Introduction to Security & Secure Design

Luke Anderson

luke@lukeanderson.com.au

10th March 2016

University Of Sydney

Overview

1. Introduction

1.1 The World We Live In

1.2 Information Security Definition

1.3 Security Requirements

1.4 Assessing Risks

1.5 Aspects of Security

1.6 Trust

1.7 Summary

Introduction

We are entering a brave new world...

Did CypherPunk[1] Come True?

A decentralised “cryptocurrency”, based on an open source code, cryptography, and a peer to peer protocol, is developed by a pseudonymous developer “Satoshi Nakamoto”. It’s now worth over $6 billion.

In Japan, half a billion dollars of Bitcoin gets stolen from an online cryptocurrency exchange website, that was originally set up to sell *Magic: The Gathering* trading cards.

[1] https://en.wikipedia.org/wiki/Cypherpunk

Did CypherPunk Come True?

Purchasing drugs is safer and easier using a client for an encrypted global onion routing network on the darknet than hanging out in Kings Cross.

US & other allied nations claim they are under constant attack from a Chinese army of government sponsored hackers. China similarly claims it is under constant attack from US government hackers. e.g. Unit 61398

Did CypherPunk Come True?

In New Zealand, the world's top *Call of Duty* player starts a political party to revenge himself on the government, who then swoop in with black helicopters on his James Bond like villain's lair, impounding his sports cars with plates like HACKER and MAFIA.

Kim Dotcom

Did CypherPunk Come True?

Julian Assange, a character practically out of a William Gibson novel

A hacker with silver hair dressed in a suit flies around the world and uses disposable mobile phones and a laptop to spray corporate and government secrets across (relatively) uncontrollable cyberspace.

He then takes refuge in the Ecuadorian embassy after potentially being "framed" by the US government for sex crimes.

Did CypherPunk Come True?

A government sponsored software worm attacks a uranium enrichment facility in Iran

— Stuxnet

An nebulous, headless, anonymous collective of hackers, cypherpunks, kids, criminals and n00bs assemble at a moment's notice to hack websites, go on social crusades, or alternately catch criminals and deliver random acts of kindness.

Did CypherPunk Come True?

In Belize, the founder of a billion dollar antivirus company sets up a drug lab to do secret research into psychotropic drugs. While there, he secretly sets up his own intelligence network by handing out bugged laptops to government officials. They find out, and he is framed for murder. He disguises himself as a drunk old man with boot polish on his face and cotton wool in his nose and lurks out the front of his house while it is raided before fleeing the country; all while blogging furiously to the world.

John McAfee

Actual News Headlines

“Accelerometer used to log smartphone keystrokes”

“Stealing ATM pins with thermal cameras”

“How to turn a phone into a covert bugging device? Infect the printer”

“Tampered heart monitors, simulating failure in human organs”

“Github SSL replaced by self-signed certificate in China”

“Youth expelled from Montreal college after finding ‘sloppy coding’ that compromised security of 250,000 students personal data”

“Chinese Army Unit Is Seen as Tied to Hacking Against U.S.”

Actual News Headlines

“At Facebook, zero-day exploits bring war games drill to life”

“15 phone, 3 minutes all that’s needed to eavesdrop on GSM call”

“Shopping For Zero-Days: A Price List For Hackers’ Secret Software Exploits”

“U.S. House approves life sentences for hackers”

“Fingerprints can now be scanned from 2 meters away”

“Breakthrough silicon scanning discovers backdoor in military chip”

“Russian nuclear warheads armed by computer malfunction”

Everything is “smart” and connected

Vulnerable to “anyone on the network” now means “every computer on every network”.

Viruses have been found pre-installed (deliberately) in **digital photo frames**, **multi-function printers** and installed on **pet RFID tags**.

Photo frames were shipped by BestBuy with viruses pre-installed. They sniffed your home traffic, infected your computers and sent your credit card information to China.

Soon every product made by man will be networked and have a chip in it. RFID is already the new barcode. Garbage bins in London now have LCDs and are networked.

Everything is “smart” and connected

Everything now runs software... but **all software is buggy**; the bigger the software, the more buggy it is.

