Docker Immich WHMCS module

A module for WHMCS that uses n8n workflows to deploy the Immich service using Docker.

- Description
- What is Immich
- Changelog
- Installation and configuration guide
  - Basic concepts and requirements
  - WHMCS setup (install/update)
  - Preparing Docker Server
  - Setting up n8n workflow
  - Add server
  - Product Configuration
  - Metric Billing
  - Email Template (puqDockerImmich Welcome Email)
  - Email Template (puqDockerImmich Update Email)
  - Email Template (puqDockerImmich Notification disk limit)
  - Configurable Option (CPU/RAM/DISK)
- Admin Area
  - Product Information
- Client Area
  - Home screen
  - IP Access Control
  - Reinstall

- Metrics

Description

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

> Before you start, it is important to read and familiarize yourself with the following articles at this link:
>
> https://doc.puq.info/books/docker-modules

The **WHMCS Docker Immich module** is designed for automated provisioning and management of **Immich** instances on a Docker server. It seamlessly integrates with WHMCS, allowing businesses to sell and manage **Immich** services efficiently.

Key Features

□□ Automated Container Management

- Automatic creation of an **Immich** container upon service order.
- Automated package upgrades and administrator password reset.

□□ Service Control & Security

- Service creation
- Service suspension and reactivation.
- Service termination
- Full reinstallation
- IP access control

□□ Advanced Diagnostic Tools

- Built-in tools for diagnosing and managing containers.

🌍 Multilingual Support

- Supports multiple languages, including **Arabic, Azerbaijani, Catalan, Chinese, Croatian, Czech, Danish, Dutch, English, Estonian, Farsi, French, German, Hebrew, Hungarian, Italian, Macedonian, Norwegian, Polish, Romanian, Russian, Spanish, Swedish, Turkish, and Ukrainian**.

⚙ Fully Customizable Workflows

- Uses **n8n workflows** to automate processes, allowing full customization for business-specific needs.

System Requirements

To run the WHMCS Docker **Immich** module, ensure you have:
✅ **WHMCS version 8+**
✅ **An n8n server** for workflow automation
✅ **A server with Docker installed** for container management (Installation Guide)

Installation & Setup

1️⃣ Environment Preparation

- Install **WHMCS 8+**.
- Set up an **n8n server** for automation workflows.
- Ensure **Docker** is installed and running. (Installation Guide)

2️⃣ Module Installation

- Upload and activate the **WHMCS Docker Immich module**.
- Configure the module settings to connect with your Docker server and n8n workflows.

3 Workflow Customization

- Utilize **n8n workflows** to automate service provisioning and management.
- Modify workflows as needed for custom business logic.

4 Testing & Deployment

- Perform a **test order** to verify automatic container creation.
- Check all service management functions (creation, suspension, unsuspension, termination).

Why Choose This Module?

Seamless automation – Reduces manual work and speeds up service deployment.
Highly customizable – Modify workflows to fit any business model.
User-friendly – Integrated within WHMCS with a simple setup process.

This module makes selling and managing **Immich instances** through WHMCS easy, automated, and flexible!

Manage | Metrics

Go to Immich | User manual

Status:	running
CPU usage:	1 CPU 99.73%
Memory usage:	778.3MiB / 1GiB 76% 24%
Disk usage:	3.0G / 9.8G 32% 68%

immich

https://1-5342.d01-test.uuq.pl/

Change Administrator Password

Version:	2.2.53
Administrator:	it@puq.pl
Users:	

Module Commands

Create | Suspend | Unsuspend | Terminate | Change Package | Container Start | Container Stop | Mount disk | Unmount disk

API Connection status

API Connection OK

Container

Refresh | Log

Status	Running
Name	1-5342.d01-test.uuq.pl_immich (8e14264d1726)
CPU usage	99.52%
Memory usage	776.1MiB / 1GiB 75.79% 24.21%
Disk IO	153MB / 555MB
Disk mounted	3.0G/9.8G 32% 68%
Disk file	3.1G
Network IO	548MB / 1.09GB

Dependent Containers

Refresh

Status	Running
Name	1-5342.d01-test.uuq.pl_ml (30d0e4fb5a29)
CPU usage	99.78%
Memory usage	210.2MiB / 1GiB 20.52% 79.48%

Status	Running
Name	1-5342.d01-test.uuq.pl_db (7987ecf80297)
CPU usage	99.99%
Memory usage	183.9MiB / 1GiB 17.96% 82.04%

Status	Running
Name	1-5342.d01-test.uuq.pl_redis (b3200c47ac80)
CPU usage	99.73%
Memory usage	9.777MiB / 1GiB 0.95% 99.05%

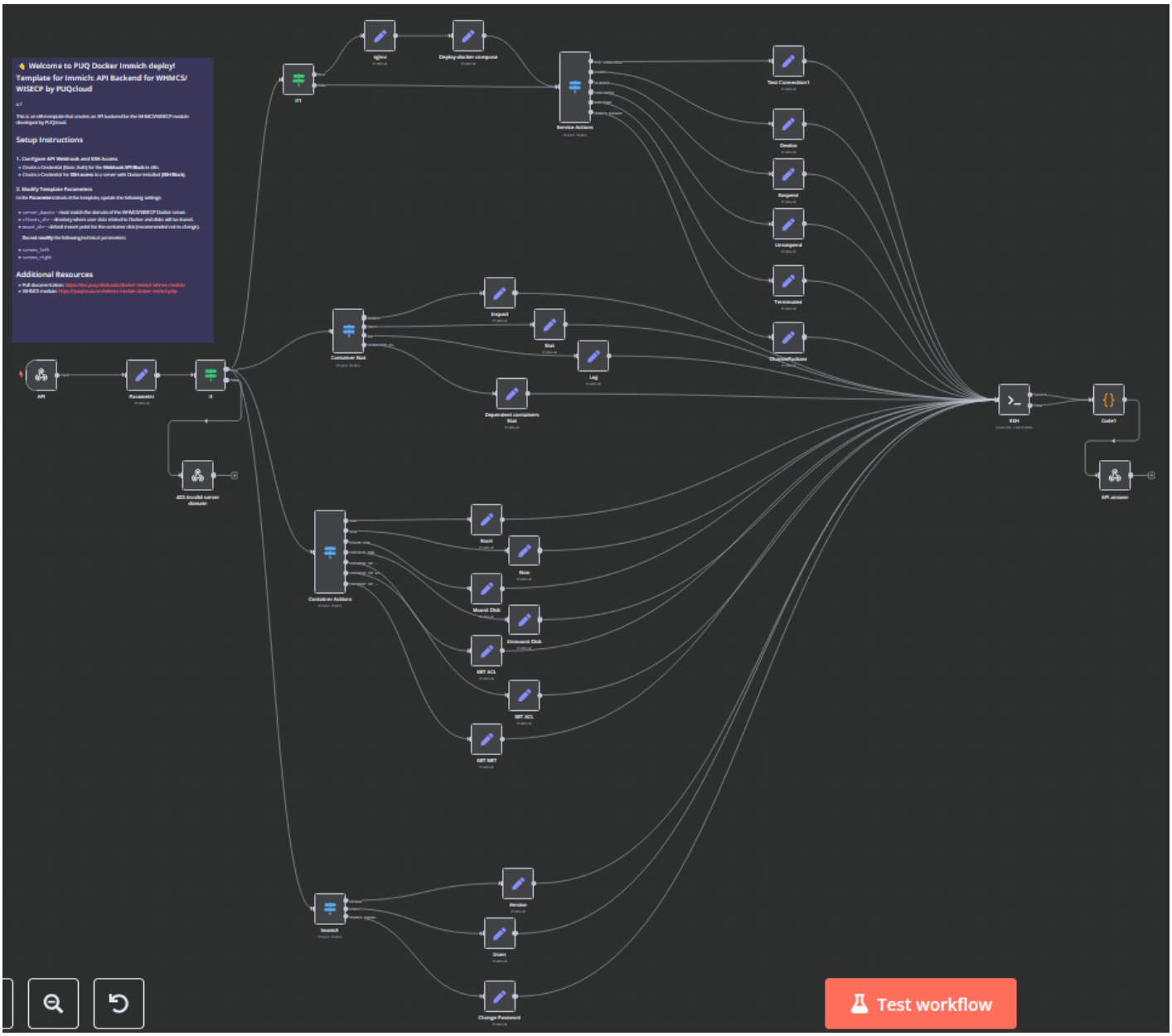
App

Refresh

Version	2.2.53
Users	it@puq.pl

Metric Statistics

Metric	Enabled	Current Usage	Last Update
Traffic IN (GB)	✓	0.49 GB	6 hours ago
Traffic OUT (GB)	✓	0.67 GB	6 hours ago


Welcome to PUQ Docker Immich deploy!
Template for Immich: API Backend for WHMCS/ WISECP by PUQcloud
v1
This is an n8n template that creates an API backend for the WHMCS/WISECP module developed by PUQcloud.
Setup Instructions
1. Configure API Webhook and SSH Access
Create a Credential (Basic Auth) for the Webhook API Block in n8n.
Create a Credential for SSH access to a server with Docker installed (SSH Block).
2. Modify Template Parameters
In the Parameters block of the template, update the following settings:
server_domain – must match the domain of the WHMCS/WISECP Docker server.
stack_dir – directory where user data related to Docker and disks will be stored.
mount_dir – default mount point for the container disk (recommended not to change).
Do not modify the following technical parameters:
screen_left
screen_right
Additional Resources
Full documentation: https://doc.puq.info/en/docker-immich-whmcs-module
WHMCS module: https://puq.hu.com/en/docs-module-docker-immich.php
Test workflow

What is Immich

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

What is Immich? An Overview of the Self-Hosted Photo and Video Storage Platform

Immich is an open-source, self-hosted solution for photo and video storage that emphasizes user privacy and control. Unlike cloud services where your data is stored on third-party servers, Immich allows you to host your media library on your own hardware. This gives you complete control over your data and helps ensure its privacy.

Key Features and Characteristics of Immich

- **Self-Hosted:** Your photos and videos are stored on your own servers, providing maximum control over data privacy and security.
- **Open Source:** As an open-source project, Immich allows anyone to inspect, contribute to, and adapt the software to suit their needs.
- **Privacy-Focused Functionality:** Immich is designed with privacy in mind, and your data is not shared with or used by third parties.
- **Active Development:** Immich is still in active development, so users should be prepared for potential bugs and changes. The developers caution against using Immich as the sole storage method for your photos and videos.

Mobile Apps and Web Interface

Immich offers mobile apps for both Android and iOS, allowing users to easily upload and access their media on the go. There's also a web interface for managing your library from any browser.

Is Immich Right for You?

Immich can be an excellent option for tech-savvy users who want control over their photos and videos. It's particularly appealing for those concerned about privacy and who prefer to keep their data on their own hardware rather than in the cloud.

However, as Immich is in active development, caution should be exercised when using it to store critical data. Nonetheless, for those seeking a self-hosted, privacy-oriented solution for media management, Immich offers a promising open-source platform.

Remember to regularly back up your data and stay updated with the latest developments in the Immich project if you decide to use it.

image-1742323351681.webp

Image not found or type unknown

Changelog

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

v1.1 Released 28-04-2025

1. Support for multi-service docker backend
2. Support custom domain names via PowerDNS
3. Support configuration options, CPU, RAM, DISK

---

v1.0 Released 18-03-2025

First version

Installation and configuration guide

Basic concepts and requirements

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

> Before you start, it is important to read and familiarize yourself with the following articles at this link:
>
> https://doc.puq.info/books/docker-modules

The **WHMCS Docker Immich module** is part of the **WHMCS Docker module series** developed by **PUQcloud**. This module enables service providers to offer **Immich Docker containers** as a service to their clients, allowing for seamless automation and integration.

The system consists of three core components:

□□ WHMCS Module

The **core component**, installed in WHMCS, manages service provisioning and automation from the WHMCS side.

□□ Debian 12 Server

A server running **Debian 12** with the following pre-installed:
□**Docker** – For container management

We have prepared instructions for installing and configuring Docker

https://doc.puq.info/books/docker-modules/page/installing-docker-for-puqcloud-modules

□□ n8n Server

This server facilitates communication between the WHMCS module and the Docker server, ensuring smooth workflow execution.

To explore n8n’s full potential, visit the **official n8n website** for documentation, tutorials, and community support.

Key Features & Concepts

⚙ Workflow Automation

n8n provides a **graphical workflow builder**, allowing users to automate various tasks, such as:
✔ **Sending notifications**
✔ **Configuring firewalls** on external routers
✔ **Managing DNS settings**
✔ **Custom automation processes** tailored to specific needs

□□ Flexibility & Customization

The module offers **personalized settings** and supports **elastic automation**, giving clients full control over their n8n workflows.

WHMCS setup (install/update)

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

> **Module is coded ionCube v13**

Supported php version:

- php 7.4 WHMCS 8.11.0 -
- php 8.1 WHMCS 8.11.0 +
- php 8.2 WHMCS 8.11.0 +

> To install and update a module, you must perform one and the same action.

1. Download the latest version of the module.

PHP 8.2

```
wget http://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-Immich/php82/PUQ_WHMCS-Docker-Immich-latest.zip
```

PHP 8.1

```
wget http://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-Immich/php81/PUQ_WHMCS-Docker-Immich-latest.zip
```

PHP 7.4

```
wget http://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-Immich/php74/PUQ_WHMCS-
```

```
Docker-Immich-latest.zip
```

> All versions are available via link:
> https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-Immich/

2. Unzip the archive with the module.

```
unzip PUQ_WHMCS-Docker-Immich-latest.zip
```

3. Copy and Replace "puqDockerImmich" from "PUQ_WHMCS-Docker-Immich" to "WHMCS_WEB_DIR/modules/servers/"

Preparing Docker Server

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

To install the Docker server for this module, please follow the instructions at the following link:

https://doc.puq.info/books/docker-modules/page/installing-docker-for-puqcloud-modules

Installation and configuration guide

Setting up n8n workflow

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Overview

The **Docker Immich WHMCS module** uses a specially designed workflow for **n8n** to automate deployment processes. The workflow provides an API interface for the module, receives specific commands, and connects via SSH to a server with Docker installed to perform predefined actions.

Prerequisites

- You must have your own **n8n** server.
- Alternatively, you can use the official **n8n** cloud installations available at: n8n Official Site

Installation Steps

Install the Required Workflow on n8n

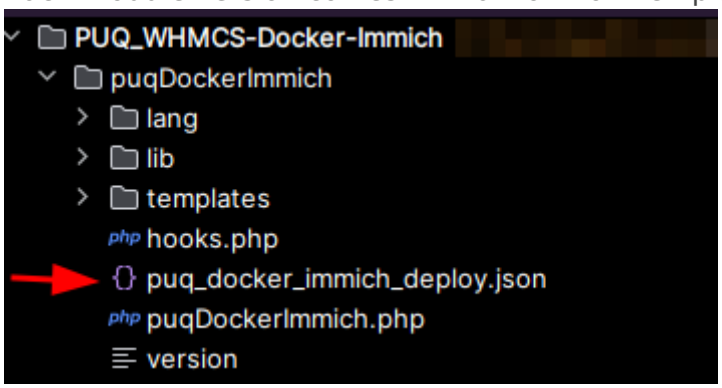
You have two options:

Option 1: Use the Latest Version from the n8n Marketplace

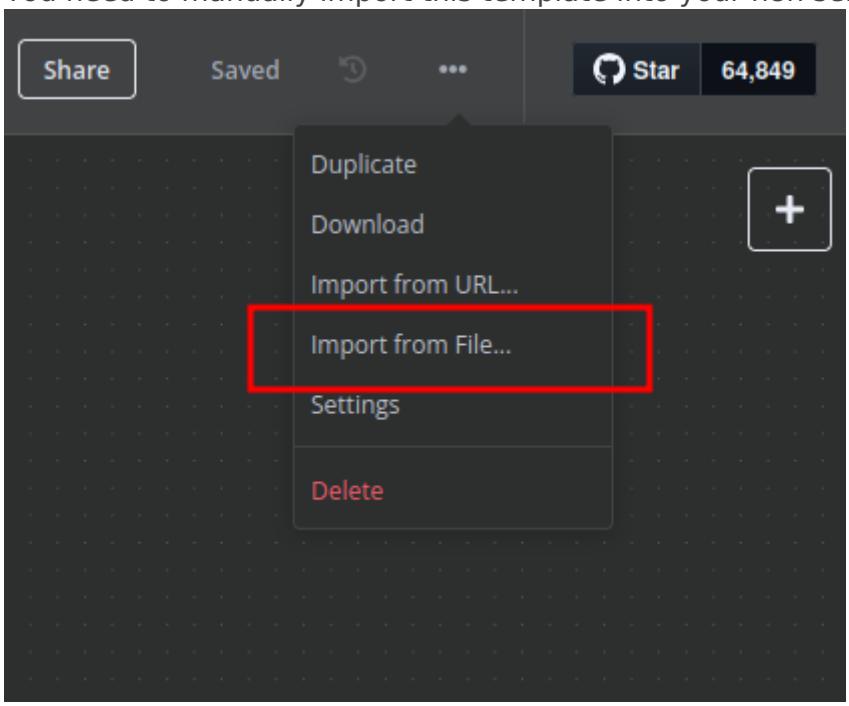
- The latest workflow templates for our modules are available on the official n8n marketplace.
- Visit our profile to access all available templates: PUQcloud on n8n

Option 2: Manual Installation

- Each module version comes with a workflow template file.



