CSOC 1020: Assignment #4(Initial Access)

August 15, 2023

**Prepared By**: *Ashish Kishor Hedau*

Table of Contents

[Unpatched version of Simple PHP Photo Gallery 2](#_Toc137283759)

[Description 2](#_Toc137283760)

[Impact 2](#_Toc137283761)

[Recommendations 2](#_Toc137283762)

[Steps to Reproduce 3](#_Toc137283763)

# Unpatched version of Simple PHP Photo Gallery

## Description

The assessed “http://192.168.195.58” application was found to implement an unpatched Simple PHP Photo Gallery (allows user to instantly create a photo gallery just by dragging and dropping .jpg photos into the image folder). The photo gallery v0.8 is vulnerable to remote code execution since the software does not adequately validate or sanitize user-provided input.

## Impact

An attacker can manipulate the URL request on client-side and upon receiving the malicious request, the server could inadvertently execute the attacker’s provided code. Depending on the extent of the vulnerability, the attacker might gain unauthorized access to the system, execute arbitrary commands, or perform actions they shouldn’t be able to perform.

## Recommendations

Update the Simple PHP Photo Gallery to the latest version available. Implement robust server-side input validation and sanitization for all user-provided inputs, ensuring that any data submitted to the server is thoroughly checked for validity and security. Regularly review and update the server and application configurations. Disable unnecessary services, ports, and features to minimize the attack surface.

## Steps to Reproduce

We started the enumeration process of the IP in scope (192.168.195.58) with a nmap scan “**nmap -sV -vv -Pn -n -p- 192.168.195.58**”.

![A screen shot of a computer

Description automatically generated]()

The web app running on port 80 has Simple PHP Photo Gallery v0.8 as highlighted below.

![A screenshot of a computer

Description automatically generated]()

The image below shows an exploit on exploitDB for Remote File Inclusion affecting the Simple PHP photo gallery.

![A screenshot of a computer

Description automatically generated]()

The highlighted box shows the path to exploit the vulnerability.

![A screenshot of a computer

Description automatically generated]()

We crafted a reverse shell payload and named it as **php\_reverse.php.**

![A black background with white text

Description automatically generated]()

We used the exploit and successfully connected back to the server hosted at our local machine (192.168.45.204).

![A computer screen shot of a website

Description automatically generated]()

For the final execution, we started the nc listener on port 445 and executed our php\_reverse.php payload in the URL.

![A screenshot of a computer

Description automatically generated]()

CSOC 1020: Assignment #4(Privilege Escalation)

August 15, 2023

**Prepared By**: *Ashish Kishor Hedau*

Table of Contents

[Pkexec utility Read and Write vulnerability 2](#_Toc137283759)

[Description 2](#_Toc137283760)

[Impact 2](#_Toc137283761)

[Recommendations 2](#_Toc137283762)

[Steps to Reproduce 3](#_Toc137283763)

## Pkexec utility Read and Write Vulnerability

## Description

The assessed IP in scope was found to have a local privilege escalation vulnerability on polkit’s pkexec utility. The pkexec application is a setuid tool designed to allow unprivileged users to run commands as privileged users according to predefined policies. The current version of pkexec doesn’t handle the calling parameters count correctly and ends up trying to execute environment variables as commands.

Source: <https://nvd.nist.gov/vuln/detail/cve-2021-4034>

## Impact

An attacker can exploit this vulnerability by carefully crafting environment variables in such a way that they appear to be valid commands. When the vulnerable pkexec is invoked, it could incorrectly interpret these crafted environment variables as commands to be executed with elevated privileges. This can allow them to perform actions that are otherwise restricted, such as modifying system files, installing software, or creating new user accounts.

## Recommendations

We recommend system administrators to monitor system logs for any suspicious activity, especially related to pkexec usage. Regularly audit system configurations to identify any unauthorized changes. Limit the use of privileged operations to only those users who truly need them and follow the principle of least privilege.

## Steps to Reproduce

We started the enumeration by downloading linpeas.sh into the target machine. We hosted a python server at our local machine to achieve this.

**Note:** /tmp directory was used to download linpeas.sh as we were unable to download it in different directories.

![A screenshot of a computer

Description automatically generated]()

After running linpeas.sh, the output revealed the CVE 2021-4034 vulnerability.

![A screenshot of a computer

Description automatically generated]()

We searched on github for CVE 2021-4034 vulnerability and found an exploit.

![A screenshot of a computer

Description automatically generated]()

We ran the exploit.py found earlier and were successful in elevating privileges to ‘**root**’.

![A screenshot of a computer

Description automatically generated]()