



**Carnegie
Mellon
University**

 Electrical & Computer
ENGINEERING

Prof. Philip Koopman

Security Plans

“Plans are nothing;
planning is everything.”

– Dwight D. Eisenhower

■ Anti-Patterns:

- No security plan
- Expecting “perfect” security
- Unrealistic security assumptions

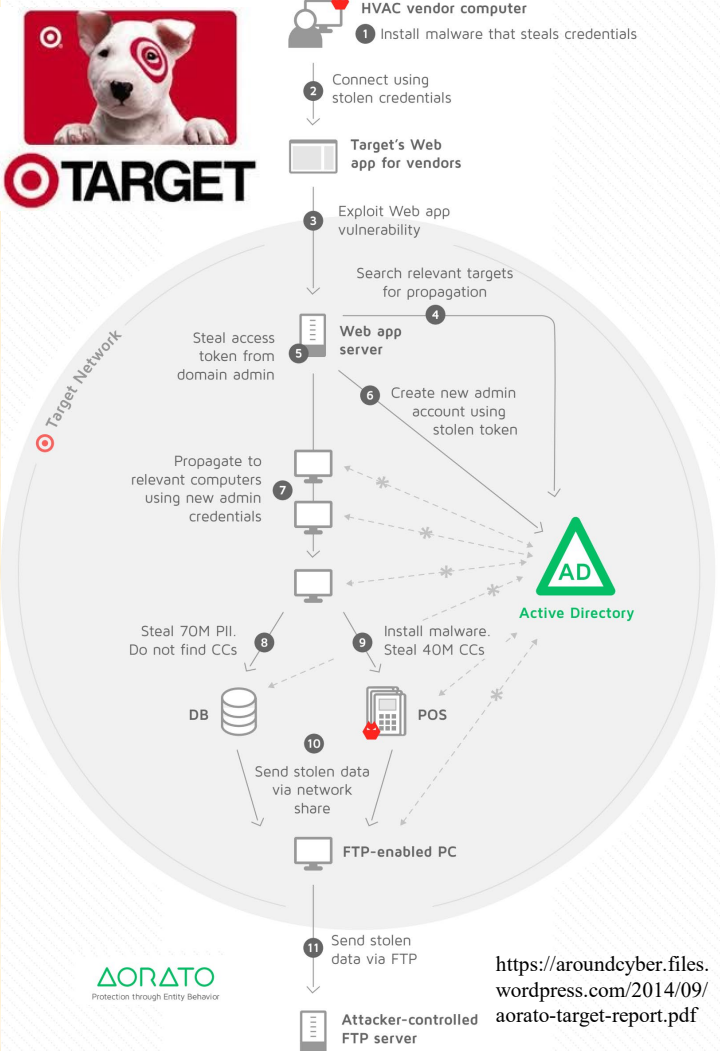
■ Plan for security: Why, What, How

- Why is security important to your system?
 - Which aspects matter; which don't?
- What types of attacks do you expect?
 - Which attackers present the most risk
- How can you mitigate the risk?
 - How do you know you succeeded?

Elements of a Security Plan

- Requirements
- Threats
- Vulnerabilities
- Mitigation
- Validation

Target Attack



Wake-up call to US credit card system

- Credit card skim at Point of Sale Terminals
 - Personal info (PII) also stolen from database
- \$150M - \$300+M costs to Target
 - Additional costs bank write-off costs
- Accelerated move to chip security in US
- Ultimately, cost CEO of Target his job

Complex attack

- Many steps, 3 month timeframe
- Generated multiple security alerts
- 11 GB of data, 40M credit cards, 70M other PII
 - <http://www.zdnet.com/article/the-target-breach-two-years-later/>
 - https://www.commerce.senate.gov/public/_cache/files/24d3c229-4f2f-405d-b8db-a3a67f183883/23E30AA955B5C00FE57CFD709621592C.2014-0325-target-kill-chain-analysis.pdf
 - <https://aroundcyber.files.wordpress.com/2014/09/aorato-target-report.pdf>

Security Requirements: Why Secure?



- **Security:** Protecting against unauthorized access, use, disclosure, disruption, modification, and destruction

[<http://csrc.nist.gov/drivers/documents/FISMA-final.pdf>]

→ (“CIA” properties)

- **Confidentiality:** is information released?

- Information is kept secret (**Secrecy**) → Encryption
- Activities can't be associated with an individual or other entity (**Privacy**)

- **Integrity:** have changes been made?

