CS 491/531
SECURITY IN CYBER-PHYSICAL SYSTEMS

Lecture 17: Review

DR. ABDULLAH AYDEGER

LOCATION: ENGINEERING A 409F

EMAIL: AYDEGER@CS.SIU.EDU
Outline

Common ICS Attack Targets

Risk and Vulnerabilities in ICS
Common ICS Attack Targets
Safety in ICS

Most ICS employ automated safety mechanisms to avoid catastrophic failures

- Would it solve the problem of critical consequences of cyber incidents?
- Many of these safety control mechanisms utilize same messaging and control protocols used by ICS operational processes
  - Some of the mechanisms are even integrated to protocol itself

Safety systems are very significant

- However, security will not be provided
Common Industrial Targets

Engineering Workstations

SCADA server/historian

Protocols
## Examples of Attack Targets

<table>
<thead>
<tr>
<th>Target</th>
<th>Attack Vectors</th>
<th>Attack Methods</th>
<th>Consequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access control system</td>
<td>-Identification cards</td>
<td>-RFID Spoofing</td>
<td>-Unauthorized physical access or access to ICS assets</td>
</tr>
<tr>
<td></td>
<td>-Business network client</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-Database integration communication channel</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-Remote user access</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Historian</td>
<td>-Business network client</td>
<td>-Installation of malware via unvalidated software</td>
<td>-Manipulation of process</td>
</tr>
<tr>
<td></td>
<td>-Database integration communication channel</td>
<td>-Database injection</td>
<td>-Credential leakage (business or control)</td>
</tr>
<tr>
<td></td>
<td>-Remote user access</td>
<td>-Insecure communication protocols</td>
<td>-Unauthorized access to ICS assets</td>
</tr>
<tr>
<td>Master or slave devices</td>
<td>-Unvalidated firmware</td>
<td>-Distribution of malicious firmware</td>
<td>-Delay system</td>
</tr>
<tr>
<td></td>
<td>-Weak communication problems</td>
<td>-Exploitation of INP</td>
<td>-Mechanical damage</td>
</tr>
<tr>
<td></td>
<td>-No authentication (or weak) for “write” operations</td>
<td>-Buffer overflow</td>
<td>-Suppression of critical status/alarms - safety</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Examples of Attack Targets

<table>
<thead>
<tr>
<th>Target</th>
<th>Attack Vectors</th>
<th>Attack Methods</th>
<th>Consequences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operator workstation (HMI)</td>
<td>-Operational applications</td>
<td>-Installation of malware via USB</td>
<td>-Plant shutdown</td>
</tr>
<tr>
<td></td>
<td>-USB</td>
<td>-Authorization of ICS HMI functions without sufficient access control mechanisms</td>
<td>-Product quality</td>
</tr>
<tr>
<td></td>
<td>-Control network</td>
<td></td>
<td>-Credential leak (control)</td>
</tr>
<tr>
<td>Telecommunication systems</td>
<td>-Public Key infrastructure</td>
<td>-Disclosure of private key via external compromise</td>
<td>-Credential leak (control)</td>
</tr>
<tr>
<td></td>
<td>-Internet visibility</td>
<td>-Exploitation of device connected to public networks</td>
<td>-Information leak</td>
</tr>
<tr>
<td></td>
<td></td>
<td>-Network access through unmonitored access points</td>
<td>-Unauthorized remote access</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-Command and control</td>
</tr>
<tr>
<td>ICS Technician</td>
<td>-Social engineering</td>
<td>-Transmission of malware on control network via unauthorized connection</td>
<td>-Plant shutdown/delay</td>
</tr>
<tr>
<td></td>
<td>-Email attachments</td>
<td>-Exploitation of applications with administrative rights</td>
<td>-Mechanical sabotage</td>
</tr>
<tr>
<td></td>
<td>-File shares</td>
<td></td>
<td>-Modification of status messages</td>
</tr>
</tbody>
</table>
Common Attack Methods

MiTM:

- Intercept traffic between two target systems
  - Inject new traffic
- Works only if the connection lacks encryption and authentication
  - Even if auth or encryption is used -> listen for key exchanges and interrupt with your own key
    - Not that simple due to long period of time to re-establish communication
- Most INP authenticate in cleartext
  - Some don’t even have authentication
Common Attack Methods

DoS:

