HTTP 기반 악성 봇넷 분석 (Kraken 봇넷을 중심으로)

장대일*, 이재서*, 박준형**, 김민수***, 노봉남****
*전남대학교 정보보호협동과정
{daels, mirr1004}@lsrc.jnu.ac.kr
**전남대학교 시스템보안연구센터
werther@lsrc.jnu.ac.kr
***목포대학교 정보보호학과
pheonix@mokpo.ac.kr
****전남대학교 전자컴퓨터공학부
bbong@jnu.ac.kr

Analysis of HTTP-Based Malicious Botnet (The cases of Kraken Botnet)

Dae-il Jang*, Jae-Seo Lee*, Jun-Hyung Park**,
Minsoo Kim***, Bong-Nam Noh****
*Interdisciplinary Program of Information Security, Chonnam University, korea
**System Security Research Center, Chonnam National University, Korea
***Dept. of Information Security, Mokpo National University, Korea
****ept. of Electronics Computer Engneering, Chonnam University, Korea

요 약

악성 봇이 현대 인터넷 보안의 큰 위협으로 등장함에 따라, 이러한 봇을 탐지하기 위한 많은 연구가 진행되고 있다. 하지만 악성 봇은 꾸준히 진화하여 탐지방법을 무력화시키고 있으며, 최근 HTTP를 이용한 악성 봇의 등장으로 그 탐지와 대응이 더욱 어려워지고 있다. 게다가 웹기반 서비스들의 증가로 HTTP를 이용하는 패킷은 통신량의 대부분을 차지하고 있으며, 이들에 대한 분석은 큰 부하를 발생시키게 된다. 이러한 문제를 해결하기 위해서는 악성 봇넷을 효과적으로 탐지하기 위한 효율적인 매커니즘을 선택하여야 하며, 본 논문에서는 대표적인 HTTP 기반 악성 봇넷인 크라켄(Kraken) 봇넷의 특성을 분석하였다.

1. 서론

HTTP 프로토콜은 최근 악성 봇넷에서 P2P, IRC와 더불어 가장 널리 사용되는 프로토콜이다. HTTP가 널리 이용되고 많은 서비스를 제공하게 되면서 방화벽이나 보안제품에서 HTTP 프로토콜에 대한 점검이 어렵게 되었다. 이러한 이유로 기존의 IRC 프로토콜을 이용하는 악성 봇넷을 제작하던 공격자들은 악성 봇넷이 방화벽 등에 차단되자 방화벽에 친화적인 HTTP 프로토콜을 악성 봇넷의 통신에 이용하게 되었다[1]. 이에 HTTP 봇넷의 탐지를 위해 HTTP 프로토콜을 이용하여 통신을 하는 악성 봇넷의 단계적인 행위에 대한 특성을 분석 할 필요가 있다.

본 논문의 2장에서는 악성 봇넷에 대해 알아보고 3장에서는 HTTP 봇넷의 구조적 특징과 사례에 대해 알아볼 것이다. 4장에서는 HTTP 기반 크라켄 봇넷의 통신과정과 봇넷이 발생시키는 UDP, TCP, HTTP 트래픽을 연결단위로 구분하여 분석할 것이다. 이를 통해 HTTP 봇넷의 동작 유형을 분석할 수 있고 동작 상황을 예측하여 HTTP 봇넷의 탐지 방법을 구상하는데 도움을 줄 수 있다.

2. 봇넷

봇(bot)은 "로봇(robot)"의 준말로서, 사용자나 다른 프로그램 또는 사람의 행동을 흉내 내는 대리자로 동작하는 프로그램을 의미한다. 봇에 해커들이 웜, 바이러스를 접목시키면서 현재의 악성 봇으로 진화하였다[2]. 악성 봇은 허가 없이 악성 프로그램을 설치하기도 하고 정보를 훔쳐가기도 하며 스팸메시지를 발송하거나 분산 공격을 하기도 한다. 시스템을 감염시킨 봇은 원격지의 공격자에게 해당 시스템의 제어권을 양도하고 공격자의 명령을 대기한다. 이렇게 감염된 시스템을 악성 봇 또는 드론(drone) 및 좀비(zombie)시스템 이라고 한다[3].

