봇넷 트래픽 특성 분석: 사례 연구[12]

김유승*, 최현상, 김인환, 권종훈, 이희조
고려대학교 컴퓨터• 통신공학부
e-mail : * {corekey}@korea.ac.kr

An Analysis on Botnet Traffic

Yu-Seung Kim, Hyun-Sang Choi, In-Hwan Kim, Jong-Hun Kwon, Heejo Lee
*Div. of Computer & Communication Engineering, Korea University

요 약

최근 DDoS 공격의 의도와 공격형태가 날로 다양해지고 그 피해규모가 심각해짐에 따라 DDoS 를 탐지하고 이를 방어하기 위한 연구들이 활발하게 진행되고 있다. 한편, 봇넷은 이러한 DDoS 공격을 수행하는 도구로서 여러 연구기관들에 의해 새로운 위협적인 요소로 보고되고 있다. 본 연구에서는 보안상 상대적으로 취약하다고 알려져 있는 교내망에서 실제 봇넷 트래픽을 찾아내고 분석하였다. 이를 통해 봇넷의 특성을 밝혀내고 이와 관련된 연구의 기초자료로 사용될 수 있을 것이다.

1. 서론

20 세기 말에 처음 출현한 이후로 DDoS 공격은 점차 공격 의도의 다양화, 피해 규모의 확대화, 수행 기술의 고도화와 같은 특성을 보이면서 정부와 민간을 가리지 않고 국가를 초월하여 주요한 네트워크 서버들을 마비시키고 있다. 특히, 봇넷(Botnet)은 악의적인 의도를 가진 소프트웨어에 감염된 불특정 다수의 PC 들로 이루어진 네트워크로 봇 마스터(Bot master)에 의해 원격 조종되어 DDoS 공격을 수행하는 주요한 도구로 이용되고 있다.

Arbor Networks 가 2007 년 발표한 조사에 따르면 봇넷이 사이버 상의 가장 위협적인 요소임을 알리고 있다. 이는 최근 발생하는 DDoS 공격의 대부분이 봇넷에 의해 이루어지는 것과 무관하지 않다. 시만텍의 발표에서도 2006 년 하반기에만 600 만대의 새로운 봇이 생성된 것으로 나타나고 있으며 이는 같은 해 상반기에 비해 약 26%나 증가한 수치이다. 2007 년에는 하루 평균 5 만 2771 대의 PC 가 새로이 봇에 감염되고 있는 것으로 알려졌다.

이러한 일련의 흐름들은 봇넷에 대한 심층적인 연구를 절실히 요구하고 있다. 본 논문에서는 보안상 상대적으로 취약하다고 알려져 있는 교내망에서 봇넷의 C&C 서버로 이용되고 있는 PC 에서 트래픽들을 수집하고 분석하였다. 이러한 분석을 통한 연구는 봇넷의 특성을 밝히는데 활용될 수 있다.

본 논문의 2 장에서는 교내망에서 봇넷 트래픽을 수집하고 이를 분석하기 위한 방법을 설명한다. 3 장에서는 수집된 트래픽 전체를 대상으로 다양한 통계를 통해 전체적인 특성을 파악하도록 한다. 4 장에서는 수집된 트래픽 중에서 실제 봇넷으로 의심되는 패킷들을 추적하여 세부적인 동작 방식을 살펴보았다. 마지막으로 5 장에서는 결론과 함께 향후 전망에 대하여 언급하기로 한다.

2. 측정 방법

봇넷 트래픽을 분석하기 위해 우선 교내 네트워크에서 봇넷의 C&C 서버로 이용되고 있는 PC 에서 트래픽을 수집하였다. 해당 호스트는 교내 한 연구실의 서버로 x.y.216.46 의 IP 를 사용하고 있는데 여기서는 botcnc 서버라고 가칭하도록 한다.

이렇게 선정된 botcnc 서버에서 트래픽을 수집하기 위해 공개 소프트웨어인 Wireshark 를 사용하였다. 수집된 패킷을 분석하기 위해서 먼저 Wireshark 소프트웨어에서 기본적으로 제공하는 패킷 분석 tool 을 사용하였고, 비정상으로 과도한 트래픽이 확인되는 구간의 패킷 내용(packet payload)을 분석하여 일반적인 봇넷에서 보여지는 트래픽과 비교 및 대조해 보았다.

