INSE6120 – Cryptographic Protocols and Network Security

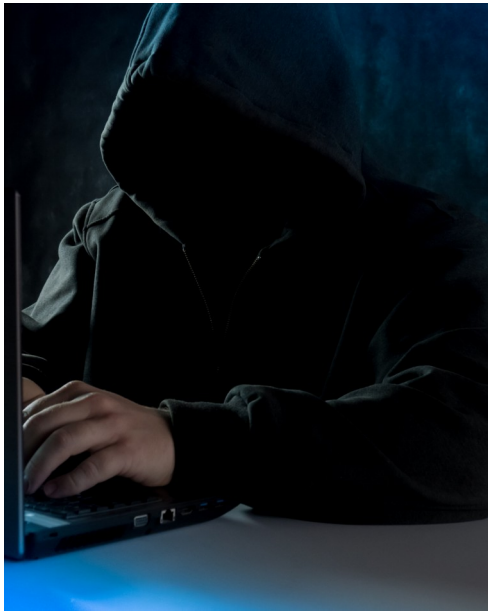
I. Pustogarov

Introduction to Security Context

Security

Goal

vs



Adversary

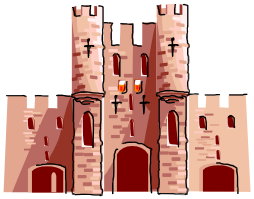
Book: Computer Security and the Internet: Tools and Jewels
http://people.scs.carleton.ca/~paulv/toolsjewels.html?

Security vs. correctness

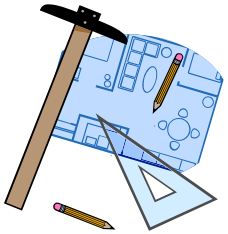
- System correctness: system satisfies specification
  - For reasonable input, get reasonable output
- System security: system properties preserved in the face of an attack
  - For unreasonable input, output not completely disastrous
- Main difference: active interference from adversary

Security stack

People

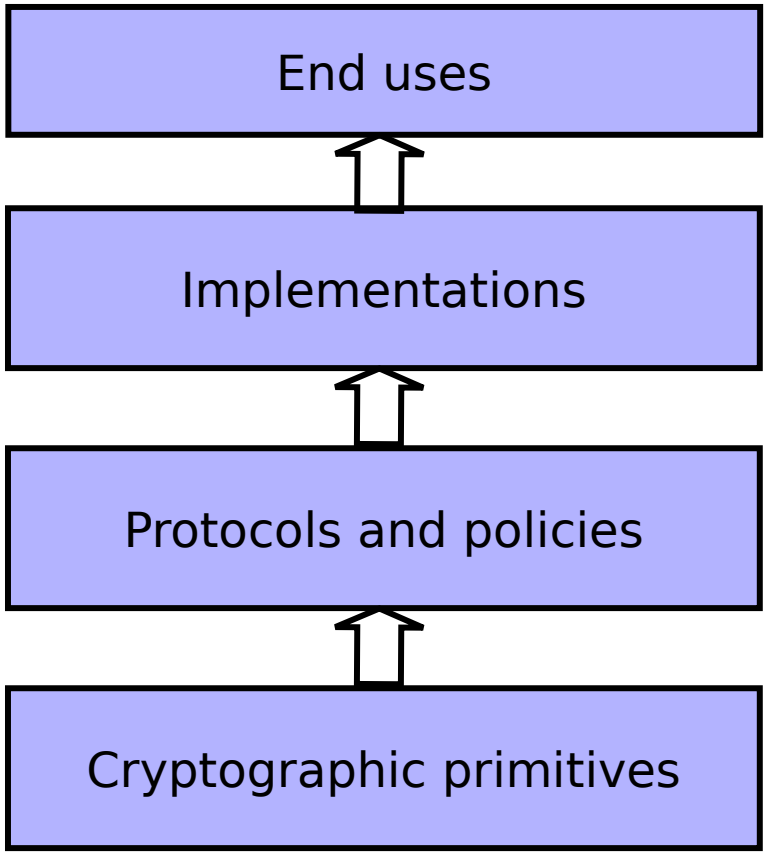


Systems



Blueprints

Building blocks



Password managers, company policies...

