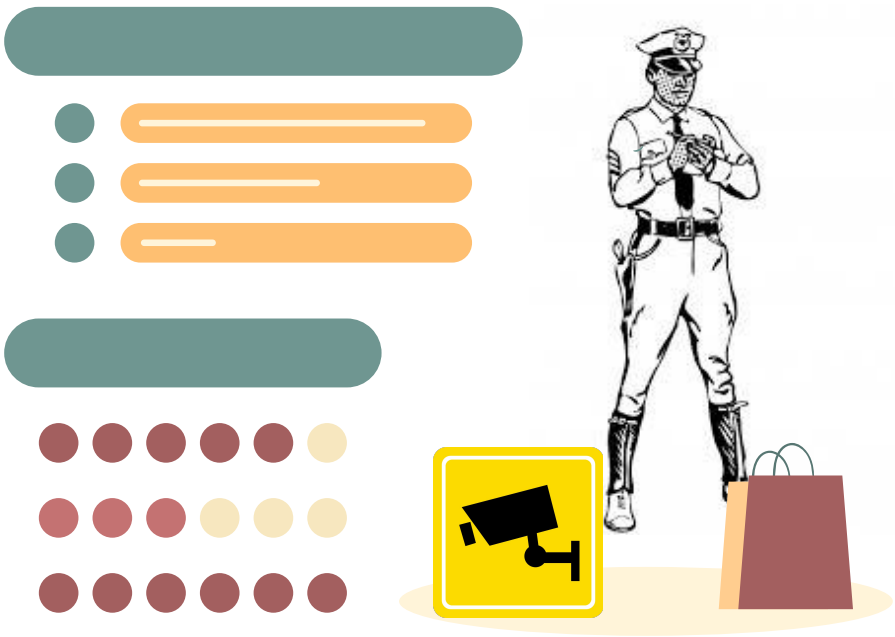
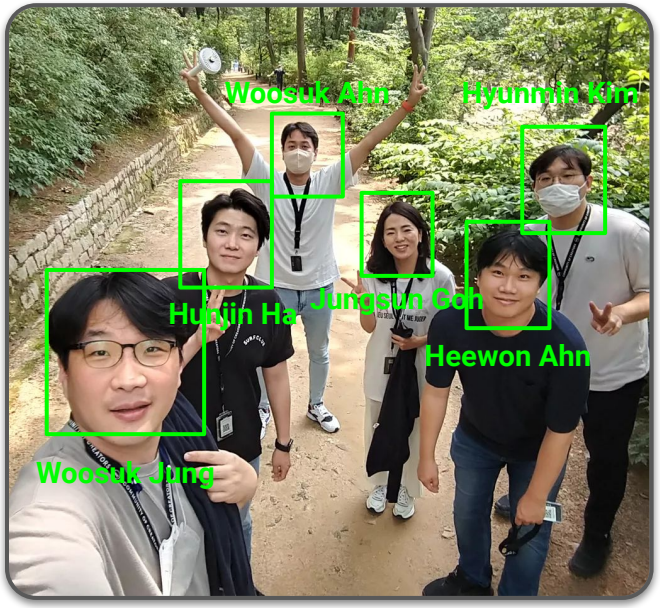
2022
LG Security Specialist
Studio Project
Team 2

Final Presentation
July 15, 2022

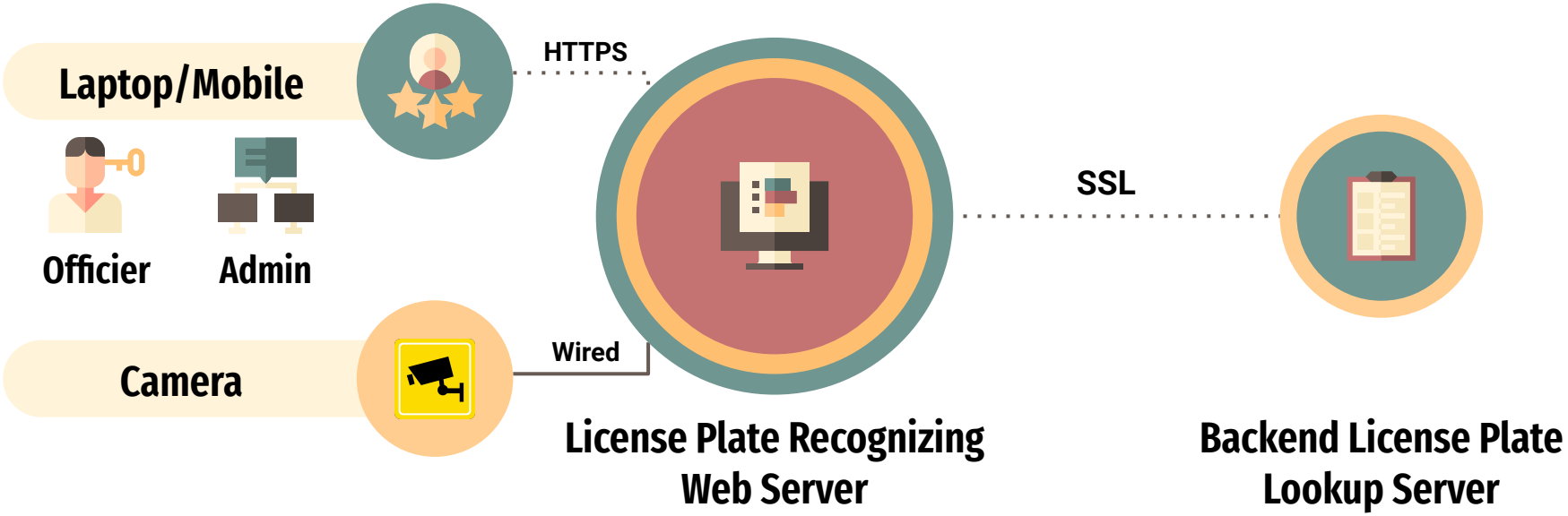


We Are AhnLab!



