

# Portfolio

이담호 (Damho Lee)

# 목차

## Contents

소개

경력 사항

연구 배경

연구 분야

1. 프로그램 분석
2. 프로그램 보안 (공격 및 방어)
3. 빅데이터

관심 분야

1. 머신러닝
2. 홈 네트워크
3. 로또 번호 예측 Telegram 챗봇

교육 이수

수상 경력

# 소개

Who am I?



## 이담호 (Damho Lee)

1989.11.04

홍익대학교 컴퓨터공학과

컴파일러 연구실 박사 수료

와이브레인 연구원 (10년차)

Phone : 010-2734-5798

Email : damho1104@gmail.com

LinkedIn : <https://www.linkedin.com/in/damho1104>

Github : <https://github.com/damho1104>

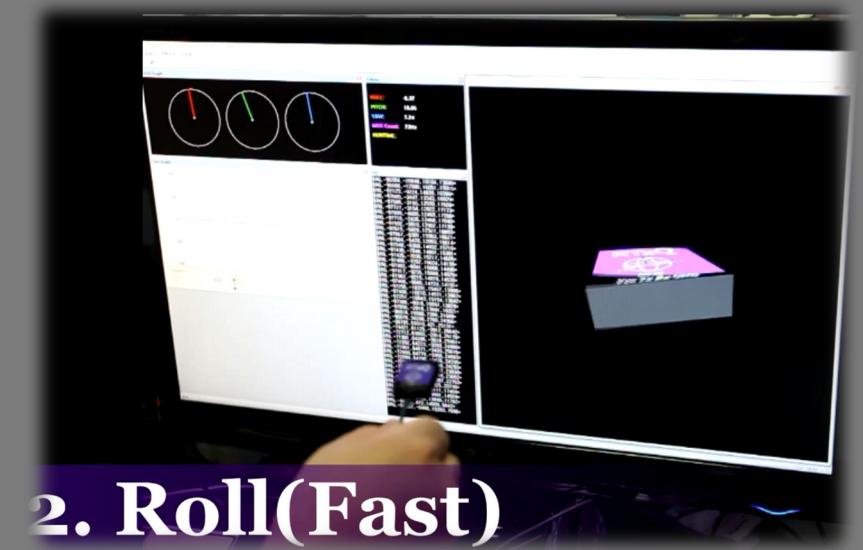
### Keyword

- C/C++, Python, Java, Kotlin, Shell script
- LLVM, Clang, Static Instrumentation, KLEE
- Program Analysis (Static / Dynamic)
- Control-flow Hijacking
- Control-flow Protection
- KLEE, Symbolic Execution
- Flask, FastAPI, SpringBoot
- Docker, AWS, Web Backend, Infra

# 경력 사항

## RUVA Tech (2011 ~ 2012)

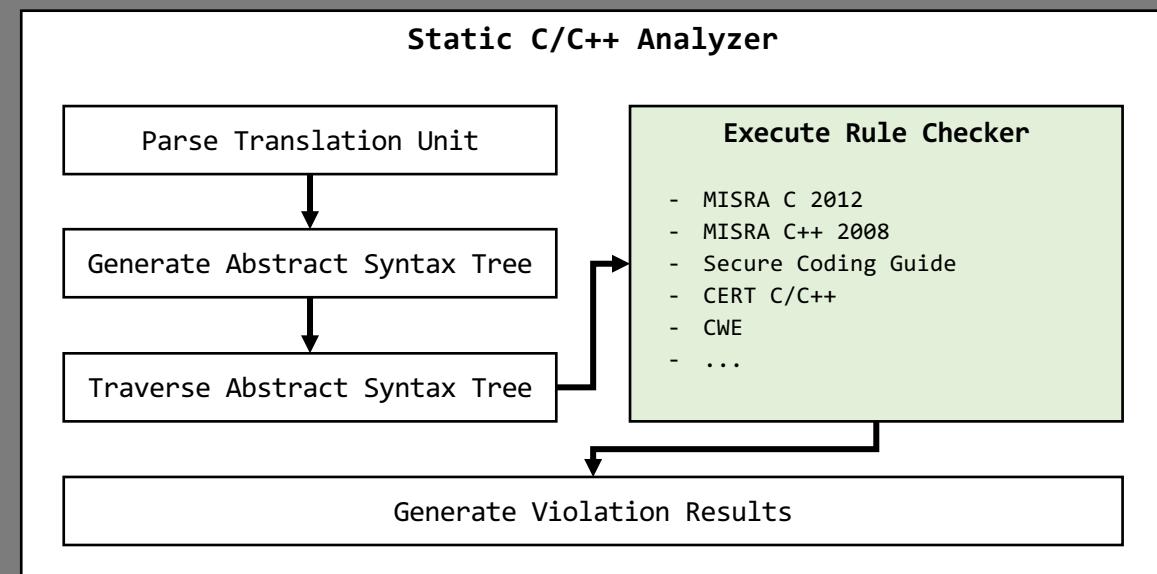
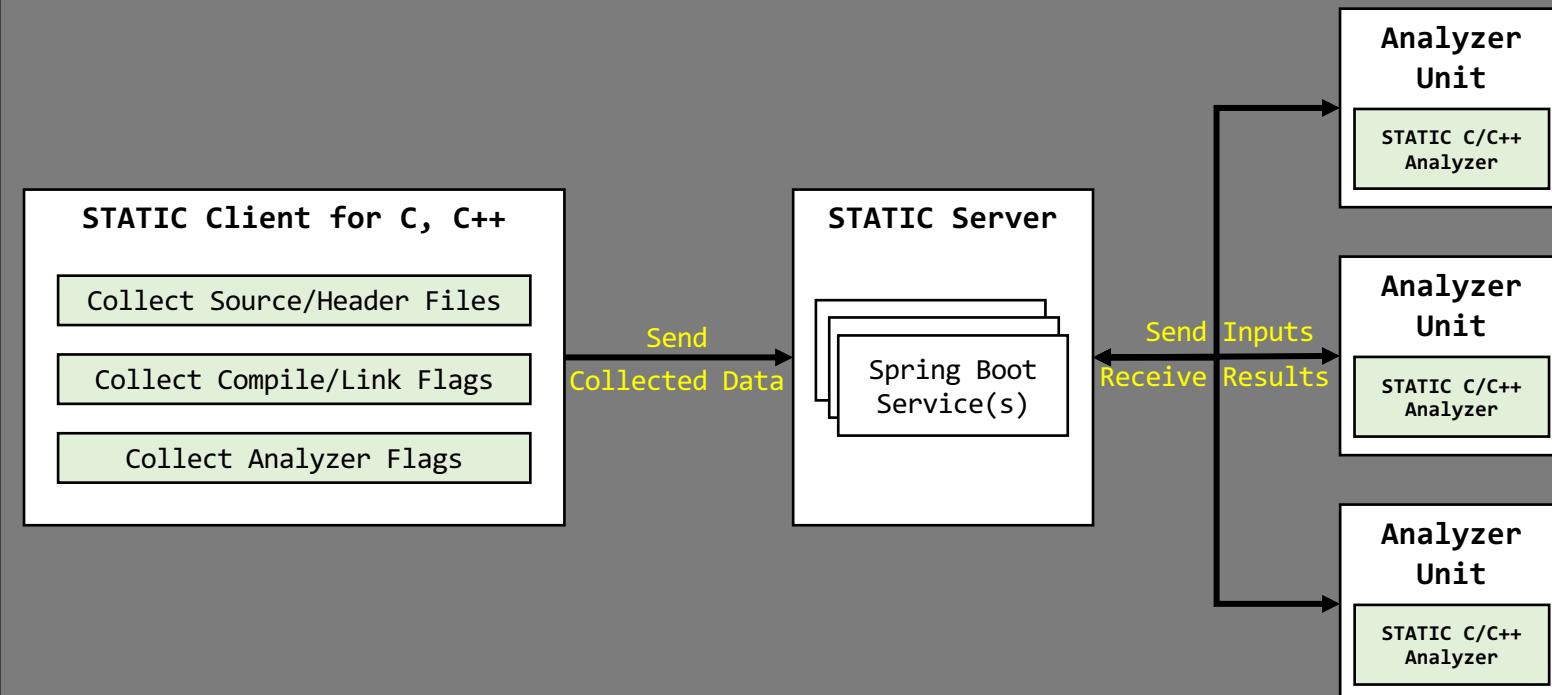
- RUVA Tech 위치 참조 센서 연동 UI
  - OpenGL 을 사용한 그래픽 표현
    - 오일러 각/쿼터니언 표현 방식 사용
    - 현재의 센서 값을 그래픽으로 표현
    - 마우스와 연동하여 게임 테스트
  - 복수의 센서를 사용한 모션 캡쳐 시스템
    - 센서 값을 인체 모형 그래픽에 표현
- 결과물
  - 논문 1건
    - 위치참조 센서를 사용한 스트레칭 자세 교정 시스템, 한국정보과학회 학술발표논문집 2012.



# 경력 사항

## 슈어소프트테크 (2019 ~)

- STATIC C/C++ Analyzer
  - Secure Coding 규칙 개발
    - CERT C/C++, CWE 규칙 기반 Suresoft Secure Coding 규칙 개발
    - Abstract Syntax Tree 기반 Checker 형태
- STATIC Client for C, C++
  - STATIC 서버에서 정적 분석을 위한 사용자 프로젝트 소스 및 빌드 정보 추출 도구 개발 및 유지보수
  - 사용 기술
    - DLL Injection
    - API Hooking
    - Compiler Preprocessor
    - Cross Compile Technique



# 경력 사항

## 슈어소프트테크 (2019 ~)

### • STATIC Analysis Service

- Spring Boot 기반 웹 백엔드 서비스
- 엔터프라이즈 급 웹 기반 정적 분석 도구 STATIC의 핵심 서비스
  - 프로젝트 분석 정보 관리 및 분석 결과 모니터링
  - Analyzer Unit 연결 관리
  - Analyzer 패치 관리
  - JPPF 기반 TU/Link 단위 병렬 분석
- 사용 기술

Java

Spring Boot

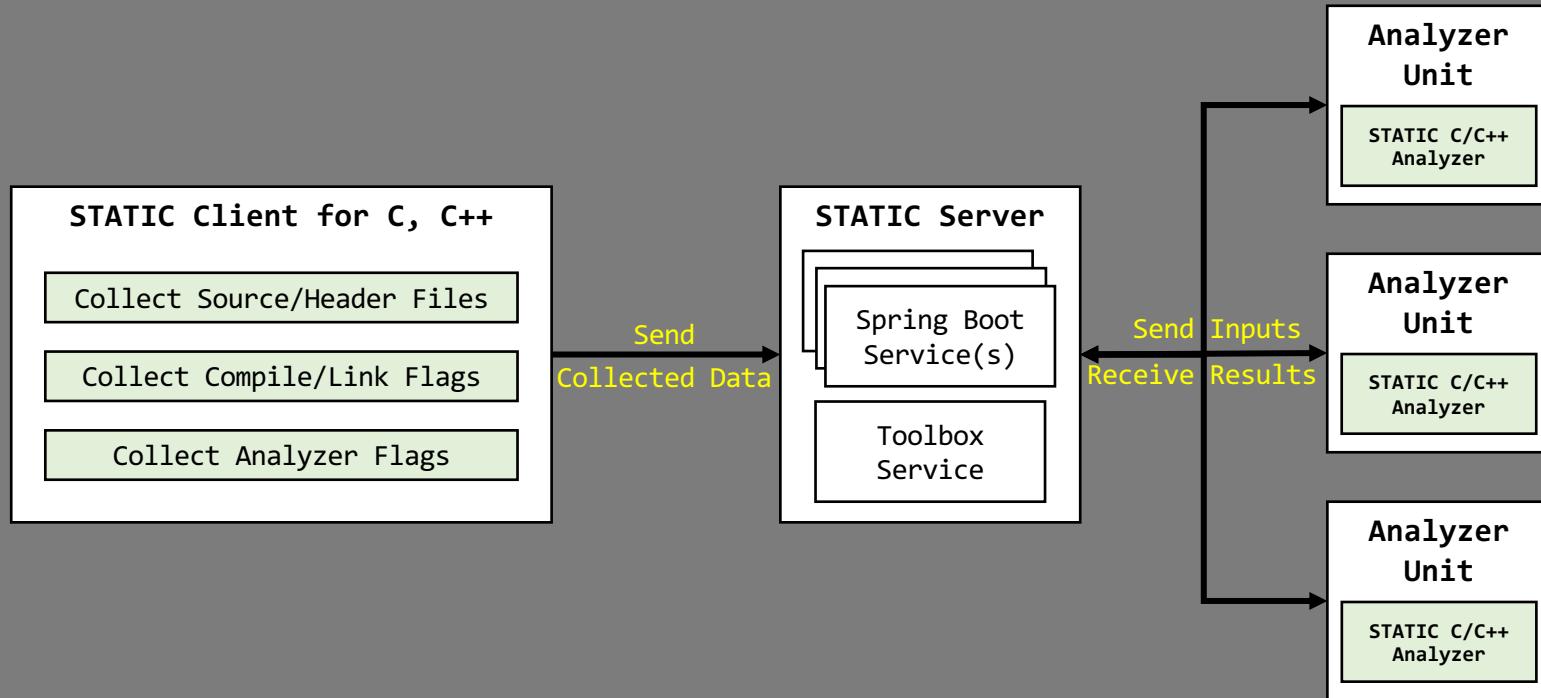
JPPF(Java Parallel Programming Framework)