Firewalls, intrusion detection...

TLS, IPsec, access control...

RSA, DSS, SHA-1...

All defense mechanisms must work correctly and securely

Bad news for security and privacy…

- Security often **not** a primary consideration
  - Performance and usability take precedence
- “I’ve got nothing to hide”
- Feature-rich systems may be poorly understood
- Implementations are buggy
- Many attacks are **non-technical** in nature
  - Phishing, social engineering, etc.
  - Understand human aspects, phycology

Why security is hard...

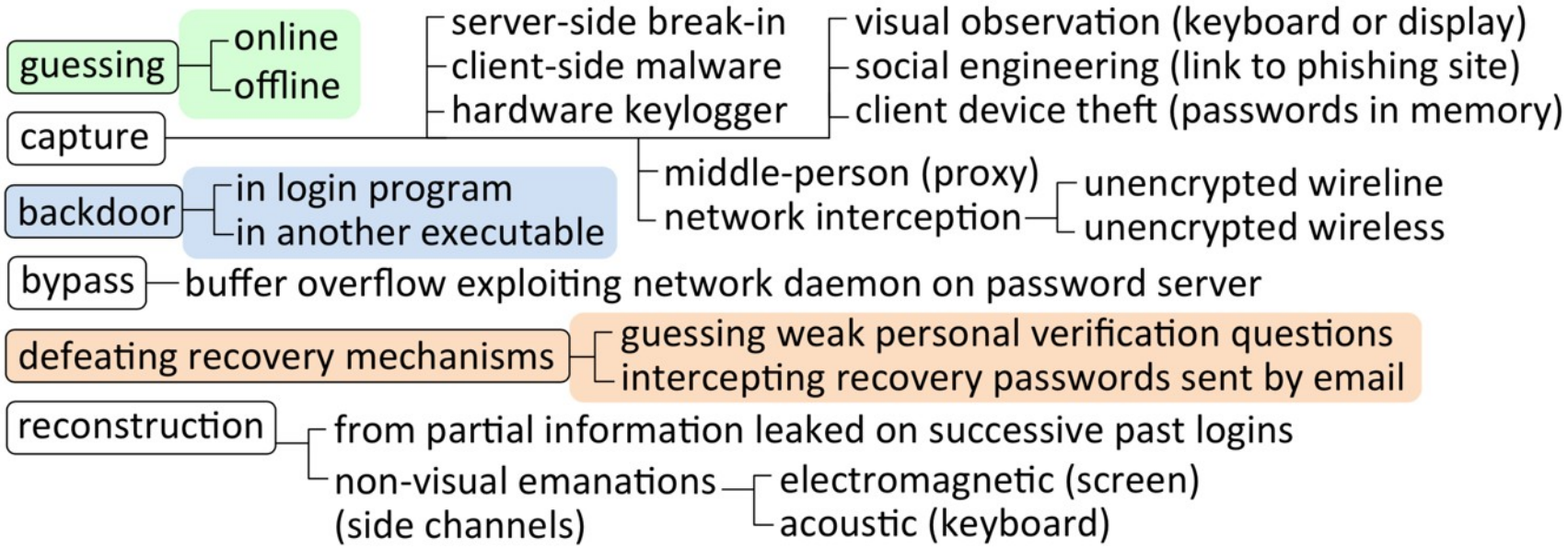
1. *intelligent, adaptive adversary*
2. *no rulebook*: attackers are not bound to any rules of play, while defenders typically follow protocol conventions, specifications, standards
3. *defender-attacker asymmetry*: attackers need find only one weak link to exploit, while defenders must defend all possible attack points.
4. *scale of attack*: the Internet enables attacks of great scale at little cost.
5. *universal connectivity*

6. *pace of technology evolution*: need continuous software upgrades and patches.
7. *software complexity*
8. *developer training and tools*
9. *interoperability and backwards compatibility*
10. *market economics and stakeholders*

11. *features beat security*
12. *low cost beats quality*
13. *missing context of danger and losses*