Members		Role & Responsibility
Woosuk Ahn	Web Backend	*Implement UI and ALPR vehicle control and recognition system and Web*
Heewon Ahn	Web Auth	*Implement the Web authentication part*
Woosuk Jung	Lookup Server	*Implement Backend License Plate Server and DB*
Hyunmin Kim	Lookup Server	*Implement Backend License Plate Server and logging part*
Hunjin Ha	Secure Channel	*Implement SSL protocol*
Jungsun Goh	Documentation	*Quality assurance and documentation*

Studio Project Overview



Security Goal

Goal

The client application should communicate securely with a backend server that contains relevant information.

Confidentiality & Integrity

An officer shall securely access the system and uses data.

Authentication

The identity of the person who is accessing the data and resources in the system shall be verified before access.

Authorization

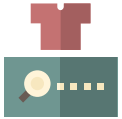
Administrators shall access and modify configuration file.

Availability

An officer shall use the system in real time at any time he or she wants.

Assets

Personal Information



License Plate Number
Vehicle Status
Owner Address
Owner Zip Code
Owner Birth of Date

User Credentials

User ID
User Password

System Config.

Number of Max User
Confidence Level
Lookup Server IP
System Logs

Threats

1 Asset Identification

Assets

Asset	Damage Scenario	Asset ID
License Plate Number	If a license plate number is manipulated, a police officer cannot get valid information or a criminal can avoid law enforcement.	AID-01
Status	If queried vehicle information is manipulated, a police officer cannot get valid information or a criminal can avoid law enforcement.	AID-02
Registration Expiration	No damage	-
Owner Name	Personally identifiable information (PII) may be leaked.	AID-03
Owner Date of Birth	Personally identifiable information (PII) may be leaked.	AID-04
Vehicle Year of Manufacture	No damage	-
Vehicle Make	No damage	-
Vehicle Model	No damage	-
Vehicle Color	No damage	-
Owner Street Address/ Location	Personally identifiable information (PII) may be leaked.	AID-05
Owner City, State and Zip Code	No damage	-
ID	Authentication information can be leaked and an abuser can access the system maliciously.	AID-06
Password	Authentication information can be leaked and an abuser can access the system maliciously.	AID-07
Configuration File	If a configuration file is manipulated, a police officer cannot get valid information or a criminal can avoid law enforcement.	AID-08
Log file	If a log file is manipulated, the server cannot provide access and query information, so the system cannot avoid non-repudiation.	AID-09

Table 2. Asset List

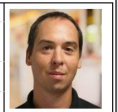
2 Threat Modeling

2) Disgruntled Employees

Type	**PID-02:** Disgruntled Employee
Motivations	Dan is a retired police officer. He wants to know about a luxury car in the neighborhood where he lives and wants to know information about this car and the owner's information. He knew that his credential is not expired, and he can access the Tartan ALPR system.
Goals	To retrieve specific vehicle information (include
Skills	
Misuse Ca	

3) Hackers

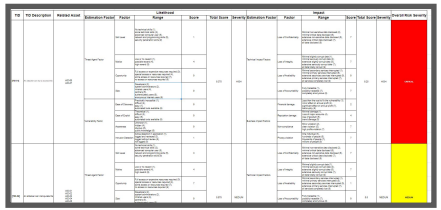
Type	**PID-03:** Hacker
Motivations	He is a champion of '2022 world CTF'. And Tartan requested him pentesting against their new ALPR system.
Goals	To accomplish personal data without detection To cause discomfort to connect the ALPR system.
Skills	He has good Code and Hacking skills He knows whole ALPR System architectures
Misuse Cases	1. Steal a mobile device from a policeman and disable alerts. 2. Intercept alert messages and change the message to a normal message. 3. Intercept response messages and restore another place and send the abnormal messages to a police officer.



Jeffrey Gennari

	Threat	Property Violated	Threat Definition
S	Spoofing identify	Authentication	Pretending to be something or someone other than yourself
T	Tampering with data	Integrity	Modifying something on disk, network, memory, or elsewhere
R	Repudiation	Non-repudiation	Claiming that you didn't do something or were not responsible; can be honest or false
I	Information disclosure	Confidentiality	Providing information to someone not authorized to access it
D	Denial of service	Availability	Exhausting resources needed to provide service
E	Elevation of privilege	Authorization	Allowing someone to do something they are not authorized to do

3 Risk Assessment



Step 4: Determining the Severity of the Risk

In this step, the likelihood estimate and the impact estimate are put together to calculate an overall severity for this risk. This is done by figuring out whether the likelihood is low, medium, or high and then do the same for impact. The 0 to 9 scale is split into three parts:

Likelihood and Impact Levels	
0 to <3	LOW

Determining Severity

However the tester arrives at the likelihood and impact estimates, they can now combine them to get a final severity rating for this risk. Note that if they have good business impact information, they should use that instead of the technical impact information. But if they have no information about the business, then technical impact is the next best thing.

Overall Risk Severity

Impact				
HIGH	Medium	High	Critical	
MEDIUM	Low	Medium	High	
LOW	Note	Low	Medium	
	LOW	MEDIUM	HIGH	
Likelihood				

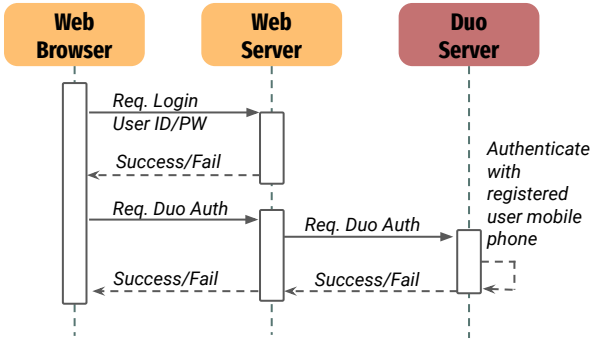
PnG, STRIDE

OWASP

Main Security Design

Web Client

2-Factor Authentication

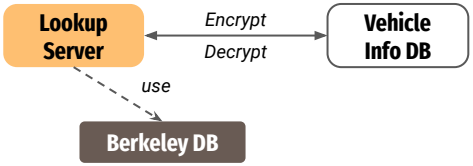


Key:

