administration task can be abused to escalate to super-admin privileges.

The first DMBS separation of duty issue is usually created in the design process and is well-known and documented by software vendors. We all know that a person who has DBA (da-

tabase administrator) privileges can read or modify any data that is stored in a database. Some years ago it was probably OK and fairly common to have super-admin DBAs who can access any data in the database. That is no longer the case because databases are required to adhere to strict regulations.

The second group of issues is less known and is sometimes associated with vulnerabilities (security flaws) or design issues in the software. For example, Oracle Database provides CREATE LIBRARY and CREATE PROCEDURE system privileges designed to give specific capabilities to the accounts or roles that hold them. The first privilege allows users to create library objects within the database that points to external files. The second one allows for the creation of database stored procedures.

These two privileges can be used — or abused, in this case — to create a library within the database associated with an external DLL (dynamic-link library) file that contains a function to execute operating system commands. After this, it is possible to create a stored procedure that calls the external library function to execute operating system commands. These operating system commands usually run under the database software owner account, giving full, unrestricted access to all confidential database data. Of course, there are countermeasures to prevent this kind of elevation-of-privileges technique, but they are usually not known and not implemented.

The administrative privileges also expand the attack surface in a database. For example, in an Oracle Database, someone who has the EXECUTE_CATALOG_ROLE role has execute permissions granted to a much bigger amount of packages and procedures that can contain escalation of privileges vulnerabilities. One such vulnerability is the SQL injection in DBMS_CDC_PUBLISH.ALTER_AUTOLOG_CHANGE_SOURCE, which allows a user with an EXECUTE_CATALOG_ROLE role to escalate to DBA privileges, in turn giving access to all database data.

ORACLE DATABASE VAULT

DBMS software vendors are aware of these issues and limitations, and they are providing some solutions to address them. An example of this is Oracle Database Vault, an add-on option installed on top of an existing Oracle Database. The main goal of Database Vault is to provide separation of duty to protect against insider threats.

Photo: Peter Kaminski (Flickr CC 2.0)

In an Oracle Database with Oracle Database Vault installed, it is possible to protect sensitive application data from DBA access. In this way, DBAs will still be able to perform their daily administrative tasks, and the sensitive application data will not be accessible to them because the system privileges do not have power over the Oracle Database Vault protected data.

This, in theory, provides the separation of duty controls required for most security regulations. Unfortunately, this add-on software is good at providing a solution to the first problem (administrative privileges giving full access to all database data), but does not adequately solve the second one (abusing certain administrative privileges to escalate privileges and compromise sensitive data). There are still many ways in which users with administrative privileges can escalate their privileges and compromise all the data stored in the database. The two examples above can be applied to a database protected with Oracle Database Vault.

NATIVE AUDITING LIMITATIONS

Another requirement of most security regulations is the need to keep audit trails of all operations in a database, especially those related to administrative tasks and that have access to sensitive data. DBMS software usually provides native auditing capabilities that can be used for recordkeeping and provides accountability of database operations. This native auditing capability is also subject to attack by privileged or administrator users.

For example, in Oracle Database the SYS user is audited in a different way than other users. The standard native database auditing options do not have an effect over the actions performed by users with SYSDBA privilege. Because of this, it is possible to perform operations as the SYS user without leaving a trace in the standard native database auditing trails. Also, because of some escalation of privilege vulnerabilities, it is possible for users with certain privileges to escalate to SYSDBA privilege and perform operations without being recorded.

EXPECT LIMITATIONS

Today's DBMS software does not offer a complete out-of-the-box solution to protect from the insider threats represented by highly privileged users. This poses a serious security risk because organizations have difficulties not only protecting sensitive data from privileged users, but also keeping records of all the activity performed by them. TeamSHATTER is working closely with database software vendors such as Oracle to find and research these types of vulnerabilities. In addition, vendors are continuously improving their software to make it more secure.

In the face of these issues, there are countermeasures that can be taken to mitigate the risks posed by high privilege misuse. Traditional security best practices such as reducing the attack surface and the principle of least privilege are good examples. There are also some other measures that are less known and specific to each database type. For example, in Oracle, you can configure the database to use a separate low-privilege user for external procedure execution, allowing you to avoid using the same user who is running the database server process.

Most importantly, you should know the limitations of your DBMS software and apply additional protections to secure them. Third-party security software can help identify these risks and implement the countermeasures needed to minimize them. Also, third-party DAM (database activity monitoring) software can help keep the required audit trails for all the operations performed by high-privilege users.

AV-School is an international educational project that unites school pupils, students, lecturers from the world of higher education, teachers of computer science, parents and Kaspersky Lab's own experts. It helps to raise loyal, competent, professional IT specialists who are able to fill key roles in the industrial and commercial sectors.

Antivirus School – A new source of IT knowledge

An innovative educational project from Kaspersky Lab

Here are just some of the features AV-School has to offer:

- → Training via virtual departments
- → Articles by experts in the antivirus industry
- → Blogs by IT security experts and specialists

Antivirus School is always ready to cooperate with true professionals!