- Unauthorized data alteration or destruction is detected or prevented (**Data Integrity**)
- Changes to system state made by authorized entities (**Authentication; Non-Repudiation**)

- **Availability:** is it working?

- Services are available when requested in a timely manner (opposite is **Denial of Service**)

- Which of these are most important to you?

- Often **Data Integrity** matters more than **Secrecy**

Threats: What Types Of Attacks?

PLC

A discrete digital computer used for automation of typically industrial electromechanical processes

DCS

A hierarchical control system with distributed elements across a facility via communications technologies

SCADA

A system for remote monitoring and control of infrastructure, typically over slow speed, long distance communication channels

IED

A microprocessor-based controllers device for controlling power system equipment, such as circuit breakers, transformers and capacitor banks.

WHICH ARE USED IN MANY INDUSTRIES



PETROCHEMICALS



TRANSPORTATION



ENERGY



MANUFACTURING

TO CONTROL MANY PROCESSES



PORT AUTOMATION



FOOD AND BEVERAGE MANUFACTURING



POWER GENERATION AND TRANSMISSION



PHARMACEUTICAL MANUFACTURING



OIL EXPLORATION AND PRODUCTION



RANSOMWARE

Threat actor locks control of crane, trapping operator; unions halt work until cranes are safe

Operational halt



REMOTE ACCESS TOOL

Threat actor sends commands, destroying sensitive equipment

Loss of capital investment



MALICIOUS INSIDER

Insider removes over-speed protection on turbine causing significant damage

Diminished generation capacity



SUPPLY CHAIN COMPROMISE

Compromise of supply chain results in production of defective batch of medication

DoJ initiates criminal investigation



DESTRUCTIVE MALWARE

Malware alters parameters on a semi-submersible rigs station keeping system causing collision

Damage to rig and reputation

INCIDENT COUNT (FY)

<http://goo.gl/AFmXIY>

FY 2010 2011 2012 2013 2014 2015

140

197

257

245

295

ATTACKS ON ICS CAN CUT DEEP INTO YOUR BOTTOM LINE

Who:

- Nation State
- Organized group
- Script Kiddie
- Casual abuser
- Insider

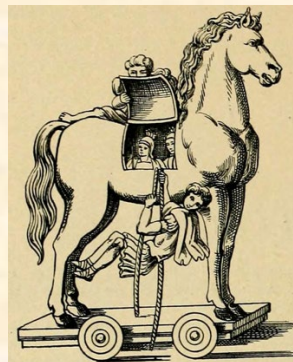
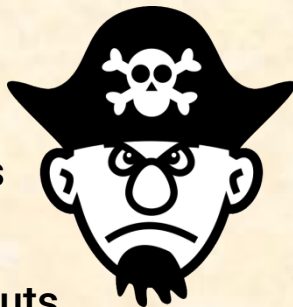


Motivation: \$\$\$\$(?)

- Politics & economics
- Surveillance
- Denial of Service
- Ransom
- Fame & Notoriety
- Just for the lulz



Vulnerabilities: Weaknesses & Tactics



<http://goo.gl/n3spNF>

■ Resource management

- Buffer overflow
- Garbage collection; timeouts

■ Improper input validation

- Crash due to exceptional inputs
- Executing input as a command (SQL)

■ Improper Authentication

- Bad configuration; information exposure
- Delayed message playback
- Man-in-the-middle attacks