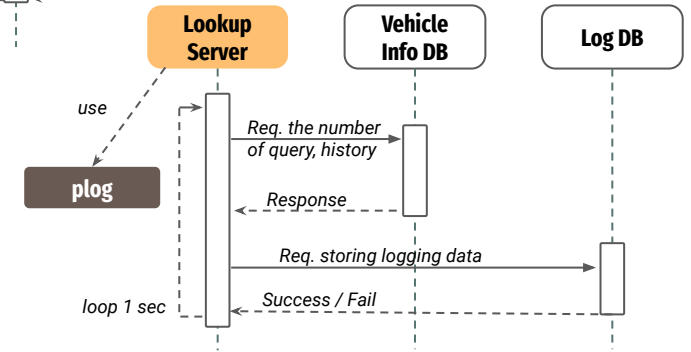
- Device
- External Device
- Database
- Open Source Library
- Certificate (X.509)

Lookup Server

Database Encryption

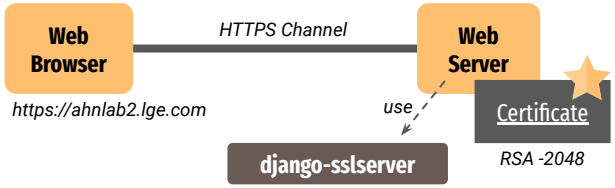


Data Logging

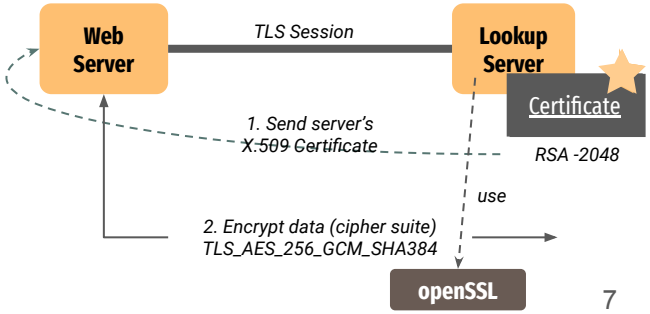


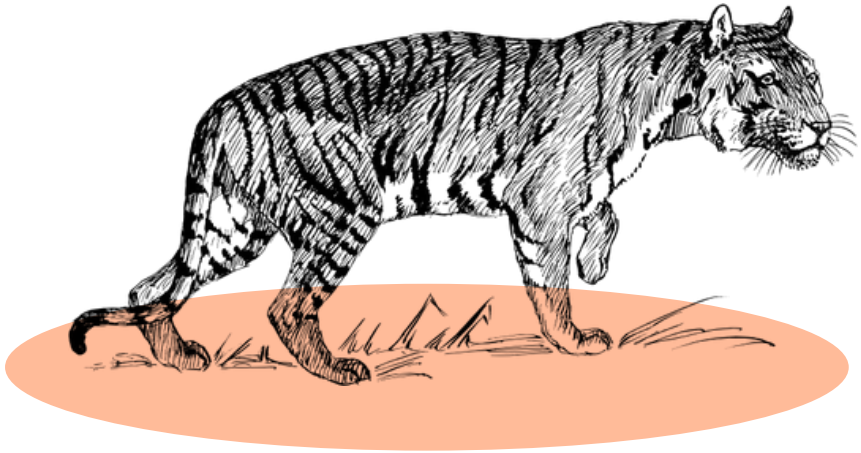
Secure Channel

HTTPS communication



TLS v1.3 communication





Phase 2
LG Security Specialist Evaluation
for Tiger Project (Team 1)

R&R Reorganization for Phase 2

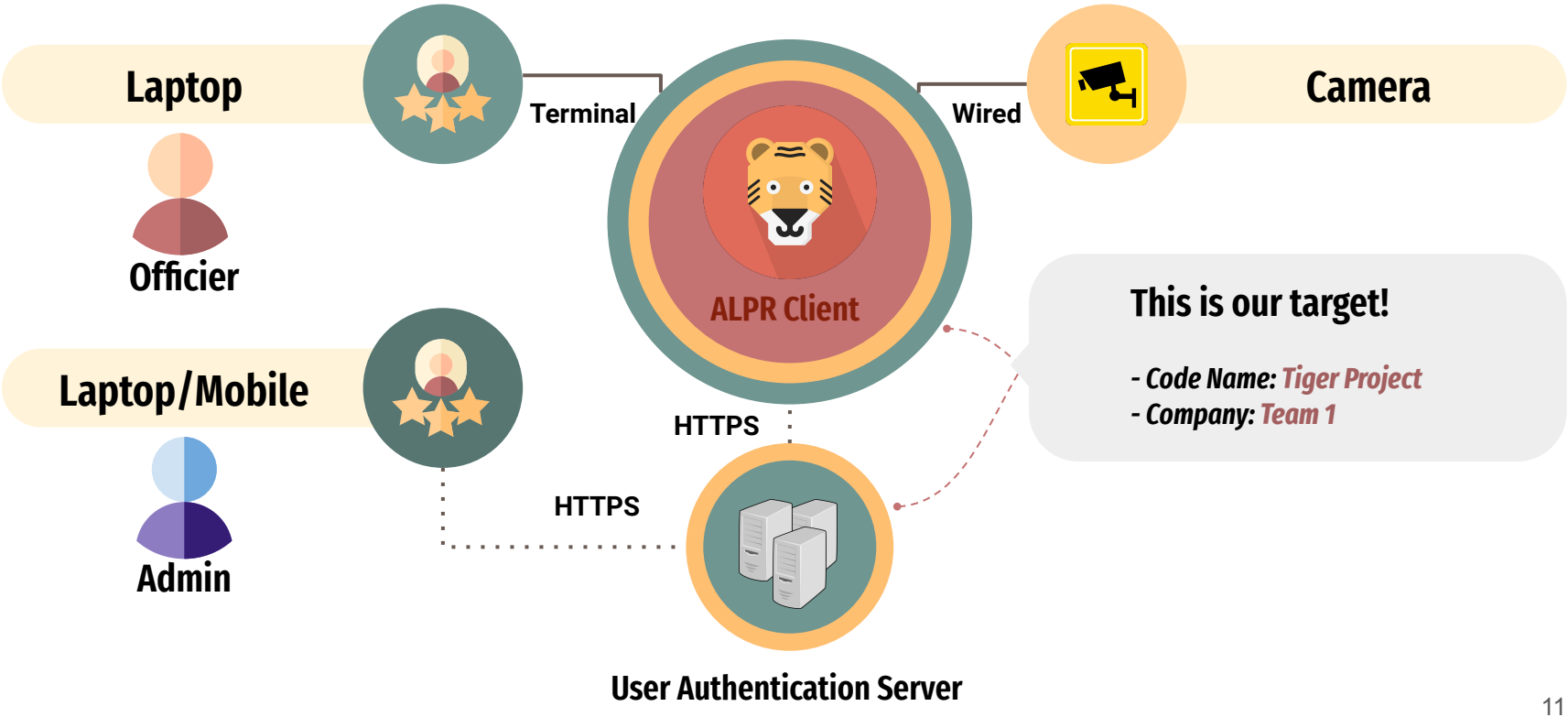
Members		Role & Responsibility
Woosuk Ahn	Web Backend	*ALPR Web server (Node.js)*
Heewon Ahn	Authentication	*Authentication for web server, “lgdemo”*
Woosuk Jung	ALPR Client	*Analysis of the “ALPR Client” program, database*
Hyunmin Kim	ALPR Client	*Analysis of the “ALPR Client” program, database*
Hunjin Ha	Secure Channel	*Analysis secure communications (HTTPS, TLS)*
Jungsun Goh	Documentation	*Static analysis(Coverity), and researching fuzzing tools*

