

**2019년도
한국정보보호학회**

**학술발표대회
논문집**

- 일시 : 2019년 2월 14일(목)
- 주최 : 한국정보보호학회 영남지부/충청지부
- 주관 : 부산가톨릭대학교
- 후원 : (주)오픈링크시스템
(주)대신정보통신 (주)아이티센
- 장소 : 부산가톨릭대학교

모시는 글

2019년 정보보호학회 영남지부/충청지부 연합학술대회를 부산가톨릭학교 교정에서 개최하게 되어 매우 뜻 깊게 생각하고 있습니다. 최근 4차 산업혁명 아젠다가 이슈화되고 있는 상황에서 정보통신 기술과 다양한 기술의 융합으로 ICT 기술에도 많은 변화를 가져왔으며, 특히 블록체인, 사물인터넷, 빅데이터, 인공지능과 같은 기술의 발전으로 ICT 기술은 인터넷이라는 네트워크 공간을 벗어나 우리의 일상 곳곳에서 폭넓게 활용되고 있습니다. 하지만, 이와 같은 기술 발전으로 정보보호의 중요성은 더욱 크게 부각되고 있으며, 이러한 기술, 산업, 시장이 제대로 성장하고 활성화되기 위해선 보안과 프라이버시 보호 등의 문제들이 반드시 해결되어야 할 것입니다. 이번 연합학술대회에 발표될 총 50편에 가까운 논문을 보더라도 이러한 경향을 쉽게 볼 수 있습니다.

본 학술대회가 정보보호 각 분야에서 축적된 그 동안의 연구 결과를 한 자리에 모여 발표하고 토론함으로써 최근 정보보호 분야의 기술 동향이나 정보 교환에 도움이 되고, 회원 상호간의 친목을 다지는 장이 되기를 바랍니다. 특히 이번 학술대회는 영남지부와 충청지부의 연합학술대회로 개최되어 더욱 뜻깊은 행사가 될 것이며, 이를 통해 학회의 영남지부와 충청지부가 좀 더 활성화되기를 기대하며, 또한 연합학술대회를 통해 두 지부의 교류가 더욱 활발히 이루어질 수 있기를 바랍니다. 또한 귀한 시간을 내주시어 “2019년 보안트랜드”의 제목으로 기꺼이 초청 강연을 해주신 이호웅 안랩 CEO님께 감사드립니다. 마지막으로 이번 2019년 한국정보보호학회 영남지부/충청지부 연합학술대회에서 우수한 연구 논문을 발표해 주시는 발표자와 학술대회 참석자, 그리고 귀한 시간을 내주셔서 논문 심사 및 행사를 진행해주신 위원장 및 위원님들께도 깊은 감사를 드립니다.

2019년 2월 일

한국정보보호학회 영남지부 지부장 신상우

충청지부 지부장 김정태

학술대회 조직

■ 지부장 : 신상옥 (부경대)
 김정태 (목원대)

■ 프로그램위원

위원장 : 이만희 (한남대)
위원 : 임강빈 (순천향대)
 최대선 (공주대)
 이종혁 (상명대)

■ 운영위원

위원장 : 이대성 (부산가톨릭대)
위원 : 김호원 (부산대)
 최윤호 (부산가톨릭대)
 신원 (동명대)
 이석환 (동명대)

행 사 일 정

■ 행 사 장 : 부산가톨릭대학교 정보공학관

□ 2월 14일(목요일)

시 간	행 사 내 용					
12:00- 13:30	등 록					
	부산가톨릭대학교 정보공학관					
	2층 202호	2층 205호	2층 206호	3층 302호	3층 305호	3층 306호
13:30- 14:30	A- 1 IoT/CPS /클라우드 보안	B- 1 IoT/CPS /클라우드 보안	C- 1 블록체인 /금융 보안	D- 1 인공지능 보안	E- 1 컴퓨터보안	F- 1 정보보호관리/정책 기타정보보호
14:30- 15:00	휴 식					
15:00- 16:00	A- 2 IoT/CPS /클라우드 보안	B- 2 모바일 보안 디지털포렌식	C- 2 블록체인 /금융 보안	D- 2 인공지능 보안	E- 2 컴퓨터보안	F- 2 컴퓨터보안
16:00- 16:30	휴 식					
16:30- 18:00	<p>□초청강연: 2019년 보안 트랜드 (장소: 대학본부관 103호 대강의실)</p> <p>- 강 사: 이호웅 안랩 CTO</p> <p>□한국정보보호학회 영남지부 총회 및 폐회</p>					

목 차

Session A-1 IoT/CPS/클라우드 보안 (정보공학관 202호)

13:30 ~ 14:30 좌장 : 김호원 (부산대)

• OCF 장치를 비 OCF 장치로 연결하기 위한 OCF 게이트웨이 보안 Naufal Suryanto, 김호원 (부산대) -----	3
• IoT-DNS 네트워크 보안기법 분석 Ismail, 김현곤, 김호원 (부산대) -----	6
• 클라우드 오토-스케일링 시스템 구축에 관한 연구 손현민, 최성철, 최현택, 이현철 (대신정보통신) -----	9
• Wireless Sensor Network 환경에서의 윈도우 기반 에너지 소모량 예측을 통한 이상 노드 탐지 기법 신진명, 최석환, 최윤호 (부산대) -----	12

Session A-2 IoT/CPS/클라우드 보안 (정보공학관 202호)

15:00 ~ 16:00 좌장 : 이임영 (순천향대)

• 클라우드 스토리지 환경에서 공모공격을 방지하는 ID 기반 프록시 재암호화 기술 김원빈, 이임영 (순천향대) -----	18
• 하드웨어를 공유하는 LEA 및 HIGHT 블록 암호 저면적 구현 기법 김보훈 박종선 (고려대) -----	21
• 최신 IoT 악성코드의 침해사례 및 특징 분석 정해선, 최슬기, 곽진 (아주대) -----	24
• 효율적인 데이터 공유를 위한 CP-ABE 접근제어 기법에 관한 연구 황용운, 이임영 (순천향대) -----	27

Session B-1 IoT/CPS/클라우드 보안 (정보공학관 205호)

13:30 ~ 14:30 좌장 : 김정태 (목원대)

• Inversionless Berlekamp-Massey Algorithm을 이용한 BCH 부호의 하드웨어 구현 Asep Muhamad Awaludin, 윤영여, 김호원 (부산대) -----	33
• 클라우드 보안 취약점 분석에 관한 연구	

손현민, 박민규, 최현택, 이현철 (대신정보통신) -----	36
• 복합체 기반 폴딩 Sbox 구현 최병준, 박종선 (고려대) -----	39
• 비트 시리얼 구조를 적용한 ARIA 암호화 하드웨어 조정훈, 김보훈, 박종선 (고려대) -----	42

Session B-2 모바일 보안 디지털포렌식 (정보공학관 205호)

15:00 ~ 16:00 좌장 : 이만희 (한남대)

• 역전파 신경망 기반의 안드로이드 악성코드 탐지 시스템 Afifatul Mukaroh, 강효은, 김호원 (부산대) -----	48
• 전자 영수증 어플리케이션의 보안 취약점 분석 Afifatul Mukaroh, 강효은, 김호원 (부산대) -----	51
• 디지털 포렌식 관점의 Windows 10 Sticky Notes 연구 on a Crypto Hardware 채진희 (순천향대), 허원석 (고려대) -----	55
• 스마트 기기의 디지털 증거 훼손 방지 방안에 대한 연구 (생체인증 방식이 적용된 단말을 중심으로) 황혜성 (서울여대), 이세영 (경북대), 허원석 (고려대) -----	59

Session C-1 블록체인/금융 보안 (정보공학관 206호)

13:30 ~ 14:30 좌장 : 신상욱 (부경대)

• 블록체인 기반의 분산 EHRs 저장소 프라이버시 보호에 관한 연구구현 Sandi Rahmadika, 이경현 (부경대) -----	66
• 블록체인 프라이버시 보호 프로토콜 동향 강원태, 이상현, 김호원 (부산대) -----	69
• 탈중앙화 데이터 마켓플레이스의 접근제어 기법에 대한 연구 김혜빈, 박지선, 신상욱 (부경대) -----	72
• 탈중앙화된 데이터 거래 플랫폼의 연구 동향 분석 노시완, 이경현 (부경대) -----	75

Session C-2 블록체인/금융 보안 (정보공학관 206호)

15:00 ~ 16:00 좌장 : 이종혁 (상명대)

• 식품 추적 시스템에서의 블록체인 솔루션 Cho Nwe Zin Latt, 이경현 (부경대)	-----	82
• 크립토재킹 공격 유형 및 동향 분석 김의진, 김득훈, 곽진 (아주대)	-----	85
• 수정 · 삭제를 위한 Hybrid Blockchain 기반의 XGS (XOR Global State) 인젝션 기술에 관한 연구 라경진, 이임영 (순천향대)	-----	88
• 기업 블록체인 거버넌스와 위험 통제 프레임워크 설계 이경모, 이경현 (부경대)	-----	91

Session D-1 인공지능 보안 (정보공학관 302호)

13:30 ~ 14:30 촘장 : 최대선 (공주대)

• 딥러닝 기반 의료분야 인공지능 기술 동향 조사 김도완, 최대선 (공주대)	-----	99
• CNN기반 악성코드 이미지화 및 분석 시스템 김용수, Le Thi Thu Huong, 김호원 (부산대)	-----	102
• 확률적 분석을 통한 Feature Selection 및 머신러닝 기반 악성코드 탐지 박승수, 이만희 (한남대)	-----	105

Session D-2 인공지능 보안 (정보공학관 302호)

15:00 ~ 16:00 촘장 : 신 원 (동명대)

• Intrusion Detection Classifier Using Feature Selection and Recurrent Neural Network Thi-Thu-Huong Le, 김용수, 김호원 (부산대)	-----	112
• 딥러닝 기반 얼굴인식 기술의 동향 권대용, 최대선 (공주대)	-----	116
• 인공지능 시스템 보안 위협과 대책 박호성, 최대선 (공주대)	-----	119
• 딥러닝 기반 악성 PowerShell 스크립트 탐지 방안 송지현(과학기술연합대학원대학교), 최선오, 김종현, 김익균(ETRI)	-----	122

Session E-1 컴퓨터보안 (정보공학관 305호)

13:30 ~ 14:30 촘장 : 서화정 (한성대)

• 마스킹 연구 동향	권용빈, 안규황, 권혁동, 서화정 (한성대)	129
• 램포트 해시체인을 적용한 QR코드 출입통제 시스템	박형민, 김가을, 이선영 (순천향대)	132
• 하이브리드 퍼저를 이용한 브라우저 취약점 분석	이정모, 이병천 (중부대)	135
• seL4 마이크로커널의 특성 분석	장희준, 김형식 (성균관대)	140

Session E-2 컴퓨터보안 (정보공학관 305호)

15:00 ~ 16:00 좌장 : 임강빈 (순천향대)

• OpenMP를 활용한 LSH DRBG 병렬 최적 구현	권혁동, 안규황, 권용빈, 서화정 (한성대)	146
• 바이너리 분석 모듈을 이용한 Driller의 접근법 분석	김주환, 유지현, 윤주범 (세종대)	149
• 암호 알고리즘에 대한 부채널 분석 기술 동향	박찬희, 이진재, 김호원 (부산대)	152
• 위협 정보 표현 규격 STIX 2.0을 이용한 IDS 로그 표현에 관한 연구	유지현, 김주환, 윤주범 (세종대)	155

Session F-1 정보보호관리/정책 기타정보보호 (정보공학관 306호)

13:30 ~ 14:30 좌장 : 허준호 (부산가톨릭대)

• An Improved Big Data Analysis Technique for the Security of Nuclear Power Plant	이상도 (한수원), 허준호 (부산가톨릭대)	161
• An Analysis of Satisfaction Level of ITSM for Improvement of IT Service Efficiency in Military	우한철 (국방부 기무학교), 허준호 (부산가톨릭대)	164
• 가스터빈 디지털트윈: 가스터빈 동작 모델링을 위한 에이전트 기반 시뮬레이션	Putri Rahmawati Binti Eldi, 강효은, 김호원 (부산대)	166
• 성공적인 전산망 분리를 위한 실무적인 고려사항	박종현, 김창훈 (대구대)	171

Session F-2 컴퓨터보안 (정보공학관 306호)

15:00 ~ 16:00 죄장 : 박영호 (세종사이버대)

- 웹 어셈블리를 활용한 초경량 블록암호 CHAM 최적화 구현
Thi-Thu-Huong Le, 김용수, 김호원 (부산대) ----- 178
- 자동 익스플로잇 생성 도구에서의 버퍼 오버플로우 탐지 방법 분석
유지현, 김주환, 윤주범 (세종대) ----- 181
- 퍼저와 메모리 버그 탐지기를 사용한 바이너리 취약점 탐지 및 분류 방법
최민준, 김현욱, 윤주범 (세종대) ----- 184
- 사물인터넷 시대에 필요한 중소기업 정보보안수준 연구
문재웅, 박영호 (세종사이버대) ----- 188

오시는 길

□ 캠퍼스 주소 : (46252) 부산광역시 금정구 오륜대로 57

□ 지하철 이용 노선 안내

1호선 장전역(4번 출구) 하차 -> 왼쪽 방향 1분 거리 마을버스(5번, 5-1번) 승차 -> 부산가톨릭대학교 하차

□ 시내버스 노선안내

36, 50, 148, 178, 247, 1002 금정구청 하차 후 도보 10분



Campus Map



- 학술대회 장소 : 정보공학관(B동) 2층, 3층 강의실
- 초청강연 및 총회 장소 : 대학본부관 1층 103호 대강의실

□ 논문발표

- * 개인당 논문 발표시간 : 15분 (질의응답 포함)
- * 행사장에 빔 프로젝트 및 컴퓨터가 준비되오니 발표 자료를 미리 설치해 주시길 부탁드립니다.

□ 등록안내

* 사전등록기간 : 2019년 2월 14일 오후 6시까지 등록 가능 (현장등록 비용 동일)

구분	등록비
일반/회원	10만원
대학원생	5만원
학부생	면제

* 기타사항

- 발표자는 반드시 등록해야 하며, 2월 14일까지 등록해주시기 바랍니다.
- 학부생 논문의 경우 반드시 지도교수는 등록해야 합니다.
- 학부생의 경우 kiisc@kiisc.or.kr로 학생증 사본을 송부해 주십시오.

* 사전 등록 방법

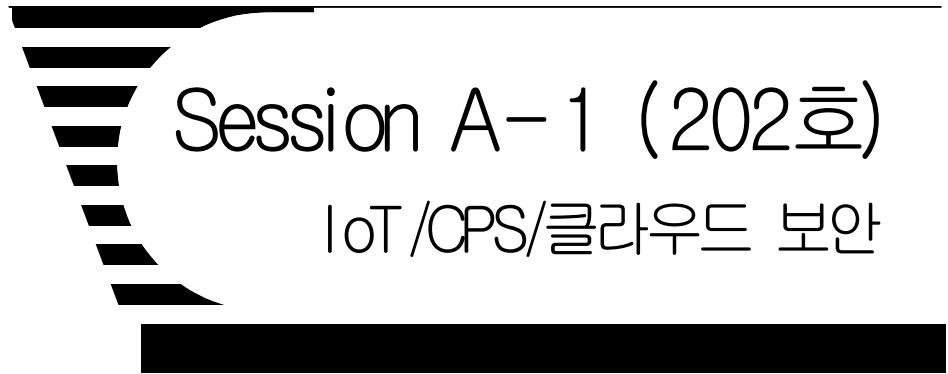
- 학회 홈페이지(www.kiisc.or.kr) 접속 → 학술행사 → 학회행사 → 사전등록바로가기
- 등록정보 작성 후 결제 방법 선택하시어 결제

□ 등록 송금처

- * 예금주 : 한국정보보호학회
- * 계좌번호 : 754-01-0008-146 (국민은행)
- 사전등록시 등록비는 위의 구좌로 송금하시고, 입금자가 대리일 경우 연락바랍니다.
- 학생의 경우 kiisc@kiisc.or.kr로 학생증 사본을 송부해 주십시오.
- 신용카드 결제 시 세금계산서 발급이 불가합니다. (부가가치세법 시행령 제 57조)

□ 등록문의처 : 한국정보보호학회

전화 : 02-564-9333 (내선번호2)
전자우편 : kiisc@kiisc.or.kr



좌장 : 김호원 (부산대)

OCF 장치를 비 OCF 장치로 연결하기 위한 OCF 게이트웨이 보안

수란토 나우팔*, 김호원*

*부산대학교 전기전자컴퓨터공학부

naufalsuryanto@gmail.com, howonkim@pusan.ac.kr

Secured OCF Gateway for Bridging OCF Devices to Non-OCF Devices

Naufal Suryanto*, Ho-won Kim*

*Department of Electrical and Computer Engineering,
Pusan National University.

Abstract

Open Connectivity Foundation (OCF) is one of the popular organizations for IoT standard. OCF devices communicate with other OCF devices using an OIC protocol that runs over the CoAP. A bridge is needed to connect the OCF devices with Non-OCF devices in order to communicate even though using a different protocol. In this paper, we propose a secured OCF gateway for bridging OCF devices to Non-OCF devices. IoTivity framework will be used as the implementation of OCF specification. The communication between OCF gateway and OCF devices will be secured by DTLS connection. MQTT is chosen for the example of Non-OCF device common protocol that is used by OneM2M, IBM Watson IoT, and other IoT platforms.

I. Introduction

There are a lot of IoT platforms developed in the market. Several large companies work together to make an IoT standard that can be widely used for several IoT products. OCF and OneM2M are examples of the organization for the common standard platform. Each platform has its own standard for communication protocol. That would be a problem when different device platforms try to communicate with each other. Maintaining the compatibility of each platform is very important so that they can communicate. The security system of each platform must also be maintained. This is one of the main objectives for the OCF standard.

1.1 OCF (Open Connectivity Foundation)

OCF (Open Connectivity Foundation) is an industry group whose stated mission to develop specification standards, promote a set of interoperability guidelines, and provide a certification program for devices involved in the Internet of Things (IoT). It has become one of the biggest industrial connectivity standards organizations for IoT.

The OCF main mission are as follow [1] :

- Provide specifications, code and a certification program to enable manufacturers to bring OCF Certified products to the market that can interoperate with current IoT devices and legacy systems.
- Make the end user's experience better by

seamlessly bridging to other ecosystems within a user's smart home and ensure interoperability with OCF compliant devices.

1.2 IoTivity Framework

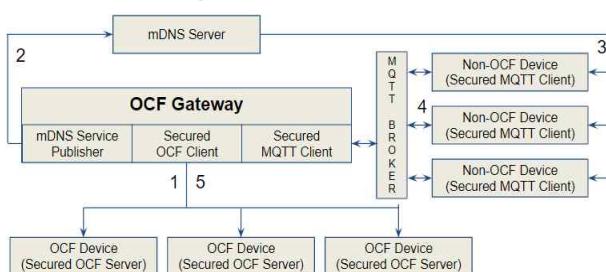
IoTivity is an umbrella of projects for building IoT devices. It is open-source and become a reference implementation of OCF specifications. It serves as a starting point for developing and certifying OCF products.

IoTivity is using OIC protocol (abbreviated to OC in code) which is a REST-like interface similar to HTTP and CoAP. The initial release of IoTivity includes functionality for:

- Resource registration
- Resource discovery
- Device discovery with filtering
- Property attributes (get/ set/ observe)
- Resource tree (resources with sub-resources)

II. OCF Gateway Architecture

We designed a gateway that would be a bridge to connect OCF devices and Non-OCF devices. OCF gateway architecture will be described in Figure 1.



[Figure 1] OCF Gateway System Overview

The OCF gateway system will work with the following flows :

1. OCF gateway through the internal OCF Client will periodically do a discovery for every OCF devices connected to the same network. If a new device found, OCF

gateway will send an observe message to that device. These features are available on the IoTivity framework.

2. OCF gateway through the internal mDNS Service Publisher will publish two mDNS services (OIC and MQTT) that can be used for both OCF and Non-OCF device for every new OCF device found. MQTT Publisher and Subscriber will be created by OCF gateway through the internal Secured MQTT Client.
3. Non-OCF device can do a discovery of OCF devices by doing browse of available MQTT Service from the mDNS Server. MQTT Service will provide the information on the MQTT topic name.
4. Non-OCF devices can get the properties of the OCF devices by subscribing the topics provided by the mDNS server. For updating the OCF device's properties, Non-OCF devices need to publish the new properties value using the same JSON structure on the same topic.
5. OCF gateway will translate and forward the data from the Non-OCF devices to the OCF devices and vice versa. The communication between OCF gateway and OCF devices using OIC protocol will be secured by DTLS connection, while communication between OCF gateway and Non-OCF devices using MQTT protocol will be secured by TLS connection.

III. Implementation

Based on the OCF gateway architecture design, we try to implement them and simulate the process to prove that the system works as planned.

First, we generate 10 OCF devices that are discoverable, observable, and secure.

```

Registering Resource :
- URI: /room/light/0
- Type Name : oic.r.light
- Interface : oic.if.baseline
Registering Resource :
- URI: /room/light/1
- Type Name : oic.r.light
- Interface : oic.if.baseline
Registering Resource :
- URI: /room/light/2
- Type Name : oic.r.light
- Interface : oic.if.baseline
Registering Resource :
- URI: /room/light/3
- Type Name : oic.r.light
- Interface : oic.if.baseline
Registering Resource :
- URI: /room/light/4
- Type Name : oic.r.light
- Interface : oic.if.baseline

```

[Figure 2] Registering OCF Devices

OCF gateway will automatically find any new devices registered in the network.

```

Found resource for the first time on server with ID: 31313131-3131
URI of the resource: /room/light/0
Host address of the resource: coap://[fe80::3cd6:587b:10c9
List of resource types:
oic.r.light
List of resource interfaces:
oic.if.baseline
Host of resource:
coap://[fe80::3cd6:587b:10c9:677f%25enp0s3]:38874
List of resource endpoints:
coaps://192.168.0.11:50528
coaps://[fe80::3cd6:587b:10c9:677f%25enp0s3]:35436
Change host of resource endpoints
Current host is coaps://192.168.0.11:50528
List of resource connectivity types:
CT_ADAPTER_IP
CT_FLAG_SECURE
CT_IP_USE_V4

```

[Figure 3] OCF Gateway found new resource

After a new device found, OCF gateway will observe that device and publish the mDNS services.

```

Observe registration action is successful:
OBSERVE Result:
Resource URI: /room/light/0
Resource attributes:
brightness : 0
state : false
in : Room Light0

```

[Figure 4] OCF Gateway successfully observe new resource

Non-OCF device can browse mDNS service for discovering any available MQTT services.

```

naufal@naufal-VirtualBox:~/iotivity-1.3.1$ avahi-browse -r _mqtt._tcp
+ enp0s3 IPv6 MQTT_room_light_1 _mqtt._tcp local
+ enp0s3 IPv6 MQTT_room_light_7 _mqtt._tcp local
hostname = [naufal.VIRTUALBOX.local]
address = [fe80::3cd6:587b:10c9:677f]
port = [1883]
txt = ["topic=/ocf/room/light/7"]
= enp0s3 IPv6 MQTT_room_light_4 _mqtt._tcp local
hostname = [naufal.VIRTUALBOX.local]
address = [fe80::3cd6:587b:10c9:677f]
port = [1883]
txt = ["topic=/ocf/room/light/4"]

```

[Figure 5] Browse MQTT services using mDNS

Non-OCF devices can select which OCF device that wants to be connected by subscribing or publishing the topic earned from mDNS.

```

Client mosqsub/4343-naufal-Vir sending CONNECT
Client mosqsub/4343-naufal-Vir received CONNACK
Client mosqsub/4343-naufal-Vir sending SUBSCRIBE (Mid: 1, Topic: /ocf/room/light/0, Qos: 0)
Client mosqsub/4343-naufal-Vir received SUBACK
Subscribed (mid: 1): 0
Client mosqsub/4343-naufal-Vir received PUBLISH (d0, q0, r0, m0, '/ocf/room/light/0', ... (48 bytes)
{"brightness":0,"state":false,"n":"Room Light0"}

```

[Figure 6] Non-OCF Devices subscribe

For experiments purposes, we try to publish the following JSON data {"state" : true, "brightness" : 50} to "/ocf/room/light/0" topic. OCF gateway will translate and forward that message to the OCF device.

```

16:49:57.038 [MQTT Call: /room/light/0 sub] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - MQTT UPDATE TRIGGERED - Post data to /room/light/0
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - POST request was successful
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - Resource URI: /room/light/0
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - Resource attributes:
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - brightness : 50
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - state : true
16:49:57.043 [Thread-234] INFO com.pnu.islab.iot.ocf.gateway.OCFGateway - n : Room Light0

```

[Figure 6] OCF Gateway forwards the message to OCF Device and update its data

IV. Conclusion

The OCF gateway architecture that we have implemented has worked well. Non-OCF devices can communicate with OCF devices through the OCF gateway. The security of each platform is still maintained.

[References]

- [1] OCF – Foundation [online] Available at: <https://openconnectivity.org/foundation> [Accessed 8 Feb. 2019]
- [2] About | IoTivity [online] Available at: <https://iotivity.org/about> [Accessed 8 Feb. 2019]
- [3] Lee Joo-Chul, Kim Hyong-Jun, and Sang-Ha Kim. “Virtualizing Non-OCF Devices into OCF Ecosystem.” 2018 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, South Korea.
- [4] Lee Joo-Chul, Kim Hyong-Jun, and Sang-Ha Kim. “Bridging OCF devices to legacy IoT devices.” 2017 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, South Korea.

IoT-DNS 네트워크 보안기법 분석

이스마일*, 김현곤*, 김호원*

*부산대학교 정보컴퓨터공학과

is.tmdg86@gmail.com, kgusrhs@gmail.com, howonkim@pusan.ac.kr

IoT-DNS Network Security Analysis

Ismail*, Hyungon Kim*, Ho-Won Kim*

*Department of Electrical and Computer Engineering,
Pusan National University.

요약

Since DNS utilized for IoT infrastructure in enterprise system, security become one of challenge to solve and ensure all data sent properly from the origin to the targeted device.

The security issues not only cover on DNS network but also in the devices area. Because the devices exposed to the internet connection, will make device easily targeted by attacker to control and get various data from this devices.

This paper will explain several probable attacks and solutions to handle it.

I. 서론

IoT requires enterprises to invest in new network and infrastructure technology to support the millions of connected objects. IoT devices utilize DNS to connect to endpoints across the internet.

DNS is an old protocol in terms of internet. Over the years, a number of attacks have been carried out on the DNS and also the DNS infrastructure has been exploited to carry out large-scale attacks.

One of famous attack for IoT devices to DNS are “rebinding attacks” that attack a half billion devices[1]. DNS rebinding attacks are when an attacker tricks a user’s browser or device into binding to a malicious DNS server and then make the device access unintended domains[2].

II. 본론

2.1 IoT-DNS Security Attack

Using DNS as Internet of Things infrastructure connect to the devices will causing many security issues related to DNS probable attacks it self, and also unsecured device transfer.

2.1.1 DNS Infrastructure Vulnerabilities

DNS plays a critical role in the process of connecting devices through internet connection and it has become a target for attacks. When DNS is not properly configured it will be vulnerable to various attacks. There are many ways to attack the DNS Network, a few of them are :

2.1.1.1 Man in The Middle Attack

Response to the DNS query are authenticated by the IP address of DNS server. So an attacker can intercept and spoof the IP address of DNS server and make a response look genuine as if it had originated from the intended DNS server and make the DNS server as bridge between the real DNS server to the client.

2.1.1.2 TCP SYN Flood Attack

DNS server use TCP port 53 for zone transfer, which is can be used by attacker for TCP SYN Flood attack. The attacker masquerades as a DNS client and spoofs the source IP address with a fake IP address, which will make the DNS server send SYN-ACK packets to fake destinations. The receivers of the SYN-ACK will simply drop them, as they not initiated it and make amounts DNS resources such as cpu, memory wasted and leads to exhaustion.

2.1.1.3 Denial of Service (DoS) and Distributed Denial of Service (DDoS) Attack

In a DoS attack, an attacker uses a single connection to flood a target DNS server with fake DNS query with objective of exhausting the DNS server. In the other side DDoS aim to exhaust the DNS server to UDP requests, both of this attacks are typically carried out to cause a loss of availability of DNS server

2.1.2 DNS Infrastructure Network Security

For handle all of this kind of attacks, there are several approachment that can be use to protect the DNS network such as :

2.1.2.1 DNS Security Extension (DNSSEC)

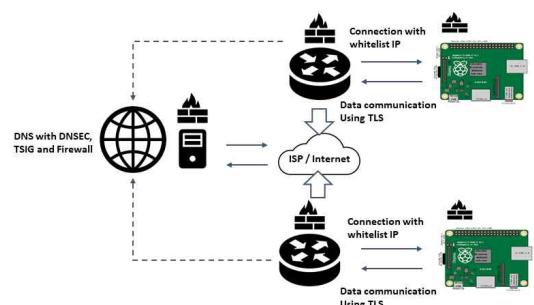
DNSSEC assures the authenticity of the origin of response and provides integrity for the data received from DNS servers. DNSSEC uses asymmetric cryptography to prevent DNS spoofing by ensuring that all responses received are digitally signed by the DNS zone administrators.[4][5]

2.1.2.2 Transaction Signature (TSIG)

TISG ensures a secure communication between primary and secondary DNS by use symmetric keys and cryptographic hash functions, and ensures that the data received in zone transfer is authentic and not modified during transit, also provides authenticity of the DNS response.[4]

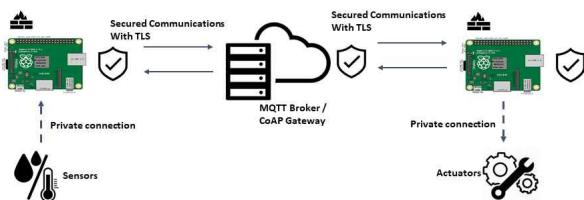
2.1.2.3 DNS Firewall

Firewall is security appliance that filters the DNS traffic to protect various DNS attacks and typically blocks certain type of DNS queries. DNS Firewall has the feature of malware protection known as Response Policy Zone (RPZ). Inside firewall also can added list of allowed IP's to request and response the DNS server. This firewall can implement for network and devices at the same time.[6]



2.1.3 Devices Data Transfer Protection

Besides DNS network protections, devices communication also one of important thing that have to secured from attacks. To handle this issue, device communication can be protect using TLS methods for TCP connection and also UDP connection



III. 결론

DNS Firewall can protects the DNS infrastructure from DoS/DDoS attacks but still many ways to attacks the DNS that not cover in this paper, moreover mDNS/DNS-SD security still need further research for implementation and limitation at scale network and security

- [3] NS1, THE INTERNET OF THINGS REQUIRES A RETHINK ON DNS, “<https://ns1.com/blog/internet-of-things-requires-a-rethink-on-dns>”, Available at February 14, 2017
- [4] ResearchGate, Domain Name System (DNS) Security: Attacks Identification and Protection Methods, “https://www.researchgate.net/publication/326894081_Domain_Name_System_DNS_Security_Attacks_Identification_and_Protection_Methods”, Available at 11 August 2018
- [5] Almira Hamzic, Isabel Olofsson, DNS and the Internet of Things: Outlining the challenges faced by DNS in the Internet of Things, KTH ROYAL INSTITUTE OF TECHNOLOGY, 12-13, 2016
- [6] Naman Gupta, Srishti Sengupta, Vinayak Naik, A Firewall for Internet of Things, 9th International Conference on Communication Systems and Networks, 1-2, 2017

[참고문헌]

- [1] theinquirer, Half a billion enterprise IoT devices vulnerable to DNS rebinding attacks, “<https://www.theinquirer.net/inquirer/news/3036359/half-a-billion-iot-devices-in-the-office-vulnerable-to-dns-attacks-warns-amis>”, Available at 23 July 2018
- [2] BLEEPINGCOMPUTER, Half a Billion IoT Devices Vulnerable to DNS Rebinding Attacks, “<https://www.bleepingcomputer.com/news/security/half-a-billion-iot-devices-vulnerable-to-dns-rebinding-attacks/>”, Available at July 20, 2018

클라우드 오토-스케일링 시스템 구축에 관한 연구

손현민*, 최성철*, 최현택*, 이현철*

*대신정보통신(주)

linuxson@dsic.co.kr, scchoi@dsic.co.kr, htchoi@dsic.co.kr, wlsqor2@gmail.com

A Study on the Construction of Cloud Auto-Scaling System

Hyun-Min Son*, Sung-Chul Choi*, Hyun-Taek Choi*, Hyun-Cheol Lee*

*DaiShin Information&Communications Co., Ltd

요약

업체 및 공공 기관들은 정보 자원을 통합, 효율적으로 운영하기 위해 업무 서비스의 클라우드 전환을 추진하고 있다. 업무시스템의 트래픽량 증가 및 예측 가능한 부하는 기존 자원 할당을 통해 지원 가능하나 예측하기 어려운 시스템의 부하 발생시 신속한 대응력 부재로 효율적이고 빠른 서비스 제공을 위한 자동 자원확장(오토-스케일링) 기술 도입이 필요하다. 본 논문에서는 클라우드 환경 기반의 자동 자원확장 기능을 업무시스템에 도입하여 DDoS와 같은 보안 공격이나 예상하지 못한 긴급 및 재난 상황 등으로 인한 시스템 부하가 급증하여도 원활한 서비스 운영이 가능한 클라우드 기반 자동 자원확장 시스템 구축 방법을 제안한다. 이를 통해 예측 할 수 없는 시스템 부하 발생시에도 서비스 연속성을 보장 할 수 있어 시스템의 신뢰도 향상에 기여 할 수 있을 것이다.

I. 서론

클라우드 컴퓨팅(Cloud Computing)이란 구름 너머에서 제공하는 컴퓨팅 서비스를 사용자가 필요할 때 빌려 사용하는 것으로 클라우드라는 명칭은 IT 디아어그램에서 인터넷을 구름으로 표현하던 것에서 유래되었다. 즉, 클라우드는 인터넷을 의미하고 인터넷에 연결된 서비스 제공자의 데이터센터에 접속하여 서비스를 사용하는 것이 클라우드 컴퓨팅이다. 스토리지·플랫폼·소프트웨어와 같은 ICT 자원을 데이터센터에 공유 풀(Shared Pool)로 집적시켜 이용자가 필요로 하는 만큼 분리하여 네트워크를 통해 가상화 서비스로 제공하고 사용하는 만큼 비용을 청구하는 방식을 의미한다. 클라우드 컴퓨팅은 ICT 자원을 물리적으로 구축·운영하는 방식에 비해 비용을 줄이고 상황에 따라 신속하게 필요한 ICT 자원을 재배치 할 수 있게 해 준다. 또한 디지털 데이터의 폭발적 증가, 모바일 기기의 확산, 사이버 보안에 대한 관심 고조 등

ICT 환경 변화에도 효과적으로 대응할 수 있다.

최근 장시간에 고성능의 자원을 활용하여 안정적이고 지속적인 자원의 공급을 필요로 하는 대규모 데이터센터는 클라우드 자원을 효율적으로 통합, 활용하기 위해 자원 가상화 특성을 활용한 오토-스케일링(Auto Scaling) 기술이 사용되고 있다. 오토-스케일링이란 사용자가 정의된 상황에 따라 가상 컴퓨터 자원(VM 또는 컨테이너)의 서버 용량(Capacity)을 자동으로 확대 또는 축소 할 수 있는 클라우드 컴퓨팅 기술을 말한다. 오토-스케일링 기법은 효율적이고 통합적으로 클라우드 자원을 제공하지만 대부분의 오토-스케일링 기법은 단순한 하드웨어의 성능을 기반으로 제공되고 있어 응용의 특성이나 서비스의 정체, 성능 판단 기준 등의 고려가 필요하다. 본 논문에서는 클라우드 환경 기반의 자동 자원확장 기능을 업무시스템에 도입하여 시스템 부하가 급증하여도 원활한 서비스 운영

이 가능한 클라우드 자동 자원확장 시스템 구축 방법을 제안한다.

II. 관련 연구

최근 클라우드 컴퓨팅 환경에서 가상화 기술을 통해 자원 규모의 확장과 축소가 용이하고 효율적인 자원 관리 기법이 제공되는 오토-스케일링 기술에 대한 연구가 이루어지고 있다. 특히 자원 사용의 비용을 최소화하기 위해 가상머신을 스케일링하는 기술[1]과 사용자가 정의한 작업의 데드라인이나 자원 사용 비용을 최소화 할 수 있는 기술[2]이 활발히 연구되고 있다. 또한 사설 및 상용 클라우드로 구성된 하이브리드(Hybrid) 클라우드 환경에서 가변적인 자원 요구에 따라 자원을 할당하고 SLA(Service Level Agreement) 위반을 최소화하는 방법 [3]과 대규모 계산처리 서비스인 HTCaS(High Throughput Computing as a Service)를 위한 가상머신 오토-스케일링을 최적화하는 방법이 연구되고 있다[4]. AWS(Amazon Web Service)의 오토-스케일링 서비스는 사용자가 정의한 규칙으로 자원의 규모를 확장하고 축소하는 규칙 기반의 오토-스케일링 기술을 통해 애플리케이션을 모니터링하고 용량을 자동으로 조정하여, 안정적으로 여러 자원에 대해 애플리케이션 규모 조정을 설정 할 수 있는 기능을 제시하였다[5].

III. 클라우드 오토-스케일링 시스템

3.1 컨테이너 기반 오토-스케일링 시스템

컨테이너(Container)는 컨테이너 이미지 안에 필요한 소프트웨어와 의존성 라이브러리를 같이 패키징하는 기술로 배포 및 라이프 사이클 관리의 복잡성을 줄이고 Host OS 상에서 컨테이너별로 격리 기능을 제공한다. 도커(Docker)란 애플리케이션을 신속하게 구축, 테스트 및 배포 할 수 있는 소프트웨어 플랫폼으로 소프트웨어를 컨테이너라는 표준화된 유닛으로 패키징하며, 이 컨테이너에는 라이브러리, 시스템 도구, 코드, 런타임 등 소프트웨어를 실행하

는 데 필요한 것이 포함되어 있다. 이미지(Image)란 클라우드 서비스에 맞게 배포하기 위해 특정 프로세스를 실행하기 위한 환경 설정을 말한다. 다음 [그림 1]은 VM과 Docker 스택을 나타낸 것으로 컨테이너 기반 오토-스케일링 시스템은 외부 트래픽에 따라 서버의 수(POD)를 자동 조절(Scale Out/in) 가능한 체계를 말한다. VM(Virtual Machine) 기반에서는 가용·사용 용량 관리가 중첩 이었다면 오토-스케일링은 업무 서비스의 트래픽 패턴 기반의 용량 관리 기능이 추가적으로 필요하다.

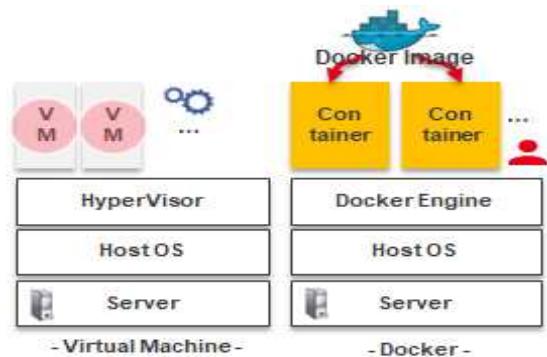


그림 1. VM과 Docker

3.2 대상 선정 및 시스템 구성 방안

확장 가능한 오토-스케일링 시스템 구축을 위한 적용 대상 선정 방안은 5단계 프로세스를 거쳐 선정 하였다.

- 적용 대상 도출 : 적용 가능한 업무서비스 도출
- 적용 가능 유형 검토 : 유형별 업무 서비스를 고려하여 적용 가능 유형 검토
- 적합성 상세 검토 : 인프라, 보안 및 트래픽 특성 등을 고려한 시스템 대상 적용 가능 세부 조사 및 상세 검토
- 유형별 적용 방안 도출 : 유형별 대표 시스템을 선정하여 전환을 위한 방안 수립
- 적용 대상 선정 : 전환 가능 업무 시스템의 대상 선정, 적용 검증 및 보완

시스템 구축을 위한 구성 방안은 기존 정보 자원과의 연계, PaaS 플랫폼 전용 풀 구성(서버, 스위치, 스토리지 등)과 수평적 자원확장을

고려하여 관리과 서비스 그룹으로 분리 구성하였고, 향후 서비스 영역 확장성과 관리 인터페이스도 고려하여 설계하였다. 또한 네트워크 인프라, 라우터 노드의 독립적 구성을 통한 보안성 강화, VM 단위 Migration 등 운영 효율성도 고려하였다.

3.3 클라우드 오토-스케일링 시스템 구축

본 논문에서 제안 방법은 네트워크 클러스터, 관리 클러스터, 서비스 노드 영역으로 구성하고 x86서버(12식, CPU : Intel 36Core, MEM : 576 GB, OS : RHEV V7.5, 가상화 : RHV V.4.1), SAN 스위치(8식), 스토리지(3식) 등으로 오토-스케일링 시스템 구축하였으며 전체 시스템 구성도는 다음 [그림 2]와 같다.

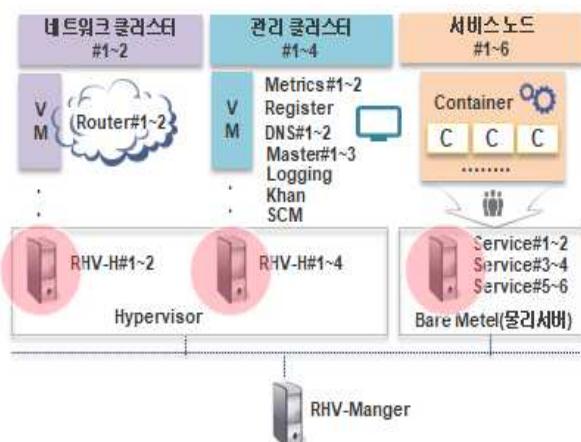


그림 2. 전체 시스템 구성도

업체 및 기관에서 정의한 업무 규칙 및 하드웨어 성능의 임계치를 통해 자원의 규모를 확장 및 축소하는 정책 기반의 자원 할당 자동화 기법을 적용하고 기준이 되는 CPU 사용량, 디스크 사용량과 같은 하드웨어의 성능치에 상한 값과 하한 값을 정하여 그 값을 기준으로 미리 정해진 수만큼 컨테이너를 추가하거나 제거하여 자원의 규모를 확장하거나 축소시킬 수 있게 하였다.

모니터링 프로세스를 통해 CPU 사용률의 임계치 기준을 마련하고 모니터링 된 CPU 값에 따라, 자원 자동 알고리즘에 의해서 필요한 만큼 WEB/WAS 수 및 설정 변경을 자동으로 적

용하여 서비스 연속성을 보장 할 수 있으며 보안 위협의 정도에 따라 등급별 추가 보안 서비스 적용이 가능 할 것이다.

IV. 결론 및 향후 연구

본 논문에서는 트래픽을 모니터링 하고 용량을 자동으로 조정하여 저렴한 비용으로 안정적이고 예측 가능한 성능 유지가 가능한 클라우드 기반 자동 자원확장 시스템을 구축하였다. 자원에 대한 규모 조정 계획을 수립하고 성능과 비용을 최적화 하거나 적절한 균형을 유지하기 위한 자원 자동 알고리즘을 통해 효율적으로 자원 규모를 조정 할 수 있게 하였다.

향후 연구 방향으로는 오토-스케일링 체계 적용을 위한 서비스 기준을 제시하여 확장 가능한 시스템 설계, 시스템 표준 기술 정의, 표준화된 전환 프로세스 정의 및 수립을 통해 지능적인 자동 자원확장 체계를 구축하는 것이다.

[참고문헌]

- [1] Dutta, S., Gera, S., Verma, A., Viswathan, B., "Smart Scale : Automatic Application Scaling in Enterprise Clouds", Proceedings of 2012 IEEE 5th International Conference on Cloud Computing, Jun. 2012, PP. 221~228,
- [2] Ming Mao, Marty Humphrey, "Auto - Scaling to Minimize Cost and Meet Application Deadlines in Cloud Workflows", SC11, Nov. 2011.
- [3] 강혜정 외 2, "과학계산 응용 실행을 위한 하이브리드 클라우드에서의 SLA기반 VM 오토-스케일링 기법", 정보과학회논문지 40권 6호, PP.266~273.
- [4] 김서영 외 6, "대규모 계산처리 서비스를 위한 가상머신 오토-스케일링 최적화 연구", 한국컴퓨터종합학술대회, 2015.06, PP.86~88.
- [5] Amazon Auto-Scaling Service, <http://aws.amazon.com/autoscaling/>

Wireless Sensor Network 환경에서의 윈도우 기반 에너지 소모량 예측을 통한 이상 노드 탐지 기법

신진명*, 최석환*, 최윤호*

*부산대학교 전기전자컴퓨터공학과

sinrayng@pusan.ac.kr

Anomaly Node Detection Method through Window based Energy Prediction in Wireless Sensor Network Environments

Jinmyeong Shin, Seok-Hwan Choi*, Yoon-Ho Choi*

*Dept. of Computer Science and Engineering, Pusan National University.

요약

Wireless Sensor Network(WSN) 환경은 그 응용 가능성으로 인해 다양한 분야에 적용되고 있으나 적은 배터리, 낮은 연산능력, 무선 환경을 통한 통신 등의 한계로 인해 보안이 취약하다. 이를 해결하기 위해 Intrusion Detection 기반의 효율적인 보안 기법들에 대한 연구가 진행되었지만 중앙 집중적 구조, 높은 연산 요구량 등 여러 한계점을 가지고 있다. 따라서 본 연구는 기존 연구들의 한계점을 해결하는 새로운 방식의 이상 노드 탐지기법을 제안한다. 제안된 기법은 일정 시간동안의 에너지 소모량에 기반해 주변 노드들의 정상 행위를 학습하고 각 노드들의 이상행위를 탐지한다. 또한 시뮬레이션을 통해 제안하는 기법의 성능을 보인다.

I. 서 론

WSN은 그 활용성으로 인해 다양한 분야에서 적용되고 있다. 그러나 적은 배터리, 낮은 연산능력, 단순한 구조 등 태생적 한계로 인해 보안에 상당히 취약하며 그로인해 다양한 공격의 대상이 된다.

Flooding, Sybil, Wormhole, Sinkhole, Gray hole 등 다양한 공격 중 대부분은 암호화 기반의 인증 기법들로 사전에 예방이 가능하지만, WSN 기기들의 낮은 연산능력과 적은 배터리로 인해 인증에 필요한 암호 연산을 수행하는 것은 상당한 부하를 초래한다.

이런 문제들을 회피하면서도 WSN 노드들의 취약성을 해결하기 위한 방법으로 이상 노드들을 탐지하는 Intrusion Detection System(IDS) 기반의 연구들이 진행 되었다[1].

G. Han와 4인은 에너지 소모량 예측만을 통해 이상 노드를 탐지하는 기법[2]을 제안하였으나 중앙 집중적인 구조와 Markov 체인 적용에

따른 상당한 연산 부하가 존재한다. N. Dharini 외 2인은 Ant Colony 알고리즘을 통한 에너지 소모량 예측 기반의 경량 이상 노드 탐지 방법을 제안하였다[3]. 하지만, WSN을 구성하는 네트워크가 단일 종류의 기기로 구성되어야 하는 한계점이 존재한다.

본 논문은 기존 연구의 한계점을 극복하는 윈도우 기반 에너지 소모량 예측을 통한 이상 노드 탐지 기법을 제안한다. 일정 시간 윈도우 동안 인접 노드의 배터리 잔량을 수집하고 이를 통해 이상 에너지 소모 패턴 파악 및 이상 노드를 색출한다.

본 논문의 구성은 다음과 같다. 2장에서 제안하는 기법에 대해 설명하고, 3장은 실험환경 및 실험 결과에 대해 논의 한다. 마지막으로, 4장에서 결론을 서술한 뒤 논문을 끝맺는다.

II. 제안 기법

본 장에서는 제안하는 윈도우 기반 에너지

소모량 예측 방법 및 이상 노드 탐지 방법을 서술한다.

2.1 원도우 기반 에너지 소모량 예측기법

네트워크에 있는 노드들은 주변 인접노드들에게 주기적으로 자신의 배터리 잔량을 알려주며 메시지를 전달받은 노드는 해당 메시지를 각자 보유한 인접 노드 n 에 대한 원도우 w_n 에 저장한다. 각 원도우에 저장된 배터리 잔량을 이용해 $t+1$ 번째와 t 번째 사이의 에너지 소모량 $C_t^{w_n}$ 을 다음과 같이 계산된다.

$$C_t^{w_n} = P_t^{w_n} - P_{t+1}^{w_n}$$

t 번째 노드 n 의 배터리 잔량 $P_t^{w_n}$ 까지의 정보를 알고 있을 때 다음 배터리의 잔량 $P_{t+1}^{w_n}$ 의 예측치 $P_{pred}^{w_n}$ 은 다음과 같이 계산된다.

$$P_{pred}^{w_n} = P_t^{w_n} - \sum_{i=1}^N \frac{C_{t-N+i}^{w_n}}{N}$$

여기서 N 은 원도우 w_n 의 크기를 나타낸다.

2.2 이상 노드 탐지 기법

이상 노드의 탐지 과정에서 인접 노드가 이상적 예측치와 다른 배터리 잔량을 보유하는 것만으로 해당 노드를 이상 노드로 판단 경우를 방지하기 위한 문턱 값을 설정할 필요가 있다. 이를 위한 문턱 값을 T_{w_n} 은 다음과 같이 계산된다.

$$T_{w_n} = \frac{\max_{C_{w_n}} + \min_{C_{w_n}}}{2}$$

여기서 $\max_{C_{w_n}}$ 과 $\min_{C_{w_n}}$ 는 각각 원도우 w_n 의 에너지 소모량 중 최대, 최소값을 나타낸다. 위의 공식을 통해 구해진 문턱 값을 이용하여 다음과 같이 노드 n 의 이상 여부를 판단한다.

$$status_n = \begin{cases} Anomaly, & \text{if } diff_{t+1}^{w_n} > T_{w_n} \\ Normal, & \text{otherwise} \end{cases}$$

여기서 $diff_{t+1}^{w_n}$ 는 $t+1$ 번째에 받은 신제 배터리 잔량과 기준의 배터리 예측량 사이의 차이를 나타내며 다음의 수식을 통해 계산된다.

$$diff_{t+1}^{w_n} = |P_{t+1}^{w_n} - P_{pred}^{w_n}|$$

즉, 예측량과 실제 배터리 잔량사이의 차가 문턱 값을 넘어가면 해당 노드가 이상행위를 하고 있는 것으로 판단한다.

III. 실험 결과

3.1 실험 환경

본 실험은 OMNET++ 5.4.1 및 Inet 4를 이용해 시뮬레이션 되었으며 100mX 100m 환경에서 2종류의 정상노드 40기, 비정상 노드 10기를 통해 실험하였다.

3.2 실험 결과

표 1. 탐지 결과

		Detection	
		positive	negative
Real	positive	8	2
	negative	3	37

비정상 행위를 수행하는 노드는 총 2종류로 각각 5개씩 Flood 공격과 Black Hole 공격을 수행하였다. 표 3은 정상 노드들 이상노드 탐지 결과로 Accuracy 0.9, Recall 0.727, Precision 0.8의 결과를 나타냄을 확인할 수 있다.

IV. 결 론

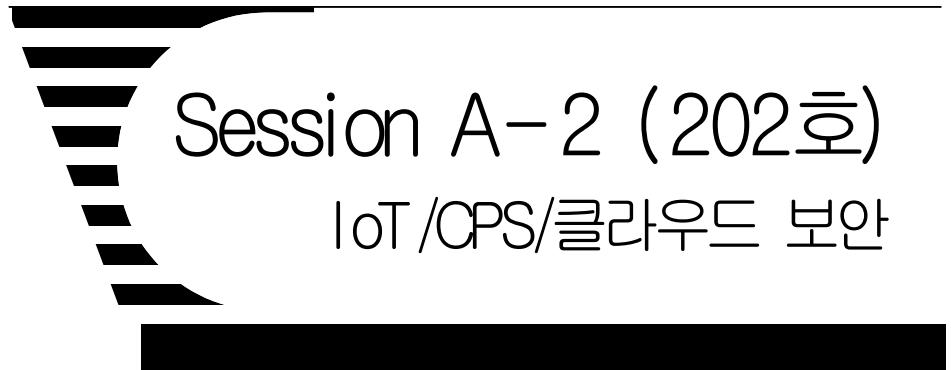
본 논문에서는 기존의 이상 노드 탐지 기법들의 한계점을 개선한 시간 원도우 기반의 이상행위 탐지 기법을 제안하였다. 제안하는 기법은 정상 노드들의 행동 패턴을 적은 비용으로 학습하고 비정상 노드의 에너지 사용량을 예측량과 비교함으로써 이상 노드를 탐지하며 시뮬레이션을 통해 그 결과를 보였다.

Acknowledgement

본 연구는 한국연구재단 논문연구과제 (NRF-2018R1D1A3B07043392) 지원으로 수행되었습니다.

[참고문헌]

- [1] N. Schweitzer, A. Stulman, R. D. Margalit and A. Shabtai, Contradiction Based Gray-Hole Attack Minimization for Ad-Hoc Networks, IEEE Transactions on Mobile Computing, Vol. 16, No.8, Aug. 2017.
- [2] G. Han, J. Jiang, W. Shen, L. Shu and J. Rodrigues, IDSEP: a novel intrusion detection scheme based on energy prediction in cluster-based wireless sensor networks, IET information Security, Oct. 2012
- [3] N. Dharini, R. Blaakrishnan and A. P. Renold, Distributed Detection of Flooding and Gray Hole Attacks in Wireless Sensor Network, 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials(ICSTM), Chennai, T.N., India, May 2015, pp.178–184



좌장 : 이임영 (순천향대)

클라우드 스토리지 환경에서 공모공격을 방지하는 ID 기반 프록시 재암호화 기술⁺

김원빈*, 이임영**

*순천향대학교 컴퓨터학과, **순천향대학교 컴퓨터소프트웨어공학과

*wbkim29@sch.ac.kr, **imylee@sch.ac.kr

ID-based proxy re-encryption scheme to prevent collusion attacks
in cloud storage environment

Won-Bin Kim*, Im-Yeong Lee**

*Dept of Computer Science and Engineering, Soonchunhyang University

**Dept of Computer Software Engineering, Soonchunhyang University

요약

최근 클라우드 스토리지가 보급됨에 따라 다양한 영역에서 데이터를 저장, 공유, 이용에 클라우드 스토리지를 활용하고 있다. 하지만 클라우드 스토리지는 항상 데이터의 유출 위험을 가지고 있다. 따라서 클라우드 스토리지 환경에서 데이터의 복호화 권한을 위임하여 데이터를 보다 안전하게 관리하는 기술인 프록시 재암호화가 제안되었다. 하지만 위임자의 ID를 공개키로 사용하는 ID 기반 프록시 재-암호화 기술에서는 위임자와 프록시 서버가 공모하여 데이터 제공자의 개인키를 복원하거나 재암호화키를 위조할 수 있는 공모공격의 위협이 제기되었다. 이를 해결하기 위해 본 연구에서는 위임자와 프록시의 공모가 불가능한 방법을 제시한다.

I. 서론

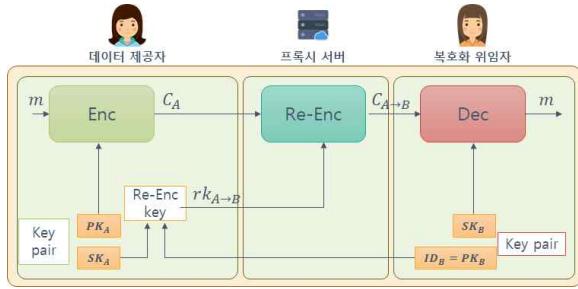
클라우드 스토리지는 데이터를 원격지(Proxy) 서버에 보관한 뒤 네트워크를 통해 데이터를 이용할 수 있는 기술이다. 이러한 클라우드 스토리지는 다양한 환경에 적용될 수 있으나 Honest-but-curious의 속성을 가지고 있기 때문에 항상 데이터가 유출될 수 있는 위험을 안고 있다.

프록시 재암호화는 데이터 제공자가 데이터를 암호화 한 뒤, 프록시 서버에서 암호화된 데이터를 재암호화하여 위임자에게 복호화 권한을 제공하는 기술이다. 이 기술에서 프록시 서버는 데이터의 원본을 알 수 없으면서도 위임자가 복호화할 수 있는 형태로 데이터를 재암호화 할 수 있어야 한다. 따라서 프록시 재암호화

기술은 데이터 제공자인 Alice가 복호화 위임자 Bob에게 데이터를 공유하는 환경에 적용될 수 있다. 하지만 복호화 위임자 Bob이 프록시 서버와 공모하여 재암호화키로부터 데이터 제공자 Alice의 개인키를 추출하거나, 새로운 재암호화 키를 위조할 수 있는 위협이 제시되었으며, 이를 공모공격이라고 한다.

공모공격은 데이터 소유자의 공개키로 암호화된 데이터를 복호화 위임자의 개인키로 복호화 할 수 있도록 하는 재암호화 키를 위조하기 위한 공격이다. 따라서 재암호화키만을 보유한 프록시 또는 자신의 공개키쌍을 가지고 있는 복호화 위임자 혼자서는 공격을 수행할 수 없다. 따라서 본 연구에서는 프록시와 복호화 위임자의 결탁을 통해 서로가 보유한 정보를 결합하여 새로운 재암호화키를 생성할 수 없도록 하여 공모공격을 방지한다.

+ 이 논문은 2016년도 정부(교육부)의 재원으로
한국연구재단의 지원을 받아 수행된 기초연구
사업임(NRF-2016R1D1A1B03935917)



(그림 30) ID 기반 프록시 재암호화 개요

II. 관련 연구

2.1 프록시 재암호화

프록시 재암호화 기술은 Alice가 소유한 데이터를 프록시가 재암호화 하여 Bob에게 복호화 권한을 제공하는 기술을 의미한다[2][3]. 프록시 재암호화 기술은 다양한 형태가 존재하며, 이 중 ID 기반 프록시 재암호화 기술은 사용자의 ID를 공개키로 이용하여 재암호화 하는 형태이다.

(그림 1)과 같이 ID 기반 프록시 재암호화는 복호화 위임자 Bob의 ID를 공개키로 사용하여 재암호화 키 $RK_{A \rightarrow B}$ 를 생성한다. 그리고 프록시는 $RK_{A \rightarrow B}$ 를 이용하여 Bob의 암호문 C_A 를 복호화 할 수 있도록 재암호화를 수행한다. 이 과정에서 프록시는 C_A 를 복호화하지 않기 때문에 메시지 원본 m 을 알 수 없다[1].

2.2 공모공격

ID 기반 프록시 재암호화 기술에서는 재암호화 키의 생성 시 데이터 제공자의 개인키와 복호화 위임자의 공개키가 이용된다. 이러한 점을 이용하여 프록시 서버와 복호화 위임자가 서로 결탁하여 공모공격을 수행할 수 있다. 프록시 서버는 데이터 소유자로부터 재암호화 키 $RK_{A \rightarrow B}$ 를 수신하며, 복호화 위임자는 자신의 개인키 SK_B 를 소유한다. 이러한 상황에서 프록시 서버는 복호화 위임자에게 재암호화 키 $RK_{A \rightarrow B}$ 를 제공할 경우 복호화 위임자는 SK_B 를 추출하거나 $RK_{A \rightarrow C}$ 를 생성할 수 있다. 따라서 공모공격을 방지하기 위해서는 프록시와 복호화 위임자의 정보를 결합하여 데이터 제공자의 개인키의 추출 또는 제 3의 재암호화 키를 위조할 수 없도록 설계하여야 한다.

III. 시스템 모델

본 장에서는 제안방식의 시스템모델을 설명하고, 이에 따른 보안 요구사항을 확인한다.

3.1 Hardness Assumption

본 연구는 다음과 같은 가정을 기반으로 한다. 여기에서는 소수 위수 q 에 대한 순환군 G_1 , G_T 가 주어졌을 때, 겹선험 사상에 의해 $\hat{e}: G_1 \times G_1 \rightarrow G_T$ 가 성립함을 기반으로 한다[2].

- m-Computational Diffie-Hellman (m-CDH) Assumption : G_1 상에서 $a, b \in \mathbb{Z}_q^*$ 이며, 튜플 $(g, g^a, g^b, g^{1/b}, g^{a/b}) \in G_1 \times G_1 \times G_T \times G_T \times G_T$ 주어졌을 때, 공격자는 $g^{ab} \in G_1$ 을 계산할 수 없어야 함
- Decisional Bilinear Diffie-Hellman (DBDH) Assumption : G_1, G_T 상에서 $a, b, c \in \mathbb{Z}_q^*$ 이며, 튜플 $(g, g^a, g^b, g^c, T) \in G_1 \times G_1 \times G_1 \times G_T \times G_T$ 주어졌을 때, 공격자는 T 가 $T = \hat{e}(g, g)^{abc}$ 인지 G_T 상의 무작위 값인지 결정할 수 없어야 함
- m-Decisional Bilinear Diffie-Hellman (m-DBDH) Assumption : G_1, G_T 상에서 $a, b, c \in \mathbb{Z}_q^*$ 이며, 튜플 $(g, g^a, g^{1/a}, g^{1/b}, g^{a/b}, g^b, g^c, T) \in G_1 \times G_1 \times G_1 \times G_1 \times G_1 \times G_1 \times G_T$ 주어졌을 때, 공격자는 T 가 $T = \hat{e}(g, g)^{abc}$ 인지 G_T 상의 무작위 값인지 결정할 수 없어야 함

3.2 보안 요구사항

본 연구는 다음과 같은 사항을 요구한다.

- 기밀성(Confidentiality) : 클라우드 스토리지에 업로드된 데이터는 정당한 소유자 이외에는 원본을 확인할 수 없도록 해야 한다.
- 단방향성(Unidirectional) : 소유자가 위임자의 암호문을 해독하지 못하면서도 소유자가 위임자에게 복호화 권한을 위임할 수 있도록 해야 한다.
- 공모공격 저항(Collusion resistance) : 프록시 서버와 복호화 위임자의 결탁을 통해 재암호화 키의 위치 또는 데이터 제공자의 개인키가 복구될 수 없어야 한다.

IV. 제안방식

본 장에서는 제안방식의 시스템모델을 설명하고, 이에 따른 보안 요구사항을 확인한다.

4.1 Setup(1^λ)

공개 파라미터를 생성하여 분산된 사용자에게 공개하며, 마스터 비밀 키(MSK)를 생성하여 안전하게 보관한다.

4.2 KeyGen($param, MSK, id_A$)

PKG는 공개 파라미터와 마스터 비밀 키, id_A 를 입력하여 사용자의 개인키와 공개키를 출력하고 안전한 채널로 사용자에게 전송한다.

4.3 Encrypt($param, id_A, m$)

데이터 제공자는 공개 파라미터, 자신의 id_A 와 함께 암호화하려는 메시지 m 을 입력하여 first-level 암호문 $C = (C_1, C_2, C_3, C_4, C_5)$ 를 획득하고, 이를 프록시 서버에 저장한다.

4.4 RKGen($param, sk_{id_A}, id_B$)

데이터 제공자는 공개 파라미터와 자신의 개인키 sk_A 와 복호화 위임자의 id_B 를 입력하여 재암호화 키 $rk_{id_A \rightarrow id_B}$ 를 생성한다.

4.5 Re-encrypt($param, rk_{id_A \rightarrow id_B}, C$)

프록시 서버는 암호문 C 와 $rk_{id_A \rightarrow id_B}$ 를 이용해 재암호화를 수행하여 second-level 암호문 $D = (D_1, D_2, D_3, D_4, D_5)$ 을 획득한다.

4.6 Decrypt($param, sk_{id_{A \rightarrow B}}, C \text{ or } D$)

복호화 단계에서는 암호문이 first-level 암호문인지, second-level 암호문인지 판단하여 복호화를 수행한다.

first-level 암호문 : $param, sk_{id_A}, C$ 를 입력받아 암호문 검증을 수행한 뒤, $m = C_4 \oplus H_3(\frac{C_3}{e(C_2, sk_{id_A})})$ 를 통해 원본 메시지 m 을 복호화 한다.

second-level 암호문 : $param, sk_{id_B}, D$ 를 입력받아 암호문 검증을 수행한 뒤, $m = D_4 \oplus H_3(\frac{D_3}{e(D_1, D_5)^{x_{AB}}})$ 를 통해 원본 메시지 m 을 복호화 한다.

V. 제안방식 분석

□ 기밀성(Confidentiality) : First-level 암호문은 데이터 제공자의 키로 암호화되어 데이터 제공자만이 복호화 할 수 있다. 또한 First-level 암호문을 복호화 위임자의 공개키로 재암호화 할 경우 복호화 위임자만이 복호화 할 수 있다.

□ 단방향성(Unidirectional) : 데이터 제공자는 복호화 위임자의 공개키를 취득하여 재암호화 키 $rk_{A \rightarrow B}$ 를 생성할 수 있다. 하지만 $rk_{A \rightarrow B}$ 를 이용하여 $rk_{B \rightarrow A}$ 를 생성할 수 없다.

□ 공모공격 저항(Collusion resistance) : 재암호화 키 $rk_{A \rightarrow B}$ 와 복호화 위임자의 키 쌍을 이용해 $rk_{A \rightarrow C}$ 또는 sk_A 를 생성할 수 없다. 따라서 프록시와 복호화 위임자의 공모를 통한 공격이 불가능하다.

VI. 결론

본 제안방식은 ID 기반 프록시 재암호화 환경에서 프록시와 복호화 위임자가 공모하여 재암호화 키 위조 또는 데이터 제공자의 개인키를 복구하는 위협을 방지한다. 이를 위해 데이터 제공자는 무작위 파라미터를 생성하고, 재암호화 키에 삽입하여 제 3자가 재암호화키를 변환할 수 없도록 하였다. 이를 통해, 본 제안방식은 클라우드 스토리지 환경에서 데이터 제공자에게 더욱 향상된 안전성을 제공할 수 있다.

[참고문헌]

- [1] 박승환, et al. "스마트카드를 이용한 프록시 재 암호화 기법 기반 콘텐츠 공유 메커니즘에 관한 연구." 정보보호학회논문지 21.3 (2011): 131-141.
- [2] Paul, Arinjita, et al. "A CCA-Secure Collusion-Resistant Identity-Based Proxy Re-Encryption Scheme." International Conference on Provable Security. Springer, Cham, (2018). pp. 111-128.

하드웨어를 공유하는 LEA 및 HIGHT

블록 암호 저면적 구현 기법

김보훈* 박종선*

*고려대학교 전기전자공학과

rlaqhgns0616@korea.ac.kr jongsun@korea.a.ckr

Low area implementation of unified hardware for
LEA and HIGHT block cipher

Bohun Kim*, Jongsun Park*

*School of Electrical Engineering, Korea University.

요약

국내 표준 암호 알고리즘인 LEA와 HIGHT는 경량 환경에서의 암호화를 위해 설계된 ARX(Add-Rotation-Xor)구조의 블록 암호이다. 본 논문에서는 두 암호 알고리즘을 하나의 하드웨어로 결합하여 구현하는 기법을 제안한다. 제안하는 하드웨어는 적은 면적 비용으로 필요에 따라 LEA-128 동작 또는 HIGHT 동작을 하도록 모드를 선택할 수 있다. 구현 결과는 Samsung 65nm CMOS 공정 라이브러리를 이용하여 합성되었고, 8650GE의 면적을 달성하여 개별 구현 대비 12%의 면적 효율을 얻었다.

I. 서론

오늘날과 같은 정보화 시대에서는 통신에서의 데이터 암호화가 필수적인 요소가 되었다. 암호화 시스템은 크게 대칭키 암호와 공개키 암호로 나뉘며, 각 분야에 다양한 암호 알고리즘이 제시되었다. 사용자는 상황과 용도에 따라 적절한 암호화 방식을 선택할 수 있다.

대표적인 대칭키 암호는 미국에서 국제 표준으로 지정된 블록 암호인 AES(Advanced Encryption Standard)이다. 우리나라에서 개발된 블록 암호로는 SEED, ARIA, HIGHT 및 LEA가 있다. 이들 중 HIGHT와 LEA는 특히 경량 구현에 용이하도록 설계된 ARX(Add-Round-Xor) 구조이다. 두 알고리즘은 하드웨어 상에서 유사한 방식으로 설계되며, 현재 가장 널리 쓰이는 128비트의 키 사이즈를 지원한다.

본 논문에서는 128비트 키 스케줄러 및 라운

드 연산에서 하드웨어를 부분적으로 공유함으로써 HIGHT와 LEA-128이 통합된 저면적 암호화 하드웨어 구조를 제안한다. 제안한 하드웨어는 Samsung 65nm CMOS standard cell 라이브러리로 합성하여 면적과 성능을 분석하였다.

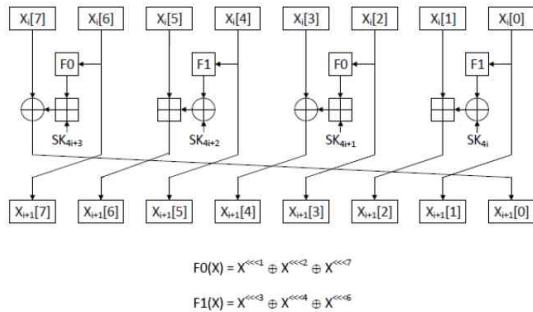
본 논문의 구성은 다음과 같다. 2장에서 HIGHT와 LEA 암호화 알고리즘에 대해서 설명하고, 3장에서 이 두 가지 알고리즘을 결합한 구조를 제안한다. 4장에서는 합성한 결과를 측정하고 분석하며, 끝으로 5장에서 결론을 맺는다.

II. 암호 알고리즘

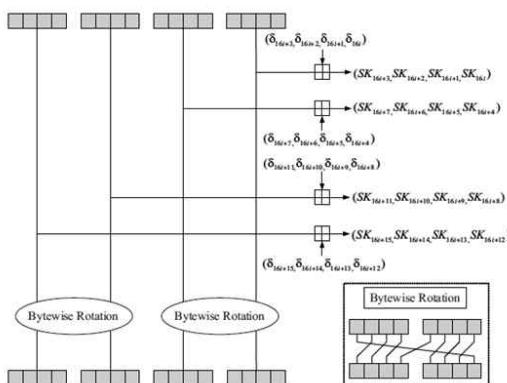
2.1 HIGHT

HIGHT는 64비트 데이터와 128비트 키를 사용하는 ARX 구조의 블록 암호 알고리즘이다. 라운드 연산은 8비트(1바이트) 단위로 이루어지

며 총 34라운드의 연산을 통해 암호화가 이루어진다.

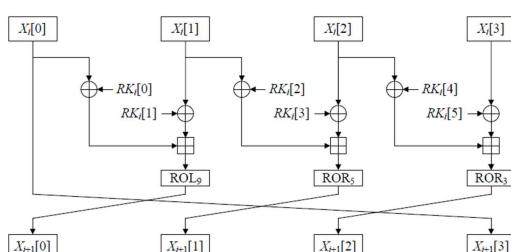


[그림 31] HIGHT 라운드 연산

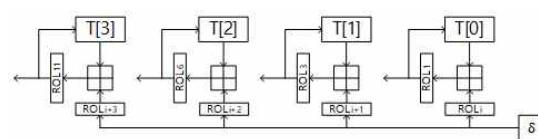


[그림 32] HIGHT 키 스케줄링

2.2 LEA



[그림 33] LEA 라운드 연산



[그림 34] LEA-128 키 스케줄링

LEA는 128비트 데이터와 128/196/256비트 키를 사용하는 블록 암호 알고리즘이다. 라운드 연산은 32비트 단위로 이루어지며 키의 길이에

따라 각각 24/28/32라운드의 연산을 수행한다. 본 논문에서는 HIGHT의 규격에 맞추어 128비트 키를 가지는 LEA-128을 대상으로 설계하였다.

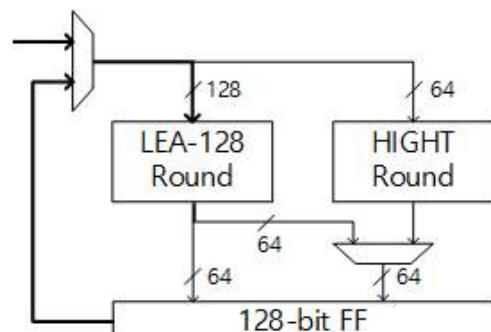
III. HIGHT와 LEA의 결합 구조

하드웨어 공유의 기본적인 방식은 MUX(multiplexer)를 이용하여 공통된 하나의 모듈에 여러 가지 입력이 들어갈 수 있도록 만드는 것이다. 이때 공유되는 모듈의 면적이 추가되는 MUX의 면적보다는 더 커야 하드웨어 공유의 면적 절감 효과가 있다. [표 1]은 Samsung 65nm CMOS standard cell 라이브러리에서 서로 다른 하드웨어들의 상대적인 면적을 나타낸 표이다. [표 1]에서 보듯이, XOR 게이트의 경우는 2to1 MUX보다 작은 면적을 가지므로, XOR 게이트를 공유하기 위해 MUX를 사용하는 것은 적절하지 못하며, 덧셈이나 플립플롭을 공유할 때에 효과를 볼 수 있다.

Module	GE	Tech.
NAND	1	Samsung 65nm
XOR	2.25	
Add	6.75	
2to1 MUX	3	
Flip-flop	6	

[표 1] NAND를 기준으로 한 상대 면적

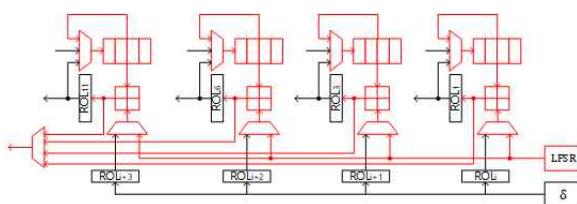
우리의 설계는 라운드 기반 즉, 한 클락 사이클에 한 라운드의 연산을 하는 하드웨어 구현을 기준으로 삼았으며, HIGHT와 LEA의 암호화 과정을 구현하였다.



[그림 35] HIGHT와 LEA-128이 결합된 하드웨어 구조

라운드 함수의 경우 HIGHT와 LEA의 전체적인 구조는 유사하지만 연산 단위가 HIGHT는

8비트, LEA는 32비트로 서로 다른 단위를 가진다. 따라서 LEA의 라운드 연산을 HIGHT 연산에 사용할 수 있더라도 데이터 패스를 잘 컨트롤 해주어야한다. 우리는 공유되는 면적보다 추가되는 MUX 및 컨트롤에 필요한 면적이 더 작을 경우에만 하드웨어 공유를 사용하였다. 이에 따라 라운드 연산에서는 [그림 5]와 같이 64비트 레지스터만을 공유하고 핵심 연산부는 LEA와 HIGHT를 분리하여 설계하였다.



