- **Name**: Iskander Nafikov
- **E-mail**: i.nafikov@innopolis.university
- **GitHub**: https://github.com/iskanred

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

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:

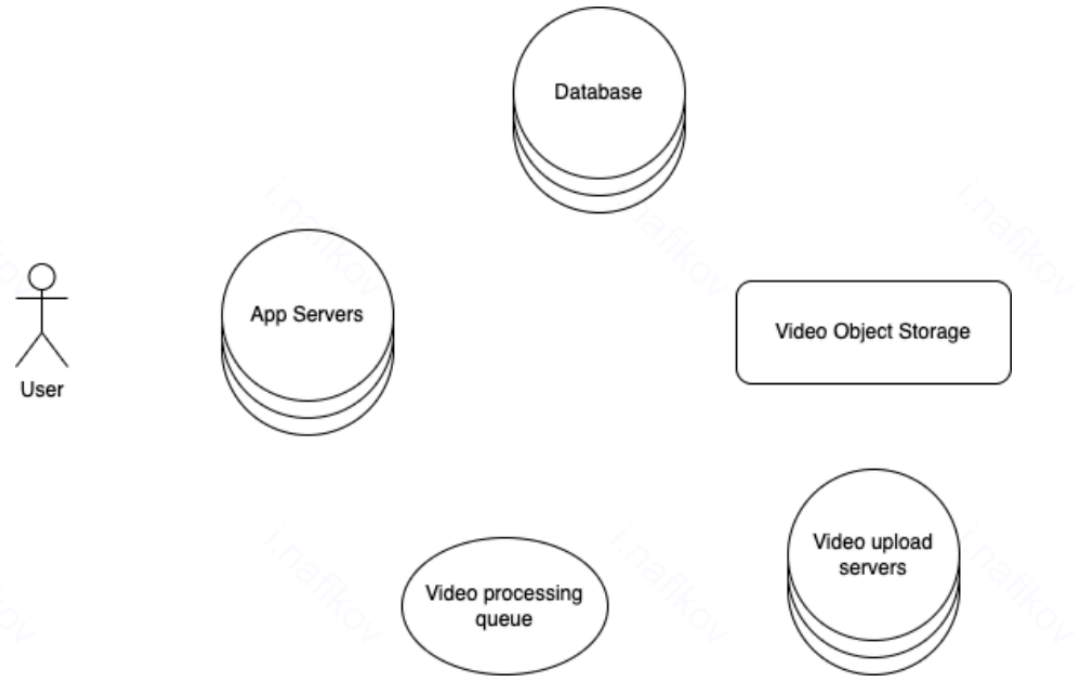
- user data
- user upload history

- 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

Task Description

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.

Implementation

Assumptions

- Application is provided only in a form of client-side mobile application, while "app servers" are considered as a back-end part of the system without need for providing front-end servers (with or without server-side rendering). Also, this means that Application server provides an API for interaction with mobile clients.
- I don't consider the mobile client application itself. Main focus for me is a back-end part of the system.
- By "private" videos I understand those which can be shared with others by specifying their accounts. After sharing they become accessible for these people in the owner's account profile. Without sharing they are "hidden".
- By "hidden (not shown in search and on user page)" videos I understand those which are "private" but not shared with anyone.
- App servers are necessary for managing videos, comments, and account's data, while upload servers are used for uploading videos and video processing queue is for downloading videos.

Entry points

ID	Entry Point	Description	Trust levels
1	**HTTPs port of App Servers**	The API will be only be accessible via TLS. API within the application is layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with

ID	Entry Point	Description	Trust levels
			Invalid Login Credentials
1.1	API Endpoints of App Severs	API for mobile applications or third-party services. All methods of interaction with our application are layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video
1.1.1	Authentication	This method accepts user supplied credentials and compares them with those in the database.	- User with Valid Login Credentials - User with Invalid Login Credentials
1.1.2	Search videos	This methods is for getting a list of videos (ID, name, preview) by different parameters including sorting parameters and paging by text query and page number.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials
1.1.3	Get user's account details	This method is for getting user's account details such as nickname, list of videos, account's creation date and a channel's description by the account ID.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials
1.1.4	Get your view history	This method is for getting your own view history with paging by user ID	- User with Valid Login Credentials
1.1.5	Manage a video	This method is to make a selected (by ID) video public, hidden, private or to delete this video. Before accepting a request to manage a video it checks if a user is authenticated and authorised (user is a owner of the video).	- Owner of the video