- You need to manually import this template into your n8n server.

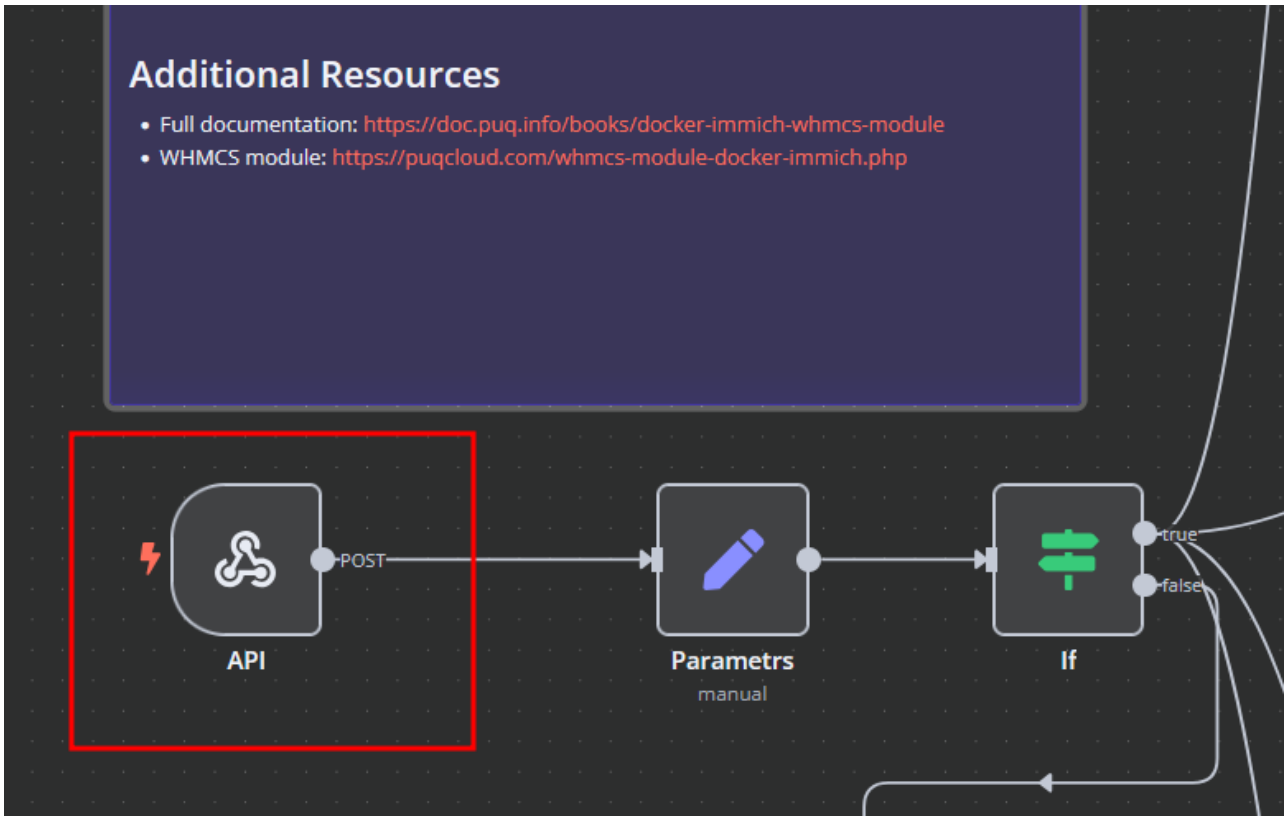


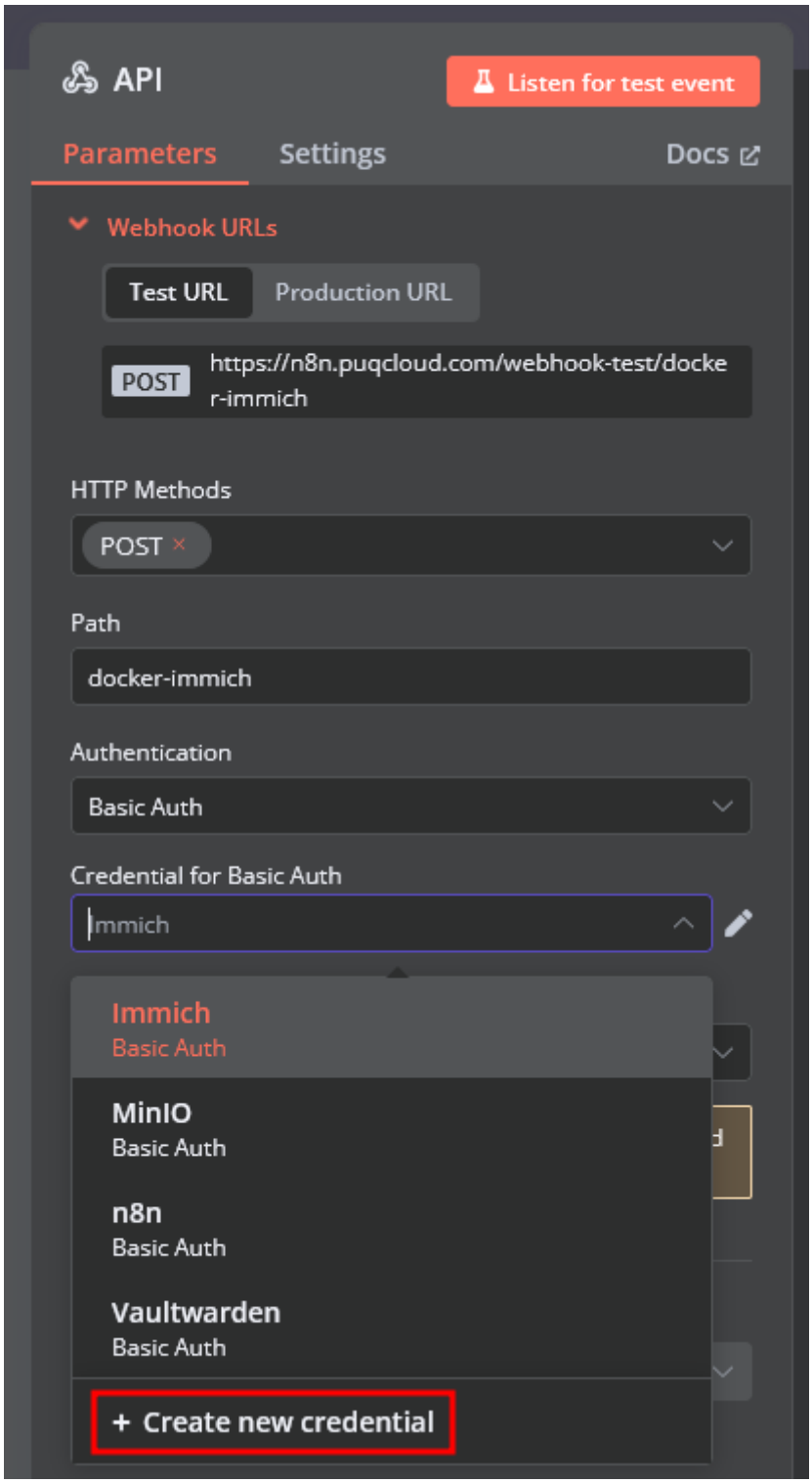
n8n Workflow API Backend Setup for WHMCS/WISECP

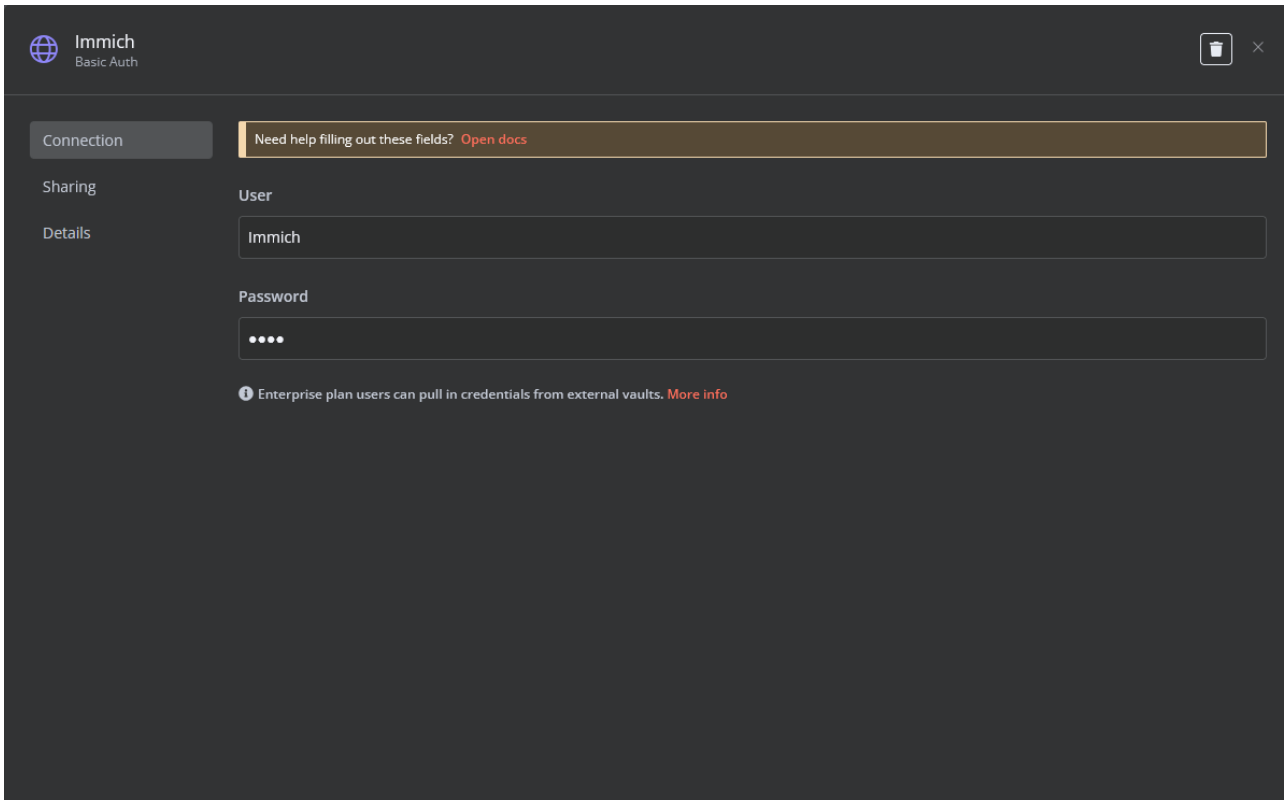
Configure API Webhook and SSH Access

- Create a **Basic Auth Credential**

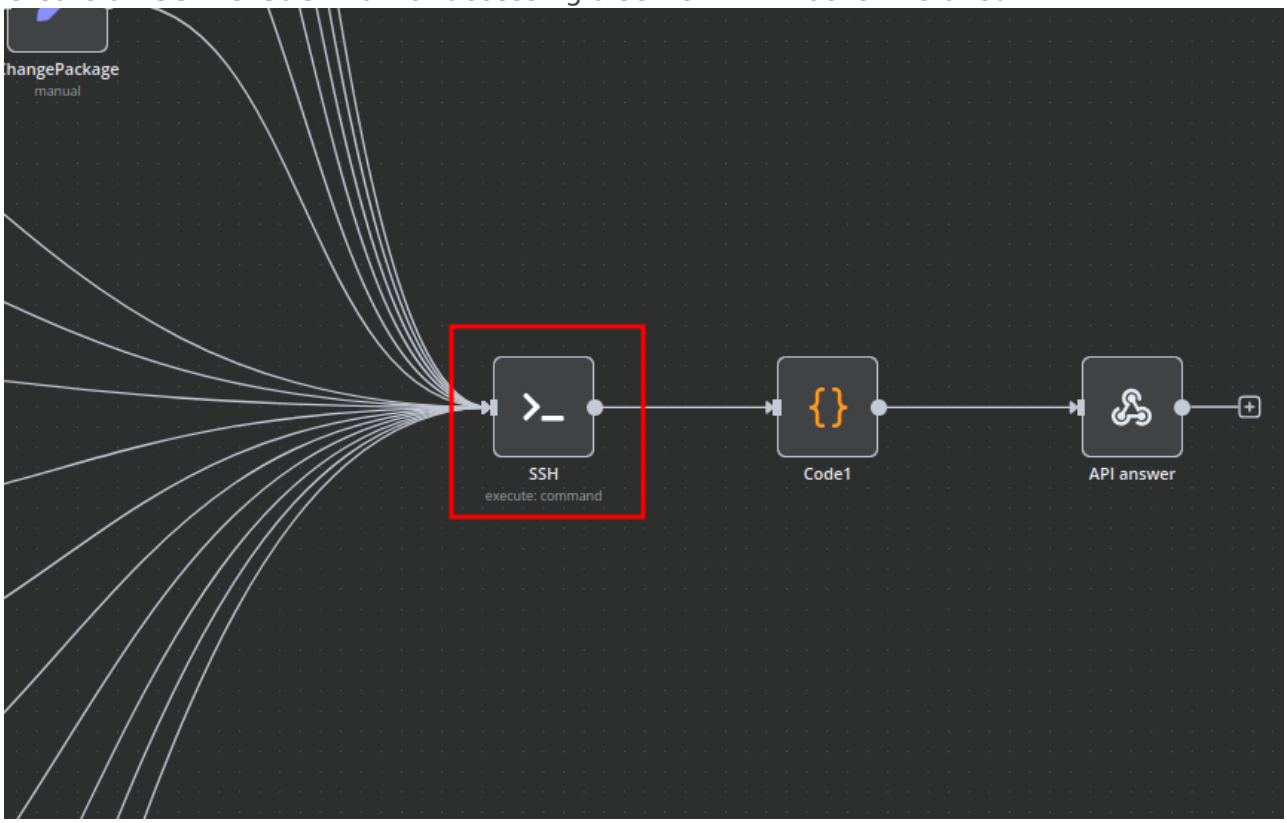
for the Webhook API Block in n8n.

