웹 기반 악성코드 유포공격의 특성 분석

유대훈, 김지상, 조혜선, 박해룡
한국인터넷진흥원

요 약

인터넷의 사용이 증가하며, 웹을 통한 악성코드유포가 주요 위협으로 등장하였다. 본고에서는 인터넷을 통한 악성코드 유포방법 중 가장 대표적 공격방법이 웹 기반 악성코드 유포공격의 특성을 분석한다.

Ⅰ. 서 론

인터넷과 웹의 사용은 해가 갈수록 증가하고 있다. 1993년 130개이던 웹 페이지의 개수는 2013년 1월 6억 7천만개를 넘었으며, 1400만명이 약간 넘던 인터넷 사용인구는 27억명 이상으로 증가하였다[1].

표 1. 웹사이트와 인터넷사용인구 수

년도(1월)	웹사이트 수	인터넷 사용인구 수
2013	672,985,183	2,756,198,420
2011	346,004,403	2,282,955,130
2009	238,027,855	1,766,206,240
2007	121,892,559	1,373,327,790
2005	64,780,617	1,027,580,990
2003	40,912,332	778,555,680
2001	29,254,370	500,609,240
1999	3,177,453	280,866,670
1997	1,117,255	120,758,310
1995	23,500	44,838,900
1993	130	14,161,570

위와 같이 인터넷사용인구가 증가하며, 인터넷을 통한 악성코드 유포가 중대한 위협으로 등장하였다. 인터넷을 통한 악성코드 유포방법에는 Drive-by download, 이메일, 업데이트서버 해킹 등이 있다. 다음은 각 공격방법과 그로 인한 주요사건을 나타낸 것이다.

표 2. 악성코드 유포공격방법과 대표사건

공격방법	설명	대표사건
Drive-by Download	웹사이트에 접속시 악성코드 설치	
이메일	악성코드를 이메일에 첨부하여 열어보도록 유도하여 악성코드 설치	RSA 해킹
업데이트 서버 해킹	S/W 업데이트 서버 해킹을 통해 악성코드 설치	SK컴즈 해킹

본고에서는 이러한 악성코드 유포방법 중 가장 널리 사용되는 Drive-by Download 공격방법의 특성을 알아보고자 한다.

Ⅱ. 관련 연구 동향

Microsoft사가 2012년 4월 발표한 자료에 의하면, 한국의 웹사이트 1천개중 17개는 악성코드 유포지로 악용되고 있으며 이는 전 세계 3위에 해당하는 수치이다[2]. 세계의 평균 악성코드 유포율이 10.85개인 것을 고려하면 한국의 악성코드 유포율이 높음을 알 수 있다. 이는 한국의 웹사이트가 공격의 대상이 되거나, 악성코드 유포에 악용되는 경우가 많음을 뜻한다.

Drive-by-download공격을 탐지하기 위한 기존의연구는 악성코드 유포지 분석이 주류를 이루었다. 기존의 연구는 크게 정적분석과 동적분석으로 나뉜다. 정적 분석 방식의 경우 Halfond와 Orso가 SQL Injection 공격으로 삽입된 코드를 탐지할 수 있는 기술을 제안하였고[3], Marco Cova는 정상적인 자바스크립트 코드의 특징을 이용한 기계학습을 통해 Drive-by download에 사용되는 악성 스크립트 코드를 탐지할 수 있는 기술을 제안하였다[4]. 그러나 정적분석을 통한 탐지방법은 공격패턴이 변화될 때마다 패턴을 변경해야 하는 단점이 있으며 악성 스크립트 코드 난독화에 따라 많은 오탐이 발생할 수 있다는 한계점이 존재한다. 이를 보완하기 위해 허니팟(honeypot)을 이용하여 웹 페이지 방문 시 발생하는 악성행위를 분석하고 탐지하는 방향으로 동적분석이 이루어졌으나, 동

적분석 방식의 탐지방법은 시스템 자원소모가 크고 전체 분석/탐지 시간도 길어지는 단점이 있다.

최근, 해커들은 악성코드 유포 사이트를 대규모로 확장해 거대한 악성코드 전파용 네트워크를 운영 중에 있다. 이러한 공격을 막기위해 유포지 분석 뿐만이 아니라, 경유지/공격코드 등의 감염경로를 분석하여 하나의 네트워크로 분석하는 연구가 활발히 진행 중이다. 한국과학기술원 논문에서는, 웹 페이지 내 포함된 난독화/은닉화 된 스크립트를 식별하는 멀티레벨 에뮬레이션을 이용하여 악성코드 유포 네트워크를 분석하는 기술을 제안하였다[5].