ID	Entry Point	Description	Trust levels
1.1.6	Request for uploading a video	The method is to request to upload a video for an owner's account. It check if a user a user is authenticated. If a check was successful it provides a user a URL of an appropriate video upload server and a secret session ID.	- User with Valid Login Credentials
1.1.7	Request for getting a video	This method is to request to receive a video stream by its ID. It checks if a user is authorised (user is a owner of the video or has an access to the video). If a check was successful it provides a user a URL of an appropriate video processing queue and a secret session ID.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
1.1.8	Get comments on a public video	This method is for displaying list of comments on a video with paging by the video ID and a page number.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
1.1.9	Get comments on a private video	This method is for displaying list of comments on a video with paging by the video ID and a page number. Before sending a list of comments it checks if a user is authenticated and authorised (user is a owner of the video or has an access to the video).	- Owner of the video - User with an access to the video
1.1.10	Get comments on a hidden video	This method is for displaying list of comments on a video with paging by the video ID and a page number. Before sending a list of comments it checks if a user is authenticated	- Channel's owner

ID	Entry Point	Description	Trust levels
		and authorised (user is a owner of the videor).	
1.1.11	Post a comment on a public video	This method is for posting a text comment on a video by the video ID.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
1.1.12	Post a comment on a private video	This method is for posting a text comment on a video by the video ID. Before uploading a comment it checks if a user is authenticated and authorised (user is a owner of the video or has an access to the video).	- Owner of the video - User with an access to the video
1.1.13	Post a comment on a hidden video	This method is for posting a text comment on a video by the video ID. Before uploading a comment it checks if a user is authenticated and authorised (user is a owner of the video).	- Owner of the video
2	**TLS port of Video Upload Servers**	The API will be only be accessible via TLS. API within the uploading videos is layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials
2.1	API Endpoints of Video Upload Severs	API for mobile applications or third-party services. All methods of interaction with upload servers are layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials
2.1.1	Upload a video	The method is to upload a video for an upload server by a secret session ID which	- Owner of the video

ID	Entry Point	Description	Trust levels
		was given by an App server.	
3	**TLS Port of Video Processing Queue**	The API will be only be accessible via TLS. API within the downloading videos is layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
3.1	API Endpoints of Video Processing Queue	API for mobile applications or third-party services. All methods of interaction with processing queues are layered on this entry point.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
3.1.1	Get a public video	The method is to receive a video steam from a video processing queue by a secret session ID which was given by an App server.	- Anonymous User - User with Valid Login Credentials - User with Invalid Login Credentials - Owner of the video - User with an access to the video
3.1.2	Get a private video	The method is to receive a video steam from a video processing queue by a secret session ID which was given by an App server.	- Owner of the video - User with an

ID	Entry Point	Description	Trust levels
			access to the video
3.1.3	Get a hidden video	The method is to receive a video steam from a video processing queue by a secret session ID which was given by an App server.	- Owner of the video

Assets

ID	Name	Description	Trust Levels
1	**Users**	**Assets relating to users' accounts**	
1.1	User Login Credentials	The login credentials that a user will use to log into the account.	- User with Valid Login Credentials - Database Server Administrator - Database Read User - Database Read/Write User - App server User Process - App server Administrator
1.2	Personal Data	The video hosting will store personal information relating to the users such as: - view history - list of hidden or private videos - comments on a private videos	- User with Valid Login Credentials - Database Server Administrator - Database Read User - Database Read/Write User - App server User Process - App server Administrator
2	**Videos**	**Assets relating to video streams**	
	Private or hidden videos	Video streams that are not available for every user	- Owner of the video - User with an access to the