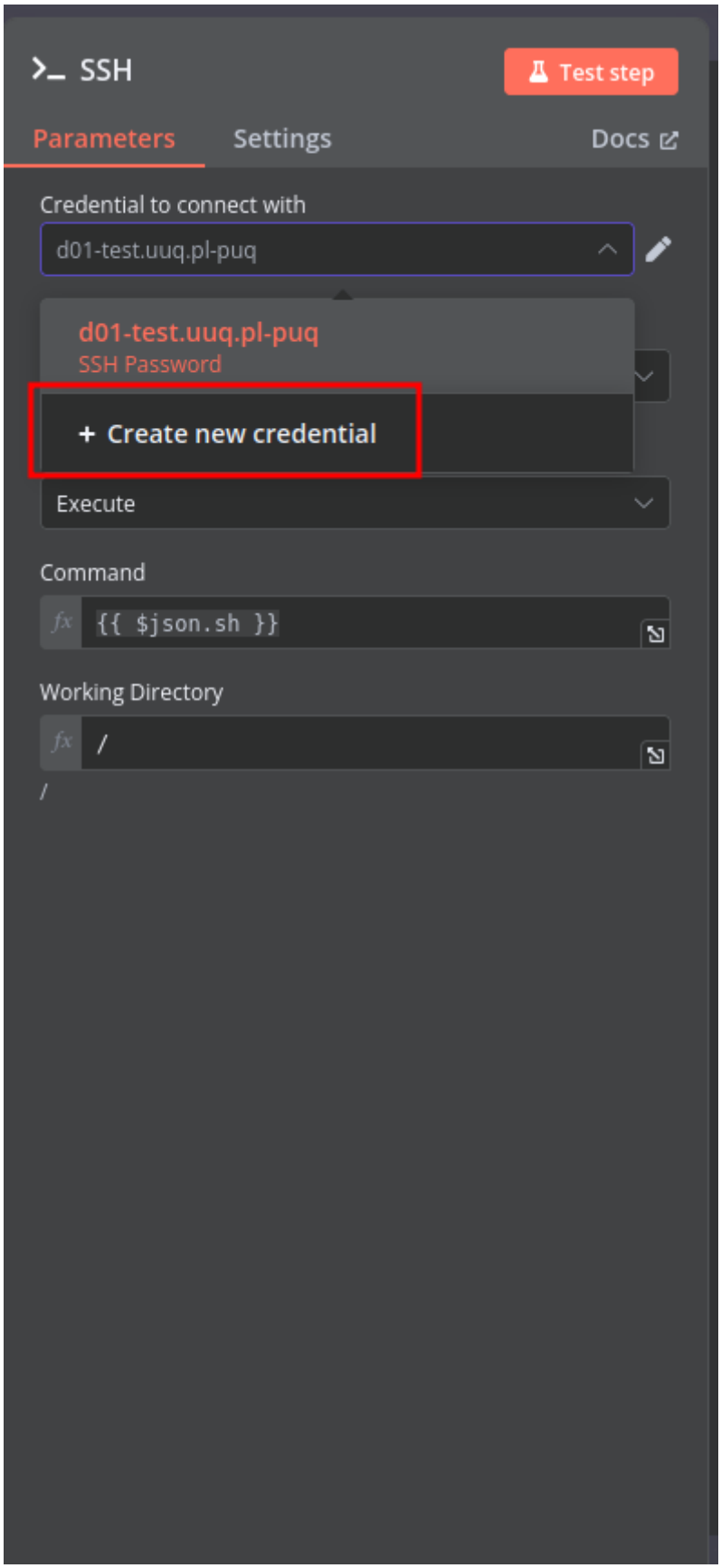


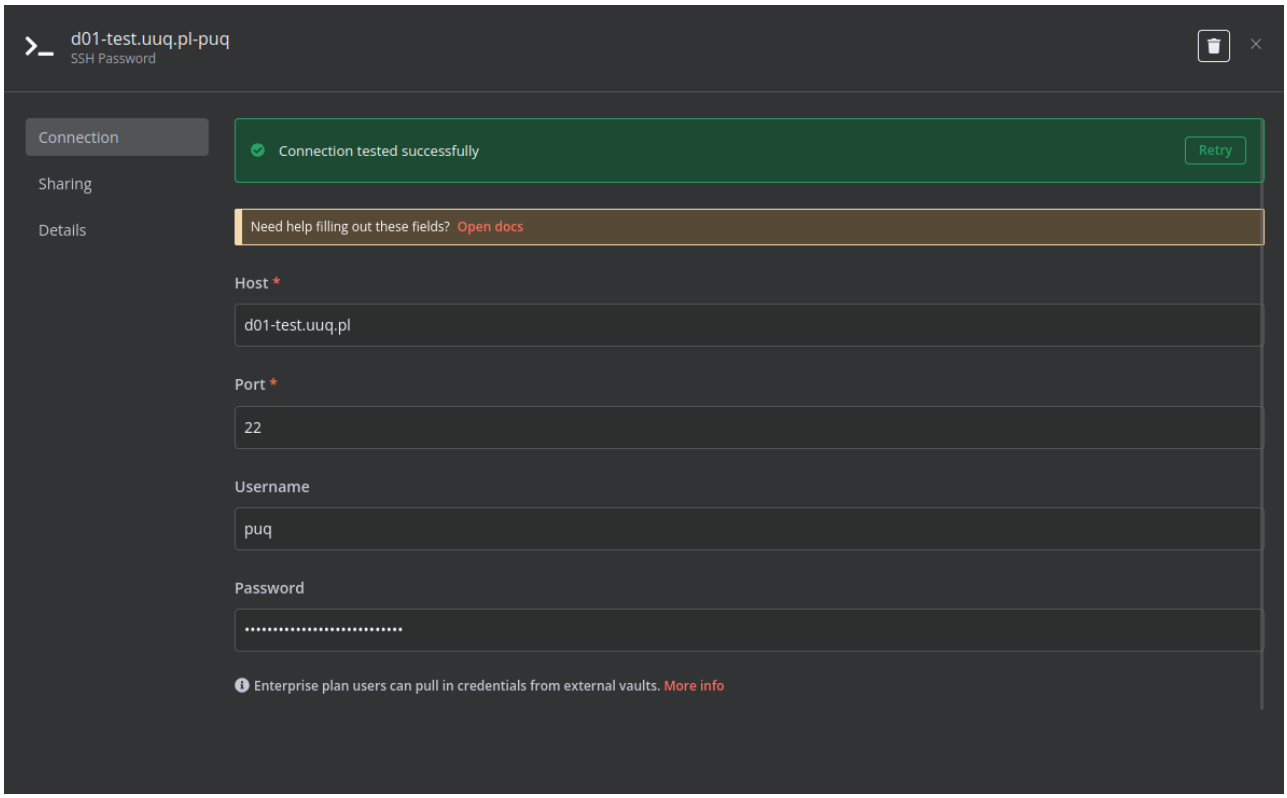

API
Listen for test event
Parameters
Settings
Docs
Webhook URLs
Test URL
Production URL
POST
https://n8n.puqcloud.com/webhook-test/docke
r-immich
HTTP Methods
POST
Path
docker-immich
Authentication
Basic Auth
Credential for Basic Auth
mmich
Immich
Basic Auth
MinIO
Basic Auth
n8n
Basic Auth
Vaultwarden
Basic Auth
+ Create new credential



- Create an **SSH Credential** for accessing a server with Docker installed.

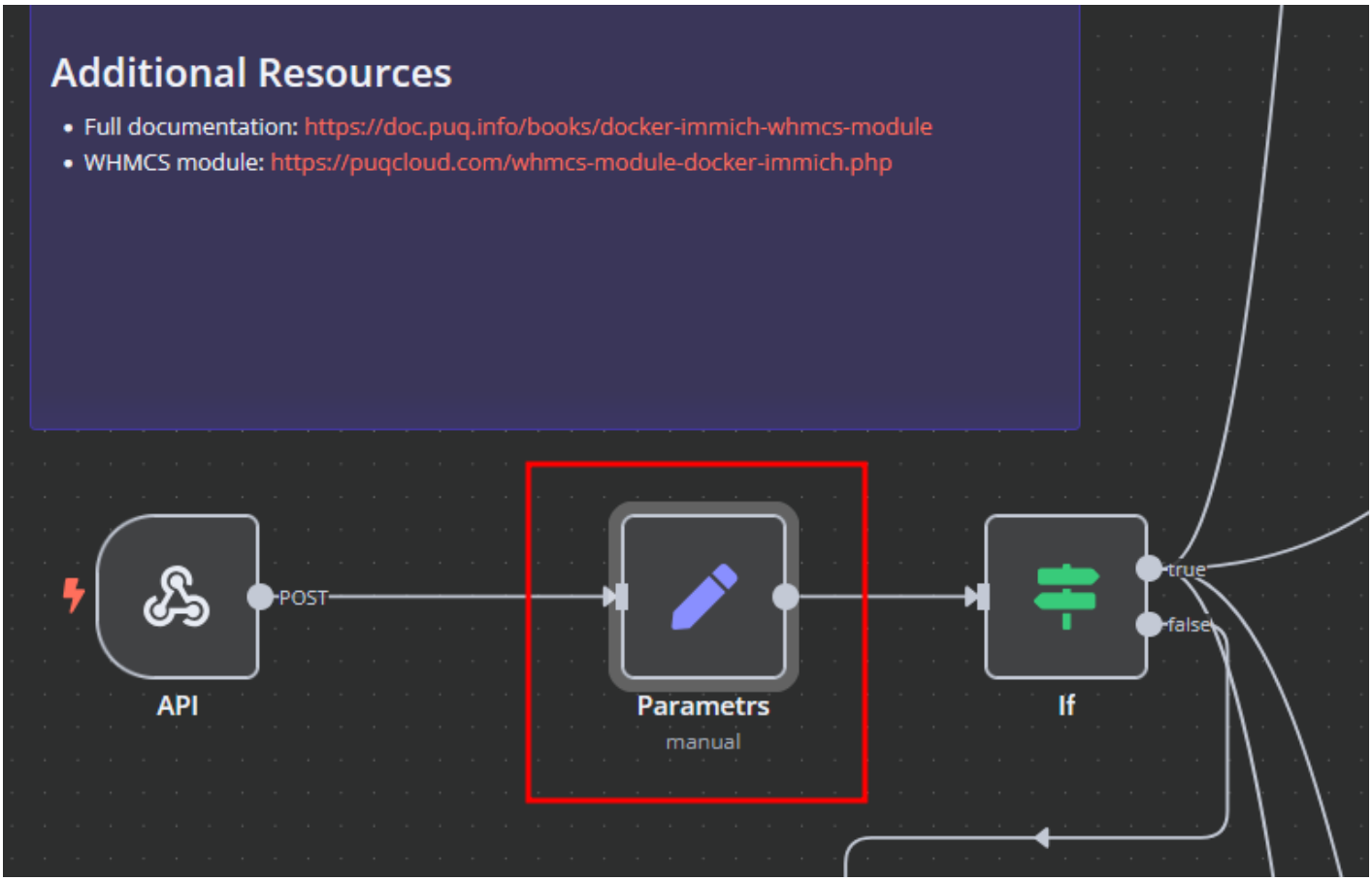


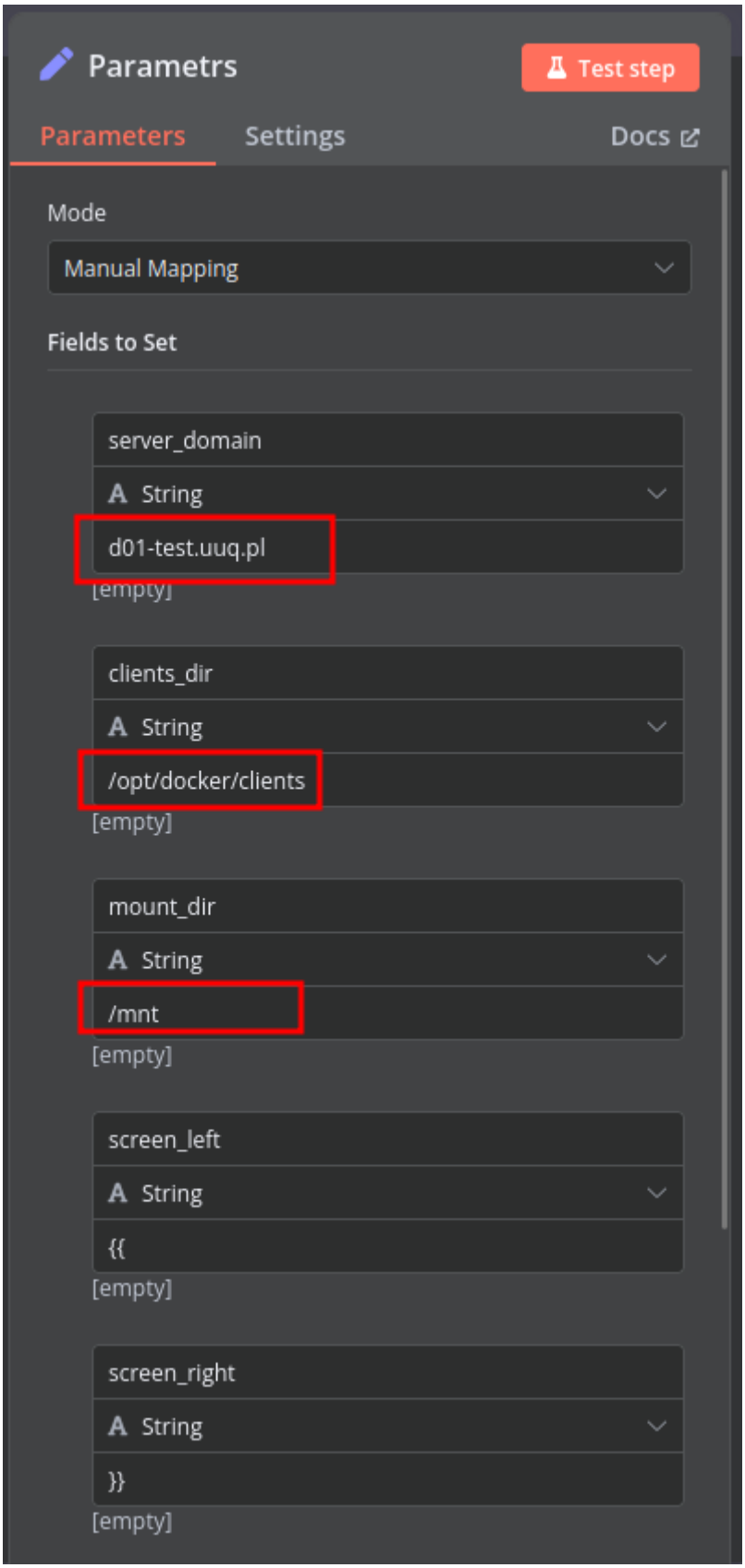

SSH
Test step
Parameters
Settings
Docs
Credential to connect with
d01-test.uuq.pl-puq
d01-test.uuq.pl-puq
SSH Password
+ Create new credential
Execute
Command
{{ $json.sh }}
Working Directory
/
/



Modify Template Parameters

In the **Parameters** block of the template, update the following settings:


Additional Resources
• Full documentation: https://doc.puq.info/books/docker-immich-whmcs-module
• WHMCS module: https://puqcloud.com/whmcs-module-docker-immich.php
POST
API
Parametrs
manual
If
true
false



- `server_domain` – Must match the domain of the WHMCS/WISECP Docker server.
- `clients_dir` – Directory where user data related to Docker and disks will be stored.
- `mount_dir` – Default mount point for the container disk (recommended not to change).

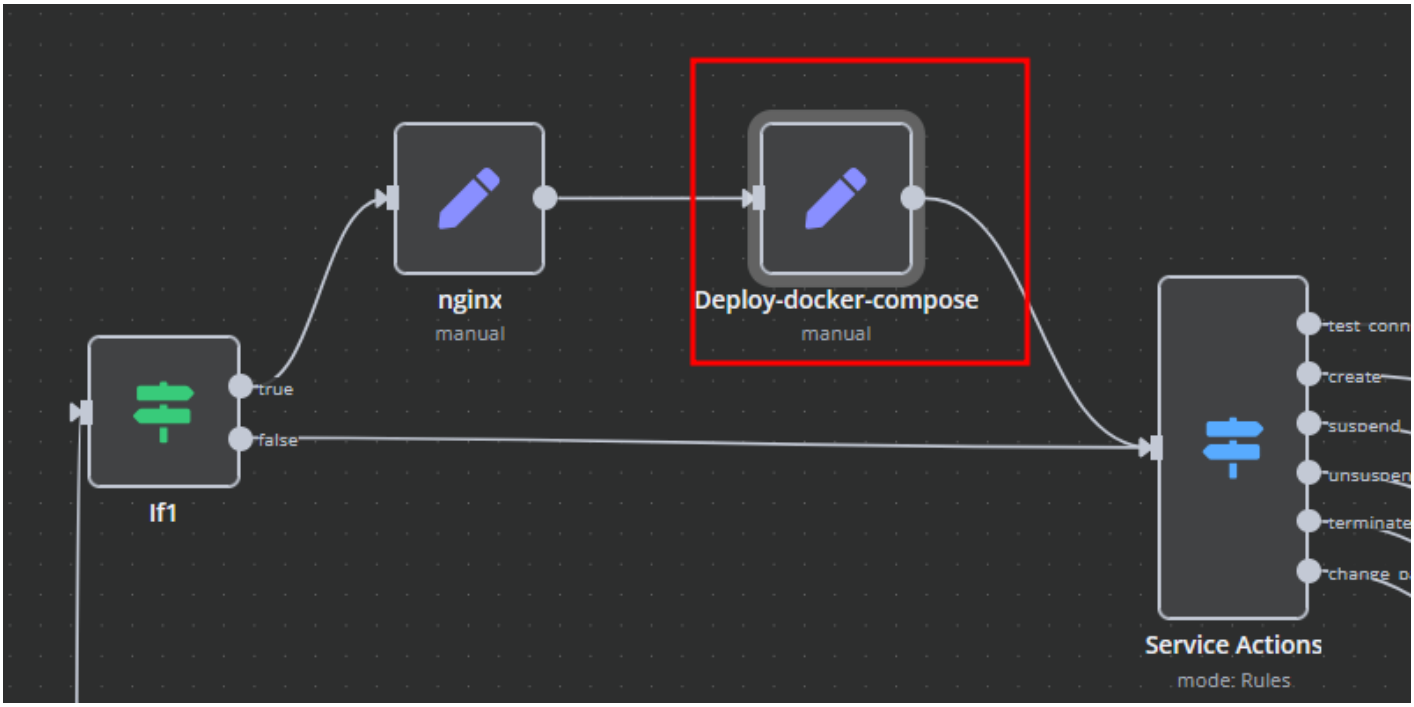
Do not modify the following technical parameters:

