오픈소스를 이용한 APT 그룹 판별 및 프로파일링 연구

Open Source using APT Group Profiling and Identification

저자
(Authors)

이현식, 문해은, 장경익, 성준영, 우상태
Hyunsik Lee, Heaeun Moon, Gyeongik Jang, Joonyoung Sung, Sangtae Woo

출처
(Source)

한국정보과학회 학술발표논문집 , 2018.6, 1198-1200 (3 pages)

발행처
(Publisher)

한국정보과학회
KOREA INFORMATION SCIENCE SOCIETY

URL

http://www.dbpia.co.kr/Article/NODE07503292

APA Style

이현식, 문해은, 장경익, 성준영, 우상태 (2018). 오픈소스를 이용한 APT 그룹 판별 및 프로파일링 연구. 한국정보과학회 학술발표논문집, 1198-1200.

이용정보
(Accessed)

국민대학교
121.139.87.***
2018/08/12 18:09 (KST)

저작권 안내

DBpia에서 제공되는 모든 저작물의 저작권은 원저작자에게 있으며, 누리미디어는 각 저작물의 내용을 보증하거나 책임을 지지 않습니다. 그리고 DBpia에서 제공되는 저작물은 DBpia와 구독계약을 체결한 기관소속 이용자 혹은 해당 저작물의 개별 구매자가 비영리적으로만 이용할 수 있습니다. 그러므로 이에 위반하여 DBpia에서 제공되는 저작물을 복제, 전송 등의 방법으로 무단 이용하는 경우 관련 법령에 따라 민, 형사상의 책임을 질 수 있습니다.

Copyright Information

Copyright of all literary works provided by DBpia belongs to the copyright holder(s)and Nurimedia does not guarantee contents of the literary work or assume responsibility for the same. In addition, the literary works provided by DBpia may only be used by the users affiliated to the institutions which executed a subscription agreement with DBpia or the individual purchasers of the literary work(s)for non-commercial purposes. Therefore, any person who illegally uses the literary works provided by DBpia by means of reproduction or transmission shall assume civil and criminal responsibility according to applicable laws and regulations.

오픈소스를 이용한 APT 그룹 판별 및 프로파일링 연구

이 현 식, 문 해 은, 장 경 익, 성 준 영, 우 상 태
엔에스에이치씨
hslee@nshc.net, hemoon@nshc.net, kijang@nshc.net, jysung@nshc.net,
stwoo@nshc.net

Open Source using APT Group Profiling and Identification

Hyunsik Lee, Heaeun Moon, Gyeongik Jang, Joonyoung Sung, Sangtae Woo
NSHC

요 약

최근 악성코드를 이용한 사이버 공격의 수와 피해가 늘어남에 따라 수년간 다양한 방식의 악성코드 탐지 기법이 연구됐다. 최근에는 악성코드 탐지뿐만 아니라 프로파일링 기술을 이용해 공격 그룹 판별 및 추적에 대한 연구도 함께 진행되고 있다. 본 논문은 악성코드 탐지 기법이 아닌 악성코드에서 추출할 수 있는 정보를 이용해 공격 그룹(이하 APT 그룹)을 그룹화하고 그룹화한 APT 그룹을 프로파일링하는 기술에 대한 연구를 진행하였다. 먼저 해당 연구는 탐지 기술을 기반으로 진행되며, 탐지 기술로는 야라(YARA) 기술을 이용하며, 탐지 대상으로는 공격 그룹이 이용하는 원격 관리 도구(Remote Access Tool)를 대상으로 연구를 진행하였다. 이 외에 공개된 APT 그룹 룰셋(Ruleset)과 자동화 추출 룰셋을 사용해 공격 그룹 탐지 방법과 공격 판별에 사용 가능성에 대해 다룬다.

