

BSc (Hons) in Information Technology
IE1212 – System & Network Programming
Year 2 : Semester I : 2020



CVE-2015-0235
GHOST
VULNERABILITY

Name: N.K.A.Ganushaka Omesh
Reg No: IT18223118
Year 2 Semester 1

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

Introduction

A vulnerability in GNU C Library (glibc), referred to as the vulnerability in GHOST, was announced to the general public on January 27, 2015. To sum up, the vulnerability allows remote attackers to take full control of a system by exploiting a buffer overflow bug in the GetHOST (hence the name) functions of glibc. Unlike Shellshock and Heartbleed, this vulnerability is serious and has many servers affected.

The vulnerability of GHOST may be exploited on Linux systems which use GNU C Library versions prior to glibc-2.18. In other words, systems using an unpatched version of glibc from versions 2.2 to 2.17 are at risk.

Many Linux distributions, like but not limited to, are potentially vulnerable to GHOST and should be

patched: CentOS 6 & 7, Debian, 7 Red Hat Enterprise Linux 6 & 7, Ubuntu 10.04 & 12.04. It is strongly recommended that all of the affected Linux servers are modified and rebooted. We'll show you how to test if your systems are vulnerable, and how to update glibc to fix the vulnerability if they are vulnerable.



What is ghost vulnerability(CVE-2015-0235)?

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

The GHOST vulnerability is a serious weakness in the Linux glibc library. It allows attackers to remotely take complete control of the victim system without having any prior knowledge of system credentials. CVE-2015-0235 has been assigned to this issue.

Ghost is a security flaw based on a buffer overflow caused by the "gethostbyname", "gethostbyname2", "gethostbyaddr" functions of glibc 2.2 libraries and those prior to version 2.18 of Linux. This flaw allows an attacker to execute, depending on the context, arbitrary code and thus to take control of the system. This buffer overflow vulnerability can be triggered both locally and remotely. It concerns servers, routers and NAS using Linux.[1]

Only the exploit offered by Qualys is open to the public as of the date of publication of this report, but it only triggers a denial of service. In addition, **no exploitation code for taking control of a remote machine is yet being suggested**. Qualys confirmed it has a tool for remotely taking control of an Exim server. This code will, however, only be released if Qualys predicts that most of the systems on the Internet will be changed.[2]

History of CVE-2015-0235

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

The first appearance of this type of flaw in glibc dates back to 2000 and was corrected on May 21, 2013, between versions 2.17 and 2.18 of glibc . However, the risk for the security of the systems having been underestimated, most of the stable Linux distributions benefiting from support did not immediately apply a correction to their library. The reason advanced by the editors is that of a problem of compatibility .

The company Qualys, supplier of security information, in October 2014, discovered ghost. Before disclosing the flaw officially the January 18, 2015, Qualys reported the problem to Linux distribution editors, who quickly released fixes. Updates are available for [Debian](#) , [Ubuntu](#) and [Red Hat](#) . Given the number of glibc-based applications, Ghost has since January 27, 2015 considered a severe vulnerability, which should be corrected immediately. The Ghost vulnerability is today classified by the American NCAS at the value 10, that is to say at the high level, and carries the reference CVE-2015-0235. [2]

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

Founders of CVE-2015-0235

Alexander Peslyak better known as **Solar Designer**, is a security specialist from Russia. He is best known for his publications on exploitation techniques, including the return-to-libc attack and the first generic heap-based buffer overflow exploitation technique, as well as computer security protection techniques such as privilege separation for daemon processes. In 2015 Qualys acknowledged his help with the disclosure of a GNU C Library `gethostbyname()` function buffer overflow (CVE-2015-0235). [3]

Why is it called the GHOST vulnerability?

It is called as the GHOST vulnerability as it can be triggered by the GetHOST functions. [1]

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

GHOST EXPLAINED

The weakness in GHOST is linked to network names and numbers, as it happens. The originate from functions of the device where the vulnerable code was found.

The functions are called `gethostbyname()` and `gethostbyname2()`, do as the names mean. They will distinguish a host's IP Address (e.g. 93. 184. 216. 34) from their Domain name (e.g. example.com).

