Threat Modeling  
Model Creation Procedure

# Revision

Version 8

3/20/25 2:03 PM

# SME

Charles Wilson

# Abstract

This document describes the procedure used to perform the threat model creation activity described in the AVCDL secondary document **Threat Modeling Report** **[2]**.

# Group / Owner

Security / Security Architect

# Motivation

This document is motivated by the need to identify design deficiencies related to cybersecurity as early in the product development process as possible. This is necessary given the nature of safety-critical, cyber-physical systems, subject to certifications such as **ISO/SAE 21434** and **ISO 26262**.

**Note:** Within the context of this document, the terms ***security*** and ***cybersecurity*** are used interchangeably. It is presumed that the term ***security*** is being used in reference to ***cybersecurity*** and not ***physical security***.

# Audience

The audience of this document is the cybersecurity practitioner who will be conducting the cybersecurity threat model creation.

# Completeness of Output

Since threat modeling is an as-is analysis, it is critical to ensure that the information gathered is as accurate and complete as possible. As with any cybersecurity assessment, it is not the place of the cybersecurity SME to make assumptions on the part of engineering. All information should be sourced from and confirmed by the owner of the element under consideration.

When information is not available for either a given section of the template or parts thereof, this should be noted. Major omissions should be recorded in the cybersecurity risk register.

# Disposition of Output

Once completed, the generated output should be managed in the organization’s document management system (DMS) as a document of record.

# Entry Criteria

This document assumes that the reader understands the purpose of cybersecurity threat modeling. Further, that the reader has read and understood the AVCDL **Threat Modeling Report** secondary document.

## Prerequisites – Development SME

### Qualifications

It is required that the development SME is both a qualified and trained development SME with expertise in the element under consideration.

**Note:** The specific qualifications and training of development SMEs necessary to assert expertise in providing accurate information regarding the element under consideration is outside the scope of this document.

**Note:** The development group is responsible for determining and making available the appropriate SME for the analysis of the element under consideration.

### Knowledge

It is required that the development SME have expertise in the element under consideration.

## Prerequisites – Threat Modeling SME

### Qualifications

It is required that the threat modeling SME is both a qualified and trained security architect (shown above on title page as **Owner**) as defined by the **NIST NCWF** role SP-ARC-002 and detailed in section **12.7 Security Architect** of the AVCDL primary document **[1]**.

### Knowledge

It is required that the threat modeling SME understands the purpose of cybersecurity threat modeling.

### Background Information

It is required that the threat modeling SME has read and understands the AVCDL **Attack Surface Analysis Report** **[6]**, **Threat Modeling Report**, and **Security Requirements Taxonomy** **[3]** secondary documents. Additionally, that the cybersecurity SME has taken training relevant to this activity.

## Prerequisites – Input Materials

It is required that the development group provide all relevant documentation related to the element under consideration necessary to complete the procedure.

**Note:** Because the scope of threat modeling is variable, the specific documentation required will be determined and provided to the development group by the threat modeling SME prior to the start of the element’s model creation.

# Model Creation Activity

The workflow diagram for the model creation activity of the **Threat Modeling Report** is shown below.

A diagram of a system model

AI-generated content may be incorrect.

Using the **system design**, the threat modeling SME works with the development SME(s) to create a model of the system suitable for threat analysis. Any existing models are used as input to this process. For this activity, models are created using a formal modeling tool when feasible. This may be a set of models depending on the complexity of the system. A **threat model** is created. The model is stored in the DMS.

**Note:** The **system design** may include information attained from attack surface analysis activities.

# Theory

The AVCDL **Understanding TARA in an AVCDL Context** **[5]** elaboration document notes that the **model creation** activity equates to the **asset identification** activity of various standards. This is shown in the following diagram. This is the fundamental task of model creation.

A diagram of a system model

AI-generated content may be incorrect.