[그림 36] 결합된 키 스케줄러 구조

키 스케줄의 경우 LEA와 HIGHT 모두 라운드마다 특정 상수와 기존의 키를 덧셈하는 연산을 포함한다. 따라서 키 스케줄러에서는 [그림 6]과 같이 128비트 레지스터뿐만 아니라 이 덧셈에 사용되는 덧셈기까지 공유하여 더욱 많은 면적을 절약할 수 있었다. 라운드 함수와 마찬가지로 LEA의 32비트 단위 연산을 HIGHT에서는 8비트 연산 4개로 나누어주어야 한다. 그림에서 빨간색으로 표시한 부분은 HIGHT 연산시에 사용되는 데이터 패스를 나타낸 것이다.

IV. 구현결과

위와 같은 구현 결과를 Samsung 65nm CMOS 공정 라이브러리를 이용하여 Synopsys Design Compiler로 합성하였다. 기존에 설계된 HIGHT와 LEA는 각각 다른 공정 라이브러리를 이용하여 합성되었다. 라이브러리에 따라 게이트의 상대적인 사이즈가 다르므로 공정한 비교를 위해 각각의 알고리즘을 Samsung 65nm 100MHz의 조건에서 설계한 결과를 바탕으로 면적을 비교 분석하였다.

Design	GE	Tech.
HIGHT	3048 [1]	250nm
	3000	Samsung

		65nm
LEA-128	5426 [2]	UMC 130nm
	6780	Samsung 65nm
HIGHT+	8650	Samsung 65nm

[표 2] 기존 결과물과의 면적 비교

[표 2]에 따르면 Samsung 65nm 공정에서 라운드 기반 설계를 했을 때 HIGHT는 3000GE, LEA-128은 6780GE의 면적을 가졌다. 우리가 설계한 HIGHT와 LEA-128이 결합된 하드웨어는 8650GE로 설계되었다. 따라서 결합된 하드웨어는 개별 면적을 합친 9780GE에 비해 12%의 면적을 절약할 수 있었다.

V. 결론

본 논문에서는 HIGHT와 LEA-128의 통합된 하드웨어 구조를 제안하였고, 개별적인 구현물에 대한 면적 효율성을 보였다. 필요에 따라 원하는 암호화를 수행하도록 동작을 선택할 수 있다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터(UD170109ED)를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- [1] Hong, Deukjo, et al. "HIGHT: A new block cipher suitable for low-resource device." International Workshop on Cryptographic Hardware and Embedded Systems. Springer, Berlin, Heidelberg, 2006.
- [2] Hong, Deukjo, et al. "LEA: A 128-bit block cipher for fast encryption on common processors." International Workshop on Information Security Applications. Springer, Cham, 2013.

최신 IoT 악성코드의 침해사례 및 특징 분석

정해선*, 최슬기**, 곽진***

*,**아주대학교 사이버보안학과

**정보보호응용및보증연구실, 아주대학교 컴퓨터공학과

*haeseon@ajou.ac.kr, **skchoi.isaa@gmail.com, ***security@ajou.ac.kr

Case and Feature Analyses of The Latest IoT Malware

Hae-Seon Jeong*, Seul-Ki Choi**, Jin Kwak***

*,***Department of Cyber Security, Ajou University

**ISAA Lab., Department of Computer Engineering, Ajou University

요약

IoT 기기의 수요는 계속해서 증가하고 있고 많은 분야에서 사용되고 있다. 하지만 IoT 기기에 대한 보안 규정 및 보안 설정 등에 대한 가이드라인이 명확하지 않아 많은 보안 문제가 존재한다. 공격자들은 이러한 IoT 기기의 취약점을 이용해 다양한 악성코드를 만들어 공격을 시도하고 있다. 더욱 정교화·고도화되고 있는 악성코드로부터 더 안전한 IoT 환경을 구축하기 위해, 본 논문에서는 최신 악성코드 사례분석을 통해 다양한 IoT 악성코드들의 공격 과정, 특징을 분석하고 대응방안 및 IoT 보안 연구의 필요성을 기술하고자 한다.

I. 서론

IoT(Internet of Things)는 전 세계적으로 보급되고 있으며, 2019년에는 142억 개의 IoT 기기가 사용될 것이라 예상된다[1]. 하지만 IoT 기기에 대한 보안 규정이나 표준이 부재하여 기본 암호 정책, 접근 제어 권한, 보안 구성과 같은 설정들이 미흡하여 다양한 보안 취약점들이 존재하고 있다[2]. 따라서, 최근에는 IoT 기기를 대상으로 하는 신·변종 악성코드들이 증가하고 있다. 카스퍼스키랩에서 2018년 09월에 발표한 자료에 의하면 2018년 상반기에 발생한 IoT 관련 악성코드의 수가 2017년에 발생한 전체 악성코드의 약 3배가량 증가한 것을 확인할 수 있다[3].

본 논문에서는 최신 IoT 악성코드

사례분석을 통해 악성코드의 동작 방식과 공격 목적 등의 특징을 분석하고, IoT 기기에 대한 보안의 필요성과 대응방안을 제시한다.

II. 최신 IoT 악성코드 사례분석

2.1 Mirai

2016년 10월 21일, IoT 기기를 대상으로 하는 Mirai 악성코드는 미국의 DNS 업체인 Dyn에 대규모 DDoS 공격을 수행하여 트위터, 넷플릭스 등 총 76개의 사이트를 마비시키거나 서비스를 지연시키는 사건을 발생시켰다[4].

이러한 Mirai 악성코드는 Mirai bot, C&C server, scan receiver, loading server로 구성되어 있다[5].

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. NRF-2017R1E1A1A01075110).

2.1.1 Mirai의 동작 과정[5][6]

- ① Mirai는 원격에서 접근 가능한 서비스인 Telnet(23번 포트), SSH(22번 포트)를 스캔해서 해당 포트가 열려있는 기기를 검색
- ② 해당 서비스가 열려있는 경우, 공장 출하 시 설정되어있는 많은 초기 계정 정보들을 대입하는 사전식 전사공격 수행
- ③ 구매 후 초기 계정 정보를 변경하지 않은 IoT 기기들에 대하여 Shell에 접근
- ④ 기기 접근이 성공하고 나면 Scan receiver로 공격에 성공한 기기의 IP주소, 계정 정보를 전송하고, scan receiver는 받은 정보를 loading server로 다시 전송
- ⑤ loading server에서는 busybox의 “wget, tftp, ECCHI, echo -ne” 명령어로 Mirai 바이너리를 기기에 다운로드
- ⑥ 바이너리 실행 시 기존 telnet, SSH 서비스를 kill하고 restart하여, C&C(Command&Control) server에서 Telnet, SSH 포트를 통해 명령을 전송
- ⑦ C&C server는 또 다른 취약한 기기를 찾기 위하여 ①~⑥의 과정을 반복하거나 DDoS 공격을 수행

Mirai의 위와 같은 자기 복제 방법은 거대한 봇넷을 형성할 수 있었으며, 이를 대규모 DDoS 공격을 수행하는데 활용할 수 있었다[5].

2.1.2 Mirai 변종 악성코드

2016년 9월 20일 Mirai의 소스코드가 공개된 이후, 많은 Mirai 변종 악성코드들이 꾸준하게 발견되고 있다[4]. 그중에서도 Satori는 2017년 11월에 최초 발견되었으며, 2018년에 발생한 IoT botnet 악성코드 중에서 53%를 차지하였다[6].

현재 Satori 악성코드에 대한 소스코드가 공개되어 있으며, Satori 악성코드는 사전식 전사공격이 아닌 알려진 원격 코드 실행 취약점을 이용하여 기기에 접근한다. Satori 악성코드는 스캐닝을 수행하지 않고 37215번 포트와 52869번 포트로 원격 접속을 시도한다. 따라서 웹처럼 자기 자신을 스스로, 빠르게 전파할 수 있는 특징을 가지고 있다[7].

2.2 VPNFilter

VPNFilter는 2018년 상반기에 CISCO Talos 팀에서 처음 발견한 악성코드로, 다단계 모듈형 플랫폼이다. 다양한 네트워킹 장치를 공격대상으로 삼고 있는 VPNFilter는 사용자의 중요 정보뿐만 아니라 ICS(Industrial Control System) 패킷을 수집하고, 자신의 흔적을 지우고, IoT 기기를 무력화시키는 등의 공격을 수행한다. 이러한 VPNFilter는 크게 3단계의 순서로 공격을 수행한다.

- 단계 1 : linux crontab job scheduler에 자신을 기록하고, 기기의 비휘발성 메모리를 조작해서 재부팅을 해도 남아있도록 하며, 다음 단계를 위해 C&C 서버의 주소를 가져온다[8].
- 단계 2 : C&C 서버 접속 시 필요한 JSON 객체를 만들고 접속한다.
- 단계 3 : 공격에 필요한 많은 기능을 포함하고 있는 여러 모듈을 추가한다[7-9].

[표 19] VPNFilter에서 사용하는 공격 모듈

모듈 이름	모듈 기능
dstr	기기내에 있는 VPNFilter의 흔적을 지우고 사용 불가능하게 함
ps	150bytes 이상의 패킷과 ICS 트래픽 수집 80번 포트(TCP)로 가는 모든 서비스를 로컬 서비스로 리다이렉트해서 sslstrip → 이를 통해 중요 계정 정보를 추출하고, 웹 트래픽에 악성코드를 삽입할 수 있어, 이는 감염된 기기에 연결되어 있는 네트워크를 감염시킬 수 있음을 의미
ssler	ssler과 비슷하게 http 통신을 검사해서 윈도우 실행 파일 존재 여부 확인, 공격자가 손상된 기기를 통과할 때 윈도우 실행 파일을 즉시 폐기 할 수 있게 함
htpx	다기능 SSH 유ти리티
ndbr	손상된 기기의 로컬 서브넷을 스캔하고 매크로
nm	서비스 거부 유ти리티
netfilter	네트워크 트래픽을 특정 인프라에 포워딩
portforwarding	손상된 기기에 SOCKS5 프록시 생성
socks5proxy	손상된 기기에게 역-TCP VPN으로 원격 공격자가 기기 내 네트워크로 접속하게 해줌
tcpvpn	

VPNFilter는 [표 1]의 모듈을 이용하여 감염된 기기에서 다른 기기에 코드 삽입을 통해 공격할 수 있고, 프록시를 사용하기 때문에 실제 공격 근원지 파악이 어렵다. 또한, C&C 서버와 데이터추출 트래픽을 난독화·암호화하여 공격에 대한 탐지를 어렵게 한다[10]. 하지만 아직도 VPNFilter가 기기에 초기 접근을 어떤 방식으로 수행하는지 명확하게 알려진 바는 없다.

III. IoT 악성코드의 특징 분석

□ IoT botnet을 활용한 DDoS 공격 수행

감염된 기기는 C&C 서버의 명령에 따라 특정 대상을 공격하는 봇넷의 일부가 되어 DDoS 공격을 수행하게 된다.

□ 감염된 IoT 기기를 숙주로 악용

IoT 기기가 악성코드에 감염되고 나면, 다른 취약한 기기를 스캔하거나 연결된 네트워크를 공격할 수 있다.

□ IoT 기기를 통한 정보 탈취

패킷을 수집하고 개인 식별 정보 등의 주요 정보를 공격자의 서버에 전송한다. 또한, 기기가 연결되어있는 네트워크의 정보 또한 수집한다.

□ IoT 기기를 통한 ICS 정보 수집

ICS 패킷을 수집하여 ICS 보안에 상당한 위협을 제공한다.

□ IoT 기기 무력화

재부팅을 수행해도 악성코드가 삭제되지 않고, 펌웨어의 값을 덮어씌워 기기를 무력화한다.

□ 공격 트래픽 추적의 어려움

C&C 서버와 데이터추출 트래픽을 난독화·암호화하고, VPN을 사용하여 공격 근원지 파악을 어렵게 한다.

IV. 대응방안 및 결론

대부분의 IoT 악성코드들이 기기에 접근하기 위한 방식으로 사용하고 있는 사전식 전사공격은 2018년 상반기 기준으로 기기 접근방식의 약 90%를 차지하고 있다[3]. 따라서, 사용하지 않는 포트를 닫고, 공장 출하 시 설정된 초기 계정을 변경하는 것만으로도 일정 수준으로 악성코드 감염을 방지 할 수 있다.

또한, 주기적으로 기기를 재부팅·초기화함으로써 악성코드를 제거해주는 방법도 악성코드 감염을 방지할 수 있는 효과를 가져올 수 있으며[10], 이미 알려진 취약점을 이용하는 악성코드들을

방지하기 위하여 기기의 펌웨어를 최신 버전으로 업데이트하는 것이 필요하다.

하지만, 최근에 발생하고 있는 신변종 IoT 악성코드들은 재부팅에 저항성을 갖고 있으며, ICS 패킷도 수집하고 직접 감염된 IoT 기기 외에 그와 연결된 기기에도 코드를 삽입하는 등 정보 수집 및 공격 기술이 점점 더 정교화되고 있다. 또한, 감염된 기기를 무력화시킴으로써 그 피해를 증가시키며, 공격 트래픽을 난독화, 암호화하거나 프록시를 사용함으로써 공격 경로를 쉽게 추적하지 못하게 하는 등 악성 행위들이 지속적으로 고도화되고 있다.

따라서, 지속적으로 발전하고 있는 악성코드들로부터 안전한 IoT 환경을 보장하기 위하여 IoT 기기 보안에 대한 다양한 연구들이 필요하다.

[참고문헌]

- [1] Gartner, “Gartner Identifies Top 10 Strategic IoT Technologies and Trends”, Nov 2019.
- [2] Infosec, “The Top Ten IoT Vulnerabilities”, Feb 2018.
- [3] Kaspersky lab, “New trends in the world of IoT trends”, Sep 2018.
- [4] Manos Antonakakis and others, “Understanding the Mirai botnet”, 26th USENIX(Security 17), Aug 2017.
- [5] McAfee, “McAfee Labs Threats Report”, Apr 2017.
- [6] SK infosec, “2019 보안 위협 전망 보고”, Sep 2019.
- [7] Trend Micro, “Source Code of IoT Botnet Author Publicly Released on Pastebin”, Jan 2018.
- [8] CISCO Korea Blog, “전 세계 50만 대의 네트워킹 장비를 공격한 신종 VPNFilter 악성코드”, <https://ciscokrblog.com/1323>, May 2018.
- [9] CISCO Talos, “VPNFilter Update – VPNFilter exploits endpoints, targets new devices”, Jun 2018
- [10] CISCO Talos, “VPNFilter III: More Tools for the Swiss Army Knife of Malware”, Sep 2018.
- [11] 한국인터넷진흥원(KISA), “2019년 7대 사이버 공격 전망”, Dec 2018.

효율적인 데이터 공유를 위한 CP-ABE 접근제어 기법에 관한 연구⁺

황용운*, 이임영**

순천향대학교 컴퓨터학과

*hyw0123@sch.ac.kr, **imylee@sch.ac.kr

A Study on CP-ABE Access Control Scheme for Efficient Data Sharing

Yong-Woon Hwang[†], Im-Yeong Lee^{**}

Dept of Computer Science and Engineering, Soonchunhyang University

요약

최근 클라우드 컴퓨팅의 발달로 인해 사람들은 클라우드 환경에서 자신의 데이터를 저장하거나 공유할 수 있다. 하지만 클라우드 환경에서도 데이터 보안 및 개인정보보호와 같은 보안 문제가 발생한다. 이에 다양한 보안 기술이 필요하며, 그 중 속성기반 암호인 CP-ABE 방식의 접근제어 기법을 사용한다. 하지만 속성의 개수가 증가함에 따라 암호문의 크기가 증가하기 때문에, 클라우드 스토리지 저장공간의 효율을 낭비할 수 있으며, 복호화하는 사용자의 연산량이 속성의 개수에 비례하여 비효율적이다. 본 논문에서는 속성의 개수와 상관없이 암호문의 크기를 고정시켜 스토리지의 저장공간의 효율을 증가시키며, 아웃소싱 기법을 지원하여 복호화하는 사용자의 연산량의 효율을 높일 수 있는 CP-ABE 방식의 접근제어 기술을 제안하여, 클라우드 환경에서의 효율적으로 데이터를 공유할 수 있다.

I. 서론

최근 클라우드 컴퓨팅의 발달로 인해 사람들은 많은 비용이 소요되는 스토리지를 대여하는 대신 필요에 따라 클라우드 스토리지에 데이터를 저장하거나 다른 사람들과 공유할 수 있다. 하지만 클라우드 환경에서도 서비스 제공업체를 완전히 신뢰할 수 없으며, 공격자로 인해 데이터가 유출되거나 손실될 수 있다. 따라서 클라우드 환경에서 데이터 소유자의 데이터의 보안이 중요하며, 다양한 보안 기술 중 속성기반 암호인 CP-ABE(Ciphertext-Policy Attribute Based Encryption)방식이 가장 적합한 암호 기술이다. 현재까지 CP-ABE 방식의 접근제어 기술에 관한 연구는 지속적으로 연구되고 있지만 다양한 보안위협에 취약한 방식들과 효율성이

부족한 방식들이 존재한다. 특히 속성의 개수가 증가함에 따라 암호문의 크기가 증가하기 때문에, 스토리지 저장 공간의 효율을 낭비할 수 있으며, 사용자가 복호화 하는 필요한 연산량이 비효율적이다. 본 논문에서는 클라우드 환경에서의 효율적인 데이터 공유를 위해 CP-ABE 방식을 활용하여 인가된 사용자만이 안전하게 클라우드 스토리지에 접근하여 데이터를 공유할 수 있는 접근제어 기법을 제안한다. 또한 속성의 개수와 상관없이 암호문의 크기를 고정시켜 클라우드 스토리지의 저장공간의 효율을 증가시키고, 아웃소싱 기법을 지원하여 복호화하는 사용자의 연산량의 부담을 줄일 수 있다.

II. 관련연구

2.1 CP-ABE

CP-ABE 방식은 속성기반 암호중 하나이며, 송신자가 암호문을 생성 시 수신자의 속성을 기

⁺ 이 논문은 2016년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구 사업임(NRF-2016R1D1A1B03935917)

반으로 접근구조를 생성하여 암호화된 데이터와 함께 수신자에게 전송하면, 수신자는 자신의 속성 집합을 암호문의 접근구조와 비교하여 만족하면 암호문을 복호화 한다[1].

2.2 기존에 제안된 CP-ABE 접근제어 기법

2013년 Yang scheme[2]은 클라우드 환경에서 안전한 데이터 접근기법으로 사용자 탈퇴시키와 암호문을 업데이트하여 탈퇴된 사용자의 접근을 차단하였다. 하지만 암호문의 크기는 속성의 개수에 비례하여, 스토리지 저장공간이 낭비되며, 사용자가 복호화시 연산은 효율성을 속성의 개수에 비례하기 때문에 비효율적이다. 2015년 Hahn scheme[3]은 일정크기의 암호문과 복호연산의 아웃소싱을 지원하는 속성기반의 안전한 데이터 공유기법을 제안하였다. 하지만 프라이버시 클라우드 환경을 기반으로 하였기에 다른 사람과 데이터를 공유하는 페블릭 클라우드 환경에서는 적합하지 않다. 2017년 Tang scheme[4]은 일정크기의 암호문을 이용한 계층적 속성기반 접근제어 기법을 제안하였지만, Yang scheme과 마찬가지로 사용자가 복호화시 연산은 비효율적이다. 이에 사용자가 암호문을 복호화시 연산량 일부를 신뢰된 기관에서 처리하여 사용자의 복호화 연산량의 효율을 높여 줄 수 있는 아웃소싱 기법이 필요하다.

III. 제안방식

본 장에서는 클라우드 환경에서 효율적인 테

이터 공유를 위한 CP-ABE 접근제어 기법을 제안한다. [그림 1]은 본 제안방식이 적용되는 환경을 시나리오로 나타낸 그림이다.

3.1 시스템 초기화 및 데이터 암호화 단계

Step 1. 초기에 사용자는 TTP에게 등록을 진행 한 후 TTP는 공개 파라미터와 마스터키 (PK, MK)를 생성한다. 이후 사용자 속성데이터와 PK, MK 를 통해 비밀키 SK 를 생성하고, 데이터 소유자에게 PK 를 사용자에게 PK 와 SK 를 안전한 채널로 전송해준다.

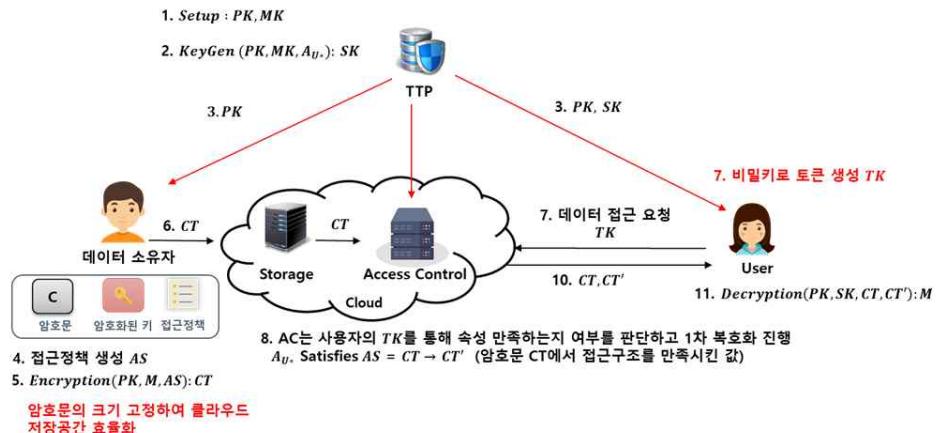
- Setup단계 : 공개파라미터 PK, MK 생성
- $KeyGen(PK, MK, A_{U*}) : SK$: 속성의 집합 A_U 와 마스터키, PK 를 통해 비밀키 SK 를 생성
- $SK : (D^{a+r}, \forall j \in S, D_j : g^r H(j)^{r_j}, D'_j : g^{r_j})$

Step 2. 데이터 소유자는 데이터를 공유할 사용자의 속성들을 기반으로 접근구조 AS 를 생성한 후 PK 와 접근구조(PK, AS)로 데이터 M 을 암호화하여 클라우드 스토리지에 저장한다.

- $Encrypt(PK, AS, m) : CT(AS, C_0, C_1, C_2)$
- $C_0 : M^* e(g, g)^{\alpha s}, C_1 : g^s, C_2 : (h \Pi_{j \in AS} g_i)^s$

3.2 사용자 데이터 접근단계

Step 3. 사용자는 TTP로부터 받은 SK 를 통해 토큰값 TK 을 생성하여, AC(Access Control)에게 접근을 요청한다.



[그림 37] 본 제안방식 시나리오

Step 4. AC는 사용자로부터 받은 TK 를 통해 1차 복호화를 진행한다. 1차 복호화하는 사용자의 속성과 암호문에 지정된 접근구조 AS 를 비교하여 만족하는지의 여부를 통해 복호화 연산을 진행하여 암호문 $CT \rightarrow CT'$ 로 변형한다. 이후 암호문 CT' 를 사용자에게 전송해준다.

□ 1차 복호화: $A_{U^*} \text{Satisfies } AS? \quad CT \rightarrow CT'$

$$CT' = e(g_i, C_2) / e(C_1, D_j^* \Pi_{j \in S} g_i) = e(g, g)^{rs}$$

Step 5. 사용자는 AC로부터 받은 CT 와 CT' 를 가지고 복호화하여 메시지 M 를 획득한다.

□ $\text{Decrypt}(PK, SK, CT, CT') : M$

$$\square M = C_0 / (e(C_1, D) CT')$$

IV. 제안방식 분석

- 인가되지 않은 사용자의 접근제어 : 본 제안방식에서 TTP에서 등록된 사용자만이 토큰을 생성하여 클라우드에 연결된 AC에 접근할 수 있다. AC에서는 사용자의 속성과 암호화된 데이터에 지정된 접근구조 속성을 비교하여 일치하는 사용자에게만 1차 복호화된 암호문 CT 와 CT' 를 전송해준다. 이에 본 제안방식에서 인가되지 않은 사용자의 접근을 차단하며, 이에 데이터에 대한 무결성 및 기밀성을 보장 할 수 있다.
- 스토리지 저장공간의 효율성 : 기존의 CP-ABE scheme은 속성의 개수에 따라 암호문의 크기가 비례하기 때문에, 속성의 개수가 많아질수록 암호문의 크기는 선형적으로 증가하여 스토리지의 저장 공간을 많이 차지하였다. 하지만 본 제안방식은 암호화하는 과정에서 C_2 연산을 통해 증가되는 속성의 수를 한가지의 속성의 수로 나타냄으로써 암호문의 크기를 일정하게 나타낼 수 있다. 이에 클라우드 스토리지 저장공간을 효율적으로 사용할 수 있다.
- 연산량 효율성 : 기존의 CP-ABE scheme에

서 복호화 연산을 사용자가 모두 연산하는 반면 본 제안방식은 아웃소싱 기법을 지원하여 복호화의 연산 일부를 AC에서 처리한다. 이에 기존 CP-ABE scheme과 비교하여 사용자가 암호문을 복호화 시 연산량의 효율을 높일 수 있다.

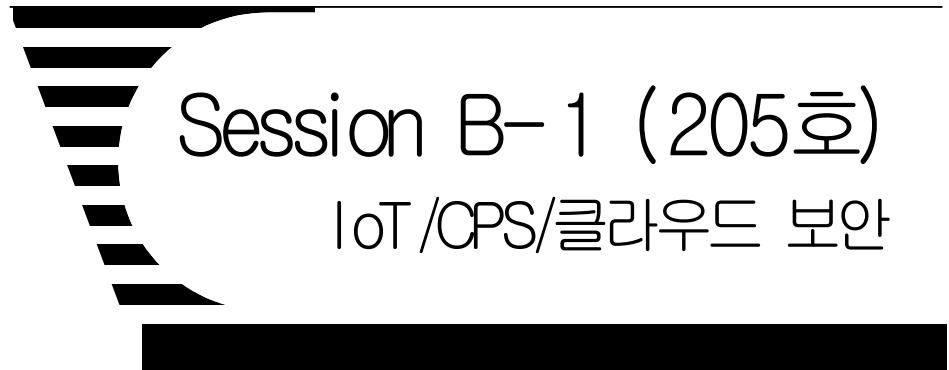
V. 결론

클라우드 환경에서 효율적인 데이터 공유를 위해 본 논문에서는 CP-ABE 기법을 활용한 접근제어 기법을 제안하였다. 제안방식은 속성의 개수와 상관없이 일정크기의 암호문을 연산함으로써 기존의 CP-ABE scheme에서 스토리지의 저장공간의 효율을 높였으며, 아웃소싱 기법을 통해 암호문을 복호화하는데 필요한 연산량의 일부를 AC에서 처리함으로써 사용자가 복호화시 연산량의 효율을 높일 수 있다.

향후 연구로는 본 논문에서 제시한 CP-ABE 접근 기법에서 사용자 탈퇴 시 발생할 수 있는 Backward security을 고려한 연구가 필요할 것으로 사료된다.

[참고문헌]

- [1] 박광용, 송유진. "속성기반 암호기술." 정보보호학회지, 제 20호 2권, pp. 85-92, 2010.
- [2] Yang, Kan, Xiaohua Jia, et al. "Attribute-based fine-grained access control with efficient revocation in cloud storage systems." Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security. ACM, 2013.
- [3] 한창희, 허준범. "고정 크기 암호 정책 속성 기반의 데이터 접근과 복호 연산 아웃소싱 기법." 정보과학회논문지, 제 43호 8권, pp. 933-945, 2016.
- [4] Teng, Wei, et al. "Attribute-based access control with constant-size ciphertext in cloud computing." IEEE Transactions on Cloud Computing 5.4 pp. 617-627, 2017.



좌장 : 김정태 (목원대)

Inversionless Berlekamp-Massey Algorithm을 이용한 BCH 부호의 하드웨어 구현

아와루딘 아셉 무하마드*, 윤영여*, 김호원*

*부산대학교 정보컴퓨터공학과

asep.muhamad11@pusan.ac.kr, yeo8006@pusan.ac.kr, howonkim@pusan.ac.kr

Hardware Implementation of Inversionless Berlekamp-Massey Algorithm for BCH Code

Asep Muhamad Awaludin*, Youngyeo Yun*, Howon Kim*

*Department of Electrical and Computer Engineering, Pusan National
University.

요약

One of widely used PUF key extraction technique is Fuzzy Extractor based on error correcting code such as BCH Code. The complexity of implementing BCH Code with hardware comes to the key equation solver in decoding step which is Berlekamp-Massey Algorithm. This paper present the hardware implementation result of inversionless Berlekamp-Massey Algorithm. The multiplication module based on LFSR is considered to reduce the hardware complexity. The implementation is evaluated by 180-nm CMOS standard cell technology. The achieved area (required number of gates) are directly impacted by the increasing number of maximum allowed bit error while 200 MHz clock rate is obtained.

I. Introduction

Error control coding is a means used to ensure the reliability of the information given from one entity to other entity. The implementation is commonly used in digital information and communication. Also, there are some implementations uses error-correcting code such storage system. The error that happen while transmitting the data can be detected and corrected on receiver side without retransmitting. The main principle of this mechanism is by computing the redundant data and sending the redundant along with data, called codeword, to the receiver (encoding). This redundant data can be used to find limited number of errors occurred on the data (decoding).

There are several well-known methods of error-correction code, Bose - Chaudhuri - Hocquenghem (BCH), Reed - Solomon, and Goppa codes. Those methods consist essentially of solving the so-called key equation [1]. It is the most complex process in decoding (e.g. BCH Decoder) which finding the error-locator polynomial $\Lambda(x)$. Some algorithms have been introduced to find the most efficient and reliable algorithm such as Euclidean algorithm, Sugiyama algorithm, and Berlekamp-Massey Algorithm (BMA). BMA has generally least hardware complexity and cost compared to other algorithm [2].

This paper mainly focused on hardware implementation of BMA Algorithm which is specialized for key equation solver in BCH Decoder. Since the inversion over finite field

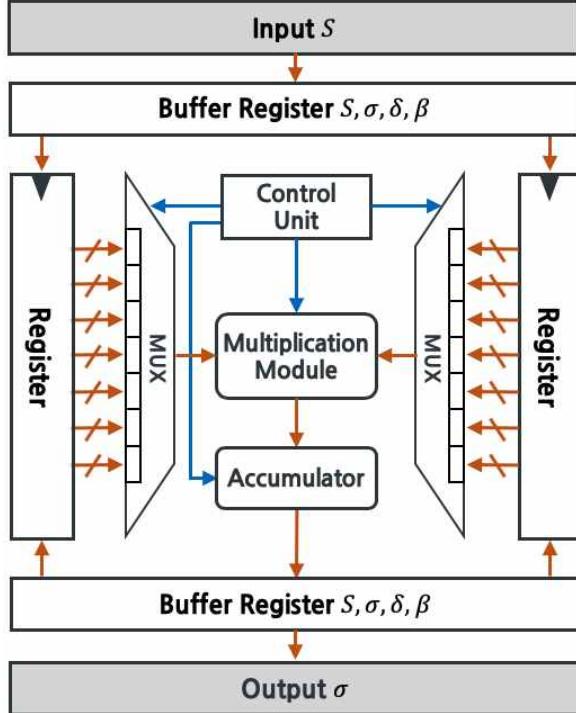


Figure 40. Inversionless BMA Hardware Architecture brings the high cost on complexity and area in term of hardware implementation, the inversionless approach [3] is considered. The architecture has been designed and implemented by 180-nm CMOS standard cell technology.

II. Inversionless Berlekamp–Massey Algorithm

Berlekamp–Massey algorithm is one of the most efficient algorithm in term of hardware implementation cost. However, some computation in this algorithm has an expensive cost that is multiplicative inverse. The new technique has been introduced to avoid the inversion within this Berlekamp–Massey Algorithm [3].

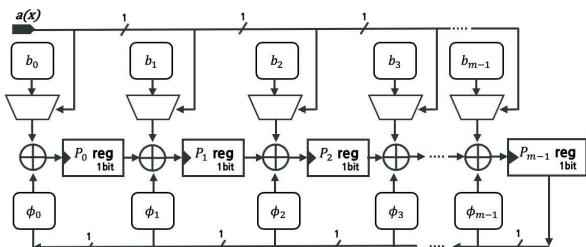


Figure 2. Multiplication over Finite Field Module

$$d_k = \sum_{i=0}^i \sigma_i^{(k)} S_{k-i} \quad (1)$$

$$\sigma^{(k+1)}(z) = \delta_k \sigma^{(k)}(z) - d_k \beta^{(k)}(z) \quad (2)$$

$$\beta^{(k+1)}(z) = \begin{cases} z\beta^{(k)}(z), & d_k = 0 \text{ or } k < 2l_k \\ z\sigma^{(k)}(z), & d_k \neq 0 \text{ and } k \geq 2l_k \end{cases} \quad (3)$$

$$l_{k+1} = \begin{cases} l_k, & d_k = 0 \text{ or } k < 2l_k \\ k - l_k + 1, & d_k \neq 0 \text{ and } k \geq 2l_k \end{cases} \quad (4)$$

$$\delta_{k+1} = \begin{cases} \delta_k, & d_k = 0 \text{ or } k < 2l_k \\ d_k, & d_k \neq 0 \text{ and } k \geq 2l_k \end{cases} \quad (5)$$

for $k = 0, 1, \dots, r-1$.

This algorithm calculate the error–locator polynomial $\sigma(z)$ with degree below or equal to $2t$. If the degree polynomial is larger than $2t$, then more than t -error occurred on the message and the key equation is not able to be solved.

III. Hardware Architecture

Figure 1 shows the hardware architecture of Berlekamp–Massey Algorithm. It consist of Control Unit, Multiplication Module, Accumulator, Multiplexer, and some registers/ buffer. Control Unit Module manages the operation state of computation such computing state of equation (1) and equation (2).

The main part of BMA hardware architecture is the multiplication module which calculates the multiplication over finite field with polynomial representation. The design uses modified Linear Feedback Shift Register (LFSR) which is well-known hardware implementation of finite field arithmetics. The figure 2 shows multiplication of

$$P(x) = a(x) * b(x)$$

$a(x)$ is processed serially and it is used to control modulo addition process of $b(x)$ value.

Table 1. Design Summary

BCH Parameter (n, k, t)	Gate Count of BMA Module	Clock Rate (MHz)
(15, 5, 3)	20584.59	200 MHz
(31, 11, 5)	31501.92	200 MHz
(63, 18, 10)	56611.92	200 MHz
(127, 120, 1)	21609.63	200 MHz
(127, 57, 11)	68394.88	200 MHz
(255, 179, 10)	71403.30	200 MHz

Both $a(x)$ and $b(x)$ value is selected by multiplexer corresponding with the its computation step. Accumulator module is used to accumulate the multiplication result in equation (1).

IV. Performance Evaluation

The evaluation is performed along with BCH Code module in order to analyze the effect of changing BCH Code parameters such length of message and number of maximum allowed bit error. Table I shows design summary of various BCH Code parameters utilized by 180-nm CMOS standard cell technology. It gives a fact that achieved clock rate and gate count are directly impacted with increasing number of maximum allowed bit error. BCH(127, 120, 1) parameter has a lowest hardware cost than BCH (63, 18, 10) parameter which has 10 maximum possible bit error. Since key equation solver is focusing on Syndrome value which is determined by number of maximum allowed bit error, it requires more register buffer to hold the variables (S, σ, δ, β). In other hand, the clock rate is likely stable in 200 MHz which is reliable enough to comply with current widely used clock in embedded system.

V. Conclusion

This paper present the hardware implementation of various parameters inversionless Berlekamp-Massey Algorithm which is used in BCH Decoder. The hardware cost (number of gates) and achieved clock rate are directly impacted with increasing number of maximum allowed bit error. However, the achieved hardware area is still high and need to be optimized compare to truncated architecture approach [4].

[References]

- [1] T. D. Mateer, "On the equivalence of the Berlekamp - Massey and the Euclidean algorithms for algebraic decoding", Proc. 12th Can. Workshop Inf. Theory, pp. 139–142, May 2011.
- [2] I. S. Reed, M. T. Shih, T. K. Truong, "VLSI design of inverse - free Berlekamp - Massey algorithm", Proc. Inst. Elect. Eng., vol. 138, pp. 295–298, Sept. 1991.
- [3] X. Youzhi, "Implementation of Berlekamp - Massey algorithm without inversion", Proc. IEE Commun. Speech Vis., vol. 138, no. 3, pp. 138–140, 1991.
- [4] J. I. Park, H. Lee, and S. Lee, "An area - efficient truncated inversionless Berlekamp - Massey architecture for Reed - Solomon decoders," in Proc. IEEE International Symposium on Circuits and Systems (ISCAS), May 2011, pp. 2693–2696.

클라우드 보안 취약점 분석에 관한 연구

손현민*, 박민규*, 최현택*, 이현철*

*대신정보통신(주)

linuxson@dsic.co.kr, passionpmk@dsic.co.kr, htchoi@dsic.co.kr, wlsqor2@gmail.com

A Study on the Cloud Security Vulnerability Analysis

Hyun-Min Son*, Min-Kyu Park*, Hyun-Taek Choi*, Hyun-Cheol Lee*

*DaiShin Information&Communications Co., Ltd

요약

최근 서버, 스토리지, 네트워크와 같은 인프라 자원 및 가상화와 분산처리 기술을 기반으로 언제 어디서나 효율적인 서비스를 이용할 수 있는 클라우드 컴퓨팅 기술이 관심을 받고 있다. 모든 정보가 중앙으로 집중되는 클라우드 컴퓨팅 환경에서 업체나 기관 등의 서비스 안정성과 보안은 클라우드 컴퓨팅 환경에서 핵심 기술이 될 것이다. 본 논문에서는 업체 및 기관의 보안 취약점 진단 데이터를 축적, 처리 및 분석하여 데이터센터 내부의 보안 위협 요소를 사전 예측하고 보안 취약점 중요도에 대한 객관적 데이터를 제공함으로써 관련 취약점 대응 시스템 구축에 필요한 객관적 근거인 취약 항목을 제시하여 보안 취약점 점검 대응의 효과를 높이는데 있다.

I. 서론

클라우드 컴퓨팅은 스토리지·플랫폼·소프트웨어와 같은 ICT 자원을 데이터센터에 공유 풀(Shared Pool)로 집적시켜 이용자가 필요로 하는 만큼 분리하여 네트워크를 통해 가상화 기술로 서비스를 제공한다. 컴퓨팅 자원을 공유하면서 생긴 새로운 해킹, 바이러스 등의 보안 위협에 의해 정보자산의 침해, 사이버 테러, 개인정보 및 중요 정보 자산의 무단 유출과 같은 역기능이 지속적으로 증가하여 심각한 사회문제로 대두되고 있다.

또한 악성 코드와 공격 기법의 고도화, 국가 간 사이버 보안 위협 증대, 침투 경로의 다양화 등 사이버 공격이 점점 지능화, 고도화되고 있고 악성 코드를 이용한 전자 금융 사기 피해가 증가하고 있으며 피싱, 패밍, 스미싱, 메모리 해킹 등 다양한 수법들이 꾸준히 증가하고 있다.

시스템이 해킹될 경우 그 결과는 다양한 현상으로 나타나며 해킹된 시스템은 악성코드를 유포하기 위한 경유지로 악용되기도 하고, 기업

의 개인정보를 유출하는 통로로 사용되기도 하며 때로는 다른 시스템을 공격하는 공격자로 사용 된다. 이 같은 문제의 해결을 위해 업체 및 기관들은 사업영역 별로 보유한 IT 인프라의 보안 취약점 진단과 대응을 위한 솔루션들을 도입, 운영해 오고 있다. 본 논문에서는 취약점 대응 시스템 구축에 필요한 취약 점검 항목을 제시하여 보안 취약점 대응 우선순위 선정의 근거를 확보하고자 한다.

II. 관련 연구

최근 업체 및 기관을 중심으로 보안 위협에 효과적으로 대응하기 위한 취약점 관리업무 프로세스의 재설계와 함께 취약점 진단 관리시스템 체계를 구축하고 있다.

KT에서는 보안 취약점 관리업무의 문제점을 제시하고 웹 기반의 취약점 진단 통합관리 체계 및 운영 프로세스를 구축하였다[1]. 한국항공대에서는 국내 소프트웨어 보안 취약점의 중요도를 정량적으로 산출 할 수 있는 보안 취약점 정량 평가 체계를 제안하였다[2], 한양대에서

는 TCP/IP 기반 DoS 공격의 보안 취약점 분석, 공격 예측 및 공격 발생 시 신속한 대응이 가능한 보안 정책을 제시하였다[3]. 순천향대에서는 국가기반 시설을 감시 및 제어하는 제어 시스템의 보안 정책 및 보안 취약점을 분석하였다[4].

또한 과학기술정보통신부, 한국인터넷진흥원에서는 기술적 취약점분석·평가 방법 상세 가이드를 발간하여 기술적 취약점 분석·평가 항목별 점검 방법을 제시하여 다양한 취약점 분석 평가를 수행 할 수 있도록 하였다[5].

III. 클라우드 보안 취약점 분석

3.1 보안 취약점 및 분석 도구

보안 취약점이란 소프트웨어나 정보시스템 상에 존재하는 보안상의 결점으로서 정보시스템에 불법적인 사용자의 접근을 허용할 수 있는 위협, 정보시스템의 정상적인 서비스를 방해하는 위협 및 정보시스템에서 관리하는 중요 데이터의 유출, 변조, 삭제에 대한 위협들을 말한다[6]. 일반적으로 해커들은 시스템의 보안 취약점을 알아낸 후, 보안 취약점을 공격하여 시스템 제어 권한을 획득하고 악성코드를 유포하거나 또 다른 시스템을 공격하는 공격 도구로 악용하기도 한다.

정보시스템에 보안 취약점이 존재하고 있는지에 대한 점검 작업을 수행한 후, 그 결과로써 정보시스템의 보안 수준을 분석하며 보안 취약점 점검 및 분석을 도와주는 자동화 도구를 보안 취약점 분석도구라 한다. 보안 취약점 분석 및 점검 도구에는 SAINT, MScan, SScan, Nessus, TIGER, COPS, K-COPS, Antisniff, Tripwire, TCP Wrapper, RTSD, nMAP, Port Sentry, SNort, Check Log 등이 있다.

3.2 클라우드 환경에서 통합 보안 취약점 점검

취약점 진단 수행에는 수동 진단과 자동 진단 방법이 있으며 자동 진단 방법에는 서버에서 지원하는 스크립트를 이용하여 취약점을 자동 진단한다. 본 논문에서는 자동 시스템 취약

점 점검 도구인 Secuguard SSE(Agent)를 사용하였고 보안 취약점 점검의 정보자산 유형별 진단 영역(진단 대수)을 Unix(78), Windows(82), WEB(28), WAS(12), DBMS(41), 네트워크(39), 보안 장비(26), 하이퍼바이저(43)로 구성하여 구성 영역별로 취약점 그룹의 점검 항목에 대한 보안 취약점을 점검하였다. 전체 시스템 구성도는 다음 [그림 1]과 같다.

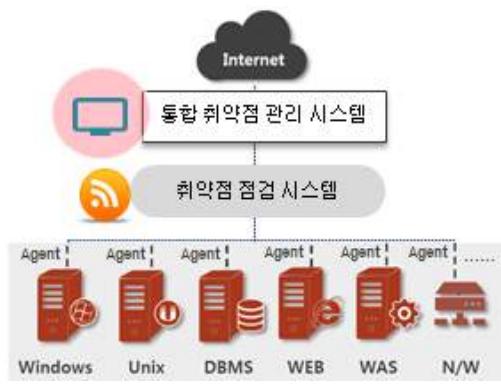


그림 1. 보안 취약점 점검 시스템

3.3 클라우드 보안 취약점 분석

취약점 그룹별 분석 결과는 다음 [표 1]과 같으며 평균 양호율은 93%로 보안 위협에 취약한 취약점 그룹은 DB관리, 보안패치, 접근제어, 설정, 권한관리, 파일 및 디렉토리 관리, DBMS 보안 설정, 서비스관리, 계정관리 등에서 나타났으며 취약점 그룹별 평균 이하의 대표 취약 항목(진단 영역)은 다음과 같다.

- DB 관리 : 패치관리 백신 프로그램 업데이트 (WIN), 패치관리 최신 HOT FIX 적용(WIN), 윈도우 인증 모드 사용(WIN)
- 보안 패치 : 최신 패치 적용(DBMS)
- 접근 제어 : 관리자 패스워드 관리(WAS), 관리자 콘솔 관리(WAS), 관리자 계정명 변경 (WAS)
- 권한 관리 : 타 사용자 권한 부여 옵션 사용 (DBMS)
- 설정 : 응답메시지 관리 및 테몬 관리(WEB)
- 파일 및 디렉토리 관리 : 파일 소유자 및 권한 설정(UNIX), World Writable 파일점검(UNIX)

[표 1] 취약점 그룹별 양호율(단위 : %)

취약점 그룹	2018년			2019년	평균
	1분기	2분기	3분기	1분기	
DB 관리	73.75	71.88	74.29	72.73	73.16
보안 패치	68.48	69.05	76.19	78.57	73.07
접근 제어	81.33	84.44	85.93	86.67	84.59
권한 관리	92.73	92.93	94.02	94.57	93.56
보안 감사 설정	99.09	100	98.55	95.65	98.32
설정	81.63	87.62	93.81	95.71	89.69
DBMS 보안 설정	93.29	94.11	96.62	96.81	95.21
서비스 관리	98.47	98.04	98.25	98.23	98.25
파일 및 디렉토리 관리	91.14	96.39	98.26	98.59	96.10
로그 관리	98.21	98.56	98.99	98.99	98.69
기능 관리	98.43	98.01	99.19	99.2	98.71
계정 관리	98.02	97.06	99.47	99.48	98.51
보안 관리	96.69	98.59	99.33	99.70	98.58
접근 관리	98.73	99.63	99.77	99.76	99.47
패치 관리	99.95	96.59	99.83	99.90	99.07

- DBMS 보안 설정 : 접속 IP 지정(DBMS), DB LINK 암호화 설정(DBMS), PL/SQL Package 사용(DBMS)
- 서비스 관리 : NetBIOS 바인딩 서비스 구동 점검(WIN), HTTP/FTP/SMTP 배너 차단 (WIN), NFS 서비스 비활성화(UNIX)

IV. 결론 및 향후 연구

모든 정보가 중앙으로 집중되는 클라우드 컴퓨팅에서 주요 시스템이 해킹 공격의 표적이 되고 그 기법이 고도화, 지능화되고 있다. 본 논문에서는 데이터센터 내부의 보안 위협 요소를 분석하고 보안 취약점 중요도에 대한 취약점 그룹 및 취약 항목의 객관적 데이터를 제시하였다.

향후 연구 방향으로는 시스템의 보안 취약점을 자동으로 점검하고 발견된 취약점 및 문제들에 대한 해결 방법을 자동으로 제공하는 지능화된 클라우드 보안 취약점 점검 및 분석 도구를 개발하는 것이다.

[참고문헌]

- [1] 문호건, 박성철, "기업보안 강화를 위한 취약점 진단 통합관리 체계 구축", 한국통신학회지 제31권 제5호, 2014.4, PP.39~45.
- [2] 안준선 외 2, "보안 취약점 중요도 정량 평가 체계 연구", 정보보호학회논문지 제25권 제4호, 2015.8, PP.921~932.
- [3] 조성현, 외2, "TCP/IP 네트워크 프로토콜의 DoS 공격 취약점 및 DoS 공격사례 분석", 정보보호학회지 제24권 제1호, 2014.2, PP. 45~52.
- [4] 최명균 외 2, "제어 시스템에 대한 보안정책 동향 및 보안 취약점 분석", 정보보호학회지 제21권 제5호, 2011.8, PP.55~64.
- [5] 한국인터넷진흥원, 과학기술정보통신부, "기술적취약점분석 · 평가방법상세가이드", 2017.12
- [6] <http://www.nilessoft.co.kr>

복합체 기반 폴딩 Sbox 구현

최병준*, 박종선*

*고려대학교 전자공학과

bjchoi@korea.ac.kr*, jongsun@korea.ac.kr*

Implementation of Composite Field Arithmetic based Folding Sbox

Byungjun Choi*, Jongsun Park*

*School of electrical engineering, Korea University.

요약

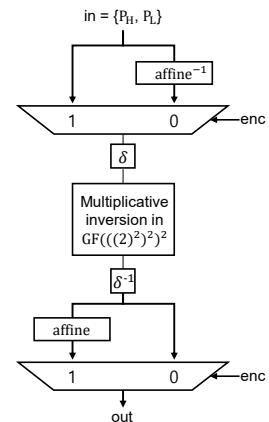
IoT(Internet of Things) 시대에 맞춰 경량 암호 알고리즘 및 저면적 HW 암호 알고리즘 구현 연구들이 진행되어 왔지만 Passive RFID(Radio Frequency IDentification)와 같은 면적 활용이 극히 제한적인 기기의 경우에는 여전히 암호 알고리즘을 넣기에 한계가 있는 상황이다. 이에 본 논문에서는 현재 국제 표준 암호 알고리즘인 AES(Advanced Encryption Standard)에서 면적 비중이 있는 Sbox를 복합체 기반에서 폴딩 구조로 설계하여 SAMSUNG 65nm 공정 기준으로 기준 저면적 복합체 기반 Sbox 대비 약 22% 감소한 337GE를 얻었으며 이를 토대로 저면적을 요구하는 기기에 활용될 수 있는 방안을 제시한다.

I. 서론

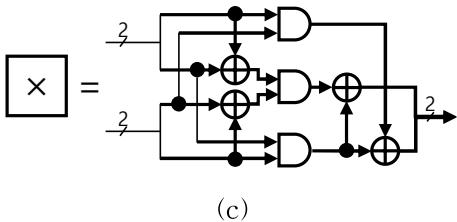
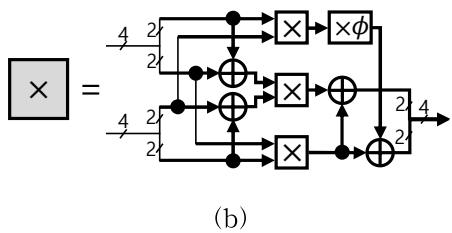
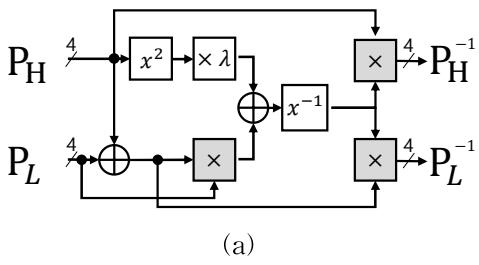
인간의 관여 없이 기기 간 정보를 교환할 수 있는 IoT(Internet of Things) 기기들은 그 특성상 개인 정보가 유출될 수 있는 보안상 취약점을 갖고 있다. 이에 저전력 및 저면적을 요구하는 IoT 기기에 맞춰 경량 암호들이 만들어졌지만 Passive RFID(Radio Frequency IDentification)와 같은 극히 제한적인 면적을 가진 기기에 경량 암호들을 넣기에는 한계가 있다. 이에 수행시간은 전보다 길어지지만 저전력과 저면적이 강점인 비트 시리얼 구조가 적용되어 이를 극복하려는 시도도 있다[1]. 본 논문에서는 현재 국제 표준 암호로 사용되고 있는 AES(Advanced Encryption Standard) 암호 알고리즘에서 면적 비중을 차지하는 Sbox를 비트 시리얼 구조에 적용 될 수 있도록 폴딩 기법을 제시하여 기존 복합체 기반 Sbox보다 약 22% 면적을 줄임을 보였다. 본 논문의 구성은 2장에서 기존 복합체 기반

Sbox를 살펴보고, 3장에서 기존보다 면적을 줄인 폴딩 구조 Sbox를 제시하며 4장에서는 기존 Sbox와 제안된 Sbox의 면적과 전력을 비교하는 실험 결과를 나타내었다. 끝으로 5장에서 결론을 맺는다.

II. 기존 복합체 기반 Sbox



[그림. 1] 복합체 기반 Sbox 구조



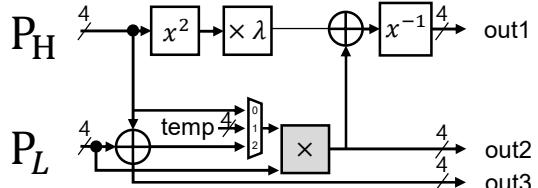
[그림. 2] (a) $GF((2^2)^2)^2$ 상의 곱 역원 회로. (b) $GF((2^2)^2)$ 상의 곱셈 회로. (c) $GF(2^2)$ 상의 곱셈 회로.

기존 AES의 Sbox HW 설계는 LUT(Look Up Table) 형식으로 설계되었으나 복합체를 사용하여 역원을 구할 때 차원을 분리하는 방법으로 암호화와 복호화에서 쓰이는 Sbox의 역연산을 공유함으로써 기존 대비 획기적인 면적 감소를 할 수 있게 되었다[2]. [그림. 1]은 이러한 복합체 기반 Sbox의 전체적인 구조를 나타내고 있다. 암호화 기준으로 보면 처음 상위 4-bit(P_H), 하위 4-bit(P_L)로 총 8-bit가 $GF(2^8)$ 상에서의 입력으로 받게 된다. 이를 하위 체로 분리하기 위해 Isomorphism function(δ)을 적용하고 $GF((2^2)^2)^2$ 상의 곱 역원 회로를 통하여 역원을 구하게 된다. 그 후 다시 원래의 체로 복구하기 위해 Inverse Isomorphism function(δ^{-1})을 거친 후 어파인이 적용되어 Sbox의 출력을 내놓게 된다.

복호화는 이의 반대 과정으로 진행된다.

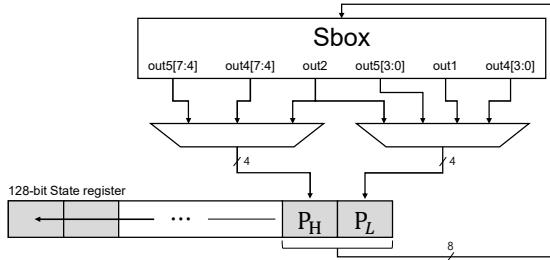
[그림. 2]의 (a)는 [그림. 1]에서 나타낸 $GF((2^2)^2)^2$ 상의 곱 역원 회로이다. 여기서 곱셈 모듈은 (b)처럼 $GF((2^2)^2)$ 상의 곱셈 회로로 나타낼 수 있고, 이는 다시 (c)와 같이 $GF(2^2)$ 상의 곱셈 회로로 만들어지는 구조로 되어 있다. 이는 $GF(2^8)$ 상에서의 곱셈 역원을 그대로 구하는 것보다 면적 측면에서 효율적인 계산 방법이며 복호화 시에도 그대로 적용될 수가 있다.

III. 제안된 복합체 기반 풀딩 Sbox

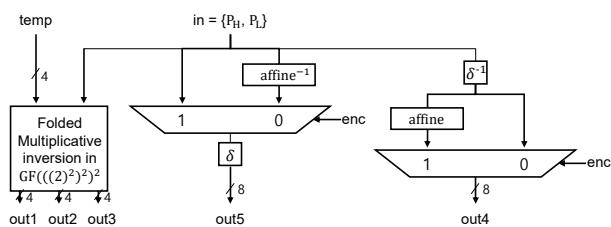


[그림. 3] 풀딩이 적용된 $GF((2^2)^2)^2$ 상의 곱 역원 회로

현재 극히 제한적인 면적을 요구하는 기기에 들어갈 수 있도록 국제 표준 암호 알고리즘인 AES를 비트 시리얼 구조로 적용한 사례가 있지만 그 AES를 구성하는 모듈 중에 Sbox에는 비트 시리얼 구조가 아직 적용되지 않았다[1]. 이에 3장에서는 Sbox를 비트 시리얼 구조에 적용할 수 있도록 풀딩 구조를 제안한다. [그림. 3]은 이러한 Sbox의 풀딩 구조에 기반이 되는 핵심 회로이다. 이는 [그림. 2]의 (a)에서 $GF((2^2)^2)$ 상의 곱셈 회로가 여러 번 사용 된 것을 하나로 풀딩함으로써 가능하다. 풀딩을 적용하게 되면 중간값을 저장하는 레지스터가 필요하게 되는데 [그림. 4]와 같이 AES의 128-bit State 값 중 8-bit를 그대로 사용한다. [그림. 5]는 이를 반영한 전체 복합체 기반 풀딩 Sbox 구조이다. 다음은 Sbox의 입력과 출력 매커니즘을 설명한다. 먼저 8-bit 입력을 받고, out5를



[그림. 4] 비트 시리얼 구조 안 풀딩 Sbox와 State 128-bit 레지스터



[그림. 5] 복합체 기반 풀딩 Sbox 구조

P_H 와 P_L 에 반영한다. 그 후에 out3를 여분의 4-bit 레지스터에 저장한다. 이때, 여분의 레지스터는 설계자에 따라 존재 위치가 다를 수 있으나 보통 Key Expansion 모듈에서 Round Key를 만들기 위한 임시 레지스터 공간을 이용한다. 이어서 [그림. 3]에서 멀스 선택 신호로 2를 선택한 뒤 out1을 P_L 에 저장한다. 그 다음 멀스 선택 신호로 0을 택한 뒤 out2를 P_H 에 저장하여 상위 4-bit의 결과를 만든다. 이제 곱 역원 회로의 하위 4-bit를 만들기 위해 멀스 선택 신호로 1을 선택하여 임시 레지스터에 저장했던 값과 P_L 값을 $GF((2^2)^2)$ 상의 곱셈 회로에 통과시켜 다시 P_L 에 적용한다. 끝으로, out4를 P_H 와 P_L 에 적용시켜 최종 Sbox의 결과를 State bit에 적용한다.

IV. 실험결과

[표. 1]은 기존 복합체 기반 Sbox와 제안한 복합체 기반 풀딩 Sbox의 면적 및 전력을 SAMSUNG 65nm 공정 기준으로 비교한 것이다. 면적에서 보면 현재까지 가장 작은 면적을 갖고 있는 satoh의 Sbox와 비교해보았을 때 약 22%의 면적 감소가 되었음을 알 수 있다. 이

[표. 1] 기존 Sbox와 면적, 전력 비교

SAMSUNG 65nm	Area(GE)	Power(mW)
canright[3]	611	0.3039
satoh[2]	432	0.2414
Ours	337	0.1495

면적은 [그림. 4]에서 약 19GE를 차지하는 멀스 구조를 제외한 것이다. 제안된 회로는 cycle 측면에서 보면 1byte의 Subbyte를 처리하기 위해 3장에서 보았듯이 5 cycle 추가되는 구조로 되어있다. cycle 대비 면적 감소를 요구하는 비트 시리얼 구조에서만 적합한 구조이며 총 수행 시간이 길어지기 때문에 전력이 낮더라도 에너지 측면의 이득은 없으므로 이를 고려한 기기에 사용되어야 할 것이다. 또한, 더욱 제한된 면적이 요구되는 기기를 사용할 시 [그림. 2] (b)의 $GF(2^2)$ 상의 곱셈 회로를 풀딩하면 cycle 수는 늘어나지만 면적 감소가 좀 더 될 것으로 사료된다.

V. 결론

본 논문에서는 비트 시리얼 구조에 맞는 AES Sbox의 풀딩 구조를 제안하여 기존 Sbox 대비 22%의 면적 감소를 이루었으며 면적 제한이 있는 기기에 탑재 될 수 있는 방안으로 기여될 수 있을 것이라 판단된다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터 (UD170109ED)를 통한 방위사업청과 국방과학 연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- [1] Jean, J., Moradi, A., Peyrin, T. and Sasdrich, P., 2017, September. Bit-Sliding: A Generic Technique for Bit-Serial Implementations of SPN-based Primitives. In International Conference on Cryptographic Hardware and Embedded Systems (pp. 687-707). Springer, Cham.
- [2] Satoh, A., Morioka, S., Takano, K. and Munetoh, S., 2001, December. A compact Rijndael hardware architecture with S-box optimization. In International Conference on the Theory and Application of Cryptology and Information Security (pp. 239-254). Springer, Berlin, Heidelberg.
- [3] Canright, D., 2005, August. A very compact S-box for AES. In International Workshop on Cryptographic Hardware and Embedded Systems (pp. 441-455). Springer, Berlin, Heidelberg.

비트 시리얼 구조를 적용한 ARIA 암호화 하드웨어

조정훈, 김보훈, 박종선*

*고려대학교 전기전자공학과

twister333@korea.ac.kr, rlaqhgns0616@korea.ac.kr, jongsun@korea.ac.kr

ARIA Encryption Hardware Applying Bit-serial Architecture

Junghoon Cho, Bohun Kim, Jongsun Park*

*School of Electrical Engineering, Korea University.

요약

본 논문에서는 ARIA 하드웨어에 비트 시리얼 기술을 적용한 하드웨어를 제안한다. ARIA 암호화 라운드 함수는 AddRoundKey, 치환 계층, 확산 계층의 각 과정이 128비트의 datapath로 진행되는데, 저면적 설계를 위해 단일 비트 단위로 연산을 수행하는 비트 시리얼 기술을 라운드 함수의 일부에 적용하였다. TSMC 250nm CMOS 공정에서 설계 및 합성을 진행한 결과, 기존에 설계된 저면적 ARIA 하드웨어와 비교했을 때 42%의 면적 효율을 얻을 수 있었다.

I. 서론

오늘날의 정보화 사회에서는 사물인터넷이나 스마트폰 인터넷 등의 다양한 플랫폼을 통해 무선 네트워크의 보급이 확장되어가고 있고, 자연히 이에 대한 정보 보안의 중요성 역시 커지며 암호 알고리즘은 비밀키 및 공개키 암호, 블록 체인, 해쉬 함수 등 다양한 방향으로 발전을 이루어 왔다.

우리나라에서는 국제 표준 블록 암호인 AES에 대응되는 알고리즘을 개발하고자 국가 보안기술연구소(NSRI)의 주도 하에 블록 암호 알고리즘 ARIA를 2003년에 개발하였고, 이후 2004년에 국가표준(KS)으로 지정하여 다양한 분야에서 널리 쓰이고 있다.

통신 기술과 암호 알고리즘의 발전과 함께 대두된 요소에는 소형화되는 통신기기에 적용할 수 있는 저면적/저전력 하드웨어의 구현이다. ARIA의 경우 라운드 함수에 AES와 같이 128비트의 평문을 동일 비트 수의 라운드

키와 XOR 연산을 진행하는 AddRoundKey 부분이 존재하는데, 본 논문에서는 저면적 하드웨어 구현을 위해 이 부분에 단일 비트 단위로 연산을 수행하는 비트 시리얼 기술을 적용한 하드웨어를 설계하였다.

본 논문의 구성은 다음과 같다. II에서는 기존 ARIA 알고리즘의 구조와 비트 시리얼 기술을 적용한 하드웨어의 구조에 대해 설명한다. III에서는 설계한 하드웨어의 합성 결과를 기준 하드웨어와 비교하고, IV에서 결론을 서술한다.

II. 비트 시리얼 기반 저면적 하드웨어 설계

2.1 ARIA 블록 암호

ARIA는 AES와 유사한 형태의 라운드 함수를 반복적으로 적용하여 암호문을 만들어낸다. 반복 횟수는 라운드 키 생성에 사용되는 암호화 키의 비트 수에 따라 결정되는데,

128/192/256비트의 암호화 키의 경우 각각 12/14/16라운드를 진행한다. 마지막 라운드의 경우 확산 계층 대신 AddRoundKey 연산을 한번 더 적용하기 때문에 필요한 라운드 키의 수는 각각 13/15/17개이다.

ARIA의 라운드 함수는 128비트의 입력과 동일한 비트 수의 라운드 키에 대해 XOR 연산을 수행하는 AddRoundKey, S-box를 이용하여 8비트 단위로 치환을 수행하는 치환 계층, 마지막으로 16X16 크기의 이진 involutory 행렬(자기 자신이 역행렬인 행렬)과 8비트 단위로 곱셈 연산을 수행하는 확산 계층으로 이루어져 있다. 또한 AES의 경우 암호화와 복호화 하드웨어의 라운드 함수 구조가 동일하므로 효율적인 하드웨어 설계가 가능하다. [2]

2.2 저면적 비트 시리얼 구조

비트 시리얼 구조는 암호 하드웨어의 저면적 설계를 위해 큰 비트 단위로 이루어지는 연산을 단일 비트 단위의 연산으로 진행하는 하드웨어 형태로, 경량 블록 암호 하드웨어인 AES, PRESENT, SKINNY에 대해 적용하는 형태로 처음 제시되었다. [3]

본 논문에서 다루는 ARIA와 가장 유사한 형태의 AES의 경우, 128비트 단위로 이루어지는 AddRoundKey 연산을 비롯해서 라운드 함수의 모든 과정을 단일 비트 단위 연산으로 진행하도록 하드웨어를 설계하였다. 그 결과 라운드 함수 연산이 한 번 진행되는 데 걸리는 시간인 latency가 크게 증가하는 단점이 있었지만, 하드웨어 면적에 대해서는 큰 이득을 얻었다.

본 논문에서 설계한 ARIA 하드웨어의 경우 AddRoundKey 부분은 AES와 똑같이 단일 비트 단위로 수행하도록 설계하였고, AES와 함수 및 구조 면에서 차이가 있는 치환 계층과 확산 계층의 경우 각각 8비트와 128비트 단위로 연산을 수행하도록 설계하였다.

설계한 하드웨어의 구체적인 구조는 다음과 같다. 128비트의 입력과 암호화 키를 각각 128개의 1비트 버퍼에 나누어 저장한다. 그리고 128 cycle 동안 1비트씩 입력과 암호화 키에 대

해 XOR 연산을 수행한다. 수행 결과는 이후의 치환 계층 연산을 위해 8비트 단위로 결합해야 할 필요가 있는데, 이를 위해 XOR 연산의 결과를 8비트 버퍼에 저장한 후 8 cycle마다 출력으로 내보낸다. 8비트 단위로 수행되는 치환 계층 연산의 경우 총 4종류의 S-box가 사용되는데, 128비트 입력을 8비트 단위로 나누었을 때 32비트 간격으로 같은 S-box가 적용된다. 따라서 치환 계층 모듈 내에서 입력에 따른 S-box shift 횟수를 줄이기 위해 AddRoundKey에서부터 입력 비트의 순서를 조정하는 것을 고려하였다. 마지막 확산 계층의 경우 8비트 혹은 1비트 단위로 연산을 수행할 때 임시 결과를 저장하는 하드웨어 공간이 과도하게 발생할 것으로 예상되어 기존 하드웨어와 같이 128비트 단위로 설계하였다.

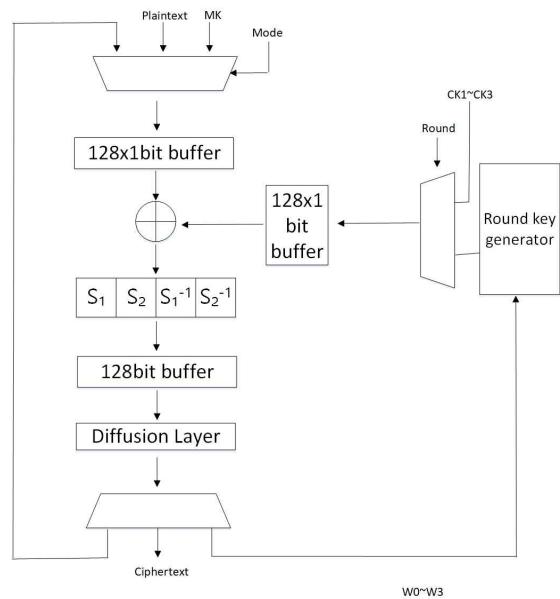


그림 1. 전체 하드웨어 구조도

III. 실험 결과

본 논문에서 제안한 ARIA 암호화 모듈 하드웨어는 Verilog HDL로 설계되었으며, TSMC 250nm 공정에서 합성되었다. 합성 결과 8029GE의 면적이 측정되었으며, 표 1에서 제시된 기준에 설계된 ARIA 하드웨어와의 비교를 통해 제안된 하드웨어가 가장 적은 면적을 가지는 것을 알 수 있다. 특히 32비트의 datapath

로 설계된 저면적 하드웨어에 비해서 42%의 면적 효율을 얻었으며, 이를 통해 비트 시리얼 기술을 토대로 한 하드웨어 구성이 datapath를 줄이는 것에 의해 면적 감소에 효과적이었다는 결론을 내릴 수 있다. 다만 면적이 감소하면서 latency가 증가하는 trade-off가 존재하는데, 라운드 함수에서의 AddRoundKey에 128cycle이 소모되기 때문에 12번의 라운드 함수가 전부 동작하는 데에는 총 1700cycle이 요구된다.

	This paper	[4]	[5]
Key length	128	128, 192, 256	128
Datapath	128	128	32
Area[GE]	8029	46100	13893
Technology [um]	250	130	350
Latency	1700	50/60/70	356

표 1. 기존 ARIA 하드웨어 디자인과의 비교

IV. 결론

본 논문에서는 ARIA 암호화 하드웨어의 면적을 줄이기 위해 기존에 다른 블록 암호 하드웨어에 먼저 제안되었던 비트 시리얼 구조를 ARIA 하드웨어의 AddRoundKey 부분에 중점적으로 적용해보았고, 그 결과 기존에 설계된 저전력 ARIA 하드웨어와 비교했을 때 42%의 면적 효율을 얻을 수 있었다. 이번에 적용된 비트 시리얼 저면적화 기술을 활용하면 ARIA 암호화 기술을 요구하는 소형 통신기기 등에 유용하게 쓰일 수 있을 것으로 예상되고 있다. 이번에 제안된 하드웨어 디자인의 범용성을 높이기 위해서는 복호화 기능을 추가하고, 192/256 비트의 암호화 키에 대해서도 대응이 가능하도록 보완이 이루어질 필요가 있다. 또한 기존 AddRoundKey에서는 면적 감소가 있었으나 control logic이 추가되면서 생긴 면적 증가가 있었기 때문에, 이를 완화하기 위한 연구가 추가적으로 요구된다.

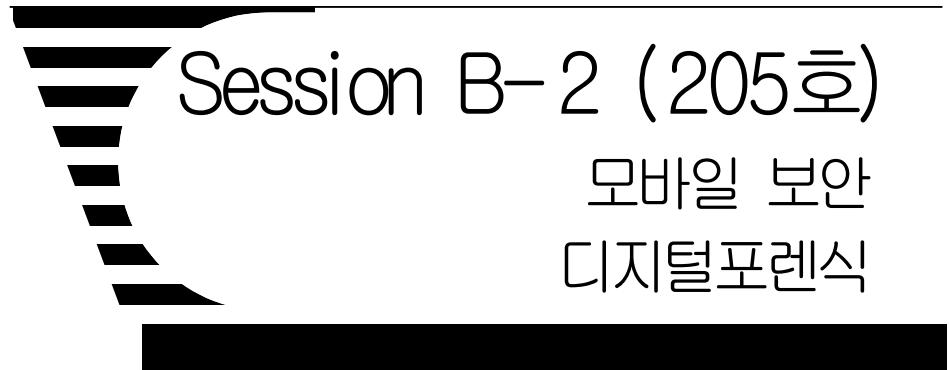
Acknowledgement

본 연구는 고려대 암호기술 특화연구센터

(UD170109ED)를 통한 방위사업청과 국방과학 연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- [1] 원동호, 현대 암호학, 도서출판 그린, 2017
- [2] Daesung Kwon, Jaesung Kim, Sangwoo Park et al., “New block cipher: ARIA”. In Proc. Information Security and Cryptology (ICISC’03), Seoul, Korea, LNCS 2971, Springer-Verlag, November 27 - 28, 2003, pp.432 - 445.
- [3] J.Jean, A. Moradi, T. Peyrin and P. Sasdrich, Bit-Sliding: A Generic Technique for Bit-Serial Implementations of SPN-based Primitives, Conference on Cryptographic Hardware and Embedded Systems 2017, September, 2017.
- [4] D.H. Kim and K.W. Shin, “An Efficient Hardware Implementation of ARIA Block Cipher Algorithm Supporting Four Modes of Operation and Three Master Key Lengths,” Journal of The Korea Institute of Information and Communication Engineering, vol. 16, no. 11, pp. 2517- 2524, Nov. 2012.
- [5] J. Park et al., “Low Power Compact Design of ARIA Block Cipher,” Proceedings of International Symposium on Circuits and Systems, pp. 313-316, May 2006.



좌장 : 이만희 (한남대)

역전파 신경망 기반의 안드로이드 악성코드 탐지 시스템

아피파틀 무카로*, 김호원*

*부산대학교 전기전자컴퓨터공학과
{afifatul.mukaroh, howonkim}@gmail.com

Backpropagation Neural Network based Malware Detection System for Android Application

Afifatul Mukaroh*, Howon Kim*

*Department of Electrical and Computer Engineering,
Pusan National University.

Abstract

Android has been a common operating system that currently people mostly use for their smartphone. Every years android users always increase because of its convenience for finding or installing any application based on what users need. Unfortunately, malware can also be found in android application. Therefore, malware detection system is needed as a part of mobile security. In this paper, it is developed malware detection system for android apps using Backpropagation Neural Network (BPNN) method. It performs that the model has accuracy 86.33%, precision 87.59%, and recall 84.67%.

Keywords : Mobile Security, Neural Network, Malware Detection

I. INTRODUCTION

Mobile phone is one of primary needs for every people. As more as people use mobile phone, as more as android user increase. Android is one of mobile operating system that currently being used in every smartphone in this world.

In android, It can not be denied that malware may be also found in the apps. Malware is a program that has malicious intent, like viruses, trojans, and worms [1]. For security, malware should be detected.

When an app is running in android, permission that being requested by the app can be read. Based on those permission, the behavior of malware can be also identified [2].

Artificial neural network (ANN) is one of best methods to read pattern or identification. This is because the ANN has ability to solve discontinuous and non-linear classification problems with robustness, adaptability, and high accuracy [3]. One of types in ANN is BPNN. BPNN learning algorithm is the best algorithm among the Multi-layer perceptron algorithms [4].

Therefore, in this paper it uses BPNN to detect malware in android apps. The result shows that the model of neural network

has accuracy 86.33%, precision 87.59%, and recall 84.67%.

II. Basic Theory

Malware detection system in this research is developed based on permission that android apps requests. The pattern of its permission is identified by BPNN in order to detect malware.

2.1. Android Application Permission

Android app permission govern what an app that has been installed in phone is allowed to do and access [5]. It can be about accessing data storage, like contacts or media files or accessing pieces of hardware like handset's camera or microphone.

The purpose of a permission is to protect the privacy of an Android user. Once an android app is installed, it should request permission to access sensitive user data [6].

