



국민대학교
전자정보통신대학
컴퓨터공학부

캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	MASK(Malware Analysis System in Kookmin)
팀 명	NCNP
문서 제목	중간보고서

Version	1.4
Date	2018-04-12

팀원	한 채연 (조장)
	김 영재
	명 준우
	이 유정
	허 준녕
지도교수	윤 명근 교수

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 MASK(Malware Analysis System in Kookmin) 를 수행하는 팀 NCNP의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 NCNP의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역


Filename	중간보고서-NCNP.doc
원안작성자	김영재, 명준우, 이유정, 한채연, 허준녕
수정작업자	한채연

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2018-04-10	한채연	1.0	최초 작성	초안 작성
2018-04-11	허준녕, 이유정	1.1	내용 추가	데이터 처리 관련 내용 추가
2018-04-12	명준우	1.2	내용 추가	라벨링 관련 내용 추가
2018-04-12	김영재	1.3	내용 추가	정적 분석 및 디버깅 관련 내용 추가
2018-04-12	한채연	1.4	내용 수정	오타자 수정 및 내용 검토

	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

목 차

1	프로젝트 목표	4
2	수행 내용 및 중간결과	5
2.1	계획서 상의 연구내용	5
2.1.1	정적 분석	5
2.1.2	동적 분석	6
2.1.3	라벨링	7
2.1.4	딤러닝	8
2.1.5	데이터 처리	9
2.1.6	Scenario 1 – 데이터베이스에 분석 결과가 존재하는 경우	10
2.1.7	Scenario 2 – 데이터베이스에 분석 결과가 존재하지 않는 경우	11
2.2	수행내용	12
2.2.1	정적 분석	12
2.2.2	동적 분석	12
2.2.3	라벨링	18
2.2.4	딤러닝	20
2.2.5	데이터 처리	21
2.2.6	웹	23
3	수정된 연구내용 및 추진 방향	27
3.1	수정사항	27
3.1.1	크롤링 자동화	27
4	향후 추진계획	28
4.1	향후 계획의 세부 내용	28
4.1.1	동적 분석	28
4.1.2	딤러닝 - 하이브리드 모델 개발	28
4.1.3	웹	28
5	고충 및 건의사항	30

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

1 프로젝트 목표

안티바이러스 테스트 업체 AV-TEST에 따르면 해마다 발견되는 악성코드는 늘어나는 추세이며, 현재 하루에 약 100만 개 정도의 악성코드가 발견되고 있다. 반면 악성코드 전문가의 수는 현격히 부족하다. 또한, 악성코드가 더욱 복잡해지고 정교해지면서 안티바이러스 제품이나 기존 탐지 솔루션은 이에 대해 효과적으로 대응하지 못한다. 따라서 본 프로젝트는 악성코드로 의심되는 파일을 분석하여 악성코드로 의심되는 파일의 정적, 동적 분석 결과와 이와 유사한 파일에 대한 분석 결과, 자체적으로 학습한 모델에 의해 결정된 악성코드의 라벨을 분석가에게 제공함으로써 보다 정교하게 악성코드를 분석할 수 있도록 하는 것을 목표로 한다. 나아가 이를 오픈소스 소프트웨어로 개발하는 것을 목표로 한다.

■ 세부 목표

- 사용자가 업로드한 파일에 대해 정적, 동적 분석을 진행하여 그 결과를 제공한다.
- 사용자가 업로드한 파일과 유사한 파일에 대한 정보를 제공한다.
- 파일에 대한 세부정보를 사용자가 한 눈에 볼 수 있도록 웹에 시각화 한다.
- 카스퍼스키 라벨을 참고하여 자체적인 바이러스 분류 기준을 세워 라벨을 제공한다.
- 주기적인 재학습을 통해 새로운 악성코드에 대해 대응을 할 수 있도록 한다.

	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

2 수행 내용 및 중간결과

2.1 계획서 상의 연구내용


사용자가 악성으로 의심되는 파일을 업로드하면, 그 파일의 md5값을 구한 후 데이터베이스에 연동된 검색엔진을 이용하여 분석 결과를 찾는다. 분석 결과가 데이터베이스에 있을 경우, 그 정보들을 웹을 통하여 사용자에게 보여준다. 분석 결과가 데이터베이스에 없을 경우, 그 파일에 대해 각각 정적, 동적 분석을 진행하고, 생성된 리포트로부터 추출된 피처를 이용하여 학습된 모델로부터 탐지 결과를 사용자에게 웹으로 보여준다. 각 세부 연구 분야는 아래 그림과 같다.



[그림 1] 연구 분야 카테고리

2.1.1 정적 분석

정적 분석이란 실제 실행 없이 파일을 분석하는 것이다. 정적 분석을 하기 위해서는 수집된 pefile만을 가지고 해당 파일이 가진 구조, 동작 등 모든 부분을 알아내야 한다. 역분석을 하기 위해 IDA Pro 툴을 이용한다. 또한 파이썬 기반의 악성코드 분석 도구로 활용되는 오픈소스 소프트웨어인 peframe을 사용하여 악의적으로 사용될 수 있는 API 정보, Anti-Debug 정보, 악성코드 여부 정보 등을 파악할 수 있다. 정적 분석을 통해 얻을 수 있는 피처로는 control-flow graph, binary sequence, mnemonic sequence 등이 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12


2.1.2 동적 분석

동적 분석이란 악성코드를 분석 환경에서 실행시킨 후 시스템의 행동 변화를 분석하는 방법이다. 악성코드를 분리된 곳에서 안전하게 실행하기 위해선 독립적인 공간이 필요하기 때문에 가상 환경을 이용한다. 그 중 악성코드 동적 분석을 위한 오픈소스 소프트웨어인 쿠쿠샌드박스(Cuckoo Sandbox)를 이용한다. 이를 이용하여 분리된 가상머신 환경에서 해당 파일을 실행한 후 어떤 행위가 일어나는지 관찰하여 구체적인 정보를 획득할 수 있다. 더 나아가 분석을 진행하게 될 guest instance를 하나 이상 구성하여 분석 효율을 높일 수 있다. 분석 후 json 포맷의 리포트가 생성되며 동적 분석을 통해 얻을 수 있는 피쳐로는 process memory, network, API Call sequence, signatures 등이 있다.

쿠쿠샌드박스는 아래와 같이 323개 함수의 API 호출을 기록하고, 자체적으로 17개의 카테고리로 분류한다.

class	description	example	# of APIs
A	file/directory	CopyFile, CreateDirectory, GetFileType, ...	47
B	registry	RegCreateKeyEx, NtCreateKey, RegDeleteValue, ...	38
C	internet explorer	CDocument_write, CScriptElement_put_src, ...	7
D	user interface	DrawText, FindWindow, LoadString, ...	11
E	net API	NetGetJoinInformation, NetShareEnum, ...	6
F	network	DnsQuery_A, GetAdaptersInfo, HttpOpenRequestA, ...	62
G	OLE	CoCreateInstance, CoInitialize, ...	3
H	process	CreateProcess, CreateThread, Module32First, ...	41
I	synchronization	GetLocalTime, GetSystemTime, ...	8
J	resource	FindResource, LoadResource, ...	6
K	services	ControlService, CreateService, ...	12
L	system	GetNativeSystemInfo, LdrLoadDll, NtClose, ...	26
M	certificate	CertControlStore, CertOpenStore, ...	5
N	encryption	CryptCreateHash, CryptGenKey, ...	19
O	exception	SetUnhandledExceptionFilter, RtlDispatchException, ...	6
P	misc	GetUserName, GetDiskFreeSpace, WriteConsole, ...	20
Q	notification	__anomaly__, __exception__, ...	4

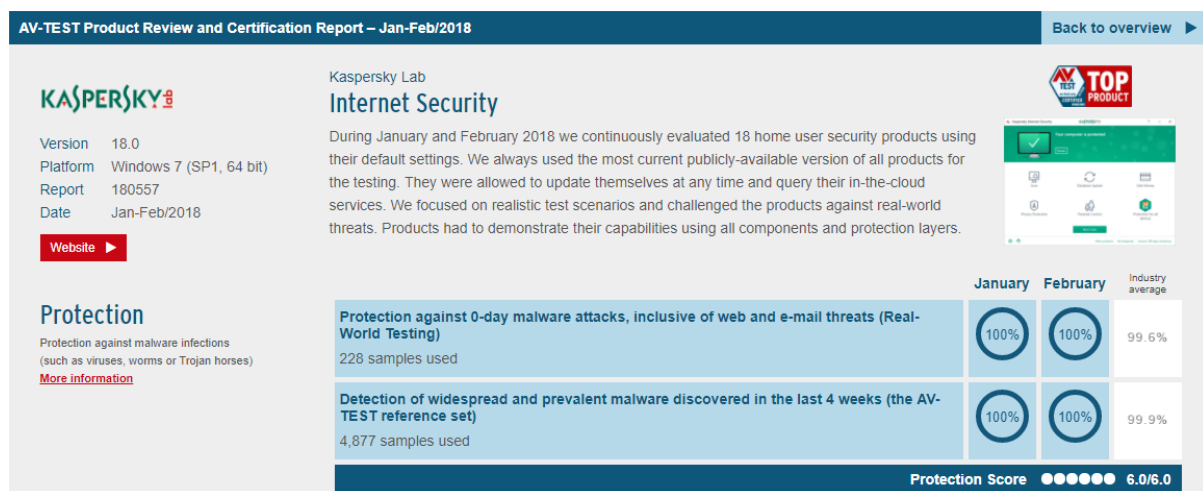
[표 1] API Table (출처 : 고동우, 김휘강(2017) "API 콜 시퀀스와 Locality Sensitive Hashing을 이용한 악성코드 클러스터링 기법에 관한 연구", 정보보호학회논문지)

	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

2.1.3 라벨링

악성코드 라벨링은 악성코드의 동적 정보와 정적 정보 등을 이용하여 특징에 따라 분류하는 방법이다. 러시아의 IT 보안 업체인 카스퍼스키 랩에서 개발한 안티바이러스인 카스퍼스키는 세계 최고의 IT 보안 테스트 및 컨설팅 서비스 제공업체인 AV-TEST에서 2016년 2월부터 현재까지 악성코드 방지점수에서 최고점수를 받고 있으며, 해외 보안 인증 VB100은 전 세계에서 실제 감염 활동이나 발견 보고가 있었던 악성코드들의 샘플 목록인 Wildlist를 오탐 없이 100% 탐지해야 인증 획득이 가능한데 카스퍼스키는 VB100 인증을 획득했다.