1. 서 론

최근 악성코드를 이용한 사이버 공격의 수와 피해가 늘어남에 따라 다양한 방식의 탐지 기법들이 연구되고 있다. 최근에는 악성코드 탐지뿐만 아니라 프로파일링 기술을 이용한 공격 그룹 판별 및 추적에 대한 연구도 함께 진행되고 있다. 본 논문은 악성코드 탐지 기법이 아닌 악성코드에서 추출할 수 있는 정보를 이용해 공격 그룹을 그룹화하고 그룹화한 APT 그룹을 프로파일링 하는 기술에 대해 연구를 진행하였다. 해당 연구는 탐지 기술을 기반으로 진행되며, 탐지 기술로는 야라(YARA) 기술을 이용하며, 탐지 대상으로는 공격 그룹이 주로 이용하는 원격 관리 도구(Remote Access Tool)에서 생성된 모듈을 대상으로 연구를 진행했다. 이 외 자동화 추출 룰셋을 이용해 공격 그룹 탐지 방법과 공격 판별에 사용 가능성에 대해 다룬다.

1.1 야라(YARA)

야라는 문자열이나 코드 패턴을 기반으로 파일을 분류할 수 있는 도구로 C와 파이썬 언어들과 비슷한 문법을 사용한다. 때문에 룰셋(Ruleset) 작성 시 매우 쉽다. 야라는 윈도우, 리눅스, macOS 등 모든 시스템을 지원하며 직접 소스 코드를 컴파일하거나 실행 파일을 사용해 설치할 수 있다. 파이썬에서 모듈을 지원하기 때문에 파이썬을 사용하면 고도화도 가능하다. 야라 룰셋은 '설명', '룰셋 이름', '데이터', '참, 거짓 판단 조건'으로 나눠서 문법이 사용되며, 파일 분류 시 단순한 문자열이나 코드 패턴을 비교할 뿐 아니라 대상 파일의 엔트리 포인트(Entry Point) 값을 지정하거나 파일 오프셋(File Offset), 가상 메모리 주소(Virtual Memory Address)를 지정할 수 있다. 정규표현식도 함께 사용하면 더욱 효율적인 패턴 매칭이 가능하다.

1.2 APT 공격 그룹

APT란 지능형 지속 위협(Advanced Persistent Threat)란 기업 조직 등 특정 대상을 선정하고 내부의 취약한 시스템을 이용해 침투한 다음 오랜 시간에 걸쳐 다양한 공격 기법을 활용해 피해자가 인지하지 못하는 방식으로 공격하는 방법을 말한다. 특히 APT는 보안업체에서 보안패치가 아직 나오지 않은 보안 취약점을 이용하기 때문에 사전 대응이 매우 중요하다. APT 공격은 '시도-성공-확산-유출'과 같이 네 단계로 공격이 이루어진다.

첫 번째 '시도' 단계에서는 먼저 외부 공격자가 사회공학적 방법 등을 비롯해 다양한 기법을 이용해 내부 사용자를 대상으로 공격을 시도한다.

두 번째 '성공' 단계는 외부 공격자가 내부 사용자나 시스템을 확보하여 내부 침투에 성공하였을 때를 의미한다.

세 번째 '확산' 단계는 확보한 시스템에 악성 행위를 실행하는 악성코드를 감염시키게 되는 단계를 의미한다.

네 번째 '유출' 단계는 감염시킨 악성코드를 통해 내부 시스템에서 유용한 데이터를 외부로 가져오는 단계로 유출뿐만 아니라 PC의 정상 사용을 못 하도록 파괴하기도 한다.

국민대학교 | IP: 121.139.87.*** | Accessed 2018/08/12 18:09(KST)

1.3 오픈소스 소프트웨어

오픈소스 소프트웨어란 소스 코드가 공개된 프로그램을 의미한다. 대부분의 오픈 소스 소프트웨어는 무료로 사용할 수 있기 때문에 프리웨어(Freeware)와 헷갈리는 경우가 많지만, 프리웨어는 무료로 사용 가능한 프로그램을 의미하며 소스 코드가 공개되는 오픈 소스 소프트웨어와는 다른 개념이다.

본 연구에 사용한 오픈소스 소프트웨어로는 야라룰셋을 추출하는 yarGen[1]와 자동화 분석환경을 제공하는 쿡쿠 샌드박스[2]를 사용했다.

yarGen은 정상 파일(Goodware)에 존재하는 스트링은 완전히 제거하진 못하지만, 정상 파일에서는 볼 수 없는 문자열들을 추출하는데 많은 도움을 주는 소프트웨어다. 또한, 정상 파일에서 사용되는 문자열 정보를 "good-strings-office.db"와 "good-opcodes-office.db"라는 데이터베이스 파일을 통해 다른 룰셋 생성 소프트웨어 보다 높은 정확성을 보인다.

