IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

THE LINUX NEW OLD VULNERABILITIES: TRIO OF BUGS

Mr. Kapil Kumar[1], Ms. Unnati Verma[2]

[1]*Assistant Professor, Department of Computer Science and Engineering, IIMT College of Engineering, Greater Noida, kapilbansal240@gmail.com*

[2]*U.G. Student, Department of Computer Science and Engineering, IIMT College of Engineering, Greater Noida, unnatiwork18@gmail.com*

ABSTRACT

The open-source operating system is free to use, share and modify for the viewers and developers which makes things easier but the attackers looks for the loop-holes. The forgotten vulnerabilities can bring high risk for the resources and opens a gate for the attackers. Over the years, Linux kernel has racked up with a lot of vulnerabilities. The latest found vulnerabilities are such examples which needs to be tackled. Through this paper, we aim to discuss the newly found old bugs in the Linux kernel.

Keywords: *vulnerabilities, linux, kernel, bugs, security, operating system.*

INTRODUCTION

A kernel is the heart of operating system in a computer system. It controls everything in the system. It is basically an interface between the hardware and software
of the computer. Be it disk management, memory management, task management or deciding the state of processes, it takes care of everything happening in the system, Kernel can be monolithic kernel, micro kernel, hybrid kernel, nano kernel and exo kernel.
Every kernel has its own functionalities and features which makes it different from the other.
In this paper, we aim to get into the depth of vulnerabilities and security of monolithic kernel, Linux.

MONOLITHIC KERNEL: LINUX

Monolithic kernels have huge lines of complex code. This type of kernel has its own space where it operates all the operating system services.
Unix, Linux, Open Vms are some of the known and most used open source operating system with monolithic kernel.
Through this paper, we're aim to talk about the Linux.

Linux is an open-source operating system which is freely available. Unlike the closed-source operating system like Windows code which can be altered only by the Microsoft, Linux's code is free to use, code, modify and share under the terms of open source licenses such as MIT, GNU Public licenses, and Apache 2.0.

BACKGROUND OF LINUX

Linux was developed by Linus Torvalds, a Finnish software engineer. He released the first version 0.02 in 1991 which he later modified as the version 1.0, in 1994, which became the heart of the operating system.

Although, Linux is not user-friendly because of being Command-line interface, unlike the Windows and Mac OS that have Graphical User interface, it is efficient and reliable system that rarely crashes.
Red Hat, Debian and Slackware are some of its distribution.

SYSTEM PROTECTION NEED

A computer system has some responsibilities towards its resources. It controls the access of processes, programs and users to secure the resource so that a user can safely share its data in a logical named space such as directory or files. Protection is needed so that only authorized user can access the data. It also makes sure that the resources are available whenever the rightful owner needs to access it.
Open source system, like Linux, brings a benefit of free development in the code which makes the work easy for the developers and viewers to look through it and makes changes, but these benefits also brings the vulnerabilities as even the attackers, who waits for the golden opportunity, can see the same code. They look for the loop-holes and flaws so that they can use it to cause harm, retrieve sensitive data from the system that have not been updated.

In this paper, we would elaborate about such vulnerabilities which was sitting in a forgotten corner of the Linux kernel from past 15 years.

SCSI

SCSI, abbreviation for Small Computer System Interface, is a standard for transporting data within connected computer with peripheral devices, originally via physical devices like a hard drive.
SCSI was published in 1986 for server setup and iSCSI is actually SCSI over TCP (Transmission Control Protocol). While we may or may not have SCSI drives in ages, but it is still around. When dealing with the storage situation, the trio of bugs become an attack surface on default Linux system.

