Access_log를 활용한 매크로 프로그램 탐지 시스템 개발

김준영[O], 김규림[*], 박한솔[*], 안승환[*], 이현탁[**], 최상용[*], 이종락[*]
[O]영남이공대학교 사이버보안학과,
[*]영남이공대학교 사이버보안학과,
[**]주식회사 시야인사이트
e-mail: fawisn@gmail.com[O], kyurimkyurim2017@gmail.com[*], hhh4081@naver.com[*], hwan5105@gmail.com[*], htlee@iseeya.co.kr[**], csyong95@gmail.com[*], jllee@ync.ac.kr[*]

Development of Macro Program Detection System using Access_log

Jun-young Kim[O], Kyu-rim Kim[*], Han-sol Park[*], Seung-hwan An[*], Hyun-tak Lee[**], Sang-Yong Choi[*], Jong-Rak Lee[*]
[O]Department of Cyber Security, Yeungnam University College,
[*]Department of Cyber Security, Yeungnam University College,
[**]SeeYaInsignt

● 요 약 ●

매크로 프로그램에 의한 피해는 꾸준히 보고되어 왔다. 최근 코로나 19로 인해 마스크 품귀 현상이 발생하여 매크로 프로그램을 이용한 구매가 사회적 이슈로 떠오르면서 매크로에 대하여 법적 처벌을 하는 등 매크로에 대한 관심이 부각되고 있다. 본 논문에서는 매크로 프로그램을 효과적으로 탐지하기 위한 방법을 제안한다. 본 논문에서는 매크로 탐지를 위해 access.log의 다양한 필드를 분석하고, 이 중 매크로 식별에 효과가 있는 필드를 추출하여 매크로를 탐지한다. 제안하는 기술의 효과성 검증을 위해 매크로의 공통적인 기능이 포함된 매크로 프로그램을 직접 제작하고, 실제 구매 웹사이트와 유사한 판매 사이트를 제작하여 실험을 통해 제안하는 방법이 매크로를 효과적으로 탐지하는 것을 보인다.

키워드: 매크로(marcro), 탐지(detect), 셀레니움(Selenium), 웹애플리케이션(Web application)

I. Introduction

IT기술의 발전은 사용자에게 편리함을 가져다주는 반면, 그 역기능 또한 확대되고 있다. 대표적인 예로 사용자의 행위를 자동으로 수행해 주는 매크로와 같은 프로그램이 있다. 최근 코로나 19로 인하여 마스크 품귀 현상이 발생하자 매크로를 이용한 온라인 쇼핑몰에서 마스크를 구매하기 위한 매크로도 생겨나고 있다. 이는 '마스크 및 손소독제 매점매석 행위 금지 등에 관한 고시(기획재정부고시 제2020-5호)'의 위반이다. 본 연구에서는 사회적 이슈가 된 매크로를 효과적으로 식별하기 위한 방법을 제안한다. 이를 위해 매크로를 분석한 다음, 일반적으로 매크로가 가지고 있는 기능을 포함한 매크로를 제작하고, 매크로 데이터 수집을 위한 서버를 구축, 실험을 통해 제안한 방법이 효과적으로 매크로를 식별할 수 있다는 것을 검증한다.

II. Preliminaries

1. Related works

1.1 국내 매크로 피해사례

2020.03.05. 서울신문에 따르면 최근 코로나19 확산으로 마스크가 귀해지자 국내 주요 전자상거래 사이트에서 '매크로'를 돌려 10만장의 마스크를 싹쓸이한 사례와 '최대 8배'의 폭리를 취한 일당이 잡혔다[1]. 또한 2019.11.14. NEWSIS에 따르면 유명 아이돌 그룹 콘서트 관람권을 매크로 프로그램을 이용해 2016년 5월부터 조직적으로 대량 구매한 뒤 2~10배의 가격으로 되파는 수법으로 약 7억원 이상의 부당이득을 챙겨 지속적이고 조직적인 매크로의 문제점이 제기되고 있다.[2]

