

LGE SW Security Specialist
CMU Studio Project
Final Report
(Team 5, 5verflow)

Date: 2 Jul. 2021

Contents

1. Introduction	3
2. Project plan	3
2.1. Roles & Responsibilities	3
2.2. Schedule	4
2.2.1 Phase 1 schedule	4
2.2.2 Phase 2 schedule	5
3. Phase 1: Secure Development	6
3.1. Requirements	6
3.1.1 Security Goal	6
3.1.2 Functional Requirements	6
3.1.2.1 Functional Requirements	7
3.1.2.2 Non-Functional Requirements	8
3.1.3 Asset Identification	9
3.1.4 Threat Modeling	12
3.1.4.1 STRIDE	12
3.1.4.2 PnG	13
3.1.5 Security Risk Assessment	18
3.1.6 Security Requirements / Mitigating Threats	19
3.2. Security Design	21
3.2.1 Overview	21
3.2.2 Architectural Drivers	22
3.2.3 Software Architecture	23
3.2.3.1 Static View	23
3.2.3.2 Server Software Block Diagram	23
3.2.3.3 Client Software Block Diagram	24
3.2.4 System Specification	26
3.3. Implementation	26
3.3.1 Input Validation	26
3.3.2 Secure Data Transmission	27
3.3.3 Secure Coding	27
3.3.4 Setup Guide	28
3.3.5 Verification (Static Code Analysis)	29
3.4. Testing	30
4. Phase 2: Security Evaluation of Classmate System	31
4.1. Analysis of Classmate System (Team 6)	31
4.1.1 Document Analysis	31
4.1.2 Implementation Analysis	34
4.2. Found Vulnerabilities	38
4.3. Attack Scenarios	40
4.4. Testing	42

5. Reflection	43
6. Conclusion	44
Appendix	44

1. Introduction

In this report, a project procedure for a secure face recognition system is presented, in which a server application performs face detection and recognition and then transmits the results to a client application through an Internet connection. This project is composed of two phases: a development and evaluation.

In phase 1, the system Secure Face ID is developed, as our objective is to build a secure product through a comprehensive consideration of a secure development process.

In order to build security into our product, we followed the procedure below:

- Training - We attended lectures from LGE (LG Electronics) and CMU (Carnegie Mellon University) and informed from fundamentals of practical tools
- Requirements and Design - First, we identified assets of the system, categorized them and defined security concerns. The next step was to use the data discovered during the asset profile creation to determine what threats may exist for our system. Their impacts on the CIA triad (confidentiality, integrity, and availability) are estimated and security requirements to solve specific security problems or to eliminate potential vulnerabilities are derived.
- Implementation and Verification - The system is implemented based on SEI CERT Coding Standards. Mitigations for high rated threats are applied, evaluated and analyzed.
- Release and Response - The software is released on [github](#) and a point of contact is provided to handle communication with customers.

In Phase 2, the system of classmates is evaluated. We analyze other systems with various perspectives to find vulnerabilities, so that we make a list of scenarios to attack. Possible attacks are performed to verify the security of the system, and then finalize the results. The result contains the differences with our system in design and implementation, and includes some improvements.

All of the artifacts produced from this project will be released on github link (<https://github.com/5verFlow/sfid-document>).

2. Project plan

This section is described based on the document "*Team 5 Project Planning*".

2.1. Roles & Responsibilities

We consist of 6 people nicknamed "5verflow". Team members have been working in various fields and had experiences. According to their experiences, the roles were assigned where members could do their best.

1. Project Description			
Project Name	Secure Face ID		
Project Schedule	2021-05-31 ~ 2021-07-02		
Update Date	2021-06-21		
Project Members	Name	Role (Phase 1)	Role (Phase 2)
	SeungWook Cha	Team Lead	Team Lead
	SungJun Lee	Requirement	Doc. Anal.
	DongHyuk Han	Infrastructure	Doc. Anal. & Test
	WooLam Kang	SW Development	Impl. Anal. & Test
	YooKyoung Choi	Requirement	Doc. Anal. & Test
	YoungJinn Lee	System Design	Impl. Anal. & Test

[Roles and Responsibilities for Team 5]

** Document : Team 5 Project Planning*

2.2. Schedule

2.2.1 Phase 1 schedule

We had planned to allocate more time to analyzing the requirements, and we'd also planned to implement the program from the early stage by separating roles. But the task of analyzing risks and mitigations required more resources than our estimation, so entire members had no way but concentrated on it. Thus it resulted in delaying the implementation schedule and decreased testing volume.

2. Project Schedule

Green Box : 1st Planning

Red Box : 2nd Planning

V : Activity

C : complete

Category	Item	Phase 1																Leading responsibility
		5/31	6/1	6/2	6/3	6/4	6/7	6/8	6/9	6/10	6/11	6/14	6/15	6/16	6/17	6/18		
Project Planning	Project Planning (Phase 1)	V	V	V	C												SW Cha	
	Project Planning (Phase 2)																SW Cha	
Functional Requirement	Function Requirement Analysis					V	V	C									YJ Lee	
	Review					V	V	C									YK Choi	
Security Requirement	Asset List						V	C									YK Choi	
	Threat Modeling					V	V	C									SJ Lee, YK Choi	
	Risk Assessment						V	C									SW Cha	
	Mitigation						V	V	C								SJ Lee	
	Review								V	C							All	
Development	Client - User interface							V	V	V	C						WL Kang	
	Client - Security (TLS)								V	V	C						DH Han	
	Server - Main Functionality							V	V	C							WL Kang	
	Server - Security (TLS)								V	V	C						DH Han	
	Server - Secure Storage											V	V	V	C		YK Choi	
	Open Source Vulnerability analysis																DH Han	
	Static Analysis											V	V	V	C		WL Kang	
Testing	Test Case										V	V	C				YK Choi	
	Function Test												V	V	C		YJ Lee	
	Security Test													V	C		YJ Lee	
Documents	Developer Guide											V	V	V	C		SJ Lee	
	Presentation											V	V	V	C		YJ Lee	

[Phase 1 Schedule]

* Document : Team 5 Project Planning

2.2.2 Phase 2 schedule

We started by analyzing artifacts of team 6. We listed up what they focused on, how they mitigated it. After that we tried to look for any missing threats and any lack of mitigations and then we made several attack scenarios for that and conducted attacks depending on scenarios. Some attack scenarios like brute force took a lot of time but it's not impact our schedule because we carried out attacks in parallel by each dedicated person.

2. Project Schedule

Green Box : 1st Planning

Red Box : 2nd Planning

V : Activity

C : complete

Category	Item	Phase 2										Leading responsibility
		6/21	6/22	6/23	6/24	6/25	6/28	6/29	6/30	7/1	7/2	
Analysis	Requirement Analysis of Team6	V	V	V	V	C						SJ Lee
	Implementation Analysis of Team6	V	V	V	V	C						WL Kang
	Test Method Analysis	V	V	V	V	C						DH Han
	Attack Scenario Listing		V	V	V	V	C					All
Testing	Given Test Case			V	V	C						YK Choi
	Attack Scenario Attempt			V	V	V	V	V	C			All
	Penetration Test					V	V	V	C			DH Han
Results	Test Results Analysis					V	V	V	C			YJ Lee
	Secure Recommendations						V	V	C			YJ Lee
Documents	Final Report							V	V	C		SW Cha
	Presentation							V	V	C		WL Kang

[Phase2 Schedule]

* Document : Team 5 Project Planning

3. Phase 1: Secure Development

3.1. Requirements

At the beginning of phase 1, we defined security goals first and functional requirements. Based on them, risks are analyzed and finally security requirements are derived.

3.1.1 Security Goal

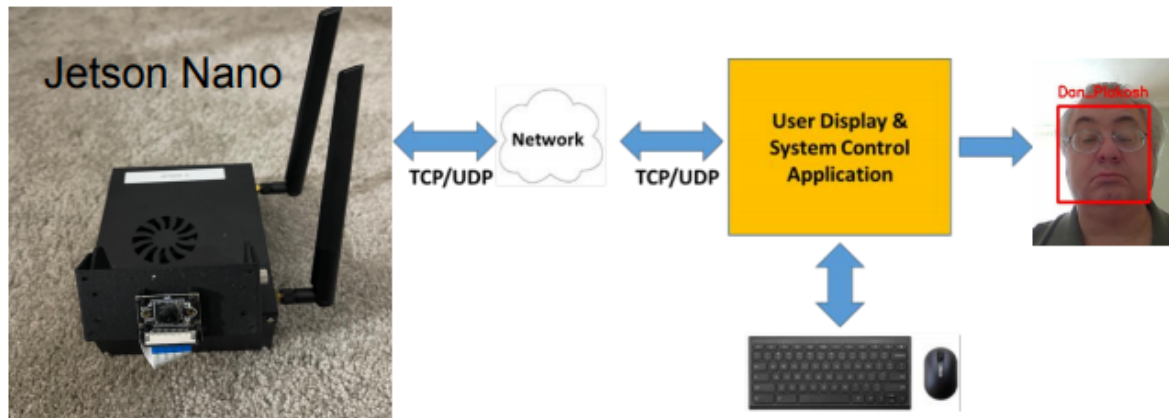
In order to derive the security goal in Secure Face ID, a thorough examination on the contexts of face recognition and video conferencing systems are performed. As a result, we derived main goals to achieve CIA triad as the followings:

- Confidentiality - the system assures personal information of the users is kept secret so that unauthorized users are prevented from obtaining access.
- Integrity - the program and the data transferred in the system can be trusted, by ensuring they are neither tampered nor repudiable.
- Availability - the system assured reliable access to resources when authorized users need them.

3.1.2 Software Requirements

Based on Professor Daniel's project description, we define functional and non-functional

requirements for the Team 5 Secure Face ID project. The document “*Software Requirement Specification*” described in detail.



[Overall system of the project]

3.1.2.1 Functional Requirements

Group	ID	Function	Use Scenario	ReqID
User Menu	FR1	The client app shall have a user selection menu. <ul style="list-style-type: none"> - SecureMode - InsecureMode - Live Mode - Playback Mode - Register Mode 	Display a user selection menu by requesting command line	Req9
Communication Mode	FR2	The client app shall be able to change communication mode with the server app to secure mode.	When a user selects the ' Secure Mode ' menu, the client app connects to the server app with secure communication protocol. *Describe communication protocol details in security requirements	Req8 Req10
Communication Mode	FR3	The client app shall be able to change communication mode with the server app to insecure mode.	When a user selects the ' Insecure Mode ' menu, the client app connects to the server app with insecure communication protocol. *Describe communication protocol details in security requirements	Req10
Operation Mode	FR4	The client app shall be able to add new user images to the image database with a user-specified name.	When a user selects the ' Register Mode ' menu, the client app shows the input command line to get an user name. If the user inputs a name, the client app displays a camera stream and then requests the	Req11

			server app to add the user's images to the image database.	
Operation Mode	FR5	The client app shall be able to display camera video stream and face recognition results from the server app.	When a user selects the ' Live Mode ' menu, the client app displays camera video stream and face recognition results received from the server app.	Req12
Operation Mode	FR6	The client app shall be able to receive video file streams and face recognition results from the server app with a user-specified filename.	When a user selects the ' Playback Mode ' menu, the client app shows the input command line to get a video file name. If the user inputs a file name, The client app displays a video file stream and face recognition results received from the server app.	Req13
Operation Mode	FR7	The client app shall be able to detect fault/error and then recover and report.	Fault/Error List Up (security) Recovery Strategy Report Format	Req17

* Document : "Software Requirement Specification"

3.1.2.2 Non-Functional Requirements

ID	Requirement	Concrete Scenario	ReqID
NFR1	Ensure application architecture is secure.	The software development should follow LG-SDL (LG Secure Development Lifecycle)	Req4
NFR2	Ensure code is written and implemented in a secure manner	The application implementation should follow LGE secure coding standard.	Req5
NFR3	Ensure application network communication is secure	The application should use TLS to provide privacy and data security for communications over the Internet	Req6
NFR4	Practice finding security flaws in code / applications both statically and dynamically	static/dynamic analysis	Req7
NFR5	Ensuring that all software in both applications are architected and coded to be secure and free of vulnerabilities	refer NFR1, NFR2, NFR3	Req15
NFR6	Analyzing the provided initial implementation for vulnerabilities and developing solutions to mitigate.	DFD, Threat Modeling, OWASP	Req18

* Document name : "Software Requirement Specification"

3.1.3 Asset Identification

Assets for this project are shown in the table below. The total number of assets is 16, and gray-colored 2 assets are excluded because we do not have control over them. One of the excluded assets is AS-004, which, as a result of the discussion, we decided to exclude routers without focusing on additional options. And the other is AS-011, an artifact of facial recognition that has been excluded by assuming FaceNet is reliable.

