

오픈소스 보안주의보

(주)비디에스케이 기술사업팀

김혜영 차장

hykim@bdsk.co.kr

1. 기업과 오픈소스

2. 2018 오픈소스 보안과 리스크 분석

3. 오픈소스 보안취약점 피해사례

4. 오픈소스 보안취약점 관리방안



1. 기업과 오픈소스

보안뉴스

2018-09-07 / 보안뉴스 원병철 기자

국내 기업 61%, 매일 5천 건 이상의 사이버 보안 경고에 노출

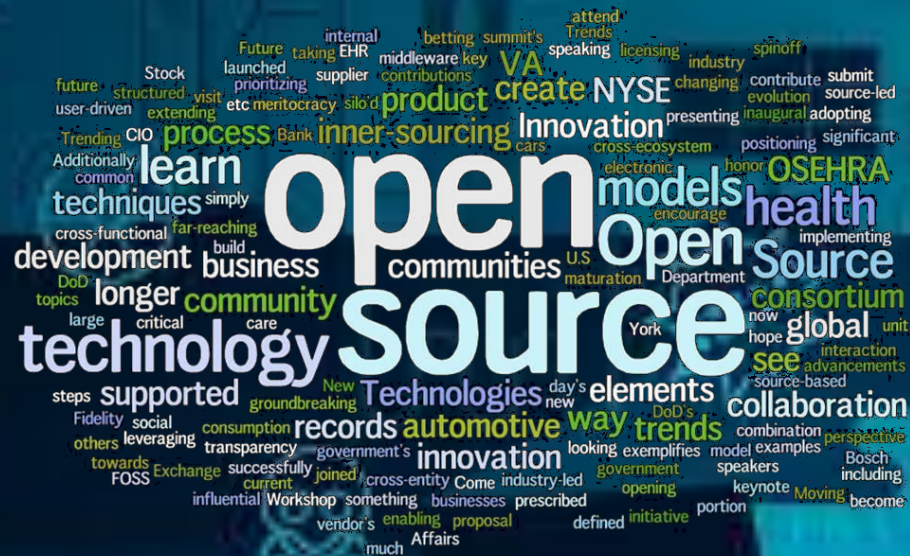
시스코 코리아(대표 조범구)는 최근 한국, 중국, 일본을 포함 아태지역 11개국 2,000여명의 보안 전문가를 대상으로 실시한 '**시스코 2018 아태지역 보안 역량 벤치마크 보고서**(2018 Asia Pacific Security Capabilities Benchmark Study)' 발표

시스코 보고서에 따르면 **국내 기업 61%가 매일 5,000건 이상의 보안 경보를 감지**하고 있으나, 보안 **경보 중 70%가 제대로 된 조치 없이 방치**되고 있는 것으로 분석됐다. 그나마 조사가 이뤄지는 30%의 보안 경보 중에도 **실제 위협으로 판단된 경우는 16%**로 파악됐다. 이는 글로벌 평균(34%) 및 아태지역 표준(44%)보다 한참 낮은 수치로, 기업들이 보안의 정확성과 효율성을 높일 필요가 있음을 시사하고 있다.

또한, 위협으로 판단된 경보 중 **문제를 해결하는 비율은 40%**로 절반에 미치지 못했다. 이는 글로벌 평균 50%, 아태지역 53%보다 낮으며, 이번 아태 11개국 조사에서 한국보다 뒤지는 국가는 태국(37%)과 베트남(39%)이 유일했다.



기업의 사이버 보안과 오픈소스



혹시 구직 중이세요?...“오픈소스’ 하세요”

2018.09.06 / 디지털데일리 백지영 기자

지난 6월 리눅스재단과 다이스그룹이 공동으로 발표한 '2018 오픈소스 직업 보고서'에 따르면 기업 채용 담당자의 87%는 오픈소스 역량을 갖춘 직원을 찾는데 어려움을 겪고 있다고 답했으며, **오픈소스 역량은 채용 조건에서 우선순위라고 꼽는 응답자는 83%**에 달했다.

리눅스 재단 관계자는

“오픈소스 기술 역량은 현재 기업에서 가장 높은 수요를 보이고 있다”

“리눅스를 비롯해 여러 **오픈소스 SW는 전체 SW 개발의 많은 부분을 차지**하고 있기 때문”

“기업들은 오픈소스 개발자에 배고픈 수준이 아니라 거의 굶주려 있다”

<https://www.github.com/>

Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 30 million developers.

3천만 개발자와 180만개 이상의 기업 및 조직

More than 1.8 million businesses and organizations use GitHub



Github로 알아보는 오픈소스 현황(1/3)

Ten years of merging

You've accomplished millions over the last decade. In 2017, the GitHub community reached **24 million developers** working across **67 million repositories**.

24 million
TOTAL USERS

67 million
TOTAL REPOSITORIES

52%
of Fortune 50
COMPANIES USE GITHUB ENTERPRISE

The 50 largest companies in the United States (by revenue) use GitHub Enterprise to build software.

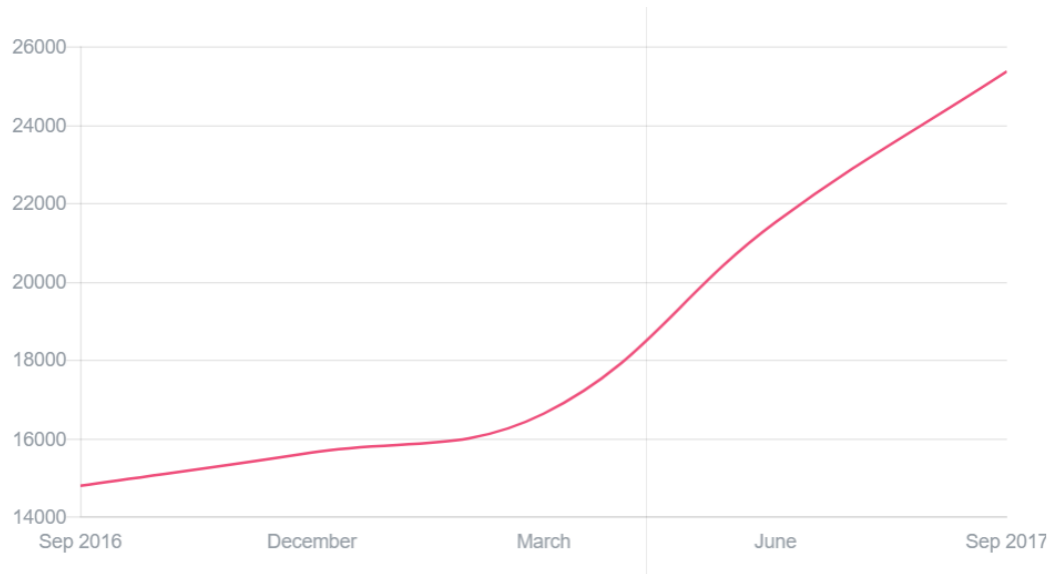
Github로 알아보는 오픈소스 현황(2/3)

A bigger, better GitHub Developer Program

The GitHub Developer Program is a way for you to get the resources you need to build great things on GitHub. This year, we made the program even bigger and welcomed **50 percent more members** than last year.

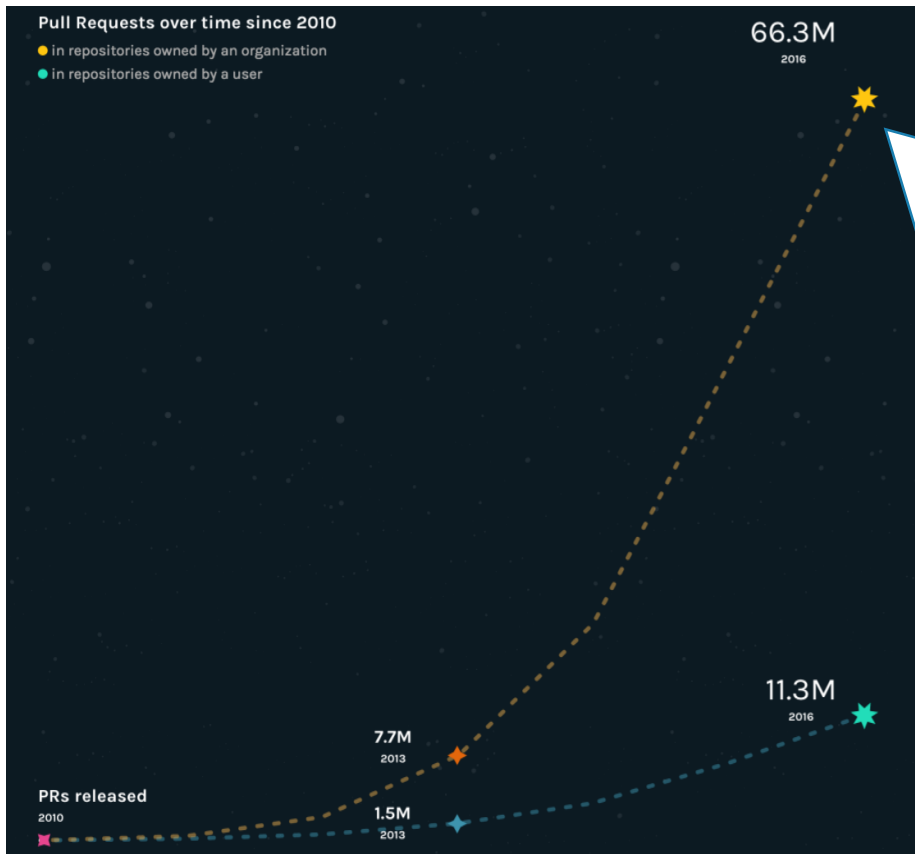
9,794

ACTIVE PROGRAM MEMBERSHIPS SINCE
SEPTEMBER 2016



Github로 알아보는 오픈소스 현황(3/3)

