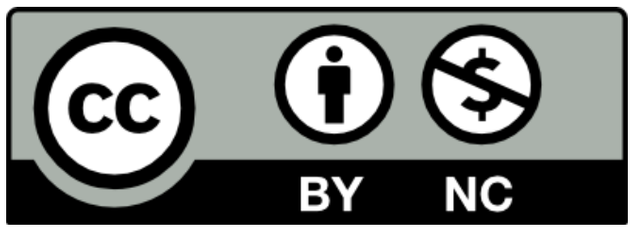
30 Years of Defending the Internet

Steven M. Bellovin
https://www.cs.columbia.edu/~smb



The 1980s: Dawn of the Internet

(All photos by the author)

1988

- May: Cliff Stoll publishes his seminal *CACM* article "Stalking the Wily Hacker"
  - He described how his site was hacked, and how he monitored the hacker and traced him
  - His book *The Cuckoo's Egg* came out in 1989
- November: The first Internet worm is launched

But all these problems are solved now, right? Let's take a look back…

Technology

- ARPANET core (IMPs)
- Some institutional Ethernets
- X.25 public nets (Tymnet, Datex, etc.)
- Dial-up phone lines for access

The Wily Hacker

- In 1986, Stoll was an astronomer working as a sysadmin at Lawrence Berkeley Labs

- There was a $.75 accounting discrepancy—and that led him to find a hacker

- Tracking the hacker took almost a year…

Attacker Characteristics

- Penetrations happened via password compromise
  - Default passwords (system/manager, field/service, guest/guest, etc.) and open accounts
  - Users sharing passwords via email, and storing passwords in files
  - Password-guessing, though Stoll didn't realize that until late in the game
- Bugs and flaws were used for privilege escalation
- The attacker was hard to trace because of stepping stones

Administrative Response

- FBI: $.75 loss? Go away!
- CIA: Fascinating
- NSA: Interesting—tell us about the attacker's techniques (and yours)
- Management: Why are you wasting your time on this?
- "The message from Germany read: 'The German State Prosecutor needs to contact high-level U.S. criminal justice persons so as to execute proper search warrants. The Bundespost cannot move until officially notified by a high-level U.S. criminal office.'" [Quotes from *The Cuckoo's Egg*]

Social Response

- Not new: “‘So what? Somebody’s always had control over information, and others have always tried to steal it. Read Machiavelli. As technology changes, sneakiness finds new expressions.’”

- “‘A computer system isn’t private like a house,’ Laurie responded. ‘Lots of people use it for many purposes. Just because this guy doesn’t have official permission to use it doesn’t necessarily mean he has no legitimate purpose in being there.’”

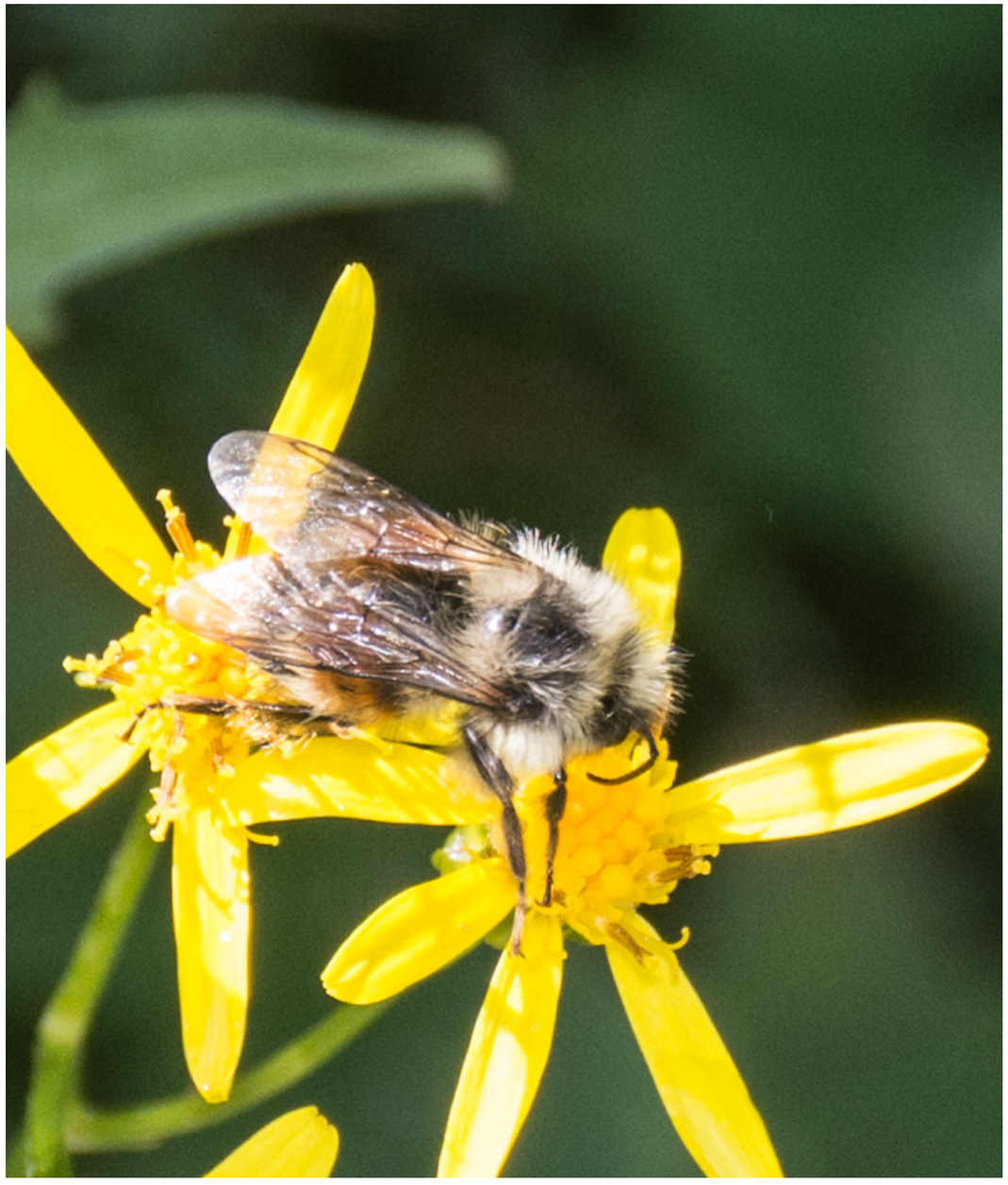
- ‘Whenever a fun-loving student breaks into systems as a game (as I might once have done), and forgets that he’s invading people’s privacy, endangering data that others have sweated over, sowing distrust and paranoia.’