ID	Name	Description	Trust Levels
			video - Video Object Storage Administrator - Video Object Read User - Video Object Read/Write User - Upload Server User Process - Video Processing Queue User Process
3	**App Servers**	**Assets relating to the underlying system of the App Servers**	
3.1	Availability of App Servers	The App Severs should be available 24 hours a day	- App Server Administrator
3.2	Ability to Execute Code as an App Server User	This is the ability to execute source code on the server as a web service user.	- App Server Administrator - App Server User Process
3.3	Ability to Collect Logs of an App Server	Collecting logs is important to monitor system's behaviour. They can give a lot of information of what is happening inside the system and what types of requests are done.	- App Server Administrator - App Server User Process
3.4	Access to an App Server	Access to the App Servers allows you to administer this server, giving you full access to the machine: memory, CPU, disk.	- App Server Administrator
4	**Video Upload Servers**	**Assets relating to the underlying system of the Video Upload Servers**	
4.1	Availability of Video Upload Servers	The Video Upload Severs should be available 24 hours a day	- Video Upload Administrator
4.2	Ability to Execute Code as a Video Upload Server User	This is the ability to execute source code on the server as a web service user.	- Video Upload Administrator - Video Upload User Process
4.3	Ability to Collect Logs of an Video	Collecting logs is important to monitor system's behaviour. They can give a lot	- Video Upload Administrator

ID	Name	Description	Trust Levels
	Upload Servers	of information of what is happening inside the system and what types of requests are done.	- Video Upload User Process
4.4	Access to a Video Upload Server	Access to the Video Upload Servers allows you to administer this server, giving you full access to the machine: memory, CPU, disk.	- Video Upload Administrator
5	**Video Processing Queue**	**Assets relating to the underlying system of the Video Processing Queue**	
5.1	Availability of Video Processing Queue	The Video Processing Queue should be available 24 hours a day	- Video Processing Queue Administrator
5.2	Ability to Execute Code as a Video Processing Queue User	This is the ability to execute source code on the server as a web service user.	- Video Processing Queue Administrator - Video Processing Queue User Process
5.3	Ability to Collect Logs of an Video Processing Queue	Collecting logs is important to monitor system's behaviour. They can give a lot of information of what is happening inside the system and what types of requests are done.	- Video Processing Queue Administrator - Video Processing Queue User Process
5.4	Access to Video Processing Queue	Access to the Video Processing Queue allows you to administer this queue, giving you full access to the machine: memory, CPU, disk.	- Video Processing Queue Administrator
5.5	Ability to change video sending bitrate	It can be important to configure speed of video streaming for specific users depending on the network conditions, region regulations, and user's statistics either manually or automatically.	- Video Processing Queue Administrator - Video Processing Queue User Process

ID	Name	Description	Trust Levels
6	**Database**	**Assets relating to the underlying system of the Database**	
6.1	Availability of Database	The Database should be available 24 hours a day	- Database Server Administrator
6.2	Ability to Execute SQL as a Database Read User	This is the ability to execute SQL select queries on the database, and thus retrieve any information stored within the database.	- Database Server Administrator - Database Read User - Database Read/Write User
6.3	Ability to Execute SQL as a Database Read/Write User	This is the ability to execute SQL. Select, insert, and update queries on the database and thus have read and write access to any information stored within the database.	- Database Server Administrator - Database Read/Write User
6.4	Access to Database Server	Access to the database server allows you to administer the database, giving you full access to the database users and all data contained within the database.	- Database Server Administrator
7	**Video Object Storage**	**Assets relating to the underlying system of the Video Data Object Storage**	
7.1	Availability of Video Object Storage	The Video Object Storage should be available 24 hours a day.	- Video Object Storage Server Administrator
7.2	Ability to Execute queries as a Video Object Storage Read User	This is the ability to execute queries on the storage, and thus retrieve any information stored within the storage.	- Video Object Storage Server Administrator - Video Object Storage Read User - Video Object Storage Read/Write User
7.3	Ability to Execute queries as a Video Object Storage Read/Write User	This is the ability to execute queries, and thus have read and write access to any information stored within the database.	- Video Object Storage Server Administrator - Video Object

ID	Name	Description	Trust Levels
			Storage Read/Write User
7.4	Access to Video Object Storage	Access to the video object storage server allows you to administer the database, giving you full access to the video object storage users and all data contained within the video object storage.	- Video Object Storage Server Administrator
8	**Mobile Application**	**Assets relating to the Video-Hosting Mobile Application**	
8.1	Login Session	This is the login session of a user to the video-hosting.	- User with Valid Login Credentials
8.2	Video Upload Session	This is the session of a user that is uploading a video on a Video Upload Server. The session is established on an App Server after checking the user is authenticated.	- Owner of the video
8.3	Video Download Session	This is the session of a user that is downloading a video from a Video Processing Queue. The session is established on an App Server after checking the user is authenticated and authorised.	- User with Valid Login Credentials - Owner of the video - User with an access to the video

