정적 분석 솔루션 개발

김수현, 김건희, 남지우

CONTENTS

01. Intro

- 팀 소개
- 프로젝트 주제
- 프로젝트 개요
- 중간발표 리마인드

02. Project

- 정적 분석
- 서비스 아키텍처
- 배포 (도커 이미지, 깃허브)
- 서비스 시연 영상

03. Outro

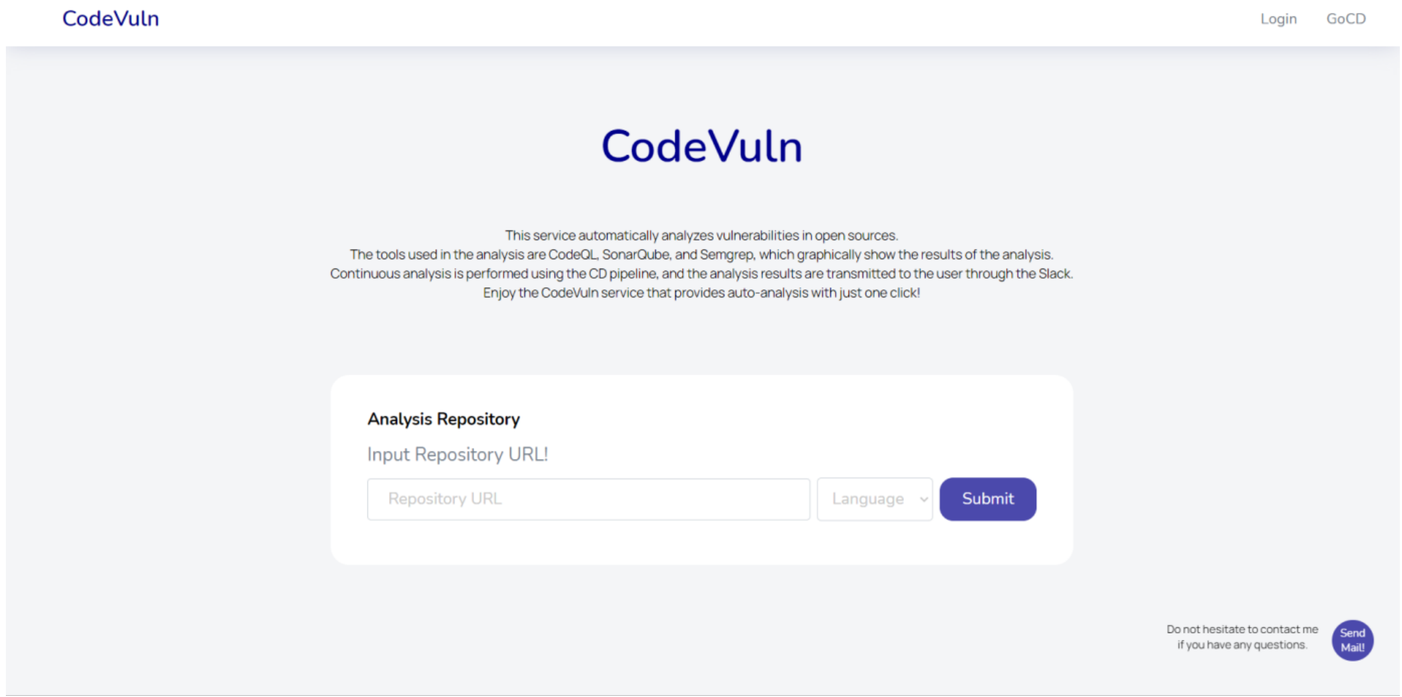
- 취약점 분석
- 산출물
- 프로젝트를 마치며
- 질문 & 답변

Final Report

Intro

1. 팀소개
2. 프로젝트 주제
3. 프로젝트 개요

codevuln.my.canva.site



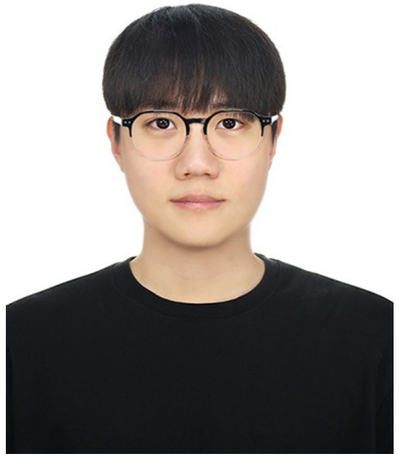
CODEVULN

CODE VULNERABILITY의 약자로,
사용자의 코드를 정적분석 해주는 의미를 담았습니다.

CODEVULN 팀 소개_팀원 소개

김수현

Project Manager



김건희

Project Agent

남지우

Project Agent

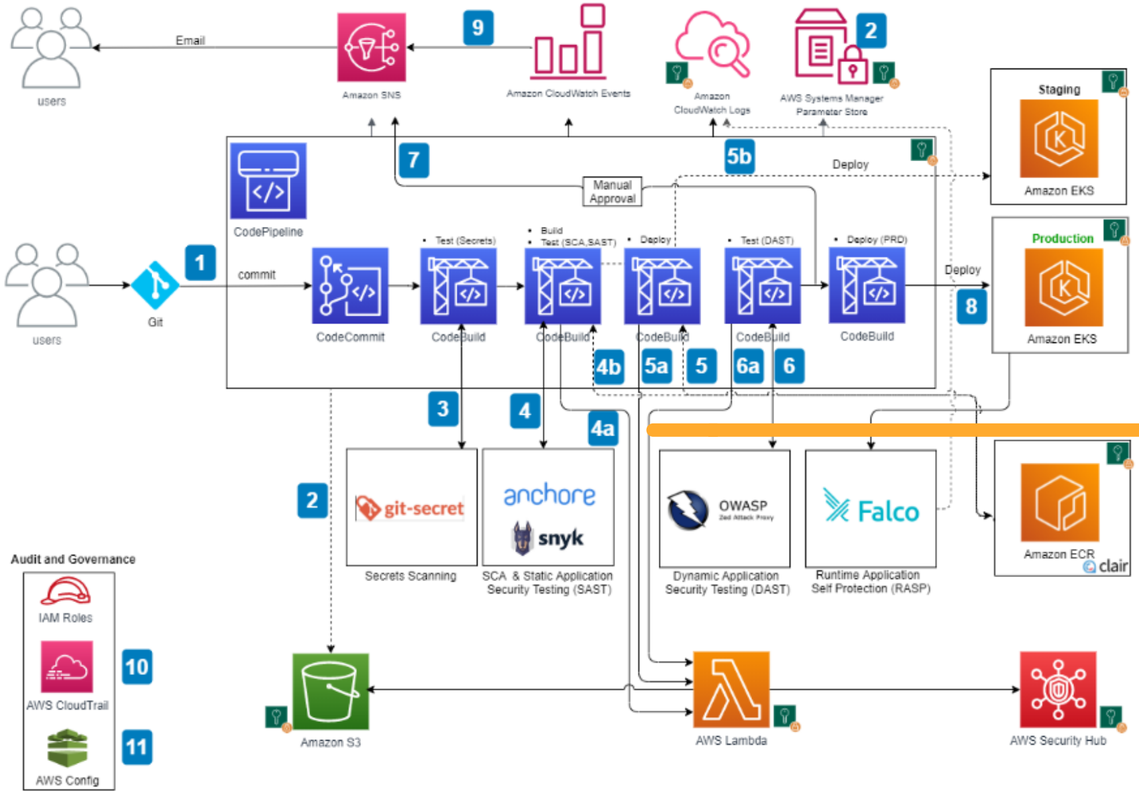
김평안

Co-Work

프로젝트 주제 : 정적 분석 솔루션 개발

정적 분석 플랫폼을 통합하여 하나의 솔루션을 구축합니다.

프로젝트 주제_주제 선정 배경



Kubernetes DevSecOps Pipeline Architecture

DevSecOps

DevSecOps 보안 피처로 사용되는 오픈소스를 통합하여 하나의 서비스를 개발 합니다.

CodeQL

SonarQube

Semgrep

세부 수행

1. 정적분석 프로그램 통합

DevSecOps 보안 피처로 사용되는 오픈소스를 통합하여 하나의 서비스를 개발 합니다.

codeQL

SonarQube

Semgrep

2. DB & WEB

솔루션을 데이터베이스와 웹 인터페이스를 통해 제공되어, 사용자가 효율적으로 접근하고 사용할 수 있습니다.