Permission is considered as normal if it doesn't pose much risk to the user's privacy or the device's operation. If the app lists normal permissions in its manifest, it will be automatically granted. If the app lists not normal permissions in its manifest, the user must explicitly agree to grant those permissions.

2.2. Backpropagation

Backpropagation is one of methods in Artificial Neural Networks. It has supervised training which uses gradient descent to reach the minimum error [7]. The strength of BPNN is in the way it updates the weight. BPNN updates the weight by calculate the gradient of loss value. Loss value that being used is Mean Square Error (MSE).

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (f_i - y_i)^2 \quad \dots \dots \dots (1)$$

MSE is the mean of total quadrat of target () minus output () .

III. Discussion

In this part is discussed about dataset, BPNN model, training, and testing.

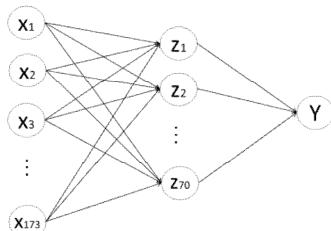
3.1. Dataset

The dataset that being used in this research is from Mahindru [2]. In the dataset there are many android application package, but here we only focus on application with communication category. It consist of 617 normal android application packages and 403 malware android packages. Each package has 173 true or false condition whether it requests a certain permission or not when it's running.

In practice, the data that being used for neural network training is 721, while for neural network testing is 300 (150 data is normal and 150 data is malware).

3.2. BPNN Model

The architecture of neural network that being used in this research is 173 – 70 – 1, which means the input layer consists of 173 nodes, hidden layer consists of 70 nodes, and output layer consist of 1 node. Figure 2 shows the details architecture of it.



[Figure 2] BPNN Architecture of the System

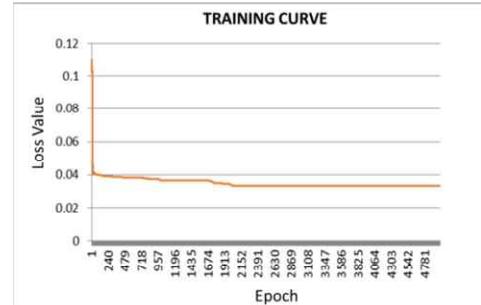
The input fhere is taken from 173 true or false condition whether an app requests a certain permission. If true, it will 1 and if false it will be 0.

The output for this neural network model is one node with 0 or 1 value. Output 1 means malware detected, output 0 means normal.

1.1. Training

Training in this research used learning

rate 0.1. It stopped when epoch reached 5000 and loss value 0.03. The training curve for this training can be seen in Figure 3. Based on this training curve, the model is actually reach minimal loss 0.03 at 2000th epoch. After that the loss value tends to be stagnant.



[Figure 3] Training Curve Based on Loss Value

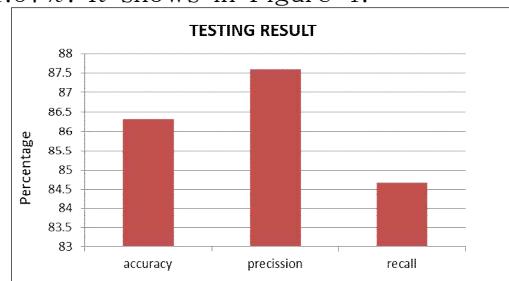
1.2. Testing

From 300 dataset that being tested with our neural network model, the true positive is 127, the true negative is 132, the false positive is 23, and the false negative is 18. Details are shown in Table 1.

[Table 1] Confusion Matrix

Class	Detected Malware	Detected Normal
Malware	127	23
Normal	18	132

From the confusion matrix, it can obtained that the accuracy of the model is 86.33%, the precision is 87.59%, and recall is 84.67%. It shows in Figure 4.



[Figure 4] Testing Result

IV. Conclusion

Research regarding malware detection on android's application is discussed in this paper. It used BPNN to read the pattern of how an android app request permission when it is running, then identify whether it has malware or it is normal. The research result shows that with BPNN, malware can be detected with accuracy 86.33%, precision 87.59%, and recall 84.67%.

References

- [1] Maughan, Douglas, 2007. Malware Detection. Germany : Springer Science & Business Media.

- [2] Mahindru, Arvind (2018): *Data set of Android permissions*. figshare. Fileset.
- [3] Jiang, J., Zhang, J., Yang, G., Zhang, D., and Zhang, L. 2010. “Application of Back Propagation Neural Network in the Classification of High Resolution Remote Sensing Image: Take Remote Sensing Image of Beijing for Instance.” In Proceedings of 18th International Conference on Geoinformatics, IEEE, 1-6.
- [4] Alsmadi, M., Omar, K., and Noah, S. 2009. “Back Propagation Algorithm: The Best Algorithm among the Multi-layer Perceptron Algorithm,” IJCSNS International Journal of Computer Science and Network Security 9.
- [5] Triggs, Robert. 2018. What Android app permissions mean and how to use them. [online] <https://www.androidauthority.com/app-permissions-886758/>.
- [6] Google Developers Team. 2019. Permissions overview. [online] <https://developer.android.com/guide/topics/permissions/overview>.
- [7] Nielsen, M. 2018. *How the backpropagation algorithm works*. [online] <http://neuralnetworksanddeeplearning.com/chap2.html>.

전자 영수증 어플리케이션의 보안 취약점 분석

지우중, 김형식

성균관대학교 전자전기컴퓨터공학과

woojoong@skku.edu, hyoung@skku.edu

Security vulnerability analysis of digital receipt application

Woojoong Ji, Hyoungshick Kim

Department of Computer Science and Engineering,
Sungkyunkwan University.

요약

전자 영수증 어플리케이션은 기존의 종이 영수증의 문제점을 대체하고자 전자형태로 스마트폰 또는 이메일로 발급 받는 영수증을 관리할 수 있는 어플리케이션을 말한다. 결제와 동시에 전자 영수증이 자동 발급됨으로써 종이 낭비 방지, 보관, 관리, 회계 처리에 매우 간편하기 때문에 많이 사용하고 있는 추세이다. 사용자들이 소비하는 모든 영수증이 해당 어플리케이션에 존재하기 때문에 사용자의 구매 내역, 개인정보, 이동 경로까지 저장하고 있다. 때문에 해당 전자 영수증을 안전하게 저장 및 관리되어야 한다. 본 논문에서는 전자 영수증 어플리케이션에 대한 보안 위협 가능성을 분석하고, 실현 가능한 다양한 공격 시나리오를 제시하였다. 제시한 공격의 가능성을 검증하기 위하여, 실제 사용 중인 전자 영수증 어플리케이션의 네트워크 트래픽을 분석한 결과, 다른 사용자의 영수증을 탈취할 수 있었다. 이를 이용하여 해당 사용자의 구매 내역뿐만 아니라 이동 경로, 개인정보 또한 획득할 수 있음을 확인하였다.

I. 서론

현재 과학기술정통신부에서 조사한 바에 의하면²⁾ 하루에 발급되는 종이영수증 및 신용카드 명세서는 하루 평균 약 4,000만 건 이상씩 발급된다. 이러한 종이 영수증의 가장 큰 문제점은 종이 영수증을 만드는 비용과 버려지는 종이 영수증을 처리하는데 발생하는 비용, 종이 영수증에서 발생하는 환경 오염 문제이다. 대부분의 사람들은 계산기에서 출력되는 영수증을 그냥 버리거나 아예 종이 영수증을 받지 않는다. 이렇게 무심코 버리는 종이 영수증에 의해 국내에서는 1년동안 종이 영수증을 만들기 위해 연간 33만 그루 이상 베어지며 종이 영수증을 만들기 위해 소비되는 물의 양은 약 15억 리터 이상 쓰인다. 또한 영수증을 만들고 폐기하는 과정에서는 발생되는 온실가스는 5만톤 이상이 발생하는 문제점이 존재한다. 이렇듯 종이 영수증은 자원낭비, 환경오염, 유해물질 발생, 개인정보 유출 등 많은 문제점이 존재한다. 이러한 문제점들로 인해 종이 영수증과 동일한 법적 효력을 가지는 전자 영수증으로 대체되고 있는 추세이다. 본 논문에서 소개하는 전자 영수증 [그림1]이란, 기존의 종이 영수증을 전자형태로 스마트폰 또는 이메일로 발급 받는 영수증을 말한다.



[그림 1] A사의 전자 영수증 어플리케이션을 통해 발급 받은 전자 영수증

전자 영수증 어플리케이션을 사용하는 사용자는 일일이 종이 영수증을 챙길 필요 없이 스마트폰으로 바로 해당 영수증을 받을 수 있으며 구매 내역을 통해 사용처, 날짜, 카드별 등으로 한눈에 확인할 수 있으며 손쉽게 관리 할 수 있어 많은 사용자들이 많이 사용하고 있는 추세이다.

²⁾ <https://news.joins.com/article/22650466>

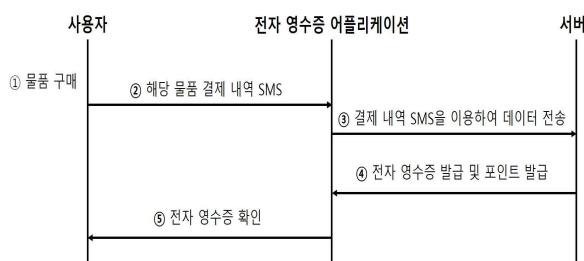
그러나 이러한 전자 영수증 어플리케이션이 보안에 취약하거나 악의적인 공격자에 의해 해당 전자 영수증 어플리케이션이 해킹을 당하게 된다면 종이 영수증과는 달리 전자 영수증 어플리케이션 특성상 사용자의 모든 구매 내역뿐만 아니라 개인 정보 유출, 사용자의 이동 경로 까지 노출될 수 있기 때문에 사용자의 편리성보다 보안적인 측면에 더욱더 신경을 써서 개발 하여야한다.

본 논문에서는 전자 영수증 어플리케이션에 대한 다양한 보안 위협 가능성을 분석하고, 실현 가능한 다양한 공격 시나리오를 제시하였다. 그래서 본 논문에서는 이러한 보안적인 문제점을 해결하기 위해 실제 사용 중인 전자 영수증 어플리케이션의 네트워크 트래픽을 분석한 결과, 다른 사용자의 전자 영수증을 탈취할 수 있었으며 사용자의 이동 경로 또한 알 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 전자 영수증 어플리케이션의 동작방식을 설명하고 3장에서는 전자 영수증 어플리케이션의 취약점을 이용한 공격 시나리오를 설명한다. 4장에서는 공격 결과를 설명하고 5장에서는 취약점개선방안, 마지막으로 6장에서는 결론을 제시한다.

II. 전자 영수증 어플리케이션

본 논문에서는 현재 존재하는 전자 영수증 어플리케이션 중 국내 최초 전자 영수증 발급 시스템을 개발한 A사에 대하여 분석하였다. A사의 전자 영수증 어플리케이션의 동작 방식은 [그림 2]와 같다.



[그림 2] A사의 전자 영수증 어플리케이션의 동작 방식

먼저 A사의 전자 영수증 어플리케이션을 설치한 사용자는 어떠한 물품을 구매하면 ①과 같이 이를 확인 할 수 있는 SMS 문자가 사용자의 스마트폰에 전송된다. 그런 다음 ②에서는 전자 영수증 어플리케이션은 해당 SMS 문자에 접근하여 이를 확인하여 거래 일시, 가격, 사업자 번호, 주소 등의 정보를 ③처럼 해당 어플리케이션 서버로 전송한다. 서버에서는 이러한 정보를 확인하여 ④처럼 전자 영수증을 발급하고 포인트를 발급한다. 하지만 발급된 전자 영수증을 확인하는 단계인 ⑤에서 다른 사용자가 사용한 전자 영수증을 탈취 할 수 있는 취약점이 발견되었다. 자세한 내용은 3장 공격 시나리오에서 자세히 설명하겠다.

III. 공격 시나리오

3.1 공격 시나리오

A사의 전자 영수증 어플리케이션의 경우 ⑤번째 과정에서 사용자가 사용한 전자 영수증을 확인 할 수 있다. [그림 3]은 ⑤번째 과정에서 사용자가 해당 전자 영수증 어플리케이션에서 영수증을 조회할 때 발생하는 패킷을 네트워크 분석 툴인 Wireshark[1]로 캡처한 모습이다.

```

GET /receipt/list.jsp?combine=ssce
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.98 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
Accept-Encoding: gzip, deflate
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
Host: 192.168.1.101 사용자의 휴대폰번호 5.k:802
Connection: keep-alive
Upgrade-Insecure-Requests: 1
  
```

[그림 3] A 어플리케이션에서 영수증을 조회할 때 발생하는 패킷 내용

[그림 3]처럼 A사의 전자 영수증 어플리케이션을 통해 사용자가 발급받은 전자 영수증을 조회할 때 보안에 취약한 HTTP 프로토콜[2]을 사용하며 URL 메타 데이터에 대한 암호화가 전혀 적용되어 있지 않다는 것을 확인 할 수 있다. 이는 악의적인 공격자가 해당 패킷을 의도적으로 발생시켜 분석한다면 충분히 공격에 대한 시나리오를 계획할 수 있다. [그림 3]처럼 A사는 사용자의 휴대폰 번호로 구분하여 영수증을 조회한다. 이는 악의적인 공격자가 해당 휴대폰 번호만 바꿔서 A사의 어플리케이션 서버에 요청하게 된다면 해당 전자 영수증 어플리케이션을 사용하는 사용자의 모든 구매내역[그림 4]에 접근 할 수 있다.

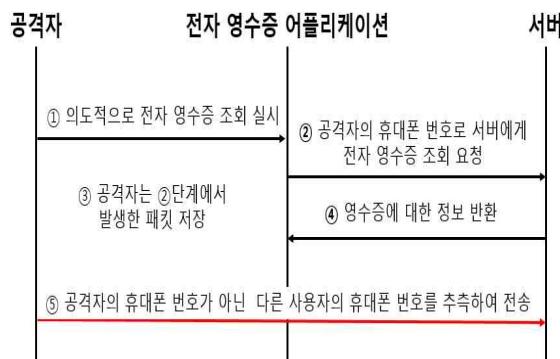
< 전자영수증 >

2019년 01월			208,990원	
			전체카드사	편집
01.23	1심 풍물스토리주점마켓	KB 국민카드	3,600원	
01.23	씨드모리스 청교관단지입구점	KB 국민카드	5,500원	
01.22	주식회사 토토리 토토리	KB 국민카드	3,600원	

[그림 4] 사용자의 모든 구매 내역에 대한 전자 영수증

3.2 재전송 공격과 추측공격

A사의 전자 영수증 어플리케이션을 공격하기 위해서는 재전송 공격과 추측공격을 이용한다. 악의적인 공격자는 [그림 3]처럼 일반 사용자로 위장하여 해당 A사의 전자 영수증 어플리케이션을 사용한다. 그런 다음 전자 영수증 조회를 의도적으로 실시한다. 그러면 [그림 3]과 같은 패킷이 발생하는데 이때 악의적인 공격자는 해당 패킷의 내용을 복사한 후 재전송 공격에 이용할 정보를 얻는다. 본 논문에서 소개하는 재전송 공격(Replay Attack)[3]이란, 사용자의 부주의 또한 중간자 공격(Man-in-the-middle)[4]으로 인해 유출된 암호나 토큰, 쿠키 등을 재사용하거나 유효한 데이터를 중간에 몰래 도청하여 얻는 정보로 재사용 또는 재전송하는 공격을 말하며 추측 공격(Guessing Attacks)[5]이란, 악의적인 공격자가 접근하고자 하는 URL 메타 데이터를 추측하는 공격 방법이다. 일반적으로는 URL 메타 데이터에 존재하는 데이터는 임의의 문자열이나 식별할 수 없는 문자열로 되어 있어 이러한 추측 공격에 대해선 안전하다. 하지만 A사의 전자 영수증 어플리케이션에서는 [그림 3]처럼 사용자의 휴대폰 번호로 사용자를 식별하기 때문에 악의적인 공격자는 다른 사용자의 휴대폰 번호를 추측하여 앞서 설명한 재전송 공격과 추측 공격을 함께 이용하여 다른 사용자의 전자 영수증을 탈취한다. [그림 5]는 본 논문에서 소개하는 공격시나리오 도식화한 것이다.



[그림 5] A사의 전자 영수증 어플리케이션 공격시나리오

IV. 공격 결과

위에서 설명한 공격시나리오를 바탕으로 상용 A사의 전자 영수증 어플리케이션에 대해 실제 공격에 대한 결과를 설명한다.

4.1 다른 사용자의 전자 영수증 탈취

[그림 3]처럼 악의적인 공격자의 휴대폰 번호가 아닌 A사의 전자 영수증 어플리케이션을 사용하는 일반 사용자의 휴대폰 번호를 추측하기 위해서는 일반적인 휴대폰 번호의 11자리 중 앞의 3자리를 제외한 8자리에 대한 번호를 추측한다. 휴대폰 번호 하나당 존재할 수 있는 번호의 수는 0에서 9이다.

이에 대해 8자리에 대해 모든 경우의 수는 1억 개이다. 하지만 일반적으로 일반 사용자가 사용할 수 있는 휴대폰 번호는 한정적이다. 이렇게 추측 공격을 수행하기 위해 일반적인 사용자의 모든 휴대폰 번호를 수집하거나 생성한다. 이렇게 존재할 수 있는 모든 휴대폰 번호를 생성한 후 [그림 3]처럼 공격자의 번호가 아닌 생성된 휴대폰 번호로 수정하여 서버로 전송한다.

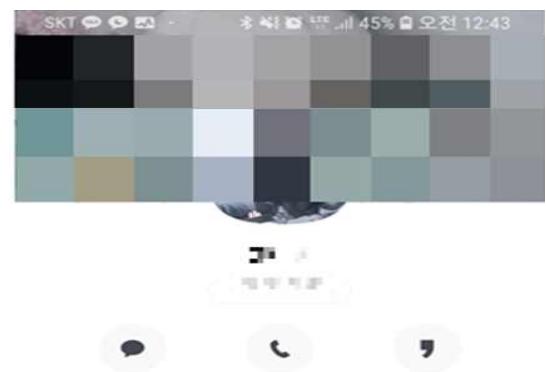
2019년 1월 24일 오전 12시 02분 기준 총 79009 번을 A사의 서버로 요청 하였으며 [그림 6]처럼 총 86명의 사용자의 휴대폰 번호와 전자 영수증 내역을 탈취 할 수 있었다.

A사의 어플리케이션을 사용하는 사용자 휴대폰 번호		
0109-5199-3.txt	2019-01-18 오후 7:04	텍스트 문서 1KB
0108-5199-0.txt	2019-01-18 오후 7:35	텍스트 문서 3KB
0108-5199-8.txt		
0103-5199-1.txt		
0103-5199-8.txt		
0103-5199-4.txt	2019년 01월 01, 16 23, 20원 01, 14 5, 300원 01, 13 15, 960원 15, 960원 01, 13 26, 600원	2019년 01월 01, 17 01, 18 01, 18 Var iousBean 6, 000원 01, 17 7, 700원 01, 13 4, 900원 01, 17 C_F모텔 45, 000원 01, 17
0105-5199-3.txt		
0109-5199-3.txt		
0106-5199-0.txt		
0106-5199-4.txt		
0108-5199-5.txt		
0109-5199-4.txt		
0103-5199-8.txt		
0109-5199-6.txt		
0104-5199-4.txt		
0102-5199-6.txt		
0106-5199-8.txt		

노출된 전자 영수증 목록		
	5KB	
	2KB	
2019-01-20 오후 2:21	텍스트 문서	3KB
2019-01-20 오후 4:50	텍스트 문서	2KB

[그림 6] A사의 어플리케이션의 사용자와 영수증 내역

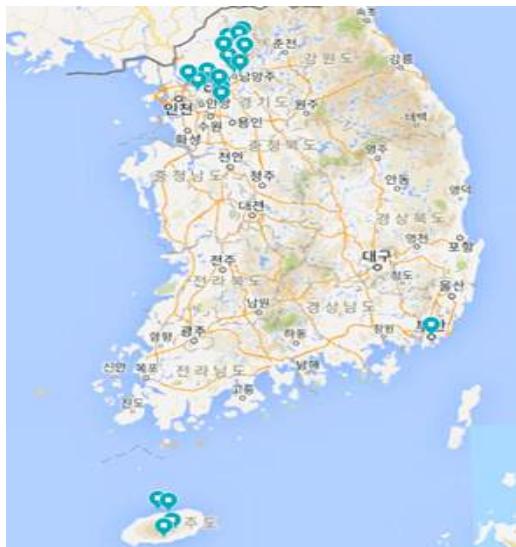
[그림 6]처럼 노출된 A사의 어플리케이션의 사용자의 휴대폰 번호와 해당 사용자의 전자 영수증 내역을 획득한 악의적인 공격자는 2차 공격을 진행할 수 있다. 예를 들어 메신저 어플리케이션을 이용해서 탈취한 휴대폰 번호로 의도적으로 친구추가를 진행 한다. [그림 7]은 악의적인 공격자가 다른 어플리케이션을 통해 해당 사용자의 사진이나 소속 등을 알 수 있기 때문에 2차 피해를 입을 수 있다.



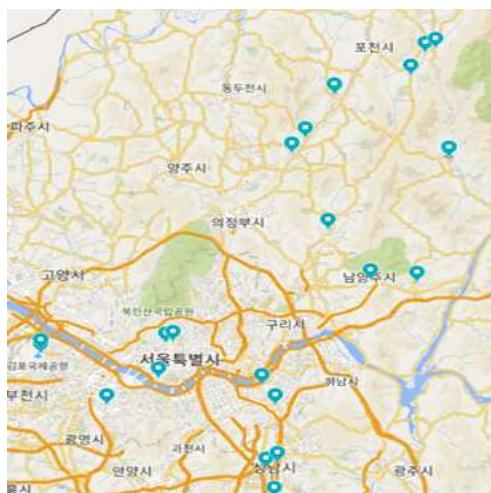
[그림 61] A사의 어플리케이션 사용자의 실제 사진

4.2 다른 사용자의 위치 추적

[그림 3]처럼 사용자가 사용한 전자 영수증에는 사용자가 사용한 위치가 적혀있다. 악의적인 공격자는 A사 전자 영수증 어플리케이션을 사용하는 사용자 모든 영수증을 탈취하여 전자 영수증에 적혀있는 주소를 이용하여 사용자의 위치를 추적하거나 사용자의 행동 범위를 알 수가 있다. [그림 8]과 [그림 9]는 Google 서비스 중 Google maps³⁾라는 서비스를 이용하여 악의적인 공격자에 의해 탈취 당한 사용자의 전자 영수증을 1개월간의 행동 범위를 추적한 결과 아래 같이 행동 범위를 알 수 있었다.



[그림 62] 사용자의 이동 경로(전체 지도)



[그림 63] 사용자의 이동 경로(상세 지도)

V. 취약점 개선방안

발견된 취약점은 전자 영수증 어플리케이션 개발업체에서 개발 단계에서 보안적인 측면을 조금만 고

려하였어도 방지 할 수 있었던 간단한 취약점이다. 이러한 공격을 큰 비용 없이 제일 간단하게 방어할 수 있는 기법은 해당 HTTP URL 메타 정보를 해시 함수를 통해 악의적인 사용자가 일반적인 사용자로 위장하여 위와 똑같은 방법을 사용하더라도 URL에는 식별 할 수 없는 해시값이 들어가 있기 때문에 본 논문에서 소개하는 2가지 공격기법을 방어 할 수 있다.

VI. 결론

본 논문은 전자 영수증 어플리케이션에 대한 취약성을 알아보았다. 해당 취약점은 HTTP URL 메타데이터를 악의적인 사용자가 마음대로 조작할 수 있다는 점과 해당 어플리케이션 서버에서 아무런 사용자의 인증을 거치지 않고 다른 사용자의 전자 영수증을 확인 할 수 있는 2가지의 취약점이 존재하였다. 이렇게 보안이 약한 서비스나 어플리케이션을 통해 노출된 개인정보로 2차 피해로 이어 질 수 있기 때문에 서비스 제공업체나 어플리케이션 개발자들은 보다 보안적인 측면을 항상 고려해야한다.

[참고문헌]

- [1] Wireshark, <https://www.wireshark.org/download.html>, 2019.01.24
- [2] Rescorla, E., & Schiffman, A. (1999). The secure hypertext transfer protocol (No. RFC 2660).
- [3] YoungJae Maeng, DaeHun Nyang. (2008). An Analysis of Replay Attack Vulnerability on Single Sign-On Solutions*. Journal of the Korea Institute of Information Security & Cryptology, 18(1), 103-114.
- [4] Ornaghi, A., & Valleri, M. (2003). Man in the middle attacks. In Blackhat Conference Europe.
- [5] Lee, S., Kim, J., Ko, S., & Kim, H. (2016, August). A security analysis of paid subscription video-on-demand services for online learning. In Software Security and Assurance (ICSSA), 2016 International Conference on (pp. 43-48). IEEE.

³⁾ <https://www.google.co.kr/maps/>

디지털 포렌식 관점의 Windows 10 Sticky Notes 사용흔적 분석에 대한 연구

채진희*, 허원석**

*순천향대학교 정보보호학과, **고려대학교 정보보호대학원 정보보호학과

A study on the use of Windows 10 Sticky Notes for digital forensics

Jin-Hee Chae*, Wonseok Heo**

*Department of Information Security Engineering, SoonChunHyang University, **Graduate School of Information Security Korea University

요약

Windows 7 이후부터 기본 응용프로그램으로 설치된 메모 프로그램인 스티키 노트(Sticky Notes)는 출시 이후부터 많은 사람들에 의해 사용되어 왔으며, 원하는 정보를 간단하게 기록할 수 있다[1][3]. 스티키 노트는 사용자 메모 프로그램이므로 작성된 내용은 사용자의 행위와 관련이 있다고 할 수 있을 것이다. 스티키 노트는 Windows 10 레드스톤1(Redstone) 업데이트 이후부터 이전 버전들과 다르게 SQLite로 데이터 저장 형태가 변경되었다[5]. 이에 본 논문에서는 디지털포렌식 관점에서 Windows 10 환경의 스티키 노트 아티팩트를 분석하여 삭제된 데이터를 확인할 수 있는 방안에 대해 제시한다.

I. 서론

컴퓨터 환경에서 사용자의 행위를 분석하기 위해서는 다양한 아티팩트(Artifact)들이 필요하다. 스티키 노트(Sticky Notes)는 주로 사용자의 일정 및 계획, 기억해야 할 일 등을 간단하게 기록할 때 사용한다. 이는 Windows Vista부터 사이드 바(Sidebar)의 가젯(Gadget)으로 내장되어 Windows 7 이후부터 기본 응용프로그램으로 설치된 기능이다(Home Basic 제외)[1][2].

2016년 기준 스티키 노트의 월간 활성 사용자는 약 800만명 수준이었으며[3], 현재는 더 많은 사용자들이 사용하는 것으로 예측할 수 있을 정도로 스티키 노트는 많이 사용되고 있는 프로그램이라고 할 수 있다.

스티키 노트는 기존의 메모 프로그램들과 마찬가지로 사용자들이 자신의 스케줄이나 간단

한 메모, 또는 아이디와 패스워드와 같은 중요 정보 등을 저장하기도 한다.

스티키 노트는 Windows 10의 레드스톤1 업데이트 이후 데이터 저장 형태가 SQLite로 변경되었으며, 이러한 구조 변경으로 인해 SQLite에 저장된 데이터를 사용자가 삭제하더라도 실제 레코드의 데이터가 남게 되는 현상이 발생하게 되었다[4].

따라서 스티키 노트의 구조를 분석함으로써 삭제된 데이터를 확인할 수 있다면, 이는 디지털포렌식 관점에서 충분한 의미를 가지고 있다고 할 수 있다. 이에 본 논문에서는 SQLite 데이터베이스의 특성을 기반으로 스티키 노트에 대한 아티팩트를 분석하고 삭제된 데이터를 확인할 수 있는 방안에 대하여 제시하고자 한다.

II. 본론

2.1 스티키 노트 데이터 저장 경로

스티키 노트 관련 아티팩트는 Windows 10 레드스톤1 업데이트 버전 1607 후 경로 및 저장 포맷이 변경되었으며, 업데이트 후의 데이터 저장 경로는 아래 [표 1]과 같다[5].

[표 1] 스티키 노트 데이터 파일 저장 경로

```
%UserProfile%\AppData\Local\Packages\Microsoft.MicrosoftStickyNotes_8wekyb3d8bbwe\LocalState\plum.sqlite
```

2.2 스티키 노트 아티팩트

Windows 10 레드스톤 업데이트 이후 스티키 노트의 데이터는 plum.sqlite 파일에 저장된다. 스티키 노트 3.0 버전 이후의 plum.sqlite 파일은 7개의 테이블(Table)로 구성되어 있으며(Media, Note, Stroke, StrokeMetadata, SyncState, UpgradeNote, User), 이 중 사용자 데이터와 관련된 내용은 Note 테이블에 존재한다.

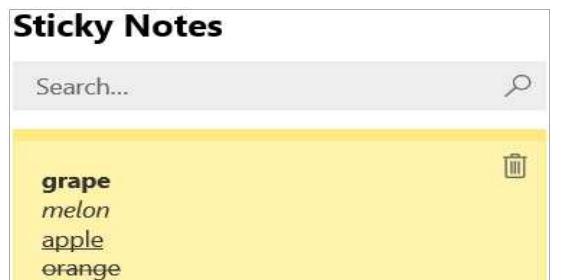
3.0 이전버전은 Media 테이블을 제외한 6개의 테이블이 존재한다.

Note 테이블의 경우 사용자 관련 데이터가 저장되어 있어 본 논문에서 주요하다 판단되는 테이블이며, 해당 테이블의 컬럼(Column)은 [표 2]과 같이 구성되어 있다.

[표 2] Note 테이블 정의

컬럼(Column)	설명(Description)
Text	메모에 저장된 텍스트 (텍스트 한 줄씩 다른 id값을 부여)
IsOpen	메모가 바탕화면에 존재할 때 활성화
Theme	노트 배경색 (Green, Pink, White, Yellow, Blue, Gray)
LastServerVersion	JSON 형식으로 저장된 텍스트 및 설정 정보 (RemoteId, ChangeKey, Id 등 포함) (동기화 시 활성화)
RemoteSchemaVersion	SQLite 버전 (동기화 시 활성화)
Id	스티키 노트 메모 고유 id값
ParentId	PC를 구분하기 위한 id값 (하나의 PC는 동일한 id를 가짐)
CreatedAt	생성된 날짜 (18자리, Windows NT time format)
UpdatedAt	업데이트 날짜 (18자리, Windows NT time format)
WindowsPosition, IsAlwaysOnTop, CreationNoteIdAnchor, IsFutureNote, RemoteId, ChangeKey, IsRemoteDataInvalid, Type, DeletedAt	현재 확인 중에 있음

스티키 노트 3.0 버전 이후부터는 디바이스 간 동기화 및 메모 관리 기능 등이 추가되었다 [6]. Note 테이블에서 한 줄의 행은 스티키 노트 메모 하나를 의미한다. [그림 1]와 같이 메모를 저장하였을 시 [그림 2]처럼 메모 하나씩 고유 id 값이 부여된다. 그리고 Text 컬럼에는 [표 3]와 같이 실제 작성된 메모 내용이 저장되며, 꾸미기 옵션(진하게, 기울기, 밑줄 등)이 활성화될 경우, 문자열 앞뒤에 특정 문자와 끝에 0이 추가된다.



[그림 64] 저장된 스티키 노트 텍스트

Id	
필터	
31f90695-df7...	
a6e8c945-1d...	
35b32ece-44...	
217f436e-2e...	
7a2c2069-61...	
7c96319c-a2...	
d6ba2f95-59...	
011ff401-625...	
2f5a6a9e-70...	

[그림 65] 삭제하기 전 메모의 id (plum.sqlite 내)

[표 3] Text 필드 저장 형태

```
\id=307fde62-eaa8-496c-a244-1bed95887b47\b
grape\b0
\id=b94427a6-7edd-4ac2-98f4-971054c47661\i
melon\i0
\id=2bb936a9-6d10-407c-93c2-5f2c580350e4\ul
apple\ul0
\id=dad33af7-8ec5-4e5f-acfb-62aee6b48c4e\strike
orange\strike0
```

Free_Pages	00102400 - 00106495	00 00 00 00 0107 44 00 07 44 00 00 00 00 00 00I
프리리스트 페이지	00106506 - 00108208	0E A5 02 3D 00 00 00 00 00 00 00 00 00 00 00 00 ... ?=...
Note	00122847 - 00122879	39 63 31 30 2D 64 61 31 31 34 30 38 35 66 39 39 ... 9c10...
Note	00122890 - 00123664	02 1A 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Free_Pages	00126976 - 00131071	00 00 00 22 20 20 20 20 20 22 62 6C 6F 63 68 53"
Note	00135122 - 00135167	92 08 D6 58 C4 77 0E 26 40 00 00 00 00 00 00 00 00 ... ?=.....I
Free_Pages	00135168 - 00139263	00 00 00 00 2D 34 32 63 62 2D 39 62 36 62 2D 61 35"

[그림 67] 삭제된 데이터 (plum.sqlite 내)

2.2 삭제 데이터 확인

SQLite는 레코드 삭제 시 해당 레코드를 데이터베이스 내에 사용되지 않는 페이지(Page)로 할당하게 된다. 다른 데이터들이 추가로 저장될 경우 기존의 페이지 내 저장 공간을 활용하지만 만약, 페이지에 더 이상의 저장이 불가능할 경우엔 추가 페이지 할당을 하게 되므로 기존 삭제된 공간을 재사용하게 되나, 그렇지 않은 경우에는 데이터베이스 엔진에 의해 프리리스트(freelist) 페이지로 계속 데이터가 저장된 상태로 남게 된다[7][8].

만약, 데이터베이스 파일에 “Vacuum 모드”가 설정되어 있다면 삭제된 영역에 대해 청소 기능이 실행되어 레코드 삭제와 동시에 해당 영역이 초기화되지만, 이는 빠른 데이터 처리를 지향하는 SQLite에 있어서 기본적으로 활성화되어 있지 않은 경우가 대부분이다. 이러한 특징 때문에 SQLite에서 삭제된 레코드, 즉, 데이터에 관한 흔적을 발견할 수 있다.

만약 사용자가 작성된 스티키 노트의 내용을 삭제할 경우 사용자 화면에서는 해당 내용이 삭제된 것으로 보이나 실제 plum.sqlite 파일에는 아래 [그림 3], [그림 4]와 같이 삭제된 메모의 id값을 기준으로 삭제된 데이터가 존재하는 것을 확인할 수 있다.

00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F	22 64 6F 63 75 6D 65 6E 74 4D 6F 64 69 66 69 65 "documentModifie
64 41 74 22 3A 20 22 32 30 31 39 2D 30 32 2D 30	dlt": "2019-02-0
37 54 31 33 3A 34 34 3A 30 37 2E 31 30 38 33 34	7113:44:07.10834
32 37 5A 22 0D 0A 7D 03 32 66 35 61 36 61 35 65	2722".J...F556a959
2D 37 30 34 62 2D 34 64 61 39 2D 39 64 36 66 2D	-704b-4da9-9def-
32 39 65 62 63 63 32 66 31 38 33 65 63 31 34 32	29ebcc2f183e.c142
31 30 64 32 2D 39 62 61 62 2D 34 31 64 35 2D 39	10d2-9bab-41d5-9
63 31 30 2D 64 61 31 31 34 30 38 35 66 39 39 39	c10-dai14085f999
08 D6 8D 02 3F 05 B3 43 08 D6 8D 02 54 C3 F5 A3	.0. .? .C.O..TA6t
0D 00 00 00 01 06 B1 00 06 B1 0E A5 02 3D 00 00±.±.¥=..

[그림 3] 삭제된 메모의 id값 (plum.sqlite 내)

삭제된 데이터는 프리리스트 페이지로 저장되거나 데이터베이스 내 할당되지 않은 공간에 남아있다. 할당되지 않은 공간은 삭제된 데이터 또는 이전에 사용된 페이지의 잔여물이 남아있기도 하며, 다른 곳에서 참조되지 않는다.

[그림 4]와 같이 SQLite 파싱(Parsing) 도구를 이용할 경우, 프리리스트 페이지에 할당된 삭제된 데이터를 찾을 수 있으나 완벽하게 복구할 수 없는 경우가 있어 이 경우 바이너리 파일에서 직접 검색하여야 한다[7].

III. 결론 및 향후 연구 과제

본 논문에서는 Windows 10 기능이자 기본 응용프로그램 중 하나인 스티키 노트를 디지털 포렌식 관점에서 구조를 분석하고 이를 통해 확인할 수 있는 데이터와 삭제된 데이터를 확인할 수 있는 방안에 대해서 제시하였다.

스티키 노트는 현재 많은 사용자가 사용하고 있으며, 저장되는 데이터는 ID나 패스워드와 같은 계정정보가 포함되거나 사건에서 중요한 정보들이 저장되었을 가능성도 있어 언급한 아티팩트가 가지는 활용가치는 크다고 할 수 있다.

다만, 본 논문에서는 plum.sqlite에 대한 아티팩트가 완전하게 분석되지 않은 상태이므로 향후 전체 컬럼에 대한 아티팩트를 추가 분석한다면 디지털포렌식 관점에서 더욱 의미 있는 결과를 도출할 수 있을 것이다.

[참고문헌]

- [1]“Using Sticky Notes in Windows Vista, 7, and 10“, Sep 7, 2018, <https://www.lifewire.com/using-sticky-notes-in-windows-7-3506962>
- [2]“Sticky Notes Substitute for Windows 7 Home Basic”, <http://www.codingwhiz.com/windows-apps/sticky-notes-substitute-for-windows-7-home-basic.html>
- [3]“Windows 10 Anniversary Update: Best New Features“, Andrew E.Freedman, April 7, 2016, <https://www.laptopmag.com/articles/windows-10-anniversary-features>
- [4] “[Tech Report] ‘구조’를 알면 ‘복구 가능성’이 보인다”, May 05, 2014, https://www.ahnlab.com/kr/site/securityinfo/secunews/secuNewsView.do?cmd=print&seq=22438&menu_dist=2
- [5] “Sticky Notes in Windows 10: Tips to use, save, format, backup, restore”, TheWindowsClub, <https://www.thewindowsclub.com/tips-to-use-format-sticky-notes-in-windows-7>
- [6] “Sticky Notes 3.0 now available to all Windows 10 April 2018 users”, Surur, Sep 25, 2018, <https://mspoweruser.com/sticky-notes-3-0-now-available-to-all-windows-10-april-2018-users/>
- [7] “Forensic Analysis of SQLite Databases: Free Lists, Write Ahead Log, Unallocated Space and Carving”, BELKASOFT, Feb 16, 2015, <https://articles.forensicsfocus.com/2015/02/16/forensic-analysis-of-sqlite-databases-free-lists-write-ahead-log-unallocated-space-and-carving/>
- [8] “Database File Format”, SQLite, Jun 18, 2004, <https://www.sqlite.org/fileformat.html>

스마트 기기의 디지털 증거 훼손 방지 방안에 대한 연구 (생체인증 방식이 적용된 단말을 중심으로)

황혜성*, 이세영**, 허원석***

*서울여자대학교 정보보호학과, **경북대학교 컴퓨터학부, ***고려대학교 정보보호대학원
정보보호학과

Study on method for preventing digital evidence damage of smart
device - Focused on device with biometric authentication

Hye-Seong Hwang*, Se-Yeong Lee**, Wonseok Heo***

*Division of information security, Seoul Women's University, **Computer
Science and Engineering, Kyungpook National University, ***Graduate
School of Information Security Korea University

요약

최근 애플의 Touch ID, Face ID와 삼성전자의 인텔리전트 스캔과 같은 생체인식
기술을 이용한 인증방식에 대한 적용이 점차 증가하고 있다. 생체인증의 경우 안전하
게 기기를 관리할 수 있고 기기에 대한 잠금 해제를 쉽게 할 수 있다는 것에서 편의
성의 장점이 있다. 그런데 이러한 인증과 연결되어 인증 시도의 임계치에 따라 인증
실패에 대한 데이터 전체 삭제 기능이 존재하므로 만약, 여러 번 인증 실패가 발생할
경우 자동으로 데이터가 삭제될 수 있다. 이는 디지털포렌식 관점에서 증거 훼손에 해
당할 수 있어 안티포렌식 기법 중 하나로 할 수 있으므로 본 논문은 생체인식 기술을
이용한 스마트 기기 인증방법 중 하나인 얼굴 인식을 기준으로 무결성 훼손을 방지할
수 있는 대안을 제시하고자 한다.

I. 서론

ICT기술이 발전함에 따라 다양한 디지털 기
기가 출시되며 되었고 그 중 스마트 기기 또한
많은 변화가 발생하게 되었다. PIN 인증에서부
터 패턴 인증, 지문 인식과 같은 생체인증 기술
이 발전하게 되었다. 이 중 스마트폰 시장 점유
율이 가장 높은 삼성전자, 애플에서 선도적으로
홍채 인식과 얼굴 인식 그리고 이를 결합한 인
텔리전스 인식을 이용한 인증방법을 적용하여
현재는 스마트폰에 어떠한 물리적 접촉 없이도
인증이 가능한 시대가 되었다. 차세대 인증방법
으로 인해 보다 편리해짐은 분명하나 디지털포

렌식 관점에서는 데이터가 훼손 될 수 있는 부
정적 영향도 존재한다.

만약 여러 번 생체인증 실패로 인해 데이터
가 초기화가 될 수 있고 피압수자가 생체 인증
으로 인증을 해제함에 따라 데이터의 변화가
발생할 수도 있다.

디지털포렌식의 경우 증거수집 절차에서 스
마트 기기와 같은 전자 매체 내에 저장된 데이
터를 추출하게 되며 이때 원본 데이터의 훼손
을 최소화하여 수집해야 한다. 이를 디지털 증
거수집의 기본원칙 중 ‘증거의 무결성 확보 원
칙’이라고 한다.[1]

디지털포렌식 분야 중 스마트 기기 포렌식은 iOS, Android OS를 사용하는 스마트 기기 내에 존재하는 디지털 데이터를 증거로서 수집하고 데이터를 분석하는 등 일련의 과정을 포함하는 디지털포렌식 분야 중 하나를 말한다.[2] 이러한 스마트 기기 포렌식의 경우 스마트 기기에 대한 이미징 기술 및 데이터 분석과 관련된 도구가 현재 많이 연구가 된 상태이다.

따라서, 본 논문은 스마트 기기의 기술적 관점에서의 안티포렌식에 대한 대응보다 관리적 관점에서의 안티포렌식 대응 방안을 제시하고자 하며, 그 중 애플 iOS의 Face ID와 삼성전자의 인텔리전트 스캔을 중심으로 설명한다.

II. 본론

2.1 스마트 기기에 적용된 생체인증 기술

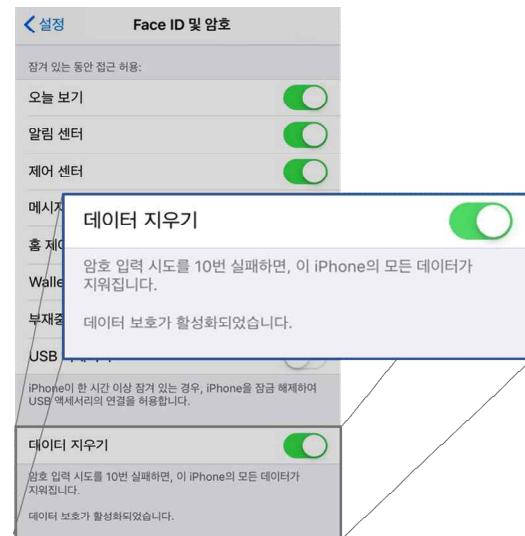
애플의 Face ID는 기기의 전면에 위치한 도트 프로젝터가 30,000개 이상의 보이지 않는 점을 사용자 얼굴에 투영하고 분석하여 얼굴 데이터를 수집한다. 수집한 데이터를 중심으로 사용자 얼굴의 특징을 맵으로 만들고 적외선 카메라로 얼굴에 대한 적외선 이미지를 촬영해, 이 두 가지 정보를 학습 데이터로 가공하여 학습하게 된다. 그 후 사용자가 기기에 대한 인증을 시도할 때 인식된 얼굴과 등록된 데이터와 대조하여 식별하는 방식이 사용된다.[3]

삼성전자의 인텔리전트 스캔은 얼굴과 홍채 정보를 모두 사용하여 인증하는 기술이다. 밝은 야외와 같이 홍채 인식이 어려운 경우에는 얼굴 인식으로 보완한다. 또한 홍채 인식 시마다 외모를 자동 업데이트하며, 외모 변화가 큰 경우 홍채 매칭 후 얼굴 정보를 업데이트를 가장 최신에 인증된 얼굴 정보를 이용하여 인증 시 대조할 데이터로 사용하는 것이 특징이다.[4]

2.2 안티 포렌식 관련 기능

잠금 기능이 있는 애플의 모든 모델에는 [그

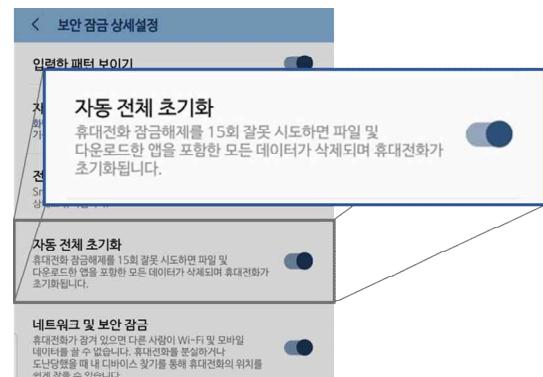
림 1]과 같이 10번의 인증 시도가 실패되면 데이터를 삭제할 수 있는 기능을 갖고 있다. 이 기능을 활성화한 후 인증 실패 임계치가 초과될 경우 스마트 기기 내의 데이터와 설정 값들이 초기화된다.



[그림 68] 데이터 초기화 기능(아이폰X)

삼성전자의 인텔리전트 스캔 기술이 적용된 갤럭시 S9를 포함한 이후 모델의 경우에도 애플의 제품과 유사하게 15번의 인증 시도가 실패되면 데이터를 삭제할 수 있는 기능을 갖고 있다.([그림 2] 참조)

만약 이 기능이 활성화 되어있을 경우 파일 및 다운로드한 앱을 포함한 모든 데이터가 삭제된다.



[그림 69] 데이터 초기화 기능(노트9)

2.3 안티 포렌식 시나리오

만약 피압수자가 Face ID나 인텔리전트 스캔 기능이 적용된 스마트 기기를 사용하는 경우 다음과 같은 안티 포렌식 상황이 발생할 수 있다.

첫 번째, 압수·수색 과정에서 여러 번 잘못된 인증을 시도하여 스마트 기기의 초기화를 유도 할 수 있다. 만약, 피압수자가 직접 스마트 기기에서 초기화 버튼을 클릭할 경우 직접 증거를 인멸 또는 변조하였다 볼 수 있어 형법 제155 조(증거인멸 등과 친족 간의 특례)의 위배되는 행위라 할 수 있다. 하지만 인증 실패로 인한 데이터 초기화에 대해서 아직까지 증거인멸로 인정하는 판례가 존재하지 않아 이를 악용할 소지가 있다.

두 번째, 수사관이 스마트 기기와 같은 정보 저장매체를 압수하여 라이브 데이터의 훼손을 최소화 하고자 전원을 유지한 채 일반적으로 압수·수색 과정에서 사용하는 정전기방지 봉투나 충격방지 봉투와 같은 차폐봉투를 이용한다. 이 때 스마트 기기의 생체인증 기능이 동작된다면 인증 실패 임계치가 초과되어 데이터 초기화가 발생할 수 있다.

2.4 대안 제시

본 절에서는 앞 절에서 제시한 각 시나리오 상황의 대안들을 제시한다.

첫 번째는 압수·수색 절차의 개시부터 차폐봉투에 정보저장매체를 보관하기까지의 과정에서 수사관은 정보저장매체를 압수할 경우 가장 먼저 전면 카메라 부분을 빛이 투과하지 않는 상태로 가려 최대한 압수대상물이 기기 잠금 해제 시도가 되지 않도록 하여야 한다. 또한 데이터 수집을 하고자 피압수자에게 기기 잠금 해제를 요청할 경우 직접 기기를 전달하는 것이 아니라 잠금 해제 비밀번호를 확인하는데, 이 때 피압수자가 인증 오류를 지속적으로 발생시킬 경우 인증 시도를 즉시 차단하고 JTAG을 이용하거나 메모리칩을 분리하는 등 물리적인 데이터 추출 방법을 이용하여야 한다.

두 번째는 압수된 정보저장매체가 차폐봉투 내에서 생체인증 기능이 동작하지 않도록 차폐봉투에 특정 무늬를 삽입하거나 생체인증 기능에 사용되는 특정 적외선 주파수를 측정하여 차단함으로써 적외선 카메라의 기능을 차단하는 방법이다.

마지막으로 첫 번째 대안과 유사하게 정보저장매체를 입수할 경우 즉시 전면 카메라의 적외선 카메라와 투광 일루미네이터, 도트 프로젝터 등 생체인증에 사용되는 카메라들에 빛이 투과되지 않는 재질의 보안 스티커를 부착하는 것이다. 대부분의 생체인증 기능을 제공하는 스마트 기기의 경우 인증 모듈이 부착된 범위가 가로 10cm를 초과하지 않으므로 정규 사이즈의 보안 스티커 제작이 가능하다.

그리고 보안 스티커를 이용할 시 스티커를 부착한 일시, 증거번호, 압수대상물 정보, 담당자 정보 등을 기입할 수 있도록 하여 해당 스티커를 부착한 시점에서는 부정적인 인증이 시도되지 않았음까지 증명할 수 있도록 하는 것도 무결성을 입증하는 근거가 될 수 있다.

III. 결론 및 향후 연구 과제

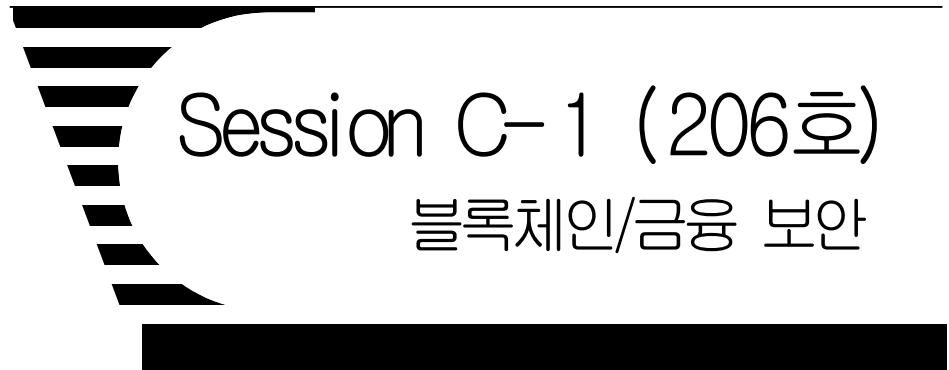
본 논문에서는 생체인식 기능 중 얼굴 인식의 기능에 초점을 맞추어 압수·수색 절차에서 발생할 수 있는 안티 포렌식 관점에서의 증거데이터 유실 상황과 그에 대한 대안을 제시하였다. 최근 아이폰XS, 갤럭시 노트9 등을 포함하여 생체인식 기능이 탑재된 스마트 기기의 출시가 증가하고 있고 사용성이 좋은 얼굴인식을 사용하는 사용자 또한 증가하고 있는 추세이다.

따라서, 디지털포렌식 관점에서의 증거인멸 방지에 대한 수준을 향상시키기 위한 노력이 더욱 필요한 상황이라 할 수 있다. 본 논문에서는 이러한 증거인멸 방지에 대해서 절차적인 측면에서만 제안하였으나, 보다 안전하고 정확도가 높은 생체인증 기술을 악용하는 안티 포렌식을 방지하기 위해서는 적외선 광장에 대한

범위 연구나 데이터 초기화 기능을 종료할 수 있는 기술적 대안 등이 추가적으로 연구되어야 할 것이다.

[참고문헌]

- [1] 이상진 등 13명, “디지털 증거 수집보존 가이드라인” TTA표준화 위원회, 2017.12
- [2] 김건우, 은성경 “시각화 기법을 활용한 스마트폰 포렌식 분석”, 디지털포렌식기술 워크샵, 2013년 8월.R
- [3] Apple, “Face ID에 적용된 첨단 기술에 관하여, <https://support.apple.com/ko-kr/HT208108>”
- [4] 이성훈 등 3명, “얼굴 인식에서의 스푸핑 공격 탐지 연구 동향” 정보통신기술진흥센터, 2018.10



좌장 : 신상욱 (부경대)

블록체인 기반의 분산 EHRs 저장소 프라이버시 보호에 관한 연구

산디 라마디카, 이경현*

*부경대학교 정보보호학협동과정

부경대학교 IT융합응용공학과

sandika@pukyong.ac.kr, khrhee@pknu.ac.kr*

Research on Privacy of Decentralized EHRs

Sandi Rahmadika and Kyung-Hyune Rhee*

Interdisciplinary Program of Information Security, Graduate School PKNU

*Department of IT Convergence and Application Engineering

Pukyong National University, Republic of Korea

Abstract

Electronic health records (EHRs) provide more than a few benefits in the digital healthcare system, yet it improves the way access to the records for the parties. Research on the development of EHRs is increasingly being carried out including the application of blockchain technology to address various challenges. The merits of these technologies give many advantages in terms of data quality control, resistant to the data tampering, hacking, and data manipulation. However, it also brings the drawbacks since the data is highly sensitive to certain parties such as medical records. In short, the nature of public blockchain exposed the data to the public on the peer-to-peer network which is likely could be misused by the adversary. In this paper, we present the privacy techniques to address the privacy issues on the blockchain network. Some remarks are also elaborated.

I. Introduction

The digital revolution on the biomedical research brings numerous benefits in the healthcare industry [1]. The current system of healthcare record is disjointed and which brings to a lack of a common standard between the parties involved [2]. Moreover, it brings the difficulty to manage the stored data since the patient does not possess the full access on the provider's database.

Blockchain comes with all the advantages of offering solutions that are faced in the centralized healthcare model. The architecture of the blockchain allows the parties to have one version of data even though data comes from various providers. Every activity that occurs is recorded and the information stored is unlikely to be changed. Therefore, the development of the blockchain in the healthcare domain is increasing lately.

The data storage is vital in the decentralized system. A straightforward storage model with guaranteed privacy protection on the peer-to-peer network is a challenge in the decentralized healthcare system. The surveys indicate that users often do not fully trust to store their data to third parties or to cloud storage providers [3].

In this paper, we present an approach to provide privacy of the parties in the public blockchain system. There are many protocols that can be used on the various model. We thoroughly select several protocols that we consider suitable for public blockchain.

II. The Core System

The ring signature protocol [4] is one of the ways which is likely suitable in order to provide the user privacy. The signer requires public keys Pk knowledge from prospective

7members as shown in Figure 1. The selected public keys are encrypted by using a trapdoor permutation function (RSA, Rabin, and Diffie-Helman).

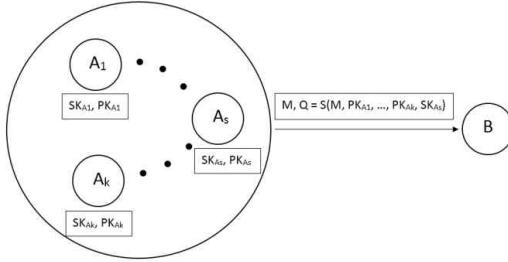


Figure 1. Ring signature in general

Sign σ ($msg, Psn, Pk1, Pk2, Pk3, \dots, Pkn$). The signature consists of the public keys ($Pk1, Pk2, Pk3, \dots, Pkn$) of the members for every message msg concatenated with the secret key Psn of the signer to produce a signature σ , and the *verify* (msg, σ). The verification process can be interpreted as accepting a group signature σ along with the messagemsg, and the output is true or false.

$$P = Hs(rA\alpha)G + Ba$$

$$P' = Hs(a\alpha R)G + Ba, \text{then}$$

$$a\alpha R = a\alpha rG = rA\alpha; P' = P$$

In the decentralized healthcare system, the patient and providers have a pair of public keys (A_n, B_n). The A_n is used to generate a one-time public key, and B_n is attached to the transaction as the tracking value.

$$\begin{aligned} Rsg = & y\alpha \oplus y\beta \oplus yy \oplus y\delta \oplus y\epsilon \\ & \oplus y\zeta \oplus y\eta \oplus \dots \oplus y\{n+1\} \end{aligned}$$

The provider uses P as a destination key for the output and attaches the new value $R = rG$ into the transaction. The PHI data with attachments to P and R values are stored into shared storage after being validated by the miner. The patient later checks every transaction using his private key ($a\beta, b\beta$) and calculates the new P' , and compares the value P received with the value P' decrypted.

The signature of a group is free to be used for any transaction without having permission from the owner of the key. For instance, a particular transaction the provider $y\beta$ uses the following keys to sign a message $h(m, yy, y\delta, ya$ and his key $y\beta$.

$$Rct = txId [(1) // (4) // (3) // (2) // (5)]$$

A technique of ring confidential transactions [5] could be applied to the system by combining transactions that have already occurred so it is possible to disguise the value of the current transaction.

III. Decentralized EHRs

A decentralized healthcare [6] system consists of the patient, doctors, and the healthcare providers. The parties manage the EHRs by leveraging the same blockchain framework. In practice, the doctor as the sender intends to send the diagnosis data to the patient. The doctor creates a ring signature group based on public key of the parties that have been known beforehand. The doctor also added its public key to the group. In a nutshell, the doctor asks the stealth address of the recipient and follow the procedure as shown in Algorithm 1.

Algorithm 1: EHRs Storage

```

1: Procedure EHRs_Storage:
2:   Generate the keys:
3:     Patient  $\leftarrow$  hash (Pub $\alpha$ , Pr $\alpha$ )
4:     Hospital  $\leftarrow$  hash (Pub $\beta$ , Pr $\beta$ )
5:     Chiropractor  $\leftarrow$  hash (Pub $\gamma$ , Pr $\gamma$ )
6:   Ring Signature:
7:   Procedure Use Public Key  $\forall$  parties  $\in$  parentkeys
8:     Create RS:
9:        $R_{sgn} \leftarrow g_\alpha(x_\alpha) \oplus g_\beta(x_\beta) \oplus \dots \oplus g_{n+1}(x_{n+1})$ 
10:    AddNewMembers:
11:      Update  $R_{sgn} \leftarrow Get\_NewPubkey \oplus PubKey_{(n+1)}$ 
12:    end procedure
13:    Procedure StealthAddr
14:      ParentKey: PubKey $_n \rightarrow$  PubKey $_n(A_n, B_n)$ 
15:      Send ( $A_n$ )  $\rightarrow$  to_sender_(via_SecureChannel)
16:      ( $A_n, B_n$ )received  $\rightarrow$  Create_New OTP
17:    end procedure
18:    Procedure Sign the PHI Data (Msg):
19:      Get Msg  $\in$  Diagnosis_Data (Hospital)
20:      Sign  $\sigma \leftarrow$  choose_signer  $\in R_{sgn}$ 
21:      RingCT  $\leftarrow$  choose_prev_txs  $\in CRS$ 

```

Key management is also a concern for the rapid system such as blockchain transaction. Weil pairing [8] technique can be used to address the problem. In short, for the blockchain system the pairing must have the following properties:

- Define the G and GT are the finite cyclic groups of prime order q , the pairing $e: G \times G \rightarrow GT$ is a map.
- Bilinear: for all $P \in G_1$, All $Q \in G_2$ and all $a, b \in \mathbb{Z}$ we have $e(aP, bQ) = e(P, Q)^{ab}$
- Non-degenerate: $e(P_1, P_2) = 1$
- The pairing must be computable. There is an efficient algorithm to compute $e(P, Q)$ for all $P \in G_1$ and $Q \in G_2$.

The decentralized EHRs system can be said as a solution to overcome the challenges in the digital healthcare domain. We assume that shared storage is interconnected to a blockchain network where miners have access into it. In terms of the type of shared storage, we do not define it in detail, instead we assume the shared storage has all the capacity needed to support the proposed system. For future work, the model and capacity of shared storage need to be observed further. The access control in the shared storage is also essential to ensure system keeps safe.

By leveraging the presented protocols, it is possible to provide privacy for the parties including in the decentralized healthcare area. The objectives are outlined as follows:

- **Untraceability.** The observer cannot trace where the transaction was received and where the data originated from [7].
- **Unlinkability.** Stealth address technique to provide the privacy for the recipient. The observer is unlikely to link a transaction to the other transaction.
- **Confidential values.** The value of the current transaction is combined with the values of the previous transaction to disguise the real value.

IV. Conclusion

We have presented the opportunities of blockchain in the digital healthcare domain. The merits of blockchain technology with all its benefits seems suitable to be applied in the healthcare area if only privacy issues can be overcome such as by implementing several protocols. A real model of storage needs to be thoroughly review for the future.

Acknowledgment

This research was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government (MSIT) (No. NRF- 2018R1D1A1B07048944).

[References]

- [1] Mamoshina, Polina, et al. "Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare." *Oncotarget* 9.5 (2018): 5665.
- [2] IBM Global Business Services FTeam, "Blockchain: The Chain of Trust and its Potential to Transform Healthcare - Our Point of View", Office of the National Coordinator for Health Information Technology, 6710 Rockledge, 2016.
- [3] H. Kopp, D. Mödinger, F. Hauck, F. Kargl, and C. Bösch, "Design of a privacy-preserving decentralized file storage with financial incentives," in Proceedings - 2nd IEEE European Symposium on Security and Privacy Workshops, EuroS and PW 2017, 2017, pp. 14 - 22.
- [4] Zhang, Fangguo, and Kwangjo Kim. "ID-based blind signature and ring signature from pairings." International Conference on the Theory and Application of Cryptology and Information Security. Springer, Berlin, Heidelberg, 2002.
- [5] Noether, Shen. "Ring SSignature Confidential Transactions for Monero." IACR Cryptology ePrint Archive 2015 (2015): 1098.
- [6] Campinha-Bacote, Josepha. "The process of cultural competence in the delivery of healthcare services: A model of care." *Journal of transcultural nursing* 13.3 (2002): 181-184.
- [7] Noether, Surae. "Review of CryptoNote white paper." HYPERLINK "http://monero.cc/downloads/whitepaper_review.pdf" (2014).
- [8] Boneh Dan, and Matthew Franklin. "Identity-based encryption from the Weil pairing." *SIAM journal on computing* 32.3 (2003): 586-615.

블록체인 프라이버시 보호 프로토콜 동향

강원태*, 이상현*, 김호원*

*부산대학교 전기전자컴퓨터공학부

kangpoong90@gmail.com, jdsd2233@gmail.com, howonkim@pusan.ac.kr

Blockchain Privacy Protection Protocol Trend

Won-Tae Kang*, Sang-Hyun Lee*, Ho-Won Kim*

*Department of Electrical and Computer Engineering,
Pusan National University.

요약

블록체인 플랫폼은 P2P를 기반으로 하고 있기 때문에 블록체인의 정보는 네트워크에 참여하고 있는 모든 노드에게 공유되게 된다. 이로 인해 블록체인 플랫폼은 투명성이라는 특징을 지니게 되는데 이는 공정거래를 유도하는 긍정적인 속성인 동시에 필요이상으로 거래정보나 개인정보를 노출시키는 부정적인 속성이 되기도 한다. 때문에 블록체인 플랫폼 내에서 필요이상으로 데이터를 노출시키지 않기 위해 많은 기관 및 기업에서 데이터 프라이버시 보호기법에 대해 연구 개발 중에 있다. 본 논문에서는 블록체인에서 데이터 프라이버시를 보호하기 위한 프로토콜 기술에 대하여 동향을 분석한다.

I. 서론

블록체인은 2008년 사토시 나가모토의 논문 Bitcoin: A Peer-to-Peer Electronic Cash System[1]에서 제시된 분산형 데이터베이스 기술로 탈중앙화, 불멸성, 투명성 등의 특징을 가진다. 기존에 제시되었던 기술에 Smart Contract 개념이 도입되며 블록체인은 플랫폼으로 확장되었다.

초기 블록체인은 단순히 암호화폐의 거래내역인 트랜잭션을 저장하는 용도로 사용되었지만, Smart Contract 개념이 도입되면서 다양한 데이터를 블록체인에 저장하게 되었다. 블록체인에 저장된 데이터는 모든 네트워크 참가자들이 열람할 수 있고, 데이터를 삭제할 수 없다는 특성 때문에 데이터 프라이버시가 보호되지 않는다는 문제점이 발생한다. 이러한 문제점을 해결하기 위해 블록체인 상에서 데이터 프라이버시를 보호하기 위한 기법들이 연구되고 있다.

본 논문은 2장에서 블록체인 네트워크에 대

한 배경지식을 알아보고, 3장에서 데이터 프라이버시 보호 기법들에 대해 기술한다. 마지막으로 4장에서 결론을 맺고 향후 연구방향을 제시하고 마친다.

II. 배경지식

2.1 블록체인

블록체인은 분산형 데이터베이스 기술의 한 종류로, 데이터를 저장하는 블록(Block)들이 체인(Chain)형태로 연결된 리스트이다. 블록은 블록 헤더(Block Header)와 블록 바디(Block Body)로 나누어지는데, 블록 헤더에는 타임 스탬프(Time Stamp), 이전 블록의 해시 값(Previous Block Hash) 등이 저장되어 있으며 블록 바디에는 트랜잭션, 즉, 블록체인 상에서 수행된 거래 데이터들이 저장되어 있다. 각 블록은 블록 헤더에 해시 값을 저장함으로써 이전 블록과 연결되어 블록의 데이터를 조작할

수 없다는 특징을 가지고 있다.

2.2 영지식 증명(Zero-knowledge proof)

영지식 증명은 암호학에서 상대방에게 누군가가 어떤 문장이 참이라는 것을 증명할 때, 그 문장이 참이라는 사실을 제외한 다른 어떤 것도 노출하지 않고 증명하는 절차이다. 어떤 문장을 증명하려는 대상을 증명자라고 하며 문장의 사실 여부를 검증하는 대상을 검증자라고 한다. 영지식 증명은 완전성, 건실성, 영지식성이라는 성질을 만족시켜야 한다.

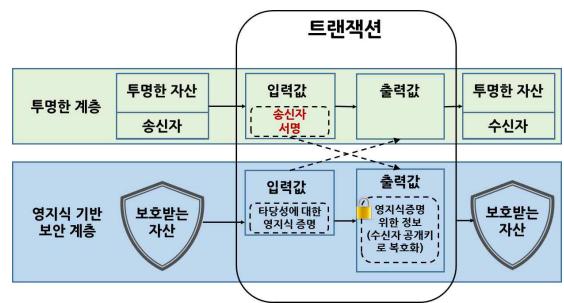
- 완전성은 어떤 문장이 참이라면, 정직한 증명자는 정직한 검증자에게 그 사실을 납득시킬 수 있어야 한다는 성질이다.
- 건실성은 어떤 문장이 거짓이라면, 부정직한 증명자가 정직한 검증자에게 그 문장이 사실이라고 납득시킬 수 없어야 한다는 성질이다.
- 영지식성은 어떤 문장이 참이라면, 검증자는 그 문장의 참, 거짓 외에 다른 어떤 것도 알 수 없어야 한다는 성질이다.

III. 블록체인 프라이버시 보호 프로토콜

블록체인은 P2P시스템을 기반으로 다수의 노드가 데이터를 공유하고 네트워크를 유지할 수 있도록 하는데 이때 블록체인 내의 데이터가 필요이상으로 노출되지 않도록 하는 것이 블록체인 프라이버시 보호 프로토콜이다. 본 장에서는 데이터 은닉을 위한 블록체인 프라이버시 보호 프로토콜에 대해 알아본다.

3.1 zk-SNARK(Zero-Knowledge Succinct Non-interactive Argument of Knowledge)

zk-SNARK는 암호화폐 플랫폼인 ZCash에서 사용하고 있는 프로토콜로 블록체인 트랜잭션에서 [그림 1]과 같이 여러 정보(수신자, 송신자, 전송 금액 등)를 노출하지 않으면서 트랜잭션의 유효성 여부를 다른 노드들에게 공유할 수 있다.[2]



[그림 76] zk-SNARK기반 트랜잭션 처리

zk-SNARK 프로토콜은 Computation → Arithmetic Circuit → R1CS → QAP → zk-SNARK의 절차를 거쳐서 프로세스가 이뤄지는데 가장 먼저 유효성 함수(validity function)의 계산과정을 사칙연산 게이트 단위의 연산회로(Arithmetic Circuit)로 표현하고 이를 전체 수식에 대한 유효성검증을 위한 R1CS(Rank 1 Constraint System)형태로 변환한다. 변환된 R1CS 형태의 값은 전체 게이트에 대해서 유효성 검사를 실시해야 유효성 검사가 완료되기 때문에 특정 포인트에서 값을 대입하여 비교만 하면 증명할 수 있도록 QAP(Quadratic Arithmetic Program)형태로 변환하여 프로토콜을 수행하게 된다. 이는 기존 증명과정에 비해 훨씬 간결(succinct)해지는 특징을 지닌다.[3]

3.2 Identity Mixer

Identity Mixer는 IBM의 블록체인 플랫폼 Hyperledger Fabric에 사용되는 프라이버시 보호를 위한 암호 프로토콜이다[4]. Hyperledger Fabric에서는 사용자 인증 및 서명을 위해 X.509 인증서를 사용하였는데, Identity Mixer는 기존의 X.509 인증서를 사용할 시 발생하는 문제점을 해결하기 위해 도입되었다. 기존의 경우 CA(Certificate Authority)에 의해 발급되는 X.509 인증서에는 사용자의 개인정보가 그대로 드러나게 되는데, 이 정보를 통해 사용자가 블록체인 플랫폼 상에서 어떤 트랜잭션을 생성 또는 실행하였는지 다른 사용자들이 알 수 있게 된다. 이러한 문제점을 해결하기 위해 Identity Mixer에서는 영지식 증명 기반의 인증서를 사용한다. 이 인증서는 사용자의 개인정보

를 드러내지 않고도 사용자를 인증할 수 있고, 하나의 비밀키에 대해 여러 개의 공개키를 사용함으로써 사용자에 대한 프라이버시 보호를 제공한다. 결과적으로 Identity Mixer는 사용자에 대한 정보를 드러내지 않고 사용자를 인증하게 함으로써, 사용자가 블록체인 플랫폼에서 어떤 활동을 하는지 드러내지 않을 수 있게 한다.

3.3 Enigma

Enigma는 기존의 다른 블록체인 플랫폼에서 사용되던 스마트 컨트랙트에 프라이버시 보호를 위한 익명성을 부여한 블록체인 플랫폼이다 [5]. Enigma에서는 데이터의 프라이버시 보호를 위한 여러 가지 기법들을 사용하는데 이 중 대표적인 몇 가지를 소개하자면 첫 번째는 데이터를 여러 개의 노드에 나누어 저장하는 기법이다. 이 기법을 사용하면 노드는 자신이 보유한 데이터만으로는 데이터 원본을 알 수 없게 되고 분산 저장된 데이터는 일반적인 방법으로 접근할 수 없기 때문에 sMPC(secure Multi-Party Computation)를 통해 쿼리를 수행해야 한다. 두 번째는 스마트 컨트랙트에 익명성을 부여한 프라이버시 컨트랙트를 사용하여 end-to-end 분산 애플리케이션을 구현하는 방법인데 이를 이용하면 데이터 익명성 처리가 가능하다. 세 번째는 데이터 처리를 위한 Off-Chain이다. 이를 통해 데이터에 대한 암호화 수행 및 원본 데이터를 누출하지 않고 컨트랙트 코드를 수행할 수 있다.

IV. 결론

블록체인은 다양한 분야의 산업에서 관심을 가지고 활용되는 기술이다. 특히 금융권과 여러 IT 기업들이 블록체인 플랫폼을 통한 시스템을 구축하려는 움직임을 보이고 있다. 기존의 블록체인 시스템에서는 사용자의 의사와 무관하게 모든 데이터가 공개된다는 단점이 있었다. 때문에 블록체인 시스템에서 사용자의 선택에 따라 데이터를 보호할 수 있는 프로토콜의 필요성이 대두되었다. 본 논문에서는 블록체인 시스템이 이러한 필요성을 충족하기 위해 도입한 다양한

데이터 프라이버시 보호 프로토콜들을 소개하였다. 이러한 프로토콜들은 추후 블록체인 플랫폼을 활용하는 응용 프로그램들에서 사용자 데이터 프라이버시 보호를 위한 핵심적인 역할을 할 것이다.

【참고문헌】

- [1] Nakamoto Satoshi, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.
- [2] Eli Ben-Sassion, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer and Madars Virza, "Zerocash: Decentralized Anonymous Payments from Bitcoin(exended version)" May. 2014.
- [3] Zcash: Privacy-protecting digital currency. (2019). [online] Available at: <https://z.cash/technology/zksnarks/> [Accessed 8 Feb. 2019].
- [4] Hyperledger Fabric - Read the Docs. (2019). [online] Available at: <https://hyperledger-fabric.readthedocs.io/en/release-1.4/idemix.html> [Accessed 8 Feb. 2019].
- [5] Guy Zyskind, Oz Nathan and Alex 'Sandy' Pentland, "Enigma: Decentralized Computation Platform with Guaranteed Privacy".

탈중앙화 데이터 마켓플레이스의 접근제어 기법에 대한 연구

김혜빈*, 박지선*, 신상욱**

*부경대학교 대학원 정보보호학협동과정

**부경대학교 IT융합응용공학과

khbin1346@pukyong.ac.kr

A Study on Access Control Mechanism of Decentralized Data Marketplace

Hye-Bin Kim*, Jisun Park*, Sang Uk Shin**

*Interdisciplinary Program of Information Security, Graduated School,

Pukyong National University

**Dept. of IT Convergence and Application Eng.,

Pukyong National University

요약

다양한 종류의 마켓플레이스 플랫폼들 중 블록체인을 이용한 탈중앙화 데이터 마켓플레이스는 중앙 관리 주체가 존재하지 않고, 개인과 개인 간의 직접적인 거래, 스마트 계약을 통한 거래의 자동화 등 중앙 집중형 플랫폼이 수행하지 못했던 기능을 수행한다. 그러나 데이터에 대한 관리 주체가 명확하지 않고, 데이터 사용자의 신원이 분명하지 않아, 무분별한 접근과 사용에 따라 여러 문제가 생길 수 있다. 따라서 중앙 집중형 플랫폼에서 중앙 관리 주체가 수행하던 접근 제어 정책을 탈중앙화 데이터 마켓플레이스에서 어떠한 방법으로 수행할 것인지에 대한 연구가 필요하다. 본 논문에서는 활용될 수 있는 접근 제어 기법 몇 가지에 대해 논하고 분석한다.

