# DNS 싱크홀 적용을 통한 악성봇 피해방지 기법 및 효과

## (Preventing Botnet Damage Technique and It's Effect using Bot DNS Sinkhole)

김 영 백 † 이 동 련 †† 최 중 섭 ††† 염 흥 열 ††††
(Young Baek Kim) (Dong-Ryun Lee) (Joongsup Choi) (Heung-Youl Youm)

**요 약** 악성봇은 해커에 의해 원격 조정되어 명령에 의해 스팸메일 발송, DDoS 공격 등의 악성행위를 수행하는 웜/바이러스 이다. 악성봇은 이전의 웜/바이러스와 달리 금전적인 이득을 목적으로 하는 경우가 많은 반면 감염사실을 피해자가 인지하기 쉽지 않아 피해가 심각한 실정이다. 이에 대한 대응 방안으로는 해커의 명령을 전달하는 명령/제어 서버의 차단이 필요하다. 이 중 악성봇 DNS 싱크홀 기법이 국내에서 적용하고 있는 봇 대응 시스템으로, 본 논문의 목적은 이 방식의 효과성을 제시하는데 있다. 본 논문에서는 1년 이상의 장기간 동안 악성봇 및 Botnet 을 관찰하여 Bot 감염 PC의 감염 지속시간, Bot 명령/제어 서버의 특성 등을 파악하고, 악성봇의 피해를 방지하기 위한 효과적인 방안인 악성봇 DNS 싱크홀의 적용 결과를 분석한다. 이를 위하여 웜샘플 분석 툴을 이용하여 자동 분석체계를 구축하였고, 이를 시스템화 하였다. 또한, 분석을 통해 현재 국내에서 적용되고 있는 봇 대응 시스템의 타당성을 검증하였다.

**키워드** : 봇, 봇넷, 싱크홀, 허니넷

***Abstract*** Bot is a kind of worm/virus that is remotely controlled by a herder. Bot can be used to launch distributed denial-of-service(DDoS) attacks or send spam e-mails etc. Launching cyber attacks using malicious Bots is motivated by increased monetary gain which is not the objective of worm/virus. However, it is very difficult for infected user to detect this infection of Botnet which becomes more serious problems. This is why botnet is a dangerous, malicious program. The Bot DNS Sinkhole is a domestic bot mitigation scheme which will be proved in this paper as one of an efficient ways to prevent malicious activities caused by bots and command/control servers. In this paper, we analysis botnet activities over more than one-year period, including Bot's lifetime, Bot command/control server's characterizing. And we analysis more efficient ways to prevent botnet activities. We have showed that DNS sinkhole scheme is one of the most effective Bot mitigation schemes.

**Key words** : Bot, Botnet, DNS Sinkhole, Honeynet

† 정 회 원 : 한국정보보호진흥원 해킹대응팀 선임연구원
ybkim@kisa.or.kr
†† 비 회 원 : 한국정보보호진흥원 해킹대응팀 연구원
ryuni@kisa.or.kr
††† 비 회 원 : 한국정보보호진흥원 해킹대응팀 팀장
jschoi@kisa.or.kr
†††† 비 회 원 : 순천향대학교 정보보호학과 교수
hyyoum@sch.ac.kr
논문접수 : 2008년 5월 29일
심사완료 : 2008년 10월 2일


정보과학회논문지: 컴퓨팅의 실제 및 레터 제15권 제1호(2009.1)

## 1. 서 론

악성봇은 "IRC Robot"에서 유래한 단어로, IRC(Internet Relay Chat)라는 인터넷 채팅에서 사용자 로그아웃 이후에도 대화방을 보전하기 위해 홀로 남아 대화방을 지키던 프로그램이 그 기원이다. 이후 IRC Robot에 여러 기능이 추가되고 해커들이 웜, 바이러스에 접목시키면서 현재의 악성봇으로 진화하게 된다. 이러한 악성봇에 대하여 다양한 대응을 하고 있으나, 국내에서 발견되는 악성 봇 명령/제어 서버와 감염 PC의 수는 여전히 상당수에 이르고 있다. 그 뿐 아니라 최근 악성봇을 이용하여 특정 사이트에 DDoS 공격을 수행하고 공격 중단을 위해 금전적인 이윤을 요구하는 등 그 악용 사례

들이 다양해지고 있다[1].

악성봇은 취약점을 가지고 있는 PC에 자동으로 전파되며, 감염 시 해커가 지정해 놓은 명령/제어 서버에 접속하여 해커로부터의 명령을 기다린다. 이렇게 악성봇 감염 PC가 접속하는 해커의 서버를 악성봇 명령/제어(C&C : Command and Control) 서버라고 하며, 명령/제어 서버와 감염 PC 들로 구성된 네트워크를 Botnet (Bot Network)이라고 한다. 명령/제어 서버의 경우 초기에는 정상적인 IRC 채팅 서버에 해커가 방을 개설하여 사용하는 경우도 있었으나 최근에는 대부분 취약한 서버를 해킹한 후 IRC 프로그램을 설치하여 사용하고 있다. 예전의 웜/바이러스의 경우 감염된 PC나 서버에 포맷 등 직접적인 악성행위가 발생하였으나, 악성봇의 경우 감염 PC나 명령/제어 서버로 악용된 서버에는 피해를 주지 않는 경우가 많아 사용자가 감염여부를 확인하기가 어렵다. 기존의 웜/바이러스의 경우 해커의 실력과시 등을 위하여 감염PC에 직접적인 해를 입혔으나, 악성봇의 경우 감염된 PC를 원격에서 조정하여 금전적 이득을 취하는 것이 주목적이기 때문에 최대한 감염사실을 숨긴 상태로 해커의 명령을 수행한다. 해커는 구성된 Botnet으로 스팸메일 발송, DDoS 공격 등의 악성행위를 수행하며, 메일발송 건수당 비용을 받거나 DDoS를 사주 받아 수행하고 수고비를 받는 경우도 많다.

악성봇 명령/제어 서버는 이러한 악성행위의 중심에 있으며, 악성봇 명령/제어 서버의 차단만으로도 해커로부터의 명령 전달을 방지할 수 있어 악성봇의 악성행위를 효과적으로 막을 수 있다. 그림 2는 악성봇 명령/제어 서버의 명령채널인 "#p4"에 접속하여 악성행위를 수행하는 것을 모니터링한 화면으로, 특정 웹사이트에서 "navy.exe"라는 악성파일을 다운로드 받은 후 실행하는 명령(.http.execute)과 주변 PC의 취약점을 찾아 악성봇을 감염시키라는 명령(.scan.startall) 2가지를 동시에 수행하고 있는 모습을 보여주고 있다. 이러한 명령을 받은 감염 PC들(P4-xxxxx)은 명령수행 결과를 해커에게 보고하고 있다.