- `screen_left`
- `screen_right`

Deploy-docker-compose

In the **Deploy-docker-compose** element, you have the ability to modify the Docker Compose configuration, which will be generated in the following scenarios:

- When the service is created
- When the service is unlocked
- When the service is updated



Expression

Anything inside {{ }} is JavaScript. Learn more

```
name: "{{ $('API').item.json.body.domain }}"

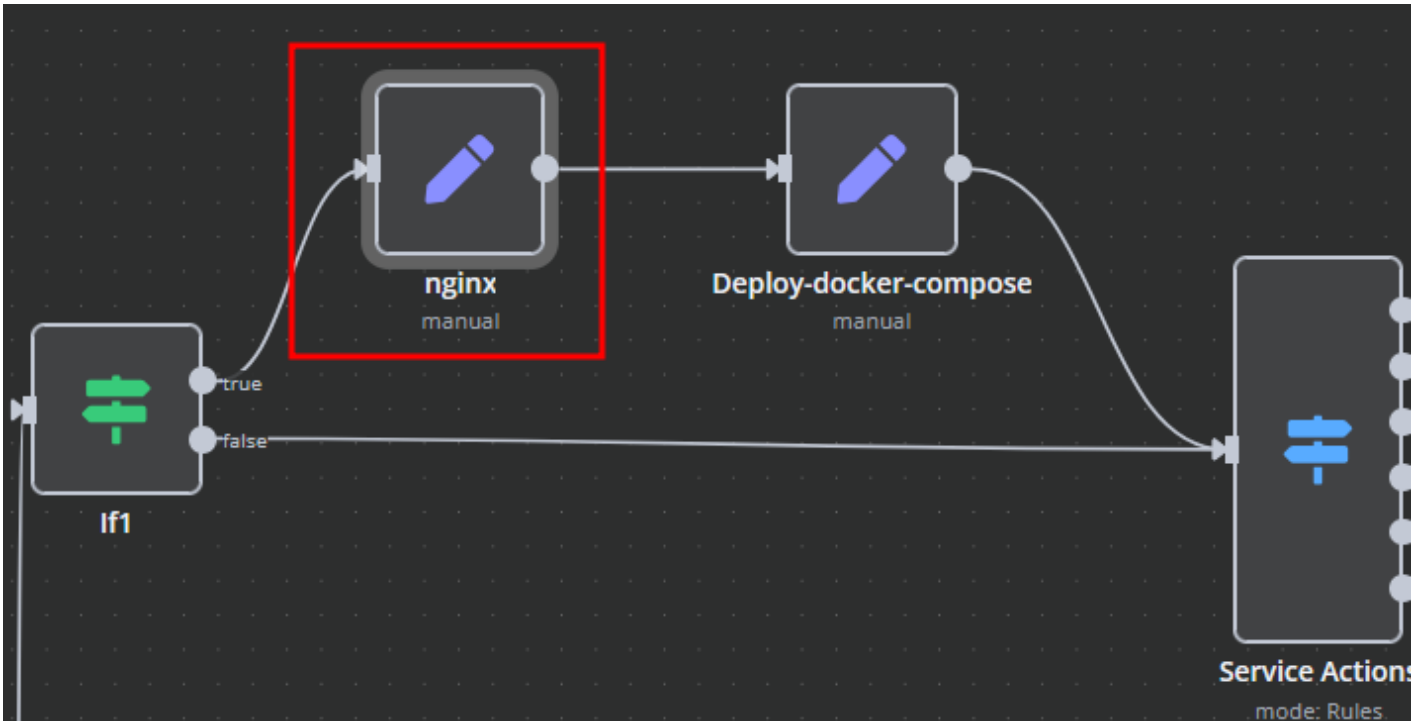
services:
  {{ $('API').item.json.body.domain }}_immich:
    container_name: {{ $('API').item.json.body.domain }}_immich
    image: ghcr.io/immich-app/immich-server:release
    restart: unless-stopped
    volumes:
      - {{ $('Parametrs').item.json.mount_dir }}/{{ $('API').item.json.body.domain }}/library:/
usr/src/app/upload
      - /etc/localtime:/etc/localtime:ro
    environment:
      - LETSENCRYPT_HOST={{ $('API').item.json.body.domain }}
      - VIRTUAL_HOST={{ $('API').item.json.body.domain }}
      - DB_HOSTNAME={{ $('API').item.json.body.domain }}_db
      - DB_PASSWORD={{ $('API').item.json.body.password }}
      - DB_USERNAME={{ $('API').item.json.body.username }}
      - DB_DATABASE_NAME=immich
      - REDIS_HOSTNAME={{ $('API').item.json.body.domain }}_redis
      - IMMICH_MACHINE_LEARNING_URL=http://{{ $('API').item.json.body.domain }}_ml:3003
    depends_on:
      - {{ $('API').item.json.body.domain }}_redis
      - {{ $('API').item.json.body.domain }}_db
    healthcheck:
      disable: false
    networks:
      - nginx-proxy_web
    mem_limit: "{{ $('API').item.json.body.ram }}G"
    cpus: "{{ $('API').item.json.body.cpu }}"

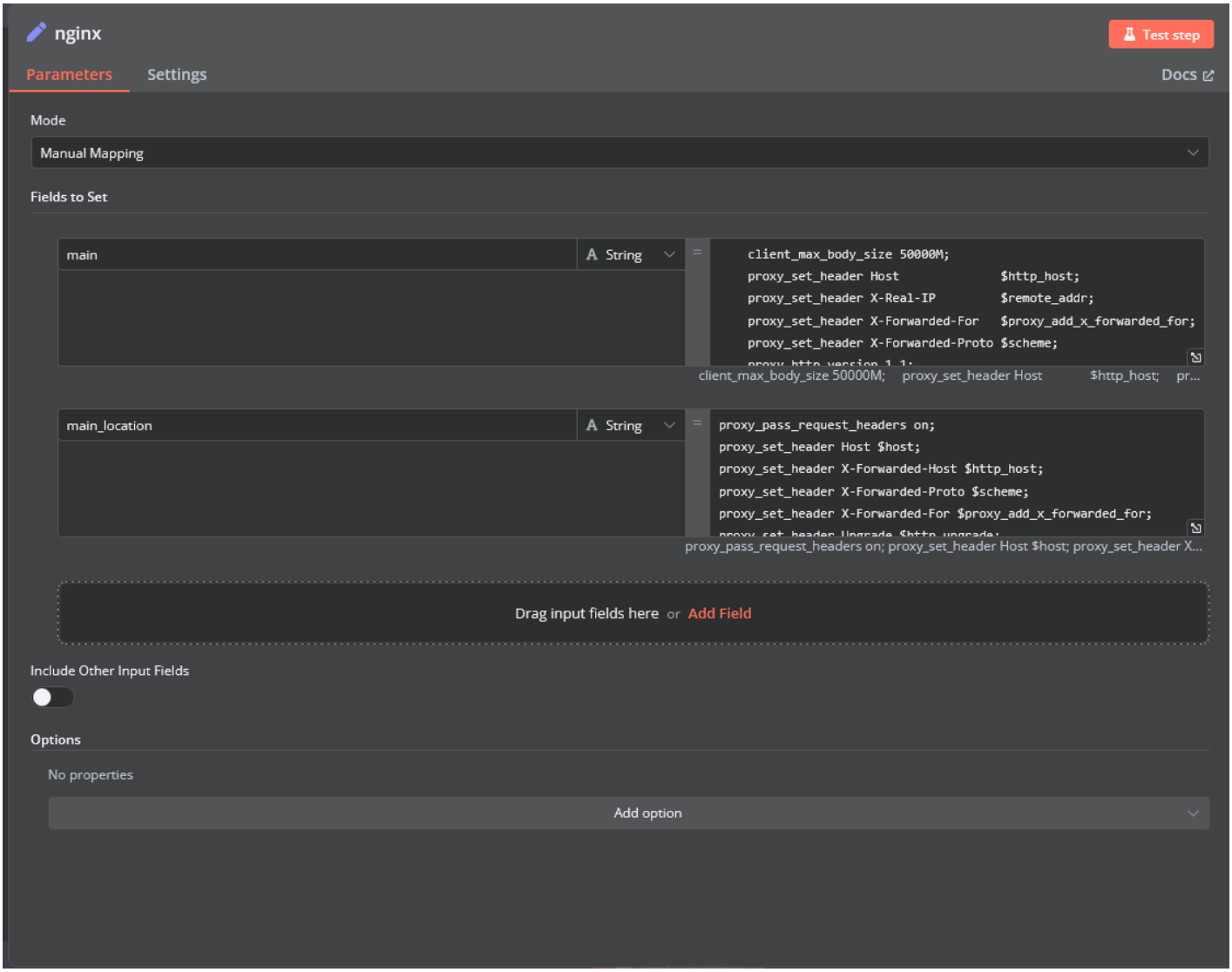
  {{ $('API').item.json.body.domain }}_ml:
    container_name: {{ $('API').item.json.body.domain }}_ml
    image: ghcr.io/immich-app/immich-machine-learning:release
    volumes:
      - {{ $('Parametrs').item.json.mount_dir }}/{{ $('API').item.json.body.domain }}/cache:/
cache
    restart: always
    healthcheck:
      disable: false
```

nginx

In the **nginx** element, you can modify the configuration parameters of the web interface proxy server.

- The **main** section allows you to add custom parameters to the **server** block in the proxy server configuration file.
- The **main_location** section contains settings that will be added to the **location /** block of the proxy server configuration. Here, you can define custom headers and other parameters specific to the root location.

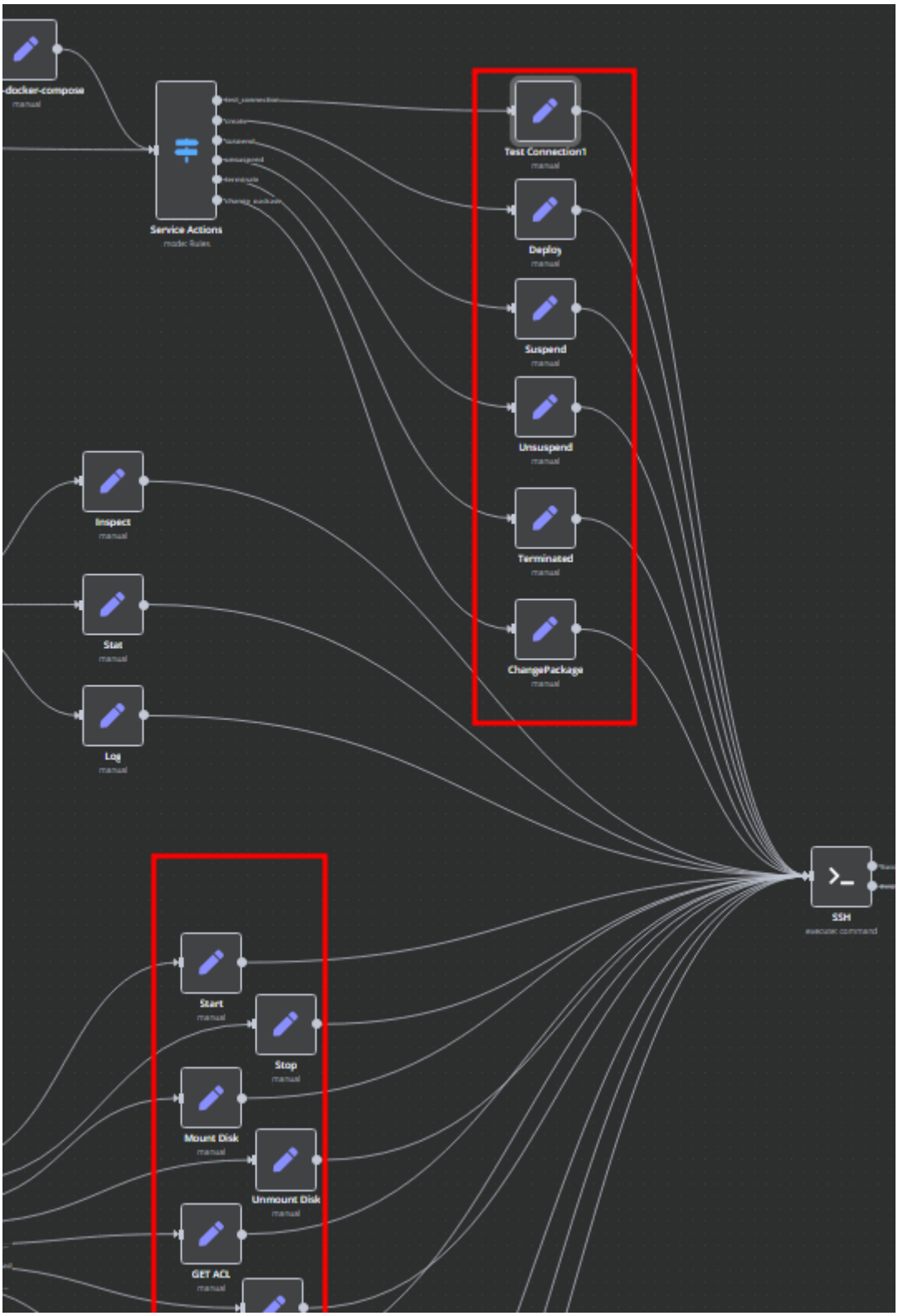




Bash Scripts

Management of Docker containers and all related procedures on the server is carried out by executing Bash scripts generated in **n8n**. These scripts return either a JSON response or a string.

- All scripts are located in elements directly connected to the **SSH** element.
- You have full control over any script and can modify or execute it as needed.


-docker-compose
manual
Service Actions
mode: Rules
Test Connection1
manual
Deploy
manual
Suspend
manual
Unsuspend
manual
Terminated
manual
ChangePackage
manual
Inspect
manual
Stat
manual
Log
manual
Start
manual
Stop
manual
Mount Disk
manual
Unmount Disk
manual
GET ACL
manual
SSH
execute command

Add server

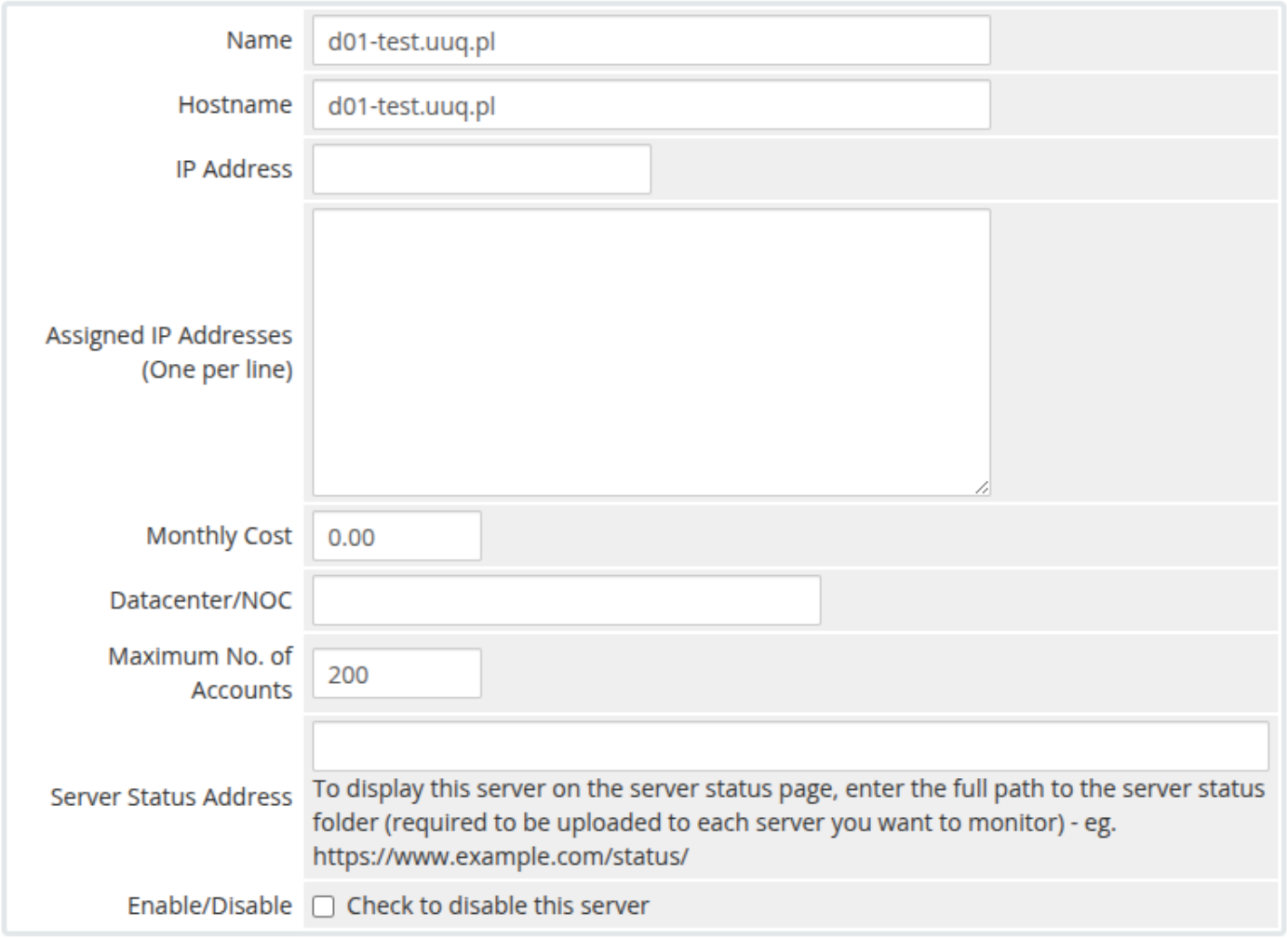
Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Add a new server to the system WHMCS.