I. 서론

다양한 종류의 데이터를 사고 팔 수 있는 데이터 마켓플레이스 플랫폼은 클라우드와 같은 기술들과 결합하여 현재까지 개발되어왔고 운영 중에 있다. 그리고 최근에는 기존 중앙 집중형 플랫폼의 단점을 보완하기 위해 중앙 관리 주체를 없애고 블록체인을 기반으로 하는 탈중앙화 데이터 마켓플레이스 플랫폼도 그 장점을 주목받고 있다. 기존 플랫폼과 마찬가지로 탈중앙화 플랫폼의 거래대상이 되는 데이터들의 종류는 다양하다. 그 중에는 개인 의료 데이터와 같이 프라이버시에 민감한 데이터도 존재한다. 이러한 데이터를 거래할 때에는 구매자와 판매자 모두가 신뢰할 수 있는 대상이어야 한다. 그

러나 탈중앙화 데이터 마켓플레이스는 각 사용자의 신원을 확보해줄 수 있는 중앙 주체가 없기 때문에, 거래 대상이 되는 데이터에 대하여 소유권이 명확하지 않거나, 무분별한 접근 및 사용으로 인한 프라이버시 침해 문제가 발생할 수 있다. 따라서 플랫폼의 장점을 살리면서 위와 같은 문제를 해결할 수 있는 방법에 대한 연구가 필요하다. 본 논문에서는 현재까지 제안되어왔던 블록체인 기반 시스템에서 활용 가능한 접근 제어 방법들 중 두 가지 기법에 대해 논하고 분석한다.

II. 관련 연구 : 블록체인과 데이터 마켓 플레이스

블록체인은 Satoshi Nakamoto에 의해 제안된 비트코인(Bitcoin)의 근간을 이루는 기술로서 수백 개의 트랜잭션들을 모아 하나의 블록으로 만든 후 체인 형태로 연결한 탈중앙화 데이터베이스(Decentralized Database)이다[1]. 전체 비트코인 시스템은 P2P 네트워크에 참여하는 노드들에 의한 합의 과정(Consensus Process)을 통해 유지·관리된다. 이로 인해 모든 노드가 동일한 내용의 블록체인을 보유하고 있으며, 블록이 체인에 한번 추가되면 그 내용을 바꿀 수 없다. 이전 블록 헤더의 해시를 다음 블록 헤더가 포함하고 있기 때문에 블록 내용이 변경되면 연결된 모든 블록의 내용이 바뀌고 모든 노드가 변경의 결과를 알 수 있다. 따라서 위·변조가 어렵고, 그로 인한 높은 안전성과 투명성을 특징으로 한다.

데이터 마켓 플레이스(Data Marketplace)는 참여자들로 하여금 데이터들을 거래 할 수 있도록 하는 플랫폼이다. 참여자는 데이터를 소유 또는 제공하는 주체인 데이터 소유자 또는 제공자(Data Provider), 데이터를 사용하고자 하는 주체인 데이터 소비자(Data Consumer), 데이터 재가공, 서비스 구축 및 제공 등의 역할을 하는 주체인 데이터 중개자(Data Intermediary)로 분류할 수 있다. 블록체인 기반 데이터 마켓 플레이스 플랫폼은 기존의 플랫폼이 제3자인 중앙 서버 또는 중개자로 인해 발생하는 수수료 문제, 정책과 관련된 지나친 의존성 문제 등을 해결하고, 데이터에 대한 거래 조건, 접근 기록 등을 스마트 계약을 이용해 블록체인 상에 기록함으로써 개인 간의 직접적이고 자동화된 거래를 할 수 있다.

III. 블록체인 기반 데이터 마켓 플레이스의 접근 제어기법

탈중앙화 데이터 마켓플레이스에서는 데이터를 거래하는 각 참여자들이 주체가 되어 데이터에 대한 접근 제어 정책을 수립해야 한다. 본 절에서는 블록체인 환경에서 사용될 수 있는 데이터에 대한 접근 제어기법 두 가지에 대해

논의한다.

3.1 스마트 계약을 이용한 제어 기법

첫 번째로 이더리움 블록체인(Ethereum Blockchain)등에서 사용되는 스마트 계약(Smart Contract)을 이용한 기법이다. 데이터 소유자가 자신의 데이터에 대한 접근 허용 사용자 리스트와 접근 권한 규칙을 내용으로 하는 계약을 생성하여 메타데이터와 함께 블록체인에 등록한다. 계약 내에는 계약을 배포한 데이터 소유자 노드의 주소가 포함되어 소유자만이 계약 상태를 수정할 수 있다. 그리고 외부 저장소에 실제 데이터와 그에 부합하는 생성한 계약의 주소가 저장된다. 스마트 계약은 데이터 소비자가 접근 권한을 요청하거나, 데이터 소유자가 접근 허용 리스트를 추가할 때, 또는 외부 저장소에서 최신 상태의 접근 허용리스트를 검색할 때 사용된다.

[2]에서 스마트 계약을 이용한 접근 제어 모델을 제안하고 있다. 여기서 데이터 소비자는 인증된 사용자로서 해당 데이터 관련 정책이 정의된 스마트 계약의 주소를 가지고 있고, 데이터 소유자가 생성한 Whitelist에 본인의 주소가 등록되어 있다. 데이터 소비자가 외부 저장소 제공자에게 데이터 접근 요청을 하면, 외부 저장소 제공자는 Random nonce값을 생성하여 소비자에게 전달한다. 소비자가 nonce값을 입력값으로 하는 트랜잭션을 계약의 주소로 전달하고, 블록체인을 스캔하고 있던 외부 저장소 제공자는 해당 트랜잭션을 보낸 노드의 주소가 기준에 데이터 소유자로부터 제공받은 최신상태의 접근 허용리스트인 Whitelist내에 존재한다면 데이터에 대한 접근을 허가한다. nonce값을 이용하는 이유는 최신 상태의 트랜잭션임을 증명하기 위함이다.

3.2 기존 접근 통제 모델을 이용한 제어 기법

접근제어를 위한 또 다른 기법으로는 기존에도 주체가 객체에 어떻게 접근하는지를 규정하기 위해 사용해왔던 접근 통제 모델(Access Control Model)이 있다. 이를 데이터 마켓 플레이스 플랫폼에 적용하기 위해서는 객체와 주체

각각의 특성과 그것들의 관계를 고려할 필요가 있다. 모델들 중 하나인 속성 기반 제어 기법(Attribute-based-Access-Control, ABAC)은 정의되어 있는 접근 정책을 이용하여 주체인 데이터 소비자의 속성과 객체인 데이터 속성간의 관계와 접근 요청이 발생하는 작업 또는 상황 문맥 정보 등을 바탕으로 데이터 소비자가 데이터에 대해 수행할 수 있는 권한을 결정한다[3]. 여기서 속성은 주체나 객체의 특성을 나타내는 값이다. 특징은 데이터 소유자가 접근하려는 소비자의 신원에 대한 사전지식을 필요로 하지 않는다는 점이다. 다른 기법으로는 역할 기반 제어 기법(Role-based-Access-Control, RBAC) 등이 있다.

Control Chain은 IoT 데이터에 대한 접근 제어를 위해 제안된 기법이며 상기된 접근 통제 기법을 적용시킨다. 해당 기법을 수행하기 위해 필요한 블록체인의 종류는 4가지이다. 각각은 수집된 센서 데이터와 가공된 데이터로부터 알아낸 상황의 문맥적인 정보가 저장된 Context, 접근 권한 허가·불허 정보를 저장하는 Accountability, 데이터와 해당 데이터를 사용하려고 하는 주체인 소비자들 사이의 관계에 대한 정의를 저장하는 Relationship, 그리고 데이터 소유자가 정의한 인증 규칙이 저장된 Rules 블록체인이다[4]. Relationship 블록체인에 등록된 관계 정보들을 RBAC나 ABAC를 비롯한 접근 통제 모델을 이용하여 그에 해당하는 인증 규칙을 Rules 블록체인에 저장한다. Accountability 블록체인은 Rules 블록체인에 등록된 인증 규칙을 바탕으로 접근 허용 정보를 저장한다. 이 정보는 네트워크의 노드들이 Context 블록체인에 저장된 상황 문맥정보와 함께 접근 여부를 판단할 때 사용될 수 있다.

IV. 결론

지금까지 블록체인 기반 데이터 마켓플레이스에서 사용될 수 있는 접근 제어 기법 중 두 가지로 스마트 계약에서 정의하는 기법과 접근 통제 모델을 이용한 기법, 그리고 적용 모델에 대해 논하였다. 스마트 계약을 적용한 모델에서

실제 데이터를 저장하고 있는 외부 저장소는 계약 내용이 아닌 주소만을 참조하고 있기 때문에, 소유자는 계약의 내용이 변경되더라도 외부 저장소에 저장되어있던 내용을 변경할 필요는 없다. 그러나 인증된 사용자를 직접 리스트에 추가하는 방식을 사용하기 때문에, 데이터를 요청하는 소비자가 많아질수록 신원을 파악해야하는 수도 늘어난다는 단점이 존재한다. 따라서 접근 제어 모델의 정책을 스마트 계약 내에 구현한 다음, 미리 블록체인에 저장된 데이터 소비자와 해당 데이터의 속성 그리고 측정된 환경조건 등을 입력 값으로 하는 트랜잭션을 생성하여 계약의 내용에 따라 접근 여부를 결정하는 방법을 제안해볼 수 있다. 이에 추가적으로 암호학적 기법을 사용하면 보안상의 안전성을 더 높일 수 있을 것이다.

[참고문헌]

- [1] Nakamoto, Satoshi. "Bitcoin: A peer-to-peer electronic cash system." 2008.
- [2] M. Laurent, et al. A blockchain-based access control scheme. SECRYPT 2018: 15th International Conference on Security and Cryptography, 2, pp.168 – 176, 2018
- [3] Hu, Vincent C., et al. "Guide to attribute based access control (ABAC) definition and considerations (draft)." NIST special publication 800.162 (2013).
- [4] Pinno, Otto Julio Ahlert, et al. "ControlChain: Blockchain as a Central Enabler for Access Control Authorizations in the IoT." GLOBECOM 2017–2017 IEEE Global Communications Conference. IEEE, 2017.

탈중앙화된 데이터 거래 플랫폼의 연구 동향 분석

노시완*, 이경현**

*부경대학교 일반대학원 정보보호학협동과정

**부경대학교 IT융합응용공학과

nosiwan@pukyong.ac.kr, khrhee@pknu.ac.kr

An Analysis on the Research Trend of the Decentralized Data Marketplace Platform

Siwan Noh* and Kyung-Hyune Rhee**

*Interdisciplinary Program of Information Security, Graduate School,
Pukyong National University.

**Department of IT Convergence and Application Engineering,
Pukyong National University.

요약

모바일 기술의 발전으로 과거에 비해서 개인이 대량의 데이터를 생산할 수 있게 되고 이러한 데이터가 높은 가치를 지니게 되면서 이러한 데이터를 거래할 수 있는 데이터마켓플레이스 플랫폼이 개발되었다. 하지만 신뢰할 수 없는 거래중개인을 배제하고 사용자간 직접적인 거래를 위해 블록체인 기술을 사용한 탈중앙화된 플랫폼이 현재 제안되어 해외에서는 테스트넷이 운영되고 있다. 국내에서는 아직 이러한 플랫폼 개발 연구가 미흡한데 본 논문에서는 국내 연구자들이 플랫폼 개발 연구에 있어 고려해야 할 보안요구사항의 분석과 해외 연구동향을 소개하여 연구방향을 제시하는 것을 목적으로 한다.

I. 서론

대한민국에서 개인정보(Personal Data)의 정의는 개인정보 보호법, 정보통신망 이용촉진 및 정보보호 등에 관한 법률 등 규정하고 있는 법률에 따라 가지는 의미가 다르지만 일반적으로 ‘살아있는 개인에 관한 정보로서 개인을 알아볼 수 있는 정보이며 해당 정보만으로는 특정 개인을 알아볼 수 없더라도 다른 정보와 쉽게 결합하여 알아볼 수 있는 정보’를 의미한다. 즉, 단순히 개인을 명확히 식별하는 정보(주민등록 번호, 이름 등)만이 아니라 개인을 식별하는데 사용될 수 있는 모든 정보를 개인정보로 정의할 수 있다.

종래의 공공기관이나 기업이 개인에 대한 정보를 생성하는 형태에서 벗어나 모바일 환경과 기술의 발전으로 오늘날의 사용자들은 일상 속에서 다양한 형태의 개인정보를 생성할 수 있

게 되었다. 유럽 소비자 위원회 위원인 Meglena Kuneva는 2009년 개인 정보는 새로운 석유와 같으며 디지털 세상의 새로운 화폐라고 말함으로써 현대 사회에서 개인정보가 매우 높은 가치를 지님을 시사했고 2025년에는 이러한 개인정보가 약 163제타바이트(ZB)에 이를 것으로 전망되고 있다.

해외에서는 이전부터 이러한 개인정보를 인공지능 에이전트에 학습시켜 새로운 서비스를 제공하는 시도가 이루어지고 있었다[1]. 하지만 Google이나 Amazon같은 대형 기업은 자신의 플랫폼을 활용해 서비스에 필요한 데이터를 수집할 수 있지만 일반적인 소규모 개발자들은 알고리즘은 개발하였으나 학습에 필요한 충분한 데이터를 얻기 어려운 문제가 있었다. 이런 문제를 해결하기 위해 일부 대형 기업이나 공공기관에서 무료로 데이터를 제공하고 있으나

[2-4] 데이터의 양이 충분하지 않고 데이터의 범주가 좁아 실제 활용하기 어렵다는 문제가 있었기 때문에 최근에는 블록체인 기반의 데이터 마켓플레이스 플랫폼이 제안되었다[5-7]. 마켓플레이스 플랫폼 상에서 사용자들은 데이터나 서비스(e.g., 다른 개발자의 알고리즘)를 거래할 수 있고 별도의 중개인을 통하지 않기 때문에 데이터를 보유한 개인이 직접 자신의 데이터를 거래하고 제어하는 것이 가능하다.

본 논문에서는 이러한 데이터마켓플레이스 플랫폼에 대한 연구동향과 플랫폼에서 고려해야 할 보안 요구사항에 대한 분석을 통해 데이터마켓플레이스 플랫폼 개발의 연구방향을 제시한다.

II. 연구 동향

이 장에서는 최근 발표된 블록체인 기반 데이터거래 플랫폼 중 다른 플랫폼과 다른 특징을 지닌 두 가지 플랫폼을 소개한다.

2.1 오션프로토콜(Ocean Protocol)

싱가폴의 비영리재단인 BigchainDB와 DEX Pte는 전 세계의 여러 데이터마켓에 존재하는 데이터들을 한 장소에서 거래할 수 있는 탈중앙화된 데이터 허브인 오션 프로토콜을 공동 개발하였다[6]. 데이터 판매자는 블록체인에 데이터 판매를 위한 스마트 컨트랙트를 등록하고 컨트랙트를 통해 거래를 수행하게 된다. 등록된 컨트랙트 주소로 구매자가 구매비용을 지불하면 판매자에게 비용이 바로 전달되는 것이 아니라 컨트랙트 주소에 구매비용이 묶이게 (locked)되고 구매자가 요청한 데이터&서비스를 제공하고 얻은 증명을 블록체인 네트워크의 검증들(verifiers)에게 검증받아야 컨트랙트에 묶여있던 구매비용을 대가로 수령할 수 있어 별도의 거래중개인의 참여없이 거래의 공정성을 보장한다.

또한 오션프로토콜은 데이터 신뢰성 문제를 해결하기 위해 큐레이션 마켓(Curation Market)의 개념을 적용하였다. 기존 데이터마켓플레이스 플랫폼에서 판매되는 데이터의 품질은 거래

중개인이 보장하였다. 한국데이터산업진흥원에서 운영하는 데이터스토어의 경우[2] 판매데이터에 대한 심사가 통과될 경우에만 실제 판매가 가능하도록 하고 있다. 하지만 별도의 관리기관이 없는 탈중앙화된 플랫폼에서는 이러한 데이터의 신뢰성을 보장할 중개인이 없어 절 낮은 데이터가 판매되어 플랫폼의 신뢰성을 저하시킬 수 있다. 오션프로토콜에서는 사용자가 자신이 보유한 화폐(ocean coin)를 특정한 데이터에 투자할 수 있도록 하여 투자한 데이터의 실제 판매(actual popularity)와 투자받은 금액(predicted popularity)에 따라 최종적으로 데이터 투자자들이 받는 보상이 다르도록 설계하여 신뢰성 있는 데이터에 대한 지표로 활용할 수 있도록 하고 있다.

2.2 코르텍스(Cortex)

이더리움 블록체인은 간단한 프로그램을 블록체인 상에 업로드하고 동작 결과에 대해 네트워크의 합의를 통해 투명한 프로그램 연산결과를 보장하는 스마트 컨트랙트를 특징으로 2세대 블록체인으로 불린다. 동일한 입력에 대해서는 항상 같은 연산결과를 내는 경우 단일 개체가 다른 목적으로 연산결과를 속이기 위해서는 네트워크의 모든 사용자로 하여금 연산결과가 자신과 동일하도록 만들어야하므로 투명한 연산결과를 보장받을 수 있다는 것을 전제로 하는데 코르텍스는 스마트 컨트랙트에 인공지능 모델을 포함시켜 누구나 비용만 지불하면 원하는 인공지능 서비스를 제공받을 수 있도록 해주는 AI Dapps 플랫폼이다[7].

개인 CPU에서 동작하는 이더리움 스마트 컨트랙트와 달리 GPU/FPGA를 사용하여 복잡한 연산이 필요한 인공지능 모델이 블록체인 상에서 동작할 수 있도록 지원하고 사용자는 플랫폼에 게시된 AI 컨트랙트를 호출함으로써 네트워크 전체 검증자의 AI 추론결과의 합의를 통해 투명한 인공지능 서비스를 제공받을 수 있다.

III. 핵심 요소 기술 분석

2장에서는 데이터마켓플레이스 플랫폼의 예시로 오션프로토콜과 코르텍스를 소개했다. 이장에서는 데이터마켓플레이스 플랫폼 개발을 위한 보안요구사항을 서술하고 요구사항을 만족하기 위한 솔루션들을 소개한다.

3.1 상호운용성(Interoperability)

대부분의 데이터마켓플레이스 플랫폼은 개별적인 블록체인 네트워크 위에서 동작하고 있는데 일반적인 블록체인 네트워크는 폐쇄적인 환경이라 다른 블록체인 네트워크와 통신할 수 없다. 즉, 원하는 데이터를 구매하기 위해서는 해당 플랫폼에 참여해야하는데 보통 플랫폼별로 별도의 화폐를 사용하기에 매번 필요한 데이터가 존재하는 플랫폼의 화폐를 구입하는 것은 매우 비효율적이다. 이를 위해 현재 블록체인 네트워크 사이의 상호운용성을 보장하기 위한 연구가 제안되었다[8-9].

상호운용성 보장기술은 연결된 모든 블록체인 네트워크의 상태(state)를 기록하는 블록체인 허브(Hub) 네트워크 구현을 기반으로 한다. 이를 통해 한 블록체인에 기록된 내용을 다른 블록체인으로 전달하는 것이 가능하고 오션프로토콜과 같은 통합 거래 플랫폼으로 본래 네트워크의 정보를 전달하여 거래하는 것이 가능하다.

3.2 증명 가능한 거래

탈중앙화된 플랫폼에서의 거래는 거래중개인을 통하지 않고 사용자간 직접 거래를 사용한다. 하지만 이 경우 판매자가 요청받은 데이터 혹은 서비스를 제공하였는지에 대한 보장이 없기 때문에 이러한 거래에 대한 증명을 네트워크에서 검증하여 거래가 정확하게 이루어졌을 경우에만 판매자에게 구매비용을 제공하는 방식이 필요하다.

현재 판매하는 대상은 특정한 데이터 혹은 서비스(알고리즘)인데 데이터의 경우 구매자로 하여금 데이터를 제공받았다는 증명으로 실제로 데이터를 보이지 않고 검증할 수 있는 Proof-of-Storage 기법이 제안되었다[10]. 서비스

의 경우 영지식 증명(zero-knowledge proof)[11]을 사용하여 실제 연산에 사용된 데이터나 알고리즘을 노출하지 않고 검증하는 방법이 제안되었다. 이러한 거래의 증거를 기반으로 거래의 완료를 검증하여 검증된 판매자에게만 거래 대가를 지급하게 된다.

3.3 프라이버시

마켓플레이스 플랫폼의 기본적인 목적은 인공지능 모델의 학습을 위한 데이터의 확보이나 이 과정에서 제공되는 데이터에 포함된 개인정보의 노출로 인한 문제가 발생할 수 있다. 때문에 실제로 데이터(raw data)를 제공하지 않고 인공지능 모델이 학습할 수 있도록 하는 방법이 제안되었는데 암호화된 데이터를 사용하여 학습하도록 하는 동형암호(Homomorphic encryption)[12]와 실제 학습에 사용된 데이터를 노출하지 않고 학습결과만을 제공하는 다자간 계산 프로토콜(Multi-Party Computation)[13]을 사용한 방법이 있다.

IV. 결론

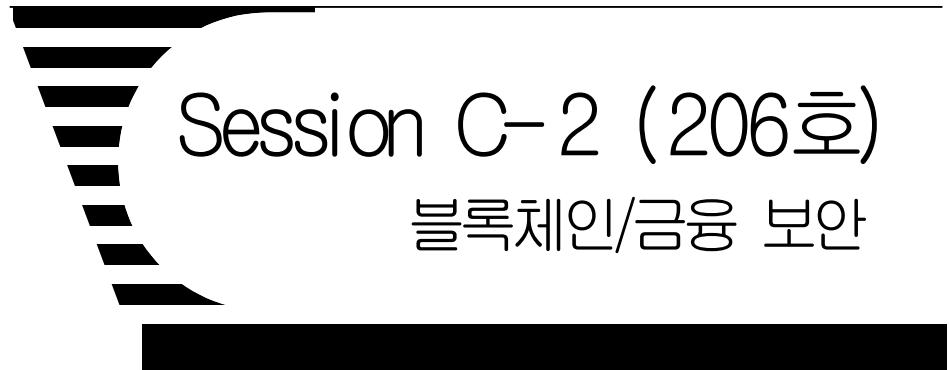
본 논문에서는 정보화시대에서 개인이 생산하는 개인정보의 가치와 이러한 데이터를 거래하는 플랫폼인 데이터마켓플레이스 플랫폼의 연구동향을 소개하고 플랫폼 구현에서 필요한 보안요구사항과 이를 해결하기 위한 알려진 솔루션을 살펴보았다. 불과 얼마 전까지 개인이 생산하는 개인정보의 범주는 매우 협소하였으나 기술의 발전으로 생산하는 데이터의 범주가 매우 넓어졌고 그 가치 또한 증가하여 이를 거래하는 플랫폼이 대두될 것으로 기대되는 현재 상황에서 개인의 프라이버시를 보장하고 거래의 공정성을 보장하는 기술의 연구가 필요하다고 생각하며 본 논문이 추후 이런 기술의 연구에 방향을 제시해줄 수 있을 것으로 기대한다.

[Acknowledgement]

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. NRF-2018R1D1A1B07048944)

[참고문헌]

- [1] P. Vepakomma, O. Gupta, T. Swedish, and R. Raskar, “Split learning for health: Distributed deep learning without sharing raw patient data,” arXiv preprint arXiv:1812.00564, 2018.
- [2] “DATA STORE.” [Online]. Available: <https://www.datastore.or.kr/>. [Accessed: 01–Feb–2019].
- [3] “Registry of Open Data on AWS.” [Online]. Available: <https://registry.opendata.aws/>. [Accessed: 01–Feb–2019].
- [4] “BigQuery Public Datasets | BigQuery,” Google Cloud. [Online]. Available: <https://cloud.google.com/bigquery/public-data/>. [Accessed: 01–Feb–2019].
- [5] B. Goertzel, S. Giacomelli, D. Hanson, C. Pennachin, and M. Argentieri, “SingularityNET: A decentralized, open market and inter-network for AIs,” Whitepaper, 2017.
- [6] Ocean protocol foundation, Ocean Protocol: A Decentralized Substrate for AI Data & Services, Whitepaper, 2018.
- [7] Z. Chen, X. Yan, W. Wang, and J. Tian, “Cortex – AI on Blockchain,” Whitepaper, 2017.
- [8] E. Buchman and J. Kwon. “Cosmos: A network of distributed ledgers.” Whitepaper, 2016.
- [9] G. Wood, “Polkadot: Vision for a heterogeneous multi-chain framework,” Whitepaper, 2016.
- [10] G. Ateniese et al., “Provable data possession at untrusted stores,” in Proceedings of the 14th ACM conference on Computer and communications security, 2007, pp. 598 - 609.
- [11] S. Goldwasser, S. Micali, and C. Rackoff, “The knowledge complexity of interactive proof systems,” SIAM Journal on computing, vol. 18, no. 1, pp.186 - 208, 1989.
- [12] C. Gentry, “Fully homomorphic encryption using ideal lattices,” Proceedings of the 41st ACM Symposium on Theory of Computing, pp.169–178, 2009.
- [13] Y. Lindell, “Secure multiparty computation for privacy preserving data mining,” in Encyclopedia of Data Warehousing and Mining, IGI Global, 2005, pp. 1005 - 1009.



좌장 : 이종혁 (상명대)

식품 추적 시스템에서의 블록체인 솔루션

초느에진랏*, 이경현**

*부경대학교 정보보호학협동과정

**부경대학교 IT융합응용공학과

chonwe1612@gmail.com

Blockchain solutions in food traceability system

Cho Nwe Zin Latt*, Kyung-Hyune Rhee**

*Interdisciplinary Program of Information Security, Graduate School,

Pukyong National University

**Department of IT Convergence and Application Engineering

Pukyong National University

요약

Blockchain, a technology that is known for helping many industries that face trust issues, can help the food industry address its problems too. With blockchain, every stakeholder who is involved in the industry can access relevant details about a food product. This blockchain can help food producers and retailers in tracking the food supply chain to know more information about safety, provenance, and authentication thereby helping them rebuild the trust of consumers. One way of solving traceability issues and ensuring transparency is by using blockchain technology to store data from the chemical analysis in chronological order so that they are impossible to manipulate eventually.

I. Introduction

The Bitcion is the first application of blockchain, it's a kind of digital currency based on blockchain [1] technologies, using for trade things on the Internet like money as we do in the real work. Because of the success of Bitcoin, people now can utilize blockchain technologies in the field and service, such as financial market, IoT, supply chain, voting, medical treatment, food industry and storage. The blockchain technologies composed of six key elements.

Blockchain[2] enables end-to-end traceability by bringing a common technological language to the food chain while allowing consumers to access the storage of food on their label through their

phone. This has raised the need to trace products through the complex supply chain from retail back to the farm: to trace an outbreak, to verify that a product is kosher, organic or allergen-free, or simply to assure transparency to consumer. When applied to their food supply chain[3], digital product information such as farm organization details, batch numbers, factory and processing data, expiry dates, storage temperatures, and shipping details are digitally connected to food items and their information such as farm origination details, batch numbers, factory and processing data, expiry dates, storage temperature and shipping details are digitally connected to food items and their information is entered into the blockchain

each step of the process. Blockchain technology will enable consumers to make more informed decisions when purchasing groceries, reduce food waste and has the potential to halt the spread of illness by contamination, and save millions of lives. A blockchain is a publicly shared, transparent, decentralized ledger for recording the history of transactions within a system. Data can only be added to the ledger, the historical data is unalterable, and the integrity of the data is achieved by consensus among distributed parties, rather than a central administrator. The verification process is randomized which means that no one participant can force a particular entry onto the blockchain. The paper is organized as followed: in section 2, 3 things that will change the world today. In section 3 food supply chain 2.0 and the section 4 is verifiability and section 5 is conclusion about this paper.

II. 3 Things that will change the world today

This will revolutionize both the pace at which contaminated food can be recovered and the trust that consumers place in the food they purchase. The potency of this technology in food production was corroborated recently in a presentation by Frank Yiannas, vice president of food safety at Walmart[4], at the annual shareholder conference. Technology to more effectively manage their supply chains Using conventional methods, it took over six days to reveal the origin of a package of sliced mango. The same result was achieved in just 2.2 seconds using blockchain technology. The greater efficiency in detecting food contamination will enable distributors to mitigate consequences of abnormalities by taking the appropriate preventative measures. In swiftly handling issues, food safety will

increase for the consumer, and it will reduce the likelihood of a producer's reputation being tarnished by a bad batch.

Food traceability is set to soon become a standard, and it is only a matter of time before other industries follow suit and embrace blockchain and enable their consumers to see what is required to produce their goods.

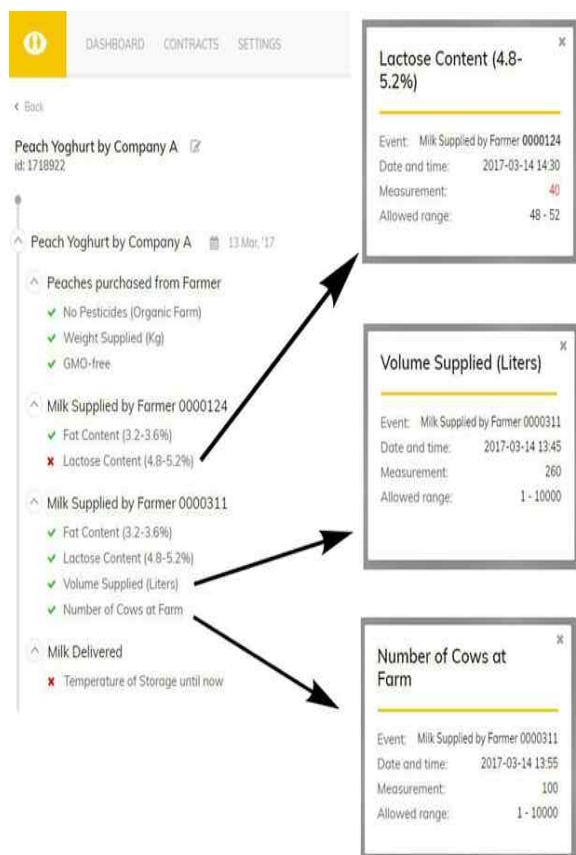
III. Food Supply Chain 2.0

Soon after the publication, we have received another press release about a similar project. It is called Food Blockchain XYZ, a new tool that aims at providing "a reliable and comprehensive overview of all aspects of quality, safety and tracing of food from farm to fork, and to seamlessly manage commercial relationships between different actors of the supply chain." To do so, Food Blockchain XYZ decided to use the so-called Food Supply Chain 2.0 [6], which combines sensor systems, Blockchain and smart contracts. They also developed a specific token, Foodcoin, that allows worldwide transactions with no fees and that is not subjected to exchange rate fluctuations. It is cryptographically secure and compatible with the Ethereum [5] Blockchain and "it permits new risk-free, flexible and creative commercial relations based on smart contracts," according to their official press release.

IV. Verifiability

From an economic point of view, Food Blockchain XYZ can be useful both for consumers and producers because the first can benefit from a higher transparency and better knowledge about food, and the second can increase their prices for their verifiably better quality products. For example, Food

Blockchain XYZ [7] can allow a company producing yogurt to create a risk-free agreement with suppliers of milk. This company would have high requirements for the quality of milk, so it might require that the composition of milk is at appropriate levels. It sources milk from two farmers; the first one provides a too low percentage of lactose in milk, while the second passes all the requirements and the company decides to order the milk from the second farmer.



[Fig. 1 Blockchain XYZ]

V. Conclusion

In this report we described about assuring food traceability with blockchain technology looks promising, there remain some limits to be considered. However, many challenges in implementation are still to be overcome, such as how to translate posted information in the system and whether companies are willing to take the leap into transparent operations. In

turns out that bitcoin is not the only application for blockchain technology. It many prove useful in a number of other areas such as financial services, personalized health and food provenance.

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. NRF-2018R1D1A1B07048944)

[참고문헌]

- [1] J. Michael, A. cohn, and J.R. BUTCHER, "Blockchain Technology," The Journal, Feb. 2018.
- [2] M. Iansiti, and K.R. Lakhani, "The Truth about Blockchain," Harvard Business Review, vol. 95, no. 1, pp. 118-127, 2017.
- [3] B. Bigliardi and E. Bottani, "Performance Measurement in The Food Supply Chain: a Balanced Corecard Approach". Facilities, vol. 28, no. 5/6, pp. 249-260, 2010.
- [4] S.P.D Sell, "Introduction to supply chain management," 1999.
- [5] N. Atzei, M. Bartoletti, and T. Cimoli, "A Survey of Attacks on Ethereum Smart Contracts (sok)," In Principles of Security and Trust. Springer, pp. 164-186, 2017.
- [6] J.F. Galvez, J.C.. Mejuto, and J. Simal-Gandara, "Future Challenges on the Use of Blockchain for Food Traceability Analysis," TrAC Trends in Analytical Chemistry, 2018.
- [7] X. Li, P.J. iang. T. Luo and Q. Wen. "A Survey on The Security of Blockchain Systems," Future Generation Computer Systems. 2017.
- [8] D. Brandon "The Blockchain: The Future of Business Information Systems," International Journal of the Academic Business World vol. 10, no. 2, pp. 33-40, 2016

크립토재킹 공격 유형 및 동향 분석

김의진*, 김득훈**, 곽진***

*:***아주대학교 사이버보안학과

**정보보호응용및보증연구실, 아주대학교 컴퓨터공학과

*dmlwls0403@ajou.ac.kr, **dhkim.isaa@gmail.com, ***security@ajou.ac.kr

Types and Trends Analysis of Cryptojacking Attack

Eui-Jin Kim*, Deuk-Hun Kim**, Jin Kwak***

*:***Department of Cyber Security, Ajou University

**ISAA Lab., Department of Computer Engineering, Ajou University

요약

블록체인은 분산원장 플랫폼을 이용해 익명성과 거래의 무결성을 보장하는 기술이다. 블록체인기술이 주목받으면서 이로부터 생성되는 가치가 존재하는 암호 화폐에 대한 관심도 증가하였다. 암호 화폐의 가치가 높아짐에 따라 암호 화폐 채굴 방식도 다양해졌으며, 이 중에서도 브라우저 기반 암호 화폐 채굴 방식이 큰 주목을 받고 있다. 그러나 브라우저 기반 채굴의 취약점을 악용한 크립토재킹(Cryptojacking)이 등장하면서 공격자는 실질적 이익을 볼 수 있게 되었다. 이를 대비하기 위해 본 논문에서는 대표적인 크립토재킹 공격 유형인 피싱, Malvertisement, 파일 공유 사이트를 이용한 공격 유형 및 동향 분석과 이에 대한 대응방안을 제시한다.

I. 서론

블록체인은 분산원장 플랫폼을 이용하여 익명성과 거래의 무결성을 보장함에 따라 향상된 보안성을 제공하는 기술이다[1]. 블록체인에 의해 획득 가능한 암호 화폐에 대한 가치가 높아지면서, 암호 화폐와 더불어 암호 화폐 채굴에 대한 관심이 증가하고 있다.

암호 화폐 채굴에는 CPU(Central Processing Unit), GPU(Graphics Processing Unit), ASIC(Application Specific Integrated Circuit)등의 채굴 방식이 있다. CPU와 GPU 채굴 방식은 전력 소모량에 비해 낮은 채굴 효율성을 가지며, ASIC는 주문형 반도체를 이용하여 CPU와 GPU 채굴 방식보다 빠르지만, 가격이 비싸다는 단점이 있다[2].

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. NRF-2017 R1E1A1A01075110).

초기에는 브라우저 기반 암호 화폐 채굴 방식이 존재하였으나 암호 화폐의 가치가 낮아 주목받지 못하였다. 그러나 암호 화폐 가치 상승으로 브라우저 기반 암호 화폐 채굴 방식이 재주목을 받게 되었다[3]. 브라우저 기반 암호 화폐 채굴 방식인 Coinhive는 웹 사이트를 통해 사용자의 PC 자원을 사용자에게 허가받은 후 채굴을 진행한다. 웹 관리자들은 소규모의 스크립트를 추가함으로써 웹 사이트에 암호 화폐를 채굴 할 수 있다.

그러나 브라우저 기반 채굴 방식의 취약점을 이용한 악의적인 공격이 발생하고 있으며 이를 크립토재킹(Cryptojacking)이라 한다[3].

본 논문에서는, 2장에서 브라우저 기반 암호 화폐 채굴 사이트인 Coinhive와 이에 대한 취약점을 악용한 크립토재킹에 대해 설명하고, 3장에서 크립토재킹 공격 유형과 동향을 분석한 뒤 4장에서 이에 대응하기 위한 브라우저 기반 애플리케이션을 분석하고, 마지막으로 결론을 맺는다.

II. 관련 연구

2.1 Coinhive

2017년 9월 브라우저 기반 암호 화폐 채굴 사이트인 Coinhive가 등장하였다. Coinhive는 CPU를 기반으로 한 CryptoNight 알고리즘으로 만들어진 모네로라는 암호 화폐를 채굴한다[4].

Coinhive는 웹 사이트 수익 창출을 위한 광고의 대안으로 제시되었다. 해당 과정은 웹 사이트에 소규모의 스크립트만 추가하면 되며, 이는 사용자의 컴퓨팅 파워를 이용해 해시값을 알아내고 암호 화폐를 지급한다[5].

[그림 1]은 Coinhive를 통해 웹 사이트 채굴을 위한 코드를 나타낸다.

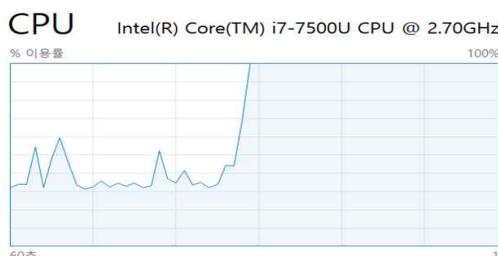
사용자의 컴퓨팅 파워를 이용하기 위해서는 사용자의 자원 활용 허가를 먼저 받아야 한다. 그러나 공격자에 의한 다양한 크립토재킹 공격을 통해서 사용자의 컴퓨팅 파워를 악용하는 공격이 가능하다는 문제가 존재한다.

```
<script src = "https://authedmine.com/lib/authedmine.min.js"></script>
<script>
var miner = new CoinHive.Anonymous('<YOUR PUBLIC SITE KEY>', {throttle: 0.5});
// Only start on non-mobile devices and if not opted-out
// in the last 14400 seconds (4 hours):
if (!miner.isMobile() && !miner.didOptOut(14400)){
    miner.start();
}
</script>
```

[그림 80] Coinhive 스크립트

2.2 크립토재킹

크립토재킹이란 암호 화폐(CryptoCurrency)와 하이재킹(Hijacking)의 합성어로 사용자의 허가 없이 컴퓨팅 파워를 통해 암호 화폐를 채굴하여 공격자의 지갑으로 보내는 것이다. 사용자의 PC를 대상으로 수행되는 공격이므로 CPU 채굴 또는 GPU 채굴을 활용하여 수행된다.



[그림 81] 크립토재킹에 의한 개인 PC 사용량

크립토재킹은 사용자의 자원을 사용하는 것

외에는 다른 피해가 없는, 채굴을 위한 마이너(Miner) 악성코드로 분류된다. 그러나 크립토재킹 공격은 장기적으로 사용자 기기의 수명 및 성능 저하의 원인이 된다. 최근에는 개인 PC의 고성능화로 인하여 개인 PC를 목표로 한 공격이 이뤄지고 있으며, 기업용 PC 또는 산업 기반시설을 목표로 한 크립토재킹 공격 또한 증가하고 있다. 기업의 서버와 같이 고사양의 시스템이 암호 화폐 채굴에 효과적이기 때문인데, 이는 기업 전체·산업 기반시설 및 국가 경제에 악영향을 줄 수 있다[6].

III. 크립토재킹 공격 유형 및 동향 분석

3.1 피싱과 악성 메일

크립토재킹의 주요 공격 방법 중 하나인 피싱(Phising)은 불법적인 경로를 통해 얻은 사용자의 개인정보 또는 이메일을 이용하는 방법으로 정상 파일로 위장한 악성 메일 혹은 지인으로 위장한다. 이후 파일을 다운로드하면 암호 채굴 스크립트 또는 프로그램이 사용자의 허가 없이 컴퓨터에 다운로드되어 사용자의 자원을 악용하는 크립토재킹이 수행된다.

3.2 Malvertisement

Malvertisement는 악성코드(Malware)와 광고(Advertisement)의 합성어이다. 이러한 악성광고는 사용자가 볼 수 없는 아이프레임(iframe)을 통해 취약점 공격사이트로 재접속하는데, 이 때 Exploit Kit이 사용된다. Exploit Kit은 사용자 PC의 취약점을 분석하며, 사용자의 PC에 취약점 패치가 되어 있지 않다면 취약점을 이용해 악성코드를 다운로드할 수 있는 셀 코드(Shell Code)가 실행된다[7].

[표 1]은 Exploit Kit을 통한 크립토재킹에 악용되는 취약점이다[6].

[표 47] 크립토재킹에 악용되는 취약점

취약점 유형	설명
CVE-2018-8174	VBScript 원격코드 실행 취약점
CVE-2016-0189	JScript와 VBScript 원격코드 취약점
CVE-2018-4878	허상포인터(Dangling Pointer) 취약점

3.3 파일 공유 사이트

파일 공유 사이트는 P2P 프로토콜을 사용하여 사용자들끼리 파일을 주고받을 수 있다. 파일 공유 사이트의 증가로 인해 해커들의 악의적인 파일 유통도 증가하였다. 예를 들어 게임 크랙으로 위장한 악성코드의 경우 백신 프로그램이 크랙을 악성코드로 인식하는 점을 이용해 백신 프로그램 종료를 유도한다[8]. 이와 같은 상황을 악용하여 공격자는 사용자 PC에 쉽게 접근하여 정상 파일로 위장한 악성코드를 사용자 PC에 설치한 후 크립토재킹을 수행한다.

3.4 크립토재킹 동향

IoT 기기의 증가에 따라 크립토재킹의 대상이 PC에서 IoT 기기로 향하고 있다. 인기 유료 앱에 악성코드를 심은 뒤 불법적 경로로 무료 배포를 하는 수법을 사용되는데, 무료 배포된 앱 안의 공유된 APK파일을 설치할 경우 크립토재킹 공격을 받기 쉬우며 앱 실행 시 정상 작동하는 것처럼 보이나 백그라운드에서 암호 화폐를 채굴한다.

공격자들은 IoT 기기의 자원을 적정량만 사용하여 사용자들이 인지하지 못하게 접근하는 새로운 방법을 제시하고 있다. 또한 충전할 때에만 암호 화폐를 채굴하거나 모바일 환경에서 악성 프로그램 설치 시 와이파이를 이용하여 데이터 사용량을 최소화하는 방법을 이용하는 등 공격이 지능적으로 발전하고 있으므로 이에 대한 대응방안이 필요하다.

IV. 대응방안 분석

4.1 NoCoin 기술

NoCoin은 암호 화폐 채굴 웹 사이트가 blacklist.txt에 저장된 블랙리스트 기반 애플리케이션이다. NoCoin은 인터넷 익스플로러나 크롬에서 확장 프로그램으로 이용이 가능하며, 현재 버전 0.4.14 기준으로 사용자가 직접 자신의 화이트리스트를 추가할 수 있다[9].

4.2 Miner Block 기술

Miner Block 기술은 블랙리스트 기반

애플리케이션이다. Miner Block은 버전 0.4.0을 기준으로 사용자가 직접 추가할 수 있으며 특정 경로를 필터링 할 수 있는 화이트리스트 기능이 추가되었다[10].

V. 결론

본 논문에서는 Coinhive의 등장으로 웹 브라우저 기반 암호 화폐 채굴이 알려졌고, 이를 통한 암호 화폐 채굴 시 취약점을 악용하는 크립토재킹에 대한 동향을 분석하였다. 최근에는 모바일을 기반으로 사용자가 인지하지 못하는 새로운 방법을 이용하는 지능적인 크립토재킹이 등장하고 있으며, 개인 PC뿐 아니라 IoT 기기도 위협 대상이 되고 있다. 따라서 웹 브라우저 기반 대응방안뿐만 아니라 모바일, IoT 기기에도 적용할 수 있는 선제적 대응방안이 필요하다.

[참고문헌]

- [01] Satoshi Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System”, 2008.
- [02] 최원석, 김형식, 이대화, “크립토재킹 연구 동향”, 정보보호학회지 28권 3호, Jun 2018.
- [03] Symantec , “Cryptojacking : A Modern Cash Cow”, Symantec ISTR, Sep 2018.
- [04] 고동현, 정인혁, 최석환, 최윤호, “크립토재킹 사이트 탐지를 위한 동적 분석 프레임워크”, 정보보호학회논문지 28권 4호, Aug 2018.
- [05] Coinhive, “Coinhive monetize your business with your users cpu power”. <https://coinhive.com/>, 2017.
- [06] Ahnlab, “월간 安 201810호”, Ahnlab, Oct 2018.
- [07] Trend Micro, “Rig Exploit Kit Now Using CVE-2018-8174 to Deliver Monero Miner”, 2018.
- [08] Ahnlab, “월간 安 201809호”, Ahnlab, Sep 2018.
- [09] Keramidas, “R. NoCoin Browser Extension”, <https://github.com/keraf/NoCoin>
- [10] Belkacim, “I. MinerBlock Browser Extension”, <https://github.com/xd4rker/MinerBlock>

수정·삭제를 위한 Hybrid Blockchain 기반의 XGS (XOR Global State) 인젝션 기술에 관한 연구⁺)

라경진*, 이임영**

순천향대학교 컴퓨터학과*,**

A Study on Hybrid Blockchain-based XGS (XOR Global State) Injection Technology for Modification and Deletion

Gyeong-Jin Ra*, Im-Yeong Lee**

Dept of Computer Science and Engineering, Soonchunhyang University***

요약

블록체인은 추가전용(Append-only) 분산 원장으로써, 임의의 수정·삭제를 불가하게 하여 전체 시스템의 무결성과 신뢰를 제공하는 데이터베이스 기술이다. 즉, 블록체인은 수정과 삭제가 아닌, 정당한 사용자의 접근권한에 따라 데이터의 읽기와 쓰기가 가능한 CRAB(Create-Retrieve-Append-Burn)방식이다. 하지만 이는 한번 생성된 데이터의 삭제가 불가하여 Privacy 침해 등과 같은 문제가 발생한다. 본 논문은 이와 같은 문제를 해결하기 위해 별도의 Storage 을 두어 데이터를 별도로 저장하고 연결이력을 Blockchain에 기록하는 On-Off Blockchain 방식의 Hybrid Blockchain 시스템을 적용한다. 또한 원장 기록과 별도로 최신 상태(State)를 분산데이터베이스로 하여 변경 가능하게하고, 임의의 인젝션을 XOR 형태로 발생시켜 State를 변경함에 따라 Off Blockchain의 수정·삭제이력을 전체 시스템의 변경 없이 조회하도록 제안한다.

I. 서론

최근 비트코인의 등장과 함께 블록체인은 추가전용(Append-only) 분산 원장으로써, 임의의 수정·삭제를 불가하게 하여 전체 시스템의 무결성과 신뢰를 제공하는 데이터베이스 기술이다. 즉, 블록체인은 수정과 삭제가 아닌, 정당한 사용자의 접근권한에 따라 데이터의 읽기와 쓰기가 가능한 CRAB(Create-Retrieve-Append-Burn)방식이다[1]. 하지만 이는 한번 생성된 데이터의 삭제가 불가하여 Privacy 침해 등과 같은 문제가 발생한다. 이를 위한 기존 방식인 Rewritable Blockchain[2]은 카멜레온해시로 충돌쌍을 통해 완전한 수정·삭제를 하여 원본데이터

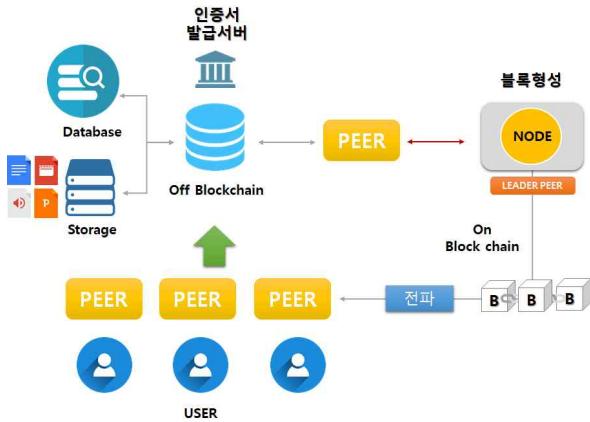
삭제가 이루어지므로 지속적인 이력 추적불가와 원본데이터 위조의 혼란을 야기한다. 본 논문은 이와 같은 문제를 해결하기 위해 별도의 Storage 을 두어 데이터를 별도로 저장하고 연결이력을 Blockchain에 기록하는 On-Off Blockchain 방식의 Hybrid Blockchain 시스템을 적용한다. 또한 원장 기록과 별도로 최신 상태(State)를 분산데이터베이스로 하여 변경 가능하게하고, 임의의 인젝션을 XOR 형태로 발생시켜 State를 변경함에 따라 Off Blockchain의 수정·삭제이력을 전체 시스템의 변경 없이 조회하도록 제안한다.

II. 관련연구

2.1 Hybrid Blockchain

기존 블록체인은 데이터를 하나의 트랜잭션으로 하여 평문 혹은 암호화 형태로 블록체인

+ 본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터육성 지원사업의 연구결과로 수행되었음
(IITP-2018-2015-0-00403)



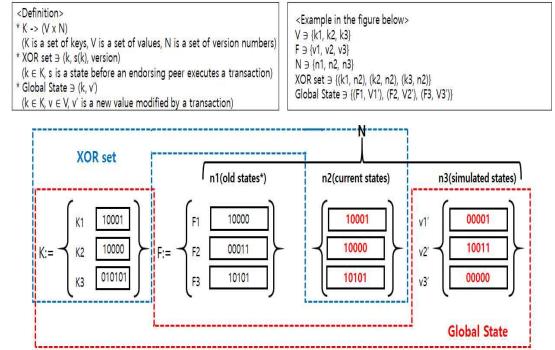
(그림 1) Hybrid Blockchain

에 기록하고 이를 모두 소지하는 분산원장의 형태를 가지다. 하지만 모든 사람들에게 공개하고, 이를 수정·삭제가 불가능함에 따라 Privacy 문제가 발생한다. 이를 위해 별도의 Storage를 두고 데이터의 원본을 저장하는 Off Blockchain과, 연결된 메타데이터 값만을 블록체인에 기록하여 분산원장으로 소지하는 On Blockchain을 결합한 Hybrid Blockchain 시스템이 제안되었다[3].

2.2 Rewritable Blockchain

미국 Accenture사의 Rewritable Blokchain은 카멜레온해시를 통해 해시의 충돌쌍을 찾는 것은 매우 어려우나, 카멜레온해시는 백도어(Back Door)성질을 이용하여, 사용자의 정당한 개인키를 통해서는 쉽게 충돌쌍을 찾도록 한다[2]. 이를 통해 트랜잭션의 내용은 변하나 해시값은 변동되지 않으므로 전체 블록체인 시스템은 유지되면서 정당한 사용자에 의한 수정·삭제가 가능하다. 하지만 원본데이터가 사라짐에 따라 이전 데이터의 접근이 불가하고 데이터 수정이력을 지속적으로 추적하기 어렵다. 또한 빈번히 이루어질 경우 PoW 기반의 Rewritable Blokchain의 전체 시스템 동기화는 매우 동작이 느려지는 단점이 존재한다.

2.3 XOR Global State(XGS)

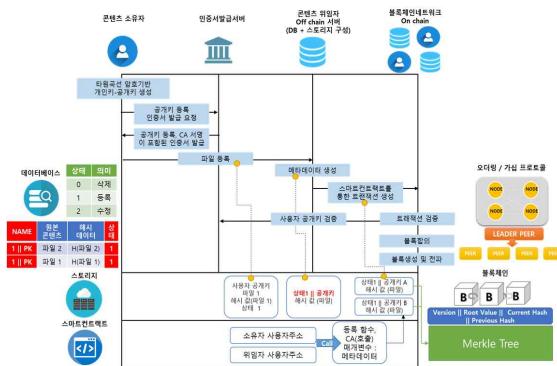


(그림 2) XGS 인제션 기술

아더리움과 하이퍼렛저는 최근에 일어난 거래에 따른 최신 State를 블록체인의 수정불가한 원장과 함께 분산데이터베이스 형태로 기록한다. 따라서 블록체인의 완전한 수정·삭제의 불가능한 성질을 최신 State의 변경에 따라 기존의 블록체인에 포함된 트랜잭션이 수정·삭제가 이루어진 것처럼 한다[3]. 본 논문에서는 이러한 성질을 통해 XOR 기반의 데이터를 Overriding 하여 수정·삭제를 하고 On Blockchain의 State를 변경하도록 임의의 트랜잭션 인제션을 주입한다. XGS의 분산데이터베이스는 (그림 2)과 같이 구성되어 있다. XOR set에 따라 Key-Value로 구성되어 있으며 수정 혹은 삭제 할 데이터의 변경된 값 만큼은 Global State에 인제션 되어 Off Blockchain의 Global State를 인위로 변경한다. 이를 통한 Linking 값을 포함한 On Blockchain이 생성되고 수정된 파일의 Linking 값의 완전한 On Blockchain의 Tracking은 수정되기 이전의 참조 값으로 Tracking이 가능하다.

III. 제안방식

본 논문의 제안방식은 컨소시엄 블록체인 기반의 데이터 등록, 수정, 삭제, 추적과정을 가진다(그림 3). 모든 과정의 사용자는 인증서를 통한 사용자 인증단계를 포함한다. 등록과정은 Off Blockchain과 On Blockchain의 Linking 단계와 서버의 On Blockchain 기록생성을 포함하고 수정, 삭제는 XGS를 통한 인제션 과정을 포함한다.



(그림 3) 전체 시나리오 동작 방식

1. 데이터 등록 단계

이 단계에서는 사용자가 Off Blockchain의 별도의 Storage에 데이터를 등록하는 단계이다. 사용자는 컨소시엄 블록체인의 CA(Certificate Authority)를 통해 인증서 기반의 개인키를 통해 인증을 수행한다. 정당한 사용자는 자신의 데이터를 등록한다.

2. Hybrid Blockchain Linking 단계

이 단계에서는 사용자가 Off Blockchain에 등록한 데이터의 On Blockchain 기록을 위해 메타데이터를 생성하는 과정이다. 이는 Name-Value 방식의 Global State 속성과 원본데이터의 해시데이터를 메타데이터로 지정한다. 데이터 등록은 0, 수정은 1, 삭제는 2와 같은 상태를 가진다.

3. 데이터 수정·삭제 단계

이 단계에서 사용자가 수정과 삭제를 요청할 시, 수정할 데이터를 등록하고 삭제할 데이터를 선택한다. XOR set에 따라 Key-Value로 구성되어 있으며 수정 혹은 삭제할 데이터의 변경된 값 만큼은 Global State에 인젝션 되어 Off Blockchain의 Global State를 인위로 변경한다. 이때 서버의 서명과 사용자의 서명과 같은 정당한 스마트컨트랙트의 유효성이 검증 될 경우에만 On Blockchain의 트랜잭션이 발생된다.

IV. 제안방식 분석

- 사용자 인증(Authentication): 정당한 사용자

는 사용자의 공개키-개인키를 통해 네트워크의 올바른 사용자임을 확인한다.

□ 신뢰성(Reliability) : 네트워크 참여자는 네트워크의 무결성과 가용성을 보장받아 전체 네트워크를 신뢰한다.

□ 효율성(Efficiency) : XOR 기반의 처리 기술은 안전하면서 전체 처리량의 오버헤드를 최소화하여 효율적으로 계산 및 가용을 제공한다.

□ 프라이버시(Privacy) : Off Blockchain에 포함된 트랜잭션은 정당한 그룹원만이 접근하여 사용되며 On Blockchain에는 데이터의 해시 값과 데이터 속성 값만으로 생성한 메타데이터로 기록하여 사용자의 정보 노출을 최소화한다.

V. 결론

본 논문에서 수정·삭제를 위한 Hybrid Blockchain 기반의 XGS 인젝션 기술을 제안하였다. 제안방식분석에 따라 보안요구사항을 만족하면서 데이터의 완전 수정·삭제를 하지 않으므로 추적을 지속적으로 수행하면서 수정·삭제를 가능하게 하였다. 또한 XOR 연산만을 사용하여 계산의 효율성을 가진다. 향후 구체화 된 환경에 적용하여 필요한 기반 프로토콜을 구현 및 실제 구현까지 확대 적용이 필요할 것으로 생각된다.

[참고문헌]

- [1] Nakamoto, S., "Bitcoin: A peer-to-peer electronic cash system", 2008.
- [2] Watanabe, Hiroki, et al. "Blockchain contract: Securing a blockchain applied to smart contracts.", Consumer Electronics (ICCE), 2016 IEEE International Conference on. IEEE, 2016.
- [3] Cachin, C., "Architecture of the hyperledger blockchain fabric", In Workshop on Distributed Cryptocurrencies and Consensus Ledgers (Vol. 310), 2016.

기업 블록체인 거버넌스와 위험 통제 프레임워크 설계

이경모*, 이경현**

*부경대학교 정보보호학협동과정

**부경대학교 IT융합응용공학과

dlrud2539@pukyong.ac.kr

Design of Framework for Enterprise Blockchain Governance

*Kyeong Mo Lee, **Kyung-Hyune Rhee

*Interdisciplinary Program of Information Security, Graduate School,
Pukyong National University

**Department of IT Convergence and Application Engineering,
Pukyong National University

요약

블록체인 기술은 최초 화폐의 교환을 목적으로 제안된 이후 다양한 어플리케이션 분야를 지원할 수 있도록 발전하고 있으며 적합하게 설계되고 구현된 블록체인 기술은 많은 경제적 이익을 가져다 줄 수 있다. 하지만 블록체인 기술에 대한 이해 부족과 블록체인이 기업의 비즈니스 목표를 효율적으로 지원하기 위한 체계화된 구조가 없어 많은 기업에서의 도입이 지연되고 있다. 이러한 문제 개선을 위해 본 논문에서는 기업 블록체인 거버넌스와 통제 프레임워크 설계를 통해 이러한 유기적인 통제 구조를 수립하며 이를 위해 다양한 접근 방법과 프로세스를 제시한다.

I. 서론

2009년 블록체인 기술은 나카모토 사토시에 의해 제안되었으며 최초 화폐의 교환 목적으로 설계 되었으나 이후 이더리움 블록체인 등 스마트 컨트랙트를 활용하여 다양한 분산 어플리케이션을 구현할 수 있도록 개발되었다[1]. 보다 최근에는 하이퍼레저 아키텍처와 같이 고속 처리와 기밀성 등이 개선된 블록체인 플랫폼이 기업에서 확산되고 있으며 블록체인은 기업의 IT 환경과 비즈니스 모델을 변화시키고 있다 [2].

하지만 여전히 많은 기업에서는 블록체인 기술이 도입되지 못하고 있으며 이는 블록체인의 기술적 이해 부족과 더불어 블록체인이 기업의 비즈니스 목표를 지원해주기 위한 구조화된 방법이 부재한 것에 기인한다. 또한 블록체인 기술의 도입 및 운영 시 이에 대한 긍정적 혹은 부정적 위험에 대해 평가되어야 하지만 이러한 구체적인 통제 방법이 부재하는 문제점이 있다.

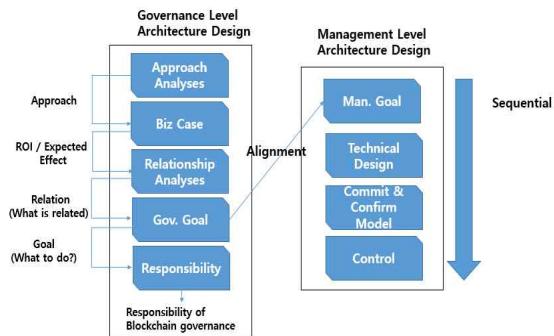
이러한 문제점은 블록체인 거버넌스 구조와 위험 통제 프레임워크를 수립함으로써 완화할

수 있다. 일반적으로 기업 거버넌스란 이사회 및 고위 경영진의 수준에서 결정된 기업의 비즈니스 목표 및 전략과 부합하여 기업 내 각 영역 및 이를 지원하는 기술 등이 기업의 궁극적 목표와 연계되는 것을 보장하며 이를 위해 구성된 책임과 지배 구조 등을 의미한다[3]. 기존에 정의된 거버넌스의 경우 다수가 존재하며 블록체인과 직접적으로 관련된 거버넌스로는 IT거버넌스와 정보보호거버넌스가 있다[4]. 이러한 거버넌스는 상위 기업의 비즈니스 목표를 지원하기 위한 IT와 정보보호 기술에 대한 전략 및 조직구조, 책임 등을 정의한다. 또한 하위 개념으로 이러한 상위 수준의 전략 및 목표에 대해 조직이 이를 운영 및 모니터링 하는 일련의 행위를 관리라고 한다[5].

본 논문에서는 IT 및 정보보호 거버넌스 모델과 상호 작용하며 비즈니스 목표를 블록체인 기술이 효율적으로 지원하기 위한 거버넌스 및 위험 통제 프레임워크 설계를 진행하며 이를 위해 필요한 접근 방법과 프로세스 및 고려사항을 세부적으로 분석하여 제시한다.

II. 설계 프로세스

본 절에서는 기업 거버넌스 및 위험 통제 프레임워크 설계를 위해 거버넌스 수준과 관리 수준의 아키텍처 설계 프로세스를 제시하며 이들 간의 관계에 대해 기술한다.



<그림1> 블록체인 거버넌스 아키텍처 구조

1.1 거버넌스 수준 아키텍처 설계

- 접근법 분석(Approach Analyse)

블록체인 거버넌스 및 위험 통제 프레임워크에 대한 핵심 접근법을 도출하기 위해 위험 통제 및 거버넌스 분야의 국제 표준 및 최선 실무 사례(Best Practice)와 라이브러리(Library)[1-8]을 분석하여 그림2와 같은 PDCA, 성숙도 모델 및 차이 분석 기법, 핵심 목표 기반의 통제 프레임워크 모델, 거버넌스와 관리의 분리 체계, 최선 실무 사례, 라이브러리 기반의 접근법을 도출하였으며 이를 블록체인 거버넌스 및 위험 통제 프레임워크 모델에 적용할 수 있다.

Reference Model	Key Approach	Explanation
ISO 27001[6]	PDCA (plan-do-check-act)	정보보호관리체계의 표준으로써 정보보호와 통제의 관점에서 지속적인 <계획-실행-확인-개선>의 프로세스를 가짐.
NIST Security Framework[4] CMMI[7]	Maturity level Gap Analysis	측정 가능한 현재 기업의 정보보호/프로세스 수준을 N 단계의 성숙도 수준으로 정의하고 이러한 현재의 프로파일을 토대로 미래 지원점과의 차이 분석과 이를 위한 개선 프로세스를 가짐.
COBIT 4.1[5]	핵심 목표기반의 통제 프레임워크	IT 거버넌스의 모델로써 IT 기술이 가져야 할 핵심 목표와 가치를 설정하고 이에 대한 다양한 프레임워크 등을 제시함.
COBIT 5.0[3]	거버넌스와 관리의 분리	전사적 IT 거버넌스의 표준으로써 상위 거버넌스 체계에서의 EDM(평가, 지시, 모니터링)-PBRM(계획, 구축, 운영, 모니터링)를 구분하는 접근방법을 취함.
NIST Security Framework[4] ITIL[8]	Best Practice/ Informative Reference	정보보호 및 IT 서비스 및 운영에 관한 최선 실무 (Best Practice) 기반의 프레임워크 및 라이브러리를 제공함.

<그림2> 접근법 분석 결과

- 비즈니스케이스

비즈니스케이스(Business Case)란 특정 프로젝트의 시작 전 혹은 제품 및 서비스의 개발/획

득/구현/유지보수/운영 등의 과정에서 대상 제품과 서비스 기술이 비즈니스 목표와 연계되고 상위 목표에 대해 유용한 가치를 제공하는지에 관한 비용 효과적인 타당성 검토를 의미한다. 이러한 비즈니스케이스는 블록체인 거버넌스 모델의 구현 시 검토되어야 하며 블록체인 기술이 적용되는 범위 내 일부 혹은 전체에 대해 평가되어야 한다. 이 경우 정량적 혹은 정성적인 이익이 발생할 수 있는지에 대해서도 검토되어야 한다.

- 관계 분석(Relationship Analysis)

비즈니스케이스에서 지정된 블록체인 기반의 제품 및 서비스의 경우 기존 IT 및 정보보호 인프라 및 기술에 대해 대체 및 보완하는 관계에 있을 수 있다. 이러한 관점에서 블록체인 거버넌스 구성을 위해서는 기업 내 블록체인 기술이 다른 거버넌스 구조에 있는 기술들과 연계점 분석이 선행되어야 한다. 이러한 연계구조는 다음과 같이 분류가 가능하다.

- 독립 : IT/정보보호 기술에 대해 독립적인 기능과 통제를 갖는다.
- 관계 : IT/정보보호 기술에 대해 의존적이거나 보완하는 기능과 통제를 갖는다.
- 공통 : IT/정보보호 기술에 대해 같은 수준의 기능과 통제가 요구된다.

• 최종 목표(Governance Goal)

이전 단계에서 도출된 결과를 통해 블록체인 기술은 사업의 목표 및 IT와 정보보호 거버넌스와 연계되며 비용 효과적으로 비즈니스 목표를 지원하도록 설계될 수 있으며. 또한 이러한 분석을 통해 도출된 블록체인의 기능과 통제는 향후 위험 관리 프레임워크 개발을 위해 사용될 수 있다.

언급한 일련의 거버넌스 수준의 설계를 통해 블록체인 거버넌스의 목표는 언급한 상위 혹은 동등한 레벨의 거버넌스 체계의 지원, 비용 효과적인 비즈니스 목표 달성을 지원으로 설정된다.

• 책임(Responsibility)

거버넌스 수준의 설계 중 책임이란 블록체인 기술이 궁극적으로 기업의 비즈니스 목표 달성을 위해 블록체인 기술의 도입 및 운영 등에 관한 전반에 관한 최종적인 결정과 궁극적 책임을 지는 주체를 설정하는 단계이며 다음과 같은 수준으로 설계가 가능하다.

- Management (CxO)

이는 블록체인에 관한 기술적 의사 결정 수준으로써 블록체인 기술이 기존 IT 및 정보보호 기술과 연계하기 위한 결정은 Management Level 이상에서 수행될 수 있음을 의미한다.

- Board(Directors) & Senior Management 블록체인 기술이 개별 혹은 동등한 수준의 IT/정보보호 기술 범위보다 상위 수준에서 프로세스 전반에 미치는 영향이 있을 경우 이는 고위 경영진 수준 이상에서 결정되어야 한다.

1.2 관리 레벨 아키텍처 분석

• 관리 목표(Management Goal)

관리의 목표는 블록체인 거버넌스에서 결정된 궁극적인 목표와 전략에 부합하도록 하는 블록체인과 관련된 계획/운영/모니터링/개선을 포괄하는 개념이다. 이러한 관리의 목표는 COBIT 참조 모델에 따라 다음 5가지로 정의 될 수 있으며[5] 의미하는 바는 다음과 같다.

- 가치 전달 : 기업의 목표와 연계하여 블록체인 기술이 가치를 전달 할 수 있도록 하는 방법.
- 위험 관리 : 블록체인 기술이 불러일으킬 수 있는 위험(긍정적/부정적)을 관리하는 것.
- 자원 관리 : 블록체인 기술과 관련된 인력, 인프라, 데이터, 어플리케이션 등 관련된 자원을 효율적으로 사용하는 것.
- 성과 측정 : 블록체인 기술이 전달할 수 있는 가치에 대해 평가하는 방법.
- 전략적 연계 : 블록체인 기술이 동등 혹은 상위 수준의 목표를 지원하며 이에 대해 효율적인 방식으로 연계될 수 있도록 하는 것.

• 기술설계(Technical Design)

기술설계란 거버넌스 수준에서 설정한 비즈니스 케이스, 관계 분석을 통해 결정된 범위 내에서 필요한 구체적인 기술 범위를 설계하는 것으로써 다음의 경우의 예로 설명할 수 있다.

- 거버넌스 수준 설계(예) : 거버넌스 설계에서 외부 기업 간 네트워킹 기능, 분산 DB 기능과 정보보호의 무결성 및 가용성, 거래의 투명성 기능과 관계가 있는 블록체인을 컨소시엄 형태로 하여 기업 간 의사

결정 시스템(비즈니스 케이스)을 구축하려고 한다.

- 기술설계 : 실제 적용될 기술의 범위인 기업 간 블록체인 트래픽에 관한 규격(포트 등). 사용될 객체 / JSON 포맷 등 기술적인 요소가 설계된다.

• 제출 및 승인(Commit & Confirm Model)

기존 기업 내 트랜잭션 승인에 대한 직무 분리와 통제의 관점에서 이러한 제출 및 승인에 관한 책임과 권한 관계를 식별하는 것은 관리 수준의 설계에서 고려되어야 한다. 이 경우 제출자(Committer)와 확인자(Confirmator)는 반드시 직무 분리가 이루어져야 하며 기업 외부와 연계될 경우 적절한 통제 절차가 수립되어야 한다. 이러한 제출 및 승인의 구성을 다음과 같이 수준에서 세분화 할 수 있다.

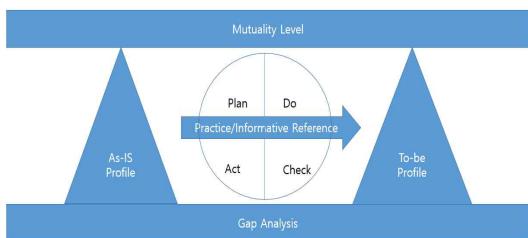
- Parallel Commit & Confirm : 이는 각 부서별로 비즈니스 로직에 따라 접근 권한과 승인 권한을 고유 부서에 한정시키는 모델로써 정보 흐름이 각 부서의 장(Manager)과 직원(Staff)으로 한정된다. 이 경우 반드시 제 3자의 감독에 관한 권한을 고려하여야 한다.

- Parallel Commit & Committee Confirm : 이는 개별 부서에서 트랜잭션이 제출되지만 전사에 영향력을 미칠 수 있으며 중요한 경우 각 부서의 장이 참여하는 위원회(Committee)에서 블록체인의 트랜잭션을 승인하는 모델을 사용할 수 있다.

- Organization Level Commit/Confirm : 이는 블록체인 기술이 컨소시엄 형태로 구성될 경우 각 비즈니스는 조직 단위로 행해지며 이러한 동일한 수준의 기업 조직은 각 컨소시엄의 대표 노드들로 트랜잭션의 주체를 구성하여야 한다.

• 통제(Control)

위험 통제는 기술 설계와 제출 및 승인 모델에 따른 잠재적 위험을 평가하고 위험에 대해 통제를 설계하여 기업 내에서 정의된 수용 가능한 위험 수준(Acceptable Level of Risk, ALR)으로 위험을 통제하는 것이다. 이를 위해 접근 방법 중 하나인 PDCA 모델을 사용하며 각 단계는 다음과 같이 설계될 수 있다.



- Plan(계획) : 현재 통제 프로파일(AS-IS Control)을 평가하고 목표 통제 프로파일(TO-BE)을 구성하는 것이 목표인 단계로써 기존 레거시 시스템 혹은 구축되지 않은 시스템에서 블록체인 기술 기반으로 변경되었을 경우 제거되었거나 추가된 통제를 식별하는 단계로 정의한다. 이 단계에서는 알려진 통제 프레임워크 및 기술인 ISO/IEC 13335-1:2004, 17799:2005, 27001, Guide 73:2002의 참조를 통해 현재의 통제 수준을 도출하고 목표 통제 프로파일과 Gap(차이)을 도출한다. 이후 “Do” 단계에서는 이러한 차이의 개선을 줄이는 활동을 할 수 있다.
- Do(통제 수행) : 도출된 현재 프로파일에서 기업이 수용 할 수 있는 통제 위험(Acceptable Level of Risk, ALR) 수준까지 위험을 관리하기 위한 통제를 수행한다. 이 경우 예방/적발/교정/보완/중복통제 등이 고려될 수 있으며 통제간의 의존성과 우선순위가 고려되어야 한다. 또한 블록체인의 경우 기존 기업의 IT 인프라 혹은 정보보호 장비와 관련된 통제를 받기 때문에 이러한 점을 고려하여 설계하여야 한다.
- Check : 지속적이고 주기적으로 통제와 위험에 대해 목표 프로파일과 외부 환경에 적응적으로 재평가하여야 하며,内外부 환경의 변화, 블록체인 기술과 관련된 이해관계자 및 승인 구조 등의 변화가 있을 시 재평가 되어야 한다.
- Act : 지속적으로 이러한 통제 프로파일의 수준에 대해 개선하며 목표 통제 프로파일과 근사값을 유지하도록 하는 활동으로 정의한다.

이와 더불어 통제 단계에서는 블록체인 기술과 관련된 통제를 최적화하기 위한 최선 실무(Practice)가 개발되어야 하며 블록체인 기술과

관련된 알려진 공격 벡터 및 취약점 등이 함께 고려되어야 한다. 또한 이러한 위험은 기술적인 부분 뿐 아니라 커플라이언스 및 관리적인 측면에서도 함께 고려되어야 할 것이다.

III. 결론

본 논문에서는 기업의 블록체인 기술 확산과 비즈니스 목표 지원을 위한 블록체인 거버넌스 및 통제 프레임워크 설계 프로세스를 제안하였다. 제안 모델에서는 다양한 방법론과 프로세스 및 고려사항을 제시하였다. 이후 이러한 거버넌스 및 프레임워크 구조를 기반으로 통제된 블록체인 기술은 기업의 비즈니스 목표를 효율적으로 지원할 수 있으며 실질적인 이익을 발생시킬 수 있을 것이다.

Acknowledgement

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. NRF-2018R1D1A1B07048944)

[참고문현]

- [1] V.Buterin, Ethereum White Paper, Available in <https://github.com/ethereum/wiki/wiki/White-Paper>, 2013.
- [2] C.Cachin, Architecture of The Hyperledger Blockchain Fabric, In Workshop on Distributed Cryptocurrencies and Consensus Ledgers, vol. 310, pp. 1-4, July. 2016.
- [3] P.Năstase, F.Năstase, and C.Ionescu, Challenges Generated by The Implementation of The IT Standards CobiT 4.1, ITIL v3 and ISO/IEC 27002 in Enterprises, Economic Computation & Economic Cybernetics Studies & Research, vol. 43, no. 1, pp 1-16, 2009.
- [4] S.J.Shackelford, A.A.Proia, B.Martell, and A.N.Craig, Toward a Global Cybersecurity Standard of Care: Exploring The Implications of The 2014 NIST Cybersecurity Framework on Shaping Reasonable National and International Cybersecurity practices, Texas International Law Journal, vol. 50, no. 305, pp. 1-33, 2015.