We started modeling threads with DFD (Data Flow Diagram) based on the results of this step.

ID	Type	Assets	Description	Security Characteristics
AS-001	External Entity	User	<p><Definition> A person who uses the Jetson Nano system through a client app.</p> <p><Damage Scenario> 1. An attacker assumes a user's password and uses it maliciously.</p> <p>2. An attacker spoofs a user's password and then threatens to publish the victim's data or perpetually block access to it unless a ransom is paid.</p> <p>3. An attacker can input invalid strings maliciously.</p> <p>4. An attacker can resolve the name of attendees by using an image database.</p>	Confidentiality Integrity Availability Authentication Authorization Non-repudiation Information-Disclosure
AS-002	Hardware	NVIDIA Jetson Nano	<p><Definition> Face recognition system using TensorRT.</p> <p><Damage Scenario> An attacker can break or steal the device.</p>	Availability
AS-003	Hardware	Raspberry Pi v2 camera	<p><Definition> Input source device of Jetson Nano.</p> <p><Damage Scenario> An attacker can break or steal the device.</p>	Availability
AS-004	Hardware	TP-Link AC-1750 Mesh Wi-Fi Router	<p><Definition> Router for communication between Jetson Nano and laptop.</p> <p><Damage Scenario> 1. An attacker can break or steal the device.</p> <p>2. An attacker can decrease availability by using DoS attacks.</p> <p><i>* This asset is only for practice purposes and it's not included in DFD.</i></p>	Availability
AS-	Hardware	Laptop	<Definition>	

005			<p>Laptop device to develop and run the client application.</p> <p><Damage Scenario> An attacker can rob the authenticated laptop and connect to the server with it.</p>	
AS-006	Application	The Client Application	<p><Definition> An application that communicates with the Jetson Nano, The server application, and sends a request to the Jetson Nano when the user enters a command, such as a communication mode(Secure Mode, Non Secure Mode) or mode of operation (Learning Mode, Run Mode, Test Run Mode). Display image frames and any accompanying amplifying analysis information received from the camera and image analysis application in the format specified.</p> <p><Damage Scenario> 1. An attacker can tamper with the attendee name by intercepting the registering request when learning mode.</p> <p>2. An attacker can insert malicious code into the name buffer by intercepting the registering request when learning mode or test run mode.</p> <p>3. An attacker can play injurious video files and then break the system.</p>	<p>Confidentiality Integrity Availability Authentication</p>
AS-007	Application	The Server Application	<p><Definition> An application that communicates with the Laptop, the client application and sends a response of the client app's requests.</p> <p><Damage Scenario> An attacker can tamper with the camera video stream so as to spoof the user's learning data.</p>	<p>Integrity</p>
AS-008	Data	Communication Protocol	<p><Definition> Secure Mode : Non Secure Mode :</p> <p><Damage Scenario> If the system uses an old version of protocol, an attacker knows the protocol vulnerabilities and exploits them easily.</p>	
AS-009	Data	Certificates	<p><Definition> The certificates to establish secure, authenticated communication with cameras and image analysis applications and user interfaces.</p>	<p>Confidentiality Authentication Non-Repudiation</p>

			<p><Damage Scenario> If the certificates are stored in insecure storage, an attacker can access that and then delete, modify or expose them.</p>	
AS-010	Data	Image file	<p><Definition> Image file of people in the imgs folder and made by user specified filename.</p> <p><Damage Scenario> An attacker can know the user's name and face id if the filename is set to plain text and then can maliciously use personal privacy.</p>	Confidentiality Integrity
AS-011	Data	Artifact of Face recognition	<p><Definition> Artifacts learned and built with AS-010. <i>* This asset is only for practice purposes and it's not included in DFD.</i></p>	Integrity Availability Authorization
AS-012	Data	Video file	<p><Definition> Video file when using face recognition system instead of camera input source.</p> <p><Damage Scenario> If an attacker tampered with a video file, it can cause crack of the system or the client application.</p>	Integrity Availability Authorization
AS-013	Data	Error/Fault Log	<p><Definition> Log of system fault or error.</p> <p><Damage Scenario> 1. An attacker can know the system vulnerabilities and exploit the vulnerability easily.</p>	Confidentiality
AS-014	Interface	Camera <-> The Server Application	<p><Definition> Includes both hardware and software interfaces.</p>	Integrity
AS-015	Networking Interface	The Server Application <-> The Client Application	<p><Definition> Network interface between the server application and the client application. The connection protocol can be changed by a user selection.</p>	Confidentiality Integrity Authentication Non-Repudiation
AS-016	Interface	The Client Application <-> The User	<p><Definition> User interface with the client application. An user can select the UI button and input string.</p>	Integrity Authentication Authorization

* document : "Project Asset List"

3.1.4 Threat Modeling

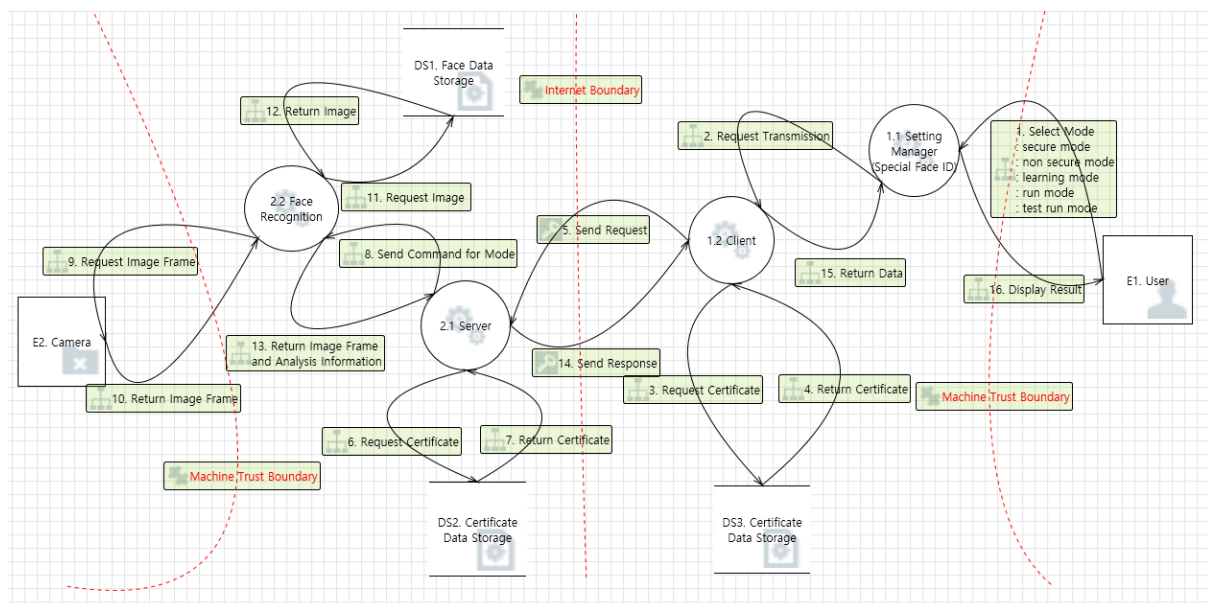
Team 5 has used Microsoft Threat Modeling Tool and adopted the STRIDE model and PnG method to identify the threats.

3.1.4.1 STRIDE

We have learned Microsoft Security Development Lifecycle for security development in this course and tried to follow the procedure. We know STRIDE threat modeling with the Microsoft Threat Modeling Tool is implemented at Microsoft and widely adopted.

So we chose the STRIDE method with assets already identified. The threat modeling file “team5_DFD.tm7” shows in detail.

- JetsonNano : server application
- User displays & system control application
- Data : video frame, meta data, picture, certificates
- Network interface
- Hardware : camera



[Team 5's Data Flow Diagram]

* document : “Security Requirements”

We mainly focused on the boundary area of DFD and extracted the following risk items, especially data exchanges between entities.

- User Interface - Invalid data could cause buffer overflow or connecting to unauthorized system
- Network Communication - Data sniffing
- Data Consistency - User data corruption

D	E	F	G	H
Diagram	Interaction	Priority	State	Description
Diagram 1	7. Return Certificate	High	Not Started	Improper data protection of "DS2. Certificate Data Storage" can allow an attacker to read information not intended for disclosure. Review authorization settings.
Diagram 1	8. Send Command for Mode	High	Not Started	"2.2 Face Recognition" may be able to impersonate the context of "2.1 Server" in order to gain additional privilege.
Diagram 1	13. Return Image Frame and Analysis Information	High	Not Started	"2.1 Server" may be able to impersonate the context of "2.2 Face Recognition" in order to gain additional privilege.
Diagram 1	11. Request Image	High	Not Started	"DS1. Face Data Storage" may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of "DS1. Face Data Storage" . Consider using a standard authentication mechanism to identify the destination data store.
Diagram 1	11. Request Image	High	Not Started	Does "2.2 Face Recognition" or "DS1. Face Data Storage" take explicit steps to control resource consumption? Resource consumption attacks can be hard to deal with, and there are times that it makes sense to let the OS do the job. Be careful that your resource requests don't deadlock, and that they do timeout.
Diagram 1	12. Return Image	High	Not Started	"DS1. Face Data Storage" may be spoofed by an attacker and this may lead to incorrect data delivered to "2.2 Face Recognition" . Consider using a standard authentication mechanism to identify the source data store.
Diagram 1	12. Return Image	High	Not Started	Improper data protection of "DS1. Face Data Storage" can allow an attacker to read information not intended for disclosure. Review authorization settings.
Diagram 1	10. Return Image Frame	High	Not Started	"E2. Camera" may be spoofed by an attacker and this may lead to unauthorized access to "2.2 Face Recognition" . Consider using a standard authentication mechanism to identify the external entity.
Diagram 1	10. Return Image Frame	High	Not Started	"2.2 Face Recognition" may be able to impersonate the context of "E2. Camera" in order to gain additional privilege.

[Threats]




* document : "Security Requirements"

3.1.4.2 PnG

1. Identify the PnG types

In addition, PnG (Persona non Grata) method is also utilized to identify security threats, as team members are novice to threat modeling and PnG method is expected to give consistent results regardless of their professionalism. The following table identifies the PnG types.