- In IT system response is slowed down until DoS is resolved, in Industrial network system shutdown is possible
  - A few examples: loss of communication with device, crashing particular services within device
- Loss of communication may lead to “Loss of Control” or “Loss of View”
  - This will result the system to move “safe state”
Common Attack Methods

Compromising HMI (Engineering Work Station):

- Obtain command and control of ICS
- Exploit device vulnerability and install remote access to the console
  - Finding vulnerabilities by penetration testing
- No knowledge of industrial protocols needed (or no ladder logic, etc.)
  - Only interpret GUI to change values within a console
Examples of ICS Incidents

**STUXNET**

- It was the first virus to include code to attack Supervisory Control and Data Acquisition (SCADA) systems (infection started 2007)
  - Poster child of industrial malware
- It is (was at the time of its discovery) the most complicated virus / worm ever discovered
- Average viruses are about 10k bytes in size
  - Stuxnet was 500 KB (and no graphics)
- It is unusual for a virus to contain one zero-day vulnerability. Stuxnet had 4
- Stuxnet also acted like a rootkit – hiding its actions and its presence
# Lessons learned from Stuxnet

<table>
<thead>
<tr>
<th>Previous Belief</th>
<th>Lesson Learned</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control systems can be isolated from other networks, eliminate risk of cyber incident</td>
<td>They are still subject to human who can use USB</td>
</tr>
<tr>
<td>PLC and RTUs don’t run modern OS, don’t have necessary attack surface</td>
<td>PLCs can be affected and have been affected by malware</td>
</tr>
<tr>
<td>Firewall/IDS are sufficient</td>
<td>Blacklisting based defense is not sufficient due to zero-day vulnerabilities, whitelist defenses should be considered against unknown exploits</td>
</tr>
</tbody>
</table>
Adobe Exploits

Example of recent shift in attack paradigm from lower-level protocol and OS to application layer

How this works?

◦ PDF attached to email from trusted source (spear phishing)
◦ Distribution of manuals/reference materials using PDF
◦ PDF feature of “Launch action” to run executable embedded within PDF
◦ Available in Kali Linux and Social Engineering Toolkit (SET)

https://github.com/trustedsec/social-engineer-toolkit
How to proceed if infection detected

Not to clean it directly

◦ May have subsequent levels of infection that exist (staying idle and undetected)
◦ Valuable info such as infection path, other compromised hosts

First step to isolate the infected host

Collect as much as possible forensics data

◦ System logs, network traffic, memory analysis data

Sandbox the infected device/system
Risk and Vulnerabilities in ICS
Statistics of ICS Incidents

80% impacting ICS are “unintentional”

- Only 35% from outsider
- Insider + unintentional is a big concern

Embedded devices and network appliances were targeted 34%

- Windows-based ICS and enterprise hosts 66%

These numbers would help to understand risks that should be prioritized

https://scadahacker.com/
Definition of Keywords

An *asset* is what we’re trying to protect

A *threat* is what we’re trying to protect against

A *vulnerability* is a weakness or gap in our protection efforts

*Risk* is the intersection of assets, threats, and vulnerabilities

Risk is a function of threats exploiting vulnerabilities to obtain, damage or destroy assets. Thus, threats (actual, conceptual, or inherent) may exist, but if there are no vulnerabilities then there is little/no risk. Similarly, you can have a vulnerability, but if you have no threat, then you have little/no risk.

- *Asset + Threat + Vulnerability = Risk*
What is Risk?

ISO defines: “potential that a given threat will exploit vulnerabilities of asset”

Risk is a function of:

- The likelihood of a given “Threat Event”
- Exercising particular potential vulnerability of an asset
- Consequences that impact operation of the asset

Threat Event:

- Threat source and actor to carry out the event
- Threat vector to initiate the event
- Threat target which the event attacks
Flowchart of Assessing Risks to ICS

1. System characterization
2. Identify threat events, sources
3. Identify vulnerabilities
4. Classify risk
5. Determine risk
6. Measure likelihood and impact
7. Mitigation plan
8. Vulnerability ranking
9. Control recommendation
Security Testing in ICS

Penetration testing in ICS?

- Requires non-production test environment

Security Audits

- Test particular system against specific set of policies, procedures or regulations
  - It usually mean known threats
  - Do not uncover unexpected or latest vulnerabilities

Security and Vulnerability Assessment

- To look at the entire solution for the system
  - This means each ICS system and subsystem/network infrastructure and so on
Theoretical Tests

Industrial systems operational integrity is critical to allow test to be run, even small risk tests can disrupt the integrity (time requirements, etc.)