- [5] S.D. Haes, W.V. Grembergen, and R.S. Debreceny, COBIT 5 and Enterprise Governance of Information Technology: Building Blocks and Research Opportunities, Journal of Information Systems, vol. 27, no. 1, pp 307–324, 2013.
- [6] G.Disterer, ISO/IEC 27000, 27001 and 27002 for Information Security Management, Journal of Information Security, Vol. 4, no. 2, pp 1–92, April. 2013.
- [7] C.P.Team, Capability Maturity Model Integration (CMMI SM), Version 1.1. CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS), 2002.
- [8] B. Orand, and J.Villarreal, Foundations of IT Service Management: With ITIL 2011, ITILyaBrady, 2011.



좌장 : 최대선 (공주대)

딥러닝 기반 의료분야 인공지능 기술 동향 조사

김도완*, 최대선†

*공주대학교 의료정보학과

dhdp200@smail.kongju.ac.kr

A Survey on the trend of medical artificial intelligence based on deep learning

Do-Wan Kim*, Dae-Sun Choi†

*Medical Information, Kongju University.

요약

의료 데이터는 ‘빅데이터’의 4가지 속성인 크기(Volume), 다양성(Variety), 수집속도(Velocity), 신뢰도(Veracity)를 모두 가지고 있다. 하지만 데이터의 의미 있는 활용은 보급에 비해 낮은 수준이고 향후 의료전문가 부족 현상도 점점 심해질 예정이며 이에 따라 임상 진단의 불일치나 의료영상 판독 지연의 경우가 흔히 발생하여 이에 대한 해결책이 필요한 상황이다. 최근 들어 앞서 말한 문제점들을 극복하기 위해 딥러닝 기반 의료분야 인공지능 기술을 도입하여 의료 데이터 분석 및 진단 보조 솔루션으로 확산하고 있다. 본 논문은 인공지능의 개념을 간략히 요약한 후 딥러닝 기반 의료분야 인공지능 기술 동향을 소개하고자 한다.

I. 서론

의료 데이터는 우리가 흔히 말하는 ‘빅데이터’의 4가지 속성인 크기(Volume), 다양성(Variety), 수집속도(Velocity), 신뢰도(Veracity)를 모두 가지고 있다. 하지만 데이터의 의미 있는 활용은 보급에 비해 낮은 수준이고 의료전문가 부족 현상도 점점 심해지고 있다. 한국보건사회연구원에 따르면 우리나라의 경우 2030년까지 의사 7,600명, 간호사 15만 8,000명이 부족할 것으로 예상되고 있다[1]. 이에 따라 임상 진단 불일치나 의료영상 판독 지연의 경우가 흔하게 발생한다.

최근 들어 이러한 문제점들을 극복하기 위해 딥러닝 기반 의료분야 인공지능 기술을 빠르게

도입하여 의료 데이터 분석 및 진단 보조 솔루션으로 확산하고 있다.

본 논문에서는 딥러닝 기반 의료분야 인공지능 기술 동향을 소개하고자 한다. II장에서는 인공지능 개념에 대해 요약한 후, 딥러닝 기반 의료분야 인공지능 기술을 소개한다. III장에서는 결론 및 향후 계획을 제시한다.

II. 딥러닝 기반 의료분야 인공지능 기술

인공지능은 ‘지능적인 기계를 만드는 엔지니어링 및 과학’을 의미한다. 인공지능이라는 개념이 처음 제안되었을 때 실현을 위한 방법론의 부재로 인해 정체기를 지속하다가 근래 머신러닝 및 딥러닝의 다양한 활용으로 급속도로 발전하고 있다[2]. 인공지능은 다양한 산업에서 적용되고 있는데, 의료분야에서는 의료영상 분야와 논문 분석, 웨비포 온콜로지와 같이 치료법을 권고해주는 역할에서도 적용되고 있다[3].

2.1 의료영상 분석

* 이 논문은 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No. 2017-0-00380, 차세대 인증 기술 개발)을 받아 수행하였습니다.

† 교신저자, sunchoi@kongju.ac.kr(Corresponding author)

의료영상 분석방법은 X-ray, CT, MRI, 초음파, 내시경 등 다양한 의료영상에 적용되고 있으며, 병변 탐지 및 정량화, 병변 분류뿐만 아니라 최근에는 인공영상의 생성이나, 영상 간 변화, 해상도 향상 등에 도입하여 활발히 연구되고 있다. Maria J. M. Chuquicusma는 DC-GANs 알고리즘을 사용하여 폐 결절 샘플을 생성하는 기술을 제안하고 생성된 폐 결절과 실제 결절을 비교를 위해 방사선사에게 Visual Turing Test를 실행하는 실험을 하였다 [4]. Figure 1은 폐 결절 샘플을 생성하는 DC-GANs의 구조를 나타낸다. Generator는 3개의 Convolutional layer로 구성되어 있으며 56 by 56 크기의 샘플을 생성하고, Discriminator는 2개의 convolutional layer로 구성되어 있고 주어진 샘플이 실제 이미지인지 생성된 샘플인지 확률점수를 출력해준다. 생성된 결절과 실제 결절을 비교하기 위해서 2명의 방사선사에게 Visual Turing Test를 실시하였다. Visual Turing Test에 대한 결과는 True Recognition Rate(TRR)과 False Recognition Rate(FRR)을 계산했다. 방사선사 1의 평균 TRR은 58.56%였고, 방사선사 2의 평균 TRR은 93.52%를 나타냈다. 결론적으로 이 실험은 제시된 DC-GANs 알고리즘이 고품질의 폐 결절 샘플을 생성할 수 있는 능력을 보여줌으로써 방사선사의 교육 목적이나 빅데이터로 훈련시키는 딥 네트워크의 실제 샘플을 생성하는 목적으로 사용될 수 있다.

한석민 외 1명은 GAN을 사용하여 임상 정보 데이터를 생성시키고 진짜 데이터와 유사한 가짜 데이터를 합성하여 별개의 전립선암을 판별

하는 Network에 적용하여 성능을 높여주는 기법을 제안하였다[5]. Figure 2는 GAN의 구조를 나타낸다. 임상 데이터는 8개의 feature(나이, PSA, TRUS class, DRE, Total Pros, Total_vol, TZ-vol, PSAD(PSA/Total_vol), PSA-TZ(PSA/TZ-vol))와 1개의 target data(전립선암 여부)로 이루어져 있다. Generator와 Discriminator의 Input에 Class를 포함시켜 Output class에 알맞은 데이터가 생성될 수 있도록 하였다.

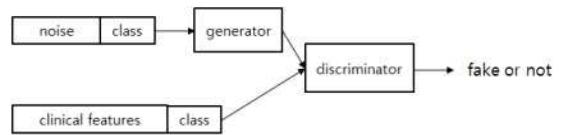


Figure 91 GAN 구조

Figure 3, 4, 5는 각각 합성 데이터를 포함하지 않고 학습시킨 경우, 합성 데이터를 포함하고 학습시킨 경우, 합성 데이터만으로 학습시킨 경우를 보여준다. 이 연구는 균형이 맞지 않는 Class balance를 완화하는 문제에도 적용할 수 있고 실제 데이터 기반으로 가상의 데이터를 생성하여 병원 외부에서 연구목적으로 사용하는 데에도 도움을 줄 수 있다.

2.2 IBM 왓슨 포 온콜로지

왓슨 포 온콜로지(Watson for Oncology)는 자연어 처리 능력을 토대로 의학 논문을 분석하고 최적의 의료법을 제시함으로써 임상 의사 결정을 도와주는 의료 인공지능 시스템이다[6]. 의료법을 초록색, 주황색, 빨간색 3단계로 권고를 해주는데, 초록색은 추천하는 치료법, 주황색은 고려해볼 수 있는 치료법, 빨간색은 권고

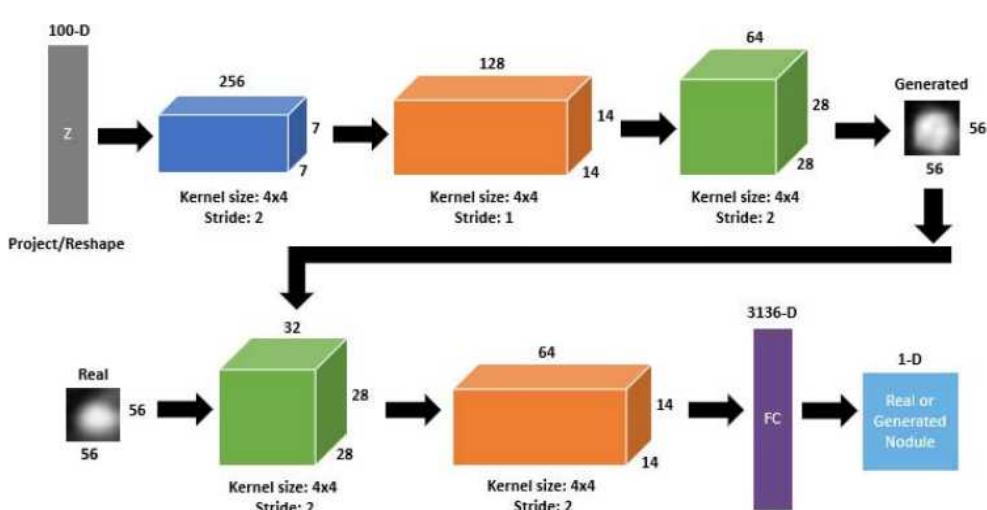


Figure 90 폐 결절 샘플 생성을 위한 DC-GANs 구조

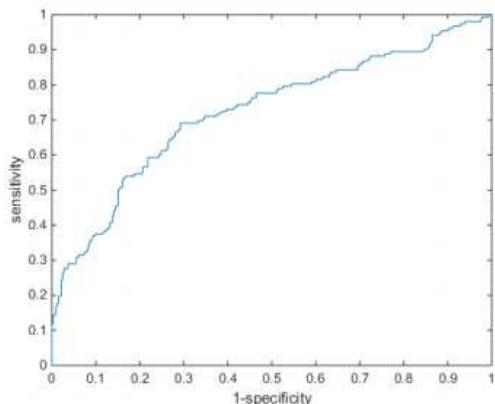


Figure 92 합성 데이터를 포함하지 않는 경우

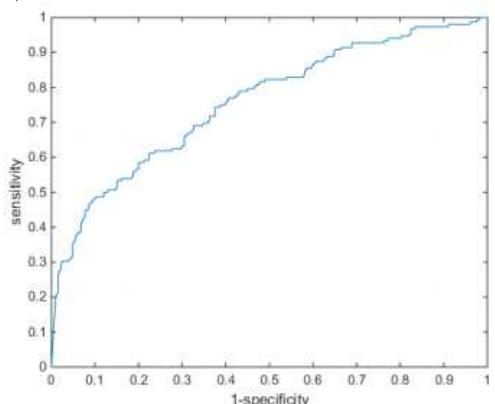


Figure 93 합성 데이터를 포함한 경우

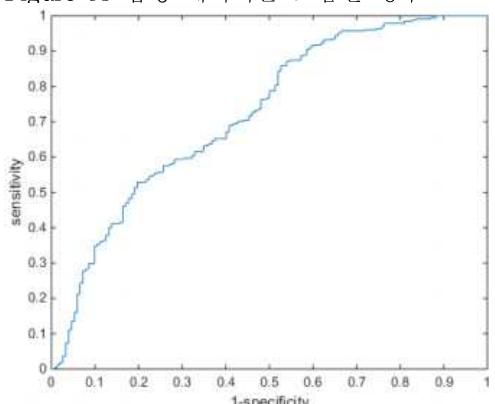


Figure 94 합성 데이터만 학습한 경우

하지 않는 치료법을 나타낸다. 각각의 치료법을 선택하면 왜 이러한 치료법을 권고하였는지 관련 논문, 임상 연구 결과 등 근거 자료들을 보여준다. 왓슨 포 온콜로지는 엄청난 분량의 암 관련 논문, 임상시험 결과들을 환자 치료에 빠르게 반영할 수 있는 강점이 있다. 하지만 왓슨의 학습이 자동으로 논문을 자동으로 읽고 스스로 판단하는 것이 아니라 왓슨을 훈련시키는 과정에서 수작업 교정에 많은 시간을 사용하는

것으로 알려져 있다.

왓슨 포 온콜로지는 국내병원에 빠르게 도입되면서 진료 보조로 활용하여 환자 만족도가 높아지고 소통이 활발해졌다. 향후 전 세계 보험환경과 의사 서비스 수준에 맞도록 왓슨 기능을 향상시킬 것이다.

III. 결론

딥러닝 의료분야 인공지능 기술이 발전함에 따라 의료 빅데이터 활용, 의료진 부족 현상, 오진, 판독지연 등에 대한 솔루션이 확산되고 있다. 하지만 환자의 의료정보(개인정보) 유출, 의료기기 해킹, 랜섬웨어 감염에 따른 업무 마비 등 사이버 공격이 잇따라 발생하고 있다. 낮은 보안의식, 보안조직의 부재 등 정보보호 대책 마련에 어려움을 겪고 있다. 따라서 의료분야의 보안의식을 키우고 의료정보보호 기술연구가 요구될 것으로 보인다.

[참고문헌]

- [1] 정규환, “인공지능 기반 의료영상 분석 기술 동향”, 정보통신기술진흥센터, 2018
- [2] 최예림, 김관호, “인공지능 개요 및 적용 사례”, Ie매거진, 2016
- [3] 최윤섭, “IBM 왓슨 포 온콜로지의 의학적 검증에 관한 고찰”, Hanyang Medical Reviews, 2017
- [4] Maria J. M. Chuquicusma, Sarfaraz Hussein, Jeremy Burt, Ulas Bagci, “How to fool radiologists with Generative Adversarial Networks? A visual turing test for lung cancer diagnosis”, ISBI, 2018
- [5] 한석민, 이동열, “Generative Adversarial Network를 이용한 전립선암 진단용 임상데이터 합성 방법”, 한국정보과학회, 2018
- [6] 이다은, “4차 산업혁명과 의료? 길병원의 왓슨 도입을 중심으로”, 한국과학기술학회, 2017

CNN기반 악성코드 이미지화 및 분석 시스템

김용수, Le Thi Thu Huong, 김호원□

부산대학교 전기전자컴퓨터공학과

{dkgoggog0329, lehuong7885, howonkim}@gmail.com

CNN-based Malicious Code Imaging and Analysis System

Yong-Su Kim, Le Thi Thu Huong, Ho-Won Kim□

*Department of Electrical and Computer Engineering, Pusan National Univ.

요약

본 논문에서는 딥러닝 기법 중 하나인 CNN(Convolutional Neural Network)를 이용하여 악성코드를 이미지화하여 효과적으로 분석하는 기법을 제안한다. 본 연구에서 개발한 시스템을 이용하여 현재 분석 방해 기법 등으로 지능화된 악성코드를 기존의 분석 기법에 비해 높은 정확도로 분류할 수 있다.

I. 서론

최근 컴퓨터와 인터넷의 보급이 크게 확산됨에 따라 정보 검색, 인터넷 거래 등 인간에게 편리한 기능을 많이 제공해 주었지만, 이에 따른 부작용 역시 급증하고 있다. 그중 분산서비스거부(DDoS) 공격, 랜섬웨어, 개인정보 유출, 해킹 등은 대부분 악성코드에 의해 발생하며, 사용자의 피해가 지속해서 증가하고 있다.

과거에는 악성코드의 전파속도가 느리고 감염되는 경로가 한정적이었으며, 전파기술 또한 단순한 구조였기 때문에 이에 대응하는 방안에 대한 연구는 비교적 어려운 편이 아니었다. 하지만 현재의 악성코드들은 은닉기술과 자가변형, 난독화, 가상 환경 우회 등의 발달한 분석 방해 기법으로 매우 지능화되어, 기존의 악성코드 분석 기법으로는 현재의 지능화된 악성코드를 분석하기에 많은 어려움을 지니고 있다.

최근 딥러닝으로 대표되는 머신러닝 기술의 발달로 인해, 많은 분야에서 인간의 분석 및 처

리 능력보다 뛰어난 효과를 보여주고 있다. 딥러닝 기술은 여러 비선형 변환기법의 조합을 통해 다양한 데이터나 복잡한 자료들을 분석하여 숨어있는 패턴을 발견해 데이터를 분류하거나 미래에 일어날 현상을 예측하는 기술이다.

본 논문에서는 이러한 딥러닝 기술을 악성코드 분석 기법에 적용하여 효과적으로 악성코드를 분석하며, 특히 이미지 분류에 효과적인 CNN(Convolutional Neural Network) 기법을 악성코드 분석에 적용하기 위해 악성코드를 이미지화하는 방법과 그에 따른 효과적인 분석 기법을 소개한다.

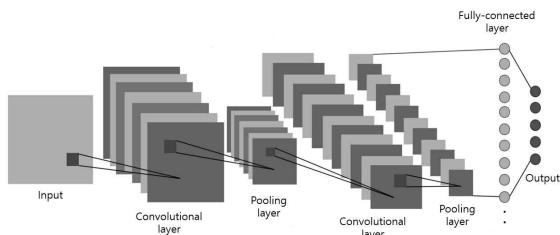
II. 본론

2.1 CNN(Convolutional Neural Network)

CNN은 기존 Neural Network에서 이미지 데이터의 공간 정보 손실이 일어난다는 단점을 극복하여 이미지의 공간 정보를 유지하면서 인

접 이미지와의 특징을 효과적으로 인식하는 특징이 있다. 또한, Filter라는 공유 파라미터를 사용하여 기존 신경망 기술에 비해 학습 시간이 월등히 감소되는 장점이 있다.

CNN의 주요 구조는 [그림 1]과 같이 Input layer, Convolutional layer, Pooling layer, Fully-connected layer, Output layer로 이루어져 있다.



[그림 95] CNN 전체 구조도

CNN의 Input layer는 특징상 주로 이미지 데이터를 읽어 들인다.

Convolutional layer는 입력 데이터를 Filter가 순회하며 합성곱 연산을 하여 Feature map을 만든다. 이러한 방법으로 여러 Filter를 사용하면 이미지의 특징을 효과적으로 추출할 수 있게 된다.

Pooling layer는 Convolutional layer의 출력 데이터를 입력으로 받아서 출력 데이터의 크기를 줄이거나 특정 데이터를 강조하는 용도로 사용된다.

Fully-connected layer는 Convolutional layer와 Pooling layer가 겹쳐진 여러 층으로부터 나온 출력 데이터를 1차원으로 펴주어 기존의 신경망 기법으로 Output layer와 연결한다. Output layer의 node 개수는 원하는 데이터 분류 개수에 맞게 설정한다.

2.2 악성코드 이미지화

본 연구에서는 CNN을 이용하여 악성코드를 분석하기 위해, 전처리 단계로 악성코드를 이미지화하는 방법을 적용하였다. 악성코드 바이트 코드는 그레이스케일 이미지로 시각화시키는 것이 가장 적절하다. 대부분의 악성코드는 동일한 그룹에 속하는 이미지가 레이아웃과 텍스트 면에서 매우 유사하게 나타난다.

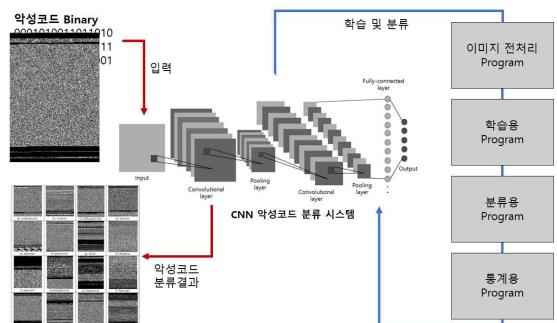
주어진 악성코드 Binary는 부호 없는 8비트 정수로 된 벡터로 읽혀진 2D 배열로 구성된다. 이 범위는 [0, 255](0:검정색, 255:흰색) 범위의 그레이스케일 이미지로 쉽게 시각화 할 수 있다. 이미지의 너비는 고정되어 있으며 높이는 파일 크기에 따라 달라질 수 있다. [그림 2]는 Alueron.gen'J 악성코드를 그레이스케일 이미지로 변환한 모습이다.



[그림 96] 악성코드 이미지화

2.3 악성코드 분석 시스템

본 연구에서 개발한 악성코드 이미지화 및 분석 시스템의 전체 구조도는 [그림 3]과 같다.



[그림 97] 개발 전체 구조도

악성코드 분석 시스템을 개발하기 위해 Python과 머신러닝 라이브러리 Tensorflow를 사용하였고, 악성코드 데이터세트는 25개 그룹의 악성코드 종류와 9,339개의 악성코드 샘플로 이루어진 Kaggle의 Malimg Dataset를 이용하였다. 주어진 데이터세트에 맞게 악성코드를 분류하기 위해 악성코드를 이미지화한 것을 CNN의 Input layer로 입력을 받고, 25개의 악성코드 그룹으로 분류하기 위해 Output layer의 node 개수를 25개로 설정하였다.

개발한 CNN기반 악성코드 분류 시스템은 주어진 악성코드를 이미지화하는 이미지 전처

리 Program, CNN 신경망에 이미지를 입력시켜 학습하는 학습용 Program, 학습이 다 끝난 상태에서 악성코드를 그룹에 맞게 분류하는 분류용 Program, 분류 결과를 통계적으로 나타내고 정확도를 실험하는 통계용 Program으로 구성되어 있다.

III. 결론

본 연구에서는 딥러닝 기술인 CNN기반의 악성코드 이미지화 및 분석 시스템을 개발하여 악성코드를 효과적으로 분류하는 방법을 제안하였다.

딥러닝을 통한 악성코드 분류 방법에서 가장 중요한 것은 첫 번째로 이미지화의 방식이다. 높은 분류 정확도를 위해서는 CNN의 정밀도와 파라미터 프로그래밍 튜닝도 중요하지만 이미지화시 사용하는 Width값이나, 악성코드 바이트 코드 중에 나타난 알 수 없는 부분을 어떻게 처리하는지 등의 방법이 본 연구에서 중요한 요소라고 판단된다.

두 번째는 CNN의 구축방법이다. CNN의 구축은 이미지처리기, 학습기, 시험기, 결과출력 부분 등으로 분류할 수 있는데, 연산 횟수와 방법에 따라서 학습기와 시험기의 작동에 너무 많은 소요시간이 요구될 수 있어 시스템의 성능에 큰 영향을 미칠 수 있다. 이 문제를 해결하기 위해 LeNet, AlexNet, GoogLeNet 등의 다양한 CNN 변형 구조를 사용하여 효과적인 성능을 얻을 수 있다.

마지막으로 절대적으로 중요한 것은 학습 데이터의 양이다. 본 연구에서는 레이블간의 불균형과 같은 문제 때문에 100%에 가까운 정확도를 기록할 수는 없었지만, 이 시스템이 협업에 적용되어 학습할 수 있는 악성코드의 데이터 양이 많아지면 이것에 비례하여 악성코드 분석 시스템의 정확도도 높아질 것이다.

[참고문헌]

- [1] Trinius, P. Holz, T. Gobel, J. and Freiling, F.C. 2009. Visual analysis of malware behavior using treemaps and thread

graphs. In International Workshop on Visualization for Cyber Security (VizSec), 33–38.

- [2] Goodall, J.H. Randwan H. and Halseth, L. 2010. Visual analysis of code Security. In International Workshop on Visualization for Cyber Sec (VizSec)
- [3] Karim, M. E., Walenstein, A., Lakhotia, A. & Parida, L. 2005. Malware phylogeny generation using permutations of code. Journal in Computer Virology, 1 (1):13–23.
- [4] Malware Detection With Convolutional Neural Networks in Python, DZone, <https://dzone.com/articles/malware-detection-with-convolutional-neural-network/>

확률적 분석을 통한 Feature Selection 및 머신러닝 기반 악성코드 탐지*

박승수*, 이만희*

*한남대학교 컴퓨터공학과

parkseungsoo.kr@gmail.com, manheelee@hnu.kr

Feature Selection using Probabilistic Analysis and Malware Detection based on Machine Learning

Seung-Soo Park*, Man-Hee Lee*

*Division of Computer Engineering, Hannam University.

요약

최근, 매일 수십만 개 이상의 새로운 악성코드가 발견되고 있으며, IoT(Internet of Things) 기기의 정보 탈취 및 영상 정보 유출, Cryptojacking, Ransomware 등 개인의 사생활을 침해하거나 금전을 탈취하기 위한 목적의 악성코드가 증가하고 있다. 이와 같이 다양한 유형으로 출현하는 악성코드에 대응하기 위한 방법이 요구된다. 본 논문에서는 정적 분석을 통해 추출한 API/DLL의 조건부 확률을 산출하여 악성파일에 높은 확률로 사용되는 API/DLL을 선정하고, 이를 바탕으로 머신러닝을 활용해 악성코드를 탐지하는 방법을 제안한다. 본 논문에서 제안하는 방법으로 92%의 정확도를 보이는 머신러닝 모델을 도출하였으며, 산출한 조건부 확률을 통해 다양한 악성코드 분석 및 탐지 분야에 기여하고자 한다.

I. 서론

국제 보안제품 성능평가기관 AV-TEST에 따르면 현재까지 약 9억 개의 악성코드 샘플이 발견되었고, 매일 35만 개 이상의 새로운 악성코드가 발생하고 있다[1]. 이처럼 악성코드의 수는 기하급수적으로 증가하고 있으며, 그 유형도 다양하게 분화하고 있다. 스마트 냉장고를 악성코드 유포지로 활용하거나 공유기를 통한 개인계정 탈취 및 DDoS(Distributed Denial of Service) 공격 사례가 있었으며, 최근에는 IP카메라의 영상 유출이 급격하게 증가하고 있다[2]. 또한, 일반 시스템을 악성코드로 감염시켜 가상화폐를 채굴하는 Cryptojacking, 가상화폐 탈취, Ransomware 등 금전적인 이득을 취하기 위한 목적의 공격이 급증했다[3]. 따라서 이와 같은

공격에 신속하고 정확하게 대응하기 위한 방법이 필요하다.

본 논문에서는 정적 분석을 통해 PE(Portable Executable) 파일로부터 악성코드가 실행되기 위해 필요한 API/DLL을 추출하여 Feature로 활용하였다. 또한, 독립적인 하나의 Feature가 악성파일 샘플에 나타날 확률 즉, 조건부 확률을 산출하여 악성코드 탐지에 유의미한 Feature를 선정하고, 이를 바탕으로 머신러닝 기술을 활용하여 악성코드를 탐지하는 방법을 제시한다.

II. 관련 연구

2.1 악성코드 분석

악성코드를 분석하는 방법으로는 크게 정적 분석과 동적 분석으로 나눌 수 있다. 정적 분석은 역공학을 통해 악성코드 파일을 실행시키지 않고 악성코드 구성 요소들의 연관성이나 호출

* 이 성과는 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2018R1A4A1025632).

관계 등을 추출하여 전체적인 구조 및 흐름을 분석하는 방법이며, PE 파일의 IAT(Import Address Table)에 포함된 API 정보를 추출하여 악성코드가 어떤 행위를 수행하는지 파악할 수 있다[4]. 동적 분석은 악성코드가 실제로 실행되었을 때 수행되는 내용을 분석하는 방법이며, 가상 머신 상에서 프로세스, 파일, 레지스트리 생성 및 수정, 네트워크 이용, API 호출 등을 실시간으로 분석하는 방법이다[4]. 동적 분석 방법은 악성코드가 동작하기 위한 조건을 만족하지 않을 경우 실제 악성코드의 기능에 대한 분석이 어려우며, 정적 분석에 비해 많은 분석 시간이 필요하다. 따라서 본 논문에서는 정적 분석을 통해 IAT(Import Address Table)을 탐색하였으며, IAT에 포함된 API/DLL을 추출하여 Feature로 활용하였다[5].

2.1 조건부 확률 기반 Feature selection

Feature selection은 모델의 정확도를 향상시키거나 Learning에 필요한 시간을 단축시키기 위해 원본 Feature 집합 중에서 가장 좋은 분류 성능을 보이는 부분 집합을 탐색하는 과정이다. 일반적으로 일변량 통계, 모델 기반 선택, 반복적 선택 등이 존재하지만, 독자적인 Feature selection 방법도 존재한다. M. Edkrantz[6]은 머신러닝을 활용하여 공개적으로 알려진 소프트웨어의 보안취약점 목록인 CVE(Common Vulnerabilities and Exposures) 중 Exploit 사례가 없는 CVE가 Exploit 될 가능성을 예측하는 연구를 진행하였다. 특히, CVE로부터 추출한 Feature의 중요도를 평가하기 위해 자체적으로 산출한 조건부 확률 즉, 각각의 Feature가 CVE Exploit에 활용될 확률을 기반으로 Feature의 중요도를 산출하여 중요도가 높은 Feature 부분 집합을 선정하였으며, 두 개의 정답 label을 갖는 Feature의 조건부 확률은 아래 [수식 1]과 같은 방법으로 산출된다[6].

$$P = \frac{\frac{\text{Feature가 label 1에 등장한 횟수}}{\text{label 1의 샘플수}}}{\frac{\text{Feature가 label 1에 등장한 횟수}}{\text{label 1의 샘플수}} + \frac{\text{Feature가 label 0에 등장한 횟수}}{\text{label 0의 샘플수}}}$$

[수식 151] 조건부 확률 계산 공식

[수식 1]을 통해 특정 Feature가 label 1에 출현할 확률을 구함으로써 이를 Feature의 중요도 평가에 활용할 수 있다. 본 논문에서도 이와 같은 공식을 통해 추출한 API/DLL Feature의 확률을 산출하여 Feature의 중요도를 평가하는데 활용하였다.

III. 실험 환경

3.1 Data set

본 논문에서는 KISA와 한국정보보호학회가 공동개최한 2018 정보보호 R&D 테이터 챌린지 대회에 사용된 정상/악성파일 5세트 중 정답 Label이 공개된 3세트를 Data set으로 활용하였으며, 이 중 2세트는 Training set, 나머지 1세트는 Test set으로 사용하였다. 1세트는 악성파일 7,000개, 정상파일 3,000개, 총 10,000개의 샘플로 구성되어 있다.

3.2 머신러닝 도구

scikit-learn은 파이썬 기반 오픈소스 머신러닝 라이브러리이며, 지도/비지도 학습 알고리즘, 모델 성능 평가에 사용되는 Cross validation, Hyper-parameter 최적화에 사용되는 Grid-search 등 다양한 머신러닝 도구를 지원하고 있다[7]. 본 논문에서는 악성파일 샘플을 예측하기 위해 지도 학습 알고리즘인 Random forests, Naive bayes, Logistic regression 등을 구현 및 활용하였다.

IV. 실험

4.1 Feature 추출

정상파일 6,000개 악성파일 14,000개로 구성된 Training set 샘플로부터 원본 Feature 집합을 추출하기 위해 앞서 설명한 정적 분석 방법

과 파일 기반의 PEFile 라이브러리를 활용하여 API 65,493개, DLL 1,554개를 추출하였다.

4.2 Feature selection

가장 좋은 분류 성능을 보이는 Feature 부분집합을 탐색하기 위해 앞서 설명한 것과 같이 총 67,047개로 구성된 API/DLL Feature 그룹의 조건부 확률을 산출하였으며, 그 확률이 0%를 넘는 API 11,917개, DLL 335개, 총 12,252개의 Feature를 최종 Feature 그룹으로 선정하였다. 54,795개의 Feature가 감소함에 따라 Learning에 필요한 시간이 매우 줄어들고 통계적으로 유의미한 API/DLL Feature를 확보하였다. 확률이 높고 악성 샘플에 등장한 횟수가 많은 API/DLL Feature들은 각각 [표 1], [표 2]와 같다.

[표 54] 상위 10개 API Feature

API	확률	악성 샘플 등장 횟수	정상 샘플 등장 횟수
getprocaddress	0.5371	9,567	3,958
getmodulehandlea	0.5912	9,124	3,029
exitprocess	0.5984	8,758	2,822
getlasterror	0.4964	7,945	3,870
closehandle	0.5024	7,705	3,664
writefile	0.5299	7,631	3,250
sleep	0.5047	7,528	3,547
loadlibrarya	0.5466	6,758	2,691
freelibrary	0.5414	6,722	2,733
getmodulefilenamea	0.5838	6,677	22,85

[표 55] 상위 10개 DLL Feature

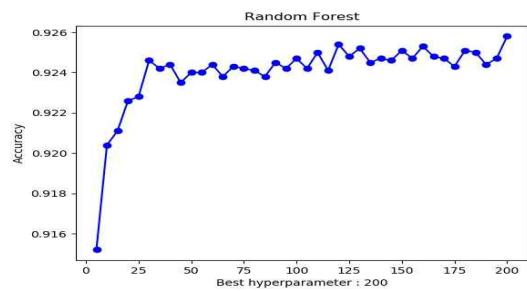
DLL	확률	악성 샘플 등장 횟수	정상 샘플 등장 횟수
kernel32.dll	0.5837	15,401	5,424
user32.dll	0.5760	10,644	3,869
advapi32.dll	0.5636	9,010	3,445
shell32.dll	0.6156	6,644	2,049
ole32.dll	0.5597	5,682	2,208
oleaut32.dll	0.5955	5,438	1,824
gdi32.dll	0.5660	5,382	2,038
comctl32.dll	0.5231	4,368	1,967
version.dll	0.5902	3,350	1,149
msvbm60.dll	0.8498	2,109	184

4.3 Hyper-parameter 최적화

Hyper-parameter는 머신러닝 알고리즘 학습 과정에 인위적인 영향을 줄 수 있는 매개변수를 의미한다. 이 값에 따라 전혀 다른 모델이 생성되므로 모델의 성능을 최적화하기 위해 적당한 Hyper-parameter를 탐색할 필요가 있다.

Hyper-parameter 최적화를 위해 Training set 을 사용하였으며, Default hyper-parameter 값 을 기준으로 탐색할 구간을 설정하고 10-fold Cross validation을 통해 가장 높은 정확도를 도출하는 Hyper-parameter 값을 탐색하였다.

Random Forests 알고리즘의 Hyper-parameter인 n_estimators는 알고리즘을 구성하는 Decision tree를 개수를 설정하는 값이며, 해당 값을 5부터 200까지 5씩 증가시켜 최적값인 200을 탐색하였고, 그 결과는 아래 [그림 1]과 같으며, 나머지 알고리즘도 동일한 방법으로 탐색하였다.



[그림 98] Random forest hyper-parameter 최적화 결과

Naive bayes 알고리즘의 Hyper-parameter인 alpha는 모델의 복잡도를 조절하는 값이며, 값이 클수록 복잡도가 감소한다. 해당 값을 0.1부터 2.0까지 0.1씩 증가시켜 최적값인 0.1을 탐색하였다.

Logistic regression 알고리즘의 Hyper-parameter인 C는 모델의 분류 오류에 대한 Penalty의 수준을 설정하는 값이며, 해당 값이 클수록 모델이 Training set에 최적화됨에 따라 Over-fitting 위험성이 증가한다. 해당 값을 0.1부터 2.0까지 0.1씩 증가시켜 최적값인 1.3을 탐색하였다.

4.4 최종 결과

머신러닝 분류 모델의 성능 평가 지표에는 Accuracy, Precision, Recall, F1-score 등이 있으며, 각 지표들이 의미하는 바는 다음과 같다. Accuracy는 정답 label 구분 없이 모든 Test

set 샘플 중 정확히 정답을 예측한 샘플의 비율이며, 탐지율을 의미한다. Precision은 악성파일로 예측한 샘플 중 실제 악성파일의 비율이며, 정확도를 의미한다. Recall은 실제 악성파일에 속한 샘플 중 악성파일로 예측한 샘플의 비율이며, 재현율을 의미한다. F1-score는 Precision 및 Recall의 조화 평균이다.

앞서 최종 선정한 Feature 부분 집합과 Hyper-parameter 값을 기반으로 최적의 모델을 생성한 후 정상파일 3,000개 악성파일 7,000개로 구성된 Test set을 분류한 결과는 아래 [표 3], [표 4]와 같다.

[표 56] 모델별 Accuracy, Precision 결과

알고리즘	Data set 종류	Accuracy	Precision
Random forest	Train	0.9575	0.9618
	Test	0.9201	0.9320
Naive bayes	Train	0.7281	0.8490
	Test	0.7209	0.8463
Logistic regression	Train	0.9513	0.9555
	Test	0.9164	0.9309

[표 57] 모델별 Recall, F1-score 결과

알고리즘	Data set 종류	Recall	F1-score
Random forest	Train	0.9781	0.9699
	Test	0.9556	0.9436
Naive bayes	Train	0.7439	0.7930
	Test	0.7347	0.7866
Logistic regression	Train	0.9759	0.9656
	Test	0.9511	0.9409

Random forest 알고리즘이 전체적으로 우수한 성능을 보이고 있으며, 92%의 탐지율을 기록하였다.

V. 결론

본 논문에서는 30,000개의 정상/악성파일로부터 추출한 API/DLL의 조건부 확률을 산출하여 악성파일에 높은 확률로 등장하는 API/DLL을 선정하였으며, 이를 바탕으로 머신러닝 기법을 활용해 악성코드를 탐지하는 방법을 제안하였다. 그 결과 Random forest 알고리즘이 가장 우수하였으며, Precision의 결과 즉, 악성파일로

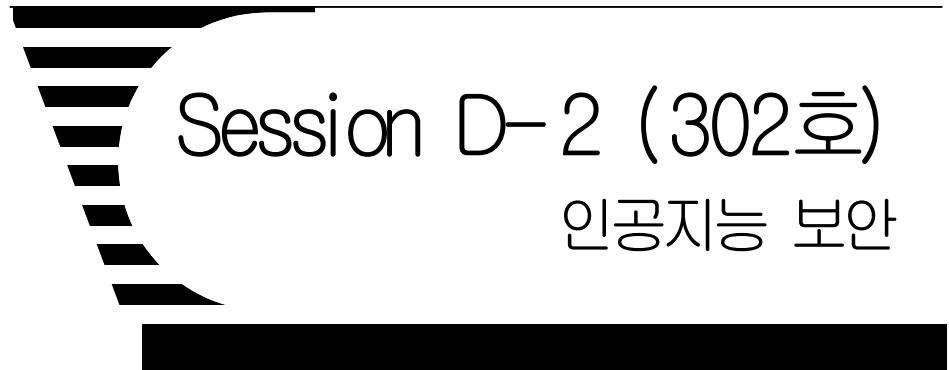
예측한 샘플 중 실제 악성파일의 비율이 93%인 것을 고려했을 때, 해당 모델은 충분히 악성코드에 대응할 수 있을 것으로 기대된다. 향후에는 더 다양한 Feature 그룹을 추가하여 모델의 성능을 향상시킬 계획이며, 각 그룹의 조건부 확률을 산출하여 다양한 악성코드 분석 및 탐지 연구에 기여하고자 한다.

Acknowledgement

본 결과(연구)는 한국인터넷진흥원(KISA)에서 운영하는 정보보호 R&D 데이터셋 [대용량 악성·정상코드 III(데이터 챌린지 2018)]을 활용하여 작성되었음.

[참고문헌]

- [1] AV-TEST, Malware Statistics, <https://www.av-test.org/en/statistics/malware/>, January, 2019.
- [2] KISA, 7대 사이버 공격 전망 2019, December, 2018.
- [3] KISA, 악성코드 익스체인지 탐지 동향 보고서 ['18년 하반기], January, 2019
- [4] 강부중, 한경수, 임을규, 악성코드 현황 및 탐지 기술, 한국정보과학회, 정보과학회지 30(1), January, 2012
- [5] 박재우, 문성태, 손기욱, 김인경, 한경수, 임을규, 김일곤, 문자열과 API를 이용한 악성코드 자동 분류 시스템 보안공학연구지원센터, 보안공학연구논문지, 8(5), October, 2011
- [6] M. Edkrantz, Predicting Exploit Likelihood for Cyber Vulnerabilities with Machine Learning, Chalmers University of Technology, Gothenburg, Sweden <http://publications.lib.chalmers.se/records/fulltext/219658/219658.pdf>, 2015
- [7] scikit-learn, User Guide, "https://scikit-learn.org/stable/user_guide.html"



좌장 : 신 원 (동명대)

Intrusion Detection Classifier Using Feature Selection and Recurrent Neural Network

Thi-Thu-Huong Le, Yongsu Kim, Howon Kim
Pusan National Univ.
lehuong7885@gmail.com, dkgoggog0329@gmail.com,
howonkim@pusan.ac.kr

Abstract

In this paper, we propose a classifier to improve intrusion detection accuracy in network security field. The proposed method bases on feature selection technique in feature engineering and recurrent neural network in deep learning. We perform experiment on the intrusion benchmark dataset, NSL-KDD 2012. The experimental result illustrate this work obtains significantly performance improvements.

Keywords. IDS, feature selection, SFS, KNN, NSL-KDD, RNN.

I. Introduction & Related Works

In recent years, intrusion detection system (IDS) becomes more and more attractive many researchers contributed in this field. The goal of IDS can detect malicious activities in security network. Besides, it identifies unauthorized use, misuse, and abuse of computer system by both internal and external system. Hence, IDS is security system used to monitor, recognize, and report malicious activities or policy violations. IDS uses different classification of attack types including probe, remote to local (R2L), user to root (U2R), and denial of service (DoS).

In traditional IDS, there are two type of IDS, including anomaly detection and misuse detection. Liao et al. [1] proposed a classification system consisting of five sub-classes such statics-based,

pattern-based, rule base, state based, and heuristic based. In statistic base, several methods are proposed such as distance based by Sabahi and Movaghfar [2], Bayesinan based by Stavroulakis and Stamp [3], game theory by Li et al. [4]. However, the existing limitation are less accuracy, poor control. In pattern based, the method are used including petrinet by Lazarevic et al. [5], keystroke monitoring by Murali and Rao [6]. However, these techniques are simple but less flexible. In rule based, there are some methods are applied such data mining by Xie et al. [7], SVM by Kolias et al [8]. However, these methods are not easily created and updated. In state based, the techniques are applied including user intention identification by Lazarevic et al. [5], Markov process model by Li et al. [4], and protocol analysis by Stavroulakis and Stamp [5]. However, these methods still

less effective and low false positive rate. In heuristic based, fuzzy logic by Mar et al. [9], genetic algorithm by Garcia et al [10] are proposed. However, these methods have fault tolerant, scalable which are problems.

In recently, deep learning field is widely used in IDS with improving accuracy performance compare to previous methods. In this work, we propose new IDS classifier including two processes. The first, applying feature selection technique to select the best subset feature by reducing high dimensional data. This technique is Sequential Forward Selection (SFS). The second, using the selected feature subset to train and test in Recurrent Neural Network (RNN) model.

II. Background

2.1 SFS

SFS is a simplest greedy search algorithm. It is a bottom-up search procedure. SFS start with an empty feature subset reaches a desired size. For every iteration which is inclusion of a new feature, the whole feature subset is evaluated which is expect for the features that are already included in the new subset. The evaluation is done by criterion function which assesses the feature that leads to the maximum performance improvement of the feature subset if it is included. To generate and evaluate feature subset, we combine SFS with machine learning model. In this work, we choose KNN model. By SFSKNN, we can select the best feature subset based on accuracy and error.

2.2 RNN

RNN is sequence model in deep learning field. There is also feedback loop connecting a neuron to itself. This model have a memory that captures the information which

has been compute so far. There are three layers in RNN model such as input layer, hidden layer, and output layer. The architecture of RNN is shown in Fig.1.

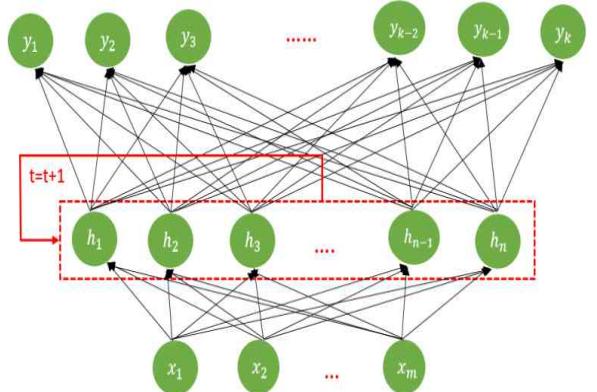


Fig 1. RNN model by time

III. Experiments

3.1 NSL-KDD dataset

We use a real-world datasets in IDS as NSLKDD [12]. To evaluate our approach on dataset, we used confusion matrix to measurement classification for each attack. NSLKDD is the new version of KDD Cup'99 dataset. This dataset is published available on their website. The authors analyzed that, in KDD Cup, train and test sets are duplicated about 78% and 75% of the records, respectively. Therefore, they did redundant records in train set and no duplicate records in new test sets. Although this dataset overcomes some limits in old dataset, it is going to keep 41 features. As a result, the number of records in train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small partition.

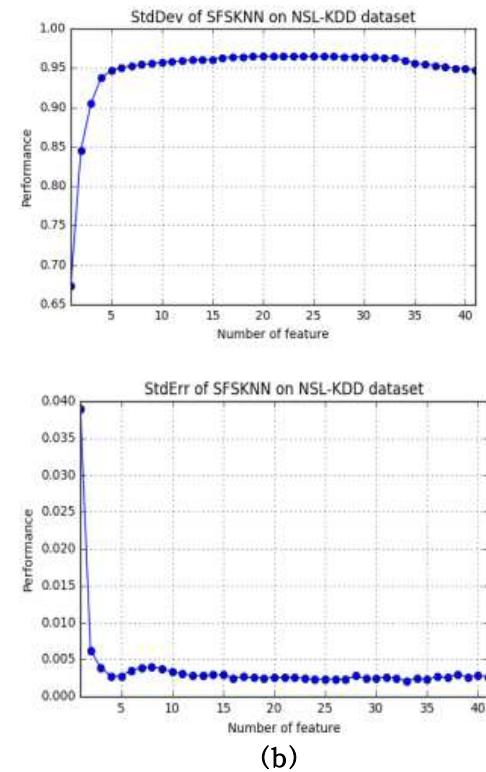
This dataset has 41 features corresponding 41-dimensional data. They are [duration, protocol_type, service, flag, src_bytes, land,

wrong_fragment, urgent, hot,
 num_failed_logins, logged_in,
 num_compromised, root_shell, su_attempted,
 num_root, num_file_creations, num_shells,
 num_access_files, num_outbound_cmds,
 is_host_login, is_guest_login, count,
 srv_count, serror_rate, srv_serror_rate,
 rerro_rate, srv_rerror_rate, same_sr_rate,
 diff_srv_rate, srv_diff_host_rate,
 dst_host_count, dst_host_sr_count,
 dst_host_same_srv_rate,
 dst_host_diff_srv_rate,
 dst_host_same_src_por_rate,
 dst_host_srv_diff_host_rate,
 dst_host_serror_rate, dst_host_srv_serro_rate,
 dst_host_rerror_rate, dst_host_srv_rerror_rate].
 Besides, five label classes (output data) consists of four types of attack and non-attack corresponding DoS, Probe, R2L, U2R, and Normal.

3.2 Experimental Results

We perform two experiments. The first is to generate feature subset selected. The best feature subset is selected based on the maximize of accuracy (StdDev) and the minimize of loss (StdErr) of SFSKNN method. The second is to evaluate our prosed model.

(a)



(b)

Fig 2. (a) accuracy (StdDev) and (b) loss (StdErr) of proposed method SFSKNN.

Fig 2 (a) and (b) present StdDev and StdErr of propose method. The best accuracy and loss at 20 features combination with 96.48 % (accuracy) and 0.25% (loss). The best feature subset is generated including [duration, protocol_type, service, flag, src_bytes, dst_bytes, is_guest_login, wrong_flagment, urgent, su_attempted, count, srv_count, serror_rate, same_srv_rate, diff_srv_rate, num_root, num_file, is_host, num_failed_logins].

Besides, Fig 3 (a) shows the classification result for each attack in the dataset. The average accuracy of the proposed model achieved 98.556%.

IV. Conclusion

In this paper, we introduce a IDS classifier

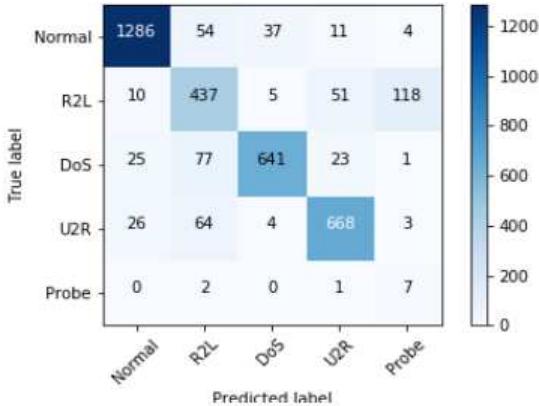


Fig 3. The classification for each attack of proposed SFSKNN+RNN model

to improve the performance of conventional RNN by embedding SFSKNN into the training procedure. Experiments with SFS and RNN achieves remarkable performance improvements in IDS field.

[ACKNOWLEDGMENT]

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (2014-1-00743) supervised by the IITP (Institute for Information & communications Technology Promotion).

[References]

- [1] H.J Liao, C.H.R Lin, Y.C. Lin, K.Y Tung, "Intrusion detection system: A comprehensive review", Journal of Network and Computer Applications, 2013, pp. 16–22.
- [2] Sabahi F, Movaghar A, "Intrusion detection: a survey", in Third international conference on system and network communication, Sliema, Malta, 2008, pp. 23–26.
- [3] Stavroulakis P, Stamp M, Handbook of information and communication security. New York: Springer-Verlag, 2010.

[4] L Li, Zhang G, Nie J, Niu Y, Yao A, "The application of genetic algorithm to intrusion detection in MP2P network", in Third international conference on advances in swarm intelligence, Shenzhen, China, 2012, pp. 390–397.

[5] Lazarevic A, Kumar V, Srivastava J, "Managing cyber threats: issues, approaches, and challenges", New York:Springer-Verlag, 2005.

[6] Murali A, Rao M, "A survey on intrusion detection approaches", in First international conference information and communication technologies, Karachi, Pakistan, 2005, pp. 233–240.

[7] Xie M, Han S, Tian B, Parvin S, "Anomaly detection in wireless sensor networks: a survey", Journal of Network and Computer Applications, 2011; pp. 1302–25.

[8] Kolias C, Kambourakis G, Maragoudakis M, "Swarm intelligence in intrusion detection: a survey", Computers & Security, 2011, pp. 625–42

[9] Mar J, Hsiao IF, Yeh YC, Kuo CC, Wu SR, "Intelligent intrusion detection and robust null defense for wireless networks", International Journal of Innovative Computing Information and Control, 2012; vol.8, pp. 3341–59.

[10] Garcia-Teodoro P, Diaz-Verdejo J, Macia-Fernandez G, Vazquez E, "Anomaly-based network intrusion detection: techniques, systems and challenges", Computers & Security, 2009, pp.18–28.

딥러닝 기반 얼굴인식 기술의 동향*

권대용* 최대선†

*공주대학교 의료정보학과

dykwon1101@smail.kongju.ac.kr

Trends in Deep Learning-Based Face Recognition

Dae-Yong Kwon* Dae-Sun Choi†

*Division of medical information, Kongju University.

요약

2014년 Deepface의 등장은 얼굴인식 분야에서 큰 파장을 가져왔다. 이미지의 해상도, 포즈, 조명 등의 문제로 실질적으로 적용하기 힘들다는 기존의 얼굴인식 기술의 한계점에 돌파구를 제시한 것이다. 얼굴인식 기술에 딥러닝이 도입되면서 얼굴인식 분야는 획기적인 변화를 겪었고, 딥러닝 기반의 얼굴인식 기술은 지금도 빠르게 발전하고 있다. 본 논문에서는 최근 몇 년간에 딥러닝 기반 얼굴인식 기술의 발전과정과 대표적인 알고리즘을 소개하고자 한다.

I. 서론

얼굴 인식(Face Recognition)[1]은 사람의 얼굴이 포함된 이미지나 영상에서 얼굴 영역을 검출한 후 어떤 사람의 얼굴인지를 판별하는 기술을 말한다. 얼굴 인식 기술은 지난 수십년 간 연구되었고 공공 안보, 일상생활, 생체 인증 등 많은 분야에서 널리 사용되어왔다. 하지만 이전의 얼굴 인식 기술에는 해결해야 할 중대한 문제들이 몇 가지 존재했다. 이미지 혹은 영상 속에서 사람의 얼굴을 판별 또는 분석하는 데 있어 그 사람의 포즈나 조명, 해상도등에 따라 성능이 크게 좌우됐다. 또한, 기존의 얼굴 인식 기술로는 복잡한 비선형으로 이루어진 얼굴에서 특징을 추출하는 데 한계가 있었다.

2014년, 처음으로 딥러닝(Deep Learning)[2]을 접목한 얼굴 인식 기술이 발표되어 LFW 데이터셋에서 97.35%의 정확도를 달성함으로써 당시의 얼굴 인식 기술에 비해 크게 향상된 인식률을 선보였다. 딥러닝이란 머신러닝 기법의 일종으로 다층으로 구성된 인공 신경망을 이용해 대용량의 데이터를 이용해 학습시키는 것으로, 딥러닝을 복잡한 비선형적인 풀 수 있다. 딥러닝을 얼굴 인식 기술에 접목함으로써 다양한 이미지에서 더욱 뛰어난 성능으로 사람의 얼굴을 식별 또는 구분할 수 있게 된다.

본 논문에서는 대표적인 딥러닝 기반의 얼굴 인식 기술들과 그 성능, 그리고 최신의 딥러닝 기반 얼굴 인식 기술들을 소개하고자 한다.

II. 딥러닝 기반 얼굴 인식 기술

2.1 DeepFace

2014년, FaceBook에서 발표한 DeepFace[3]는

* 본 연구는 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No.2016-0-00173, 펀테크 서비스 금융사기 방지를 위한 비대면 본인확인)을 받아 수행

† 교신저자(Corresponding author)

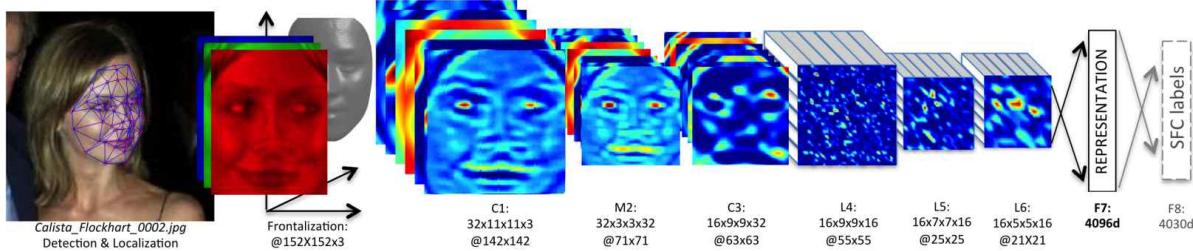


그림 105. DeepFace 아키텍처 개요

최초로 얼굴인식 기술에 딥러닝을 접목한 연구다. DeepFace는 LFW 데이터셋에서 97.35%에 획기적인 정확도를 선보였다. 사람이 이미지를 봤을 때의 정확도는 97.53%로, 최초로 제한 없는 조건 하에서 인간의 성능에 근접하였다.

DeepFace에서는 이미지 속의 얼굴 영역에 3D 모델링을 통해 다른 방향을 보고 있는 얼굴을 정면으로 정렬한다. 그 후 국소 연결 컨볼루션 계층(Locally-connected layer)을 포함한 9개 층의 CNN을 이용해 학습 시킨다(그림1).

DeepFace는 뛰어난 성능을 선보이지만 매우 많은 양의 데이터와 컴퓨팅 파워를 요구한다는 한계점을 가진다. Facebook은 DeepFace를 학습시키는데 자체적으로 수집한 400만 장 이상의 얼굴 이미지를 사용된다. 또한 심층 신경망에 1억 2천개 이상의 파라미터가 사용되어서 학습시키는데 매우 큰 컴퓨팅 파워를 요구한다.

2.2 VGGFace

VGGFace[4]는 옥스퍼트 대학의 Visual Geometry Group에서 제안하였다. 인터넷에서 대규모 데이터셋을 수집한 후, 이 데이터셋으로 VGGNet을 학습시킨 후 triplet loss function을 통해 네트워크를 조정한다. VGGNet은 아주 작은 3×3 컨볼루션 필터를 사용하고 2×2 풀링 후 피쳐맵의 수를 두 배로 늘렸다는 특징이 있다. 이에 따라 네트워크의 층 수가 16~19개로 증가했고 비선형적인 매핑에 더욱 견고해졌다. VGGFace는 LFW 데이터셋에 대해 98.95%의 정도를 보인다.

2.3 ResNet

ResNet[5]은 2016년에 등장한 신경망 아키텍처로 현재까지도 가장 높은 성능과 범용성을

보이고 있다. ResNet은 Gradient Vanishing, 연산량의 과도한 증가 등으로 레이어의 수가 늘어날수록 성능이 떨어지는 타 네트워크의 단점을 개선했다. 이는 ResNet의 가장 큰 특징인 ‘Residual Block’의 도입으로 가능해졌다. 즉, 레이어의 입력을 추가로 뺀 Residual Function $F(x) = H(x) - x$ 를 최소화한다(그림 2).

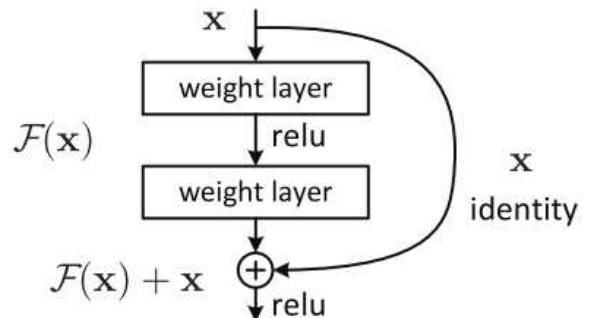


그림 106. Residual learning

손실 함수 $F(x)$ 가 0이 되는 것이 목표이므로 $H(x)$ 를 x 로 매핑시키는 것이 학습의 목표가 된다. 2018년 100개 층으로 구성된 ResNet 아키텍처를 사용한 Arcface 연구는 LFW 데이터셋에 대해 99.83%에 정확도를 선보였다.

2.4 얼굴인식 기술 전반

딥러닝 기반 얼굴 인식 기술이 높은 성능을 보이기 위해서는 많은 부분을 고려해야 한다. 신경망 아키텍처를 구성하고, 적절한 레이어의 개수를 정해야 한다. 또한 모델의 구조에 따라서는 하나의 네트워크가 아닌 다중의 네트워크를 사용할 수도 있다. 신경망을 구성한 후에는 분별력 있게 이미지에서 피쳐를 학습하기 위한 적절한 loss function을 정의한다. loss function은 동일 인물로부터 추출된 특징의 분산은 작

Method	Public.Time	Loss	Architecture	Numbr of Networks	Traing Set	Accuracy
DeepFace	2014	softmax	Alexnet	3	Facebook (4.4M,4K)	97.35±0.25
DeepID2	2014	contrastive loss	Alexnet	25	CelebFaces+ (0.2M,10K)	99.15±0.13
DeepID3	2015	contrastive loss	VGGNet-10	50	CelebFaces+ (0.2M,10K)	99.53±0.10
FaceNet	2015	triplet loss	GoogleNet-24	1	Google (500M,10M)	99.63±0.09
Baidu	2015	triplet loss	CNN-9	10	Baidu (1.2M,18K)	99.77
VGGface	2015	triplet loss	VGGNet-16	1	VGGface (2.6M,2.6K)	98.95
light-CNN	2015	softmax	light CNN	1	MS-Celeb-1M (8.4M,100K)	98.8
Center Loss	2016	center loss	Lenet++7	1	CASIA-WebFace, CACD2000, Celebrity+ (0.7M,17K)	99.28
L-softmax	2016	L-softmax	VGGNet-18	1	CASIA-WebFace (0.49M,10K)	98.71
Range Loss	2016	range loss	VGGNet-16	1	MS-Celeb-1M, CASIA-WebFace (5M,100K)	99.52
L2-softmax	2017	L2-softmax	ResNet-101	1	MS-Celeb-1M (3.7M,58K)	99.78
Normface	2017	contrastive loss	ResNet-28	1	CASIA-WebFace (0.49M,10K)	99.19
CoCo loss	2017	CoCo loss	-	1	MS-Celeb-1M (3M,80K)	99.86
vMF loss	2017	vMF loss	ResNet-27	1	MS-Celeb-1M (4.6M,60K)	99.58
Marginal Loss	2017	marginal loss	ResNet-27	1	MS-Celeb-1M (4M,80K)	99.48
ShereFace	2017	A-softmax	ResNet-64	1	CASIA-WebFace (0.49M,10K)	99.42
CCL	2018	center invariant loss	ResNet-27	1	CASIA-WebFace (0.49M,10K)	99.12
AMS loss	2018	AMS loss	ResNet-20	1	CASIA-WebFace (0.49M,10K)	99.12
Cosface	2018	cosface	ResNet-64	1	CASIA-WebFace (0.49M,10K)	99.33
Arcface	2018	arcface	ResNet-100	1	MS-Celeb-1M (3.8M,85K)	99.83
Ring loss	2018	Ring loss	ResNet-64	1	MS-Celeb-1M (3.5M,31K)	99.50

표1. LFW 데이터셋에서의 얼굴 인식 기술별 정확도

개하고, 다른 인물로부터 추출된 특징의 분산은 크게 하는 것이 바람직하다. loss function은 앞서 VGGNet에서 사용한 triplet loss function을 비롯해 contrastive loss function, softmax를 개선한 여러 loss function 등이 연구되고 있다. Table 1은 LFW 데이터셋에 대한 딥러닝 기반 얼굴 인식 기술들 별 정확도를 보여준다.

III. 결론

본 논문에서는 딥러닝 기반 얼굴인식 기술의 연구 동향에 대해 살펴보았다. 딥러닝을 얼굴인식 기술에 접목시킴으로써 기존에 비해 높은 성능을 보이고 있다. DeepFace를 필두로, 사람보다 정확하게 이미지나 영상에서 얼굴을 검증 또는 식별할 수 있으며, 얼굴의 방향이나 조명 변화 등 얼굴 인식에 장애가 되는 요인들에 강인한 기술들 역시 등장하고 있다. 하지만 실제 환경의 다양한 조건 속에서 완벽하게 한계점을 극복했다고 보기는 어렵다. 따라서 지속적인 관심과 연구가 필요하다.

[참고문헌]

- [1] Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, "Face recognition: A literature survey", ACM computing surveys (CSUR), Dec, 2003
- [2] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton, "Deep learning", nature, May, 2015
- [3] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, Lior Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification", IEEE, 2014
- [4] R. Ranjan, S. Sankaranarayanan, A. Bansal, N. Bodla, J. C. Chen, V. M. Patel, C. D. Castillo, and R. Chellappa. Deep learning for understanding faces: Machines may be just as good, or better, than humans. IEEE Signal Processing Magazine, 2018.
- [5] K. He, X. Zhang, S. Ren, and J. Sun. "Deep residual learning for image recognition", CVPR, 2016.

인공지능 시스템 보안 위협과 대책²⁾

박호성*, 최대선*□

*공주대학교

hspark@kongju.ac.kr

Security Threats and Measures for Artificial Intelligence System

Hosung Park*, Daeseon Choi*□

*Kongju National University.

요약

인공지능이 생활과 산업 전반에 활용될 것으로 예상됨에 따라 인공지능 시스템의 보안 취약점 및 그 대책에 관한 관심이 높아지고 있다. 인공지능의 오작동을 유발할 수 있는 다양한 공격 방식들이 제안되고 있으며, 보안 사고 사례들과 실험 결과들 역시 속속 등장하고 있다. 이러한 보안 위협에 대응하기 위한 방어 전략들 역시 활발히 연구되고 있지만 아직은 충분한 대책들이 마련되었다고 보기 어렵다. 본 논문에서는 인공지능 시스템을 향한 대표적인 보안 위협들과 그 대책에 대해 알아본다. 특히, 보안 대책은 실질적인 대응을 위해 기술적인 방법뿐만 아니라 정책적인 방법들을 포함한다.

I. 서론

인공지능의 눈부신 발전에 따라 인공지능 시스템의 보안 취약점에 관한 관심 역시 높아지고 있다. 이미 인공지능의 오작동을 유발할 수 있는 다양한 공격 방식들이 제안되었으며, 실제 보안 사고 사례들이 발생하고 있다. 향후 인공지능이 금융, 의료, 생산과 같은 분야에서 더 중요한 역할을 담당하게 될수록 보안 취약점은 더 큰 위협으로 다가오게 될 것이다.

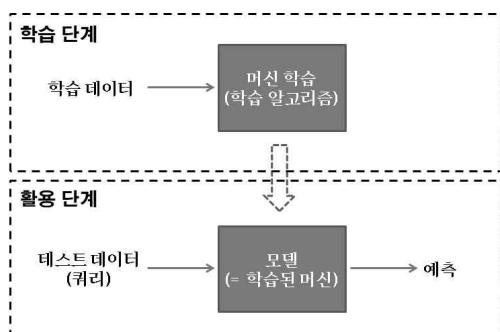


그림 107. 인공지능 시스템 동작 과정

2) 본 연구는 2019년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No.2016-0-00173, 핀테크 서비스 금융사기 방지를 위한 비대면 본인확인)을 받아 수행 □ 교신저자 (corresponding author)

인공지능 시스템은 그림 1과 같이 크게 학습 단계와 활용 단계로 구분할 수 있다. 학습 단계는 머신이 다양한 학습 데이터를 학습 알고리즘을 통해 스스로 학습하는 과정이며, 활용 단계에서는 학습을 마친 머신 즉, 모델을 사용하여 실제 서비스를 제공하는 단계를 뜻한다. 인공지능 시스템에 대한 보안 위협은 공격 시점에 따라 학습 단계에서의 공격과 활용 단계에서의 공격으로 분류할 수 있다. 각 단계에서 대표적인 공격 방법들과 그에 대한 보안 대책들을 소개한다. 보안 대책은 공격에 대응하기 위해, 현 상황에서 가능한 기술적인 방법들과 정책적인 방법들을 포함한다.

II. 인공지능 보안 위협

2.1 학습 단계 공격 (오염 공격)

그림 2는 학습 단계에서의 공격 과정을 보여준다. 학습 단계에서의 공격은 공격자가 모델에 의도적으로 잘못된 학습 데이터를 주입함으로써 모델 자체를 오염시킨다. 오염된 모델은 활용 단계에서 오동작을 일으켜 잘못된 예측값을 출력하게 된다. 따라서, 오염(poisoning) 공격이

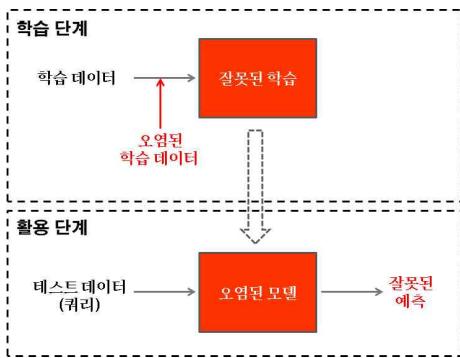


그림 108. 학습 단계에서의 공격

라고 부른다. 최소한의 오염 데이터 추가로 오작동을 최대화하는 것을 목적으로 한다[1]. 학습 데이터가 실시간으로 업데이트되는 서비스(active learning)가 오염 공격에 노출되기 쉽다.

2.2 활용 단계 공격 (회피 공격)

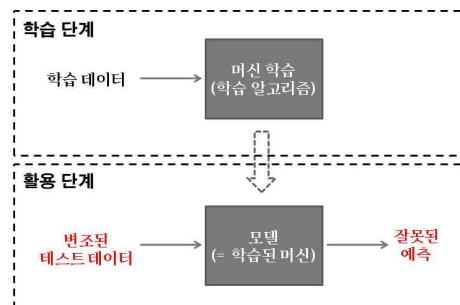


그림 109. 활용 단계에서의 공격

그림 3은 활용 단계에서의 공격 과정을 보여준다. 학습 단계에 대한 개입이 없기 때문에 모델은 오염되지 않았지만, 활용 단계의 테스트 데이터를 변조함으로써 AI 시스템의 오작동을 유발한다. 회피(evasion) 공격이라고 부른다. 최소한의 변조로 인간 및 탐지 시스템에 적발되지 않는 상태에서 모델이 해당 테스트 데이터에 대한 예측을 잘못하도록 하는 것을 목적으로 한다. 회피 공격을 가장 많이 적용하고 있는 분야는 이미지와 음성이다. 테스트 데이터의 변형을 노이즈 추가라는 손쉬운 방식으로 할 수 있기 때문이다. 현재 가장 많이 쓰이는 공격 기술은 Carlini-Wagner(CW)[2]이다.

III. 인공지능 보안 대책

3.1 학습 단계 보안 대책

학습 데이터 오염 방지를 위해 다음과 같은 보안 대책들이 필요하다.

첫째, 데이터의 출처를 명확히 해야 한다. 출처가 불분명한 데이터의 경우, 데이터의 오염 확률이 높고 데이터의 신뢰성이 떨어지기 때문이다. 데이터의 출처에 따른 데이터 신뢰성 판단 그리고 신뢰성에 따른 관리가 필요하다.

둘째, 오염 데이터 유입 방지 대책이 필요하다. 데이터 출처 인증, 데이터 전송 과정 위·변조 방지, 담당자에 대한 보안 교육 등의 데이터 수집 과정 보호와 인터페이스 통제, 원격 접속 통제, 네트워크 보안 등의 학습 데이터 접근 제어로 분류할 수 있다.

셋째, 오염 데이터 탐지 및 필터링이다. 데이터 학습 전에 오염 데이터를 탐지하고 제거하는 과정을 말하며, 기본적인 원리는 통계적 이상치를 찾는 것이다[3]. 다시 말해, 학습 데이터의 분포 및 양상과 다른 데이터를 오염 데이터로 판단하고 이를 제거한다. 또한, 예상 가능한 오염 데이터의 경우, 규칙(rule) 기반 필터링이 가능하다. 예를 들어 채팅 로봇의 경우 비속어, 부적절한 단어가 속한 학습 데이터는 미리 제거할 수 있다.

넷째, 인공지능 시스템 정보 유출을 방지해야 한다. 공격자는 인공지능 시스템의 정보들을 더 많이 알수록 더 정교한 오염 데이터를 생성할 수 있다. 따라서 인공지능 시스템의 정보 유출 방지는 공격 성공률을 떨어뜨릴 수 있는 효과적인 방어 전략이다. 보호할 필요성이 있는 정보들은 메타 데이터, 특징, 학습 데이터, 학습 레이블 등의 데이터 정보와 학습 알고리즘, 파라미터, 모델 구조, 상위 모델 정보 등의 모델 정보가 포함된다.

3.2 활용 단계 보안 대책

변조 데이터에 의한 회피 공격에 대응하기 위한 대책들은 다음과 같다.

첫째, 변조 데이터 학습을 통한 회피 공격 방지이다. 그림 4와 같이 기존의 학습 데이터와 변조 데이터를 함께 학습하여 활용 단계에서 변조 데이터가 입력되더라도 오동작을 유발하지 못하도록 하는 방법이다. 쉽게 말해 모델에

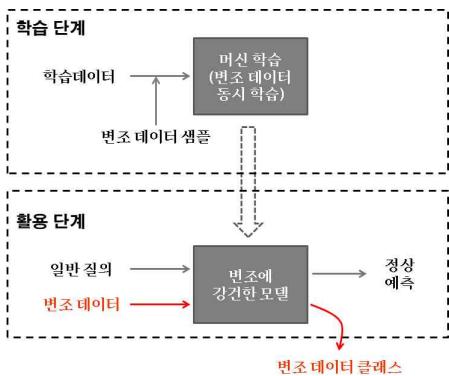


그림 110. 변조 데이터 학습을 통한 방어

제 변조 데이터의 유형을 알려주고, 변조 데이터가 질의로 입력되는 경우 별도로 구분할 수 있도록 해주는 방법이다. 단순하고 효과적인 방법으로 모델 본래 목적의 정확도를 떨어뜨리지 않는다는 장점을 갖는다. 더 강한 방어 기능을 제공하기 위해 다양한 변조 데이터를 미리 확보 혹은 생성하는 것이 중요하다.

둘째, 변조 데이터 탐지 및 필터링이다. 질의(테스트 데이터)가 모델에 입력되기 전에, 변조 데이터를 탐지하여 제거 혹은 역변조하여 회피 공격을 방어한다. 학습된 원본 데이터와 비교하여 왜곡이 심하다고 판단되는 데이터는 변조 데이터로 간주하여 제거한다. 왜곡 차가 비교적 적을 경우는 가장 가까운 원본 데이터를 찾아 역변조 함으로써 변조 데이터 입력 가능성을 줄인다. 변조 데이터 탐지의 기본 원리는 학습 데이터 분포와 다른 데이터 즉, 통계적 이상치 탐지(outlier detection)이다[4].

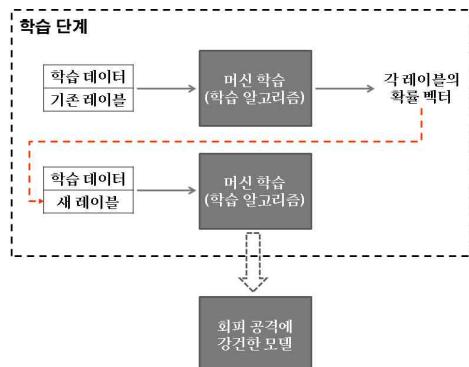


그림 111. 강건한 모델 생성(증류법)

셋째, 모델 변형을 통해 회피 공격에 강한 모델을 생성하는 방법이다. 기본 원리는 공격자가

변조를 위한 적절한 노이즈 값을 찾기 어렵게 모델을 변형하는 것이다[5][6]. 그림 5는 대표적인 모델 변형 방법인 증류법(distillation)[5]을 보여준다. 첫 번째 학습은 일반적인 학습법을 따르지만 두 번째 학습 단계를 추가하여 첫 단계에서 각 레이블의 확률값을 새로운 레이블로 사용하여 학습한다. 변조 데이터가 모델의 오작동을 유발할 만큼 충분히 정밀하지 못하게 생성되도록 유도한다.

넷째, 학습 데이터 오염 방지에서와 같은 이유로 인공지능 시스템 정보유출을 방지하여 공격자가 정밀한 변조 데이터를 생성하지 못하도록 한다.

IV. 결론

보안 위협과 그에 따른 방어가 항상 그려졌듯이 인공지능 시스템에 대한 보안 공격 및 방어도 서로를 극복하며 발전해가고 있다. 특히 현시점에서 인공지능 보안 대책은 아직 상용 서비스를 제공하기에 충분하다고 보기 어렵다. 따라서 안전한 인공지능 서비스를 위해서는 최신 기술들에 대한 지속적인 관심과 적용이 요구된다.