## Scope

When performing threat model creation, it is essential to limit the scope of the element under consideration. The goal of the threat modeling exercise is to establish that appropriate controls are in place to ensure the cybersecurity posture of the element, not other elements which connect to it. When properly created, the threat models of multiple interacting elements will require no modification if composed into a larger abstraction.

**Note:** There is a very strong temptation to try and consider the other elements when threat modeling. Doing so will only waste time as the development SME is providing information only for the element under consideration. Additionally, other elements will be subject to threat models of their own.

**Note:** There is also a very strong temptation to try fit everything conceivable into the element scope. This will result in either an incomplete model or an overly complicated one.

## Goal

The goal of threat modeling is the establishment of a set of controls necessary to ensure that data crossing a threat boundary can do so securely. Threat modeling cannot assess whether behaviors within a process are correct.

# Data Flow Diagram Entities

The AVCDL video **Cybersecurity Requirements-based Threat Modeling [7]** describes the creation of a data flow diagram (DFD) from the block view of a system and shows how that decomposition changes based on the level of abstraction. There are five entity types used within the DFD. These are **resources**, **processes**, **interactors**, **boundaries**, and **data flows**.

**Note:** The combination of a **resource** and its associated owning **process** separated from other processes by a **boundary**, constitute an asset.

**Note:** With the exception of boundaries, entity instance names should be unique. In the case of data flows the name should be of the form **<source> to <destination>**. This is so that issues can be unambiguously identified.

## Resources

In their simplest form, resources are data stores. They represented diagrammatically by cylinders as shown below.

A black background with a black square

AI-generated content may be incorrect.

Resources have the following attributes:

* + Has unique ID
  + Is read-only
  + Is structured
  + Contains PII
  + Contains credentials
  + Is encrypted
  + Is integrity checked
  + Data type
  + Analog
  + Executable
  + Configuration data
  + Data log
  + Audit log

## Processes

Processes are responsible for management of resources. They are represented diagrammatically by circles as shown below.

A black background with a black square

AI-generated content may be incorrect.

Processes have the following attributes:

* + Has unique ID
  + Is authenticated
  + Is authorized
  + Type
  + Driver
  + Service
  + Process
  + Class
  + OS
  + Third-party
  + Self
  + External

**Note:** In **DFD3 [9]**, the process entity is represented using a rounded rectangle. The choice of symbol in this instance should be made in order to communicate consistently.

## Interactors

Interactors are non-process entities external to the element. They are represented diagrammatically by rectangles as shown below.

A black background with a black square

AI-generated content may be incorrect.

Interactors have the following attributes:

* + Type
  + Person
  + Analog

## Trust Boundaries

Trust boundaries are used to denote a specific region of trust. They are represented diagrammatically by dashed arcs as shown below.

A red text on a black background

AI-generated content may be incorrect.

Boundaries have the following attributes:

* + Type
  + Machine
  + Network
  + Interprocess
  + OS
  + Resource
  + External interactor

**Note:** Abstractions such as hypervisors would use **machine** boundaries.

**Note:** For emphasis, boundaries are drawn with the convex side toward the element being modeled. The resultant diagram emphasizes the element having a series of boundary shells. This is illustrated in the following diagram where the resource is the focus of protection.

A diagram of process boundary

AI-generated content may be incorrect.

## Data Flows

Data flows are used to denote any data passing from one non-boundary entity to another. They are represented diagrammatically by solid arcs with an arrow indicating the destination of the data flow, as shown below.

Blue text on a black background

AI-generated content may be incorrect.

Data flows have the following attributes:

* + Requires acknowledgement
  + Has sequence numbers
  + Has authenticated source
  + Has authenticated destination
  + Is safety-critical
  + Integrity check
  + None
  + Simple (CRC)
  + Cryptographic
  + Encryption
  + None
  + Symmetric
  + Asymmetric

**Note:** Double-ended arced arrows are not used as they imply symmetries which are not necessarily true. Always show each data flow separately. Additionally, analysis is not performed on outbound data flows as they do not impact the cybersecurity of the element under consideration.