The best antivirus software for Windows Home User



[그림 2] AV-TEST의 Protection 평가에서 최고 점수를 받은 카스퍼스키

VB100 results from 2018-02 (latest) on Windows 7 Professional, Windows 10 Professional

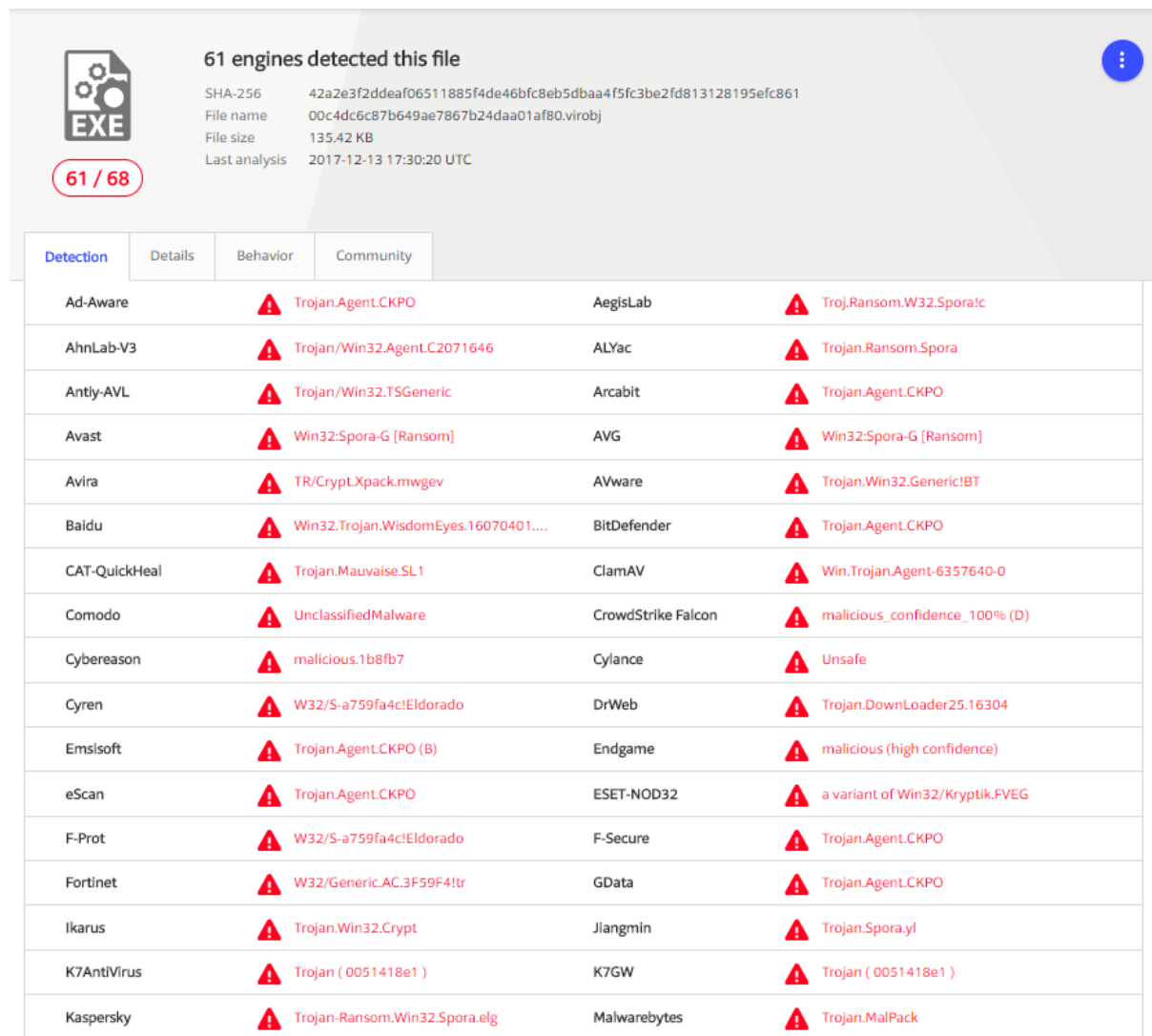
Read the full review, or download it.

Tested product	Result	RAP Overview	WildList (%)	WildList (%)	False positives	False positives
Kaspersky Lab Kaspersky Endpoint Security 10 for Windows	Passed VIRUS 100		100.00	100.00	0	0

[그림 3] VB100 인증을 획득한 카스퍼스키

구글의 자회사인 바이러스토탈은 파일에 대한 분석과 결과를 제공해주는 온라인 서비스이다. 바이러스토탈은 분석결과로 약 60여개의 안티바이러스의 탐지결과와 진단명을 제공한다. 바이러스

토탈의 기능을 간편하게 사용하기 위한 API가 제공되는데 이를 통하여 악성코드의 분석결과를 받아온다. 분석결과에서 카스퍼스키의 진단명을 받고 연구하여 이를 기반으로 독자적인 라벨을 제작한다.



61 engines detected this file

SHA-256: 42a2e3f2ddea06511885f4de46bfc8eb5dbaa4f5fc3be2fd813128195efc861
File name: 00c4dc6c87b649ae7867b24daa01af80.virobj
File size: 135.42 KB
Last analysis: 2017-12-13 17:30:20 UTC


61 / 68

Detection	Details	Behavior	Community
Ad-Aware	Trojan.Agent.CKPO	AegisLab	Troj.Ransom.W32.Spora.c
AhnLab-V3	Trojan/Win32.Agent.C2071646	ALYac	Trojan.Ransom.Spora
Antiy-AVL	Trojan/Win32.TSGeneric	Arcabit	Trojan.Agent.CKPO
Avast	Win32:Spora-G [Ransom]	AVG	Win32:Spora-G [Ransom]
Avira	TR/CryptXpack.mwgev	AVware	Trojan.Win32.Generic!BT
Baidu	Win32.Trojan.WisdomEyes.16070401....	BitDefender	Trojan.Agent.CKPO
CAT-QuickHeal	Trojan.Mauvaise.SL1	ClamAV	Win.Trojan.Agent-6357640-0
Comodo	UnclassifiedMalware	CrowdStrike Falcon	malicious_confidence_100% (D)
Cybereason	malicious.1b8fb7	Cylance	Unsafe
Cyren	W32/S-a759fa4c!Eldorado	DrWeb	Trojan.DownLoader25.16304
Emsisoft	Trojan.Agent.CKPO (B)	Endgame	malicious (high confidence)
eScan	Trojan.Agent.CKPO	ESET-NOD32	a variant of Win32/Kryptik.FVEG
F-Prot	W32/S-a759fa4c!Eldorado	F-Secure	Trojan.Agent.CKPO
Fortinet	W32/Generic.AC.3F59F41tr	GData	Trojan.Agent.CKPO
Ikarus	Trojan.Win32.Crypt	Jiangmin	Trojan.Spora.yl
K7AntiVirus	Trojan (0051418e1)	K7GW	Trojan (0051418e1)
Kaspersky	Trojan-Ransom.Win32.Spora.elg	Malwarebytes	Trojan.MalPack

[그림 4] 바이러스토탈의 파일 분석결과 화면

2.1.4 딥러닝

본 프로젝트는 정적, 동적 분석 결과로부터 피처를 생성하여 딥러닝 모델을 설계할 예정이다. 먼저 정적 분석 결과를 이용한 피처 생성 방법은 다음과 같다. IDA Pro를 이용하여 입력 파일을 idb 파일을 생성한다. 생성된 idb파일은 ida python을 이용하여 opcode sequence를 추출한 다음


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

n-gram 기법을 이용해 2, 3, 4 – gram을 만든다. 만들어진 집합을 피쳐 해싱(Feature Hashing) 기법을 이용하여 입력 벡터(Input Vector)를 제작할 예정이다. 쿠쿠샌드박스로부터 생성된 리포트에서 추출할 수 있는 API Call Sequence로부터 n-gram 기법을 이용해 2, 3, 4 – gram을 만든다. 마찬가지로 만들어진 집합을 피쳐 해싱 기법을 이용하여 입력 벡터를 제작할 예정이다. 딥러닝은 크게 3가지 모델을 만들 예정이다. 첫 번째 모델은 opcode sequence를 피쳐로 사용하는 모델, 두 번째 모델은 API Call Sequence를 피쳐로 사용하는 모델, 세 번째 모델은 opcode sequence와 API Call Sequence를 합쳐서 피쳐로 사용하는 모델이다.