[참고문헌]

- [1] M. Luis, et al. "Towards poisoning of deep learning algorithms with back-gradient optimization," AISeC, Nov. 2017.
- [2] N. Carlini et al. "Towards evaluating the robustness of neural networks," IEEE Symposium on SP, May 2017.
- [3] J. Steinhardt, et al. "Certified defenses for data poisoning attacks," NIPS, Dec. 2017.
- [4] Tianyu Pang, et al. "Towards robust detection of adversarial examples," NIPS, Dec. 2018.
- [5] N. Papernot, et al. "Distillation as a defense to adversarial perturbations against deep neural networks," IEEE Symposium on SP, May 2016.
- [6] B. Wang, et al. "With Great Training Comes Great Vulnerability: Practical Attacks against Transfer Learning," USENIX, Aug. 2018.

딥러닝 기반 악성 PowerShell 스크립트 탐지 방안

송지현*, 최선오**, 김종현**, 김익균**

*과학기술연합대학원대학교 ICT(정보보호공학), **한국전자통신연구원

dmon95@naver.com, suno@etri.re.kr, jhk@etri.re.kr, ikkim21@etri.re.kr.

Deep Learning based Malicious Powershell Script Detection Method

Jihyeon Song*, Sunoh Choi**, Jonghyun Kim**, Ikkyun Kim**

*ICT(Information Security Engineering), University of Science and Technology, **Electronics and Telecommunications Research Institute.

요약

수년 전부터 PE 형태에서 Fileless 형태로 악성코드의 형태가 진화하고 있다. 사이버 공격자들은 Fileless 공격에 효율적인 PowerShell 스크립트를 사용하여 악성 행위를 하도록 하며, 공격에 사용되는 PowerShell 스크립트는 그 자체가 정상 스크립트일 수 있기 때문에 스크립트만으로 악성 여부를 판별하기가 어렵다. 기존에 악성 PowerShell 스크립트를 탐지하기 위한 방안이 몇 가지 존재하지만, 탐지를 우회하는 방법들이 공개되었고 그 외의 다른 방안에 대한 연구는 부족한 상황이다. 본 논문에서는 정상 및 악성 PowerShell 스크립트에서 텍스트 키워드를 추출하고 이를 특징으로 하여 딥러닝 모델에 적용해 악성 PowerShell 스크립트를 탐지할 수 있는 방안을 제시하고자 한다.

I. 서론

사이버 공격자들은 악성코드를 탐지 및 분석을 어렵게 하기 위해, 수년 전부터 PE 형태에서 Fileless 형태로 악성코드를 진화시키고 있다.

Fileless 악성코드는 악성 실행파일이 파일 시스템에 존재하지 않고 동작하는 악성코드로 JavaScript, PowerShell 스크립트 등을 사용해 악성 행위를 하며, 지속성을 유지하기 위해 레지스트리 키를 사용하는 등의 방법을 취하기도 한다[1]. JavaScript는 난독화 기술이 발전되어 있고, 취약한 웹사이트를 감염시켜 해당 사이트에 접속했을 때 악성 JavaScript를 실행하도록 하는 것이 가능하기 때문에 기존의 악성코드에서 많이 사용해왔다. 그러나 악성코드의 형태가 진화하면서, Fileless 공격에 효율적인 PowerShell 스크립트를 사용하는 비율이 증가하고 있다.

PowerShell은 Microsoft에서 개발한 명령 줄

인터페이스 쉘 및 스크립트 언어로, Windows XP 이상 대부분의 버전은 PowerShell을 지원한다. PowerShell을 사용하여 운영체제(Linux, macOS, Windows) 및 프로세스 관리를 할 수 있고, 레지스트리 및 인증서 등에 액세스하는 것이 가능하다[2]. 사이버 공격자의 입장에서 PowerShell을 이용하면 운영체제의 주요한 기능에 액세스 가능하며, 실행 혼적을 거의 남기지 않을 수 있기 때문에 분석가들의 공격 탐지 및 분석을 어렵게 할 수 있다. PowerShell 스크립트를 단독으로 사용하는 경우뿐만 아니라, JavaScript를 사용해 PowerShell을 실행하는 것이 가능하기 때문에 난독화 된 JavaScript에 PowerShell 스크립트를 숨겨놓는 때도 있다.

Fileless 공격에 사용되는 PowerShell 스크립트 자체는 정상 스크립트일 수도 있다. 따라서 스크립트 내용만으로 PowerShell 스크립트의 악성 여부를 판단하는 것은 정확하지 않다. 악성 PowerShell 스크립트를 탐지하는 방안은 일

부 존재하지만 이를 무력화할 수 있는 우회 방안들이 제시되었으며[6], 그 외에 다른 탐지 방안에 관한 연구가 부족한 상황이다.

본 연구에서는 PowerShell 스크립트로부터 텍스트 키워드를 추출하는 방법과 이렇게 추출된 특징 데이터를 사용하여 악성 PowerShell 스크립트를 탐지할 수 있는 딥러닝 모델을 제시하고자 한다.

II. 관련 연구

2.1 PDF 내 악성 JavaScript 탐지

기존의 악성코드에서는 JavaScript를 주로 사용해왔으며, 악성 JavaScript를 탐지하는 연구는 이미 활발하게 진행되었다. 이러한 연구는 악성 PowerShell 스크립트를 탐지하기 위한 연구의 참고사항이 될 수 있다.

Laskov와 Srndic의 연구[3]에서는 정적 분석 기반으로 JavaScript가 포함된 악성 PDF 문서를 탐지하기 위해 PDF 문서에 삽입된 JavaScript를 추출하여 코드를 토큰화하고, 그것을 특징으로 사용하였다. One-Class Support Vector Machine(OCSVM)을 사용해 학습시킨 후 악성 여부를 판단하였다.

2.2 악성 PowerShell 스크립트 탐지

악성 PowerShell 스크립트를 탐지하기 위한 연구는 일부 존재하며, PowerShell에서 실행된 명령 또는 명령 실행 순서를 기반으로 기계 학습을 통해 탐지하는 연구 등이 있다.

Microsoft와 Ben-Gurion 대학의 연구[4]에서는 Microsoft 기업 네트워크에서 실행된 PowerShell 명령에서 EncodedCommand를 사용해 인코딩된 명령을 디코딩하는 등의 전처리 과정을 거친 후, 어떤 문자가 명령에서 몇 번 사용되는지 횟수를 확인한다. 영문자가 아니거나 희귀한 문자는 제외하고 1.4% 이상의 빈도 수를 보이는 문자 집합(61자) 대소문자를 구분하기 위한 case bit(1자)를 합한 62 길이의 벡터를 딥러닝 모델에 적용하여 악성 PowerShell 명령을 탐지하였다.

고려대학교 정보보호대학원의 연구[5]에서는

명령 실행 모니터를 통해 PowerShell에서 실행되는 명령 및 인자들을 모니터링하고, 분류기에 전달하여 CNN을 사용해 특징을 추출한 후 RNN에 실행 순서대로 전달하여 악성 행위 여부를 판단한다.

앞서 소개한 Microsoft와 Ben-Gurion 대학의 연구는 PowerShell 명령의 일부 문자를(정적 특징) 특징으로 사용하였고, 고려대학교 정보보호대학원의 연구는 모니터링을 통해 얻은 PowerShell에서 실행되는 명령 및 인자들의 실행 순서(동적 특징)를 이용해 PowerShell 기반 악성코드를 탐지하였다.

이 외에 PowerShell의 이벤트 로그를 사용하여 악성 PowerShell 스크립트를 탐지하는 방안 등이 있지만, 이는 악성 PowerShell 스크립트 작성 시 이벤트 로그를 남기지 않도록 설정하여 쉽게 우회가 가능하며[6], PowerShell 스크립트를 이용한 공격에서 사용된 스크립트 자체는 정상일 가능성이 있기 때문에 스크립트 자체만으로 악성 여부를 판단하기는 것은 어려울 수 있다는 한계가 존재한다. 따라서 기계 학습을 통해 정상 및 악성 PowerShell 스크립트의 텍스트 키워드의 문맥을 이해하고 판별해내는 방안이 필요하다.

III. 실험 환경

3.1 전처리

정상 PowerShell 스크립트를 토큰 집합으로 구문 분석하도록 하는 [System.Management.Automation.PSParse::Tokenize]를 사용하여 [그림 1]과 같이 Type이 ‘Keyword’, ‘Variable’, ‘CommandParameter’, ‘Command’인 명령만 반환하도록 하였다. 그리고 그중에서 ‘Content’만을 추출하여 정상 스크립트의 텍스트 키워드로 사용하였다.

악성 OLE 파일은 데이터를 추출하기 전에 python-oletools 패키지의 olevba 모듈 중 vbaparser.analyze_macros()를 통해 파일을 구문 분석하여 [그림 2]와 같이 PowerShell을 통해 실행되는 악성 PowerShell 스크립트를 먼저 검색한 후, ‘Keyword’를 추출하였다[7].

정상 및 악성 샘플에서 추출한 텍스트 키워드를 모두 소문자로 변환한 후, 중복을 제거하고 시퀀스를 생성하였다. 그리고 [그림 3]과 같은 형식으로 학습 데이터를 구성하였다. 데이터의 내용은 ‘파일 이름, 1000(dummy), 1000(dummy), 시퀀스 개수, 악성 여부(정상은 0/악성은 1), 시퀀스1, 시퀀스2, ...’이다.

Content	:	param
Type	:	Keyword
Start	:	36
Length	:	5
StartLine	:	3
StartColumn	:	1
EndLine	:	3
EndColumn	:	6
Content	:	infile
Type	:	Variable
Start	:	52
Length	:	7
StartLine	:	3
StartColumn	:	17
EndLine	:	3
EndColumn	:	24

[그림 112] 정상 PowerShell
키워드

```
keyword=Auto_Open
keyword=Shell
keyword=powershell
keyword=net.webclient
keyword=downloadstring
keyword=new-object
keyword=http://192.168.63.135/payload.txt
keyword=192.168.63.135
keyword=powershell.exe
keyword=Shell
keyword=WScript.Shell
keyword=Powershell
keyword>CreateObject
keyword=Net.WebClient
```

[그림 113] 악성 PowerShell 키워드

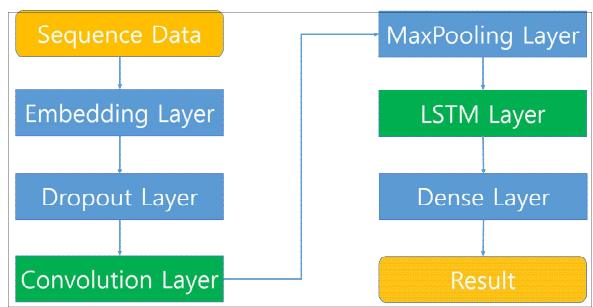
```
mal1, 1000, 1000, 7, 1, 1004, 1005, 1006, 73, 1007, 1008, 1009
mal10, 1000, 1000, 2, 1, 1017, 1018
mal100, 1000, 1000, 7, 1, 1004, 1005, 1006, 73, 1019, 1020, 1009
mal11, 1000, 1000, 6, 1, 1022, 1004, 1026, 1009, 1027, 1028
mal12, 1000, 1000, 2, 1, 1004, 1009
```

[그림 114] 학습에 사용한 데이터

3.2 딥러닝 모델

악성 PowerShell 스크립트 탐지를 위한 딥러닝모델은 [그림 4]와 같다. PowerShell로부터 추

출한 시퀀스 데이터를 입력으로 받으면 embedding layer를 거치게 되고 그다음으로 convolution layer를 통해 시퀀스 데이터의 특징을 학습하게 된다. 마지막으로 LSTM layer를 통해 시퀀스 데이터에 대한 학습을 하게 된다. 그리고 최종적으로 PowerShell 스크립트의 정상 및 악성 여부를 판단하게 된다.



[그림 115] 악성 PowerShell 스크립트 탐지 딥러닝 모델

3.3 실험 데이터

인터넷[8][9][10][11]에서 수집한 360개의 정상 PowerShell 스크립트 파일 샘플과 ESTsecurity[12]에서 제공받은 악성 PowerShell 스크립트가 포함된 OLE(Object Linking and Embedding) 개체 파일 351개 및 malshare.com[13]에서 수집한 악성 PowerShell 스크립트 9개를 합한 360개의 악성 파일 샘플을 데이터 샘플로 사용하였다.

학습에는 정상 데이터 샘플 360개와 악성 데이터 샘플 360개 총 720개의 파일 중에서 랜덤하게 80%인 576개를 선택하여 사용하고 나머지 20%인 144개를 테스트에 사용하였다.

IV. 실험 및 성능 평가

총 5회에 걸쳐 80%의 데이터를 가지고 학습을 하고 20%의 데이터를 사용하여 테스트를 수행하였다.

총 5회의 실험에서 탐지율(Recall)은 [표 1]과 같이 96.97%의 탐지율을 보여주었고, 오탐율(FPR: False Positive Rate)은 두 번째와 네 번째 실험에서만 오탐이 발생하여서 평균적으로

는 0.5%의 오탐율을 보여주었다.

이 실험을 통하여 PowerShell 스크립트로부터 키워드 시퀀스를 추출하고 이를 딥러닝 모델로 학습하여 PowerShell 스크립트의 정상 및 악성 여부를 탐지하는 것이 어느 정도 가능한 것을 확인할 수 있었다.

향후에는 좀 더 많은 정상 및 악성 PowerShell 스크립트를 수집하여 실험을 진행 할 예정이며, 관련 연구와의 비교실험도 수행하려고 한다.

	Recall	FPR	TP	FP	FN	TN
실험1	97.29	0	72	0	2	71
실험2	98.66	1.42	74	1	1	69
실험3	96.05	0	73	0	3	69
실험4	98.57	1.33	69	1	1	74
실험5	94.28	0	66	0	4	75
평균	96.97	0.5				

[표 1] 성능 평가 결과

V. 결론

본 논문에서는 정적 분석을 기반으로 악성 PowerShell 스크립트를 탐지하기 위해 정상 및 악성 파일 샘플로부터 텍스트 키워드를 추출하고, 이를 시퀀스로 치환하여 각 샘플 파일이 가지는 시퀀스 데이터를 이용해 딥러닝 모델을 학습시키는 방법을 제안하였다.

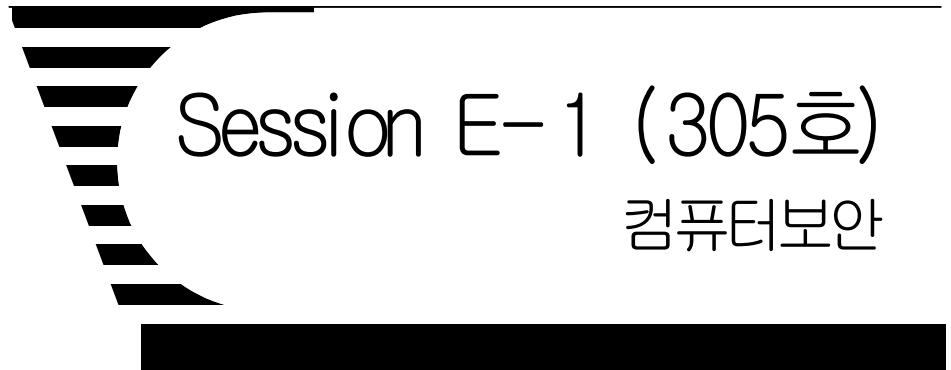
본 연구에서 진행한 실험에서는 정상 PowerShell 스크립트 및 악성 OLE 파일에서 추출한 악성 PowerShell 스크립트를 데이터 샘플로 사용했지만, 향후에는 PowerShell을 이용하는 실제 Fileless 악성코드로부터 PowerShell 스크립트를 탐지 및 추출하는 기능과 난독화된 스크립트를 난독화 해제하여 구문 분석하도록 하는 전처리 과정을 추가하는 연구를 진행 할 예정이다.

[참고문헌]

- [1] <https://zeltser.com/fileless-malware-beyond-buzzword/>
- [2] Microsoft, <https://docs.microsoft.com/ko-kr/>

<powershell/scripting/overview?view=powershell-6>, August, 2018.

- [3] Pavel Laskov, Nedim Srndic, Static Detection of Malicious JavaScript-Bearing PDF Documents, Proceedings of the Annual Computer Security Applications Conference. ACM, pp.373-382, Dec, 2011.
- [4] Danny Hendler, Shay Kels, Amir Rubin, Detecting Malicious PowerShell Commands using Deep Neural Networks, Proceedings of the 2018 on Asia Conference on Computer and Communications Security. ACM, pp.187-197, June, 2018.
- [5] 이승현, 문종섭, 명령 실행 모니터링과 딥러닝을 이용한 파워쉘 기반 악성코드 탐지 방법, 정보보호학회논문지, v.28, no.5, pp.1197-1207, Oct, 2018.
- [6] Michael Gough, <https://www.slideshare.net/Hackerhurricane/you-can-detect-powershell-attacks>, June, 2018.
- [7] Philippe Lagadec, <http://www.decalage.info/en/python/olevba>, Sep, 2015.
- [8] <https://www.robvanderwoude.com/powershellexamples.php>
- [9] <https://github.com/microsoftgraph/powershell-intune-samples>
- [10] <https://github.com/Azure-Samples/powerbi-powershell>
- [11] <https://aka.ms/PowerShellCorpus>
- [12] <https://www.estsecurity.com/>
- [13] <https://malshare.com/search.php?query=powershell>



좌장 : 서화정 (한성대)

마스킹 연구 동향

권용빈* 안규황* 권혁동* 서화정*†

*한성대학교 정보시스템공학과

dove333@naver.com tigerk9212@gmail.com korlethean@gmail.com hwajeong84@gmail.com

Research trends in masking

Yong-Been Kwon* Kyu-Hwang An* Hyeok-Dong Kwon*

Hwa-Jeong Seo*†

*Division of information system engineering, Hansung University.

요약

부채널분석 기술의 발달과 IoT(Internet Of Things)환경이 도래하면서, 앞으로의 암호기기에 부채널분석에 대한 대응기법의 적용은 필수불가결하다. 그 중에서도 기기에서 발생하는 전력을 이용하는 전력분석 기술과 대응기법인 마스킹(Masking)기법은 현재 활발한 연구가 진행 중이다. 본 논문에서는 이러한 마스킹기법에 대한 기존의 연구들을 분석하여 미래 연구에 대한 방향을 제시한다.

I. 서론

부채널분석 기술이란 암호기기가 작동하는 동안에 물리적으로 발생하는 전력, 전자기파, 시간을 분석하여 키와 관련된 정보 또는 키를 찾아내는 기술이다. 이전의 부채널분석에 대한 인식은 부채널분석을 시도하는 공격자가 암호기기에 가까이 접근하기 어렵다는 이유와 분석에 대한 연구 등한시되었다. 하지만 최근 분석 기구의 발달과 IOT환경에 따른 암호기기에 대한 접근성의 향상에 부채널분석 기술은 다시 활발하게 연구되고 있는 분야이다. 그 중에서도 전력분석은 대응기법이 적용되지 않은 많은 암호기기들의 키를 복원해내는데 성공한 강력한 분석법이다. 따라서 많은 암호기기들에 기본적인 대응기법을 적용하는 것이 필수적이다. 대표적인 대응기법은 마스킹기법이다. 마스킹기법은 구현 시 암호알고리즘의 특성과 암호가 사용되는 환경을 고려하여 만들어져야 한다. 본 논문에서는 마스킹기법에 대한 개념과 연구 동향을 살펴본다.

II. 전력분석

전력분석은 크게 파형을 단순하게 분석하는 기법인 SPA(Simple Power Analysis)와 여러 파형을 수집한 뒤 통계적으로 분석하는 기법인 DPA(Differential Power Analysis)로 분류된다. 일반적으로 키를 획득할 수 있는 강력한 분석 방법은 DPA이고 순서는 다음과 같다. 먼저, 전력모델을 설정한다. 기기가 전력을 소모하는 방식을 알아야한다는 것을 의미하며 주로 1의 개수를 세는 해밍웨이트(Hamming Weight)모델이나 변경되는 비트의 수를 나타내는 해밍디스턴스(Hamming Distance)모델에서 상수를 조절하여 이용한다. 다음으로 올바른 암호로 암호기를 동작할 때 발생하는 전력파형을 수집한다. 수집된 파형은 시작점을 맞추거나 공격지점 외의 부분을 제거하는 등의 전처리과정을 통해 총 DPA시간을 줄일 수 있다. 파형을 수집하고 나서는 키와의 연산을 거친 비트들의 버스내의

흐름을 고려하여 공격지점을 결정한다. 공격지점을 결정한다는 것은 그 지점에서의 비트를 예측한다는 것을 의미한다. 단순히 모든 비트를 예측하는 것은 공격에 대한 이점이 없으므로 알고리즘 구조에 맞게 좋은 공격지점을 찾는 것이 중요하다. 예를 들어 입출력 4비트 SBOX의 경우 예측 시 2^4 개의 경우의 수만을 고려하기에 좋은 공격지점이 된다. 다음으로 예측된 비트와 수집한 패형을 비교한다. 이 때, 두 집합 사이의 상관관계를 계산하거나, 예측된 비트에 기반하여 패형을 두 그룹으로 나눈 뒤 관계가 있는지 판단하는 방법 등을 이용한다. 결과적으로 예측된 키가 옳다는 관계가 나오면 그 키를 전체 키의 부분으로 하고 다음 부분의 비트를 분석하여 전체 키를 찾을 수 있다.

III. 마스킹

통계를 이용한 전력분석이 가능한 이유는 키와 평문이 조합된 비트들의 흐름을 살펴 특정 지점에서의 비트가 예측가능기 때문이다. 따라서 대응기법으로서 연산이 일어나기 전에 난수를 더하여 비트를 예측할 수 있도록 만드는 기법이 마스킹기법이다. 기본적인 마스킹연산은 난수를 더하는 형태이다. 선형연산의 경우에는 더해준 난수를 모든 연산을 마친 뒤 뺄셈으로써 정상적인 결과를 얻을 수 있다. 하지만, 비선형연산의 경우는 같은 방법으로 상용하는 결과를 얻을 수 없다. 그렇다고 하여 마스킹을 제거하고 비선형연산을 한다면 예측된 부분이 노출되므로 피해야 한다. 이러한 문제가 비선형연산에 있지만, 많은 암호알고리즘에서 비선형연산을 수행하기 때문에 이러한 문제들을 고려하여야 한다.

3.1 고차마스킹 공격과 대응기법

마스킹기법이 적용된 알고리즘을 분석해내기 위한 공격들이 키를 찾아내는데 성공하였고 따라서 기존의 마스킹기법을 1차마스킹기법이라고 부르게 되었다. 1차마스킹된 알고리즘을 공격하는 방법으로 마스킹 테이블의 약점을 이용하거나 각 라운드 별로 사용하는 난수가 같다는 점을 이용하여 비트를 찾아내는 등의 방법들이 있고 이를 고차전력분석이라고 한다. 물

론, 이러한 방법은 필요한 패형의 개수를 증가시키기 때문에 공격자로 하여금 더 많은 자원을 소모하게 한다. 반면, 고차전력분석을 막기 위해 마스킹 테이블을 다르게 설계하거나 각 라운드에서도 서로 다른 난수를 더하는 등 마스킹 알고리즘을 다르게 설정하는 방법을 고차마스킹기법이라고 한다. 이러한 기법 또한 암호기기가 더 많은 자원을 소모하도록 한다. 따라서 환경에 따라 적합한 수준의 마스킹기법을 적용하여야 할 것이나 공격자의 환경과 공격수준을 고려했을 때, 적어도 1차마스킹기법은 반드시 적용해야 할 것이다.

3.2 마스킹 변환

암호 알고리즘은 선형, 비선형 연산이 혼재한다. 따라서 선형 연산을 위해 마스킹된 값은 비선형 연산을 수행하기 위한 마스킹된 값으로 변환되어야 하며 이 때 마스킹 전의 값을 노출하지 않아야 한다. 이러한 변환을 위해 많은 연산을 요구한다. 특히 저전력 환경을 위한 암호 알고리즘의 경우에는 보다 효율적인 변환이 요구된다.

3.3 축소마스킹

일반적으로 마스킹기법은 다양한 마스크값을 이용할수록 안전성이 증가하고 동시에 연산량 즉 자원소모량도 증가하게 된다. 따라서, 저전력 암호기기를 위한 마스킹기법으로 알고리즘 초반과 맨 마지막 부분에만 마스킹을 적용할 수 있고 이를 축소마스킹이라고 부른다.

IV. 연구동향

본 절에서는 마스킹과 관련한 최신 연구 동향을 살펴본다.

4.1 고차 분석에 안전하고 효율적인 마스킹

최근 연구에서는 하드웨어로 구현된 AES에 대해 고차마스킹으로의 확장이 가능하면서도 적은 난수 발생, 적은 저장 공간과 빠른 속도를 가지는 효율적인 마스킹을 제안하였다[1]. 1차마스킹 구현에 대하여 가장 적은 공간을 요구하며, 충분히 안전한 난수를 발생시키기 위한 효율적인 방법을 적용하였다.

4.2 효율적인 마스킹 변환

LEA는 저전력 환경을 위해 설계된 블록 암호 알고리즘이다. 특징으로 Addition, Rotation, XOR연산으로 이루어진 ARX구조를 가진다는 점이다. 이렇게 선형 연산과 비선형 연산이 섞여있는 구조는 마스킹변환이 필요하다. 마스킹 변환 시 반드시 고려해야 할 점은 원래의 값이 노출되지 않아야 한다는 점과 저전력 환경을 감안하여 연산량을 줄이고 연산속도를 빠르게 해야한다는 점이다. 마스킹변환에는 산술 마스킹된 값을 불 마스킹된 값으로 변환하는 AtoB(Arithmetic-to-Boolean)과 그 반대인 BtoA(Boolean-to-Arithmetic)이 있다. 일반적으로 BtoA기법은 Goubin이 제안한 기법을 이용하며 연산량이 적어 효율적이다. 하지만 AtoB연산의 경우 연산량이 많아 사전 계산된 테이블을 이용하는 C-T기법이나 Debraize기법을 LEA에 적용하는 연구가 있었다[3]. 최근 연구에서는 기존 LEA에 적용된 C-T기법에서 C-테이블을 만들지 않고 알고리즘 상에서 발생한 캐리를 바로 덧셈하는 방법으로 처리하여 필요한 메모리의 수를 절반으로 줄인 마스킹 최소 단위 8기준 2^8 바이트를 저장할 메모리만을 필요하도록 하였다. 동시에 처리 속도를 향상시키기 위해 반복문 내 바이트 단위 연산을 반복문 밖에서 워드 단위로 연산하여 효율적인 연산이 가능케 하였다. 그 결과 이전까지 가장 빠르다고 알려진 C-T기법 보다도 17%정도의 속도 향상과 C-T 기법과 Debraize기법이 요구하는 메모리의 절반인 256바이트의 메모리만을 요구하는 효율적인 LEA 마스킹 기법을 제안하였다[2].

4.3 축소마스킹에 대한 공격

SIMON 암호 알고리즘은 저전력 환경에 적합한 블록 암호 알고리즘이다. 따라서 효율적인 마스킹을 위해 축소마스킹을 적용할 수 있는데 이 경우 알고리즘 중반에 해당하는 비트들의 해밍웨이트를 알 수 있다. 이 해밍웨이트 값이 특정 값을 만족하는 평문을 찾아 차분 공격을 진행한다. 차분 특성을 만족하는 횟수가 많은 예측키의 비트 중 차분의 영향을 받는 비트만을 옳은 키의 비트의 일부로 하여 전체키를 복구하였다. 많

은 라운드를 마스킹 할수록 더 많은 평문이 필요하지만 10라운드 이내로 마스킹이 적용된 SIMON 알고리즘은 취약함이 발표되었다[3].

V. 결론

마스킹기법은 각 암호 알고리즘의 특성에 맞게 다양하게 구현될 수 있다. 따라서 암호 알고리즘을 잘 분석한다면 적은 연산으로도 강력한 암호 강도를 보장할 수 있다. 최근 연구들은 저전력 환경에 적합하도록 효율적인 마스킹기법을 제안하면서 부채널분석에 대한 충분한 안전성을 가지는지 검증하고 있다. 전력분석에 대한 연구가 활발하게 일어나고 저전력 환경이 중요해지는 만큼 대응기법인 마스킹기법을 암호 알고리즘을 고려하여 효율적이고 안전하게 설계하는 것이 중요하다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터 (UD170109ED)를 통한 방위사업청과 국방과학 연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- [1] H. Gross, S. Hangard and T. Korak, An efficient side-channel protected AES implementation with arbitrary protection order, *Cryptographers' Track at the RSA Conference*, pp. 95–112, February, 2017
- [2] E. Park, S. Oh and J. Ha, Masking-Based Block Cipher LEA Resistant to Side Channel Attacks, *Journal of The Korea Institute of Information Security & Cryptology* 27(5), pp. 1023–1032, October, 2017.
- [3] J. Kim, K. Hong, S. Kim, J. Cho and J. Kim, Side Channel Attacks on SIMON Family with Reduced Masked Rounds, *Journal of The Korea Institute of Information Security & Cryptology* 27(4), pp. 923–941, August, 2017.

램포트 해시체인을 적용한 QR코드 출입통제 시스템

박형민, 김가을 이선영*

*순천향대학교 정보보호학과

phmin911@gmail.com, autumn666@naver.com, *sunlee@sch.ac.kr

QRcode Access Control system based on Lamport Hash-chain

Park Hyeong Min, Kim Ga Eul, *Sun-Young Lee*

*Dept. of Information Security Engineering, Soonchunhyang University

요약

기업의 정보 유출 방지와 사내 방범을 위해 출입통제 시스템이 중요시되고 있다. 기존의 RFID 시스템은 무선통신으로 데이터를 송수신하는 과정에서 쉽게 공격자에 의해 데이터를 탈취 당할 수 있고 도청, 재전송 공격 등이 가능하다. 따라서 본 논문에서는 효율적 비동기화 방식의 일회용 패스워드 인증 기법인 램포트 해시체인을 QR코드 생성에 적용한 출입통제 시스템을 제안한다.

I. 서론

최근 정보 유출 사고가 급증하면서 기업에서도 정보 유출 방지를 위하여 보안에 관심을 기울이고 있다. 기업의 정보 유출 방지와 사내 방범을 위해 건물 내의 출입통제 시스템은 보안되어야 할 기본적인 요소이다. 현재 기업에서 사용하고 있는 RFID(Radio-Frequency Identification) 카드를 이용한 출입통제 시스템은 이용이 간편하다는 장점이 있다. 하지만 RFID 카드를 분실하거나 도난을 당하였을 경우 출입의 제한이 있을 뿐만 아니라 RFID 카드 복제에 따른 보안 문제가 발생할 수 있다. 또한 RFID 시스템은 리더기와 태그가 무선통신으로 데이터를 송수신하기 때문에 공격자가 쉽게 도청을 하여 재전송 공격을 할 수 있다[1].

이에 따라, 본 논문에서는 램포트 해시체인(Lamport Hash-chain)을 적용한 QR코드를 애플리케이션에서 생성한 후 리더기에 태그 하여 인가된 사용자임을 인증하는 QR코드 출입통제 시스템을 제안하여 RFID 카드를 이용한 출입통제 시스템의 보안 문제를 해결하고자 한다. 본 논문의 2장에서는 QR코드, 램포트 해시체인,

RFID 시스템에 대해 분석하고, 3장에서는 2장에서 연구한 내용을 기반으로 램포트 해시체인을 적용한 QR코드 출입통제 시스템을 제안하고 4장에서 결론을 맺는다.

II. 관련 연구

2.1 QR코드

QR코드는 작은 정사각형의 점을 가로, 세로 동일하게 병렬시킨 2차원 코드로써 숫자, 영자, 한자, 한글, 기호, 이진수, 제어 코드 등 모든 데이터를 처리 할 수 있다. 위치 찾기 심볼과 셀로 이루어져 있으며 셀을 이용하여 데이터를 저장한다. 숫자 최대 7,089자, 문자 최대 4,296자, 한자 최대 1,817자 저장할 수 있다. 위치 찾기 심볼은 배경에 영향을 받지 않도록 하여 어느 방향에서도 안정적인 고속 인식을 하게 해준다. 전용 스캐너뿐만 아니라 스마트폰으로도 정보를 송수신이 가능하며, 물류관리, 마케팅 등 다양한 분야에서 사용되고 있다[2].

2.2 램포트 해시체인

램포트 해시체인은 클라이언트에서 생성한

비밀 값에 연속해서 해시함수를 연산하여 해시체인을 생성하고, 생성한 해시값을 역순으로 인증을 위해 사용하는 효율적 비동기화 방식의 일회용 패스워드 인증 기법이다[3][4][5]. Fig. 1.은 램포트 해시체인의 생성과 사용 방향을, Fig. 2는 램포트 해시체인을 통한 인증과정을 보여준다.

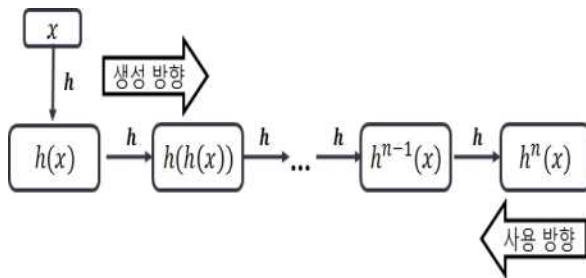


Fig. 1. 램포트 해시체인 생성, 사용 방향 과정

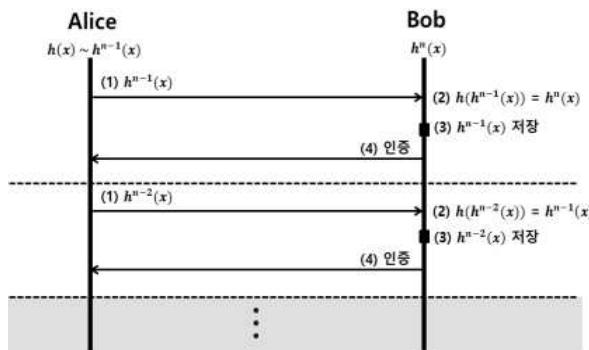


Fig. 119. 램포트 해시체인을 통한 인증과정

Step 1. Alice는 Bob에게 $h^{n-1}(x)$ 를 전송한다.

Step 2. Bob은 Alice에게 받은 $h^{n-1}(x)$ 에 해시연산하여 가지고 있던 $h^n(x)$ 와 비교한다.

Step 3. Bob은 $h^{n-1}(x)$ 를 저장한다.

Step 4. Bob은 Alice를 인증한다.

만약 공격자가 해시값 탈취에 성공해도 해시함수의 특성에 의해 클라이언트에서 생성한 비밀 값을 복원할 수 없으며, 다음에 사용될 해시값 또한 유추하기 어렵다. 클라이언트에 생성한 비밀 값에 해시함수를 연산한 횟수에 따라 인증 가능한 횟수가 제한된다.

2.3 RFID 시스템

RFID 시스템은 무선통신 기술을 사용하여 리더기와 태그 사이에서 데이터를 전달하는 방식이다. 태그는 안테나와 IC칩으로 구성되어 있으며 IC칩에는 정보가 기록되고 안테나를 통해 리더기에 정보를 송신한다. 태그에 저장된 정보는 반복적으로 사용 가능하며 설정에 따라 ID 정보, 실시간 정보 등을 전송할 수 있다. RFID 시스템은 도청, 재전송 공격 등에 취약하다[1].

III. 제안 시스템

본 논문에서는 램포트 해시체인으로 일회용 QR코드들을 생성하고 이를 스마트폰 내 애플리케이션을 통해 인증하는 출입통제 시스템을 제안하였다. Fig. 3은 사전 등록 과정을 보여준다.

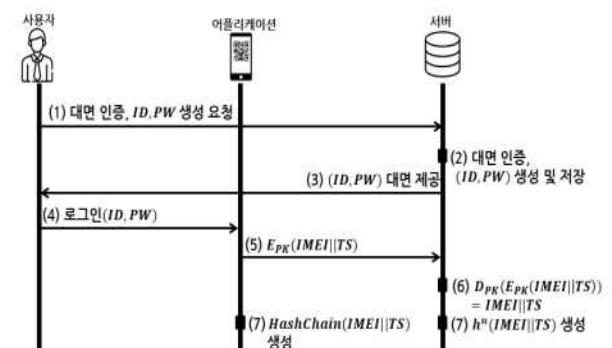


Fig. 120. 사전 등록

Table 1. 사전 등록 프로토콜 기호

Notation	Description
Hashchain	해시함수를 $(n+1)$ 회 연산한 값
ID, PW	사용자의 ID, PW
IMEI	국제단말기식별번호
TS	타임스탬프
PK	대칭키

Step 1. 사용자는 서버에게 대면 인증과 사용자의 ID, PW 생성을 요청한다.

Step 2. 서버는 사용자를 대면 인증하고, 사용자의 ID, PW를 생성하고 저장한다.

Step 3. 서버는 사용자에게 사용자의 ID, PW를

대면 제공한다.

Step 4. 사용자는 서버로부터 받은 사용자의 ID, PW로 애플리케이션에 로그인한다.

Step 5. 애플리케이션은 단말기의 IMEI값과 TS를 사전에 공유 받은 대칭키로 암호화 후 서버에게 전송한다.

Step 6. 서버는 사전에 공유한 대칭키로 Step 5.의 값을 복호한다.

Step 7. 어플리케이션과 서버는 각각

$\text{Hashchain}(\text{IMEI} \parallel \text{TS})$ 과 $h^n(\text{IMEI} \parallel \text{TS})$ 을 생성한다.

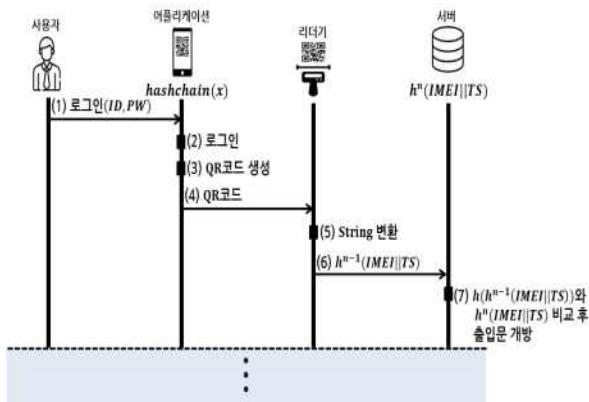


Fig. 4. QR코드를 사용한 인증 프로토콜

사용자는 초기 등록 당시 서버에게 전달받은 ID, PW로 애플리케이션에 로그인한다. 애플리케이션은 등록 당시 생성한 해시체인으로 QR코드를 생성 후 리더기에 입력한다. 리더기는 QR코드를 String 변환하여 $h^{n-1}(\text{IMEI} \parallel \text{TS})$ 을 획득 후 서버에게 전달한다. 서버는 리더기에게 받은 $h^{n-1}(\text{IMEI} \parallel \text{TS})$ 에 해시연산을 한 후 사전등록 당시 생성한 $h^n(\text{IMEI} \parallel \text{TS})$ 와 비교하여 출입문을 개폐한다. 이후 과정은 동일하다.

QR코드 생성에 사용되는 해시값이 탈취되어도 그 해시값은 다시 사용되지 않으므로 재전송 공격을 방지할 수 있으며, 해시함수의 특성상 탈취당한 값으로 클라이언트에서 생성한 비밀 값을 유추할 수 없다는 장점이 있다.

IV. 결론

본 논문에서는 램포트 해시체인으로 일회용 QR코드들을 생성하고 이를 스마트폰의 애플리케이션을 통해 인증하는 출입통제 시스템을 제안하였다. 제안 시스템을 사용할 시 RFID 시스템의 도청, 재전송 공격 등의 취약점을 방지한다. 일회용 QR코드의 사용으로 재전송 공격을 방지할 수 있으며 해시함수의 일방향성을 통해 클라이언트에서 생성한 비밀 값을 유추할 수 없다는 장점이 있다.

[참고문헌]

- [1] Ari Juels, RFID Security and Privacy: A Research Survey, IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 24, NO. 2, FEBRUARY 2006
- [2] DENSO WAVE INCORPORATED, QR코드란, <https://www.qrcode.com/ko/about/>
- [3] L. Lamport, “Constructing digital signatures from a one-way function,” Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, Oct. 1979.
- [4] RFC 2289, A One-Time Password System, Feb. 1990. <https://www.ietf.org/rfc/rfc2289.txt>
- [5] L. Lamport, “Password Authentication with Insecure Communication”, Communications of the ACM (24.11), pp880-772., 1981.11.24.

하이브리드 퍼저를 이용한 브라우저 취약점 분석

이정모, 이병천

*중부대학교 정보보호학과

Analysis of Browser Vulnerabilities using Hybrid Fuzzer

Jungmo Lee, Byoungcheon Lee

*Department of Information Security, Joongbu University

요약

최근 웹 기반 서비스가 증가하면서 브라우저의 취약점을 이용한 웹서비스 공격이 증가하고 있어서 브라우저의 취약점에 대한 선제적인 대응이 필요하다. 브라우저 취약점을 탐색하기 위한 작업에는 Auditing과 Fuzzing이 존재하는데 본 논문에서는 하이브리드(Hybrid) 퍼저(Fuzzer)인 Nduja를 이용해 브라우저의 취약점을 분석해본다. 퍼징을 진행하면서 여러 다양한 크래시를 수집하고 수집한 결과들을 분석하는 과정을 제시한다.

I. 서론

최근 웹을 기반으로 하는 서비스가 증가하면서 많은 사람들은 브라우저를 이용해 다양한 종류의 인터넷 서비스들을 사용하고 있다. 그중에서 구글이 Chromuim 엔진을 기반으로 만든 크롬 브라우저는 현재 가장 많은 이용자수를 가지고 있다[1]. 악의적인 목적을 가진 해커들은 브라우저에 존재하는 취약점들을 이용해 많은 문제를 일으키고 있다.

시만텍(Symantec)에서 발표한 2017년 사이버 범죄 및 보안 위협 동향에 대한 분석을 담은 인터넷 보안 위협 보고서(ISTR)[2]에 따르면 웹에 관련된 악성코드 및 피싱이 가장 많이 이루어지고 있으며 이는 대부분 웹 브라우저에서 활동되는 자바스크립트의 취약점을 노린 공격이다. 철저한 보안과 완벽에 가까운 소스코드를 가지고 있다고 공언하는 크롬 브라우저이지만 지금도 Chrome 자바 스크립트 엔진인 V8에서 발생한 Use-after-Free 버그(CVE-2018-17465), BigInt64Array Out-of-Bound Write (CVE-2018-16065) 등 취약점이 많이 발견되고 있다. 이 중 몇몇 취약점들은 블랙마켓에서 거액으로 판

매가 되고 있고 공격자들은 해당 취약점을 가지고 악의적인 스크립트를 실행하는 등 사용자들의 컴퓨터를 위협하고 있다.

따라서 브라우저 취약점이 악의적인 의도를 가진 해커에 의해 악용되기 전에 개발자가 분석을 진행하는 버전인 Debug 버전과, 일반 사용자들에게 배포되는 버전인 Release 버전 단계에서 크래시(Crash) 및 취약점을 미리 발견해 개선하도록 해야 한다. 이러한 취약점들을 발견하는 방법에는 눈으로 소스코드를 읽고 수동적으로 취약점을 발견하는 방법인 오디팅(Auditing) 방법과 소프트웨어에 무작위 데이터를 반복하여 입력해 소프트웨어의 로직에 크래시를 유발함으로써 보안상의 취약점을 자동으로 찾아내는 퍼징(Fuzzing)을 이용하는 방법이 있다.

본 논문에서는 하이브리드(Hybrid) 퍼저(Fuzzer)인 Nduja를 이용해 브라우저의 취약점을 탐색할 것이다. 그리고 이를 토대로 나온 취약점을 분석하는 과정을 살펴보고 퍼저에 대한 대응책을 제시한다.

II. 퍼징

2.1 퍼징 기법의 종류

퍼징에는 Table. 1에서 제시한 바와 같이 다양한 기법들이 사용된다. 본 논문에서는 효율성이 높은 하이브리드 퍼징 방식을 사용한다.

Table 1. Classification of Fuzzing Technique
s

퍼징 기법	설명
Dumb Fuzzing	입력의 형태를 모르는 환경에서 임의의 값으로 입력 데이터를 생성 또는 변이하는 퍼징 기법
Smart Fuzzing	입력의 형태를 하는 환경에서 입력의 타입 또는 형태에 따라 입력 데이터를 생성 또는 변이하는 퍼징 기법
Mutation Fuzzing	기존에 정상적인 입력 데이터의 일부를 조작하여 새로운 입력 데이터를 만들어 내는 퍼징 기법
Generation Fuzzing	분석된 입력의 형태에 따라 적합한 새로운 입력 데이터를 만들어 내는 퍼징 기법
Hybrid Fuzzing	Mutation과 Generation 퍼징 방법의 장점을 결합한 방식으로 비교적 높은 커버리지를 가지므로 예측하기 어려운 상황들을 다양하게 이끌어내는 퍼징 기법

2.2 퍼징 도구 Nduja

Nduja[3]는 Rosario Valotta가 개발한 웹브라우저 퍼징 도구이다. 기존의 퍼저들은 DOM [4]Level 1 또는 [5]Level 2에 대한 API를 이용한 퍼징을 수행했지만 Nduja에서는 DOM Level 2 및 [6]Level 3 API를 많이 활용하고 좀 더 복잡한 퍼징 알고리즘을 이용해 다양한 크래시와 새로운 형태의 취약점을 발견할 수 있다. 전체적인 퍼징 메커니즘은 Fig. 1과 같다[7].

Nduja는 Javascript로 만들어진 퍼저이며 이를 이용하여 무작위로 DOM 요소들을 만들고 DOM 처리 함수를 무작위로 호출하며 임의로 DOM 요소를 삭제하는 등 다양하고 복잡한 과

정을 뒤섞어 브라우저를 분석한다. 이런 과정을 거치게 되면 다양한 객체가 할당되고 해제되는 과정을 반복하는데 이 과정에서 어떠한 객체가 이미 메모리가 해제된 객체에 접근하게 되는 Use-After-Free(UAF) 버그가 발생할 수 있으며 이는 브라우저 해킹에서 가장 많이 이용되는 버그이다. 이 도구는 많은 UAF 유형의 취약점을 발견했으며 2012년경 브라우저 취약점 패치와 악용에 가장 많이 사용된 퍼저로 알려졌다.

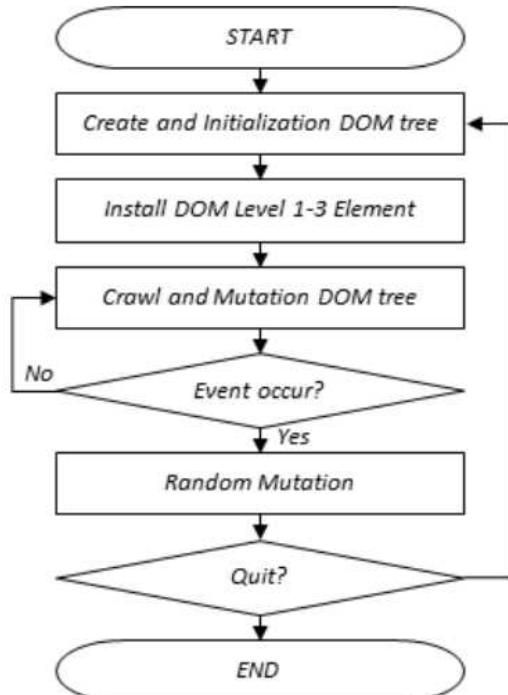


Fig. 1 Nduja Fuzzing Mechanism

III. Nduja 환경 구성 및 방법

Nduja는 루비(Ruby) 언어로 개발된 웹 브라우저 퍼징 프레임워크인 그라인더(Grinder)[8]를 기반으로 작동하며 전체적인 구성은 노드(Node)와 서버(Server)로 이루어져 있다. 노드는 퍼징을 수행하는 단위 객체를 의미하며, 서버는 그라인더를 통해 퍼징을 관리하고 크래시를 분류해 수집하는 등 관리 역할을 한다.

전체적인 운영 환경을 Table. 2 및 Fig. 2에 제시하였다. 여기에서 호스트는 여러 PC를 돌리고 있는 것처럼 구성하기 위해 VMware가 설치된 PC를 의미하며, 노드는 하나의 호스트

컴퓨터에 설치된 여러개의 GuestVM에 설치된 그라인더 노드를 의미한다. 서버는 노드에서 생성된 크래시를 분류하고 관리하기 위해 그라인더 서버가 설치된 우분투 환경의 PC를 말한다.

Table 2. System Configuration of Experimental Environments

	Host	GuestVM	Server
OS	Win7	Win7(32bit)	Ubuntu14
CPU	Intel i5	Intel i5	Intel i7
RAM	8GB	1GB	8GB

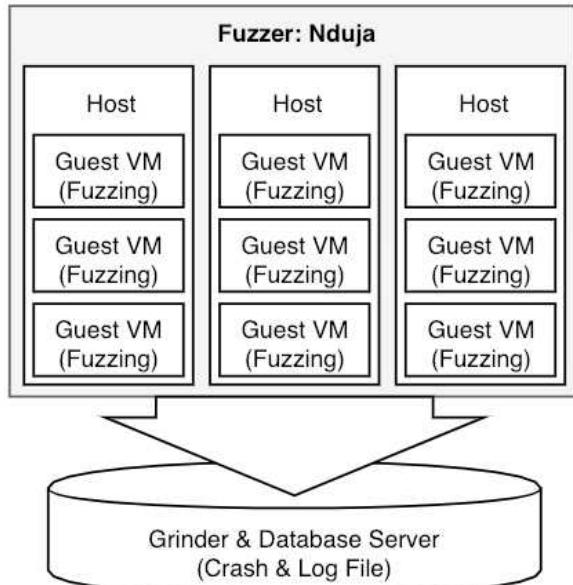


Fig. 2 Operations in Developed Environment

각각의 노드에 있는 브라우저에서 크래시가 발생하면 노드 내의 `~/grinder/node/crashes` 디렉토리 내 2개의 파일(.crash, .log)에 이들 정보가 저장된다. 그리고 이 파일들은 그라인더 서버로 전송된다. .crash 파일에는 call stack, disassembly, register info 등 유용한 디버깅 정보가 담겨져 있으며 .log 파일에는 재생성(reproduce)이 가능한 테스트 케이스 정보가 저장된다. 이것은 브라우저를 페징하는 과정에서 생성된 로깅(logging) 정보가 들어 있으므로 크래시를 트리거 할 수 있다. 또한 크래시 파일을

이용한 브라우저 공격의 성공 가능성은 디버거 모듈이 대상 브라우저 프로세스를 후킹하고 Windbg에 있는 !exploitable API를 사용하여 판단할 수 있다. 서버로 전송되는 이러한 정보들은 보안상 민감한 내용이므로 암호화되어 전송하도록 되어 있다. 그라인더의 장점 중 하나는 웹 서버를 구축해 놓은 상태이므로 외부에서 로그인하여 퍼저들이 정상적으로 작동하고 있는지, 어떠한 취약점이 발생되고 있는지 외부에서도 분석이 가능하도록 크래시 파일과 로그 파일을 다운로드 및 상세보기가 가능하다.

IV. 실험 및 결과

퍼저는 Nduja를 사용했으며 하나의 호스트 내에 여러 퍼저를 구동시키기 위하여 가상으로 환경을 구성할 수 있는 VMware를 이용해 여러 노드를 구성하고 IE11, Chrome 61을 대상으로 페징을 진행하였다. 해당 버전들은 실험 당시 가장 최신 버전을 대상으로 진행했다. 이렇게 구축한 시스템으로 약 7일간 페징을 수행하였고 페징한 결과는 Table. 3 와 같다.

Table. 3. Experimental Results

Target	Type	Count
IE11	Stack Overflow	6
Chrome	Read Access Violation	61
Chrome	Write Access Violation	1
Chrome	Execute Access Violation	564

이 실험에서는 IE와 크롬을 포함해 약 630여 개의 크래시가 발생하였으며 다양한 취약점을 탐지할 수 있었다. 오류로 분류될 수 있는 크래시에 한해서만 수집하였으며 이것은 공격자가 Heap Feung Shui[9], Heap Spray[10] 공격 등을 통해 원하는 코드를 주입한 후, 해당 코드로 실행흐름을 변경할 수 있는 가능성이 높기 때문이다.

V. 크래시 사례 분석

위에서 생성된 덤프 파일과 크래시 로그는 매우 많기 때문에 그중에 하나의 크래시 로그 파일을 사례로 분석해본다.



Fig. 124 Chrome Crash File

Fig. 3의 사례는 Read Access Violation의 사례로 해당 프로세스가 읽기(Read) 권한이 없는 메모리 영역에 접근해 발생한 오류이다. 이는 메모리 주소가 누출되어 권한이 없는데도 필요한 주소나 정보를 읽어올 수 있는 트리거(Trigger)가 된다.

```
Call Stack:
0x6699722 - chrome_child!v8::internal::anonymous namespace)::Invoke
0x6699720 - chrome_child!v8::internal::Eval::Execution::Call
0x669920A2 - chrome_child!v8::Script::Run
0x66998875 - chrome_child!blink::V8ScriptRunner::runCompiledScript
0x66997F25 - chrome_child!blink::ScriptController::executeScriptAndReturnValue
0x66998856 - chrome_child!blink::ScriptController::evaluateScriptInMainWorld
0x66998A1 - chrome_child!blink::ScriptController::executeScriptInMainWorld
0x66998978 - chrome_child!blink::ScriptLoader::doExecuteScript
0x66998978 - chrome_child!blink::ScriptLoader::doExecuteScript
0x66998E53 - chrome_child!blink::ScriptController::parseScript
0x66992604 - chrome_child!blink::HTMLScriptRunner::runScript
0x66992486 - chrome_child!blink::HTMLScriptRunner::execute
0x66992474 - chrome_child!blink::HTMLDocumentParser::runScriptsForPausedTreeBuilder
0x669940C6 - chrome_child!blink::HTMLDocumentParser::processTokenizedChunkFromBackgroundPa
0x66999408 - chrome_child!blink::HTMLDocumentParser::pumpPendingScript
0x66959408 - chrome_child!blink::HTMLDocumentParser::resumeParserInAfterScriptExecution
```

Fig. 125 Call Stack of Chrome Crash

위의 사례에 대한 콜 스택을 보면 맨 아래에 있는 함수가 최초로 실행된 곳이며 맨 위에 있는 함수가 결론적으로 문제가 발생한 함수 지점을 나타낸다. 그리고 Windbg를 사용해 덤프 파일을 분석한다. 크로미움(Chromium)은 모든 함수에 대해 소스가 공개 되어 있으므로 Windbg로 디버깅 시 심볼 서버를 크로미움으로 설정한다[11]. 그리고 kb(Display Stack Backtrace) 명령어를 이용해 콜스택을 확인하고 Windbg 기능 중 하나인 Local view(로컬 창으로 변수 보기)와 소스 코드를 보며 최초 혹은 중간 지점부터 호출한 함수에서 시작해 맨 위에 있는 오류 발생 지점 함수까지 따라가며 분석을 진행하게 된다.

VI. 결론 및 향후 연구

본 논문에서는 현재 보안 시장의 뜨거운 이슈로 부상하고 있는 웹브라우저의 취약점에 대

해 분석하는 방법 중 대표적인 방법인 퍼징에 대해 소개하였다. 그 중 Nduja11을 이용해 하이브리드 퍼징을 진행하였고, 그 결과 많은 크래시 데이터를 얻을 수 있었다.

그런데 취약점 제보를 목적으로 브라우저 취약점 분석을 하고 있는 연구자들이라면 모두 이러한 퍼져들을 돌리고 있을 것으로 생각되는데 같은 퍼징 알고리즘 및 엔진을 이용해 취약점을 점검하면 비슷하거나 똑같은 취약점이 나올 것이다. 또한 구글에서는 이미 브라우저를 구성하고 있는 전체적인 클래스를 대상으로 대규모로 퍼징 테스트를 진행하고 있다. 그러므로 우리가 발견한 크래시를 해당 소프트웨어 벤더사에 제보를 해도 중복될 가능성이 높게 된다. 따라서 이러한 문제점들을 극복하기 위해서는 좀 더 다양한 방법론을 사용하는 퍼져 엔진을 제작하기 위한 연구개발이 필요하며, 여러 웹 API 중 소수의 선택된 클래스를 대상으로 취약점을 집중적으로 찾을 수 있는 퍼져를 개발하여 이용할 필요가 있다.

또한 취약점 발견에만 몰두하지 말고 크래시 결과를 분석하는 과정에서 존재하는 메모리 보호기법에 대한 연구도 필요하다. 즉 특정 프로세스가 사용하는 메모리 주소를 무작위로 부여하여 익스플로잇(Exploit)을 방어하기 위한 ASLR(Address Space Layout Randomization)이나 외부 접근 및 영향을 차단하여 제한된 영역 내에서만 프로그램을 동작시키는 Sandbox와 같은 기법들을 우회할 수 있는 공격기술을 연구해야 할 필요가 있다.

[참고문헌]

- [1] Browser Market Share Worldwide <http://gs.statcounter.com/>
- [2] Symantec Internet Security Threat Report <https://www.symantec.com/content/dam/symantec/ko/docs/infographics/istr-23-infrastucture-attacks-stealthy-mining-threats-go-big-and-small-ko.pdf>
- [3] Rosario Valotta, "Taking Browsers Fuzzin

- g To The Next (DOM) Level”, DeepSec 2012.
- [4] Document Object Model (DOM) Level 1 Specification <https://www.w3.org/TR/DOM-Level-1/>
- [5] Document Object Model (DOM) Level 2 Core Specification <https://www.w3.org/TR/DOM-Level-2-Core/>
- [6] Document Object Model (DOM) Level 3 Core Specification <https://www.w3.org/TR/DOM-Level-3-Core/>
- [7] Event and Command based Fuzzing Method for Verification of Web Browser Vulnerabilities <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.836.8133&rep=rep1&type=pdf>
- [8] Grinder, <https://github.com/stephenfewer/grinder>.
- [9] Alexander Sotirov ‘Heap Feng Shui in JavaScript’ <https://www.blackhat.com/presentations/bh-europe-07/Sotirov/Presentation/bh-eu-07-sotirov-apr19.pdf>
- [10] Heap Spraying https://en.wikipedia.org/wiki/Heap_spraying
- [11] Debugging Chromium on Windows <https://www.chromium.org/developers/how-tos/debugging-on-windows>

seL4 마이크로커널의 특성 분석

장희준, 김형식

성균관대학교 소프트웨어대학

hj.jang@skku.edu, hyoung@skku.edu

Analysis on seL4 Microkernel's Properties

Heejun Jang, Hyoungshick Kim

College of Software, Sungkyunkwan University.

요약

seL4 마이크로커널은 L4 마이크로커널을 개량한 오픈소스 보안 마이크로커널로서 설계 단계부터 정형 검증을 고려하여 개발하고 모든 과정이 검증된 최초의 완전히 정형 검증된 상용 마이크로커널이다. seL4는 보안시스템을 구현하기 위해, 정형 검증 기법이 적용된 최소한의 메커니즘을 제공하며, 기존에 개발된 마이크로커널 중 범용성을 갖고 최악의 조건일 때의 실행시간 측정에 대한 높은 수준의 보증 분석을 거친 유일한 마이크로커널이며, 안전한 시스템을 구축하기 위해 가장 많이 참조된다. 본 논문에서는 seL4에 대한 조사를 진행하여 L4를 어떻게 발전시켜 seL4를 완성했는지, seL4를 기반으로 하는 OS 구조는 어떠한지, 어떻게 보안성을 보장하는지, 검증을 위해 어떤 정형 검증 방법을 사용했는지, 현재 남아있는 연구과제는 무엇인지 다룰 것이다.

I. 서론

컴퓨터 시스템의 보안성 및 안정성은 OS 커널의 보안성 및 안정성에 의존한다. 모놀리틱 커널 구조를 가진 일반적인 OS의 경우 다양한 하드웨어에 대한 지원과 기능성이 늘어나면서, 커널의 크기와 복잡성이 증가하고 있다. 일반적인 OS의 커널의 코드는 수천만 줄에 이르렀고, 이러한 코드 크기의 증가는 OS의 TCB 크기증가를 의미하고, TCB의 크기 증가는 버그, 취약점 발생의 증가를 일으켰다. 결국, 커널의 보안성을 높이기 위해서는 커널의 코드 크기와 TCB 크기를 줄여야 한다. 이런 관점에서 마이크로커널은 기존 커널보다 보안 측면에서 우수하다고 주장되었고, 이 주장을 뒷받침할 수 있는 양적 연구가 실행되었다[1].

마이크로커널은 커널의 가장 기본적인 부분(메모리 관리, 스케줄링, 기본적인 IPC 등 핵심 기능들)만을 담당하며, 나머지 부분(디바이스 드라이버, 인터럽트 핸들러 등의 기타 OS 서비스)들은 사용자 모드에서 실행되는 커널을 말한다. 마이크로커널 기반 시스템의 경우, 응용프로그램은 서비스를 호출하기 위해 IPC 통신을 이용한다. 사용자 모드에서 실행되는 서비스들은 독립성이 보장되고 context-switching 엔진으로 동작하며, 서비스를 수행하는데 필요한 최소한의

권한만 받는다. 이 서비스들 중 하나가 실행 중 문제가 생긴다고 하더라도, 전체적인 시스템에는 영향이 없어 시스템이 비정상적으로 종료되는 경우를 방지할 수 있다. 마이크로커널의 대표적인 예로 MINIX3, L4, seL4 등이 있다.

seL4는 L4 마이크로커널을 개량한 오픈소스 보안 마이크로커널로서 설계 단계부터 정형 검증을 고려하여 개발되고 모든 과정이 검증된 최초의 완전히 정형 검증된 상용 마이크로커널이다. seL4 마이크로커널은 capability를 기반으로 한 접근제어를 사용하여, 시스템의 접근 권한을 세부적으로 제어하고, 최소한의 권한만 갖도록 설계할 수 있다. 또한 보안 시스템을 구현하기 위해, 정형 검증 기법이 적용된 최소한의 메커니즘을 제공한다. seL4는 기존에 개발된 마이크로커널 중 범용성을 갖고 최악의 조건일 때의 실행시간 측정에 대한 높은 수준의 보증 분석을 거친 유일한 마이크로커널이며, 안전한 시스템을 구축하기 위해 가장 많이 참조된다.

본 논문에서는 seL4에 대한 조사를 진행하였다. L4를 어떻게 발전시켜 seL4를 완성했는지, seL4를 기반으로 하는 OS 구조는 어떠한지, 어떻게 보안성을 보장하는지, 검증을 위해 어떤 정형 검증 방법을 사용했는지, 현재 남아있는 연구과제는 무엇인지 다룰 것이다.

II. seL4로의 변화(L4 to seL4)

seL4 마이크로커널은 L4 마이크로커널이 개량된 커널로 L4 마이크로커널 가족에 포함된다. L4 마이크로커널에서 seL4 마이크로커널로의 가장 큰 변화는 어셈블리 언어로만 구현되었다가, 어셈블리 언어의 비중을 현저히 낮추고 커널의 소스를 C/C++ 언어로 대체한 것이다. 그리고 L4 마이크로커널에서 발생하는 문제를 해결하고자 설계 및 구현 부분에서 변화가 있었고[2], 아래에서 그에 대한 사항을 자세히 다루고자 한다.

2.1 설계

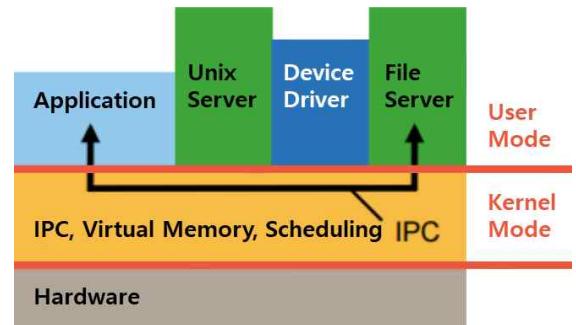
L4 마이크로커널에서 Long IPC가 기존 POSIX read-write 인터페이스를 서버로 사용할 때 주로 이용되었으나, 기능성 측면에서 최소성 원칙을 위반하기 때문에 seL4에서는 배제되었다. L4 마이크로커널은 동기 IPC 모델을 통신, 동기화, 신호 메커니즘에 사용하였고, 캐시 및 TLB의 오염을 방지하기 위해 IPC 작업의 대상으로 스레드를 사용하였고, DoS 공격과 같은 long IPC에 의해 발생할 수 있는 공격을 보호하기 위해 IPC 작업에 타임아웃을 둘었으며, “clans and chief”라는 메커니즘을 통해 IPC 통제를 하고, 프로세스 계층 구조를 사용하여 DoS 공격을 발생시킬 수 있는 확인되지 않은 TCB 할당 문제를 해결하였다. 이에 반해 seL4에서는 L4와 같은 동기 IPC 모델을 유지하면서, 해당 모델을 비동기 알림으로 보완하여 비동기 IPC 통신을 수행하였고, 본질적으로 포트와 비슷한 IPC 앤드 포인트를 IPC 작업 대상으로 채택했으며, long IPC를 지원하지 않기 때문에 타임아웃을 두 개의 플래그 값으로 대체하였고, 앤드 포인트에서의 capability를 기반 접근 제어를 통해 IPC 통제를 하였고 프로세스 계층 구조를 capability로 대체하였다.

2.2 구현

기존 L4 마이크로커널에서 존재하던 프로세스 커널 스택, 가상 TCB 배열, 비이식성, 비표준 호출 규칙, 어셈블리 등 seL4에서 더 이상 필요 없는 부분들은 모두 제거되었다. L4 마이크로커널에서 사용하던 레이지 스케줄링은 실시간 기반 시스템에 사용되는 seL4에 적합하지 않아 베노 스케줄링으로 대체 되었고, 우선순위를 무시한 채 커널이 프로세스 간의 전환을 시도하던 직접적인 프로세스 전환 방식은 우선순위 기반 전환으로 대체되었다. L4 마이크로커널에서 사용되던 비선점형 스케줄링 방식은 정형 검증을 위해 seL4에서도 그대로 유지되었다.

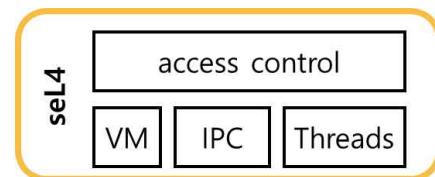
L4 마이크로커널에서 seL4 마이크로커널로 바뀌면서, 설계, 구현 부분에서 많은 변화가 있었고, 특히 L4 마이크로커널에서 파생된 seL4, OKL4, Fiasco.OS 등의 커널들은 최소성 원칙을 지키기 위해 대체되거나 추가되거나 삭제된 부분이 많았다. 최소성 원칙의 핵심 결과는 모든 디바이스 드라이버가 사용자 레벨 프로세스로 만드는 것이며, 이것은 모든 L4에서 파생된 커널에 적용되고 있다.

III. seL4 기반 OS 구조



[그림 126] seL4 기반 OS 구조.

그림 1은 seL4 마이크로커널을 기반으로 하는 OS의 구조이다[1]. kernel mode에 있는 IPC, VM등은 seL4 마이크로커널에 존재하는 것이고, 그 이외에 디바이스 드라이버나 파일 서버와 같은 것들은 사용자 모드에서 응용프로그램 같은 형태로 존재하고 있다.



[그림 127] seL4 구성 요소.

그림 2는 seL4 마이크로커널의 구성 요소들을 그린 것이다. seL4의 구성 요소에는 접근 제어, 가상 메모리, IPC, 스레드 등이 있으며, seL4 마이크로커널은 IPC 통신, 하드웨어 예외처리 관리, 프로세스 관리(스케줄링), 메모리 관리 등을 수행한다.

IV. 보안성 및 정형 검증

4.1 보안성

seL4 마이크로커널의 API는 필수적으로 커널 레벨에서 동작하며, 동적인 시스템 구조를 허용한다. 커널의 접근 제어 정책들의 더 높은 수준의 개념은 데이터 읽기, 메시지 송신 등과 같이 각각의 권한들과 추상화 객체들의 집합을 통해 시스템의 접근 제어 설정들을 저장하는 대신 각각의 커널 객체와 capability들로 추상화된다.[3] 권한 제한은 접근 제어 정책이 향후의 실행 상태에 대한 시스템의 구체적인 capability 및 커널 객체에 대한 정적인 안전 근삿값을 명시한다. 즉 시스템은 제어 정책이 예측하는 것보다 더 많은 권한을 얻을 수 없다.

seL4 마이크로커널은 무결성, 기밀성을 만족시킨다. 무결성은 구성 요소가 어떤 행동을 하든 접근 제어 정책이 명시적으로 수정하도록 허용하지 않는 시스템의 데이터를 수정할 수 없다는 것을 의미한다. 기밀성은 적절히 설정된 시스템에서 확실한 권한 없이 실행하거나 다른 요소에 대한 정보를 얻는 것이 불가능한 것을 의미한다. 즉 권한이 없는 스레드가 다른 스레드의 데이터를 읽을 수 없다. seL4에서 정보는 정책에서 명시적으로 허용하는 경우에만 접근할 수 있다. seL4는 접근 제어 정책, 커널 객체, capability 등을 이용해서 무결성과 기밀성을 만족시키고, 입증함으로써 보안성을 보장한다.

4.2 정형 검증

seL4에서 증명하는 것은 기능적 정확성이며, 정형적으로 시스템의 C언어와 커널 코드의 표현 사이의 일치성을 증명한다. 즉 두 코드 사이의 일치성이 증명되면, C언어에서 보안성이 증명될 때, 그 보안성이 커널에서도 증명된다는 것이 보장된다.[4] seL4 마이크로커널에서 정형 검증을 위해 상호작용적이고, 기계 보조적 및 기계 검사 증명 기술을 사용하였다.

먼저 추상적 명세에서 시스템 호출 인수가 이진 형식으로 인코딩되는 방법과 추상적인 논리적 용어로 각 시스템 호출의 효과, 예외처리 또는 VM 오류가 생길 때 발생하는 일을 설명하는 등 외부 인터페이스를 지정하기에 충분한 세부 사항을 확인한다.

다음 Haskell로 정형 검증 도구에서 실행 가능한 형태로의 변형이 이뤄진다. 이때, 생성된 정의에 대한 확신을 찾고 중간자 프로토타입으로만 사용되는 Haskell이 아닌 C언어를 찾기 때문에 정확성이 중요하지 않은 채 변환이 이뤄진다. 이때 실행 가능한 형태에는 최종적인 C 구현에서 있을 것으로 예상되는 모든 데이터 구조 및 구현 세부 정보가 포함된다.

마지막으로 컴파일러와 하드웨어가 옳다는 가정하에 소스코드에서 정형 검증을 통해 seL4가 정형적으로 검증되었다는 것을 입증한다. 정형 검증이 옳음을 증명하기 위해 Hoare 로직과 Isabelle/HOL 검증 도구를 사용한다.

V. 한계점 및 향후 연구과제

현재 seL4는 동기 IPC 모델에 비동기 알림을 이용하여 비동기 IPC 통신을 하고 있는데, 이를 발전시켜 완전한 비동기 IPC 모델을 만들어 비동기 IPC 통신을 하는 것이 남아있다. 시스템의 고수준 보안 분석과의 연결은 현재 비정형적인 상태로 남아 있으며, 통신 코드 정리는 플랫폼이 제공하는 모든 통신 기본 요소를 다루지 않고, 단대단(end to end) 시스템에서 자동화를 위한 연구가 남아있다.

seL4에서 사용하는 구조 기반 접근법은 시스템의

구성 요소 구조에 의해 시행되는 특징들만 표현할 수 있고, 런타임 때 해당 구조가 변경되거나 관심 있는 특징들이 너무 많거나 신뢰할 수 있는 구성요소의 행동에 크게 의존하는 경우 반환되는 경우가 줄어든다는 한계점이 있다.

이러한 한계를 완화하기 위해서는 위 구조에서 신뢰할 수 있는 빌딩 블록으로 사용하기 위해 사전 검증된 보증된 구성요소 라이브러리를 사용한다. 이러한 라이브러리에는, 입력 정제, 출력 필터, 다운 그레이더, 런타임 모니터와 같은 C언어에서 잠재적으로 생성될 수 있는 보안 패턴이 포함될 수 있으며 재사용 가능한 암호화 모듈, 키 저장소, 파일 시스템, 네트워크와 같은 인프라 구성 요소가 포함된다.

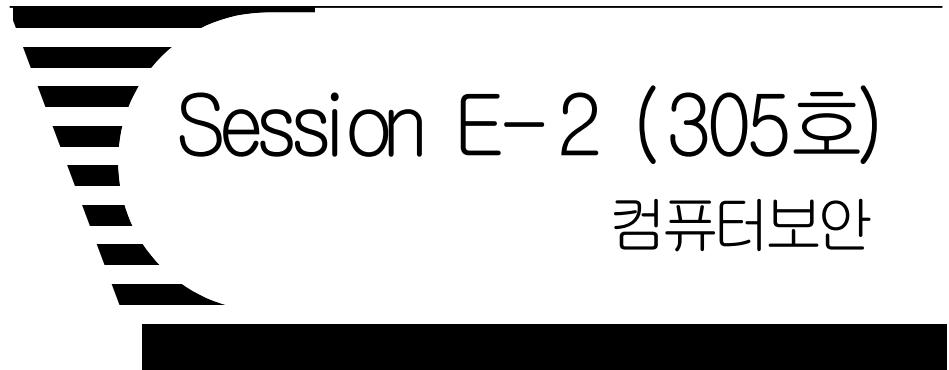
보안 속성이 둘 이상의 구성요소에 의존하는 경우에는 상호 작용 및 구성의 신뢰성에 추론해야 하며, 이때, 신뢰할 수 있는 구성요소가 있는 동시성 추론, 프로토콜 및 정보 흐름 추론을 진행해야 한다.

VI. 결론

본 논문에서는 seL4 마이크로커널이란 무엇인지, L4 마이크로커널에서 seL4 마이크로커널로 변하면서 어떤 부분이 바뀌었는지, seL4 마이크로커널 기반 OS 구조는 무엇인지, 보안성은 어떻게 증명하고, 정형 검증은 어떤 방법을 통해 진행되는지, seL4 마이크로커널의 한계점과 향후 연구과제에 대하여 조사하였다. 본 논문을 통해 연구자들에게 seL4 마이크로커널에 대한 전반적인 지식과 함께 seL4의 동향 파악에 대한 도움을 줄 수 있을 것이다.