THE TRIO OF BUGS

Linux supports old hardware which makes it reliable but this reliability bring security holes within old programs. That's what happened with the Linux's Small Computer System Interface (SCSI) data transport drive.
A security company, GRIMM researcher found the trio security holes- CVE-2021-27365, CVE-2021-27363, CVE-2021-27364.
These new old bugs which was inside the Linux kernel from 15 years had been sitting inconspicuously in the mainline code. The first two vulnerabilities has Common Vulnerability Scoring System (CVSS) score above 7, which is high, while one of them turned out usable as Local Privilege Escalation (LPE) attack. These flaws have lurked in Linux code since 2006 without detection.
The sprawl of Linux kernel have so much code that the small bugs slipped detection. So, after this researcher Adams Nichols has suggested a strategy to load as little code as possible. When the server needs high throughput, low latency networking for data transfer and storage, RDMA (Random Direct Memory Access) technology comes in the light. It has several implementation but the Infiniband, found in the ib_iser kernel module is related to the topic.
Now the question arise, if we do not use the SCSI or iSCSI will it automatically run. Well, this leads our attention towards the concept of on-demand kernel module loading.

ON-DEMAND KERNEL

To improve the compatibility, the Linux kernel can load kernel on-demand. When a particular code needs some functionalities, the kernel loads it, like support for uncommon protocol families. It brings improvement but opens the fate for the local attackers as unprivileged users gets a way to load dubious kernel module which later can be exploit. In 2009, grsecurity's GRKERNSEC_MODHARDEN was introduced as a defense. The support various between the Linux distribution.

BUGS IDENTIFICATION

The trio of bugs vulnerabilities brings security holes of local elevation of privileges, information leaks and denial of services.

(1) CVE-2021-27365

This vulnerability is found in the version 5.11.3 in Linux kernel. This is a heap buffer overflow type of vulnerability which was found in the iSCSI subsystem. The iSCSI data structures can exceeds the PAGE_SIZE value as it doesn't have appropriate length constraints or checks. It can be triggered by setting an iSCSI string attribute to value larger than one page and then to read it.

The kernel- 4.18.0-240.e18 source code's line 3397 in *drivers/scsi/libiscsi.c* has a sprintf call used for seq file to back the iSCSI attribute which is used on the user supplied value with a single page buffer. The unprivileged user can this send netlink message to the iSCSI subsystem, that sets the attribute like hostname, username, etc related to the iSCSI connection via helper function in *drivers/scsi/libisci.c.*

This bug was introduced in 2006 when the iSCSI subsystem was being developed under commits a54a2caad and fd7255f51a. It can be found in iscsi_host_get_param() in *drivers/scsi/libiscsi.c* which affected the versions tested on RHEL 8.1, 8.2 and 8.3. It can lead to LPE, information leak and denial of services problems by the attackers.

(2) CVE-2021-27363

The other vulnerability discovered by GRIMM, is a kernel pointer leak in the version 5.11.3 of Linux kernel. This vulnerability can determine the address of the isci_transport structure registered with the iscsi subsystem.

Unprivileged users can avail transport's handle via sysfs file system, at *sys/class/iscsi_transport/$TRANSPORT_NAME/handle*. The handle is leaked when the show_transport_handle() function (in *drivers/scsi/scsi.transport_iscsi.c*) is called. In kernel module's global variable, this handle is actually the pointer to an isci_transport structure.

This vulnerability affects the versions tested on RHEL 8.1, 8.2 and 8.3 which can leak information.

(3) CVE-2021-27364

This is the last vulnerability of the trio which is an out-of-bound kernel read in the libiscsi module. It can be triggered via a call to send_pdu which is present in the lines 3747-3750 in *drivers/scsi/scsi_transport_iscsi.c* in the kernel 4.18.0-24.0.el8 source code.

This is somewhat simliar to the CVE-2021-27365 bug which provides the unprivileged user to craft netlink messages. These messages specifies buffer size that leads the driver failure to validate, which thus leads to controllable out-of-bounds read. The calculation of size of preceding header are not validated for multiple user-control values, allowing for a read of up to 8192 bytes at a controllable 32-bit offset from the original head buffer.