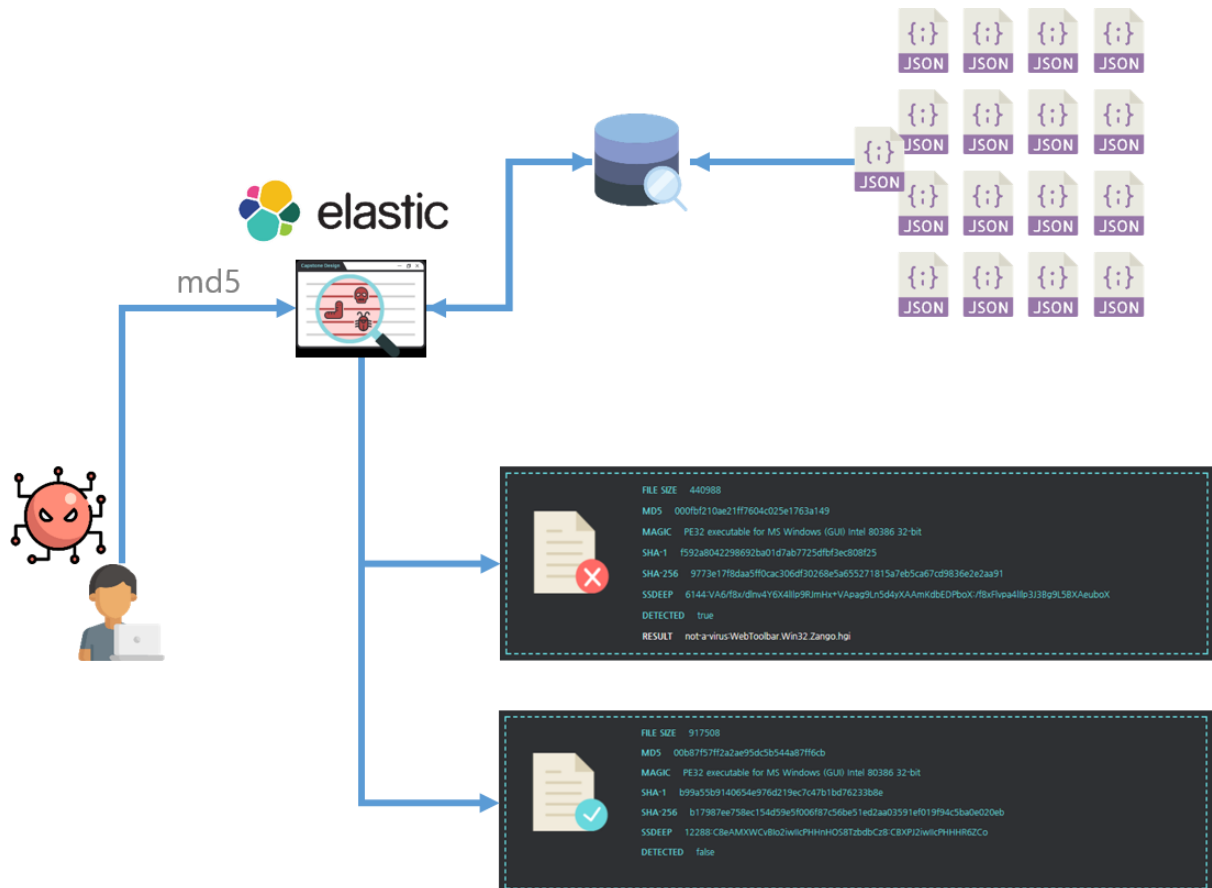
2.1.5 데이터 처리

본 프로젝트는 대량의 악성코드 데이터 샘플을 이용하여 정적 분석, 동적 분석, 라벨링, 딥러닝을 진행하게 되는데, 이러한 빅데이터를 기본적인 저장매체를 통해 관리 및 검색하는 것은 효율적이지 못한 프로젝트 진행을 유발하게 된다. 따라서 효율적인 데이터 관리 및 처리를 위해 데이터 관리 시스템과 운용 방법에 대해 연구한다.

준비 중인 서비스 중 유사한 데이터를 검색해주는 서비스는 또 하나의 데이터 처리 이슈에 해당한다. 유사도 비교를 위해 어떠한 알고리즘을 사용할 것인지 연구한다. 또한, 대량의 데이터에 대해 단순히 전체 검색을 통한 유사도 비교를 하는 것은 서비스 속도 저하를 유발하고 서버 과부하에 원인이 될 수 있다. 따라서 빠르고 성능적으로 효율적인 검색 방법 대해 연구를 진행한다.

 <div> <p>국민대학교</p> <p>컴퓨터공학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

2.1.6 Scenario 1 – 데이터베이스에 분석 결과가 존재하는 경우

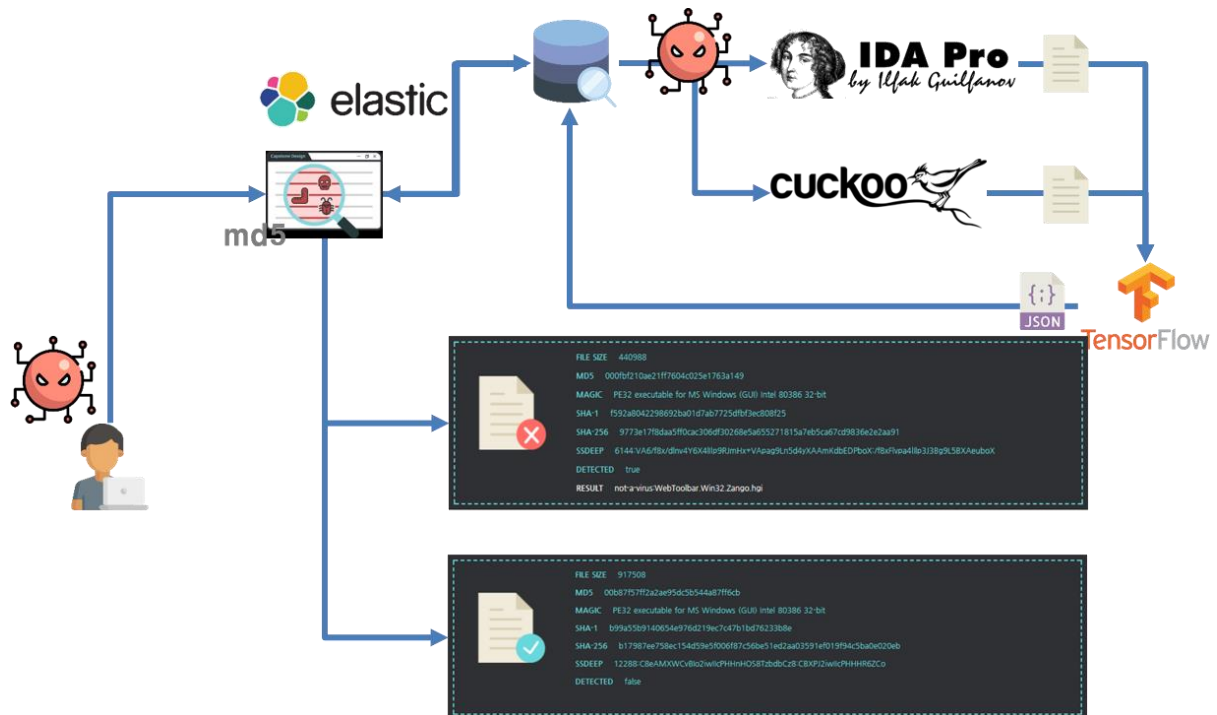


[그림 5] 데이터베이스에 분석 결과가 존재하는 경우

사용자가 악성으로 의심되는 파일을 웹에 업로드하면, 그 파일의 md5를 구한 후 데이터베이스에 연동된 엘라스틱서치를 이용해 분석 결과를 찾는다. 리포트가 데이터베이스에 존재하면 결과와 그 연관된 정보들을 웹을 통하여 사용자에게 보여준다.