봇넷(botnet)은 봇의 집단이다. 악성 봇넷은 공격자의 명령에 따라 동작하는 악성 봇에 감염된 좀비 시스템이 인터넷 공간에 형성하고 있는 네트워크를 의미하고 명령 및 제어 서버와 좀비 시스템으로 구성된다. 공격자인 봇마스터는 명령 제어 서버에 연결하여 좀비 시스템들에게 명령을 전달하고 좀비 시스템은 명령에 따라 동작한다.

2.1 악성 봇넷의 구조적 특징

대부분의 악성 봇들은 악성 봇을 구성하는 모듈 구성이나 수행하는 기능적인 영역에 따라 통신 영역, 제어 영역, 주요 기능 영역, 확장 영역, 자기업데이트 영역으로 구분된다. 통신영역은 공격자와의 통신 채널을 확립하는 기능, 제어영역은 공격자의 명령 수신 및 실행 기능, 주요 기능 영역은 숙주 시스템에 악성 봇을 적재 또는 실행하는 기능, 확장 영역은 네트워크에 연결된 취약한 시스템을 찾고 공격하여 봇을 감염시키는 기능, 자가 업데이트 영역은 좀비 시스템에 감염된 악성 봇을 최신상태로 유지하는 기능을 한다[4].

악성 봇넷의 C&C모델은 중앙집중형 모델, P2P모델, 랜덤형 모델 3가지로 구분된다[5]. 중앙집중형 모델은 IRC 봇을 비롯한 현재 존재하는 대부분의 봇넷이 갖는 기본구조이지만 중앙 관리 서버를 차단함으로 봇넷의 활동을 봉쇄할 수 있다. P2P모델은 봇넷을 관리하기 위한 특별한 서버를 두지 않고 봇들이 서로 통신함으로써 활동한다. 이 때문에 관리와 명령하달이 어렵지만 차단하기 어렵다는 장점을 갖는다. 랜덤형 모델은 높은 생존률을 보이고 단일 봇은 다른 봇이나 C&C에 대한 어떠한 정보도 가지고 있지 않아[6] 인터넷을 무작위로 스캔하여 발견된 다른 봇에 메시지를 전송한다. 구현이 쉽고 단일 봇의 탐지로 봇넷을 탐지하기 힘들다는 장점을 갖지만 메시지 소실율이 높다는 단점이 있다.

3. 악성 HTTP 봇넷

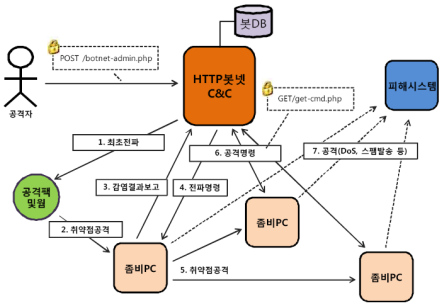
최근 악성 봇넷에서 통신을 위해 가장 많이 사용되는 방법 중 하나는 HTTP이다. HTTP를 사용하면 다음과 같은 두 가지의 큰 장점을 갖는다. 첫 번째, HTTP는 인터넷 트래픽의 많은 양을 차지하고 있어 HTTP를 사용하는 봇넷을 탐지하기 어렵다. 두 번째, 대부분의 방화벽이나 게이트웨이가 보안 정책상 HTTP를 검사하지 않아 통신이 원활하다[5]. 이러한 장점을 이유로 최근 많은 봇들이 통신 채널로 HTTP를 선택하고 있다.