### • STATIC Toolbox Service

- Flask 기반 웹 백엔드 서비스
- STATIC 을 이용하는 다수의 사용자들의 패키지 버전을 관리하는 서비스
- 사용 기술

Python

Flask



# 경력 사항

## 슈어소프트테크 (2019 ~ 2022)

- STATIC Client for C#

- C# 프로젝트 소스 및 빌드 정보를 통해 정적 분석 수행하는 클라이언트 개발
  - Roslyn .Net Compiler Platform 라이브러리 사용한 소스 및 빌드 정보 추출
  - FxCop, StyleCop 연동
  - C# Metric Calculator 개발
    - PLOC, LOC, Cyclomatic Complexity, Nesting Depth

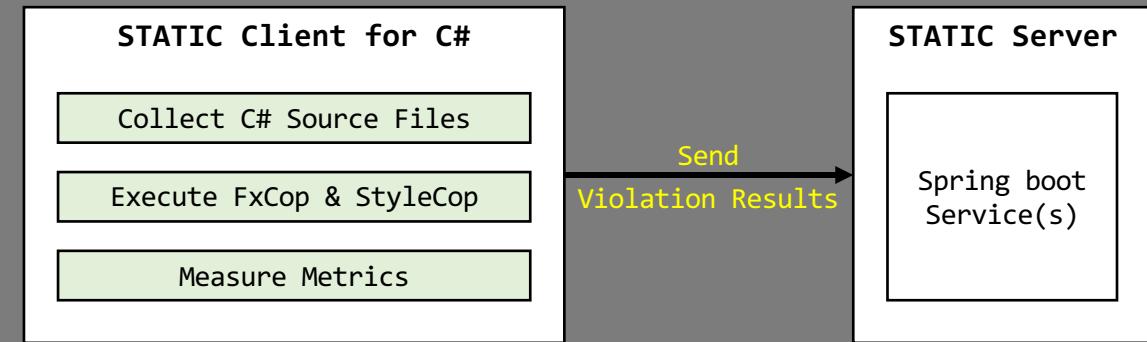
- Continuous Integration System

- STATIC 서비스 및 클라이언트 빌드, 테스트 및 배포 자동화 시스템, 사내 Code Sign Service 구축 및 관리
  - Framework

Gitlab CI/CD

Jenkins

Flask

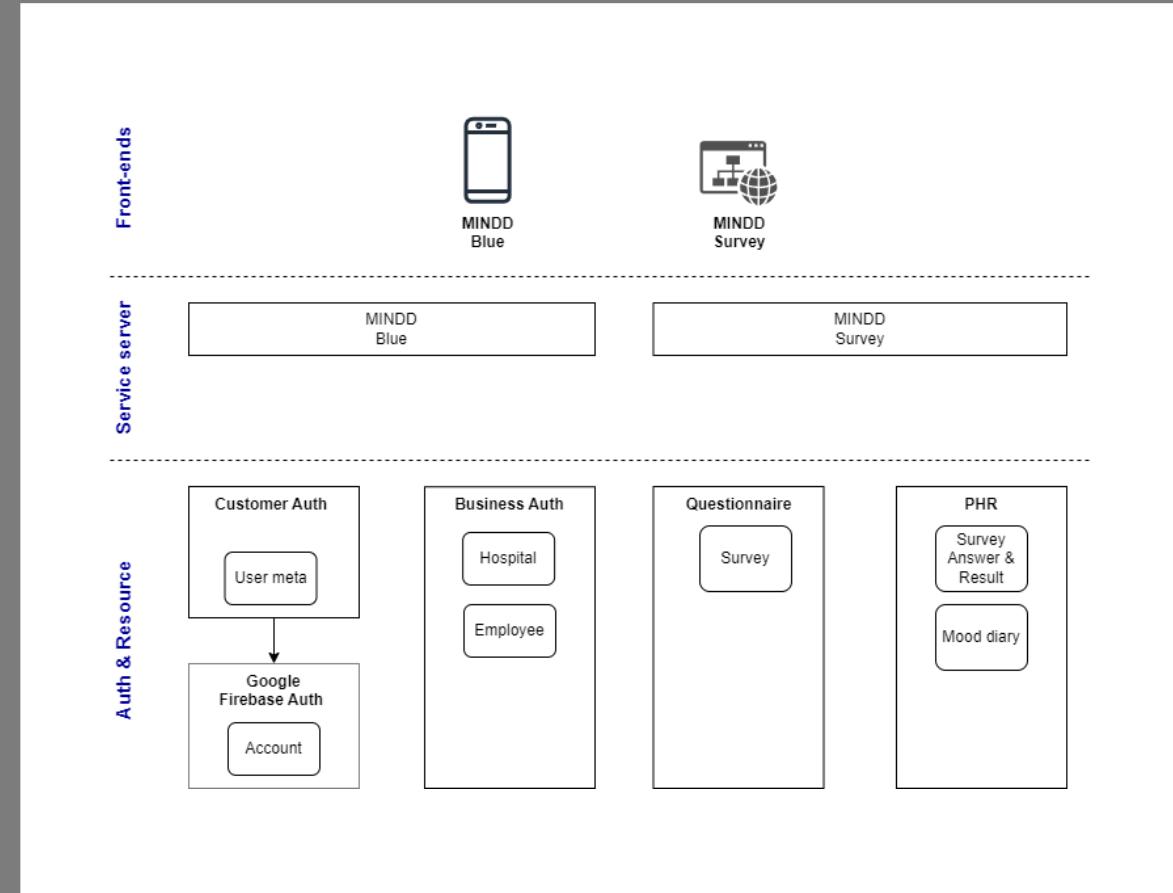


# 경력 사항

## 와이브레인 (2022 ~)

### • 정신 건강 관리 서비스 MINDD

- 온라인 척도검사를 위한 웹 백엔드 서비스 MINDD Survey 개발
  - 기존 병원에서 종이로 척도검사를 진행했던 방식을 온라인으로 가져와서 진행하는 서비스
  - B2B (병원 대상)
  - 담당: API 백엔드 서비스 및 기록 저장 서비스 개발
- 개인 정신 건강 관리를 위한 웹 백엔드 서비스 MINDD Blue 개발
  - 개인 정신건강을 일부 온라인 척도검사와 일기를 통해 관리, 일상 루틴을 등록하고 관리
  - B2C
  - 담당: API 백엔드 서비스 및 기록 저장 서비스 개발
  - 사용 기술
    - MSA
    - Springboot Webflux
    - MongoDB
    - Kotlin

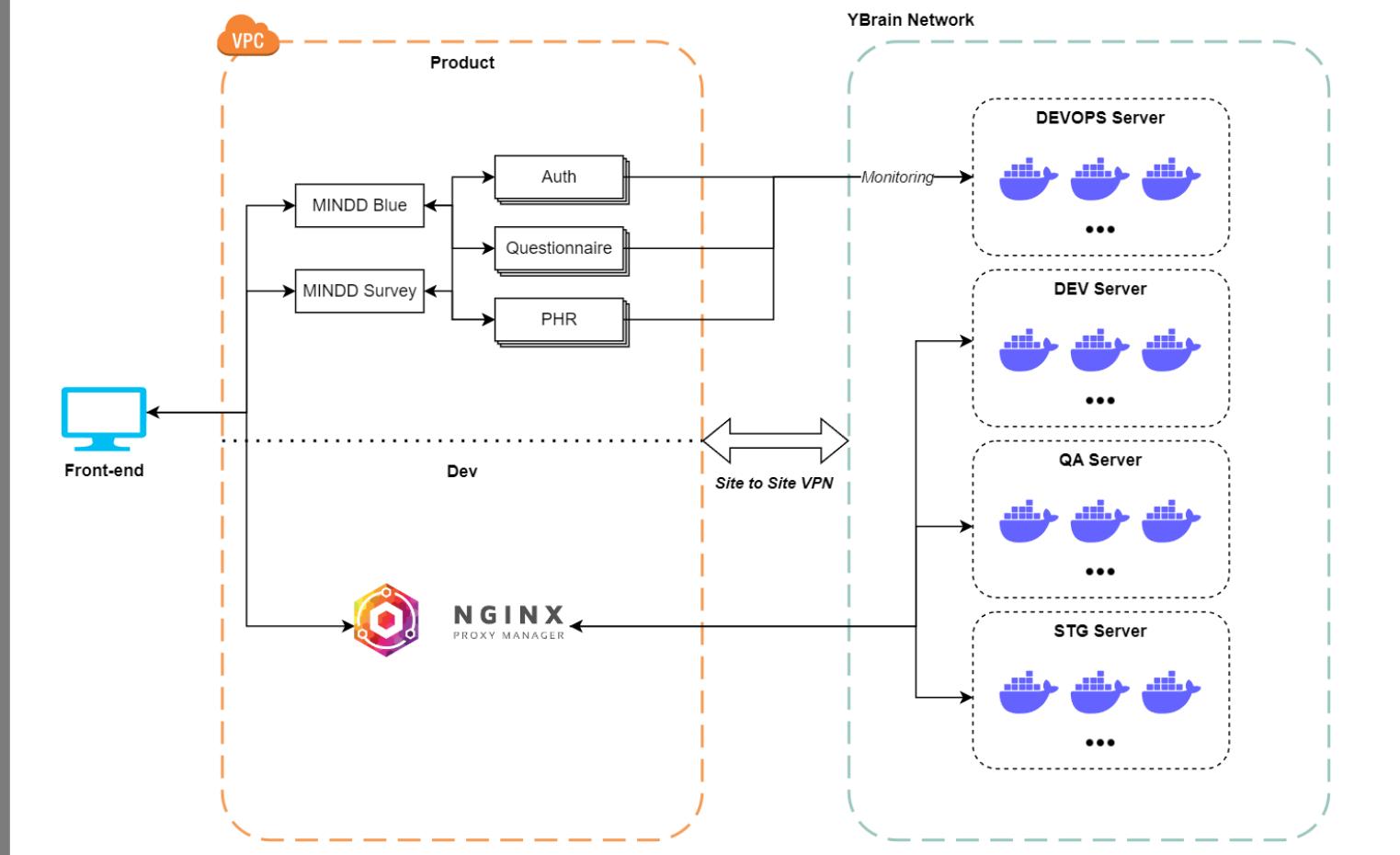


# 경력 사항

## 와이브레인 (2022 ~)

- **인프라 관리**

- 원격 접속을 위한 VPN 서버 구축
  - Docker 기반 Wireguard VPN 서버
- 사내 개발 환경 백엔드 서비스 인프라 구축
  - Docker 기반 Nginx Proxy Server
  - Docker 기반 로그 모니터링 도구
- APM 도구 Signoz 구축 및 관리
- AWS
  - EC2
  - Lambda
  - API Gateway
  - Beanstalk
  - WAF