 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

2.1.7 Scenario 2 – 데이터베이스에 분석 결과가 존재하지 않는 경우



[그림 6] 데이터베이스에 분석 결과가 존재하지 않는 경우

사용자가 업로드한 파일에 대한 리포트가 데이터베이스에 없으면 그 파일에 대해 각각 정적 분석과 동적 분석을 진행 후 리포트를 생성하고, 이로부터 피처를 추출한다. 추출한 피처를 이용하여 학습된 모델로부터 탐지 결과를 구한다. 이 결과들을 데이터베이스에 저장하고, 웹을 통해 사용자에게 결과를 보여준다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

2.2 수행내용


2.2.1 정적 분석

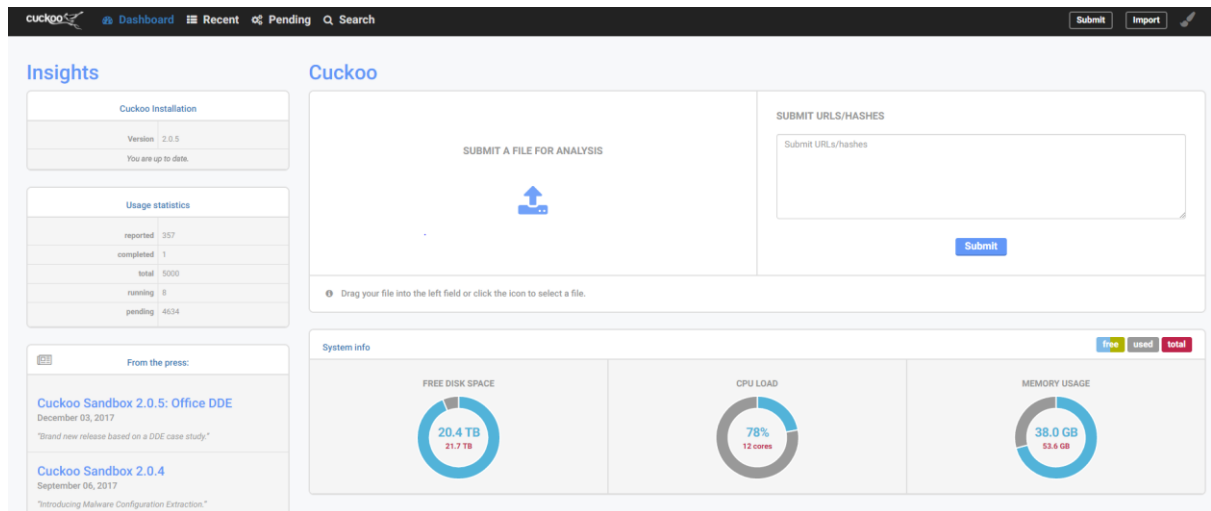
본 프로젝트에서는 정적 분석을 위해 IDA Pro와 peframe을 이용하였다. 정적 분석 정보를 이용하여 악성코드의 특징을 추출해낼 수 있는데 주로 Control-flow graph, Binary sequence, Mnemonic sequence, Opcode sequence 등이 사용된다. I. Santos 외 6명의 연구에서는 opcode sequence를 이용하여 변종 악성코드를 탐지할 수 있음을 보였고, I. Santos 외 2명의 연구에서는 머신러닝을 이용한 악성코드 탐지에서 opcode sequence를 악성코드의 특징으로 학습시켰을 때 높은 정확도로 악성코드 탐지가 가능함을 보였다. A. Shabtai 외 4명의 연구에서는 opcode sequence n-gram 기법을 사용하여 알려지지 않은 악성코드를 높은 정확도로 탐지해낼 수 있음을 보였으며 Xin Hu 외 3명의 연구에서는 정적 분석 정보에서 특징을 추출할 때 opcode sequence를 사용하는 것이 Control-flow graph, binary sequence, mnemonic sequence를 사용하는 것보다 효과적이라고 주장하였다. 따라서 본 프로젝트에서는 IDA를 사용하여 얻어낸 정적 분석 정보 중 opcode sequence를 다중 n-gram 기법으로 가공하여 악성코드의 특징으로 사용한다.

2.2.2 동적 분석

본 프로젝트에서는 동적 분석을 위해 오픈소스 소프트웨어인 쿠쿠샌드박스를 이용하였다. 샌드박스 환경은 가상머신 종류 중 하나인 버추얼박스(virtualbox)로 구성하였다. 쿠쿠샌드박스는 비정상적인 접근을 탐지하기 위해 의도적으로 설치해 둔 시스템을 의미하는 허니팟에 기초한다. 따라서 악성코드가 잘 동작해야 효과적인 분석을 할 수 있기 때문에 고의로 취약한 환경을 구성하였다. 방화벽과 윈도우 업데이트를 비활성화시키고, UAC를 비활성화시켰다. 또한 리눅스의 root 계정으로 시스템을 운영하는 것과 같이 Administrator 계정을 활성화하였다. 쿠쿠 코어가 샌드박스를 제어하기 위해서는 샌드박스의 아이피를 알아야 하는데, 아이피가 유동적으로 변경되면 코어가 제어를 할 수 없기 때문에 쿠쿠 엔진과 동일한 아이피 대역으로 설정하였다.


쿠쿠샌드박스는 웹 인터페이스(장고)를 제공한다. 이 웹 인터페이스가 사용하는 데이터베이스인 몽고디비(MongoDB)를 구축하여 사용자가 편리하게 브라우저를 이용하여 분석을 요청하거나 분석된 결과를 볼 수 있도록 하였다. 쿠쿠 코어나 샌드박스 설정을 위해서 설정 파일을 수정하였다.

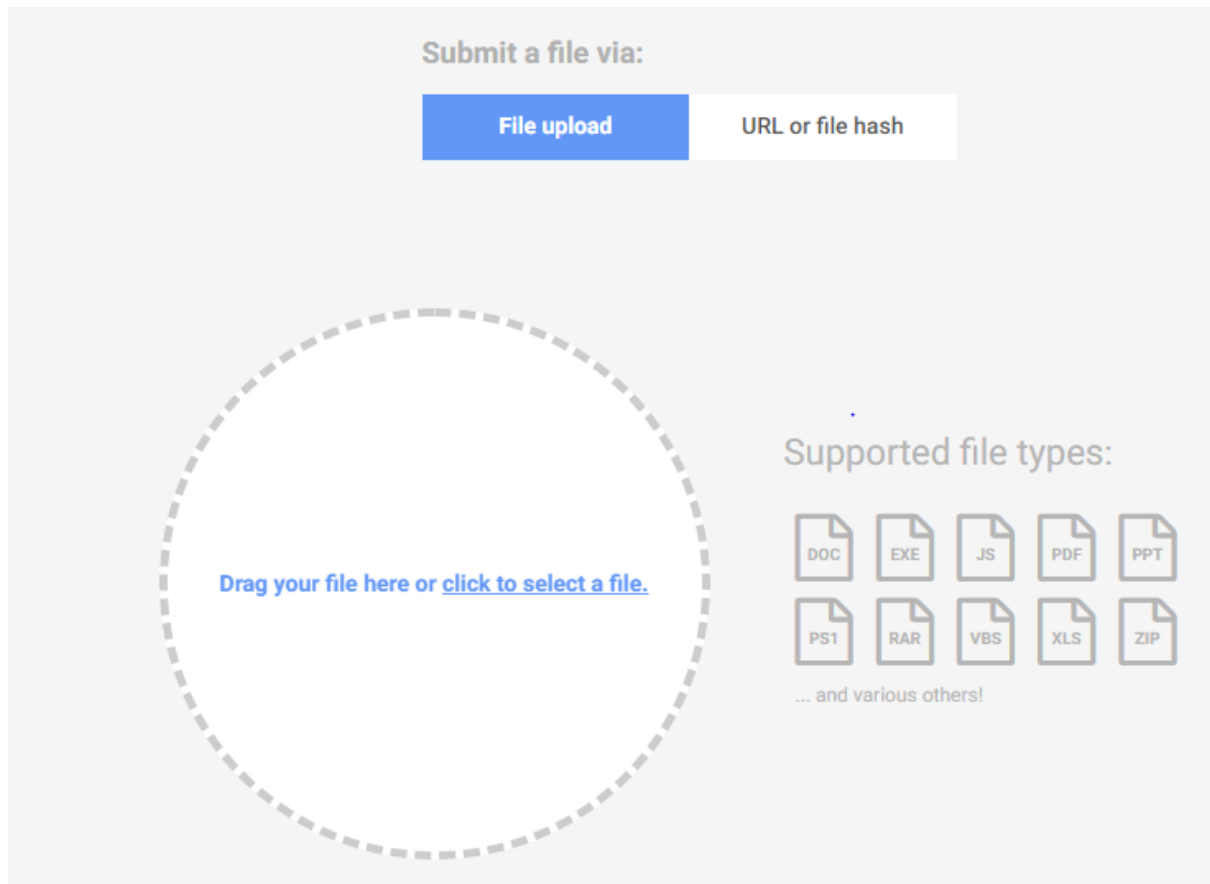
 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12



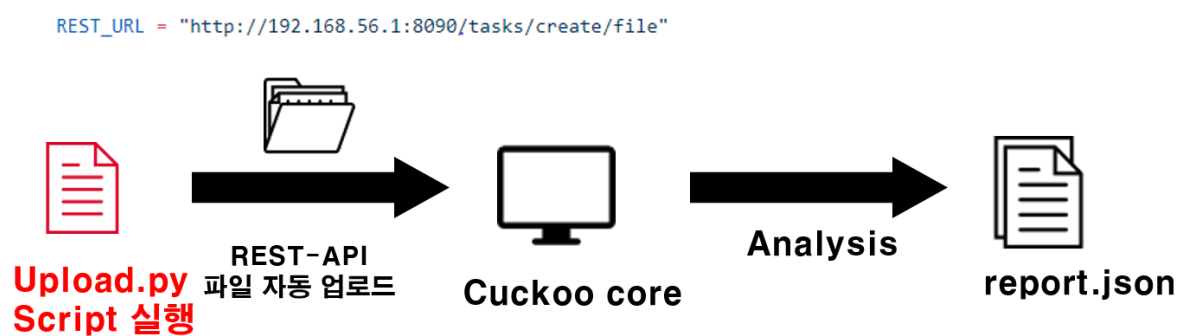
[그림 7] 쿠쿠샌드박스의 웹 인터페이스

악성으로 의심되는 파일을 분석하기 위해서는 웹 인터페이스를 이용하여 드래그 앤 드랍으로 파일을 전달하는 방법이 있지만, 자동화를 위하여 쿠쿠샌드박스에서 제공하는 REST API를 사용하였다. REST API를 이용하여 분석을 진행할 대량의 파일 업로드를 자동화하였다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

