

*Department of Computer Science
Southern Illinois University Carbondale*

**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

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

Examples of Attack Targets

Target	Attack Vectors	Attack Methods	Consequences
Operator workstation (HMI)	<ul style="list-style-type: none"> -Operational applications -USB -Control network 	<ul style="list-style-type: none"> -Installation of malware via USB -Authorization of ICS HMI functions without sufficient access control mechanisms 	<ul style="list-style-type: none"> -Plant shutdown -Product quality -Credential leak (control)
Telecommunication systems	<ul style="list-style-type: none"> -Public Key infrastructure -Internet visibility 	<ul style="list-style-type: none"> -Disclosure of private key via external compromise -Exploitation of device connected to public networks -Network access through unmonitored access points 	<ul style="list-style-type: none"> -Credential leak (control) -Information leak -Unauthorized remote access -Command and control
ICS Technician	<ul style="list-style-type: none"> -Social engineering -Email attachments -File shares 	<ul style="list-style-type: none"> -Transmission of malware on control network via unauthorized connection -Exploitation of applications with administrative rights 	<ul style="list-style-type: none"> -Plant shutdown/delay -Mechanical sabotage -Modification of status messages

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

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

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



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:

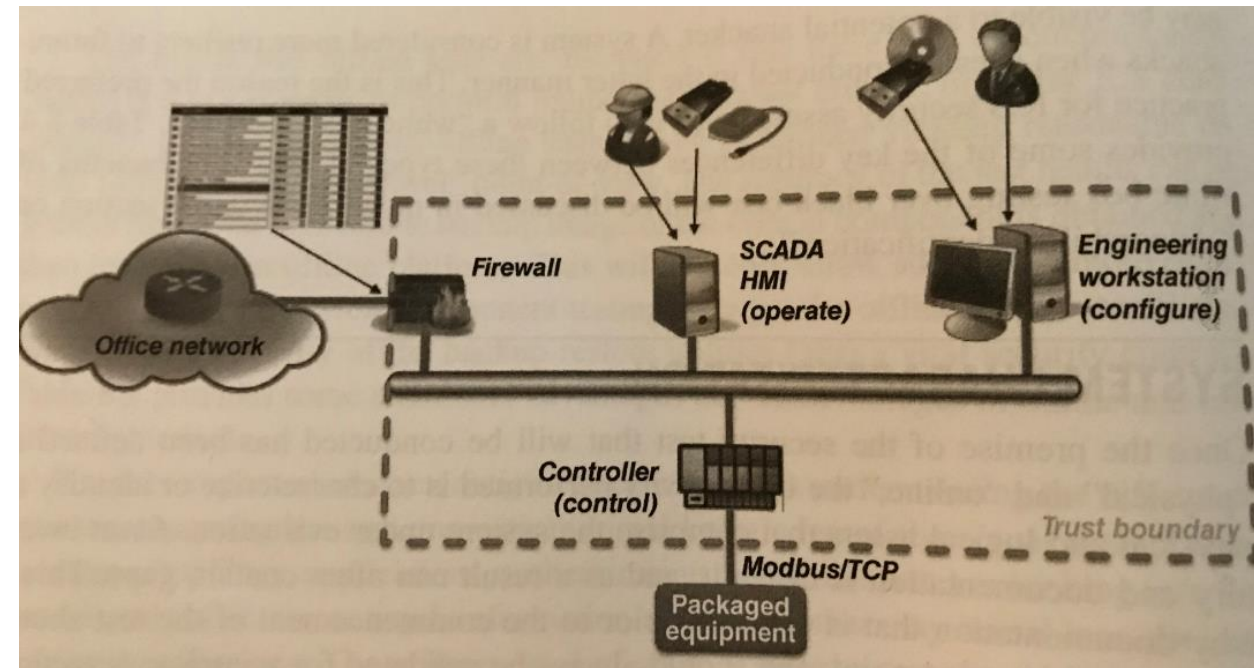
- Not connected to physical process and not performing real-time control operations
 - Difficult to include 3rd party
- Reflects subset of overall architecture, can omit key components