Ⅲ. Drive-by Download 공격의 특성

1. Drive-by Download 공격의 개요

Drive-by Download 공격은 다음과 같은 과정을 통해 이루어진다.

① 피해자는 리다이렉션 코드가 삽입된 경유지에 접속한다.
② 경유지에 삽입된 리다이렉션 코드는 취약점 공격코드가 삽입된 공격지를 호출한다.
③ 공격지의 공격코드는 피해자의 시스템의 취약점을 이용하여 적절한 권한을 획득한다.
④ 공격지에 삽입된 리다이렉션 코드는 악성코드가 삽입된 유포지를 호출한다.
⑤ 이를 이용하여 유포지에서 악성코드를 다운로드하고 실행시킨다.

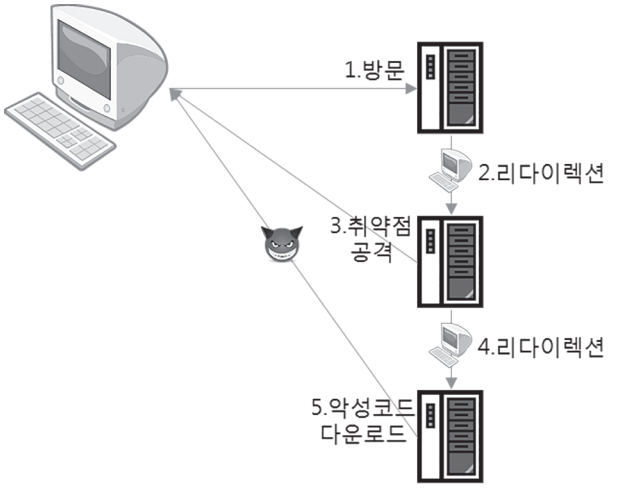

그림 1. Drive-by Download 공격

2. 공격에 사용되는 공격코드의 특성

최근Microsoft사에서 분석한 취약점 악용유형을 살펴보면, 아래와 같이 웹을 이용한 유형 (HTML과 JavaScript 악용유형)이 가장 많은 비중을 차지하며, OS 및 문서취약점이 다음 순위인 것으로 확인된다.

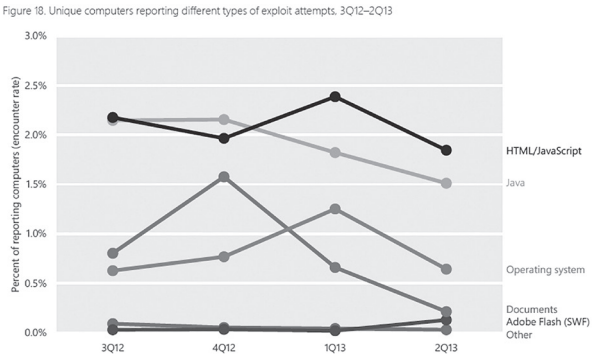

그림 2. 최근 취약점 탐지현황[6]

웹을 통한 악성코드 유포유형에 대한 보다 상세한 동향파악을 위해 국내 웹사이트를 통하여 악성코드를 유포하는 URL을 샘플링하여 분석 한 후 유형을 분류하였다. 분석결과 JAVA가 59%로 가장 많았으며, Internet Browser 가 36%, SWF가 5% 순으로 나타났다.

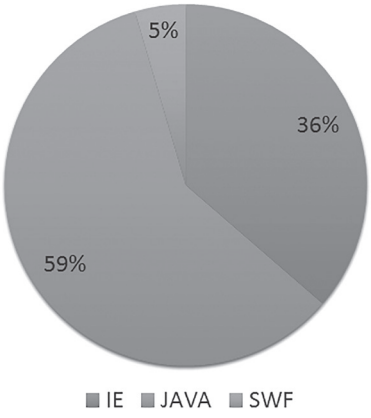

그림 3. 애플리케이션 별 취약점 현황

또한 해당 어플리케이션 취약점을 공격하는데 사용된 공격코드를 살펴본 결과, 대부분 난독화되어 있었으며, 난독화 방법은 대표적으로 Escape 문자열을 사용한 방법 및 자바스크립트의 변수나 함수명의 랜덤화와 문자열 패킹을 사용한 난독화 방법이 사용되었다. 이 기법은 악성코드 유포를 위해 삽입된 자바스크립트 코드 내의 변수와 함수명을 랜덤 문자열로 치환하고 이 문자열을 패킹하여 해독을 어렵게 만든다.

