

and a typical pentest

Threat Modeling

- Step 1: Read all documentation
- Step 2: Use the application to perform basic tasks
- Step 3: Watch how data moves
- Step 4: Define user roles
- Step 5: Define assets
- Step 6: Define components
- Step 7: Access matrix
- Step 8: Component diagram
- Step 9: Threat tree

Threat Based Testing – Threat Modeling Introduction to Threat Modeling – What is Threat Modeling

- Software development is about creating applications that enable users to perform some tasks
- Secure development requires determining what a user shouldn't do and ensuring that the code properly restricts users to authorized actions.
- Threat modeling is a design activity to do just that

Introduction to Threat Modeling – How Threat Modeling Helps

Threat Modeling enables you to:

- Identify threats
- Identify vulnerabilities
- Identify mitigating factors
- Perform risk analysis
- Prioritize security fixes
- Derive security test cases

Introduction to Threat Modeling - Threat Modeling Walkthrough

- For our threat modeling walkthrough, we will have an example in which we model a simple online store application that allows users to buy sport equipment
- We will apply the process step-by-step to our example

Threat Based Testing – Threat Modeling The Threat Modeling Process

Threat Modeling Process

- Collecting Information
- Decomposing the Application
 - Identifying Entry Points
 - Identifying Assets
 - Identifying Roles
- Building the Activity Matrix
- Building the Threat Profile
 - Identifying Threats
 - Classifying Threats
 - Building Threat Trees
 - Identifying Vulnerabilities
- Analyzing Risks



The Threat Modeling Process – Collecting Information

Background information:

- Can be collected relatively fast
- Is crucial for a good start of a threat model
- Helps to understand the application and its basic purposes
- Provides better understanding of threat mitigations
- Can be used throughout the entire iterative threat modeling process

The Threat Modeling Process – Collecting Information

- There are four main sources of background information:
 - Specifications
 - Implementation Assumptions
 - External Dependencies
 - Internal and External Security Notes

The Threat Modeling Process – Collecting Information: Specifications

- Usually include:
 - Customer requirements
 - Intended purposes
 - Use cases
- Define the primary functionality of the system
- Scope the threat model by providing common and uncommon uses of the System Under Test (SUT)
- Can be used later to analyze the threats that emerge depending on the specific use case

The Threat Modeling Process – Collecting Information: Implementation Assumptions

- Decisions made before developing or during architectural or project revisions
- Capture basic architectural and design assumptions that may raise security issues
- List features that may increase the attack surface of the SUT
- Help in defining mitigations to specific threats.

The Threat Modeling Process – Collecting Information: External Dependencies

- List the software components which the SUT relies to function properly
- Can be used to construct dependency contracts to capture third-party security concerns

Threat Based Testing – Threat Modeling The Threat Modeling Process – Collecting Information: Security Notes

- Internal and external security notes
- Hidden security concerns and steps that were take against them
- Are used to capture the security assumptions from an architectural point of view
- Help to make the threat model more clear
- Aid in defining mitigations for threats discovered during step 4 (Building the Threat Profile)

The Threat Modeling Process – Collecting Information: Example

- High-level information for our online store application:
 - The application stores customer data such as shipping and billing addresses, and credit card numbers
 - The application interfaces with a 3rd party payment processing system
 - The application interfaces with a separate inventory system to manage stock and re-orders

Threat Based Testing – Threat Modeling The Threat Modeling Process

Threat Modeling Process

- Collecting Information
- Decomposing the Application
 - Identifying Entry Points
 - Identifying Assets
 - Identifying Roles
- Building the Activity Matrix
- Building the Threat Profile
 - Identifying Threats
 - Classifying Threats
 - Building Threat Trees
 - Identifying Vulnerabilities
- Analyzing Risks



The Threat Modeling Process - Decomposing the Application

Decomposing the application...

