ELEC5616 COMPUTER & NETWORK SECURITY

Lecture 15:

Overview of Network Security

THE VENDORS WILL SAVE YOU!

Since the early days of the Internet, security vendors have offered products to help keep you secure from the scary realm of the Internet:

Network Scanning Tools

Firewalls

Virtual Private Networks (VPN)

Intrusion Detection Systems (IDS)

Public Key Infrastructure (PKI)

Biometrics

Unified Threat Management (UTM) Appliances

EXCEPT...

Venders are more interested in:

- Making the software work out of the box than be secure out of the box
- Know customers can’t tell the difference between snake oil & real security

On top of that...

- Only hackers run network scanning tools
- Firewalls are walls designed to have holes in them
- Virtual Private Networks work on top of the Internet
- Intrusion Detection Systems only detect old attacks well, not new ones
- Public Key Infrastructure requires complex infrastructure and management
- Biometrics can be fooled and can’t sanely revoke or issue new keys

SO WHAT’S GOING ON?

We are building the digital world on a foundation of mud...

– Operating systems like Microsoft Windows

–The IP stack and 802.11/WEP

– Poor user protocols such as Telnet, FTP, RSH and HTTP

– Poor network protocols such BGP and DNS

– Poor network management protocols such as SNMP

– Poor security programming languages such as C

– Lack of proper infrastructure

– Lack of quality developers

– Poor design and programming practice

(design choices, implementation, assumptions, ...)

SCARY EXAMPLES: DDOS

July 1999:	The Computer Emergency Response Team (CERT) issues an advisory on Denial-of-service attacks
Sep 1999:	Packet Storm receives copies of DDoS tools
Nov 1999:	CERT warns of new class of attacks (DDoS) and tools in circulation at CISAC Information Warfare conference
Dec 1999:	Packet Storm receives latest copies of TFN and trinoo (DDoS attack tools)
Dec 1999:	Packet Storm release new tools and launches Storm Chaser 2000: Next Generation CyberDefence.

SCARY EXAMPLES: DDOS

Feb 7 2000:	**Yahoo - 3 hour outage**
Feb 8 2000:	**E-bay - 5 hour outage**
Feb 8 2000:	**buy.com - 4 hour outage - first day of IPO!**
Feb 8 2000:	**Amazon - 3:45 outage**
Feb 8 2000:	**CNN - 3:30 outage**
Feb 9 2000:	**ZDnet - 3:15 outage**
Feb 9 2000:	**E*trade - 2:45 outage**

The attack:

An amplified denial-of-service attack on the routers connecting these websites to the Internet

- Amplified Ping and SYN floods

SCARY EXAMPLES: DDOS

“Still no news on who is behind the concerted DoS attacks that so crippled America’s ability to buy Pokemon trading cards earlier this week.” - Need to Know www.ntk.net

“In a case like this, there is no Interpol, no Pinkerton’s that you can turn to for help” - Wall Street Journal

“Like a distributed pizza attack where you call every pizza shop in town and deliver them to your worst enemy” - Bruce Schneier

“A 16-year-old Montreal boy will be sentenced in April for his admitted guilt in paralyzing the Web sites of several U.S. companies, such as Yahoo, Amazon and eBay, while acting as the hacker Mafiaboy in February 2000.

The unidentified boy, who quit school and works a menial job, Thursday pleaded guilty to five counts of mischief, 51 counts of illegal access to a computer and one count of breach of bail conditions…” -- IDG

SCARY EXAMPLES: DDOS

Low Orbit Ion Canon: open-source DDoSing tools

– Used by Anonymous during their attacks on the Church of Scientology, Recording Industry Association of America (RIAA) and so on

DDoS for fun and/or profit

– CloudFlare is a company that exists purely to mitigate DDoS attacks

– Reddit DDoSed for no apparent reason during Boston bombing coverage

– DDoS attacks seen against SpamHaus (an anti-spam organisation) large enough to cause problems with the core layers of the Internet itself

SCARY EXAMPLES: CONFICKER

A botnet which had more computing power than any existing supercomputer

“Fought back” against computer security researches: multiple variations that repaired previous vulnerabilities and became more difficult to stop

Would you believe...

Conficker was first released three weeks after a fix was released to the public?

In a perfect world, how could a patched vulnerability cause an issue?

October 23, 2008: The patch for that issue (MS08-067) was released

January, 2009: 30% (or more) of Windows PCs remained unpatched

The vulnerability hit Windows 2000, Windows XP, Windows Vista, Windows Server 2003, Windows Server 2008...

SCARY EXAMPLES: CONFICKER

Paper: *Automatic patch-based exploit generation is possible*

Given a program **P** and a patched program **P’**

Generate an exploit for the unknown vulnerability present in **P**

The technique demonstrated in the paper...

Automatically generated exploits take **minutes**

Once an exploit is known, vulnerable machines are compromised in **minutes**

Distributing updates to everyone takes a theoretical minimum of **hours/days**

Distributing updates to everyone takes months once users become **involved**

SCARY EXAMPLES: “INTERNET CENSUS 2012”

Researcher wants to map the entirety of the Internet

It’s a computationally intensive task, so would likely be expensive...

Or...