최종 삽입 될 HTML 코드:

```
<iframe src=http://advtds.fastfind.info/advtds/out.php?s_id=20 width=1 height=1
style="display:none"></iframe>
```

난독화 된 자바스크립트 코드:

```
function BD37A78D25DEEF10B10A677B5F0(B9D5D6B429B3B9BD29A08C8){
return(parseInt(B9D5D6B429B3B9BD29A08C8,16));}function
D5281A4C55A9736772D3539EA51(D6242D36DFD76213ED900E11FDA){function
C56A17251C947C7EF(){var D83D6CE95B0A38CD6F=2;return
D83D6CE95B0A38CD6F;}var D71C351C9A9105908A5D4D9624954="";for(
CEDB124A2EA9FE61EB10A584FE0E8=0;CEDB124A2EA9FE61EB10A584FE0E8<lt;
D6242D36DFD76213ED900E11FDA.length;CEDB124A2EA9FE61EB10A584FE0E8++=
C56A17251C947C7EF()){D71C351C9A9105908A5D4D9624954+=
(String.fromCharCode(BD37A78D25DEEF10B10A677B5F0(
D6242D36DFD76213ED900E11FDA.substr(CEDB124A2EA9FE61EB10A584FE0E8,
C56A17251C947C7EF())));} document.write(D71C351C9A9105908A5D4D9624954);
}D5281A4C55A9736772D3539EA51("3C696672616D65207372633D687474703A2F2F
6164767464732E666173746696E642E696E666F2F6164767464732F6F75742E7068
703F735F69643D32302077696474683D31206865696768743D31207374796C653D22
646973706C61793A6E6F6E65223E3C2F696672616D653E");
```

그림 4. 공격코드 난독화 예

네트워크 기반 탐지나 백신의 경우 공격코드를 탐지하기 위해 시그니처 방법을 주로 사용하는데, 이러한 난독화 기법은 네트워크 기반 탐지 및 백신탐지를 어렵게 한다. 실제 탐지된 공격코드를 곧바로 다수의 백신으로 점검해 본 결과, 많은 백신이 진단하지 못하는 것으로 확인되었다.

그림 5. 난독화 공격코드 백신탐지 결과 예

난독화된 코드를 복호화 하여 악용된 취약점을 자세히 확인해 본 결과, 악용 취약점은 CVE-2013-0422, CVE-2013-2465, CVE-2012-1889 등인 것으로 확인되었다.

표 3. 악용된 취약점 상세내역

CVE Num	대상	내용
CVE-2012-0507	JAVA Applet	AtomicReferenceArray.set()함수 취약점을 통한 원격코드 실행
CVE-2012-1723	JAVA Applet	JIT 컴파일을 할 때 부정확한 코드 검증 취약점을 통한 원격코드 실행
CVE-2012-4681	JAVA Applet	"acc" private field를 리플렉션의 권한 변경을 통한 원격코드 실행
CVE-2012-5076	JAVA Applet	com.sun.org.glassfish.gmbal.* 패키지 취약점을 통한 원격코드 실행
CVE-2013-0422	JAVA Applet	리플렉션 함수들이 재귀적으로 호출을 통한 원격코드 실행
CVE-2013-2465	JAVA Applet	storeImageArray()함수 취약점을 이용한 원격코드 실행
CVE-2013-0634	Adobe Flash Player	플래시 클래스에 버퍼오버플로우 취약점을 이용한 원격코드 실행
CVE-2012-1889	Microsoft Internet Explorer(6/7)	IE XML 코어 서비스 취약점을 이용한 원격코드 실행
CVE-2012-1889	Microsoft Internet Explorer(8/9)	IE XML 코어 서비스 취약점을 이용한 원격코드 실행
CVE-2012-1875	Microsoft Internet Explorer	IE 8.0 동일 ID 속성 취약점을 이용한 원격 코드실행
CVE-2010-0806	Microsoft Internet Explorer	IE 6/7에 대한 비유포 포인트에 대한 액세스 요청 원격코드 실행
CVE-2012-1876	Microsoft Internet Explorer	IE 6~9 미리보기 취약점(힙 오버플로우)를 이용한 원격코드 실행

또한, 공격자는 Driver-by Download공격 시, 공격 성공률을 높이기 위해서 아래와 같이 여러가지 취약점을 복합적으로 이용하는 다중 공격방식을 취하므로, 감염방지를 위해서는 웹 브라우저 및 JAVA, Flash Player에 대한 보안 업데이트 관리를 동시에 철저하게 하는 것이 필요하다.

