

2020 춘계학술 발표대회

[논문집]

2020년 5월 22일(금) ~ 23일(토)

장소: 오션스위츠 제주호텔



주최 사단법인 한국인터넷정보학회 주관 제주대학교 SW중심대학사업단

후원 (주)카카오, (주)LGCNS, (주)아이티센, (주)콤텍정보통신, (주)한컴위드, 도서출판 길벗, (주)넷케이티아이, LG히다찌(주)

B-4 ☆ 건설의 객관적 평가 방법 및 시스템 개발	29
문초이, 유현중, 이지원, 김은빈, 김수복, 계슬아, 위예진(순천향대), 백유상(고려대), 이언석(순천향대)	
B-5 2차원 이미지(깊이 카메라)를 통한 실시간 환경에서 물체의 부피를 구하는 알고리즘 연구	31
배성현(성균관대)	
B-6 자동차 번호판 검출을 위한 인조 번호판 생성 시스템	33
김상준, 이유진, 박구만(서울과기대)	
B-7 요구 공학을 이용한 DAQ system의 비기능적 요구사항 추출	35
김경식, 이석원(아주대)	

C. 인터넷보안 (좌장 : 강지원 교수 (세종대))

C-1 안드로이드 디바이스 자원 사용량 분석을 이용한 크립토재킹 탐지 기술	41
이희승, 곽진(아주대)	
C-2 비정형 시나리오 기반 원자력시설의 사이버보안 훈련 평가 체계 강화 방법론 제안	43
김상우, 류진호(한국원자력통제기술원)	
C-3 CLAPP 기반 안드로이드 악성코드 분류 기법	45
정해선, 김득훈, 곽진(아주대)	
C-4 Just Walk Out 서비스에서의 안전한 소비자 그룹화 방법	47
이민경, 곽진(아주대)	
C-5 웹사이트 광고 배너 분석을 이용한 웹툰 침해 의심 사이트 탐지 기술	49
김의진, 최슬기, 곽진(아주대)	
C-6 지멘스 S7 PLC 취약점을 이용한 공격 방법 연구	51
김준원, 김장훈, 이민규, 서정택(순천향대)	
C-7 PLC 제어로직 변조를 통한 DoS 공격 및 대응방안 연구	53
김장훈, 김준원, 정다운, 서정택(순천향대)	

안드로이드 디바이스 자원 사용량 분석을 이용한 크립토재킹 탐지 기술

이희승*, 박진**

*아주대학교 컴퓨터공학과 정보보호응용및보증연구실

**아주대학교 사이버보안학과

e-mail : *hslee.isaa@gmail.com, **security@ajou.ac.kr

Detection Technique of Cryptojacking Using Android Device Resource Utilization Rate Analysis

Hee-Seung Lee*, Jin Kwak**

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

**Department of Cyber Security, Ajou University

요 약

최근 암호 화폐가 주목을 받으면서 다른 사용자의 디바이스를 침해하여 암호 화폐를 채굴하는 방식이 생겨나고 있다. 암호 화폐를 채굴 하는 방식은 디바이스에 직접 설치되는 방식과 불특정 다수의 자원을 점유하기 수월한 브라우저의 취약점을 이용한 방식이 있다. 이 중 브라우저의 취약점을 악용한 채굴 방식 중 웹사이트에 접속만해도 스크립트가 실행되어 채굴이 되는 크립토재킹은 사용자가 채굴행위에 대해 인지하기가 어려워 지속적으로 증가하고 있다. 크립토재킹 공격이 성공하면 사용자 디바이스의 자원을 점유하여 CPU나 네트워크 대역폭 같은 자원들의 사용량이 높아지게 된다. 이때 디바이스의 성능이 저하되면서 소유 디바이스의 가용성이 침해된다. 이에 따라 본 논문에서는 안드로이드 디바이스에서 자원 사용량을 분석하여 크립토재킹을 탐지하는 기술을 제시한다.

1. 서 론

최근 암호 화폐가 주목을 받으면서 다른 사용자의 디바이스를 침해하여 암호 화폐를 채굴하는 방식이 생겨나고 있다. 이러한 방식 중 웹사이트에 접속만 해도 웹 브라우저를 통해 채굴 프로그램을 동작시키는 크립토재킹은 공격의 난이도가 낮은 편에 속해 이를 이용한 공격이 증가하고 있다[1]. 이는 웹사이트에 심어둔 악성코드를 활성화시켜 피해자의 CPU나 인터넷 대역폭을 암호화폐 채굴에 이용하는 공격이다.

파이어아이스는 2018년 크립토재킹의 영향을 받는 상위 10개 국가를 발표했으며, 한국은 8.43%로 4위에 올라있었다[2]. 또한 한국인터넷진흥원의 자료에 따르면 2018년 1분기 국내 웹 번조 침해사고의 약 27%가 크립토재킹 공격임이 확인되었다[3].

