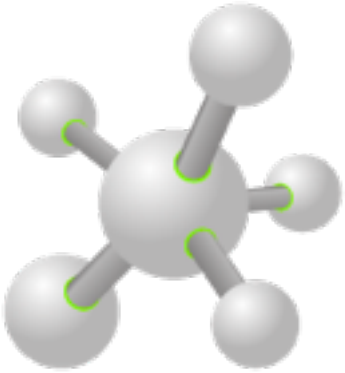
info@thinkst.com
research@thinkst.com
http://www.thinkst.com

ThinkstScapes Quarterly Report 2014 / Q1



ThinkstScapes Quarterly

2014 / Q1

Themes in this edition

- *Threat Intelligence*
- *Hardware attacks*
- *Defensive options*
- *Liability and insurance*

Introduction

WELCOME to ThinkstScapes Quarterly for the first quarter of 2014. Compared to 2013 Q1, this quarter saw an increase in the number of events and their length: an extra 218 talks but also a significant jump in the number of conference days (an almost 100% increase). This quarter is typically light in terms of content, with major conferences including RSA, Shmoocon and CanSecWest.

The themes in the quarter include the persistent hardware attacks which we've seen for some time, a raft of defensive options for network owners, and the introduction of threat intelligence for the first time. Threat intelligence is riding the hype wave and it remains to be seen whether it goes the same way as NAC; however indications at this point are that it can yield useful information for securing networks and we are cautiously optimistic. Software liability and insurance are potential alternatives to compliance and get a cursory glance in this edition.

We hope you enjoy this edition of ThinkstScapes Quarterly!

The explosion of security events worldwide means that industry participants are increasingly swamped by speakers vying for our attention. The ThinkstScapes Quarterly Edition focuses on identifying key research and trends curated from conferences held in the previous 3-month period.

Summary

Conference overview

Q1 saw 40 events covering 1318 talks spread over 95 conference days.

Table 1 below shows the 40 events in calendar format.

Table 1: Conferences 2014/1—2014/3

Start Date	End Date	Conference
2014-01-14	2014-01-16	Flocon
2014-01-14	2014-01-17	S4
2014-01-17	2014-01-19	Shmoocon
2014-01-19	2014-01-19	Suits and Spooks
2014-01-20	2014-01-20	BSidesColumbus
2014-01-23	2014-01-23	PCILondon
2014-01-27	2014-01-28	OWASPAppSecCalifornia
2014-01-30	2014-01-30	e-crime
2014-02-14	2014-02-15	Nullcon
2014-02-16	2014-02-17	ASIS
2014-02-17	2014-02-20	Fast
2014-02-23	2014-02-24	BSidesSF
2014-02-23	2014-02-26	NDSS
2014-02-24	2014-02-28	RSA
2014-02-26	2014-02-28	ESSoS
2014-02-27	2014-02-28	ICCNSS
2014-02-27	2014-02-27	Trustycon
2014-02-28	2014-02-28	Metricon9
2014-03-01	2014-03-02	VEE
2014-03-03	2014-03-05	CODASPY
2014-03-03	2014-03-07	FinanCryptandInfoSec
2014-03-03	2014-03-05	FSE
2014-03-06	2014-03-08	RootedCon
2014-03-07	2014-03-09	DakotaCon
2014-03-07	2014-03-09	Eth0
2014-03-08	2014-03-08	CrikeyCon
2014-03-08	2014-03-08	DefconKerala
2014-03-10	2014-03-11	BSidesVancouver
2014-03-10	2014-03-11	EUSmartGridCyberSecurity
2014-03-12	2014-03-14	CanSecWest
2014-03-13	2014-03-14	ASIS Internaional
2014-03-17	2014-03-20	AppSecAsiaPac
2014-03-17	2014-03-19	IFIP
2014-03-19	2014-03-20	CartessAsia
2014-03-19	2014-03-20	Troopers
2014-03-20	2014-03-21	BSIDESAustin
2014-03-22	2014-03-22	BSidesSLC
2014-03-25	2014-03-28	Blackhat Asia

Talks Overview

From these 40 conferences, 22 presentations were selected for inclusion. The number of selected talks is not fixed between different editions of ThinkstScapes; rather, the selection represents a mixture of talks that are required reading combined with work that is notable, comprehensive or provides an alternative viewpoint to previously recommended pieces of work.

In Table 2, the chosen 22 talks are listed along with their authors. Each title links directly to the summary in this report, and one can return to this summary table at any time by following the “[talk index]” link that appears at the bottom of every summary entry in this report.