Honeypots

- Stoll wanted the attacker to stay on longer, so he could be traced

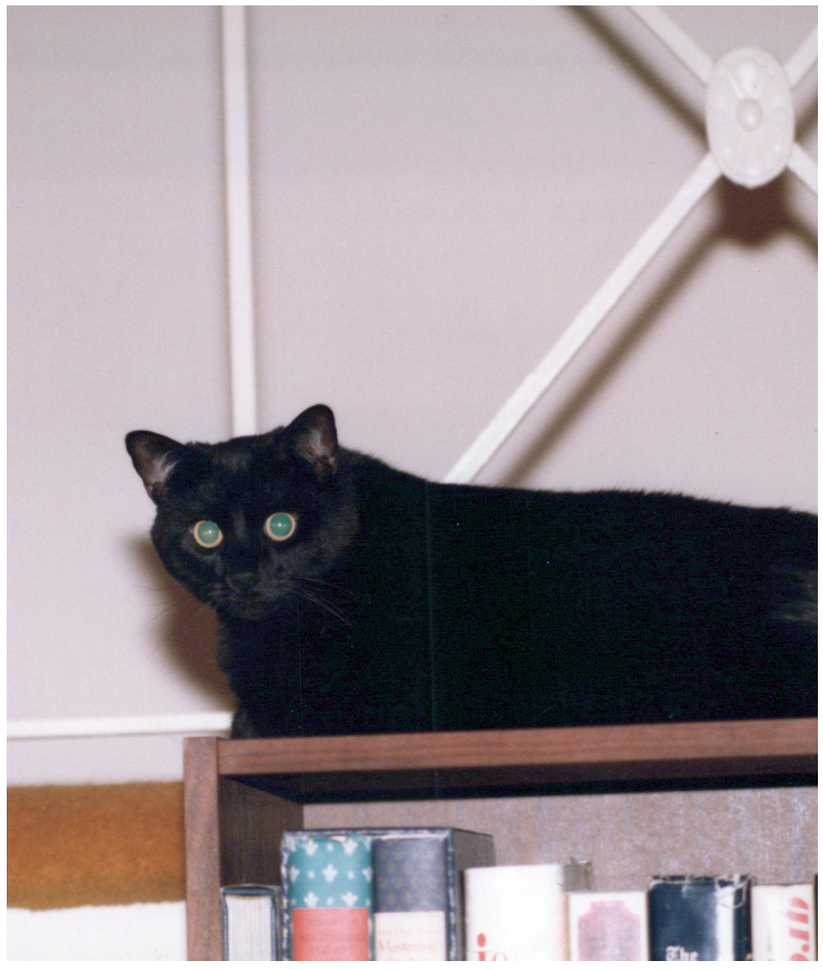
- He created fake documents about the Strategic Defense Initiative (an anti-missile system)

- Someone (probably a Soviet bloc agent) even sent a physical letter asking for more documents!



Internet of Things

- “That computer is the Petvax. Stored within it are patient records, analysis programs, medical data, and scans of people’s brains. This hacker was playing games with medical tools. Break this computer, and someone’s going to get hurt. A bad diagnosis or a dangerous injection. Or what?” (A bevatron control computer was also hacked.)