No.	Description
-----	-------------

1	<div data-bbox="355 241 542 477">  </div> <div data-bbox="566 239 1037 472"> <p>Name: Peter</p> <p>Age: 25</p> <p>Job: Student</p> <p>Interest: Software Hacking, Online Chat</p> </div> <p>Description: Peter is a student hacker pursuing a reputation. He is not interested in monetary assets, just wants to be notorious by taking down prestigious universities or companies. He spends most of his days on the computer.</p>
2	<div data-bbox="355 678 542 904">  </div> <div data-bbox="566 676 850 907"> <p>Name: JooYoung</p> <p>Age: 37</p> <p>Job: Investor</p> <p>Interest: Stock, Cooking</p> </div> <p>Description: JooYoung works for an investment company. The company office is located in a shared office space in Gangnam. She enjoys spending her time with her child and baking some scones. She majored in mathematics, and is very good with calculation.</p>
3	<div data-bbox="355 1151 555 1377">  </div> <div data-bbox="566 1146 1048 1377"> <p>Name: Dave</p> <p>Age: 55</p> <p>Job: Team Leader of Tartan Incorporate</p> <p>Interest: Real Estate</p> </div> <p>Description: Dave is a 55-year-old employer of Tartan Incorporate. He is a leader of another team and is also gathered at the same office with the team '5verflow'. He works on a development project similar to that of '5verflow' but is reluctant to work diligently. In addition, he is hostile to the leader of '5verflow' and wants to spoil his project.</p>

2. Identify PnG


The following table summarizes the identification of PnG.


Attacker	PnG Method	Descriptions
Peter	Motivations	<ul style="list-style-type: none"> • Increase recognition


	Goals	<ul style="list-style-type: none"> ● Intrude into Web server of Tartan Inc. where developers share their files (source code and documents) ● Break in the developers' laptops and get some confidential information such as source code
	Skills	<ul style="list-style-type: none"> ● Strong coding skills ● Extensive knowledge of hacking skills ● Utilize various hacking tools
	Misuse Cases	<ul style="list-style-type: none"> ● Make Distributed DoS attack on the web server to get notorious reputation ● Download confidential data (source code or document) from web server and upload them to public internet ● Upload malicious script on the web server of Tartan Inc. ● Hack the router located in the conference room and flood invalid network control messages (ARP, DNS, etc.) ● Steal data files from laptops and make a disclosure of developing procedures of Tartan Inc.
JooYoung	Motivations	<ul style="list-style-type: none"> ● Monetary gain
	Goals	<ul style="list-style-type: none"> ● Steal valuable assets: Jetson Nano, laptops, router ● Acquire some classified information
	Skills	<ul style="list-style-type: none"> ● Lacks technical skills to hack networking or computers. ● However, she is located at the same workspace with Team '5verflow', so she can easily notice when the conference room is empty. ● Also, she is acquainted with some brokers who buys and sells classified business information

	Misuse Cases	<ul style="list-style-type: none"> • Sneak into the conference room and steal properties (Jetson Nano, laptops, router, etc.) • Sneak in the conference room and install a hidden camera or microphone for eavesdropping
Dave	Motivations	<ul style="list-style-type: none"> • Hostility and laziness
	Goals	<ul style="list-style-type: none"> • Ruin the project of team '5verflow' • Get documents and source codes from '5verflow' and exploit them to his project
	Skills	<ul style="list-style-type: none"> • Lots of job experience in Tartan Inc. and know some habits of the team leader of '5verflow', such as frequently used password and source code locations. • Acquainted with the team leader of '5verflow' so he can enter into the conference room, steal a glance at the password of the router. • Intermediate level of coding and hacking
	Misuse Cases	<ul style="list-style-type: none"> • Sniff the data and information exchanged through the router using the password he stole. • Establish a ssh connection to Jetson Nano equipment, copy source code, and modify or delete data in the file system.

3. Indicate whether the PnG identification resulted in new threats (compared to STRIDE results)

Threats discovered by PnG		Mitigation	STRIDE appearance
 <p>Peter</p>	Make Distributed DoS attack on the web server to get notorious reputation	Backup important files in secure storage	Denial of Service

	Download confidential data (source code or document) from web server and upload them to public internet	Choose secure web server service with sufficient level of authentication	Information disclosure
	Upload malicious script on the web server of Tartan Inc.	Choose secure web server service that checks suspicious behavior	Tampering
	Hack the router located in the conference room and flood invalid network control messages (ARP, DNS, etc.)	Setup a high level of authentication method on router (WPA2 ~ 3). Periodically revise password	Spoofing, Denial of Service
	Steal data files from laptops and make a disclosure of developing procedures of Tartan Inc.	Setup firewall on the laptops	Information disclosure
 <p>JooYoung</p>	Sneak into the conference room and steal properties (Jetson Nano, laptops, router, etc.)	Be sure to lock up the conference room	<p>X</p> <p>The results from STRIDE does not cover the threats out of the systemic scope that may easily arise by attackers not skilled with software hacking</p>

	Sneak in the conference room and install a hidden camera or microphone for eavesdropping.	Be sure to lock up the conference room	Information disclosure-
 Dave	Sniff the data and information exchanged through the router using the password he stole.	Setup a high level of authentication method on router (WPA2 ~ 3). Periodically revise password. Cover up the information sheet of router	Spoofing
	Establish a ssh connection to Jetson Nano equipment, copy source codes and modify or delete data in the file system	Change the default configurations such as ssh connection id and password (e.g., root/root, lg/lg), and router settings (given password)	Spoofing, Tampering, Denial of Service

Using the PnG method, team 5 members have identified possible threats and compare the results from those using STRIDE method. Although all the team members are novice to threat modelling, it was able to get consistent results. STRIDE method is one of the good approaches toward systematic threat modeling, though it is hard to detect threats out of the systemic scope that may arise on software development cycle, even by the attackers who are not skilled with hackings. In addition, teammates have derived the mitigation strategies and applied those immediately to the team project.

3.1.5 Security Risk Assessment

To rate the threats, Team 5 has adopted OWASP Risk Rating Methodology, known as a means to easily and more accurately assess the likelihood and impact of a web application vulnerability. The document “*Security Requirements*” described in detail, especially on the sheet “*OWASP Risk assessment*”.

Likelihood Factors						Technical Impact Factors			
Vulnerability Factors				Overall Likelihood	Likelihood Level	Loss of Confidentiality	Loss of Integrity	Loss of Availability	L Acc
Ease of Discovery	Ease of Exploit	Awareness	Intrusion Detection						
9 ▾	9 ▾	6 ▾	3 ▾	6.13	High	2 ▾	1 ▾	9 ▾	
9 ▾	9 ▾	6 ▾	3 ▾	6.125	High	2 ▾	1 ▾	9 ▾	
9 ▾	9 ▾	6 ▾	3 ▾	6.125	High	2 ▾	1 ▾	9 ▾	
9 ▾	9 ▾	6 ▾	3 ▾	6.125	High	2 ▾	9 ▾	9 ▾	
9 ▾	9 ▾	6 ▾	3 ▾	6.125	High	2 ▾	9 ▾	9 ▾	
9 ▾	9 ▾	6 ▾	3 ▾	6.125	High	2 ▾	9 ▾	9 ▾	

[Risk Rating using OWASP]

* document : "Security Requirements"

Team 5 selected 19 threats from the OWASP risk assessment and grouped into the following 5 categories to define risk.

- Input Validation
- Secure Data Transmission
- Authentication
- Secure Data Store
- Logging

3.1.6 Security Requirements / Mitigating Threats

To fill out the security requirements, we used the quality attribute scenarios and defined 6 parts to characterize quality attributes. (Stimulus, Source of the stimulus, Environment, Artifact stimulated, Response, Response measure)

Quality Attribute Requirement Scenario	
	Desc.
Stimulus	Write invalid form of IP address (ex. 123.456.789)
Source	User input for IP address
Environment	Before connecting to JetsonNano server
Artifacts	Configuration data
Response	Check whether the input IP address is on the valid range
Response Measure	100 percent of detecting invalid IP address
	Desc.
Stimulus	Sniffing data on network between JetsonNano and user laptop
Source	Attacker connected on the same network
Environment	Secure mode operation with connection
Artifacts	Data on transmission
Response	Encrypting data during transmission
Response Measure	100 percent of transmitted data is encrypted
	Desc.
Stimulus	connection from unknown client
Source	unidentified user
Environment	Server is listening to connection request
Artifacts	Server system
Response	authenticated with 2 factor method
Response Measure	always deny for authentication failed

[Team 5's Quality Attribute Requirement Scenario]

* document : : "Security Requirements"

Category	Security Requirements ID	Security Requirements	TID	
Input Validation for Client Application	SR 1-1	Client Application must check if the format of input IP address is in valid format	170	attacker can input to extreme might causes attack might to simply leads to SERVICE
	SR 1-2	Server and Client should check respectively whether the input for Username field on the Register mode is valid as a filename.	170	An attacker can overflow using as the input of using special
	SR 1-3	Client should check if the input of the Port field is within the valid port number range.	170	An attacker can number or string Port field and overflow.
	SR 1-4	Server and client should check input validation respectively whether the input for video file name field on the Playback mode has video file format such as .mp4.	170	An attacker can overflow using as the input, filename which is to execute or save file.
	SR 1-5	Client should check whether the image received from server is format of jpeg before displaying it.	N/A	An attacker can transmitted from jpeg format can
	SR 1-6	Client should compare the number of detected face and the number of its information, which are received from server, and they should be same.	N/A	By tempering number of detected face matched to the information.
Secure Data Transmission	SR 2-1	After connection establishment all the data transferred between server and client must be securely encrypted	N/A	INFORMATION data over the
	SR 2-2	Must check integrity of all the transmitted data between server and client	N/A	TAMPERING network
Secure Authentication	SR 3-1	Server and Client must mutually authenticate each other with X.509 certificates	98, 105	Server or client an attacker (if unauthorized)

[Team 5's Security Requirement & Mitigation]

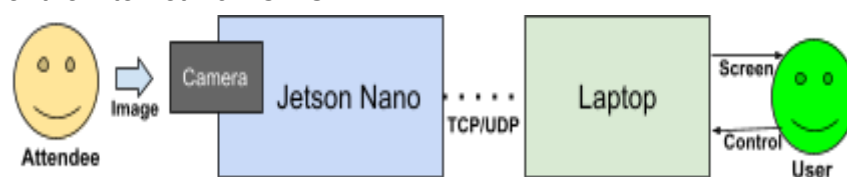
* document : "Security Requirements"

3.2. Security Design

Based on the security requirement we decided, security design was performed for each server and client side. Connections between them are also considered. The document "Software Architecture Design" described in detail.

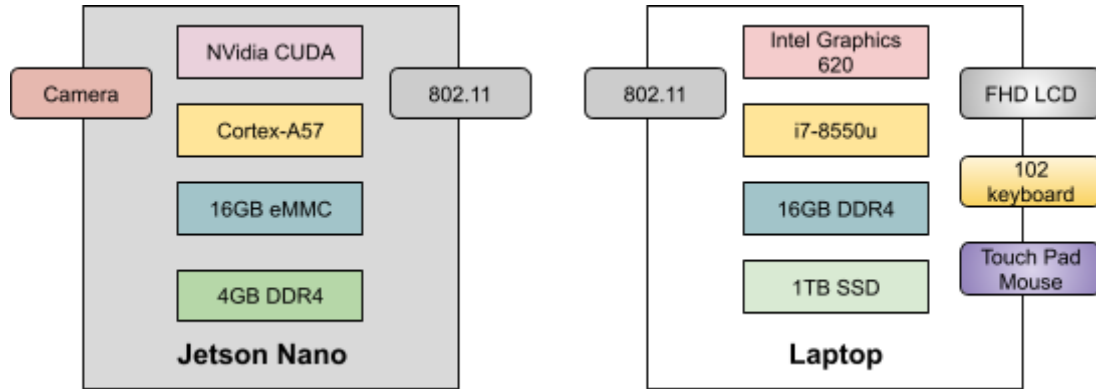
3.2.1 Overview

The Secure Face ID consists of Jetson Nano server and Laptop client and they are connected over the internet via TCP/UDP.