More info available at: **www.av-school.com** | **www.av-school.ru** | **www.av-school.pl**

Winning The War But Losing Our Soul

There was lots of noise and distraction on the crowded expo floor of the RSA Conference this year. After a grueling couple of years, vendors were back in force with big booths, big news and plenty of entertainment designed to attract visitor traffic. Wandering the floor, I saw magic tricks, a man walking on stilts, a whack-a-mole game, a man dressed in a full suit of armor and a 15-foot-long racetrack that I would have killed for when I was 10.

BY PAUL ROBERTS

Paul is a veteran IT security journalist. He currently serves as editor at ThreatPost.com and security evangelist at Kaspersky Lab.

The most telling display, however, may have been in Booth 556, where malware forensics company HBGary displayed a simple sign noting it had decided to remove its booth and cancel scheduled talks by its executives. This after the online mischief-making group Anonymous broke into the computer systems of the HBGary Federal subsidiary and stole proprietary and confidential information. The HBGary sign stayed up for a couple of days, got defaced by someone at the show and was later removed. When I swung by HBGary's booth on Thursday, it was a forlorn, empty patch of brown carpet where some marketing types were holding an impromptu bull session.

It would be easy to say that the lesson of HBGary is that anyone can get hacked. After all, the company's founder, Greg Hoglund, is one of the smartest security folks around, hands down. He's a recognized expert on malware and literally wrote the book on rootkit programs. HBGary Federal's customers included the U.S. Department of Defense, as well as spy agencies like the CIA and NSA.

Or maybe the lesson of HBGary is simply not to kick the hornet's nest, so to speak, by needlessly provoking groups like Anonymous that have shown themselves to be hungry for publicity and have little to lose in a confrontation. Maybe the lesson is simply that if you're going to kick the hornet's nest, as HBGary Federal CEO Aaron Barr was determined to do, then at least spend some time securing your Web and email infrastructure and following password security best practices before you commence said kicking.

But I think the real lesson of the hack — and of the revelations that followed it — is that the IT security industry, having finally gotten the attention of lawmakers, Pentagon generals and public policy establishment wonks in the Beltway, is now in mortal danger of losing its soul. We've convinced the world that the threat is real: omnipresent and omnipotent. But in our desire to combat it, we are becoming indistinguishable from the folks with the black hats.

Of course, none of this is intended to excuse the actions of Anonymous, which HBGary President Penny Leavy rightly labeled "criminals," rather than politically-motivated "hacktivists," in a conversation with Threatpost. The attack on HBGary was an unsubtle, if effective, act of intimidation designed to send a message to Barr and other would-be cyber-sleuths: Stay away.

We can see their actions for what they are and sympathize deeply with Barr, Hoglund and his wife, Leavy, for the harm and embarrassment caused by the hackers from Anonymous, which published some 70,000 confidential company emails online for the world to see. Those included confidential company information, as well as personal exchanges between HBGary staff that were never intended for a public airing. It's easy to point the finger and chortle upon reading them, but how many of us (or even the Anonymous members) could stand such scrutiny?

It's harder to explain away the substance of many other email messages that have emerged in reporting by Ars Technica and others. They show company executives like Barr mining social networks for data to "scare the s***" out of potential customers, in theory to win their business. While "scare 'em and snare 'em" may be business as usual in the IT security industry, other HBGary Federal skunkworks projects clearly crossed a line, including a proposal for a major U.S. bank (allegedly Bank of America) to launch offensive cyber attacks on the servers that host the whistleblower site WikiLeaks.

HBGary also was part of a triumvirate of companies, including Palantir and Berico Technologies, that was working with the law firm of the U.S. Chamber of Commerce to develop plans to target progressive groups, labor unions and other left-leaning nonprofits that the Chamber opposed through a campaign of false information and entrapment. Other leaked email messages reveal work with General Dynamics and a host of other companies to develop custom, stealth malware and collaborations with other companies selling offensive cyber capabilities, including knowledge of previously undiscovered (zero-day) vulnerabilities.

Look, there's nothing wrong with private companies helping Uncle Sam to develop offensive cyber capabilities. In an age of sophisticated and wholesale cyber espionage by nation states opposed to the United States, the U.S. government clearly needs to be able to fight fire with fire. Besides, everybody already knew that Greg Hoglund was writing rootkits for the DoD, so is it right to say we're "Shocked!" to read his email and find out that what we all suspected was true? I don't think so.

What's more disturbing is the way that the folks at HBGary — mostly Barr, but others as well — came to view the infowar tactics they were pitching to the military and its contractors as also applicable in the civilian context. How effortlessly and seamlessly the focus on

Continued on page 35

How Secure Are Your Passwords?

BY DAVID EMM

David is a senior regional researcher in Kaspersky Lab's Global Research & Analysis Team. He is a well-known presenter of information on malware and other IT threats at exhibitions and events.

Notwithstanding the technical nature of today's malware, cybercriminals often start by trying to exploit human weaknesses as a way of spreading their programs. This should come as no surprise. Humans are typically the weakest link in any security system. Securing a house is one example: You can have the finest burglar alarm in the world, but if you don't set it, it offers no protection at all. The same is true for online security.