Stoll's Analysis

- Passwords were a problem—memorable ones were guessable; random ones were stored in files—but worked
- Systems weren't secure as shipped
- Usable security: "'We've got to turn this around,' Bob [Morris] said. 'Secure computers might keep the bad guys out, but if they're so balky that nobody will use 'em, it won't be much progress.'"
- People didn't install patches
- How do you disclose vulnerabilities responsibly?

Lots of Hackers

- Even in 1986, hackers weren't novel
- Most folks Stoll contacted had hacker problems
- Internet connectivity for a host was rare—but dial-up modems were common
- Attacker goal: shell access, then root access (or equivalent)
  - In other words, *system* access

The Internet Worm

- Multi-platform (4.3BSD Vax and Sun 3)
- Multi-vector (sendmail hole, fingerd buffer overflow, password-guessing, transitive trust)
- Encrypted payload
- Nothing malicious—but it multiplied far too quickly and clogged hosts

Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now
Vector	*Passwords*				
Perpetrators	*Joy hackers*				
Motive	*Money; curiosity*				
Defense	*Hardening hosts*				
Attack Surface	*High*				

Early 1990s: Firewalls and the Web

Attack Surface

Stoll: “Tightening one computer was like securing an apartment house. But a network of computers, all sharing files and interchanging mail, well, this was like securing a small city. Bob [Morris], as chief scientist of the Computer Security Center, directed that effort.”

- Too many machines were vulnerable; hardening them all was too hard
- We needed a scalable solution: firewalls