1. Use the most naive hack possible (Telnet with credentials)
2. Filter through the specs of the owned devices (1.2 million) for a good subset
3. Select a subset (420k) and upload a specially crafted botnet to them

(minimal combinations such as root:root, admin:admin, no passwd, etc)

Good news: all data was released into the public domain for further study

Bad news: researcher created a 1.2 million device botnet for “fun research”

ECONOMIES OF THE INTERNET

Organisations need to understand that the Internet brings numerous advantages but equally large disadvantages as well...

– Information leakage (US govt documents => Wikileaks)

– Operationally exposing your internals to the world 24x7x365

– Increased risk associated with increased chance of compromise

– Ease at which attackers can launch attacks and get away with them

There's no Internet Police and no international jurisdiction

HOW CAN THIS HAPPEN?

- **Security is always catch-up**
  - Always a significant time delay between finding, reporting, advising and fixing problems
- **Security is usually reactive**
  - Security is perceived as a cost centre, not a profit centre
- **Homogenous nature of the Internet (monocultures)**
- **Heterogeneous nature of the Internet (interoperability)**
- **Political issues, export restrictions**
  - The government really doesn’t want you to be *that* secure
    - They want to raise the bar to their level
- **Patents**
- **Humans use the Internet**

HOW CAN THIS HAPPEN?
THE INTERNET IS A MONOCULTURE

Most blogs in the world run WordPress

- Bugs found every week and most installations not kept up to date

Most computers on the Internet run Windows

- With over 63,000 known bugs

Most nameservers run Bind

- “Buggy Internet Name Daemon” or “300,000 lines of bad code” (Bernstein)

Most web servers run Apache or Nginx

- Historically the buggiest UNIX program (vying with bind)

Software is almost always an exact copy…

HOW CAN THIS HAPPEN?
THE INTERNET IS A MONOCULTURE

The result..?

Software is almost always an exact copy…

If you can break into one version, you can break into them all

Thus, an attack against *any* of these tools will result in numerous owned machines

Even with only a small percentage of instances, that’s enough to wage digital war

(and people are kind enough to leave old versions running too!)

Bad news:

More and more devices (smart phones, tablets, watches, pacemakers, fridges) are coming online where security is not even on the feature list

MOST USERS HAVE *NO CLUE* ABOUT SECURITY

Gives power to those controlling their computer:
usually not the user but instead of the owner of a botnet

Allows companies with horrific security records to remain in business

Enable advertising snake oil security instead of developing the real thing

Are almost always the easiest way to break security:

- IT send an email reminding users to never to reveal passwords to anyone only to receive emails with passwords from people not reading the message

People may not be so calm if they actually understood the machine sitting in their living room could be actively working against them..?

COMMON BELIEFS ARE *WRONG*

The common security philosophy is that if you secure the perimeter, you can keep the insides soft and gooey (marshmallow)

This has always been a very bad assumption.

Nowadays it is even worse; your network is like Afghanistan:

- There is no border.
- You cannot trust anyone.
- There are simply too many ways into your network:
  - Internet connections (T1, cable, ADSL, frame relay …)
  - Every machine, including those ones set up by an intern four years ago running a system or tool that no-one has updated since
  - 802.11 wireless networks (the record is well over 15 kilometres with a good antenna and amplifier)
  - Third party connections (vendors, partners, clients …)

Users are 90% of the problem and they are already inside!

HOSTS ARE WEAK

When not weak due to bugs, are often weakly configured

Default configurations are usually insecure

Too many exposed services, exposed code

Programs are written poorly in bad languages

Hosts have users which further erode security

Administrators don’t know they’ve been hacked until well after the fact

In short there are too many ways to successfully attack hosts that can then be used to attack others:

- Remote exploit to gain access to the system
- Subversion of system to gain privileges
- Leverage access to other systems across the whole network
  - Through trust relationships, packet sniffing, keystroke logging etc.

AS TIME GOES ON, IT JUST GETS WORSE

Modern security still has all the problems of non-modern security:
the only difference is technology is more pervasive so we have more

OWASP put out a Top 10 Security Flaws list each year

The items in the top 10 do change, but not as the old issues go away, simply as new technology introduce new issues that no-one has had to deal with before

A1 - Injection
A2 - Broken Authentication and Session Management
A3 - Cross-Site Scripting (XSS)
A4 - Insecure Direct Object References
A5 - Security Misconfiguration
A6 - Sensitive Data Exposure
A7 - Missing Function Level Access Control
A8 - Cross-Site Request Forgery (CSRF)
A9 - Using Components with Known Vulnerabilities
A10 - Unvalidated Redirects and Forwards

NOTE...

None of these problems are stopped by encryption

None of these problems are stopped by firewalls

None of these problems are stopped by VPNs

None of these problems are stopped by biometrics

None of these problems are stopped by IDSs

None of these problems are stopped by PKIs

Some are a result of lack of strong authentication

Some are a result of bad programming

Some are a result of poor security administration

REFERENCES

OWASP Top 10 Security Flaws for 2013

– https://www.owasp.org/index.php/Top_10_2013-T10

Automatic Patch-Based Exploit Generation is Possible (2008)

– Brumley et al.