SQLite

HTML

FLASK

세부 수행

3. GoCD

GoCD 파이프라인을 설정하여 GitHub에서 코드가 머지될 때마다 자동으로 솔루션이 실행됩니다.

GoCD

4. SNS

실행 결과는 슬랙 및 다양한 SNS 채널을 통해 알림으로 제공됩니다.

SLACK

Final Report

Project

1. 정적 분석
2. 서비스 아키텍처
3. 배포 (도커 이미지, 깃허브)
4. 서비스 시연 영상

정적분석, SAST

Static Application Security Testing

소프트웨어의 소스 코드, 바이트 코드 또는 바이너리 코드를
분석하여 보안 취약점을 찾아내는 정적 분석 방법

AST

Abstract Syntax Tree

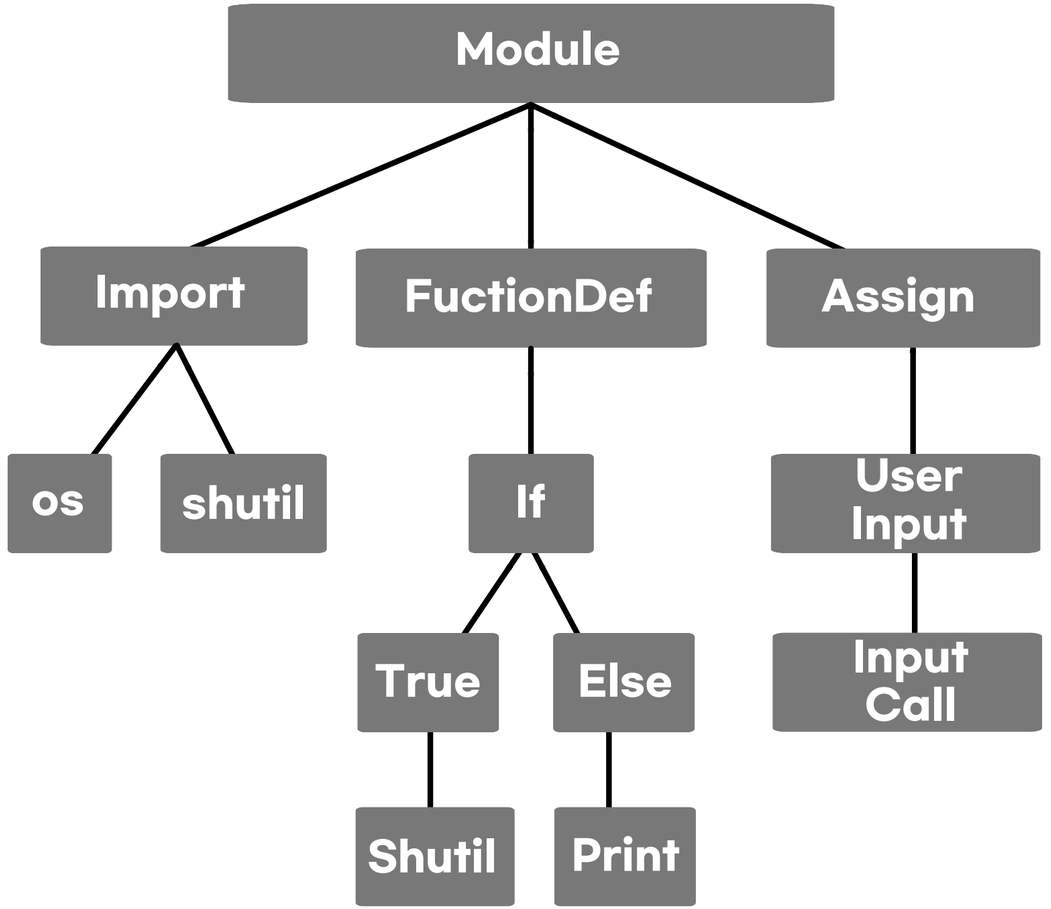
프로그래밍 언어의 소스 코드 구문을 표현하는 계층적인 트리 구조

정적분석, SAST_AST

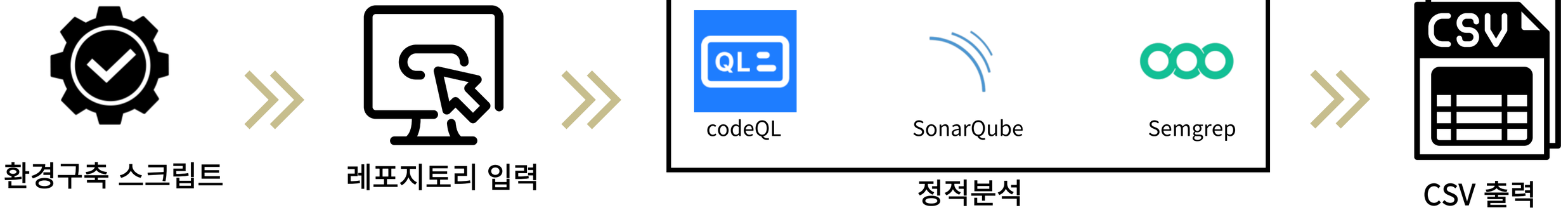
```python
import os
import shutil

def delete_file_safe(filename):
    if os.path.isfile(filename):
        shutil.rmtree(filename, ignore_errors=True)
    else:
        print("File does not exist")

user_input = input("Enter the filename to delete: ")
delete_file_safe(user_input)
```



서비스 아키텍처 - CLI



서비스 아키텍처 - CSV Colunm

codeQL

tool	date	time	name	explanation	severity	message	path	start_line	start_col	end_line	end_col
CodeQL	2024-05-07	13:12:49	Unsafe jQuery plugin	A jQuery plugin th	warning	Potential	/js/jquery.	626	8	626	36
CodeQL	2024-05-07	13:12:49	Unsafe jQuery plugin	A jQuery plugin th	warning	Potential	/js/jquery.	634	24	634	50
CodeQL	2024-05-07	13:12:49	Unsafe jQuery plugin	A jQuery plugin th	warning	Potential	/js/jquery.	651	7	651	34
CodeQL	2024-05-07	13:12:49	Unsafe jQuery plugin	A jQuery plugin th	warning	Potential	/js/jquery.	655	7	655	34
CodeQL	2024-05-07	13:12:49	Unsafe jQuery plugin	A jQuery plugin th	warning	Potential	/js/jquery.	688	7	688	42

```
624 // if a pager selector was supplied, populate it with the pager
625 if(slider.settings.pagerSelector){
626     $(slider.settings.pagerSelector).html(slider.pagerEl);
```

서비스 아키텍처 - CSV Colunm

Semgrep

tool	date	time	severity	path	start_line	end_line	start_col	end_col	message	rule_id	lines	metadata
Semgrep	2024-05-07	13:11:42	WARNING	/home/co	352	352	14	27	Using use	php.lang.s	@	{'category': 'security', 'co
Semgrep	2024-05-07	13:11:42	WARNING	/home/co	472	472	26	43	Using use	php.lang.s		{'category': 'security', 'co

