We Have a Problem With the ‘Security’

Marcus J. Ranum
mjr@ranum.com

A Note On Words

- When I say “secure” you can think “resilient” (or vice-versa)
  - They are not the same thing but they are so interdependent that they act like they are the same
- When I say “complex” you can think “failure-prone” or “insecure”
  - Again: not the same thing but so interrelated that they track together

In Case You Missed It

- Security is not very good
  - Operations
  - Software
  - Governance
- We have had a computer security industry for 30+ years and there has not been substantial improvement
  - By “substantial” I mean: no major or significant problems have been solved

Why?

- There are too many reasons and the interactions are subtle
- Candidates:
  - It’s very hard
  - Nobody really wants it
  - Doing other things is more fun
  - It hasn’t really mattered because of resilience in the rest of the system
  - Complexity

Let’s Talk Complexity

- We don’t even build resilient systems
  - To build resilient systems we have to manage complexity much better than we do
  - Instead, we build high-maintenance systems
- Put another way: “it’s 2019, why do we still need system administrators?”
  - We need them to install patches
  - We need them to install upgrades
  - It is easier to fiddle with stuff than build resilience

Complexity: The Conundrum

- Since complexity is the enemy of security (and resilience and low cost operations) why does our response to security problems seem to always involve making the system more complex?
  - It's like putting out a fire with diesel fuel or building a castle out of duct tape
  - Security is the universe of band-aids and patches instead of steady improvement in software resilience and quality, and operational simplicity

A Hypothesis

- Maybe Bill Joy (Sun Micro.) was right, that “network effect” makes systems much more powerful
  - It also makes them much more complex (thus less resilient or secure)
  - It also allows them to do more important stuff (that’s what “powerful” means)
- Our computer systems are getting more important so they are getting worse

Our Responses

- Our responses to the situation appear to be becoming more desperate and less well thought-out
  - “Let’s add AI to everything!”
  - DEVOPS
  - Containerization
  - Virtualization
  - Automating patching so that our machines now reboot at random times with random results

An Example

- A client comes to me and says “we want to do software defined networking and maybe micro-segmentation/zero trust”
  - The problem is that they have currently got a large single-zone network with absolutely no interior landscape – everything can talk to everything
  - They have demonstrated that they *do not know how* to do SDN or segmentation
  - It’s like a child asking for a table-saw because they don’t know how to use a hand saw safely

An Example (continued)

- Meanwhile, the same client has been moving projects piece-meal to the cloud and some developer groups have switched to DEVOPS; there is no central planning of this, it just happens organically
  - In other words: they are making the problem worse, as fast as they can while simultaneously asking “how do we manage this problem?”

We Call It “Immaturity”

- We need better CTOs and better customers
  - CTOs that can resist fads and marketing
  - Customers that are patient and understand what they are buying (don’t just run out and buy it because the sticker says ~~big data~~ ~~cloud~~ ~~virtualization~~ AI on it
  - Searching for panaceas makes the problem worse, not better

The Problem With Patches

- Thanks to endless floods of security vulnerabilities and patches resulting from them, system configuration is a moving target
  - You can no longer set your system up and drop into maintenance mode because ***software rots*** and cannot function without intervention unless it is isolated from other networks and systems
  - Thanks to auto-patch, you can't even isolate systems because they complain they can't get patches

There Are Rays of Hope

- NASA's software development process during the shuttle and Apollo program set a high bar for excellence through engineering discipline (Margaret Hamilton)
  - There are spots around the industry where development is a sober engineering process
- Fly-by-wire aircraft do not use web browsers as a primary control interface, yet, and we do not control nuclear weapons with SMS texts

Now: What Worries Me?

- After painting such a rosy picture of our software/technology environment…

A Foundation Of Duct Tape

- What you, and all computer users in the future will have to work with is a computing infrastructure that is built on foundations of duct tape
  - Unfortunately, once you have this, your best remaining option when something goes wrong is to grab more duct tape and start putting it on all the things
  - Auto-patching just re-arranges the duct tape periodically

The Rest of The World's Problem

- Once you step outside of the United States and its sphere of influence, all software and hardware look suspiciously like a trojan horse
- Oddly, right now, the US is screaming about this very problem itself, regarding Huawei
  - Is China subverting Huawei's products in order to gain control of the duct tape that comprises the infrastructure?
  - Has the US subverted Facebook, Apple, and Google? (That's the big duct tape)