Evaluation Plan

P1	*Jul. 6 - Jul. 7*
P2	*Jul. 8 - Jul. 9*
P3	*Jul. 12 - Jul. 14*

	P1	P2	P3	
Install Product	Install Docker, Node.js in our analysis environment			***2 days***
Static Analysis	Analyze using Coverity for C++, Node.js(Javascript)			***2 days***
Code Review	Review all source code and system configurations			***4 days***
Pen Testing	Analyze using Kali linux tool, and fuzzing test tool for JPEG			***4 days***
Documentation	Make a report and presentation			***3 days***

Overview of Target



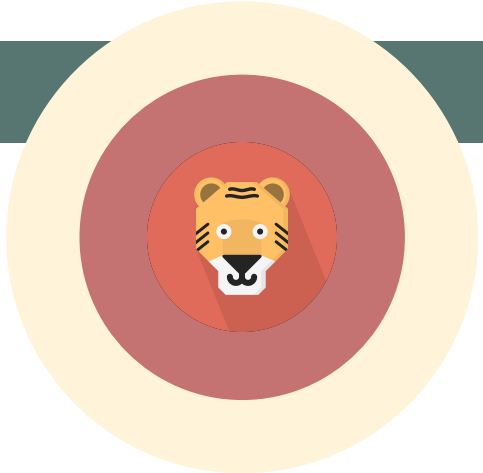
Assets of Target

	Assets	
1	User Information (ID / Password / Extra Information)	Protected
2	Recognized plate number on client	Not Protected
3	Plate number for query server	Not Protected
4	Private key and certificate for TLS	Not Protected
5	Vehicle information sent by server	Protected
6	User and Vehicle Information DB	Protected
7	Output Video file in client	Protected
8	Keys for encryption	Protected

Evaluation Method

Code Review

- Web server
- ALPR Client
- System Configurations



Static Analysis

- Coverity 2022.03
- CERT-C++ Coding Standard
- CWE Top25
- OWASP Top10

Penetration & Fuzz Testing

Kali Linux
Pen Testing OS Toolkit

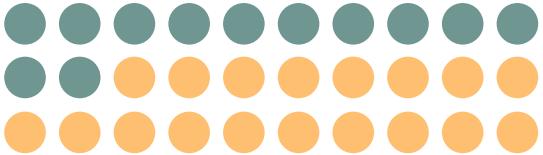
VEGA
Web Security Scanner

NMAP
Port Scanner

SQLMAP
SQL Injection Scanner

Evaluation Result

Web Interfaces Code Review



Require Login	6 / 13
Not Require Login	7 / 13

Static Analysis

Web Server Flaws	11
ALPR Client Flaws	10

VEGA

High(5) Low(2) Info(2)

Report: *"Session Cookie Without Secure Flag"*
→ Consider that attack using Session Cookie

NMAP

```
PORT     STATE SERVICE
80/tcp   open  http
139/tcp  open  netbios-ssn
443/tcp  open  https
445/tcp  open  microsoft-ds
6063/tcp open  x11
9310/tcp open  sapms
9922/tcp open  unknown
```

Rule of Coverity:
MISSING_SAMESITE_ATTRIBUTE_SESSION_COOKIE_EXPRESS

It detects that there are CSRF vulnerability in the system.

Found Vulnerabilities

	Summary	Impact	Method
1	*An attacker can change a user's password by exploiting a bug in which some functions do not authenticate.*	EoP	Code Review
2	*An attacker can send "Verification Code" to the any email address infinitely as an admin*	EoP	Code Review
3	*An attacker can bypass required administrator approval to activate a new account by using a bug.*	EoP	Code Review
4	*To stop the server, attackers can send a lot of request messages for server to trigger email authentication.*	DoS	Code Review
5	*An attacker can send messages to the server rapidly, the server cannot service at that moment.*	DoS	Code Review
6	*An attacker can access the server via SSH (port 9922) and remove database file.*	Tampering	NMAP
7	*An attacker can obtain an unprotected client key and receive information about the plate number from the server.*	Spoofing	Static Analysis
8	*An attacker can retrieve license plate number by intercepting an authentication token stored in a cookie*	Spoofing	Static Analysis VEGA

Elevation of Privilege (1/3)

Elevation of Privilege

CVSS 5.2

An attacker can change a user's password by exploiting a bug where some functions do not authenticate