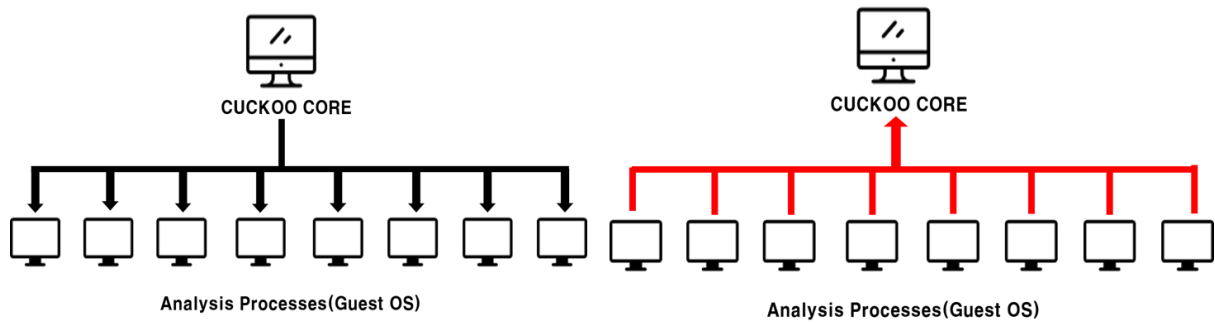


[그림 8] 웹 인터페이스를 이용하여 파일을 업로드하는 방법



[그림 9] REST API를 이용한 대량의 파일 업로드 자동화

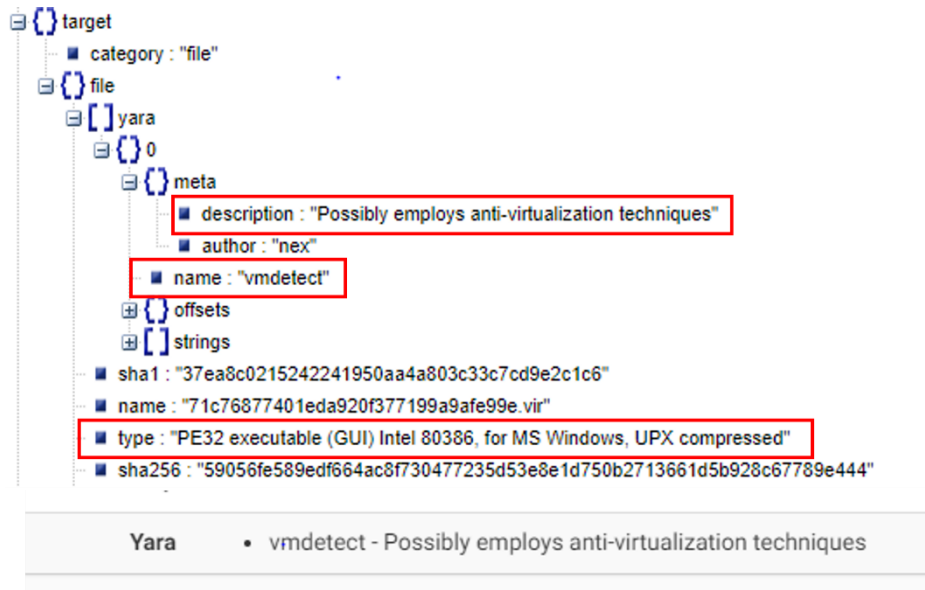
분석의 효율성을 높이기 위해 분석을 진행하는 guest instance의 개수를 1개에서 현재 8개로 늘려서 n개의 분석 파일에 대해 분석 시간을 1/n만큼 단축할 수 있다.



[그림 10] 쿠쿠 코어와 guest instance 구조도

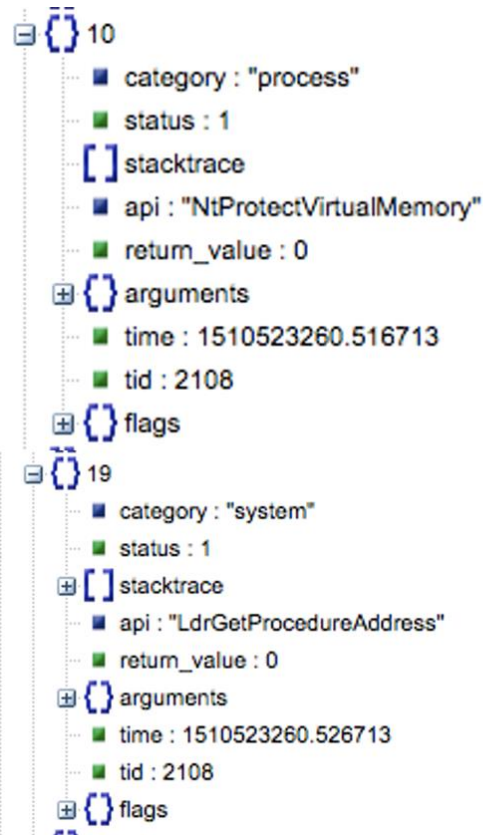
동적 분석 결과로 json 포맷의 리포트가 생성된다. 리포트에서 확인할 수 있는 결과로는 아래와 같다. 이들은 덤퍼링 모델을 생성할 때 추출될 피처의 후보가 된다.

- ① Process memory : 프로세스에 대한 메모리 덤프 분석 정보이다.
- ② Target : Yara rule에 의해 탐지되었을 경우 나타나는 정보이다.



[그림 11] Yara rule에 의해 탐지되었을 때 리포트에서의 Target 정보

- ③ 네트워크 프로토콜, 악성코드를 실행한 host 정보이다.
- ④ 정적 분석 결과(Strings..)
- ⑤ Behavior(API 통계, API call sequence..)



[그림 12] 한 프로세스의 10번째, 19번째 API 호출 기록(리포트)

⑥ Signatures : 위의 악성코드 정보들을 바탕으로 나타난 악성코드 특징(description)이다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

Signatures


Queries for the computername (1 event)
Checks amount of memory in system, this can be used to detect virtual machines that have a low amount of memory available (1 event)
A process attempted to delay the analysis task. (1 event)
Drops a binary and executes it (1 event)
Checks adapter addresses which can be used to detect virtual network interfaces (1 event)
Potentially malicious URLs were found in the process memory dump (50 out of 124 events)
Attempts to identify installed AV products by installation directory (3 events)
Deletes its original binary from disk (1 event)
A process performed obfuscation on information about the computer or sent it to a remote location indicative of CnC Traffic/Preperations. (4 events)

[그림 13] 웹 인터페이스에서 확인 가능한 파일에 대한 signatures

- ⑦ Score : signatures로 식별한 패턴을 통해 의심스러운 평균 수준을 수치화한 정도이다.

<p>Score</p> <p>This file shows some signs of potential malicious behavior. The score of this file is 1.2 out of 10.</p>
<p>Score</p> <p>This file shows numerous signs of malicious behavior. The score of this file is 4.2 out of 10.</p>
<p>Score</p> <p>This file is very suspicious, with a score of 5.4 out of 10!</p>

[그림 14] 웹 인터페이스에서 확인 가능한 파일에 대한 score

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

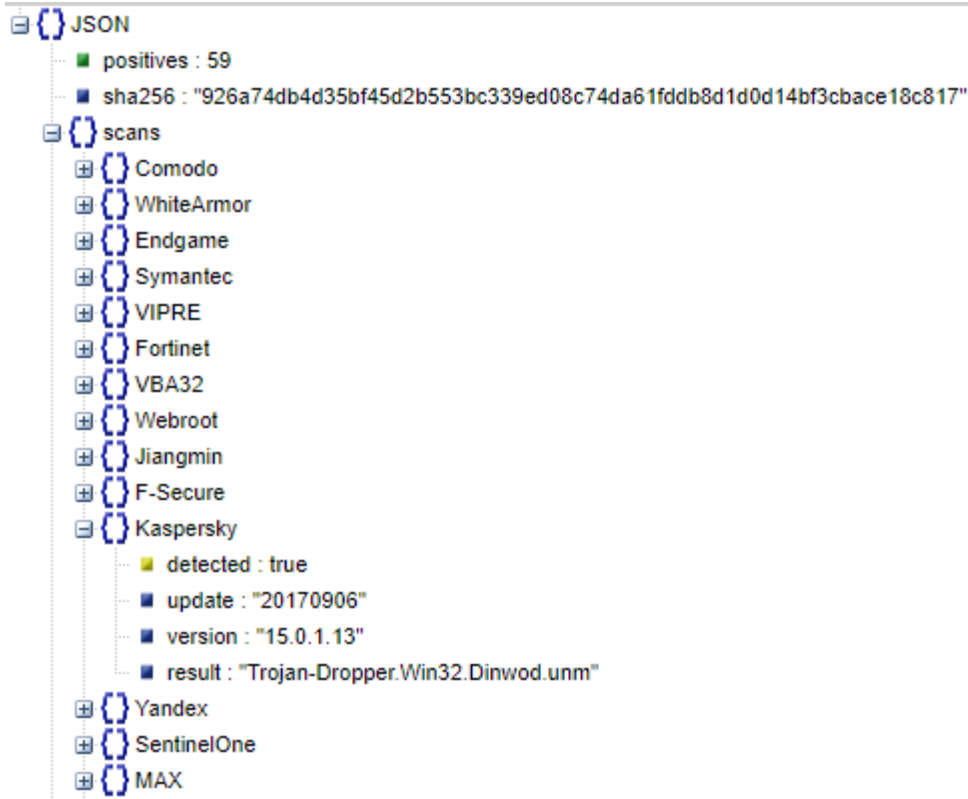
2.2.3 라벨링

본 프로젝트에서는 IT 보안 테스트 및 컨설팅 서비스 제공업체인 AV-TEST에서 우수한 평가를 받는 안티바이러스인 카스퍼스키의 라벨을 연구하고 독자적인 라벨을 제작한다. 카스퍼스키 안티바이러스 개발업체인 카스퍼스키 랩에서는 악성코드를 다음 7가지로 분류하며 이를 가장 일반적인 악성코드 분류 방법으로 정의한다..