Table 2: Selected Talks from 2014/1—2014/3

Title	Authors
Introductory work, reviews and security theory	
Collaboration across the Threat Intelligence Landscape	Merike Kaeo
Data Breach Resolution for Insurance Carriers	Paul Paray
Microsoft Vulnerability Research: How to be a finder as a vendor	Jeremy Brown, David Seidman
New Frontiers in Cryptography	Dan Boneh, Chris Palmer
Scan all the things – Project Sonar	Mark Schloesser
Software Liability?: The Worst Possible Idea (Except for all Others)	Jake Kouns, Josh Corman
Attacks	
Building Trojan hardware at home	JP Dunning
Bypassing EMET 4.1	Jared Demott
Fuzzing the easy way, using Zulu	Andy Davis
Harvard Architecture Exploitation – Coming to a Smart Grid SoC Near You!	Nathan Keltner, Josh Thomas
Owning a building: Exploiting Access Control and Facility Management Systems	Billy Rios
Project Robus: Master Serial Killer	Adam Crain, Chris Sistrunk
Power Attack: An Increasing Threat to Data Centers	Haining Wang, Zhang Xu, Zichen Xu, Xiaorui Wang
The Sniper Attack: Anonymously Deanonymizing and Disabling the Tor Network	Aaron Johnson, Rob Jansen, Florian Tschorsch, Björn Scheuermann
USB attacks need physical access right? Not any more...	Andy Davis
Defense	
Argus with Netmap : Monitoring Traffic at 10Gbits/s Line Rate Using Commodity Hardware	Harika Tandra
Fix What Matters: Why Using CVSS for Remediation is Nuts	Michael Roytman
Hiding the breadcrumbs: Anti-forensics on SAP systems	Will Vandevanter, Juan Perez-Etchegoyen
Honeywords: A New Tool for Protection from Password Database Breach	Ronald Rivest, Kevin Bowers
Raising Costs for Your Attackers Instead of Your CFO	Aaron Beuhring, Kyle Salous
Writing Secure Software Is Hard, but at Least Add Mitigations!	Simon Femerling

Themes

Threat Intelligence

Unsurprisingly threat intelligence was a major theme in Q1. Like most buzzwords its exact definition depends on who you ask, but the general idea that keeping track of security incidents and attacker capabilities in order to inform defensive decisions seems to be agreed upon. We include Kaeo's talk on sharing threat intelligence and Roytman's discussion on prioritising remediation based on factors other than CVSS scores. There were many more talks on threat intelligence, mostly from vendors looking to sell re-branded appliances with "Threat Intelligence" added to their brochures.

Hardware attacks

This theme has been present in various guises over a number of recent editions. Attacks on hardware are proliferating: Dunning shows how to Trojan hardware at home, Keltner and Thomas talk about attacking Harvard architecture devices, Rios attacks building management systems, Crain and Sistrunk reveal SCADA bugs, Wang *et al* present power DoS attacks and Davis hammers USB drivers.

Defensive options

Defense talks constitute just under a third of the work included in this edition, which is higher than usual. The defensive options are actionable and useful for operators and administrators; defense talks tend to be concrete and practical. Tandra speaks on high-speed network monitoring, Vandevanter and Etchegoyen discuss SAP forensics, Rivest and Bowers present a way to detect password breaches, Beuhring and Salous argue for application whitelisting using tools likely already installed in your organization and Femerling releases a tool to quickly scan binaries for obvious security gaps.

Liability and insurance

Compliance has not worked to secure organizations (Target's PCI compliance being proof by example), and there are growing calls for alternatives. In this edition we highlight two approaches, software liability and cyber insurance, by Paray and Corman respectively.

Comments

RSA

The RSA Conference is the largest security event by far and we have records of 380 talks at the 2014 US conference. It was also the subject of a boycott in relation to the revelations that RSA included a defective random number generator in the BSafe library, ostensibly an intentional weakening of the library paid for by the NSA. However the boycott did not appear to have much effect and the most visible outcome was a new conference called TrustyCon, whose speakers spoke on privacy, surveillance and cryptography. It remains to be seen whether the protest conference was merely a reaction to the revelations, or whether it has the legs to return next year.

Recommended talks

Noted industry talk:

> Jared Demott, "Bypassing EMET 4.1".

Applying defensive thinking beyond common metrics:

> Michael Roytman, "Fix What Matters: Why Using CVSS for Remediation is Nuts".

Administrivia

The contact address for ThinkstScapes is thinkstscapes@thinkst.com; general queries can be sent to info@thinkst.com.

Summaries of selected work

Introductory work, reviews, surveys and security theory

Collaboration across the Threat Intelligence Landscape [slides]

Merike Kaeo

With threat intelligence gaining momentum, an underpinning of this new movement is access to comparable data. For individual organization s this means both a source of data as well as a common format or syntax to make comparisons.

Getting into sharing

This talk provided an overview of current efforts to share security data, covering efforts to enumerate security attributes (e.g. MITRE's C* projects), IETF standards relevant to collaboration, NIST documents, taxonomies and frameworks such as IODEF and Veris, and transports to carry the data (like TAXII and RID).