3. 트래픽 특성 분석

botcnc 서버에서 다음과 같은 시간대에 해당 호스트에서 수집된 패킷들을 대상으로 한다.

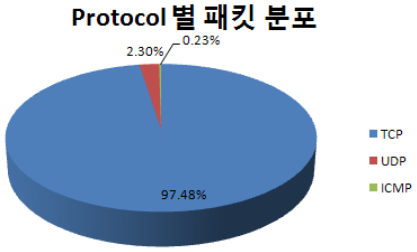
시작 시간 : 2008/5/27 10:49:04
종료 시간 : 2008/5/27 11:36:38
총 소요 시간 : 47 분 33 초 (2853 초)

먼저 이들 패킷의 전반적인 특성은 다음과 같다.

총 패킷 개수 : 25,094 개
평균 PPS (초당 전송 패킷 수) : 8.794 개/sec
평균 패킷 크기 : 199.210 Bytes
총 패킷 크기의 합 : 4.767 Mbytes
평균 초당 전송 바이트 : 1751.92 Bytes/sec
평균 초당 전송 비트 : 13.69 Kbit/sec

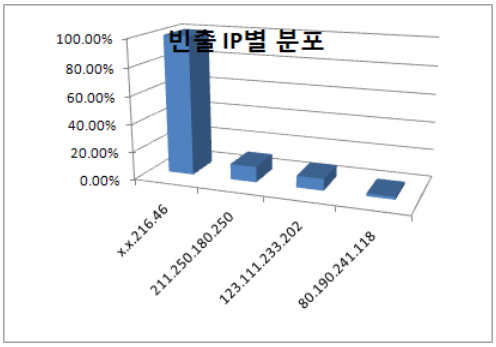
여기에서 패킷들의 평균 크기는 약 200 Bytes 로 나타났는데 분포를 세밀히 살펴보면 40~159 Bytes 사이의 패킷이 거의 90% 가까이 차지하고 있었다.

총 패킷들을 프로토콜 별로 분류한 결과 아래와 같이 거의 대부분의 트래픽이 TCP 로 이루어져 있음을 알 수 있다. 그리고 TCP 중의 절반 정도는 HTTP 로 이루어져 있었다.



(그림 1) Protocol 별 패킷 분포

전체 트래픽으로부터 botcnc 서버가 어떠한 대상과 주로 통신하는지 알아보기 위해 패킷의 송신지와 수신지 주소를 분석해 보았다. 송신지나 수신지 전체를 통틀어 가장 빈번하게 출현하는 IP 별로 정리하였을 때 (그림 2)과 같이 나타났다.

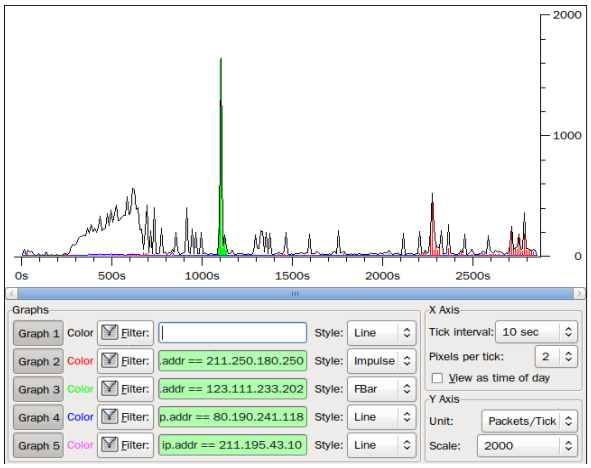


(그림 2) 빈출 IP 별 분포

전체 트래픽은 송신지나 수신지에 x.y.216.46 의 IP 가 존재하는 경우를 기준으로 수집하였기 때문에 당연히 이 IP 는 100%의 출현 빈도를 보인다. 이어서 211.250.180.250, 123.111.233.202, 80.190.241.118 의 IP 순으로 빈번하게 사용되었다. 한편, 각 패킷의 목적지 IP 별로 출현 빈도를 조사하였을 때에도 앞에서와 동일한 순서의 결과를 얻을 수 있었다.