1.2 해외 매크로 피해사례

2020.04.19. 게임 전문매체 '유로게이머에 따르면 오픈소스 기반의 매크로인 '버드 봇'으로 닌텐도 스위치를 구매하는 자동화 봇(매크로)이 등장하였고, 이로 인해 닌텐도의 품귀 현상이 발생했다. 국내 온라인 최저가가 3배, 오프라인 매장은 대기표를 받고 예약을 하는 일까지 발생하였고 해외의 경우 공급 망에 마비가 될 정도로 영향을 미쳤다고 전했다[3].

2019.06.30. 중앙일보에 따르면 세계적으로 유명한 아이돌 그룹 'BTS'의 콘서트 표의 공식 가격은 9만 9천원이다. 일보 오사카의 공연 예매가 시작되자마자 5분도 안 돼 매진되고, 해외 티켓 중개 사이트에 한화 480만원으로 등록되며 판매되었다. 이에 대해 전문 암표상이 매크로 프로그램을 제작하여 구매했다는 것이 경찰의 설명이 있다고 전했다[4].

1.3 Access_log

Access_log는 일반적으로 웹서버에서 접속자의 정보를 보여주는 로그로, 서버의 설정과 저장되는 정보는 Fig.1, Fig. 2, Table. 1.과 같다.

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\"
          \"%{User-Agent}i\"" combined
```

Fig. 1. Access_log Structure