Cybercriminals continue to make extensive use of social engineering to trick people into doing something that they shouldn't. Phishing scams, for example, are designed to lure people to a fake Web site to disclose their personal information, such as usernames, passwords, PINs and any other information that cybercriminals can use. The classic phishing scam takes the form of a speculative email or instant message spammed to millions of addresses in the hope that enough people will fall for the scam and click on the link. Just like pickpockets, cybercriminals follow the crowds, targeting the many social networking sites that increasing numbers of visitors flock to these days.

One of the problems with social engineering-based attacks is that they form a moving target. Successive scams never look quite the same, in turn making it difficult for individuals to know what's safe and what's unsafe. However, people aren't only susceptible due to a lack of awareness. Sometimes the lure of free audio or video content, or naked pictures of the latest celebrity, can entice people into clicking on a link that simply should be ignored.

Sometimes people cut corners to make their lives easier and don't understand the security implications of those actions. This is evident in password creation. More business than ever is accomplished online through shopping, banking, paying bills, professional networking and other avenues. So it's not uncommon to have 10, 20 or more online accounts, but remembering (or even choosing) a unique password for each account can be difficult. The temptation is to use the same password for each account or to use something like a child's name, spouse's name or location name that has personal significance and is therefore easy to remember. Another common approach is to recycle passwords, such as using "myname1," "myname2," "myname3" and so on for successive accounts.

Using any of these approaches increases the likelihood of a cybercriminal guessing the password. It also means that if one account is compromised, a cybercriminal may get easy access to other accounts. Unfortunately, this risk isn't obvious to nontechnical staff or to members of the general public. And even when they're made aware of the potential danger, they often don't see a feasible alternative because they can't possibly remember 10, 20 or more passwords.

So how can you create a secure password that's easy to remember but distinct from all the others you use? One solution is to use the name of the online resource as the core of your password, and then mix it up by applying the same four-step rule (or five or six, depending on your comfort level). This may involve swapping certain characters, adding numbers, mixing uppercase and lowercase characters, or adding nonalphanumeric characters. This will create a unique password that's hard to guess, but all you have to remember is the same four-step rule.

Let's show how this might work by taking three fictional online resources:

http://www.sampleshop.com
http://www.samplebank.com
http://www.mysocnet.com

We would then use "sampleshop," "samplebank" and "mysocnet" as the core of each password.

Let's use the following as our simple four-step method:
1. Capitalize the fourth character.
2. Move the second-to-last character to the front.
3. Add the numeral "1" after the second character.
4. Add a semicolon to the end.
This would result in the following passwords for each of the above accounts:
os1amPleshp
ns1amPlebak
em1ysOcnt

They're all unique and do not appear in the dictionary. They all mix uppercase and lowercase characters, numeric characters and non-alphanumeric characters. Yet all you have to do is remember is the same four-step rule each time.

An alternative solution is to start with a memorable phrase, such as:

The quick brown fox jumps over the lazy dog

Next, use the initial characters of each word to create the core of your password (in this case, "tqbfjotld"). Then apply a similar four-step rule to mix things up:
1. Capitalize the second character.
2. Add the numeral "2" after the third character.
3. Add a comma to the beginning.
4. Put the last character of the online resource at the beginning.

For the three fictional online accounts previously listed, this would result in the following passwords:
p,tQb2fjotld
k,tQb2fjotld
t,tQb2fjotld

Once again, the same four-step rule generates a unique password for each online account.

Unfortunately, cybercrime is here to stay as both a product of the Internet age and part of the overall crime landscape. So we can't hope simply to win the war, but we do need to find ways to mitigate the risks associated with going online. Clearly, legislation, law enforcement and technology all have a part to play in this risk realm. However, since many of today's cyber attacks target human fallibility, it's essential to find ways to patch these human vulnerabilities, just as we strive to secure computing devices. The use of sensible passwords is a key part of this patching process.

Tips for creating a secure password

* Include punctuation marks and/or numbers.
* Mix capital and lowercase letters.
* Include similar looking substitutions, such as the number zero for the letter "O" or "$" for the letter "S".
* Create a unique acronym.
* Include phonetic replacements, such as "Luv 2 Laf" for "Love to Laugh".

Things to avoid:
* Don't use a password that is listed as an example of how to pick a good password.
* Don't use a password that contains personal information (name, birth date, etc.).
* Don't use words or acronyms that can be found in a dictionary.
* Don't use keyboard patterns (asdf) or sequential numbers (1234).
* Don't make your password all numbers, uppercase letters or lowercase letters.
* Don't use repeating characters (aa11).

Tips for keeping your password secure:
* Never tell your password to anyone (this includes significant others, roommates, parrots, etc.).
* Never write your password down.
* Never send your password by email.
* Periodically test your current password and change it to a new one.

(Source: Google Online Security)

Authentication Comes Of Age

Continued from 23
space (NSTIC), a voluntary framework of interoperable certificates.