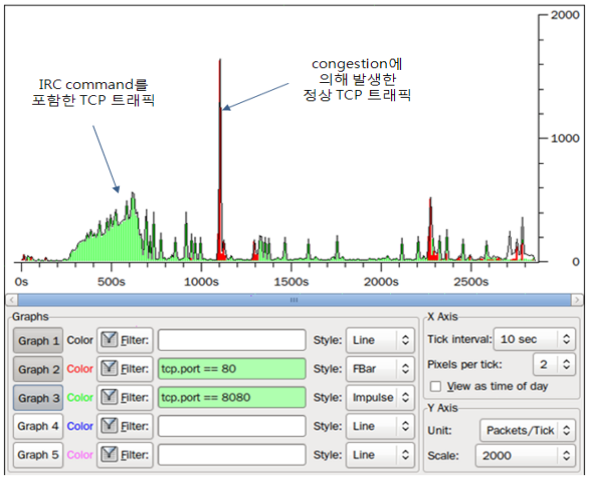
이번에는 앞에서 분석한 대로 botcnc 서버와 빈번하게 통신하는 IP 에 유의하여 세부적인 트래픽 특성을 살펴보았다. 먼저, 시간에 따라 PPS(Packets per Second)를 아래의 (그림 3)와 같이 표현하였다.

그림에서 검은 색 선은 각 시점에서 전체 트래픽의 PPS 를 의미하며 그 외 색깔로 표시된 영역은 각 시점에서 그림 하단에 표시된 색깔의 IP 가 송신지나 수신지 주소로 사용된 패킷의 PPS 를 나타낸다. 여기에서 패킷 수집 시작 이후 1100 초 부근의 시점에서 botcnc 서버는 123.111.233.202 의 IP 와 갑자기 많은 수의 패킷을 주고 받는 것을 볼 수 있다. 반면, 2300 초와 2700~2800 초가 지난 구간에서는 문제의 호스트가 211.250.180.250 의 IP 를 가진 외부 호스트와 많은 수의 패킷을 주고 받았다.



(그림 3) 빈출 IP 에 따른 PPS 분포

한편, 300 초에서 1000 초 이후의 구간에서 대량의 트래픽이 무색으로 표시되는데, 이것은 해당 구간에서 botcnc 서버가 특정한 IP 를 가진 외부의 호스트와 1:1 통신하지 않고 여러 대의 호스트들과 N:1 의 통신 형태를 가지기 때문으로 해석할 수 있다. 이 구간에서 나타나는 패킷의 내용을 통해 이들은 주로 80 및 8080 포트를 사용하는 TCP 패킷들로 이루어져 있음을 알 수 있었다. (그림 4)에서 80 포트와 8080 포트를 사용하는 TCP 패킷의 트래픽을 나타내었다.



(그림 4) 포트별 PPS 분포

앞에서 살펴본 1100 초 부근의 구간에서 211.250.180.250 의 IP 와 통신하는 영역과 80 포트를 사용하여 통신하는 구간이 겹치고 있다. 이 구간의 패킷을 살펴보면 botcnc 서버가 211.250.180.250 의 IP 를 가진 호스트와 정상적인 html 메시지를 주고 받는 것으로 보인다. 패킷의 수는 상당히 높게 나타나지만 실제 전송된 bytes 로는 크지 않은데 이것은 이 때 두 호스트 간에 congestion 이 발생하여 window size 가 아주 작은 TCP 패킷들과 그에 따른 ACK 메시지 들이 관찰되었다.

전반적으로 전 영역에 걸쳐 많은 양의 트래픽을 유발하는 부분은 8080 포트를 사용하는 TCP 패킷들이다. 이들은 잘 살펴보면 내부 메시지에 IRC 명령을

상당수 포함하고 있어 botnet 의 동작이 의심된다. 다음 장에서 이 IRC 명령 메시지들을 더 세밀히 추적해 보도록 한다.

4. 봇넷 트래픽 추적

8080 포트는 본래 HTTP 프로토콜이 사용하는 포트이나 여기서는 교내망 외부에 존재하는 봇마스터가 교내망의 botcnc 서버를 봇 C&C 서버로 삼아 IRC 프로토콜로 명령을 내리는데 사용하고 있었다.