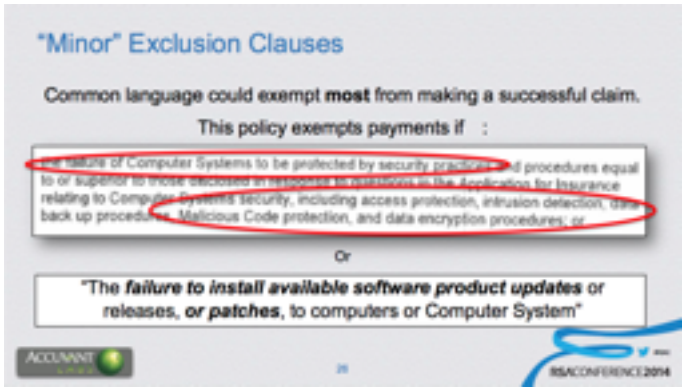
Takeaway: Sharing intelligence is key and the speaker argues that attackers share intelligence all the time, it is silo'd defenders who do not share that suffer due to their isolation. In our experience organization s are generally cautious about sharing their intelligence and data. The speaker states that sharing is a necessity and should be implemented.

[talk index]

Data Breach Resolution for Insurance Carriers [slides]

Paul Paray

Cyber insurance is seeing increased interest, and this talk provided an introduction into the nitty gritty of those offerings.



How to not pay out data breach insurance

Compliance regimes attract richly deserved flak, and insurance is supposedly a market-based alternative to compliance. A breach incurs costs directly in the form of lawyer fees, consultants, PR, fines and mandatory monitoring/notification. These costs are real and demonstrable; compliance can help reduce some of these (e.g. the likelihood of fines), but other costs cannot be fixed through compliance. Insurance can fill that gap.

When a cyber insurance policy is underwritten, the underwriter must assess the policy holder's risk level to determine their premium. However, this assessment is typically questionnaire-based and the policy lasts years, even if your risk profile changes. This aspect is one of the hairy areas of cyber insurance and has not been resolved.

As for claims, the terms in many policies will apparently make it difficult to receive compensation. The speakers cite an example in which an insurer will not payout claims if the compromised systems were missing patches, which surely rules out most environments. The fine print matters.

Takeaway: The cyber insurance field is still new and growing, and the industry is feeling its way into fresh territory.

[talk index]

Microsoft Vulnerability Research: How to be a finder as a vendor [slides]

Jeremy Brown, David Seidman

Short description of MSVR

Bug bounty programs are all the rage, whether they be homegrown as in the case of Google, Microsoft, Mozilla and others, or externally managed by teams like Bugcrowd or HackerOne. However not many organization s have the size and focus to coordinate security issues found by their *own* employees in third party software. Microsoft is one of the few exceptions to this; the Microsoft Vulnerability Research program (MSVR) supports Microsoft employees who find bugs in external software.

The talk walks through the underpinnings of the program and explains the 7-step process by which bugs are reported to external vendors. The process is essentially mandatory for all bugs found on company time, and optional for bugs found in personal time. The program relies on a security contact database to report issues to the correct people at the third party, a support group who help the bug-finders by liaising with vendors in a consistent and transparent manner, and ensuring that the reported issues are valid and verifiable before contacting the vendor.

The benefit to such a program is that bug reports emanating from Microsoft will have been reviewed and examined, and are likely to of good quality. (It also has the benefit of providing Microsoft with a cudgel to swing at other large vendors whose employees take aim at MS products and drop 0day, such as Google.)

Disclosure policies have been around for sometime, and security consultancies will usually have a stated policy, however the MSVR is more than a disclosure policy. Microsoft's sheer size almost necessitate such a model: bug reports being issued by random application groups could vary wildly in quality and excitement and quickly cause reputational damage to Microsoft.

Takeaway: The basics of the MSVR program are not cumbersome even for mid-sized organization s. A disclosure policy combined with reviews of bugs before they go out and a fixed point of contact are the essence of the program and worthwhile duplicating.

New Frontiers in Cryptography [video]

Dan Boneh, Chris Palmer

Applications of obfuscated cryptographic programs

Crypto has come under intense scrutiny in recent months with the various OpenSSL issues (more this week), combined with the Snowden revelations about the ability of nation states to monitor both cleartext and certain enciphered communications.

Crypto starts out in extremely academic and theoretical terms, and eventually some of it makes its way into products used by endusers. This talk consisted of two discussions on the leading edge of applied cryptography, which are soon to impact those endusers.

