Security Analysis Profile v 2

Bruce Powel Douglass, Ph.D.
IBM IoT

Abstract

The Security Analysis Profile (SecPro2) is a UML profile specifically to enhance system designers, systems engineers, and enterprise architects to specify and design security systems. SecPro2 is intended to facilitate the identification of security needs and requirements, identify and characterize assets of value in the system or enterprise, identify and evaluate the effectiveness of countermeasure, and model attack behavior with or without countermeasure mitigation. SecPro2 provides a number of views – diagrams, tables and matrices – for this purpose. It is intended that the profile provides security relevant metadata either as a stand alone model or as a part of an encompassing UML or SysML model and to support exchange with risk analytic tools such as *Blade Risk Analyzer* from KDM Analytics.

Concepts and views

SecPro2 is based upon the OMG *Operational Threat and Risk Sharing and Federation Model (T&RM)* (current in the standardization process). It has a separate metamodel that is explicitly related to the T&RM conceptual model but is itself based on the MOF and UML metamodels, making it suitable for a UML profile. Some key concepts represented in this profile are:

Security: freedom of a system from external intrusion, interference or theft.

Asset: a security relevant feature of a system that the system wishes to protect. Assets include the following properties:

- Access Level Permitted
- Accountability
- Asset Kind
    - Actor
    - Information Asset
    - Current Asset
    - Resource Asset
    - Physical Asset
    - Service Asset
    - Security Asset
    - Tangible Asset
    - Intangible Asset
- Availability
- Clearance Required
- ID
- Integrity
- Value

Asset Context: The system or extra-system elements surrounding or enshrouding one or more assets; e.g. a safe in which money is kept and a network are both part of a asset context. An asset context may be decomposed into contained asset context elements.
Accessible Resource: Either an Asset or Asset Context

Vulnerability: a weakness in the security field of an asset that may be exploited by an attack

Threat: a means by which the security shield of an asset may be penetrated through a vulnerability. Types of threats include (but are not limited to):

- Damage
- Deception
- Disclosure
- Disruption
- Intrusion
- Theft
- Interception
- Usurpation

Attack: The realization of a threat invoked by a threat agent

Attack Chain: A type of attack that is composed of sub-attacks. Sometimes known as a *Cyber Killchain*.

Threat Agent: A human or automated threat source that invokes an attack, typically intentionally

Threat Source: An element that may cause a security violation, intentionally or otherwise (e.g. storm knocks out the power, unlocking the doors of a facility).

Countermeasure: A means by which a vulnerability is protected from attack. Countermeasures may be passive or active, and may be implemented by design elements, policies, procedures, labeling, training, or obviation. Countermeasure types include (but are not limited to):

- Access Control
- Accounting
- Active Detection
- Authentication
- Recovery
- Boundary Control
- Backup
- Encryption
- Deterrence
- Obviation
- Nonrepudiation
- Policy Action
- Response
- Scanning Detection

Role: A part a person plays in a context, e.g. a user, administrator, or trusted advisor.

Authenticated Role: A role with explicit authentication, which typically includes a set of permissions.

Permission: The right or ability to perform an action that deals with an Accessible Resource. A role may be granted permissions to perform different kinds of access to an asset.

Access: A type of action that can be performed on a resource. This includes (and may be extended)

- No access
- Unrestricted access
- Read access
- Modify access
- Open access
- Close access
- Entry access
- Exit access
- Create access
- Delete access
- Remove access
- Invoke access
- Configure access
- Interrupt access
- Stop access

Security Violation: The inappropriate intrusion, interference, or theft of an asset; this may be the result of an attack (intentional) or failure (unintentional)

Security Posture: The set of assets, asset contexts, vulnerabilities, and countermeasures aggregate to form the system Security Posture.

Risk: the possibility of an undesirable event occurring or undesirable situation manifesting. Risk is the product of (at least) two values: likelihood and severity.

Risk Number: The numeric value associated with a risk (likelihood x severity).

Views

Several distinct views are provided by the profile. These are split into diagrams and tables/matrices.

Security Analysis Diagram (SAD)

SAD is a logical causality diagram that shows how assets, events and conditions combine to express vulnerabilities, how countermeasures address vulnerabilities, and how attacks cause security violations. The intention is to identify when and where countermeasures are or should be added to improve system security. This diagram uses logical operations (AND, OR, NOT, XOR, and so on) to combine the presence of assets, asset context, situations, and events. This is similar to a Fault Tree Analysis diagram used in safety analysis.