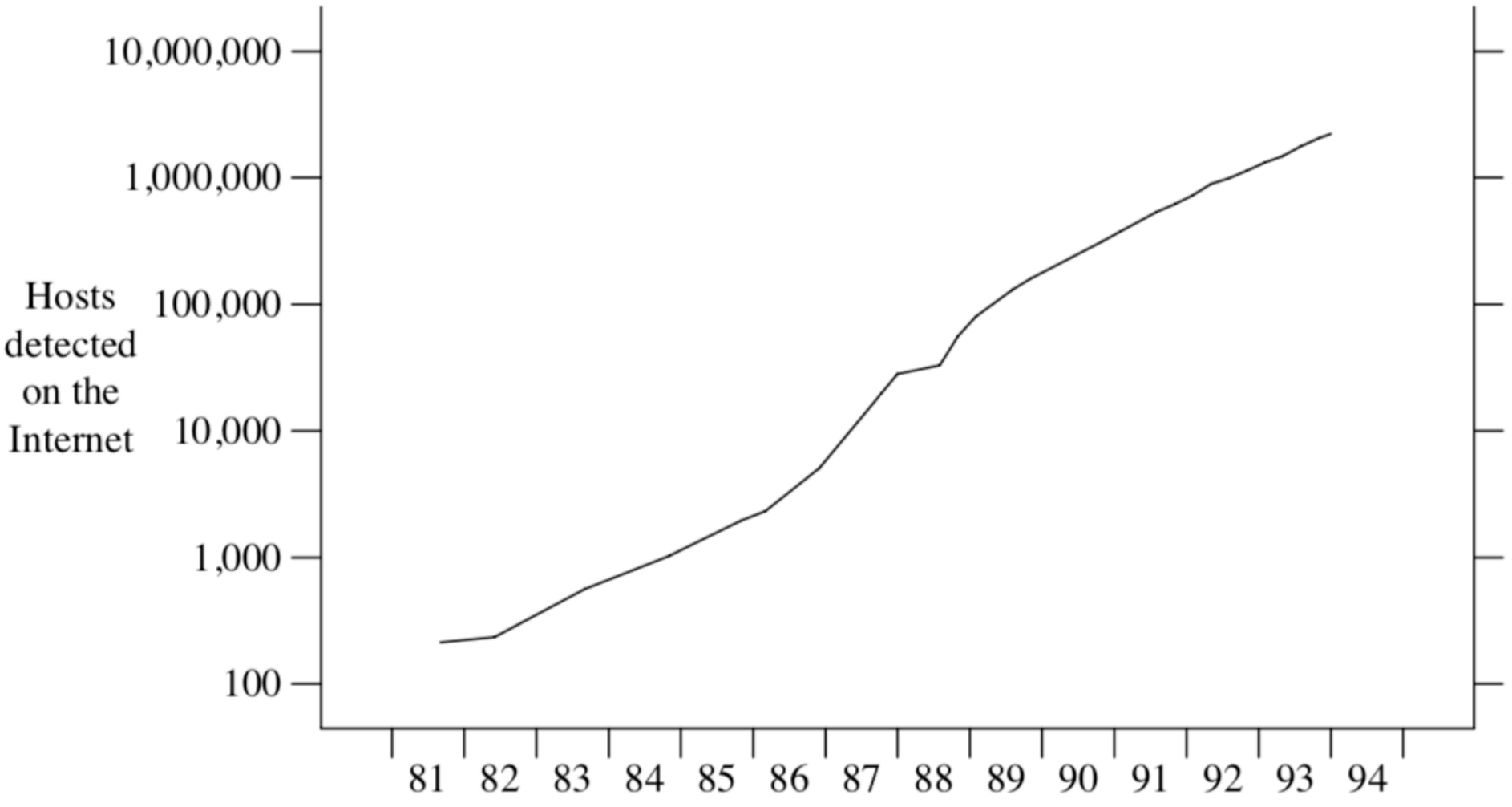
Firewalls: Evolution, not Invention

- The notion of a single gateway host for an organization wasn't new
  - Many motives, but security was one
  - Modern packet filter: Mogul, 1989
  - Cryptographic network access control: Estrin and Tsudik, 1989
  - Application firewall: Cheswick, 1990
  - A quasi-theory of firewalls (Cheswick and Bellovin, 1994)
  - The beginning of commercial firewalls

Scalability

From Cheswick and Bellovin (1994):

- Firewalls have a much smaller attack surface
- Ease of administration; professional administration
- Chokepoint (customs inspection) for traffic
- More logging
- Firewalls are the “network response to a host security problem” (Bellovin, 1994)



Growth of the Net

Monitoring an Attacker

- Firewalls are also a great place to monitor traffic
  - "An Evening with Berferd" (Cheswick, 1992)
  - "There Be Dragons" (Bellovin, 1992)
- By definition, all traffic passes through that point
  - Easier for us than for Stoll—he had to monitor multiple entry points to his net

Firewall Limitations

- "By its nature, a firewall is a very strong defense against attacks at a lower level of the protocol stack… [F]irewalls provide almost no protection against problems with higher level protocols" (Cheswick and Bellovin)
  - Email? Web?
  - "A recent sendmail bug provides a sterling example. Problems with certain mail header lines could tickle bugs in delivery agents. Our firewall, and many others, paid almost no attention to headers"
- Firewalls depend on topology—if there is connectivity that doesn't pass through the chokepoint, the firewall provides no protection

The Web

- The Web made the Internet accessible to ordinary users
- Consumer ISPs started to appear—with no firewalls
- But that mattered little, because the web wasn't (quite) an attack vector
  - Viruses mattered more, and those were often spread via floppy disks

Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now
Vector	*Passwords*	*Bugs*			
Perpetrators	*Joy hackers*	*Joy hackers*			
Motive	*Money; curiosity*	*Curiosity*			
Defense	*Hardening hosts*	*Firewalls*			
Attack Surface	*High*	*Moderately low*			

Late 1990s: The Diet of Worms

(Sorry, I only have snake pictures)

Growing Attacks

- In 1996, Aleph One published "Smashing the Stack for Fun and Profit"
  - A cookbook for how to carry out buffer overflow attacks
- Many new sites joined the Internet; most had inadequate administration
- Code bases grew much faster than quality control
  - "Ship first, debug later"

More Targets

- Many—most?—consumer-facing companies were on the Web
- That includes financial institutions
- Spammers had discovered the joys of email
- Some US military sites were penetrated
  - It seemed like enemy action and the President was notified
  - It was Israeli and California teenagers
- In Operation Eligible Receiver, the NSA showed that the US power grid was vulnerable

Worms

- The existence of many buggy hosts made worms feasible
  - Example: the ILOVEYOU worm spread via email and low-grade social engineering
  - SQL Slammer spread via UDP
  - Code Red infected IIS web servers (unknown to most users, lots of ordinary hosts seemed to run web servers and database servers)
- Most worms had no particular goal—but they clogged the net
  - The Blaster worm blocked CSX railroad's signaling network

What Happened?

- There was more buggy code than imagined, and much of it was not stopped by firewalls
  - ILOVEYOU was email, and passed right through
  - Traveling laptops were infected by Code Red (which only infected Microsoft web servers) and brought it home
- Telecommuting and traveling employees (via home broadband) were often unprotected by firewalls—VPNs were not yet common
  - Besides, family members often shared the employee's laptop
- There were often business-to-business links that bypassed the firewall
- *The properties necessary for successful firewalling no longer held*

Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now
Vector	*Passwords*	*Bugs*	*Bugs*		
Perpetrators	*Joy hackers*	*Joy hackers*	*Joy hackers*		
Motive	*Money; curiosity*	*Curiosity*	*Mischief*		
Defense	*Hardening hosts*	*Firewalls*	*Firewalls*		
Attack Surface	*High*	*Moderately low*	*Very high*		

Mid-2000s: Follow the Money

A Phase Change

- Circa 2003, destructive worms stopped happening
- "Computer security experts and law enforcement officials are struggling to understand the motives of a mysterious software author who appears intent on prying open many of the electronic locks on the Internet." (NY Times)
- "'I think the motivation is clear: it's money,' said Mikko H. Hypponen"
- Yup!

There's Gold in Them Thar Hosts!