본 논문에서는 2007년의 악성봇 데이타 산출 결과를

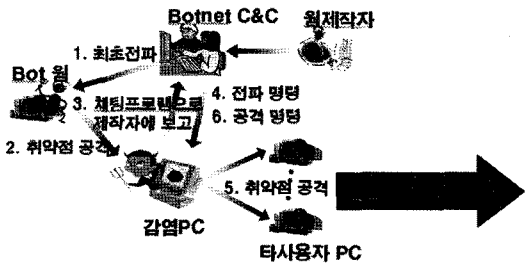


그림 1 악성봇 감염 과정

```
* Now talking in #p4
* Topic is ".scan.startall |.http.execute http://71.XX.XX.238/navy.exe c:\navy.exe"
* Set by MDfgdsfee3 on Thu Jun 29 18:14:43
(P4-pyfawp) download to c:\navy.exe finished.
(P4-ryhobok) Receiving file.
(P4-pyfawp) opened c:\navy.exe.
(P4-ajbpsdg) download to c:\navy.exe finished.
(P4-ulkfpxz) download to c:\navy.exe finished.
(P4-ulkfpxz) open c:\navy.exe.
(P4-ajbpsdg) open c:\navy.exe.
```

그림 2 악성봇 명령 전달 화면

중심으로 악성봇의 주요 특성에 대하여 살펴보고자 한다. 악성봇 데이타 산출은 허니넷에서 수집한 악성코드 샘플을 분석한 결과와, 악성봇의 명령/제어 서버 도메인으로의 접속을 우회시키는 악성봇 DNS 싱크홀 시스템의 운영 결과를 바탕으로 하였다. 결과 데이타들은 1년 이상의 시스템 운영을 통해 축적된 데이타를 통계화 한 것으로 장기간 동안의 악성봇 트렌드를 읽고자 하였으며, 국내 악성봇 감염 PC 및 악성봇 명령/제어 서버로 구성된 Botnet에 대한 구체적인 현황 및 대응 방향을 구현하고 분석하였다.

## 2. 허니넷 시스템 구축 및 운영 결과

### 2.1 허니넷 시스템 구축

허니넷(Honeynet)이란 해커나 악성코드로부터의 공격을 받아 해커의 행위나 악성코드의 전파 특성들을 분석하기 위해 만든 취약한 네트워크로, 유입되는 트래픽은 모두 악성 트래픽으로 간주되어 분석하게 된다. 그러나 허니넷 내부에 아무런 시스템이 없게 되면 해커나 악성코드가 공격을 하지 않게 되므로, 취약점 모듈을 이용하여 악성코드를 수집하는 프로그램이나, 실제 가상의 기업 환경을 꾸며 해커를 유도하는 형태로 운영하고 있다.

허니넷은 다양한 형태로 구성하고 있으나 본 논문의 결과는 UNIX 기반의 오픈 소스 프로젝트인 nepenthes 프로그램[2]을 기반으로 한 허니넷에서 수집한 악성코드 샘플을 중심으로 분석하였다. nepenthes는 악성코드 수집을 위하여 악성코드가 주로 사용하는 윈도우즈 취약점을 에뮬레이션 하는 모듈을 사용하며, 윈도우 운영체제가 아닌 UNIX 운영체제에서 실행된다. 이 경우 운영체제가 UNIX 이므로 운영체제 자체가 감염되지 않아 허니넷 내부에서 외부로의 공격이 발생하지 않는다는 장점이 있다. nepenthes는 취약점을 에뮬레이션하여 악성코드의 공격을 기다리는 모듈과 공격 발생시 실제 윈도우즈 OS 처럼 동작하여 전파되는 악성코드를 다운로드 하는 모듈, 그리고 다운로드된 모듈을 분석을 위해 전송하는 모듈 등으로 구성되어 있다. 이번 연구에는 이러한 nepenthes 모듈을 허니넷에 위치한 PC에 설치하여 수집한 일일 약 200개 가량의 악성코드 샘플이 분석에 사용되었다[3].

### 2.2 악성봇 샘플 자동분석 시스템 구축

nepenthes에 의하여 허니넷에서 수집된 악성코드들은 1차로 악성코드 분석 S/W[4]를 통하여 악성코드의 백신 진단여부 검사, 감염 시 초기 악성행위 관찰 등이 이루어진다. 악성코드 분석 S/W는 국외 백신회사인 NORMAN 사의 제품을 구매하여 사용하였다. 악성코드 분석 S/W 에서는 수집된 악성코드 별로 파일시스템 변경사항, 레지스트리 변경사항, 윈도우즈 서비스 등록사항, 네트워크 트래픽 발생사항에 대하여 분석결과를 그림 3과 같이 text 파일 형태로 출력한다. 이 중에서 "Network service" 부분에서 연결을 시도하는 서버가 악성봇의 명령/제어 서버로써, 그림 3에서는 TCP/45569 포트로 접속을 시도하고 있음을 알 수 있다.

악성코드 분석 S/W는 허니넷에서 당일 수집된 200개 가량의 악성코드에 대해 분석을 수행한다. 이러한 분석결과를 사람이 모두 확인하여 악성봇 명령/제어 서버 목록을 추출해 내기는 시간이 많이 걸리므로, 보다 효과적으로 분석결과를 추출하기 위하여 악성봇 샘플 자동분석 시스템을 구축하였다. 자동 분석 시스템에서는 1차 분석결과를 악성봇 명령/제어 서버별로 분류하고 접속하는 IRC 채널정보 등을 DB에 저장하여 향후 악성봇 DNS 싱크홀 시스템에 적용할 수 있도록 한다. 자동 분석 시스템은 악성코드 분석 S/W[4]의 후처리 시스템으로, 분석결과에서 악성봇 명령/제어 서버의 도메인명, IP, 사용포트, 백신진단명, IRC 채널명, USERNAME, NICKNAME 등의 정보를 악성코드 분석 S/W의 분석결과 출력 파일에서 추출하여 그림 4와 같이 DB에 저장한다.