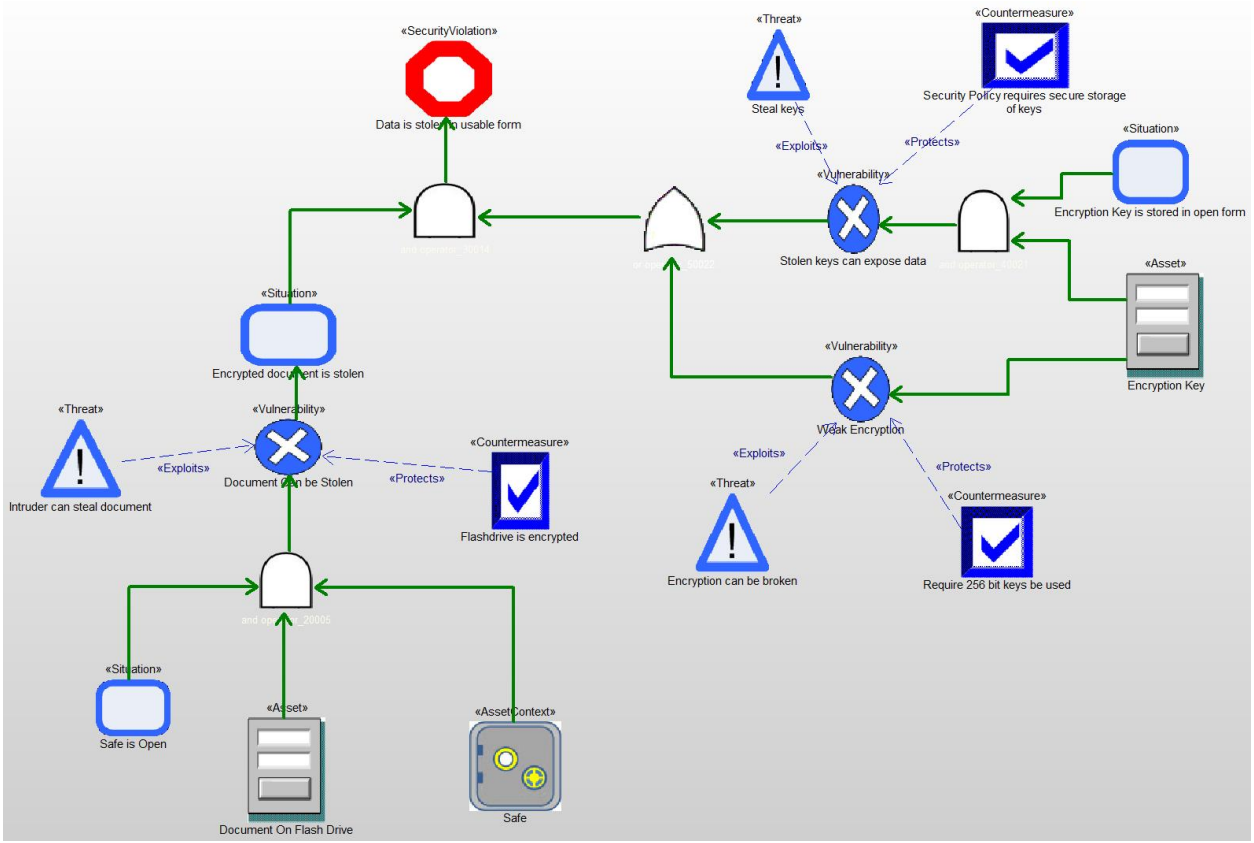


Figure 1: Sample Security Analysis Diagram

Asset Diagram (AD)

AD is a structural diagram intending to show the relation among assets, asset contexts, vulnerabilities, counter measures, supporting security requirements, and security-relevant design elements.

Here are a couple of examples:

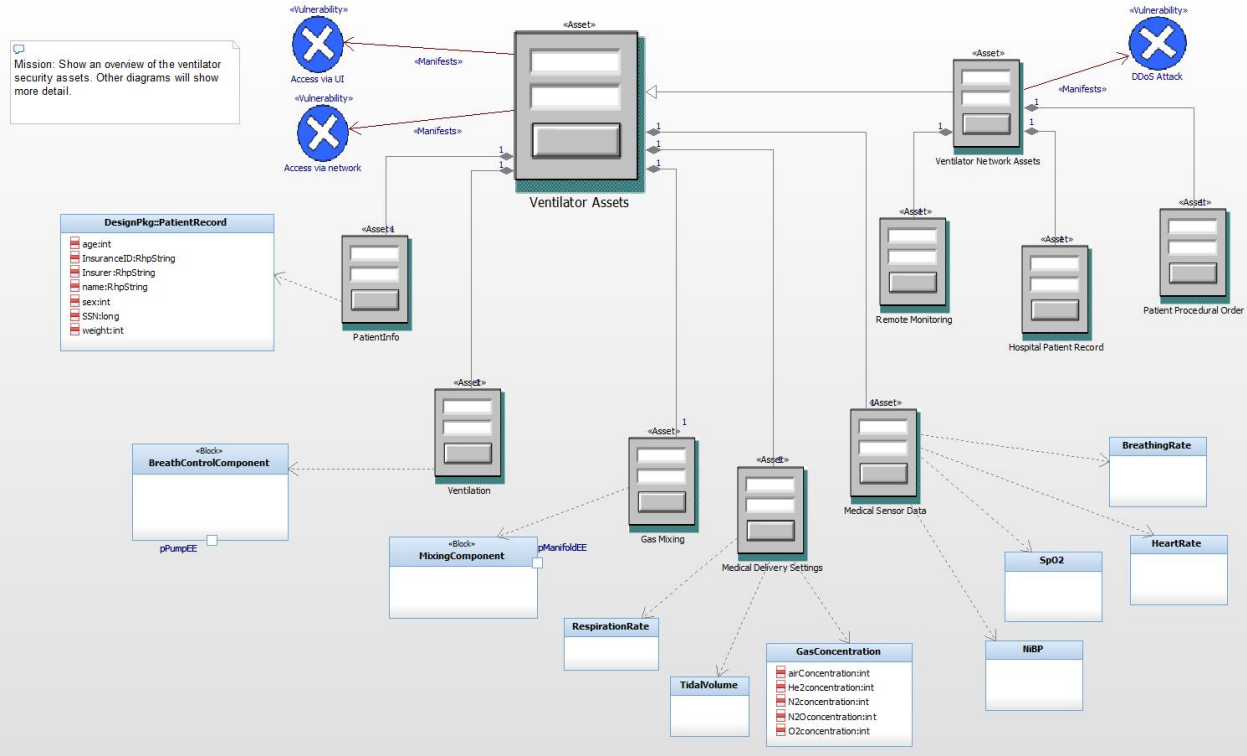


Figure 2: Asset Diagram with icons

Figure 2 shows an overview of the assets within a medical ventilator using the iconic depiction of assets. This can also be shown using standard UML box notation. This has the advantage being able to easily show the values of the security metadata (stored in tags). See Figure 3.