```
127.0.0.1 - - [06/Jun/2020:18:22:16] "GET /index.php HTTP 1.1"
  ①           ②                      ③    ④         ⑤
200 1665 "http://www.google.com" "Mozilla/5.0 ……"
⑥   ⑦          ⑧                    ⑨
```

Fig. 2. Content for each field of Access_log

Table 1. Description to content

No	Description
①	원격 호스트 IP
②	요청 날짜/시간
③	method(메서드)
④	요청 URL
⑤	프로토콜 버전
⑥	HTTP 상태 코드
⑦	HTTP 헤더를 제외한 전송 크기
⑧	referrer(리퍼러)
⑨	User-Agent

기록되는 정보 중 데이터 전송 크기, Referrer, User-Agent등의 정보는 요청패킷을 만들 때 정상과 동일하게 조작이 가능하며, Method, URL, Protocol 버전의 경우 클라이언트에서 자동으로 생성되는 정보이다. 반면 원격 호스트 IP, 요청날짜/시간의 경우 사용자와 프로그램이 접속할 때 위변조가 불가능한 기록이다. 따라서 매크로와 사용자의 접속을 구분하기 위한 특성인자로 활용 가능한 정보는 각 아이피별 접속 빈도를 확인 할 수 있는 원격 호스트 IP와(①) 날짜와 시간(②)이다. 본 연구에서는 매크로 식별을 위해 Access_log에 기록된 단위 시간당 요청 수를 활용한다.

III. 액세스 로그를 통한 매크로 식별

아파치 톰캣의 액세스 로그를 이용하여 상품 자동 구매 매크로 탐지를 위한 아키텍처는 Fig. 3과 같다.

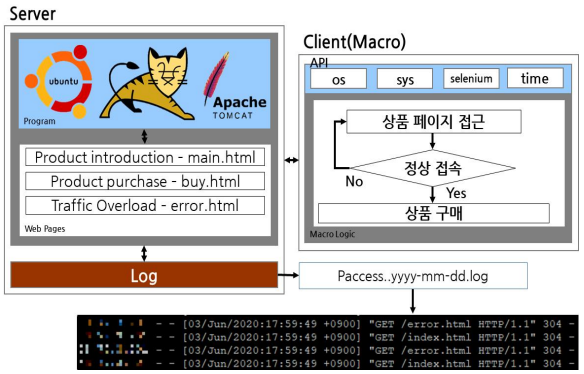


Fig. 3. Test System Architecture

로그 확인을 위한 웹 서버는 Ubuntu 18.04.4에 아파치 톰캣을 이용하여 웹 서비스를 구축하고, 상품 메인 페이지인 index.html, 트래픽 과부하로 인한 접근 오류 현상을 재현하기 위해 199/200의 확률로 출력하는 error.html 페이지와 1/200의 확률로 출력되는 정상 상품 구매 페이지 buy.html로 개발하였다.

클라이언트에서 동작하는 상품 자동 구매 매크로는 자동화된 매크로의 동작을 시각적으로 보여주기 위해 Chrome driver[5]와 Selenium[6]을 사용한다. Selenium은 웹 페이지에서 Element를 xPath로 찾아 클릭하는 방식으로 동작하여 실제 사용자가 사용하는 방식과 가장 유사하게 웹 브라우저 자동화하기 위해 사용한다.

클라이언트에서 테스트 웹 서버를 대상으로 구매 매크로를 동작하면, 매크로는 첫 페이지(index.html)에 접근 후 id가 "btn11"인 버튼을 클릭하여, 마스크 구매 시도를 한다. 웹 서버는 트래픽 과다 현상을 구현하기 위해 1/200 확률로 정상 구매 페이지로 응답한다. 매크로는 트래픽 과다로 인하여 구매 페이지 접근 실패 시 첫 페이지(/index.html)로 돌아가 작업을 반복하다가 구매 성공 페이지가 출력되면, 구매 버튼을 눌러 자동으로 구매를 진행한다.

본 연구에서 제안하는 매크로와 정상 사용자를 구분하기 위한 수식은 다음 식(1)과 같다.

$$Normal = [0 \le Normal \le R_{hs} + (R_{hs} \times 0.5)] \quad \text{--- (1)}$$

-R_{hs}= 정상 사용자의 초당(seconds, s) 페이지 접근(hit, h) 횟수

