A Step-by-step Guide to Create Your First Threat Model (Template Included)

Introduction - What is Threat Modeling

A structured and repeatable process to identify threats and mitigate them against valuable assets in a system. We cannot build secure systems until we understand the applicable threats to our applications/ systems/platforms/infrastructure/services/APIs etc. Threat Modeling involves (i) visually modeling a system (ii) identifying potential threats (iii) validating and/or designing security controls to mitigate risk(s).

Threat Modeling versus Threat Intelligence

While both Threat Modeling (TM) and Threat Intelligence (TI) focus on identifying threats in order to act on them or mitigate them, Threat Modeling aligns well with the Security architecture/design portion of

Secure Development Lifecycle, whereas Threat Intelligence aligns well with security operations. Threat Modeling is relevant to identifying threats in a particular system/application/platform/service that we are building before that system is deployed in production, whereas Threat Intelligence is relevant to a comprehensive list of Threats to a whole organization with reference to systems that are already in production/non-prod/pre-prod/laptops/desktops, etc.

Threat Modeling alignment to NIST CSF

Both Threat Modeling (TM) and Threat Intelligence (TI) maps into NIST CSF Identify (ID) → Risk Assessment (ID.RA) category

Function	Category	Sub-category
IDENTIFY (ID)	**Risk Assessment (ID.RA):** The organization understands the cybersecurity risk to organizational operations (including mission, functions,image, or reputation), **organizational assets, and individuals.**	**ID.RA-3:** Threats, both internal and external, are identified and documented

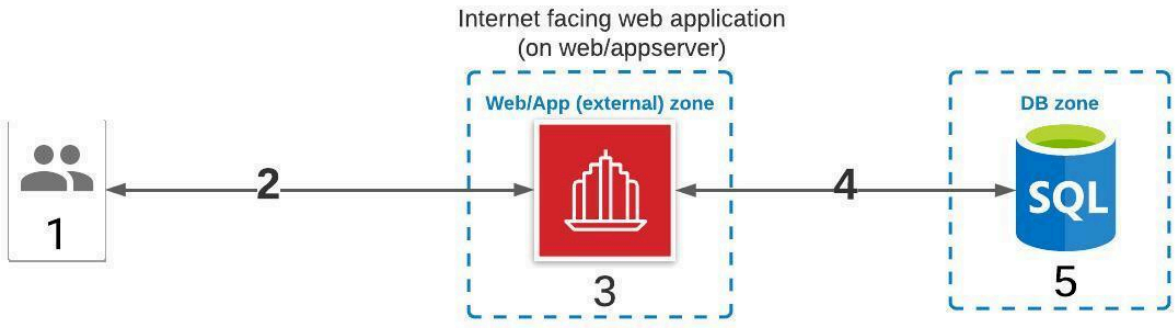
A simple, six-step approach to Threat Modeling

The following describes a simple, six-step approach to perform threat modeling:

1. Create an architecture diagram of the application/system by:
    a. depicting each architectural component as one of the four threat modeling elements. Any architectural component which is not an actor/data flow/data store would be a process from the threat modeling perspective.
    b. assign a number to each architectural component for each reference in later steps.
2. List down each architectural component matching the assigned numbers or identifiers in the diagram (eg. as rows in a spreadsheet) along with mapping to the corresponding threat modeling element those components fall into.
3. For each such architectural component, duplicate the row as many times as there are applicable threats based on the STRIDE applicability matrix and assign an applicable threat for that component in each row. For example, for an actor, there would be two rows (one for Spoofing threat and second row for Repudiation threat as there are two applicable threats as per STRIDE applicability matrix. Similarly, there would be four rows for a database, as there are four applicable threats for a data store).
4. Think about how such a threat could make a contact or exploit a vulnerability in the component and manifest into a real risk to the application/system that is being threat modeled. Write down or explain the threat description in a simple sentence or two
5. Think about if the threat is real or not and how a set of security controls (one or many) that are already in place or going to be implemented could mitigate the potential risks. Propose such mitigation plan in a simple sentence or two
6. Identify the appropriate security control(s) from NIST CSF. Each such security control should be placed in the next column on the same row. Note that there could be many-to-many relationships between potential threats and possible mitigation controls. (one security control may mitigate multiple threats and one threat may need multiple controls for risk mitigation).

Let's take a simple internet facing web application architecture to walk through the six (6) steps described above.