- Leads to theoretical tests

Standardized method of completing questionnaire

- Like interview

Dept of Homeland Sec (DHS) ICS Cyber Emergency Response Team (ICS-CERT) developed Cyber Security Evaluation tool (CSET) for offline tests

- Security practices are compared against recognized industry standards
- Answers generate output with the recommendation list
Online – Offline Physical Tests

Online test:
- Evaluation is performed on actual running industrial network
  - Contains volatile ICS components
- Represents completely functional and operational ICS architecture
  - Including 3rd party components

Offline test:
- Not connected to physical process and not performing real-time control operations
  - Difficult to include 3rd party
- Reflects subset of overall architecture, can omit key components
System Characterization

First activity to perform for physical and online test

Use zone concept for better analysis

- Create trust boundary
- All external entry points require penetration
Device Scanners

Ping command:
- Basic device identification tool, built-in to most commercial OS
- Not effective in ICS due to security appliances rarely forward ping (ICMP)

Arping and arp-scan:
- Based on ARP protocol (MAC layer) can be used to identify hosts

Network mapper or nmap:
- Data collection via network-based, external packet injection and analysis
- Host discovery, host service detection, OS detection, spoofing, execute customized code
Device Scanners

Network statistics or netstat tool

- Command-line feature is available on most OS
- Useful when trying to identify applications and services running on particular host
- Does not inject packets on network which could compromise time-sensitive communication between ICS
- Friendly and passive
Vulnerability Scanners

OpenVAS open-source, and many commercial tools (Tenable Nessus, SAINT scanner)

Identify vulnerabilities that may exists comparing with database of known vulnerabilities

- Depends on product’s database, different results

https://tools.kali.org/vulnerability-analysis/openvas
Traffic Scanners

Collect raw network packets and provide them for host identification, firewall rule set, etc.

Basic form is tcpdump for Linux, windump for Windows

- Purpose is to capture and save network traffic

Wireshark

- Uses pcap (file style of tcpdump)
- Used for analysis of network traffic
- Not recommended to use for raw packet collection
  - Memory performance issues
Examples of Live Host Identification

Quiet Scanning Techniques:
- Single ARP request via `arping`
- Scan entire subnet via `arp-scan (-l)`
Examples of Live Host Identification

Noisy/Dangerous Scanning Techniques:

- Ping sweep on a single subnet via nmap:

```
root@Hacker:~# nmap -sn 192.168.56.0/24
```

Starting Nmap 6.46 (http://nmap.org) at 2014-06-19 07:39 IST
Nmap scan report for 192.168.56.100
Host is up (0.008s latency).
MAC Address: 00:80:27:7A:C0:DB (Cadmus Computer Systems)
Nmap scan report for 192.168.56.103
Host is up (0.0017s latency).
MAC Address: 08:80:27:FC:15:EA (Cadmus Computer Systems)
Nmap scan report for 192.168.56.110
Host is up (0.00823s latency).
MAC Address: 08:80:27:00:24:86 (Cadmus Computer Systems)
Nmap scan report for 192.168.56.115
Host is up (0.011s latency).
MAC Address: 08:80:27:A8:01:85 (Cadmus Computer Systems)
Nmap scan report for 192.168.56.113
Host is up.
Host down: 256 IP addresses (5 hosts up) scanned in 20.37 seconds
```

- Create and send specific packets on network via hping3