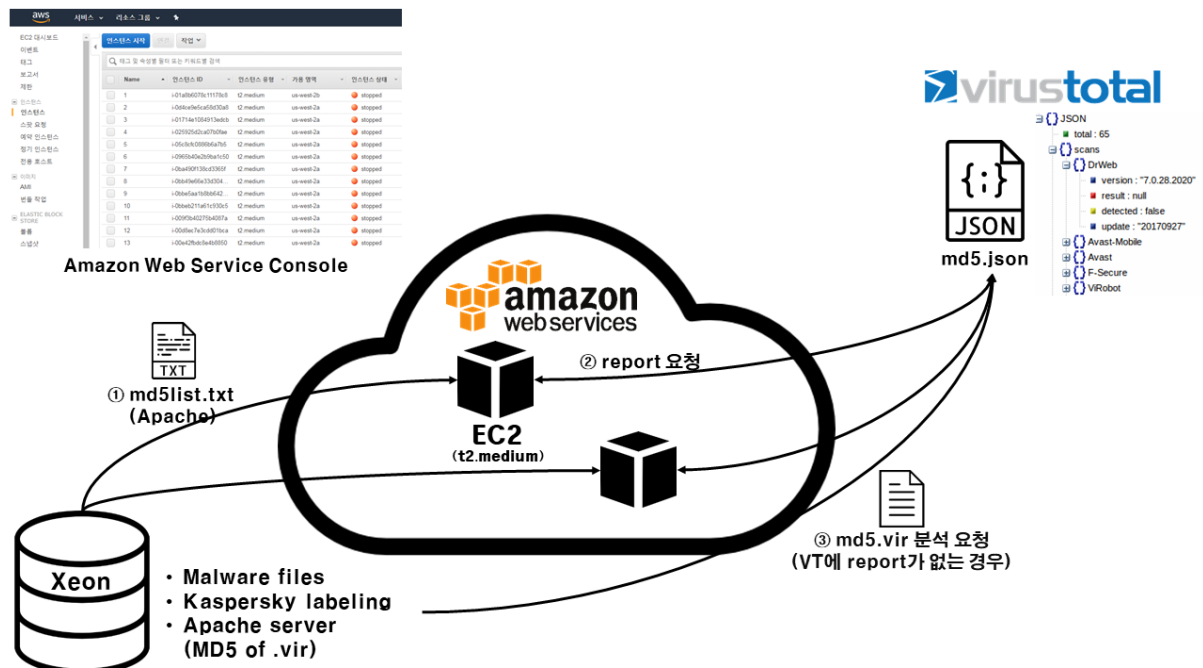
1. 바이러스(Virus) : 사용자 동의없이 기존 프로그램에 설치되는 자체 복제 프로그램 코드 유형으로 감염되는 객체의 종류, 공격대상을 선택하는데 사용하는 방법 혹은 공격방법을 통해 더 세분화될 수 있다.
2. 웜(Worm) : 자체 복제 프로그램으로 바이러스의 하위 구분으로 간주될 수 있으나 기존 프로그램을 감염시키지 않는 대신 네트워크 취약점을 조작하여 다른 시스템에 퍼져나갈 기회를 찾기 전까지 공격대상의 컴퓨터에 스스로 설치된다.
3. 트로이잔(Trojan) : 정상파일처럼 보이나 실제로는 유해한 행동을 수행하도록 설계되어 있는 악성코드의 유형이다. 트로이잔은 스스로 복제 되지 않기 때문에 퍼져나가지 않는다. 그러나 인터넷의 범위가 넓어지며 더 많은 대상에게 도달하기 쉬워졌다.
4. 랜섬웨어(Ransomware) : 랜섬웨어는 공격 대상의 금전탈취를 목적으로 설계된 악성코드 유형이다. 일단 실행되면 공격 대상의 시스템을 잠그거나 파일을 암호화하여 대상의 행동을 제한하고 팝업창이나 피싱 등의 형태로 금전을 요구한다.
5. 백도어(Backdoor) : 공격자가 시스템 설계자나 관리자에 의해 고의적으로 남겨진 시스템의 보안상 허점을 이용하여 시스템에 허가되지 않은 접근을 가능하게 하는 악성코드의 유형이다.
6. 루트킷(Rootkit) : 공격대상이 설치한 기존 보안 소프트웨어의 존재와 작동을 숨기도록 설계된 특수한 형태의 악성코드 유형이다.
7. 다운로드(Downloader) : 추가적으로 악성코드를 다운로드 하는 악성코드의 유형이다. 악성코드가 설정한 웹사이트로 접속하여 추가적인 악성코드를 다운받아 감염시킨다.

바이러스토탈은 파일에 대한 분석과 결과를 제공해주는 온라인 서비스로 분석결과로 약 60여개의 안티바이러스 엔진의 탐지 결과를 보여준다. 바이러스토탈 에서는 파일의 분석 결과를 받아올 수 있는 API를 제공한다. 제공된 API를 통하여 악성코드의 분석을 요청하면 json 형식의 파일로 분석 결과를 받게 되며 해당 파일을 파싱하여 각 악성코드의 카스퍼스키의 탐지결과를 얻어낸다. 바이러스토탈 API를 통하여 분석 요청을 할 때의 효율성을 높이기 위하여 AWS의 EC2 Instance를

이용하여 분산 분석을 한다.



[그림 15] 바이러스토탈 API를 통하여 받을 수 있는 json 형태의 분석 결과

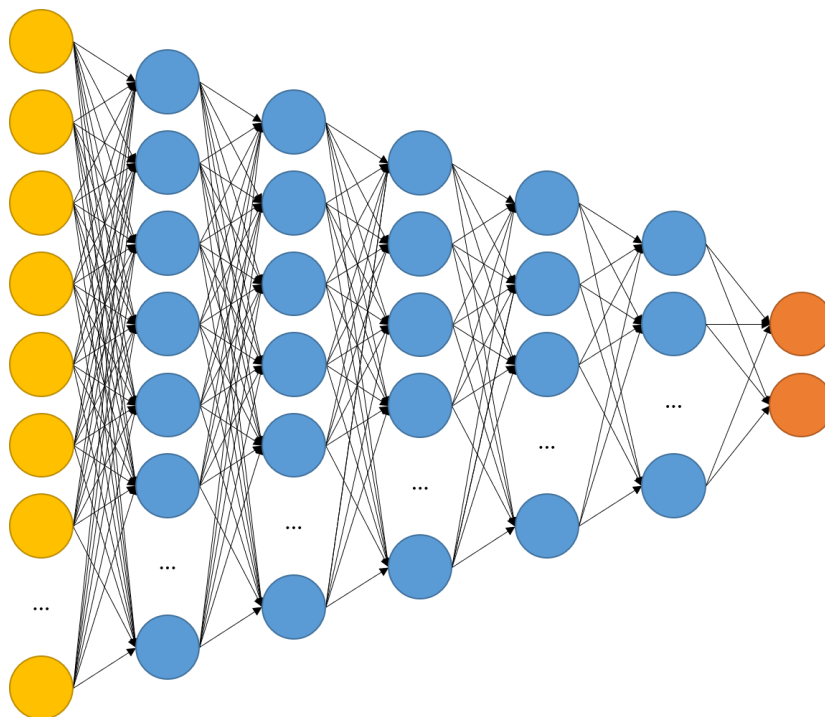


[그림 16] AWS EC2 인스턴스를 통한 바이러스토탈 분석요청 프로세스

2.2.4 딥러닝

본 프로젝트에서는 텐서플로우(Tensorflow)를 이용하여 ANN(Artificial Neuron Network) 모델을 만든다. 먼저 virussign(<http://www.virussign.com/>) 에서 다운로드 받은 악성코드 중 10,000개를 악성코드 샘플로, Microsoft 사의 소프트웨어, Adobe사의 소프트웨어, Nexon사의 게임, 주요 Antivirus 소프트웨어 파일 중 10,000개를 정상 파일 샘플로 사용하여 학습을 하였다.