수집된 트래픽으로부터 발견된 IRC 명령들 중에서 빈번하게 나타나는 것들을 <표 1>에 정리하였다.

<표 1> 발견된 IRC 명령 list

명령	매개변수	의미
PASS	<password>	접속암호 설정
NICK	<nickname>[<hopcount>]	대화명 지정
USER	<username><hostname> <servername><realname>	사용자 이름 지정
QUIT	[<Quit message)]	클라이언트 접속 완료
ADMIN	[<server>]	서버의 관리자 확인
JOIN	<channel>{,<channel>} [<key>{,<key>}]	특정 채널에 합류
LIST	[<channel>{,<channel>} [<server>]]	채널의 토픽들을 나열
PART	<channel>{.<channel>}	해당 채널에서 벗어남
MODE	<channel|nickname>[[+|-]modechars [parameters]]	채널의 운영자나 유저에게 속성을 지정함
NAMES	[<channel>{,<channel>}]	채널의 대화명들을 나열

실제로 확인된 몇 가지 IRC 명령 들을 아래에 나열하였다.

서버 상태 메시지

```
:MyIRCD 001 A_YmFkXzg5ZDhkODU_C :Welcome to the MyNet Internet Relay Chat Network A_YmFkXzg5ZDhkODU_C
:MyIRCD 002 A_YmFkXzg5ZDhkODU_C :Your host is MyIRCD[0.0.0.0/8080], running version hybrid-7.2.3
:MyIRCD 003 A_YmFkXzg5ZDhkODU_C :This server was created Mar 9 2006 at 00:31:58
:MyIRCD 004 A_YmFkXzg5ZDhkODU_C MyIRCD hybrid-7.2.3 CDGabcdfgiklnorsuwxyz biklmnopstveI bkloveI
```

'MyNet Internet Relay Chat' 이라는 채널을 쓰고 있고 'version hybrid-7.2.3'의 버전인 것을 알 수 있다.

```
:MyIRCD 255 A_YmFkXzg5ZDhkODU_C :I have 21 clients and 0 servers
:MyIRCD 265 A_YmFkXzg5ZDhkODU_C :Current local users: 21 Max: 68
:MyIRCD 266 A_YmFkXzg5ZDhkODU_C :Current global users: 21 Max: 68
:MyIRCD 250 A_YmFkXzg5ZDhkODU_C :Highest connection count: 68 (68 clients) (178 connections received)
:MyIRCD 422 A_YmFkXzg5ZDhkODU_C :MOTD File is missing
```

연속된 패킷에서 이 채팅 서버에 접속한 호스트들에 대한 정보를 보여준다. 최대 68 개의 클라이언트를 수용할 수 있는 이 서버는 현재 21 개의 호스트가 접속해 있는 것을 알 수 있다.

접속 시도

```
PASS iloveyou!!!
NICK A_YmFkXzg5ZDhkODU_C
USER nobody unknown unknown :noname
```

주기적으로 호스트에서 IRC 서버로 메시지를 보내 접속을 시도 하는 것을 볼 수 있다. 이때 동일한 PASS 를 이용하고 랜덤으로 생성된 NICK 의 정보만 다르게 하여 접속하는 것을 볼 수 있다.

```
:A_YmFkXzg5ZDhkODU_C!nobody@218.238.105.194 JOIN :#armyclass
```

접속이 성공하게 되면 위와 같은 메시지를 botcnc 서버에서 봇마스터로 보내게 된다.

PING, PONG 메시지

```
PING :MyIRCD
PONG MyIRCD
```

PING, PONG 메시지를 통해서 botcnc 서버는 봇마스터가 계속 접속하고 있는지를 확인한다. 약 205 초 마다 1 번씩 위와 같은 메시지로 호스트와 서버 간에 통신이 이루어짐을 확인하였다. PING, PONG 각각 패킷의 경우 1108 개의 메시지가 112 개의 호스트에서 보내 온 것을 확인 하였고 각 호스트당 약 10 개의 PING, PONG 메시지를 보낸 것을 알 수 있다.

관리자 명령

ADMIN, QUIT, PART 등과 같은 채널 관리자가 사용하는 메시지가 한 IP 에서만 만들어지는 것을 볼 수 있었다. 이것으로 이 IP 는 봇마스터일 것이라고 생각할 수 있다.