INSIDE NSTIC

The NSTIC framework is not another single sign on like Microsoft Passport, but rather an additional layer of authentication like the online SSL (Secure Sockets Layer) system. Whereas SSL uses a trusted third-party to authenticate the site you're trying to access, NSTIC establishes an ecosystem of identity providers, or third-party identity clearinghouses, that confirm you are who you say you are by validating individual "trustmarks" from a variety of different attribute providers — with some of whom you may already have accounts. In theory, the interoperability within NSTIC could make authentication roughly analogous to using your ATM card at different machines worldwide. You could, for example, establish certain trustmark attributes with Google, others with Verizon, and still other attributes with additional partners within the framework, then use these credentials interchangeably as needed.

For example, if you currently swipe your driver's license to gain access to a bar with your date of birth, the bar might also capture your electronically stored name, address and biometric data. Under NSTIC, that wouldn't be the case. You could choose an attribute provider that supplies only values (the birth date is March 31, 1974) or claims (the individual is older than 21). In this way, you control the relay of attributes. By selectively providing PII information, proponents of NSTIC argue there is an inherent anonymity built into the framework since no third-parties would be able to see all of your personal attributes.

While still theoretical, the use of behavioral modeling and interoperable authentication tokens may one day provide organizations with more oversight and individuals with more granular control and privacy than current authentication systems. The devil, of course, remains in the actual details.

Hacking Blackberry In 10 Seconds

Continued from 15
likely to fall before we get a chance to try.

We're in second position to target the Blackberry, though, and it seems likely that we'll get our shot. I held off booking tickets to be sure I would be ready in time. With one day to go and a reliable exploit in hand, I book tickets and I'm off to Vancouver.

At CanSecWest, there is some discussion between RIM and the Pwn2Own organizers. It turns out RIM released an updated version of the BlackBerry firmware to some of their carriers. We didn't know that there was a newer version—AT&T, Rogers and others did not list the updated version on their websites. Unfortunately, the updated version actually fixes the vulnerability I wanted to use. With one day to go I have to change the exploit to use a different, nonpublic and unpatched bug. I need to write the new exploit quickly, so I pick a vulnerability from my private collection that is similar to the original.

Instead of socializing with friends, my teammate Vincenzo Iozzo and I end up sitting in a hotel room for most of the first day to get the new exploit working. The process goes quickly between the two of us. The new vulnerability is more complex, and it is hard to get it to work as reliably as the old one. After some last-minute hacking on the floor of the hotel lobby, everything finally works, just in time for our turn at Pwn2Own.

Aaron Portnoy from TippingPoint runs our exploit against the BlackBerry from CanSecWest, and within 10 seconds we have copied the BlackBerry Messenger Contact list and photo from the phone. It is officially certified as hacked.

The inevitable press coverage for the event would have you believe that these devices instantly go down. Once you spend four weeks researching an unknown device and writing an exploit, you can indeed hack it on stage in 10 seconds.

Stuxnet Under The Microscope

Continued from 17
ability to spread via USB drives. While a slower method of propagation, hand-carried drives defeat many air-gapped defenses that may have been counted on to secure facilities, Tofino's Byres said.

"Worms don't have a single vector of attack. If you focus on a single vector, you are limiting your defense. You are taking on a Panzer with a spear," he said.

PERSISTENT PEST

Byres and his colleagues also found that Stuxnet has a superlative ability to persist in a network. While Iran claimed to have cleaned up its infection, for example, Byres said he believes the program persists in a network too well to be easily eradicated. He pointed to the level of attention directed at his site and other control-system sites as circumstantial evidence supporting his theory.

"There is no way that they cleaned up Stuxnet," Byres said. "I've tried to clean up Stuxnet… On an individual machine, it's a piece of cake. On a network, it is a living hell, because it is aggressive and it spreads in so many different ways."

Stuxnet spreads through many vectors, including peer-to-peer using remote procedure calls, which are frequently used by applications to communicate between Windows systems. The ability to jump back to just-cleaned systems makes it difficult to clean up an entire network. Moreover, the ability to hide in embedded systems such as programmable logic controllers means that even after an infection is cleaned from a compromised computer, the payload may still be active and hidden. For that reason, companies should have a robust way of detecting infected systems.

"Accept the fact that you are going to get some infections, but absolutely protect the crown jewels," Byres said. "There are many systems that, if they go down, are going to affect production, but there are a few systems that are going to result in deaths. Protect those."

Winning The War But...

Continued from 31

"advanced persistent threats" shifted from government-backed hackers in China and Russia to encompass political foes such as ThinkProgress or columnist Glenn Greenwald. Anonymous may have committed crimes that demand punishment, but it's up to the FBI to handle that, not a large U.S. bank or its attorneys.

The HBGary emails cast the shenanigans on the RSA expo floor in a new and scarier light. What other companies facing the kind of short-term financial pressure that Barr and HBGary Federal felt might also cross the line, donning the gray hat or the black one? What threat to all of our liberties does that kind of IT security firepower pose when it's put at the behest of corporations, government agencies, stealth political groups or their operatives? Bruce Schneier, our industry's Obi-Wan Kenobi, has warned about this very phenomena of how the military's ever-expanding notion of "cyber war," like the Bush era's War on Terror, does little to promote security but a lot to promote inchoate fear. That inchoate fear then becomes a justification for further infringement on our liberties.