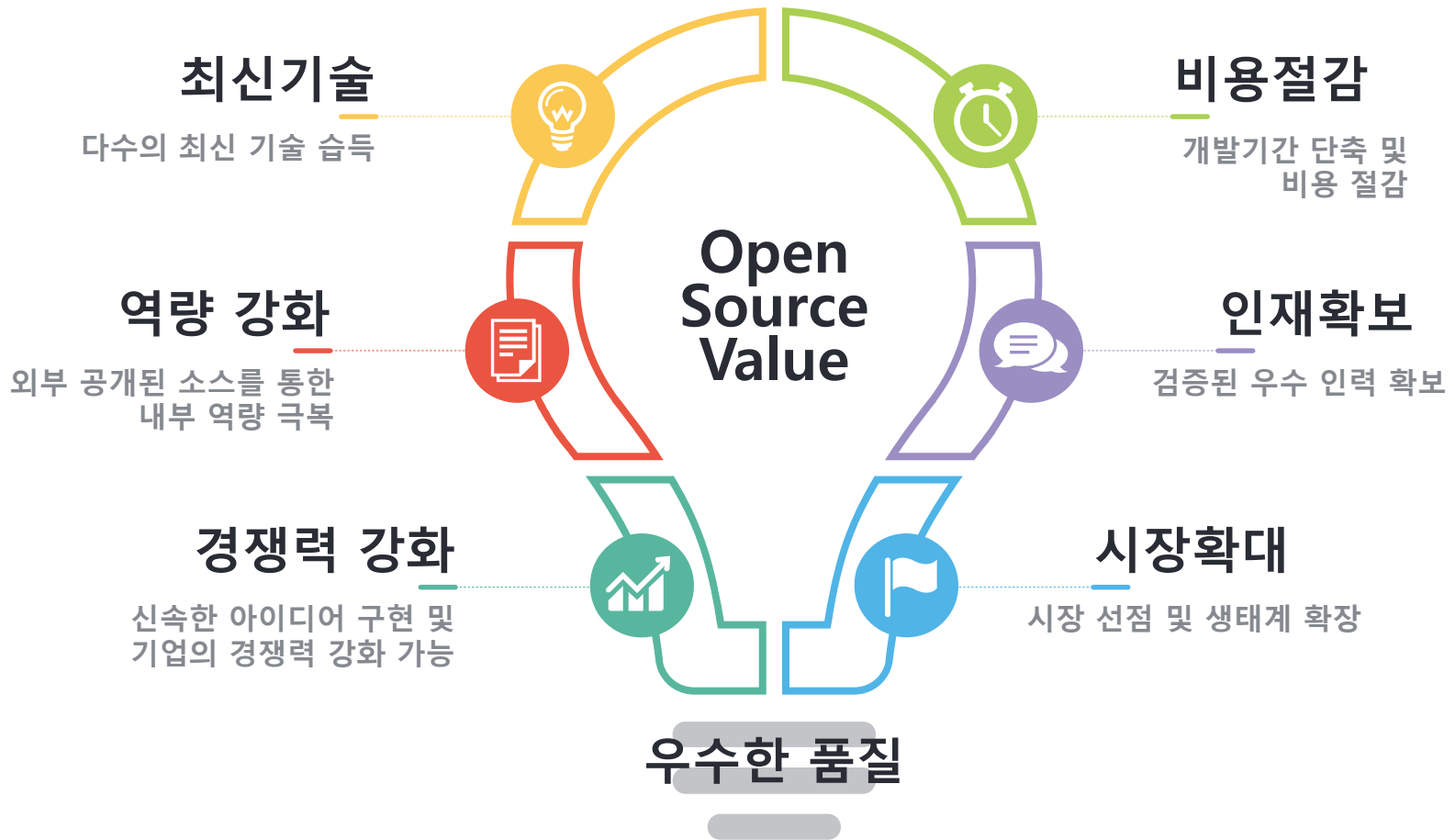
Mature Open Source Ecosystem



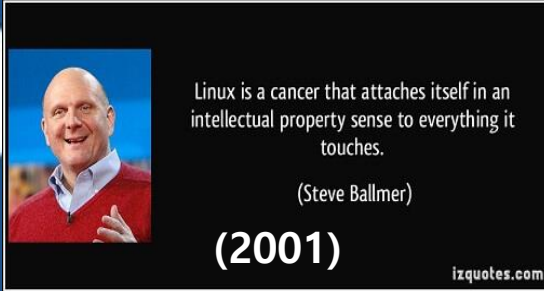
Projects with the most contributors

	MICROSOFT/VSCODE	15K
	FACEBOOK/REACT-NATIVE	8.8K
	NPM/NPM	7.6K
	ANGULAR/ANGULAR-CLI	7.4K
	TENSORFLOW/TENSORFLOW	7.3K
	FORTAWESOME/FONT-AWESOME	6.8K
	ANGULAR/ANGULAR	6K
	DOCKER/DOCKER	6K
	JLORD/PATCHWORK	5.9K
	ANSIBLE/ANSIBLE	5.9K

기업의 오픈소스 채택 이유



오픈소스의 가치



(2018) Github 75억 달러 인수(약 8조 원)

Open Source License



Open Source Security



GitHub Security Bug Bounty

Software security researchers are increasingly engaging with Internet companies to hunt down vulnerabilities. Programs by [Google](#), [Facebook](#), [Mozilla](#), and others have helped to create a strong bug-hunting community. Our bounty program gives a tip of the hat to these researchers and provides some cold hard cash for their efforts.

If you've found a vulnerability, [submit it here](#). You can find more information in the [rules](#) and [FAQs](#). You can also check the current rankings on the [leaderboard](#).







Happy bug hunting!



Github 보안관리 (Bounty hunters)

Leaderboard

These are the current top 10 bounty hunters based on total points earned across all targets. For listings by target, visit their individual [pages](#). For the full list of contributors, check out [GitHub's bounty hunters](#).

1		Aleksandr Dobkining src404 onerroralert(document.domain) @adob	30,750 pts	   
2		joernchen of Phenoelit @joernchen	18,500 pts	   
3		Tanner @Cache-Money	16,000 pts	 
4		Ioannis Kakavas @jkakavas	15,600 pts	
5		kyprizel @kyprizel	14,000 pts	  
6		Orange Tsai @orangetw	12,500 pts	 
7		Markus Fenske @ibblue	10,000 pts	
8		Choongwoo Han @tunz	9,500 pts	  
9		Blake Burkhart @bburky	7,500 pts	
10		Abhishek Dharani @Dharani-abss	6,000 pts	 

Github 보안관리 (Security alerts)

Viewing and updating vulnerable dependencies

Managing alerts for vulnerable dependencies in your organization's repositories

Organization owners and repository admins receive security alerts when GitHub detects a vulnerable dependency in an organization repository. You can specify additional organization members or teams to also receive security alerts for vulnerable dependencies.

- 1 On GitHub, navigate to the main page of the repository.
- 2 Under your repository name, click **Settings**.
- 3 In the left sidebar, click **Alerts**.
- 4 Type the name of the person or team you'd like to receive alerts when GitHub detects a vulnerable dependency, then click their username or team name to select it.
- 5 After you've selected all of the people or teams you'd like to receive alerts, click **Save changes**.

The screenshot shows the GitHub interface for the repository 'octo-org / octo-project'. The 'Insights' tab is selected, and the 'Dependency graph' section is active. A yellow alert banner at the top states: 'We found potential security vulnerabilities in your dependencies. Some of the dependencies defined in these manifest files have known security vulnerabilities and should be updated: ./Gemfile.lock 34 vulnerabilities found'. Below this, a list of dependencies is shown, including 'rails / rails actionpack', 'rails / rails activerecord', 'rails / rails activesupport', 'thoughtbot / terrapin', 'jnunemaker / httparty', 'rails / jquery-rails', and 'mikel / mail'. A detailed popup for 'rails / rails actionpack' shows '6 known vulnerabilities found' with a list of CVEs (CVE-2016-2098, CVE-2016-0751, CVE-2015-7576, CVE-2014-7829, CVE-2014-7818, CVE-2014-0130) and their severities (High, Moderate). It also suggests updating 'actionpack' to version '3.2.22.2'.