접속 성공

```
Botmaster -> C&C
PASS iloveyou!!!
NICK test
USER kkk "kkk.co.kr" "x.y.216.46" :test
USERHOST test

C&C -> Botmaster
:MyIRCD NOTICE AUTH :*** Looking up your hostname...
```

비밀번호, 닉네임 등을 정하고 kkk.co.kr 로 불리는 서버에 test 로 접속을 시도하는 모습이다.

```
Botmaster -> C&C
LIST

C&C -> Botmaster
:MyIRCD 321 test Channel :Users Name
:MyIRCD 322 test #mndjob 1 :
:MyIRCD 322 test #111 86 :
:MyIRCD 322 test #armyclass 3 :
:MyIRCD 322 test #epc 2 :
:MyIRCD 322 test #atc 4 :
:MyIRCD 323 test :End of /LIST
```

접속 성공 후 LIST 명령을 통해서 현재 채팅서버에 열린 채널들과 접속한 호스트 수를 보여준다

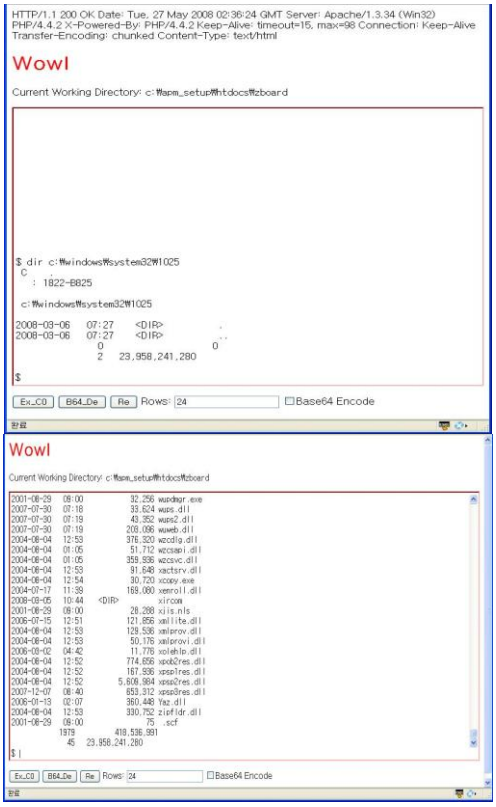
```
Botmaster -> C&C
JOIN #mndjob
MODE #mndjob

C&C -> Botmaster
:test!kkk@211.250.180.250 JOIN :#mndjob
:MyIRCD 353 test = #mndjob :test @A_udq8vL_BrKDEyNS4xMzEuMTc5KS1ENDA1N0ExMi12MS4y
:MyIRCD 366 test #mndjob :End of /NAMES list.
```

채팅방으로 봇마스터가 접속하게 하면서 채팅방의 상태 등의 정보를 받는다.

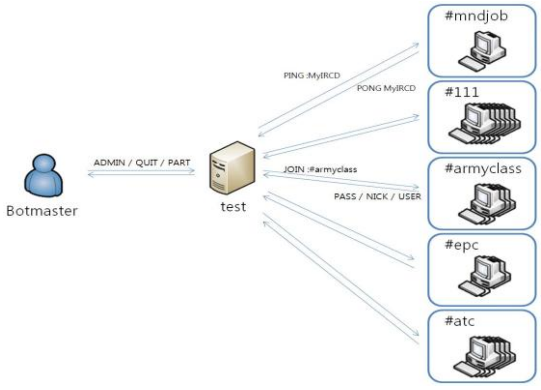
봇마스터의 호스트 폴더 검색

(그림 5)에서처럼 html 파일 내에 textarea 필드에 자바스크립트를 실행하여 호스트의 폴더 정보를 가져온 결과를 보여주고 있다. 아래에는 $ 로 시작하는 프롬프트가 있고 이곳을 통해서 쉘을 쓰듯이 웹 페이지에서 호스트에 폴더 위치를 변경하거나 목록을 보는 등의 실행을 하는 것을 볼 수 있다. 윈도우 시스템 폴더나 로그폴더들을 검색하고 있다.