DB에 저장된 악성봇 명령/제어 서버의 목록은 이후 실제 접속시도를 통해서 접속 가능한 Active 서버인지 아니면 이미 차단되었거나 도메인이 삭제된 서버인지 여부를 체크하는 과정을 거치게 된다. 이 과정에서 Active 한 서버에 대해서는 별도로 도메인 목록을 저장하여 이를 악성봇 DNS 싱크홀 시스템에 적용하게 된다. DNS 싱크홀 시스템에서는 ISP 등과의 협조를 통해 해당 도메인을 차단/우회 조치하고, 수집된 악성코드 샘플은 백신

```
{ Changes to filesystem }
  * Creates file C:\WINDOWS\SYSTEM32\symantecch.exe.

{ Changes to registry }
  * Creates key "HKCU\Software\Obsidium".
  * Creates key "HKCU\Software\Obsidium\{7D9F39B5-C47 .....
  * Creates value "Windows Update Drive"="symantecch.exe" .....
  * Sets value "restrictanonymous"="" in key "HKLM\Syst .....

{ Network services }
  * Looks for an Internet connection.
  * Connects to "loxxx.xxxxx.it" on port 45569 (TCP).
  * Connects to IRC Server.
  * IRC: Uses nickname [XP]|036109.
  * IRC: Uses username jbatsmp.
  * IRC: Joins channel ##und3r with password dr0ws1ng69.
  * IRC: Sets the usermode for user [XP]|036109 to +n+x+B+i.
.....
```

그림 3 악성코드 분석 S/W[3] 분석 결과

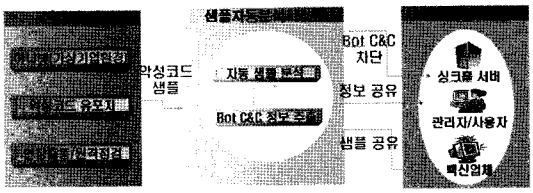


그림 5 악성봇 샘플 자동 분석 시스템

회사에 전달하여 향후 백신에서 탐지 후 치료가 가능하도록 하고 있다.

### 2.3 악성봇 샘플 자동분석 시스템 운영 결과

2007년 동안 수집된 악성코드 개수는 약 38,000여개로 월평균 3,200 여개의 악성코드를 수집하여 분석하였으며, 수집하는 악성코드는 MD5 해쉬값 계산을 통하여 중복 수집되는 경우를 방지하였다.

악성코드 분석 S/W[4]를 통하여 분석된 1차 분석결과를 바탕으로 분류해 보면 Spybot 계열이 45%로 가장 많았고, SDBot, IRCBot 등 봇 계열이 주를 이루었다. 또한 악성코드 분석 S/W에서 분석하지 못하는 신규 악성코드도 23% 가량 수집되었다. 특히 Spybot의 경우 370여 가지의 변종이 발견되었는데, 이는 악성봇의 소스코드가 공개되어 소스코드의 일부 수정만으로도 변종

| DOMAIN_NM | PORT | USER_NM | NICK_NM | CHANNEL_NM | CHANNEL_PASS | USER_MODE | SIG_NM |
|---|---|---|---|---|---|---|---|
| col :us | 8885 | dbnksq | USA\|94954 | #coko# | cokobotpass | +x+B | W32/Spybot.AHVS. |
| ak.i | 18555 | XP-1810 | [P0\|USA\|60022] | #aa | p00n3d | -x+i | no signature detectior |
| you strangled.net | 9908 | XP-5038 | [P0\|USA\|72521] | #1217 | 18952346 | | W32/Rdriv.A. |
| my. at | 1863 | XP-5090 | [P0\|USA\|62826] | ##921? | ㅐㄴㄱㅐㄴㄱㅐㄱ | is | W32/SDBot.AMPE. |
| my. at | 1863 | XP-5090 | [P0\|USA\|62826] | ##922? | ㅐㄴㄱㅐㄴㄱㅐㄱ | is | W32/SDBot.AMVO. |
| get om | 16666 | XP-3660 | [P0\|USA\|68222] | #VV# | MM | | W32/SDBot.AKYC. |
| dar oul.info | 6667 | XP\|USA-2 | XP\|USA-211915 | #불프# | 松 | | W32/Spybot.APPE. |
| 0x2 ar.info | 8081 | tlyhljc | tlyhljc | #.s | dfert | | BAT/Qhost.A. |
| bot ds.com | 65146 | opkqcqsx | [FUCKOFF]-744 | #a | imallowed2020 | +xt | W32/Spybot.BEUV. |
| h.rt g | 13830 | XP-3872 | [P0\|USA\|64026] | #mirg# | p00n3d | -x+i | |

그림 4 악성봇 명령/제어 서버 정보 DB

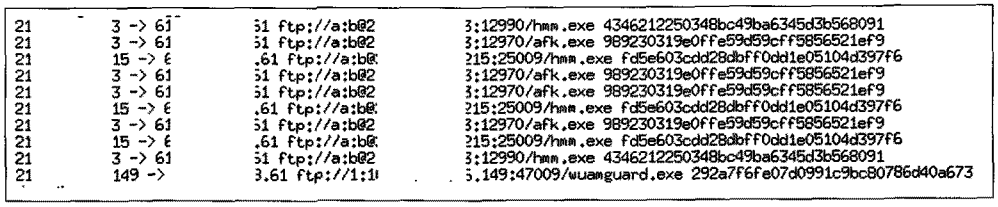


그림 6 악성코드 수집 및 MD5 해쉬값 계산

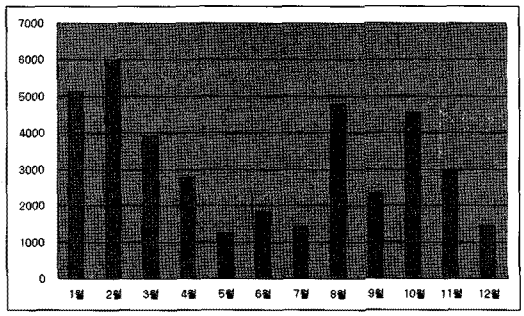


그림 7 악성코드 수집갯수

표 1 수집된 악성코드 종류

| 악성코드 명 | 수집비율 |
|---|---|
| W32/Spybot.xxx | 45.4% |
| W32/Malware.xxx | 7.4% |
| W32/SDBot.xxx | 5.8% |
| Bobax | 3.2% |
| W32/Ircbot.xxx | 2.9% |
| Korgo | 2.8% |
| W32/Sality.xx | 1.8% |
| Sasser | 1.1% |
| W32/Virut.x | 0.5% |
| ETC | 6.4% |
| no signature detection | 22.8% |

생성이 가능하기 때문이다. 같은 악성봇 계열인 SDBot과 Ircbot을 포함하면 악성봇 계열은 전체 수집 악성코드 중 54%나 차지한다. Spybot, SDBot 등은 모두 악성봇 계열 웜/바이러스로 자동분석 S/W[4]에 포함된 NORMAN사의 백신 프로그램에서 진단하는 진단명으로 다른 백신 프로그램에서는 다르게 진단할 수도 있다.