Trust levels

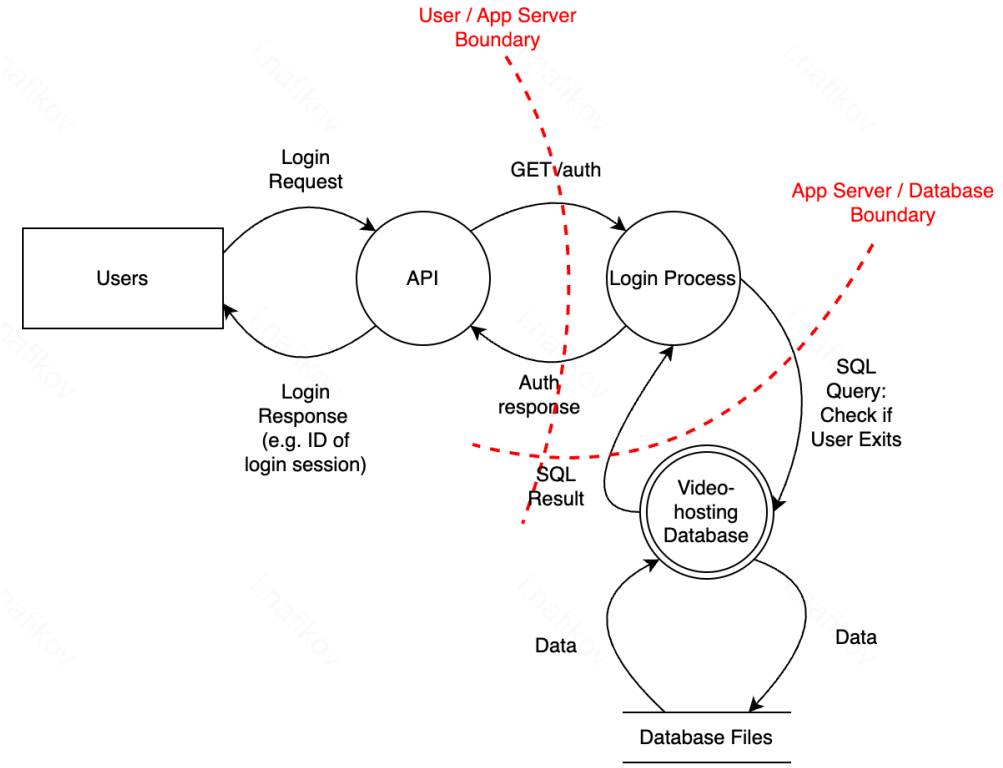
ID	Name	Description
1	Anonymous User	A user who has connected to the video-hosting server but has not provided valid credentials.
2	User with Invalid Login Credentials	A user who has connected to the video-hosting server and has logged in using valid login credentials.
3	User with Valid Login Credentials	A user who has connected to the video-hosting server is attempting to log in using invalid login credentials.
4	User with an access to the video	A user who has an access to some private video.
5	Owner of the video	A user who has uploaded a video which can be public, private, or hidden.

ID	Name	Description
6	App Server User Process	This is the process/user that an App Server executes code as and authenticates itself against the database server.
7	App Server Administrator	They can configure App Server machines, collect logs, and etc.
8	Video Upload User Process	This is the process/user that an Video Upload Server executes code as and authenticates itself against the Video Object Storage and an App Server.
9	Video Upload Administrator	They can configure Video Upload Server machines, collect logs, and etc.
10	Video Processing Queue User Process	This is the process/user that an Video Upload Server executes code as and authenticates itself against the Video Object Storage and an App Server.
11	Video Processing Queue Administrator	They can configure Video Processing Queue machines, collect logs, change bitrate, and etc.
12	Database Read User	The Database user account used to access the database for read access.
13	Database Read/Write User	The Database user account used to access the database for read and write access.
14	Database Server Administrator	The Database server administrator has read and write access to the database that is used by the video-hosting.
15	Video Object Storage Read User	The Video Object Storage user account used to access the database for read access.
16	Video Object Storage Read/Write User	The Video Object Storage user account used to access the database for read and write access.
17	Video Object Storage Server Administrator	The Video Object Storage server administrator has read and write access to the Video Object Storage that is used by the video-hosting.