In other words, these functions do a DNS (domain name servers) search for users, so the software need not cover the intricacies of the DNS protocol. [4]

```
#include<netdb.h>
#include<stdio.h>

int main()
{
    struct hostent *result;
    unsigned char *host;

    result = gethostbyname("example.com");
    host=(unsigned char *)result->h_addr;

    printf("%u.%u.%u.%u\n",host[0],host[1],host[2],host[3]
);

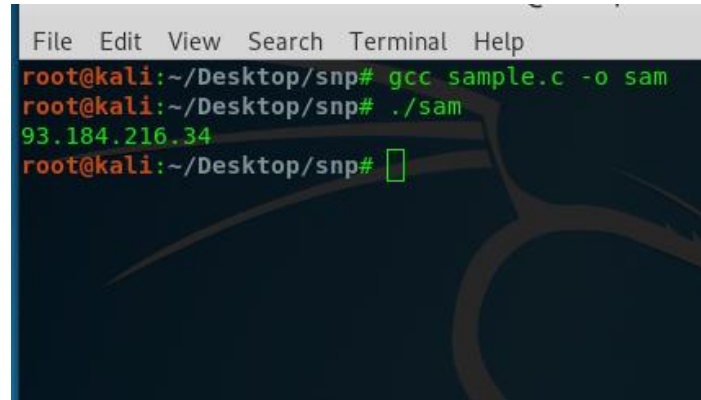
    return 0;
}
```

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

And execution of above code looks something like this:



```
File Edit View Search Terminal Help
root@kali:~/Desktop/snp# gcc sample.c -o sam
root@kali:~/Desktop/snp# ./sam
93.184.216.34
root@kali:~/Desktop/snp#
```

By the way, even if your software does not call `gethostbyname()` directly, you might end up calling it indirectly as a side-effect of doing anything, something, with a computer name involved.

For example, if your program looks up email addresses, calls for updates retrieves posts from online forums or any of the network-related behaviors that are completely unexceptional, it almost certainly causes name-to-number lookups at some point.

So if these lookups are focused on data obtained from the outside, like the email address of a sender in the obtained email headers, then hackers might very well be able to choose which data is passed on to the `gethostbyname()` function of your Linux machine.[4]

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

How the Bug is worked?

It works out that `gethostbyname()` has a smart functionality, where it figures out if users call it with name that is already a IP Address .In that scenario, running a DNS search will be a waste of time so it doesn't concern.

Unfortunately, the code running through the name to see whether it's a IP Address has a buffer overflow, so if user intentionally send an extremely long number laid out correctly. Then an attacker might be able to trigger messages or network requests that crash the program; and with a bit (or, more likely, a lot) of trial and error, they might be able to trigger the crash in a way that gives them the chance over the computer. This is recognized as an exploit of Remote Code Execution (RCE).[4]

What versions and operating systems are affected?

The first vulnerable version of the GNU C Library affected by this is `glibc-2.2`, released on November 10, 2000. We identified a number of factors that mitigate the impact of this bug. In particular, we discovered that it was fixed on May 21, 2013 (between the releases of `glibc-2.17` and `glibc-2.18`). Unfortunately, it was not recognized as a

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

security threat; as a result, most stable and long-term-support distributions were left exposed including Debian 7 (wheezy), Red Hat Enterprise Linux 6 & 7, CentOS 6 & 7, Ubuntu 12.04, for example. [5]

What are the bad effects of this vulnerability?

Successful exploitation of this vulnerability can result in remote code execution, so it has the potential to be pretty bad. This issue can also be exploited locally in some cases, allowing an unprivileged user to gain additional access. In contrast to a vulnerability like Heart bleed, this issue is not always exploitable. In fact, in a general sense, this is not an easy bug to exploit.

The `gethostbyname()` function calls are used for DNS resolving, which is a very common event. To exploit this vulnerability, an attacker must trigger a buffer overflow by supplying an invalid hostname argument to an application that performs a DNS resolution. There is a remote code execution risk due to this vulnerability. An attacker who exploits this issue can gain complete control of the compromised system.

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

What is glibc?

The GNU C Library or glibc is an implementation of the standard C library and a core part of the Linux operating system. Without this library a Linux system will not function. This means any program that links to this library and calls either of the two vulnerable “get host” functions is vulnerable. [6]

How to check what C library (Glibc) version do we use in the Linux system?