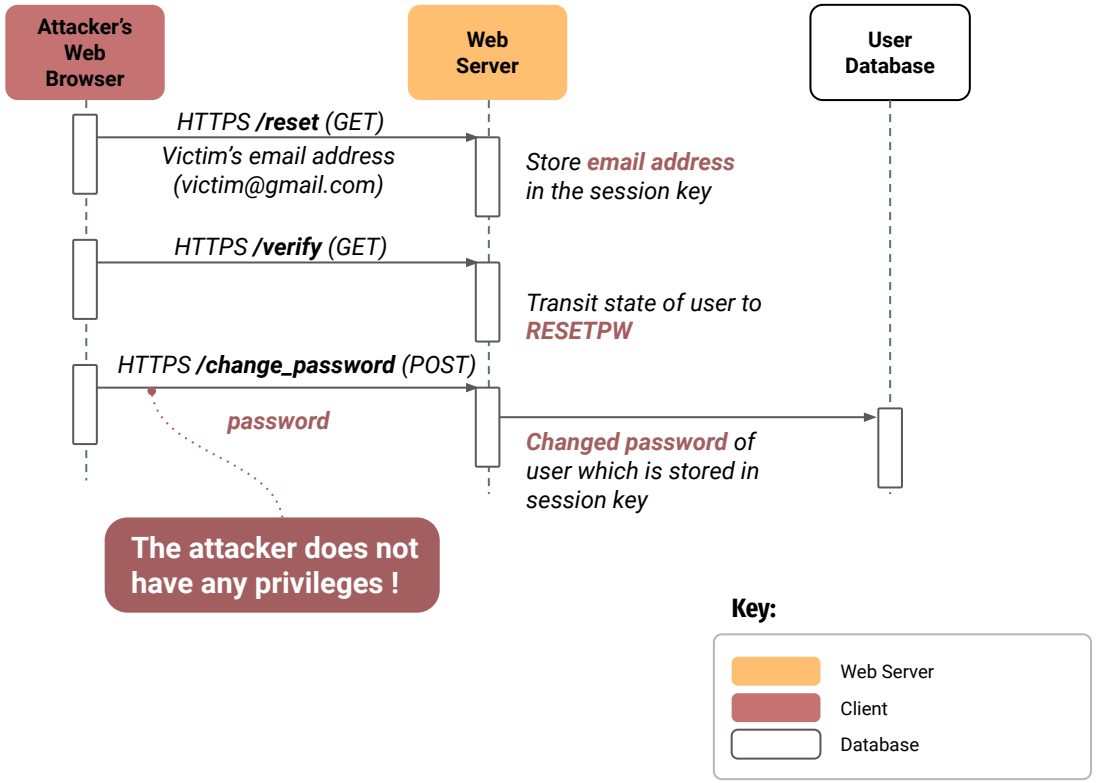
⚠ Some functions have a bug where it is bypassed when the "verifycode" and "answer" of more than 30 characters is transmitted.

Sequence of Attack

1. **Send /reset (HTTPS GET)**
2. **Send /verify (HTTPS GET)**
3. **Send /change_password (HTTPS POST)**

Mitigation

Functions should authenticate user



Elevation of Privilege (2/3)

Elevation of Privilege

CVSS 4.5

An attacker can send "Verification Code" to the any email address infinitely as an admin

⚠ Register function does not have authentication mechanism.

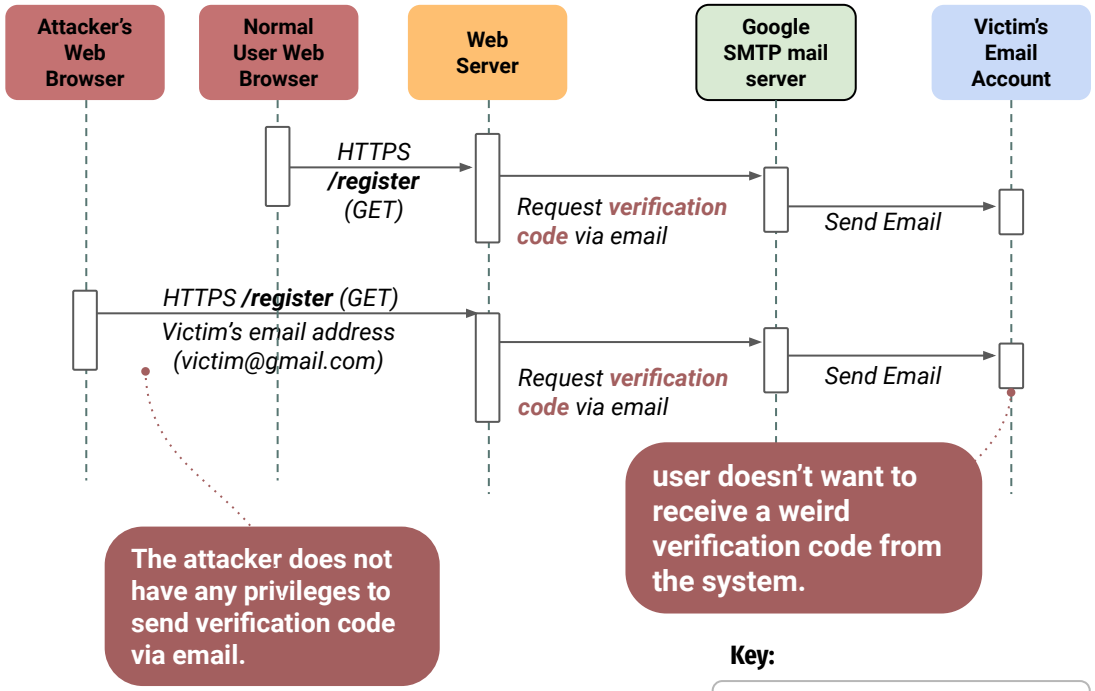
Sequence of Attack

Precondition: A user who wants to register a new account resisters an email address and get verification code

1. Send /register (HTTPS GET)

Mitigation

Functions should authenticate user



Key:

- Web Server
- Client
- External Device
- Victim

Consequence

The user cannot distinguish the real verification code.

