리눅스 Ghost 취약점 대응방안 권고

'15. 01. 29 / KISA 취약점점검팀

□ 개요

o 미국 보안회사 'Qualys'에 의해 Linux 시스템의 GNU C 라이브러리(glibc)의 특정 함수에서 임의코드를 실행할 수 있는 취약점이 공개(2015.1.27)

※ 해당 취약점은 'CVE-2015-0235' 지정, 도메인 네임을 IP로 변환하는 기능이 포함된 서비스(메일, 웹 등)들은 해당 취약점에 영향을 받을 수 있음

□ 취약점 상세분석

o 리눅스 계열에서 사용하고 있는 glibc 라이브러리에 존재하는 __nss_hostname_digits_dots 함수에서 잘못된 메모리 사용으로 인해 메모리 변조 가능

※ glibc 라이브러리 : GNU C 라이브러리로 리눅스 운영체제에서 기본적으로 사용하고 있는 라이브러리

※ __nss_hostname_digits_dots 함수 : 도메인 주소를 IP 주소로 변환하는 gethostbyname 함수 호출 시 호스트 이름이 ip형태일 경우, 내부적으로 호출되는 함수

o __nss_hostname_digits_dots 함수 내 메모리 사이즈 연산이 잘못되어 메모리 복사과정에서 오버플로우 발생

```
int __nss_hostname_digits_dots(const char *name, struct hostent *resbuf, char
                    **buffer, size_t *buffer_size, size_t buflen, struct hostent
                    **result, enum nss_status *status, int af, int *h_errnop)
```

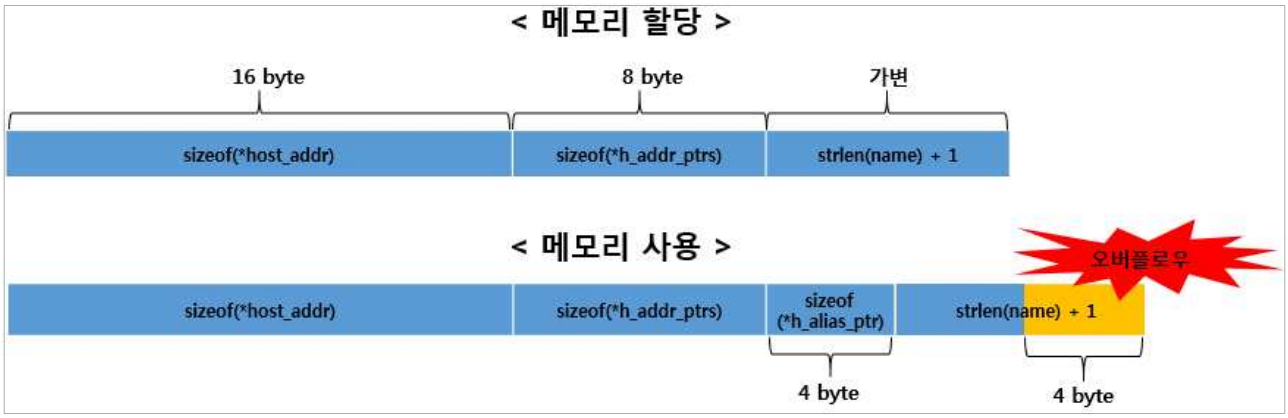
```
....
// *buffer_size는 기본적으로 1024 byte, *buffer는 기본 사이즈로 할당되어 있음
// size_needed가 *buffer_size보다 큰 경우에는 *buffer를 size_needed만큼 메모리 재 할당
size_needed = ( sizeof(*host_addr) + sizeof(*h_addr_ptrs) + strlen(name) + 1 );


....
host_addr = (host_addr_t *)*buffer;
h_addr_ptrs = (host_addr_list_t *)( (char *)host_addr + sizeof(*host_addr) );
h_alias_ptr = (char **)( (char *)h_addr_ptrs + sizeof(*h_addr_ptrs) );
```

```
// hostname의 메모리 시작주소 연산 시,
// size_needed 에 포함되지 않은 sizeof(*h_alias_ptr) 추가(4byte)
hostname = (char *)h_alias_ptr + sizeof(*h_alias_ptr);

....
// 추가된 sizeof(*h_alias_ptr)에 의해 strcpy 시 4바이트 오버플로우 발생(32bit 기준)
resbuf->h_name = strcpy(hostname, name);
....
```

[취약한 glibc 소스코드]



※ 32bit 운영체제 기준 4byte 오버플로우

o 취약점에 영향 받는 프로그램에서 공격자가 조작한 문자열을 인자로 gethostbyname 함수를 호출할 경우, 메모리 변조를 통해 임의코드 실행 가능