# 연구 배경

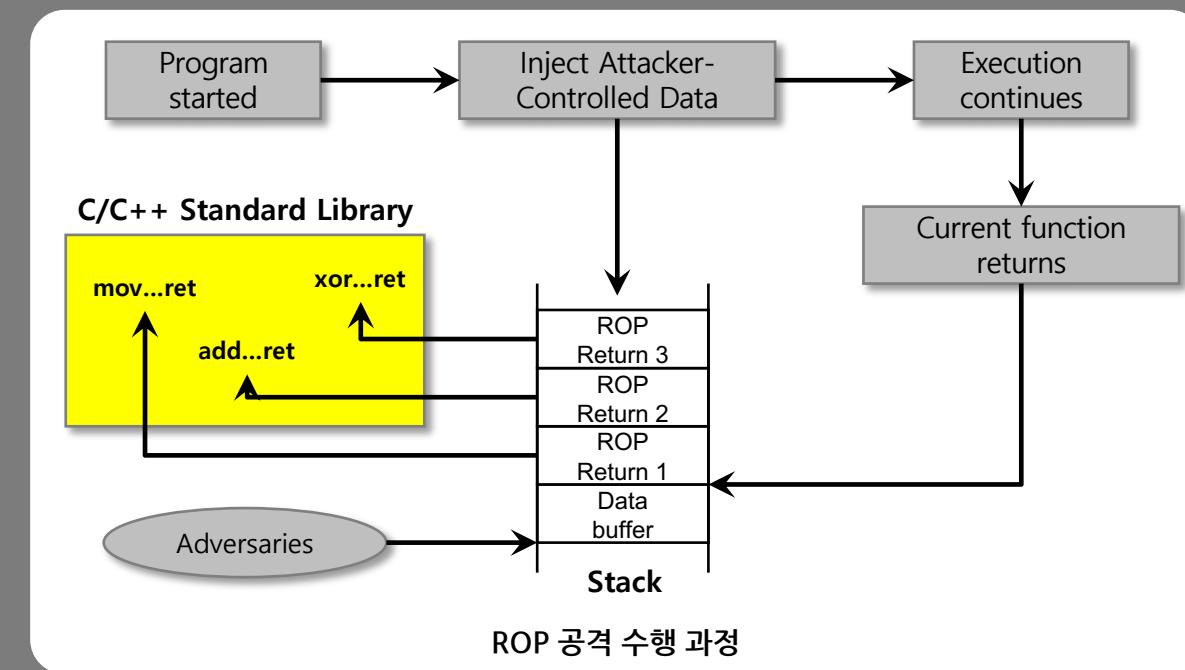
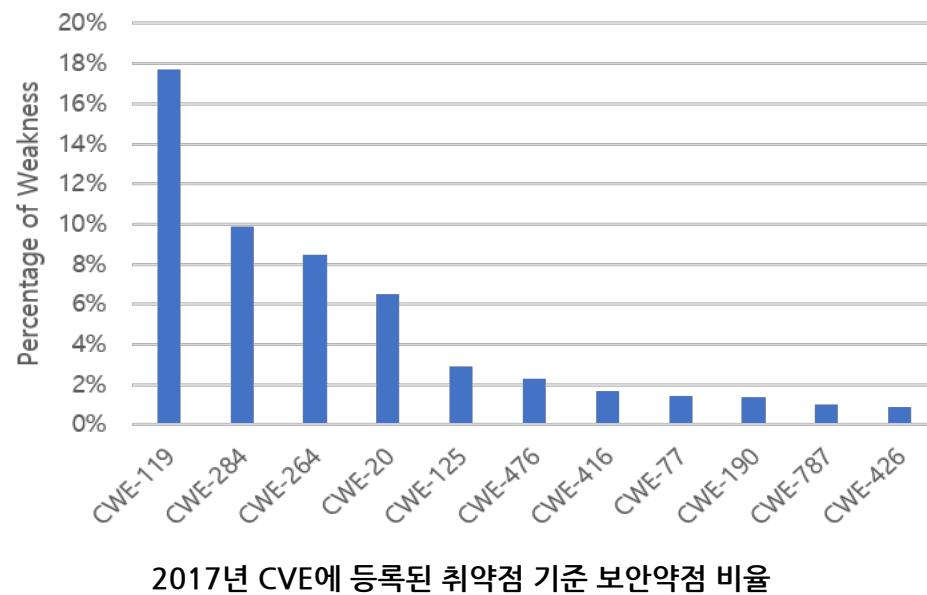
## Background

- 시스템 개발 언어의 문제

- C/C++ 언어 문제
  - 약한 타입시스템
  - 포인터의 무제한적 사용
  - 통제되지 않는 타입 변환
  - 배열의 불완전 처리
- 언어의 문제로 인해 보안약점 발생  
예: 버퍼 오버플로우

- 메모리 안전성 부재로 인한 공격 존재

- 메모리 안전성 부재로 인해 제어흐름 데이터 변조 가능
  - 제어흐름 데이터 예: 코드 포인터
- 제어흐름 데이터 변조를 시작으로 프로그램 공격 수행
  - 코드 재사용 공격 (Code Reuse Attack)
  - Return-Oriented Programming (ROP)



# 연구 분야 1

## 프로그램 분석

- **C/C++ 시큐어코딩 연구**

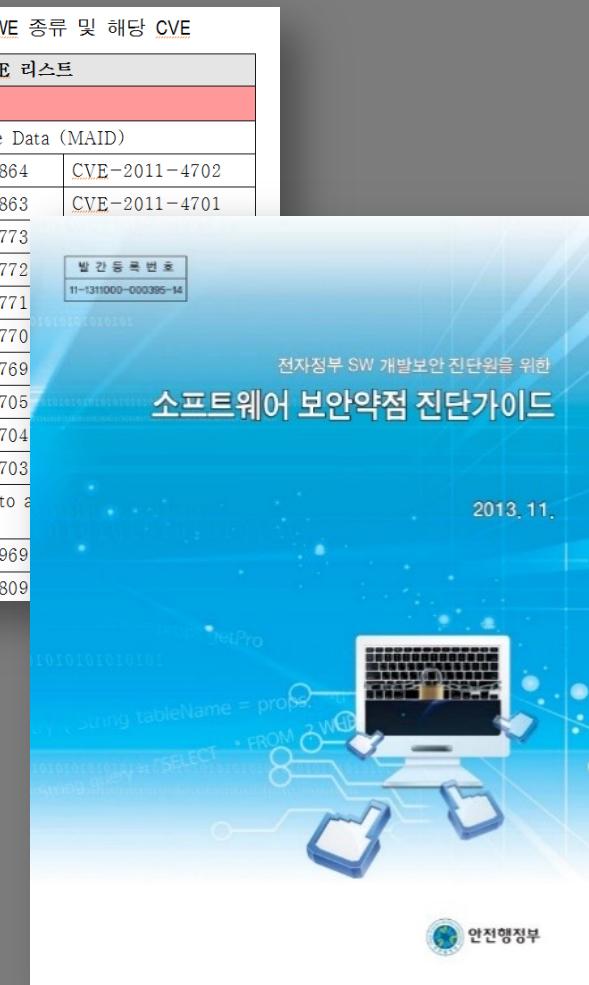
- 47개의 보안약점에 대응되는 3가지 계층, 14가지 규칙 분류
- 시큐어코딩 가이드를 통해 보안 지식 학습 가능

- **보안약점 진단 규칙 설계**

- C++ 관련 보안약점 하위 85개 항목의 분석
- 보안약점 탐지 규칙을 기술하기 위한 언어 정의

표 9. 안드로이드 시큐어 코딩 규칙 유형이 원화하는 CWE 종류 및 해당 CVE

규칙	CWE	CWE 설명 / CVE 리스트		
입력 값 검증				
CWE-471		Modification of Assumed-Immutable Data (MAID)		
	<a href="#">CVE-2014-1977</a>	<a href="#">CVE-2011-4864</a>	<a href="#">CVE-2011-4702</a>	
	<a href="#">CVE-2013-2301</a>	<a href="#">CVE-2011-4863</a>		<a href="#">CVE-2011-4701</a>
	<a href="#">CVE-2013-2300</a>	<a href="#">CVE-2011-4773</a>		
	<a href="#">CVE-2013-0720</a>	<a href="#">CVE-2011-4772</a>		
	<a href="#">CVE-2013-0719</a>	<a href="#">CVE-2011-4771</a>		
	<a href="#">CVE-2013-0718</a>	<a href="#">CVE-2011-4770</a>		
	<a href="#">CVE-2012-6422</a>	<a href="#">CVE-2011-4769</a>		
	<a href="#">CVE-2011-4867</a>	<a href="#">CVE-2011-4705</a>		
	<a href="#">CVE-2011-4866</a>	<a href="#">CVE-2011-4704</a>		
	<a href="#">CVE-2011-4865</a>	<a href="#">CVE-2011-4703</a>		
CWE-22		Improper Limitation of a Pathname to a ('Path Traversal')		
	<a href="#">CVE-2014-1975</a>	<a href="#">CVE-2014-1969</a>		
	<a href="#">CVE-2014-1506</a>	<a href="#">CVE-2014-0809</a>		



# 연구 분야 1

## 프로그램 분석

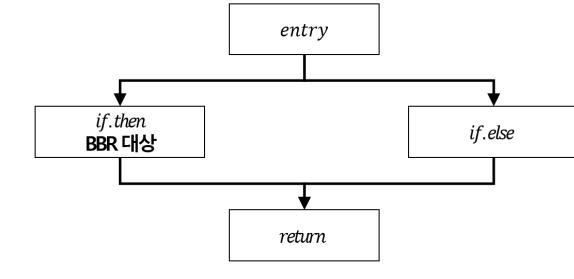
- Basic Block Restriction (BBR)
  - 정상적으로 종료되는 입력으로 동적 분석하여 **실행되지 않은 기본 블록들의 접근 제한**
  - Basic Block Profiling을 통한 실행되지 않은 기본 블록 수집
  - Framework
    - LLVM**
    - Compiler-RT**
- Unreachable Region Analysis based on Input Conditions
  - 미리 지정한 입력 조건에 따라 실행되지 않을 Basic Block 탐색 방법
  - Framework
    - KLEE Symbolic Executor**

## Basic Block Restriction

```

1 float divide(float a, int b){
2     float c = 0.0;
3     if(b == 0)
4         printf("divide by 0\n");
5     else
6         c = a/b;
7     return c;
8 }
  
```

(a) 예제 코드



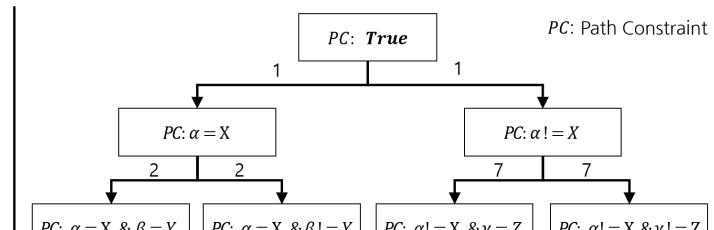
(b) 제어흐름 그래프

## Unreachable Region Analysis based on Input Conditions

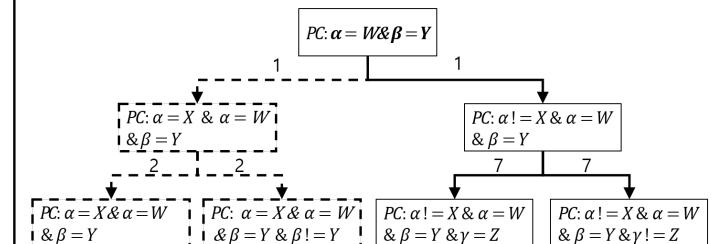
```

1 if(argv[1][0]=='X'){ //argv[1][0]:α
2     if(argv[2][0]=='Y') //argv[2][0]:β
3         printf("arg2=Y\n");
4 }
5
6 else{
7     if(argv[3][0]=='Z') //argv[3][0]:γ
8         printf("arg3=Z\n");
9 }
  