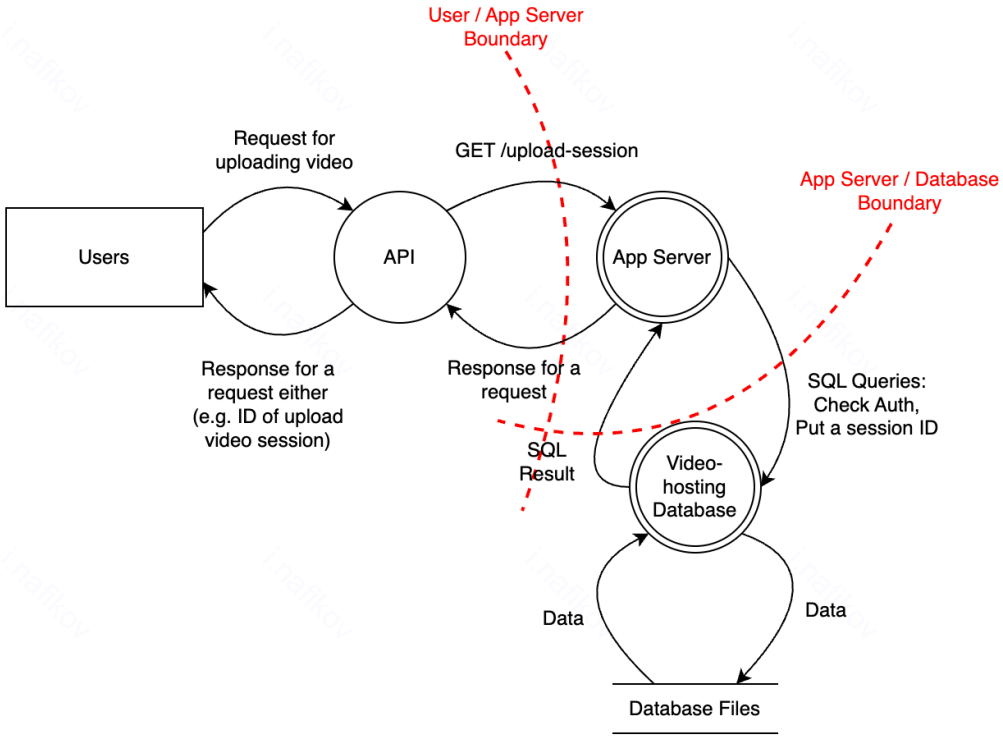
Data flows

I selected several data flows for the whole process of uploading a video which contains the 3 stages:

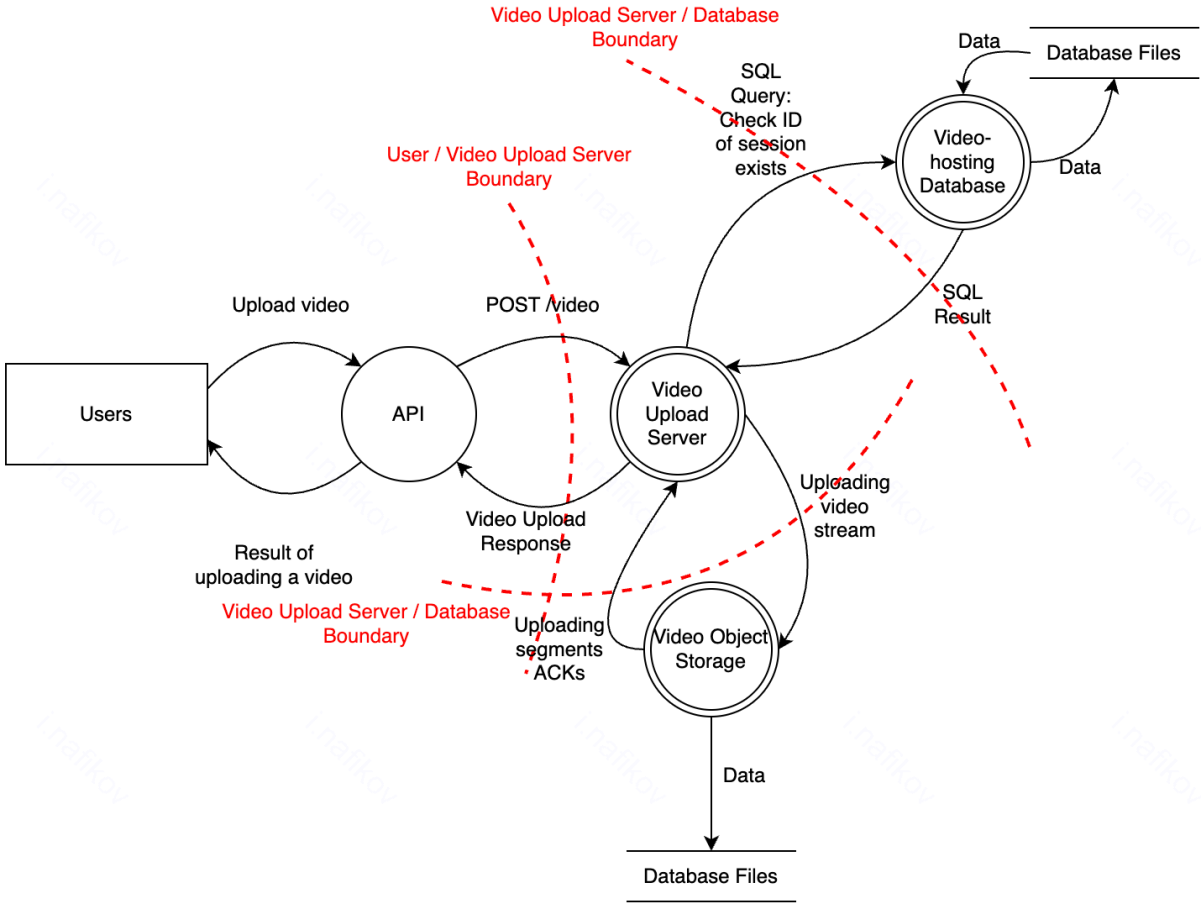
1. Log in



2. Request for uploading a video



3. Upload a the video



2. Determine threats

Task Description

Now when you have decomposed the system you can determine possible threats. Categorisations 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”.
- **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 realisation, 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 prioritisation, 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.

Implementation

Asset	Category	Threat	Vulnerability	Score	Countermeasure
Users	**========**	**========**	**========**	**===**	**========**
User Login Credentials	Information Disclosure	User credentials are exposed and obtained by an attacker.	Passwords are stored in plain text in the database.	4.2	Store passwords using strong hashing algorithms. Use HTTPS for all communications.
Personal Data	Information Disclosure	Personal data may be accessed by unauthorized users.	Lack of proper access controls or data encryption.	4.2	Implement access controls and data encryption for sensitive data.
Videos	**========**	**========**	**========**	**===**	**========**
Private or hidden videos	Information Disclosure	Hidden or private videos are exposed to unauthorized users.	Lack of proper access controls or video files encryption.	2.0	Implement access controls and video files encryption.
App Servers	**========**	**========**	**========**	**===**	**========**
Availability of App Servers	Denial of Service	App Servers become unavailable.	No protection from DoS or DDoS attacks	8.6	Implement rate limiting, IP whitelisting, and load balancing.

Asset	Category	Threat	Vulnerability	Score	Countermeasure
Ability to Execute Code as App Server User	Elevation of Privilege	An attacker executes arbitrary code through input manipulation.	Insufficient validation of input data leading to remote code execution exploits.	7.6	Use strict input validation to prevent RCE.
Ability to Collect Logs of an App Server	Information Disclosure	Logs containing sensitive information are accessed by unauthorised users.	Logs are stored in a location with inadequate access controls.	4.8	Implement strict logging policies and use access controls for log files.
Access to an App Server	Elevation of Privilege	Unauthorized users gain access to administrative functionalities of a App Server.	Weak access controls or hardcoded credentials.	7.2	Enforce strong password policies and multi-factor authentication for admin access.
Video Upload Servers	**========**	**========**	**========**	**===**	**========**
Availability of Video Upload Servers	Denial of Service	Video Upload Servers become unavailable.	No protection from DoS or DDoS attacks	6.0	Implement rate limiting, IP whitelisting, and load balancing.
Ability to Execute Code as Video Upload Server User	Elevation of Privilege	An attacker executes arbitrary code through input manipulation.	Missing input sanitization on file uploads.	5.3	Use strict video files validation to prevent RCE.
Ability to Collect Logs of Video Upload Servers	Information Disclosure	Logs containing sensitive information are accessed by unauthorised users.	Logs are stored in a location with inadequate access controls.	2.6	Implement strict logging policies and use access controls for log files.