The easiest way to check the version number is to run the following command: [7]

```
ldd --version
```

Sample outputs from Ubuntu Linux 12.04.5 LTS:

```
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.
omesh@ubuntu:~$
```

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

A list of affected Linux platforms

- RHEL (Red Hat Enterprise Linux) version 5.x, 6.x and 7.x
- CentOS Linux version 5.x, 6.x & 7.x
- Ubuntu Linux version 10.04, 12.04 LTS
- Debian Linux version 7.x
- Linux Mint version 13.0
- Fedora Linux version 19 or older
- SUSE Linux Enterprise 11 and older (also OpenSuse Linux 11 or older versions).
- SUSE Linux Enterprise Software Development Kit 11 SP3
- SUSE Linux Enterprise Server 11 SP3 for VMware
- SUSE Linux Enterprise Server 11 SP3
- SUSE Linux Enterprise Server 11 SP2 LTSS
- SUSE Linux Enterprise Server 11 SP1 LTSS
- SUSE Linux Enterprise Server 10 SP4 LTSS
- SUSE Linux Enterprise Desktop 11 SP3
- Arch Linux glibc version <= 2.18-1 [7]

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

GHOST vulnerability check (exploit)

You may test or replicate the bug using C code below:

```
#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);
}
```

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

Compile the program and run: [7]

```
$ gcc ghosttest.c -o ghosttest  
$ ./ghosttest
```

Sample outputs from patched Kali Linux v 19: [7]

```
root@kali:~/Desktop/snp# vi GHOST.c  
root@kali:~/Desktop/snp# gcc GHOST.c -o ghost  
root@kali:~/Desktop/snp# ./ghost  
not vulnerable  
root@kali:~/Desktop/snp#
```

Sample outputs from unpatched Ubuntu version 12.04 LTS: [7]

```
omesh@ubuntu:~/Desktop/testRun$ gcc GHOST.c -o ghost  
omesh@ubuntu:~/Desktop/testRun$ ./ghost  
vulnerable  
omesh@ubuntu:~/Desktop/testRun$
```

BSc (Hons) in Information Technology

IE1212 – System & Network Programming


Year 2 : Semester I : 2020

How do list applications depend upon vulnerable Glibc?

We can see vulnerable services by typing the following command:

If the system uses a vulnerable version of glibc user, by executing the following command as root, you can find out the program is affected by this vulnerability (execution by a non-root user will give incorrect results) [7]

```
root@nas01:/# lsof | grep libc | awk '{print $1}' | sort | uniq
acpid
atd
atop
awk
bash
console-k
cron
dbus-daem
dnsmasq
gdbus
getty
gmain
grep
init
in.tftpd
lsof
mdadm
named
polkitd
rpcbind
rpc.idmap
rpc.mount
rpc.statd
rs:main
rsyslogd
sm-notify
sort
ssh-agent
sshd
udev
unig
vnstatd
```



A list of all the services/binaries that rely on the glibc libraries. A reboot is necessary to fix the GHOST on a Linux based server/desktop/laptop.

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

Fix CVE-2015-0235 on the Ubuntu Linux [7]

Type the following apt-get command as the root user:

```
sudo apt-get clean
sudo apt-get update
sudo apt-get upgrade
## only run dist-upgrade on a Ubuntu if you want to upgrade kern
##sudo apt-get dist-upgrade
```

Finally, reboot Ubuntu Linux server by typing the following command and you can see the output: [7]

```
sudo reboot
```

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

GHOST Vulnerability Test to See If a Linux OS Is Secure

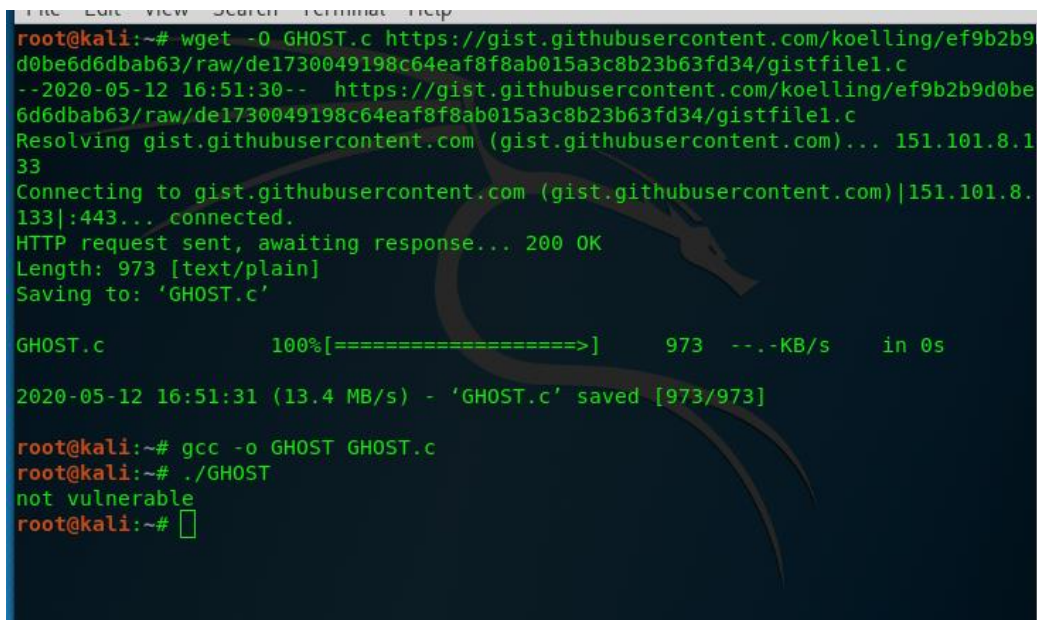
There is another method to test and find out if the Linux server or desktop powered by Linux is secure or not. This is a method that remotely accesses (this time code is in a server machine) the code and execute. [7]