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

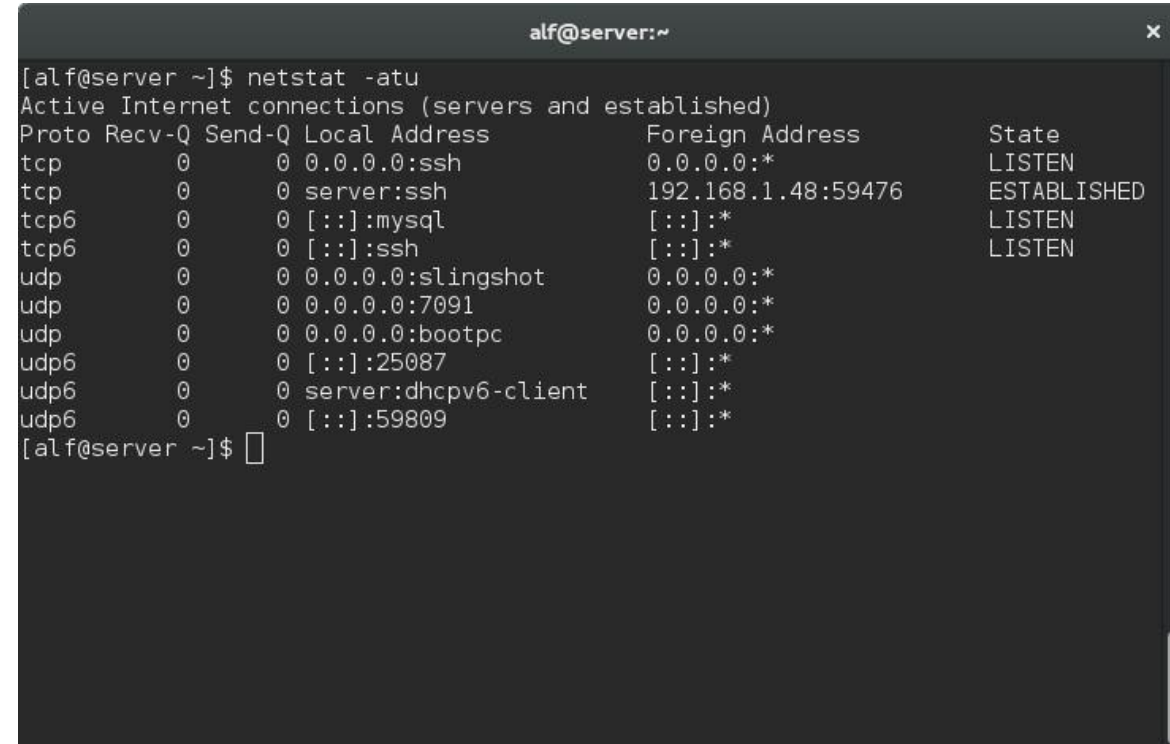
Network mapper or *nmap*:

- 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

A terminal window titled 'alf@server:~' showing the output of the 'netstat -atu' command. The output lists active Internet connections, including listening and established connections for various protocols like tcp, tcp6, udp, and udp6.

```
alf@server ~]$ netstat -atu
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:ssh             0.0.0.0:*               LISTEN
tcp        0      0 server:ssh             192.168.1.48:59476     ESTABLISHED
tcp6       0      0 [::]:mysql             [::]:*                 LISTEN
tcp6       0      0 [::]:ssh               [::]:*                 LISTEN
udp        0      0 0.0.0.0:slingshot      0.0.0.0:*
udp        0      0 0.0.0.0:7091           0.0.0.0:*
udp        0      0 0.0.0.0:bootpc         0.0.0.0:*
udp6       0      0 [::]:25087             [::]:*
udp6       0      0 server:dhcpv6-client   [::]:*
udp6       0      0 [::]:59809             [::]:*
```