### 2.4 악성봇 명령/제어 서버 정보 수집 결과

수집된 악성코드를 분석하여 악성코드가 해커로부터 명령을 받기 위해 접속하고자 하는 명령/제어 서버를 추출하여 분석을 실시하였다. 명령/제어 서버의 IP를 국가별로 분류해 보면 미국이 전체의 56.2%로 가장 많은 수를 차지하였고, 한국이 5.1%로 두 번째, 캐나다가 4.5%로 세번째였다. 그러나 이 수치는 국내에서 수집된 악성코드 샘플을 위주로 분석하였기 때문에 국내의 수

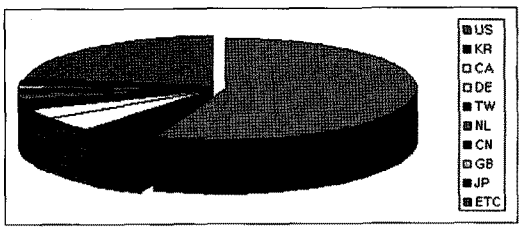


그림 8 명령/제어 서버 국가별 분류

치가 높게 나타났을 수 있다. 실제로 미국 시만텍사의 '07년 상반기 위협보고서에 따르면 한국의 명령/제어 서버 국가 순위는 4위로 나타났다[5].

명령/제어 서버가 명령을 전달하기 위한 프로그램인 IRC 프로그램이 서비스 목적으로 사용하는 포트는 TCP/80이 19.9%로 가장 높았고, TCP/6667, TCP/53 순이었다. 특히 IRC의 기본 포트인 TCP/6667, 6668, 7000 포트 보다 TCP/80의 비율이 더 높은 것을 알 수 있었다. 이는 단순히 IRC 사용 포트를 차단하는 것만으로는 명령/제어 서버로의 연결 시도를 막을 수 없다는 것을 의미하며, 침입차단시스템 등의 보안장비에서 TCP/80의 경우는 대부분 차단하고 있지 않으므로 이러한 포트를 사용함으로써 감염PC가 보안장비를 우회하여 명령/제어 서버로 연결될 수 있도록 하고 있음을 알 수 있다. 최근 급증하는 HTTP 봇의 영향으로 TCP/80 포트를 사용한다고 볼 수도 있으나, 수집한 샘플이 주로 IRC 봇이어서 그러한 가능성은 적어 보인다.

1년 동안 수집한 악성코드에서 추출된 명령/제어 서

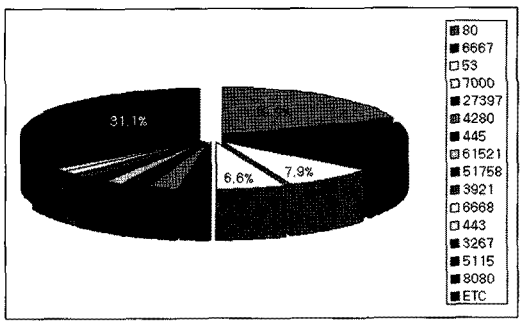


그림 9 명령/제어 서버 사용포트

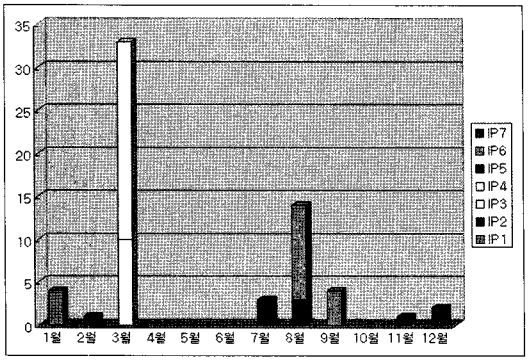


그림 10 명령/제어 서버 IP 변경 활동

버의 도메인 명을 기준으로 하여 수집된 횟수를 보면, 150회 이상 발견된 도메인도 있고, 50회 이상 발견된 도메인도 7개나 된다. 이는 악성봇 명령/제어 서버의 활동 지속시간이 상당히 길다는 것을 의미한다. 그러나 이러한 악성 도메인의 경우 해당 IP의 소유자가 자신의 서버가 해킹되어 악용되고 있음을 인지하게 되면 해커의 입장에서는 더 이상 해당 IP를 사용할 수 없게 된다. 이 경우 해커는 도메인은 그대로 두고 IP만 다른 서버로 변경하여 사용하게 되는데, 이러한 징후도 이번 연구에서 감지되었다. 악성봇 명령/제어 서버의 활동기간 동안 IP 변경 횟수를 산출하여 본 결과, 10회 이상 IP가 변경된 사례가 3개로 관찰되었다. 그림 10은 특정한 하나의 악성봇 명령/제어 서버 도메인에 대하여 IP 변경 내용 및 탐지 횟수를 나타낸 것으로, 총 7회 IP가 변경되었고, 해당 명령/제어 서버의 경우 3월에 가장 많이 탐지되었으나, 이후 활동이 없다가 다른 IP로 7월부터 다시 활동을 시작한 것을 확인할 수 있다.

명령/제어 서버에 직접 접속하여 IRC의 기본 명령어인 접속 클라이언트 숫자 확인 명령어를 통하여 접속해 있는 감염 PC 수를 측정해 본 결과, 평균적으로 2,600대의 PC가 감염되어 접속하고 있는 것으로 파악되었고, 이 중 100대-500대 PC가 감염되어 활동 중인 소규모 Botnet 이 가장 많았으나, 5,000개 이상의 PC가 접속하고 있는 대규모 Botnet도 14.1%나 존재 하였다. 최근 초고속 인터넷 서비스가 광랜 서비스로 진화하면서 상당

표 2 명령/제어 서버 규모

| 명령/제어 서버 규모 | 비율 |
|---|---|
| 5000개 이상 | 14.1% |
| 2000-5000 | 17.1% |
| 1000-2000 | 18.1% |
| 500-1000 | 11.2% |
| 100-500 | 20.1% |
| 100 이하 | 19.4% |

수의 개인 사용자들의 인터넷 환경에서 최대 100Mbps 속도가 나오고 있다. 이러한 점을 가만해 보면 5,000 대 가량의 감염 PC는 상당한 양의 트래픽을 발생시킬 수 있어, DDoS 공격 등에 악용되는 경우 공격을 당하는 피해 서버뿐만 아니라 감염 PC가 다수 존재하는 중소기업이나 아파트 단지 내 네트워크에도 영향을 미칠 수 있어 피해가 커질 우려가 있다[6,7].