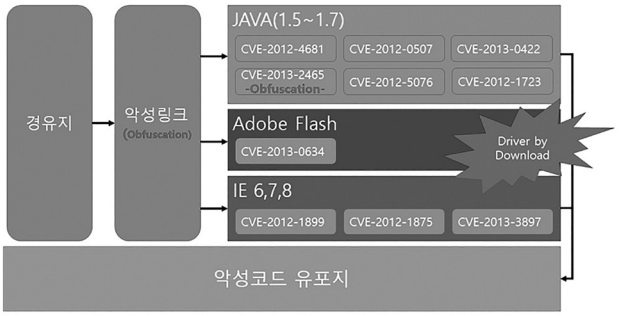

그림 6. 취약점 공격 개요

3. 공격에 사용되는 주소의 특성

3.1 악성코드 유포그룹의 개요

최근 발생하는 상당수의 공격들은 웹을 통한 악성코드 유포방법인 Drive by Download 공격에 의해 발생하고 있으며 이러한 피해는 점점 심각해지고 있다. 웹을 통해 전파되는 악성코드는 악성코드 유포그룹이라 불리는 거대 네트워크를 통해 사용자에게 전파되며, 악성코드 유포그룹은 사용자가 처음 접속하는 경유지 사이트, 취약점을 이용한 공격코드를 다운로드 하는 Exploit site 그리고 마지막으로 악성코드 배포에 악용되는 악성코드 유포지 사이트로 구성되어 있다. 이 악성코드 유포그룹의 구조와 특성을 파악하고 이해하는 것은 악성코드 유포그룹의 활동을 탐지하고 예측한다는 면에서 그 의미가 크다. 본고에서는 2013년 1년 동안 수집된 국내 180만개 주요 사이트를 통해 유포된 악성코드, 유포지, Exploit site, 그리고 경유지 사이트의 탐지 결과를 기반으로 악성코드 유포그룹의 구조와 특성을 파악 한다.

3.2 악성코드 유포에 사용되는 주소들의 재악용

[7]에 따르면 탐지된 사이트의 상당수가 2번 이상 탐지되지 않았으며, 특히 탐지된 경유지의 경우 2번 이상 탐지되지 않은 사이트의 비율이 53%로 과반수가 넘으며, 탐지 횟수가 2회 이하인 경우는 경유지 사이트 전체의 78%에 달한다. 이는 경유지의 대부분이 재사용 되지 않으며 그 이유는 경유지 사이트를 계속 변경하여 악성코드 유포그룹의 추적을 피하고 경유지를 다양화하여 보다 많은 PC들에 악성코드의 감염을 확산하려는 의도로 추정된다. 또한 탐지된 전체 사이트 중 18%는 Exploit

site와 유포지 모두로 사용되거나 경유지와 Exploit site 모두로 사용되는 등 중복해서 사용되었으며 이는 공격자에 의해 감염된 사이트의 일부가 악성코드 전파에 중복 재사용됨을 의미한다.

Exploit site와 악성코드 유포지는 경유지 사이트에 비하여 상대적으로 재사용되는 비율이 높다. 이는 악성코드 유포그룹이 보다 많은 PC에 악성코드를 감염시킬 뿐만 아니라 탐지되는 것을 회피하기 위해 상당수의 경유지 사이트를 계속 변경하는 것으로 판단된다. 반면 공격자가 직접 공격코드를 심는 Exploit 사이트나 악성코드를 유포하는 악성코드 유포지의 재사용비율이 비교적 높다는 것은 exploit site나 유포지는 공격자에 의해 관리된다는 것을 의미한다.