The first speaker spoke on options for improving the x509 PKI. Possible options include having CAs rely on DNS to determine if they should issue certificates, certificate pinning (which is seeing increased adoption), extensions to TLS, and a public log similar to Bitcoin in which all certificate signings are published. Each is an extension and improvement on the current PKI, but each also comes with drawbacks. What's apparent is that changes are needed, but no one route has yet won out.

The second speaker spoke on obfuscated cryptographic programs. A recent advance in theoretical cryptography is a technique for embedding a key in a program which is provably not extractable. Although very exciting this comes with the caveat that the programs are extremely slow and very large. The hope is that with time the technique can be optimized to the point where cryptographic program obfuscation become practical, making many crypto problems trivially solvable. The magic crypto dust might actually arrive!

Takeaway: The past year has seen renewed interest in the application of cryptography, and this talk showed what the next couple of years might hold.

[talk index]

Scan all the things – Project Sonar

[slides]

Mark Schloesser

Internet-wide scanning has been around for a while but its cost has dropped in recent times as bandwidth and storage prices have plummeted. Global lists of vulnerable systems are known to be offered by government contractors such as End Game Systems, but now the same information is within reach of researchers.

Internet-scale is quite a bit of data

Last year in ThinkstScapes we included ZMap, a high speed port scanner. In this edition, we came across Project Sonar which is an effort to provide Open Source data on Internet-exposed systems. It relies on ZMap, Masscan (another high speed port scanner) and other tools to produce the raw data, which is then saved, processed and examined.

Current analysis mimics other efforts such as Shodan and the EFF SSL Observatory, but includes efforts to highlight common consumer hardware when narrowing vulnerability hunting efforts and also tracking defacements. Future plans are to better its publication, analysis and reporting interfaces.

Right now, one can download raw data for scans of all IPv4 hosts on port 80 and 443, including SSL certificate dumps, as well as scans on popular ports, from https://scans.io. Some of the data is extremely fresh (i.e. weeks old at the most).

Takeaway: In one sense Internet-scale refers not to the expansion of scope, but the reduction of complex matters until they are within the grasp of small players. Internet-wide scanning data is trivially accessible to individuals now, and defensive strategy must take that into account. At the very least, companies should be examining this data to determine their exposure therein.

[talk index]

Software Liability?: The Worst Possible Idea (Except for all Others)

[slides]

Jake Kouns, Josh Corman

Software vendors have avoided assuming the liability for defects in their products through the use of EULAs, and the speakers believe this should and will change.

Why liability is coming

They list arguments often given against software liability: stifling of innovation, more barriers to entry, impact on the broader economy and hurting vendors.

However they argue that in the US there is already product liability for goods that place life or limb in peril, which cannot be contracted away. This could be immediately applicable where software forms a piece of a larger product (e.g. vehicles). There is also liability for data breaches which, while not directly software liability, is certainly related. One way this might be extended is for small companies who suffer financially in a data breach to recoup costs from their software providers.

In the end, they do not provide a firm direction for where software liability is moving but they are confident that liability is finally coming to the software world. Their key message is that we need to get the incentives for liability right, otherwise the emphasis is placed on the inputs rather than the outcomes; that road leads to regulation-driven compliances regimes such as PCI DSS.

Instead, liability regimes should not be prescriptive on what needs to be done while still allowing for liability. They argue it should extend to intangibles and not be limited to instances of physical harm.

Takeaway: The regulatory regime around the security of software is going through a period of upheaval. Regulation of vulnerability and exploit sales is under discussion, as is liability for loss and insecure practices. This talk sheds light on the potential for software liability.

[talk index]

Attacks

Building Trojan hardware at home

[slides]

JP Dunning

Troy would be safe, but not your desktop

The release of the NSA's capability menu in December last year showed what type of surveillance is possible with nation state resources. The raft of commercial hardware keystroke loggers show that those simple technologies are equally accessible for consumers.

This talk shows how easy it is to build custom surveillance gear and embed it in legitimate looking hardware. By utilizing an open source hardware platform called Glitch, the speaker demonstrated opening up various devices (including a mouse, a keyboard, a point-of-sale terminal, and a desktop computer), and wiring the Glitch board into the USB lines. Conceptually, there is nothing new about this except that it shows that previously specialized hardware attacks such as modifying target hardware is now performed without special hardware. Instead of searching of keystroke loggers by inspecting the USB ports, opening up the hardware could soon become standard.

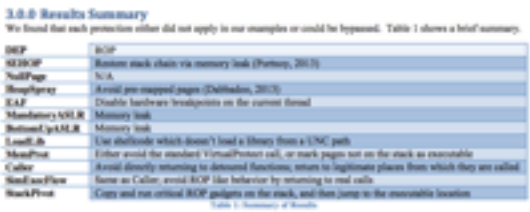
Takeaway: The takeaway here is the commoditization of surveillance gear, to the point where even if nation states are not in your threat model, these attacks should be.