14. *managing secrets is difficult*
15. *user non-compliance (human factors)*: users bypass or undermine computer security mechanisms which impose inconveniences without visible direct benefits (in contrast: physical door locks are also inconvenient, but benefits are understood).
16. *error-inducing design (human factors)*: it is hard to design security mechanisms whose interfaces are intuitive to learn, distinguishable from interfaces presented by attackers, induce the desired human actions, and resist *social engineering*.
17. *non-expert users (human factors)*

Example attacks on password systems



Goal vs Adversary

- Policy (goals)
- Threat model (assumptions about the adversary)
- Mechanisms (how to achieve goals)

Any of these can go wrong

Security goals

- Confidentiality
  - Information is accessible only to authorized parties
- Integrity
  - Data, software or hardware remaining unaltered, except by authorized parties.
- Authorization
  - Accessible on by those approved by the resource owner or domain administrator.
- Availability
  - Remaining accessible for authorized use.
- Authentication
  - Provides assurances that the identity of a principal involved in a transaction is as asserted
- Accountability
  - The ability to identify principals responsible for past actions

Goal vs Adversary

- Policy (goals)
- Threat model (assumptions about the adversary)
- Mechanisms (how to achieve goals)

Any of these can go wrong

Goal vs Adversary

- Policy (goals)
- Threat model (assumptions about the adversary)
- Mechanisms (how to achieve goals in the presence of the attacker)

Any of these can go wrong

Examples

- Policy: Password recovery questions
  - This changes your policy. Before: you can log in if you know the password. Now: you can log in if you know the password OR if you know answers to security questions. This effectively weakens the security of your system. This happened to Sarah Palin. Be conservative when defining policy.
- Threat model: Human factors, weak encryption
  - People will choose simple predictable passwords, people will clink on random links to sites that are not original sites.
  - Certificate pinning
- Mechanisms: compromised hardware/software
  - Backdoor in router firmware (Zyxel)
  - NSA Clipper chip.

What code to trust?

- What code can we trust?
  - Consider "login" or "su" in Unix/Linux
- Is Windows/OSX binary reliable?
  - Does it send your password to someone?
  - Does it have backdoor for a “special” remote user?
- Can't trust the binary
  - Let’s use open-source: Ubuntu?
  - Check source code, or write your own (and compile)

What code to trust?

- Who wrote the compiler?
- Consider a Trojaned compiler:
  - Compiler inserts backdoor into binary only when you compile the login process' source code
- Ok, inspect the source code of the compiler… Looks good? Recompile the compiler!
  - Does this solve the problem?

A bugged compiler

```
compile(s)
char *s;
{
        ...
}
```

```
compile(s)
char *s;
{
        if(match(s, "pattern")) {
                compile("bug");
                return;
        }
        ...
}
```

```
compile(s)
char *s;
{
        if(match(s, "pattern1")) {
                compile ("bug1");
                return;
        }
        if(match(s, "pattern 2")) {
                compile ("bug 2");
                return;
        }
        ...
}
```

Moral

"*The moral is obvious. You can't trust code that you did not totally create yourself. (Especially code from companies that employ people like me.)*"

"*No amount of source-level verification or scrutiny will protect you from using untrusted code.*"

Today, Ken Thompson works as a distinguished engineer for Google

Is this a real problem?

- Yes, it happened to a Delphi compiler in 2009!
  - Win32/Induc-A

- http://www.microsoft.com/security/portal/threat/encyclopedia/entry.aspx?name=Virus%3AWin32%2FInduc.A#tab=2

Similar “attacks”

Visual Studio 2015 C++ Compiler Secretly Inserts Telemetry Code Into Binaries (infoq.com)

Microsoft

421

Posted by msmash on Friday June 10, 2016 @09:00AM from the Microsoft-things dept.

Reader edxwelch writes:

> Reddit user *sammiesdog* discovered recently that Visual Studio 2015 C++ compiler was inserting calls to a Microsoft telemetry function into binaries. "I compiled a simple program with only main(). When looking at the compiled binary in IDA, I see a call for *telemetry_main_invoke_trigger* and *telemetry_main_return_trigger*. I cannot find documentation for these calls, either on the web or in the options page," he wrote. Only after the discovery did Steve Carroll, the dev manager for Visual C++ admit to the "feature" and posted a workaround to remove it.