3.1 악성 HTTP 봇넷 구조 및 특징

악성 HTTP 봇넷은 HTTP의 특성 상 중앙집중형 봇넷을 구성하고 C&C와 좀비 시스템 사이의 연결을 유지하지 않는다. 악성 HTTP 봇 클라이언트는 C&C서버로 정기적으로 폴링(poling)하여 연결을 유지한다[7]. 다음의 그림 1은 악성 HTTP 봇넷 명령과 제어 체계를 보여준다.

악성 HTTP 봇넷은 중앙집중형 봇넷 토폴로지를 형성한다. 하지만 위에서 언급했다시피 중앙집중형 토폴로지는 중앙 서버가 탐지되어 차단되면 더 이상 봇넷이 활동할 수 없는 단점을 지니고 있다. 이에 많은 악성 HTTP 봇넷은 멀티서버 구성 봇넷 토폴로지를 구성한다.

이러한 구조적 특징 이외에 악성 HTTP 봇넷은 SSL을 이용하여 암호통신을 하거나 페이로드 부분을 암호화하여 분석 및 탐지를 어렵게 한다. 또한 일반적인 HTTP 트래픽과 비교하였을 때 봇넷에 의해 발생하는 HTTP 트래픽은 응답이 매우 독특하거나 비정상적인 HTTP 헤더 필드나 페이로드를 갖는다[5].



(그림 1) 악성 HTTP 봇넷 명령과 제어 체계

3.2 악성 HTTP 봇넷 사례

악성 IRC 봇의 단점을 보완하기 위해 악성 P2P 봇으로 그리고 네트워크 레벨에서 탐지를 어렵게 하기 위해 악성 HTTP 봇으로 발전하였다. HTTP를 그대로 사용하는 봇이 있는 반면, 변경된 HTTP를 이용하거나 다른 통신 프로토콜을 이용하는 봇도 있다. 또 HTTP를 이용하여 명령을 하달하고 모듈을 업데이트하는 등 봇넷이 수행하는 모든 행위를 하기도 하고 일부 기능만을 부분적으로 수행하기도 한다. 다음의 표 1은 2008년 4월 현재 가장 많은 스팸메시지를 보내는 악성 봇넷을 조사한 것이다. 표 1과 같이 상위 10개의 봇넷 중에 8개의 봇넷이 HTTP를 사용하는 악성 HTTP 봇넷에 해당한다.

<표 1> 대량스팸발송을 위한 상위 악성 봇넷[9]

이름	형태	좀비 수
srizbi	HTTP	315,000
Kraken	HTTP	185,000
Rustock	HTTP	150,000
Cutwail	HTTP	125,000
Storm	HTTP, P2P	85,000
Grum	HTTP	50,000
Onewordsub	-	40,000
Ozdock	HTTP	35,000
Nucrypt	HTTP	20,000
Wopla	HTTP	20,000
Spamthru	P2P	12,000

4. 악성 HTTP 봇넷 분석:크라켄(Kraken)

본 논문에서는 악성 HTTP 봇넷 중에서 크라켄을 중심으로 HTTP 봇넷을 분석하였다. 크라켄은 기본적으로 감염된 시스템으로부터 스팸을 발송하기 위해 디자인되었으며, 크라켄에 감염된 좀비 시스템은 여러 가지 동적인 DNS 해결 서비스로부터 체계적으로 생성된 서브도메인에 의한 C&C 서버로의 접근은 시도한다[10]. C&C 서버와

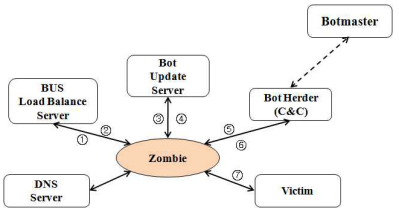
연결되면 일련의 과정을 거쳐 스팸메시지를 보내게 된다.

4.1 크라켄 특징

크라켄은 다음과 같은 주요 특징을 갖는다.