Impact

이슈의 잠재적 영향

likelihood

이슈가 발생할 가능성

confidence

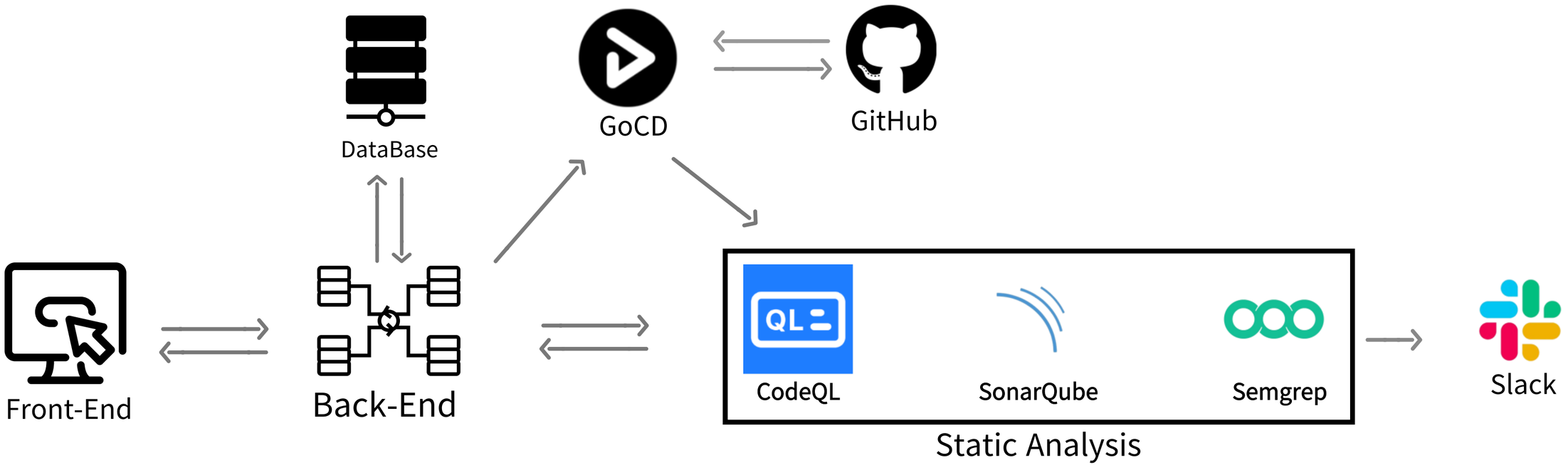
이슈 신뢰도

서비스 아키텍처 - CSV Colunm

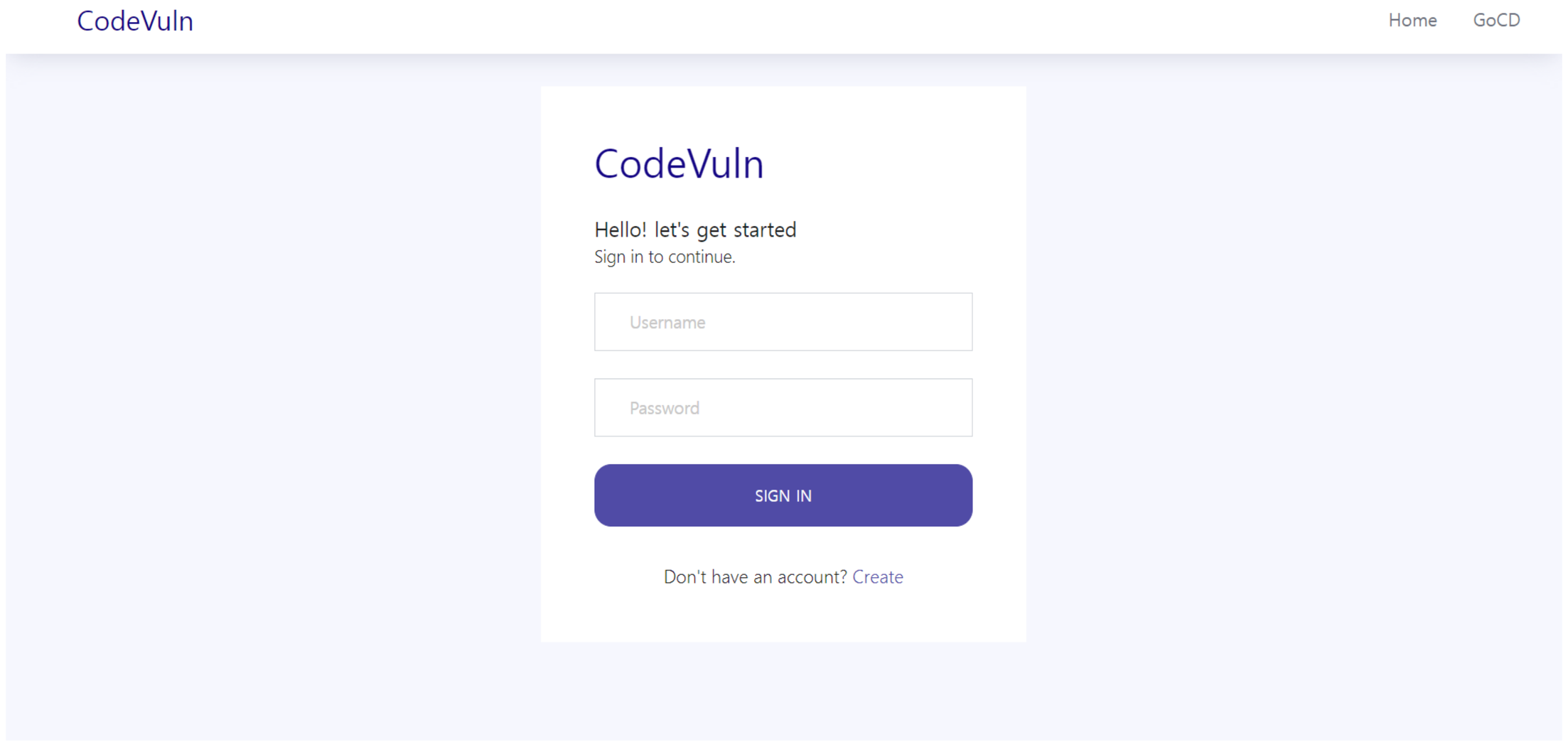
SonarQube

tool	date	time	rule	severity	message	path	start_line	start_colur	end_line	end_column
SonarQub	2024-05-08	18:26:47	javascript:S2703	BLOCKER	Add the "l	gnuboard5	5	9	5	10
SonarQub	2024-05-08	18:26:47	javascript:S2703	BLOCKER	Add the "l	gnuboard5	14	8	14	11
SonarQub	2024-05-08	18:26:47	php:S6600	CRITICAL	Remove th	gnuboard5	2	0	2	28
SonarQub	2024-05-08	18:26:47	php:S121	CRITICAL	Add curly	gnuboard5	6	4	6	6
SonarQub	2024-05-08	18:26:47	php:S115	CRITICAL	Rename th	gnuboard5	9	11	9	19

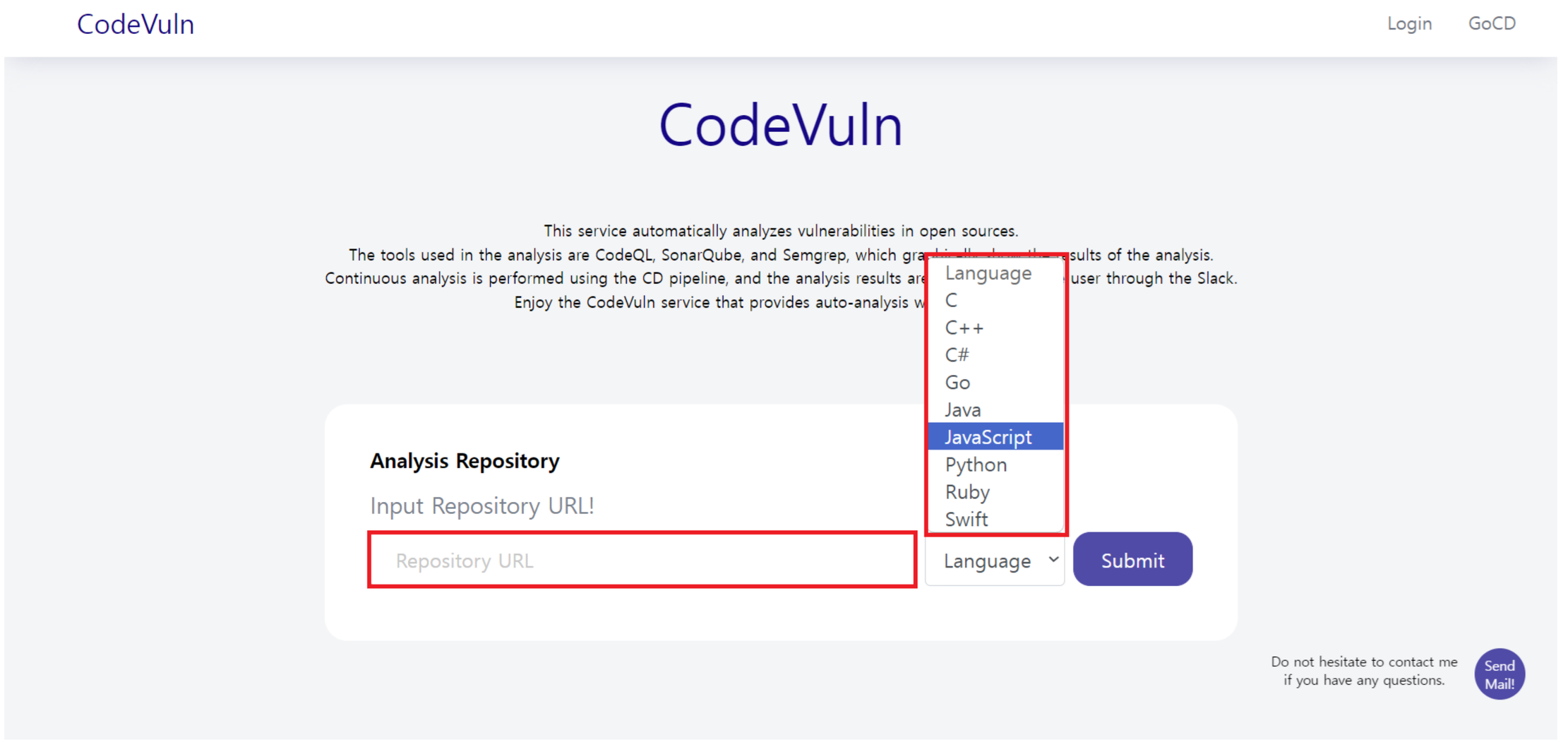
서비스 아키텍처 - WEB


DataBase
GoCD
GitHub
Front-End
Back-End
CodeQL
SonarQube
Semgrep
Static Analysis
Slack