A Microsoft spokesperson confirmed the existence of this behavior to InfoQ, adding that the company wil be removing it in a future preview build. For those who wish to get rid of it, the blog writes:

> Users who have a copy of VS2015 Update 2 and wish to turn off the telemetry functionality currently being compiled into their code should add *notelemetry.obj* to their linker command line.

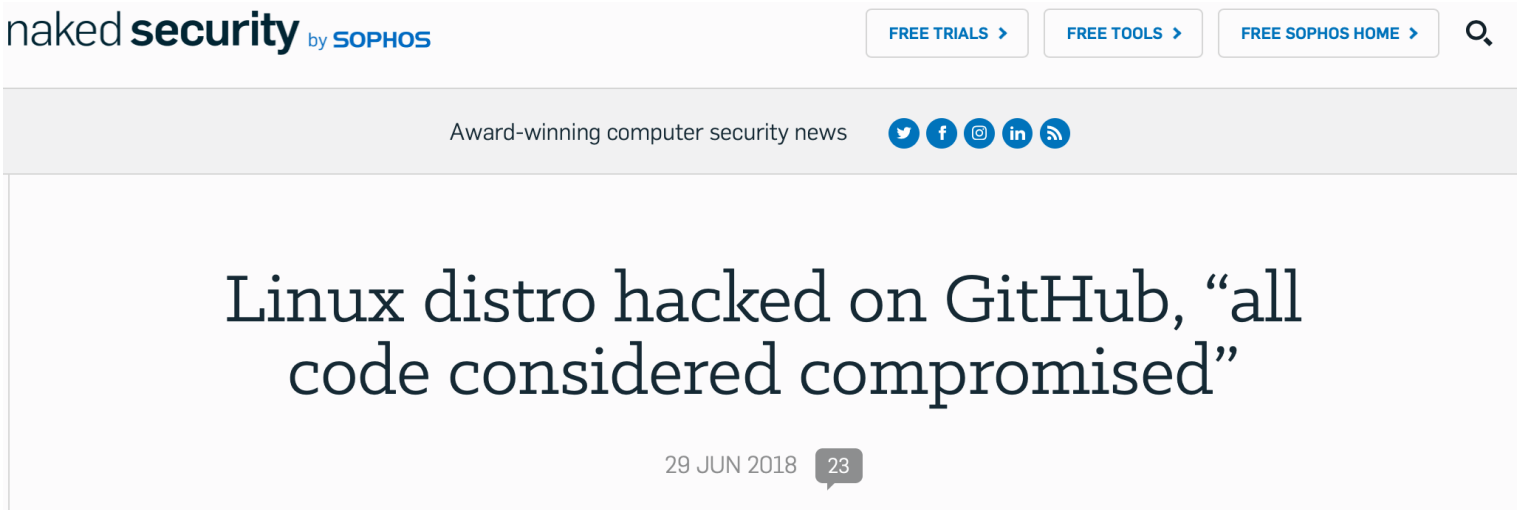
ars TECHNICA BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE STORE