- Attackers had figured out that there was money to be made from hacking
- Worms that shut down the Internet are bad for business
- Besides: why waste time on joy-hacking when you can profit from your skills?

Spammers

- Originally, spam was sent via open relays, but those were being closed down
- Better idea: hack endpoints; let them send spam
- The spammers *paid* the hackers
- There was now a profit motive for hacking and the market worked its magic

Phishing

- People did online banking—and logged in with passwords
- Phishing emails and keystroke loggers could collect those passwords
  - Again, email passes through firewalls
  - Drive-by downloads and buggy code let other applications be attacked, e.g., Flash
- TLS certificates didn't help—people didn't notice the *absence* of a correct certificate

The Threat Model Changed

- The 1990s were the domain of amateur hackers
- Their resources and skills were limited
  - The security troubles then were caused by poor code (Windows was then new to the Internet) and inexperienced users who were using inadequate systems
- In the 2000s, there were new, tempting targets: bank accounts
- The attackers were more skilled

But it took defenders too long to catch on to the change!

Software Quality

- Software quality started to improve
  - Microsoft got religion after the Gartner Group warned companies to ditch IIS
- The end of the `.com` boom drove a lot of smaller companies out of business
  - There was less pressure to ship fast
- But *no one* had the time or money to rewrite all of the old, buggy code
  - And backwards compatibility was important

Hardening Hosts

- We started to see hardened hosts
  - iOS is one of the most secure operating systems
  - Windows Vista and Windows 7 showed the benefits of Microsoft's security effort
- Sandboxing went mainstream
  - The old Trusted Computing Base was pronounced dead (though it had been on life support for 10 years)

Sandboxing

- The TCB model: trust the kernel; it doesn't matter what users do
  - This was a model for timesharing systems: protect users from each other
- Newer problem: users are affected by trouble at the application layer
  - (Actually, graphically shown by the 1988 Internet Worm)
- Sandboxing: limit the abilities—and hence the damage that can be caused by—a subverted application
- A new OS feature—because applications will *always* have bugs

Patching

- All operating systems and most applications alert users to patches
- Microsoft instituted “Patch Tuesday”: scheduled releases of updates, to ease the sysadmin load
  - Sometimes, there’s a serious enough attack in the wild that out-of-cycle patches are released
  - Attackers reverse-engineer patches
  - Sometimes, patches “brick” systems or cause other serious problems

Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now
Vector	*Passwords*	*Bugs*	*Bugs*	*Bugs; phishing*	
Perpetrators	*Joy hackers*	*Joy hackers*	*Joy hackers*	*Thieves*	
Motive	*Money; curiosity*	*Curiosity*	*Mischief*	*Money*	
Defense	*Hardening hosts*	*Firewalls*	*Firewalls*	*Firewalls; patches; sandboxes*	
Attack Surface	*High*	*Moderately low*	*Very high*	*Very high*	

2009-Now: Miltarizing the Net

Stuxnet! Snowden!
Sony! Shamoon!

- The Internet is now part of most countries' critical infrastructure
  - Targets attract weapons
  - Information attracts spies
- The militaries and spy agencies of the world understand this

Attacker Skills

- High-end attackers today are *very* skillful, and have vast resources
- Spear-phishing works better if you know a lot about the target
- But lesser threats are still serious when employed by skilled agents
  - Why use a complex exploit when a simpler one works?
- It's proven relatively easy for even less-developed countries to build up sophisticated military hacking skills

Sophisticated Attacks

- Operating system defenses are quite good—but they're not perfect
  - 15 years ago, buffer overflows were ~50% of all attacks
  - Technology has stopped most of those—but we now have things like ROP and other code reuse attacks
- Today's exploits are multistage: get a beachhead, escape the sandbox, do privilege escalation
  - Chain together multiple vulnerabilities to penetrate a system

A Vast Array of Targets

- Other countries' government and military infrastructure
  - Also, other critical infrastructure, e.g., the power grid
- Well-placed monitoring points
- Defense contractors
- Other countries' commercial technologies
- Personal information databases
- Dissidents
- More or less anything else imaginable

Ordinary Thieves are Better

- Why steal credit card numbers one at a time when you can steal 50 million at once?

- Massive information thefts
  - Within the last week, Marriott and Quora disclosed massive breaches (500 million and 100 million users' data taken)

- "Why rob banks? That's where the money is."