3.3 악성코드 유포에 사용되는 주소들의 활동기간

앞서 설명과 같이 [7]에 따르면 동일한 사이트가 재악용되는 비율은 Exploit site나 유포지에 비해 경유지가 높은 경향을 보이는 반면 그 활동 기간이 하루인 사이트의 비율은 경유지 사이트에 비해 Exploit site나 악성코드 유포지가 월등히 높게 나타난다. 이는 악성코드를 유포하는 악성코드 유포그룹의 구성비가 경유지, Exploit site 그리고 유포지의 순으로 낮아지는 것과 같다. 〈그림 7〉은 경유지, Exploit site 유포지 그리고 악성코드의 각 사이트별 활동 기간을 통계적으로 도시화 한 것이다. 〈그림 7〉에서 보는 바와 같이 악성코드의 활동기간이 악성코드 유포그룹 사이트들에 비해 상대적으로 긴 이유는 악성코드는 다른 악성코드 유포그룹 사이트들에 비해 새로운 악성코드를 제작하는데 필요한 시간 및 비용이 상대적으로 더 크기 때문으로 추정할 수 있다.

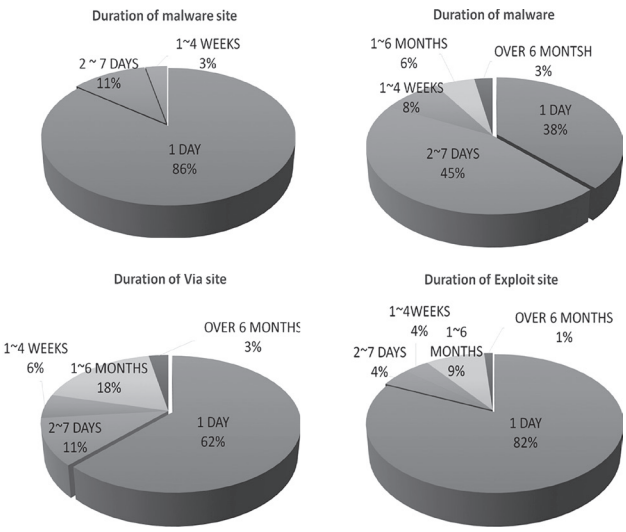

그림 7. 악성코드 유포사건의 요소별 활동기간[7]

3.4 악성코드 유포에 사용되는 주소들의 활동기간와 탐지 횟수 간 상관관계

[7]에 따르면 경유지의 경우 활동 기간과 탐지 횟수에는 상대적으로 높은 상관성을 보보이나 대부분 사이트에서는 활동 기간과 탐지 횟수는 큰 상관관계는 보이지 않는다. 〈그림 8〉는 활동기간별 탐지 횟수를 보이며 가로축은 각 사이트의 활동기간을 세로축은 각 사이트의 탐지 횟수를 나타낸다. 〈그림 8〉에서 보듯 유포지는 악성코드 경유지와 Exploit site에 비해 활동 기간이 매우 짧으며 소규모로 운영됨을 알 수 있다.

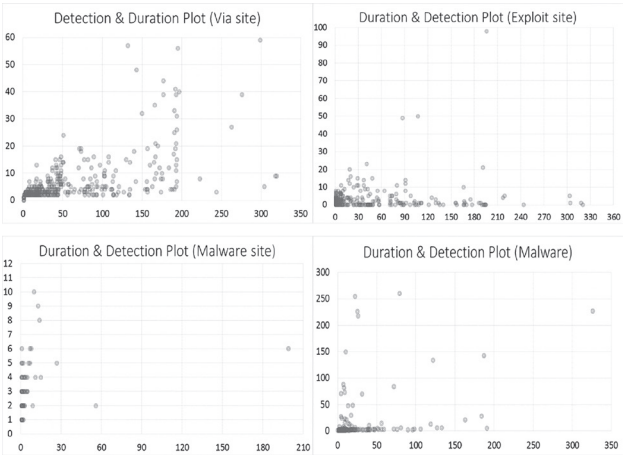

그림 8. 악성코드 유포사건의 요소별 활동기간-탐지횟수 관계[7]

3.5 악성코드 유포에 사용되는 주소들의 활동기간와 탐지 횟수 간 상관관계

〈그림 9〉은 악성코드 재배포율을 나타낸 것으로 각 유포지 사이트 별로 동일한 악성코드가 재 배포되는 비율을 나타내는 값을 의미 한다. 〈그림 9〉에서 보듯 활동 일수가 하루 이상인 경유지의 84%는 매번 다른 악성코드를 유포함을 의미한다. 이는 악성코드 유포그룹이 시시각각으로 변화하며 새로운 악성코드를 유포하는 것으로 볼 수 있다.