(그림 5) 봇마스터의 호스트 폴더 검색 행위

이와 같이 수집된 패킷들의 내용에서 나타나는 IRC 명령의 추적을 통해서 아래의 (그림 6)과 같이 1 개의 봇마스터와 1 개의 IRC 서버를 발견하였고, 5 개의 채팅방과 112 대의 감염된 호스트가 접속한 것을 확인하였다. 외부의 봇마스터는 교내망의 botcnc 서버(IRC 서버이자 C&C 서버)를 통해 채팅 서버를 관리하고 감염된 호스트 통제와 같은 악성 행위를 하는 것으로 나타났다.



(그림 6) 트래픽 분석을 통해 밝혀진 봇넷의 구조

5. 결론 및 향후 전망

본 논문에서는 최근 들어 사이버 공격의 주요한 위협이 되고 있는 봇넷의 사례를 연구하기 위해 교내망에서 봇넷 트래픽을 수집하고 분석하여 실제 봇넷이 동작하고 트래픽 패턴을 찾아내었다. 패킷의 내용을 통해 추적한 결과, 수집된 트래픽은 IRC 프로토콜을 사용하는 봇넷으로 확인되었으며, 트래픽을 수집한 대상 서버는 외부의 봇마스터로부터 명령을 받아 하위의 봇호스트들을 통제하고 명령을 전달하는 것으로 분석되었다.

이러한 봇넷 트래픽을 탐지하는데 특정 송신지의 주소별로 트래픽을 분류하여 PPS 가 높은 순으로 찾아내는 방법은 큰 효과를 보이지 못한다. 즉, 봇마스터와 봇 C&C 서버 사이의 트래픽은 상대적으로 적고, 봇 C&C 서버와 봇호스트들간의 트래픽은 대량이긴 하지만 1:N 의 통신 형태를 가지기 때문이다. 따라서, IRC 를 이용하는 봇넷에서는 이러한 방법을 통해 봇마스터나 봇 C&C 서버는 찾아낼 수 있지만 인터넷에 폭넓게 산재해 있는 봇호스트들을 찾아내는 것이 어렵다. 이러한 봇넷을 찾아내기 위해 침입 탐지 장비와 같은 곳에서 패킷의 내용 중 특정 봇이 가지는 시그너쳐를 기반으로 탐지할 수도 있으나 조금만 봇이 변형되어도 탐지가 어려워지는 문제가 있다.

최근에는 IRC 봇이나 HTTP 봇과 같은 기존의 중앙 집중식 명령 전달 및 제어 구조를 가진 봇과 달리 P2P 프로토콜을 사용하여 더 유연한 공격 구조를 지닌 봇이 공격자의 위치 탐지를 더 어렵게 하고 있다. 추후에 이러한 P2P 봇의 트래픽을 분석하여 특성을 밝히는 것도 필요하리라 본다.

참고문헌

[1] Dean Turner, Marc Fossl, Eric Johnson, Trevor Mack, Joseph Blackbird, Stephen Entwisle, Mo King Low, David McKinney, Candid Wueest. “Symantec Global Internet Security Threat Report”, 2008

[2] Arbor Networks. “Worldwide Infrastructure Security Report”, 2007

[3] 전용희, “봇넷 기술 개요 및 분석”, 정보보호학회지, 제 18 권 제 3 호, pp. 101-108, Jun 2008

[4] Hyundo Park, Peng Li, Debin Gao, Heejo Lee, Robert H. Deng, "Distinguishing between FE and DDoS Using Randomness check", Information Security Conference(ISC2008), LNCS, Vol. 5222, pp. 131-145, Sep. 15. 2008

[5] 최현상, 권종훈, 김인환, 이희조, "악성코드를 이용한 봇넷 공격", 경영과 컴퓨터, pp. 144-147, Jul. 2008

[6] 최현상, 이희조, "행위기반의 봇넷 탐지기술", 정보보호 21C, Vol. 86, pp. 80-83, Oct. 1. 2007

[1] 본 연구는 지식경제부 및 정보통신연구진흥원의 IT 핵심기술개발사업의 일환으로 수행하였음. [2008-S-026-01, 신종 봇넷 능동형 탐지 및 대응 기술]

[2] 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음. (IITA-2008-C1090-0801-0016)