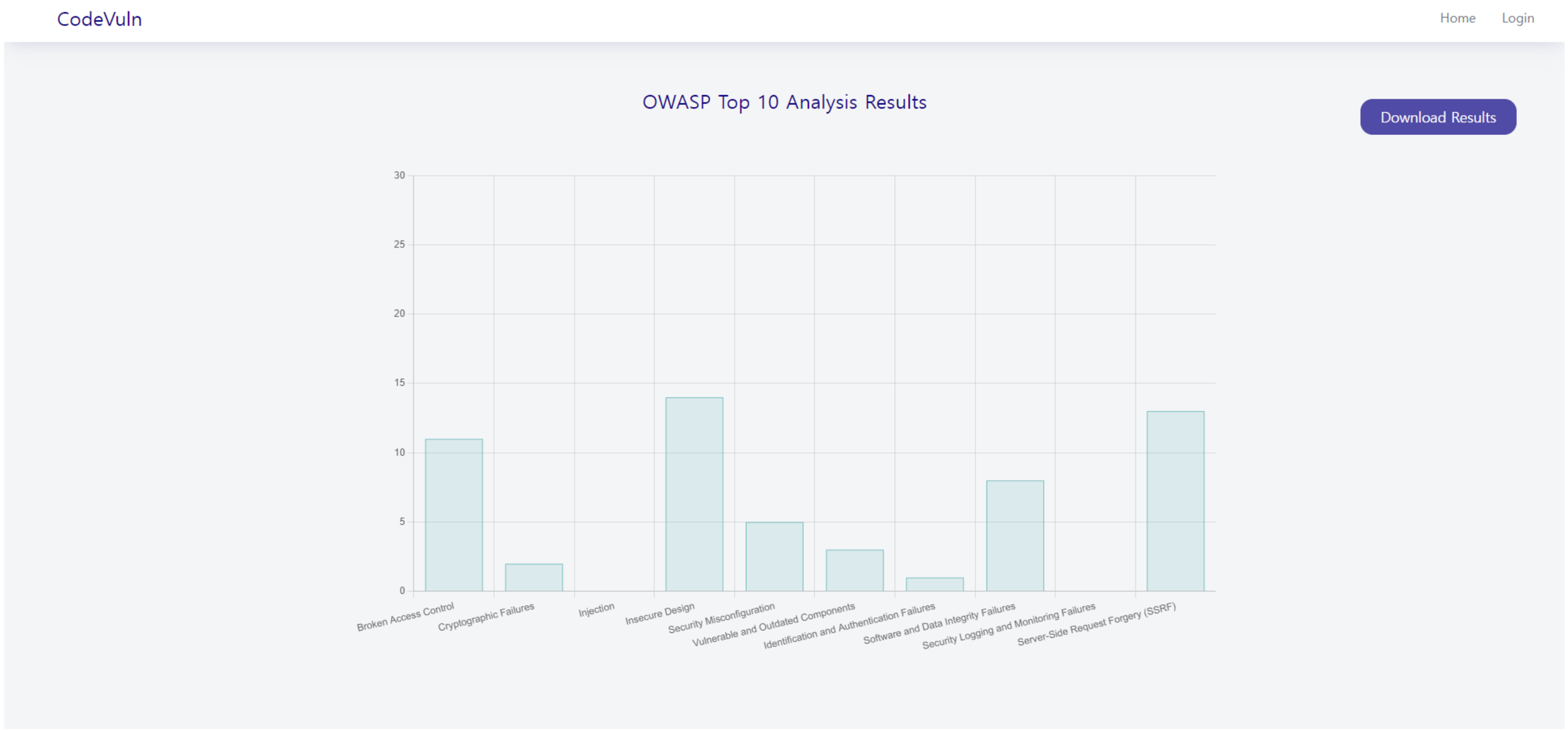
서비스 아키텍처 - Login



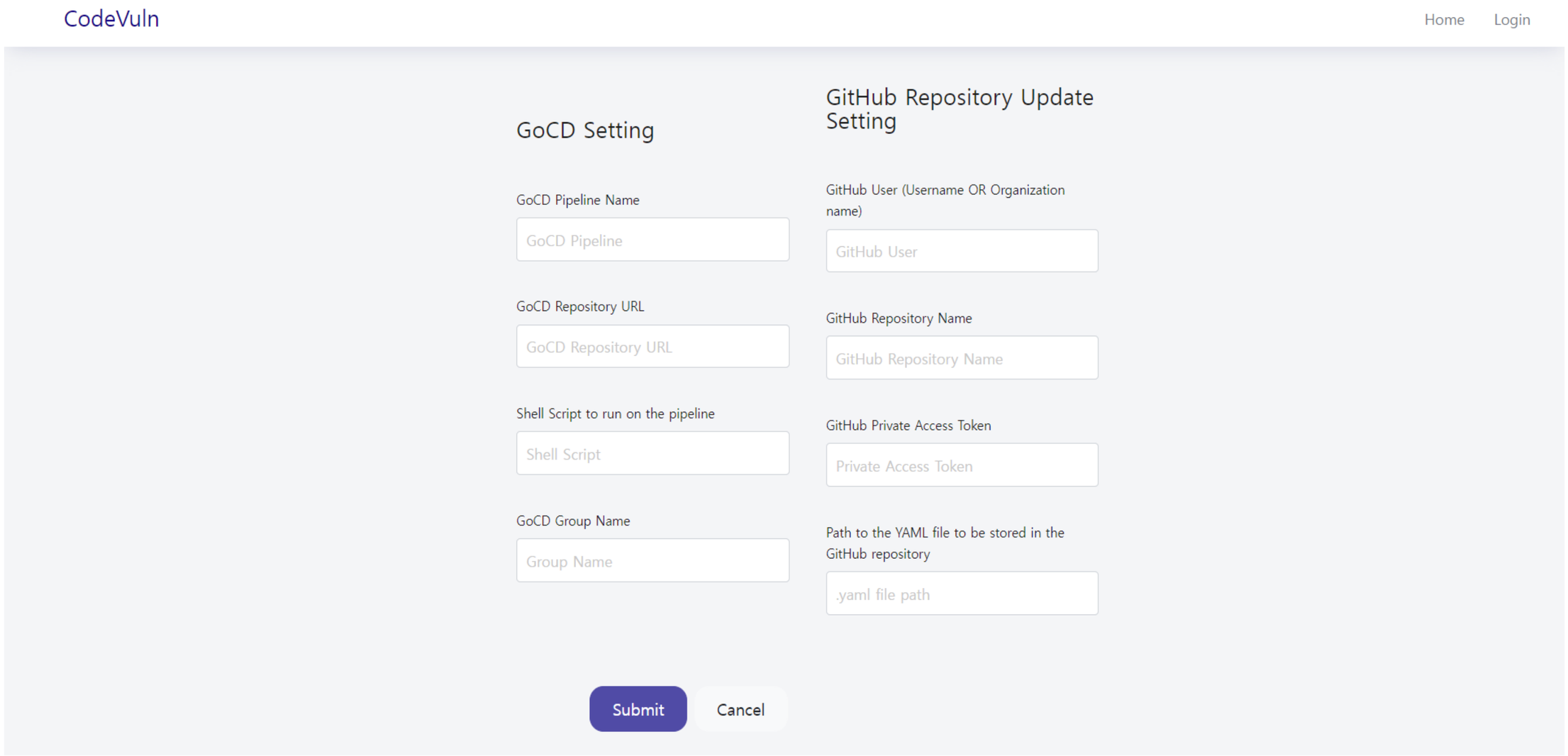
서비스 아키텍처 - Input github link & language



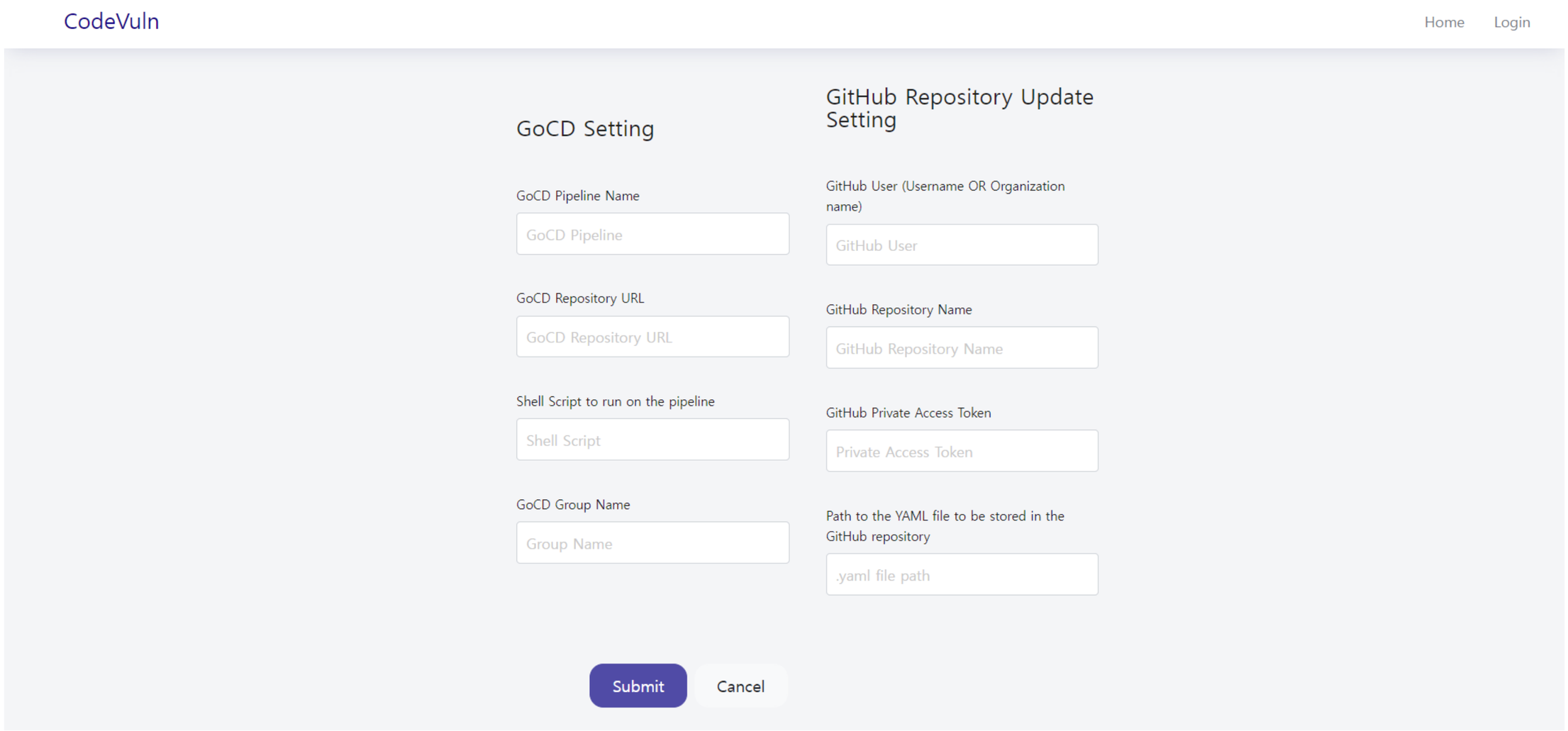
서비스 아키텍처 - Result



서비스 아키텍처 - gocd



서비스 아키텍처 - 시연영상

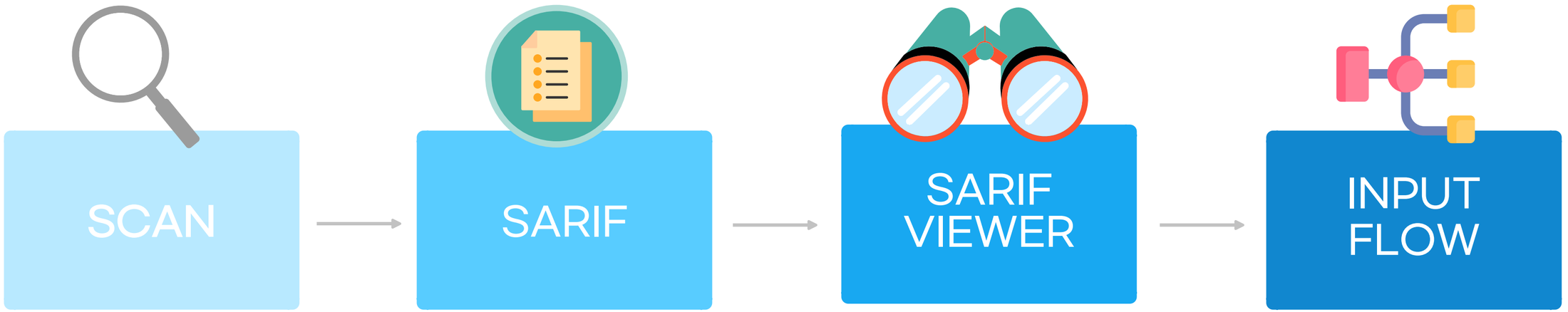


Final Report