## 3. 악성봇 DNS 싱크홀 구축 및 운영 결과

### 3.1 악성봇 DNS 싱크홀 시스템 구축

악성봇 감염 PC가 명령/제어 서버에 접속하기 위해서는 악성봇 서버의 도메인에 대한 IP를 얻기 위하여 감염 PC가 사용하는 DNS 서버에 질의를 하게 된다. 이 때 DNS 서버에서는 해당 도메인을 관할하는 DNS 서버에게 IP를 얻어와서 감염 PC에게 알려주고 감염 PC는 응답받은 IP로 접속하는 과정을 거치게 된다. 그러나 악성봇 DNS 싱크홀이 적용된 DNS 서버의 경우에는 사전에 악성봇 명령/제어 서버로 알려진 도메인은 감염 PC로부터 DNS 질의를 받을 때 해당 도메인을 관할하는 DNS 서버에게 물어보지 않고 직접 특정 IP(싱크홀 서버 IP)를 응답하게 되고, 감염 PC는 해커의 서버 대신에 싱크홀 서버로 접속하게 된다. 이렇게 되면 악성봇은 감염 후 해커의 명령/제어 서버에 접속하여 악성 행위 명령을 전달받는데, 명령/제어 서버로의 접속이 차단되므로 더 이상 악성 행위를 할 수 없게 된다. 악성봇에 감염된 수많은 PC 치료가 현실적으로 어려운 반면 악성봇에 감염된 PC가 유지하는 명령/제어 서버로의 연결을 차단함으로써 공격자로부터 감염 PC가 원격에서 조정되는 것을 방지하여 수많은 악성봇의 추가 활동을 무력화할 수 있으므로, DNS 싱크홀 방법은 현재 알려진 악성봇 대응방법으로는 가장 효과적인 것으로 알려져 있다. 이를 위하여 2005년 국내 8개 ISP에 적용하기 시작하여, 현재는 악성봇 명령/제어 서버 도메인 목록 약 1,200여개를 공유하고 있다.

### 3.2 악성봇 싱크홀 시스템 운영 결과

악성봇 싱크홀은 2005년부터 운영되기 시작하였지만, 통계 시스템 구축이 완료되어 운영된 2007년 데이타를 중심으로 운영결과를 분석하였다. 악성봇 싱크홀에 적용하는 악성 도메인 목록에는 수집한 악성봇 샘플에서 추출한 도메인과 국외 기관과 정보공유를 통해 추출한 도메인이 포함되어 있다. 이렇게 수집된 도메인 개수는 계속적으로 증가하여 현재 8,800개('07.12 현재) 이상의 도메인이 수집되었다. 그러나 이전에 사용하던 도메인이나 이미 차단된 도메인은 현재 접속이 불가능 하므로 매일 이를 체크하여 접속이 가능한 도메인 1,200여개('07.12 현재)만을 악성봇 싱크홀에 적용하고 있다.