Elevation of Privilege (3/3)

Elevation of Privilege

CVSS 6.8

An attacker can bypass required administrator approval to activate a new account

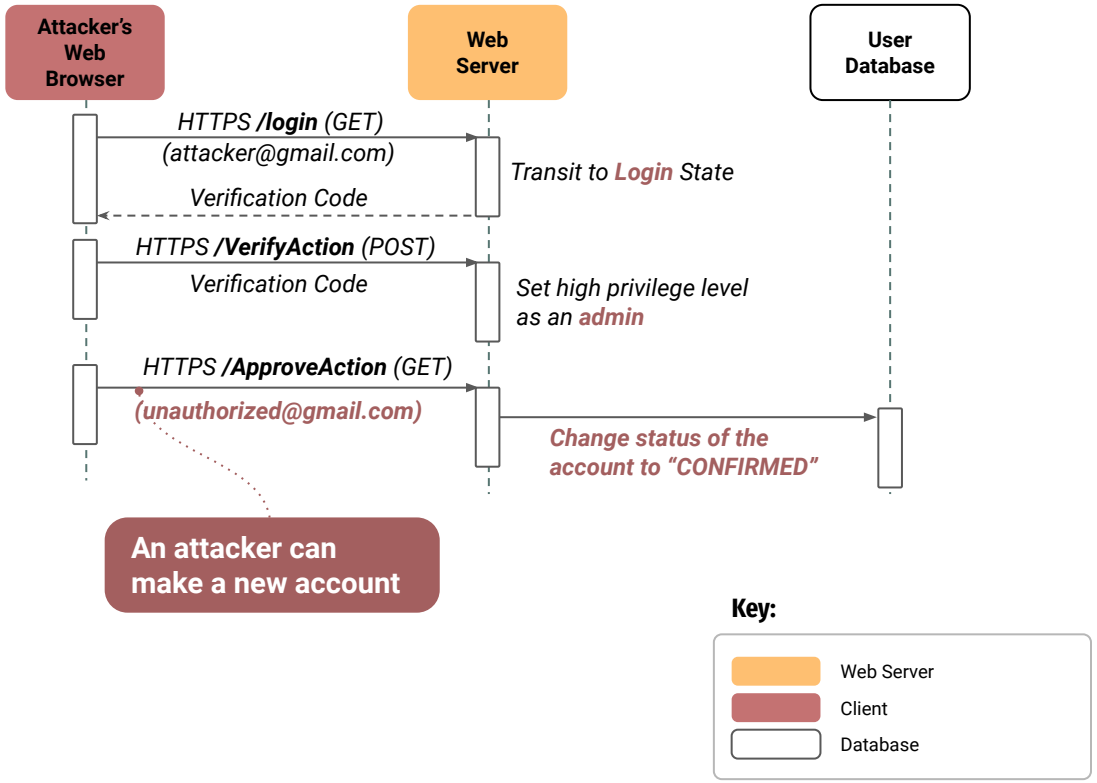
- Some functions give a level of admin without any authentication
- The system should give high privilege level once the admin is logged on

Sequence of Attack

1. Send /login (HTTPS GET)
2. Send /VerifyAction (HTTPS POST)
3. Send /ApproveAction (HTTPS GET)

Mitigation

Functions should check whether the session is admin level or not



Denial of Service (1/2)

Denial of Service

CVSS 6.0

To stop the server, an attacker can send a lot of request messages for server to trigger email authentication

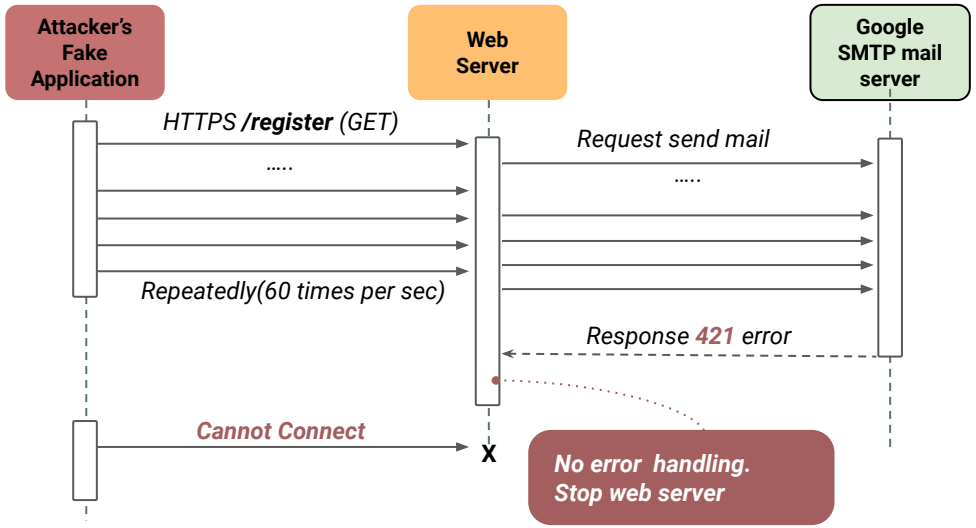
Some functions do not error handling(not using callback function) at google SMTP server error

Sequence of Attack

1. Make a fake application to send messages
2. Send a lot of messages to web server

Mitigation

Add to callback function to catch an exception



At util.js line 82,

```
82:transporter.sendMail(mailoptions);
```

```
82:transporter.sendMail(mailoptions,
function(err, info){
if(err){console.error(err);}});
```

Key:

- Web Server
- Client
- External Module

Denial of Service (2/2)

Denial of Service

CVSS 3.4

The resources of the web server rapidly increase due to sending a large number of HTTP request messages rapidly

Node.js supports single thread, if a lot of HTTP requests are received, resources (CPU, RAM) usage increase rapidly. So the web server cannot service at that moment.