[talk index]

Bypassing EMET 4.1

[slides]

Jared Demott



EMET defences with their bypasses

We have mentioned EMET on numerous occasions as an important line of defense in large scale Windows deployments. EMET is a collection of runtime checks which can be enabled on 32-bit software running on Windows, and aims to protect against 0day threats by detecting code that looks like shellcode.

We also covered the Bluehat prize in which Microsoft rewarded researchers for contributing new ideas towards EMET checks; the speaker behind this talk won third prize in that competition and was apparently of the belief that the defences chosen over his were insufficient. So he set out to prove that very thing through an examination of the latest EMET edition.

The paper provides a highly accessible description of the defences included in EMET, and he breaks down the process by which each can be bypassed. The cumulative effect is a working exploit which does not trigger any of the twelve EMET checks.

Takeaway: In truth, it should come as no surprise that EMET can be bypassed. Microsoft do not pitch it as an ironclad defense and there has been previous work on bypassing EMET. Instead, the takeaway is to set expectations: EMET does not defend against attackers willing to customize exploits but the level of knowledge required to do so cuts out a large number of novice exploit writers. At the bottom-end of the market, EMET detects and blocks Metasploit-quality exploits.

[talk index]

Fuzzing the easy way, using Zulu

[slides]

Andy Davis

Fuzzing usually has large setup costs. The slowest option is to write a custom fuzzer which inevitably starts resembling one of the public frameworks such as Peach or Sulley. Even if one of those frameworks is chosen in the beginning, the fuzzer still requires manual configuration that can take hours at the minimum.

This talk introduced a fuzzer called Zulu whose aim is to quicken the setup of fuzzer. It does so primarily by configuring through a GUI rather than text files or code. In addition, the fuzzer supports a range of input options to capture the data which is to be fuzzed, including serial ports, USB data, file data, and network traffic via PCAP files.

Zulu successes

Zulu supports integration with VMWare to monitor whether the fuzzer has had an effect.

Takeaway: The contribution of Zulu is to lower the bar for fuzzing. The code has been open sourced, and even more researchers will be able to fuzz applications.

[talk index]

Harvard Architecture Exploitation – Coming to a Smart Grid SoC Near You!

[video]

Nathan Keltner, Josh Thomas

In planning future defences against memory corruption, one solution that is often proposed is a fundamental shift in the basic design of memory space that would make overwriting code virtually impossible. Current architectures such as x86 or x64 mix code and data in a single memory space, which underpins the majority of memory corruption exploitation. This is inherent to the Von Neumann architecture which is present in almost every personal computing device.

Memory map on a Harvard device

It is not, however, the only architecture. The Harvard architecture relies on multiple separate types of memory, some of which cannot be written into. In a perfect world a true Harvard architecture stores code solely in read-only memory so memory corruption bugs cannot impact execution flow, but in practice this is not the case. Harvard architectures are found in SoCs such as the Teridian, which is widespread in smart meters. The first half of this talk provided background on the Harvard architecture and low-level exploitation hints when dealing with Flash memory.

The second half of the talk focused on the challenges that are unique to exploitation on Harvard architectures. While simple examples were absent, what was apparent was that even with a fundamental shift in architecture, there are still vectors available to attackers. Examples include the update mechanism as well as abuse of software timers.

Takeaway: Harvard architectures are by no means a silver bullet; while they prevent entire classes of bugs they are not, strictly speaking, security designs.

[talk index]

Owning a building: Exploiting Access Control and Facility Management Systems

[slides]

Billy Rios

With all the SCADA and hardware hacking talks in recent times, it is a little surprising that actual security hardware and devices do not get as much attention as they should. The hotel room lock hacking from 2012 touched on this, but the principle is broader than simply locks.

In this talk, the speaker shows attacks on building management systems which include locks and access control, but also important functions such as video feeds, HVAC, alarms, lighting, energy and billing.

Searching for buildings on the Internet

He found vulnerabilities in two vendor products, namely Niagara Framework and MetaSys. The vulnerabilities were quite standard (remote code execution issues, no authentication, file retrieval and so on), but its their impact which is notable. They expose their interfaces to the network, in some cases via web services, which makes remote attack simple.

Even more concerning is that an scan for these systems on the broader Internet found 50,000 buildings exposed.

Takeaway: His recommendation is to speak to your facilities people and determine whether any of these systems are used: Tridium – Niagara, Johnson Controls – MetaSys, Automated Logic – WebCTRL, or Delta Controls – eneliWEB. The presentation also lists access control and surveillance systems that could be network-exposed.