```

(a) 예제 코드



(b) 입력의 조건을 추가하지 않은 기호 실행 트리



(c) 입력의 조건을 추가한 기호 실행 트리



# 연구 분야 2-1

## 프로그램 보안(공격)

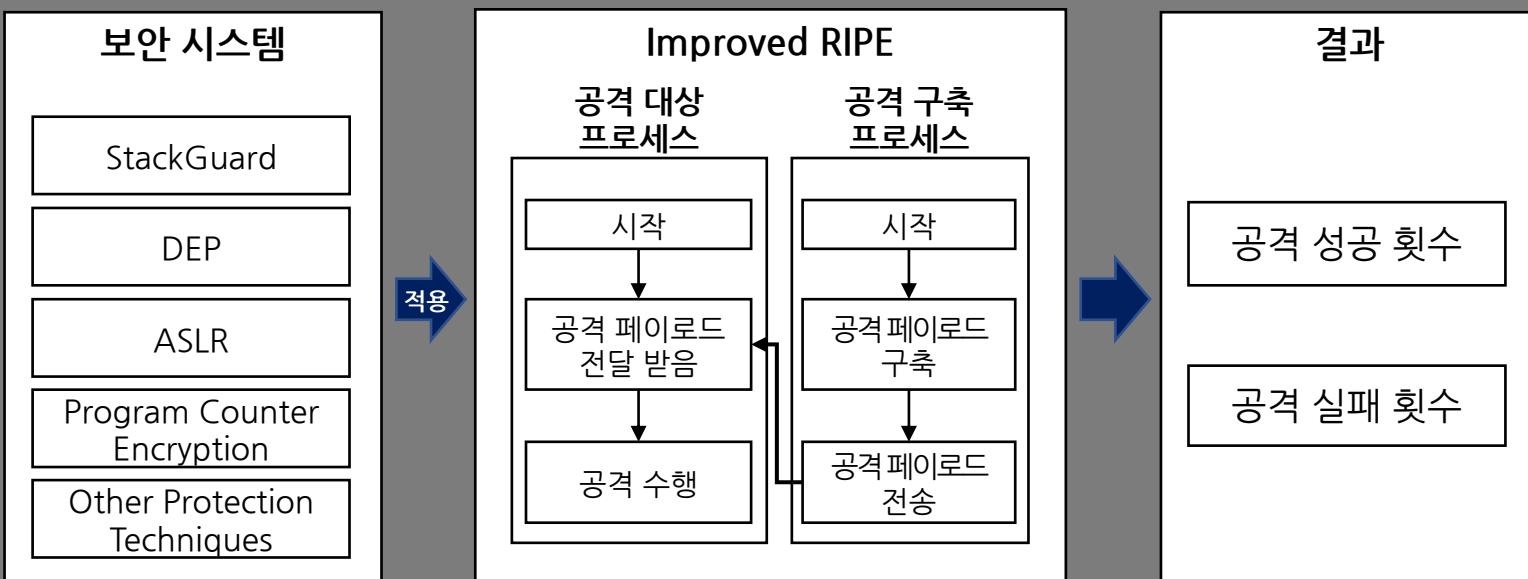
### • 보안 시스템 검증 벤치마크 RIPE 확장

- OS 수준, 컴파일러 수준 보안 능력 검증
- 888 가지 버퍼 오버플로우 기반 제어흐름 탈취 공격
  - RIPE 벤치마크 대비 38가지 공격 추가
  - 방어 검증 요소 추가

```
865. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l bss -f strcat
866. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l bss -f sscanf
867. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l bss -f fscanf
868. [PASS] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f memcpy
869. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f strcpy
870. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f strncpy
871. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f sprintf
872. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f snprintf
873. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f strcat
874. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f strcpy
875. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f strncat
876. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f sscanf
877. [FAIL] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f fscanf
878. [PASS] ./build/ripe_attack_generator -t indirect -i returnintolibc -c baseptr -l data -f homebrew
879. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f memcpy
880. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f strcpy
881. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f strncpy
882. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f sprintf
883. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f snprintf
884. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f strcat
885. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f strncat
886. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f sscanf
887. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f fscanf
888. [FAIL] ./build/ripe_attack_generator -t direct -i rop -c ret -l stack -f homebrew

=====
#PASS : 400
#FAIL : 488
#TOTAL: 888
=====

crl@crl-VirtualBox:~/RIPE/RIPE-revised$
```



# 연구 분야 2-1

## 프로그램 보안(공격)

### • 참여 프로젝트

- 모바일 기기 시스템 계층별 보안 강화 기법 연구. 국방과학연구소(위탁과제).  
2014.03~2015.02
- 고품질 융합 소프트웨어 개발 지원 도구 연구. 정보통신기술진흥센터(미래창조과학부 지원과제).  
2014.01~2016.12

### • 연구 결과물

#### • 논문 2건

실행시간 침입 방지 평가 프로그램 RIPE의 개선과 확장, KCC 2014.

실행시간 침입 방지 평가 프로그램(RIPE)의 개선, KIISE 2015.

#### • 프로그램 등록 1건

개선된 실행시간 방어 기법 평가 프로그램, 한국저작권위원회 (C-2014-022811).

2014년 한국컴퓨터종합학술대회 논문집

### 실행시간 침입 방지 평가 프로그램 RIPE의 개선과 확장\*

이 담호<sup>01</sup>, 이 담호<sup>01</sup>

Improvement and Expansion of Rute

Damho Lee<sup>01</sup>, Hyungyu L  
Ho

비파 오버플로우 기법은 비파의 경계 값을  
복사되는 현상을 의미한다. 2011년 John Wil-  
ian 프로그램 보안 기법 평가 도구인 RIPE를  
보유한 공유하는 때문에 기밀성에 근거한 프-  
로그램을 확장하는 것은 중요한 보안 기법  
이다. 이러한 이유로 많은 보안 기관은  
정보의 기밀성을 보장하였다. 또한 네트워-  
크 기법 기반의 RIPE를 확장시켰다.

#### 1. 서 론

프로그램 공격은 공격자가 프로그램 제어 흐름을  
할 수 있는 데이터를 원하는 값으로 변조하면서  
된다. 프로그램 공격 기법의 대표적인 사례는 비파 오-  
버플로우[1]이다. 비파 오버플로우는 비파의 경계를  
넘지 않는 함수나 프로그램을 통해 비파의 크기 이상  
데이터가 복사되는 현상을 말한다. 비파의 경계 밖으로  
데이터가 복사되면 함수의 코드 포인터와 같은 제어  
판권원 데이터와 포인토어 프로그램은 공격자가 뒤  
제어 흐름으로 변경된다. 이러한 공격은 예방하기  
ProtoPolice[2]와 StackShield[3] 등의 방어 기법이  
있고 또한 방어 기법들을 평가하는 도구들도 만-  
들었다.

2011년에 John Wilard는 비파 오버플로우 기-  
법에 대한 프로그램 보안 기법 평가 도구인 RIPE  
발표하였다. RIPE는 공격 코드, 위치등의 공격  
조합하여 만들어진 비파 오버플로우 공격으로 훈-  
련과 커널에서 제공하는 방어 기법들을 평가하는  
수행한다. RIPE 프로그램 보안 기법 평가 도구는

\* 본 연구는 미래창조과학부 및 정보통신산업진흥원의  
연구센터육성 지원사업/IT융합고급인력과정지원사업의  
결과로 수행되었다 (NIPA-2014-H0301-14-1023)

\* 본 연구는 국방과학연구소에서 수행중인 “사이버 공격  
비한 금융 대기능 모바일 단말기의 보안 SW 개발”  
제원으로 수행되었다 (UD140009ED).

홍익대학교 | IP: 223

홍익대학교 | IP: 223.194.69.\*\*

ISSN 2383-630X(Print) / ISSN 2383-6296(Online)  
Journal of KIISE, Vol. 42, No. 8, pp. 1049-1056, 2015. 8  
<http://dx.doi.org/10.5626/JOK.2015.42.8.1049>

### 실행시간 침입 방지 평가 프로그램(RIPE)의 개선

(Improvement of Runt  
Evaluation)

이현규<sup>\*</sup> 이 담호<sup>\*</sup>  
(Hyungyu Lee) (Damho Lee)

이상훈<sup>\*\*\*</sup> 김훈  
(Sanghoon Lee) (Hoonkyu

요약 2011년에 발표된 RIPE는 프로그램 공격  
오버플로우 기반 공격에 대한 완화 기법들을 평-  
행 실행되도록 구성되어, RIPE가 실행될 때에는 공격자  
할 수밖에 없게 된다. 그 결과 공격 무단은 방어가  
된다. 이 논문에서는 RIPE의 공격과 방어 무단이  
폐지 난독화와 같은 기밀성이 근거한 방어 기법을  
에 대한 방어 능력을 실험할 수 있도록 실험 모드  
문자열 공격을 수행하도록 38 가지 베이트의 공격을  
되었고, 보호 효과 평가의 정확성을 높아졌다.  
키워드: RIPE, 비파 오버플로우, 주소 공간 난독화

**Abstract** Runtime Intrusion Prevention Eval  
for evaluating mitigation techniques against RIPE  
is built as a single process, defense and at  
address space layouts when RIPE is tested. As a  
for defense routines without restriction. We sepa

제 C-2014-022811 호

### 프로그램 등록증

1. 프로그램의 제호 개선된 실행시간 방어 기법 평가 프로그램  
(명칭)

2. 저작자 성명 (법인명) 고려대학교 산학협력단  
서울특별시 성북구 암암로 3. 생년월일 (법인등록번호) 114471-0002565

4. 창작연월일 2014년05월

5. 공표연월일 2014년06월25일

6. 등록사항 저작자 : 고려대학교 산학협력단,  
등록자 : 2014.05, 공표 : 2014.06.25

7. 등록연월일 2014년09월16일

『저작권법』 제53조에 따라 위와 같이 등록되었음을 증명합니다.