쿡쿠 샌드박스 는 악성코드를 자동으로 분석하는 샌드박스 소프트웨어다. 쿡쿠샌드박스는 가상환경에서 샘플을 실행시키고 가상환경에서 파일 실행전과 실행후의 변화를 비교하여 결과를 보여줌으로, 많은 샘플들을 분석하기 위해서는 필요한 소프트웨어다. 하지만, 샘플들의 기술적인 코드 분석은 되지 않기 때문에 샌드박스 결과를 보고 판단하여 세세한 분석 작업도 필요하다.

2. 공격 그룹 판별 데이터

2.1 탐지 활용 방법

샘플을 탐지하기 위해서는 야라를 사용한다. 야라는 문자열과 같은 데이터를 이용해 탐지도 가능하지만, 코드 시그니처를 이용해서도 탐지할 수 있다. 보통 공격 그룹을 탐지 하기 위해서는 해당 샘플들이 남기는 흔적(feature)를 사용해 추적하게 된다.

탐지에 활용할 수 있는 데이터로는 샘플에서 데이터를 숨기기 위해 사용하는 디코딩 코드나 정상 파일에서 확인할 수 없는 문자열 데이터들을 사용할 수 있다. RAT와 같은 경우는 보통 키로깅 기능이 있으며 방향키, 기능키를 표현하기 위한 문자열들이 존재하며, 통신을 위해 사용하는 코드 패턴들을 그룹화할 때 사용할 수 있는 탐지 데이터로 사용할 수 있다.

2.2 공격 그룹 판별 데이터

데이터 중에 키로깅과 같은 기능을 가지고 있는 RAT와 같은 경우는 다음과 같은 문자열 데이터를 가지고 있는 것을 확인할 수 있다. 다음은 다크코멧 RAT에 존재하는 키로깅 관련 문자열이다. [3]

[ESC], [<-], [NUM_LOCK], [DEL], [INS], [SNAPSHOT], [LEFT], [RIGHT], [DOWN], [UP]

표 1. 키로깅 문자열 데이터

이러한 문자열 데이터를 사용하게 되면, 다른 RAT에서도 키로깅 기능을 사용할 수 있기 때문에 다른 데이터들도 함께 추가하여 그룹화를 나누는 것이 좀 더 확실한 그룹화에 도움이 된다. 그러한 데이터로는 다음과 같이 코드 시그니처들을 사용할 수 있다.

50 8D 45 8C 50 6A 00 6A 00 6A 04 6A 00 6A 00 6A 00 8B 45 F8 E8 F2 19 F9 FF 50 8B 45 FC E8 E9 19 F9 FF 50 E8 A7 39 F9 FF

표 2. 그룹화 코드 시그니처

해당 코드는 방화벽 우회를 위해 iexplore.exe를 생성한 후 해당 프로세스에 스레드를 생성하여 악성 행위를 하는 코드다. 이러한 코드 패턴들과 함께 탐지하게 되면 좀 더 확실한 그룹화가 가능하다.