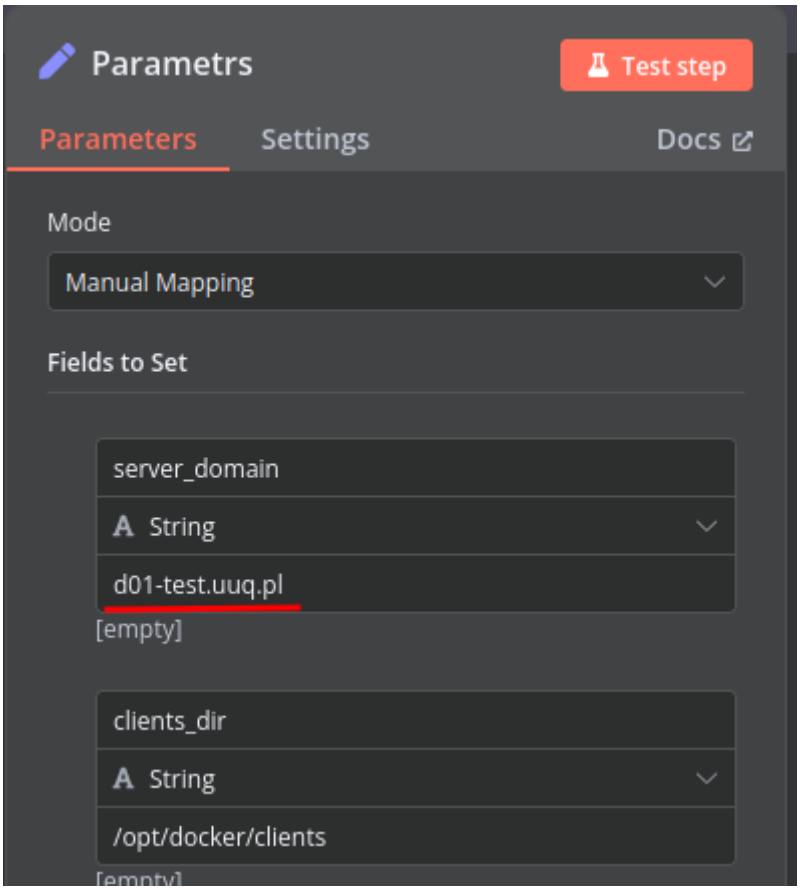
```
System Settings->Servers->Add New Server
```

- Enter the correct **Name** and **Hostname**



Attention: Important Information

The **hostname** field represents the actual domain of the server running Docker and must match the **server_domain** parameter in the **n8n workflow**. If they do not match, communication will not function correctly.
Additionally, this domain must be configured so that all its subdomains resolve to the IP address of the server running Docker.

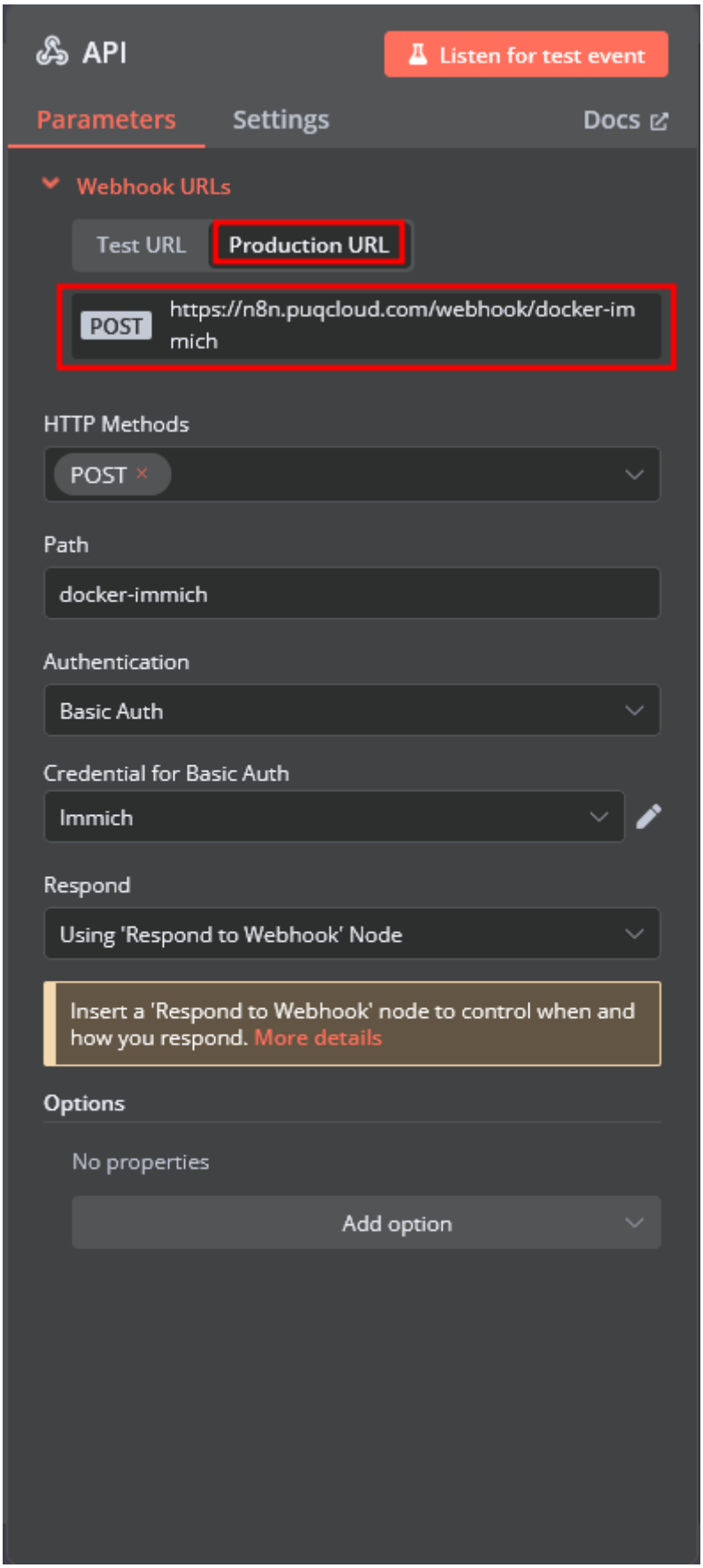


In the **Server Details** section, select the **"PUQ Docker Immich"** module and enter the correct **username** and **password** for the **API endpoint** in the n8n workflow.

Additionally, in the **Access Hash** field, insert the **URL of the API entry point** for the n8n workflow.

Server Details

Module	PUQ Docker Immich Test Connection ✔ Connection successful. Some values have been auto-filled.
Username	Immich
Password	••••
Access Hash	https://n8n.puqcloud.com/webhook/docker-immich
Secure	☑ Check to use SSL Mode for Connections


API
Listen for test event
Parameters
Settings
Docs
Webhook URLs
Test URL
Production URL
POST
https://n8n.puqcloud.com/webhook/docker-immich
HTTP Methods
POST
Path
docker-immich
Authentication
Basic Auth
Credential for Basic Auth
Immich
Respond
Using 'Respond to Webhook' Node
Insert a 'Respond to Webhook' node to control when and how you respond. More details
Options
No properties
Add option

Product Configuration

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Add new product to WHMCS

```
System Settings->Products/Services->Create a New Product
```

In the **Module settings** section, select the **"PUQ Docker Immich"** module

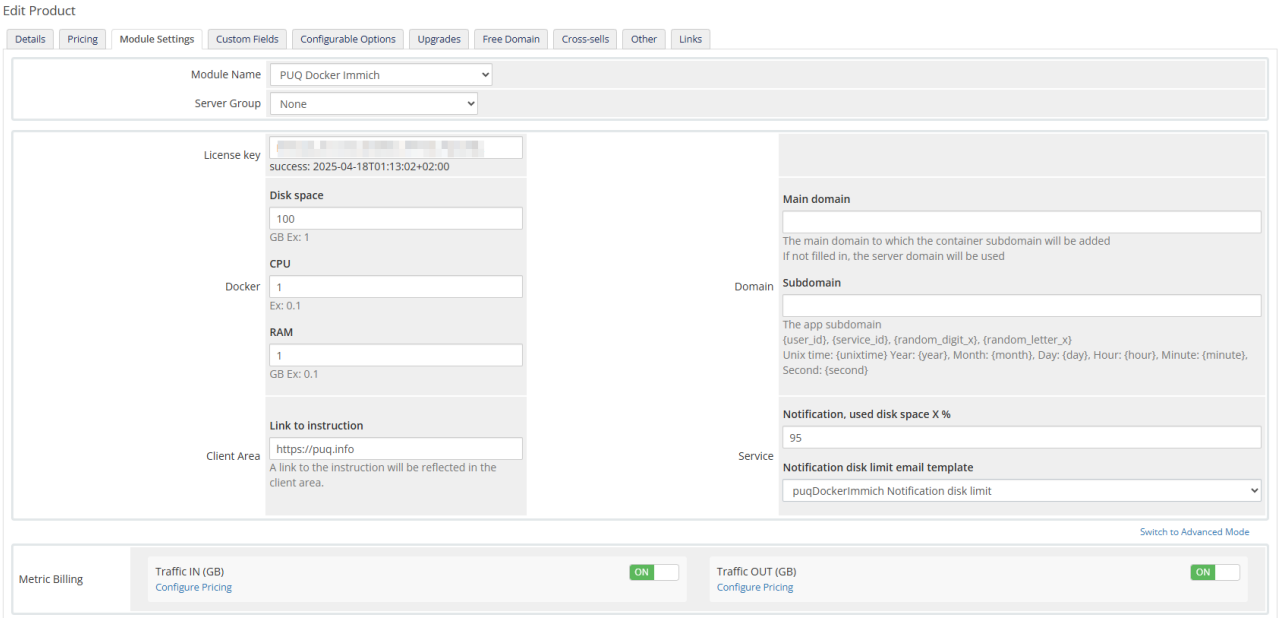
- **License key –** A pre-purchased license key for the **"PUQ Docker Immich"** module. For the module to work correctly, the key must be active
- **Disk space** – defines the allocated disk size for the Docker container.
- **CPU** – sets the CPU usage limit for the Docker container.
- **RAM** – specifies the amount of RAM allocated to the Docker container.
- **Link to instruction** – URL to a guide that will be displayed in the client panel if provided.
- **Main domain** – defines the primary domain for the web interface of the application. If not set, the main domain will be taken from the **hostname** parameter in the server settings.
- **Subdomain** – a personal subdomain assigned to each service. If left empty or if the subdomain is already taken, it will be automatically generated in the format **{user_id}-{service_id}**.
-

Supported Macros for **App Subdomain**:

- **{user_id}** – Client ID
- **{service_id}** – Service ID
- **{random_digit_x}** – Random number (x defines the length)
- **{random_letter_x}** – Random letter (x defines the length)
- **{unixtime}** – Unix timestamp
- **{year}, {month}, {day}, {hour}, {minute}, {second}** – Date and time

values

- **Notification, used disk space X %** – The percentage value that sets the threshold for the container's disk space usage will trigger a notification message to the client once the threshold is reached.
- **Notification disk limit email template** – The email template for the notification that will be sent when the threshold is reached.



Installation and configuration guide

Metric Billing

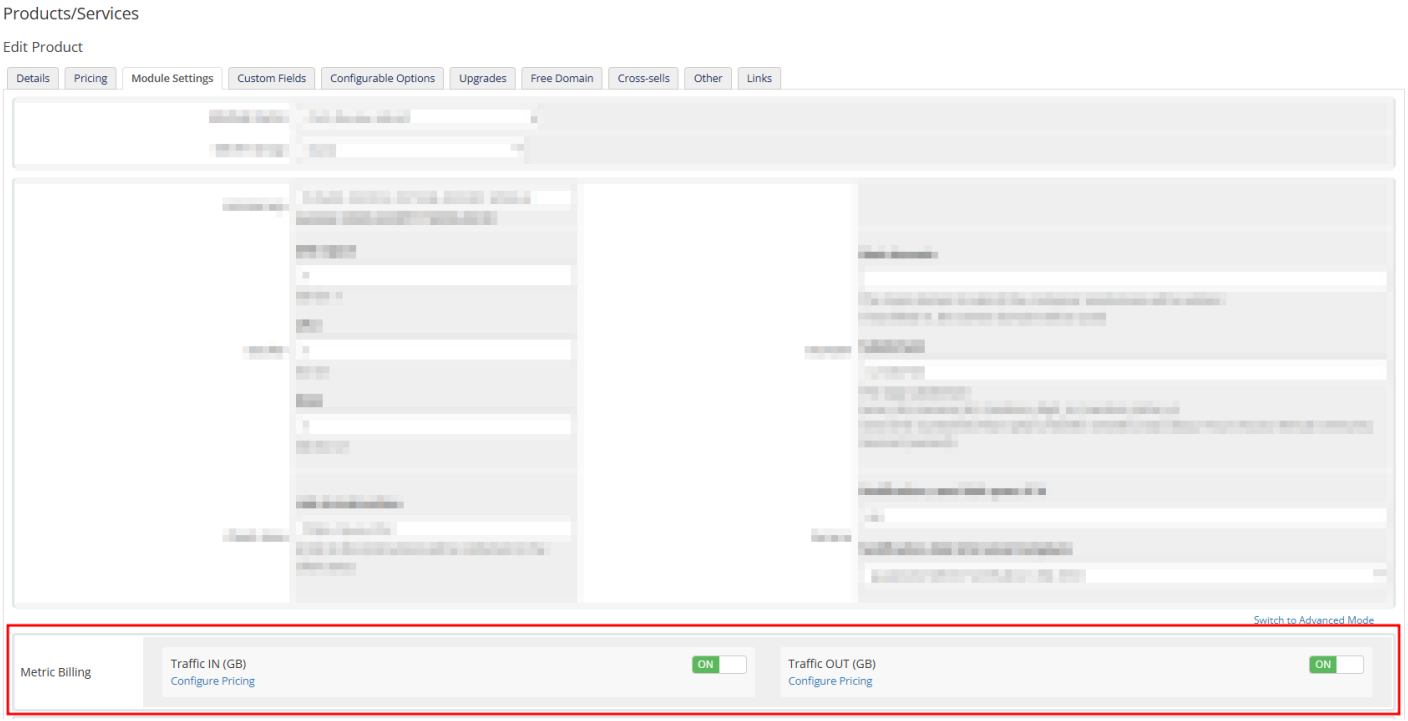
Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

To bill certain metrics separately, you can use the standard WHMCS Metric Billing mechanism.