This bug affected versions tested on RHEL 8.1, 8.2 and 8.3 and which can cause denial of services and information leak. It was found in iscsi_if_recv_msg() function in *drivers/scsi/scsi_transport_iscsi.c.*

TECHNICAL ANALYSIS

To demonstrate the use of first two vulnerabilities, GRIMM has developed a Proof of Concept (PoC) exploit. PoC, currently, supports for the 4.18.0-147.8.1.el8_1.x86_64, 4.18.0-193.14.3el8_2.x86_64 and 4.18.0_240.el8.x86_64 releases of Linux kernel.

The vulnerability of other versions of Linux kernel is no less, but before exploitation the system address and structure offsets are needed in them.

KASLR Leak

KASLR stands for Kernel Address Space Layout Randomization which is a computer security technology to prevent exploitation of memory corruption vulnerability. This has to be bypassed before modification of kernel structure and function on pointer changing starts. Kernel's base address is randomized in Linux to hinder the exploitation process. However, the local user often bypasses the KASLR because of the numerous sources of local information leak. To bypass KASLR, two separate information is leaked.

The first information leak comes from a non-null terminated heap buffer. The kstrdup function is called when an iSCSI string attribute is set, via the iscsi_switch_str_param function, on the user provided input. However, the user's input is NULL terminated as the buffer containing the user input is not initialized upon allocation thus the kernel does not enforce it. So, the kstrdup function copies any non_NULL bytes after the client input, which, by reading back the attribute, can be later retrieved.

In the second information leak, the target module's iscsi_transport structure address is obtained. Thi structure defines how the target handles the various iSCSI requests. As the target kernel module is in the global region, the information leak can be used to obtain the address of its kernel module.

TESTING

To test the provided PoC exploit, kernel version and RHEL compatibility have to be ensured. To check the release versions with cat/etc/redhat-release and kernel version with uname -r, the exploit will detect and warn if you are on an unsupported version.

CONCLUSION

The discussed vulnerabilities are from a very old driver in Linux kernel but due to a new technology, RDMA, it became a highlight. These vulnerabilities are patched by the vendors, and users are recommended to upgrade to version 8.0.10. The real problem for the Linux kernel is because of the tension between the compatibility and security. The risks needs to be understand by the administrators and operators. As Adams Nicholas has suggested the code should be written as little as possible so that such vulnerabilities could not stay within the kernel for long 15 years which might exploit a lot of resources.

REFERENCE

1. https://resources.whitesourcesoftware.com/blog-whitesource/top-10-linux-kernel-vulnerabilities
2. https://nakedsecurity.sophos.com/2021/03/17/serious-security-the-linux-kernel-bugs-that-surfaced-after-15-years/
3. https://blog.grimm-co.com/2021/03/new-old-bugs-in-linux-kernel.html
4. https://www.smartbrief.com/branded/F6D8DF94-3AEB-4A28-8343-BDDAF6341D54/3B2C2EB6-E5C7-483C-A082-E4022017B145
5. https://nvd.nist.gov/vuln/detail/CVE-2021-27365
6. https://nvd.nist.gov/vuln/detail/CVE-2021-27363
7. https://nvd.nist.gov/vuln/detail/CVE-2021-27364
8. https://www.scmagazine.com/home/security-news/vulnerabilities/three-flaws-that-sat-in-linux-kernel-since-2006-could-deliver-root-privileges-to-attackers/
9. https://penpaperpins.com/a-trio-of-vulnerabilities-in-the-linux-kernel-can-give-attackers-root-privileges
10. https://securitytoday.com/articles/2019/08/19/the-dangers-of-opensource-vulnerabilities-and-what-you-can-do-about-it.aspx#:~:text=The%20reason%20behind%20the%20open,that%20have%20not%20been%20updated.
11. https://www.geeksforgeeks.org/system-protection-in-operating-system/?ref=rp
12. https://www.redhat.com/en/topics/linux/what-is-the-linux-kernel
13. https://www.britannica.com/technology/Linux
14. https://www.zdnet.com/article/what-are-open-source-operating-systems-everything-you-need-to-know/