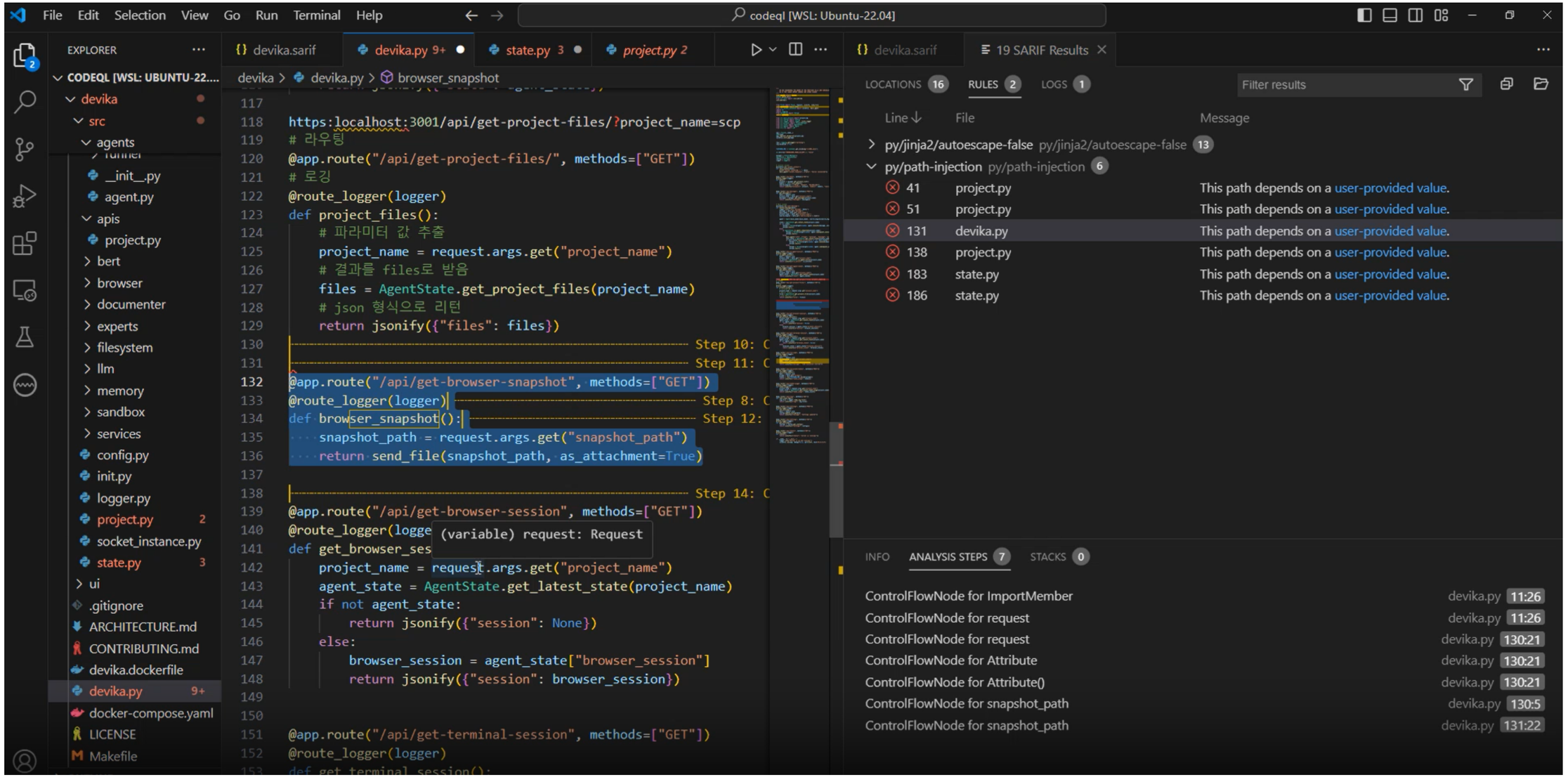
Outro

1. 취약점 분석
2. 산출물
3. 프로젝트를 마치며
4. 질문 & 답변

오픈소스 취약점 분석 - FLOW



오픈소스 취약점 분석 - Sarif Viewer

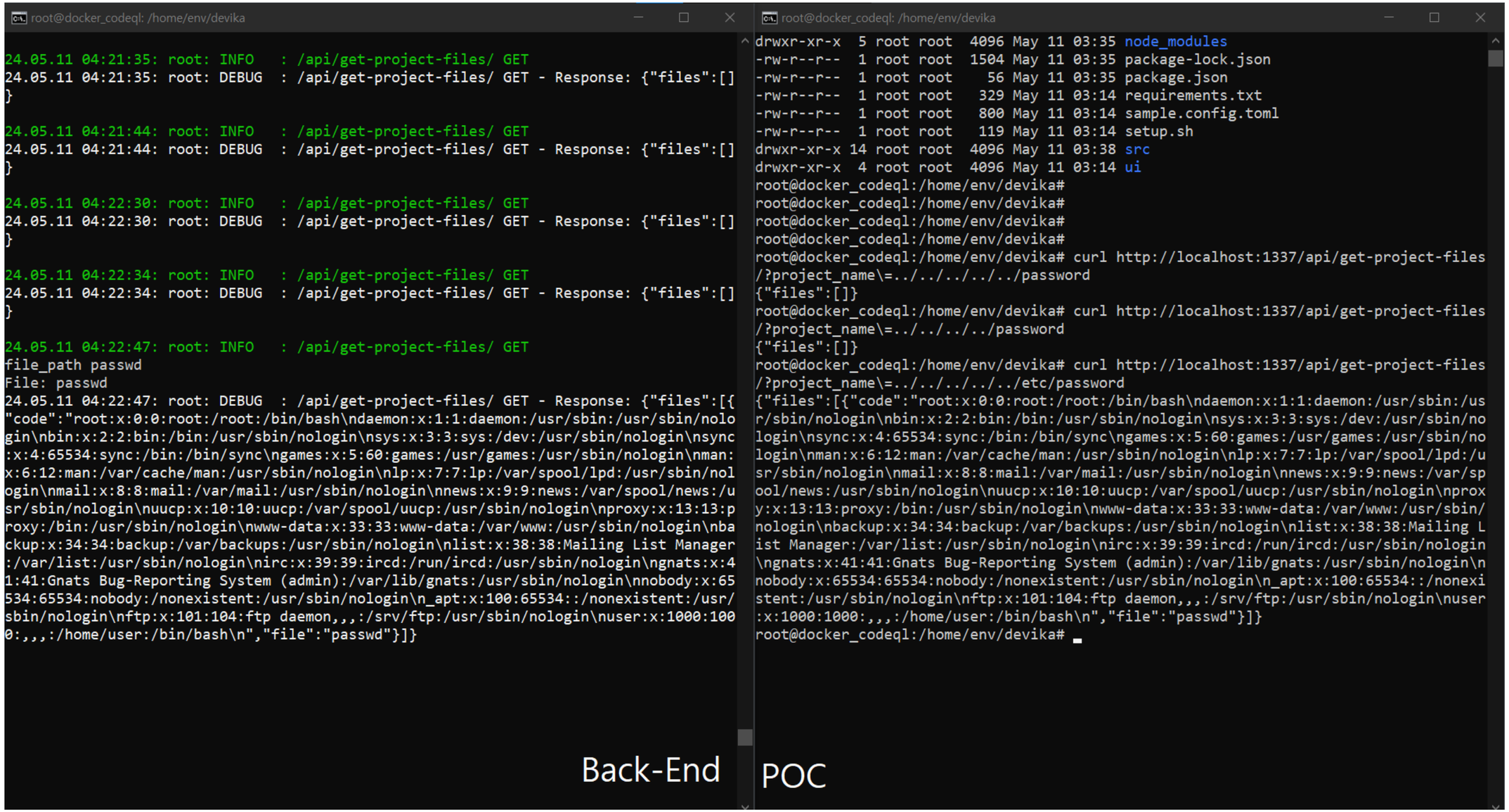


오픈소스 취약점 분석 - Path Injection

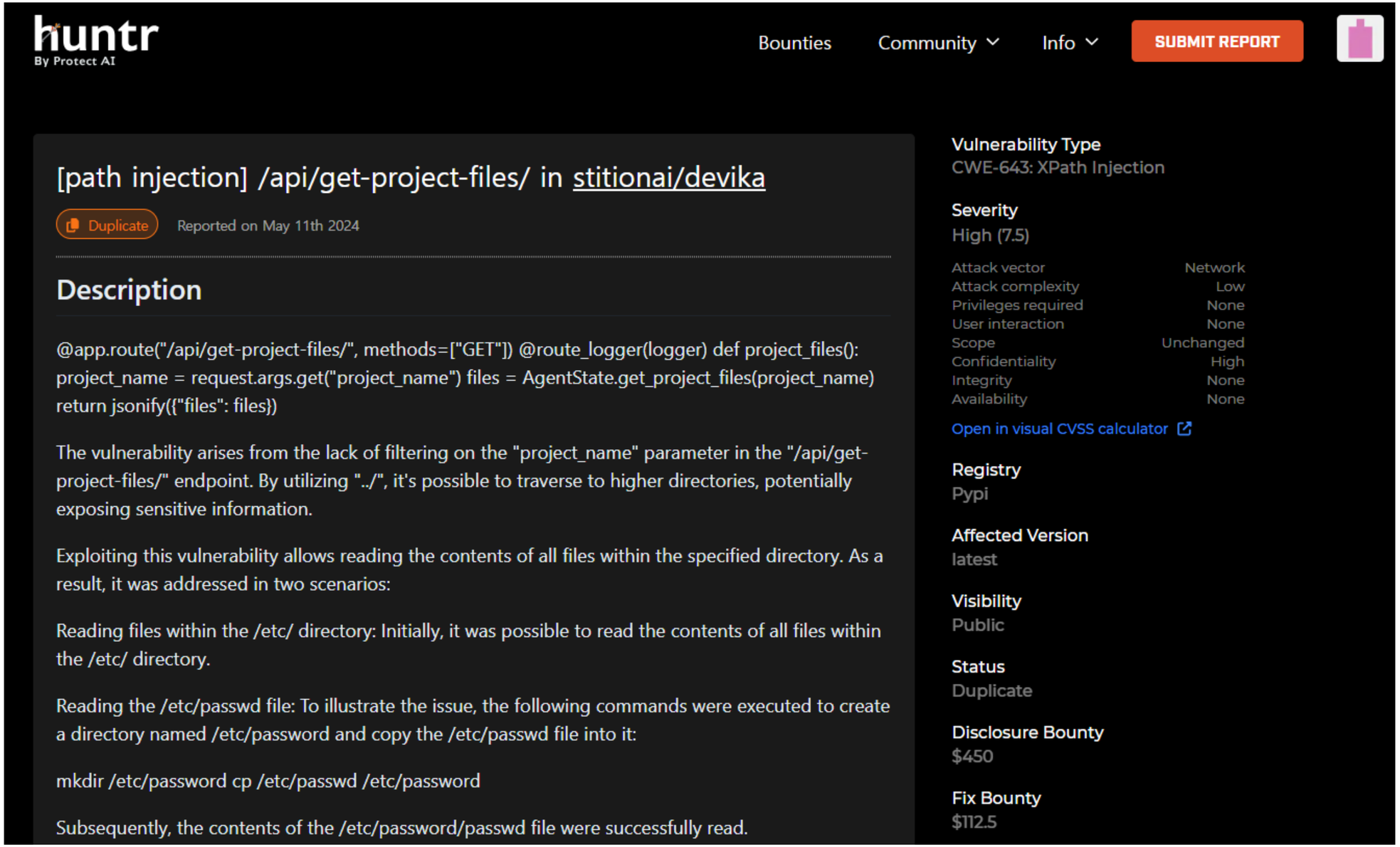
```python
@app.route("/api/get-project-files/", methods=["GET"])
@route_logger(logger)
def project_files():
    project_name = request.args.get("project_name")
    files = AgentState.get_project_files(project_name)
    return jsonify({"files": files})
```

curl http://localhost:1337/api/get-project-files/?project_name₩=../../../../../etc

