Group no: - 8

Group Member:- Shrut Shah – 19BCP125

Shubham Kathiriya – 19BCP127

Vedant Patel – 19BCP138

Subject: - Cyber Security Lab

Division:-2

Lab 4:- Vulnerabilities of Operating System

Aim:-

Exploiting the Vulnerabilities of Operating System.

Introduction:-

1. What are Vulnerabilities?

- Vulnerability is effectively an error in the code or the logic of operation within the OS or the application software. Because today's OSs and applications are very complex and include a lot of functionality, it's difficult for a vendor's development team to create software that contains no errors.

- Unfortunately, there’s no shortage of virus creators and cybercriminals that are ready to devote considerable effort to investigating how they can benefit from exploiting any vulnerability – before it’s fixed by the vendor issuing a software patch.

2. Operating System Vulnerabilities?

- CodeRed, Sasser, Slammer and Lovesan (Blaster) are examples of worms that exploited vulnerabilities in the Windows OS – whereas the Ramen and Slapper worms penetrated computers via vulnerabilities in the Linux OS and some Linux applications.

3. Which is Safer, Windows or Linux?

- The obvious answer is Linux. There are a few reasons for the increased security of the Linux platform versus Windows. Firstly, Windows is a closed-source commercial application. It is also created by a profit-driven company. So, it should be no surprise that when Microsoft creates a new version of Windows, they may be cutting corners and rushing to release to obtain revenue streams; they sometimes overlook finer points like security and proper interactions between applications that ensure a secure, stable environment. If this seems strange, it is the Agile or Xtreme programming methodology: get a working product to consumers and patch problems afterwards.
- Secondly, Linux is open source. That allows the community to see and test the underlying code as well as seeing where updates get installed. The

testing is done by the community, which can often catch things before they become a problem.

- Finally, Linux has lower usage statistics for desktop computing. According to statcounter.com, in April 2019, the Linux market share was 1.63% versus Windows, which was 79.24%. So, writing viruses to target Linux systems for desktops isn't as lucrative. Thus, it has lower appeal.
- It is important to note that Android is based on the Linux operating system and is currently the most used operating system in the world. There is more vulnerability for Android than forLinux desktop systems. This is due to the popularity of the OS and the fragmented population of devices that are typically managed by the manufacturer (i.e., Samsung, Huawei, etc.). Worldwide, Android devices are 75.85% of the mobile market and Windows phones are only .28%. The Linux to Windows ratio is almost exactly the oppositefrom desktop to mobile.

Windows Vulnerabilities:-

- Windows has quite a few vulnerability issues. Things like browser vulnerabilities, issues with mounting of devices (USB, external hard drives, etc.), and even font drivers can all be methods for ingress into the system.

1. Windows Font Type Vulnerability:-

- Open Type Font is used by Adobe type libraries to allow cross-platform compatibility when viewing PDFs. This allows Macs, Linux systems, and Windows to all support and views the same documents. However, the 'atmfd.dll' file in the Adobe Type Manager Library created a buffer underflow that would allow remote code execution. A buffer underflow or underwrite is a data threat that takes place when the data is stored in a temporary holding space (swap, memory, stack, etc.) during a buffer data transfer and is being fed at a slower rate than where it is being read from.

2. Driver Improper Interaction with Windows Kernel Vulnerability:-

- By using a stack-based buffer overflow in the RtlQueryRegistryValues function in win32k.sys, local users can gain privileges and bypass the User Account Control (UAC) feature through a specially crafted REG_BINARY value for a SystemDefaultEUDCFont registry key.

3. Windows Fax Services Cover Page Editor Vulnerability:-

- If you're still faxing, here's another reason to leave that remnant of the 80's behind: a heap-based buffer overflow in the CDrawPoly:
- Serialize function in fxscover.exe in Microsoft Windows Fax Services Cover Page Editor 5.2 r2 could allow remote attackers to execute arbitrary code via a long record in a Fax Cover Page (.cov)file.

4. Win32k Improper Message Handling Vulnerability:-

- Another kernel-mode vulnerability, this flaw also involves win32k.sys in the kernel-mode drivers. In this case, they do not properly handle window broadcast messages and could potentially allow local users to gain privileges through a specially crafted application.

5. Win32k.sys Elevation of Privilege Vulnerability:-

- Windows 7's win32k.sys in kernel-mode drivers could enable local users to gain privileges through a specially crafted application.