"We reinforce the notion that we're helpless — What person or organization can defend itself in a war? — and others need to protect us. We invite the military to take over security, and to ignore the limits on power that often get jettisoned during wartime," Schneier observed. That kind of conflation is clear reading Barr's emails, where the line blurs between sales-oriented tactics and offensive actions. The security industry veterans I spoke with at this year's show were as aghast at Barr's trip far off reservation, but they also expressed a weary recognition that, in the security business, this is where things are headed.

What's the alternative? Schneier notes that focusing on cybercrime as crime, rather than war, tends to avoid the problems with demagoguery. Focus on cybercrime and hacking in the same way that you focus on other types of crimes: As long-term problems that must be managed within the context of normal life, rather than wars that pose an existential threat to those involved and must be won at all costs. The United States needs peacetime cybersecurity "administered within the myriad structure of public and private security institutions we already have" rather than extra-judicial vigilantism and covert ops of the kind the HBGary emails reveal. Here's hoping HBGary is the wake-up call the industry needed to reverse course.

Seven Recommendations For A Safer Facebook

Continued from 11

start for security education, but it's not enough. Unfortunately, the general level of user awareness when it comes to security and privacy online is sub-optimal. Educating computer users is a task for the whole IT industry, not just security companies.

The problems with social networks is that they struggle to find a balance point between usability and security, as you can't have both at the same time -- not in this world at least. They need their websites to have state of the art usability, and security features will always come in the way of that.

At the same time, no technology can guarantee 100 percent security, as there will always be the human weakness. These recommendations are not to be seen as a silver bullet for securing Facebook, but as seven realistic and doable steps that can dramatically increase the safety and privacy of all Facebook users.

On the users' side, I'm always giving this advice to anyone who asks me about privacy and social networks: as long as you have a social networking account, make sure you operate under the assumption that sooner or later, the things you do online can be seen by anyone. Expect the best, but think of the worst. Don't upload a picture, don't post a link or a comment unless you are prepared to take responsibility for your actions.

I know it might be hard to decide, but if in doubt, just don't do it. Don't do it unless it's something that you're ready to share with any person from your past, present or future – or even more. Be honest to yourself first and you won't have any problems. I think it's called common sense.

Online Shopping Carts

Continued from 13

WordPress and Joomla that are "annoyingly running on stupidly old and vulnerable versions." Emails are sent to the hacked site's owner and hosting company as soon as the compromises are discovered. If Burn doesn't receive a reply within a week, he contacts the Web site owner directly by phone.

"Thankfully, those with the better hosting companies tend to have their sites cleaned for them, which saves a world of hassle," Burn said. "I've seen some cleaned within the hour and others still not cleaned several months later. Still other cases are still open around a year after initial discovery."

WhiteHat Security's Grossman said he hopes to start seeing more website owners migrating to hosting providers that specialize in keeping customer software up to date with the latest security enhancements.

"It's a lot easier to keep sites from getting hacked than it is to clean them up afterward, because the attackers usually are installing backdoors and rootkits on the sites they hit. The type of model where hosting providers manage security for their clients is gaining ground, because in so many cases it never gets taken care of otherwise," Grossman said.

While he acknowledges that such full-featured hosting services are more expensive than the average monthly hosting plan, Grossman said he hopes the industry gains enough ground to become more competitive on price. "I hope we'll see a whole new generation of hosting providers that will differentiate themselves on security," Grossman said.

Until then, we repeat our cautionary advice: You get what you pay for.

Tales From An Emergency Response Team

Q&A

Alexey Polyakov

In this interview, Kaspersky Lab's Alexey Polyakov talks about the work of the company's Global Emergency Reponse Team (GERT), trends in malware infections and some of the most common problems found in the typical corporate environment.

Can you describe the work of the Global Emergency Response Team (GERT)?

The Global Emergency Response Team was created to help Kaspersky Lab corporate customers to effectively mitigate virus-related incidents, share our knowledge and expertise with corporate IT, and perform post-mortem and forensic analysis.

The Global Emergency Response Team consists of few worldwide distributed locations: one in Moscow, Russia, and another one in Seattle, USA. We provide 24x5 service to our existing corporate customers in Eastern Europe, European Union, U.K., and North and South Americas. All members of our team are malware analysts with solid experience and able to identify malware presence and necessary mitigation steps with minimal information from the infected users. We also have homegrown tools that help us effectively fight various malware incidents.

What does your job entail?

Typically, our corporate customers call the first line on corporate support when they experience a possible malware outbreak. The role of corporate support is to investigate and resolve nonvirus-related cases. However, they escalate to GERT if the case involves malware or requires specific virus-related knowledge. Our experts are ready to receive and process malware-related incidents 24x5 around the world. We use a set of tools that help us with identifying and mitigating malware remotely. About 90 precent of incidents are resolved within 24 hours or less.

What types of security threats are you encountering in a typical corporate environment?