<https://help.github.com/articles/about-security-alerts-for-vulnerable-dependencies/>

Apache Software Foundation Security Team



The Apache Way

Contribute

ASF Sponsors

THE APACHE SECURITY TEAM

The Apache Security Team exists to provide help and advice to Apache projects on security issues and to provide co-ordination of the handling of security vulnerabilities. All members of the Security Team are also [members](#) of the Apache Software Foundation.

REPORTING A VULNERABILITY

We strongly encourage folks to report security vulnerabilities to one of our private security mailing lists. A [list of security contacts for Apache projects](#) is available. If you can't find a project's security contact, please report then please use the general security address below.

Please note that the security mailing lists should only be used for reporting and fixing such vulnerabilities. We cannot accept regular bug reports or other security issues. A security report that does not relate to an undisclosed security problem in an Apache product will be ignored.

Also note that the security team handles vulnerabilities in Apache products, not other projects. Security issues for other projects should be sent to root@apache.org only.

The general security mailing list address is: security@apache.org. This is a private mailing list.

Please note that we do not use a team OpenPGP key. If you wish to encrypt your e-mail, please use the public keys of the members of the Apache Security Team and be aware that it may take us a little longer to respond to you. Please do not contact these members individually about security issues.

- Mark Cox - 5B25 45DA B219 95F4 088C EF8A 36CE E4DE B00C FE33 - pgp.mit.edu
- Bill Rowe - B1B9 6F45 DFBD CCF9 7401 9235 193F 180A B55D 9977 - pgp.mit.edu
- Mark Thomas - A9C5 DF4D 22E9 9998 D987 5A51 10C0 1C5A 2F60 59E7 - pgp.mit.edu
- Yann Ylavic - 8935 9267 45E1 CE7E 3ED7 48F6 EC99 EE26 7EB5 F61A - pgp.mit.edu

The keys for all of the above can also be obtained in a single file from [The Apache Software Foundation](#).

VULNERABILITY INFORMATION

Information on the published vulnerabilities for an Apache project can usually be found on the project's web pages. For convenience a [list of security information pages for Apache projects](#) is available. If you can't find the information you are looking for on the project's web site, you should ask your question on the project's user mailing list. The security lists **should not be used to ask questions about**:

- how to configure the product securely;
- if a published vulnerability applies to the version of the Apache product you are using;
- if a published vulnerability applies to the configuration of the Apache product you are using;
- obtaining further information on a published vulnerability;
- the availability of patches and/or new releases to address a published vulnerability.

The relevant project's users list is the place to ask such questions. Any such questions sent to the Apache Security Team or to a project security team will be ignored.

VULNERABILITY HANDLING

An overview of the vulnerability handling process is:

- The reporter reports the vulnerability privately to Apache.
- The appropriate project's security team works privately with the reporter to resolve the vulnerability.
- A new release of the Apache product concerned is made that includes the fix.
- The vulnerability is publicly announced.

A more detailed description of the process has been written for committers. Reporters of security vulnerabilities may also find it useful.

Apache Software Foundation Security Team



Google Custom Search

The Apache Way

Contribute

ASF Sponsors

ASF PROJECT SECURITY INFORMATION

Pages maintained by ASF projects to provide information on known security vulnerabilities are listed below. The security contact for reporting new vulnerabilities is also shown. Note that whilst all projects have a security team, not all project security teams have a dedicated address for reporting new vulnerabilities.

To report a vulnerability in an Apache project that is not listed below, please contact the [Apache Security Team](#).

Apache project security page	Security Contact		
Apache Portable Runtime (APR)	Apache Security Team	Apache NiFi	Apache NiFi Security Team
Apache Ant	Apache Security Team	Apache OFBiz	Apache OFBiz Security Team
Apache Apex	Apache Apex Security Team	Apache OpenMeetings	Apache OpenMeetings Security Team
Apache CloudStack	Apache CloudStack Security Team	Apache OpenOffice	Apache OpenOffice Security Team
Apache Commons	Apache Security Team	Apache ORC	Apache ORC Security Team
Apache CouchDB	Apache CouchDB Security Team	Apache Sentry	Apache Sentry Security Team
Apache Geronimo	Apache Geronimo Security Team	Apache SpamAssassin	Apache SpamAssassin Security Team
Apache Guacamole	Apache Guacamole Security Team	Apache Shiro	Apache Shiro Security Team
Apache Hadoop	Apache Hadoop Security Team	Apache Sling	Apache Sling Security Team
Apache Hive	Apache Hive Security Team	Apache Spark	Apache Security Team
Apache HTTP Server	Apache HTTP Server Security Team	Apache Struts	Apache Struts Security Team
Apache Ignite	Apache Ignite Security Team	Apache Tomcat	Apache Tomcat Security Team
Apache Jackrabbit	Apache Jackrabbit Security Team	Apache Traffic Control	Apache Traffic Control Security Team
Apache Kafka	Apache Kafka Security Team	Apache Traffic Server	Apache Traffic Server Security Team
Apache Libcloud	Apache Libcloud Security Team	Apache Trafodion	Apache Trafodion Security Team
Apache Metron	Apache Metron Security Team	Apache Zeppelin	Apache Zeppelin Security Team
		Apache ZooKeeper	Apache ZooKeeper Security Team

Linux Foundation CII

THE **LINUX** FOUNDATION PROJECTS



Core Infrastructure Initiative.
Fortifying our future.

운영위원회

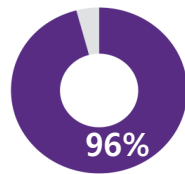


Facebook
Amazon Web Services
HPE Software
Cisco
VMWare
Intel
NetApp
Bloomberg
Microsoft
Qualcomm Technologies Inc.
Google
Lyft
Hitachi
IBM
NEC
Rackspace
Fujitsu

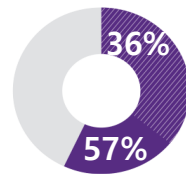
2. 2018 오픈소스 보안과 리스크 분석

2018 오픈소스 보안과 리스크 분석 보고서

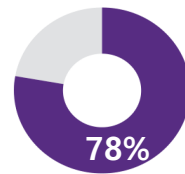
- 상용 애플리케이션의 오픈소스 보안, 컴플라이언스 및 코드 품질 리스크 현황 제공
- 2017년 1,100개 이상 익명의 상용 소프트웨어 코드 대상으로 블랙덕 검증 후 데이터 도출



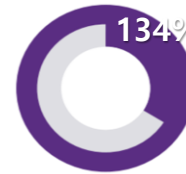
상용 애플리케이션의 96%에서 오픈소스 발견



코드내 오픈소스 비율 36% ⇒ 57% 증가



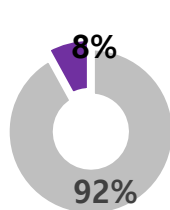
코드의 78%는 최소 한 개 이상의 보안취약점 보유



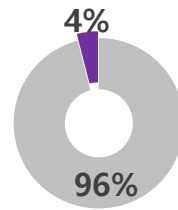
애플리케이션 당 취약점 작년대비 134% 증가



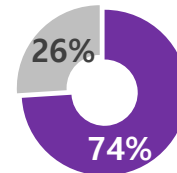
발견된 보안취약점의 54% 이상이 고위험군



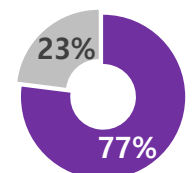
검사된 애플리케이션의 8%에서 Struts 발견, 그 중 33% 여전히 미조치



취약점 공개된 지 4년, 애플리케이션의 4% Heartbleed 포함



애플리케이션의 74%에서 라이선스 위반 발견(GPL 44%)

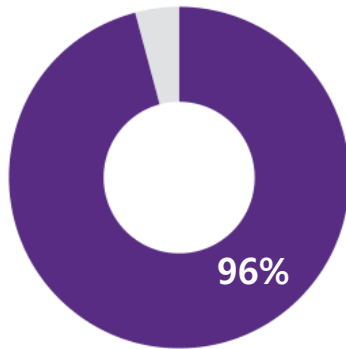


IoT 평균 77% 오픈소스 활용, 평균 677개의 보안취약점 보유

2018 오픈소스 보안과 리스크 분석 보고서

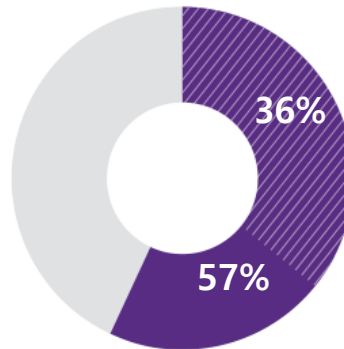
상용 애플리케이션 코드의 절반 이상 오픈소스 사용, 78% 보안 취약점 보유