Step 1: Create an architecture diagram and label the artifacts



Step 2: List down each architectural component

Artifact depicted in the diagram	TM Element
(1) Human user (customer/employee/partner) using a web browser	Actor
(2) Data flow between user/browser and web/app server	Data flow
(3) Web Application (app.organization.com)	Process
(4) Data flow between web/app server and database	Data flow
(5) Database	Data store

Step 3: Identify and assign potential threats from STRIDE applicability matrix

Artifact depicted in the diagram	TM Element	Applicable Threats (STRIDE Classification)
(1) Human user (customer/employee/partner) using a web browser	Actor	Spoofing
(1) Human user (customer/employee/partner) using a web browser	Actor	Repudiation
(2) Data flow between user/browser and web/app server	Data flow	Tampering
(2) Data flow between user/browser and web/app server	Data flow	Information disclosure
(2) Data flow between user/browser and web/app server	Data flow	Denial of service

Step 4: Describe threat description

Analyze Model		Identify Threats	
Artifact depicted in the diagram	TM Element	Applicable Threats (STRIDE Classification)	Threat description
(1) Human user (customer/employee/partner) using a web browser	Actor	Spoofing	An attacker could pretend to be a valid customer and try to access unauthorized details
(1) Human user (customer/employee/partner) using a web browser	Actor	Repudiation	An authorized user (e.g., w/ admin privs) might delete/edit customer data and could claim to have not performed that action
(2) Data flow between user/browser and web/app server	Data flow	Tampering	An attacker could modify data as it traverses internet to the web/app server
(2) Data flow between user/browser and web/app server	Data flow	Information disclosure	An attacker could sniff network traffic to read sensitive data in transit
(2) Data flow between user/browser and web/app server	Data flow	Denial of service	An attacker could launch DoS/DDoS to degrade the availability of a web application/service to users

Step 5: Propose risk mitigation plan

Artifact depicted in the diagram	TM Element	Applicable Threats (STRIDE Classification)	Threat description	How we plan to mitigate the risk(s)
(1) Human user (customer/employee/partner) using a web browser	Actor	Spoofing	An attacker could pretend to be a valid customer and try to access unauthorized details	Implemented or plan to implement strong authentication
(1) Human user (customer/employee/partner) using a web browser	Actor	Repudiation	An authorized user (e.g., w/ admin privs) might delete/edit customer data and could claim to have not performed that action	Implemented or plan to implement log monitoring for operations on sensitive data by users
(2) Data flow between user/browser and web/app server	Data flow	Tampering	An attacker could modify data as it traverses internet to the web/app server	Implemented or plan to implement encryption of data in-transit using strong cryptography
(2) Data flow between user/browser and web/app server	Data flow	Information disclosure	An attacker could sniff network traffic to read sensitive data in transit	Implemented or plan to implement encryption of data in-transit using strong cryptography
(2) Data flow between user/browser and web/app server	Data flow	Denial of service	An attacker could launch DoS/DDoS to degrade the availability of a web application/service to users	1. Implemented or plan to implement firewalls at appropriate levels in the network to reduce the attack surface 2. Implemented secure network configuration

Step 6: Identify appropriate security controls from NIST CSF

Analyze Model		Identify Threats		Mitigation Plan			
Artifact depicted in the diagram	TM Element	Applicable Threats (STRIDE Classification)	Threat description	How we plan to mitigate the risk(s)	Relevant or applicable NIST CSF control(s)		
(1) Human user (customer/ employee/ partner) using a web browser	Actor	Spoofing	An attacker could pretend to be a valid customer and try to access unauthorized details	Implemented or plan to implement strong authentication	PR.AC-7: Users, devices, and other assets are authenticated (e.g., single-factor, multi-factor) commensurate with the risk of the transaction (e.g., individuals' security and privacy risks and other organizational risks)		
(1) Human user (customer/ employee/ partner) using a web browser	Actor	Repudiation	An authorized user (e.g., w/ admin privs) might delete/edit customer data and could claim to have not performed that action	Implemented or plan to implement log monitoring for operations on sensitive data by users	PR.PT-1: Audit/log records are determined, documented, implemented, and reviewed in accordance with policy	DE.AE-3: Event data are collected and correlated from multiple sources and sensors	

(2) Data flow between user/browser and web/app server	Data flow	Tampering	An attacker could modify data as it traverses internet to the web/app server	Implemented or plan to implement encryption of data in-transit using strong cryptography	PR.DS-2: Data-in-transit is protected		
(2) Data flow between user/browser and web/app server	Data flow	Information disclosure	An attacker could sniff network traffic to read sensitive data in transit	Implemented or plan to implement encryption of data in-transit using strong cryptography	PR.DS-2: Data-in-transit is protected		

For full threat model, refer to “Threat Model for 2-tier web app” worksheet at:
➕ Template: Creating a Manual Threat Model in Six Steps
Architecture diagrams are on the first worksheet “Architecture diagrams” for additional reference.