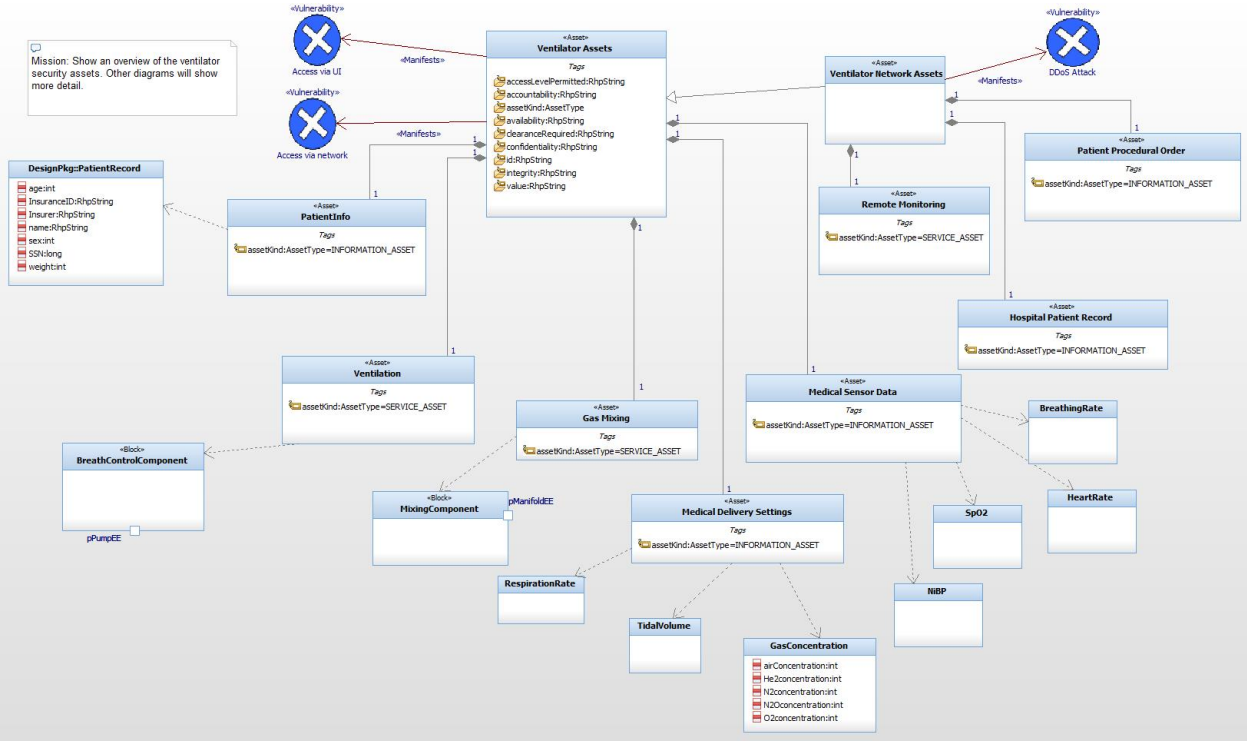


Figure 3: Asset diagram with standard UML notation

The next figure shows an asset diagram focusing on roles, permissions, vulnerabilities, countermeasures and related requirements.

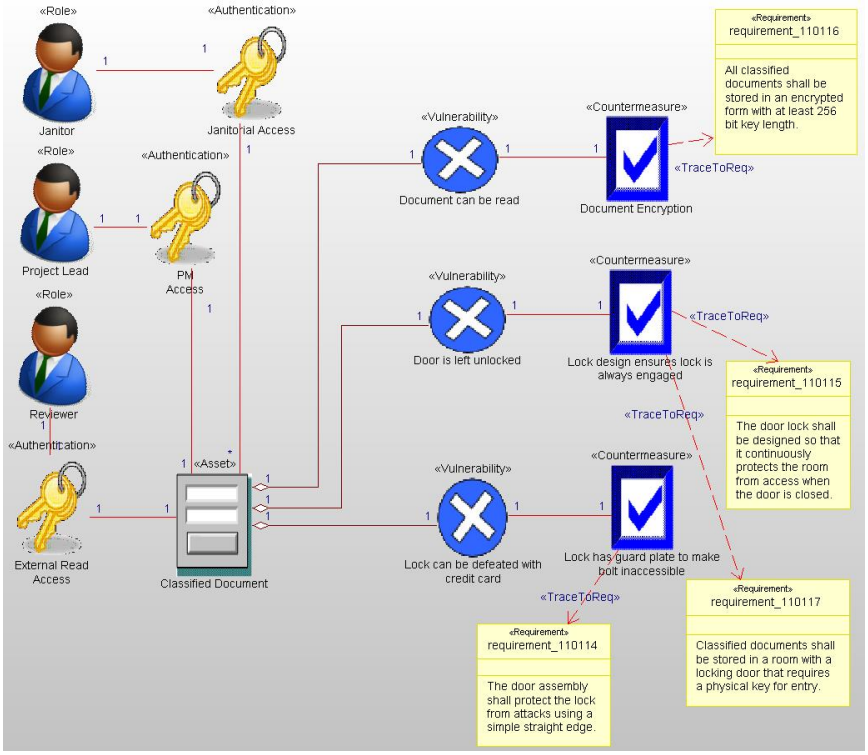


Figure 4: Asset diagram with roles

Attack Flow Diagram (AFD)

An AFD is a specialized UML activity diagram, with explicit Attack and Mitigation actions. Attack actions may be further stereotyped into standard attack chain action subtypes including: Weaponization, Delivery, Installation, Reconnaissance, Command and Control, Exploitation, and Actions On Objective.