이에 따라 안드로이드 디바이스 환경에서 자원사용량을 분석하여 지속적으로 증가하고 있는 크립토재킹 공격을 탐지하는 방안을 제시하고자 한다.

본 논문에서는 2장에서 브라우저 취약점 기반 크립토재킹 동작 과정 및 안드로이드 디바이스에서 자원 사용량 분석에 대해 설명하고, 3장에서 크립토재킹을 탐지하기 위한 제안 사항을 제시한 뒤 4장에서 제안 사항에 대한 실험결과를 다루고, 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

2.1 브라우저 취약점 기반 크립토재킹 동작 과정

크립토재킹 공격이 성공할 경우 다음과 같은 과정으로 동작 된다[1].

Step 1. 피해자는 크립토재킹 공격이 성공하여 악성 스크립트가 삽입된 웹사이트에 접속

Step 2. 스크립트를 지원하는 브라우저는 웹사이트에 삽입된 악성 스크립트를 실행

Step 3. 스크립트가 실행되는 동안 CPU와 같은 자원들을 점유하며 채굴 진행

Step 4. 채굴된 코인들은 공격자의 '코인지갑'으로 전송

2.2 크립토재킹 안드로이드 디바이스 자원 사용량 분석

리눅스 기반 안드로이드 디바이스에서 크립토 재킹으로 인해 브라우저 상에서 동작하고 있는 채굴기의 자원 사용량을 분석한다.

리눅스 기반 안드로이드 디바이스는 PC와 연결하여 디바이스의 상태 및 자원 사용량을 확인해 볼 수 있다. adb shell을 이용한 방법으로 cmd창에서 명령어를 입력하여 확인할 수 있다[4].

(표 1) 안드로이드 디바이스 CPU 자원 모니터링

```
#> adb shell "top -s 9"
#> adb shell "top -n 1 -k %-CPU -o %CPU,PID,USER,ARGS -s 1"
```

top 명령어는 안드로이드 디바이스의 CPU 자원을 모니터링 하는 명령어로 첫 번째 명령어는 그림1, 그림2와 같이 출력된다. 두 번째 명령어는 특정 항목만 출력되도록 하는 명령어이며, CPU 사용량과 프로세스 ID, 프로세스를 실행한 유저, 실행된 프로그램을 뜻한다.

크립토재킹 사이트는 웹사이트 내에 크립토재킹 스크립트를 포함하고 있는 심플 테스트베드를 제작하였고, 해당 테스트베드에 접속하여 크립토재킹이 실행되었을 때 자원 사용량을 측정 및 분석하였다.

```

$ top -m 1 -n 1
top: 17:00:10 up 19:00, 19990-16n-1utasks: 488 total, 3 running, 476 sleeping,
Mem: 5487240k total, 5237844k used, 2295396k free, 118808k buffers
Swap: 2097148k total, 1065312k used, 1031836k free, 1634288k cached
PID USER      PR  NI  VIRT  RES  SHR  S CPU% TIME+ ARGS
13853 u0_625    10  -10 1.2G 196M 104M S 36  6  0:10.81 com.android.chr+
14388 u0_1667   20  0 1.1G 118M 103M S 70  0  0:02.57 com.android.chr+
13894 u0_a159   20  0 1.2G 221M 129M S 48  8  0:04.42 com.android.chr+
13614 u0_a159   16  -4 1.3G 238M 188M S 40  0  0:12.12 com.android.chr+
14489 shell     20  0 12M 4.5M 3.8M S 28  8  0:00.00 com.android.chr+
3396 system     5  -15 3.1G 23M 19M R 23  3  0:00.29 surfaceflinger
3364 system   -15  -15 1.45M 4.9M 4.2M S 23  3  0:00.00 android.hardware+

```

(그림 1) 정상 사이트의 브라우저 자원 사용량

```

$ top -m 1 -n 1
top: 17:00:10 up 19:00, 19990-16n-1utasks: 488 total, 4 running, 477 sleeping,
Mem: 5487240k total, 5237156k used, 230104k free, 83152k buffers
Swap: 2097148k total, 1083700k used, 1007448k free, 1061468k cached
PID USER      PR  NI  VIRT  RES  SHR  S CPU% TIME+ ARGS
13851 u0_1630    10  -10 1.5G 205M 117M S 78  3  0:34.75 com.android.chr+
26307 u0_a52    20  0 4.5G 236M 175M S 72  5  0:00.00 com.android.chr+
3525 system     5  -15 3.1G 314M 176M S 82  5  0:00.00 system_server

```

(그림 2) 크립토재킹 사이트의 브라우저 자원 사용량

(표 2) Thread 개수에 따른 (표 3) 정상 사이트 행위에 따른 크립토재킹 자원 사용량

사용 Threads	CPU 자원 사용량	행위	CPU 자원 사용량
10	180 ~ 792	새로고침 및 사이트 접속	80 ~ 150
9	163 ~ 786		
8	124 ~ 780	동영상 재생	60 ~ 80
7	92 ~ 713		
6	83 ~ 608	검색 서비스	80 ~ 90
5	64 ~ 510		
4	54 ~ 417	사이트 로그인	50 ~ 60
3	40 ~ 308		
2	31 ~ 206	메일 서비스	40 ~ 60
1	20 ~ 104	웹문 서비스	50 ~ 80

3. 제안 사항

2.2절에서 분석한 크립토재킹의 특징을 이용하여 시간당 평균 자원 사용량을 구한다.