상용SW의 오픈소스 사용 비율



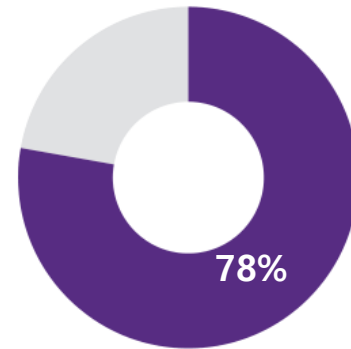
오픈소스 컴포넌트
상용 애플리케이션의 **96%에서 발견**
(애플리케이션 당 평균 257 컴포넌트 포함)

상용SW 오픈소스 코드 사용 비율



상용 코드베이스 내
오픈소스 **코드 비율 57%**
작년 36% ⇒ 58% 증가

오픈소스 보안취약점 비율



코드의 **78%**는 최소 한 개
이상의 보안취약점을 가지며,
평균 64개의 취약점 존재

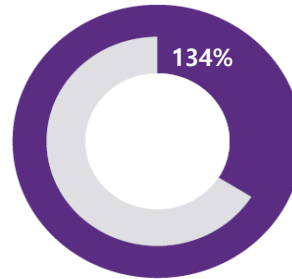
2018 오픈소스 보안과 리스크 분석 보고서

애플리케이션 평균 64개 취약점 발견, 작년대비 134% 증가

'17년 신규 오픈소스 보안취약점 수



평균 취약점 발견 수



보안취약점 발견 시점



2017년 4,800개의
새로운 오픈소스
보안취약점 발견
'16년 3,623개 ⇒ 32% 증가

애플리케이션 당
평균 64개의 취약점 발견
'16년 평균 27개 ⇒
134% 증가

검증을 통해 식별된
보안취약점은 평균적으로
약 6년 전에 공개
'17년 4년

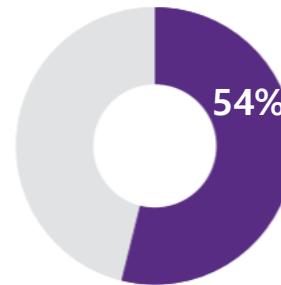
2018 오픈소스 보안과 리스크 분석 보고서

범용적 컴포넌트에서 주로 발견, 54% 이상이 고위험군

Top 10 high-risk components found

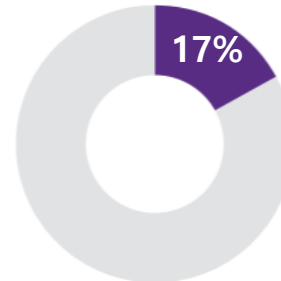
Apache Commons Collections (3.2.1)	9.36% of codebases
Spring Framework (version unspecified)	6.54%
Apache XML Xalan-Java (2.7.1)	5.30%
Node.js (version unspecified)	5.12%
FreeBSD (version unspecified)	4.95%
zlib (1.2.8)	4.77%
Sun Java Platform Standard Edition (JRE) (J2RE) (version unspecified)	4.24%
zlib (version unspecified)	4.06%
Sun Java Platform Standard Edition SDK (J2SDK) (JDK) (version unspecified)	3.89%
Open BSD (version unspecified)	3.18%

Attackers continue to exploit unpatched software to conduct attacks against critical infrastructure organizations. As many as 85% of targeted attacks are preventable, according to US-CERT.



발견된 보안취약점의
54% 이상이 High-risk

이러한 High-risk의
보안취약점은
Apache Commons
Collections와 Spring
Framework를 포함하는
범용적인 버전에서 발견



17%가 Heartbleed,
Logjam, Freak, Drown,
Poodle과 같이 널리
알려진 보안취약점을
포함

2018 오픈소스 보안과 리스크 분석 보고서

오픈소스 사용에 대한 산업간 연구결과
범용적인 컴포넌트가 전 산업 애플리케이션에 모두 영향을 줌

산업	애플리케이션의 흔한 고위험군 컴포넌트	고위험군 컴포넌트를 지닌 코드베이스 비율
Aerospace, Aviation, Automotive, Transportation Logistics	zlib	17%
Big Data, AI, BI, Machine Learning	Spring Framework	9%
Computer Hardware & Semiconductors	libxml2, zlib	17%
Cyber Security	Apache Log4j	12%
Ed Tech	Zend Framework	12%
Energy & Clean Tech	Apache Xerces-C++ XML Parser	33%
Enterprise Software/SaaS	Spring Framework	5%
Financial Services & FinTech	Spring Framework	10%
Healthcare, Health Tech, Life Science	libtiff, libxml2	4%
Internet & Mobile Apps	Spring Framework	21%
Internet of Things	OpenSSL, Apache Tomcat	10%
Internet & Software Infrastructure	Node.js	28%
Manufacturing, Industrials, Robotics	Sun Java Platform Standard Edition (JRE) (J2RE), Apache Tomcat	9%
Marketing Tech	Symfony	15%
Retail & E-commerce	Apache Commons Collections	13%
Telecommunications & Wireless	Chromium Source	25%
Virtual Reality, Gaming, Entertainment, Media	zlib	25%

산업별 보안 위험 및 라이선스 이슈 현황

산 업	코드베이스 내 오픈소스 비율	높은 보안 위험을 지닌 코드베이스 비율	라이선스 문제를 지닌 코드베이스 비율
Aerospace, Aviation, Automotive, Transportation Logistics	53%	30%	78%
Big Data, AI, BI, Machine Learning	45%	25%	72%
Computer Hardware & Semiconductors	74%	22%	72%
Cyber Security	36%	41%	76%
Ed Tech	45%	15%	77%
Energy & Clean Tech	11%	33%	78%
Enterprise Software/SaaS	46%	17%	83%
Financial Services & FinTech	27%	34%	78%
Healthcare, Health Tech, Life Science	48%	31%	71%
Internet & Mobile Apps	57%	60%	64%
Internet of Things	77%	15%	75%
Internet & Software Infrastructure	65%	67%	78%
Manufacturing, Industrials, Robotics	32%	9%	91%
Marketing Tech	76%	23%	77%
Retail & E-commerce	71%	32%	61%
Telecommunications & Wireless	64%	38%	100%
Virtual Reality, Gaming, Entertainment, Media	70%	50%	92%

11%~77%

9%~67%

61%~100%

오픈소스 보안 취약 현황 및 문제점

2018 오픈소스 보안과 리스크 분석 보고서



모든
사이버테러의
80% 이상
애플리케이션
레벨에서 발생

상용
애플리케이션
96%
오픈소스 활용

코드의 절반은
취약점 보유,
발견된 취약점의
절반은
고위험군

보안취약점
신규 약 5천 건,
코드당 64개
취약점 보유



3. 오픈소스 보안취약점

피해사례

EQUIFAX – 2017년 최악의 보안사고(Apache Struts2)