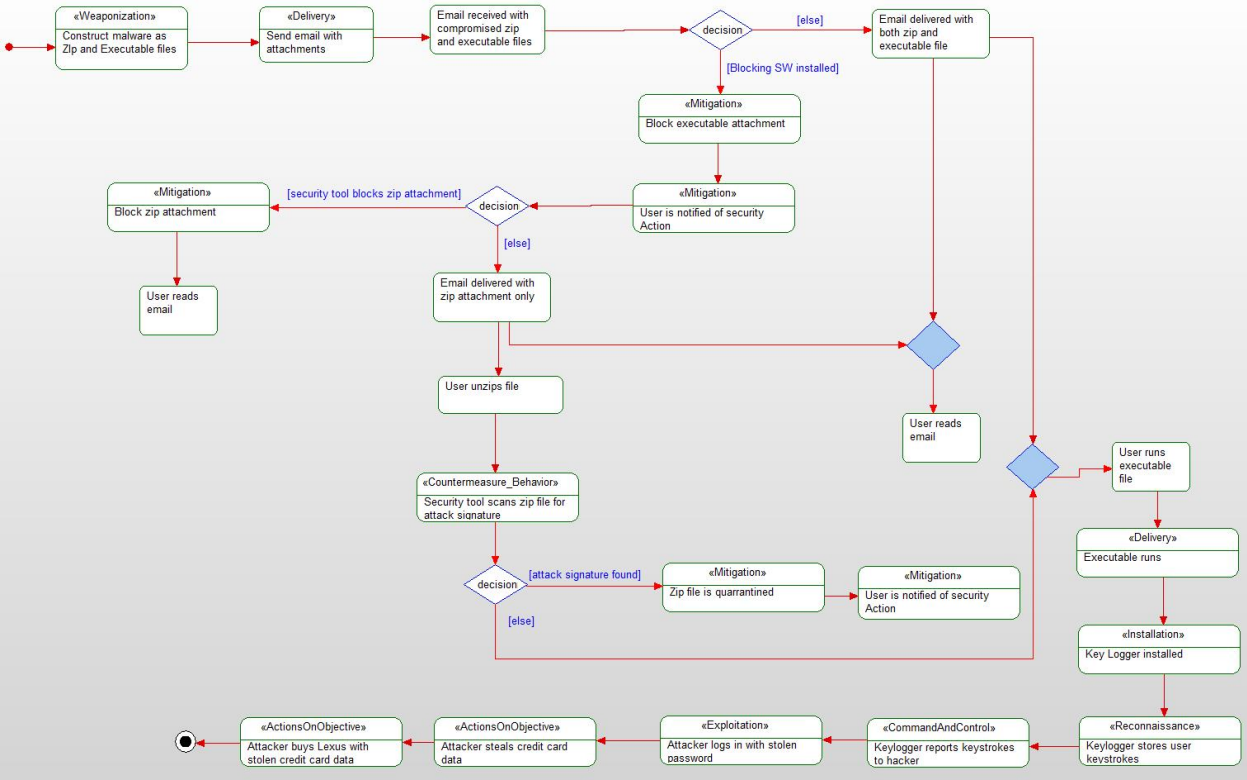


Figure 5: Attack Flow Diagram

Attack Scenario Diagrams (ASD)
It is envisions that standard (unmodified) sequence diagrams will serve the purpose of displaying specific attack/mitigation scenarios.

Security Posture Table (SAT)
The SAT is a tabular summary for assets, asset context, vulnerabilities, and countermeasures and their important security-relevant metadata, including Name, Description, Risk Number, Severity, Probability, Consequence, and Impact.

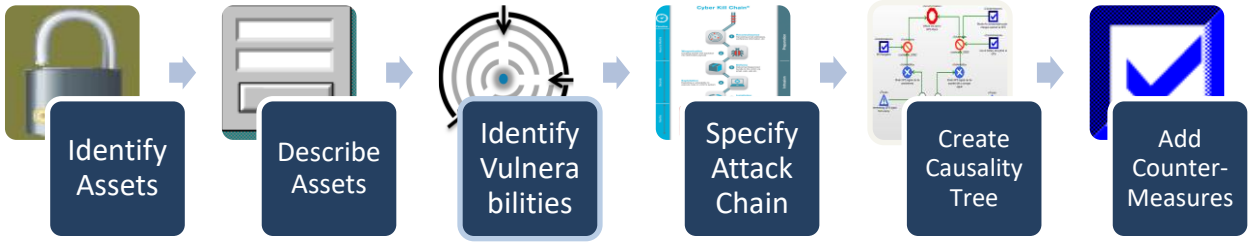
A number of other tables are provided including tables of Assets, Countermeasures, and Roles.

A number of matrices are provided to show the relations between elements including:

- Asset – Authentication
- Asset – Vulnerability
- Countermeasure – Vulnerability
- Requirement – Security Element
- Role – Asset

A security analysis workflow

The profile is process-agnostic, although the Harmony Agile Model-Based Systems Engineering (Harmony aMSBSE) and Harmony for Embedded Software (Harmony ESW) processes both contain workflows for security analysis. This section abstracts and describes that workflow for security analysis.