[참고문헌]

- [1] Simon Biggs, Damon Lee, and Gernot Heiser , The Jury Is In: Monolithic OS Design Is Flawed: Microkernel-based Designs, Proceedings of the 9th Asia-Pacific Workshop on Systems, ACM, August, 2018.
- [2] Kevin Elphinstone, Gernot Heiser, From L3 to seL4 what have we learnt in 20 years of L4 microkernels?, SOSP '13 Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles, Pages 133–150, November, 2013.
- [3] Gerwin Klein, June Andronick, Ihor Kuz, Toby Murray, Gernot Heiser and Matthew Fernandez, Formally verified software in the real world, Communications of the ACM, 61(10), pp. 68–77, October, 2018.
- [4] Gerwin Klein, Kevin Elphinstone, Gernot Heiser, June Andronick, David Cock, Philip Derrin, Dhammadika Elkaduwe, Kai Engelhardt, Rafal Kolanski, Michael Norrish, Thomas Sewell, Harvey Tuch, and Simon Winwood, seL4: Formal Verification of an OS Kernel, In ACM Symposium on Operating Systems Principles, 207–220, October, 2019.



좌장 : 임강빈 (순천향대)

OpenMP를 활용한 LSH DRBG 병렬 최적 구현

권혁동* 안규황* 권용빈* 서화정*†

*한성대학교 IT응용시스템공학과

korlethean@gmail.com tigerk9212@gmail.com veyoung@gmail.com hwajeong84@gmail.com

LSH DRBG parallel optimization using OpenMP

Hyeok-Dong Kwon* Kyu-Hwang An* Youg-Been Kwon*

Hwa-Jeong Seo*†

*Department of Applied IT Engineering, Hansung University.

요약

결정론적 난수 발생기(Deterministic Random Bit Generator, DRBG)는 미국 표준 기술 연구소(National Institute of Standards and Technology, NIST)에서 권고하는 난수 발생 알고리즘으로 입력에 따라 의사 난수를 생성한다. DRBG의 내부에는 의사 난수 함수를 사용하는데 아직까지 구현된 적 없는 LSH DRBG를 구현하며, 나아가 DRBG의 규격 별로 또는 DRBG의 함수 중 유도함수와 내부 출력 생성 함수에서 반복적으로 호출되는 의사 난수 함수를 OpenMP를 통해 병렬화 처리를 적용하여 최적화를 하고자 한다.

I. 서론

일반적으로 난수 발생기는 무작위 값을 발생시켜야 하나 컴퓨터상에서는 완전 난수 생성이 어려우므로 입력 값에 따라 출력 값이 고정된 결정론적 난수 발생기를 사용한다.

결정론적 난수 발생기의 내부에는 의사 난수 함수를 사용하는데 해당 부분이 반복적으로 호출되며 결과 값은 독립적이므로 병렬화 적용이 가능하다. 이에 따라 국산 해시 함수인 LSH를 사용한 결정론적 난수 발생기를 구현하며 이에 병렬화를 적용하여 효율적인 최적화를 본다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구 동향을 확인하며 3장에서 제안 기법에 대해 살펴보며 4장에서 성능평가를 내리고 5장에서 결론을 맺는다.

II. 관련 연구 동향

2.1 결정론적 난수 발생기 (Deterministic Random Bit Generator)[1]

입력 값에 따라 출력이 고정된 난수 발생기를 의미한다. 알고리즘 내부에 적용하는 의사 난수 함수의 종류에 따라 해시 기반 DRBG와 HMAC 기반 DRBG로 나뉘지만 둘의 내부 구조는 다르다. 본 논문에서 언급하는 DRBG는 모두 해시 기반 DRBG임을 명시하는 바이다.

2.2 LSH (Lightweight Secure Hash)[2]

LSH는 2014년 국가 보안 기술연구소에서 개발한 암호 학적 해시 함수로 Wide-pipe merkle Damård 구조가 적용되었으며 224, 256, 384, 512 비트의 출력 규격을 제공한다. 현재 국내 TTA 표준으로 제정되어 있다.

2.3 SSE (Streaming SIMD Extensions)[3]

SSE는 SIMD(Single Instruction Multiple Data)에 속하는 명령어 셋으로 SIMD의 특성을 활용하여 하나의 명령어로 절차 지향적 데이터를 한 번에 처리할 수 있기에 전체적인 동작 속도를 향상시킨다.

2.4 AVX (Advanced Vector Extensions)[4]

AVX는 2008년 발표된 SIMD 명령어 셋으로 기존의 16바이트에서 증가한 32바이트를 동시에 계산 가능하다. 특히 부동소수점 연산이 뛰어나기 때문에 모델링, 실험, 그래픽 분야에서 유용하게 활용할 수 있다.

III. 제안 기법

DRBG에 병렬화를 적용하는 방법에는 데이터 병렬화와 태스크 병렬화 두 가지 방법이 있으며 이에 대해 서술한다.

3.1 데이터 병렬화 (Data Parallelism)

LSH는 총 6개 규격을 지원하며 후행 규격은 선행 규격의 출력을 대기하므로 병렬화를 거친다면 성능 향상을 기대할 수 있다. 단, 모든 규격별로 작업 시간이 동일하지는 않기 때문에 프로그램이 종료되기 위해서는 모든 규격이 완료되기를 대기해야 하며 이러한 요구 사항을 적용한 모델이 [그림. 1]로 각 규격마다 병렬적으로 동작함을 확인할 수 있다.

Algorithm 1

1. read test vectors
2. loop count = number of hash output types
3. operate OpenMP to for loop
4. For i = 1 to loop count
 - 4.1 DRBG using hash[i] type
 - 4.2 write DRBG result
 - 4.3 waiting until other processing finished
5. DRBG finished

[그림. 130] 데이터 병렬화가 적용된 LSH DRBG

Algorithm 2

- 1.1 If hash bits is 224 or 256
seed bits = 440
- 1.2 If hash bits is 384 or 512
seed bits = 888
2. len = ceil (seed bits / hash bits)
3. counter = 0x01
4. input = counter || seed bits || msg from parameter
5. operate OpenMP to for loop
6. For i = 1 to len
 - 6.1 output[i] = Hash(input)
 - 6.2 counter += 1
 - 6.3 refresh input value
 - 6.4 waiting until other processing finished
7. final output = (output[1] || ... || output[n]) mod 2^{seed bits}

[그림. 131] 태스크 병렬화가 적용된 유도함수

3.2 태스크 병렬화 (Task Parallelism)

DRBG에서 유도함수와 내부 출력 생성 함수

의 반복적인 의사 난수 함수 호출 부분을 병렬화한다. 반복 횟수 산정은 유도함수에서는 규격에 따라 2회, 3회로 고정되며 내부 출력 생성 함수는 출력 길이에 따라 비례한다. 각 함수마다 병렬화가 적용된 모델은 [그림. 2], [그림. 3]과 동일하다.

Algorithm 3

1. len = ceil (output bits from parameter / hash bits)
- 2.1 If hash bits is 224 or 256
seed bits = 440
- 2.2 If hash bits is 384 or 512
seed bits = 888
3. data = state V from parameter
4. operate OpenMP to for loop
5. For i = 1 to len
 - 5.1 output[i] = Hash(data)
 - 5.2 data = (data + 1) mod 2^{seed bits}
 - 5.3 waiting until other processing finished
6. final output = (output[1] || ... || output[n]) mod 2^{output bits}

[그림. 132] 태스크 병렬화가 적용된 내부 출력 생성 함수

IV. 성능 평가

성능 평가에 앞서 DRBG 구현 및 작업 환경은 [표. 1]과 동일함을 명시하며 테스트에 사용한 구현물은 깃허브¹⁾에서 열람이 가능하다.

[표. 163] 작업 환경

운영 체제	Windows 10 Pro
프로세서	Intel Core i7-8550U CPU 1.8GHz
컴파일러	MinGW
IDE	Eclipse Photon (4.8.0)
언어	C

4.1 데이터 병렬화 성능 평가

LSH 규격별로 병렬화를 적용하여 6개 규격이 동시에 출력을 진행하도록 한다. 모든 규격 출력이 완료된 시점을 1회로 설정하여 테스트 벡터 60종을 1,000회 반복하며 예측 내성 지원 설정을 한다. 대조군은 동일 환경에서 병렬화를 적용하지 않으며 실험 결과는 [표. 2]와 같다.

[표. 78] 데이터 병렬화 비교

	평균 수행 시간(ms)	cpb
대조군	73	491
병렬화	27	181

객관적인 평가를 위해 수행 시간을 CPB(Clo

1) https://github.com/korlethean/drbg_with_mp

ck cycle Per Bytes)로 환산한다. 비교 결과 병렬화 적용 모델이 약 2.71배의 효율을 보인다.

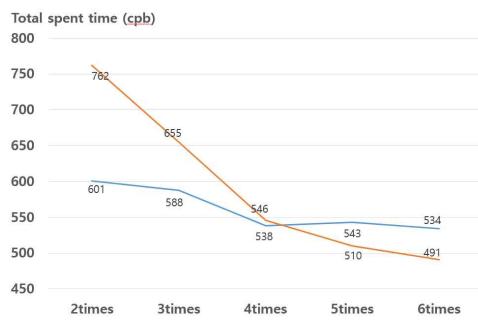
4.2 태스크 병렬화 성능 평가

DRBG의 내부 구조에서 의사 난수 함수를 반복적으로 호출하는 부분에 병렬화를 적용한다. 4.1절과 동일한 환경에서 진행하며 비교 결과는 [표. 3]과 같다.

[표. 79] 태스크 병렬화 비교

	평균 수행 시간(ms)	cpb
대조군	73	491
병렬화	149	1001

태스크 병렬화를 적용할 경우 오히려 대조군에 비해 약 2.04배 저하된 성능을 보여준다. 이는 병렬화 과정에서 발생하는 오버헤드로 인한 것으로 판단되었다. 오버헤드는 필연적으로 발생하기 때문에 반복 횟수를 증가시키면 효율이 증가한다. 이에 따른 결과는 [그림. 4]와 같다.



[그림. 133] 출력 길이별 비교 그래프

출력 길이가 기존 대비 5배 이후일 때 병렬화 모델의 동작 효율이 좋으며 출력 길이가 증가하더라도 동작 효율의 변화가 크지 않음을 확인 가능하다.

V. 결론

본 논문에서는 DRBG 구조에 병렬화를 적용하여 최적화 구현을 시도하였고 그 성능이 통상적인 DRBG 동작보다 뛰어남을 확인하였다.

데이터 병렬화는 다수 규격의 DRBG 출력에 유리하며 태스크 병렬화는 표준 규격에는 부적합하지만 이론상으로 병렬화 적용을 통해 성능 향상이 가능함을 보였다.

병렬화 기법은 성능 개선을 위한 방법 중 하나로 다양한 분야에 활용된다. 이때 병렬화를

적용하면 필연적으로 오버헤드가 발생한다는 점을 항상 염두에 두며 맹목적인 적용이 아닌 적절한 병렬화 적용을 시도해야만 의도한 성능 향상을 이끌어 낼 수 있다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터 (UD170109ED)를 통한 방위사업청과 국방과학 연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- National Institute of Standards and Technology, “Recommendation for Random Number Generation Using Deterministic Random Bit Generators”, Available: <https://www.nist.gov/publications/recommendation-random-number-generation-using-deterministic-random-bit-generators-2>
- Telecommunications Technology Association, “Hash function Full structure and compression function of LSH”, Available: http://commitee.tta.or.kr/data/standard_view.jsp?nowPage=2&pk_num=TTAK.KO-12.0276&commit_code=TC5.
- Srinivas K. Raman, Vladimir Pentkovski, Jagannath Keshava, “Implementing Streaming SIMD Extensions on the Pentium III processor”, Available: <https://www.computer.org/csdl/mags/mi/2000/04/m4047.pdf>
- Chris Lomont, “Introduction to Intel® Advanced Vector Extensions”, Available: http://software.intel.com/sites/default/files/m/d/4/1/d/8/Intro_to_Intel_AVX.pdf

바이너리 분석 모듈을 이용한 Driller의 접근법 분석

김주환*, 유지현*, 윤주범*

*세종대학교 정보보호학과

kjh97852003@naver.com, yjhy8783@naver.com, jbyun@sejong.ac.kr

Driller's approach analysis using binary analysis module

Juhwan Kim*, Jihyeon Yu*, Joobeom Yun*

*Department of Computer and Information Security, Sejong University.

요약

현대 사회에서 바이너리 분석은 시스템 보안이라는 분야에서 매우 중요한 주제로 남아 있다. 바이너리의 구조에 대해 많은 정보들은 프로그램 분석에 대해 어렵게 만들며, 바이너리 분석은 실제로 컴파일된 코드에 대하여 증명할 수 있는 유일한 방법이다. 본 논문에서는 소프트웨어 취약점을 이용한 자동화 공격 방어 시스템 대회인 Cyber Grand Challenge에서 3위를 차지한 shellphish팀의 Mechaphish 시스템의 구성 요소 중 차세대 바이너리 분석 시스템인 angr와 소프트웨어 테스팅 기법인 퍼저를 결합한 시스템인 Driller 모듈을 분석하고자 한다.

I. 서론

현대 사회에서 바이너리 분석은 시스템 보안이라는 분야에서 매우 중요한 주제로 남아 있다. 특히 바이너리 코드는 운영체제 구조와 성능에 중요한 어플리케이션으로 컴파일되는 경우가 많으며, 컴파일이 되기 전의 소스 코드를 분석하여 보안에 관한 취약점이 없다는 것을 입증하더라도 컴파일이 된 후에는 유지되지 않을 수 있다[1]. 이처럼 바이너리 분석이 대두되면서 바이너리의 취약점을 찾는 시스템과 결합을 한다거나, 더 나아가 찾은 취약점을 이용하여 익스플로잇(exploit)까지 생성하는 연구들[2],[3]이 진행되고 있다.

본 논문에서는 미국의 DARPA에서 진행한 소프트웨어 취약점을 이용한 자동화 공격 방어 시스템 대회인 Cyber Grand Challenge에서 3위를 차지한 shellphish팀의 Mechaphish[4] 시스템의 구성 요소 중 Driller[3] 모듈을 분석하고자 한다. 이는 차세대 바이너리 분석 시스템인 angr[1]와 소프트웨어 테스팅 기법인 퍼저(fuzzer)를 결합한 시스템이며, 소프트웨어의 취약점을 발견하는데 큰 기여를 했다.

II. 연구 배경

2.1 angr

angr는 기존에 연구되었던 접근법들이 하나로 통합된 바이너리 분석 모듈로 사용자가 바이너리를 분석 시스템에 로드하여 제어할 수 있다. angr는 다양한 바이너리 포맷을 가상 주소 공간에 표현하기 위해 메모리 이미지를 추출하는 CLE(CLE Loads Everything)[5] 로더를 사용하며, 바이너리 영역의 주소 뿐만 아니라 같이 로드된 공유 라이브러리 등에 대해서도 주소 공간을 확인할 수 있다. angr는 기본 블록(basic block)의 단위로 코드를 분석하기 때문에 이러한 주소들을 기반으로 디스어셈블리(disassembly)하여 기본 블록을 추출할 수 있으며, 바이너리의 특정 시점을 시뮬레이트하여 기호 실행(symbolic execution)을 수행할 수도 있다. 예를 들어, 특정 시점 B에 도달하기 위해서 입력 값이 요구되면 기호 실행을 통해 특정 시점 A에서 단계를 하나씩 밟아 특정 시점 B에 도달하기 위한 표준 입력 값을 찾을 수 있다. 또한 특정 시점의 레지스트리나 메모리, 파일 시스템 데이터를 확인하거나 주어진 주소들을 기반으로 CFG(Control Flow Graph)를 표현할 수 있으므로 바이너리 분석에 용이하다.

2.2 Driller

Driller는 앞서 말했듯이, 소프트웨어 테스팅 기법인 퍼저와 바이너리 분석 모듈인 angr를 결합한 시스템이다. 일반적인 퍼저는 무작위 입력 값을 바이너리

리에 입력하여 크래시(crash)를 발견하는 것을 목표로 하지만, 어플리케이션의 어느 경로를 목표로 해야 하는지에 대한 제어가 없기 때문에 낮은 코드 커버리지를 가진다는 단점이 있다. 반면에 기호 실행을 동적으로 실행하는 콘콜릭 실행(concolic execution)은 바이너리에 기호 변수를 사용하여 조건문과 같은 제약 조건을 찾고 해를 구하여 특정한 경로에 대한 입력 값을 모델링할 수 있다. 하지만 콘콜릭 실행은 조건문의 분기 같은 경우에 참과 거짓의 상태 모두를 충족하게 됨으로써 두 경로를 모두 탐색해야 한다[3]. 이는 바이너리의 크기가 거대할수록 경로 폭발(path explosion)의 문제를 야기할 확률이 높아지므로 시스템의 자원이 고갈되는 현상이 발생할 수 있다. Driller는 이 두 가지 기존 접근법들의 단점을 보완하여 퍼저를 기반으로 선택적인 콘콜릭 실행을 하였다. AFL Fuzz[6]를 사용하여 퍼징을 수행한 후에 더 이상 퍼저가 새로운 상태 전환을 찾지 못한다면, 퍼저가 생성한 입력 값을 기반으로 콘콜릭 실행이 호출된다. 이는 기존 입력 값의 흐름대로 조건문이 있는지 먼저 찾고, 조건문을 발견하면 기존 입력 값으로 이어지는 상태와 반대되는 상태로 진입하기 위한 입력 값을 생성하여 퍼저에게 전달한다. 결과적으로 퍼저는 새로 생성된 입력 값을 기반으로 다시 퍼징을 수행하므로 새로운 상태를 찾게 되고 위의 과정을 반복하게 된다.

III. Driller의 설계

3.1 SimState

Driller에서 사용하는 angr는 앞서 말했듯이, 바이너리 분석 모듈이다. 이는 바이너리의 특정 시점을 시뮬레이트 할 수 있는데 이 시점을 하나의 프로그램 상태(program state)라 칭하며, 특정 객체인 SimState로 작업하는 것이다. [그림 1]은 .entry_state() 함수를 호출함으로써 바이너리의 entry point에서 실행할 준비가 된 상태를 초기화하는 과정이다. 상태 생성자들에 대한 함수는 다음과 같다.

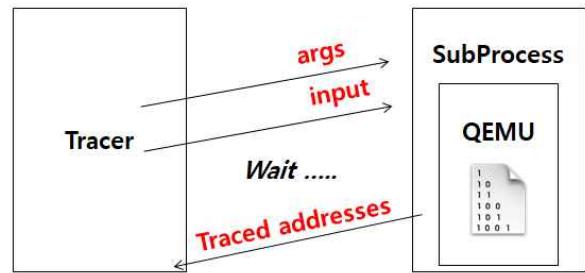
- .blank_state() : 이 생성자는 비어 있는 상태를 만들며, 대부분의 데이터(레지스터, 메모리, 파일 시스템 등)는 초기화되지 않은 상태로 남아있다. 특정 주소를 매개 변수로 주면 상응되는 상태가 생성된다.

- .entry_state() : 이 생성자는 메인 바이너리의 entry point에서 실행할 준비가 된 상태를 생성한다.

- .full_init_state() : 이 생성자는 메인 바이너리의 entry point 이전에 실행되어야 하는 모든 초기화를 통해 실행하기 위한 준비된 상태를 생성한다. 예를

```
In [1]: import angr
In [2]: p = angr.Project('./kisa/dataset1/q02', auto_load_libs=False)
In [3]: state = p.factory.entry_state()
In [4]: state
Out[4]: <SimState @ 0x8048340>
```

[그림 134] SimState 객체의 초기화



[그림 135] Tracer 개요

들어, 공유 라이브러리 생성자를 수행하고 entry point로 전너뛴다.

- .call_state() : 이 생성자는 매개 변수로 주어진 함수를 실행할 준비가 된 상태를 생성한다.

Driller에서는 .full_init_state() 함수로 초기 상태를 생성하며, 해당 상태를 기반으로 기호 실행을 할 준비를 마친다.

3.2 Tracer

Driller에서 사용하는 Tracer 모듈은 [그림 2]와 같이 QEMU[7]를 사용하여 동적으로 바이너리의 주소들을 추적하는 과정이다. 여기서 QEMU란 소프트웨어 전체를 가상 머신 위에서 실행하는 가상화 소프트웨어로 Driller에서는 자체적으로 수정한 QEMU를 사용하였다. 퍼저가 더 이상 상태를 찾지 못하면, 콘콜릭 실행이 호출되는데 그 사이에 Tracer 모듈이 먼저 실행되어 바이너리에 동적 트레이싱(dynamic tracing)을 수행한다. 바이너리 경로와 퍼저가 무작위로 생성한 입력 값을 QEMU에게 제공하면 해당 입력 값으로 인해 바이너리에 거치게 되는 기본 블록의 진입 주소들을 반환받게 되며, 이들은 Driller의 핵심 기술인 기호 실행에서 참조할 주소들로 사용된다.

3.3 Simulation Manager

Driller에서 가장 중요한 인터페이스로 사용되는 simulation manager는 프로그램의 상태 공간을 탐색하며, 기호 실행을 제어할 수 있다. 이를 통해 프로

```
In [3]: state = p.factory.entry_state()
In [4]: simgr = p.factory.simgr(state)
In [5]: simgr
Out[5]: <SimulationManager with 1 active>
In [6]: while True:
...:     simgr.step()
...:     if len(simgr.active) >= 2:
...:         break
...:
In [7]: simgr
Out[7]: <SimulationManager with 2 active>
In [8]: simgr.active
Out[8]: [<SimState @ 0x80484b9>, <SimState @ 0x80484e3>]
```

[그림 3] Simulation manager를 이용한 기호 실행

[그림 137] 실제 바이너리를 대상으로 한 Drilling 과정

그램의 특정 시점이 초기화된 상태를 기준으로 앞으로 나아갈 수 있다. 이러한 기능으로 조건문과 같이 분기로 나뉘어지는 기본 블록에 도달했을 때, simulation manager는 [그림 3]과 같이 두 가지 상태를 가지게 된다. 이처럼 simulation manager는 여러 상태를 다룰 수 있으며, 이 상태들은 stashes라 불리는 메소드로 조직화된다. 기본적인 stash는 active stash이며, 시뮬레이션을 수행할 때마다 stash가 바뀔 수 있다. 예를 들어, 하나의 상태가 더 이상 유효하지 않은 명령이나 유효하지 않은 instruction pointer를 포함하게 된다면 더 이상 시뮬레이션을 할 수 없기 때문에 active stash 내의 상태가 deadended stash로 이동하게 된다. stash는 사용자가 새롭게 정의할 수 있으므로 Driller에서는 diverted라는 stash를 생성하여 퍼저가 생성한 입력 값으로 인해 진입되는 상태의 반대 상태를 찾아 diverted stash로 이동시켰다.

Simulation manager의 탐색 기법은 사용자가 정의 할 수 있는 여러 기법이 있다. 즉, 기호 실행을 수행하기 전에 프로그램 상태가 탐색되는 패턴을 수정하는 것이다. Driller는 Tracer와 DrillerCore를 사용하는데 이 탐색 기법들 중 모듈 이름과 동일한 Tracer 기법은 이전에 설명한대로 미리 동적 트레이싱한 주소들을 매개 변수로 주어서 해당 주소들을 따라 실행을 유발한다. 퍼저가 생성한 입력 값을 기반으로 거치게 된 주소들만을 실행하므로 바이너리의 진입점부터 순차적으로 코드 커버리지를 넓힐 수 있게 된다. DrillerCore는 새로운 상태 전환을 찾기 위해 입력 값을 기호적으로 따라가는 탐색 기법이다. 이는 Tracer 기법과 함께 사용해야 하며, 앞서 설명한 diverted stash를 생성하는 역할을 한다. 결과적으로 Driller는 diverted stash가 발견되면, 해당 상태로 진입할 수 있는 표준 입력 값을 생성하며, 이를 퍼저에게 전달하고 기호 실행을 종료한다.

IV. 실험 결과

[그림 4]는 조건문에 특정한 입력 값이 정확히 일치해야만 분기 안으로 진입할 수 있는 바이너리를 대상으로 Driller를 수행한 과정이다. 현재 active stash와 반대되는 diveted stash를 발견하고 해당 상태를 진입하기 위해서 “SEJONG_Netsec_Lab”을 포함한 입력 값을 찾은 것을 알 수 있다.

V. 결론

본 논문에서는 Driller에 대한 핵심적인 접근법을 분석하기 위해 전체적인 흐름과 기술적인 내용을 분석하였으며, Driller가 내재하고 있는 바이너리 분석 모듈을 어떻게 활용했는지에 대해서도 분석하였다. 끝으로 Driller의 실제 바이너리의 실험 결과를 [그림 4]를 통해 보였다.

VI. Acknowledgements

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학 ICT연구센터지원사업의 연구결과로 수행되었음(IITP-2018-2018-0-01423).

[참고문헌]

- [1] Y.Shoshtaishvili, R.Wang, C.Salls, N.Stephens, M.Polino, A.Dutcher, J.Grosen, S.Feng, C.Hauser, C.Kruegel and G.Vigna, "(State of) the art of war: offensive techniques in binary analysis." IEEE Symposium on Security and Privacy. 2016.
 - [2] S.K.Cha, T.Avgerinos, A.Rebert, and D.Brumley, "Unleashing mayhem on binary code," In Proceedings of the IEEE Symposium on Security and Privacy, 2012.
 - [3] N.Stephens, J.Grosen, C.Salls, A.Dutcher, R.Wang, J.Corbetta, Y.Shoshtaishvili, C.Kruegel, and G.Vigna, "Driller: Augmenting fuzzing through selective symbolic execution," Network and Distributed System Security Symposium, 2016.
 - [4] Y.Shoshtaishvili, A.Bianchi, K.Borgolte, A.Cama, J.Corbetta, F.Disperati, A.Dutcher, J.Grosen, P.Grosen, A.Machiry, C.Salls, N.Stephens, R.Wang and G.Vigna, "Mechanical phish: resilient autonomous hacking," IEEE Security and Privacy Magazine, 2018.
 - [5] CLE, <https://github.com/angr/cle>
 - [6] American Fuzzy Lop,
<http://lcamtuf.coredump.cx/afl/>
 - [7] QEMU,
<https://qemu.weilnetz.de/doc/qemu-doc.html>

암호 알고리즘에 대한 부채널 분석 기술 동향

박찬희*, 이진재*, 김호원*

*부산대학교 전기전자컴퓨터공학과

{chan70921, wlswo3149, howonkim}@pusan.ac.kr

Side Channel Analysis Technique Trends for Cryptography Algorithm

Chan-Hui Park*, Jin-Jae Lee*, Ho-Won Kim*

*Department of Electrical and Computer Engineering,

Pusan National University

요약

부채널분석은 하드웨어에서 암호 알고리즘이 동작할 때 발생하는 소비전력 또는 전자기파를 수집/분석하여 비밀키를 알아내는 강력한 공격 방법으로 알려져 있다. 하드웨어 기술의 발달과 사물인터넷 기술의 보편화로 인해 부채널분석의 위험성은 점차 증가하고 있다. 본 논문에서는 다양한 부채널 분석 기법과 이를 검증할 수 있는 시스템에 대해서 기술한다.

I. 서론

최근 하드웨어 기술의 발달과 함께 사물인터넷 기기들이 보편화되고 있다. 신용카드에 부착된 IC칩, 전자여권, 스마트폰, 스마트워치 등 사용자의 개인정보가 저장된 디바이스는 보안취약점이 발견될 경우 큰 혼란을 야기시킬 수 있다.

사물인터넷 환경에서 개인정보를 보호하기 위해 다양한 암호 알고리즘들이 사용되고 있다. 대표적으로 AES, RSA, ECC, SHA-2와 같은 암호 알고리즘들이 사용되고 있지만, 이러한 암호 알고리즘의 경우 취약점이 존재하고 보안취약점을 악용하여 중요정보를 탈취한 사례들이 증가하고 있다.

그중에서도 하드웨어로부터 발생하는 전력 또는 전자파를 탐지하여 암호 모듈 동작시 중요정보를 해독하는 부채널분석에 대한 위협이 점차 증가하고 있는 추세이다. 부채널분석은 1996년 Kocher가 제안한 개념으로 연산에 걸리

는 시간을 측정하여 비밀키 정보를 획득하는 소요시간 분석(Timing Attack)[1]을 제안하였으며, 그 후 소비전력분석 공격(Power analysis Attack)[2], 전자파분석 공격(Electromagnetic Attack)[3] 등 다양한 공격 기법들이 있으며, 최근에는 공격기법뿐만 아니라 부채널분석에 대응할 수 있는 대응기법들에 대한 연구가 활발히 진행되고 있다.

부채널분석에 대한 취약점이 점차 알려지게 됨에 따라 IT 제품에 대한 안전성을 검증하고 이에 대한 평가 방법의 필요성이 대두되면서 국내외에서도 보안 취약성을 평가하기 위한 방법론 및 기준에 대한 연구가 진행되고 있다.

대표적으로 CC(Common Criteria), FIPS(Federal Information Processing Standard) 140-3, EMV(Electromagnetic Vulnerability), PCI PED를 통해 부채널분석에 대한 취약성 평가를 실시하고 있다.

II. 부채널분석 기법

부채널분석은 크게 침투형(Invasive)와 비침투형(non-Invasive)로 나눌 수 있다. 침투형 분석의 경우 칩의 표면을 제거하고 탐침을 함으로서 이루어진다. 본 논문에서는 비침투형 부채널분석 기법인 전력분석기법과 전자파분석기법에 대해서 기술한다.

2.1 SPA(Simple Power Analysis)

SPA는 암호 모듈이 동작할 때 전력 소모량을 한번만 측정한 후 비밀 키의 값을 알아내는 공격이다. 공개키 암호 RSA를 분석할 경우 아래 그림 1과 같은 패턴을 획득할 수 있다.

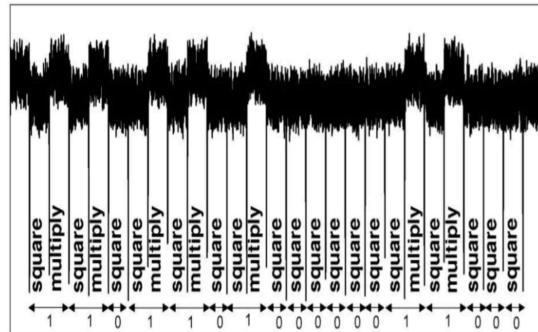


그림 138 RSA 연산시 전력 소모 패턴

비밀키의 비트가 0일 때 Square 연산, 1일 때 Square와 Multiply 연산이 수행되기 때문에 위와 같은 패턴을 획득할 수 있으며, 패턴을 분석하여 비밀키를 유추하는 것이 가능하다.

2.2 DPA(Differential Power Analysis)

DPA는 다수의 전력소모패턴을 수집하고 통계적인 방법을 통해 전력소모량의 평균을 계산한 후 비밀 키를 유추한다. 이 방법의 경우 잡음을 제거하고 다수의 데이터를 활용하기 때문에 SPA보다 유의미한 결과를 획득할 수 있다.

2.3 EM(Electromagnetic Analysis)

EM은 암호 모듈이 동작할 때 kftod하는 전자기파를 수집/분석하여 비밀키의 정보를 알아내는 방법이다. 최근에는 EM을 활용하여 안드로이드 스마트폰에서 동작하는 암호알고리즘의

비밀키를 알아내는데 성공하면서 본 공격의 위험성이 증명되었다[4,5].

III. 부채널분석 검증 시스템

본 장에서는 부채널분석이 가능한 검증 보드 및 시스템에 대해서 기술한다. 초기 부채널분석을 위한 장비의 가격이 고가에 형성되어 있었으나 최근 ChipWhisperer와 같은 저가의 장비가 출시되면서 부채널분석에 대한 연구가 활발히 진행될 것으로 예측된다.

3.1 ChipWhisperer[7]



그림 139 ChipWhisperer-Lite 보드

캐나다회사 NewAE Technology에서 2013년 출시된 저가의 부채널분석이 가능한 보드로 오픈소스로 모든 소스코드와 회로도가 깃허브를 통해 공개되어 있는 것이 가장 큰 특징이다.

AES, DES, RSA에 대한 분석이 가능하며, SASEBO-W 보드를 활용하면 스마트카드에 대한 분석도 가능한 것으로 알려져 있다.

3.2 DPA Workstation[6]

DPA Workstation은 전력 분석 및 전자기파 분석, 오류주입이 가능한 부채널분석 장비로서 AES, RSA, ECC, DES, SHA 암호 알고리즘에 대한 분석이 가능하다. 또한 데이터 수집 및 신호처리, 그리고 시각화 기능을 포함하고 있으며, 소스코드가 제공되어 사용자가 유연하게 소스코드를 수정하여 활용가능하기 때문에 기능을 추가하거나 수정하여 성능향상이 가능하다.

3.3 SCARF[8]

한국전자통신연구원(ETRI)에서 개발한

SCARF는 소비전력, 전자기파, 오류주입 등 부채널 정보 분석이 가능한 장비이다. 안전성 검증보드와 소프트웨어 검증보드 3종, 하드웨어 검증보드 1종, 카드타입 검증보드 2종으로 이루어져 있으며, 파형수집, 시각화를 제공하고 있다. DES, AES, SEED, ARIA, RSA, ECC에 대한 분석이 가능하다.

IV. 결론

본 논문에서는 암호 알고리즘에 대한 부채널 분석에 대한 연구 동향에 대해 기술하였다. 이와 같이 부채널 분석에 대한 위험성은 점차 증가하고 있다. 소비전력 또는 전자기파를 수집할 수 있는 장비와 기술을 통해 각종 디바이스 또는 스마트폰의 비밀정보를 획득할 수 있으며 이를 통해 사용자의 개인정보를 획득하는 것이 가능하다. 따라서 부채널공격에 대응하기 위한 기법들에 대한 연구가 활발히 이루어질 것으로 예상된다.

[참고문헌]

- [1] Kocher, Paul C. "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 1996.
- [2] Kocher, Paul, Joshua Jaffe, and Benjamin Jun. "Differential power analysis." Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 1999.
- [3] Quisquater, Jean-Jacques, and David Samyde. "Electromagnetic analysis (ema): Measures and counter-measures for smart cards." Smart Card Programming and Security. Springer, Berlin, Heidelberg, 2001. 200-210.
- [4] Goller, Gabriel, and Georg Sigl. "Side channel attacks on smartphones and embedded devices using standard radio equipment." International Workshop on Constructive Side-Channel Analysis and Secure Design. Springer, Cham, 2015
- [5] Genkin, Daniel, et al. "ECDSA key extraction from mobile devices via nonintrusive physical side channels." Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2016
- [6] Rambus homepage,
<https://www.rambus.com/security/dpa-countermeasures/dpa-workstation-platform/>
- [7] ChipWhisperer,
<https://newae.com/tools/chipwhisperer/>
- [8] Kim, Juhan, et al. "SCARF: profile-based side channel analysis resistant framework." Proceedings of the International Conference on Security and Management (SAM). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 2012.

위협 정보 표현 규격 STIX 2.0을 이용한

IDS 로그 표현에 관한 연구

유지현*, 김주환*, 윤주범*

*세종대학교 정보보호학과

yjhy8783@naver.com, kjh97852003@naver.com *jbyun@sejong.ac.kr

A Study on the IDS log expression using STIX 2.0

Jihyeon Yu*, Juhwan Kim*, Joobeom Yun*

*Department of computer and information security, Sejong University

요약

본 논문은 국내에서는 미국과 같은 국가적인 차원의 기반시설 보호를 위해 보안 로그에 관한 표준화나 시스템 간 정보 교환 규격에 대한 가이드가 부족한 상태임을 인지하고, 정보공유체계에서 CTI(Cyber Threat Intelligence)정보 표현 규격으로 현재 미국에서 정보 표현 규격으로서 표준화가 진행 중인 STIX v2.0을 제시한다. STIX v2.0은 미국에서 사이버 위협 정보를 표현 및 공유하기 위해 개발한 표현 규격이다. 기존 레거시 탐지체계에서의 정보들은 표현방법이 상이하기 때문에 정보 공유 및 분석에서 어려움이 있다. STIX v2.0을 통해 일관된 표현을 정보공유 시에 사용한다. 이는 정보공유체계에서 호환성, 가독성, 일관성 등에서 효과적이고 국가적인 차원에서 사이버 위협에 대한 신속한 대응이 가능할 것이다. 본 논문에서는 다양한 탐지 정보들 중에서 IDS 로그를 STIX v2.0을 이용하여 표현하는 방안에 대해 제시한다.

I. 서론

최근 국방망 해킹, 이란의 STUXNET 등의 주요기반시설 침해사고가 발생하여 사이버 위협에 대한 국가적인 보안이 중요시 되고 있다. 주요 정보통신기반시설에 사이버 위협이 가해지는 경우 최대 심각한 국가적 혼란까지 예상되기 때문에 사이버 위협 침해사고에 대한 대비로서 주요시설 간에 정보공유는 매우 중요하다. 현재 국내에 있는 기존 레거시 장비체계는 정보표현 방법이 각각 상이하기 때문에 정보 공유 시에 어려움이 있고 이를 근본적으로 해결하기 위해 장비 교체를 하는 것은 많은 비용이 드는 부담이 있다.

본 논문에서는 세계적으로 표준화되고 있는 사이버 보안 위협 표현 STIX v2.0에 대해 조사하여 현재 다른 국가에 비해 빈약하다고 판단되는 IDS 정보공유체계의 정보 표현 및 공유 규격으로 STIX v2.0을 제시한다.

II. 관련 연구

2.1 STIX

미국의 국토안보부는 사이버 위협에 대응을 위해 효율적이고 안전한 정보공유 체계 구축의 필요성을 인지하였으나, 사이버 위협 정보가 표준화가 되지 않아 일관성 있는 분석의 어려움을 느끼고 이를 극복하기 위해 2012년부터 규격개발에 착수하였다. 이에 미국의 국토안보부는 MITRE 조직을 통해 2013년 4월에 사이버 위협 정보 표현 규격인 STIX(The Structured Threat Information eXpression) v1.0을 발표했다[1][2]. 계속해서 새로운 버전들을 공개하였고 16년 5월에 발표된 STIX v1.2.1 부터는 OASIS CTI TC에서 관리하고 있다. OASIS CTI TC는 지능형 사이버 위협 대응의 모델링, 분석, 공유의 필요성을 해결하고자 설립되었다. 이는 일련의 정보 표현과 프로토콜을 정의하며, OASIS 공개 표준 프로세스에 따라 개발 및 표준화를

위해 미국의 국토안보부에서 발표한 세 가지 규격으로 전환되었다[3]. 이 세 가지 규격은 STIX, CYBOX(Cyber Observable Expression), TAXII(Trusted Automated Exchange of Indicator Information)이다. STIX는 정보 공유 시의 사이버 위협이라고 확신되는 정보를 표현하기 위한 규격, CYBOX는 모든 관찰된 정보를 표현하기 위한 규격, TAXII는 STIX로 표현된 정보를 자동으로 교환하기 위한 송수신을 하는 서버 규격으로서 각각의 기능을 담고 있다.

최근 STIX의 정보 표현 범위를 늘리기 위해 STIX, CYBOX를 결합한 STIX v2.0이 공개되었고, 본 논문에서는 가장 최근 발표된 STIX v2.0을 사용한 IDS 로그 정보 표현을 제시한다.

2.2 STIX v2.0

STIX v2.0은 지능형 사이버 위협 대응을 교환하는데 사용되는 언어 및 JSON 직렬화 포맷이다. 이를 통해 조직들은 일관성 있고 기계 판독이 가능한 방식으로 서로 사이버 위협 정보를 공유할 수 있으므로, 보안 커뮤니티들은 물리적인 위협에 비해 가능성이 높은 컴퓨터 기반의 공격들을 더 잘 이해할 수 있을 것이다. 이에 이들은 이러한 공격들을 보다 빠르고 효과적으로 보고 예측하며 응답할 가능성이 높다. STIX v2.0은 공동 위협 분석, 자동화 위협 교환, 자동화 탐지 및 대응 등과 같은 다양한 기능을 설계하였다.

STIX v2.0의 데이터 모델은 CTI에서의 개념을 대표적으로 표현하기 위해 12개의 SDO(STIX Domain Objects)를 정의한다. 이는 목차와 내용을 나누는 것처럼 각 객체에 해당 객체에 대한 속성, 정보를 포함하여 내용 파악, 공유 면에서 용이하게 설계하였다. 또한 CTI 개념을 설명하는데 사용되는 객체 간의 관계를 나타내기 위해 2개의 SRO(STIX Relationship Objects)를 정의한다. 예를 들어, 악의적인 의도를 가지고 있는 것으로 판단되는 개인, 조직을 표현하는 객체인 Threat Actor 객체와 공격 패턴을 표현하는 객체인 Attack-pattern 객체의 관계를 Threat Actor 'uses' Attack-pattern 이

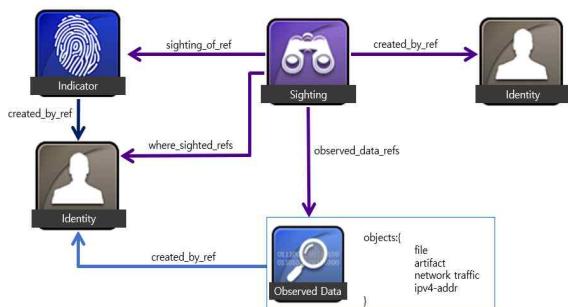
며, 이 때 Relationship Type인 'uses'를 나타내기 위해 SRO를 사용한다[4].

III. STIX 2.0을 이용한 IDS 로그 표현

현재 기존 레거시 장비에서 구동되는 탐지체계는 각각 표현방법이 상이하다. 기존에 쓰던 탐지체계를 바꾸는 비용을 줄이기 위해 정보를 전달할 때에는 STIX로 탐지정보를 표현하여 공유하는 방법이 적절하다. STIX v2.0을 이용한 정보 표현 객체와 이들의 관계를 제안하는 바는 [그림 1]과 같다. IDS 로그를 나타내기 위해 STIX의 SDO인 indicator 객체, sighting 객체, identity 객체, observed data 객체를 사용한다. 기존의 STIX 표준에서 각각의 객체는 다음과 같이 이용되었다.

- indicator : CTI를 탐지하는데 사용한 규칙을 나타내는 SDO 객체
- identity : 사람, 조직, 그룹 등 주체를 나타내는 SDO 객체
- observed data : 시스템 및 네트워크에서 관찰된 데이터를 나타내는 SDO 객체
- sighting : 관찰된 정보 중 CTI의 요소를 가지고 있다고 판단되는 데이터를 분류하기 위한 SRO 객체

IDS 로그를 표현하기 위해 IDS 로그 탐지결과를 sighting 객체를 통해 나타내고, IDS 시스템을 identity 객체로, IDS 탐지규칙은 indicator 객체로, 탐지된 로그는 Observed data 객체로 표현하여 sighting 객체의 특정 속성을 통해 IDS 시스템, IDS 탐지규칙, 탐지된 로그가 참조



[그림 1] IDS 로그 다이어그램

<표 1> sighting 객체 세부 속성

STIX 객체	속성	비고
sighting	type	객체 유형(sighting)
	id	sighting 객체의 ID
	created_by_ref	sighting 객체를 생성한 주체의 ID
	created	sighting 객체의 최초 생성 시각
	modified	sighting 객체의 최근 수정 시각
	revoked	sighting 객체의 폐지 여부
	first_seen	탐지규칙에 일치하는 객체가 관찰된 최초 시각
	last_seen	탐지규칙에 일치하는 객체가 관찰된 최근 시각
	count	탐지규칙에 일치하는 객체가 관찰된 횟수
	sighting_of_ref	탐지규칙(indicator)의 ID
	observed_data_refs	탐지대상(observed_data)의 ID 목록
	where_sighted_refs	탐지주체(identity)의 ID 목록

될 수 있도록 한다. 각 객체의 속성은 사용자가 새롭게 정의할 수 있고, 변경할 수 있다. 제시하고자하는 sighting 객체의 세부 속성은 <표 1>과 같다.

sighting_of_ref, observed_data_refs, where_sighted_refs를 제외한 나머지 속성들은 STIX 표준으로 공개된 기본 속성이다. 속성 중 sightng_of_ref는 IDS 탐지규칙을 포함하고 있는 indicator 객체를 참조하며, observed_data_refs는 해당 탐지규칙에 의해 탐지된 대상을 포함하고 있는 observed_data 객체를 참조하고, where_sighted_refs는 해당 IDS의 시스템 정보를 나타내는 identity 객체를 참조한다. 각 객체들을 참조하는 이유는 객체들 간의 관계를 표현하여 공유된 정보에 대한 신속하고 정확한 해석을 하기 위함이다.

IV. 결론

본 논문에서는 점점 심각해지는 사이버 위협에 대응하기 위해 주요 정보시설 간 정보공유 체계 구축을 통해 사이버 위협에 신속히 대응해야 한다는 필요성을 인지하고, 정보공유체계에서 현재 미국에서 표준화가 진행 중인 STIX v2.0을 정보공유 시 정보표현 규격으로 제시한다. 표준화된 정보 표현은 국가 관제체계에서 사이버 위협에 대한 정보를 전파할 시 호환성, 가독성, 일관성, 정확성 등에서 이점을 보일 것

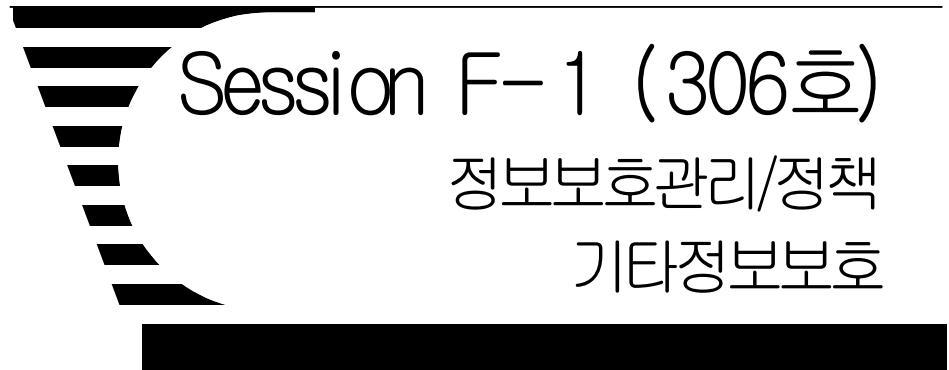
으로 기대한다.

[ACKNOWLEDGMENT]

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학 ICT연구센터지원사업의 연구결과로 수행되었음(IIITP-2018-2018-0-01423).

[참고문헌]

- [1] 국외 사이버 위협 정보공유의 체계조사, <http://www.kisa.or.kr/uploadfile/201402/201402141548019564.pdf>
- [2] STIX Version 1.2.1 Part 1: Overview, <http://docs.oasis-open.org/cti/stix/v1.2.1/cs01/part1-overview/stix-v1.2.1-cs01-part1-overview.pdf>
- [3] OASIS CTI TC 표준화 규격, <https://www.oasis-open.org/standards>
- [4] STIX 2.0 Specification Part 1: STIX Core Concepts, <https://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part1-stix-core.pdf>



좌장 : 허준호 (부산가톨릭대)

An Improved Big Data Analysis Technique for the Security of Nuclear Power Plant

Sangdo Lee^{1,*}, Jun-Ho Huh^{2,**}

¹Senior Manager of Security & ICT Department, Cyber Security Control Team, Korea Hydro & Nuclear Power (KHNP) Co. LTD, Republic of Korea (ROK)

*First Author's email: piterlee@gmail.com

²Assistant Professor of Department of Software, Catholic University of Pusan, Republic of Korea.

** Corresponding Author's email: 72networks@pukyong.ac.kr or 72networks@cup.ac.kr; Tel.: +82-510-0662

Abstract

Recently, there has been speculation in the ROK on the permanent shutdown of Shin-Gori Nuclear Power Units 5 and 6. The Korean Government organized a speculation committee with the participation of citizens and announced the final recommendation. The result was to recommend the restart of the construction of nuclear power. The background of the initial speculation was to reflect the people's concern on the safety of nuclear power. In line with the severity of radiation leaks of nuclear power plants, the decision was made with the de-nuclear awareness. Additional opinion in the speculation survey demonstrates an increasing need for the supplementation of safety. As such, the safety of nuclear power plants is a serious national concern. The Republic of Korea suffered direct and indirect damage from the Fukushima nuclear accident in Japan and also experienced cyber threats toward Republic of Korea Hydro & Nuclear Power Co., Ltd. in December 2014. With the above matters in mind, this study aims to suggest a measure for improving security in nuclear power plants. Based on overseas cyber attacking cases and an attacking scenario on the control facility of a nuclear power plant, the study designed and proposed a nuclear power plant control network traffic analysis system that satisfies security requirements and in-depth defense strategy.

I. Introduction

Among cases of information system hacking over the last couple of years around the world, a mobile service-based cyber attack that would cause national risk would be cyber terrorism on the nuclear power plant, an infrastructure facility. Two important cases are an attack on Stuxnet, an Iranian nuclear facility, and the cyber threat

on ROK Hydro & Nuclear Power Co., Ltd. The former showed that nuclear power could be shut down by direct cyber attack and the latter demonstrated that a cyber hacking attack on a nuclear power plant could cause psychological damage for the people in the country. After these events, security has been reinforced as the risk of cyber attack against the nuclear power plant was found [1-4].

The estimated damage by cyber attack on national infrastructure facilities is so large that it may affect the entire nation and society. In particular, in case of a nuclear power plant, it can cause a huge amount of damage such as power outage and radiation leak, causing huge social chaos and continual damage. To reinforce security against such threats, it is necessary to analyze the weaknesses of the control system periodically and to strengthen the checks after the security measure. As a measure, the Cyber Security Plan (CSP), a measure of identifying the digital equipment of the power plant control network and of evaluating risk level, has been carried out [5-6].

However, there is a lack of measure to maintain the integrity of the control facility in terms of operating and technical measures after identifying digital equipment and checking weaknesses in the nuclear power plant control facilities. While digital equipment identification, weakness checks, and media control are already compulsory, it is necessary to develop real-time detection measures against the newest malicious codes that can be introduced into the control facility of power plants.

II. Big Data Analysis for Security

Figure. 1. is a visualization graph provided by TensorBoard, which is a visualization tool of TensorFlow. As TensorFlow teaches the neural network, it shows the process of optimization and graphs the indicators. Biase means the machine learning time and the horizontal line is the Biase value. This corresponds to b when $y=ax+b$ and can be divided into 2 classes if the value is adjusted. A and B in [Fig. 17] mean that 500 times of learning were taken with G-Descent \Rightarrow

Adam order and 500 times of execution were carried out. The top 2 lines and the 2 bottom lines indicate that the Biase value depends on which one is first studied. Figure B on the top row shows that although class classification is difficult to achieve despite the 500:500 learning curve, the bottom line D shows the accurate Biase value. As a result, it was found that it is advantageous to learn Adam first rather than to use G-Descent first in class classification.

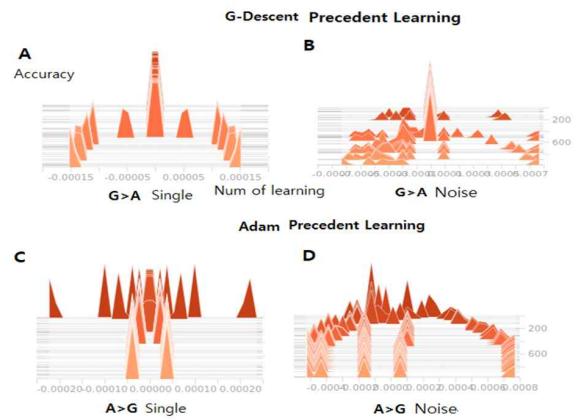


Figure. 1. Visualization Performance Evaluation Provided by TensorBoard

Figure. 2 is a graph comparing the accuracy of single learning and noise insertion learning. The vertical axis indicates accuracy and the horizontal axis indicates learning unit. The vertical scale indicates the accuracy of a single cell with 100 times of learning, the blurred line indicates accuracy when verifying with learning data, and the darker line indicates accuracy with validation data after learning with learning data. The reason why the diagonal line (dotted line) appears is that the noise is caused by the phenomenon in which the accuracy suddenly fell back up again while double learning was processed. As it is a result of the experiment, it is expressed without removal. The above

figure is a curved graph with winding in case of single learning (C, D) and the noise learning (A and B) shows a curved graph without winding which shows more stability. The currently experimented learning data are 2~3,000 cases, and the graph was not shown in detail. If it is experimented with at least 100,000 cases, it may produce a more detailed graph.

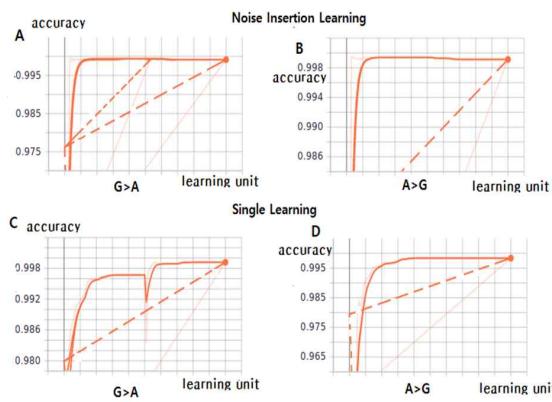


Figure 2. Graph Comparing the Accuracy of Single Learning and Noise Insertion Learning

III. Conclusion

This study designed and proposed a nuclear power plant control network traffic analysis system that satisfies security requirements and in-depth defense strategy on the basis of overseas cyber attack cases and control facility attacking scenarios against recent nuclear power plants.

To enhance security of the nuclear power plant, the study collected big data such as internet flow into the control facilities, network traffic of intranet, and security equipment events and compared and verified them with machine learning analysis. After measuring accuracy and time, the study proposed the most suitable analysis algorithm for the power plant by comparing and analysis. To find a suitable algorithm, the study compared and tested Adam, G-Descent

and CNN algorithms. In the 1st test result, in terms of the analysis study, Adam algorithm was the predominance and G-Descent had the highest accuracy. As CNN had a difference in speed more than double compared with other algorithms, it was excluded. The algorithm that is suitable for control network analysis requires not only accuracy but also analysis time. Real-time analysis of an attack against power plants needs to have quick analysis time, but accuracy also cannot be compromised.

Conflicts of Interest: The authors declare no conflict of interest.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2017R1C1B5077157).

[References]

- [1] S Park, J-H Huh,; "Effect of Cooperation on Manufacturing IT Project Development and Test Bed for Successful Industry 4.0 Project: Safety Management for Security," Processes, MDPI, Vol.6, No,7, 1-31, 2018.
- [2] Diederik P.K.; Jimmy L.B., "Adam: A Method for Stochastic Optimization". ICLR, 2015.
- [3] LeCun Y., Bengio Y., Hinton G.; "Deep learning," Nature, 521, pp. 436-444, 2015.
- [4] Sermanet P et al.; "Overfeat: Integrated recognition, localization and detection using convolutional networks," CoRR, 2013.
- [5] S Lee, J-H Huh, "An effective security measures for nuclear power plant using big data analysis approach," The Journal of Supercomputing, Springer, 1-28, 2018.
- [6] Vosoughi, S. et al., "The spread of true and false news online," Science, 359, 2018.

An Analysis of Satisfaction Level of ITSM for Improvement of IT Service Efficiency in Military

Han-Chul Woo^{1,*}, Jun-Ho Huh^{2,**}

¹Assistant Professor (Lieutenant Commander) of Defense Security Command, Republic of Korea.

¹Ph.D candidate of Dept., of Defense Acquisition Program, Kwangwoon University, Seoul, Republic of Korea.

*First Author's email: woocking@hanmail.net

²Assistant Professor of Department of Software, Catholic University of Pusan, Republic of Korea.

** Corresponding Author's email: 72networks@pukyong.ac.kr or 72networks@cup.ac.kr; Tel.: +82-510-0662

Abstract

Following the expansion of the IT services in the military acquisition sector such as Defense Electronic Procurement, military export/import support system, etc., the customers' dependence on IT for conducting their business with military or the related companies is increasing, as well as the military's dependence on the same technology for their services to the public. However, the issues pertaining to the simplified/integrated management of complex IT service management systems, including slow system recovery, lack of an integrated customer service window or insufficient information sharing have become the priority problems the IT managers are required to solve. A proper business process and IT service system should be established along with efficient IT process to maintain or improve the quality of service or develop a new business model. In that regard, ITSM (IT Service Management) is the key to the solution. The ITIL (IT Infrastructure Library)-based ITSM is one that supports efficient system management without complicating the service process by assuming the role of guiding how to strategically achieve the management objectives while consistently improving the quality of the service.

I. Introduction

Over the last few decades, the defense business has transformed into a more sophisticated form introducing some innovative technologies of the 21st century and the defense contractors are consistently approaching companies having some cutting-edge technologies regardless of their

sizes. The digitalized and automated technologies have become the integral sources of advanced modern weapon systems [1-3].

In most cases, the development and approval process for defense projects takes a long period of time so that those small/medium-sized weapon system development companies may not be able to keep their business until the actual

implementation of their new systems.

It would be wise for those who are innovative but small companies to partner up with an established defense contractor to avoid any unnecessary and costly processes which can be expected when dealing directly with the Ministry of Defense by themselves. This method allows them to be supported, guided, and funded when they are in need as long as the contract is valid while avoiding the typical bureaucracy they are not familiar with.

For an example, companies like techUK are often assisting small and medium-sized companies (SME's) interested in participating in defense systems by offering some foundational supports. The Chairman of techUK, Tim Gibson, explained that due to the strict regulations, it is quite difficult for the SME's to participate in military supply chains especially when they wish to export their product or technology overseas so that it would be a good idea to establish a partnership with some multinational corporations who are familiar with the military markets in NATO, North America, Japan, and/or Australia [4-6].

Recently, the concept of ITSM is being discussed as a solution to the issues pertaining to the IT service management in businesses. It has been found that the wider use of IT infrastructure makes a company depend on the quality, volume, and availability provided by the ICT infrastructure. That is, the IT infrastructures are designed focusing on the technology rather than the rational and efficient customer-oriented process due to the stereotype view of 'IT is the business' or 'Business is the IT'. The military agencies also increasingly paying attention to the ITSM following the trend and the Defense

Acquisition Program Administration (DAPA) of the Republic of Korea (ROK) is proactively adopting the concept by systemizing it to apply to their operations.

II. ITSM (IT Service Management)

Figure. 1 is describing the elements forming an ITSM system and the most important element is the IT process which is the process of providing and supporting the IT service. The next most important part involves manpower and organization. It is essential that the manpower with the skills and abilities necessary for providing an optimal IT service should be fostered/secured, establish an organization with them, and assign an appropriate role to each of them.

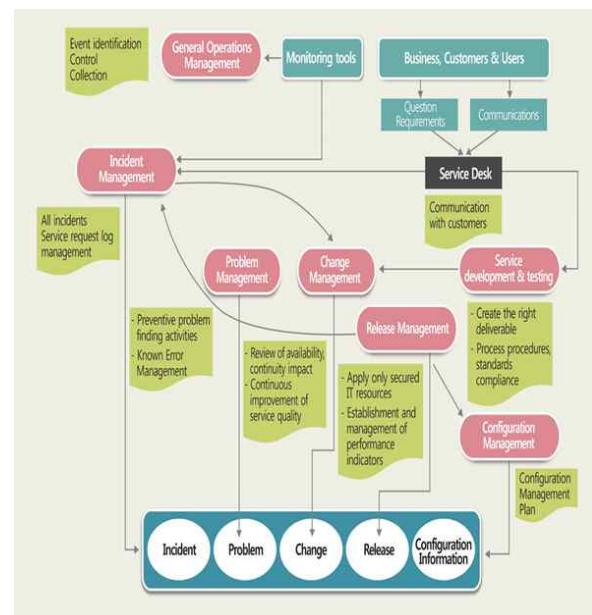


Figure. 1. The ITSM

The technology used here refers to a tool or solution which is required to provide an optimal IT service and without such an element, the effects of ITSM cannot be realized properly. Recently, many solution companies have released the solutions related to ITSM.

III. Analysis of Application of Military Acquisition IT Service ITSM

As they also handled customers' varying requests individually through their own systems and managed the data separately, it was impossible to check their operating statuses comprehensively so that the efficient management of operating personnel and utilization of valuable data were not achieved as they had first intended. The IT services prior to the introduction of ITSM are represented as 'the dispersed points for complex and diversified systems, operating organizations and supporting environments, and customer service requests'. Much changes have been made since the introduction of ITSM in 2009 and among them, the largest one was 'integrated IT service management' which allowed flexible use of manpower following the improved operational efficiency and reduced processing time based on a single point of contact (i.e., unifying all the systems into an IT service desk) for all the requests and information of accumulated data. Figure. 2 shows the ITSM System Process Procedure (Fault Processing) as a feature of the system service after the introduction of ITSM as an 'information of unified systems/data'. Analysis revealed that the greatest effect of ITSM was the 'completion of an integrated support infrastructure' based on increased management efficiency, improved service reliability, and expeditious business support. Meanwhile, the IT service desk described in Figure. 3 allows a quick response to the IT-related problem through user interface which shows the overall situation of requests for troubleshooting.

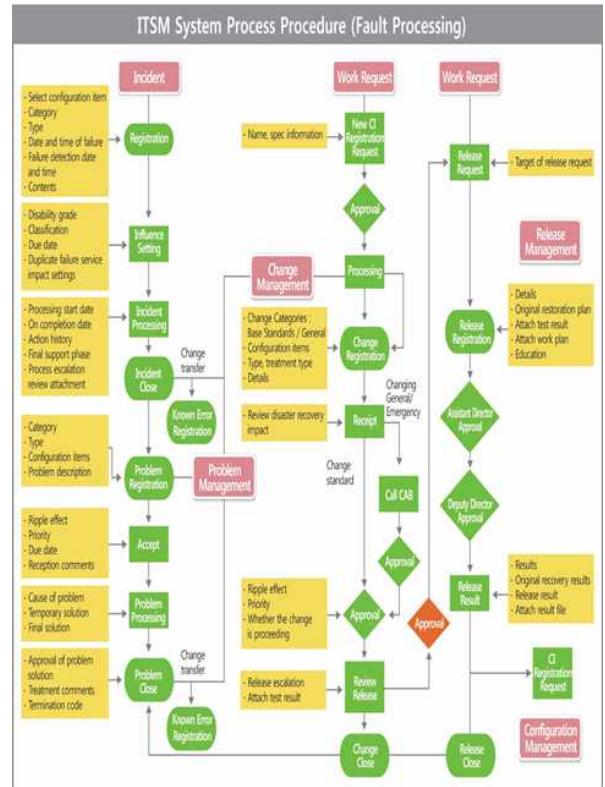


Figure. 2. The ITSM System Process Procedure



Figure. 3. The IT Service Desk of DAPA.

IV. Conclusion

In this paper, the effects of introduction and application of ITSM to increase the efficiency in the IT service for military acquisition sector have been analyzed. Along with military supplies and weapon systems procurement/acquisition, this sector is a vital national service and has a variety of special IT services supporting the business. After the

introduction of ITSM, the IT services dispersed throughout each acquisition sector have been integrated into a single IT service to solve the problems pertaining to the previous services offered individually. Most of all, the greatest achievement is that a framework which allows an effective IT service support has been created. However, it is still necessary to place ITSM in the center of the IT service in the military acquisition sector by developing the ITIL insufficient in the existing ITSM system to actively utilize it for achieving a higher level of customer satisfaction and organizational ROI (Return on Investment).

The key factor in the service quality to obtain the reliability of service is service value which is one of the components of a service quality directly affected by the human and the resulting qualities. At the same time, a decisive parameter for increasing the service reliability is service satisfaction which should be complemented or improved to secure a higher level of reliability.

Conflicts of Interest: The authors declare no conflict of interest.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2017R1C1B5077157).

[References]

- [1] Stanley, Jeffrey E., et al. "Correlating network services with operational mission impact." Military Communications Conference, MILCOM 2005, 2005, IEEE.
- [2] Dimou, I., et al. "Demonstration of a

Cross Security Domain Service Management Capability for Federated Missions." Military Communications Conference (MILCOM 2014), IEEE, 2014.

- [3] Pathmanand, Ukrist. "Globalization and democratic development in Thailand: the new path of the military, private sector, and civil society." Contemporary Southeast Asia (2001): 24-42.
- [4] Man, Mandy Mok Kim, and S. A. Wafa. "The relationship between distinctive capabilities, innovativeness, strategy types and the performance of small and medium-size enterprises (SMEs) of Malaysian manufacturing sector." The International Business & Economics Research Journal 8.11 (2009): 21-33.
- [5] Shyy, D. J. "Military usage scenario and IEEE 802.11 s mesh networking standard." Military Communications Conference, 2006. MILCOM 2006. IEEE, 2006.
- [6] Balci, Osman, and William F. Ormsby. "Network-centric military system architecture assessment methodology." International Journal of System of Systems Engineering 1.1-2 (2008): 271-292.

가스터빈 디지털트윈 : 가스터빈 동작 모델링을 위한 에이전트 기반 시뮬레이션

엘디 푸트리 라마와티 빈티*, 강효은*, 김호원*

*부산대학교 전기전자컴퓨터공학과

{putrirmwati, hyoeun405, howonkim}@gmail.com

Digital Twin for Gas Turbine : Agent-Based Simulation approaches for Modelling Gas Turbine Operation

Putri Rahmawati Binti Eldi*, Hyoeun Kang*, Howon Kim*

*Department of Electrical and Computer Engineering, Pusan National University.

Abstract

In this paper, an agent-based simulation modeling approach is presented. This simulation is used for communication between each component on gas turbine based on the Brayton cycle. The implementation involves a power generating process that first creates large datasets of dimensionless ambient coefficients and then applies those coefficients to a variety of turbine components. The results of this simulation are ambient temperature can control generated power based on the scenario and the other controlled attributes will be added in the future.

Keywords : Agent-Based Simulation, Gas Turbine, Brayton Cycle

I. Introduction

Driven by the Industry 4.0 vision, and the development of big data analytics, faster algorithms, increased computation power, and amount of available data enable the simulation with ability of real-time control and optimization of products and production lines, which is referred to as a Digital Twin, using a digital copy of the physical system to perform real-time optimization[5].

Real-time control needs simulation method, one of that is Agent-Based simulation, The concept of agent-based simulation is usually selected for modeling the human-oriented problems, such as found in the engineering design process[4].

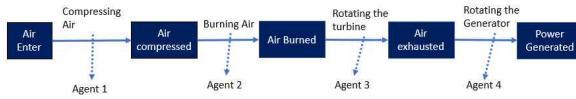
Our work will be discussed in the context of modeling gas turbine operation, How the ambient temperature affects the generated power as one of the attributes that will be controlled by Artificial Intelligence (AI) in the future. In the end, this simulation is for AI's environment.

II. Proposed Method

2.1 Modelling Paradigm

In the modeling paradigm, it separates into two, first is time-driven and second is event-driven. In Time-driven there is 2 part, system dynamics, and Dynamic System while Event-driven has two parts, Agent-Based and Discrete-Event[4] which are used in this research.

Discrete Event Simulation (DES) is used to analyze in every change of events while Agent-Based Simulation is used to model the different resources activities/states.

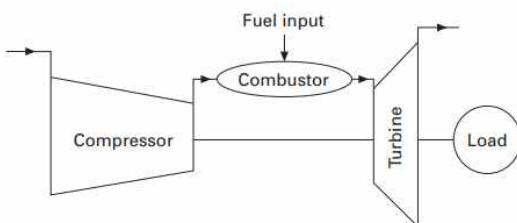


[Figure 148] Agents in Gas Turbine
There were many different resource types that have been considered including humans and major facilities[1].

There are two kind of ABS, first is An agent is a computational system that is situated in a dynamic environment and is capable of exhibiting autonomous and intelligent behavior, second is An agent may have an environment that includes other agents. The community of interacting agents as a whole operates as a multi-agent system[2].

2.2 Gas Turbine Operations

Gas Turbine is power generation that needs higher inlet pressure than the exit. normally, the compressor is used to provide an increase in pressure into the turbine. if the compressor discharge flow is expanded, the turbine power will less. Because of that, energy is added into compressor discharge air as combustor that burning fuel. Clearly, the power output from the gas turbine depends on the efficiency of the compressor, combustor, and turbine[3].



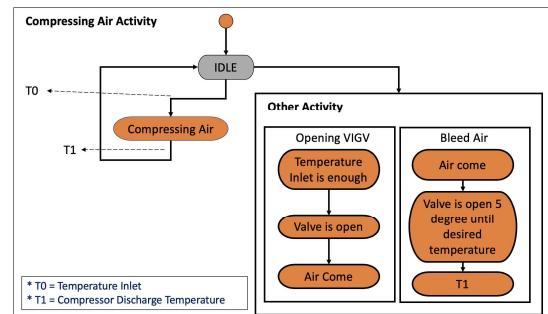
[Figure 2] Schematic layout of single-shaft gas turbine

III. Model Scenario

This scenario is just the example gas turbine operation (not exactly same). The goal for this scenario is getting the probability of power generated by a given inlet temperature/ ambient temperature range. There are 4 agents in this model, compressor, combustor, turbine, and generator. Each agent has their own activity that can affect the other agent.

3.1. Compressor Agent scenario

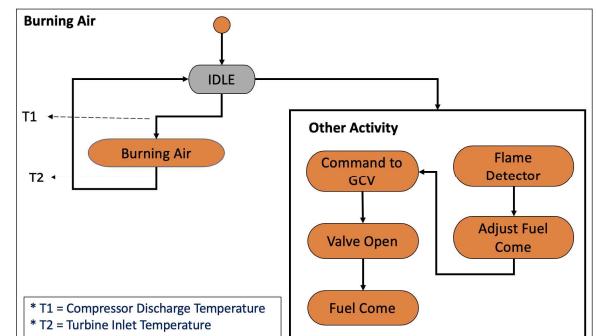
The main activity in the compressor compresses the air, but there are another two activities in the compressor, those are opening VIGV and bleed valve (see Figure 3).



[Figure 3] Compressor scenario

3.2. Combustion Chamber scenario

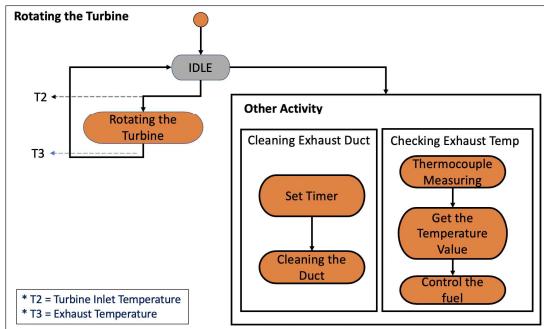
Combustion chamber or combustor have to burn the fuel for increasing the energy, and the other activity is control the fuel flow entering the chamber(see figure 4).



[Figure 4] Combustor scenario

3.3. Turbine scenario

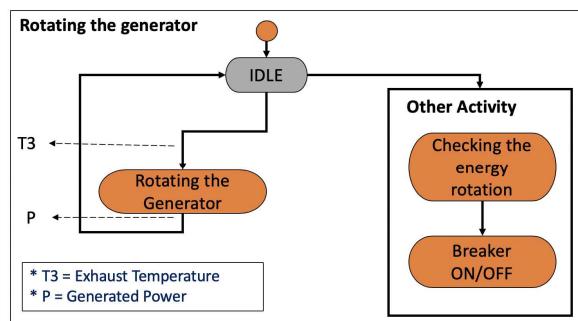
In the combustion chamber, before hot air from combustion chamber enters the duct, need to clean the duct because if there is another hot air there, it will become fault in there. And after the turbine spin, need to check the exhaust temperature to send the feedback control to the combustion chamber (see Figure 5).



[Figure 5] Turbine scenario

3.2.4 Generator scenario

In the generator, its about electricity. If the energy rotation from the turbine is so high, it can cause the short in the gas turbine. Because of that, we need a breaker to control the generator (see Figure 6).

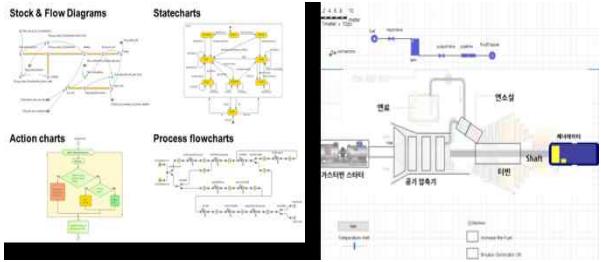


[Figure 6] Generator scenario

IV. Simulation Results

The simulation is using Anylogic. In the main display, the agent is represented by the rectangle and a slider for controlling the ambient temperature. Besides that, there is also information about the duct and generator breaker.

In Figure 7, it shows the running



[Figure 7] Simple Gas Turbine Simulation simulation with generated power 1.04 MWatt and there is information for increasing the fuel flow because the desired power cannot be achieved.

V. Conclusions

In this gas turbine simulation using Anylogic, ABS method can be approached. Also, the communication between the agent can clearly see in the simulation. If we want to control the ambient temperature, then we could predict the power generated by this simulation.

[References]

- [1] Abdelghany, M., Eltawil, A. B., & Abdou, S. F. (2016). A Discrete-Event and Agent-Based Hybrid Simulation Approach for Healthcare Systems Modeling and Analysis. International Conference on Industrial Engineering and Operations Management (pp. 1921–1928). Kuala Lumpur: IEOM Society International.a
- [2] Monostori, L., VanCzca, J., & Kumara, S. (2006). Agent-Based Systems for Manufacturing. Annals of the CIRP.
- [3] Razak, A. (2007). Industrial Gas Turbines. Boca Raton: CRC Press LLC.
- [4] Yu, T.-T., aj. P., Crowder, R. M., & Wills, G. B. (2012). Approaches to modelling the gas turbine maintenance process. Journal of Computing and Information Science in Engineering.
- [5] Zhang, H., Liu, Q., Chen, X., Zhang, D., & Leng, J. (2017). A Digital Twin-Based Approach for Designing and Multi-Objective Optimization of Hollow Glass Production Line. IEEE Access, 26901–26911.

성공적인 전산 망 분리를 위한 실무적인 고려사항

박종현*, 김창훈*

*대구대학교 컴퓨터공학

johpark74@naver.com

Practical considerations for a successful network separation

Jong-hyun Park*, Chang Hoon Kim*

*Department of Computer Engineering, Daegu University.

요약

업무 전산망분리는 금융·공공기관을 시작으로 내부에서 운영 중인 중요 정보자산을 외부의 사이버 위협으로부터 보다 완벽하게 보호하기 위하여 도입·운영 중인 네트워크 보안의 한 분야이다. 이는 전산망을 운영하는 모든 기업·학교·관공서 등 대부분의 기관에 적용할 수 있으나, 구축 시 소요되는 많은 비용과 시간 그리고 구축 실패에 따른 손실 등을 고려할 때 설계 단계에서부터 운영단계까지 전 과정에 대한 심도 있고 다양한 분석을 요구한다. 본 논문에서는 업무 망 분리 구축에 필요한 구성 방법 및 각 구성 유형별 장단점을 소개하고 망 분리 구현을 위한 여러 가지 고려사항들을 실무적인 관점에서 분석하고 그 결과를 제시한다.