We see different malware attacks specific for every region we serve. There is much malware that targets banks and financial institutions in Brazil. In the U.S., we see frequent outbreaks with the Sality virus, Kido worm, and various fake antivirus and rogue security tools. Fake antivirus is also popular in Europe. In Eastern Europe and Russia, we also see lots of SMS thefts. Beside just software malware, we have encountered a few cases with firmware. There was a case with Internet router hacking.

Within the last 12 months of active engagement with our corporate users, we noticed that the majority of virus-related incidents happen due to underestimated design issues or unnoticed weaknesses in security solutions of corporate security policy. Here are a few examples:

Partially protected network environment. AV solution installed only on some part of the network

Multiple-vendor antivirus protection. Network segments are protected by different AV vendor products. Not all AV vendors can catch malware early enough. This means that part of the network will be constantly attacking other systems

Missing security updates or outdated AV signature versions. Update schedule either not configured or configured to install updates too infrequently

Configuration issues with network shares. Typically, no access control, full access to everyone, or unnecessary write permissions to a wide group

Excluding nonexecutable files from regular system scans. Nonexecutable files can also contain threats.

Based on your GERT data, how are these infections happening?

Let me give a recent example. You probably know that malware might take advantage of missing security patches. This can happen when either a schedule is not in the place or the update frequency is too infrequent. We have seen that some corporate customers still do not pay attention to updating their systems with urgent updates.

A good example is the well-known Windows OS vulnerability MS10-046 that was fixed by Microsoft on August 2, 2010. This vulnerability was used by at least two threats: the Stuxnet worm and Sality, an aggressive file infector. Stuxnet might infect end-user systems when someone inserts an infected USB memory card. However, the Sality virus used to propagate via open network shares, incoming infected files or infected removable media like USB sticks. Now we can see that the new Sality modification uses the new ".lnk" vulnerability as another opportunity method of propagation. We have

observed it aggressively spreading through this vector.

This Sality variant consists of several files. One of them is aturun.inf. This file is auto executed when new removable media is inserted into a computer. It contains a link to an infected .dll located on remote host. The remote host is an infected system in the organization network that was previously infected by the Sality virus, or it can be a remote host outside of your organization network, like a public file exchange. Although Microsoft released a patch long ago, we still see signs of infection in the wild.

And there is another good example that is related to IT security policy. This is the AV scanner configuration for email attachments on portable media, like external USB devices. In some cases, a scan of nonexecutable files can be disabled for performance reasons. However, some malware can be written as nonexecutable files targeting particular products file formats, e.g., AutoCAD files.

Let's assume there is a midsized company called Project Designer that has offshore developers. The remote team works on the design of a particular project component and sends AutoCAD drawings weekly to the headquarters team. Then, HQ engineers copy the drawings into a common folder for shared use. Let's also assume that HQ engineers always use a corporate AV product to scan USB memory cards and email attachments received from offshore teams. The AV product is configured by the company's security administrator via the management console. For performance reasons, the AV scan configuration excludes all nonexecutable files or files bigger than particular size. The email scanning can also be configured to skip big files as well as nonexecutable files.

The attacker can create a malicious component that will run when AutoCAD loads an autolist script. When executed, this script might create other malicious components, automatically start up services, download updates from the Web or propagate to open network shares. A good example is Trojan.Acad.Dwgun. It as arrives as a dwg file, which appears to be a drawing file for AutoCAD. Its malicious function activates when someone opens the file. The attacker can craft a file of a specific file size to avoid AV scanning.

Depending on payload function, the

attack might erase all project files from the file system or copy project files to external locations. It can also spread to other servers that can provide shared resources.

What's the bigger point of weakness: people or technology?

It looks like both sides must know the rules and not allow malware to propagate in the network. We have cases where IT security policy was not configured properly or when malware was used to employ zero-day exploits.

Recently there was a malware outbreak in an organization when someone executed an email attachment. The person did not really know that the attachment was infected and blindly followed this social engineering attack. Once the attachment is executed, it can take advantage of a software vulnerability. We have seen this many times in the past year in Office documents, PDF files, Java scripts and other objects.

On the other hand, we also see cases when known system vulnerabilities are not patched in the organization or patched quite late. There are cases when IT security policies miss an insignificant piece, but later that piece may become a really dangerous security policy threat—for example, the case with remote file exchange or accepting files from other regions without scanning.

From our experience, I would say that about 90 percent of virus outbreaks happen due to the following factors:

Wrong file share configuration (allow autorun, write/execute access to everyone, store executables files with permission to execute, having blank password for accessing fileshares)

Successful social engineering attacks (executing email attachments without scanning, browsing Internet from corporate mission-critical systems, using unscanned USB devices, installing unverified software downloaded from the Web)

Weak IT security policy that allows threats described earlier in this article

Do you think desktop virtualization will lead to better security?

Well, virtualization was initially proposed as a good way to improve performance of individual boxes—to be able to run all sorts of possible tasks with minimal hardware resources. Virtualization's main driver is cost efficiency, not security. However, virtualization provides another way of securing corporate environments. It has to be protected the same way as regular physical systems, with proper security patch policies, AV solutions and so on.