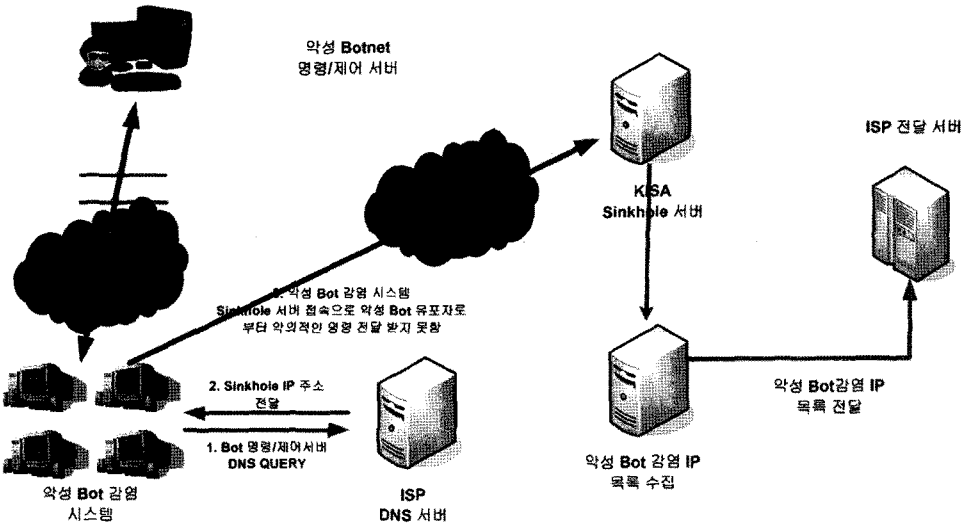


그림 11 악성봇 DNS 싱크홀 시스템

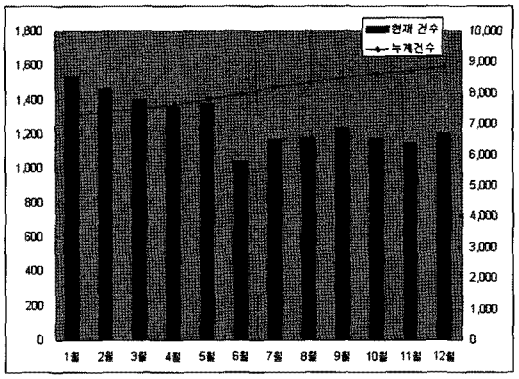


그림 12 DNS 싱크홀 적용 도메인 수

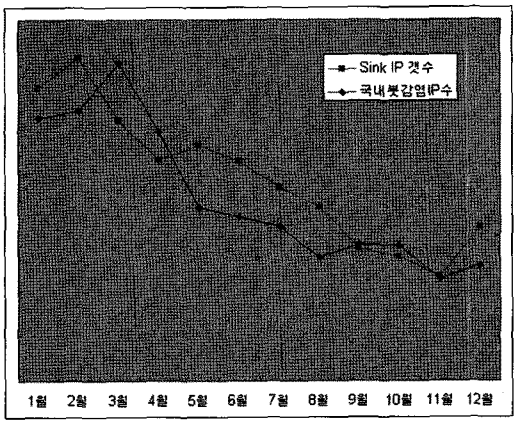


그림 13 싱크홀 IP수와 봇감염 IP수 비교

악성봇 싱크홀이 적용된 ISP의 감염 PC는 해커가 운영하는 명령/제어 서버에 접속하지 않고, 싱크홀 서버로 우회되어 접속하게 되는데, 이를 관찰하여 보면 악성봇에 감염된 IP 개수를 확인할 수 있다. 이렇게 싱크홀로 유입된 감염된 IP 개수 추이와 악성봇 감염율 추이를 살펴보면 전체적으로는 비슷한 추이를 가지나, 싱크홀로 유입된 IP 개수가 감소한 3월, 9월에는 허니넷으로 유입된 국내의 악성봇 감염 IP가 증가한 것을 살펴볼 수 있다. 이는 싱크홀 된 IP 개수가 줄어들게 되면 해커로부터 명령을 전달받아 악성행위를 수행하는 국내의 악성봇 감염 IP가 증가할 수 있다는 점을 보여준다. 그러나 최근에 다양한 HTTP 봇이 많이 감염됨에 따라 IRC 봇의 경우 전체적인 감염 IP가 줄어들고 있는 추세에 있다고 할 수 있다.

지난 2007년 2월에는 국외 ROOT DNS 서버가 DDoS 공격을 받아 일부 DNS 서버에 장애가 발생한 사건이 있었다. 공격은 12시간이 넘게 지속되었으며, 국내의 악성코드 감염 PC도 상당수가 공격에 악용되었다. 공격에 사용된 악성코드는 Virut라는 바이러스 였는데, 실행파일에 감염되는 파일바이러스 이기도 하지만 악성봇과 같이 IRC를 이용하여 해커의 명령/제어서버에 접속 후 악성행위 명령을 기다리는 형태의 바이러스 였다. Virut에 감염 시 접속하는 명령/제어 서버인 proxima.ircgalaxy.pl 도메인을 악성봇 DNS 싱크홀에 적용시켜 본 결과, 싱크홀 유입 IP 개수가 3배 가량 급증한 것을 알 수 있다. 이후 국내 ISP 등에서 해당 도메인을 차단 조치하여 이후에 접속하는 감염IP는 줄어들었지만, 만일 국내 모든 DNS 서버가 싱크홀을 적용 중이었다면 국내발 DDoS 공격은 상당부분 감소시킬 수 있었을 것이라

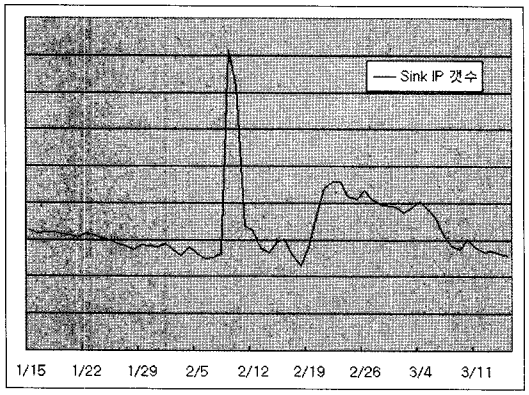


그림 14 Virut의 명령/제어 서버를 싱크홀에 적용한 결과

고 판단된다. 이에 따라 2007년에는 국내 주요 ISP에 싱크홀 적용을 완료하였고, 현재 웹호스팅 업체 및 주요 대학의 DNS 서버에도 적용을 권고하고 있다.

### 3.3 악성봇 감염 IP 분석

싱크홀로 유입된 악성봇 감염 IP를 ISP별로 분류해 본 결과, 악성봇 감염 기간은 2일 이하가 가장 많았다. 물론 초고속 인터넷 가입자의 경우 IP가 접속할 때 마다 변경되는 유동 IP 사용 고객이 다수 있지만, 최근에는 유동 IP라 하더라도 매번 새로운 IP로 바뀌는 경우는 많지 않고, 국외 보안 업체의 감염 IP 통계 등에도 IP 기준으로 통계를 산출하고 있어 같은 방식을 사용하였다. 국내 ISP 중 초고속 인터넷 서비스를 하고 있는 주요 ISP의 경우 악성봇 감염 후 1주일이 지나면 85% 가량이 치료가 되는 것으로 나타났고, 특히 B ISP의 경우 3일이 지나면 80% 가량이 치료가 되는 것으로 나타났다. B ISP는 2007년에 전용 무료 백신 보급 등을 통해 악성봇 치료에 많은 노력을 기울여 다른 ISP 에 비해 빠른 치료가 되는 것으로 추정된다. 반면 CATV 인터넷 사업자의 경우 2일 이하 치료되는 숫자보다 1주일 가량 감염이 지속되는 사례가 오히려 증가한 것을 확인할 수 있는데, CATV 가입자는 상대적으로 무료백신을 사용하는 수가 적다는 것이 영향이 있는 것으로 보인다.

악성봇 감염율은 허니넷에 유입되는 IP 중 악성봇이 주로 사용하는 18개 포트를 목적지 포트로 하는 IP만을 추려낸 후, 전체 유입 IP 중 국내 IP 수의 비율을 산정한 것이다. 악성봇의 전파에 이용되는 주요 포트 목록은 악성봇 소스코드에 명시되어 있는 취약점 공격 모듈을 중심으로 작성한 것으로, 악성봇에서는 각각의 취약점에 대하여 별도로 감염전파를 위한 공격을 수행하거나 모든 취약점에 대한 공격을 동시에 진행 할 수도 있다. 따라서 허니넷에 유입되는 트래픽 중 이러한 취약점 포트에 대한 공격은 악성봇이 감염시도를 위해 발생시키는 트래픽이라고 볼 수 있으며, 정상적인 경우 유입 트래픽

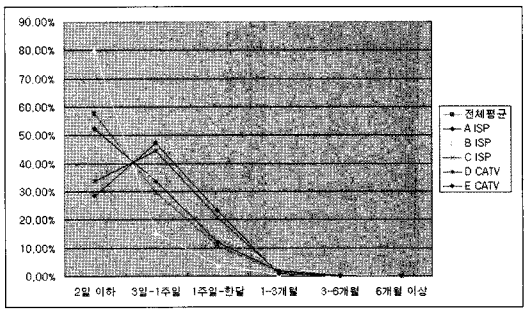


그림 15 ISP 별 봇 감염 지속시간

표 3 악성봇의 전파에 이용되는 주요 포트 목록

| 포트 | 관련 취약점 및 웜/악성 Bot |
|---|---|
| 23 | Cisco Telnet |
| 80 | WebDAV, ASN.1-HTTP, Cisco HTTP |
| 135 | DCOM, DCOM2 |
| 139 | NetBIOS, ASN.1-NT |
| 143 | IMail |
| 445 | NetBIOS, LSASS, WksSvc, ASN.1-SMB, DCOM |
| 903 | NetDevil |
| 1025 | DCOM |
| 1433 | MS-SQL |
| 2967 | Symantec Exploit |
| 2745 | Bagle, Bagle2 |
| 3127 | MyDoom |
| 3140 | Optix |
| 5000 | UPNP |
| 6101 | Veritas Backup Exec |
| 6129 | Dameware |
| 17300 | Kuang2 |
| 27347 | Sub7 |