[Context Overview]

The camera captures video conference attendees and the server analyzes it using MTCNN and FaceNet. The CUDA GPU is utilized during image processing. The image and data are transferred to the client and the user sees the image and data at the client laptop.



[System Overview]

3.2.2 Architectural Drivers

- SW Main Features

This is a summary of functional requirements.

ID	Level-1
MF-1	Application Login
MF-2	Live Mode (Run Mode)
MF-3	Playback Mode (Test Mode)
MF-4	Register Mode (Learning Mode)
MF-5	Error Report

- Quality Attributes

This system shall meet quality attributes and security has top priority.

ID	Quality Attribute	Scenario	Priority
QA-01	Performance	The system must show video as close to real time as possible.	3
QA-02	Security	The system must be secured from any threats as specified.	1
QA-03	Reliability	The system should provide tolerance of network failure thus it could transfer images continuously.	2

- Constraints

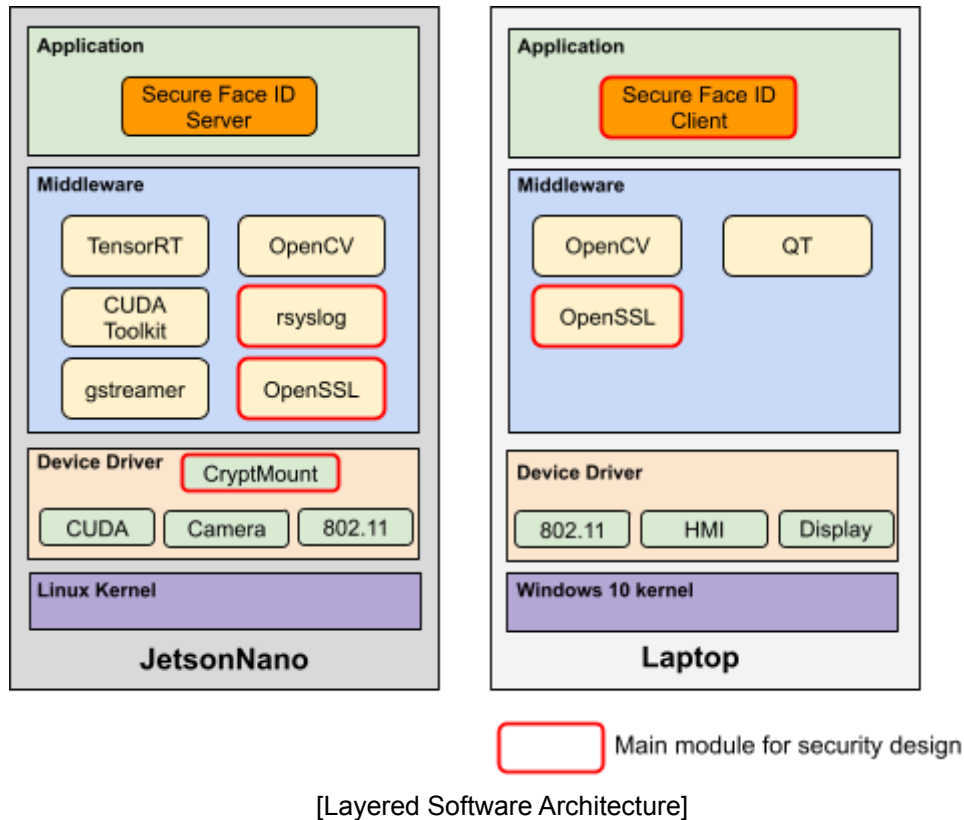
ID	Description
----	-------------

TCN-01	The server allows only one client connection.
--------	---

3.2.3 Software Architecture

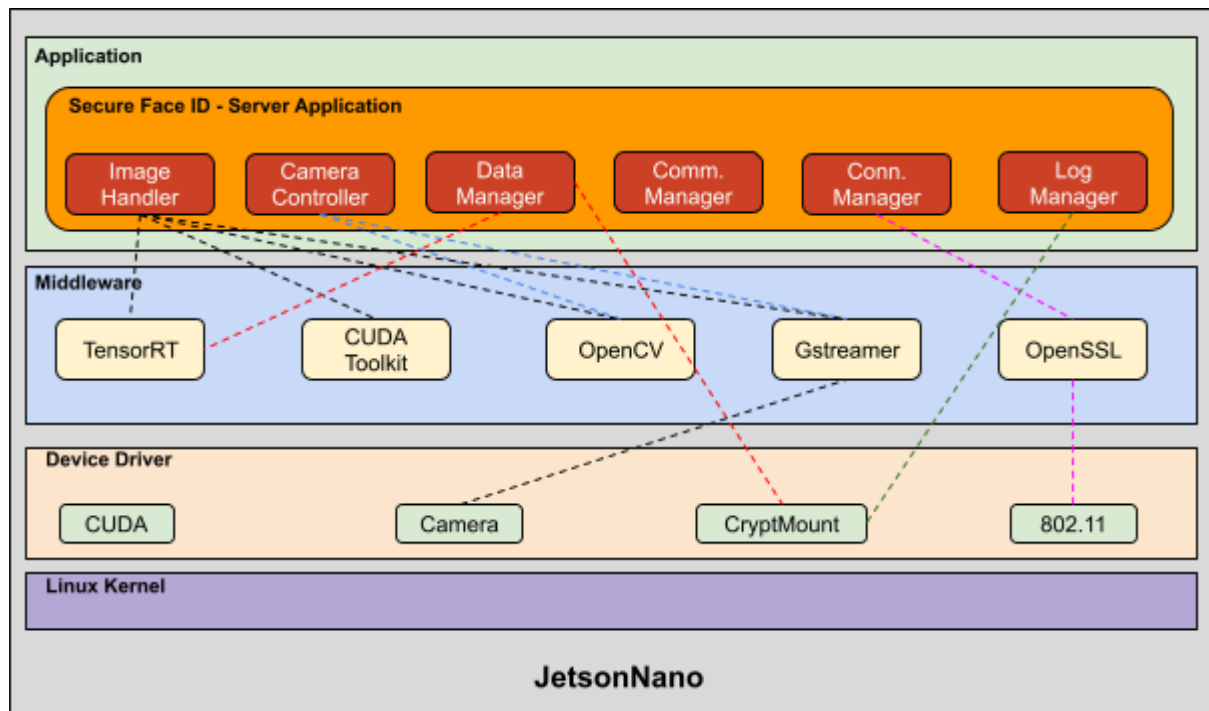
3.2.3.1 Static View

The diagram below shows static layered blocks.



3.2.3.2 Server Software Block Diagram

The diagram below shows server side software modules and their dependencies.



[Server Software Block Diagram]

FaceNet

The face recognition library includes TensorRT and OpenCV with CUDA accelerator.

OpenSSL v1.1

To protect against data sniffing on the network between JetsonNano and user laptop, use OpenSSL for data encryption.

To protect against connection from unknown clients, used openssl for authentication key from trusted certificate authority.

CryptMount

To protect credentials and image files. secure storage is implemented using cryptmount which is a software tool for managing encrypted file systems under the Linux family of operating systems.

rsyslog

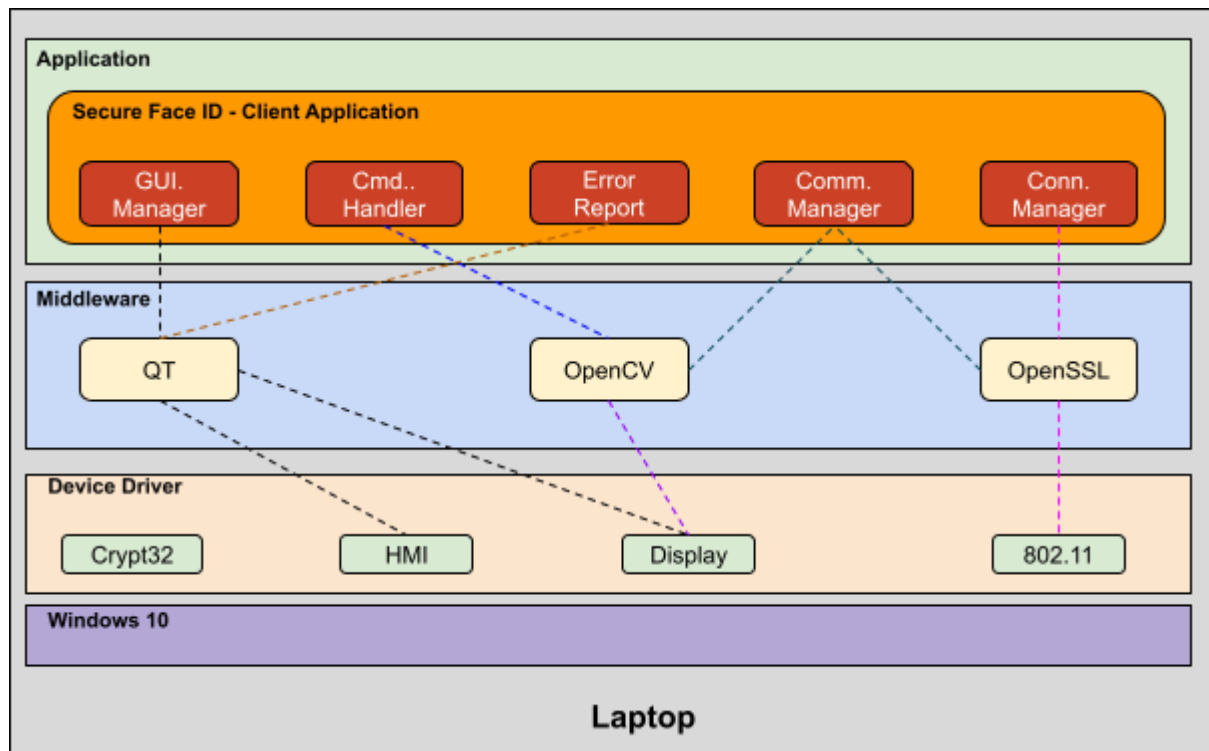
To handle errors properly and verify connection status.

Secure Face ID Server

To recognize faces over the camera, the server application also has input validation checking logic.

3.2.3.3 Client Software Block Diagram

The diagram below shows client side software modules and their dependencies.



[Client Software Block Diagram]

QT Framework

GUI toolkit for windows 10

OpenCV

The computer vision library toolkit for machine learning and artificial intelligence.

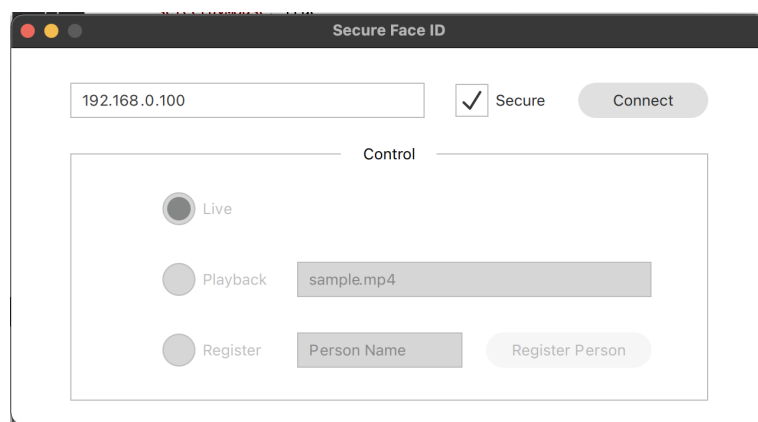
OpenSSL v1.1

To protect against data sniffing on the network between JetsonNano and user laptop, use OpenSSL for data encryption.