To configure it, you need to enable the required metrics and set the desired prices. Metrics work based on the standard WHMCS mechanism, the description of which can be found here:

https://docs.whmcs.com/products/configuration-options/usage-billing/



Configure Pricing

Traffic IN (GB)

Metric Type: Monthly

Metric Unit: GigaBytes

Pricing

Scheme:

(•) Per Unit ⓘ ( ) Total Volume ⓘ ( ) Graduated ⓘ

Quantity Included

0.00

Price Per GB			
PLN	EUR	UAH	USD
1.00	1.00	1.00	1.00

Close Save

Email Template (puqDockerImmich Welcome Email)

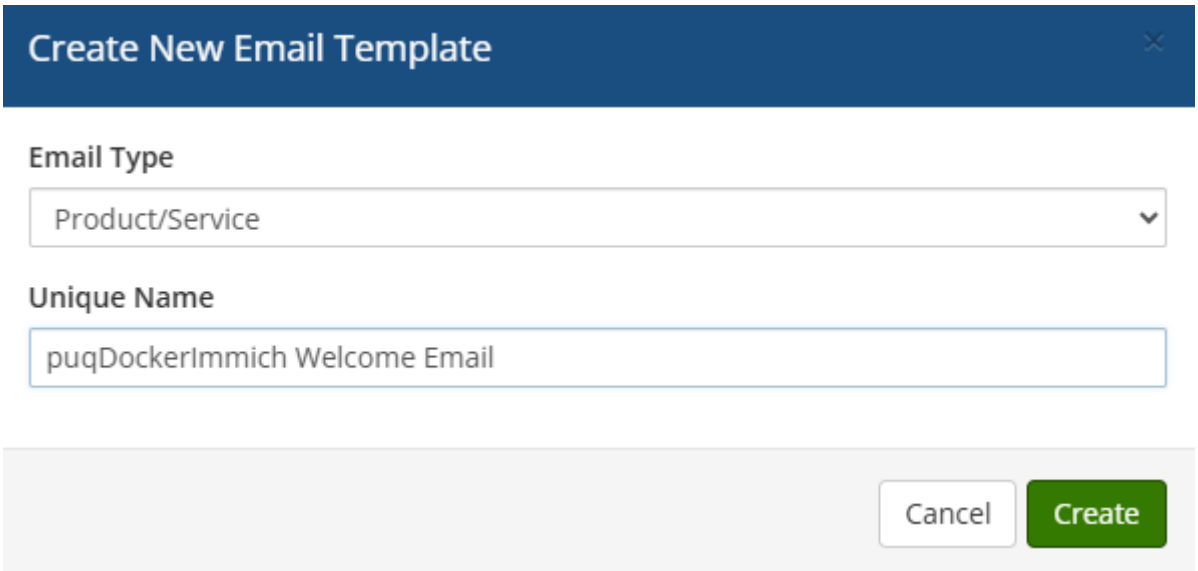
Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Create an email template for customer notifications.

```
System Settings->Email Templates->Create New Email Template
```

- **Email Type:** Product/service
- **Unique Name:** puqDockerImmich Welcome Email



Subject:

```
Immich Order Information
```

Body:

```
Dear {$client_name},

Your order has been accepted for implementation.

Product/Service: {$service_product_name}
Payment Method: {$service_payment_method}
Amount: {$service_recurring_amount}
Billing Cycle: {$service_billing_cycle}
Next Due Date: {$service_next_due_date}


The installation and setup of your Immich instance is in progress.
Within the next 4-5 minutes, you will be able to use your Immich instance.

Upon your first login, you will need to create an account.

Here is the link to your Immich server.

https://{$service_domain}/

Thank you for choosing us.

{$signature}
```

Subject: Immich Order Information

File Edit View Insert Format Table Help

Paragraph Verdana 11pt

Dear **{$client_name}**,

Your order has been accepted for implementation.

Product/Service: **{$service_product_name}**
Payment Method: **{$service_payment_method}**
Amount: **{$service_recurring_amount}**
Billing Cycle: **{$service_billing_cycle}**
Next Due Date: **{$service_next_due_date}**

The installation and setup of your Immich instance is in progress.
Within the next 4-5 minutes, you will be able to use your Immich instance.

Upon your first login, you will need to create an account.

Here is the link to your Immich server.

https://{$service_domain}/

Thank you for choosing us.

{$signature}

P » STRONG 82 WORDS POWERED BY TINYMCE

Email Template (puqDockerImmich Update Email)

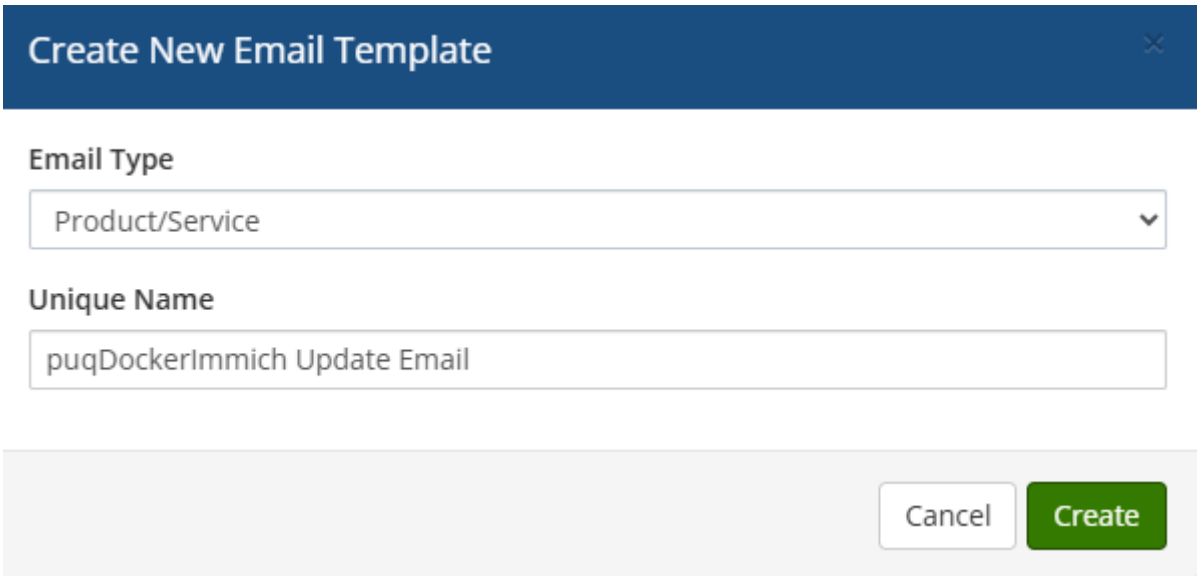
Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Create an email template for customer notifications.

```
System Settings->Email Templates->Create New Email Template
```

- **Email Type:** Product/service
- **Unique Name:** puqDockerImmich Update Email



Subject:

```
Immich Update Information
```

Body:

```
Dear {$client_name},

Your instance is currently being updated.
You will be able to use your Immich server again within 3 minutes.

Here is the link to your Immich server.

https://{$service_domain}/
Thank you for choosing us.

{$signature}
```

Subject: Immich Update Information

File Edit View Insert Format Table Help

Paragraph Verdana 11pt

Dear **{$client_name}**,

Your instance is currently being updated.
You will be able to use your Immich server again within 3 minutes.

Here is the link to your Immich server.

https://{$service_domain}/
Thank you for choosing us.

{$signature}

P 38 WORDS POWERED BY TINYMCE

Email Template (puqDockerImmich Notification disk limit)

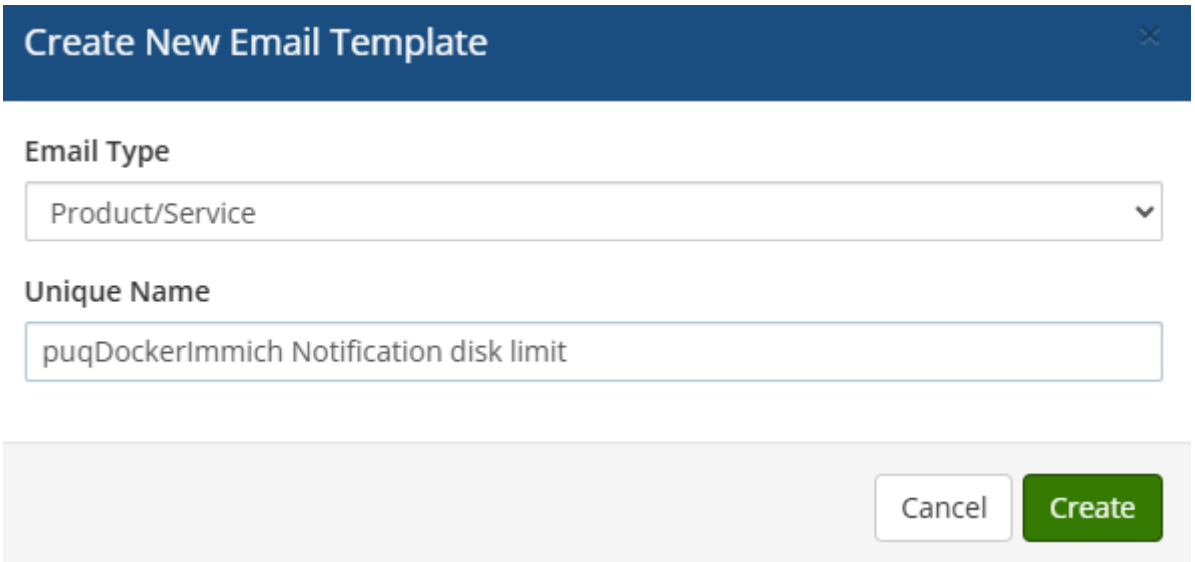
Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Create an email template for customer notifications.

```
System Settings->Email Templates->Create New Email Template
```

- **Email Type:** Product/service
- **Unique Name:** puqDockerImmich Notification disk limit



Subject:

```
Disk space usage {$disk_used_percentage}%
```

Body:

```
Dear {$client_name},

We want to inform you that your Immich service is running low on disk space.
Please take action to prevent service interruptions.

Service Details:

Product/Service: {$service_product_name}
Domain: {$service_domain}
Total Disk Space: {$disk_total}
Used Disk Space: {$disk_used} ({$disk_used_percentage}%)
Consider freeing up space or upgrading your plan if needed.

{$signature}
```

Subject: Disk space usage {$disk_used_percentage}%

File Edit View Insert Format Table Help

Paragraph Verdana 11pt

Dear **{$client_name}**,

We want to inform you that your Immich service is running low on disk space.
Please take action to prevent service interruptions.

Service Details:

Product/Service: **{$service_product_name}**
Domain: **{$service_domain}**
Total Disk Space: **{$disk_total}**
Used Disk Space: **{$disk_used} ({$disk_used_percentage}%)**
Consider freeing up space or upgrading your plan if needed.

{$signature}

P 58 WORDS POWERED BY TINYMCE

Configurable Option (CPU/RAM/DISK)

Docker Immich module **WHMCS**

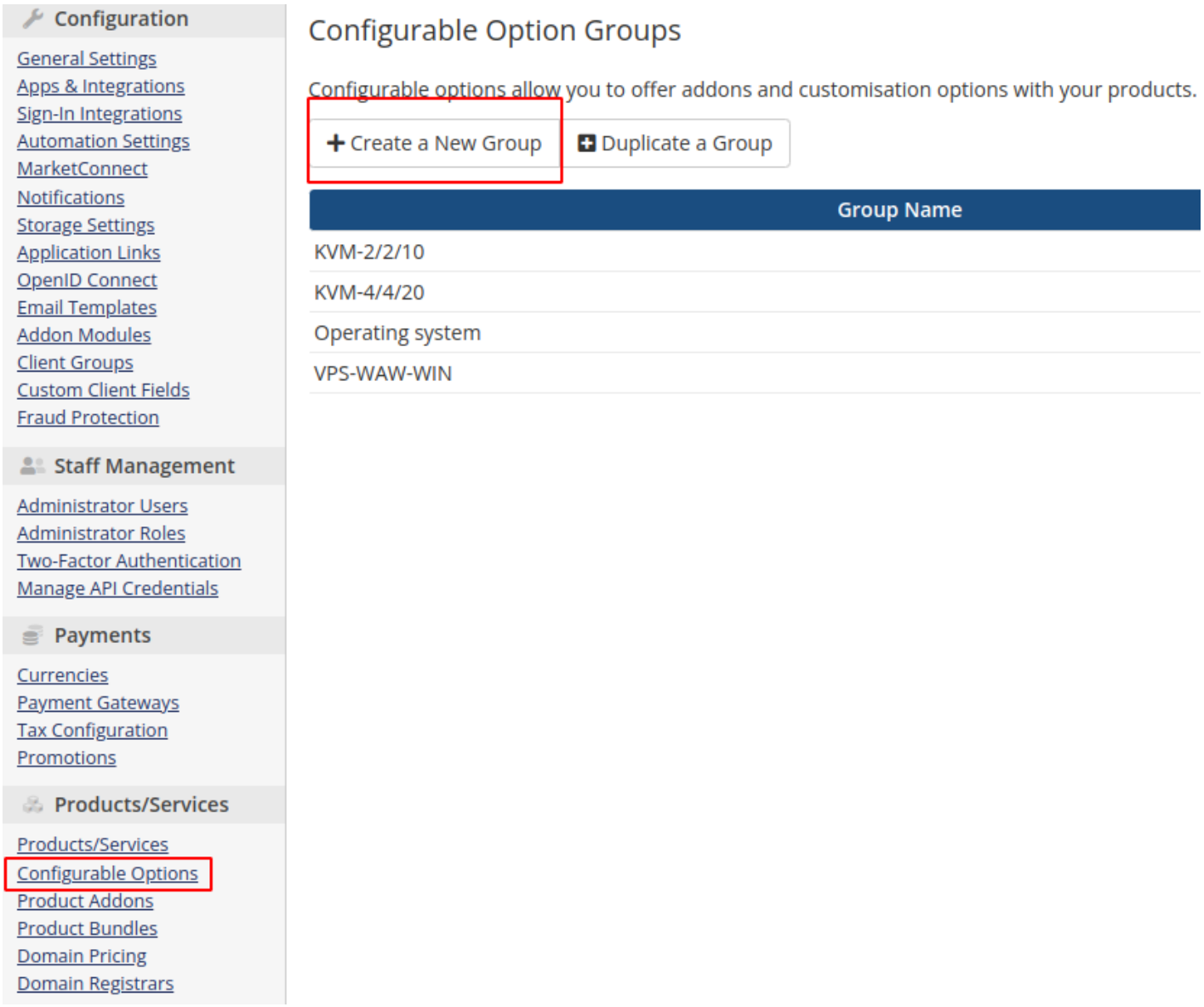
Order now | Download | FAQ | n8n

In order for the client to have a choice of operating system in the configuration of the virtual machine. You need to configure the Configurable Option and connect them to the package.