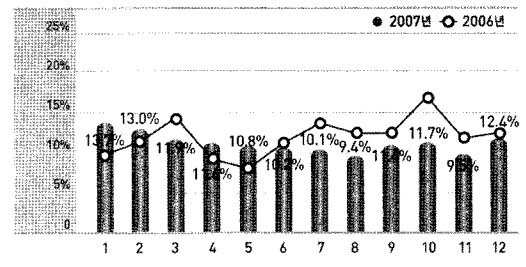


그림 16 악성봇 감염율

이 없는 허니넷에서 이를 탐지함으로써 전체 국내 인터넷 환경에서의 악성봇 관련 트래픽을 유추해 볼 수 있다.

악성봇 감염율은 2005년부터 산출을 시작하여, 2005년에는 평균 18.8%였던 감염율이 2006년에는 12.5%, 2007년에는 11.3%로 감소되었다. 국외 미국 시만텍 보고서에서도 2006년 6위였던 악성봇 감염 국가 순위가 2007년에는 9위로 하락하였다. 이와 같이 악성봇 감염율

이 하락하고 있는 원인은 국내의 악성봇 DNS 싱크홀 적용에 있는 것으로 판단되며, 향후에 웹호스팅 업체나 대학의 DNS 서버에 적용을 확대시킨다면 감염율을 좀 더 낮출 수 있을 것이다.

## 4. 일본의 CCC 프로젝트와의 비교 분석

일본에서는 악성봇 대응을 위하여 정부차원의 CCC (Cyber Clean Center)[8] 프로젝트를 수행중이다. CCC 프로젝트는 ISP 및 백신사와의 협조를 통해 운영되고 있다. 운영방법은 허니넷을 공격하여 악성봇을 전파한 IP 목록을 ISP에 전달하여 ISP가 해당 가입자에게 감염사실을 알리면, 가입자는 악성봇 치료 홈페이지에 접속하여 백신 프로그램을 다운로드 받아 치료하는 방식이다. 그러나 이러한 방식은 ISP로부터 감염사실을 통보받은 가입자가 치료 홈페이지에 직접 접속하여 치료를 해야 하므로, 가입자의 적극적인 참여가 필요해 일본에서도 통보받은 가입자의 30%만이 치료 홈페이지에 접속하여 백신을 다운로드 받고 있으며 실제 다운로드된 백신으로 완전히 치료가 되었는지도 확인하기 어려워 보인다.

이에 비해 악성봇 싱크홀 방식은 감염자의 참여가 없이도 ISP의 협조만으로도 악성봇에 의한 악성행위를 차단할 수 있다는 장점이 있으며, 실제 악성봇 감염자의 상당수가 보안에 대한 지식이 부족하고 백신 프로그램을 사용하지 않는 점을 고려해 보면 보다 효과적인 방법으로 판단된다. 그러나 악성봇 싱크홀 방식에서는 감염자 PC에 보안 취약점이 여전히 남아 있어 또 다른 악성봇에 감염될 가능성이 있으므로, CCC 프로젝트와 같은 감염자 PC 자체의 취약점 제거 방안이 향후에 마련되어야 할 것으로 보인다.

## 5. 결론 및 향후 연구방향

악성봇은 해커가 원격에서 조정을 통해 추가적인 악성코드 다운로드, 스팸메일 발송, DDoS 공격 등 다양한 악성행위가 가능해 이전과 달리 금전적 이득을 목적으로 하는 해킹 공격에 많이 사용되고 있다. 특히나 악성봇은 소스코드가 공개되어 약간의 수정만으로도 다양한 변종이 탄생하므로 백신 프로그램만으로 피해를 방지하기에는 한계가 있다. 이러한 점에서 악성봇에 해커의 명령을 전달하는 명령/제어 서버 차단은 피해방지를 위해 중요하다. 본 논문에서 제시한 악성봇 DNS 싱크홀 기법을 직접 운영해본 결과 Virut 바이러스 등의 피해방지에 큰 효과를 보는 등 악성봇 감염 PC의 차단 및 활동제어에는 효과적인 기법임이 입증되었다.

본 논문에서는 1년 이상의 장기간에 걸쳐 Botnet을 관찰하여 그 특성을 파악하고자 하였고, 악성봇 DNS 싱크홀 기법 역시 1년 이상의 적용 결과를 바탕으로 하여 그 효과를 검증하였다. 악성봇 DNS 싱크홀 기법을 통하여 현재 문제가 되고 있는 DDoS 공격이나, 다량의 스팸메일 전송 등 악성행위를 상당부분 감소시킬 수 있으므로, DNS 서버를 운영하는 ISP, CATV/SO, 웹호스팅 업체, 대학 등에서는 도입을 적극적으로 검토하여야 할것으로 판단된다. 또한 국내에서 적용되고 있는 봇 대응 기법의 정당성을 제시하였다.

향후의 연구방향으로는 최근에는 BlackEnergy Bot 등 증가하고 있는 HTTP 봇에 대한 대응 방안 마련이 필요하다. 기존의 악성봇 명령/제어 서버는 주로 인터넷 채팅 프로토콜(IRC)을 이용하여 구성되었다. 그러나 최근에는 IRC 대신 HTTP 프로토콜을 이용하여 구성되고 있다. 이와 같은 방식으로 전파된 HTTP 봇의 경우