It's No Longer the System

- Attackers today go after *services* rather than systems
- Email accounts are a major target—and for those, you need a login and password, not root access
  - Intelligence agencies want to spy—but ordinary thieves want password reset emails
- Systems may be compromised, but only as a way to get at the data on them

Scalable Defenses

- Machine learning and large cloud email providers (Google, Microsoft, Yahoo) are a good way to defeat spam
- Amazon AWS, Google, Microsoft, IBM run very secure computing and storage complexes
- Chromebooks—the return of timesharing service bureaus
- Google can often detect nation-state attacks

Defeating Phishing

- Two-factor authentication is becoming more common
- Corporations have long used it, but now consumers are starting to
- Phones are a common second factor, whether via apps (reasonably secure) or SMS (not so much)

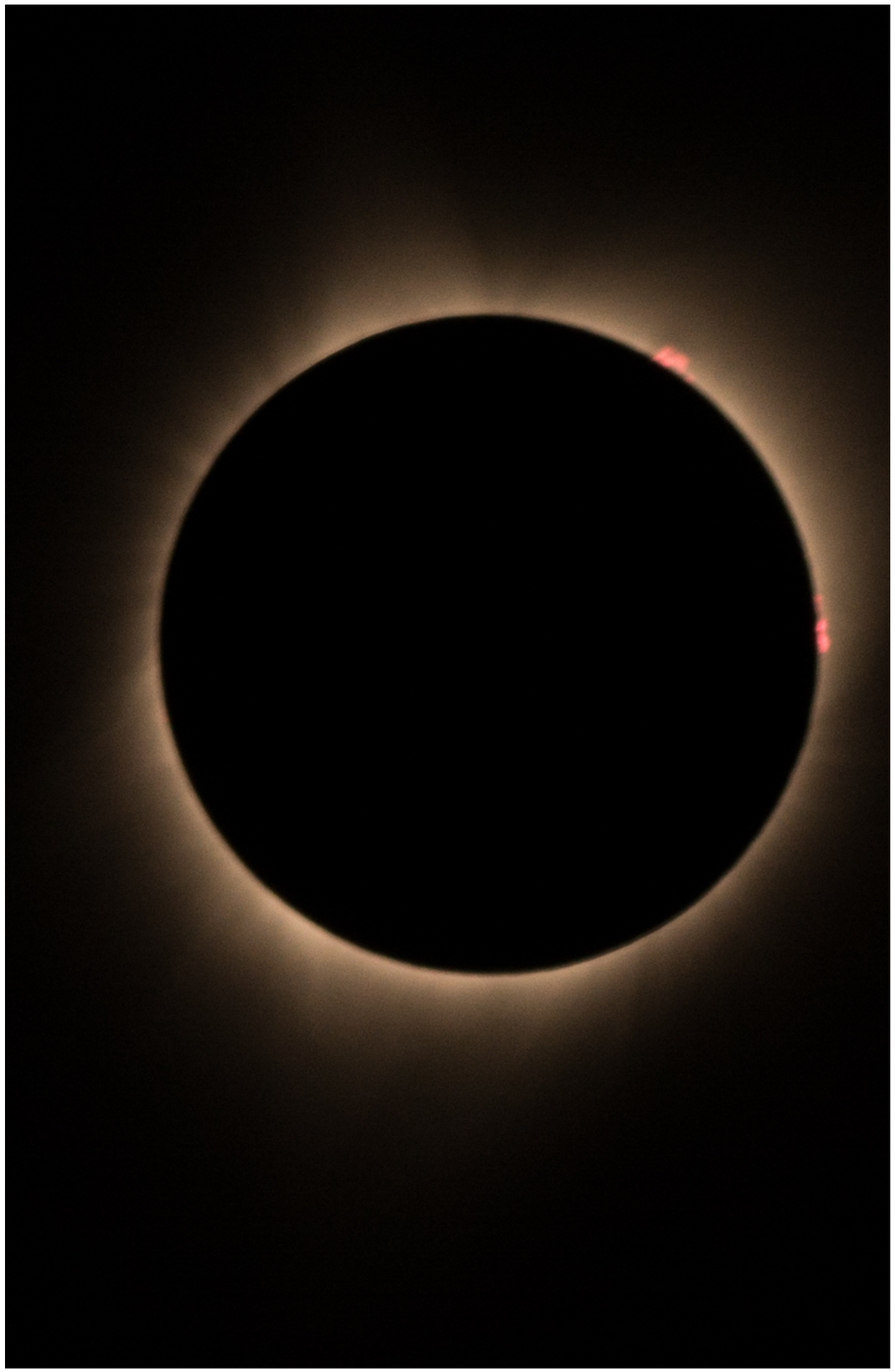
Patching

- Some vendors (Microsoft for Windows 10; Google for Chromebooks) push patches to users
  - Some of these patches still break things
- Corporations are starting to realize the importance of speedy patching—and many have development and deployment styles that are amenable to this
  - Equifax was hit hard because they didn't patch a critical system

Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now
Vector	*Passwords*	*Bugs*	*Bugs*	*Bugs; phishing*	*Bugs; spear-phishing*
Perpetrators	*Joy hackers*	*Joy hackers*	*Joy hackers*	*Thieves*	*Thieves and spies*
Motive	*Money; curiosity*	*Curiosity*	*Mischief*	*Money*	*Money; intelligence; military*
Defense	*Hardening hosts*	*Firewalls*	*Firewalls*	*Firewalls; patches; sandboxes*	*2FA; auto-patch; cloud; spam filters*
Attack Surface	*High*	*Moderately low*	*Very high*	*Very high*	*High*