[talk index]

Project Robus: Master Serial Killer

[video]

Adam Crain, Chris Sistrunk

Superficially this talk presented the background that led to the release of 15 SCADA advisories by a two-man team against SCADA devices that implement DNP3, a network protocol for data exchange between components in a smart grid. The details of the vulnerabilities are not notable as they are standard memory corruption problems and exploits were not provided. Instead, our interest was piqued due to the inroads made by a new fuzzer for DNP3. The model is not unfamiliar: take a complex new interoperability protocol produced by a standards committee, give it a couple of years for vendors to throw an implementation together and deploy, then security researchers take interest and write fuzzers for the protocol.

Vendor response timelines

The end result is predictable. Software untested for security defects tends to have security defects, and the 15 advisories produced (with more to come) signal that for all the SCADA noise emanating from the security community, vendors are still playing catchup.

An interesting addition were the timelines for patches and in this the talk provides a fresh perspective. For each of the advisories they showed the response timelines; the average was 75 days from report to fix with the shortest time being 12 days. This is concerning as it appears that SCADA vendors may be unable to respond quickly to an actual attack.

Takeaway: Industrial control security remains a lofty goal but this research shows that the industry remains in the early stages of securing their products.

[talk index]

Power Attack: An Increasing Threat to Data Centers [slides]

Haining Wang, Zhang Xu, Zichen Xu, Xiaorui Wang

A curious idea was presented in this paper about a novel Denial-of-Service attack. Starting with the observation that the electrical power supply of many data centers is over-subscribed since hosted servers almost never all run at 100% simultaneously, the authors explore the inescapable "what-if": can servers be synchronized to increase their power consumption beyond the supply rating and thereby cause power outages in the centre?

Attack in a nutshell

Taking the view that attackers access data centers typically in one of the cloud models (PaaS, IaaS, or SaaS), they explore how power consumption can be influenced in each of these circumstances. Their experiments show that, on a local scale, individual servers or racks can be pushed to consume more power than their local power distribution units can provide causing their breakers to trip.

It turns out that the attack, while simple to describe, has the inherent problem that influencing the power consumption of most machines in a data centre synchronously is, by itself, a very hard problem. Regardless, this was an academic paper so such realities are out of scope.

Takeaway: The risk of this attack is extremely low at this point, but it does raise questions about power supply in data centers and introduced a new form of Denial-of-Service.

[talk index]

The Sniper Attack: Anonymously Deanonymizing and Disabling the Tor Network [slides]

Aaron Johnson, Rob Jansen, Florian Tschorsch, Björn Scheuermann

Anonymity online is hard to achieve when facing opponents with access to legal tools (i.e. subpoenas), and perhaps the most well known counter to such threats is Tor, which lets users hide their IP addresses from the services they access. (By no means is Tor the only option, but is does seem to have the largest marketshare.)

Driving Tor relays offline

Tor works as follows: a client accessing, say, Google, choses three Tor relays and constructs a layered message which indicates which relays the request must be passed through. The trick is that each layer is, in essence, only readable by the two nodes who must exchange the message. It means that no relay can figure out both the sender of the request as well as its intended recipient.

The authors of this work discovered an attack which could be used to knock Tor relays offline. The details of the attack are quite simple: the attacker initiates a large download via Tor, with the victim relay being the first node along the Tor circuit. By two alternative tricks, it is possible to keep the download data flowing into relay, without it sending out any data. This fills the buffers and, with enough parallel downloads, will cause memory exhaustion on the relay. At this point, either the OS fails or, more typically, the Tor process will be killed.

At first glance this is a Denial-of-Service, but the authors go on to show how this can be leveraged into decloaking users by removing legitimate Tor nodes from the network, and waiting for users to utilize malicious nodes.

Takeaway: The attack is notable for a few reasons. Tor has been subject to much academic research in terms of theoretical attacks against its privacy defences, but its increasing real-world importance means that engineering choices are now being subject to attacks. By no means is this attack sophisticated, which suggests that further bugs lie in the intersection between the protocol specification and its implementation.

[talk index]

USB attacks need physical access right? Not any more... [slides]

Andy Davis

USB vulnerabilities require physical access. According to conventional wisdom, that is. In this talk the speaker shows how this has not been the case for a number of years now, at least on the Windows platform.

He starts by walking through his history of USB attacks, which all required physical access. While Remote Desktop on Windows supported access to USB devices, the client was still responsible for interacting with the device and so any vulnerabilities could only be triggered on the attacker's own client.

What this means for defenders

However, starting in Windows Server 2008 R2 SP1 Microsoft introduced the RemoteFX feature which includes USB redirection. This meant that USB device drivers could be installed on servers, and the RDP client could simply proxy USB packets from the device on the client-side, to the driver at the server. In other words, remote USB attacks became possible.

He demonstrates this through a USB vulnerability in an audio driver, where the USB commands are sent by a malicious client and exploit the server-side driver.

Takeaway: This opens up a range of attacks on RDP servers. USB bugs have typically received low attention due to their access constraints, but RemoteFX provides a remote path to reach them. The solution is to disable RemoteFX if not needed.