To protect against connection from unknown clients, used openssl for authentication key from trusted certificate authority.

Secure Face ID Client

To defend invalid input data of IP address and user name, the client application has input validation checking logic.



[Client application Qt GUI]

3.2.4 System Specification

Technical Specification of Server

CPU, GPU	Quad-core ARM Cortex-A57 , 128 NVIDIA CUDA® cores
DRAM	4 GB 64-bit LPDDR4, 1600MHz 25.6 GB/s
Storage	16 GB eMMC 5.1
Video	250MP/S Encoder, 500MP/S Decoder
Camera	12 lanes (3x4 or 4x2) MIPI CSI-2 D-PHY 1.1
Network	Gigabit Ethernet, 2.4Ghz 5.8Ghz Dual Band 802.11 A/C
OS	Linux
Software	OpenCV, FaceNet, OpenSSL

Technical Specification of Client

CPU, GPU	8th Gen. Intel® Core™ i7-8550U 1.80GHz/4.00GHz
DRAM	16GB DDR4 2400MHz - 8GB x 2
Storage	1TB M.2 SSD
Network	Gigabit Ethernet, 2.4Ghz 5.8Ghz Dual Band 802.11 A/C
OS	MS Windows 10
Software	QT, OpenCV, OpenSSL

3.3. Implementation

3.3.1 Input Validation

In the client application, there are several input items needed to consider verification. IP address, User name and Image data as well. Attackers can try to input malicious texts to the normal inputs like IP address and user name, also to temper an image data during communications.

For the normal inputs, the client application denies malicious user inputs by preventing special characters and too long inputs.

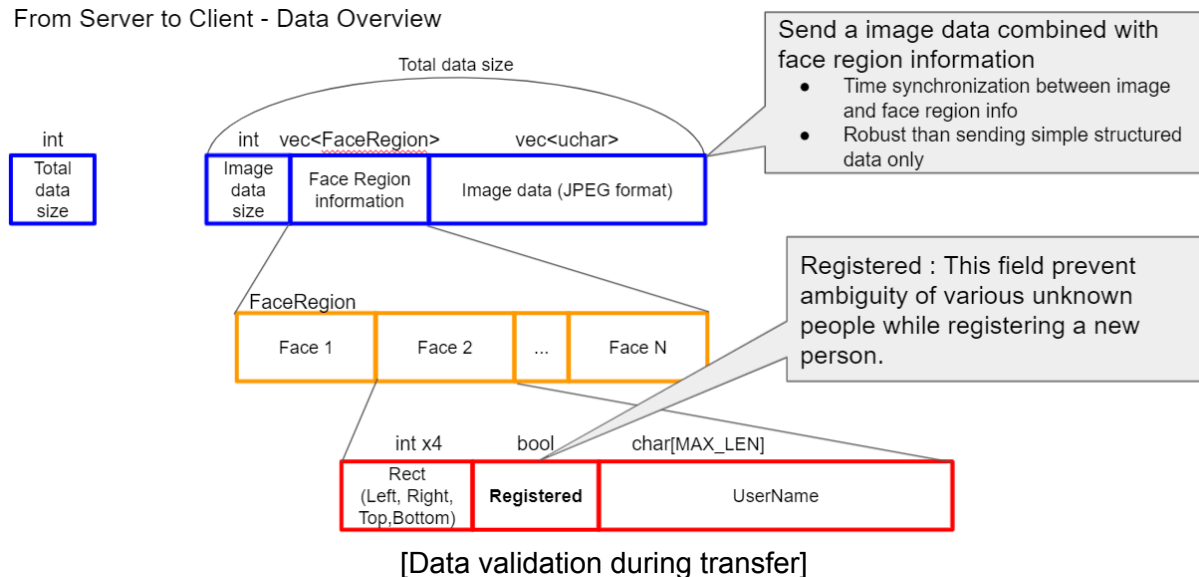
For the image data, the client application checks the image format of received data at client side. Basic logic is the investigation of SOI (Start of Image) and EOI (End of Image). SOI and EOI values for JPEG format are fixed values, so that they can be easily checked if the data format is correct.

3.3.2 Secure Data Transmission

TLS is enforced on the client and server for a secure channel. In addition, the server sends image data combined with face region information for time synchronization between image and face region info and for robustness.

- X.509 Certificate
 - Long key length: 4096 bits, AES-256 encrypted
 - Stored in secure storage (encrypted, not accessible to unauthorized user)
 - Permission to the keys are restricted so that only the owner can read and no one is able to write and execute
 - Certificate status (valid or revoked) are managed
- TLS
 - TLS v1.3: faster handshake and stronger security by removing static key exchanges
 - Cipher Suite: TLS_AES_256_GCM_SHA384
 - AES 256: according to NIST Recommendations
 - GCM: provides both confidentiality and authentication using AAD
 - SHA384: susceptible to length extension attack
- Data format

From Server to Client - Data Overview



3.3.3 Secure Coding

To protect software vulnerabilities and exploits, team 5 analyze the code using static analysis tools, the Flawfinder and Cppcheck, and fix the code which has a weakness found from static analysis tools.

To protect against software vulnerabilities and exploits, Team 5 used static analysis tools, Flawfinder, and Cppcheck to analyze code and fix code with vulnerabilities found by static analysis tools.

- Static Analysis with Flawfinder

We chose flawfinder for static analysis because flawfinder is a simple program that examines C/C++ source code and reports possible security weaknesses ("flaws") sorted by risk level. Furthermore It is free .

- Fix flaws which can be occurred overflow
 - Fix strcpy to strncpy to guarantee that storage for strings has sufficient space for character data and the null terminator.
 - Fix sprintf to snprintf to guarantee that storage for strings has sufficient space for character data and the null terminator.
 - Fix destination length of memcpy
- Ignore false positive
 - Ignored flaws about statically-sized arrays because check the size when it is used.
 - Ignore flaws about possible race conditions when the file opens/reads. Team 5 have not spent time fixing the flaws because it's difficult to be attacked.

ID-7	src/pnet_rt.cpp:41: [2] (misc) open: Check when opening files - can an attacker redirect it (via symlinks), force the opening of special file type (e.g., device files), move things around to create a race condition, control its ancestors, or change its contents? (CWE-362).	ignore-false positive	<pre>protofile.close(); protofile.open("temp.prototxt", ios::out); //flawfinder_5verflow : ignore protofile.write(contents.c_str(), contents.size());</pre>
ID-8	src/baseEngine.cpp:55: [1] (buffer) read: Check buffer boundaries if used in a loop including recursive loops (CWE-120, CWE-20).	ignore-false positive	<pre>std::cout << "size" << trtModelStream.size() << std::endl; file.read(trtModelStream_data(), size); //flawfinder_5verflow : ignore file.close();</pre>
ID-9	src/faceNet.cpp:43: [1] (buffer) read: Check buffer boundaries if used in a loop including recursive loops (CWE-120, CWE-20).	ignore-false positive	<pre>std::cout << "size" << trtModelStream.size() << std::endl; file.read(trtModelStream_data(), size); //flawfinder_5verflow : ignore file.close();</pre>
ID-10	src/main.cpp:122: [1] (buffer) read: Check buffer boundaries if used in a loop including recursive loops (CWE-120, CWE-20).	ignore-false positive	<pre>unsigned char c; if ((r = read(0, &c, sizeof(c))) < 0) { //flawfinder_5verflow : ignore return r;</pre>
ID-11	src/main.cpp:766: [1] (buffer) read: Check buffer boundaries if used in a loop including recursive loops (CWE-120, CWE-20).	ignore-false positive	
ANALYSIS SUMMARY:			
Hits = 11			

[Final results of static analysis with flawfinder]

* *document* : “Static Analysis”

3.3.4 Setup Guide

Refer to “Developer Guide” or <https://canvas.cmu.edu/groups/59086/collaborations/2592>

3. Guide for build and run our system

1.1 How to get the source code

Github : <https://github.com/5verFlow/sfid-document>

1.2 How to build the system

1.2.1 Server (Linux, Jetson Nano)

Step 1. Install Secure Storage

This project uses cryptmount, an encrypted filesystem, to store registered user image files and credentials.

1. Install cryptmount
\$ sudo apt install cryptmount
2. Set up crypt mount for a target filesystem
\$ sudo cryptmount-setup

[Developer Guide]

* document : "Developer Guide"

3.3.5 Verification (Static Code Analysis)

We'd conducted static code analysis by using flawfinder and fixed issues up.

Hits = 16		
Lines analyzed = 3454 in approximately 0.33 seconds (10488 lines/second)		
Physical Source Lines of Code (SLOC) = 2577		
Hits@level = [0] 56 [1] 5 [2] 10 [3] 0 [4] 1 [5] 0		
Hits@level+ = [0+] 72 [1+] 16 [2+] 11 [3+] 1 [4+] 1 [5+] 0		
Hits/KSLOC@level+ = [0+] 27.9395 [1+] 6.20877 [2+] 4.26853 [3+] 0.388048 [4+] 0.388048 [5+] 0		
Minimum risk level = 1		

[Result of Static Analysis]

* document : "Static Analysis"

3.4. Testing

We wrote test cases according to the security requirements, and each item is mapped to the security requirements. For example,

- Test case ID : TC-03
- Category : [Input validation] - Verify input username using VALID format
- Associated security requirement : SR 1-2
- Descriptions : This verifies **SR 1-2** that server and client should check respectively whether the input for username field on the register mode is valid as a filename.

Category	Test Case ID	Test Descriptions	Test Step
Precondition			Prepare the server application on J... fixed port number to connect with th... application. Execute the client application on wi...
[Input validation] Verify input IP address using VALID format	TC-01	This Verifies SR 1-1 that Client Application must check if the format of input IP address is in valid format.	[Positive] 1. Select Insecure mode by uncheck... check box. 2. Select 'Live' radio button. 3. Enter a valid ip address. 4. Click 'Connect' button
[Input validation] Verify input IP address using INVALID format	TC-02	This Verifies SR 1-1 that Client Application must check if the format of input IP address is in valid format.	[Negative] 1. Select Insecure mode by uncheck... check box. 2. Select 'Live' radio button. 3. Enter a invalid ip address. 4. Click 'Connect' button
[Input validation] Verify input username using VALID format	TC-03	This Verifies SR 1-2 that Server and Client should check respectively whether the input for Username field on the Register mode is valid as a filename.	[Positive] 1. Select secure mode by checking... box. 2. Select 'Register' radio button. 3. Click 'Connect' button with valid I... 4. Enter valid user name. 5. Click 'Register Person' button w... person is recognized. 6. Change mode to 'Live' by selectin...
[Input validation] Verify input username with duplication	TC-04	This Verifies SR 1-2 that Server and Client should check respectively whether the input for Username field on the Register mode is valid as a filename.	[Positive] 1. Select Insecure mode by uncheck... check box. 2. Select 'Register' radio button. 3. Click 'Connect' button with valid I... 4. Enter an user name same with T... 5. Click 'Register Person' Button. 6. Change mode to 'Live' by selectin...