Asset	Category	Threat	Vulnerability	Score	Countermeasure
Access to a Video Upload Server	Elevation of Privilege	Unauthorized users gain access to administrative functionalities of a Video Upload Server.	Weak access controls or hardcoded credentials.	6.0	Enforce strong password policies and multi-factor authentication for admin access.
Video Processing Queue	**========**	**========**	**========**	**===**	**========**
Availability of Video Processing Queue	Denial of Service	Video Processing Queue become unavailable.	No protection from DoS or DDoS attacks	6.0	Implement rate limiting, IP whitelisting, and load balancing.
Ability to Execute Code as Video Processing Queue User	Elevation of Privilege	An attacker executes arbitrary code.	Lack of proper access controls.	5.3	Implement access controls.
Ability to Collect Logs of Video Processing Queue	Information Disclosure	Logs containing sensitive information are accessed by unauthorised users.	Logs are stored in a location with inadequate access controls.	2.6	Implement strict logging policies and use access controls for log files.
Access to Video Processing Queue	Elevation of Privilege	Unauthorized users gain access to administrative functionalities of a Video Processing Queue.	Weak access controls or hardcoded credentials.	6.0	Enforce strong password policies and multi-factor authentication for admin access.
Ability to Change Video Sending Bitrate	Denial of Service	Changing bitrate leads to performance degradation for users decreasing UX.	Lack of network monitoring for video streaming to a user or lack	3.1	Introducing mechanisms for monitoring bitrate anomalies and queue changeover.

Asset	Category	Threat	Vulnerability	Score	Countermeasure
			of absence of queue changeover.		
Database	**========**	**========**	**========**	**===**	**========**
Availability of Database	Denial of Service	Database become unavailable.	No protection from DoS or DDoS attacks	8.6	Restrict network access from outside making it available only for our servers and introducing failover mechanisms.
Ability to Execute SQL as Database Read User	Information Disclosure	Unauthorized users can extract sensitive data from the database.	Weak credentials or lack of query restrictions.	4.4	Enforce strong password policies and multi-factor authentication for admin access and use parameterized queries.
Ability to Execute SQL as Database Read/Write User	Information Disclosure	Unauthorized users can extract sensitive data from the database.	Weak credentials or lack of query restrictions.	7.2	Enforce strong password policies and multi-factor authentication for admin access and use parameterized queries.
Access to Database Server	Elevation of Privilege	Unauthorized access to the database server allows data manipulation.	Poor network isolation practices or weak password policies.	8.0	Restrict network access from outside making it available only for our servers and enforce strong password policies and multi-factor authentication for admin access.
Video Object Storage	**========**	**========**	**========**	**===**	**========**

Asset	Category	Threat	Vulnerability	Score	Countermeasure
Availability of Video Object Storage	Denial of Service	Video Object Storage become unavailable.	No protection from DoS or DDoS attacks	6.0	Restrict network access from outside making it available only for our servers.
Ability to Execute queries as Video Object Storage Read User	Information Disclosure	Unauthorized extraction of video metadata or private/hidden video fragments.	Weak credentials or lack of query restrictions.	2.2	Enforce strong password policies and multi-factor authentication for admin access and use parameterized queries.
Ability to Execute queries as Video Object Storage Read/Write User	Tampering	Unauthorized modification of stored video data.	Weak credentials or lack of query restrictions.	6.2	Enforce strong password policies and multi-factor authentication for admin access and use parameterized queries.
Access to Video Object Storage	Elevation of Privilege	Unauthorized access to the database server allows data manipulation.	Poor network isolation practices or weak password policies.	6.6	Restrict network access from outside making it available only for our servers and enforce strong password policies and multi-factor authentication for admin access.
Mobile Application	**========**	**========**	**========**	**===**	**========**
Login Session	Spoofing	Session hijacking allows attackers to introduce themselves as another user and gain access to its hiddn videos,	Insecure session management practices.	7.1	Use secure cookies, implement session expiration, and monitor for unusual session activities.

Asset	Category	Threat	Vulnerability	Score	Countermeasure
		comments and view history.			
Video Upload Session	Information Disclosure	Unauthorized access to user's upload sessions resulting in accessing hidden videos.	Insecure session management practices.	5.4	Use secure cookies, implement session expiration, and monitor for unusual session activities.
Video Download Session	Information Disclosure	Unauthorized access to user's download sessions resulting in accessing hidden videos.	Insecure session management practices.	5.4	Use secure cookies, implement session expiration, and monitor for unusual session activities.