Create Group

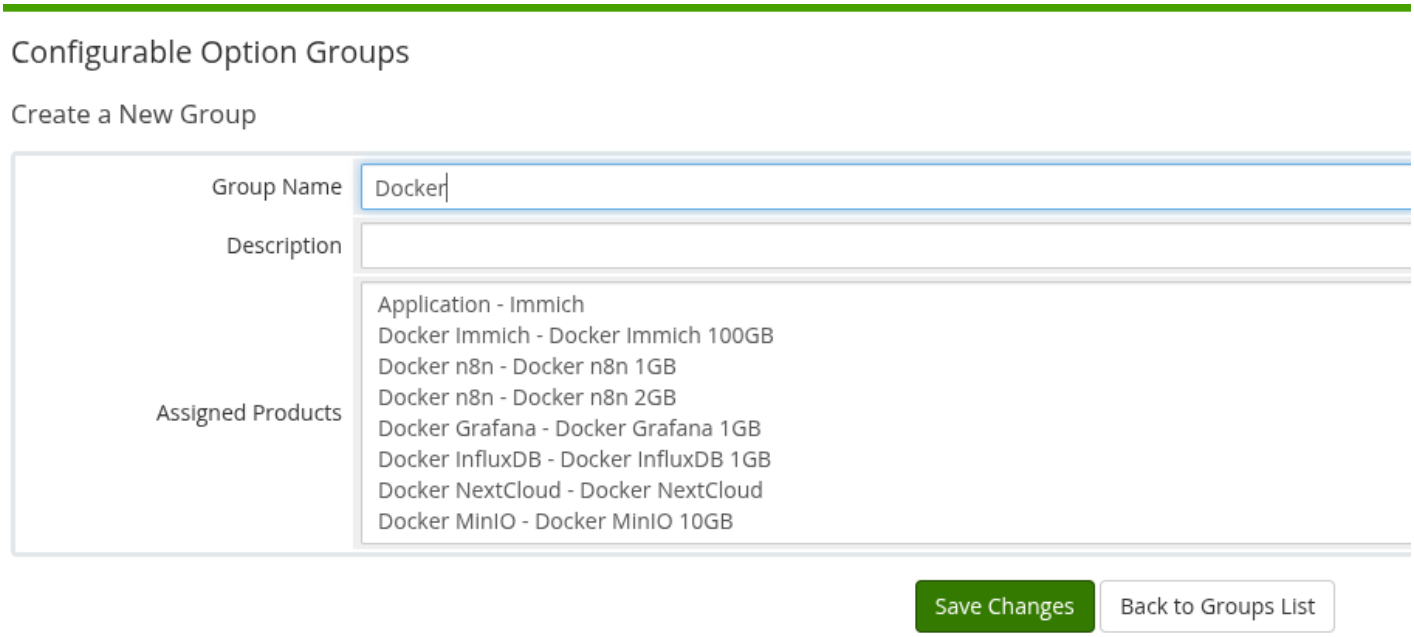
Add new Configurable Option to WHMCS

```
System Settings->Configurable Option->Create a New Group
```

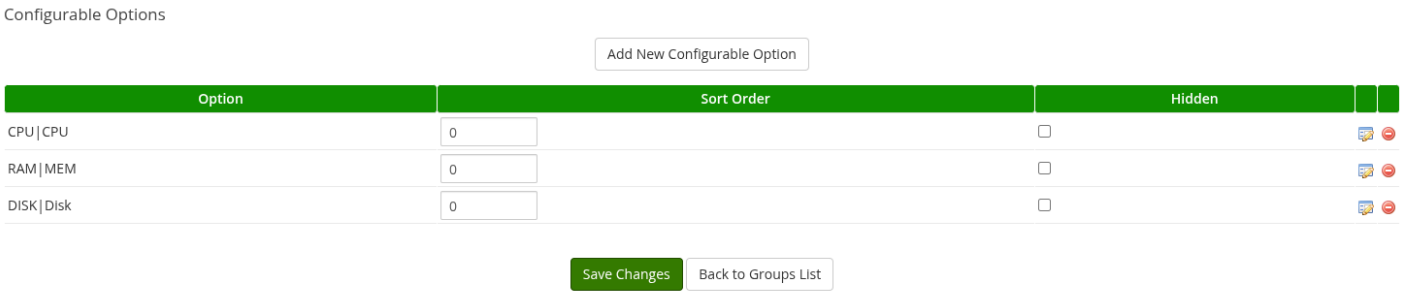


Enter the group name, description, and select the products you need.

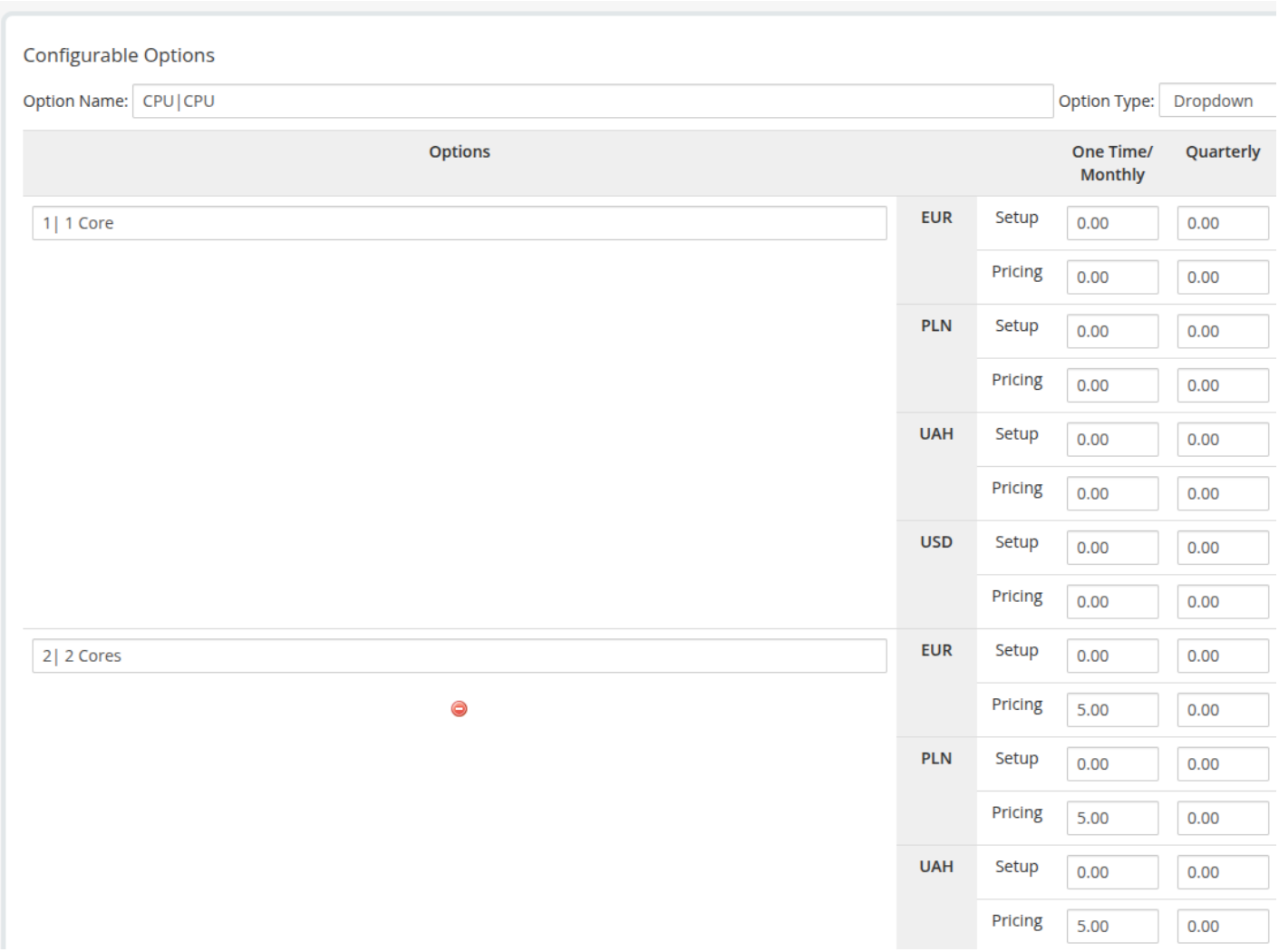
Group Name: Docker



Add Configurable Option

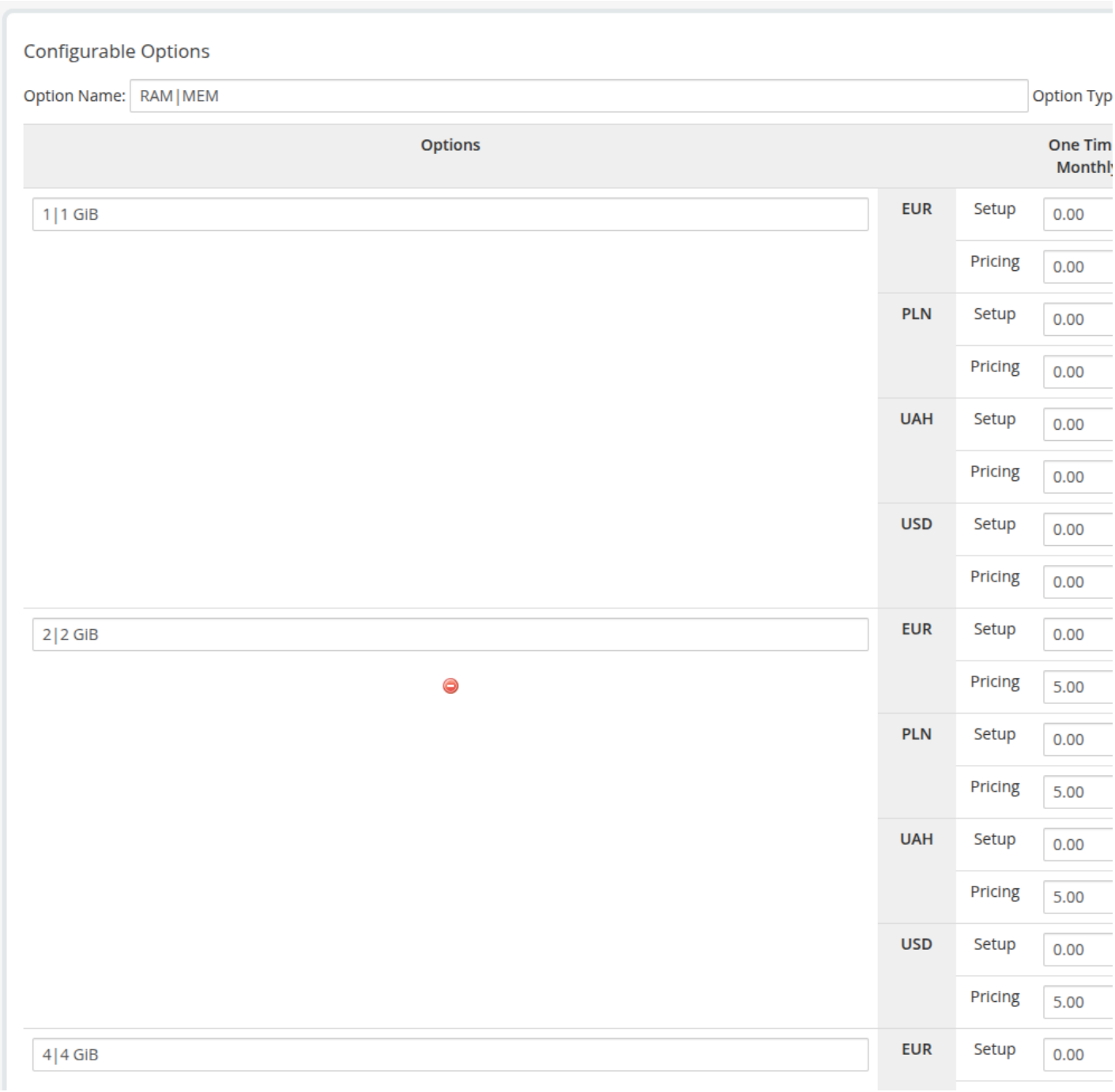


CPU|CPU - The first element before | must be left as is, but the second can be changed to suit your needs



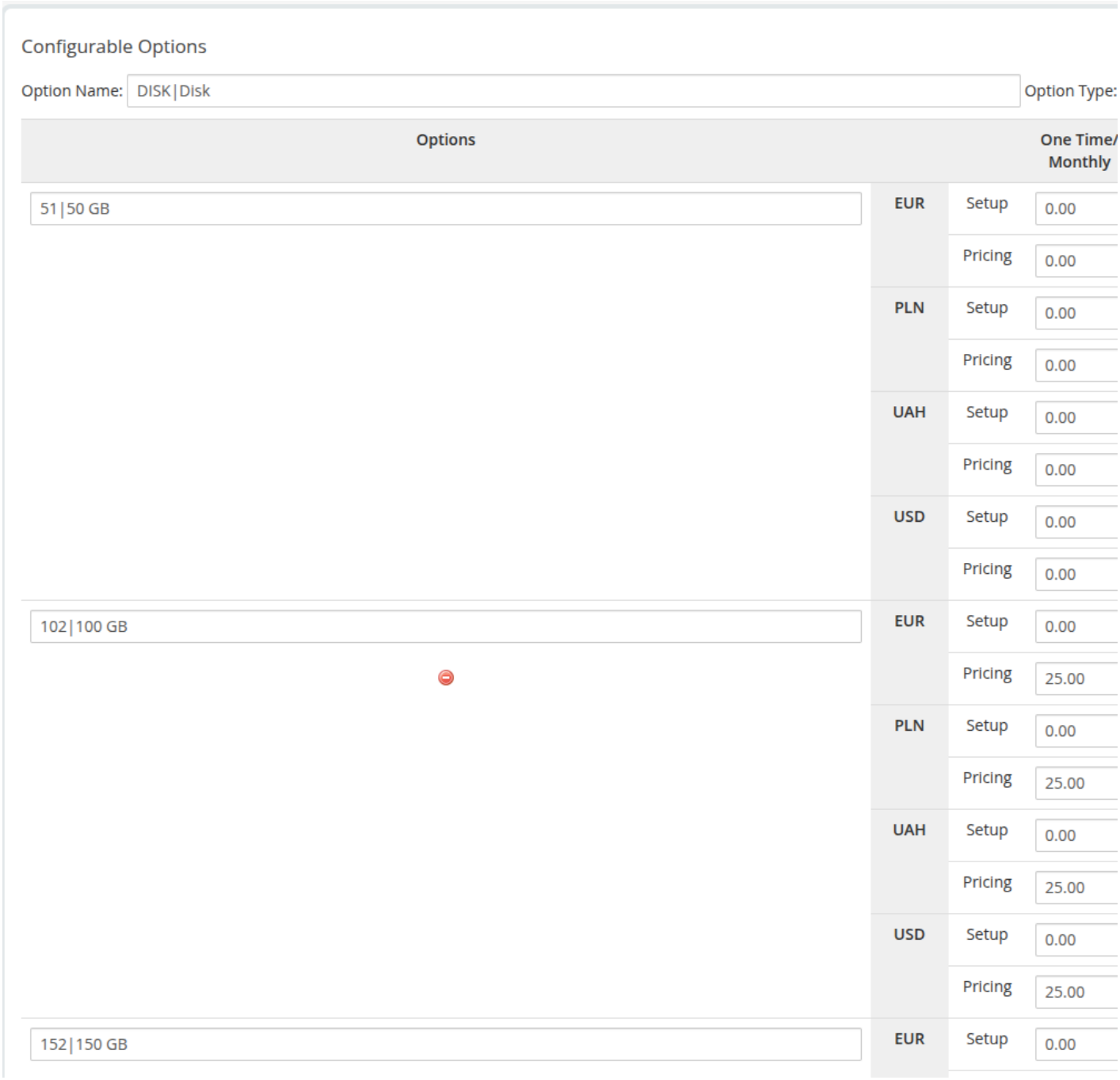
In the option, the first element is responsible for the quantity and the second element is responsible for the description.

RAM|MEM - The first element before | must be left as is, but the second can be changed to suit your needs



In the option, the first element is responsible for the quantity and the second element is responsible for the description.

DISK|Disk - The first element before | must be left as is, but the second can be changed to suit your needs



In the option, the first element is responsible for the quantity and the second element is responsible for the description.

Admin Area

Product Information

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Admin Panel – Container Management Overview

The **admin panel** is structured into two main sections with additional control buttons for container management.

Control Buttons

- **Container Start / Stop** – Start or stop the running container.
- **Mount Disk / Unmount Disk** – Attach or detach the container's disk to the host system. This modifies the **fstab** file to ensure proper mounting.

Container Status & Resource Monitoring

- **Status** – Displays the current state (Running / Stopped).
- **Name** – Unique identifier and domain of the container.
- **CPU Usage** – Shows current CPU load.
- **Memory Usage** – Displays RAM consumption in real-time.
- **Disk IO & Disk Mounted** – Tracks disk input/output operations.
- **Disk File** – Indicates the actual disk image size.
- **Network IO** – Shows network traffic statistics.
- **Log Button** – Loads and displays the container logs for debugging and monitoring.

Application Information

- **Version** – Displays the installed application version.
- **Owner** – Administrator managing the application.
- **Users** – List of assigned users with access to the application.