크립토재킹 사이트에서는 그림2와 같이 CPU 자원 사용량이 792로 나왔으며, 정상 사이트에서는 사이트 접속과 새로고침 시 자원 사용량이 최대 150으로 나와 스크립트에서 CPU 자원 사용량에 대한 기준치를 150으로 정했다.

(표 4) 안드로이드 디바이스에 적용되는 스크립트

```

#!/system/bin/sh
average=150 // CPU 자원 사용량에 대한 기준치
while :
do
sum=0
for i in `seq 0 9`; do
usagearr[$i]="top -n 1 -k -%CPU -o %CPU,PID,USER,ARCS -s 1 |
grep chr | head -1 | sed -e 's/^ */g' | cut -d ' ' -f 1 |
awk '{printf \"%d\",$1}'"
if [ -z "$usagearr[$i]" ]; then
usagearr[$i]=0
fi
let sum=$sum+$usagearr[$i]
done
let sum=$sum/10
if [ $sum -gt average ]; then
service call isms 7 i32 0 s16 "com.android.mms.service" s16 "Phone
number" s16 "null" s16 "Detecting CryptoJacking!" s16 "null" s16 "null"
am force-stop com.android.chrome
fi
done

```

adb shell을 이용하여 안드로이드 디바이스 내의 스크립트를 작성한 후, 계속해서 자원 사용량을 분석할 수 있도록 백그라운드로 실행시켰다.

(표 5) 스크립트 백그라운드 실행

```
#> adb shell sh -T- /script_PATH/script.sh
```

4. 실험 결과

시간당 평균 자원 사용량은 브라우저의 프로세스 중 CPU 사용량이 가장 높은 1건을 10초간 기록하여 평균을 구했다. 또한, 표 4 스크립트에서 나타난 기준치인 150보다 높은 CPU 사용량이 탐지될 경우 디바이스의 자원탈취 및 채굴이 진행되지 않도록 브라우저의 프로세스를 종료시킨 후, SMS를 전송하여 사용자가 인식할 수 있도록 하였다.

Detecting CryptoJacking! 오전 11:54

Detecting CryptoJacking! 오전 11:57

(그림 3) 크립토재킹 탐지 및 사용자에게 알림

제안 사항에서 분석한 특징과 시간당 평균 자원 사용량에 기반하여 크립토재킹 탐지 실험을 한 결과 자원 사용량이 높아지는 브라우저의 프로세스를 탐지하였다. 하지만, 크립토재킹이 정상적인 사용과 비슷한 자원 사용량을 점유하도록 설정된 경우 탐지하는데 어려움이 존재한다.

5. 결론

본 논문에서는 안드로이드 디바이스에서 자원 사용량 분석을 이용한 크립토재킹 탐지를 위해, CPU 자원의 시간당 평균값을 산출하여 탐지 방안을 제시하였다. 또한, 탐지 및 알람을 위해 shell 스크립트를 이용하였다. 모든 크립토재킹에 대한 탐지를 해야 하나, 소량의 자원을 사용하는 경우 탐지하는데 한계가 있고, shell을 이용하여 스크립트를 백그라운드로 실행한 것으로 스크립트가 종료되거나, 디바이스가 재부팅될 경우 다시 기기를 연결하여 스크립트를 실행해야 하는 제약이 있다.

향후, 자원 사용량이 낮은 크립토재킹을 탐지할 수 있는 추가모듈과 안드로이드 디바이스에서 스크립트를 실행하는 방안을 연구할 계획이다.

참고 문헌

- [1] 고동현, 정인혁, 최석환, 최윤호, "크립토재킹 사이트 탐지를 위한 동적 분석 프레임워크", 정보보호학회논문지, 제28권 제4호, pp.963-974, 2018.08
- [2] Fireeye, "How the Rise of Cryptocurrencies Is Shaping the Cyber Crime Landscape: The Growth of Miners", <https://www.fireeye.com/blog/threat-research/2018/07/cryptocurrencies-cyber-crime-growth-of-miners.html>, 2018.07.
- [3] 이재훈, "2018년 3분기 사이버 위협 동향 보고서", 한국인터넷진흥원, pp.30-44, 2018.10.22.
- [4] Android 스튜디오, "Android 디버거 브리지(adb)", <http://developer.android.com>, 2020.02.17.