The Future



Attack Matrix

	1988	1990-1995	1996-2003	2003-2008	2009-now	The Future
Vector	*Passwords*	*Bugs*	*Bugs*	*Bugs; phishing*	*Bugs; spear-phishing*	*?*
Perpetrators	*Joy hackers*	*Joy hackers*	*Joy hackers*	*Thieves*	*Thieves and spies*	*?*
Motive	*Money; curiosity*	*Curiosity*	*Mischief*	*Money*	*Money; intelligence; military*	*?*
Defense	*Hardening hosts*	*Firewalls*	*Firewalls*	*Firewalls; patches; sandboxes*	*2FA; auto-patch; cloud; spam filters*	*?*
Attack Surface	*High*	*Moderately low*	*Very high*	*Very high*	*High*	*?*

Is There Hope?

- Can we solve some of these problems?

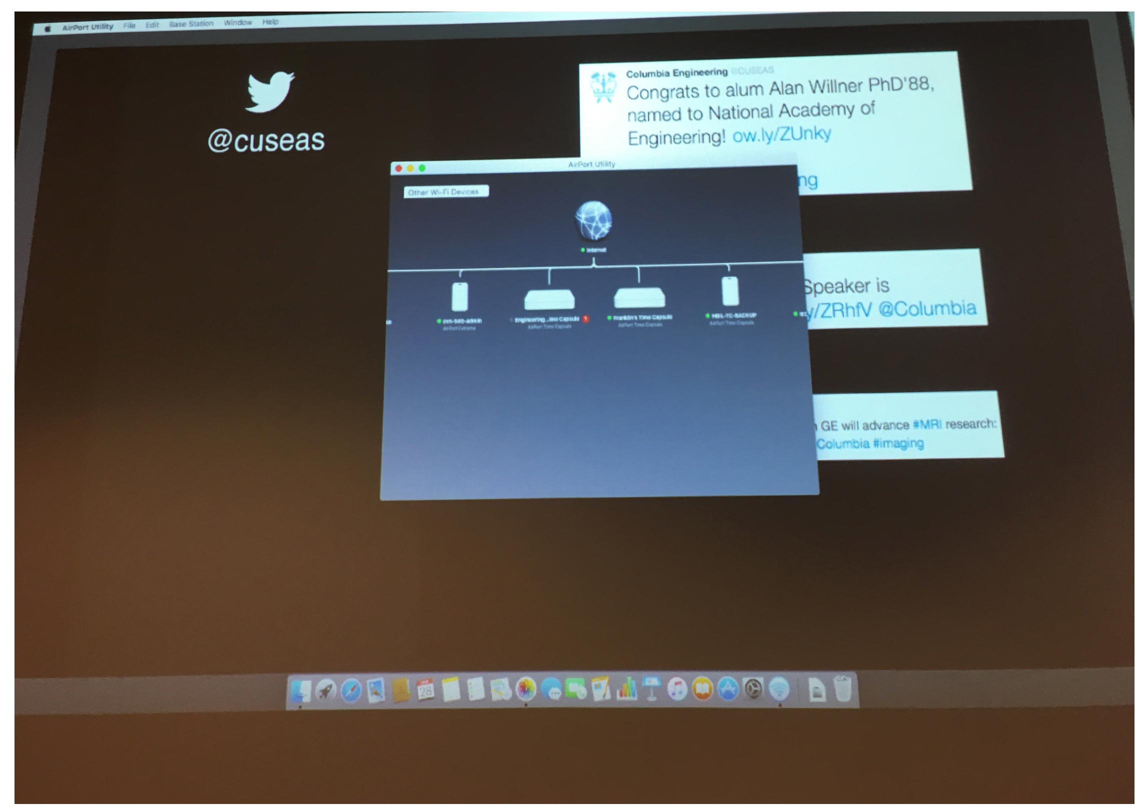
- Or will we dive deeper into the swamp?

The Internet of Things

- Inexperienced developers
- Platforms that aren't hardened
- Repetition of old mistakes
- No economic model for patches
- Devices that outlive their support lifetime
- Physical-world consequences


Service Status
DETOUR
GOOD SERVICE
DELAYS
Railroad
Bus
Subway
TURNSTYLE
OPEN DAILY
9 AM - 9 PM


Terminal C Departures
Departing to
Airline Flight Gate Sch Time Status
Albany
Asheville
Atlanta
Beijing
Bogota
Boston
Boston
Boston
Burlington
Burlington
Charleston
Charleston
Charlotte
Charlotte
Chicago O'Hare
Chicago O'Hare
Welcome to Newark Liberty International Airport
September 12 1:37 PM
Defibrillator


@cuseas
Columbia Engineering
Congrats to alum Alan Willner PhD'88, named to National Academy of Engineering! ow.ly/ZUnky