Vulnerability Scanners

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

Identify vulnerabilities that may exist 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)

```
root@debian:~# arping -c 2 192.168.178.27
ARPING 192.168.178.27
60 bytes from 08:00:27:c9:7c:85 (192.168.178.27): index=0 time=396.617 usec
60 bytes from 08:00:27:c9:7c:85 (192.168.178.27): index=1 time=313.585 usec

--- 192.168.178.27 statistics ---
2 packets transmitted, 2 packets received, 0% unanswered (0 extra)
rtt min/avg/max/std-dev = 0.314/0.355/0.397/0.042 ms
root@debian:~#
```

```
File Edit View Search Terminal Help
root@kali:~# arp-scan --interface=eth0 --localnet
Interface: eth0, datalink type: EN10MB (Ethernet)
Starting arp-scan 1.9 with 256 hosts (http://www.nta-monitor.com/tools/arp-scan/)
172.16.44.1      00:50:56:c0:00:08      VMware, Inc.
172.16.44.2      00:50:56:fa:49:a4      VMware, Inc.
172.16.44.140    00:0c:29:2d:9c:10      VMware, Inc.
172.16.44.141    00:0c:29:0a:56:4f      VMware, Inc.
172.16.44.145    00:0c:29:5f:1d:1f      VMware, Inc.
172.16.44.148    00:0c:29:0f:46:91      VMware, Inc.
172.16.44.149    00:0c:29:df:37:17      VMware, Inc.
172.16.44.153    00:0c:29:ec:fd:52      VMware, Inc.
172.16.44.254    00:50:56:fe:c9:1a      VMware, Inc.

9 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.9: 256 hosts scanned in 2.203 seconds (116.21 hosts/sec). 9 responded
root@kali:~#
```

Examples of Live Host Identification

Noisy/Dangerous Scanning Techniques:

- Ping sweep on a single subnet via nmap:

```

root@Qhacker:~# nmap -sn 192.168.56.0/24
Starting Nmap 6.46 ( http://nmap.org ) at 2014-06-19 07:38 IST
Nmap scan report for 192.168.56.100
Host is up (0.00058s latency).
MAC Address: 08:00:27:7A:CC:DB (Cadmus Computer Systems)
Nmap scan report for 192.168.56.103
Host is up (0.0017s latency).
MAC Address: 08:00:27:FC:15:EA (Cadmus Computer Systems)
Nmap scan report for 192.168.56.110
Host is up (0.00023s latency).
MAC Address: 08:00:27:00:24:06 (Cadmus Computer Systems)
Nmap scan report for 192.168.56.115
Host is up (0.011s latency).
MAC Address: 08:00:27:A0:16:85 (Cadmus Computer Systems)
Nmap scan report for 192.168.56.113
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 28.97 seconds
root@Qhacker:~# █
  
```

- Create and send specific packets on network via hping3

```

root@ddos: ~
File Edit View Search Terminal Help
root@ddos:~# hping3 -h
usage: hping3 host [options]
-h --help          show this help
-v --version       show version
-c --count         packet count
-i --interval     wait (uX for X microseconds, for example -i u1000)
                  --fast      alias for -i u10000 (10 packets for second)
                  --faster    alias for -i u1000 (100 packets for second)
                  --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 --rawip      RAW IP mode
-1 --icmp       ICMP mode
-2 --udp        UDP mode
-8 --scan       SCAN mode.
Example: hping --scan 1-30,70-90 -S 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

```
C:\Users\SIU[redacted]>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::8cb2:9d7d:c0bd[redacted]
    IPv4 Address. . . . . : 131.230.166.[redacted]
    Subnet Mask . . . . . : 255.255.255.192
    Default Gateway . . . . . : 131.230.166.254
```

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

Category	Potential Vulnerabilities
Network	Physical Security Configuration Errors or Management Port Security Use of Vulnerable INP Lack of IDS Capabilities
Config	Poor Account Management/Password Policies Lack of Patch Management Ineffective Whitelisting
Platform	Insecure Embedded Apps/Untrusted 3 rd Party Apps Lack of System Hardening

Common ICS Vulnerabilities

Category	Potential Vulnerabilities
ICS Apps	Code Quality Lack of Authentication Vulnerable INP
Embedded Devices	Config Errors Vulnerable INP Insufficient Access Control
Policy	Security Awareness Social Engineering Physical Security Access Control

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?