Module Commands: Create | Suspend | Unsuspend | Terminate | Change Package | Container Start | Container Stop | Mount disk | Unmount disk

API Connection status: API Connection OK

Container

Refresh | Log

Status	Running
Name	1-5342.d01-test.uuq.pl_immich (8e14264d1726)
CPU usage	99.42%
Memory usage	772MiB / 1GiB 75.39% / 24.61%
Disk IO	155MB / 555MB
Disk mounted	3.0G/9.8G 32% / 68%
Disk file	3.1G
Network IO	552MB / 1.11GB

Module Commands: Create | Suspend | Unsuspend | Terminate | Change Package | Container Start | Container Stop | Mount disk | Unmount disk

API Connection status: API Connection OK

Refresh | Log

```
    at MachineLearningRepository.predict (/usr/src/app/dist/repositories/machine-learning.repository.js:97:15)
    at async MachineLearningRepository.detectFaces (/usr/src/app/dist/repositories/machine-learning.repository.js:106:26)
    at async PersonService.handleDetectFaces (/usr/src/app/dist/services/person.service.js:239:52)
    at async JobService.onJobStart (/usr/src/app/dist/services/job.service.js:153:28)
    at async EventRepository.onEvent (/usr/src/app/dist/repositories/event.repository.js:126:13)
    at async Worker.processJob (/usr/src/app/node_modules/bullmq/dist/cjs/classes/worker.js:394:28)
    at async Worker.retryIfFailed (/usr/src/app/node_modules/bullmq/dist/cjs/classes/worker.js:581:24)
[31m[Nest] 6  - [39m03/18/2025, 1:24:29 AM [31m  ERROR[39m [33m[Microservices:{"id":"894964e9-afb0-41c9-97e7-4877582f7ca3","source":"upload"}][39m [31mUnable to run job handl
er (faceDetection/face-detection): Error: Machine learning request '{"facial-recognition":{"detection":{"modelName":"buffalo_l","options":{"minScore":0.7}},"recognition":{"modelNam
e":"buffalo_l"}}}' failed for all URLs[39m
[33m[Nest] 6  - [39m03/18/2025, 1:24:29 AM [33m   WARN[39m [33m[Microservices:MachineLearningRepository][39m [33mMachine learning request to "http://1-5342.d01-test.uuq.pl_m
l:3003" failed: fetch failed[39m
Error: Machine learning request '{"facial-recognition":{"detection":{"modelName":"buffalo_l","options":{"minScore":0.7}},"recognition":{"modelName":"buffalo_l"}}}' failed for all UR
Ls
    at MachineLearningRepository.predict (/usr/src/app/dist/repositories/machine-learning.repository.js:97:15)
    at async MachineLearningRepository.detectFaces (/usr/src/app/dist/repositories/machine-learning.repository.js:106:26)
    at async PersonService.handleDetectFaces (/usr/src/app/dist/services/person.service.js:239:52)
    at async JobService.onJobStart (/usr/src/app/dist/services/job.service.js:153:28)
    at async EventRepository.onEvent (/usr/src/app/dist/repositories/event.repository.js:126:13)
    at async Worker.processJob (/usr/src/app/node_modules/bullmq/dist/cjs/classes/worker.js:394:28)
    at async Worker.retryIfFailed (/usr/src/app/node_modules/bullmq/dist/cjs/classes/worker.js:581:24)
[33m[Nest] 6  - [39m03/18/2025, 1:24:55 AM [33m   WARN[39m [33m[Microservices:MachineLearningRepository][39m [33mMachine learning request to "http://1-5342.d01-test.uuq.pl_m
l:3003" failed: fetch failed[39m
[31m[Nest] 6  - [39m03/18/2025, 1:24:55 AM [31m  ERROR[39m [33m[Microservices:{"id":"9dc0b614-4286-46fe-b423-cd5a9a6ab499","source":"upload"}][39m [31mUnable to run job handl
er (faceDetection/face-detection): Error: Machine learning request '{"facial-recognition":{"detection":{"modelName":"buffalo_l","options":{"minScore":0.7}},"recognition":{"modelNam
e":"buffalo_l"}}}' failed for all URLs[39m
Error: Machine learning request '{"facial-recognition":{"detection":{"modelName":"buffalo_l","options":{"minScore":0.7}},"recognition":{"modelName":"buffalo_l"}}}' failed for all UR
Ls
    at MachineLearningRepository.predict (/usr/src/app/dist/repositories/machine-learning.repository.js:97:15)
    at async MachineLearningRepository.detectFaces (/usr/src/app/dist/repositories/machine-learning.repository.js:106:26)
    at async PersonService.handleDetectFaces (/usr/src/app/dist/services/person.service.js:239:52)
```

Dependent Containers

Refresh

Status	Running
Name	1-5342.d01-test.uuq.pl_ml (30d0e4fb5a29)
CPU usage	70.77% / 29.23%
Memory usage	224.3MiB / 1GiB — 21.9% / 78.1%

Status	Running
Name	1-5342.d01-test.uuq.pl_db (7987eef80297)
CPU usage	100%
Memory usage	185.9MiB / 1GiB — 18.16% / 81.84%

Status	Running
Name	1-5342.d01-test.uuq.pl_redis (b3200c47ac80)
CPU usage	99.67%
Memory usage	10.04MiB / 1GiB — 0% / 99.02%

App

Refresh

Version	2.2.53
Users	it@puq.pl

Metric Statistics

Metric Statistics

Metric	Enabled	Current Usage	Last Update
Traffic IN (GB)	✓	0.00 GB	4 hours ago
Traffic OUT (GB)	✓	0.01 GB	4 hours ago

Refresh Now

Addons

Reg Date	Name	Pricing	Status	Next Due Date		

Client Area

Home screen

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

Client Area Overview – Logical Structure

The **main screen** of the client area is divided into **three logical sections**:

1. **Navigation Block**
   - **"Go to Immich"**: Direct link to access the application.
   - **"User Manual"**: Opens the official documentation or user guide.
2. **Resource Usage Block**
   - Displays real-time statistics on container resource usage:
     - **CPU Usage**: Number of allocated CPUs and current load.
     - **Memory Usage**: RAM consumption, helping clients understand available capacity.
     - **Disk Usage**: Storage consumption within the container.
   - This section is crucial for users to **monitor performance** and determine whether they need to **upgrade their package**.
3. **Application Information & Controls**
   - **Application Version**: Displays the installed software version.
   - **Owner Information**: Indicates the primary administrator of the application.
   - **User List**: Shows active users associated with the instance.
   - **Reset Password Button**: Allows the client to reset the administrator password for the application.

This **clear structure** ensures that users have **quick access** to their application, **real-time monitoring** of resource usage, and **essential management functions** in one place.

Manage | Metrics

Go to Immich | User manual

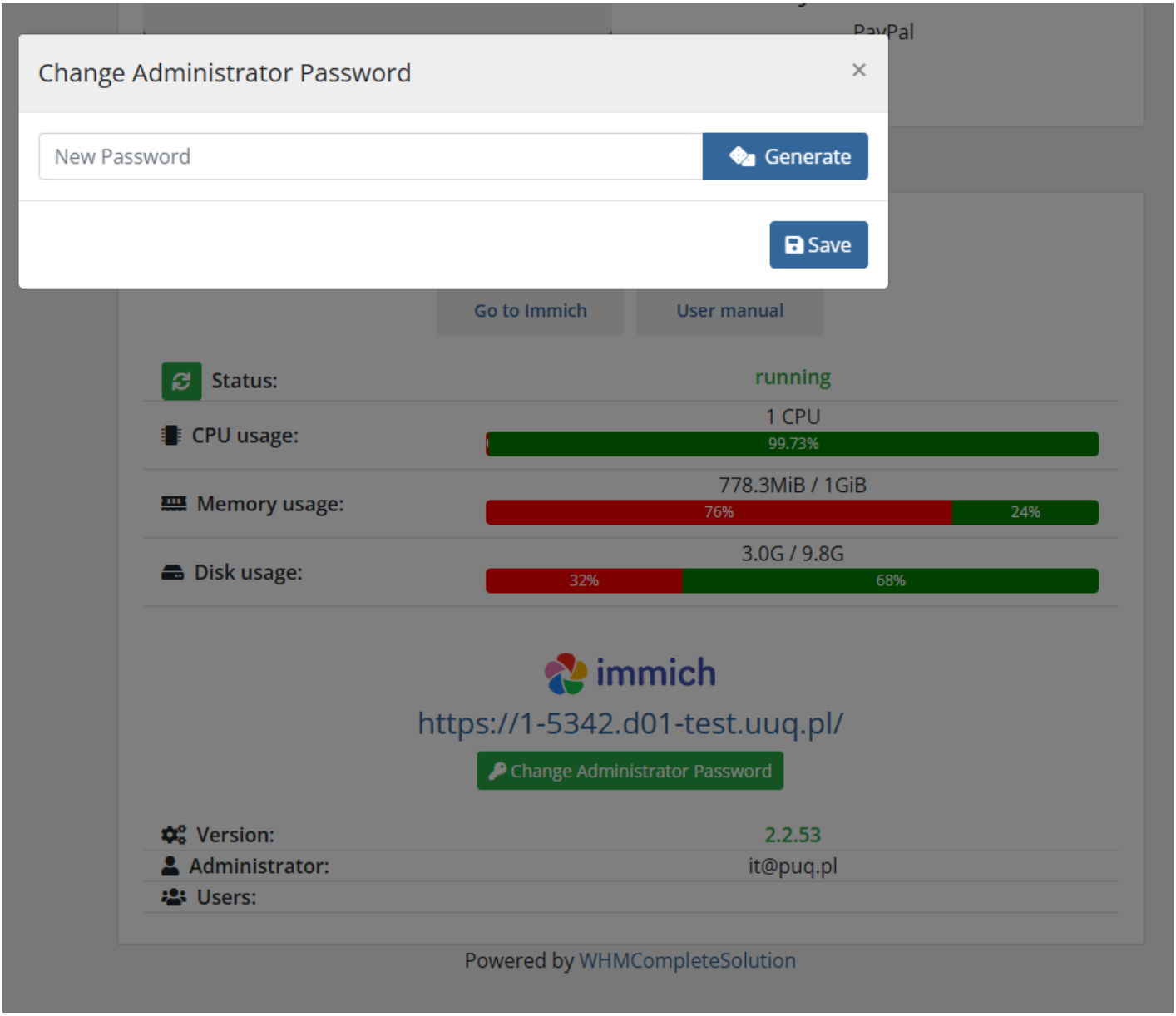
Status:	running
CPU usage:	1 CPU 99.73%
Memory usage:	778.3MiB / 1GiB 76% 24%
Disk usage:	3.0G / 9.8G 32% 68%

immich

https://1-5342.d01-test.uuq.pl/

Change Administrator Password

Version:	2.2.53
Administrator:	it@puq.pl
Users:	


PayPal
Change Administrator Password
New Password
Generate
Save
Go to Immich
User manual
Status:
running
CPU usage:
1 CPU
99.73%
Memory usage:
778.3MiB / 1GiB
76%
24%
Disk usage:
3.0G / 9.8G
32%
68%
immich
https://1-5342.d01-test.uuq.pl/
Change Administrator Password
Version:
2.2.53
Administrator:
it@puq.pl
Users:
Powered by WHMCompleteSolution

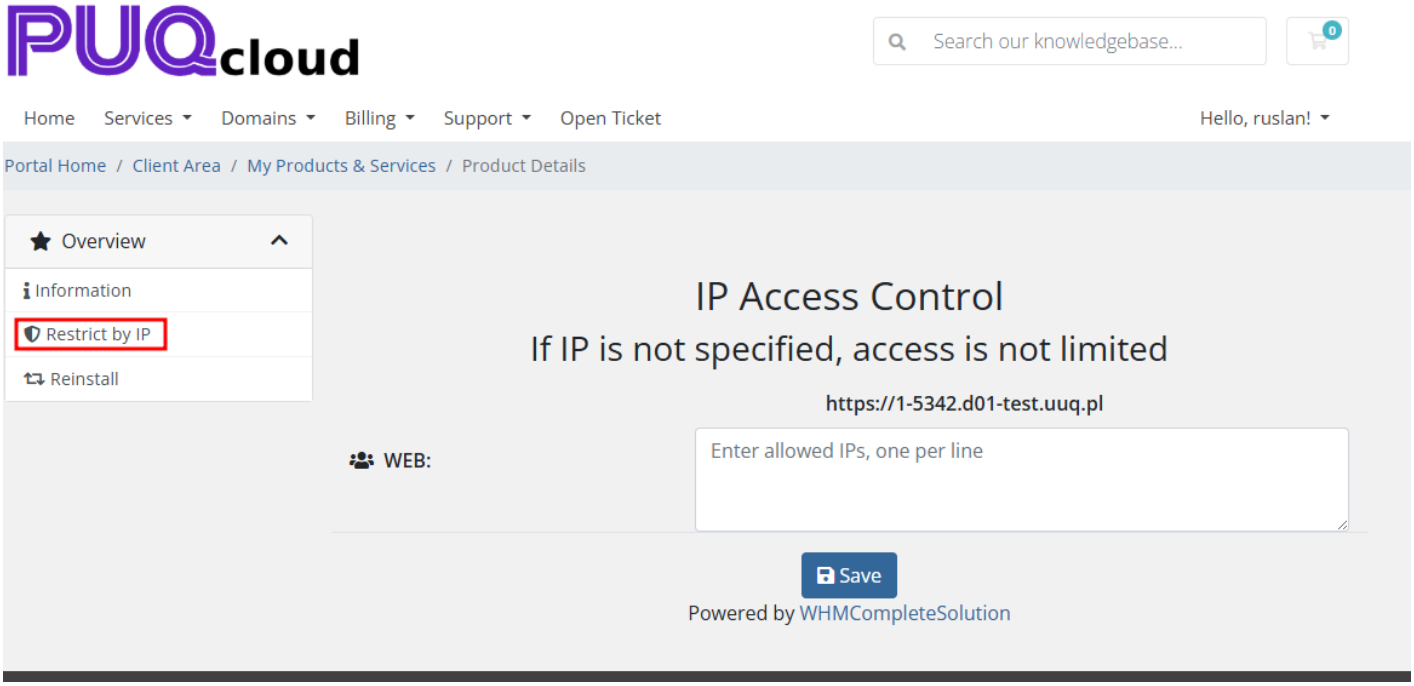
IP Access Control

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

In the client area, the client can configure access to their resource by entering allowed IP addresses in the appropriate section on the **Restrict by IP** page.

If no IP addresses are specified, access is open to all IP addresses.



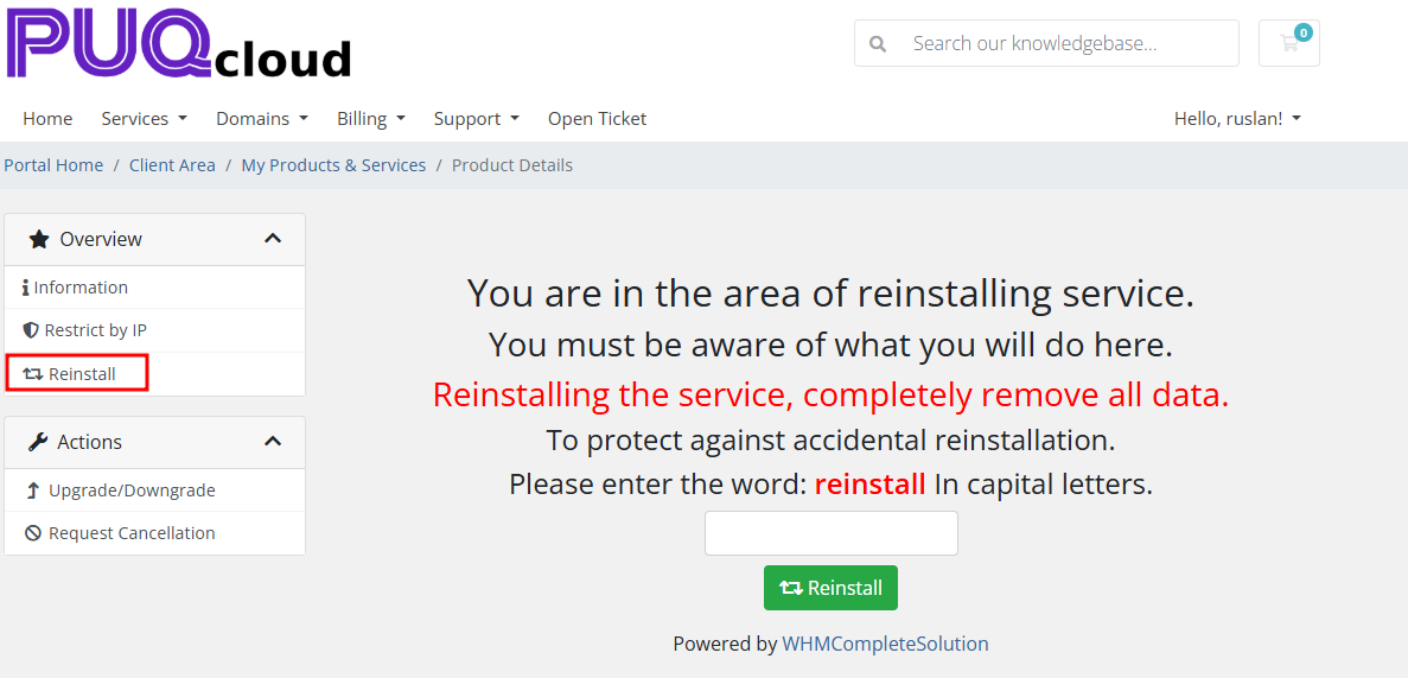
Reinstall

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

The client has the option to **fully reinstall the application**, which will result in **complete data loss**.

This action can be performed on the **Reinstall** page, which also includes **protection against accidental reinstallation**.



Client Area

Metrics

Docker Immich module **WHMCS**

Order now | Download | FAQ | n8n

If you use metrics for application traffic billing, the Metrics tab will display the usage statistics for the metrics.

