

Agent-based Integration

- [Auto Installer in Atlassian Jira Integration via Log Source](#)
- [Auto Installer in Atlassian Confluence Integration via Log Source](#)
- [Auto Installer in Aquila CATO Network Integration via Linux](#)
- [Auto Installer in Aquila CATO Network Integration Windows](#)
- [Auto Installer of Intercom Activity Logs](#)

Auto Installer in Atlassian Jira Integration via Log Source

Aquila–Atlassian Jira Integration via log source automation

An Auto Installer for Aquila–Atlassian Jira Integration via log source automation builds upon the structured authentication and API connectivity defined in the [referenced guide](#), but shifts the configuration process into a fully automated workflow. Instead of manually executing each setup step, the integration leverages automated scripts or installers to establish secure communication with Jira using OAuth 2.0 (3LO) and API keys, enabling systems to authenticate and interact with Jira APIs seamlessly. OAuth 2.0 serves as a secure authorization mechanism that allows external applications to access Jira resources on behalf of a user or service account without exposing credentials, while API keys provide an additional method for authenticated requests . By embedding these processes into automation, the installer can dynamically configure credentials, register applications, and connect log sources to Jira, ensuring that system events are automatically transferred to our back-end. This approach not only simplifies deployment but also enhances consistency, scalability, and real-time incident tracking within integrated environments.

Prerequisites:

- **Service User (linux user)**
- **Service Group (linux group)**
- **Atlassian Jira Cloud ID**
- **Atlassian Jira User Email (Admin)**
- **API key**

Creating the setup installer

Linux:

1. **Login to the Log Collector.**
2. **Open Terminal and type the following commands.**

- `cd Documents`

- `nano setup.sh`

3. **Paste this Script** *Contact Aquila Tech Support for the full source code.

- Please Contact "Aquila Platform Defence and Technical Enablement" for the source code. Thank you!.

4. **Save the Script and change it to executable file.**

- `chmod +x setup.sh`

5. **Now execute the file in higher privileged.**

- `sudo ./setup.sh`

6. **Follow the on screen instruction to fully integrate the system.**

7. **Restart the service.**

```
sudo systemctl daemon-reload
```

```
sudo systemctl restart jira-audit.service
```

If you need further assistance, kindly contact our support at support@cytechint.com for prompt assistance and guidance.

Auto Installer in Atlassian Confluence Integration via Log Source

Aquila–Atlassian Confluence Integration via log source automation

An Auto Installer for Aquila–Atlassian Confluence Integration via log source automation builds upon the structured authentication and API connectivity defined in the [referenced guide](#), but shifts the configuration process into a fully automated workflow. Instead of manually executing each setup step, the integration leverages automated scripts or installers to establish secure communication with Confluence using OAuth 2.0 (3LO) and API keys, enabling systems to authenticate and interact with Confluence APIs seamlessly. OAuth 2.0 serves as a secure authorization mechanism that allows external applications to access Confluence resources on behalf of a user or service account without exposing credentials, while API keys provide an additional method for authenticated requests. By embedding these processes into automation, the installer can dynamically configure credentials, register applications, and connect log sources to Atlassian Confluence, ensuring that system events are automatically transferred to our back-end. This approach not only simplifies deployment but also enhances consistency, scalability, and real-time incident tracking within integrated environments.

Prerequisites:

- **Service User (linux user)**
 - **Service Group (linux group)**
 - **Atlassian Confluence Cloud ID**
 - **Atlassian Confluence User Email (Admin)**
 - **API key**
-

Creating the setup installer

Linux:

1. **Login to the Log Collector.**
2. **Open Terminal and type the following commands.**

- `cd Documents`

- `nano setup.sh`

3. Paste this Script *Contact Aquila Tech Support for the full source code*.

- Please Contact "Aquila Platform Defence and Technical Enablement" for the source code. Thank you!.

4. Save the Script and change it to executable file.

- `chmod +x setup.sh`

5. Now execute the file in higher privileged.

- `sudo ./setup.sh`

6. Follow the on screen instruction to fully integrate the system.

If you need further assistance, kindly contact our support at support@cytechint.com for prompt assistance and guidance.