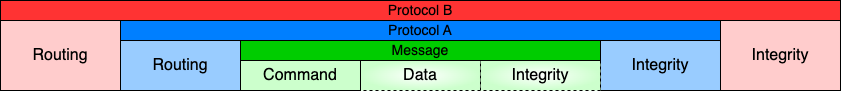
# Methodology

## Resolution Determination

The first step in creation of the threat model is the determination of the model’s resolution.

**Note:** It is not the goal of the threat model creation activity to produce a single all-encompassing diagram of the system. It is expected that there will be a number of diagrams necessary to cover the model. The exact number will depend upon the complexity of the element’s scope.

To appreciate why multiple diagrams are necessary, consider the following diagram from the AVCDL **Cybersecurity Requirements Taxonomy [8]** video.



A black background with pink circles and a black background

AI-generated content may be incorrect.

Here we have a message with multiple intermediators between the intended source and destination. This is a fairly typical situation. If the model covers the dataflows shown in black, but excludes those in red (dashed), certain types of cybersecurity design flaws will be overlooked.

Consider the following concrete example of a simple system with a core service providing information to the larger system.

A diagram of a service application

AI-generated content may be incorrect.

We can see that the core service process uses a read-only configuration file and read-write application data file. If we create a DFD directly from this diagram, we get the following:

A diagram of a service

AI-generated content may be incorrect.

The translation to a DFD is fairly straightforward.

**Note:** The resources have been annotated to indicate the distinction between structured and unstructured data.

There is, however, a level of abstraction not being represented. The following exposes this level.

Diagram of a database manager

AI-generated content may be incorrect.

The **core service** accesses the **configuration data** and **application data** via the **filesystem** and **database manager** services (shown in **red**). Additional trust boundaries are now evident.

But even this representation hides information. The following diagram reveals one additional level of abstraction.

A diagram of a server

AI-generated content may be incorrect.

Here we see the intermediation between the **database manager** and the file holding the database.

## Resource Ownership

When constructing the element model, it is important to ensure that every resource has one owning (managing) process. As seen about in the discussion of DFD levels, the **core service data** is managed by only a single process regardless of the level. One can consider that ownership is delegated as the level increases.

**Note:** Resources with multiple owners indicate a fundamental flaw in the element’s design.

## Operating System Abstraction

One of the most overlooked abstractions is that of the operating system. The cybersecurity controls that can be applied to an element are constrained by the sophistication of the operating system.

Consider the following operating system decomposition by scale.

A screenshot of a computer program

AI-generated content may be incorrect.

As we move from **tiny** to **large** scale, more cybersecurity controls become useable. At the same time, there are many more processes and trust boundaries to account for.

Fundamentally, the operating system provides four core processes. These are:

* + Filesystem
  + Process manager
  + Interprocess communications (IPC) manager
  + Network manager

Depending upon the scale of the operating system, one or more of these may be absent.

In medium and large scale operating systems, a number of other processes (managers) are provided. If we think about the element from the point of view of an individual process within the element, we have the following arrangement.

A circular diagram of a computer process

AI-generated content may be incorrect.

This view makes it clear where trust boundaries exist.

**Note:** In this view, a resource is treated like any other external entity (managed by some operating system process).

## Open Source / Third-Party Abstraction

Both open source and third-party processes should be treated the same as operating systems ones.

**Note:** It is just as important to understand the trust relationships between the element and any open source or third-party processes as it is to do so with the operating system processes.

# DFD Creation

Since attempting to perform threat modeling using a single DFD representing all possible data flows for all workflows is infeasible, a set of DFDs must be created. These allow for the ease of analysis and issue tracking.

## Element-only DFD

This first DFD created focuses only on those entities within the element owned by the organization and the external entities with which they interact.

**Note:** This view will not be used in the threat modeling activity. It serves as a reference to anchor the element’s workflow DFDs.