Buggy Code

- Buggy code will *not* go away

- Newer programming languages have fewer error-prone constructs than do C and C++

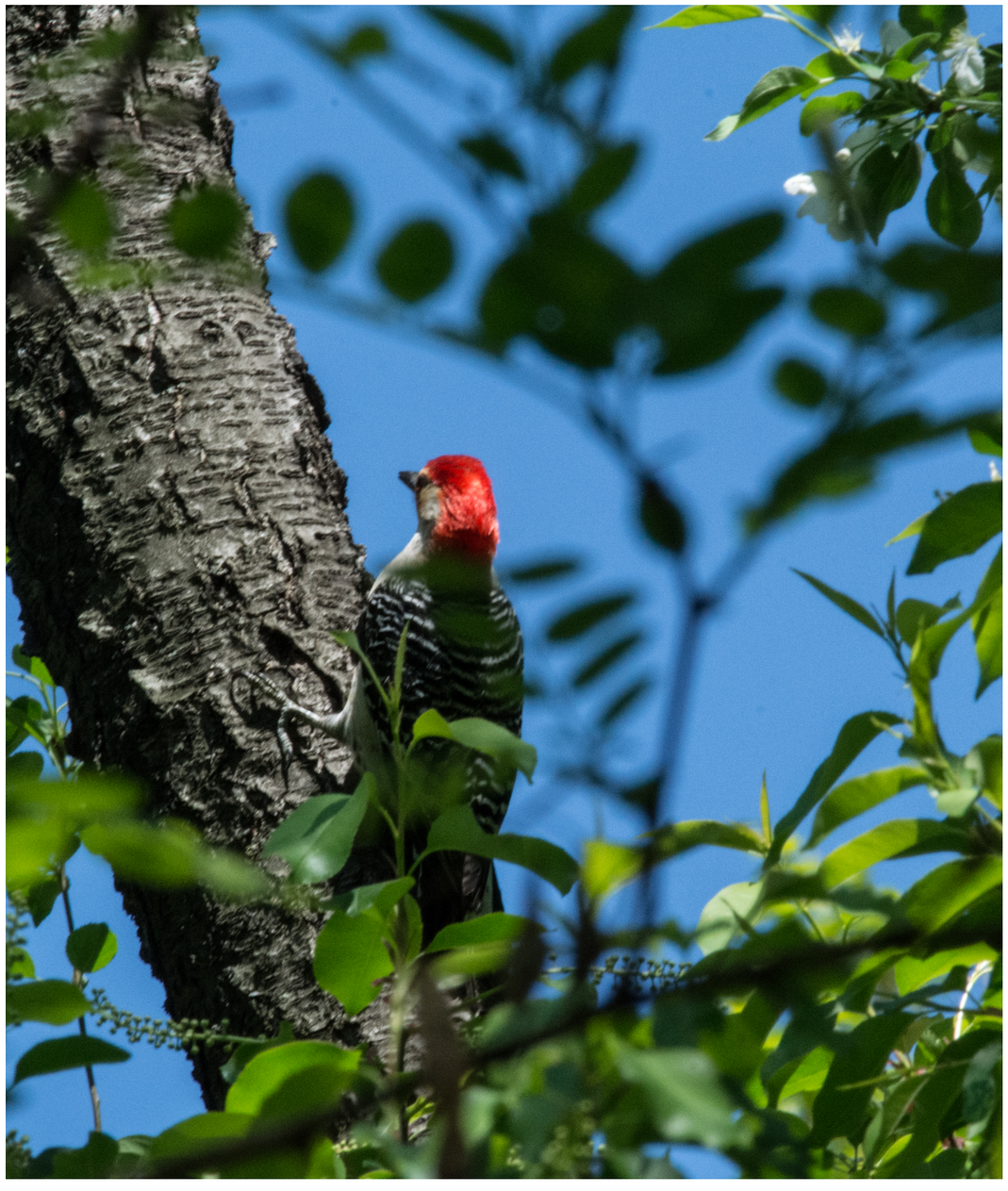
- But it's still possible to make higher-level mistakes

Better Ways to Cope with Bugs

- Formal methods are showing (some) promise

- Agile development methods allow rapid testing and deployment

- Large-scale systems involve many replicas—try out new versions on a few at a time



Defending the Net?

- Can the military defend the civilian parts of the net?
- Do we want it to try?
  - Privacy issues
  - What if it's an ordinary thief and not a foreign government?
- Besides, governments keep trying to mess with our crypto

A Look Back at 1988

- We know how to solve Stoll's password problem—password managers and 2FA—but deployment remains a challenge
- We have even more serious attackers going after data that does exist
- We can do patching, but not for IoT
- We do have information-sharing
- Buggy code still bedevils us
- Tracing and attribution are hard, though sometimes possible
- Morris' usability concerns remain real

Are we climbing out of the swamp?

Questions?

(these slides at https://www.cs.columbia.edu/~smb/talks/30-years.pdf)