[Team 5's Test Cases]

* document : "Test Cases"

4. Phase 2: Security Evaluation of Classmate System

4.1. Analysis of Classmate System (Team 6)

4.1.1 Document Analysis

- System Requirements

As a result of analysis of the team 6 requirements document, almost everything is the same with us but team 6 defines 2 additional requirements more than us. It's about using user authentication and recovering from networking errors.

ID	Description			
CMU-REQ-D-01	Establishing connection between Client and Server			
CMU-REQ-D-02	A user should be able to initiate a video feed, and a feed in Support Secure Mode			
CMU-REQ-D-03	A user should be able to initiate a video feed, and a feed in Support Non Secure Mode			
CMU-REQ-D-04	Support Learning Mode - Register new person to the Server			
CMU-REQ-D-05	Support Run Mode - System identifies faces and performs facial recognition			
CMU-REQ-D-06	Support Test Run Mode ; A user should be able to tune image analysis with local file			
CMU-REQ-D-07	Display Result - Face-recognized images			
CMU-REQ-D-09	The system must use 2FA			Yellow Cell : Not considered a requirement in a Team 5 project
CMU-REQ-D-10	User credentials must be protected			
CMU-REQ-D-14	Reliability, the video is reliably delivered.			
CMU-REQ-D-15	Recover from networking errors asap			
CMU-REQ-Q-01	Architecture Docs			
CMU-REQ-Q-02	Coded to be secure and free of vulnerabilities			
CMU-REQ-Q-03	Proper fault/error detection, recovery, reporting			
CMU-REQ-Q-04	Analying initial implementation for vulnerabilities			
CMU-REQ-Q-05	Developing solutions to mitigate			

- Security Goal

Defined as "Protecting the user's privacy information in our system" in a document.

So we have focused on trying to attack the goal.

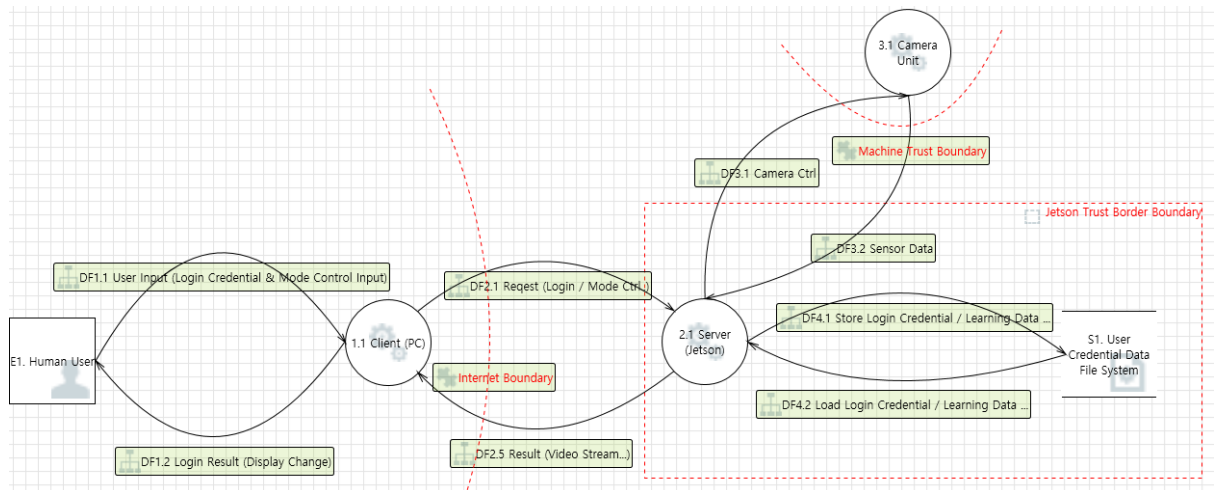
- Threat Modeling

Team 6 used STRIDE and PnG methods for threat modeling and also brainstormed.

The total number of threats is 28, 21 identified by STRIDE, 3 by PnG, and 4 by brainstorming.

- STRIDE with Microsoft Threat Modeling Tool

The threats are almost all about the JetsonNano Server boundary that has user credential data storage and about communication for video stream.





[Team 6's Data Flow Diagram]


○ PnG

The threats are about user authentication and about communication for user credentials. It seems that these are not identified on STRIDE by Microsoft Threat Modeling Tool.

1. Identify the PnG types, goals, motivations, skills

	Type	Internal Engineer
	Goal	Ruin the administrator's reputation
	Motivation	Revenge to the administrator
	Skill	manipulate the user credential data, find out the administrator's password from the previous one that is used to other system
	Misuse case	1. Change the image data not to recognize registered users. 2. Disclose administrator's ID/Password to the employees in the company.

	Type	Spy
	Goal	Steal all components of the system
	Motivation	Competitors request
	Skill	Physical power and ability to use various equipment
	Misuse case	1. Steal the client and server

	Type	Hacker
	Goal	Post the achievements of hacking on the internet
	Motivation	Strives for recognition
	Skill	Extensive knowledge of network protocols and hacking program
	Misuse case	1. Sniff the communication channel between server and client to get user credential data.

○ Brainstorming

The threats are physical network attack, certificates attack, face recognition data attack and root key used for encryption attack. It seems more details and consideration for the team 6 system.

● Risk Assessment

Team 6 used OWASP Risk Rating Methodology and identified 3 critical risks, 13 high risks and 12 medium risks. Critical risks are about threat of user credentials and user id/password.

Threat Group	Factors for Estimating Likelihood					Factors for Estimating Impact					Overall Risk Severity
	Estimating Factors	Factors	Range	Likelihood		Estimating Factors	Factors	Range	Impact		
				Score	Severity				Score	Severity	
Information Disclosure	Threat Agent	Skill level	6 - Some technical skills	6.125	HIGH	Technical Impact	Loss of confidentiality	9 - All data disclosed	6.5	HIGH	Critical
[Threat] If the user credential data is stored as plain text, it can be disclosed.		Motive	4 - Possible reward				Loss of integrity	7 - Extensive seriously corrupt data			
		Opportunity	4 - Special access or resources required				Loss of availability	3 -			
		Group Size	7 -				Loss of accountability	7 - Possibly traceable			
	Vulnerability	Ease of discovery	7 - Easy	Business Impact	Financial damage	7 - Significant effect on annual profit					
		Ease of exploit	8 -		Reputation damage	7 -					
		Awareness	6 - Obvious		Non-	5 - Clear					

[Team 6's Risk Assessment]

- Security Requirements
 - Manage user info like learned user face photo and user name securely
 - Manage user id / pass securely
 - Manage communication channel securely
- Evaluate of documents & Results of Analysis

As team 6 system's security goals, it seems that they focused on protecting user credentials and user id/password. So we, team 5, also have tried to attack their security goals.

We also analyze the threat list, security requirements, and mitigation of the Team 6 project, then identify if anything is missing.

When trying to compare and match TRID(Threat ID), SRID (Security Requirements ID) and MID (Mitigation ID), TR03, TR1 and TR52 do not have security requirement id even though they have mitigation id. Furthermore, threats are about server input validation and mitigation is about client input validation. So it looks like team 6 is missing a security requirement.

Written by Team 6							
Checked and Analyzed by Team 5							
TR-ID	Threat	Review	SR-ID	Security Requirement	MI-ID(matched to SR-ID)	Mitigation	Evaluation by team 5 -> Evaluate
TR-01	If the user credential data is stored as plain text, it can be disclosed.	User credential should be kept securely	SR-13, SR-12	User Credential Data should be encrypted in the storage. Use well-known cryptographic libraries and robust algorithms.	MI-03, MI-07	Encrypt user credential data in storage - Use OpenSSL library of latest version (1.1.1k) - Use an algorithm that are stronger than AES256 - Use cbc of gcm mode Integrity Check with hash - Use an algorithm that are stronger than sha256	
TR-02	An attacker modify user credential data.	User credential should be kept securely	SR-11, SR-10, SR-12	the system shall know the change of the user credential data. The system must perform an integrity check before using user credentials. Use well-known cryptographic libraries and robust algorithms.	MI-03, MI-07	Integrity Check with hash function - Use OpenSSL library of latest version (1.1.1k) - Use an algorithm that are stronger than sha256 Encrypt user credential data in storage - Use OpenSSL library of latest version (1.1.1k) - Use an algorithm that are stronger than AES256	
TR-03	An attacker modify user credential data and then server can use it without checking.	User credential should be kept securely	SR-11, SR-09, *No matching SR	the system shall know the change of the user credential data. Server and client must communicate using an encrypted channel. Input validation check is required in Server side.	MI-05, (It seems to assume that do not considering in server side.) MI-02, MI-07	Input validation check - Input sanitization Integrity Check with hash function - Use OpenSSL library of latest version (1.1.1k) - Use an algorithm that are stronger than sha256 Communicate using Encrypted channel - using protocol TLS1.2 or higher - Consider mutual authentication	Need to make sure the server is doing input validation.

[Team 5's Analysis about Team 6's Threats]

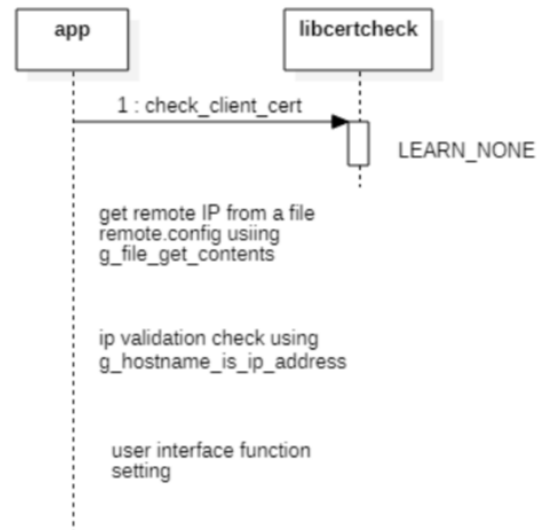
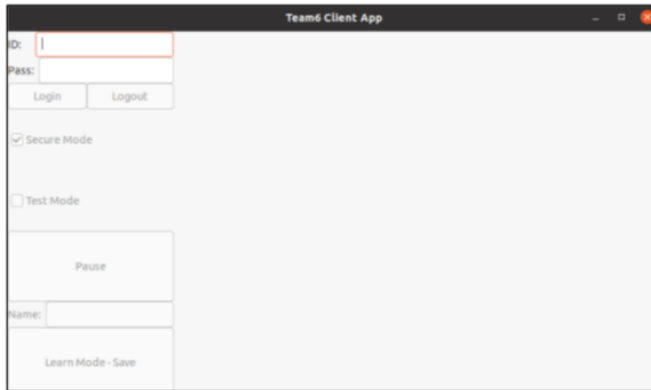
4.1.2 Implementation Analysis

We analyzed the source code of team 6 project and then made a sequence diagram. It describes a main sequence diagram focused on finding vulnerabilities that can attack the security goal of team 6 protecting users' privacy.