2014년 09월 17일  
한국저작권위원회  
국립현대미술관  
한국문화재청



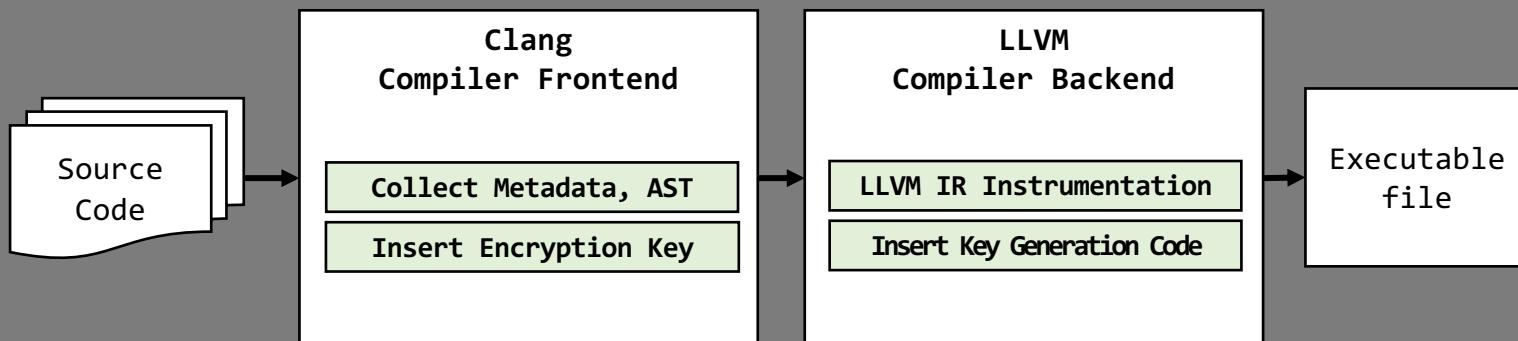
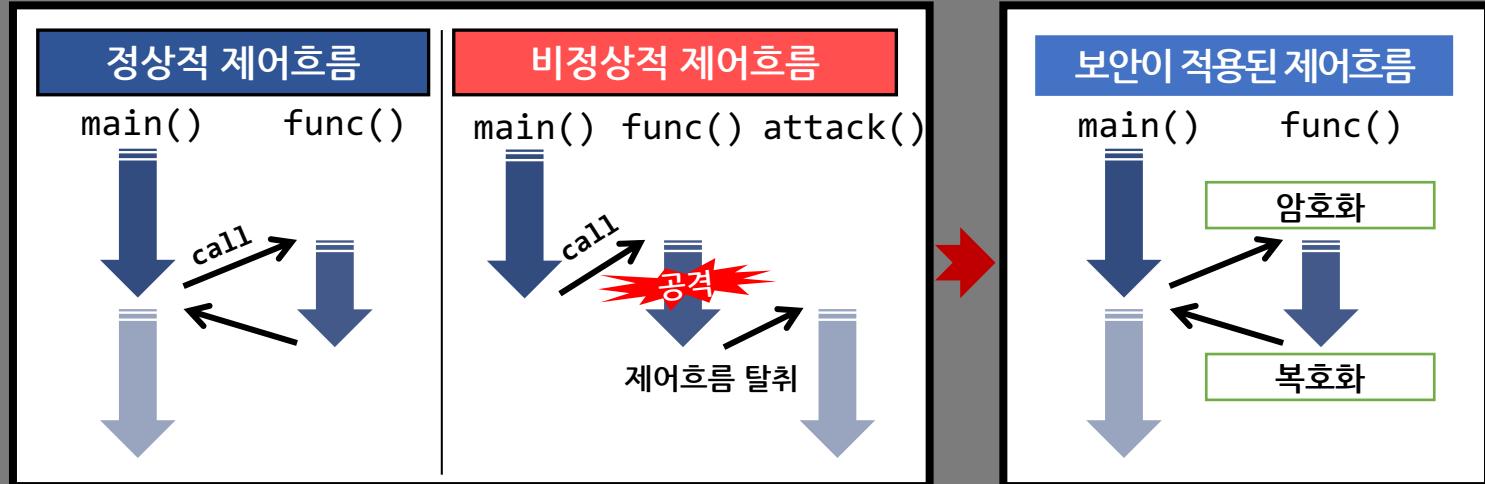
## 연구 분야 2-2

### 프로그램 보안(방어)

- Control-Flow Data Protection
  - 실행시간에 제어흐름 데이터 암호화, 제어흐름 데이터 사용시 복호화
  - 암호 키는 프로그램 실행시간에 변경
  - 맑은 업무

Collect Metadata in AST phase  
LLVM IR Instrumentation(LLVM IR Pass)

- Framework
  - Clang
  - LLVM
  - Compiler-RT
  - LibTooling



## 연구 분야 2-2

### 프로그램 보안(방어)

- VTable Pointer Protection

- 실행시간에 VTable Pointer 생성 시 암호화, VTable Pointer 사용 시 복호화
- 암호 키는 객체마다 unique 한 난수
- 사용 기술

- Clang

- Source Code Rewriter**
  - Class Hierarchy Collector**

- LLVM

- LLVM IR Instrumentation**

- Compiler-RT(Runtime Library)

- Pseudorandom Number Generator**
  - Use Intel rrand opcode**

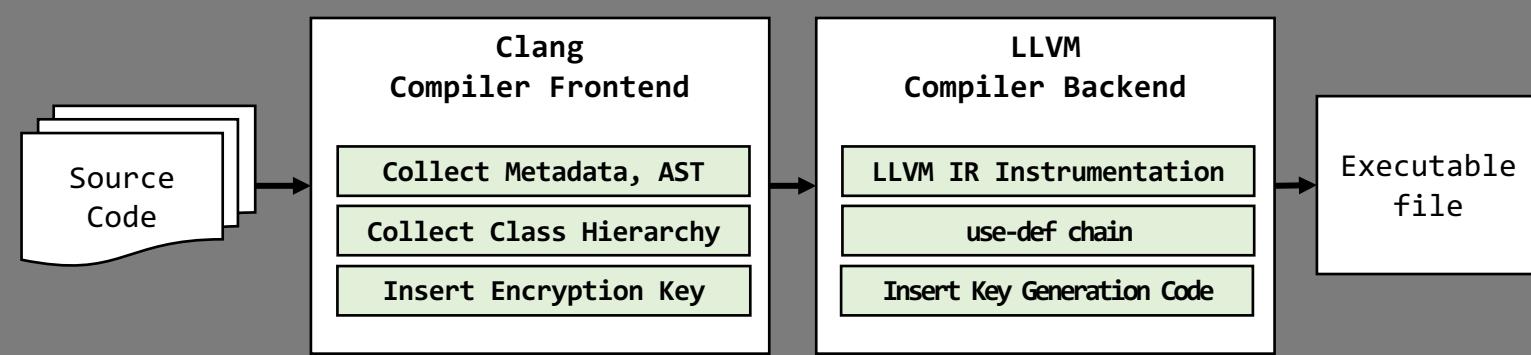
- Framework

- Clang**

- LLVM**

- Compiler-RT**

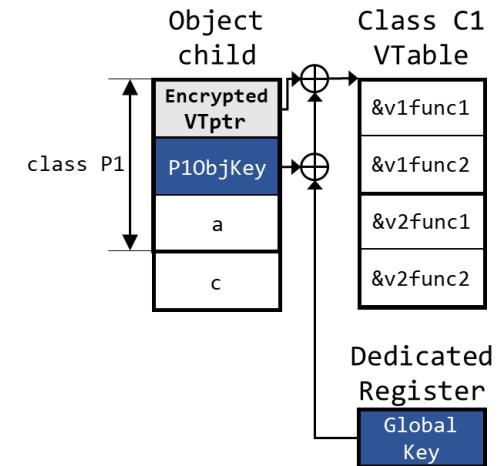
- LibTooling**



#### Source Code

```
1. class P1 {
2.     public:
3.     virtual void v1func1();
4.     virtual void v1func2();
5.     int a;
6.     unsigned long long P1objKey;
7. };
8. class C1 : public P1 {
9.     public:
10.    virtual void v2func1();
11.    virtual void v2func2();
12.    int c;
13. };
14. int main(int argc, char **argv){
15.     C1 *child = new C1();
16.     ...
17.     return 0;
18. }
```

#### Memory Layout



# 연구 분야 2-2

## 프로그램 보안(방어)

### • 참여 프로젝트

- **프로그램 실행 흐름의 처방적 통제 및 응용.**  
한국연구재단(교육부 지원과제).  
2015.04~2018.10
- **고품질 융합 소프트웨어 개발 지원 도구 연구.**  
정보통신기술진흥센터(미래창조과학부 지원  
과제).  
2013.04~2016.12

### • 연구 결과물

- **논문 1건, 프로그램 등록 2건, 특허 출원 1건**  
LLVM 환경에서 가상 함수 테이블 포인터 인코딩,  
홍익대학교, 2016.

**가상 테이블 포인터 인코딩 기법을 적용한 LLVM(엘엘브이엠) 컴파일러,** 한국저작권위원회  
(C-2015-025903).

**개선된 가상 함수 테이블 포인터 인코딩을 적용한 LLVM(엘엘브이엠) 컴파일러,** 한국저작권위원회  
(C-2016-022293).

**가상 함수 테이블 포인터 암호화 시스템 및 그 방법,** 출원번호 10-2016-0183304.

석사학위논문

LLVM 환경에서

가상 함수 테이블 포인터 인코딩

Virtual Function Table Pointer Encoding  
in LLVM Environment

홍익대학교 대학원

컴퓨터공학과 컴퓨터공학 전공

이 담 호

2016년 2월

출원번호통지서

제이지 1/4

관인생략

출원번호통지서

출원일자 2016.12.30  
특기사항 심사청구(유) 공개신청(무)  
출원번호 10-2016-0183304 (접수번호 1-1-2016-1294976-76)  
출원인 명칭 홍익대학교 산학협력단(2-2006-051462-3)  
대리인 성명 특허법인 세원(9-2011-100121-1)  
발명자 성명 표창우 이달호  
발명의 명칭 가상 함수 테이블 포인터 암호화 시스템 및 그 방법

특허청장

<<안내>>

1. 귀하의 출원은 위와 같이 정식적으로 접수되었으며, 이후의 심사 진행상황은 출원번호를 통해 확인하실 수 있습니다.  
2. 출원번호로 접수일로부터 다음날까지 등록된 날입영수증에 성명, 날짜자본 등을 기재하여 가까운 우체국 또는 은행에 납부하여야 합니다.  
※ 날짜자본호: 0111(기본코드)+접수번호  
3. 귀하의 주소, 연락처 등의 변경사항이 있을 경우, 즉시 특허고객번호 정보변경(경정), 정정신고서를 출원하여 출원 이후의 각종 통지서를 정식적으로 받을 수 있습니다.  
※ 특허고객번호: http://www.kipo.go.kr/특허고객번호선택.html  
4. 특허(实用新型)등록 출원은 명세서 또는 도면의 보정이 필요한 경우, 등록결정 이전 또는 의견서 제출기간 이내에 출원서에 최초로 첨부된 명세서 또는 도면에 기재된 사항의 범위 안에서 보정할 수 있습니다.  
5. 외국으로 출원하고자 하는 경우 PCT 제도(특허·실용신안)나 마드리드 제도(상표)를 이용할 수 있습니다. 국내출원일을 외국에서 인정받고자 하는 경우에 국외로 출원하는 경우 출원일로부터 일정한 기간 내에 외국에 출원하여 우편전송을 인정받을 수 있습니다.  
※ 제도 인원: http://www.kipo.go.kr/특허고객번호선택.html  
※ 우편전송기간: 특허·실용신안은 12개월, 상표·디자인은 6개월 이내  
※ 미국특허청은 출원일로부터 16개월 이내에 미국특허청으로부터 출원서, 출원인의 미국개설대리인에게 우편으로 출원서를 제출하여야 합니다.

제 C-2015-025903 호

프로그램 등록증

1. 프로그램의 제호 가상 테이블 포인터 인코딩 기법을 적용한 LLVM(엘엘브이엠) 컴파일러  
(명칭)  
2. 저작자·설명 고려대학교 산학협력단  
(법인명)  
서울특별시 성북구 안암로 114471-0002565  
(법인등록번호)

4. 창작연월일 2015년10월20일  
5. 출표연월일 2015년10월28일  
6. 등록사항 저작자 : 고려대학교 산학협력단,  
창작 : 2015.10.20, 출표 : 2015.10.28  
7. 등록연월일 2015년11월04일

「저작권법」 제53조에 따라 위와 같이 등록되었음을 증명합니다.