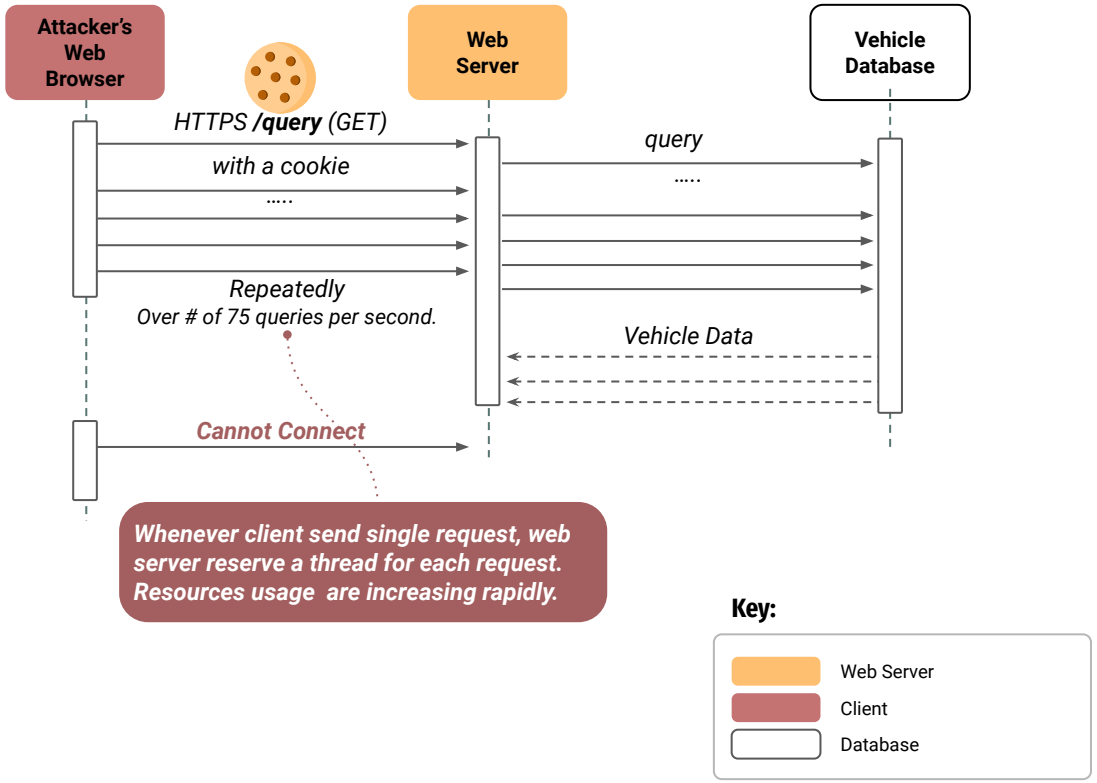
Sequence of Attack

Assume that an attacker get a user's cookie

1. Send /query (HTTP GET) with session ID repeatedly and rapidly.

Mitigation

Web Server should check a request rate, and it should be able to drop the packet in peak load



Tampering

Tampering

CVSS 6.8

An attacker can access the server via SSH (port 9922) and remove database file.

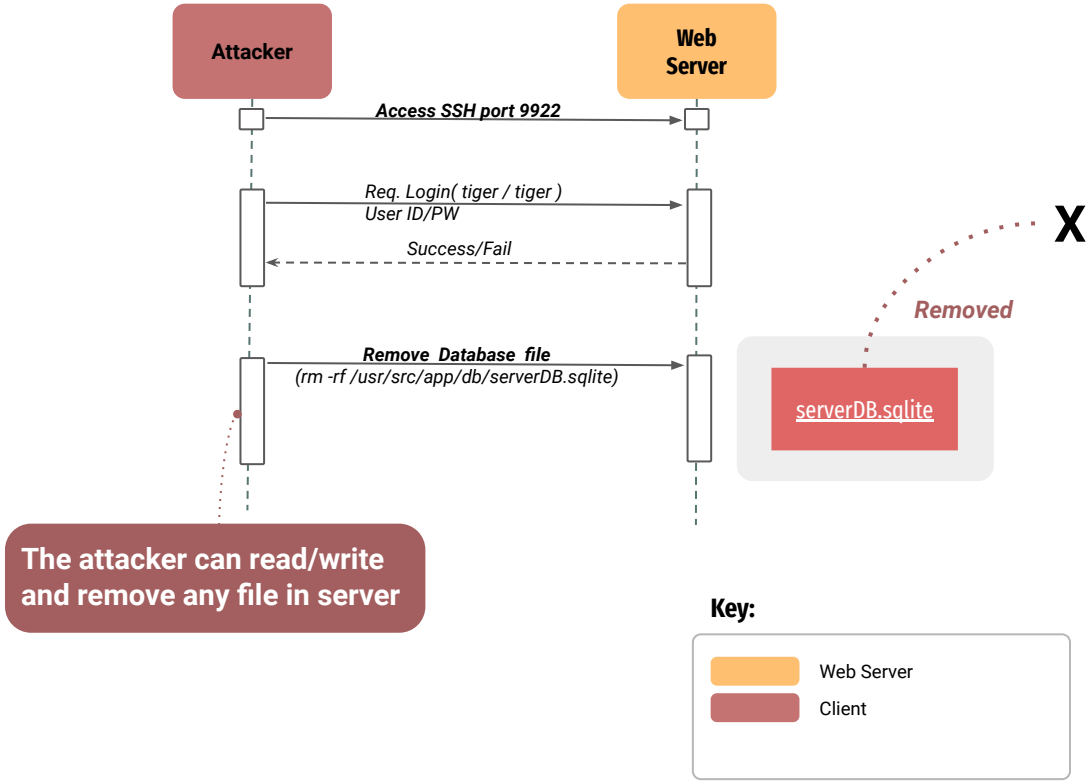
Backdoor ports can be used by Attackers

Sequence of Attack

1. **Access Server by SSH Port 9922**
2. **Login Admin ID & Password (tiger/tiger)**
3. **Remove Database file**
   (File : /usr/src/app/db/serverDB.sqlite)
4. **Server cannot register new users**

Mitigation

1. Backdoor port is disabled for Production S/W
2. the security strength of the password should be higher.



Spoofing (1/2)