· 패킷(페이로드) 암호화
· 정상적인 HTTP 프로토콜 사용
· C&C 서버와 통신을 통해 자체적인 모듈 업데이트
· 다양한 종류의 C&C 서버
· UDP, TCP, HTTP 등 다양한 프로토콜 사용
· UDP&TCP 447, TCP 443,80 등 다양한 포트 사용
  - 모듈 버전 v316 이후 UDP 랜덤포트 사용

위의 특징에서 볼 수 있듯 크라켄은 그림 2와 같이 다양한 C&C 구조를 가지고 있다.



(그림 2) 크라켄 봇넷 구조

가장 대표할만한 악성 HTTP 봇넷에 걸맞게 복잡하면서도 구조적인 것을 볼 수 있다. 이렇게 복잡한 C&C 구조는 많은 수의 좀비를 효율적으로 관리한다. 좀비들의 초기 UDP 통신을 원활히 수행하기 위한 로드밸런스 서버와 좀비 시스템의 모듈을 업데이트하기 위한 봇 업데이트 서버, 실제적인 스팸 메일 발송을 명령하는 C&C 서버 등이 크라켄 봇넷을 구성하고 있다.

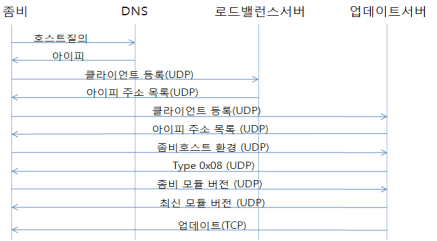
크라켄은 많은 좀비 시스템을 관리하기 위해 많은 C&C를 필요로 한다. 이에 로드밸런스 서버나 업데이트 서버를 두어 트래픽을 분산시키고 봇의 업데이트나 C&C로의 연결을 원활하게 한다.

4.2 크라켄 통신과정 분석

크라켄은 UDP 447 포트, TCP 443, 447 포트, HTTP 80 포트를 이용하여 좀비 호스트를 관리하고 스팸메시지 발송을 위한 명령을 하달한다. 각 포트 별 수행하는 역할은 다르고 최신버전으로 모듈이 업데이트 되었을 경우 UDP 447 포트만을 사용하는 것이 아닌 랜덤한 포트를 이용하여 통신한다. 좀비와 서버간의 통신에 있어 모든 페이로드는 자체 알고리즘에 의해 암호화 되어 있다. 또한 전송되는 패킷마다 패킷타입으로 역할을 규정하고 있다.

크라켄에 처음 감염 되었을 때 좀비는 스팸메시지를 발송할 수 있는 환경이 되는지 검사하기 위해 구글에 SMTP 메시지를 3차례 발송한다. 이후 C&C를 찾기 위해 DNS에 질의한다. 호스트 부분은 랜덤한 문자열이고 *.dyndns.org, *.dynserv.com, *.mooo.com, *.yi.org 등의 서버에 1초~4초 간격으로 질의한다. DNS 질의는 C&C와 연결이 이루어진 이후에도 계속 진행된다.

DNS를 통해 질의 호스트의 아이피를 받은 후 서버의 UDP 447 포트로 클라이언트 등록 패킷을 전송한다. 클라이언트 등록 패킷은 크라켄 내부적으로 0x01의 타입 번호를 가지고 서버에서 패킷에 대한 응답이 올 때 까지 계속해서 보내게 된다. 그 간격은 10초 ~ 20초 사이로 랜덤하게 전송된다. 서버에서 응답 패킷으로 아이피 주소 목록에 대한 패킷이 전송된다. 패킷 타입은 0x02이고 로드밸런스 서버일 경우 업데이트 서버 아이피를 5개 전송한다. 업데이트 서버일 경우 다른 아이피에 대한 정보가 없다. 로드밸런스 서버에 접속하여 아이피 리스트 목록을 받을 경우 좀비 호스트는 받은 목록 중 하나를 랜덤하게 선택하여 클라이언트 등록 패킷을 전송한다. 응답패킷으로 비어있는 아이피 주소 목록을 받으면 타입 0x03인 좀비 호스트의 컴퓨터 환경에 대한 정보를 전송하게 된다. 좀비의 모듈 버전이나 윈도우 버전, 서비스팩 설치 유무, 사용언어, 국가코드, CPU, 메모리 용량 등 많은 정보를 전송한다. 응답으로 타입 0x08이 오는데 이는 단순한 응답패킷의 역할을 하고 좀비는 다시 좀비 모듈의 정보를 전송한다. 서버에서는 최신 모듈의 정보를 전송하고 TCP 447 포트를 이용하여 좀비 모듈을 업데이트 한다.