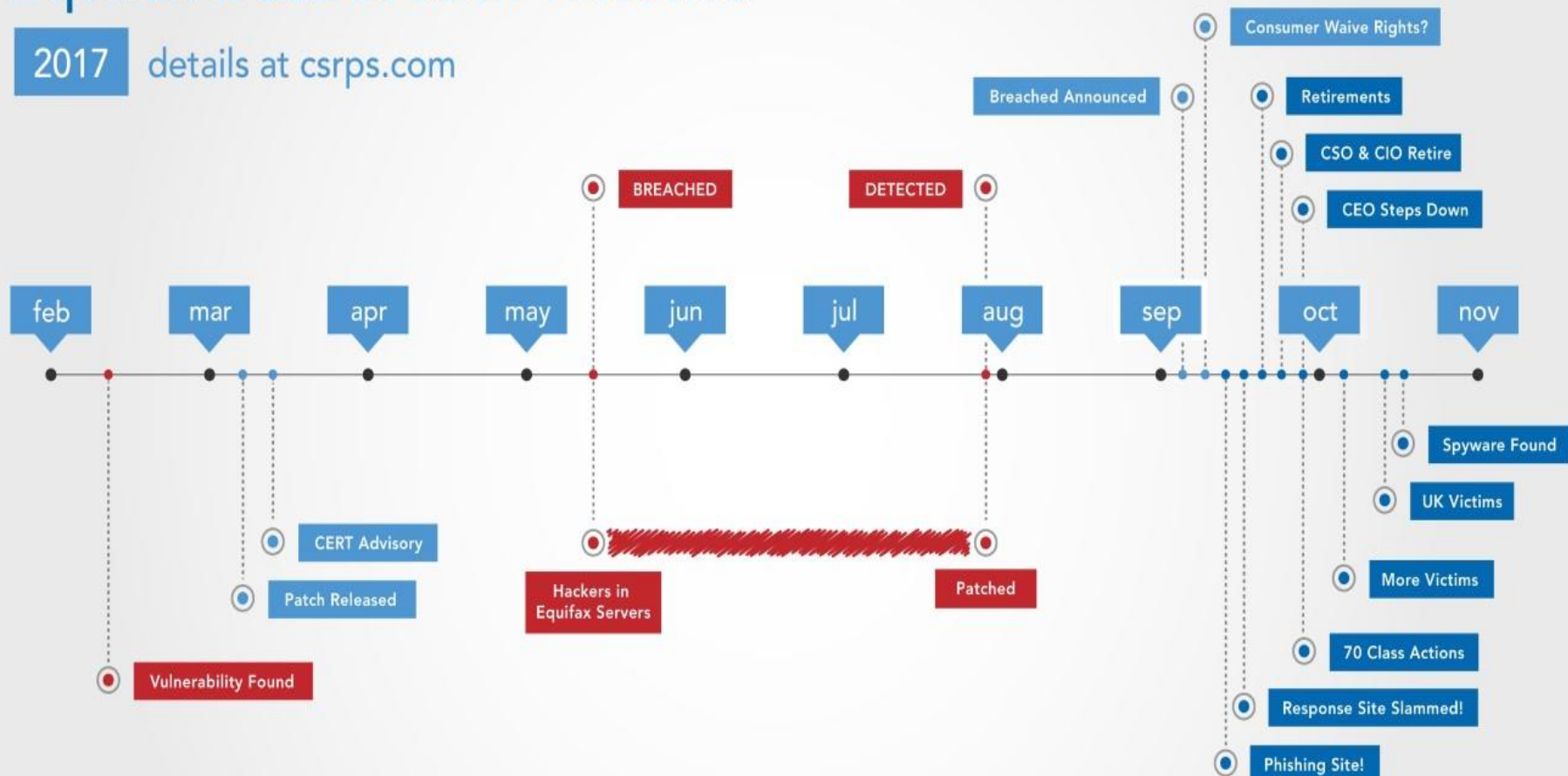
EQUIFAX

에퀴팩스 피해자 1억 4700만명
생년월일, 신용카드번호, 운전면허번호 등 주요정보유출

EQUIFAX Data Breach Timeline

Equifax Data Breach Timeline

2017 details at csrps.com



<https://csrps.com/meticulous-timeline-equifax-data-breach>

가상화폐 채굴 신종 해킹 크립토재킹(Cryptojacking) 급증

- 암호화폐 채굴 '크립토재킹' 골드러시...작년 8500% 증가, 2018.4.3
- “해커 먹잇감된 ‘오픈소스 SW’...가상화폐 채굴에 악용”, 2018.7.5
- 기업도 위험하다...테슬라 해킹해 가상통화 채굴한 해커들, 2018.3.2
- 테슬라의 AWS 클라우드 해킹... 채굴용 해킹 대상 개인→기업 확대
지난 1월초부터 한 달 넘게 채굴에 악용
- SK인포섹 "상반기 취약점 43%가 오픈소스 관련...보안설정 소홀" 2018.7.5
악성코드 80% 이상은 가상화폐 공격용...빗썸 관리자·서버 공격 가능성
- 리눅스, 자바 등 오픈소스 보안취약점 타겟

스마트카, 제조업 오픈소스 보안



마이크 피튼처(Mike Pittenger)
전 블랙덱소프트웨어 부사장
'2017 오픈소스 보안 4대 전망' 기자간담회

4월

“자동차 1억 개 이상의 코드라인 포함,
자동차 제조업체의 오픈소스 관리 부실은
대규모 자동차 리콜 사태 발생할 수도”



IT >
글로벌
리포트

혼다 생산공정, 워너크라이 공격당해...자동차 1000대 생산 지연

김범수 기자 > 김중형 인턴기자 >

기사

100자평(0)

다운로드 이메일 공유 +크게 | -작게

입력 : 2017.06.22 17:14

일본 자동차 제조 기업 혼다가 랜섬웨어 워너크라이(WannaCry) 사이버 공격의 새 희생자가 됐다.

미국 파이낸셜타임스(FT)는 21일(현지시간) 혼다의 일부 컴퓨터에 랜섬웨어가 발견돼 일요일부터 화요일까지 일부 공정이 멈췄다고 보도했다. 도쿄 6시간에 있는 사야마(Sayama·狭山) 공정의 생산이 일시적으로 멈췄으며, 약 1000대의 차량생산이 지연됐다.

워너크라이 랜섬웨어 공격은 마이크로소프트 운영체제 중 구형 버전의 약점을 이용한다. 해커는 악성 코드를 PC에 침투시켜 중요 파일을 암호화해 접근하지 못하게 하고 몸값을 요구한다. 지난 10일에는 국내 웹 호스팅 업체 인터넷나야나에서 랜섬웨어 공격을 당했다. 인터넷나야나 측에서 자체해결하지 못하고 해커에 13억원을 주고 암호 해제 프로그램을 받아 복구 작업을 진행해 논란이 되기도 했다.

http://biz.chosun.com/site/data/html_dir/2017/06/22/2017062202375.html

교육용 오픈소스를 활용한 랜섬웨어

Open Source Hidden Tear, EDA2, Heimdall을 기반으로 랜섬웨어 제작

Released on October 26, the Heimdall ransomware is self-contained in one 482-line PHP file, which produces the GUI below. If used by attackers, they would deploy the ransomware by uploading this PHP file to compromised servers and accessing the file's URL.

The image displays the user interface of the Heimdall ransomware. On the left, there is a sidebar with sections: 'Heimdall Information' (showing contact details like 'email@email.com', '2 bitcoin', and 'Cont: 123adsd'), 'Heimdall Encrypted' (with a password input field and 'Send' button), 'Heimdall Decrypted' (with a password input field and 'Send' button), 'Heimdall Message', and 'Heimdall Resume' (showing statistics like 'Total of 2010 files', 'Files cryptography: 0', and 'Files decryption: 0'). A table at the bottom has columns for 'Number', 'Path/File', and 'Status'. On the right, a code editor shows the PHP source code for 'Heimdall Ransomware', including a file tree on the left and code for license, class creation, and a function. A video player at the bottom shows a play button and a progress bar at 5:11 / 5:14.

오픈소스를 활용한 Korean 랜섬웨어

2018년 이후 가상화폐 송금을 목적으로 한 랜섬웨어 기승

- 2018년 카카오톡 위장 랜섬웨어 변종 발견(Hidden-Tear 오픈 소스 기반으로 제작)

바탕화면 파일 중 '.txt', '.doc', '.docx', '.xls', '.xlsx', '.ppt' 등 특정 확장자의 파일을 암호화한 뒤 1비트코인 요구