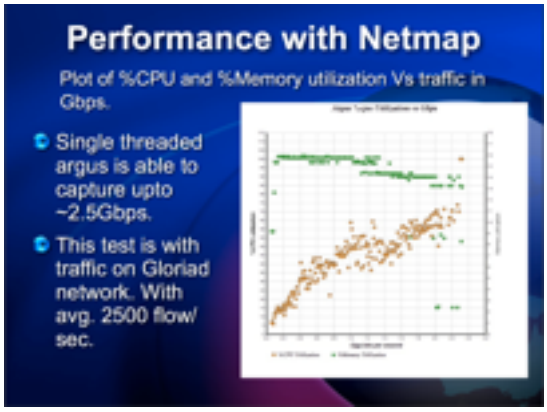
[talk index]

Defense

Argus with Netmap : Monitoring Traffic at 10Gbits/s Line Rate Using Commodity Hardware [slides]

Harika Tandra

At last year's FloCon, there was a talk on the inherent issues that superlinear growth in bandwidth brings to security folks. In essence, hard decisions would be needed once networks exceeded the capabilities of monitoring equipment, and that speaker then suggested that traffic sampling was the only available route.



Netmap performance

This talk hints at that future. Ostensibly, it shows how to achieve Gigabit monitoring capability with commodity hardware (FreeBSD on Dell servers using the Netmap framework). A single threaded application could consume 2.5Gbs and they spoke of extending the monitor to multiple threads and thereby increase the monitoring capacity.

But the fact remains that storing the data (or even sifting it for signs of attacks before storing) consumes significant time, more time than is available to the hardware. These incremental improvements do not radically change the view presented last year.

Takeaway: We still don't know how to deal with the scaling of bandwidth in defending networks.

[talk index]

Fix What Matters: Why Using CVSS for Remediation is Nuts [slides]

Michael Roytman

CVSS is the de facto method by which vulnerability impact is communicated, but there is a growing that the plain CVSS score has limited usefulness. This talk focused on the use of CVSS scores in prioritising patches and fixes. The speaker presents a fairly simple alternative which relies on more data than just the bug publisher's risk analysis.

Measures to judge vulnerability severity

It is not the first talk to highlight CVSS weaknesses by a long shot, but the examples provided are worth looking at. When it comes to prioritising patches, organizations might choose to start with those with a CVSS score of 10 and work their way down but this is wrong.

The speaker shows an analysis of breach data which combines breaches with the vulnerabilities used in each breach. Properties about the incident could then be explored such as what the CVSS score of the vulnerability used was, or whether the exploit was public and, if so, where it was published. The analysis then concluded by looking at the predictive value of each property and found that a CVSS 10 score was far inferior to the presence of an exploit on ExploitDB and Metasploit when predicting if a breach would occur.

Takeaway: This analysis is squarely within the recent trend of threat intelligence. While not as sophisticated as it could be, the talk provides a concrete path for alternatives to CVSS when prioritising fixes.

[talk index]

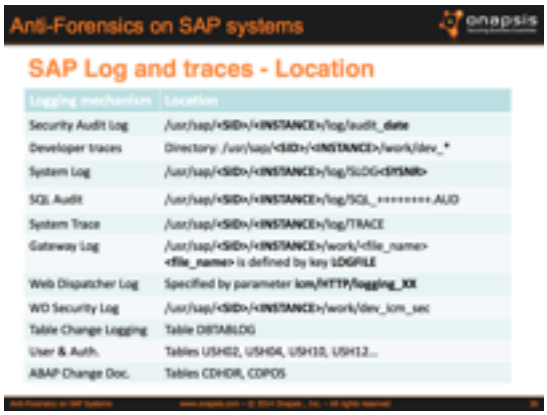
Hiding the breadcrumbs: Anti-forensics on SAP systems [slides]

Will Vandevanter, Juan Perez-Etchegoyen

SAP has a small but focused group of researchers who continually find new vulnerabilities in the ERP behemoth. This talk by employees of a SAP security company, took a slightly different approach in that it offered practical advice for how SAP attackers might hide their tracks, i.e. anti-forensics. Four approaches were described.

For defenders, anti-forensics work is useful as it provides a minimum bar below which attackers will definitely be able to hide their tracks. SAP system owners should ensure that their installations do not have the vulnerabilities listed in the talk, as those issues allow attackers to cover their traces.

Takeaway: The presentation gives SAP owners hints on how to lock their SAP installations to preserve forensic information.