Step 1: Identify Assets

Assets are system or environment features or properties that have value that your system wishes to protect. Create one or more asset diagrams to visualize these assets.

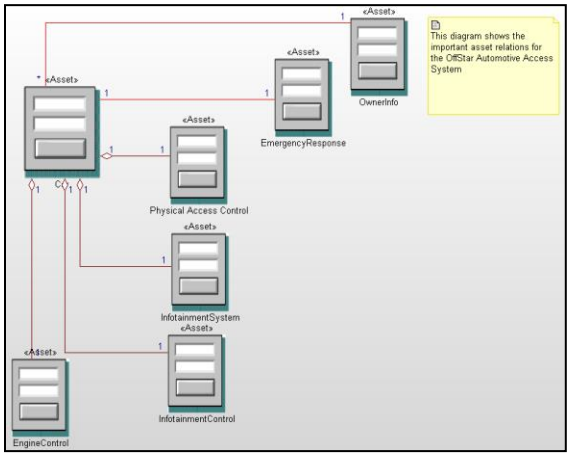


Figure 6: Add Assets

Step 2: Describe Assets

Assets have a number of properties, as described above, not all of which may apply to you. At minimum, identify the asset value and asset kind. Some assets are more valuable than others and should receive greater attention than lesser valued assets.

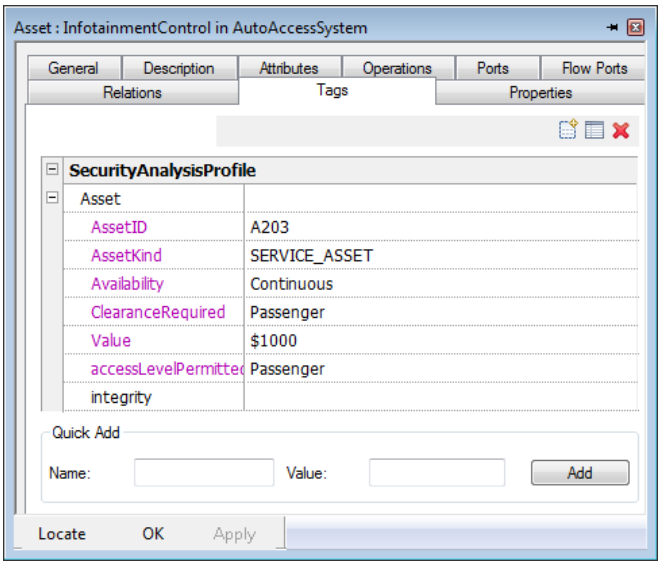


Figure 7: Characterize assets

Step 3: Identify Vulnerabilities

Vulnerabilities are weaknesses expressed either by assets themselves or by their context. Identify these vulnerabilities and exposed them explicitly. If you are using known technology, then sources such as the Common Vulnerability Enumeration (CVE) or Common Weakness Enumeration (CWE) are good sources of information.