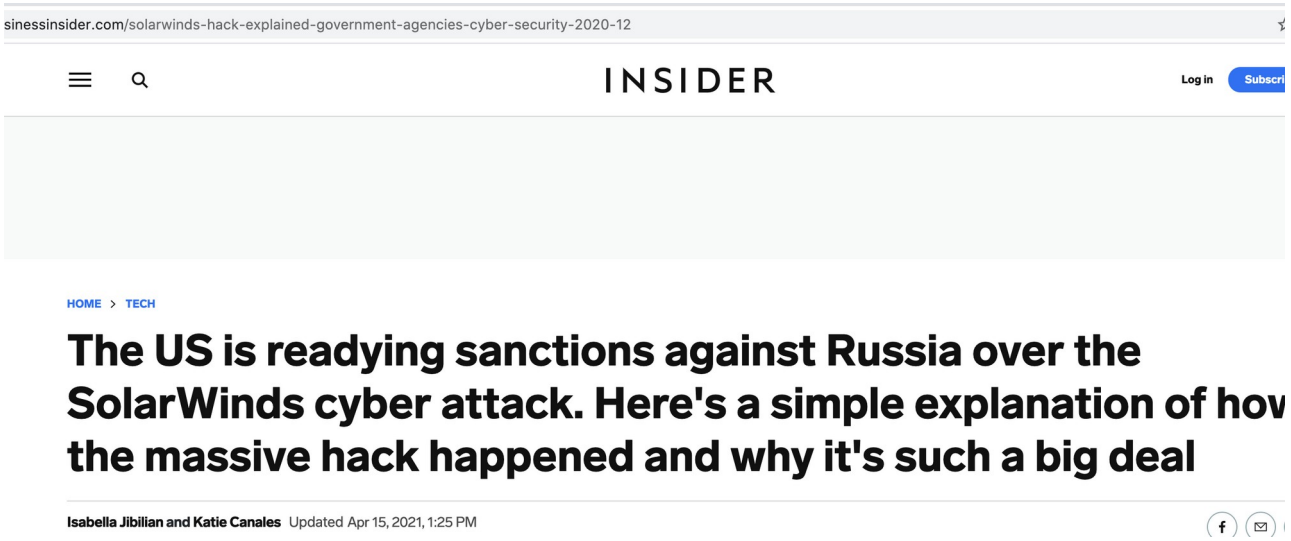
POISONING THE WELL —

Widely used open source software contained bitcoin-stealing backdoor

Malicious code that crept into event-stream JavaScript library went undetected for weeks.

DAN GOODIN - 11/26/2018, 5:55 PM


naked security by SOPHOS
FREE TRIALS
FREE TOOLS
FREE SOPHOS HOME
Award-winning computer security news
Linux distro hacked on GitHub, “all code considered compromised”
29 JUN 2018
23


sinessinsider.com/solarwinds-hack-explained-government-agencies-cyber-security-2020-12
INSIDER
Log in
HOME > TECH
The US is readying sanctions against Russia over the SolarWinds cyber attack. Here's a simple explanation of hov the massive hack happened and why it's such a big deal
Isabella Jibilian and Katie Canales Updated Apr 15, 2021, 1:25 PM

What else can go wrong?

- CPU
  - AES NI, random number generation
  - Other number theoretic attacks (e.g., bugged prime number generator)
- BIOS
- Remote management tools (Intel AMT)
- Firmware (from any device – e.g., Ethernet)
- Crypto engines

Optional reading

- Reflections on trusting trust
  - Communications of the ACM (August 1984)
  - https://www.ece.cmu.edu/~ganger/712.fall02/papers/p761-thompson.pdf
- Malicious cryptography (book, 2004)
  - Adam Young and Moti Yung
- Designing and implementing malicious hardware
  - USENIX LEET 2008
  - https://www.usenix.org/legacy/event/leet08/tech/full_papers/king/king.pdf
- Beware of Snake Oil (blog post, 1991)
  - http://www.philzimmermann.com/EN/essays/SnakeOil

BULLRUN Bottom Line

- Groundbreaking capabilities
- Extremely fragile
- Do not ask about or speculate on sources or methods underpinning BULLRUN successes
- Indoctrination required for access to secure COI

PTD "We penetrate targets' defences."

BULLRUN

- Covers the ability to defeat encryption used in specific network communications
- Includes multiple, extremely sensitive, sources and methods

PTD "We penetrate targets' defences."

Network Security Technologies

- Secure Sockets Layer/Transport Layer Security (SSL/TLS) (webmail)
- Secure Shell (SSH)
- Encrypted chat
- Virtual Private Networks (VPNs)
- Encrypted VoIP

PTD "We penetrate targets' defences."

Sensitivities

- Cryptanalytic capabilities
  - Are extremely difficult and costly to acquire
  - Require a long lead time
  - Depend on sensitive sources
  - Are very fragile
  - If lost, may never be regained
- The mere "fact of" a capability is very sensitive:
  - An adversary who knows *what* we can/cannot break is able to elude our capabilities even without knowing the technical details of *how* the capabilities work

PTD "We penetrate targets' defences."