When a 25-GPU cluster can **crack every standard Windows password in less than 6 hours**, what password are you going to pick?

How are we going to protect all these things adequately? Who on earth is going to write antivirus for a photoframe?!? Certainly not John McAfee!

Our traditional models of how we think about security are breaking down. Fast.

… and now the bad news…

Nothing is secure in the digital world

The digital world behaves very differently to the physical world:

○ Everything digital is made of **bits**

○ Bits have no uniqueness

○ It's easy to copy bits perfectly

Therefore, if you have something, I can copy it.
e.g. Information, privileges, identity, photos, videos, software, digital money, secrets, etc.

Much of information security revolves around making it hard to copy bits.
This is like trying to make water not wet.

Matt's[2] Definition of Information Security

You spend X so that your opponent has to spend Y to do something you don't want them to do.
Y is rarely greater than X... and there are may opponents.

It's all a resources game:

○ Time

○ Money $$$

○ Computational Power (== time X $$$)

[2] Matt Barrie authored much of the content in these slides

Matt's Definition of Information Security

Implications:

- Given enough resources, someone will get in.
- Given enough attackers, someone will get in.
- Given enough time, someone will get in.

Thus, all systems can and will fail.

The trick is to raise the bar to an adequate level of (in)security for the resource you're trying to protect

Security Requirements

Everything you've been taught in engineering revolves around building dependable systems that always work.

Security engineering traditionally revolves around building dependable systems that work in the face of a world full of clever, malicious attackers (sometimes those hackers are a government).

Reality is complex and requirements differ between systems

Miss this and ...

"Dropbox Security Bug Made Passwords Optional For Four Hours"

Bank Security Requirements

Bookkeeping System

Core of a banks operations, most likely threat is internal staff.

Goal: Highest level of integrity

ATMs

Threats: Petty thieves, Money Mules[3]

Goals: Physical security, customer authentication

Internet Banking

Threats: Website hacks, customer phishing

Goals: Authentication, Availability

Vault/Safe

Threats: Physical break-ins

Goals: Physical integrity, difficult to transport/open

[3] e.g. In https://en.wikipedia.org/wiki/Carbanak

Military Security Requirements

Electronic Warfare Systems

Objective: Jam enemy radar without being jammed yourself.

Goals: Covertness, availability

Result: Countermeasures, countercountermeasures, etc.

Military Communications

Objective: Low probability of intercept (LPI).

Goals: Confidentiality, covertness, availability

Compartmentalisation

Objective: Separate those in simple admin roles from knowing about missile locations

Goals: Confidentiality, availability, resistance to traffic analysis?

Nuclear Weapons:

Objective: Keep weapon control within the chain of command.

Hospital Security Requirements

Web Based Technologies

Objective: Harness economies of the internet e.g. online reference books.

Goals: Integrity of data

Remote Access for Doctors

Goals: Authentication, confidentiality, availability

Patient record systems

Goal: "Nurses may only look at records of patients who have been in their ward the last 90 days"

Goal: Anonymity of records for research purposes

Paradigm shifts introduce new threats:

Shift to online drug databases means paper records are no longer kept.

Results in new threats to: availability (e.g. DoS) and integrity (e.g. tampering of information)

Why do systems fail?

Systems often fail because designers:

- Protect the **wrong things**
- Protect the **right things** in the **wrong way**
- Make **poor assumptions** about their systems
- Do not understand the **threat model properly**
- Fail to account for **paradigm shifts** (e.g. the Internet)
- Fail to understand the **scope** of their system

Focus on the important risks

	Extreme	High	Medium	Low	Negligible
Certain	1	1	2	3	4
Likely	1	2	3	4	5
Moderate	2	3	4	5	6
Unlikely	3	4	5	6	7
Rare	4	5	6	7	7

1. **Severe:** Must be managed by senior management with detailed plan
2. **High:** Detailed research and management required at senior levels
3. **Major:** Senior management attention is needed
4. **Significant:** Management responsibility must be specified
5. **Moderate:** Manage by specific monitoring or response procedures
6. **Low:** Manage by routine procedures
7. **Trivial:** Unlikely to need specific application of resources