2.3 실제 탐지 활용 데이터

그룹화에 사용하는 데이터는 다음과 같은 형태를 보인다. 우선 키로깅의 경우는 아래와 같이 입력 시 분기 코드가 실행되면서 문자열 데이터를 기록하게 된다.

```
loc_481BF2:                             ; CODE XREF: sub_4818F8+B6↑j
                                        ; DATA XREF: sub_4818F8:off_481985↑o
        lea     eax, [ebp+var_4] ; jumptable 004819AE case 44
        mov     edx, offset aSnapshot ; "[SNAPSHOT]"
        call    sub_4055C8
        jmp     loc_481D10
;  ---------------------------------------------------------------------------

loc_481C04:                             ; CODE XREF: sub_4818F8+B6↑j
                                        ; DATA XREF: sub_4818F8:off_481985↑o
        lea     eax, [ebp+var_4] ; jumptable 004819AE case 37
        mov     edx, offset aLeft_0 ; "[LEFT]"
        call    sub_4055C8
        jmp     loc_481D10
;  ---------------------------------------------------------------------------

loc_481C16:                             ; CODE XREF: sub_4818F8+B6↑j
                                        ; DATA XREF: sub_4818F8:off_481985↑o
        lea     eax, [ebp+var_4] ; jumptable 004819AE case 39
        mov     edx, offset aRight ; "[RIGHT]"
        call    sub_4055C8
        jmp     loc_481D10
;  ---------------------------------------------------------------------------

loc_481C28:                             ; CODE XREF: sub_4818F8+B6↑j
                                        ; DATA XREF: sub_4818F8:off_481985↑o
        lea     eax, [ebp+var_4] ; jumptable 004819AE case 40
        mov     edx, offset aDown ; "[DOWN]"
        call    sub_4055C8
        jmp     loc_481D10
;  ---------------------------------------------------------------------------

loc_481C3A:                             ; CODE XREF: sub_4818F8+B6↑j
                                        ; DATA XREF: sub_4818F8:off_481985↑o
        lea     eax, [ebp+var_4] ; jumptable 004819AE case 38
        mov     edx, offset aUp ; "[UP]"
        call    sub_4055C8
        jmp     loc_481D10
```

그림 1 키로깅 코드

그룹화를 위해 해당 코드 패턴을 룰셋으로 사용할 수도 있지만, 매우 간단한 코드이고 계속 반복되기 때문에 코드 패턴보단, 코드에서 사용하는 문자열 데이터를 탐지 룰셋으로 사용할 수 있다. 또한, 방화벽 우회에서 사용했던 코드는 다음과 같은 형태를 가지고 있는 것을 확인할 수 있다.

그림 2 인젝션 코드

국민대학교 | IP: 121.139.87.*** | Accessed 2018/08/12 18:09(KST)

해당 코드 같은 경우는 C:₩Program Files₩Internet Explorer₩iexplore.exe라는 문자열도 존재하지만, 앞서 호출되는 코드 패턴들을 이용한다면 다크코멧 RAT를 분류할 때 중요한 데이터로 충분히 사용 가능하다.

그뿐만 아니라 다음과 같은 디코딩 코드도 그룹화를 나눌 때 유용하게 사용할 수 있다. 해당 코드를 보면 v5, v6, v12, v11 네 개 변수의 하드 코딩된 값을 사용해 디코딩 과정을 진행하는 것을 볼 수 있으며, 이때 4개에 키에 따라 인코딩되는 데이터가 달라지는 것을 예상할 수 있다. 이는 곳 특정 공격 그룹에서 해당 디코딩 코드를 주로 이용한다면, 아래 코드만으로도 그룹화하여 샘플을 정리할 수 있기 때문에 유용한 데이터가 된다. 코드는 다음과 같다.

```
int __usercall sub_402C98@<eax>(_BYTE *a1@<edx>, int a2@<ecx>, int a3)
{
  _BYTE *v3; // edi@1
  int v4; // esi@1
  unsigned int v5; // edx@1
  unsigned int v6; // ecx@1
  int v7; // esi@2
  unsigned int v8; // ST0C_4@3
  int result; // eax@3
  bool v10; // zf@3
  signed int v11; // [sp+Ch] [bp-8h]@1
  unsigned int v12; // [sp+10h] [bp-4h]@1

  v3 = a1;
  v4 = a2;
  v5 = 0x1D91E44;
  v6 = 0x40E0E1E;
  v12 = 0x1D91E44;
  v11 = 0x40E0E1E;
  if ( a3 > 0 )
  {
    v7 = v4 - (_DWORD)v3;
    do
    {
      v8 = v5 >> 8;
      *v3 = v11 ^ v12 & BYTE1(v5) ^ v3[v7] ^ (v12 >> 16) & BYTE3(v12) ^ BYTE1(v6) & (v6 >> 16) & BYTE3(v6);
      result = (v6 >> 8) | (v12 << 24);
      v5 = (v5 >> 8) | ((16 * v11 ^ (v11 ^ 2 * (v11 ^ 4 * v11)) & 0xFFFFFFF0) << 20);
      v6 = result;
      ++v3;
      v12 = v8 | ((16 * v11 ^ (v11 ^ 2 * (v11 ^ 4 * v11)) & 0xFFFFFFF0) << 20);
      v10 = a3-- == 1;
      v11 = result;
    }
    while ( !v10 );
  }
  return result;
}
```

그림 3 디코딩 코드

3. 공격 그룹 탐지 방법

3.1 탐지 방법

다음은 탐지한 샘플을 분류하는 전체적인 구상도다.