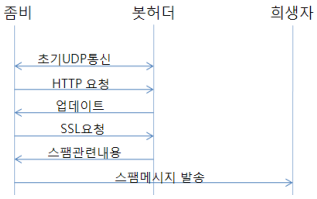


(그림 3) 크라켄 UDP & TCP 통신 과정

업데이트 이후 좀비는 봇허더에 접근하면서 UDP 랜덤 포트를 이용하여 크라켄 상태 패킷을 전송한다. 초기 통신 과정을 진행하고 HTTP를 이용하여 80포트나 443포트를 통해 명령을 주고받고 모듈을 업데이트 한다. 현재 좀비가 스팸메시지를 발송할 수 있는 상태인지를 전송한 후 타입 0x01을 제외한 나머지 UDP 통신을 수행한다.

크라켄은 HTTP POST 메시지를 이용하여 업데이트에 대한 통신을 하는데 이때 서버에 요청하는 파일 타입은 매우 다양하다. JPEG나 GIF 같은 그림파일부터 동영상 파일, 음악 파일, 일반적인 확장자의 웹페이지 요청 등 매우 다양한 파일을 요청하는 것처럼 보이지만 내용은 업데이트에 대한 것으로 동일하다. 이후 443 포트를 이용하여 스팸 템플릿이나 스팸메시지 내용, MX 서버 리스트 등을 전송받는다. SSL 포트를 이용하지만 SSL로 암호화된 데이터가 아닌 기존의 암호화 방식을 이용하여 데이터를 암호화 한다. 이후 좀비는 봇허더의 명령에 따라 스팸메시지를 발송한다. 스팸메시지 발송 시 전송받은 MX 서버 리

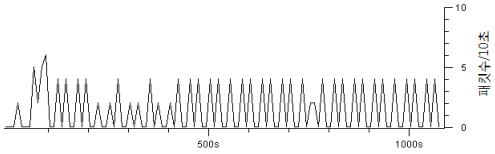
스트를 DNS에 질의하는 과정을 거친다.



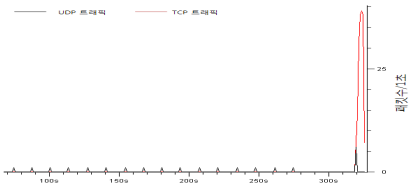
(그림 4) HTTP 통신과정

4.3. 크라켄 트래픽 분석

크라켄은 초기에 DNS에 질의하는데 그림 5는 이때의 트래픽 변화량을 보여준다. 그림 6은 UDP&TCP 447포트의 트래픽 변화량이다. 크라캔이 발생시키는 패킷이 거의 동일한 시간 간격으로 보내는 것을 확인할 수 있다.