Attacking randomness

- Dual Elliptic Curve Deterministic Random Bit Generator (Dual_EC_DRBG)

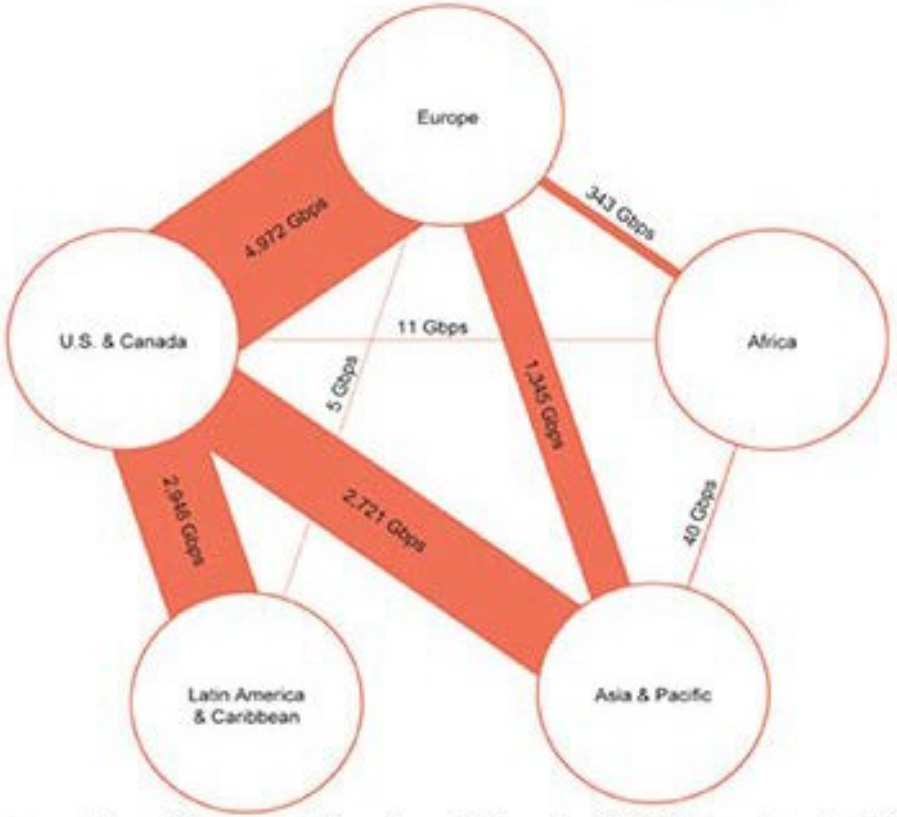
- Part of Bullrun
- Implemented in RSA BSAFE crypto library (since 2004)
- Became known in 2013 (RSA was paid US$ 1o million)

- Weaknesses are known to security community since 2007
- But still was included in Windows Vista (2007); removed in Windows 10.

(TS//SI//NF) Introduction

U.S. as World's Telecommunications Backbone

- Much of the world's communications flow through the U.S.
- A target's phone call, e-mail or chat will take the **cheapest** path, **not the physically most direct** path – you can't always predict the path.
- Your target's communications could easily be flowing into and through the U.S.