매크로와 정상 사용자를 구분하는 단위 시간당 요청횟수 산정을 위해 총 100회의 접속 시도를 분석한 결과 총 걸린 시간은 32초로 초당 약 3.1회 방문하였다. 인터넷 환경과 사용자 간의 차이 등을 고려해 오차 범위를 50%로 설정하여 일반 사용자의 정상적인 접근 속도를 산출하고 정상적인 속도보다 빠른 접근을 매크로로 정의한다. 정상 사용자는 약 초당 3.1회 페이지 접근하며, 오차 범위 50%를

반영하여, 초당 4.65회 이상 페이지 접근을 요청하는 IP를 매크로로 분류한다.

$$Normal = [0 \le Normal \le 3.1 + (3.1 \times 0.5)]$$ ---(2)

$$Normal = [0 \le Normal \le 4.65]$$ ---(3)

매크로를 50회 실행하여 위에서 산출한 공식의 정확도는 Table. 2와 같이 2회를 제외하고는 제안하는 산출식의 범위 내로 확인된다.

Table 2. Result for macro program

hit	time	h/s	hit	time	h/s
178	10.8	16.5	144	10.3	14.0
430	25.4	17.0	156	10.9	14.3
545	31.4	17.4	58	4.8	12.1
75	5.5	13.6	332	22.8	14.6
9	2.0	4.6	227	20.9	10.9
309	19.0	16.3	120	8.9	13.5
93	6.7	14.0	222	15.6	14.2
73	5.4	13.6	200	14.4	13.9
436	27.1	16.1	227	16.3	14.0
167	11.2	15.0	21	2.5	8.4
248	15.9	15.6	123	9.6	12.8
97	6.7	14.5	155	11.4	13.7
115	8.0	14.4	190	11.4	16.7
58	4.5	12.9	330	18.8	17.6
56	4.5	12.4	70	4.8	14.6
41	3.7	11.2	244	14.7	16.7
41	3.6	11.5	532	30.2	17.6
382	24.9	15.3	214	12.8	16.7
543	35.2	15.4	37	3.1	11.9
58	4.7	12.3	511	29.4	17.4
297	20.3	14.7	221	13.5	16.4
92	7.7	11.9	5	1.3	3.8
19	2.3	8.3	167	10.7	15.7
268	18.2	14.8	37	3.6	10.3
283	19.2	14.8	140	9.3	15.1

산출 식에 맞지 않은 2회의 페이지 접근 횟수는 각각 9회, 5회로 매우 적은 횟수로 구매 페이지에 접근하는데 성공하였음을 알 수 있다. 공식의 정확도 산출을 위한 페이지 요청 횟수와 시간은 클라이언트인 매크로에서 측정하였기 때문에 프로그램 최초 실행 시에 존재하는 딜레이가 포함되어 초당 페이지 요청 횟수(h/s)에 큰 영향을 끼친 것으로 분석된다. 이와 같은 내용을 고려하면, 매크로 프로그램에 대한 False Positive는 Fig. 4와 같이 0~4%로 확인된다.

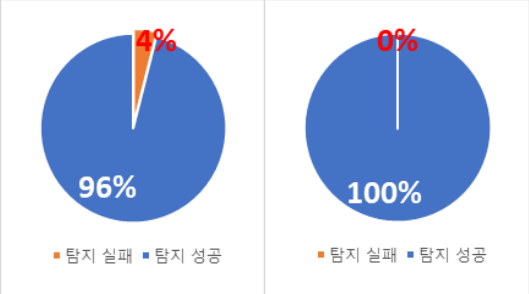


Fig. 4. Detection test result

실제 환경에서는 NAT환경에서 여러 유저가 동시에 접속하여 해당 사용자들이 초당 4.65회 이상 접근 하거나, 새로 고침(F5)을 과도하게 눌러 매크로로 탐지되는 등 다양한 변수가 존재할 수 있다.

본 논문에서는 매크로를 실시간으로 보다 쉽고 빠르게 탐지하고, 직관적인 결과 도출을 위해서 빅 데이터 분석에 사용하는 Splunk[7]를 이용하여 서버의 액세스 로그를 실시간으로 분석하고, 시각화 하도록 하였다.

실시간 요청 횟수를 차트와 표로 출력하기 위해 사용한 Splunk의 검색 설정은 Fig. 5와 같다