[그림 14]은 악성코드 탐지를 위한 심층 신경망 모델의 구조이다. 심층 신경망의 입력 층 노드 개수는 12,288개이며, 2, 3, 4 – gram을 Feature Hashing 기법을 이용해 가공을 하였다. 심층 신경망의 은닉층 개수는 총 5개이며 각각 4096, 1024, 256, 64, 16개의 노드를 가지고, 활성화 함수로는 ReLU를 사용하였다. 또한 과적합을 막기 위해서 드롭 아웃 기법을 사용하였다. 제안하는 모델은 30%의 정보를 잊어버리도록 설정하였으며 마지막 은닉층을 거쳐 출력층에 도달할 때 소프트 맥스 함수를 이용하여 주어진 파일이 정상 또는 악성에 속할 확률로 변환한 뒤, 더 높은 확률을 선택하여 정상과 악성을 판별한다.



[그림 17] 악성코드 탐지를 위한 모델 구조

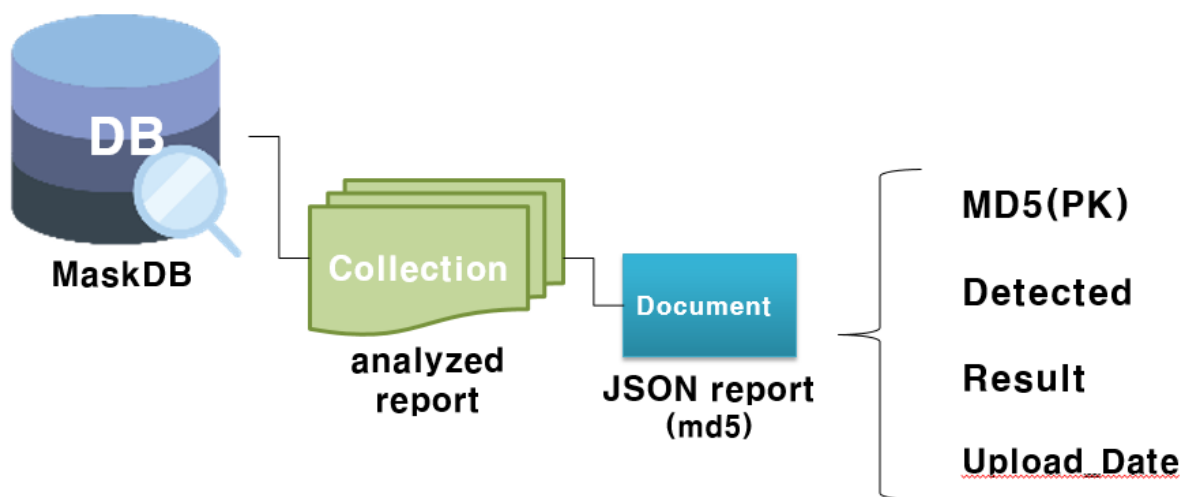
2.2.5 데이터 처리

본 프로젝트의 데이터 처리에 대한 수행한 내용은 다음과 같다.

첫째로 대량의 악성코드 샘플 데이터를 어떻게 운용하여 서비스 운영과 팀원의 연구에서 데이터 사용에 대한 시간 비용을 최소화 할 것 인지에 대한 이슈가 있었다.

이를 데이터베이스의 운용을 통해 이슈를 해결하고자 했다. 이때 팀원들의 경우 실험적인 데이터의 사용 및 저장을 하게 되는데, RDBMS를 사용하게 될 경우 컬럼의 추가가 유연하지 못해 적합하지 않았다. 따라서 유연성(flexibility)을 고려하여 json 타입의 비정형 데이터를 사용하는 NoSQL 데이터베이스를 사용하기로 하였다. 그 중 데이터의 삭제가 잦지 않고 분석 리포트가 계속 쌓이는 프로젝트의 특성을 고려하여 R/W(Read & Write) Performance가 뛰어난 몽고디비(Mongo DB)를 선택하게 되었다.

DB에는 정적, 동적 분석 리포트를 다루는 analyzed_report 컬렉션을 만들었고, md5, 바이러스 유무, 바이러스 라벨, 업로드 날짜를 파싱하여 analyzed_report 컬렉션에 자동 업로드 되도록 하는 스크립트와 md5 검색을 통해 원하는 리포트를 조회할 수 있는 스크립트를 작성하였다. 분산 DB의 구축 또한 고려해보았지만 이와 관련하여 이경용 교수님과 상담을 진행한 결과 가벼운 데이터 처리 연산을 주로 하게 되고 데이터의 양이 속도적으로 문제될 정도는 아니라는 답변을 듣게 되었고, 한정적인 서버 자원을 고려하여 단일 서버 환경으로 구축하였다.



[그림 18] 데이터베이스

둘째로 본 프로젝트는 업로드 한 악성으로 의심되는 파일에 대해 분석 후 이와 유사한 파일

의 기존 분석 결과를 동시에 보여주는 서비스 제공을 목표로 하였다.

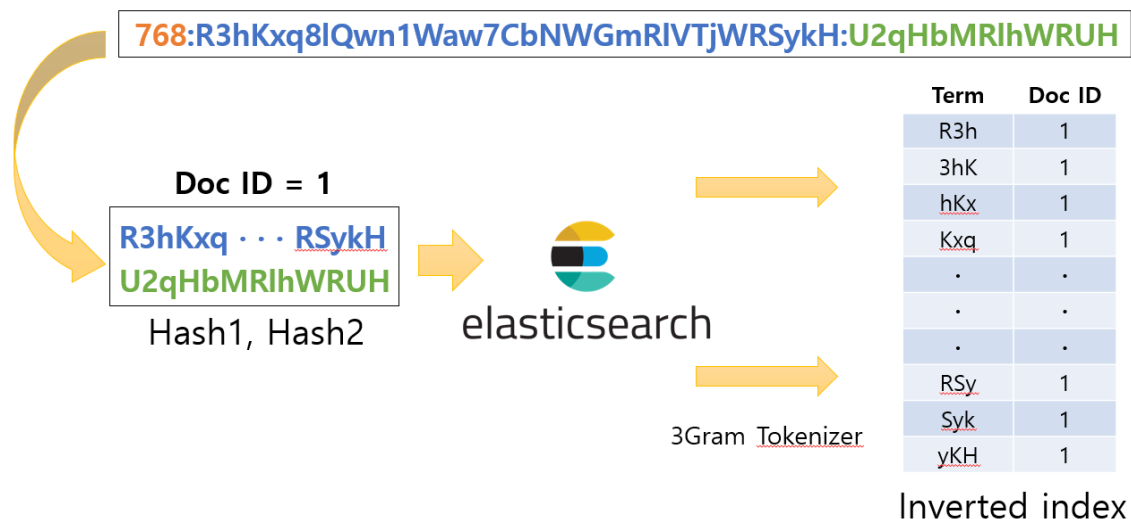
유사한 파일을 보여주기 위해서는 파일 유사도 측정 기법을 사용하여야 하는데, 이를 위해 SSDeep이라는 파일 유사도 측정 툴을 사용하였다. SSDeep은 하나의 스트링을 특정 알고리즘을 통해 hash 값을 도출해 내고, 같은 방법으로 또 다른 스트링의 해시 값을 도출해 내어 두 해시 값 간의 차이를 비교하여 유사도를 측정하게 된다.

이때 행위 기반의 유사도 측정을 위해 파일 전체에 대한 SSDeep 알고리즘 적용 대신, 파일의 opcode sequence를 추출하여 SSDeep 해시 값을 도출해 진행하게 되었다.

이렇게 도출된 SSDeep 해시 값들은 SSDeep에서 제공하는 Compare함수를 사용하여 유사도 점수를 얻을 수 있다. 하지만 전체 리포트 n개에 대해서 전부 Compare 함수를 사용해 유사도를 비교하는것은 n번의 함수 호출이 일어나게 되므로 검색하는데 많은 시간이 소요되게 된다.

이를 극복하기 위하여 도출된 SSDeep 해시를 엘라스틱서치를 이용해 역 인덱싱하여 검색 하는 방법을 고안하였다. 해시 전체를 통째로 역 인덱싱 하게 되면 완전히 같은 파일이 아닌 이상 검색되지 않게 된다. 따라서 해시 스트링을 n gram tokenize하여 역 인덱싱 해야 한다. 본 프로젝트는 실험적인 결과를 통해 3 gram tokenize를 하기로 하였다.

Block size : Hash1 : Hash2

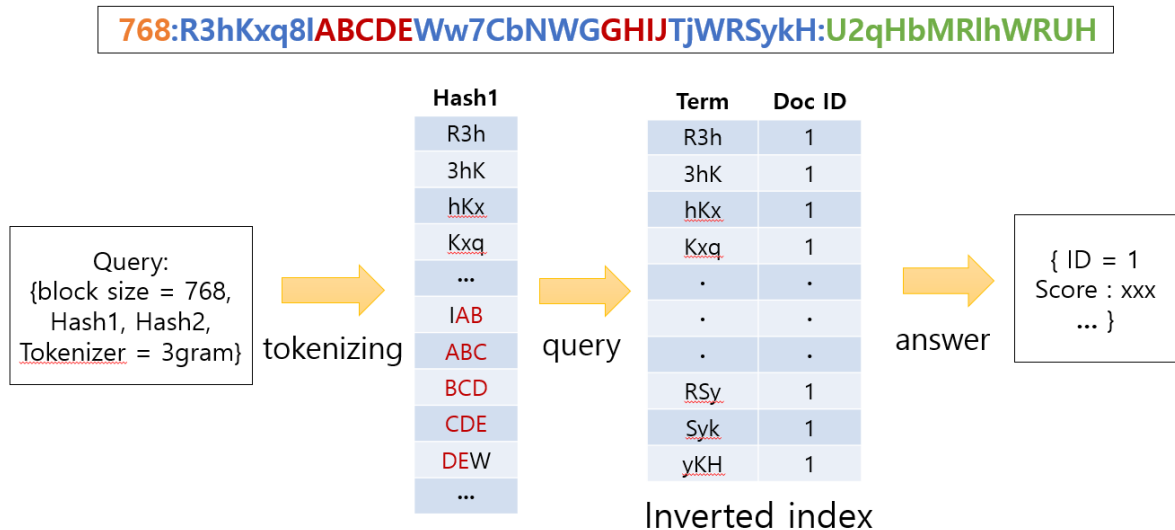


[그림 19] 3gram tokenize한 문서를 역 인덱싱 하는 과정

전체 리포트에 대해 역 인덱싱이 완료가 되면 업로드 된 파일을 3gram tokenize 한 후 준비 된

인덱스에 쿼리를 보내게 되면 워드가 가장 많이 매칭된 순서대로 리포트를 보여주게 된다.