The Huawei/Snowden Problem*

- Can we trust hardware and software that is built for us by people that we don't trust?
  - The answer is rather obvious: no
  - But most of the world is expected to
    - But! RSADSI wouldn't take $24mn from the NSA to backdoor BSAFE, which became an early reference implementation in most internet cryptography between 1994 and 2005
      - It is *pure coincidence* that happened right after the clipper chip idea was finally shot down

* See the book by Olav Lysne by this name

The Standards Body Problem

- Why does SSL not do bi-directional authentication?
  - It was originally designed to
- Probably for the same reason that your browser comes populated with ~200 certificates that can authenticate anyone
  - It's as though it was *designed* to be easy to "man in the middle" – which is odd for a protocol that is supposed to protect transactions

The Hidden CPU Problem

- Why and how did something like Intel Management Engine happen? (Or the hidden instruction set in the Via processor?)
  - We expect CPUs to do what they are told
  - Operating systems *depend* on there being things like separate memory, trusted processing modules, etc.
  - The whole notion of “operating system security” falls apart in the face of IME

See Christopher Domas’ presentation
At Black Hat entitled “God mode unlocked”

Addendum on Hidden CPUs

- Even if IME is a well-intentioned set of poor technical decisions, having a poorly documented CPU inside your CPU, running a full-blown multitasking kernel with an IP stack …
  - No mature and reliable engineering process would permit such a thing to happen without customers knowing about it
  - It utterly shatters any trust we might have in Intel's design discipline

MINIX is now the most widely-deployed operating system, ever

Consider Hacking as Complexity

- Hacking is a form of new complex actions our systems must deal with, unexpectedly, from outside
  - When we have protocol flaws, processors with hidden CPUs, and we don't know what kind of other processors are hidden on our motherboard – *how can we produce any kind of reliable computing?* How do we build resilient systems on top of this mess? Sure, our foundation may be made of duct tape but we don't even know what's in the duct tape!

Snowden As A Warning

- Snowden's leaks were followed by the Vault 7 leaks, and the leaks from the NSA contractor* who took home a collection of malware and it was stolen by hackers
- Shamoon/Stuxnet/Flame – the "Equation Group" attack tool-stack spawned a whole new generation of malware
  - These leaked tools have been easy to leverage into consumer/criminal tools like Wannacry

* Nghia Hoang Pho, 67

The Problem With All This

- For the last 15 years I have been referring to cyberwar as “the department of glass houses, developing stone-throwing technology”
  - And now we are there
  - We all have to deal with this huge unexpected complexity
  - Your system may be built in accordance with best practices and it could still have open backdoors in it that allow anyone to take it over at any time, if a certain piece of information leaks

So, How to Remain Cheerful?

- This is not how:

In recent cybersecurity tests of major weapon systems DOD is developing, testers playing the role of adversary were able to take control of systems relatively easily and operate largely undetected.

DOD's weapons are more computerized and networked than ever before, so it's no surprise that there are more opportunities for attacks. Yet until relatively recently, DOD did not make weapon cybersecurity a priority. Over the past few years, DOD has taken steps towards improvement, like updating policies and increasing testing.

Federal information security—another term for cybersecurity—has been on our list of **High Risk** issues since 1997.

Weapon Systems Cybersecurity:
DOD Just Beginning to Grapple with Scale of Vulnerabilities
GAO-19-128: Published: Oct 9, 2018

Where Do We Go From Here?

- Software/hardware/configurations – the *whole thing* – become a strategic resource that will be manipulated
  - Everything is being built on a battlefield
  - Perhaps some governments (that do not want to operate under US hegemony) will need to start thinking about how to develop their own hardware/OS/network stacks
  - An insane amount of wasted work
  - A good business opportunity

What Can We Do?

- Our best option is to explain to governments that when they behave this way they make their own job harder, they make their own systems less resilient, they make everyone's outcomes less reliable
  - They need to stop
  - They are not smart enough or wise enough to figure this out on their own (or we wouldn't be in the situation we are in)

The Cheerful Close

- It's my responsibility to end on an upbeat and inspiring note but this is not an upbeat or inspiring situation
  - I think that the best we can take away is to show increased vigilance when we build important systems
  - We may get an opportunity to build a whole new computing ecology (we should!) someday
    - If we do, let's not make these same mistakes

The Cheerful Close

- Computing is much faster and cheaper!
- Perhaps Artificial Intelligence will solve all this

Thank You

The Depressing Close

- Our problem appears to be boiling down to “we should just be happy if we still have a technological civilization in 50 years.”
  - The global CO2 rise is tracking scientists’ worst projections so the worst case scenario means we won’t have to worry about malware for much longer