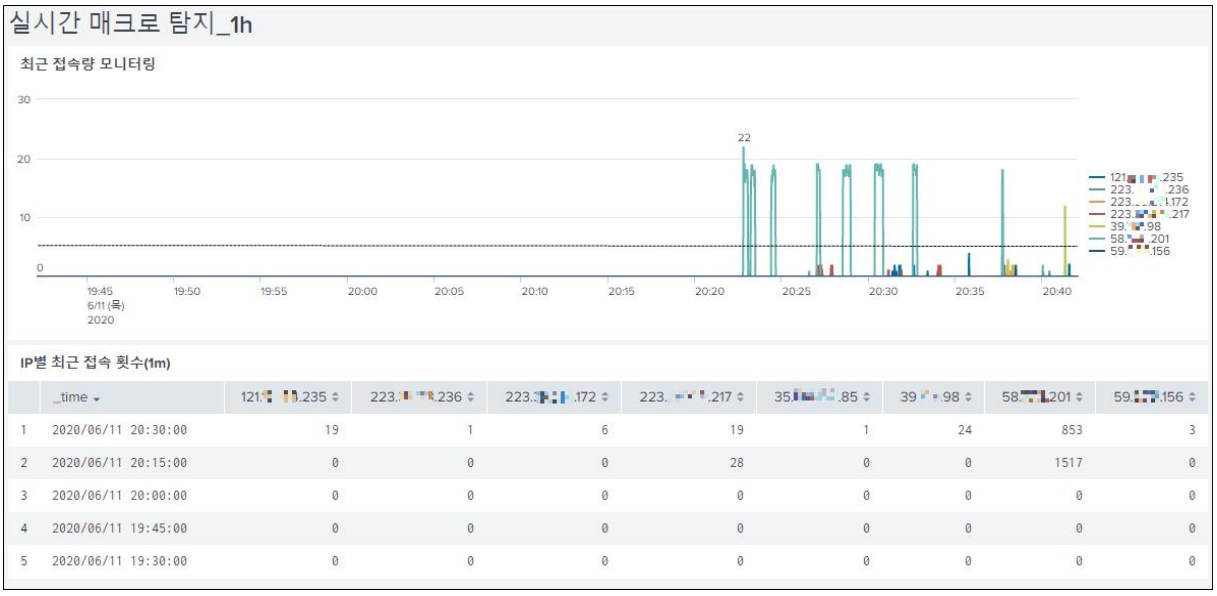
```
index="web_access_log" | timechart count
                                span=1s by clientip
```

Fig. 5. Splunk search filter

웹 서버에서 가져온 로그를 1초 간격의 시간과 Clientip을 각각의 행과 열로 하여 출력 하게 하고, 이를 시각화 도구를 이용해 차트로 변환한다. Fig. 6에 있는 차트에서 58.xxx.xxx.201과 39.xxx.xxx.98 IP에서 검은색 기준선(초당 5회)을 넘는 페이지 요청을 하였음을 확인할 수 있으며, 아래 표를 보면 223.xxx.xxx.236은 총 2370번의 페이지 요청을 하며 지속적으로 매크로를 사용한 접근을 시도하고 있으며, 39.xxx.xxx.98은 총 24번의 페이지 요청 횟수로 지속적인 접근 시도는 하지 않고 있음을 한눈에 확인 할 수 있다.

IV. Conclusions

본 논문에서는 코로나19에 따른 매크로가 사회적 이슈로 떠오르면서 마스크 구매 대란과 같은 매크로에 의한 피해 사례를 조사하고, 매크로에 대응하기 위하여 정상적인 사용자의 사용 패턴에 기반을 두어 매크로를 구분하기 위한 기준을 제시하고, 제시한 기준에 따라 매크로를 실시간으로 탐지할 수 있는 방법을 제안하였다. 그리고 실험을 통해 정상적인 사용자의 사용 패턴에 기반을 두고 매크로 판단의 기준을 설정하여 매크로를 탐지하는 것이 효율적이다는 것을 실증적으로 보여주었다. 본 연구를 바탕으로 실시간 매크로 탐지를 강화하기 위해서 더욱 다양한 기준 선정 방법과 실시간 탐지 시스템 개발을 지속적으로 연구할 예정이다.



	_time	121. .235	223. .236	223. .172	223. .217	35. .85	39 .98	58. .201	59. .156
1	2020/06/11 20:30:00	19	1	6	19	1	24	853	3
2	2020/06/11 20:15:00	0	0	0	28	0	0	1517	0
3	2020/06/11 20:00:00	0	0	0	0	0	0	0	0
4	2020/06/11 19:45:00	0	0	0	0	0	0	0	0
5	2020/06/11 19:30:00	0	0	0	0	0	0	0	0

Fig. 6. Visualization with Splunk

REFERENCES

[1] Why is there insufficient mask on the Internet...Use macros for monopoly, *https://www.seoul.co.kr/news/newsView.php?id=20200306009015*

[2] Illegal ticket made with macros, trades up to decuple, *https://newsis.com/view/?id=NISX20191114_0000829792*

[3] American teenage boy spreads macro 'bird bot' to buy Nintendo Switch, https://hypebeast.kr/2020/4/teen-reseller-creates-bird-bot-nintendo-switch

[4] BTS fan meeting ticket 90,000 won, Sold for 5.5 million won. Macro polarity, *https://news.joins.com/article/23511130*

[5] *https://chromedriver.chromium.org/*

[6] *https://www.selenium.dev/*

[7] *www.splunk.com*