- Client Initialization
 - Check client certifications
 - Load configuration from config file and check validation of server IP address

Client Initialization

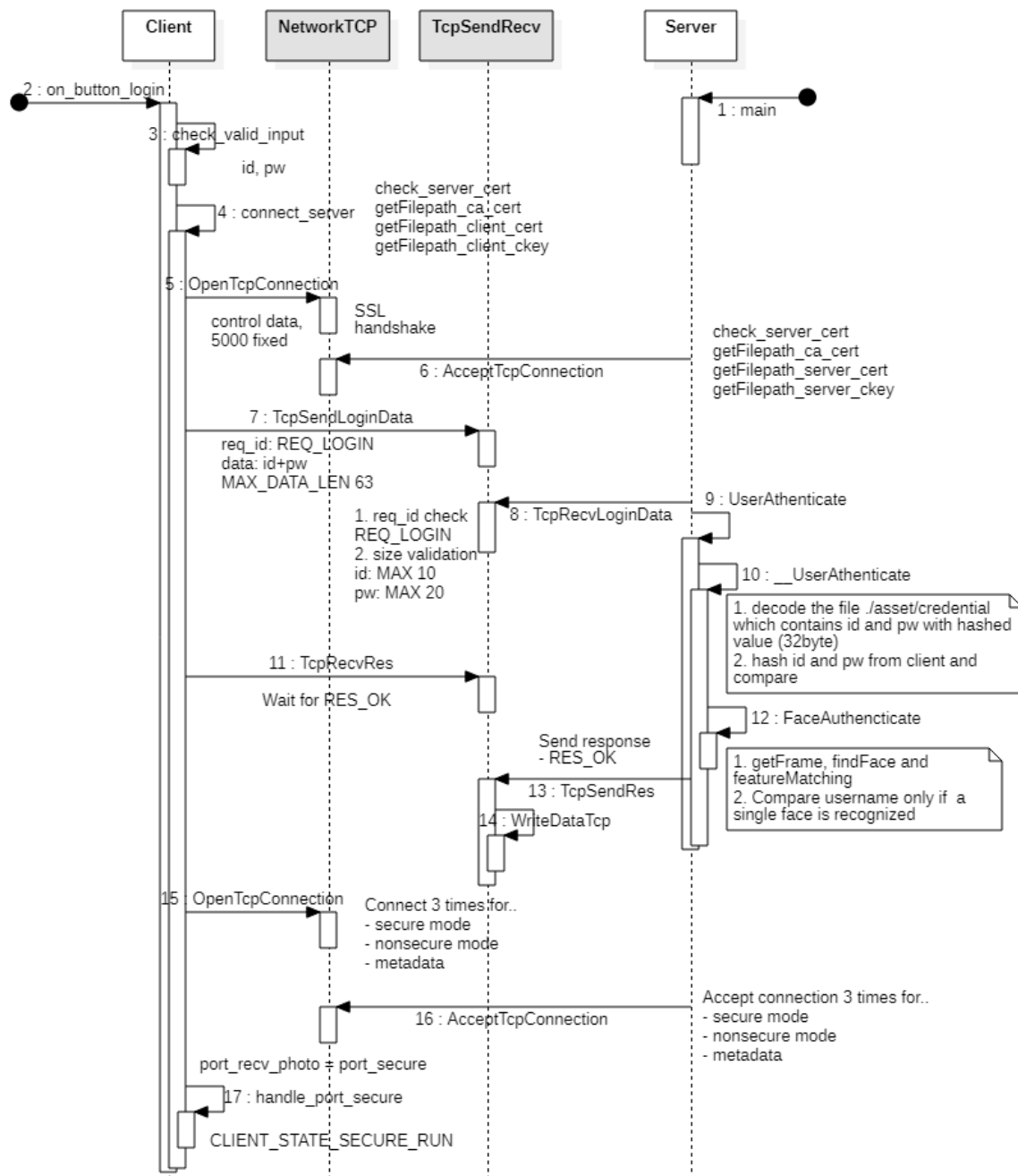
get server IP address from config file as plain text
IP validation



[Sequence diagram - Client Initialization]

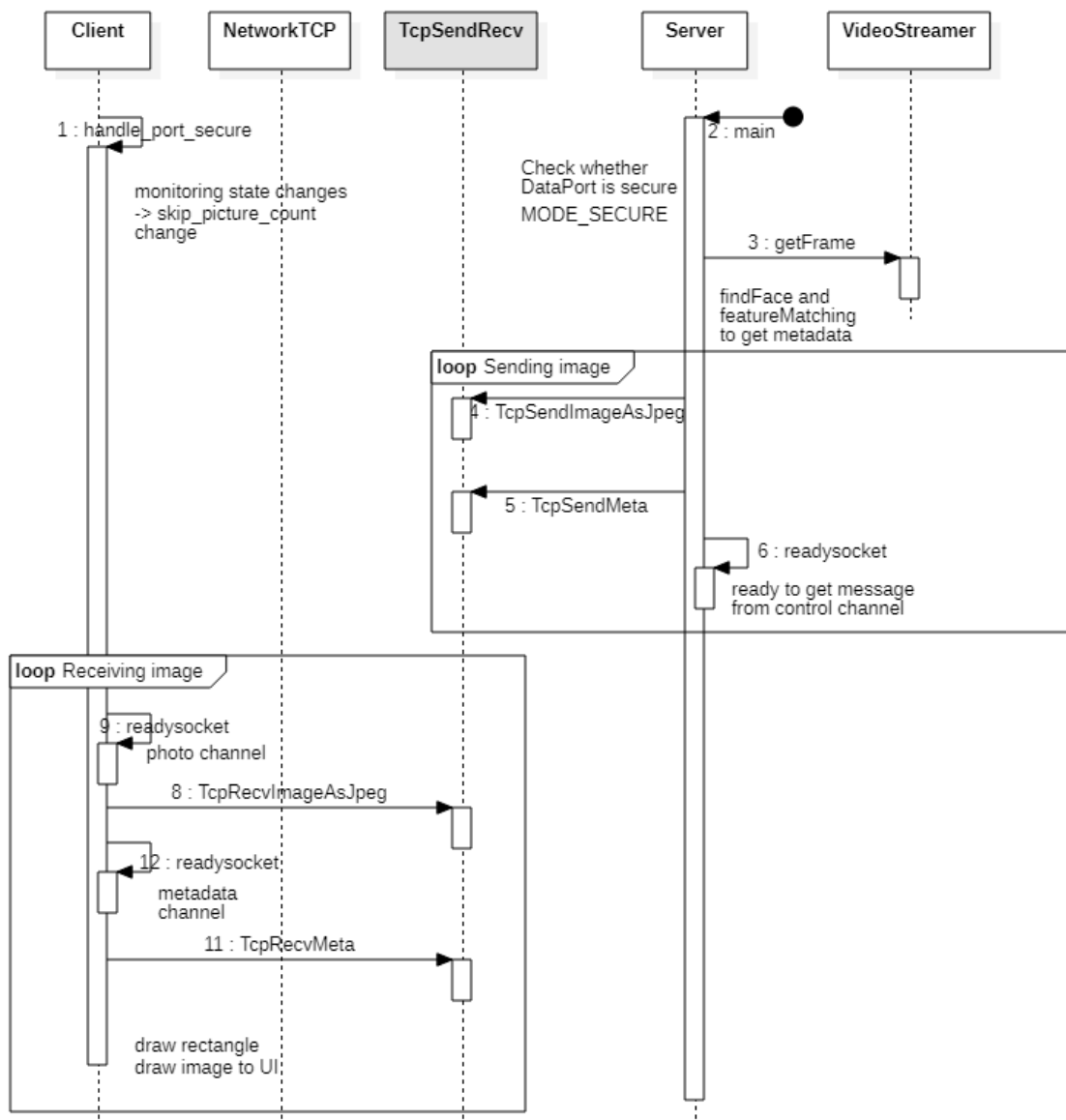
* document : Sequence Analysis of server and client

- Login
 - Client
 - Check input validation for ID/Password
 - Establishing connection using certification
 - Server
 - Authentication of ID/Password with credential file
 - Authentication of user face with ID
 - Image is not sent to client



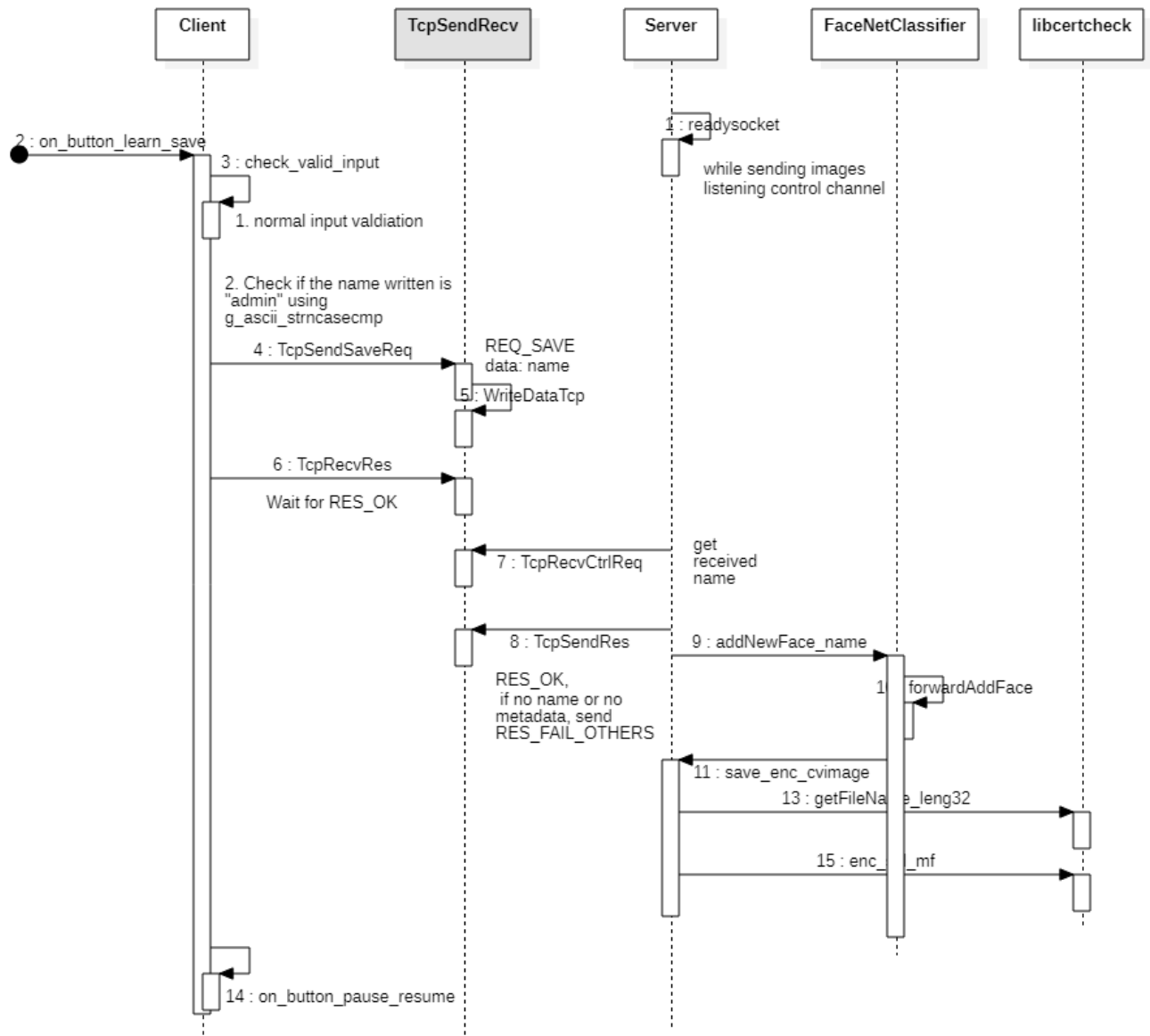
[Sequence diagram - Login]

- Data communication
 - There are two channels
 - A transfer channel for image
 - A transfer channel for metadata of image information



[Sequence diagram - Data communication]

- Save user image
 - Client
 - Check input validation for the name
 - Admin is not allowed
 - Server
 - Save image with encryption



[Sequence diagram - Save user image]

* document : "Sequence Analysis of server and client "

4.2. Found Vulnerabilities

Using the analysis techniques mentioned above, we found the vulnerabilities on the software. The list of found vulnerabilities, attack scenarios to exploit them, expected results and their impacts, and the recommendations to mitigate the threats are provided in another document ([Attack Scenario](#)).