GHOST.C Glibc Vulnerability Test

Type the following wget command to download GHOST.C on a Linux based system:

```
wget -O GHOST.c https://gist.githubusercontent.com/koelling/ef9b2b9d0be6d6dbab63/raw/de1730049198c64eaf8f8ab015a3c8b23b63fd34/gistfile1.c
```

Compile it and execute the command. So you can see the output:



```
root@kali:~# wget -O GHOST.c https://gist.githubusercontent.com/koelling/ef9b2b9d0be6d6dbab63/raw/de1730049198c64eaf8f8ab015a3c8b23b63fd34/gistfile1.c
--2020-05-12 16:51:30-- https://gist.githubusercontent.com/koelling/ef9b2b9d0be6d6dbab63/raw/de1730049198c64eaf8f8ab015a3c8b23b63fd34/gistfile1.c
Resolving gist.githubusercontent.com (gist.githubusercontent.com)... 151.101.8.133
Connecting to gist.githubusercontent.com (gist.githubusercontent.com)|151.101.8.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 973 [text/plain]
Saving to: 'GHOST.c'

GHOST.c          100%[=====>]      973  --.-KB/s   in 0s

2020-05-12 16:51:31 (13.4 MB/s) - 'GHOST.c' saved [973/973]

root@kali:~# gcc -o GHOST GHOST.c
root@kali:~# ./GHOST
not vulnerable
root@kali:~#
```


BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

CONCLUSION

This weakness cannot be the last to be found at glibc. Rapid patching of any application can be a difficult activity due to the comprehensive library use. By having good, tested patching procedures, both now and in the future you can ease the job and will the effort needed. This particular vulnerability is important but it is not so easy to manipulate. There will be patching of the systems at some stage, so make sure that you include this step in a patch management protocol that has been reviewed.

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

References:

[1]A. Sarwate, "The GHOST Vulnerability | Qualys Blog", *Qualys Blog*, 2020. [Online]. Available: <https://blog.qualys.com/laws-of-vulnerabilities/2015/01/27/the-ghost-vulnerability> [Accessed: 13-May- 2020].

[2]"Ghost (Vuln rabilit )", *Fr.wikipedia.org*, 2020. [Online]. Available: [https://fr.wikipedia.org/wiki/Ghost_\(Vuln%C3%A9rabilit%C3%A9\)#Principe](https://fr.wikipedia.org/wiki/Ghost_(Vuln%C3%A9rabilit%C3%A9)#Principe) [Accessed: 13- May- 2020].

[3]"Solar Designer", *En.wikipedia.org*, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Solar_Designer [Accessed: 13- May- 2020].

[4]P. Ducklin, "The GHOST vulnerability – what you need to know", *Naked Security*, 2020. [Online]. Available: <https://nakedsecurity.sophos.com/2015/01/29/the-ghost-vulnerability-what-you-need-to-know/> [Accessed: 13- May- 2020].

[5]*Grahamcluley.com*, 2020. [Online]. Available: <https://www.grahamcluley.com/ghost-vulnerability-faq/> [Accessed: 13- May- 2020].

BSc (Hons) in Information Technology

IE1212 – System & Network Programming

Year 2 : Semester I : 2020

[6]"GHOST –Another scary vulnerability that needs your attention", *Visolve.com*, 2020. [Online]. Available: <https://www.visolve.com/knowledge-hub/articles/ghost-vulnerability.html> [Accessed: 13- May- 2020].

[7]"How To Patch and Protect Linux Server Against the Glibc GHOST Vulnerability # CVE-2015-0235 - nixCraft", *nixCraft*, 2020. [Online]. Available: <https://www.cyberciti.biz/faq/cve-2015-0235-patch-ghost-on-debian-ubuntu-fedora-centos-rhel-linux/> [Accessed: 13- May- 2020].