Audit locations on SAP

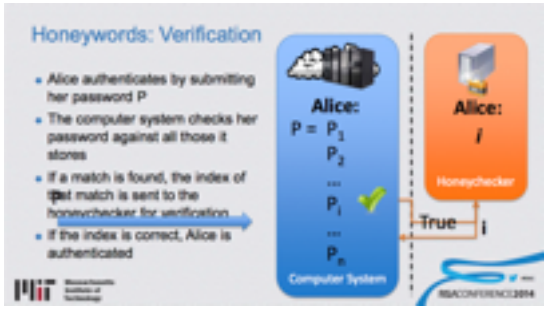
[talk index]

Honeywords: A New Tool for Protection from Password Database Breach [slides]

Ronald Rivest, Kevin Bowers

Data breaches are often detected through increased spam or fraudulent activity on stolen accounts, but other options exist. This talk presented a simple concept to aid breach detection and reduce the chance of stolen credentials being used.

Simply put, multiple passwords are stored for every user but only one of them is valid. When a user tries to authenticate, an external "Honeychecker" is queried to find the index of the correct password. If the correct password is given, then authentication succeeds; if the supplied password is not present at all in the list of potential passwords then it is simply treated as an invalid password. However, if the user-supplied password is in the list but not at the correct index then a breach has been found.



Verification flow

The solution is far from perfect, but it demonstrates the potential for builtin defences we've often advocated. Honeywords require an extra checking service with a separate database (i.e. incorporates elements of distributed security) which are potentially subject to attacks; but the approach's benefit is that it can help detect data breaches timeously. There are challenges in producing believable alternative passwords which the presentation covers in detail, but overall the idea of lures and bait is attractive for self-defending applications.

Takeaway: Honeywords are a form of application self-defense that can help detect data breaches.

[talk index]

Raising Costs for Your Attackers Instead of Your CFO [video]

Aaron Beuhring, Kyle Salous

Reasons to whitelist applications

As longtime proponents of application whitelisting, we were happy to see a talk that provides practical advice and suggestions to Active Directory administrators for implementing Microsoft's AppLocker. The argument the speakers make is that whitelisting is likely already available in your corporate network, either through Microsoft or with your anti-virus product, and so what remains is not deployment but configuration.

If one is thinking of rolling out AppLocker, then this talk provides concrete advice and lessons learned.

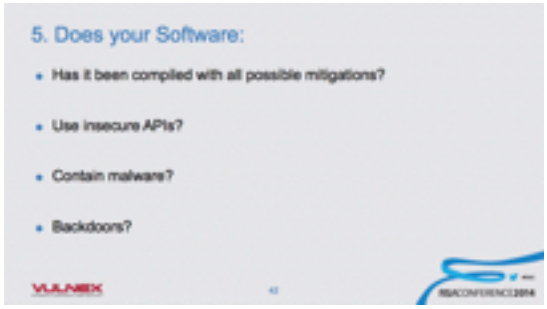
Two other techniques discussed in the talk for frustrating attackers were application-aware firewalling on Windows and reputation-based whitelisting for both network and file operations. File reputation has the potential to counter the typical custom payloads that sidestep signature-based AV, in that if a file has never been seen before then it has a poor reputation and one can act on that fact.

Takeaway: The main thrust of the talk was that these defences are likely part of your security suite already deployed and so the cost is time rather than licenses.

[talk index]

Writing Secure Software Is Hard, but at Least Add Mitigations! [slides]

Simon Femerling



Checks to perform

For many corporates, evaluating the security of third party tools is an outsourced job due to the specialist nature of the work. However, for poorly written tools this is an expensive exercise as serious issues will be found very quickly yet the consultants still need to be paid for all their time; it's a wasted exercise.

This talk introduced BinSecSweeper, a tool which evaluates ELF and PE binaries (and their related libraries) for obvious issues including missing defences such as the enabling of ASLR and DEP/NX, overflow protections, safe SEH handling and use of unsafe APIs, among other checks. This is a list of basic security designs which, if missing, highlight poor posture and can remove the need for expensive consulting in the initial PoC or exploratory phase of software purchasing.

We have previously included research about the danger of old libraries in packaged tools, and BinSecSweeper can help quickly determine if packaged libraries are improving, or if they still remain vulnerable.

Takeaway: The tool is not a traditional static analysis tool which tries to determine under which circumstances a vulnerability occurs. Rather, it lets the user quickly determine at a glance if a binary has enabled compiler-level protections and avoided poor patterns.

[talk index]

Conclusion

The first quarter of 2014 introduced two new themes and the continuation of two previous ones. We are relatively pleased with the steady increase of concrete, actionable, defense focused talks that are bubbling up (especially since some of them have covered techniques we have long been in favor of).

Although this quarter had a large number of conferences, they are typically not the sorts of events that bring out the heavy hitting offensive research and stunt hacks.

We expect the pace and quality of research to accelerate as the year progresses.

This update was written by marco@thinkst.com. Please contact me if you have queries or comments relating either to this report, or the ThinkstScapes service.