AS Lab Assignment

Threat Modeling

Intro

In this assignment, you will perform threat modeling for an example application. There is a company that wants to implement a youtube-like application. At this stage they are designing the system and ask you for a security consulting. They want to know what potential issues they may have and how to mitigate them.

There are different approaches to threat modeling, but in this assignment you will be applying process that mostly follows the one that described here:

https://owasp.org/www-community/Threat_Modeling_Process

Your submission have to include:

- Tables with entry points, assets, trust levels
- Data flow diagrams
- Threats summary table
- Report that contains clarifying information about your results

You can provide tables and diagrams inside the report or as standalone files - it is up to you.

Application

You were provided with the following information about the system.

Features:

- upload and delete videos
- get video streams with desired quality
- search available videos (filtering and sorting by some attributes)
- display user profile and uploaded videos
- some videos are private (only specific user) and some are hidden (not shown in search and on user page)
- display view history
- display, create and delete comments on video

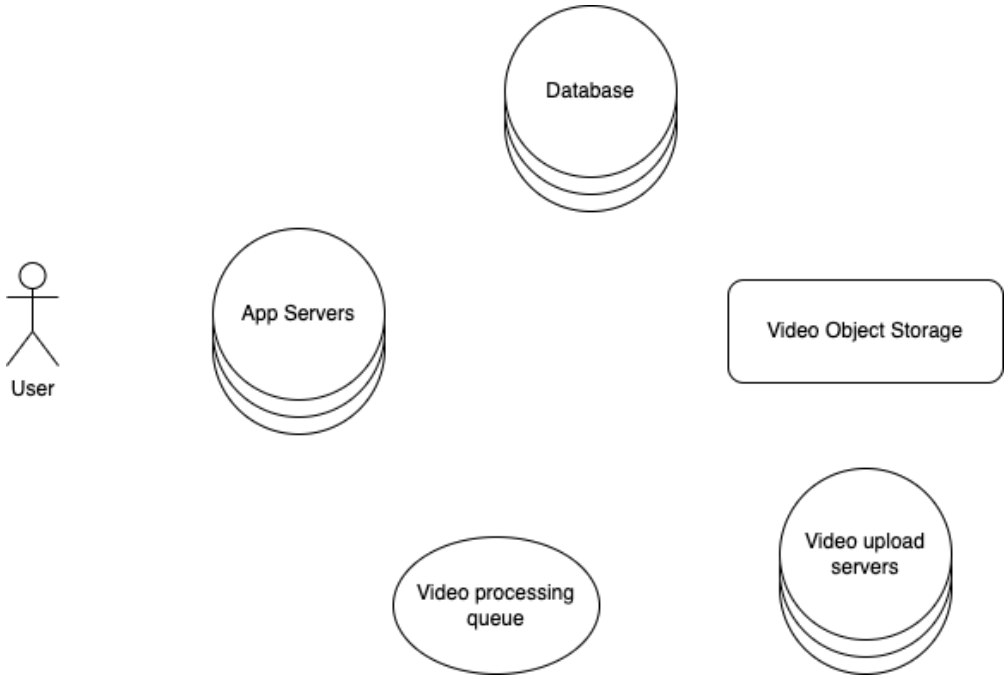
System entities:

- user data
- user upload history
- user view history
- video data
- video objects
- comments

System components:

- Application servers
- Databases
- Video processing queue
- Video uploading servers
- Video object store

Below is a diagram with system components.



1. Decompose the application

At this step you should get an understanding of the application and how it interacts with external entities. This involves gathering information about:

- Entry points - interfaces through which potential attackers can interact with the application.
- Assets - something that the attacker is interested in, it can be some data or a state of the system (for example availability).
- Trust levels - access rights that the application will grant to external entities.
- Data flows - shows flow of control through system components for particular use cases.

Your task is to:

1. Describe entry points, assets and trust levels in form of tables
2. Select at least 3 use cases that you think are the most interesting and prepare Data Flow Diagrams (DFD) for them.

Since the system itself is abstract and not real, you may make assumptions about the system, but please state them explicitly - in the form of clarification in your report.

For reference, please use:

https://owasp.org/www-community/Threat_Modeling_Process#decompose-the-application

As a tool for DFD you can use:

https://www.microsoft.com/en-us/download/details.aspx?id=49168

https://1modm.github.io/threatmodel.html

2. Determine threats

Now when you have decomposed the system you can determine possible threats.

Categorizations such as STRIDE allow to identify threats in the application in a structured and repeatable manner.

Your task is to apply STRIDE for each asset in the application and come up with a summary table with the following columns:

Asset	Category	Threat	Vulnerability	Score	Countermeasure

Asset - for example "User credentials".

Category - according to STRIDE, for example "Information disclosure".
Note that you can skip category, if you think there is no threat for that data flow that falls in that category.

Threat - a threat itself that falls into category, for example "User credentials are exposed and obtained by an attacker".

Vulnerability - a particular flaw in the system that may be exploited and lead to the threat realization, for example "During the authentication process password is passed as plain text" or "Password is stored as plain text in the database".

Score - there are different approaches for threat prioritization, but in this task you will try to do it based on Common Vulnerability Scoring System (CVSS).
https://www.first.org/cvss/calculator/3.0

Countermeasure - provide countermeasures that can be implemented in the system to mitigate that particular vulnerability.
Note that if you made some assumptions during the decomposition part about the environment the system or components are running in, countermeasures may not be required (for example, if a user communicates with our system through a dedicated protected channel, we may say that it is okay to pass a password as plain text) - but in this case assumptions should be explicitly stated in the decomposition part.

For reference, please use:

https://owasp.org/www-community/Threat_Modeling_Process#determine-and-rank-threats

https://owasp.org/www-community/Threat_Modeling_Process#determine-countermeasures-and-mitigation

https://github.com/OWASP/threat-model-cookbook