2016년 랜섬웨어



오픈소스 보안취약점을 응용한 웹 해킹기법

오픈소스 보안취약점 활용 주요 웹 해킹 기술

Download Attack

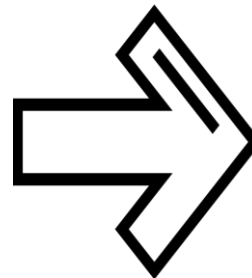
Webshell upload

Parameter

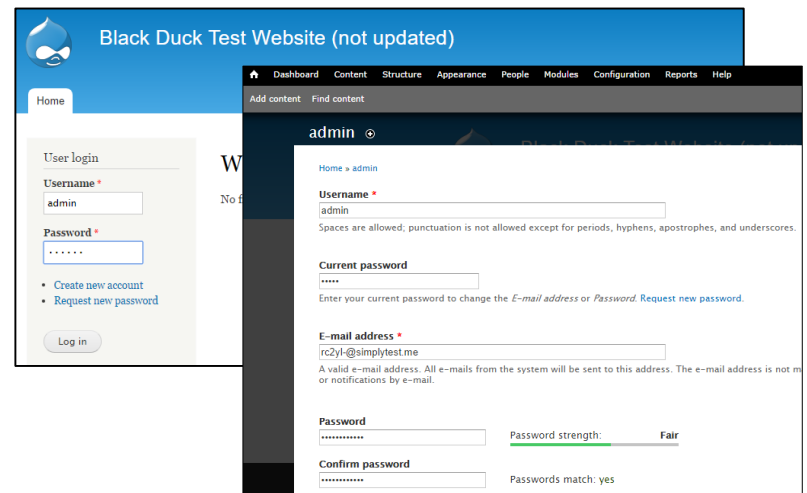
Tampering

SQL Injection

Cross-site script



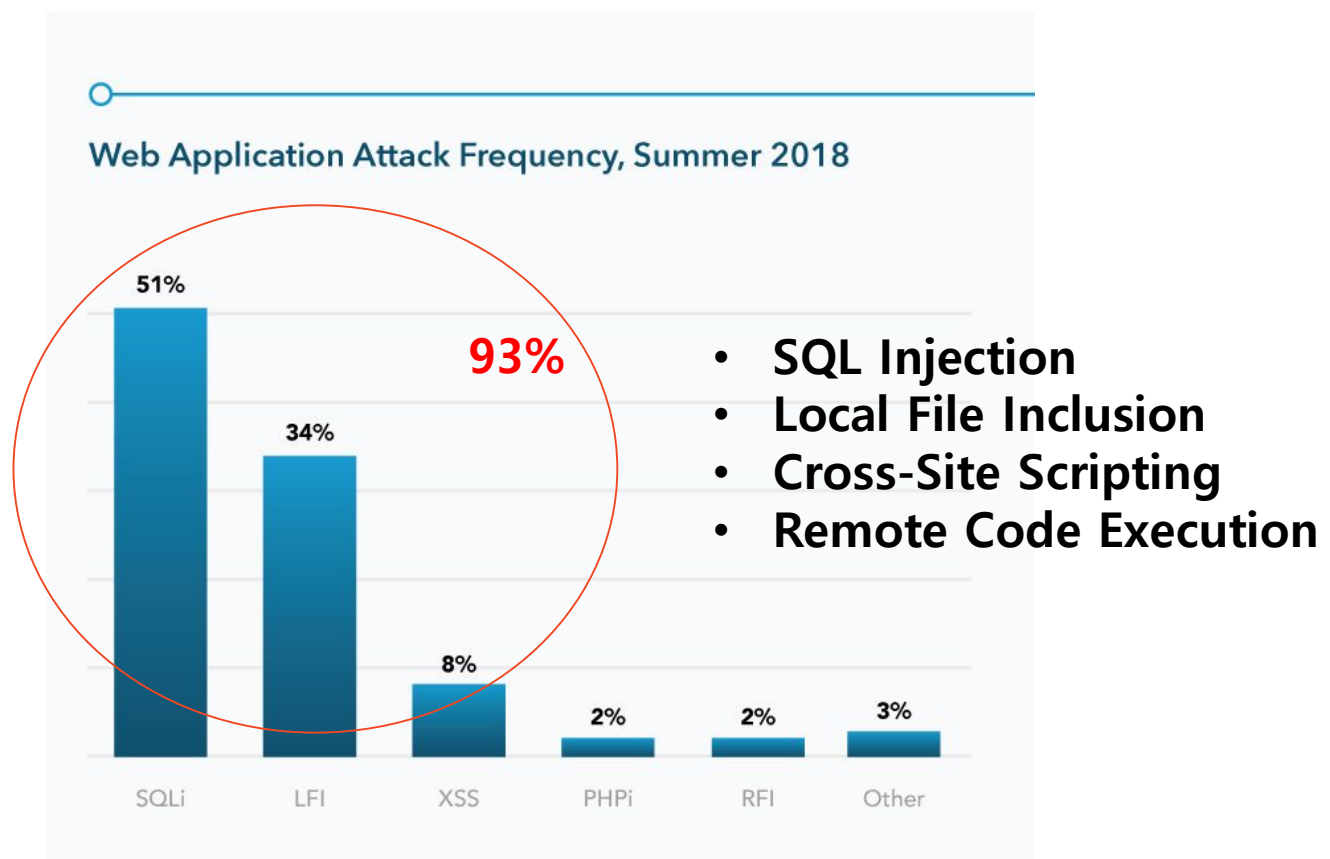
```
Drupal 7.x SQL Injection SA-CORE-2014-005
A GUEST OCT 15TH 2014 13,727 NEVER
USE PASTEBIN A LOT? UPGRADING TO A PRO ACCOUNT UNLOCKS MANY COOL FEATURES.
KEEP CALM AND GO PRO
Not a member of Pastebin yet? Sign Up, it unlocks many cool features!
Python 1.01 kb
1. #Drupal 7.x SQL Injection SA-CORE-2014-005 https://www.drupal.org/SA-CORE-2014-005
2. #Creditz to https://www.reddit.com/user/fyukyuk
3. import urllib2,sys
4. from drupalpass import DrupalHash # https://github.com/cvangysel/gitexd-drupalorg/blob/master/drupalorg/drupalpass.py
5. host = sys.argv[1]
6. user = sys.argv[2]
7. password = sys.argv[3]
8. if len(sys.argv) != 3:
9.     print "host username password"
10.    print "http://nope.io admin wowsecure"
11.    hash = DrupalHash("SS5CTo9G7Lx28r-zCfnp4uB2HJlknDKv6QTqHaf82HlbhPT2KSTzKzHL", password).get_hash()
```



웹애플리케이션 3대 공격기법

가장 널리 사용되는 웹애플리케이션 대상 공격은 SQL Injection

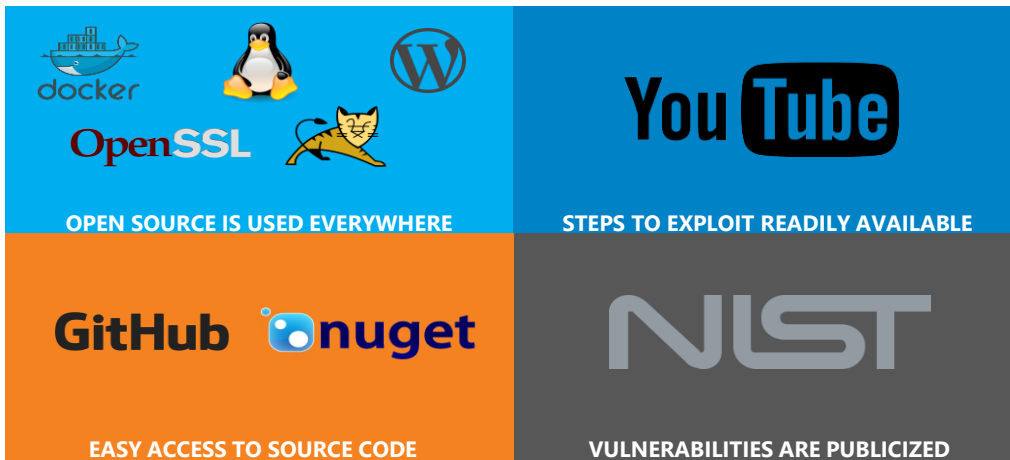
SQLi, LFI, XSS 유형의 공격이 93%



©2018 AKAMAI

가장 빠르고 효과적인 해킹도구 오픈소스

해커들이 얻는 보안취약점 정보 획득 경로



Home > 뉴스 > 전체기사

사이버전 해커들도 오픈소스 대거 활용하기 시작

좋아요 82개

입력: 2017-04-13 11:23



가

가



오픈소스, 1단계 공격 효율성 높이고 위험에도 효과적
하지만 방어 난이도 낮아진다는 단점도 있어...양날의 검

[보안뉴스 문가용 기자] 국가의 후원을 받고 활동하는 해킹 팀들이 오픈소스를 점점 더 많이 사용하고 있다는 소식이다. 이들의 오픈소스 활용이 왜 재미있다면, 여태까지 국가를 등에 업은 공격자들이라고 하면 돈이 충분해 굳이 무료 툴을 사용할 필요가 없었고, 오픈소스가 보안의 측면에서 그다지 안정적이지 않았기 때문이다.



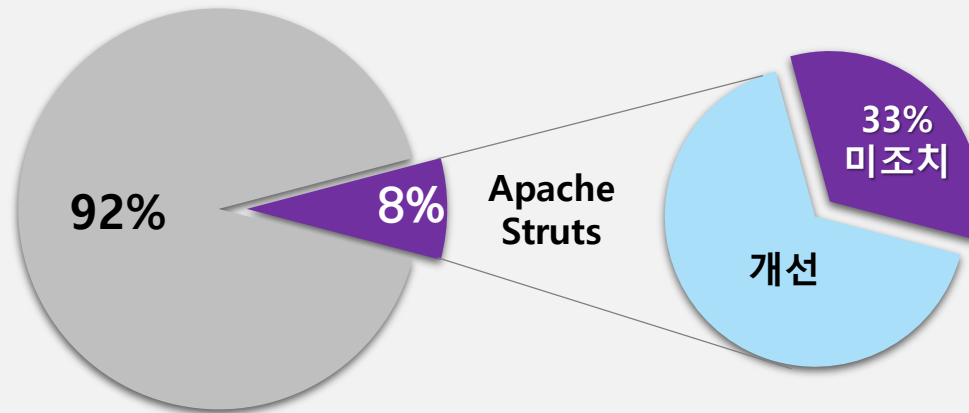
<http://www.boannews.com/media/view.asp?idx=54245>

4. 오픈소스 보안취약점 관리방안

오픈소스 보안취약점 개선 안 하는 것인가? 못 하는 것인가?