오픈소스 취약점 분석 - Path Injection



오픈소스 취약점 분석 - Path Injection



huntr
By Protect AI

Bounties Community Info SUBMIT REPORT

[path injection] /api/get-project-files/ in stitionai/devika

Duplicate Reported on May 11th 2024

Description

@app.route("/api/get-project-files/", methods=["GET"]) @route_logger(logger) def project_files(): project_name = request.args.get("project_name") files = AgentState.get_project_files(project_name) return jsonify({"files": files})

The vulnerability arises from the lack of filtering on the "project_name" parameter in the "/api/get-project-files/" endpoint. By utilizing "../", it's possible to traverse to higher directories, potentially exposing sensitive information.

Exploiting this vulnerability allows reading the contents of all files within the specified directory. As a result, it was addressed in two scenarios:

Reading files within the /etc/ directory: Initially, it was possible to read the contents of all files within the /etc/ directory.

Reading the /etc/passwd file: To illustrate the issue, the following commands were executed to create a directory named /etc/password and copy the /etc/passwd file into it:

mkdir /etc/password cp /etc/passwd /etc/password

Subsequently, the contents of the /etc/password/passwd file were successfully read.

Vulnerability Type
CWE-643: XPath Injection

Severity
High (7.5)

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	None
Scope	Unchanged
Confidentiality	High
Integrity	None
Availability	None

Open in visual CVSS calculator

Registry
Pypi

Affected Version
latest

Visibility
Public

Status
Duplicate

Disclosure Bounty
$450

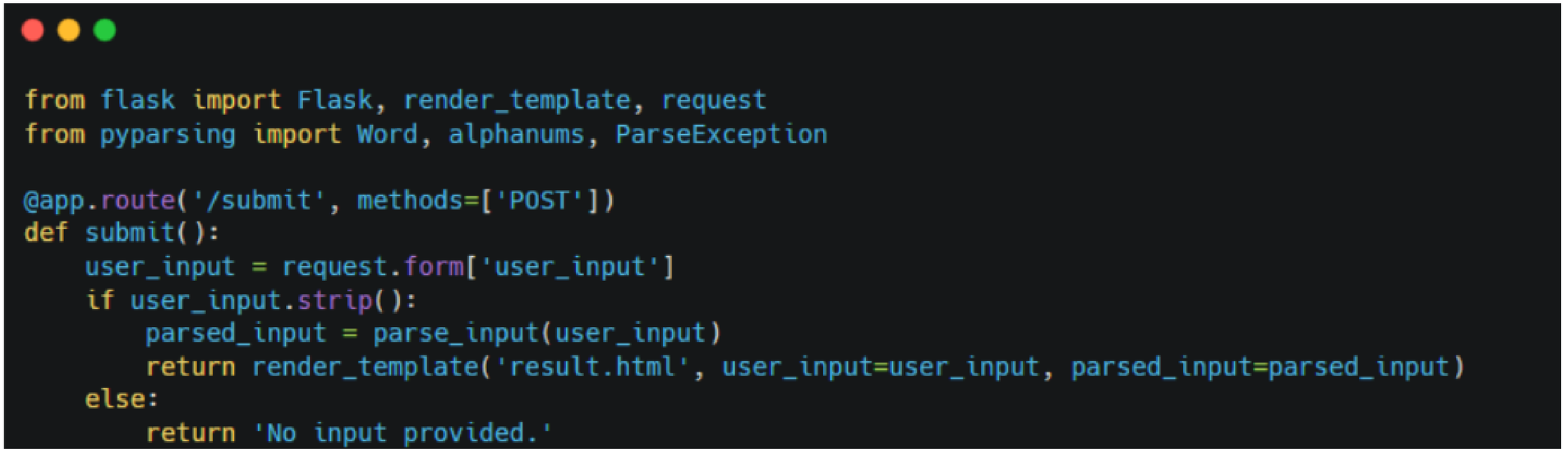
Fix Bounty
$112.5

오픈소스 취약점 분석 - XSS

```
{% endfor %}
{% if not embed %}
</body>
</html>
{% endif %}
"""

template = Template(jinja2_template_source)
```