Similar SSDeep string



[그림 20] 업로드 한 file의 ssdeep을 인덱스에 쿼리를 보내는 과정

2.2.6 웹

본 프로젝트의 웹에서 수행한 내용은 다음과 같다.

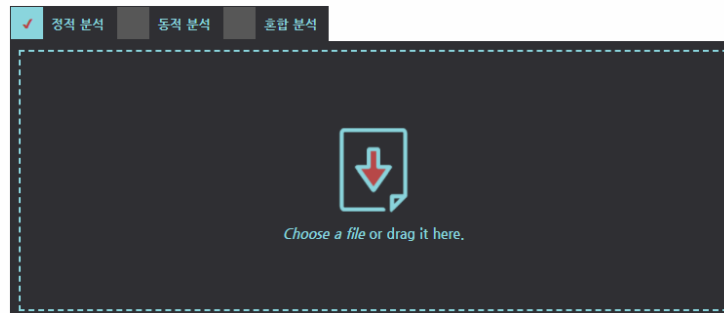
모든 레이아웃은 부트스트랩(Bootstrap) 프레임워크를 이용하여 사용자의 브라우저 크기에 따라 변하는 반응형으로 제작하였다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

NCNP : MASK 분석 통계 Sign in



악성코드 검사




[그림 13] 웹 초기화면

먼저 페이지 간의 이동을 위해 상단에 네비게이션 바를 배치하였다. 좌측에 팀 로고(NCNP)와 프로젝트 로고(MASK)를 배치하였고, 프로젝트 로고를 클릭 시 초기화면으로 돌아올 수 있도록 하였다. 또한 브라우저의 폭이 768px 보다 작으면 목록형 네비게이션 바로 변경되도록 하였다.

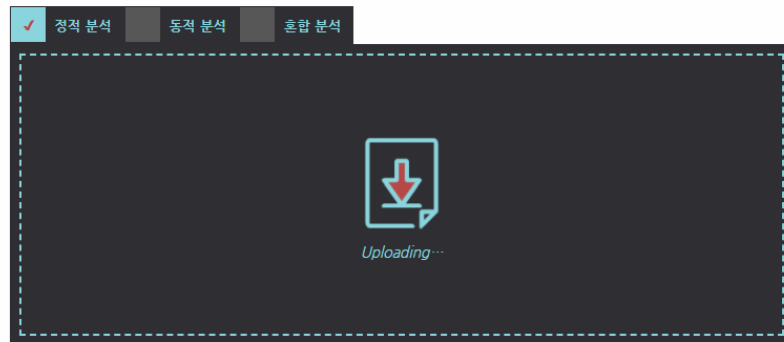


[그림 14] 브라우저의 크기에 따라 반응형으로 동작하는 네비게이션 바

사용자가 악성으로 의심되는 파일을 업로드 할 수 있도록 폼을 만들었고, 이 폼에 파일을 드래

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

그 앤 드랍을 하거나 원하는 파일을 선택하여 업로드 할 수 있도록 하였다. 폼의 좌측 상단에는 radio button을 배치하여 사용자가 정적 분석, 동적 분석, 정적과 동적을 이용한 혼합 분석, 이 세 가지 분석 방법 중 하나를 선택할 수 있도록 하였다.




[그림 15] 업로드시 사용자에게 보여지는 폼

업로드가 시작되면 업로드가 완료되는 동안 사용자에게 서비스가 멈춘 것이 아닌 업로드 중인 것을 알리기 위해 움직이는 이미지와 Uploading 문구를 출력할 수 있도록 했다. 스크립트 단에서는 Ajax를 이용해 업로드 완료 이벤트를 Listen하며 이벤트 감지 시 다음 페이지로 분석 요청을 보내도록 코드를 작성하였다.

업로드가 완료되어 분석요청이 오게 되면 분석 프로그램을 실행시키도록 하였고, 분석 프로그램을 실행할 때 인자로 업로드 된 파일의 경로를 넘기도록 하였다. 분석이 완료 된 후 에는 생성 되는 리포트를 데이터베이스에 업로드와 동시에 결과 페이지에 리포트 폼을 전송하도록 하였다.

결과 페이지가 분석이 완료된 리포트 폼을 받게 되면, 리포트에 있는 분석한 파일의 정보(md5, 바이러스 유무, 바이러스 분석 결과, 리포트 수정 날짜), 이와 유사한 파일에 대한 분석 결과, AI, 시그니처를 파싱해 분석 결과 화면을 구성하여 사용자에게 제공할 수 있도록 하였다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12


NCNP: MASK

분석

통계

Sign in

파일 정보



파일 MD5	virussign.com_00cee1918c8925a2a7782f1a0d5759c7
바이러스 유무	True
바이러스 분석 결과	None
리포트 수집 날짜	None

AI

SAFE

100%

시그니처

Primary

유사 파일

[그림 16] 사용자에게 보여지는 분석 결과 화면

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

3 수정된 연구내용 및 추진 방향

3.1 수정사항

3.1.1 크롤링 자동화

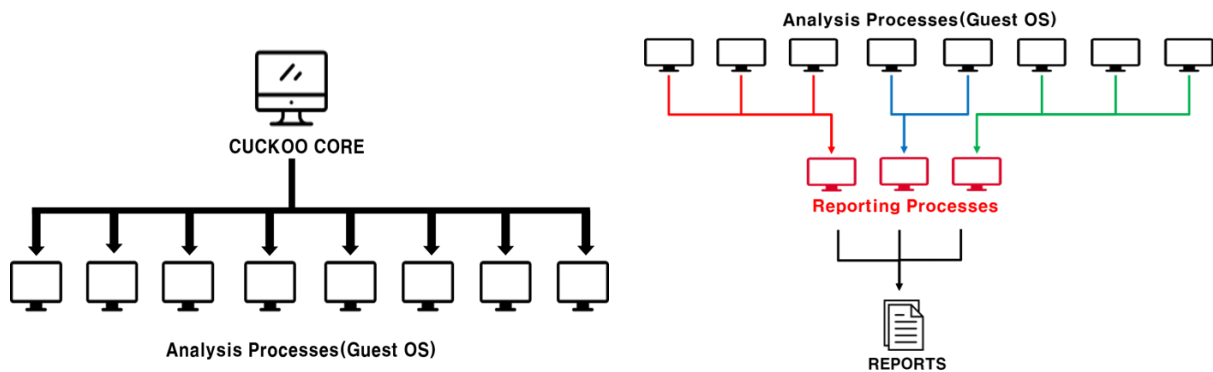
악성코드를 수집할 때 악성코드 크롤러를 개발하여 수집하려 하였으나 유료서비스나 토렌트 등을 이용해야 하기 때문에 자동화가 어렵다. 따라서 수집 채널을 수동으로 변경하였다.

4 향후 추진계획

4.1 향후 계획의 세부 내용

4.1.1 동적 분석

현재는 쿠쿠 코어가 guest instance들에게 분석 파일을 보내고, 분석 결과를 바탕으로 리포트를 추출해내는 리포팅 작업까지 진행하는데, 이렇게 될 경우 쿠쿠 코어의 과부하가 일어날 수 있다. 따라서 분석 프로세스와 리포팅 프로세스를 분리함으로써 시스템의 안정성을 높인다.



[그림 21] 쿠쿠 코어와 리포팅 프로세스의 분리 구조도

4.1.2 딥러닝 - 하이브리드 모델 개발

현재 정적 분석에 대한 학습 결과만 딥러닝을 이용해 악성코드 탐지 모델에 적용하고 있다. 추후에 동적 분석 결과로부터 추출된 피쳐도 적용되는 딥러닝 모델인 하이브리드 모델 개발을 목표로 한다.

4.1.3 웹

현재 웹은 정적, 동적, 혼합 분석 기능 중 정적 분석 기능만 제공한다. 향후 웹에서는 앞서 언급한 동적, 혼합 분석 기능을 모두 제공할 예정이고, 동적 분석 시에 나타나는 파일의 C&C 서버를 시각화하여 분석 결과 화면에 제공할 예정이다. 추가로 통계 메뉴에서 매일 분석되는 악성코드의 수와 발견되는 악성코드의 종류를 도식화한 표 등 각종

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

시각화 자료를 제공할 것이다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	MASK(Malware Analysis System in Kookmin)	
	팀 명	NCNP	
	Confidential Restricted	Version 1.4	2018-APR-12

5 고충 및 건의사항

없음.