2015년 11월 05일

제 C-2016-022293 호

프로그램 등록증

1. 프로그램의 제호 개선된 가상 함수 테이블 포인터 인코딩을 적용한 LLVM(엘엘브이엠) 컴파일러  
(명칭)  
2. 저작자·설명 홍익대학교 산학협력단  
(법인명)  
서울특별시 마포구 화우산로 270171-0011448  
(법인등록번호)

4. 창작연월일 2016년08월24일  
5. 출표연월일 -  
6. 등록사항 저작자 : 홍익대학교 산학협력단, 창작 : 2016.08.24  
7. 등록연월일 2016년09월19일

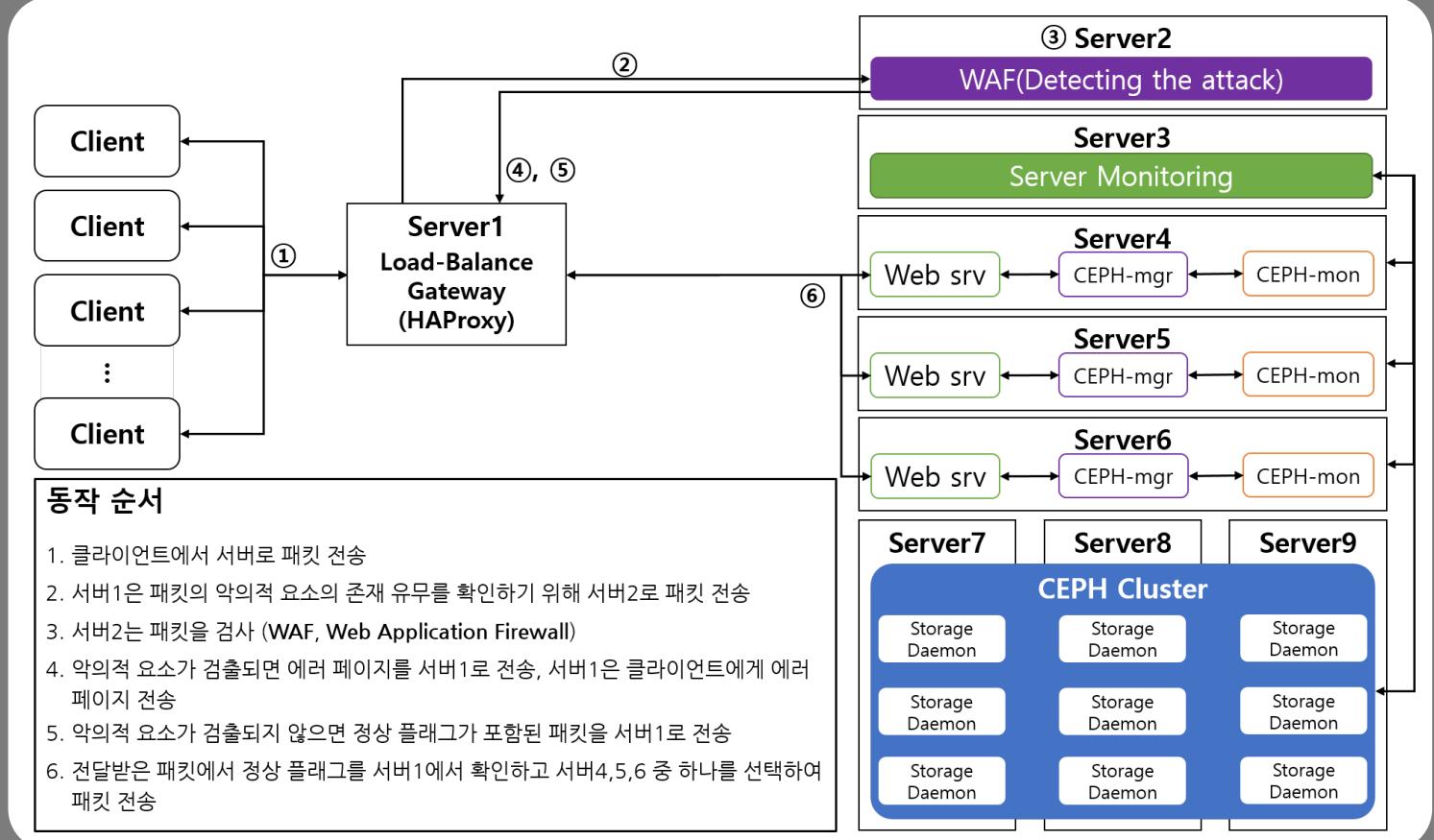
「저작권법」 제53조에 따라 위와 같이 등록되었음을 증명합니다.

2016년 09월 20일  
한국저작권위원회

# 연구 분야 3

## 빅데이터

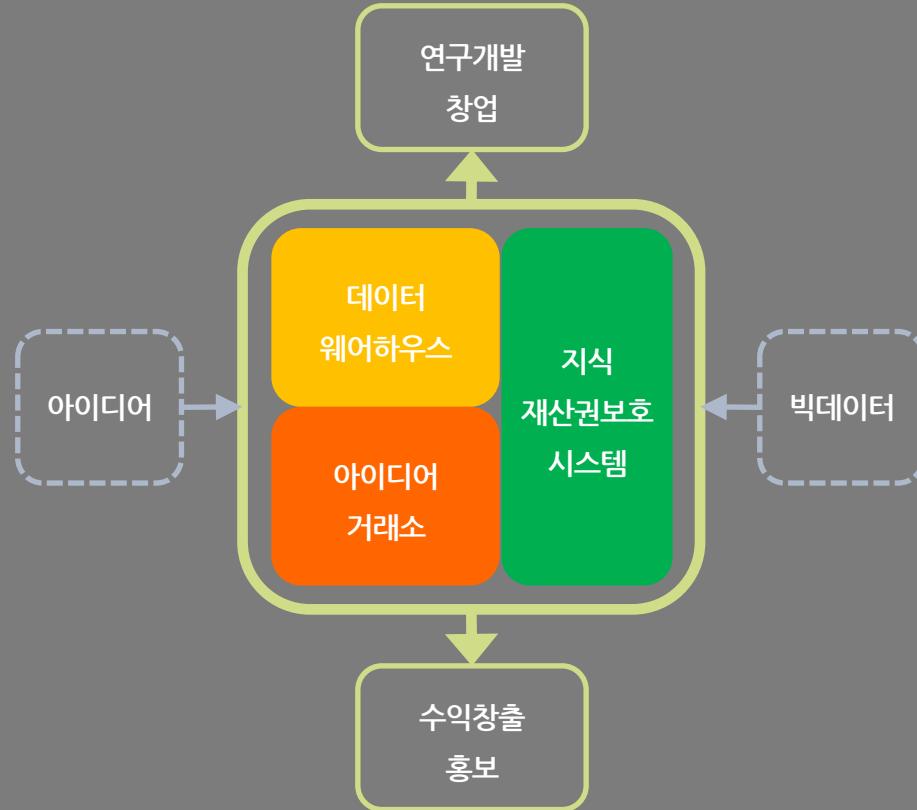
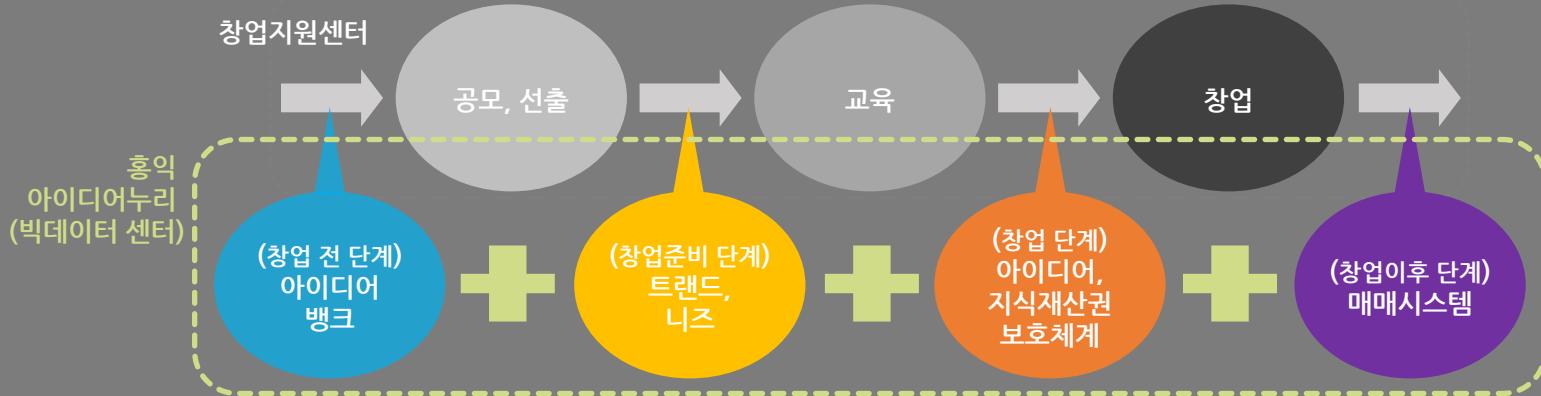
- 빅데이터 저장을 위한 서버 구축
  - Distributed Object Storage System
  - Compatible Amazon S3 Protocol 사용
  - Framework  
CEPH  
HAProxy (Load-Balance Gateway)



# 연구 분야 3

## 빅데이터

- 참여 프로젝트
  - 상상력을 디자인하는 홍대 캠퍼스타운 조성 사업. 서울특별시 캠퍼스타운 조성사업.  
2017.05.01 ~ 현재



# 관심 분야 1

## 머신러닝

### • 종목별 연관 단어를 활용한 주가 예측

- 3년간(2012~2014) 주간별 뉴스 데이터를 수집 후 주식 종목별 연관 단어 추출
- 연관 단어 유사도와 주가상승률을 매핑 후 클러스터링
- 클러스터링된 단어가 2015년도 뉴스에 나오는 경우 해당 종목의 주가상승률의 +/- 예측
- 결과