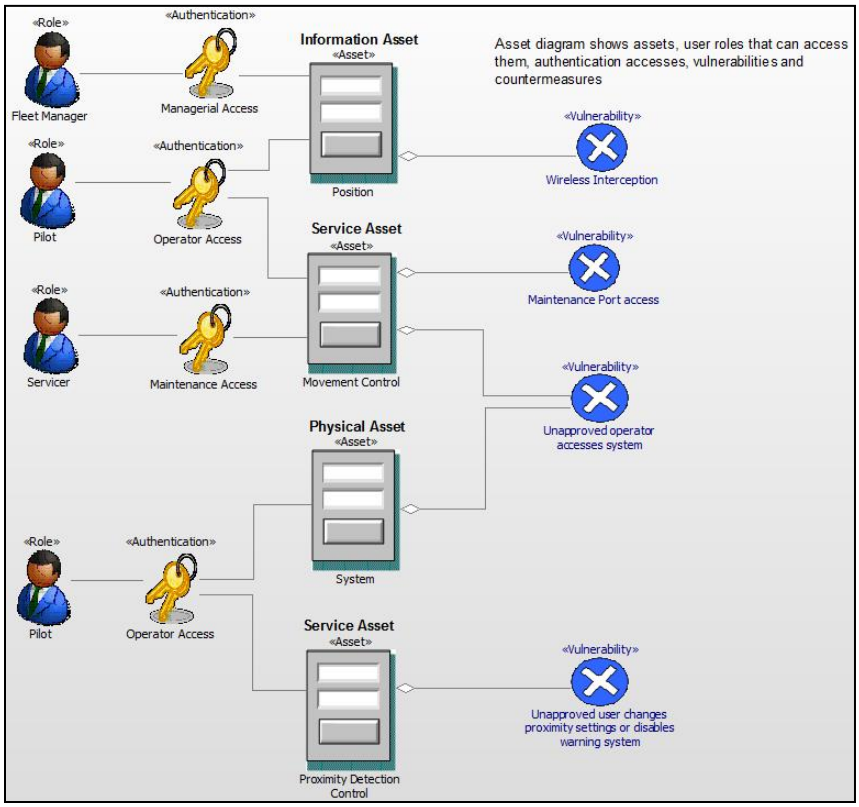


Figure 8: Identify vulnerabilities

Step 4: Specify Attack Chains

Most attacks are not a single action, but an orchestrated series of actions meant to defeat countermeasures, gain access, compromise a system and then perform "actions on objective" to exploit the asset. Use the Attack Flow Diagram or Attack Scenario Diagrams to model and understand how an attack achieves its goals and where countermeasures might be effective.

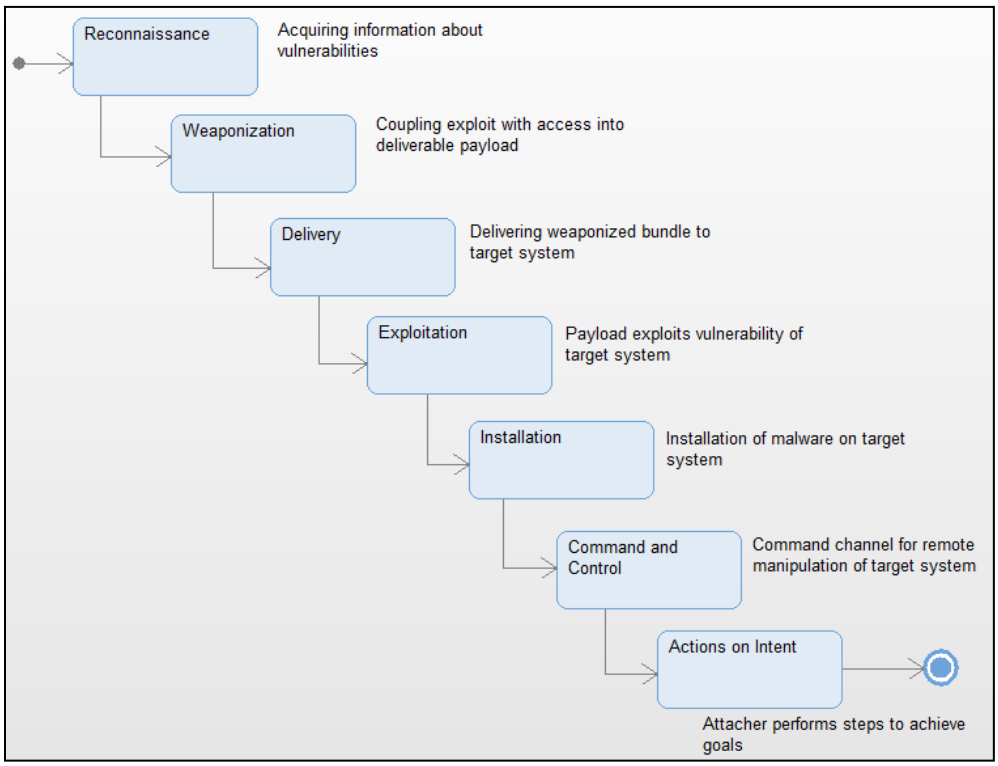


Figure 9: A prototypical cyber-attack chain

Step 5: Create Causality Trees

Express your understanding of the causal relations between attacks, vulnerabilities, and countermeasures on SADs.