It looks promising that you can put all sorts of applications on just a single box and change virtual tasks at any moment to do what you need. But what happens when a session is infected? If the infected virtual session does not have important data, it is better to discard it and recover from a repository. The downloaded data can be saved somewhere else, e.g., on the remote media. However, when a new instance of a virtual session is recovered, it might need the downloaded data from the previous session. What if this downloaded data is infected? Will it infect the new virtual session or, if data is shared between other virtual sessions, will it infect all of them?

Another interesting design advantage of virtualization is the host OS. Unlike the guest OS, the host OS is protected from external unauthorized access. The security solution can be installed just on the host OS and verify all guest OSes before or while they load. This would probably be best from a performance angle. However, there is just a small problem: How do you install patches on the host OS and how do you update signatures if the AV solution is in the host OS? Since the network operations are done from the host OS, it can be also potentially vulnerable to typical network attacks.

In summary, virtualization is a great addition to a good IT security policy, but virtualization does not make a corporate network more or less secure. It is better to follow the same security guidance we discussed above to protect the guest and host OS.

BY WOLFGANG KANDEK

Wolfgang is chief technology officer at Qualys. He has over 20 years of experience in developing and managing information systems. His focus has been on Unix-based server architectures and application delivery through the Internet.

Web Browsers: The Inconvenient Truth

The security of Web browsers should be a primary area of focus for computer end users and IT administrators alike. The Web browser, in its variants of Internet Explorer, Firefox, Chrome, Safari, Opera and others, is the most-used program on modern desktop and laptop computers. Computer owners access their browsers to use applications, update their social network statuses, get the news, play games, read email, listen to music, watch movies, edit documents and perform a plethora of other activities. In order to support the advanced media types that are necessary for these uses, browsers support the installation of extensions called plug-ins.

Attackers have long adapted their operations to this model, and as a result, they target the majority of their attacks on vulnerabilities in the browser and its installed plug-ins. Attackers aim to install a new piece of software (malware) on users' computers with the ultimate goal of gaining complete remote control. Once in control of a computer, attackers can search for valuable information, monitor the computer for interesting transactions, and use the computer to send spam and participate in distributed DDoS (distributed denial of service) attacks.

Even though the browser plays a primary role in our Internet use, statistics gathered here at Qualys show that the security of Web browsers is in dire shape. We base our assessment on the results of a free browser security evaluation service for computer end users called BrowserCheck (browsercheck.qualys.com). Our data for the last six months of 2010 encompasses 200,000 samples and indicates that 75 percent (the red line in the following chart) of all users who visit BrowserCheck have vulnerabilities that allow attackers to take control of their machines. About a quarter of these vulnerabilities are attributable directly to the browser (blue line), while three quarters of the vulnerabilities (the delta between the red and blue lines) are caused by outdated browser plug-ins.

The worst offenders in the plug-in space are Oracle's Java and Adobe's Reader and Flash. These plug-ins are installed on more than 80 percent of all tested computers and are frequently so outdated that they contain exploitable vulnerabilities. As such, they represent an irresistible target for malicious attackers.

For end users, the recommendation is clear: Keep your computer updated. Unfortunately, this is a complex undertaking because each software vendor (and sometimes even its product line) has a proprietary update mechanism. Among the most popular plug-ins listed in the chart, we found separate update mechanisms for the browser itself, Flash, Reader, Java, Silverlight, QuickTime and Shockwave. Only Microsoft's Windows Media Player gets updated jointly with the Windows operating system and the company's Internet Explorer browser.

IT administrators have additional challenges to overcome. Frequently, a fear of introducing incompatibilities into the corporate IT environment causes companies to mandate the use of outdated versions of software, including Web browsers. As an example, we often see usage numbers of Internet Explorer 6 topping 30 percent in the corporate space, whereas overall usage of this outdated browser is roughly 12 percent (according to www.ie6countdown.com) and virtually nonexistent in our BrowserCheck data, with less than 1 percent.