```
root@ddos:~# hping3 -h
usage: hping3 host [options]
  -h --help show this help
  -v --version show version
  -c --count packet count
  -i --interval wait (us for X microseconds, for example -i 1 u1000)
    --fast alias for -i u1000 (10 packets for second)
    --faster alias for -i u1000 (100 packets for second)
  -f --flood sent packets as fast as possible. Don’t show replies.
  -n --numeric numeric output
  -q --quiet quiet
  -I --interface interface name (otherwise default routing interface)
  -V --verbose verbose mode
  -D --debug debugging info
  -z --bind bind ctrl-z to ttl (default to dst port)
  -Z --unbind unbind ctrl-z
  --beep beep for every matching packet received
```

Mode
default mode TCP
  -0 --raw1p RAW IP mode
  -1 --icmp ICMP mode
  -2 --udp UDP mode
  -8 --scan SCAN mode.
ex: hping --scan 1-30,40-50 -> www.target.host
Suggested ICS Actions

Instead of ping sweep:
• Perform physical verification
• Conduct passive network listening
• Scan subset of targets

Instead of port scan:
• Do local verification (netstat)
• Scan duplicate or test system on non-production network

Instead of vulnerability scan:
• Non-production network
Command Line Tools

No packet injection

To display network configuration values, `ipconfig` can be used
Steps to be taken for System Characterization

Use `arp-scan` to identify network-connected hosts

Confirm identified hosts are authorized for the network. If not, physically inspect and take actions. Update system architecture with newly discovered info

Collect host info for each connected device, including hardware and OS info
  - Can be obtained via `systeminfo`

Collect app info for each device including vendor, name, patches, etc.
  - Can be obtained via `wmic`

Consolidate this info into database with appropriate classified policies
Vulnerability Identification

Vulnerability is not only unpatched software but also use of unnecessary services/apps

- Cannot be fully detected by scanning for presence (or absence) of software

Vulnerability can exist in form of:

- Improper authentication
- Poor credential management
- Improper access control
- Inconsistent documentation
Vulnerability Identification

Assessment phase depends on scanning tool

Involves review of relevant apps, host, config files

Physical aspect of ICS is inspected

Security controls are reviewed

Objective is to identify backdoors (holes) that may exist in the network perimeter
# Common ICS Vulnerabilities

<table>
<thead>
<tr>
<th>Category</th>
<th>Potential Vulnerabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>Physical Security</td>
</tr>
<tr>
<td></td>
<td>Configuration Errors or Management</td>
</tr>
<tr>
<td></td>
<td>Port Security</td>
</tr>
<tr>
<td></td>
<td>Use of Vulnerable INP</td>
</tr>
<tr>
<td></td>
<td>Lack of IDS Capabilities</td>
</tr>
<tr>
<td>Config</td>
<td>Poor Account Management/Password Policies</td>
</tr>
<tr>
<td></td>
<td>Lack of Patch Management</td>
</tr>
<tr>
<td></td>
<td>Ineffective Whitelisting</td>
</tr>
<tr>
<td>Platform</td>
<td>Insecure Embedded Apps/Untrusted 3rd Party Apps</td>
</tr>
<tr>
<td></td>
<td>Lack of System Hardening</td>
</tr>
</tbody>
</table>
# Common ICS Vulnerabilities

<table>
<thead>
<tr>
<th>Category</th>
<th>Potential Vulnerabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICS Apps</td>
<td>Code Quality, Lack of Authentication, Vulnerable INP</td>
</tr>
<tr>
<td>Embedded Devices</td>
<td>Config Errors, Vulnerable INP, Insufficient Access Control</td>
</tr>
<tr>
<td>Policy</td>
<td>Security Awareness, Social Engineering, Physical Security, Access Control</td>
</tr>
</tbody>
</table>
Example of Manual Vulnerability Scanning

1. Use “wmic” to list all installed apps running on Windows server

2. SCADA app software is shown as “XYZ” with vendor name “ABC” and version “2.3”

3. Using OSVDB with “ABC” keyword several results are returned

4. Compare your system to see if you have that vulnerability mentioned

5. Install the patches if available + needed
Important Tips for Vulnerability Scanning

Should never be used on online ICS without prior testing and approval from directly responsible for operation of ICS

A system has no vulnerabilities does not mean that it has been configured in a secure manner

- Neither we can say that is fully secure
Risk Classification and Ranking

Compare the threats and vulnerabilities identified

- Important to make effective security program that addresses not only operational security but also business operations

Last step before taking actions (applying policies, etc.)

- Take into account the consequence to operations that would occur, if cyber event occurs

For instance gas pipelines that controlled by ICS;

- If a real battle fought, much harder for victory
- But cyber war
Estimate Consequences and Likelihood

Microsoft model DREAD (Damage Potential, Reproducibility, Exploitability, Affected User, Discoverability):

- Provides qualitative method of assigning value to each classification
- Consequence is not dependent on time
- Consider how easy to obtain knowledge (malware code) to exploit vulnerability
  - If no proof of concept has ever been developed, less likely to be exploited
- The skill level of attacker for that exploit
  - A script kiddie could perform this attack?