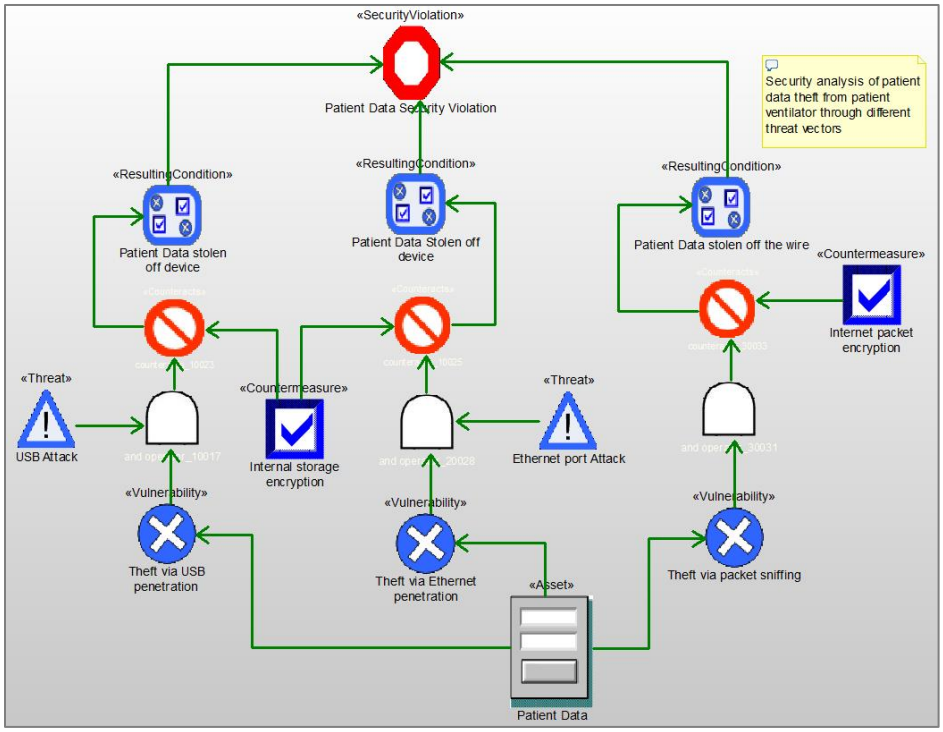


Figure 10: Security Analysis Diagram shows the causality tree for a set of attacks

Step 6: Add Countermeasures

Finally, add countermeasures that prevent, detect, or otherwise mitigate the attacks. These countermeasures should be related to system requirements that specify the need for the countermeasures, and design measures or policy actions that realize the countermeasures.

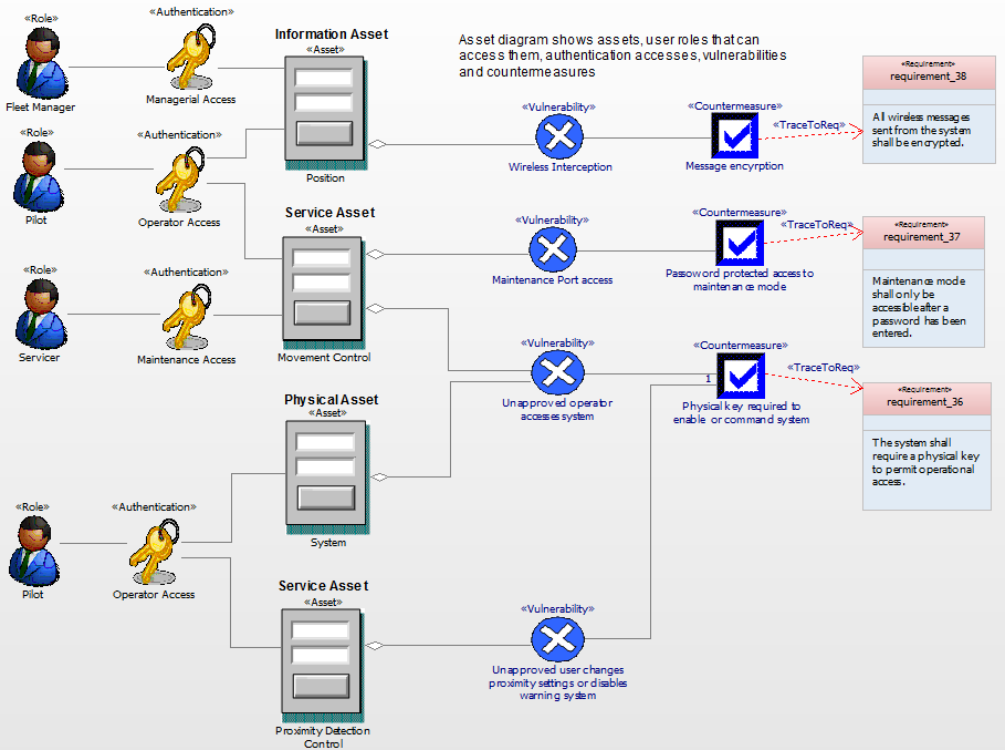


Figure 11: Add countermeasures, security requirements, and security design elements