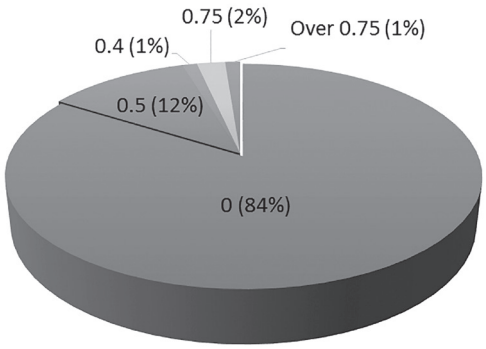

그림 9. 악성코드 재배포율

Ⅳ. 결 론

인터넷의 증가에 따라 인터넷을 통한 침해사고가 증가하고 있으며, 이 중 가장 주요한 경로가 악성코드 유포이다. 악성코드 유포방법 중 가장 널리 사용되는 것이 웹사이트를 방문한 사용자의 취약점을 공격하여 악성코드를 설치하는 Drive-by Download 공격이다.

앞에서 알아본 바와 같이 악성코드 유포를 위한 공격코드는 대부분 Escape 문자열, 패킹기법 등을 이용하여 난독화되어 있으며, 여러 취약점을 복합적으로 이용하였다. 또한 악성코드 유포에 사용되는 경유지는 재사용되는 비율이 낮은 반면 유포지는 재사용되는 비율이 높고 활동기간이 짧아 공격자에 의해 집중 관리됨을 알 수 있었다. 그리고 하나의 유포지에서 동일한 악성코드를 유포하는 비율이 낮아 악성코드 유포그룹이 계속해서 변화하며 새로운 악성코드를 유포함을 알 수 있었다.

이와 같이 Drive-by Download 공격은 계속해서 변화하는 악성코드 유포그룹에 의해 진행되므로 악성URL을 차단하거나 시그니처 방식의 악성코드 차단만으로는 효과적으로 대응하기 어렵다. 따라서 기존에 알려진 악성 URL을 차단하는 것 외에도 신규 악성 URL을 단시간내에 탐지하여 선제적으로 차단해야 한다. 또한 취약점이 알려지는 시점과 해당 취약점에 대한 패치가 나오기 전에 이를 악용하는 제로데이공격의 시작 시점간의 기간이 점점 짧아짐에 따라 이에 대응할 수 있는 연구가 더욱 진행되어야 할 것이다.

참 고 문 헌

[1] http://www.internetlivestats.com/total-number-of-websites

[2] http://www.bloter.net/archives/152202

[3] W.G.J. Halfond and A. Orso, "Amnesia: analysis and monitoring for neutralizing sql-injection attacks", Proceedings of the 20th IEEE/ACM international Conference on Automated software engineeringm, page 174-183, 2005

[4] Marco Cova, Christopher Kruegel and Giovanni Vigna, "Detection and Analysis of Drive-by-Download Attacks and Malicious JavaScript Code", 2010

[5] 최상용, 강익선, 김대혁, 노봉남, 김용민, "Multi-Level Emulation for Malware Distribution Networks Analysis", 2013

[6] Microsoft Security Intelligence Report

[7] Han Young-Il, Lee Tae-Jin, and Park Hea-Ryong. "Structural and characteristic analysis of malware network." Korean Institute of Communications and Information Science 2014. Winter, 2014 Feb. 22.

약 력

유 대 훈

2006년 한양대학교 이학사
2013년 한양대학교 이학석사
2013년~현재 한국인터넷진흥원 정보보호기술 개발팀
관심분야: 암호알고리즘, 침해사고 탐지 및 분석

김 지 상

2000년 아주대학교 학사
2000년~2003년 LG전자 대리
2003년~현재 KISA 책임연구원
관심분야: 해킹 및 침해대응 분석

조 혜 선

2013년 세종대학교 컴퓨터공학과 학사
2013년~현재 한국인터넷진흥원 정보보호기술 개발팀
관심분야: 악성코드, 네트워크보안, 시스템보안

박 해 룡

1999년 전남대학교 이학사
2001년 서울대학교 이학석사
2006년 전남대학교 공학박사
2000년~현재 한국인터넷진흥원 정보보호기술개발 팀장
관심분야: 악성코드 탐지 및 분석, 클라우드 보안, 암호 알고리즘 설계 및 분석