6. GDI Access Violation Vulnerability:-

- Primarily a Windows API for displaying graphics, the Graphics Device Interface (GDI) in win32k.sys in the kernel-mode drivers does not properly validate user-mode input. This could give remote attackers the ability to execute arbitrary code or cause a memory corruption denial of service (DoS) with specially crafted data.

7. Insecure Library Loading Vulnerability:-

- Windows Address Book (WAB) is a component that allows users to use a single list of contacts shared across multiple applications.

- Unfortunately, an untrusted search path vulnerability in wab.exe 6.00.2900.5512 in WAB could allow a local attacker to gain privileges via a Trojan horse wab32res.dll file in the current working directory.

Linux Vulnerabilities:-

1. Buffer overflow impacting python 27, python 36 and python 38:-

- A stack-based buffer overflow was discovered in the ctypes module provided within Python. Applications that use ctypes without carefully validating the input passed to it may be vulnerable to this flaw, which would allow an attacker to overflow a buffer onthe stack and crash the application.

- The highest threat from this vulnerability is to system availability.
- This vulnerability has a critical risk as this can be exposed over any network, with low complexity, no privileges and without user interaction.

2. xterm vulnerability with Ubuntu 20.10, 20.04 LTS, 18.04 LTS and 16.04 LTS and Red Hat Enterprise Linux 8:-

- Xterm incorrectly handles certain character sequences. A remote attacker could use this issue to cause xterm to crash, resulting ina denial of service, or possibly execute arbitrary code.
- This vulnerability has a critical risk as this can be exposed over any network, with low complexity, no privileges and without user interaction.

3. Screen update for SUSE Enterprise Server 12-SP2 to 12-SP5:-

- Fixed double width combining char handling (UTF-8 character) that could lead to a denial of service or code execution.
- This vulnerability has a critical risk as this can be exposed over any network, with low complexity, no privileges and without user interaction.

4. Security Update for Mozilla Thunderbird:-

- Memory safety bugs present in Firefox 85 and Firefox ESR 78.7. Some of these bugs showed evidence of memory corruption and we presume that with enough effort some of these could have been exploited to run arbitrary code. This vulnerability affects Firefox < 86, Thunderbird < 78.8, and Firefox ESR < 78.8.

- This vulnerability has a high risk as this can be exposed over any network, with low complexity, no privileges but does require user interaction.

5. **Kernel Update for Oracle Linux 6 and 7**:-

- Xen is vulnerable to a denial of service, caused by error handling issues in mapping. A local attacker could exploit this vulnerability to crash the corresponding backend driver, potentially affecting the entire domain running the backend driver.
- This vulnerability has a moderate risk as this can be exposed overa local network, with low complexity, no privileges and withoutuser interaction.

Mac Vulnerabilities:-

1. **Dock Vulnerability**:-

- The Dock in Apple OS X versions before 10.10 improperly manages the screen-lock state. This could allow attackers in physical proximityto access an unattended workstation. Newer versions of OS X do not have this flaw, so upgrading to a newer version effectively remediates the vulnerability.

2. **Mail Vulnerability**:-

- Versions of Mail before 10.10 do not properly recognize the removal of a recipient address from a message. This could allow remote attackers to obtain sensitive information by reading a message intended exclusively for other recipients.

3. **Launch Services Vulnerability**:-

- Launch Services in OS X before 10.10.3 could allow local attackers to cause a denial-of-service (Finder crash) via specially crafted localization data.

4. **App Store Vulnerability**:-

- The App Store process in Commerce Kit Framework in OS X before 10.10.2 places Apple ID credentials in App Store logs, which could allow local users to obtain sensitive information by simply readingthe log files.

5. **PDF Password Vulnerability**:-

- The UserAccountUpdater in OS X 10.10 before 10.10.2 stores a PDF document's password in a printing preference file, allowing local users to obtain sensitive information by reading said file.

6. **User Documentation Vulnerability**:-

- The User Documentation component in OS X through 10.6.8 uses HTTP sessions for updates to App Store help information. This couldallow a man-in-the-middle attacker to execute arbitrary code by spoofing the HTTP server.

7. Error Logging Vulnerability:-

- New error logging features in OS X 10.10 that include unsafe additions to the dynamic linker could allow local attackers to gainunfettered root privileges.

Conclusion:-

- A vulnerability is effectively an error in the code or the logic of operation within the OS or the application software. Because today’s OSs and applications are very complex and include a lot of functionality, it’s difficult for a vendor’s development team to create software that contains no errors.