Spoofing

CVSS 6.0

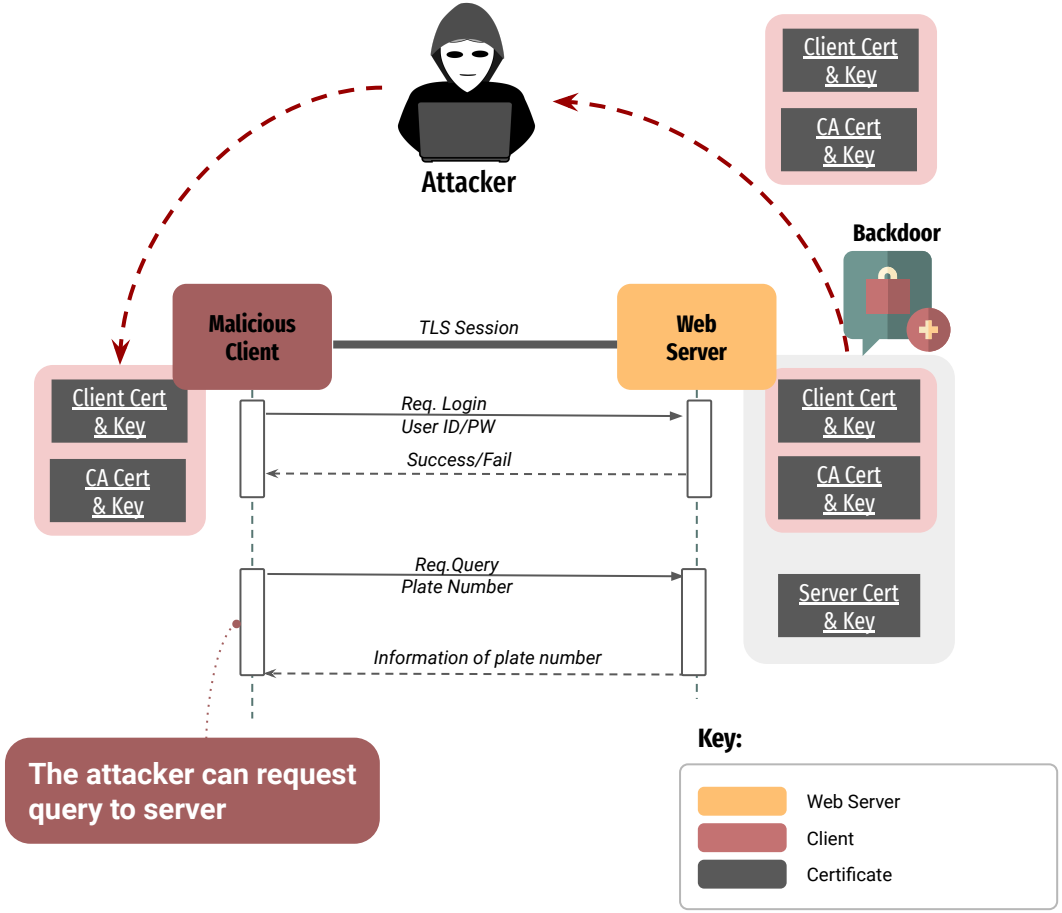
An attacker can connect a malicious client to the server with an unprotected client key and certificate

Sequence of Attack

1. **Get client key & cert through backdoor port**
   (via SSH port 9922 : Vulnerability#6)
2. **Implement malicious client code & execute**
3. **Connect to server**
4. **Login**
5. **Request information that attacker wants.**

Mitigation

A private key should be stored in hardware-based protection, such as a Hardware Security Module (HSM).



Spoofing (2/2)

Spoofing

CVSS 5.7

An attacker can retrieve license plate number without any authentication using cookies

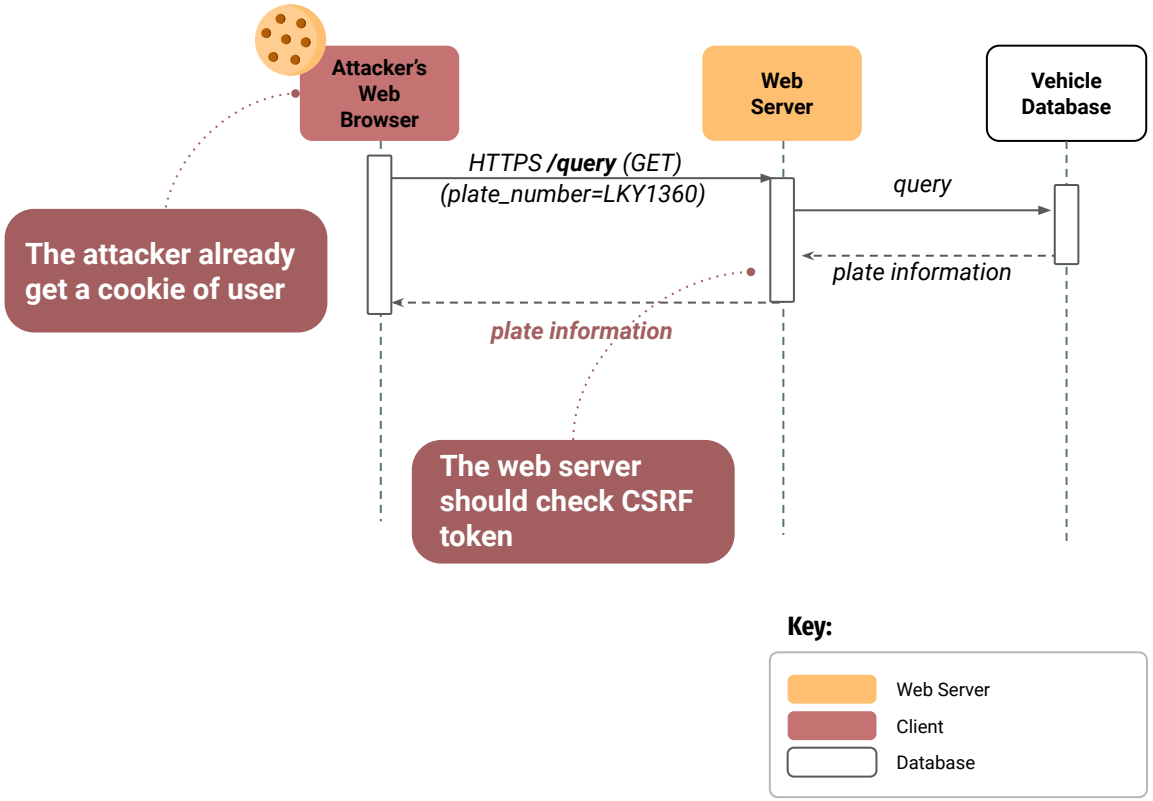
- The system doesn't have any protection mechanism of CSRF
- If an attacker get a cookie of victim, he can spoof as a general user.

Sequence of Attack

1. Send /query (HTTP GET) with cookie

Mitigation

This system should check CSRF_Token to prevent using cookie.



Lessons and Learned

Overall

This course addressed an introduction of many tools, but it was ***not easy to find and apply proper and effective tools*** for the specific project due to language limitations, lack of time and background knowledge. Additionally, tools were definitely ***helpful for identifying vulnerabilities but alerted many false alarms*** especially in case of static code analysis and software composition analysis tool.

As a result, we learned there were no substitute tools for an ***experienced engineer*** and it is necessary to apply both tools and code review. If given the opportunity to do this project again, we will apply ***both more effective tools and code review systematically at the earlier stage*** for the phase 1 and 2.