■ Bad Cryptography

- Weak algorithms
- Non-random keys
- Insecure protocols

■ Back door

- e.g. Not-so-secret factory password

■ Brute Force

- Dictionary of common passwords
- Random parameters to see what breaks

■ Phishing

- Trick victim into installing malware
- Trick victim into revealing info

■ Worms that self-propagate

- USB flash drive infections

■ Trojan Horse

- e.g., Fake software update

■ Side Channel

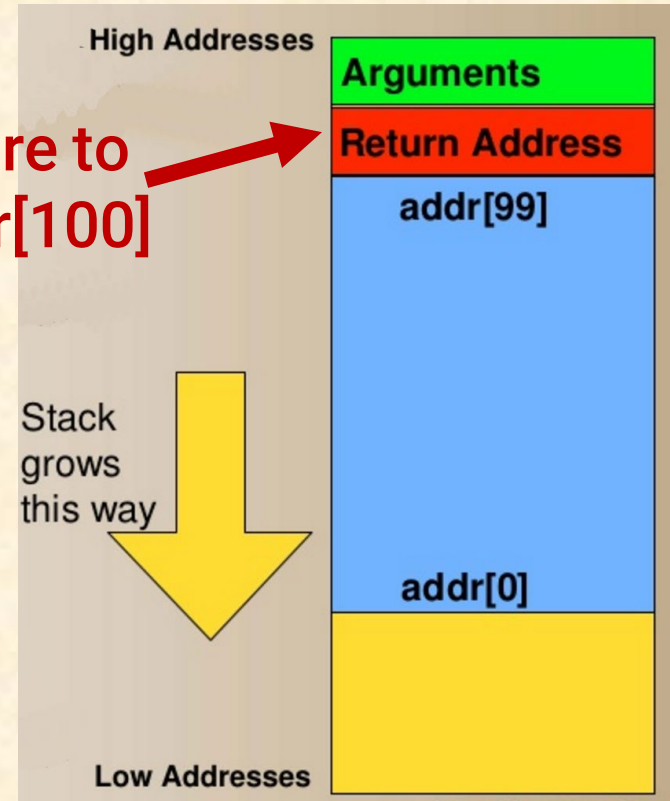
- Power consumption reveals secret keys

Mitigation: Good Security Design

- Pay attention to resource management
 - Especially, avoid buffer overflows
- Validate all inputs
 - Check size & data validity
- Ensure authentication is done properly
 - No master password!
 - Configuration, permissions
- Use strong crypto & proven protocols
 - If you write it yourself, probably it's broken
 - Principle of Least Privilege (don't run as root)
- Secure boot
 - Authenticate updates
- Avoid "security via obscurity"
 - Proprietary network protocols are NOT secure
 - Rely primarily on a unique, per-system crypto key



Store to
addr[100]



<http://goo.gl/2ta7Qv>

Validation: How Do You Know You're Secure?



■ Check code quality

- Buggy code is probably insecure
- Static analysis, stack checker tool, dynamic checkers
- Peer reviews (e.g., follow Cert C 98 Coding Standard)
 - <http://www.open-std.org/jtc1/sc22/wg14/www/docs/n1255.pdf>

■ Penetration testing

- Mostly helps deal with known exploits

■ Penetration analysis

- Hire a “red team” to attempt to penetrate system

■ Plan to be imperfect

- What if assumptions are violated (e.g., customer firewall)
- Defense in depth
- Plan for security patches (on-line updates are a vulnerability too!)

Elements of a Security Plan

A written document with plan for making your product secure:

■ Security Requirements

- What does security mean to your product?

■ Threat Model

- Who is likely to attack you, and why?
- What is the operational environment?

■ Vulnerabilities

- What are the paths of attack on your system?
- What are the likely objectives of an attacker?

■ Mitigation Strategies

- Rank probably & severity (to the degree you can) and prioritize risk (e.g., with a Risk Table)
- How will you address each risk, including ones that appear after deployment?

■ Validation Strategies

- Does your mitigation really work?





IN THE RUSH TO CLEAN UP THE DEBIAN-OPENSSL FIASCO, A NUMBER OF OTHER MAJOR SECURITY HOLES HAVE BEEN UNCOVERED:

AFFECTED SYSTEM SECURITY PROBLEM

FEDORA CORE	VULNERABLE TO CERTAIN DECODER RINGS
XANDROS (EEE PC)	GIVES ROOT ACCESS IF ASKED IN STERN VOICE
GENTOO	VULNERABLE TO FLATTERY
OLPC OS	VULNERABLE TO JEFF GOLDBLUM'S POWERBOOK
SLACKWARE	GIVES ROOT ACCESS IF USER SAYS ELVISH WORD FOR "FRIEND"
UBUNTU	TURNS OUT DISTRO IS ACTUALLY JUST WINDOWS VISTA WITH A FEW CUSTOM THEMES

The "Debian-OpenSSL fiasco" was a major security problem discovered in the Debian Linux distribution and its version of the cryptographic library called OpenSSL. With just a tiny change in the software, which was intended to have no effect on security, its random number generator was completely crippled, as was the security of all cryptographic keys generated by the system. The problem was created when a Debian developer removed one line of code which was crucial, even though it could seem like it did nothing useful.

https://www.explainxkcd.com/wiki/index.php/424:_Security_Holes