International Internet Regional Bandwidth Capacity in 2011

Source: Telegeography Research

(TS//SI//NF) PRISM Collection Details

Current Providers

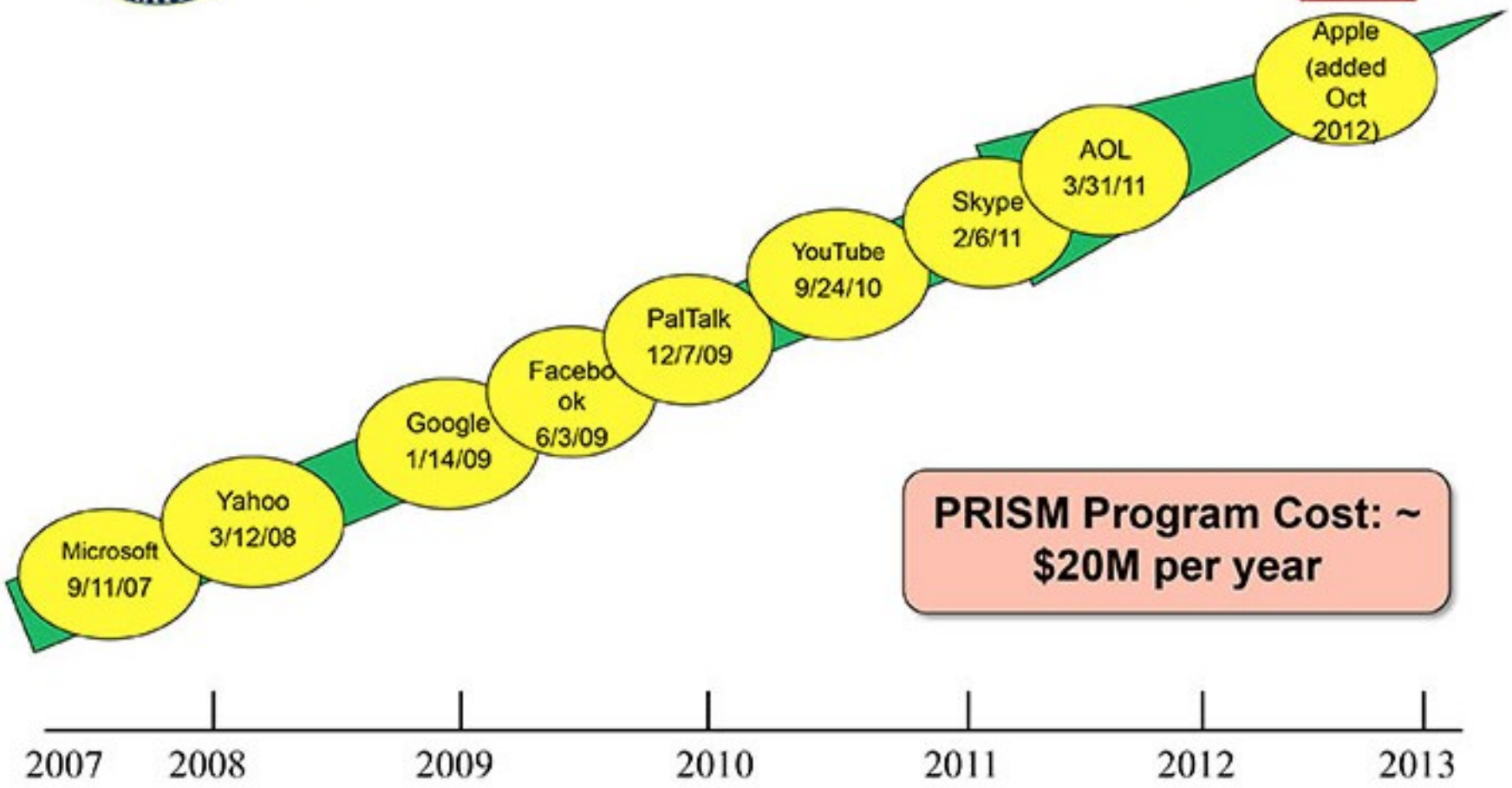
- Microsoft (Hotmail, etc.)
- Google
- Yahoo!
- Facebook
- PalTalk
- YouTube
- Skype
- AOL
- Apple

What Will You Receive in Collection (Surveillance and Stored Comms)? It varies by provider. In general:

- E-mail
- Chat – video, voice
- Videos
- Photos
- Stored data
- VoIP
- File transfers
- Video Conferencing
- Notifications of target activity – logins, etc.
- Online Social Networking details
- **Special Requests**

Complete list and details on PRISM web page:
Go PRISMFAA

(TS//SI//NF) Dates When PRISM Collection Began For Each Provider



(TS//SI//NF) What's Next

- Plan to add Dropbox as PRISM provider
- Want to add Cyber Threat Certification
- Expand collection services from existing providers
- Change UTT tasking system to allow tasking of phone numbers and sending one selector to multiple providers