- Is key to define the main elements of a threat model
- Provides a more structured and formal approach to threat modeling
- Is a great exercise to understand the inner workings of the software being modeled
- Helps to find threats during threat discovery phase

The Threat Modeling Process - Decomposing the Application

- Decomposing the application consists of three steps:
 - Identifying Entry Points
 - Identifying Assets
 - Identifying Roles

Threat Based Testing – Threat Modeling The Threat Modeling Process – Decomposing the Application: Identifying

The Threat Modeling Process – Decomposing the Application: Identifying Entry Points

- Find the sources of input to your application. List all the points in which your system receives data from outside
- List all components that receive hidden sources of input such as components that interact with the file system, registry, RPC/DCOM, memory, etc.
- Collect entry points by looking at background information (use cases or external dependencies will reveal entry points for a threat model)

The Threat Modeling Process – Decomposing the Application: Identifying Entry Points Example

- Identifying entry points in our online store application:
 - Front-end Web server
 - Merchandise database
 - Interface with 3rd-party credit card processing system
 - Interface with inventory system

The Threat Modeling Process – Decomposing the Application: Identifying Assets

- To find assets one needs to think about what the attacker will target
- When enumerating threats during the next step, you will see that most threats relate to an attacker exploiting or stealing an asset
- While doing this exercise you might start encountering threats. Note them down for later use.

The Threat Modeling Process – Decomposing the Application: Identifying Assets Example

- Identifying assets in our online store application:
 - Customer data
 - Checkout cart
 - Merchandise
 - Inventory system
 - The Web application

Threat Based Testing – Threat Modeling The Threat Modeling Process – Decomposing the Application: Identifying Roles

- Roles reflect the different privileges included in your application
- They are nouns that usually translate to the different users of the system (user, admin/root, guest, wheel, etc), but can also can refer to different privilege levels such as user mode vs. kernel mode
- Each entry point and asset will have an associated list of roles
- Noting down the roles per entry point or asset might reveal escalation of privilege or information disclosure threats

The Threat Modeling Process – Decomposing the Application: Identifying Roles Example

- Identifying roles in our online store application:
 - Web customer
 - 3rd-party payment processing system

Threat Modeling Process

- Collecting Information
- Decomposing the Application
 - Identifying Entry Points
 - Identifying Assets
 - Identifying Roles
- Building the Activity Matrix
- Building the Threat Profile
 - Identifying Threats
 - Classifying Threats
 - Building Threat Trees
 - Identifying Vulnerabilities
- Analyzing Risks



Threat Based Testing – Threat Modeling The Threat Modeling Process – Building the Activity Matrix

- The activity matrix is a set of explicit mappings between roles and asset
- Each <role, asset> pair lists the access types granted to a role for the asset
- The activity matrix is used in later steps to derive threats to the system based on improper asset access

Threat Based Testing – Threat Modeling The Threat Modeling Process - Building the Activity Metrix: Even

The Threat Modeling Process – Building the Activity Matrix: Example

Building the activity matrix for our online store application:

	Customer Data	Merchandise	[]
Web Customer	Read: Own = always Other = never	Read: always Modify: never	
	Modify: Own = always Other = never		
[]			

Threat Modeling Process

- Collecting Information
- Decomposing the Application
 - Identifying Entry Points
 - Identifying Assets
 - Identifying Roles
- Building the Activity Matrix
- Building the Threat Profile
 - Identifying Threats
 - Classifying Threats
 - Building Threat Trees
 - Identifying Vulnerabilities
- Analyzing Risks



The Threat Modeling Process – Building the Threat Profile

A Threat Profile is:

- A list of threats
- A threat tree for each of the discovered threats
- A description of mitigations
- A list of vulnerabilities

Threat Based Testing – Threat Modeling The Threat Modeling Process – Building the Threat Profile

- This step uses all the information collected to this point from the previous steps:
 - Use cases serve to identify threats in specific scenarios
 - Security notes, external dependencies, and implementation assumptions imply where to look and narrow the scope
 - Data Flow Diagrams are great resources to understand the attack surface of your application
 - Assets are target of threats
 - Entry points give context, help to identify attacks
 - Roles affect threat mitigations