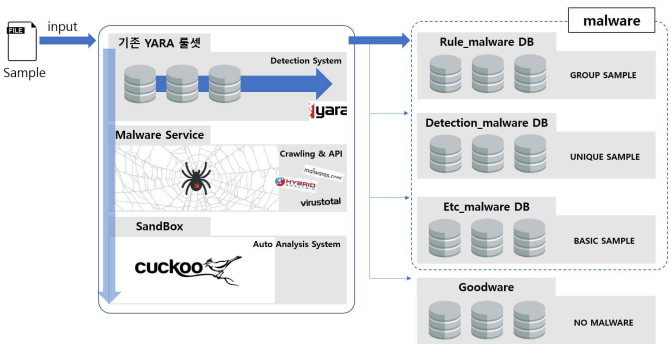

그림 4 공격 그룹 탐지 방법 구상도

크게 세 가지 탐지 방법을 이용하게 된다. 우선 기존에 분석된 데이터를 기반으로 하는 탐지 방법과 다양한 악성코드 서비스에서 나온 데이터를 이용한 탐지 기법, 그리고 자동화 분석 시스템을 이용한 탐지 방법을 이용한다.

3.2 데이터를 이용한 탐지 방법

데이터를 이용한 탐지 기법은 기존에 분석된 악성코드에서 그룹 판별에 사용할 수 있는 데이터를 사용해서 탐지하는 방법으로 야라 룰셋을 사용해 데이터를 분류하는 작업을 실행한다. 해당 탐지 기법을 사용하여 공격에 이용하는 악성코드를 그룹화할 때 사용할 수 있다.

3.3 데이터 수집을 이용한 탐지 방법

다음 기법은 이미 분석돼 악성 행위 유무가 확인된 데이터를 이용하는 방법으로 이미 사전에 나온 데이터 정보를 사용해 해당 샘플의 악성코드 유무를 확인하는 단계다. 해당 단계는 그룹화 샘플 대상엔 포함되진 않지만, 악성코드 데이터베이스에 관리함으로 추후 그룹화 대상 파일로 사용할 수 있다.

3.4 자동화 분석을 이용한 탐지 방법

자동화 분석을 이용한 방법은 앞서 확인한 두 가지 탐지 기법 외에 파일을 탐지하려는 방법으로 자동화 분석은 가상환경에서 실행 전과 후의 변화 및 통신 데이터 등을 사용해 악성 행위 유무를 확인하기 때문에 앞서 확인하는 기법 두 가지 외의 악성파일일 경우 탐지가 가능하며 정상파일 유무를 가릴 때 사용할 수 있다.

4. 결론 및 향후 연구

본 연구는 공개된 오픈소스 소프트웨어를 사용해 샘플들을 프로파일링하는 초기 방법인 분류작업에 대한 연구를 진행하고 있다. 이러한 분류 작업 외에도 수많은 데이터를 통해 그룹화가 가능한 만큼 다음에는 프로파일링 가능한 데이터 추출과 세분화된 그룹화 작업이 필요하다.

5. 감사의 글

이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임(No.2016-0-00081, 악성코드 全 생명주기 통합 프로파일링 및 공격그룹 식별 기술 개발)

6. 참고 문헌

[1] yarGen : https://github.com/Neo23x0/yarGen
[2] Cuckoo sandbox :
https://github.com/cuckoosandbox/cuckoo
[3] http://ieeexplore.ieee.org/document/5403021/

국민대학교 | IP: 121.139.87.*** | Accessed 2018/08/12 18:09(KST)