## Non-element Process DFDs

Because the non-element processes (operating system, third-party, and open source) are not monolithic, a DFD should be created for each. This allows specific cybersecurity controls to be assessed without distracting from the element’s workflows.

## Workflow DFDs

The workflow-level DFDs are limited in scope to a single operational activity. This both allows for easy addition of new workflows and also simplifies analysis (and by extension mitigation).

**Note:** It is understood that as the complexity of the element increases, so does the number of workflow DFDs required to fully describe the element.

# Threat Model Template

The element cybersecurity requirements catalog may be documented using the **AVCDL threat modeling model template** Microsoft Excel workbook **[4]**.

**Note:** Other forms of documentation are permissible so long as they provide the information laid out in this document.

There are eleven sheets in the workbook. They are:

* + [Cover sheet](#_Cover_Sheet)
  + [Revision history](#_Revision_History)
  + [Reference documents](#_Reference_Documents)
  + [Diagrams](#_Diagrams)
  + [DFDs](#_DFD_Entities)
  + [Resource entities](#_Resource_Entities)
  + [Process entities](#_Process_Entities)
  + [Interactor entities](#_Interactor_Entities)
  + [Boundary entities](#_Boundary_Entities)
  + [Data flow entities](#_Data_Flow_Entities)
  + [Legend](#_Legend)

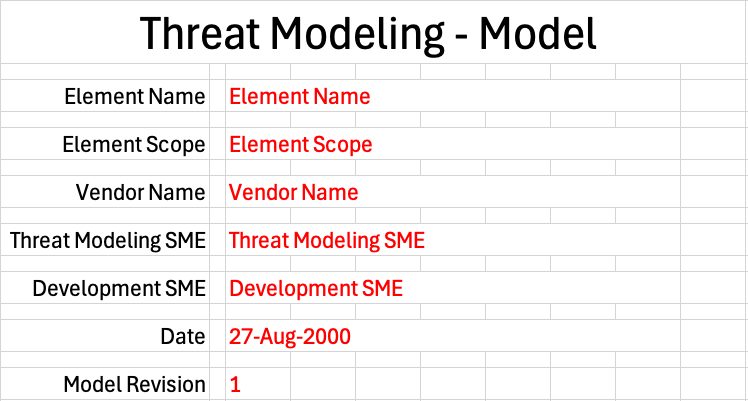
These will be addressed in turn.

## Duplication of Rows in the Various Sheets

When there is the need to add rows to the various sheets of the workbook, be sure to duplicate an existing row. This is because validation checks are attached to some of the cells which also enables the use of dropdown lists.

## Cover Sheet

The **cover sheet** of the workbook is shown below:



Fields to be completed are shown in **red**.

### Element Name

The **element name** is the element being modeled.

### Element Scope

The scope of the element is left to the discretion of the customer. Examples of element scopes include system, sub-system, component, component software (non-OS), and component OS.

**Note:** In this context, **the** **customer** refers to the entity requiring the activity.

### Vendor Name

This is the name of the vendor responsible for the element being modeled.

### Threat Modeling SME

This is the threat modeling subject matter expert creating the threat model.

### Development SME

This is the development subject matter expert providing element information for the creation of the threat model.

### Date

This is the date when the element’s threat model was created or updated. The date should be updated whenever the model is updated.

### Revision

This is the revision number of the document. The revision number is a monotonic and increasing integer, starting at 1. It should be incremented every time the document is updated.

## Revision History

The **revision history** sheet of the workbook is shown below:

A close-up of a computer screen

AI-generated content may be incorrect.

### Revision

The **revision** corresponds to that listed on the cover sheet.

### Author

The **author** corresponds to the threat modeling SME listed on the cover sheet.

### Description

This is a brief description of changes made to the threat model since it was last updated.

## Reference Documents

**Note:** These references are those necessary for the creation of the element’s threat model.

The **references** sheetof the workbook is shown below:

A document with text on it

AI-generated content may be incorrect.