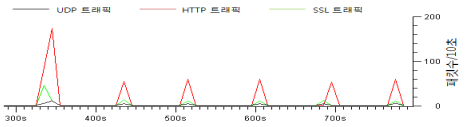


(그림 5) DNS 트래픽 변화량

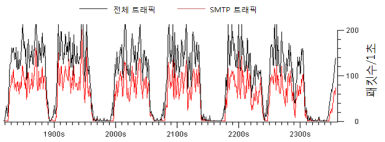


(그림 6) UDP & TCP 447 포트 트래픽 변화량

업데이트 이후 좀비는 봇허더와 통신에서 그림 7에서처럼 전체적으로 큰 차이를 보이지 않지만 초기에 트래픽 양이 월등히 많은 것을 볼 수 있다. 이는 봇허더와의 초기 통신 시 좀비 모듈을 업데이트하기 위한 것임을 알 수 있다. 봇허더와 통신 이후 좀비는 봇허더의 명령에 따라 스팸메시지를 전송하게 된다. 이 과정에서도 UDP와 TCP 447 포트를 이용한 통신이나 DNS 질의와 같이 균일한 간격을 유지하면서 트래픽을 유발하는 것을 확인할 수 있다.



(그림 7) 좀비와 봇허더와의 통신 트래픽



(그림 8) 스팸메시지 발송 시 트래픽

이와 같이 크라켄은 UDP, TCP, HTTP등 다양한 프로토콜을 사용하여 봇허더와 통신하고 통신과 스팸메시지를 발송하는 과정에서 트래픽이 일정한 간격으로 발생되는 특징이 나타났다. 이러한 규칙적인 특징은 보안 장비를 통해서 미리 감지가 가능할 것으로 예측된다.

5. 결론

악성 봇의 공격은 오늘날 큰 위협으로 대두되었다. 이러한 봇들이 구성하는 봇넷은 분산서비스거부공격이나 스팸메시지 발송 등 조직화된 악성 행위를 일삼고 있다. 또한 봇의 탐지를 어렵게 하고 봇넷의 확산을 위해 IRC 기반에서 P2P나 HTTP로 프로토콜을 변경하고 단일 C&C에서 다중 C&C로 변화하는 등 진화를 거듭하고 있다.

본 논문에서는 네트워크 레벨에서 HTTP 기반 봇넷의 탐지를 위해 대표적인 HTTP 기반 봇넷인 크라켄 봇의 트래픽을 분석하였다. 좀비가 봇허더를 찾고 스팸메시지를 보내는 과정에서 발생시키는 트래픽은 일정한 패턴을 갖는다. 통신하는 과정에서 나타나는 단계적 행위 특성은 HTTP 기반 봇넷을 미리 감지할 수 있고 이러한 특성을 침입탐지시스템이나 방화벽에 적용하여 봇넷의 확산을 방지할 수 있을 것으로 기대한다.

참고문헌

[1] Markus J, Zulfikar R., “Crimeware: Understanding New Attacks and Defenses“, Addison Wesley Professional, ISBN 0-321-50195-0, April 2008.

[2]인터넷침해사고대응지원센터, “2007 정보시스템 해킹 바이러스 현황 및 대응”, 한국정보보호진흥원(KISA), 2007년 12월

[3] Basudev Saha, Ashish Gairola., “봇넷: An Overview,“CERT-In, June 2005.

[4] Markus J, Zulfikar R., “Crimeware: Understanding New Attacks and Defenses“, Addison Wesley Professional, ISBN 0-321-50195-0, April 2008.

[5] Trend Micro, “Taxonomy of Bptnet Threats“, November 2006

[6] Evan Cooke, Farnam Jahanian, and Danny McPherson, The Zombie Roundup:
Understanding, Detecting, and Disrupting 봇넷s, Proc. of Steps to Reducing Unwanted Traffic on the Internet Workshop (SRUTI ’05), Boston, 2005.

[7] Niels Provos, Thorsten Holz., “Virtual Honeypots: From Botnet Tracking to Intrusion Detection“, Addison Wesley Professional, ISBN 978-0-321-33632-3, July 2007.

[9] Joe Stewart, “Top Spam Botnets Exposed“, Secure Works, http://www.secureworks.com/research/threats/topbotnets, April 2008.

[10] Pedram Amini, “Kraken Botnet Infiltration“, http://dvlabs.tippingpoint.com/blog/2008/04/28/kraken-botnet-infiltration, April 2008.