10%의 연관 단어가 포함

주가상승률 예측 정확도: 12.8%

- Framework

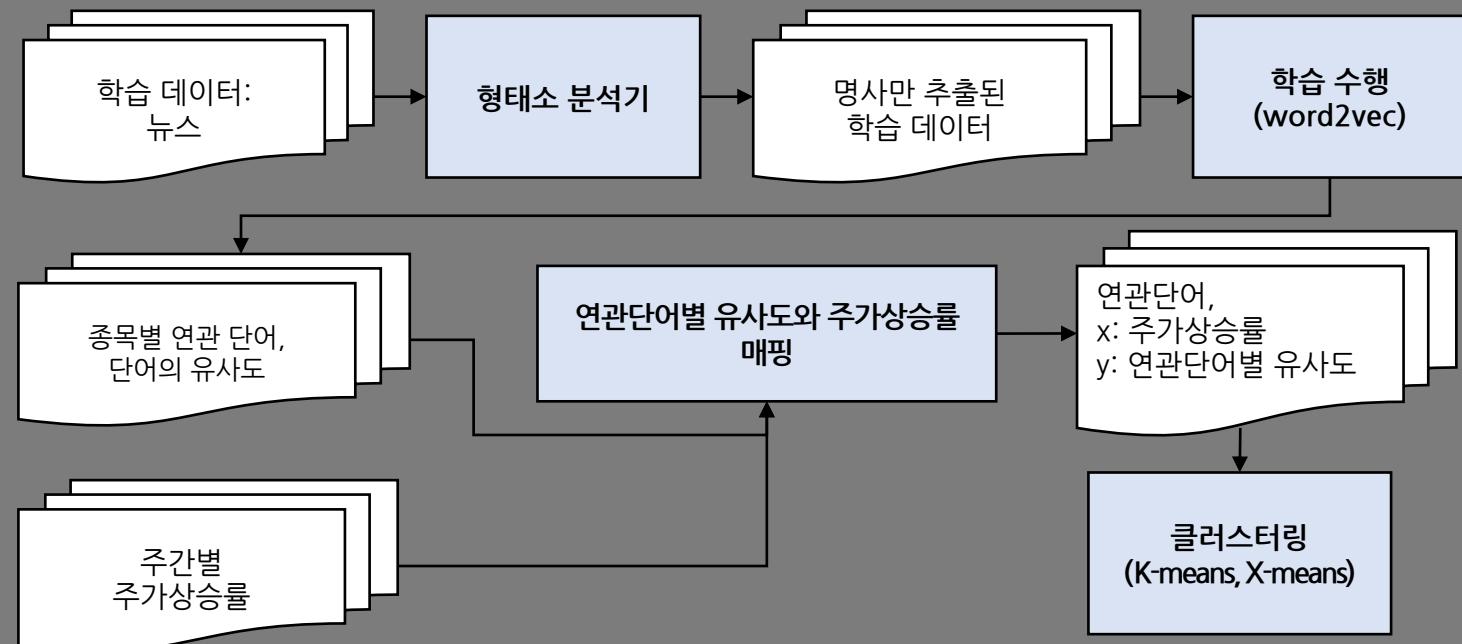
N2H4 인터넷 뉴스 크롤러

Google Finance

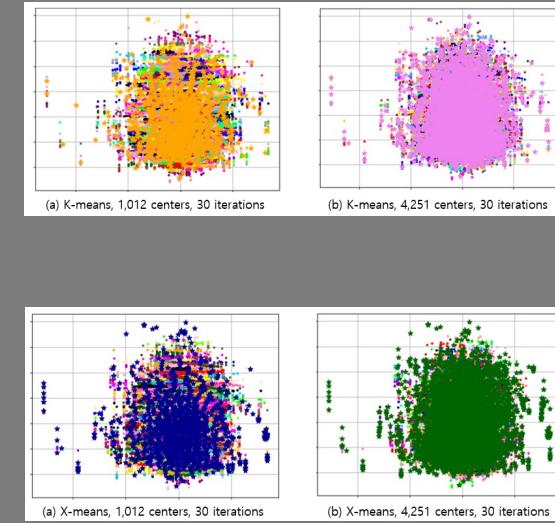
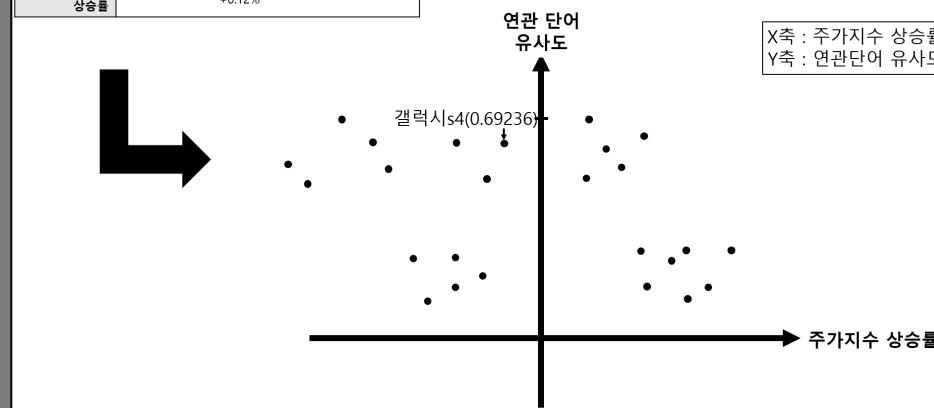
koNLPy 형태소 분석기

gensim(Python Library)

word2vec 알고리즘



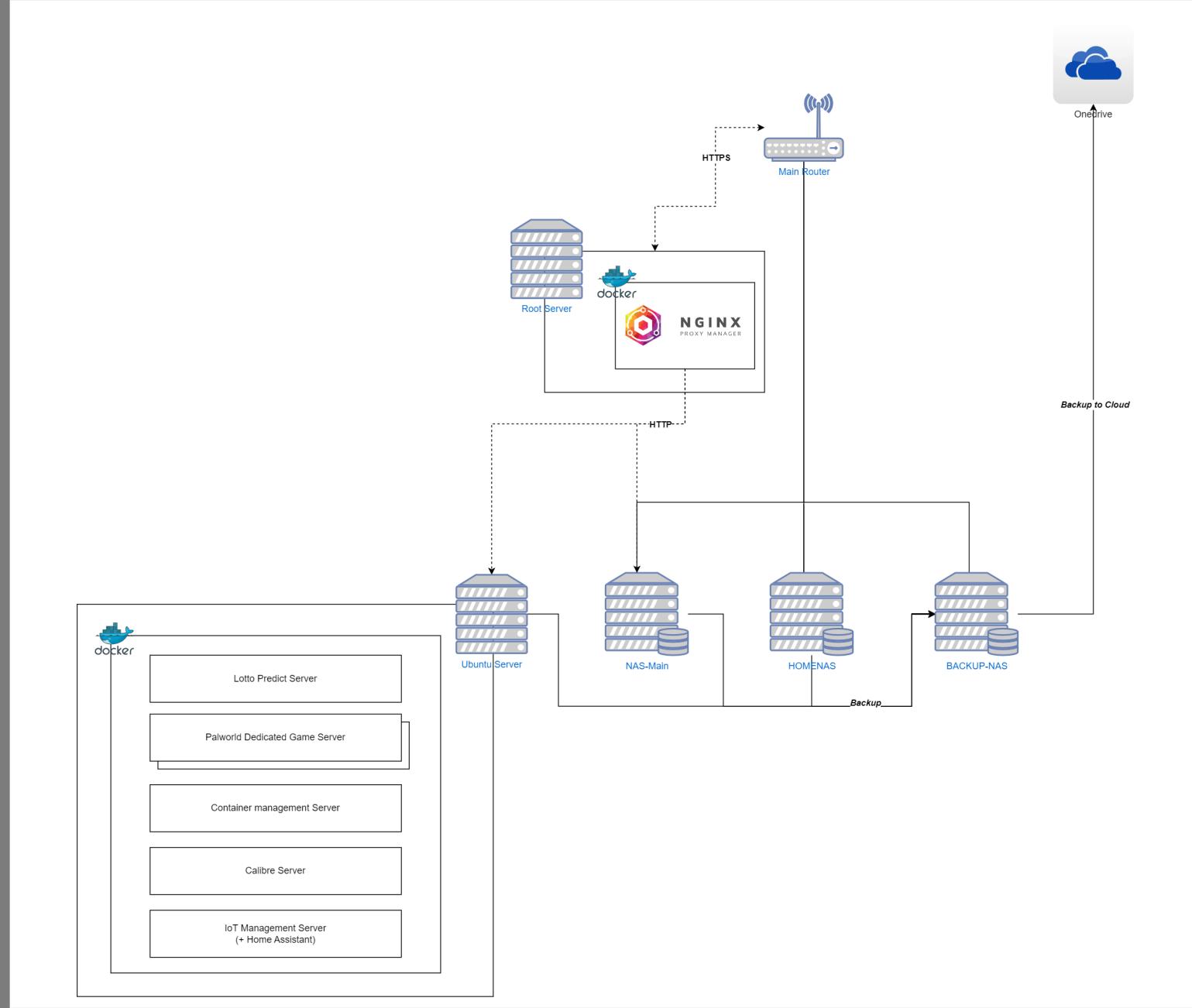
날짜	2012-01-02~2012-01-08	...		
종목	삼성전자	...		
연관 단어	갤럭시s4	IT	스마트카메라	...
유사도	0.69236	0.542123	0.491278	...
주가지수 상승률	+0.12%			...



# 관심 분야 2

## 홈 네트워크

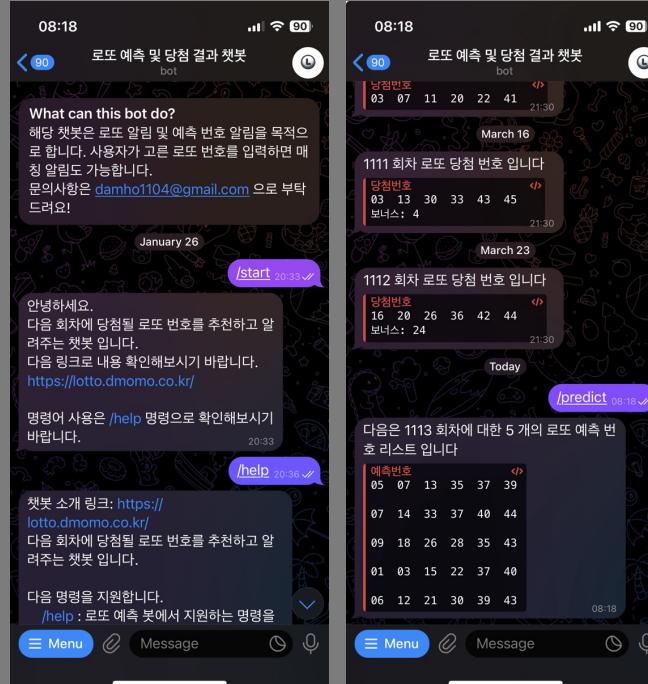
- **홈 네트워크 구축**
  - 사이드 프로젝트
    - 로또 번호 예측 Telegram 챗봇 서버
  - 운영
    - 패월드 게임 서버
    - E-book 관리 서버(Calibre Server)
    - Nginx Proxy 서버
    - 영상 관리 Plex 서버
    - IoT 관리 서버
    - 중요 데이터(가족 영상, 사진, ...) 백업 시스템
  - 사용 기술
    - Docker
    - Python
    - Bash Shell Script
    - HTTP
    - DNS



# 관심 분야 3

## 사이드 프로젝트

- **로또 번호 예측 Telegram 챗봇 서버**
  - 기능
    - 최신 회차 로또 번호 수집
    - 전체 당첨 번호 기반 딥러닝을 통한 번호 예측
    - AI를 사용한 번호 예측
      - OpenAI GPT Fine-tuning
      - Anthropic Claude3 Opus
    - Telegram 챗봇을 통한 서비스 지원
      - 사용자 관리
      - 다음 회차 번호 추천
      - 당첨 번호, 사용자 지정 로또 번호 당첨 결과 알림
  - 사용 기술
    - Docker
    - Python
    - FastAPI
    - Uvicorn
    - Tensorflow
    - APScheduler



**Lotto Predict Server Container**

**FastAPI**

**Korea Lotto Recommender**

최신 회차인 1113회의 로또 번호는 16, 20, 26, 36, 42, 44입니다. 다음 회차인 1114회 로또 번호를 예측하기 위해 이전 당첨 번호들의 통계적 분석을 바탕으로 예측 번호를 제시하겠습니다.

이전 번호들을 충분히 봐도 최근 추세, 그리고 번호 간의 조합 패턴을 분석하여 다음과 같이 10개의 번호 조합을 예측합니다. 이 예측은 확률적인 분석에 기반하며, 당첨을 보장하지는 않습니다.

