

과제 개요

2023 차세대 보안리더 양성 프로그램 화이트햇 스쿨 1기

<https://docs.google.com/spreadsheets/d/1MpW4AEuwX2UpZluTP6cEY5l2qcPLdLyzx8amnuJxhY/edit?usp=sharing>

과제

<도전! AST 맞보기>

대상:

- 1) AST를 추출할 C언어 기반의 프로그램 (교육생 제작) 또는 오픈소스 코드 활용
- 2) AST를 분석할 C언어 기반의 프로그램 (교육생 제작)
- 3) AST를 추출한 python언어 기반의 프로그램 (멘토 제공) – generate_ast.py
pip install pycparser 후에 실행

내용:

- 한 파일 기준으로 Line of Code (LOC)가 500줄 이상인 C언어 소스코드 또는 자체제작 소스코드
- AST 구조 분석기 제작
 - <기본>
 - => 함수 개수 추출하기
 - => 함수들의 리턴타입 추출하기
 - => 함수들의 이름 추출하기
 - => 함수들의 파라미터 타입, 변수명 추출하기
 - => 함수들의 if조건 개수 추출하기

제출기한: 10/15 (일) 23:59:59

제출파일: .zip파일을 개인클라우드에 업로드 후 링크를 외부공개하여 공유

- 1,2,3 소스파일

- ast 파일

- 대상소스코드정보 (소스코드, 프로젝트 URL)

- 사진+설명(대충) 담긴 PDF 파일

제출메일: danpoong@g.skku.edu

메일제목:

[화이트햇스쿨_1기_프로그래밍기초_과제] [팀명]

[화이트햇스쿨_1기_프로그래밍기초_과제] [개인명]

<심화>

- => github에서 버그를 포함하는 소스코드의 AST를 추출해서 AST기반 버그패턴화 제작한 버그패턴이 다른 프로젝트의 소스코드에서도 1개라도 검출되는지 확인

과제 파일

1. test.c 파일 -> 1. AST 추출대상인 C언어 기반의 프로그램
2. ast_function_information_parser.c 파일 -> 2. AST를 분석할 C언어 기반의 프로그램
3. generate_ast.py 파일 -> 3. AST를 추출할 파이썬 언어 기반의 프로그램
4. json_c.c 파일 -> 4. ast_function_information_parser.c를 구동시킬 json 추출 라이브러리
5. result.json 파일 -> AST 트리 구조를 담은 json파일(AST파일)

보고서 포함 총 6개의 파일로 구성된다.

테스트 케이스(테스트 함수)

1. gaussianrandom 함수

```
unsigned long int gaussianrandom(unsigned long int stddev, unsigned long int avr)
{
    unsigned long int x1, x2, y1;

    x1 = (unsigned long int)rand() / RAND_MAX;
    x2 = (unsigned long int)rand() / RAND_MAX;
    y1 = sqrt(-2 * log(x1)) * cos(2 * M_PI * x2);
    y1 = (stddev * y1) + avr;

    return y1;
}
```

함수의 리턴 타입과 파라미터 변수의 타입이 unsigned long int로 3개의 단어로 되어있는 변수명도 나타낼 수 있는지를 테스트 하기 위함이다.

2. if_function 함수

```
unsigned long int if_function(float first, float second)
{
    if (first > 0)
    {
        return 0;
    }
    else if (first == 0)
    {
        return 1;
    }
    else if (first < 0)
    {
        if (first == -1)
        {
            if (first == -1)
            {
                if (first == -1)
                {
                    if (first == -1)
                    {
                        return 4;
                    }
                }
            }
        }
        return 2;
    }
    if (first > 100)
    {
        if (first == -1)
        {
            if (first == -1)
            {
                if (first == -1)
                {
                    if (first == -1)
                    {

```

함수에서 사용된 if문의 개수를 잘 출력할 수 있는 지 확인하기 위한 테스트 함수이다.
총 15개의 if문이 사용되었다.

3. main 함수

[illegible]

파라미터가 없는 함수를 테스트하기 위해서 또한 while문, for문, 중첩 for 문안에서 작성된 if문도 카운트 할 수 있는지 테스트하기 위함이다. 총 11개의 if문이 사용되어 있다.

예상 테스트 결과

결과 출력

결과 출력

함수의 총 개수 : 3

1 번째 함수 이름 : gaussianrandom

1 번째 함수 리턴타입 : unsigned long int

1 번째 함수에서 사용된 if문 개수 : 0

1 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - stddev 파라미터 타입 - unsigned long int

파라미터 이름 - avr 파라미터 타입 - unsigned long int

2 번째 함수 이름 : if_function

2 번째 함수 리턴타입 : unsigned long int

2 번째 함수에서 사용된 if문 개수 : 15

2 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - first 파라미터 타입 - float

파라미터 이름 - second 파라미터 타입 - float

3 번째 함수 이름 : main

3 번째 함수 리턴타입 : int

3 번째 함수에서 사용된 if문 개수 : 11

3 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - parameter로 사용된 변수 없음 파라미터 타입 - parameter로 사용된 변수 없음

다음과 같은 형식과 값으로 출력되어야 한다.

실제 테스트 결과

결과 출력

함수의 총 개수 : 3

1 번째 함수 이름 : gaussianrandom

1 번째 함수 리턴타입 : unsigned long int

1 번째 함수에서 사용된 if문 개수 : 0

1 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - stddev 파라미터 타입 - unsigned long int

파라미터 이름 - avr 파라미터 타입 - unsigned long int

2 번째 함수 이름 : if_function

2 번째 함수 리턴타입 : unsigned long int

2 번째 함수에서 사용된 if문 개수 : 15

2 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - first 파라미터 타입 - float

파라미터 이름 - second 파라미터 타입 - float

3 번째 함수 이름 : main

3 번째 함수 리턴타입 : int

3 번째 함수에서 사용된 if문 개수 : 11

3 함수에서 사용된 파라미터 이름과 파라미터 타입 정보 :

파라미터 이름 - parameter로 사용된 변수 없음 파라미터 타입 - parameter로 사용된 변수 없음

 터미널이 작업에서 다시 사용됩니다. 닫으려면 아무 키나 누르세요.