보안 사고와 보안취약점이 존재함에도 불구하고도 위험 수용

글로벌 1,100개 상용 소프트웨어 대상 Apache Struts 현황조사



* 2018 Open Source Security & Risk Analysis, Blackduck Synopsys

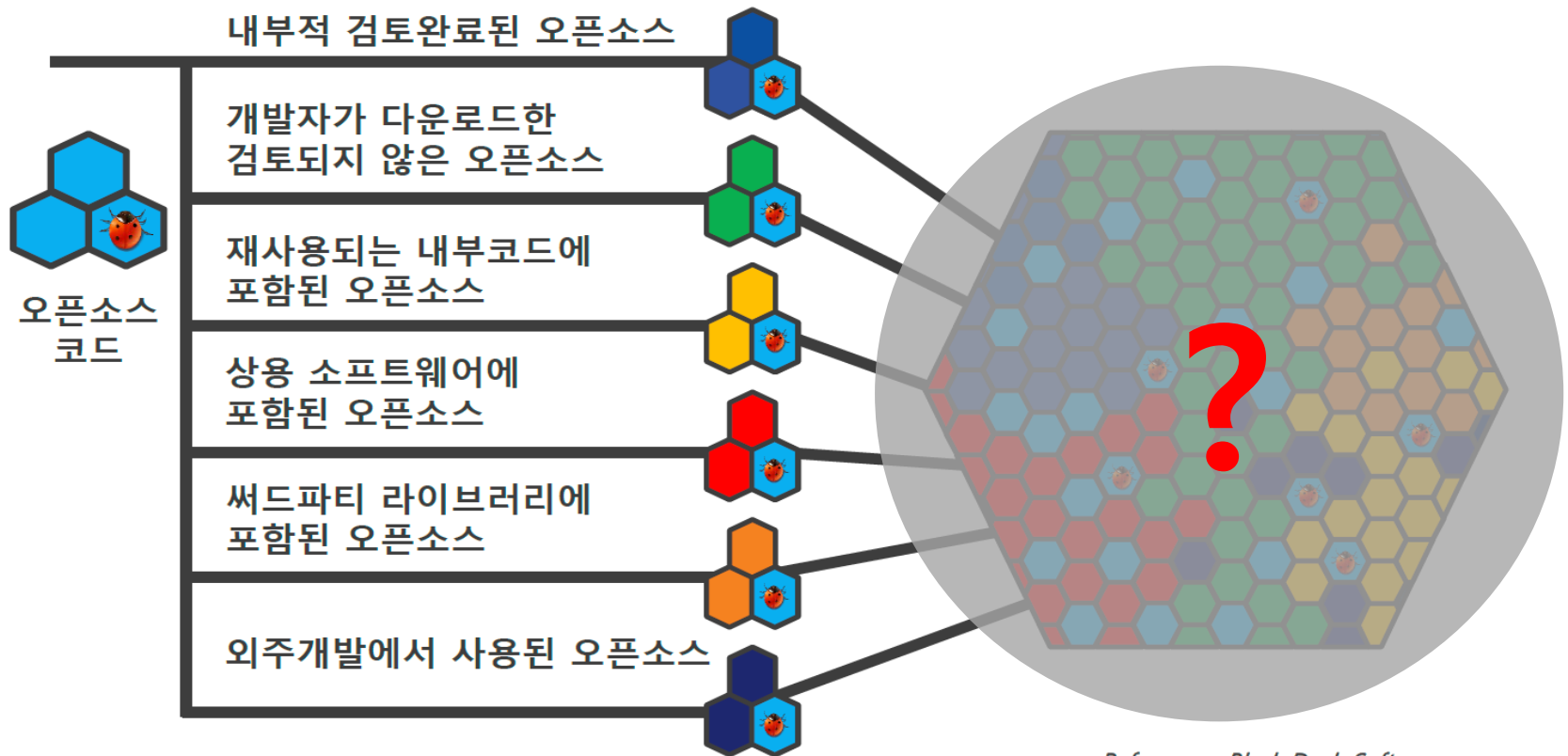
오픈소스 보안취약점 개선 안 하는 것인가? 못 하는 것인가?



VISIBILITY

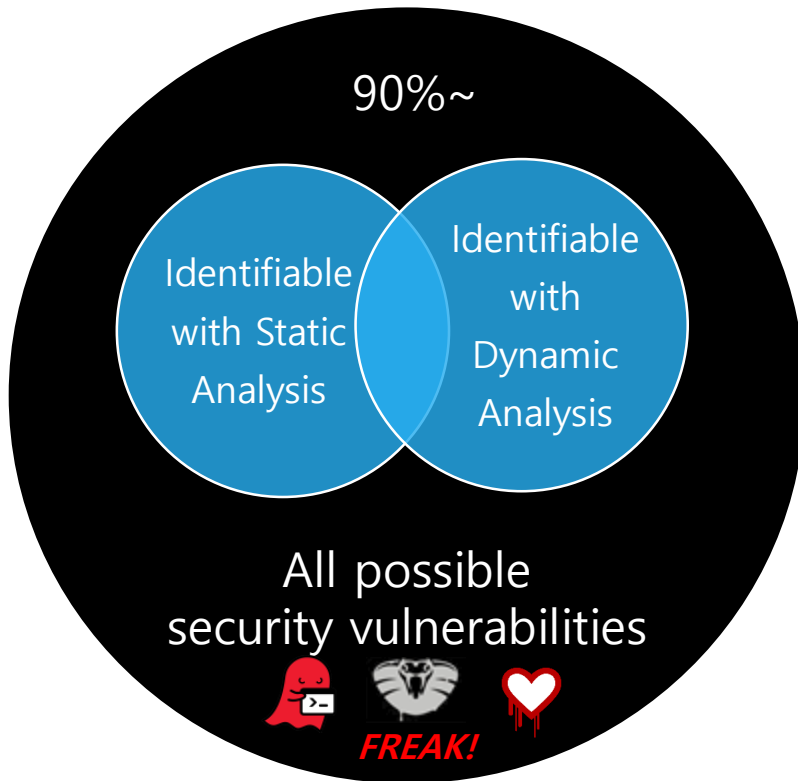
다양한 경로의 오픈소스 유입과 활용

보이지 않으면 대응도 불가능합니다.