- 05, 12, 23, 28, 34, 41
- 07, 15, 21, 29, 33, 42
- 02, 09, 16, 25, 37, 44
- 04, 11, 19, 27, 36, 45
- 06, 13, 20, 32, 38, 43
- 08, 14, 22, 30, 39, 40
- 03, 10, 18, 26, 35, 41
- 01, 17, 24, 31, 33, 46
- 09, 16, 23, 29, 34, 42
- 02, 08, 15, 21, 36, 44

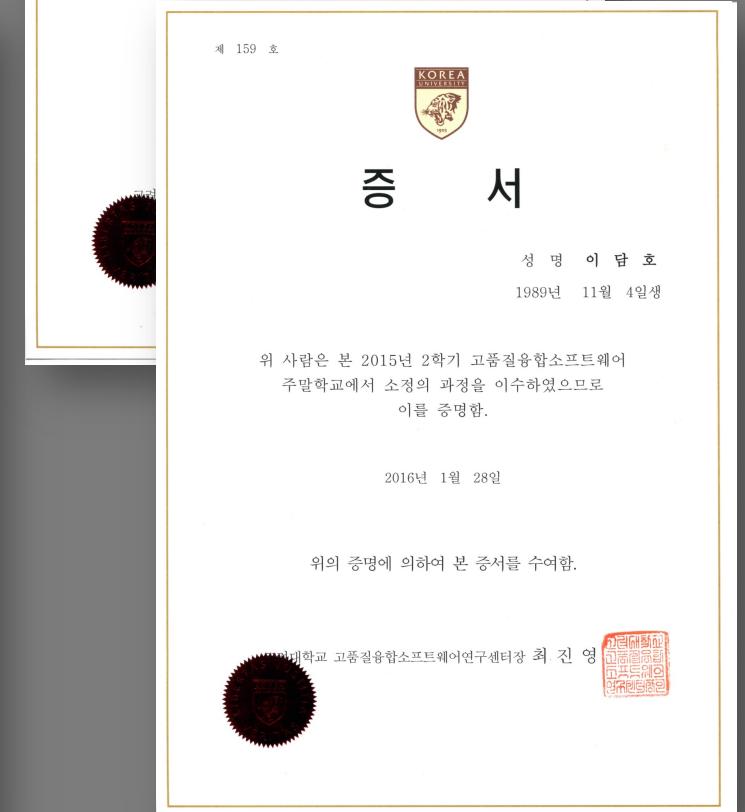
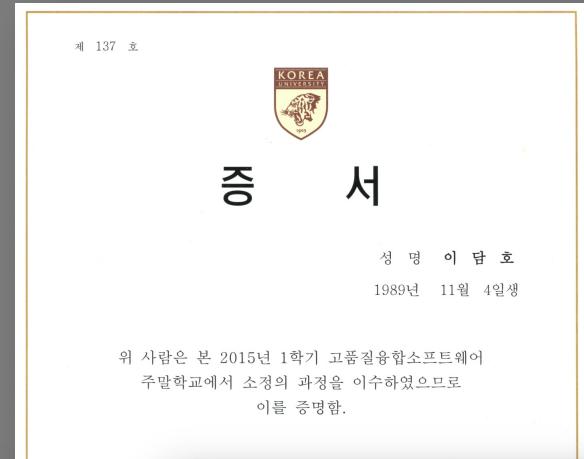
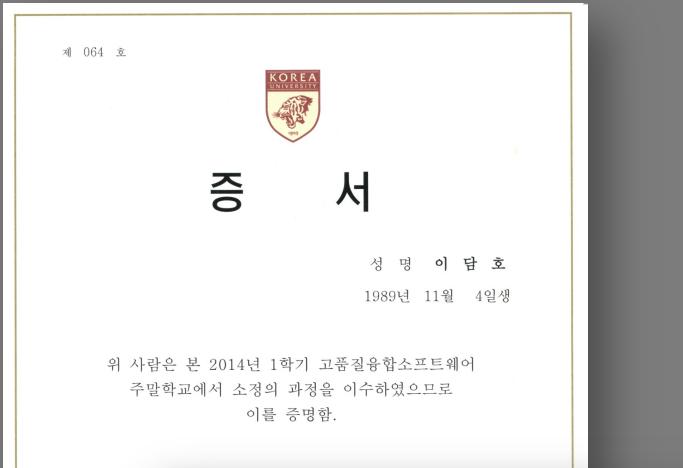
각 조합은 무작위성과 통계적 분석을 기반으로 생성되었으나, 참고용으로 활용해주시기 바랍니다.  
추첨 결과는 운과 확률에 의해 결정되므로, 책임감 있는 로또 구매를 당부드립니다.

예측 번호 뿐만 아니라 당첨 일정 기능까지 사용하시려면 다음 링크를 참고해보세요.  
<https://lotto.dmmomo.co.kr>

# 교육 이수

## 고품질 융합소프트웨어 주말학교

- “고품질 융합 소프트웨어 개발 지원 도구 연구” 프로젝트에서 개최
- 기간: 2014.03 ~ 2016.01
- 서로 다른 분야의 기초 과정 교육 이수
  - 프로그램 정적 분석
  - 프로그램 테스팅
  - 프로그램 보안 취약점 탐지 방안



# 교육 이수

## Completion Kauffman FastTrac

- “고품질 융합 소프트웨어 개발 지원 도구 연구” 프로젝트에서 진행한 기업가 정신 교육
- 기간: 2014.07.14 ~ 16
- 기업을 확장하거나 창업을 하는데 있어 필요한 기업가 정신을 전문가에게 지도 및 검증
- 실적
  - 특허 1건 출원
  - 리뷰 등록 서버 및 리뷰 등록 방법, 2014.



### Certificate of Completion Kauffman FastTrac®

Presented To:

Damho Lee

For successfully completing FastTrac® Planning the Entrepreneurial Venture™ Program

Kauffman FastTrac® - KEF : No. 040

Presented on 16, July, 2014

Ph.D Yon Ho Yu, Facilitator

Pyoung Hee Choi  
Managing Director of Venture CEO Mentoring Ce

관인생략

출원번호통지서

출원일자 2014.10.02  
특기사항 심사청구(유) 공개신청(우)  
출원번호 10-2014-0132974 접수번호 1-1-2014-0943561-85  
출원인명 체 고려대학교 산학협력단(2-2004-017068-0)  
대리인성명 김동률(9-2012-000021-4)  
발명자성명 김미경 김광연 강현자 박무규 기영준 연광호 김현민 이노 이환  
택 김문학 이담호 김경민 임지은  
발명의명칭 리뷰등록서버 및 리뷰등록방법

특허청장

<<안내>>

1. 귀하의 출원은 위와 같은 절차적으로 접수되었으며, 이후의 심사 진행상황은 출원번호를 통해 확인하실 수 있습니다.  
2. 출원에 따른 수수료는 접수일로부터 다음날까지 동봉된 납입영수증에 명기, 납부번호 등을 기재하여 기재한 후 우체국 또는 은행에 납부하여야 합니다.  
※ 납부기번호: 01311(파크원드)+접수번호  
3. 귀하의 주소, 연락처 등의 변경사항이 있을 경우, 즉시 [출원인코드 정보변경(경정), 정정신고서]를 제출하여야 출원 이후의 각종 통지서를 정상적으로 받을 수 있습니다.  
※ 특허청(patent.go.kr) 접속 > 미리보기서식으로드록한법 시행규칙 찾기 제9호 서식  
4. 특허(실용신안등록)출원은 명세서 또는 도면의 보정이 필요할 경우, 등록결정 이전 또는 의견서 제출기간 이내에 출원서에 최초로 첨부된 명세서 또는 도면에 기재된 사항의 범위 안에서 보정할 수 있습니다.  
5. 외국으로 출원하고자 하는 경우 PCT 제도(특허·실용신안이나 마드리드 제도(상표)를 이용할 수 있습니다. 국내출원인을 외국에서 인정받고자 하는 경우에는 국내 출원일로부터 일정한 기간 내에 외국에 출원하여야 우선권을 인정받을 수 있습니다.  
※ 제도안내 : <http://www.kipo.go.kr/특허마당-PCT/제도안내>  
※ 우선권 인정기준 : 한·실용신안은 12개월, 상표는 6개월 이내  
※ 출원인은 출원일로부터 일정한 기간 내에 출원인으로서의 자격을 증명할 때, 시장의 미국개설되어, 우선으로부터 16개월 이내에 미국특허심사청탁하여 [국제적교환허가서(PTO/SB-39)]를 제출하거나  
나라에 우편으로 출원서류를 제출하여야 합니다.

<http://www.patent.go.kr/jsp/kiponet/ir/receipt/online/appNoOffcAct.so>

2014.10.02

# 교육 이수

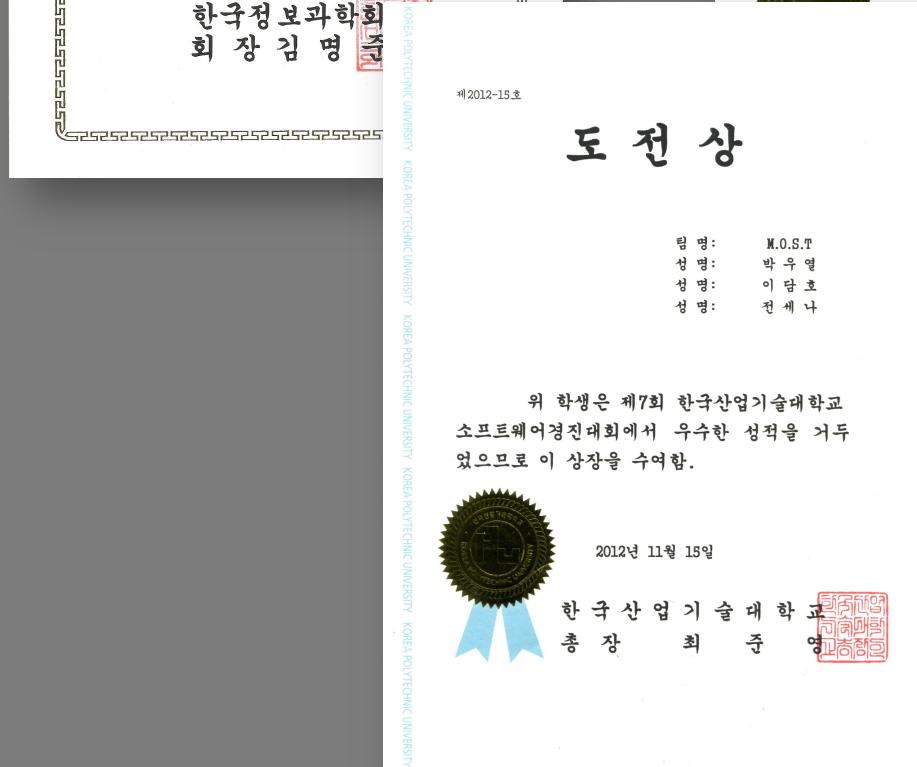
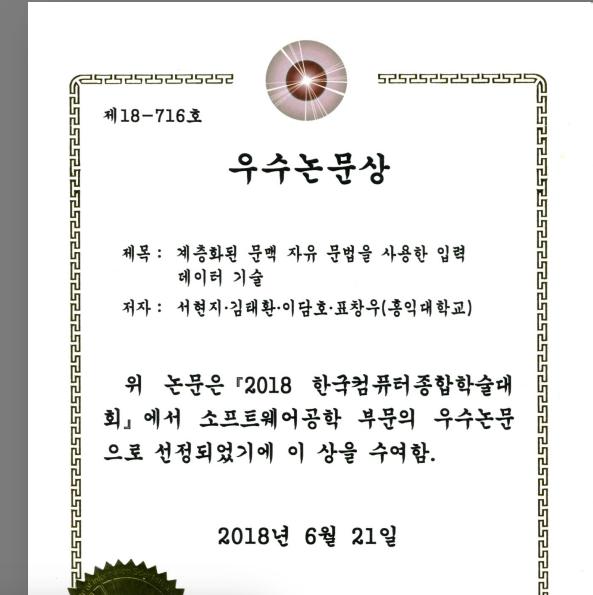
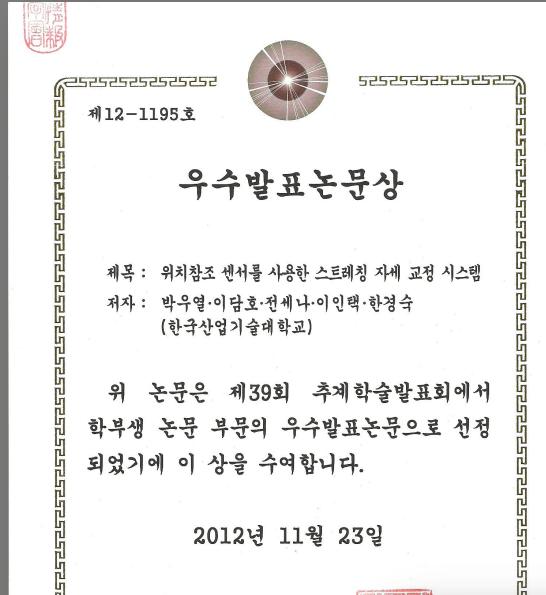
## Oracle Certified Professional

- Oracle Database 10g Administrator  
교육 이수
- 기간: 2013.09 ~ 2013.10



# 수상 경력

- 정보과학회 우수논문상 1건
- 정보과학회 우수발표논문상 1건
- 한국산업기술대학교 소프트웨어경진 대회 도전상 1건



# 수상 경력

- 슈어소프트테크 슈퍼히어로(우수사원상)
  - 2년 연속 수상
    - 2022
    - 2021