I. 서론

우리나라의 망 분리는 2006년 “해외발 국가 기관 해킹 실태 및 대처방안” 일환으로 국가기관 업무전산망과 인터넷 분리 방침이라는 대통령 보고 자료에서 처음 언급되기 시작했으며 2007년 국가/공공기관 업무전산망 분리 실무매뉴얼을 제작하여 방통위, 기재부 등에서 시범적으로 망 분리 사업을 추진하였다[1]. 2008년부터 1, 2차 국가기관 망 분리 사업을 진행하였고, 2010년에는 지자체 및 산하기관으로 확대되어 2019년 현재에는 대부분의 기관에서 완성단계에 이르고 있다. 업무 망 분리는 비용적인 측면에서의 상당한 단점에도 불구하고 기관 내부의 전산 자산보호에 많은 장점이 있기 때문에 민간 기업에서도 추진중에 있다. 국가 주도의 망 분리 사업은 막대한 비용과 시간이 투자되는 대규모 기관 프로젝트로 CEO(Chief Executive Officer)의 강력한 리더쉽이 발휘되지 않는다면 자칫 불필요한 예산 낭비를 초래하는 결과를 가져올 수 있는 위험성은 항상 존재한다.

이런 CEO의 강한 의지를 배경으로 조직 전체가 일사분란하게 “보안”이란 원칙에 입각하여

모든 업무 프로세스를 정렬시켜 나가도록 함으로써 성공적인 망 분리의 완성단계에 한발 더 다가설 수 있는 것이다. 망 분리 구축은 한 번의 기업 성과를 위한 이벤트로 끝나는 보여주기식의 과업이 아닌 기업 전체에 영향을 미치며 향후 수십년을 내다보며 준비를 해야 하는 기업의 생존과 직결된 중요한 전략임을 인식해야 한다.

성공적인 망 분리 구축을 위해서 기업은 보안에 대한 명확한 목표를 정의한 후 실행에 옮겨야 한다. 본 논문에서는 성공적인 망 분리 구축을 위한 설계단계에서 요구되는 구성 유형과 실행단계에서 필요한 다양한 고려요소들을 분석하고 각각의 장점 및 단점에 대해서 기술하고 기본적인 해결책에 대해서 제시한다.

II. 망분리 유형 및 장단점

망 분리를 구현하는 방법 중 대표적인 방식은 “물리적 망 분리”이다. 즉, 물리적으로 내·외부 통신망을 분리 운영함으로써 외부망으로부터의 사이버 공격과 내부의 중요한 자료의 유출을 원천적으로 차단하는 것이 기본 목표이다.

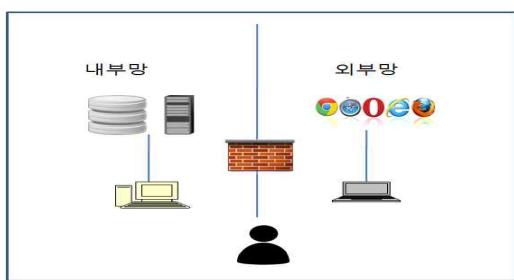


그림 6 물리적 망 분리 개념도

물리적 망 분리는 하나의 책상에 내·외부망 연결을 위한 데스크톱 컴퓨터 2대를 사용하게 되며, 각 PC는 고유의 업무를 수행함으로써 서로간의 간섭이나 위험요소로부터 독립적으로 동작하게 된다. 하지만 책상 위의 여유 공간이 넉넉지 않다면 컴퓨터를 구성하는 본체, 마우스, 키보드, 모니터 각 2세트의 공간이 사용자들의 업무 불편을 초래하는 요소로 영향을 미칠 수도 있을 것이다. 이런 불편함을 개선하기 위하여 PC의 주변기기(모니터, 키보드, 마우스 등)를 공유해서 사용하고 본체만 별도로 사용하기 위한 KVM(Keyboard, Video Monitor, Mouse) 스위치의 사용도 고려해 볼 수 있을 것이다. 하거나의 본체에 내·외부용 하드디스크와 네트워크 장비를 스위칭하기 위한 별도의 전환스위치를 사용할 수도 있을 것이다.

또한 이런 물리적 형태의 망 분리는 사용상의 불편함과 함께 고려해야 되어야 할 요소가 비용적인 측면에서 다른 방식에 비해 소요 비용이 크다는 점이다. 모든 PC는 2대씩 제공되어야 하며, 모니터와 같은 주변기기를 공유한다 해도 별도의 스위칭 장치를 준비해야 할 것이다.

이런 단점을 극복하기 위하여 “논리적인 망 분리”를 고려해 볼 수 있다. 개념은 물리적인 로컬 PC와 해당 물리적 PC를 경유하여 서버기반의 가상 PC에 접속함으로써 사용자PC를 논리적으로 서로 독립성을 유지하며 운영할 수 있는 망 분리 방식이다.

논리적 망 분리는 1대의 PC를 이용하여 내·외부망에 모두 연결이 가능하다. 물리적 PC를 이용하여 내부망에 접속하고 가상 PC를 연결하여 인터넷을 사용함으로써 외부에서 발생하는

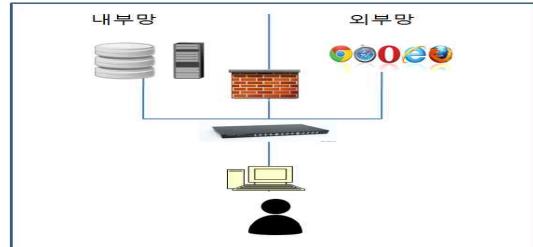


그림 7 논리적 망분리 개념도

바이러스·웜 등의 피해를 내부망과 독립적으로 동작시킬 수 있다는 것이 기본 개념이다. 논리적 망 분리의 대표적인 구현 방법으로는 SBC(Server-Based Computing) 방식과 VDI(Virtual Desktop Infrastructure) 방식이 있다. 두 방식 모두 사용자 컴퓨팅 환경은 로컬 컴퓨터 하드디스크가 아닌 서버 기반의 공간에 가상적으로 할당하여 운영하고 있으나 사용에 있어 SBC방식은 인터넷 컴퓨터는 기존 인터넷 용도로 사용을 하고 업무망은 SBC 전용 프로그램을 통하여 활용하게 된다. 즉 필요한 몇 개의 프로그램(ex,웹, CS 프로그램)을 사용자 환경에 배포함으로써 불필요한 프로그램의 사용을 통제할 수 있는 장점은 있으나, 배포 시 가상화 환경의 제약에 따라 일부 오류가 발생하기도 한다. 그리고 VDI 방식은 원격터미널을 이용하여 윈도우환경의 시스템이나 컴퓨터에 원격 접속하는 방식으로 별도의 Thin client 형태의 소형 컴퓨팅 단말기를 이용하여 가상머신(Virtual Machine)에 접속하여 컴퓨팅 자원을 사용한다. 기업 내 네트워크 연결이 가능한 모든 단말기에서 가상머신으로 접속을 하면 사용자의 동일한 컴퓨팅 환경 접속이 가능하여 업무의 연속성이 보장될 수 있으며, 이를 응용한다면 스마트 워크센터나 클라우드 컴퓨팅 환경 구현이 가능할 것이다. 하지만 수 천명의 가상 머신을 운영하는 기업에서는 직원들의 동일한 출근시간과 동 시간대 모든 PC의 부팅으로 인해 속도가 저하되는 부트스톰(Boot storm)을 피하기는 매우 어려운 실정이다. 또한 가상화 환경에서의 운영 프로그램의 경우 일부 프로그램 개발사에서는 일반 운영체제와 다른 라이선스 정책으로 별도 프로그램 비용이 추가될 수 있는 경우도 있다.

또 하나의 가상화 방식으로 PC 가상화를 이용한 네트워크 분리가 있다. 이 방식은 논리적 망 분리 개념을 응용한 것으로 사용자 PC의 운영체제를 업무용과 인터넷용으로 분리하여 구성하고 각 영역별 네트워크 접속을 분리하여 다른 네트워크에 연결하도록 하는 방식이다. 일반적으로는 업무 망 접속 시 실제 운영체제를 활용하고 인터넷은 가상화 운영체제로 보안 게이트웨이를 이용해 사용함으로써 2대의 PC를 이용하는 효과를 얻을 수 있다.

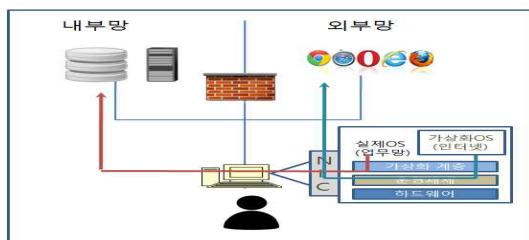


그림 8 PC 가상화를 이용한 망 분리

마지막 망 분리 방식은 하이브리드(hybrid) 형태의 구성으로써 사용자 환경에 따라 위에서 언급한 여러 가지 방식을 혼합하여 사용하는 것이다. 사용자 PC는 물리적인 망 분리를 적용하고 교육장이나 고객용 PC는 논리적 망분리, 외부 용역직원용 PC는 VDI방식을 적용하며, 외부 출장자를 위한 컴퓨팅 환경은 SBC방식으로 적용하는 것이다. 이런 하이브리드 방식을 적용하기 위해서는 각 구현방식의 장단점과 기업의 업무방식을 적절히 고려하여 선정함으로써 최대의 시너지를 창출할 수 있는 기회를 제공한다.

III. 망 분리에 따른 고려사항

1.1 망 연계를 통한 연계방안

망간 자료를 교환하는 유형에는 크게 서버 간 통신과 사용자 PC간 자료 교환, PC와 서버 간 통신으로 나누어 생각해볼 수 있다. 첫 번째의 경우는 서버 망 분리에 따른 웹서버와 웹애플리케이션 서버 간 통신이다. 외부에 공개된 홈페이지와 같은 시스템을 인터넷망(DMZ망)에서 웹서버, 웹 애플리케이션서버, 데이터베이스 서버로 운영하는 경우는 별도 내부망과 연결접점이 없으나, 일반적으로는 내부망에 통합 데이터

베이스를 구성하여 운영을 하는 경우에 반드시 서버 간 망 연계에 대한 부문을 고려해야 한다. 망 연계장비의 운영은 방화벽 장비와 유사하게 IP와 서비스 포트 기반의 허용 정책이 정의되어야 함으로서 서버 간 IP와 서비스 포트를 사전에 인지하고 있어야 한다.

두 번째 경우로 사용자 PC(내부망 PC와 인터넷망 PC)간 자료를 교환할 경우이다. 이 경우는 내부에서 작성한 문서를 메일 등을 통해 외부에 전달해야 하거나 외부에서 수집된 자료를 업무에 활용하기 위해서 내부망으로 전달해야 하는 경우로 예를 들 수 있다. 사용자 PC간 자료 교환은 망간 자료 전송 시스템을 활용하거나 보안USB 매체를 활용하게 되며, 자료 교환 시 반드시 백신 등을 이용하여 악성코드 점검 후 자료를 활용해야 한다. 불가피하게 일반USB를 이용하여 자료 교환이 필요한 경우는 관리자의 사전 승인을 득한 후 그 빈도를 최소화하여 사용할 수 있도록 한다.

망간 자료 전송 시스템을 활용하는 경우 공유 스토리지를 이용한 자료 전송 방식, 자동전환 스위치를 이용한 자료 전송 방식, 중계시스템을 이용한 자료전송, 일방향 전송장치를 이용한 자료전송 등의 여러 가지 방식이 있으며 기업의 업무절차 등을 고려하여 선택 운영하면 된다. 또 한 가지 실무적 입장에서 알아두어야 할 점은 인터넷PC에서는 자료 유출 등을 예방하기 위하여 문서편집 프로그램(한글, 워드, 오피스 등) 운영을 제한하여야 하며 외부로부터 유입된 파일이나 외부사이트 접속을 통한 악성코드 감염으로부터 사용자PC를 보호하기 위하여 컴퓨터를 재시작할 경우 초기화 될 수 있도록 조치하여야 한다.

마지막의 경우는 내부용 PC가 외부망에 연결이 필요한 경우이다. 이런 구성은 엄밀하게 말한다면 망 분리에 대한 위반 사항이며 가급적 해당 구성이 발생하지 않도록 하는 것이 최선이다. 하지만 실무적인 관점에서 본다면 충분히 발생할 수 있는 경우이며 이에 대한 대응방안을 사전에 고민해야 한다. 예를 듣다면 규모가 크지 않은 기업에서 내부 업무시스템에 지도서비스를 연계하여 구축한다고 가정 했을 경우

해당 지도연계 서비스 구축을 위한 막대한 비용을 감당하지 못해 해당 업무서비스를 구현하지 못하게 되나 외부에 오픈된 지도 서비스를 망 연계하여 이용 구축 한다면 비용 대비 효과 즉, 가성비 측면에서 선택의 여지가 없는 것이다. 하지만 이런 괴할 수 없는 망 연계 접점이라고 하더라도 연결 대상 시스템에 대한 사전 검증 확인 후 극히 제한적으로 허용되어야 하며 가급적 일방향(내부에서 외부)으로 연결해야 한다. 이것은 접속 대상 시스템을 경유한 악성 코드(웹쉘, 랜섬웨어 등)가 내부시스템으로 유입되는 경로를 차단하기 위해서이다. 또한 논리적 망 분리 구성 시 고려해야 할 요소 중 내부 망 PC의 자료를 외부망 PC로의 복사 및 붙여 넣기에 대한 기능 제한과 화면캡쳐를 통한 자료 연계 부분도 반드시 사용을 차단해야 할 기능적인 요소 중 하나이다.

1.2 주변기기에 대한 고려사항

망 분리는 많은 예산과 노력이 필수적이며, 기업 입장에서는 뚜렷한 성과도 나타나지 않는 모험적인 시도일 수 있다. 그래서 담당자로써 자원중복을 최소화함으로써 예산을 절약하는 방안을 모색하게 된다. 하지만 일반적인 망 분리 구축에 있어 구축비용을 최소화 할 수 있는 방안을 쉽게 찾을 수는 없다. 그래서 망 연계 시스템 등 반드시 있어야 하는 필수목록에 대한 최소화 보다는 복합기, 키보드, 마우스 등 주변기기에 대한 망 간 공통 활용을 통한 투입 예산의 절약 방안을 고려해 볼 수 있다. 망 간 다른 컴퓨팅 자원으로써 모니터와 마우스, 키보드를 공통으로 사용하도록 스위치 장치를 통해 구축하고 해당 스위칭을 통해서 내부 및 외부 망간 컴퓨팅 자원을 교차적으로 활용하게 할 수 있다. 그리고 복합기의 경우 내부 및 외부망에 동시 접속 가능한 네트워크 접점을 가지게 된다면 이 또한 망 간 연계의 경로가 될 수 있으므로, 이 경우 전문화된 망 간 연계시스템 구축을 통해 보안상 취약할 수 밖에 없는 경로를 제한해야 한다. 그래서 하나의 복합기를 여러 사람이 동시에 사용하게 하기 위해서는 내부망

의 PC는 네트워크를 통해 연결하고 외부망의 경우 LPT 프린트 포트를 이용하여 프린트서버로 공유하여 복합기를 사용한다.

물론 가장 이상적인 주변기기에 대한 방안은 망별로 분리된 자원을 사용하는 것이겠지만 예산이 충분하지 않은 중소 규모기업에서는 최대한 자원을 중복 활용하게 함으로써 비용에 대한 위험을 조금이나마 줄여 나갈 수 있을 것이다.

IV. 결론

망 분리는 외부의 모든 사이버위협으로 원천적으로 기업의 중요자산을 보호할 수 있는 대안으로 대두되었으나 기업 특성상 망간의 자료를 데이터를 교환함으로써 여러 가지 보안적인 위협요소가 파생되었다. 이는 발전소와 같은 완전 폐쇄적인 기반시설의 SCADA망과 같은 특수한 상황이 아니면 어느 기업이나 망 연계에 대한 접점은 반드시 발생할 것이며, 이 접점에 대한 관리가 망 분리의 성공과 실패를 좌우하는 기준이 될 것이다. 그리고 망 분리가 기업에서 성공적으로 자리 잡기 위해서는 철저한 사전준비를 통해 구성원들의 이해와 적극적인 참여가 반드시 필요하며 투입예산 대비 보안의 강화에 따른 기업의 대외 신인도 향상, 보안사고 시 발생되는 여러 가지 부대비용에 대한 절약효과 등과 비교를 통해 기업이 반드시 달성해야 할 목표임을 전 구성원이 이해하고 동참하도록 환경을 조성해야 한다.

[참고문헌]

- [1] <http://www.owasp.org> OWASP top 10 및 국가정보원 8대 취약점
- [2] 안랩연구소, 이용진, 망 분리의 필요성 및 방식
- [3] 국가정보원, "국가·공공기관 업무전산망 분리 및 자료전송 보안 가이드 라인"



좌장 : 박영호 (세종사이버대)

초경량 웹 어셈블리를 활용한 블록암호 CHAM 최적화 구현

안규황* 권혁동* 김현준* 서화정*†

*한성대학교 정보시스템공학과

tigerk9212@gmail.com, hdgwon@naver.com, khj930704@gmail.com, hwajeong84@gmail.com

Implementation of ultra-light block cipher CHAM optimization using Web Assembly

Kyuhwang An* Hyeokdong Kwon* Hyunjun Kim* Hwajeong Seo*†

*Division of information system, Hansung University.

요약

웹 페이지를 구현하기 위해선 주로 자바스크립트를 사용한다. 자바스크립트는 high-level 언어로, 사용하기 편하다는 장점이 있지만, 그만큼 상대적으로 low-level 언어인 C/C++보다 구현 성능이 떨어진다. 이러한 자바스크립트의 단점을 보완하고자 웹 어셈블리라는 기술이 탄생하였다. 웹 어셈블리는 C/C++로 작성된 코드를 웹 어셈블리 파일인 이진 바이너리 파일로 변환한 후 하나의 웹 페이지로 변환하는 기술이다. 단순히 자바스크립트로 구현하는 것보다 low-level 언어를 이용하는 웹 어셈블리로 구현하는 것이 엄청난 성능향상을 보여주며 다양한 분야에 활용될 것을 기대하고 있다. 또한 웹 어셈블리는 기존의 언어들과 다르게 연산을 수행하는 별도의 선형 메모리에서 연산을 수행하며, 사전에 권한을 부여 받은 객체만 선형 메모리에 접근할 수 있다. 따라서 자바스크립트보다 보안성과 성능 면에서 웹 어셈블리가 뛰어나다. 이를 실제로 검증하기 위해 본 논문에서는 자바스크립트로 구현한 CHAM과 웹 어셈블리로 구현한 CHAM간의 성능 비교를 하였으며 실제로 자바스크립트로 구현한 CHAM보다 웹 어셈블리로 구현한 CHAM이 약 6.15배의 성능 향상을 볼 수 있었다.

I. 서론

자바스크립트는 웹 페이지 내에서 동적으로 객체를 활용하고 싶을 때 사용되는 객체 기반의 스크립트 프로그래밍 언어이다. 자바스크립트는 컴파일을 수행하는데 있어 JIT(Just-In-Time) 컴파일 방법을 사용한다. JIT 컴파일이란 프로그램을 실제 실행하는 시점에 인간의 언어로 작성된 자바스크립트 코드를 기계어로 번역하여 컴파일하는 기법이다.

자바스크립트가 최초로 탄생했을 때는 JIT 컴파일 기법을 사용하지 않아 엄청나게 느린 환경을 가지고 있었지만, 2008년에 들어 많은 웹 브라우저들이 JIT 컴파일러를 수용함에 따라 자바스크립트의 성능이 엄청나게 향상될 수 있었다.

자바스크립트는 과거에 프론트단(front-end)에서 주로 활용이 되었지만 근래에 들어

node.js와 같이 백단(back-end)을 커버해주는 서버 사이드 네트워크 프로그래밍에도 활용되며, 모바일 웹을 개발하는데 사용되고 있다.

모바일 웹의 경우 치명적인 단점을 가지고 있다. 바로 속도이다. 모바일 웹은 네이티브 웹 & 앱에 비해 속도가 많이 느려 연산이 많이 필요로 하는 작업을 할 경우 완벽히 수행하지 못하는 등 문제를 발생시켰다. 이를 해결하기 위해 마이크로소프트사에서 개발한 ActiveX가 배포되었지만, ActiveX를 이용할 경우 보안에 취약한 등 새로운 문제가 발생하였고 해당 기술을 적용할 수 있는 웹 브라우저에서만 호환이 가능하다는 치명적인 단점이 발생하였다.

웹 어셈블리(Web Assembly)는 앞에서 언급한 속도와 같은 성능에 대한 문제점을 해결해 줄 수 있는 대책으로 기계어로 그 즉시 컴파일시키기 때문에 JIT 컴파일 기법보다 훨씬 좋은

성능을 자랑한다.

웹 어셈블리는 자바스크립트로 작성하는 것 이 아닌 C/C++로 작성한 코드를 웹상에서 컴파일하는 저수준 바이트코드로 바이너리 형식을 통하여 실행되며 독립적인 메모리를 할당하여 사용한다. 웹 어셈블리를 활용한다면 기존에 웹 혹은 사물인터넷 플랫폼 상에서 단순히 자바스크립트로 사용하였던 암호 알고리즘들에 대해 보다 빠른 성능을 보일 것을 기대한다. 동작 구조는 fig. 1과 같다.

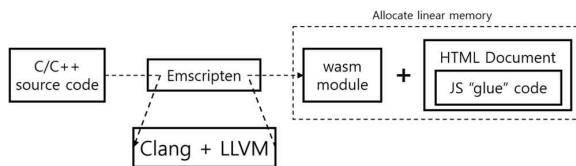


Fig. 11. How to run web assembly

본 논문의 구성은 다음과 같다. 2장에서는 웹 어셈블리에 대해 설명한다. 3장에서는 웹 어셈블리를 이용하여 구현한 CHAM에 대하여 성능 평가를 하겠으며, 4장에서 끝을 맷도록 하겠다.

II. 웹 어셈블리를 이용한 최적 구현

본 절에서는 CHAM을 구현하기 위해 참조 논문[1]에서 나타낸 바를 웹 어셈블리로 구현²⁾하였으며 아래 각주에서 확인 가능하다. 구현한 환경은 table. 1과 같다.

웹 어셈블리를 구현함에 있어 가장 중요한 것은 웹 어셈블리를 지원하는 웹 브라우저를 이용해야 하며, 해당 웹 브라우저가 웹 어셈블리 기능을 지원한다고 하더라도 그 중에 지원하지 않는 모듈도 존재한다. 따라서 본인이 원하는 모듈[2]이 무엇인지 확인한 이후에 지원하는 웹 브라우저를 선택해서 구동해야한다. 웹 어셈블리를 지원하는 브라우저별 최소 버전은 table. 2와 같다.

Table 1. The information of experiment environment

OS	macOS 10.12.6
IDE	Brackets 1.10
Web Browser	Chrome 69.0.3497.100

2)https://github.com/kyu-h/WebAssembly_CHAM

Table 2. The list of Web Assembly support web browser

Chrome	57
Edge	16
Firefox	52
Explorer	Not Support
Safari	11

웹 어셈블리는 기존 웹 구동 방식과 다르게 선형 메모리(Linear Memory)를 할당하고 해당 영역에서 동작한다. 기존 메모리 영역의 경우 관리자의 승인 없이도 데이터가 유동적으로 이동할 수 있다. 하지만 선형 메모리의 경우 기존 메모리에 있던 데이터를 관리자의 승인 하에 이동 후 선형 메모리에 저장된 데이터를 보다 안전하게 사용할 수 있다.

만약 웹 어셈블리를 이용하여 컴퓨터에 저장된 파일을 읽어서 사용하고자 할 때, 관리자가 선형 메모리에 파일을 할당하지 않는다면, 웹 어셈블리로 만든 html과 같은 폴더 내부에 있어도 해당 파일을 읽을 수 없다.

따라서 기존 메모리 방식보다 선형 메모리를 이용한 웹 어셈블리 방식이 데이터를 안전하게 보관할 수 있음을 fig. 2와 같이 보여준다.

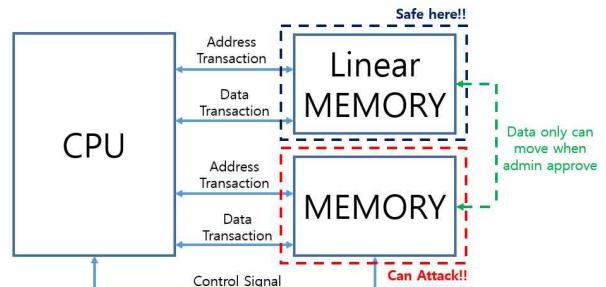


Fig. 12. How to allocate linear memory

III. 성능평가

CHAM에 대하여 3개의 모듈에 대해 1회 암호화 수행하는데 있어, 자바스크립트는 0.008초가 걸렸으며, 웹 어셈블리는 0.0013초가 걸려 약 6.15배 성능 향상을 볼 수 있다. 이는 table. 3과 같다.

앞에서도 설명했듯이 웹 어셈블리는 기존 컴파일 방식과 다르게 독자적인 선형 메모리상에

서 컴파일을 수행한다. 웹의 경우 프론트, 앤드 단을 자유롭게 상호작용하며 수행되어야하기 때문에 권한을 할당 받은 파일만 접근할 수 있는 특성을 갖고 있는 웹 어셈블리의 경우 1회 암호화 수행하는 것 보다 성능 저하를 예상하였다.

이를 검증하기 위해 테스트 벡터로 작성 된 메모장을 읽어 3개의 모듈에 대하여 각각 100회 총 300회 암호화 연산을 수행하였을 경우 얼마만큼의 연산 속도가 측정되는지 실험하였다.

각 모듈에 대해 100회 암호 연산을 수행할 때 자바스크립트는 0.51초, 웹 어셈블리는 0.0083초가 걸려 약 6.14배 성능 향상을 볼 수 있었으며, 각 모듈에 대해 10,000회 암호 연산을 수행할 때 자바스크립트는 0.259초, 웹 어셈블리는 0.0873초가 걸려 약 2.96배 성능 향상을 볼 수 있었다.

각 모듈에 대해 10,000회 암호화를 진행할 때는 앞에서 실험한 1, 100회보다 성능을 떨어졌지만, 그래도 단순히 자바스크립트로 구현한 것 보다 2.96배의 성능 향상을 하였으며, 2.96배도 엄청난 성과이다.

Table. 3. Performance comparison between javascript and web assembly implementing CHAM

Language	Time(s)	cpb
Time taken to perform		
1 encryption operation		
Javascript	0.008s	385,714
Web Assembly	0.0013s	62,678
Time taken to perform		
100 encryption operation		
Javascript	0.051s	2,458,928
Web Assembly	0.0083s	400,178
Time taken to perform		
10,000 encryption operation		
Javascript	0.259s	12,487,500
Web Assembly	0.0873s	4,209,107

IV. 결 론

본 논문에서는 초경량 블록암호 CHAM에 대하여 웹 어셈블리를 이용하여 최초로 구현하였다. 구현한 결과물과 코드에 대해 깃허브[3]에서 확인할 수 있다.

CHAM을 웹상에서 구현할 때 웹 어셈블리를 이용함으로써 단순히 자바스크립트로 구현한 것 보다 최대 6.15배 빠르게 구현하였다. 또한 기존의 자바스크립트는 단순히 CPU 메모리에 올려 암호화를 수행하기 때문에 보안성이 높지 않으나, 웹 어셈블리 같은 경우 선형 메모리상에 올려 암호화를 수행함으로써 권한을 부여받지 않은 객체는 접근이 불가하다. 따라서 자바스크립트보다 높은 보안성을 제공한다. 추후 연구로는 표준화된 국제 블록암호 혹은 해시함수에 웹 어셈블리를 적용하는 방안에 대해 확실해 볼 계획이다.

Acknowledgement

본 연구는 고려대 암호기술 특화연구센터(UD170109ED)를 통한 방위사업청과 국방과학연구소의 연구비 지원으로 수행되었습니다.

[참고문헌]

- [1] B. W. Koo, D. Roh, H. Kim, Y. Jung, D. G. Lee, D. Kwon, “CHAM: A Family of Lightweight Block Ciphers for Resource-Constrained Devices.” *International Conference on Information Security and Cryptology*. Springer, Cham, 2017.
- [2] MDN web docs, “Browser compatibility, a available: https://developer.mozilla.org/ko/docs/WebAssembly#Browser_compatibility
- [3] Github, “CHAM with Web Assembly,” a available: https://github.com/kyu-h/WebAssembly_CHAM

자동 익스플로잇 생성 도구에서의 버퍼 오버플로우 탐지 방법 분석

유지현*, 김주환*, 윤주범*

*세종대학교 정보보호학과

yjhy8783@naver.com, kjh97852003@naver.com, *jbyun@sejong.ac.kr

Buffer Overflow Detection Method Analysis in Automated Exploit Generation Tool

Jihyeon Yu*, Juhwan Kim*, Joobeom Yun*

*Department of Computer and Information Security, Sejong University.

요약

본 논문에서는 고도화되고 있는 소프트웨어 취약점 공격의 위험성과 현재 보안전문가들이 수동으로 대응하는 비효율적인 상황에 따른 자동 소프트웨어 취약점 분석의 필요성을 보여준다. 현재 자동 소프트웨어 취약점 분석 분야는 기초적인 단계로서, 미국 등 선진국에서도 활발히 연구 중인 분야이다. 본 논문에서는 자동 소프트웨어 취약점 탐지 방법의 이해를 돋고자 깃헙(github)에 공개되어 있는 자동 익스플로잇 생성 도구인 Zeratool, 그 중에서도 시스템 해킹에서의 대표적인 취약점인 버퍼 오버플로우를 어떻게 탐지하는지에 대해 분석하여 기술하였다.

I. 서론

최근 악의적인 해커들의 소프트웨어 취약점 공격이 점점 다양해지고 기술적이고 조직적인 체계로 변화하고 있다. 이를 방어하기 위해 취약점을 사전에 발견하고 분석하여 소프트웨어 보안성을 높이는 일이 매우 중요하다. 현재까지의 소프트웨어 취약점 발견은 대부분 보안전문가들이 수동으로 취약점을 분석하는 많은 비용과 인력이 필요한 일로 큰 규모의 소프트웨어 취약점 분석은 현실적으로 불가능에 가깝다. 이를 해결하기 위해 국내를 포함한 미국 등 선진국에서는 소프트웨어 자동 취약점 분석에 대한 연구가 진행 중이다.

본 논문에서는 깃헙(github)에 오픈소스로 공개되어 있는 소프트웨어 자동 취약점 분석/익스플로잇 도구인 Zeratool[1]에서 버퍼 오버플로우 취약점을 탐지하는 방법에 대해 기술하고자 한다.

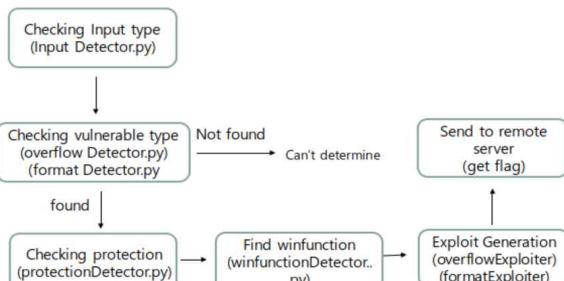
II. 본론

2.1 버퍼 오버플로우 (Buffer Overflow)[2]

버퍼 오버플로우는 시스템 해킹의 대표적인 취약점 중 하나이다. 컴퓨터에서 버퍼란 데이터를 일시적으로 저장해두는 메모리상의 공간을 의미하고 버퍼 오버플로우는 프로세스가 데이터를 버퍼에 저장할 때 원래 크기보다 초과되는 데이터를 입력하면 정상적인 경우에는 데이터가 변조되지 않아야 할 영역에 데이터가 덮어 씌워지는 것을 의미한다. 보통 개발된 프로그래밍 언어의 버퍼 오버플로우에 취약한 함수가 있는 경우 이를 이용해 시스템 권한 상승 코드나 악의적인 코드를 덮어 씌워 공격을 시도하며 공격이 성공할 경우 시스템의 권한을 상승시키거나 악의적인 행위를 할 수 있다.

2.2 자동 익스플로잇 생성 도구(Zeratool)

본 논문에서는 자동 소프트웨어 취약점 탐지



[그림 1] Zeratool 동작 순서

방법을 분석하기 위해깃협에 공개되어있는 자동 익스플로잇 생성 도구인 Zeratool을 사용하여 분석하였다. Zeratool은 CTF(Capture The Flag) 문제를 해결하는데 목적을 둔 파이썬 기반 소프트웨어 자동 분석/공격 도구이다. 현재 Zeratool이 탐지/분석할 수 있는 취약점은 버퍼 오버플로우와 포맷 스트링 취약점으로 바이너리 파일 및 소프트웨어를 분석하여 취약점 종류를 탐지하고 탐지된 취약점을 이용한 익스플로잇 코드를 생성하는 기능을 가지고 있다.

Zeratool의 전체적인 동작 순서는 [그림 1]과 같다. Zeratool은 가장 먼저 분석할 바이너리 파일에 입력 값을 주는 방법을 검사한다. 이에 따라 취약점 탐지, 익스플로잇 코드 생성에 차이가 있기 때문이다. 그 후 해당 바이너리 파일에 버퍼 오버플로우 또는 포맷 스트링 취약점이 있는지 검사한다. 다음으로 익스플로잇 코드 생성을 위해 바이너리 파일에 적용된 보호기법을 검사하고 winfunction을 탐지한다. 여기서 winfunction이란 바이너리 파일에서 취약점이 존재하는 function이다. 예를 들어, 취약한 함수 system("/bin/sh")이 main에 존재한다면 winfunction은 main이 된다. 다음으로 실제로 바이너리 파일의 취약점을 이용해 공격할 수 있는 익스플로잇 코드가 생성된다. 마지막으로, Zeratool은 CTF 문제풀이에 목적을 두었기 때문에, 원격서버에 익스플로잇 코드를 전송해 해당 서버의 쉘을 획득 후 Flag 값을 확인한다.

2.3 Zeratool OverflowDetector

Zeratool에서 버퍼 오버플로우를 탐지하는 부분은 Zeratool 내의 lib 폴더에 overflowDetector

```

if inputType == "STDIN":
    state = p.factory.full_init_state(args=argv)
elif inputType == "LIBPWNABLE":
    handle_connection = p.loader.main_object.get_symbol('handle_connection')
    state = p.factory.entry_state(addr=handle_connection.rebased_addr)
else:
    arg = claripy.BVS("arg1", 300 * 8)
    argv.append(arg)
    state = p.factory.full_init_state(args=argv)
    state.globals['arg'] = arg
    
```

[그림 2] input type에 따른 state 생성

파이썬 스크립트 파일로 구현되어 있다.

가장 먼저, 바이너리 분석을 위해 파이썬 바이너리 분석 프레임워크인 angr[3]에서 바이너리 파일 프로젝트를 실행한다. 그 후, [그림 2]와 같이 input type에 따라 state를 생성하는데, 여기서 state란 해당 바이너리 파일의 상태이다. state는 바이너리 파일의 실행 경로, 레지스터, 아키텍처, 현재 파일 실행 구간 등의 파일 정보를 담고 있다. angr는 바이너리 파일을 시뮬레이션으로서 실행하기 때문에 생성한 state를 기반으로 분석한다.

[그림 3]은 constraints를 이용하여 return을 변조하거나 winfunction에 도달할 수 있는 input 생성이 가능한지 확인하는 구간이다. simgr은 angr의 시뮬레이션 state이고 Simgr.unconstrained는 제약조건이 없는 구간을 말하는데 어떤 입력함수나 분기문이 나오지 않아 실행 경로 상 어떠한 제약이 없이 그대로 실행되는 구간을 말한다. 이 구간 이후 분기문이 있는 경우 그에 따른 실행 경로가 나뉘지고 경로별로 취약한 부분을 확인하기 위해 반복문으로 경로들이 실행된다. 실행 경로의 state를 저장한 후 pc 레지스터에 저장된 주소 값을 받아오는데 이는 분기문 또는 입력함수 이후 return이나 winfunction이 나올 가능성이 높다는 아이디어에서 비롯된다. 실제로 간단한 문제

```

for path in simgr.unconstrained:
    state = path.state
    eip = state.regs.pc
    bits = state.arch.bits
    state_copy = state.copy()

    #Constrain pc to 0x41414141 or 0x414141414141
    constraints = []
    for i in range(bits / 8):
        curr_byte = eip.get_byte(i)
        constraint = claripy.And(curr_byte == 0x41)
        constraints.append(constraint)
        print(constraints)

    #Check satisfiability
    if state_copy.se.satisfiable(extra_constraints=constraints):
        for constraint in constraints:
            state_copy.add_constraints(constraint)
    
```

[그림 3] constraints를 이용한 BOF 확인

일수록 분기문만 통과한다면 winfunction이 바로 실행되는 구조일 것이다. 그 후 저장한 pc 레지스터가 가리키는 주소를 1바이트씩 가져와 angr 내 모듈 claripy를 통해 constraint를 생성하게 된다. 여기서 constraint란, curr_byte(pc레지스터에 저장된 주소 1바이트)== 0x41(A) 가 되는 제약조건이다. angr에서는 바이너리 파일에 주는 입력 값(input)을 생성하는 solver engine이라는 모듈이 있다. 제약조건은 solver engine이 input을 생성해낼 때 제약조건에 맞는 input을 만들기 위해 사용한다. 그 후 se.satisfiable 메소드를 통해 생성한 제약조건에 맞는 input값을 생성해낼 수 있는지 확인 후 state에 제약조건을 추가한다. state에 추가한 제약조건을 바탕으로 angr가 시뮬레이션을 돌려 입력 값을 생성한다. 즉, 결론적으로 return 변조나 winfunction 도달할 수 있는 input을 생성할 수 있는지 검사하는 것이다. 제약조건에 따른 input이 생성되었다면, 그 input엔 제약조건 AAAA가 포함되어 있을 것이고 그 위치가 return이나 winfunction의 주소 위치가 될 것이다.

[그림 4]는 제약조건에 따른 input이 제대로 생성되었는지 확인하는 구간이다. stdin_str은 생성된 input값을 저장한다. 앞서 설명했듯이 버퍼 오버플로우 취약점이 존재한다고 판단하려면, 제약조건에 따라 input으로 ret나 winfunction이라고 판단되는 주소를 AAAA로 변조하는 input값을 생성할 수 있어야 한다. if문을 통해 생성된 input에 A가 있는지 확인함으로써 변조/도달 가능 여부를 확인한다. if문을 통과하지 못한다면 input으로 해당 주소를 A로 변조하지 못하기 때문에 버퍼 오버플로우 취약점이 아니라 판단한다. 이는 pc 레지스터가 담고 있는 주소가 winfunction이 아닌 정상적인 공유라이브러리 함수일 경우 버퍼와의 메모리 상 거리가 멀어 input으로 버퍼 오버플로우를 발생시킬 수 없기 때문에 input에 A가 들어있지 않을 것이라는 아이디어에서 비롯된 검사

```
stdin_str = str(state_copy.postx.dumps(0).replace('\x00',''))
.replace('\x01',''))
if 'A' in stdin_str:
```

[그림 4] if문을 통한 input 생성 확인



[그림 5] Zeratoor 실행결과 화면

방법이다.

III. 실행결과

[그림 5]는 버퍼 32byte + SFP 4byte = 36 byte 후 return 주소를 덮어씌워 공격하는 버퍼 오버플로우 취약점 바이너리파일을 Zeratoor이 탐지한 결과화면이다. 36 byte 이후 return 주소 부분이 제약조건에 따라 AAAA로 변조되는 input을 생성해낸 결과를 볼 수 있다.

IV. 결론

본 논문에서는 소프트웨어 자동 익스플로잇 도구 Zeratoor에서의 버퍼 오버플로우 탐지방법에 대해 분석하였다. Zeratoor은 버퍼 오버플로우의 기본 원리인 input을 통한 return 주소 변조를 이용하여 변조 가능여부를 통해 취약점을 판별한다. 하지만 가능성성이 높은 주소에 대해 검사하기 때문에 복잡하거나 다른 취약점과 결합된 구조의 버퍼 오버플로우 취약점에 대해 탐지해내지 못한다는 한계가 있다. 추후 많은 버퍼 오버플로우 취약점을 포용하기 위해 추가적인 아이디어 도입을 통해 정확성을 높이는 작업이 필요하다.

[ACKNOWLEDGEMENT]

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학 ICT연구센터지원사업의 연구결과로 수행되었음(IITP-2018-2018-0-01423).

[참고문헌]

- [1] Zeratoor, <https://github.com/ChrisTheCoolHut/Zeratoor>
- [2] 이구호, “버퍼 오버플로우 공격과 방어 기술에 대한 분류법,” 석사학위논문, 아주대학교, 2008년 8월.
- [3] Angr Documentation, <https://docs.angr.io/>
- [4] 강상용, 이권왕, 노봉남, “동적 기호실행을 이용한 윈도우 시스템을 Use-After-Free 취약점 자동 탐지 방법,” 정보보호학회논문지, 27(4), pp.803-810, 2017년 8월.

퍼저와 메모리 버그 탐지기를 사용한 바이너리 취약점 탐지 및 분류 방법

최민준, 김현욱, 윤주범

세종대학교 정보보호학과

choiminjun7077@gmail.com, hyunwook711@naver.com, jbyun@sejong.ac.kr

Binary Vulnerability Detection and Classification Method by Using Fuzz Testing with Memory Bug Detector

Minjun Choi, Hyunwook Kim, Joobeom Yun

Department of computer and information security, Sejong University.

요약

현재 바이너리의 취약점을 발견하기 위해 여러 연구가 진행되고 있다. 하지만 취약점의 종류를 파악하기 위해서는 일일이 별도의 분석이 필요하다. 따라서 본 논문에서는 퍼저를 사용하여 바이너리의 충돌(Crash)유발 값을 알아내고, 메모리 버그 탐지기를 사용하여 별도의 분석 없이 취약점의 종류를 빠르게 파악하기 위한 실험을 진행하였다. 실험 결과 AFL(America Fuzz lop) 퍼저의 화이트박스(White box) 기법만 사용했을 때 가장 많은 충돌이 발생했으며, 메모리 버그 탐지기 옵션을 사용한 바이너리에 AFL퍼저로 알아낸 충돌유발 값을 입력할 경우 다른 방식보다 빠르게 취약점의 종류를 파악하여 분류할 수 있었다.

I. 서론

대부분의 바이너리는 예기치 않은 입력이 있을 때 충돌(Crash)이 발생한다. 특히 C 및 C++ 와 같은 메모리를 보호하지 않는 언어의 인기 때문에 브라우저 및 OS 커널과 같은 대규모 프로젝트에서도 예기치 않은 충돌이 발생하고 있다.[1] 현재 충돌을 발견하기 위해 여러 연구가 진행되고 있다. 그 중 퍼저(Fuzzer)는 입력값을 변경하여 바이너리를 테스트하는 도구이다. 기호 실행 도구 등 다른 도구와 비교할 때 매우 단순한 방법으로 동작하지만, 충돌을 발견하는데 효과적인 도구이다.[2] 특히 자동화된 도구로 해킹 방어 대결을 펼치는 대회, Cyber Grand Challenge 에서 shellphish팀의 Mechaphish[3] 시스템도 AFL(americian fuzzy lop)[4] 퍼저가 포함된 시스템을 구축하여 3위를 차지하였다. AFL퍼저는 테스트할 바이너리의 소스코드가 있으면, 메모리 버그 탐지기인 ASan(AddressSanitizer)[5] 과 함께

사용해서 더욱 빠르게 취약점을 분류할 수 있다. 따라서 본 논문에서는 효율적으로 취약점 종류를 파악하고 분류하기 위해 AFL퍼저와 메모리 탐지기인 ASan에 대해 집중적으로 분석한다.

II. 관련연구

1.1 기호실행

바이너리 입력값에 대한 실행 가능한 모든 경로를 분석하는 기호실행 기법은 분기 문이나 반복문의 값을 기호로 표현하여 취약점으로 도달할 수 있는 최적 경로를 파악한다. 하지만 대규모 프로젝트의 경우 경로 폭발(Path Explosion) 등의 문제가 발생할 수 있다.[6]

1.2 퍼저

입력값을 변경하여 바이너리를 테스트하는 퍼저는, 수행 방식에 따라 둠(Dumb) 퍼저와 스마트(Smart) 퍼저로 나눌 수 있다. 먼저 둠 퍼

저는 주어진 입력값을 무작위로 변경하여 바이너리의 충돌을 유발한다. 따라서 대상 바이너리의 분석이 필요 없다는 장점이 있다. 반대로 스마트 페저는 바이너리의 충돌을 높이기 위해 바이너리의 형식(Format)을 분석한 후 적절한 입력값을 생성하여 바이너리의 충돌을 유발한다. 본 논문에서는 분석의 오류를 줄이고자 단순히 입력값만 주면 바이너리의 충돌을 유발하는 텀 기반의 AFL페저를 사용하였다.

III. AFL퍼저와 ASan

코드 커버리지를 높이기 위해 유전 알고리즘을 사용하는 AFL퍼저는 블랙박스(Black box) 검사와 화이트박스(White box) 검사를 지원하며 병렬 퍼징을 통해 빠른 퍼징이 가능하다. 하지만 AFL퍼저 단독으로 사용할 경우 취약점의 종류를 파악하기 위해 소스코드를 분석해야 한다.

빼른 취약점 분류를 위해 AFL퍼저와 구글에서 제공하는 취약점 탐지 도구인 ASan을 함께 사용하면 취약점의 종류를 쉽게 파악할 수 있다. ASan은 메모리 개체 사이에 Redzone을 두어 메모리 접근을 확인하며 Use after free, Memory leaks 등 7가지 취약점을 탐지할 수 있다. ASan을 사용하기 위해서는 CLANG과 GCC(4.8버전 이상)로 컴파일 시 [그림 1]과 같이 -fsanitize=address 옵션만 붙여주면 사용할 수 있고 바이너리만으로 취약점을 파악할 수 있다. 하지만 입력이 필요한 바이너리의 경우 충돌을 유발하는 입력값을 입력해야 취약점 종류를 파악할 수 있다.

```
root@ubuntu:/home/zz/Desktop/afl-2.52b# afl-gcc -fsanitize=address  
-fno-stack-protector q02.c -o q02asan  
afl-cc      by <lcamtuf@google.com>  
q02.c: In function 'solve':  
q02.c:20:3: warning: implicit declaration of function 'system' [-Wim  
plicit-function-declaration]  
    system("/bin/sh");  
    ^  
q02.c: In function 'main':  
q02.c:24:3: warning: implicit declaration of function 'read' [-Wim  
plicit-function-declaration]  
    read(0,buf,1024);  
    ^  
afl-as      by <lcamtuf@google.com>  
[+] Instrumented 6 locations (64-bit, ASAN/MSAN mode, ratio 33%).
```

[그림 18] ASan을 사용하기 위한 컴파일 과정

예를 들면 [그림 2]의 경우 입력값을 주지 않아도 때문에 바이너리가 실행 중인 상태이며

[그림 3]처럼 AFL퍼저를 통해 알아낸 충돌유발
값을 바이너리에 입력해야 실행이 완료되면서
[그림 4]와 같이 취약점 유형 등을 알 수 있다.
또한 컴파일 시 -m32옵션을 사용해서 컴파일
하면 보다 더 다양한 정보를 확인할 수 있다.

```
[root@ubuntu:/home/zz/Desktop/afl-2.52b# ./q02asan
```

[그림 19] 입력 값을 넣기 전 Q02 바이너리 실행 상태

root@ubuntu: /home/zz/Desktop/afl-2.52b/result4/crashes

(그림 20) AFL퍼저로 생성된 Q02바이너리 충돌유발
값

```
root@ubuntu:/home/zz/Desktop/afl-2.52b# ./q02asan  
^N^C^NyRzyyyyyyNyyyyyykyyyyyyyyy<96>f<85>yyyyyyyyü  
=====  
==67293==ERROR: AddressSanitizer: stack-buffer-overflow on  
address 0x7ffe56a1c7c0 at pc 0x71cd  
c770 sp 0x7ffe56a1bf18
```

[그림 4] 충돌유발 값이 입력된 Q02 바이너리

하지만 충돌 유발 값을 입력하지 않으면 [그림 5]와 같이 정상적으로 실행이 완료되면서 취약점 종류를 파악할 수 없다. 따라서 AFL 퍼저로 충돌유발 값을 빠르게 찾는 게 중요하다.

```
root@ubuntu:/home/zz/Desktop/afl-2.52b# ./q02asan  
??  
root@ubuntu:/home/zz/Desktop/afl-2.52b#
```

(그림 5) 정상 실행된 바이너리

IV. 실험

실험에 사용된 바이너리는 스택 베퍼 오버플로 문제가 있는 KISA 데이터 챌린지 1차 시범 문제 중 Q02, 2차 시범문제 중 Q08 소스코드를 사용했으며 ASan에 사용될 충돌유발 팩을 빠르게 찾기 위해 [표 1]과 같이 소스코드를 컴파일 하여 어떤 방법이 가장 많은 충돌을 유발하는지 실험했다.

〔표 1〕 실험용 바이너리의 컴파일 유형 및 분석기법

No.	컴파일 유형, 분석 기법
1	AFL-GCC 컴파일 (화이트박스)
2	GCC 컴파일 (블랙박스)
3	AFL-GCC 컴파일, ASan 활성화 (화이트박스+ASan)

또한 퍼저에 사용된 초기 입력값은 .txt, .drc 등 다양한 확장자를 가진 9개의 파일을 입력으로 사용했다.

Q02, Q08 바이너리 대상으로 5분간 AFL퍼저를 동작시킨 결과 Q02의 경우 [표 2]와 같이 AFL(화이트박스), AFL(화이트박스+ASan), AFL(블랙박스) 순으로 충돌유발 값을 많이 찾았고, Q08의 경우 [표 3]과 같이 AFL(화이트박스), AFL(블랙박스), AFL(화이트박스+ASan) 순으로 충돌유발 값을 많이 찾았다. 테스트 시 AFL 퍼저가 충돌유발 값을 잘 발견할 수 있도록 일반 GCC 컴파일 도구가 아닌 AFL-GCC 컴파일 도구를 사용해 인스트루먼테이션(Instrumentation)된 바이너리를 생성하여 테스트했기 때문에 같은 시간에 화이트박스가 충돌유발 값을 가장 많이 찾은 것으로 예상되며, AFL(화이트박스+ASan)도 AFL-GCC를 사용해 컴파일 하지만 별도의 메모리 영역(Redzone)까지 감시하기 때문에 AFL(화이트박스)보다 충돌유발 값을 많이 찾지 못한 것으로 보인다. 또한 블랙박스 검사는 별도의 가상화 소프트웨어인 QEMU[7]를 통해 바이너리를 테스트하기 때문에 AFL(화이트박스)보다 충돌유발 값을 찾는 속도가 느렸던 것으로 판단된다. 하지만 AFL퍼저의 경우 입력값을 랜덤으로 변경하여 테스트하기 때문에 초기에 주어진 입력값에 따라 실험 결과가 달라질 수 있다.

〔표 2〕 1차 시범문제 중 Q02 바이너리 실험 결과

바이너리 분석기법	충돌 수(중복 포함)
AFL(화이트박스)	124k
AFL(블랙박스)	1083
AFL(화이트박스+ASan)	27.4k

〔표 3〕 2차 시범문제 중 Q08 바이너리 실험 결과

바이너리 분석기법	충돌 수(중복 포함)
AFL(화이트박스)	57.1k
AFL(블랙박스)	789
AFL(화이트박스+ASan)	457

V. 결론

실험 결과 스택 버퍼 오버플로 취약점이 있는 KISA 데이터 챌린지 Q02, Q08 바이너리의 경우 AFL(블랙박스), AFL(화이트박스+ASan) 보다 AFL(화이트박스)을 사용했을 때 같은 시간 내 더 많은 충돌을 발견할 수 있었고, AFL(화이트박스+ASan), AFL(블랙박스)로 바이너리 충돌 값을 알아낸 후 충돌 값을 입력하여 취약점의 종류를 파악하는 것보다 AFL(화이트박스)로 충돌 값을 알아낸 후 -fsanitize=address옵션으로 컴파일된 바이너리에 충돌유발 값을 입력하면 더 빠르게 취약점의 종류를 파악할 수 있었다.

[ACKNOWLEDGEMENT]

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터지원사업의 연구 결과로 수행되었음 (IITP-2018-2018-0-01423)

[참고문헌]

- [1] Han, Wookhyun, et al. "Enhancing Memory Error Detection for Large-Scale Applications and Fuzz Testing." Symposium on Network and Distributed Systems Security (NDSS). 2018.
- [2] Klees, George, et al. "Evaluating Fuzz Testing." Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2018.
- [3] Mechanical Phish. <https://github.com/mechaphish>
- [4] American Fuzzy Lop. <http://lcamtuf.coredump.cx/afl>.
- [5] Serebryany, Konstantin, et al. "AddressSanitizer: A Fast Address Sanity

- Checker." USENIX Annual Technical Conference. 2012.
- [6] 오상환, 김태운, and 김환국. "SW 보안 취약점 자동 탐색 및 대응 기술 분석." 한국산학기술학회 논문지 18.11 (2017): 94–103.
- [7] QEMU.
<https://qemu.weilnetz.de/doc/qemu-doc.html>

사물인터넷 시대에 필요한 중소기업 정보보안수준 연구*

문재웅[†], 박영호[‡]

세종사이버대학교 정보보호대학원

A Study on the Level of Information Security for Small and Medium Businesses in the Age of the Internet of Things*

Moon Jae Woong[†], Park Young-Ho[‡]

Graduate School of Information Security Sejong Cyber University

요 약

4차 산업혁명 시대를 맞이하여 초연결, 초지능, 초정보 등 다양한 형태의 많은 사회적, 기술적 변화가 일어남에 따라 여러 곳에서 정보보호 취약 요소가 쉽게 노출이 되고 있으며, 그로 인하여 국내외 많은 기업들이 보안사고와 사이버공격에 직면하고 있다. 본 논문에서는 사물인터넷(IoT)의 보안위협요인들과 대표적인 공격유형을 분석하고 그에 따른 중소기업의 보안 취약점과 대응방법, 백업 등에 대한 구체적 사례를 제시하여 중소기업 특성을 고려한 사물인터넷 보안에 대하여 정보보호 구축 방향을 제시하고자 한다.

I. 서 론

사물인터넷(Internet of Things)의 개념은 인간과 사물, 서비스 세 가지 분산된 환경요소에 대해 상호 협력적으로 센싱, 네트워킹, 정보처리 등 지능적 관계를 형성하는 사물 공간 연결망이라고 한다. 우리는 정보화 시대와 스마트폰의 시대를 거쳐서 사물인터넷 기술의 발전을 통하여 모든 사물과 사람이 연결되는 초연결 사회로 나아가고 있다. 사물인터넷이 현재 연결되어 있는 숫자는 관련 시장조사에 따르면 약 100~150억 개의 사물이 인터넷에 연결되어 있고 2020년까지 200~700억 개로 그 수가 증가할 것으로 전망된다. 시장조사 업체 맥肯지는 2025년까지 인류의 삶을 가장 급진적으로 변화시키는 것이 사물인터넷 기술이고 전 산업에 핵심 기술로 적용될 것이라고 전망하였다. 따라서 사물인터넷은 국가 경쟁력에 결정적 영향을 미칠 수 있는 기술이

며, 그래서 세계 여러 선진 국가들도 핵심 원천 기술 개발과 서비스 활성화를 위해 사활을 걸고 있다. 이런 상황에서 우리나라 국내 산업의 99.9%를 차지하고 있는 중소기업에 대하여 사물인터넷 시대에 필요한 최소한의 맞춤형 정보보호 구축 방안을 제시하고 연구함으로써 기업의 핵심기술과 자산가치를 보호하여 지속적인 성장을 할 수 있도록 하는 것이 필요하다.

본 논문에서는 사물인터넷 보안사고로 인한 피해사고를 구체적으로 분석하여 중소기업이 피해를 입지 않고 지속적인 성장을 할 수 있도록 최소한의 적정 수준의 중소기업 정보보호 실행 방안을 제시하므로 중소기업 특성을 고려한 사물인터넷 보안에 대한 정보보호 구축 방향을 제시하고자 한다.

II. 사물인터넷 시장과 전망

* 주저자 : jwmoon10@gmail.com
† 교신저자: youngho@sjcu.ac.kr

* 이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임”(NRF-2017R1A2B4011599)

2.1 사물인터넷의 핵심기술

1) 센싱기술

사물인터넷에서는 센싱 모듈을 통해 수집되는 정보를 인터넷을 통하여 공유하기 위해 기본적인 신호처리 및 알고리즘 수행이 가능한 모듈을 포함한 내장된 스마트 센서 기술이 필요하다.

2) 인터페이스기술

사물인터넷 서비스 인터페이스 기술은 사물인터넷의 주요 구성요소(인간, 사물, 서비스)를 통해 특정 기능을 수행하는 응용서비스와 연동하는 기술로서, 사물인터넷의 다양한 서비스 기능을 구현하기 위해서는 ① 정보의 검출, 가공, 정형화, 추출, 처리 및 저장기능 등 검출정보 기반 기술 ② 위치판단 및 위치확인기능, 상황인식 및 인공지능 등 위치정보 기반기술 ③ 정보보안 및 프라이버시 보호기능, 인증 및 인가기능 등 보안기능 ④ 온톨로지(Ontology: 인간이 보고 듣고 느끼고 생각하는 것에 대해 컴퓨터에서 처리할 수 있는 형태로 표현하는 모델) 기술을 통해 다양한 서비스를 제공할 수 있는 인터페이스 역할을 수행할 수 있어야 한다.

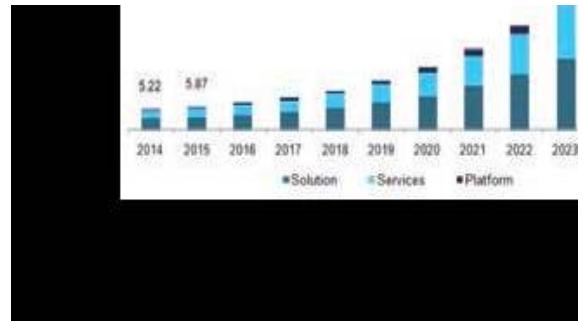
3) 네트워킹 기술

사물인터넷에서 네트워킹 기술은 분산된 환경에서 존재하는 다양한 디바이스들의 물리적인 연결을 수행하는 유무선 네트워킹 기술로서, WPAN(Wireless Personal Area Networks), 와이파이, 3G~4G LTE, 5G, 블루투스, 이더넷, 위성통신, BcN(Broadband convergence Networks) 등을 이용할 수 있다.

2.2 산업용 사물인터넷 시장

산업용 사물인터넷 시장규모는 2015년 900억불에서 2020년에는 1,100억불로 증가하며 IIoT의 2030년 글로벌 경제에 미치는 추가적 영향이 14.2조 달러에 달하고 연평균 성장률은 약 8.03%에 달할 것으로 예측하였다. 2016년 글로벌 IIoT 시장규모는 1,092억 달러였으며 2025년에는 9336억 불에 달하고 성장률은 27.8%를 보일 것으로 Grand View Research에서 예측하였다. 세부적인 컴포넌트별 규모를 보기위해 독일의 시장을 근거로 살펴보겠다. 아래 [그림1]의 컴포넌트별 시장규모의 시장 점유율을 살펴보면 산업용 IoT 솔루션은 2016년

최대 시장 점유율을 55%를 차지했으며, 솔루션 제공 업체는 운영을 혁신하고 새로운 비즈니스 모델을 만들 수 있도록 다양한 산업 분야 및 비즈니스에 정보시스템, 장치 및 센서를 통합하는 데 점점 더 노력하고 있고 의사 결정을 지원하고 제품 및 서비스를 향상시키는 다양한 출처의 데이터를 기반으로 한 분석을 통해 혁신적인 통찰력을 제공하므로 향후 상당기간 높은 성장이 기대된다. IIoT 서비스는 2017년부터 2025년까지 연평균 29%의 CAGR을 기록하며 이 분야의 성장은 클라우드 컴퓨팅 시장 개발, 지속 가능한 스마트 지원에 대한 정부의 노력 증가로 인한 효과가 크다.



[그림1] 독일의 IoT 시장의 Component별 규모(단위:10억달러)

2.3 산업용 사물인터넷 도입에 따른 효율성

사물인터넷 기술을 제조업에 도입한다면 비용절감과 생산효율화를 동시에 달성할 수 있다. 이는 제조공장의 모든 자원과 설비를 실시간 데이터에 기반하여 최적화해 유휴자원을 줄이고 가용성을 극대화할 수 있기 때문이다. 생산자들은 중앙관제를 통해 여러 공장에 있는 기계설비들을 실시간으로 모니터링할 수 있다. 공장 안에 기계장치와 시스템이 연결되면 다양한 정보가 제공되고, 이러한 정보를 바탕으로 인간의 간섭 없이 생산시스템 자동화가 가능하다. 따라서 제조에서 판매까지 전체 공급체인에 대한 정보를 실시간으로 제공하여 체계적인 관리가 수월해질 전망이다. 맥킨지 앤 커퍼니(McKinsey & Company)의 2015년 보고서에 따르면, 사물인터넷 응용기술을 생산 공정에 도입함으로써 10~20%의 에너지를 절감할 수 있으며, 20~25%의 노동효율성 증가가 발생할 것으로 예상된다.

2.4 사물인터넷 도입에 따른 산업전반의 변화

사물인터넷, 클라우드, 빅데이터, 인공지능 등으로 인

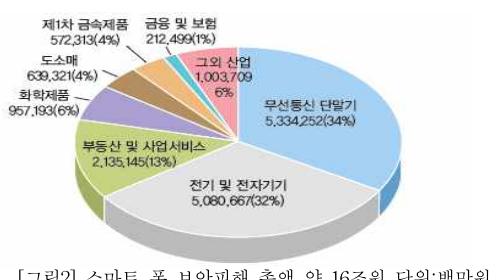
하여 가까운 미래에 기존에는 없거나 전혀 생각하지 않았던 혁신적인 사업 기회가 창출될 수 있다. 사물인터넷 혁명은 에너지, 의료, 제조업 등 다방면에 영향을 미치고 인간과 기계의 상호작용에 근본적인 변화를 가져 올 전망이다. 기존의 시장이 소품종 대량생산 이었다면 앞으로는 다품종 소량생산이 될 것이다.

III. 사물인터넷 보안 위협 요인

3.1 사물인터넷 피해현황

사물인터넷은 소형화와 신속성, 저 전력 등 환경요인의 제약으로 무선 전송매체의 비중이 매우 높은 것이 사실이다. 이러한 전송매체의 사용은 기존 무선 네트워크의 취약성을 이어받고 있어서 위험성이 매우 높다.

한국인터넷진흥원 전망에 따르면 사물인터넷의 해킹에 따른 경제적 손실이 18조에 이를 것이라고 했다. 이는 자연재해(2.7조) 및 사이버공격(3.6조)으로 인한 피해를 비교해 볼 때 엄청난 수치임을 알 수 있다. 향후 해킹에 따른 피해규모는 보다 더욱 커질 것으로 전망한다. [그림2, 그림3] 와 같이 스마트 폰과 자동차 산업에 따른 피해가 16조원과 24조원으로 예측하고 있다. 특히 스마트폰은 다양한 서비스 빛 제어가 가능하고, 휴대가 용이하여 편리성과 이동성, 실시간성 등의 다양한 장점을 갖고 있지만, [그림2] 과 같이 스마트 폰의 무선 전송매체 및 운영체제, 서비스 환경 등에 따른 취약성 총 피해액이 약 16조원에 이르고 있음을 알 수 있다. 이중 무선통신 단말기로 인한 피해가 34%이며, 전기 및



[그림2] 스마트 폰 보안피해 총액 약 16조원 단위:백만원



[그림3] 자동차 보안피해 총액 약 24조원 단위:백만원

전자기기로 인한 피해가 32%로 가장 큰 비중을 차지하고 있으며, 이 밖에도 부동산과 사업 서비스, 화학, 도소매, 금속, 금융 및 보험 등 여러 산업 분야에서도 보안 피해가 발생하고 있음을 알 수 있다.

또한 최근 보안과 무관하다고 여겨졌던 자동차 분야를 보면 [그림3]와 같이 보안 피해액이 약 24조원에 이르고 있어, 앞으로 사물인터넷 산업 전반으로 피해가 확산될 것으로 예상된다. 자동차가 인공지능에 가깝게 될 수록 오디오기기, 자동차 소프트웨어, 블루투스, ECU 단말기 등에 취약점을 찾아내어 해킹 공격이 직, 간접적으로 들어오기 때문에 피해가 증가할 것이다.

3.2 사물인터넷 위협요인 분석

3.2.1 가로채기 공격 유형

무선 전송 매체를 사용하는 사물인터넷은 많은 기기들로부터 적은 용량의 데이터를 수집 및 전송하며, 시간적, 공간적 제한은 받지 않기 때문에 다수의 사물들을 연결하는데 사용되고 있다. 그러나 가로채기(intercept) 공격에 취약하여 위협요인이 되고 있다. 이에 대한 대응 기술로 무선인증과 데이터 암호, 보안채널(secure channel), 터널링(tunneling) 등이 있지만 사물인터넷 기기의 수가 급속하게 증가할 경우, 성능 및 관리상의 문제들이 발생한다.

3.2.2 방해 및 위로변조 공격 유형

방해(interrupt) 공격은 유무선의 경계 없이 시도되고 있으며 이에 따른 피해도 지속적으로 증가하고 있다. 대표적인 공격 유형은 ‘서비스 거부공격(denial of service)’이 있으며 우리가 흔히 보는 세트톱박스, 냉장고, TV, 컴퓨터, 가전제품 등이 무선 랜이나 블루투스로 연결되어 있어서 이런 것들을 통한 공격이다. 그리고 위조(fabrication)와 변조(modification) 공격은 유무선 네트워크상에서 전송 또는 저장 데이터에 대한 위로변조나 사용자를 위장한 공격 등을 포함하며 복합 형태로 공격이 이루어진다. 이에 대한 사례로 원격지에서 인가된 사용자를 위장해 자동차의 제어장치를 조작하거나 정상적인 기기를 가장해 공격하는 등 인간의 생명까지 위협하는 설정이다. 또한 위로변조 공격을 통해 2차

적인 공격으로 이어져 막대한 경제적 손실까지 입힐 수 있다.

3.2.3 바이러스 및 웜 공격 유형

사물인터넷은 궁극적으로 기기들을 관리할 시스템을 필요로 하기 때문에 바이러스 및 웜의 유입 가능성은 배제할 수 없다. 그러나 이와 같은 공격에 대응하기 위해 백신을 사용할 경우 속도 저하문제와 이를 악용한 공격 가능성도 매우 높다. 사물들을 관리할 시스템(스마트 기기 등)이 바이러스나 웜에 걸릴 경우 인터넷이나 와이파이에 연결되어 있어서 피해 확산이 매우 빠르고 경제적 손실이 매우 크므로 공격차단의 보안 제품이 필요하다.

3.2.4 IoT 제품의 보안 취약요소

사물인터넷 환경에서 보안 취약 요소가 생길 수 있다. 여러 가지 들어난 취약요소 중에서 IoT 제품과 연관된 취약요소를 확인해 본다.

1) 평문로컬 API

장치에 내장된 구성요소 및 소프트웨어는 간혹 LAN로컬 통신을 위한 최신 암호화 표준을 사용하지 못하는 경우가 있다. 만약, 평문으로 통신할 경우 큰 위협 요소가 될 수 있는데, 이러한 경우에는 HTTPS 또는 SSH와 같은 일반적인 암호 프로토콜을 사용하여 통신할 필요가 있다.

2) 평문 클라우드 API

주요 메이저 서비스 제공업체는 개인정보보호 및 신뢰성을 보장하기 위해 일반적으로 암호화 방식을 채택하고 있는 방식이다. 그러나 IoT 장비와 연결되는 일부 서비스의 경우, 일반적인 표준을 준수하지 못하는 경우가 있는데 클라우드 API가 평문으로 통신하게 될 경우, 보안 위협요소가 크게 증가하게 된다.

3) 비 암호화된 저장

데이터를 평문으로 저장하게 될 경우, 인가되지 않는 자가 해당 데이터에 대한 접근이 가능할 수 있다. 데이터는 인가된 사용자만 접근할 수 있도록 암호화하여 보관해야 한다.

4) 원격 쉘 접근

IoT 제품에 원격 쉘로 접근이 가능한 경우 보안 취약요소가 될 수 있다. 원격 쉘은 개발 단계에서 유용하게 사용될 수 있으나, 제품의 개발이 끝나고 실제 IoT 제품의 출시가 필요할 경우에는 해당 IoT 제품의 원격 쉘 접근을 할 수 없도록 조치를 하여야 한다.

5) 백도어 계정

IoT 장치 제조업체는 간혹 기본 계정 또는 서비스 계정을 포함한다. 이 계정은 종종 추측 가능한 암호를 사용하는 경우가 있다. 해당 계정은 장치 고유의 암호로 보호될 수 있으나, 패스워드 생성 알고리즘의 경우 쉽게 유추가 가능할 수 있다.

6) UART 액세스

UART(Universal Asynchronous Receiver Transmitter) 인터페이스는 간혹 직렬 케이블 연결을 통해 정상적인 인증 메커니즘을 우회하는 방법으로 IoT 장치에 공격자가 접근하여 장치를 변경할 수 있다. 또한, UART 인터페이스는 일반 사용자의 권한을 초과하는 루트 액세스의 권한을 부여하는 권한이 있다.

3.3 중소기업 기술유출 사례

2018년 국회 산업통상자원 중소벤처기업위원회에 따르면 지난해 중소기업의 기술유출 건수는 78건으로 2016년 대비 20건 증가했다. 총 피해액은 1022억 원으로 2015년 902억 원과 비교하면 100억 원 이상 늘어났다. 지난 5년간 해외 기술유출 시도 적발은 총 152건이며 이 중 중소기업의 기술유출 시도는 102건으로 전체의 67%를 차지했다. 대기업의 해외 기술유출 건수인 35건(23%)보다 3배가 넘는 수치다. 최근에는 사이버 해킹으로 인한 기술유출 피해액이 연간 3000건 이상 발생하는 등 그 수법도 날로 다양해지고 있으며 중소기업의 사이버 상의 기술유출 시도는 취약한 중소기업의 현실을 볼 때에 속수 문책으로 당할 수밖에 없다. 기술이 유출된다는 것은 빼앗긴 회사의 생존을 위협할 수 있는 중대한 사안이다.

IV. 결론

4차 산업혁명 시대를 들어서면서 우리에게는 가장 많이 일상생활에서 직접 보고 느끼고 또한 다양한 산업

분야에서 쓰이는 사물인터넷의 응용분야와 범위가 빠르게 진화되고 있는 것이 사실이다. 그러나 이런 상황에서 사물인터넷에 대한 공격기술도 다양한 공격대상과 유형으로 나타나고 있으며, 사물인터넷 기술의 보편화, 기술발전 등에 저해요인으로 작용할 수도 있다. 국내 산업의 99%를 차지하는 중소기업이 사물인터넷 기술을 발전시키고 산업을 키우는 곳이면서 동시에 사물인터넷 기술과 제품을 통하여 경쟁력을 확보하고 성장을 할 수 있어야 한다. 4차산업혁명 시대의 사이버테러의 피해로부터 중소기업을 보호하여 경제적 피해를 입지 않고 안전적이고 지속 가능한 성장을 하도록 만들어야 한다. 따라서 본 논문은 사물인터넷의 보안 기술과 위협요인들의 대표적인 공격유형을 살펴보고 사고 예방, 보안기술 적용방안, 사물인터넷 기술개발, 대응방안을 분석하고 그에 따른 중소기업의 취약점을 확인하고 대응방법, 백업에 대한 구체적 사례를 제시함으로써 중소기업에게 최소한의 유용한 자료로 활용될 것을 기대한다.

[참고문헌]

- [1] 김종덕, “사물인터넷 도래 현황과 전망” 서울:텔코경영연구원, pp15-18, 2015.
- [2] 공만식, 최홍준, 유보현, “사물인터넷(IoT)기술동향과 전망”, 대한기계학회, 56(2), pp32-36, 2016.
- [3] 한국인터넷진흥원 “사이버침해사고(신고/조사) 기업 규모별 현황”, 2017년 통계 참조
- [4] 최용수, “산업용 사물인터넷(IoT) 시장전망과 기술 동향”, 전자공학회지, 44(5), pp. 43-49, 2017.
- [5] 김양훈, 장향배, “적정 수준의 중소기업 정보보호 추진방향”, 한국정보보호학회지, 23(4), pp45-46, 2013.
- [6] 전정훈, “사물인터넷의 보안 위협 요인들에 대한 분석”, 융합보안논문지, 15(7), pp. 48-52, 2015.
- [7] 이형택, “2017년 랜섬웨어 침해공격 및 2018년 공격 전망”, 한국랜섬웨어침해대응센터 자료 제공, 2018.
- [8] 이동혁, 박남재, “IoT 제품 보안 인증 및 보안성 유지 관리방안”, 한국통신학회지, 33(12), pp28-34, 2016.



(주)오픈링크시스템



에리콤 어플리케이션 클라이언트 가상화 솔루션
[서버 베이스 컴퓨팅 클라우드]
Any Application from Any Device, Anywhere

ICT 분야 전문 기업!

IT CONVERGENCE

빅데이터 및 오픈소스

SI | ITO

Global IT Leader!

모든 비즈니스 영역을 통합하는 통찰력으로
고객의 니즈를 완벽히 분석한 최적의 서비스로
미래를 선도하는 최첨단 기술력으로

미래의 가치를 먼저 생각하는 기업

Total Solutions

- SI-NI 사업
- Print On Demand 솔루션 사업

Smart Service

- Mobile 솔루션 사업
- 금융 솔루션 사업

Art Technologies

- 산업용 PDA 사업



큰대 믿을 i동

 대신정보통신주식회사

Daishin Information & Communications Co., Ltd.

본사 : 광주광역시 서구 상무중앙로 110

서울 : 서울특별시 금천구 가산디지털1로, 205-28 대신정보통신빌딩

Tel_062-225-7350 Fax_062-226-0716

Tel_02-2017-5000 Fax_02-2107-5015

www.dsic.co.kr

대한민국 신성장 핵심 정보기술의 중심

아이티센이 여는 4차 산업혁명 세상



아이티센

주식회사 아이티센은 컨설팅, 구축, 운영에서 융복합서비스까지
Total IT/ICT 분야의 지능정보화 서비스를 제공합니다.

신뢰를 바탕으로 고객의 성공을 이끌 수 있도록
끊임없이 도약하는 아이티센이 되겠습니다.