SIU Southern Illinois University
CARBONDALE

Department of Computer Science
Southern Illinois University Carbondale

CS 491/531
SECURITY IN CYBER-PHYSICAL SYSTEMS

Lecture 14: Risk and Vulnerability in ICS

DR. ABDULLAH AYDEGER

LOCATION: ENGINEERING A 409F

EMAIL: AYDEGER@CS.SIU.EDU

Outline

Vulnerability Identification/Scanning

Risk Classification and Ranking

Recall: Statistics of ICS Incidents

80% impacting ICS are “unintentional”

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

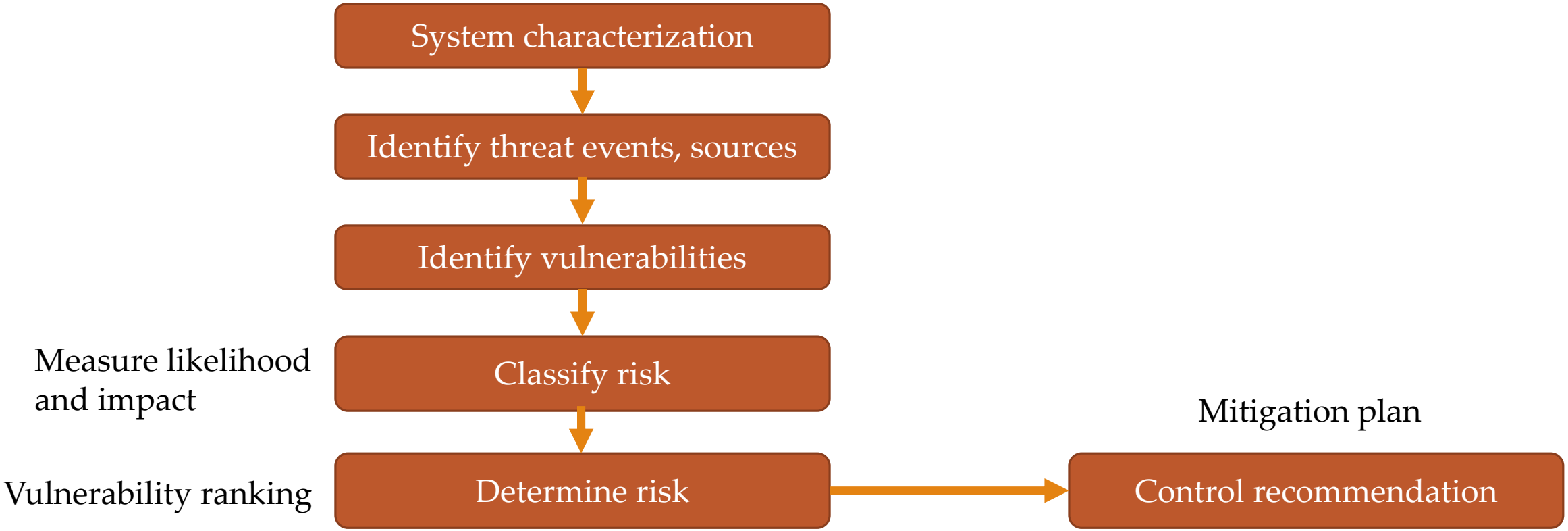
Embedded devices and network appliances were targeted 34%

- Windows-based ICS and enterprise hosts 66%

These numbers would help to understand risks that should be prioritized

https://scadahacker.com/

Recall: Flowchart of Assessing Risks to ICS



Recall: Scanning Industrial Networks

Device Scanners

Vulnerability Scanners

Traffic Scanner

Recall: 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 exists 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 3rd 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

Steps of Vulnerability Identification

Devices with little or no security feature are identified

- So they can be placed in special security zones and secured separately

Networks are reviewed to detect possible communication hijacking (MitM) opportunities

- Every component connected to ICS is assessed to discover improper features (i.e., no patch)

Good practice to work with suppliers

- So that they can also keep their vulnerabilities updated

Vulnerability Scanning

Some automated tools as we discussed earlier

Manual tests for critical host:

- Collecting info using command-line tools
- Comparing info of OS, apps and services against known vulnerabilities
- Two popular vulnerability database:
  - National Vulnerability Database (NVD) by NIST https://nvd.nist.gov/
  - Open-source Vulnerability Database (OSVDB) https://cve.mitre.org/data/refs/refmap/source-OSVDB.html

WhiteSource Vulnerability Database: https://www.whitesourcesoftware.com/vulnerability-database/

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

Authenticated Scan

Performs *white box* assessment

- Authenticating remotely on the device and performing variety of internal audits
  - Including network statistics

Provides accurate reflection of security posture of target

- Not just what is visible to attacker

More friendly on target and does not typically inject hostile traffic info network

White Box vs. Black Box

White	Black
Intent is to identify security vulnerabilities that could leak to an exploit	Represents system in a way that attacker sees it
Requires asset owner to disclosure significant info for test purposes	Protect intellectual property
Provides most comprehensive look at vulnerabilities and risk	Does not provide complete exposure to risk

Types of Vulnerability Scanning

Active mechanisms

- Place some packet on network
- "Aggressiveness" of the scan and impact on the target can be controller

Passive scanner

- Snapshot view of vulnerabilities on target
- Able to enumerate network and detect new devices are added
  - Well suited for ICS due to static nature of network topology and regular traffic patterns

Host-based scanner

- Must be installed in host
  - Not really acceptable within ICS zone

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

Configuration Auditing

Compliance auditing

- Compares current config of host against set of acceptable settings
- The settings may be determined by organization's policy, regulatory standard, etc.