Reference: Black Duck Software

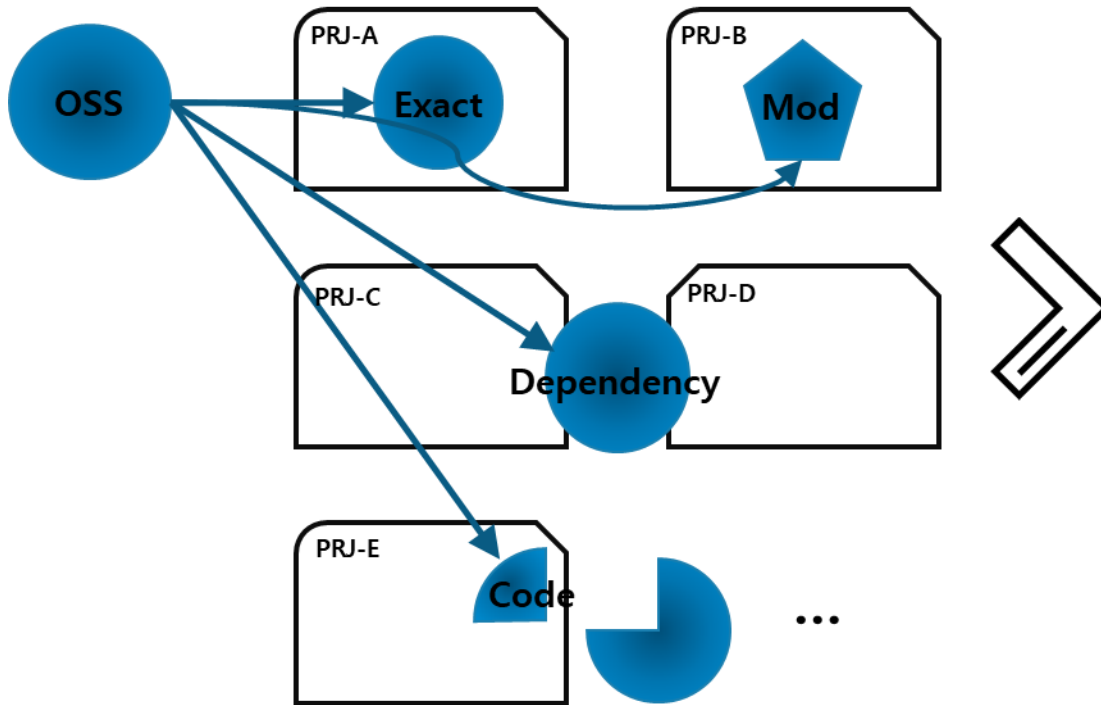
자동화 테스트 도구의 한계





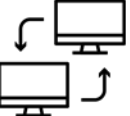

일반적인 보안취약점만 발견하는
자동화된 테스트 도구의 한계

오픈소스 활용유형에 따른 조치방법 다양

오픈소스 사용 패턴



조치방법

-  Upgrade patch
-  Code Modify
-  Redevelopment
-  Delete



패턴확인
조치방법
어려움

오픈소스 보안 라이프사이클에서 사용자 역할

오픈소스 취약점 조치는 결국 사용자 몫

Discovering Vulnerabilities



오픈소스
보안 취약점 발견

Releasing Fixes



오픈소스 보안취약점
수정 및 릴리즈

Notifying Users



사용자의 오픈소스
보안취약점 인지

Adopting Published Fixes



사용자의 오픈소스
보안취약점 조치 관리

- 오픈소스 현황 파악
- 스스로 정보 검색
- 오픈소스 보안취약점 관리
- **책임은 사용자에게!!**

오픈소스 가시화 및 관리



INVENTORY
오픈소스 사용
목록 구축



MAP
알려진
보안취약점 맵핑



IDENTIFY
라이선스와 보안
리스크 식별



TRACK
오픈소스 리스크
정책 집행



ALERT
새로운 보안취약점
모니터링

5가지 주요 업무 자동화 및 오픈소스 검증내역서(BoM) 관리



안전한 IT 시스템 운영 및 자산보호

현재 오픈소스 취약점을 파악하고 진단하여
조직에 맞는 최적의 보호체계 수립

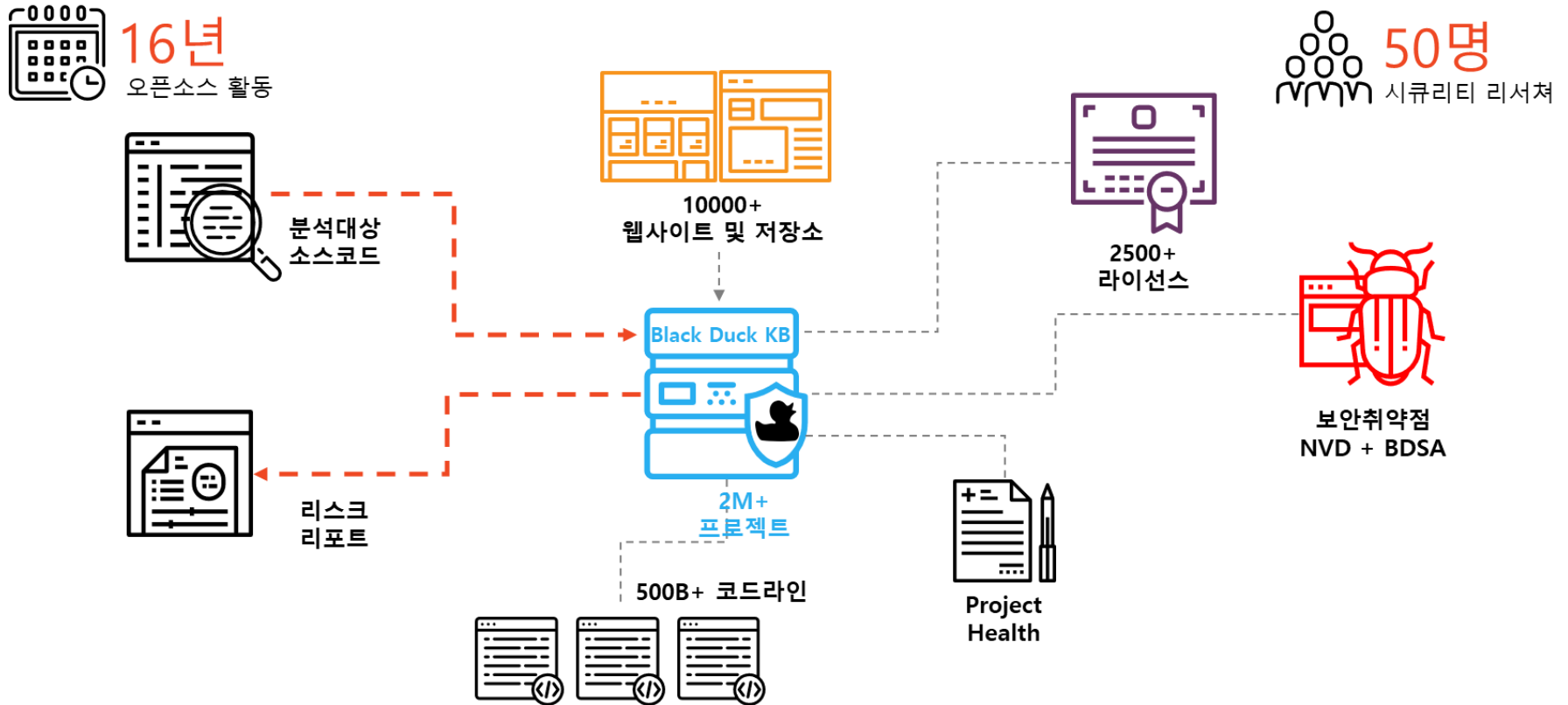


BDSK 오픈소스 보안 컨설팅 수행절차



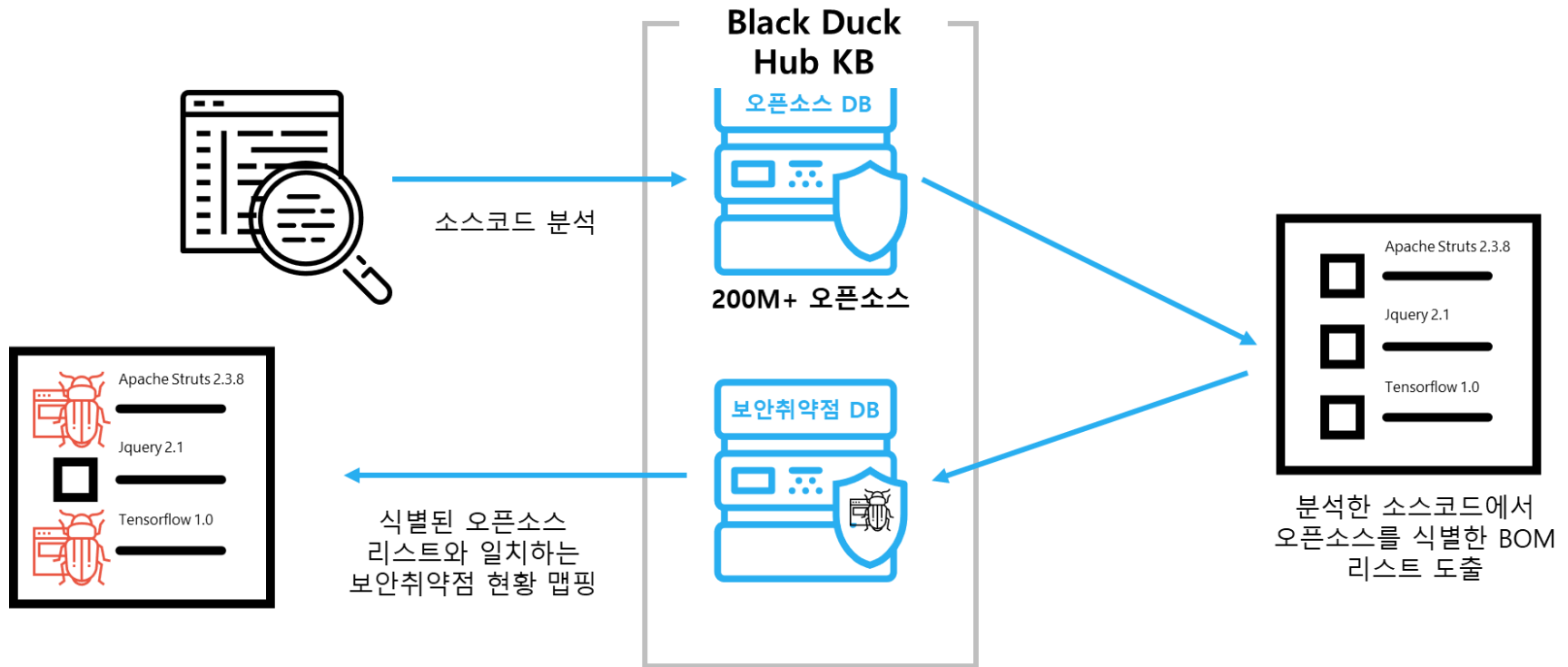
Blackduck Hub 오픈소스 관리도구

- 200만개 이상의 오픈소스 프로젝트를 포함한 전세계 최고의 오픈소스 데이터베이스



오픈소스 데이터베이스의 중요성

- 상용코드에서 오픈소스 보안취약점을 찾아내기 위해서는 취약한 보안의 원인을 제공하는 오픈소스의 식별을 정확하게 할 수 있는 오픈소스 DB의 양과 퀄리티가 중요함



BDSK 오픈소스 보안취약점 종합관리서비스

(前 블랙덕소프트웨어코리아)

BDSK는 2006년에 설립된 오픈소스 관리 솔루션 및 컨설팅 공급 기업입니다.

국내 오픈소스 활용과 거버넌스 활성화에 대한 인식 제고 및 확산에 앞장서 왔으며, 국내외 다양한 오픈소스 관련 단체 및 선도기업과 협업하여 오픈소스 컴플라이언스 및 거버넌스에 기여하고 있습니다.

BDSK





오픈소스는 소스가 공개되어 보안에 취약하다?
사용자가 오픈소스 보안 수준을 결정한다.

감사합니다.



BDSK

(주)비디에스케이
(前 블랙덕소프트웨어코리아)

김혜영 차장 hykim@bdsk.co.kr