The Threat Modeling Process - Building the Threat Profile

- Building the Threat Profile is achieved with the following four steps:
 - Identifying Threats
 - Classifying Threats
 - Building Threat Trees
 - Identifying Vulnerabilities

The Threat Modeling Process – Building the Threat Profile: Identifying Threats

Threats are possible attacks

- A threat is what an attacker might try to do to an asset or through an entry point
- Threats spring out of the "never" entries in the activity matrix

Threats have the following characteristics:

- They are usually expressed as verbs (actions)
- They involve at least one entry point or one asset
- They are written in the following form:
 - Attacker verb to\from\with\etc asset (for goal)

The Threat Modeling Process – Building the Threat Profile: Identifying Threats Example

- Threat examples from our online store activity matrix:
 - Threat #1: Attacker steals customer information.
 - Threat #2: Attacker connects to merchandise database to delete merchandise thus causing a denial of service

The Threat Modeling Process – Building the Threat Profile: Classifying Threats

STRIDE

- Spoofing
- Tampering with Data
- Repudiation
- Information Disclosure
- Denial Of Service
- Escalation of privilege

The Threat Modeling Process – Building the Threat Profile: Classifying Threats Example

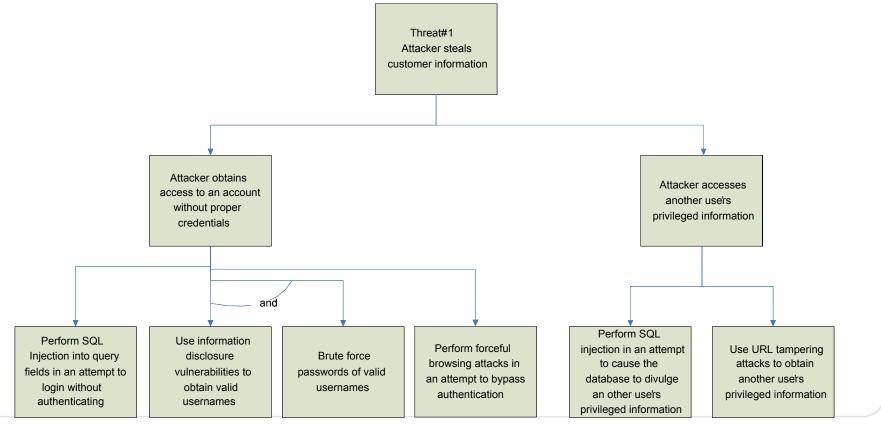
- Classifying our online store threats:
 - Threat #1: Attacker steals customer information
 - Information Disclosure
 - Threat #2: Attacker connects to merchandise database to delete merchandise and cause denial of service
 - Denial of Service

The Threat Modeling Process – Building the Threat Profile: Building Threat Trees

- Threat trees provide comprehensive details about a threat, describing the conditions required to realize it:
 - The root node is the threat
 - Child nodes are the conditions necessary for the threat to realize
- Threat trees are used during penetration testing to construct test cases from the condition nodes

The Threat Modeling Process – Building the Threat Profile: Building Threat Trees Example

- Threat tree example from our online store:
 - Threat #1: Attacker steals customer information





The Threat Modeling Process – Building the Threat Profile: Identifying Vulnerabilities

- Threats and conditions can be mitigated or unmitigated
- Attack paths can be built by identifying unmitigated routes from the leaf conditions to the root threat
- Unmitigated attack paths yield vulnerabilities
- Vulnerabilities inherit the root threats' STRIDE classifications



The Threat Modeling Process – Building the Threat Profile: Identifying Vulnerabilities Example

- Vulnerability from our online store threat tree:
 - A SQL injection vulnerability in the query fields allows an attacker to obtain access to an account without proper credentials
 - Information Disclosure

Threat Based Testing – Threat Modeling The Threat Modeling Process - Building the Threat Profile

The Threat Modeling Process – Building the Threat Profile

- Suggestions for building the threat profile:
 - Arrange a meeting to brainstorm on threats
 - Don't think too much into solutions or mitigations
 - Identify each threat with a proper ID