The summary of the found vulnerabilities are given in the table below.

Method	Number of found vulnerabilities	takeaway
Manual Code	17	Source code has full and correct information about the software,

Review		so it's a good method to find internal security holes for attackers accustomed to programming languages. Risks on this method are: <ul style="list-style-type: none"> • illegible code might be almost impossible to read • takes time to understand without documents
Documents	8	The number we found is less than those from code review and more than those from tests. Documents show well-organized information to the program and specifically, security-related documents were very helpful to find out what the developers have neglected. Risks on this method are: <ul style="list-style-type: none"> • If a document is not written well or not updated for the latest commit of source code, it might misguide the reader.
Tests	4	The smallest number of vulnerabilities are found from tests, which includes static and dynamic analysis. As the developers have already run several tests, a few new results were found. Dynamic analysis takes some time to set up environments and derive meaningful scenarios. Risks on this method are: takes time for preparations <ul style="list-style-type: none"> • Meaningful run scenarios should be derived for efficient testing, which requires some understanding of software.
Total	29	

The number of found vulnerabilities is in proportion to the easiness of the method and the quantity of information the material possesses, with addition/subtraction to external factors such as: already handled by the programmers.

$$N_v = Easiness \times Quantity\ of\ Information + \alpha_{EXT}$$

Source code itself includes full description, thus providing plentiful information of the system. More accustomed to the programming language an attacker is, less time taken to derive exploitable vulnerabilities. As all the attackers (team 5) members are familiar with C and C++, the majority of vulnerabilities are found from source code written in C and C++.

CIA	Number of found vulnerabilities	takeaway
Confidentiality	3	As TLS authentication covers many vulnerabilities related to confidentiality features, it was hard to find out exploitable vulnerabilities. In addition, Team 6 has implemented two-factor authentication which makes it a lot more difficult. In case only single authentication methods were installed such as login ID and password, it might have been possible to exploit vulnerabilities using brute-force attack.

Integrity	5	<p>To harm integrity features it is required for an attacker to be able to access the data. As data on transmission is protected by TLS encryption and data stored in the server is unable to access unless its IP address and SSH login ID and passwords are revealed.</p> <p>We (Team 5) have derived several vulnerabilities as the IP address of the server is revealed by plain-text configuration file of the client application, and penetration into the server's file system was successful from brute-force attack on SSH login ID and password.</p> <p>Yet most of the attacks on integrity have failed as the confidential data stored in the server (registered users' names and photos, credentials for application login) are encrypted (hashed), and data on transmission is protected by TLS and TCP/IP.</p>
Availability	16	<p>Most of the vulnerabilities we found are to attack availability features by cracking files or systems.</p> <p>As it tends to be more simple to break systems or resources than sneak into the system. For example, simply occupying certain resources could harm availability. In addition, it is hard or sometimes impossible to be handled from the application software perspective. From our experiences in phase 1, we cannot help but assume that availability is assured by OS or network router, so we exploited those experiences to attack the other's.</p>
Complex	5	<p>Some of the vulnerabilities we found affect multiple components. One of them, weak SSH ID and password leads to penetration that makes numerous attacks simple and possible.</p>
Total	29	

4.3. Attack Scenarios

The tables below summarize some of the attack results.

4.3.1 Scenario 1

Problem Description	Server IP address in plaintext
Related Security Issues	Authentication, Integrity, Availability
Activity/Tool used	ZZUF
Trigger, Evidence, Proof-of-concept	From the development guide document and artifacts, we discovered that the server IP written in the remote.config file is plaintext. We also found this issue in the source code as well.
Impact, Risk Rating, and/or Severity	It is easy to compromise the config file, and the client cannot access the server if the server IP address is tampered.
Containing artifacts	Attack Scenario document
Date found	2021/06/23

Issuer	Seungwook Cha
--------	---------------

4.3.2 Scenario 2

Problem Description	Video Sniffing without Authentication
Related Security Issues	Confidentiality
Activity/Tool used	None
Trigger, Evidence, Proof-of-concept	The vulnerability is found from code review, that it is practicable for an attacker to intervene and establish connections right after the authentication.
Impact, Risk Rating, and/or Severity	The attacker can sniff the video, thus private information could be revealed.
Containing artifacts	Attack Scenario document
Date found	2021/06/24
Issuer	Woolam Kang

4.3.3 Scenario 3

Problem Description	A face can have various names
Related Security Issues	Authentication, Authorization
Activity/Tool used	None
Trigger, Evidence, Proof-of-concept	From the code review, we found that there is no procedure or method to separate registered faces and new faces. Therefore, a client can add another name for the same face.
Impact, Risk Rating, and/or Severity	If a face is registered with another name, it could be exploited to the elevation of privilege or spoofing.
Containing artifacts	Attack Scenario document
Date found	2021/06/25
Issuer	Seungwook Cha

4.3.4 Scenario 4

Problem Description	Disassemble and modify the binary
Related Security Issues	Confidentiality and integrity
Activity/Tool used	rizin, radare2
Trigger, Evidence, Proof-of-concept	from manual code review, it is found out authentication took place only once and the

	credentials are not used anymore
Impact, Risk Rating, and/or Severity	The impact is severe since unauthorized users are able to access the system
Containing artifacts	Vulnerability Attack Report
Date found	2021/06/29
Issuer	Donghyuk Han

4.4. Testing

Additional information and details on attacks and testings are described in [Vulnerability Attack Report](#) and [Attack Scenario](#)

Vulnerability Attack Report

Team 5

|

[Team 5's Vulnerability Attack Report]

* document : Vulnerability Attack Report

ID	Reconnaissance Phase	Condition	Vulnerability	
AS-00 (example)	How did we find out? (Analysis documents, Code Review, nmap, packet sniffing, Nessus, etc.)	Reproducible Situation (e.g., Server Application runs with ./LgFaceRecDemoTCP_Jetson_NanoV2 20000 (invalid port num))	Vulnerabilities found (e.g., Server Application does not allow a port except 5000)	Attac (e.g.,
AS-15	Team 6 Artifacts - Source Code : Manual Code Review	during connection establishing	Sniffing - there are 4 sockets in a port to connection but log in is conducted only in first socket.	timin it cou after
AS-13	Team 6 Artifacts - Source Code : Manual Code Review	run the client app	Denial of Service, Information disclosure server IP in remote.config file is a plaintext	immu
AS-30	Team 6 Artifacts - Source Code : Manual Code Review	when press 'pause' button to enter Learning mode	Any faces can be the candidate to save no matter what if the face is already registered. That is, a person can have different names	Try to name
AS-16	Team 6 Artifacts - Source Code : Manual Code Review	during connection establishing	Same as above (AS-15)	timin If we conft
AS-20	Team 6 Artifacts - Source Code : Manual Code Review	when client request connection	no consideration for the several connection attempt of clients simultaneously	what at the
AS-22	Security Testing - nmap (network scanner)	when server is listening port	there is no recovery logic when failed to handshaking	any z

5. Reflection

Throughout the entire phase of the secure development lifecycle and brief experience on DevSecOps, some of the valuable lessons are derived.

The first impression of the secure development lifecycle was that security considerations and decisions should be made in every step of the development cycle. It was a bit surprising since Team 5 members used to regard the security considerations as an additional process. Detecting and resolving security issues early in the life cycle significantly reduces costs. Several flaws found later in implementation and testing phases are hard or impossible to be handled. Thus it'd be much better if developers are skilled and well-experienced and take thorough actions in each phase. Especially in the implementation phase, some codes were first written to meet the functional requirements and after the functionalities are confirmed security-related features are added. These two-step writings have delayed the development period.

Phase 2 of the project also provides takeaways. Some of the key takeaways are as follows.

First, being a red team to attack the system or software is an important phase to discover vulnerabilities. Though similar actions are performed in phase 1 of the project - think like an attacker and try to destroy the system and softwares, phase 2 gave many new considerations on vulnerabilities. This comes from two following factors: 1) Developers give a friendly look at their system and softwares so it's hard to take an objective view. 2) As Team 6 (developers of the target software) took separate steps, their derivation on the design and implementations are somewhat different to attackers' perspectives. This results in a diversity gain that attackers could find out some vulnerabilities that developers couldn't realize. These lessons have something in common with the benefits of DevSecOps, of which one of the main benefits is improved security. Collaboration between development and security teams improves software security.

Second, various techniques to find vulnerabilities should be utilized, including manual code review, documents analysis, static and dynamic tests, etc. Considering the performers' specialty and professionalism, some techniques might be more powerful. In the case of Team 5, manual code reviews result in the largest number of exploitable vulnerabilities found. As all the members are developers and are accustomed to C and C++, it was able to discover many things within rather a short period of time. In addition, the source codes are well organized in a very readable way and it also helped a lot. Such a whitebox analysis is very effective when developers act a role as a black hat. Team 5 put a lot of effort on code review and even drew a sequence diagram, and these activities helped a lot to understand the systems and find various vulnerabilities. When resources are restricted in a way that insufficient human resources, few computer devices for analysis, or time is limited, analysis on documentation would be very helpful as well as static analysis. In case of having rich resources, code review with an aid of documents as well as static and dynamic tests is going to be fulfilled. In short, Although some of the techniques were very useful in some cases, various techniques should be conducted to get diverse findings.

Comparing the outputs of Team 6's and those of Team 5's was also helpful and interesting since the two different outputs are meant to fulfill the same objectives. One lesson from team 6's is the importance of two-factor authentication, which makes it a lot more difficult to exploit

or even find vulnerabilities. Binary obfuscating is also a good method to deter reverse engineering, of which impact would be enormous owing to its ability to do many attacks. Some of the same features such as TLS are proven to be effective methods to protect data in air from numerous attacks.

6. Conclusion

Software security is growing as one of the most essential features. Therefore the security concerns should be reflected in each step of the development process. Throughout the development cycle, security issues are addressed and identified to meet the security goals. Derived security requirements are reflected in design and implementations. Developers should be aware of security risks and vulnerabilities so that security objectives are integrated as early as possible and the cost of security be reduced. To improve security integration, collaboration between the development team and the security team is essential.

Development team should ensure the software is built with secure features. In order to apply various techniques and policies, prioritization should be made. There always is no time and resources to handle all the vulnerabilities and apply all the methods. Vulnerabilities with significant impacts must be handled first and mitigations techniques and test methodologies that cover a diverse range of threats should be chosen.

To deal with evolving security threats and challenges, developers should be trained to give a rationale for the decisions on choosing threats and mitigation methodologies. Greater trust in the security of developed software and embracing new technologies would enable enhanced revenue growth and expanded business offerings.

Appendix

Team 5's artifacts

Artifact	Description
Team 5 Project Planning	Team organization and project plan, Each responsibility
User requirement Document	User requirements
Project Requirement Document	Project requirements
Software Requirement Specification	Functional requirements
Project Asset List	Asset List
Security Requirements	Quality Attribute

Software Architecture Design	Architecture document
Static Analysis	Flawfinder report
Test Cases	Test Cases
Developer Guide	Guide documents for classmate
Phase 1. Presentation	Presentation for phase 1
Attack Scenario - Brainstorming	Attack scenario, Analysis of security requirement and test case, Static analysis (FlawFinder, CppCheck, Code X-ray)
Sequence Analysis of Server and Client	Code analysis
Vulnerability Attack Report	Description of attack scenarios and results
Final Report	Final report
Phase 2. Presentation	Presentation for phase 2