### Name

This is the name of the document being referenced.

### Description

This is a brief description of the document being referenced.

### Location

This is the location of the document being referenced. It may be a physical location or a URL.

## Diagrams

The **diagrams** sheet of the workbook is shown below:

A close-up of a document

AI-generated content may be incorrect.

Element diagrams are inserted images with attendant explanations. The intent of these diagrams is to provide a mechanism for establishing the various entities necessary for the creation of an accurate threat model. These diagrams show the various relationships between various features of the element, a context which lists alone cannot provide.

The utility of the element model diagram is that it aides in the collection of the threat model inputs. In practice, each of these features should be uniquely enumerated to clearly document issues identified in the analysis.

The unique identifiers (names) of the various parts of the diagram should correspond to those used in the other sections of the threat modeling analysis workbook.

There are many possibilities for the diagrams included. These range from annotated photographs or CAD drawings in the case of physical elements, block diagrams which are useful at the system level, and DFDs for purely software scope elements.

The requirement for this section is that at least one visualization that appropriately conveys the structure of the element be included.

## DFDs

The **DFDs** sheet of the workbook is shown below:

A close-up of a document

AI-generated content may be incorrect.

Threat model DFDs are inserted images with attendant explanations. These diagrams provide a visualization of the information contained in the following sheets.

The utility of these diagrams is that they allow for the threat modeling SME performing the threat model analysis to provide workflow-specific DFDs showing only necessary entities.

**Note:** For ease of use by the threat modeling SME responsible for analyzing the model, it is recommended to replicate the **DFDs** sheet and add a suffix to the name.

## Resource Entities

The **resource entities** sheet of the workbook is shown below:

A close-up of a document

AI-generated content may be incorrect.

### ID

This is the unique threat model ID of the resource.

### Name

This is the name of the resource.

### Description

This is the description of the resource.

### Has Unique ID

This indicates whether the resource has a unique identifier within the context of the element.

### Is Read-only

This indicates whether the resource is read-only.

### Is Structured

This indicates whether the resource storage is structured (database). This determines whether there is an intermediating process needed to manage it (DBMS).

### Contains PII

This indicates whether the resource contains PII data.

### Contains Credentials

This indicates whether the resource contains credential data.

### Is Encrypted

This indicates whether the resource is encrypted.

### Is Integrity Checked

This indicates whether the resource provides data integrity checking.

### Resource Type

This is the type of data stored in the resource. The type may be one of the following:

* + generic
  + analog
  + executable
  + configuration data
  + data log
  + audit log

### Notes

This is a general notes field.

## Process Entities

The **process entities** sheet of the workbook is shown below:

A close-up of a document

AI-generated content may be incorrect.

### ID

This is the unique threat model ID of the process.

### Name

This is the name of the process.

### Description

This is the description of the process.

### Has Unique ID

This indicates whether the process has a unique identifier within the context of the element.

### Is Authenticated

This indicates whether the process is authenticated before use.

### Is Authorized

This indicates whether the process is authorized for the operations it carries out.

### Process Type

This is the type of the process. The type may be one of the following:

* + process
  + driver
  + service / daemon

### Process Class

This is the owner of the process. The class of the process carries both trust and ownership implications. The type may be one of the following:

* + OS
  + Third-party
  + Self
  + External

**Note:** **Third-party** encompasses all processes which are neither OS nor self in ownership within the scope of the element. Open source software processes are included in this.

**Note:** **External** encompasses all processes which are outside of the scope of the element.

### Notes

This is a general notes field.

## Interactor Entities

The **interactor entities** sheet of the workbook is shown below:

A close-up of a table

AI-generated content may be incorrect.

### ID

This is the unique threat model ID of the interactor.

### Name

This is the name of the interactor.

### Description

This is the description of the interactor.