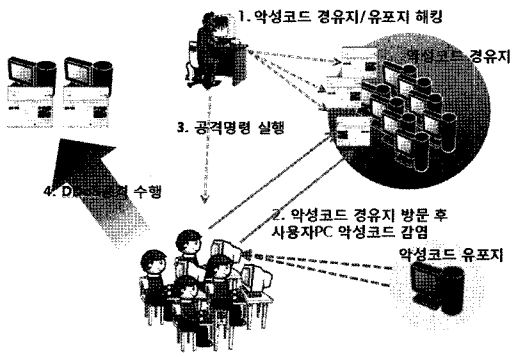


그림 17 HTTP 봇을 이용한 DDoS 공격

표 4 악성봇 싱크홀과 CCC 프로젝트 비교

| | 악성봇 싱크홀 | CCC 프로젝트 |
|---|---|---|
| 적용대상 | 참여한 ISP의 가입자 대부분에 적용(능동적) | 참여한 ISP 중 감염된 가입자에게 e-mail을 통해 알림(피동적) |
| 적용방법 | 참여한 ISP 가입자에 필수적용 | 참여한 ISP 중 감염치료를 희망하는 가입자가 백신 치료(감염자의 30%) |
| 치료결과 | 명령/제어 서버에 접속하지 못하므로 악성행위 불가능 | 백신치료에 의존하고 있어 치료하지 못하는 경우가 발생할 수 있음 |
| 재발여부 | 취약점이 여전히 존재하여 또 다른 악성봇에 재감염 가능 | 백신치료 후 보안 업데이트 등의 절차로 인해 재감염 가능성이 적음 |

DDoS 공격 등에 특히 빈번하게 사용되고 있다. HTTP 봇의 경우 IRC 봇과 달리 윈도우즈 취약점을 직접 공격하기 보다는 유명 홈페이지에 악성코드를 은닉하는 방법을 통해 사용자가 유명 홈페이지에 접속시 자동으로 다운로드되어 감염되도록 하는 방법을 사용하고 있다. 이러한 경우 기존의 허니넷으로는 악성코드 샘플을 수집하기 힘드므로, 가상의 agent가 주요 홈페이지를 반복적으로 자동 방문하여 악성코드 샘플을 다운로드 하는 형태의 시스템 운영이 필요하다. 현재 오픈소스 그룹에서 개발한 agent 프로그램을 도입/수정하여 시험적으로 악성코드 샘플을 수집중이다[9].

또한 명령/제어 서버가 존재하는 않는 Storm Bot (Peacomm)과 같은 형태의 P2P 봇의 경우 IRC 봇이나 HTTP 봇에 비하여 대응 방안이 좀 더 어렵다고 할 수 있다. HTTP 봇의 경우 악성봇 DNS 싱크홀 기법을 통하여 명령/제어 서버를 우선 차단하고 향후의 변화 방향을 연구하여야 할 것 이며, P2P 봇의 경우에도 실제 공격사례 등을 중심으로 대응방안 연구가 필요하다.

## 참 고 문 헌


[1] 한국정보보호진흥원, "2007 정보시스템 해킹·바이러스 현황 및 대응", *한국정보보호진흥원*, pp. 41-73, 2007.

[2] P. Baecher, M. Koeetter, T. Holz, M. Dornseif, and F. C. Freiling, "The nepenthes platform: An efficient approach to collect malware," In *Proceedings of 9th Symposium on Recent Advanses in Intrusion Detection(RAID'06)*, pages 165-184, 2006.

[3] F. Freiling, T. Holz, and G.Wicherski, "Botnet Tracking: Exploring a Root-Cause Methodology to Prevent Distributed Denial-of-Service Attacks," In *Proceedings of 10th European Symposium On Research In Computer Security(ESORICS05)*. Springer, July 2005.

[4] NORMAN Sandbox Information Center, Internet: *http://www.norman.com/microsites/nsic/*

[5] SYMANTEC, "Symantec Internet Security Threat Report," *SYMANTEC*, Sep. 2007.

[6] Jianwei Zhuge, Thorsten Holz, Xinhui Han, Jinpeng Guo, Wei Zou, "Characterizing the IRC-based Botnet Phenomenon," Reihe Informatik. TR-2007-010, December 3, 2007.

[7] The Honeynet Project. "Know Your Enemy: Tracking Botnets," March 2005. Internet: http://www.honeynet.org/papers/bots/.

[8] Cyber Clean Center, Dec 2007, Internet: *https://www.ccc.go.jp/*

[9] M. A. Rajab, J. Zarfoss, F. Monrose, and A. Terzis. "My botnet is bigger than yours(maybe, better than yours)," In *Proceedings of HotBots'07*, 2007.



김 영 백

1995년 2월 순천향대학교 정보통신공학과 졸업. 1997년 2월 순천향대학교 정보통신공학과 석사. 1996년 12월~2000년 3월 한전KDN 주임. 2002년 9월~현재 순천향대학교 정보보호학과 박사과정(수료). 2000년 4월~현재 KISA 인터넷침해사고대응지원센터 해킹대응팀 선임연구원. 관심분야는 인터넷침해사고대응, 정보보호



이 동 련

1999년 2월 아주대학교 정보및컴퓨터공학 졸업. 2001년 2월 아주대학교 정보통신전문대학원 석사. 2000년 12월~현재 KISA 인터넷침해사고대응지원센터 해킹대응팀 주임연구원. 관심분야는 인터넷침해사고대응, 정보보호



최 중 섭

1993년 2월 인천대학교 전자계산학과 졸업. 1995년 8월 숭실대학교 대학원 컴퓨터학과 석사. 2000년 8월 숭실대학교 대학원 컴퓨터학과 박사. 1986년 1월~1994년 3월 대우통신(주) 연구원. 1995년 6월~1996년 2월 한국전산원 연구원. 2000년 7월~현재 KISA 인터넷침해사고대응지원센터 해킹대응팀 팀장. 관심분야는 인터넷침해사고대응, 정보보호



염 흥 열

1981년 2월 한양대학교 전자공학과 졸업. 1983년 9월 한양대학교 전자공학과 석사. 1990년 2월 한양대학교 전자공학과 박사. 1982년 12월~1990년 9월 한국전자통신연구소 선임연구원. 1990년 9월~현재 순천향대학교 공과대학 정보보호학과 정교수. 1997년 3월~2000년 3월 순천향대학교 산업기술연구소 소장. 2000년 4월~2006년 2월 순천향대학교 산학연컨소시엄센터 소장. 1997년 3월~현재 한국통신정보보호학회 총무이사, 학술이사, 교육이사, (현)논문지편집위원장, (현) 상임부회장. 2003년 9월~2004년 3월 ITU-T SG17/Q10 Associate Rapporteur. 2004년 3월~현재 ITU-T SG17/Q9 Rapporteur. 2008년 10월~현재 ITU-T SG17 부의장 2005년 3월~현재 국내 ITU-T SG17 국내 분과위원회 의장. 2006년 11월~2008년 2월 (구)정보통신부 정책자문단 정보보호 PM. 2006년 11월~현재 (현) IITA 정보보호 PM. 관심분야는 네트워크 보안, IPTV 보안, USN 보안, 홈네트워크 보안, 응용보안, 이동통신보안