Nessus vulnerability scanner provides this audit

- Can be performed on config of OS, apps, antivirus, database, network infrastructure, etc.

US DoE funded project to develop set of security config guidelines for ICS components

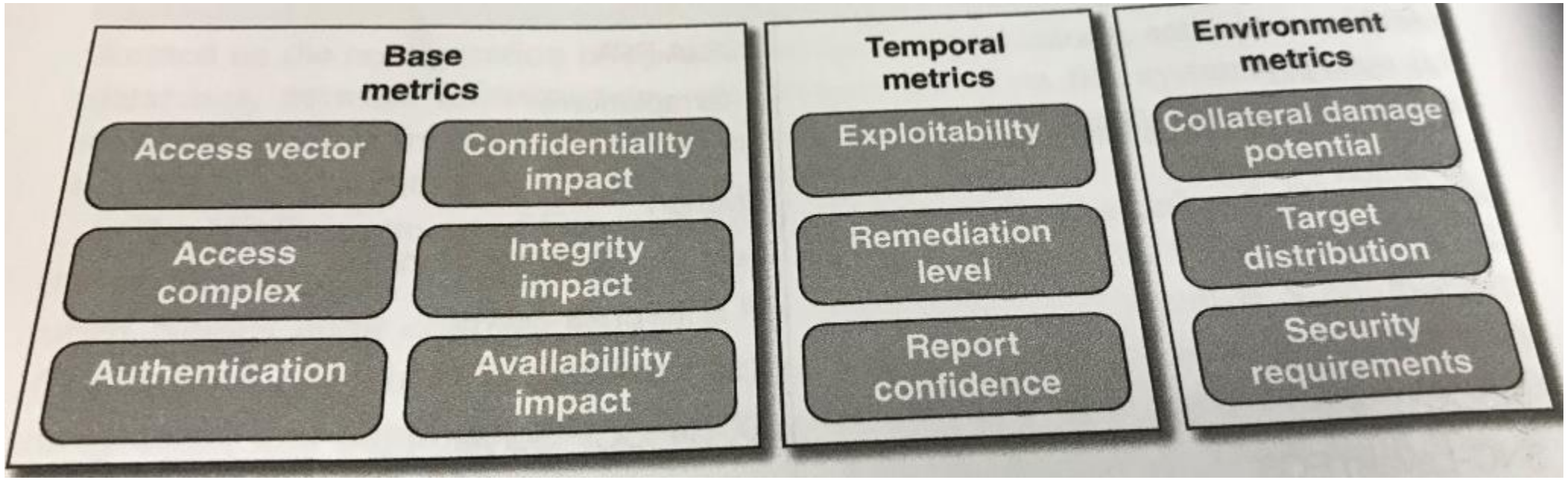
- Can be used in Nessus

Vulnerability Scoring/Prioritization

Common Vulnerability Scoring System (CVSS) globally accepted industry standard for determining severity of system vulnerability

- Base Metric: mandatory component used to present characteristics of vulnerability
  - Constant with time and across different environments
  - Provided by the party responsible for disclosing the vulnerability
- Temporal Metric: Change over time but not across different environment
- Environmental Metric: reflects environmental characteristics

Example fields for Metrics



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*

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?

DREAD Model

Rating	High	Medium	Low	Indirectly Measures
D	Can subvert security Get full trust authorization Upload content	Leaking sensitive info	Leaking trivial info	Consequences
R	Can be reproduced every time, does not require a timing window No authentication required	Can be reproduced only with a timing window/particular situation Authorization required	Very difficult to reproduce Requires admin rights	Likelihood
E	Novice programmer could make the attack in short time Simple toolset	Skilled programmer could make the attack Exploit and tools are publicly available	Attack requires extremely skilled person and in-depth knowledge Custom exploit/tools	Likelihood

DREAD Model

Rating	High	Medium	Low	Indirectly Measures
A	All users Default config Key assets	Some users Non-default config	Very small percentage of users (anonymous users) Obscure feature	Consequences
D	Published info explains the attack Vulnerability is in the most commonly used feature Very noticeable	Vulnerability is in seldom-used part of product Only a few users should come across Would take some time to see malicious use	Bug is obscure Requires source code Administrative access	Likelihood

Risk Reduction and Mitigation

Should not be a one time tactical investment

- Instead a long-term strategic investment