Axioms of Information Security

- **Information security is a resource game**
- **All systems are buggy**
- **The bigger the system the more buggy it is**
- **Nothing works in isolation**
- **Humans are most often the weakest link**
- **It's a lot easier to break a system than to make it secure**

A system can be

A product or component

e.g. software program, cryptographic protocol, smart card

...plus infrastructure

e.g. PC, operating system, communications

...plus applications

e.g. web server, payroll system

...plus IT staff

...plus users and management

...plus customers and external users

...plus partners, vendors

...plus the law, the media, competitors, politicians, regulators...

It's a lot easier to break a system than to make it secure

Aspects of Security

Authenticity

Proof of a message's origin
Integrity plus freshness (i.e. message is not a replay)

Confidentiality

The ability to keep messages secret (for time t)

Integrity

Massages should not be able to be modified in transit
Attackers should not be able to substitute fakes

Non-repudiation

Cannot deny that a message was sent (related to authenticity)

Availability

Guarantee of quality of service (fault tolerance)

Covertness

Massage existence secrecy (related to anonymity)

Definitions

Secrecy

A technical term which refers to the effect of actions that limit access to information

Confidentiality

An obligation to protect someone or some organisation's secrets

Privacy

The ability and/or right to protect the personal secrets of you or your family; including invasions of your personal space

Privacy does not extend to corporations

Anonymity

The ability/desire to keep message source/destination confidential

Passive Attacks

Those that do not involve the modification or fabrication of data.

Examples include: eavesdropping on communications. Interception:

- An unauthorised party gains access to an asset
- Release of message contents: an attack on **confidentiality**
- Traffic analysis: an attack on **covertness**

Active Attacks

Fabrication:

An unauthorised party inserts counterfeit objects into the system

Examples include masquerading as an entity to gain access to the system

An attack on **authenticity**

Interruption:

An asset of the system is destroyed or becomes unavailable or unusable

Examples include denial-of-service attacks on networks

An attack on **availability**

Modification:

An unauthorised party not only gains access to but tampers with an asset

Examples include changing values in a data file or a virus

An attack on **integrity**

Trust

A **trusted** system is one whose failure can break security policy.

A **trustworthy** system is one which won't fail.

A NSA employee caught selling US nuclear secrets to a foreign diplomat is **trusted** but not **trustworthy**.

In information security **trust is your enemy**.

Trust is your enemy

You cannot trust software or vendors

They won't tell you their software is broken

They won't fix it if you tell them

You cannot trust the Internet nor its protocols

It's built from broken pieces

It's a monoculture; something breaks = everything breaks

It was designed to work, not be secure

You cannot trust managers

They don't want to be laggards nor leaders

Security is a cost centre, not a profit centre!

You cannot trust the government

They only want to raise the resource game to their level

Trust is your enemy

You cannot trust your employees or users

They are going to pick poor passwords

They are going to mess up the configuration and try to hack in

They account for 90% of security problems

You cannot trust your peers

They are as bad as you

You cannot trust algorithms nor curves

Moore's law does not keep yesterday's secrets

Tomorrow they might figure out how to factor large numbers

Tomorrow they might build a quantum computer

You cannot trust the security community

They are going to ridicule you when they find a problem

They are going to tell the whole world about it

You cannot trust information security

It's always going to be easier to break knees than break codes

Trust is your enemy

You cannot trust yourself

You are human

One day you will screw up

Tenet of information security

Security through obscurity does not work

Full disclosure of the mechanisms of security algorithms and systems (except secret key material) is the only policy that works.

Kirchhoff's Principle: For a system to be truly secure, all secrecy must reside in the key.

If the algorithms are known but cannot be broken, the system is a good system.

If an algorithm is secret and no-one has looked at it, **nothing** can be said for its security

Morals of the story

- Nothing is perfectly secure
- Information security is a resource game
- Nothing works in isolation
- Know your system
- Know your threat model
- Trust is your enemy
- All systems can and will fail
- Humans are usually the weakest link
- Attackers often know more about your system than you do