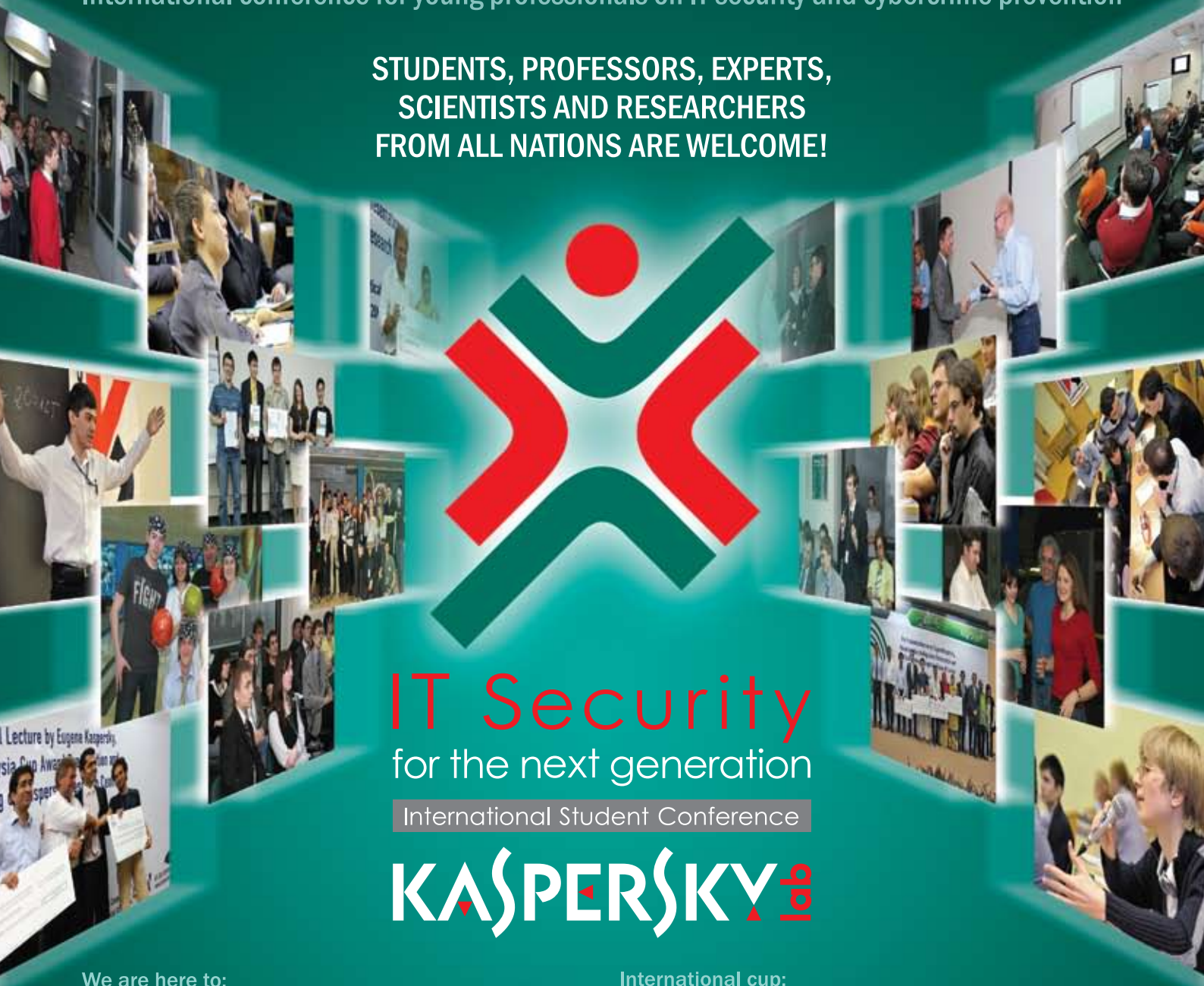
In the security industry, our challenge is to start working together with software vendors. We need to devise a universal update mechanism that works for all software components. Google's Chrome browser has started to break important ground here by introducing an automatic update program that is used for both the browser and Adobe Flash. Google Chrome users now receive the newest version of Flash automatically and often a week or more before other browser users can access it.

However, the drive to make security vendors and software vendors work together ultimately must start with the user community. So make sure your voice is heard: Contact your vendor representatives and talk to them about your problems and concerns. Tell them that you want software that updates itself automatically and has its security settings turned on by default.

«IT SECURITY FOR THE NEXT GENERATION»

International conference for young professionals on IT security and cybercrime prevention

STUDENTS, PROFESSORS, EXPERTS, SCIENTISTS AND RESEARCHERS FROM ALL NATIONS ARE WELCOME!



IT Security
for the next generation

International Student Conference

KASPERSKY lab

We are here to:

- Discuss the hottest topics in the IT security industry
- Share knowledge, ideas and experience
- Develop innovative new ways of thinking
- Give expert workshops, advice and support
- Encourage professional growth and skills development
- Provide exciting career opportunities
- Create an international community for young professionals

For more info visit:

http://www.kaspersky.com/educational-events
http://www.facebook.com/KasperskyConference

International cup:
14-15 APRIL 2011 (TUM, Munich, Germany)

Russia and the CIS cup:
10-11 MARCH 2011 (MSU, Moscow, Russia)
Countries: Russia, Moldova, Ukraine, Belorussia, Armenia, Azerbaijan, Turkmenistan, Uzbekistan, Kazakhstan, Kirgizia, Tajikistan.

Asia – Pacific, MEA and Africa cup:
4-6 March 2011
(MARA University, Kuala Lumpur, Malaysia)
Countries of Asia – Pacific, MEA and Africa.

European cup:
28-30 JANUARY 2011
(University of Applied Sciences, Erfurt, Germany)
All European countries plus Turkey and Israel.

JOIN US AND BECOME AN ACTIVE MEMBER OF THE EXPERT COMMUNITY!

WE

- Interview the company's leading experts
- Discuss the hottest topics around
- Educate users about cyber security
- Bring you the latest technology news

Watch other Kaspersky Lab webcasts on our YouTube channels:

YouTube http://www.youtube.com/Securelist

YouTube http://www.youtube.com/Kaspersky

KASPERSKY lab