Manual Threat Modeling Tool using a spreadsheet

The template for creating a threat model manually in six steps using a spreadsheet is made available at the link below. The template could be customized further to make it work with any security standard or framework instead of NIST CSF or with an organization's internal security standard.
➕ Template: Creating a Manual Threat Model in Six Steps

References

1. Microsoft Security Development Lifecycle
2. Introduction to Microsoft SDL Threat Modeling
3. Threat Modeling - Designing for Security
4. Securing Systems - Applied Security Architecture and Threat Models

Appendix 1: Primer to STRIDE framework

Threat Classifications

There are **six classifications** of Threats dubbed as **STRIDE (S**poofing**, T**ampering**, R**epudiation**, I**nformation disclosure**, D**enial of service**, E**levation of privilege**)** as described below. The STRIDE approach to threat modeling was invented in 1999.

Threat Classification	**Definition**	**Sample Threats**	**Desired security control to mitigate the threat**	**Risk mitigation solution**
Spoofing	Impersonating something or someone else	Pretending to be a valid user or server	**Authentication**	Enforce strong authentication techniques like 2FA for human authentication, client certs for non-human (API) clients
Tampering	Modifying data/ code unauthorized	Modifying code (or library) on a system / data on disk	**Integrity**	Enforce strong cryptography/ hashing
Repudiation	Claiming to have not performed an action	Remove record of modification of a file / resource	**Non-Repudiation**	Enforce logging on key events of interest. Use digital signatures
Information disclosure	Exposing information to someone not authorized	Gathering sensitive information from log files	**Confidentiality**	Enforce strong cryptography/ encryption
Denial of service	Deny or degrade service to legitimate /	Crashing a website	**Availability**	Use Throttling to control resource usage or design/build resiliency at the server level
Elevation of privilege	Gain capabilities without proper authorization	Allowing remote user to run commands, switch from a limited user to admin	**Authorization**	Enforce principle of least privilege

Threat Modeling Elements

There are four elements used in Threat Modeling:

1. **Actor** - Users (typically human users, but don't need to be. It could be clients like browsers or devices with IP address or physical address)
2. **Data Store** - Databases, File systems, LDAP, Cookies, Memory-Cache
3. **Data Flow** - HTTPS, IPSEC, RPC
4. **Process** (runs code) - Web application/service, OS process, VM/Host/Server

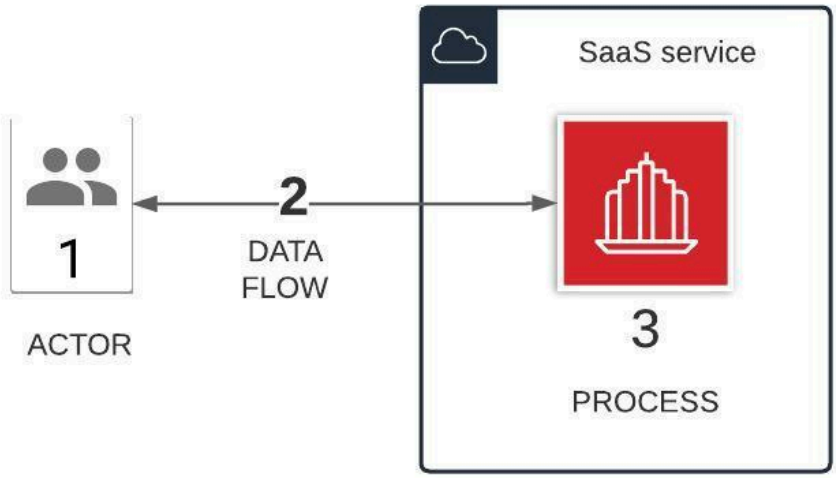
STRIDE applicability to TM elements

Not all the threats apply to every element in the architecture diagram. Matrix of the applicability of threats to actors is shown in the table below:

	Spoofing	Tampering	Repudiation	Information Disclosure	Denial of service	Elevation of privilege
Actor	X		X			
Data store		X	X	X	X	
Data flow		X		X	X	
Process	X	X	X	X	X	X

Appendix 2: Sample Threat Models

SaaS application (public cloud hosted)



Refer "Threat Model for SaaS application" worksheet at:
Template: Creating a Manual Threat Model in Six Steps