□ 영향 받는 버전

o glibc 2.2~2.17 버전의 모든 리눅스 시스템

※ glibc 2.18~2.20 버전, 2.1.3 이하버전은 취약하지 않음(2.2와 2.20은 같지 않음)

glibc 버전 확인 방법

getconf –a| grep glibc
`park@ubuntu:/$ getconf -a| grep glibc`
`GNU_LIBC_VERSION glibc 2.15`

ldd —version
`park@ubuntu:/$ ldd --version`
`ldd (Ubuntu EGLIBC 2.15-0ubuntu10) 2.15`
`Copyright (C) 2012 Free Software Foundation, Inc.`
`This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.`
`Written by Roland McGrath and Ulrich Drepper.`

o 업데이트된 후 버전이 바뀌지 않는 일부 운영체제도 있으니 취약점 확인 방법으로 패치 검증 권고

□ 취약점 확인 방법

o glibc를 사용하는 시스템에서 터미널 프로그램 실행 후 아래내용을 GHOST.c로 저장

```
/* ghosttest.c:  GHOST vulnerability tester */
/* Credit: http://www.openwall.com/lists/oss-security/2015/01/27/9 */
#include <netdb.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#define CANARY "in_the_coal_mine"
struct {
  char buffer[1024];
  char canary[sizeof(CANARY)];
} temp = { "buffer", CANARY };
int main(void) {
  struct hostent resbuf;
  struct hostent *result;
  int herrno;
  int retval;
  /*** strlen (name) = size_needed - sizeof (*host_addr) - sizeof (*h_addr_ptrs) - 1;
***/
  size_t len = sizeof(temp.buffer) - 16*sizeof(unsigned char) - 2*sizeof(char *) - 1;
  char name[sizeof(temp.buffer)];
  memset(name, '0', len);
  name[len] = '\0';
  retval = gethostbyname_r(name, &resbuf, temp.buffer, sizeof(temp.buffer), &result,
&herrno);
  if (strcmp(temp.canary, CANARY) != 0) {
    puts("vulnerable");
    exit(EXIT_SUCCESS);
  }
  if (retval == ERANGE) {
    puts("not vulnerable");
    exit(EXIT_SUCCESS);
  }
  puts("should not happen");
  exit(EXIT_FAILURE);
}
```

o GHOST.c 파일을 아래의 명령어로 컴파일 실행

```
gcc –o GHOST GHOST.c
```

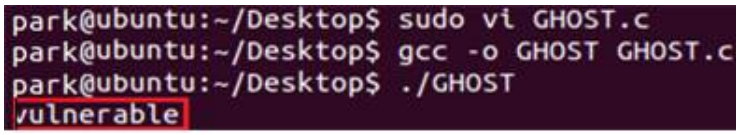
[GHOST.c 컴파일 실행]

※ gcc가 설치되어 있지 않은 경우, 'sudo apt-get install gcc' 명령어로 설치

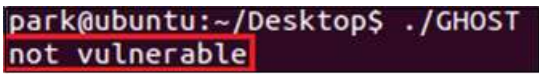
o 아래의 명령어로 실행파일 실행 후 취약점 여부 확인

```
./GHOST
```

[GHOST 실행]



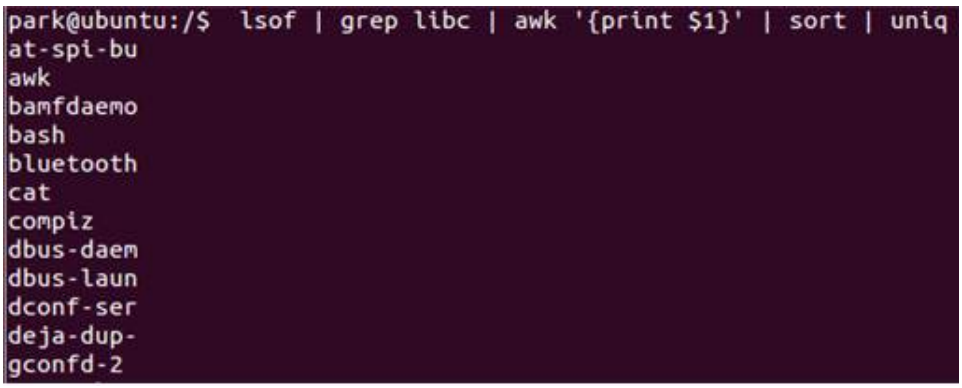
[CVE-2015-0235에 취약한 화면]



[CVE-2015-0235에 취약하지 않은 화면]

o glibc 라이브러리에 의존하는 패키지 및 어플리케이션 확인 방법

```
lsof | grep libc | awk '{print $1}' | sort | uniq
```



[lsof 명령어를 이용한 glibc 라이브러리 확인 방법]

□ 해결 방법

<업데이트 버전 설치>

o 취약한 glibc 버전을 사용하고 있는 경우, 운영체제 제조사 홈페이지를 방문하여 패치 방법 확인

CentOS	http://lists.centos.org/pipermail/centos/2015-January/149413.html
Debian	https://security-tracker.debian.org/tracker/CVE-2015-0235
Redhat	https://rhn.redhat.com/errata/RHSA-2015-0090.html
Ubuntu	https://launchpad.net/ubuntu/+source/eglibc
GUN C Library	http://www.gnu.org/software/libc/

o 패키지 버전 확인 방법

OS 종류	패키지 버전 확인 방법
CentOS / Redhat	**> rpm -qa | grep glibc**
Ubuntu / Debian	**> apt-cache show eglibc-source | grep Version**

※ 운영체제 제조사 홈페이지를 방문하여 최신 버전을 확인하고, 아래 방법을 이용하여 현재 버전을 확인 후 최신 버전이 아닌 경우, 패키지 업데이트

o 패키지 업데이트 방법

OS 종류	패키지 업데이트 방법
CentOS / Redhat	**> yum install glibc**
Ubuntu / Debian	**> apt-get clean && apt-get update && apt-get upgrade**

※ 패키지 업데이트 후 시스템 재부팅 수행

<상위 버전 라이브러리로 컴파일>

o 취약한 버전의 라이브러리를 포함하여 컴파일된 프로그램의 경우 상위 버전의 라이브러리로 재 컴파일이 필요