### Interactor Type

This is the type of the interactor. The type may be one of the following:

* + Person
  + Analog

### Notes

This is a general notes field.

## Boundary Entities

The **boundary entities** sheet of the workbook is shown below:

A close-up of a table

AI-generated content may be incorrect.

### ID

This is the unique threat model ID of the boundary.

### Name

This is the name of the boundary.

### Description

This is the description of the boundary.

### Boundary Type

This is the type of the boundary with respect to the element. The type may be one of the following:

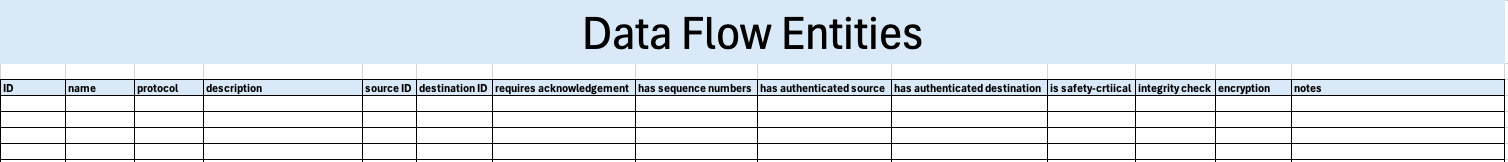
* + Interprocess
  + Machine
  + Network
  + OS
  + Resource
  + External Interactor

### Notes

This is a general notes field.

## Data Flow Entities

The **data flow entities** sheet of the workbook is shown below:



### ID

This is the unique threat model ID of the data flow.

### Name

This is the name of the data flow.

**Note:** It is recommended that the name be of the form **<source> to <destination>**.

### Protocol

This is the name of the protocol used by the data flow.

### Description

This is the description of the data flow.

### Source ID

This is the unique threat model ID of the source of the data flow.

### Destination ID

This is the unique threat model ID of the destination of the data flow.

### Requires Acknowledgement

This indicates whether the data flow’s protocol requires acknowledgement of receipt of the data.

### Has Sequence Numbers

This indicates whether the data flow’s protocol contains sequence numbers.

### Has Authenticated Source

This indicates whether the data flow’s protocol has source authentication.

### Has Authenticated Destination

This indicates whether the data flow’s protocol has destination authentication.

### Is Safety-critical

This indicates whether the data flow is safety-critical in nature.

### Integrity Check

This indicates the data flow protocol’s data integrity checking mechanism. This may be one of the following:

* + None
  + Simple (CRC)
  + Cryptographic

### Encryption

This indicates the data flow protocol’s encryption mechanism. This may be one of the following:

* + None
  + Symmetric
  + Asymmetric

### Notes

This is a general notes field.

## Legend

The **legend** sheet of the workbook is shown below:

A table with text on it

AI-generated content may be incorrect.

The **legend** sheet information is used to make the completion of the document easier by providing dropdown lists for common values. It also ensures that spelling errors do not creep into the generated material.

**Note:** The legend sheet should not be edited. If an unlisted value is required, the template should be separately revised.

# Exit Criteria

This procedure is considered complete once the generated output has been entered into the organization’s DMS as a document of record.

**Note:** The processes and procedures for entering documents into the DMS, or the updating thereof, are outside the scope of this document.

# References

1. **AVCDL** (AVCDL primary document)
2. **Threat Modeling Report** (AVCDL secondary document)
3. **Security Requirements Taxonomy** (AVCDL secondary document)
4. **AVCDL threat modeling model template** (AVCDL template)
5. **Understanding TARA in an AVCDL Context** (AVCDL elaboration document)
6. **Attack Surface Analysis Report** (AVCDL secondary document)
7. **Cybersecurity Requirements-based Threat Modeling** (AVCDL video)
8. **Cybersecurity Requirements Taxonomy** (AVCDL video)
9. **DFD3**[**https://github.com/adamshostack/DFD3**](https://github.com/adamshostack/DFD3)