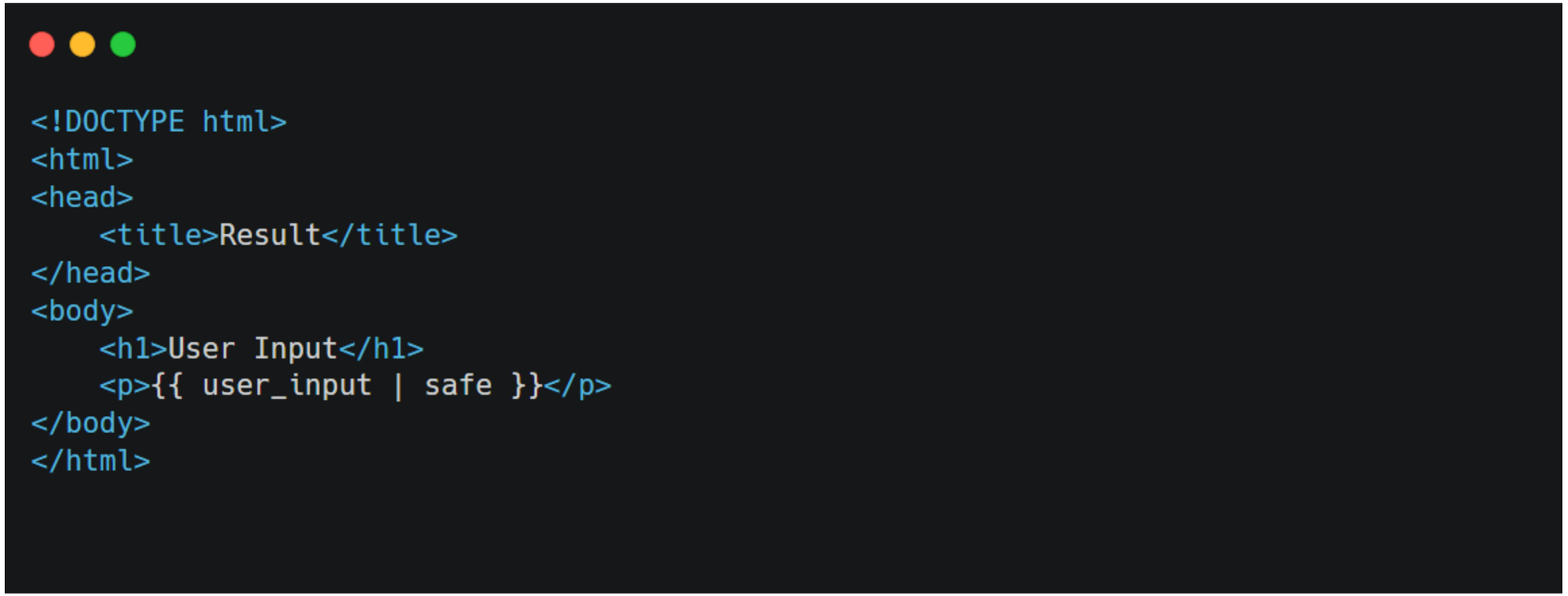
오픈소스 취약점 분석 - XSS



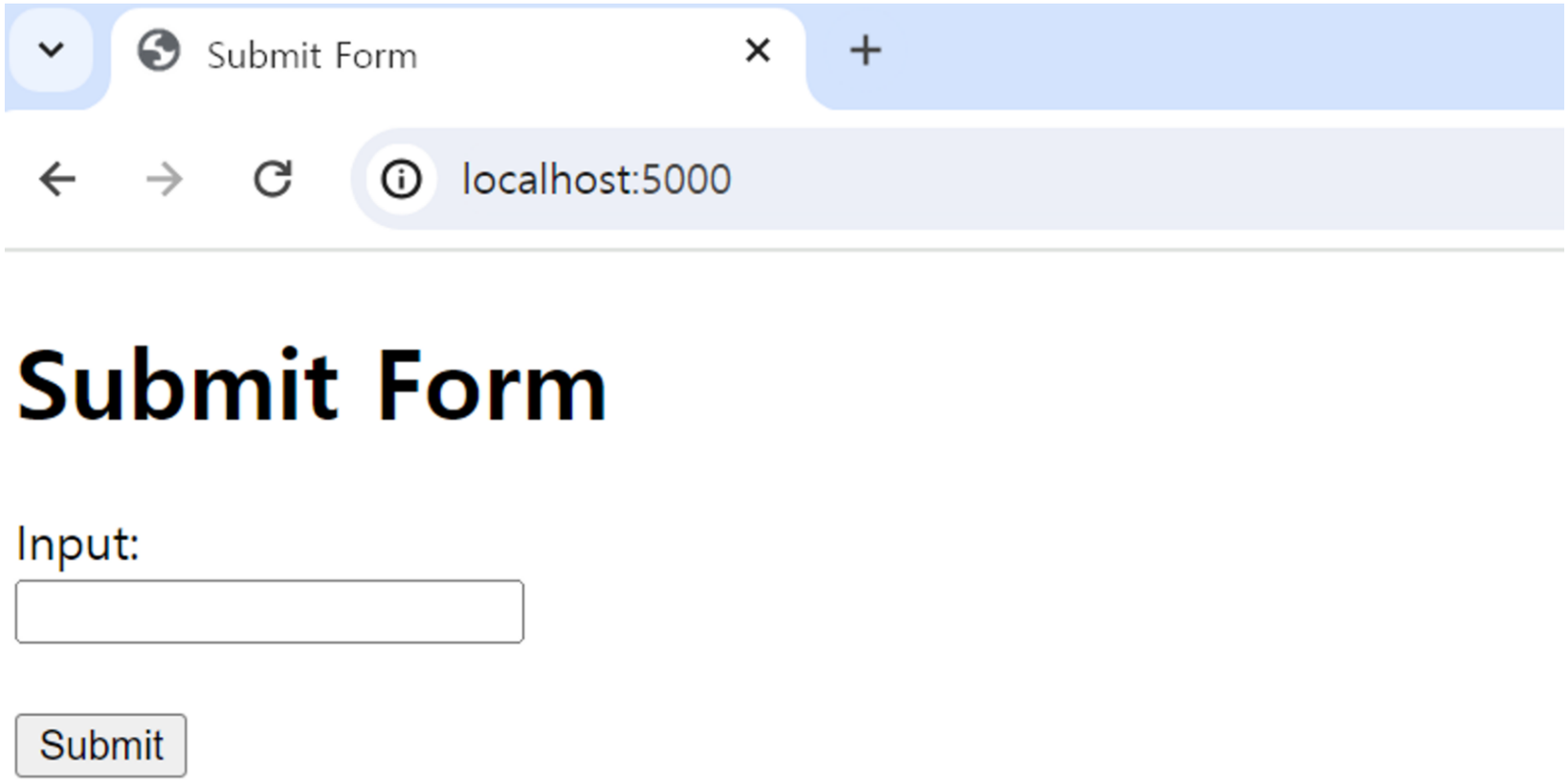
```
from flask import Flask, render_template, request
from pyparsing import Word, alphanums, ParseException

@app.route('/submit', methods=['POST'])
def submit():
    user_input = request.form['user_input']
    if user_input.strip():
        parsed_input = parse_input(user_input)
        return render_template('result.html', user_input=user_input, parsed_input=parsed_input)
    else:
        return 'No input provided.'
```

오픈소스 취약점 분석 - XSS



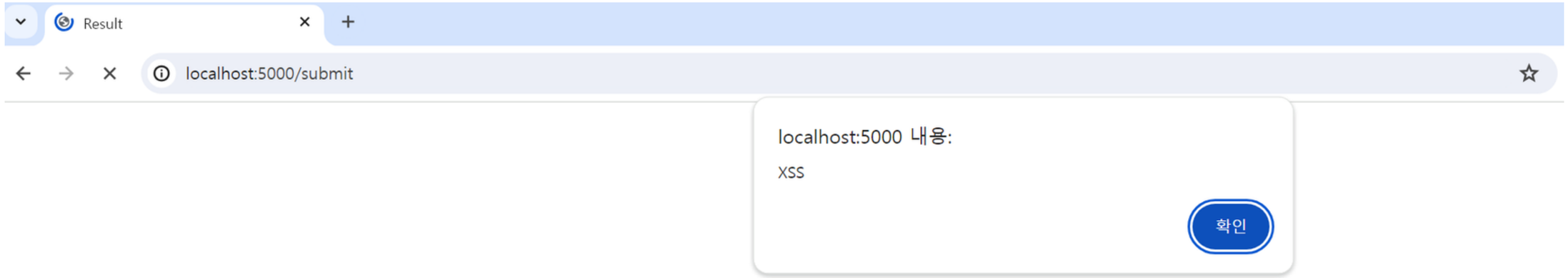
```
<!DOCTYPE html>
<html>
<head>
    <title>Result</title>
</head>
<body>
    <h1>User Input</h1>
    <p>{{ user_input | safe }}</p>
</body>
</html>
```


Submit Form
localhost:5000
Submit Form
Input:
Submit

오픈소스 취약점 분석 - XSS

Input:

<script>alert('XSS')</script>



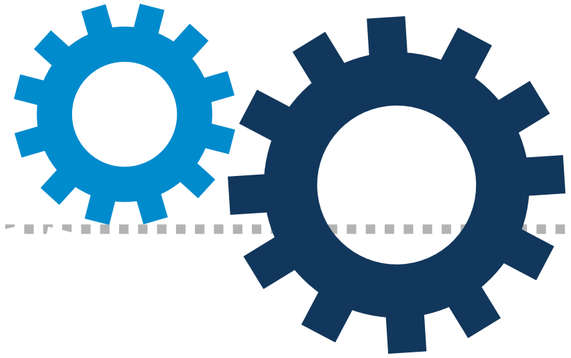
산출물

DOCKER IMAGE

GIT HUB

THESIS

정적 분석 솔루션 기대효과

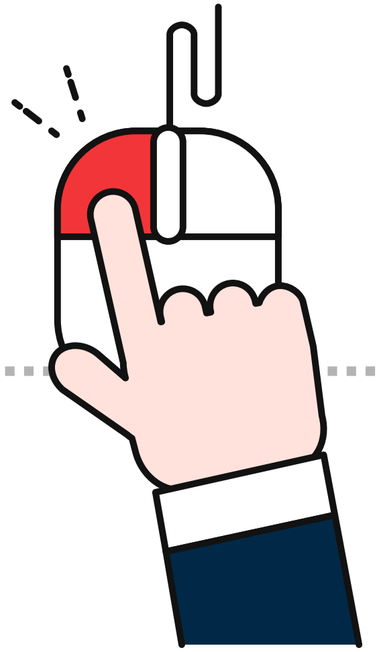


환경구축 소요시간 감소

쉘 스크립트로 환경구축을 자동화 하였습니다.

리서치 소요시간 감소

레포지토리만 입력 받으면 되기 때문에
프로그램 사용법의 대한 리서치를 줄일 수 있습니다.



간단한 프로그램 실행

한 번의 실행으로 세 개의 프로그램을
실행할 수 있습니다.

향후 계획

slack 알림 정보 개선

서비스 안정화

서비스 배포

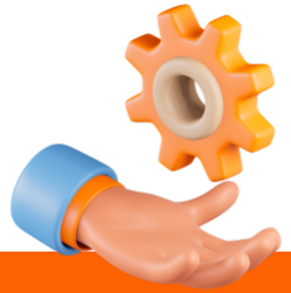
프로젝트를 마치며

정적 분석

- AST TREE 구조 이해
- CodeQL, Semgrep, SonarQube 구축 및 사용 경험

서비스 개발

- Front-End / Back-End 서비스 개발 경험
- Session, threading



CI/CD & 취약점 분석

- DevSecOps, CD 구축 경험
- 취약점 분석 및 리포팅 경험

감사합니다.