Auto Installer in Aquila CATO Network Integration via Linux

Overview

The Aquila CATO Auto Installer automates the deployment and configuration of the CATO Linux Client for secure network connectivity. It simplifies installation, service setup, auto-connect configuration, and startup persistence across Linux environments.

Reference: [Aquila CATO Network Integration \(Linux\)](#)

Prerequisites

- **Service User (linux user)**
 - **Service Group (linux group)**
 - **Python Version**
 - **Account ID**
 - **Cato API key**
-

Creating the setup installer

Linux:

1. **Login to the Log Collector.**
2. **Open Terminal and type the following commands.**

- `cd Documents`

- `nano setup.sh`

3. **Paste this Script** ***Contact Aquila Tech Support for the full source code.**

- Please Contact "Aquila Platform Defence and Technical Enablement" for the source code. Thank you!.

4. **Save the Script and change it to executable file.**

- `chmod +x setup.sh`

5. Now execute the file in higher privileged.

- `sudo ./setup.sh`

6. Follow the on screen instruction to fully integrate the system.

If you need further assistance, kindly contact our support at support@cytechint.com for prompt assistance and guidance.

Auto Installer in Aquila CATO Network Integration Windows

Overview

The Aquila CATO Auto Installer automates the deployment and configuration of the CATO Linux Client for secure network connectivity. It simplifies installation, service setup, auto-connect configuration, and startup persistence across Windows environments.

Reference: [AQUILA - Cato Network Integration\(Windows\)](#)

Prerequisites

- **Admin Privilege in running Powershell**
 - **Cato Account ID**
 - **Cato API key**
-

Creating the setup installer

Windows:

1. **Login to the Log Collector.**
2. **Open Powershell as Administrator and run this command to bypass execution policy.**

- `Set-ExecutionPolicy -Scope Process -ExecutionPolicy Bypass`

3. **Run the setup provided by cytech **Contact Aquila Tech Support for the full source code..***

- `.\aquila_cato_windows_setup.ps1`

4. **Follow the on screen instruction to fully integrate the system.**
-

If you need further assistance, kindly contact our support at support@cytechint.com for prompt assistance and guidance.

Auto Installer of Intercom Activity Logs

Overview

A lightweight, dependency-free Bash collector that polls the **Intercom Admin Activity Logs API** on an hourly schedule, flattens each event to a dot-notation NDJSON record, rotates output files at a configurable size, and automatically purges old archives — all running as a systemd user service with no root access required.

Prerequisites

- **Intercom API token**
 - **Intercom API Version**
-

Steps to generate an Intercom API Token

1. Go to the official Intercom Developer Hub:
[Intercom Developer Hub](#)
 2. Log in to your Intercom workspace.
 3. Open your workspace's **Developer Hub**:
 - In Intercom, go to:
Settings → Developer Hub → Your Apps
 4. Click **New App** (or “Create App”).
 5. Enter:
 - App name
 - Description (optional)
 6. After creating the app:
 - Open the app
 - Go to **Configure → Authentication**
 7. You'll see your **Access Token** there.
 8. Copy the token and store it securely.
 9. You can also find the API Version and copy as well.
-

Creating the setup installer

Linux:

1. **Login to the Log Collector.**
2. **Open Terminal and type the following commands.**

- `cd Documents`

- `nano setup.sh`

3. **Paste this Script** *Contact Aquila Tech Support for the full source code.

- Please Contact "Aquila Platform Defence and Technical Enablement" for the source code. Thank you!.

4. **Save the Script and change it to executable file.**

- `chmod +x setup.sh`

5. **Now execute the file.**

- `./setup.sh`

6. **Follow the on screen instruction to fully integrate the system.**

7. **Restart the service.**

```
sudo systemctl daemon-reload
```

```
systemctl --user start intercom-collector.service -f
```

***Provide the "activity_logs.ndjson" Directory to Cytech Support for Integration.
Example: /home/user/intercom/activity_logs.ndjson***

For the Back-end add this to Custom configurations:

```
processors:  
  - decode_json_fields:  
      fields: ["message"]  
      target: ""  
      overwrite_keys: true  
      add_error_key: true
```

```
- add_fields:
  target: event
  fields:
    module: intercom
- add_fields:
  target: component
  fields:
    id: intercom-logs
```

If you need further assistance, kindly contact our support at support@cytechint.com for prompt assistance and guidance.