

Docker MinIO

WHMCS module

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

- [Description](#)
- [What is MinIO](#)
- [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 \(puqDockerMinIO Welcome Email\)](#)
 - [Email Template \(puqDockerMinIO Update Email\)](#)
 - [Email Template \(puqDockerMinIO Notification disk limit\)](#)
- [Admin Area](#)
 - [Product Information](#)
- [Client Area](#)

- Home screen
- IP Access Control
- Reinstall
- Metrics

Description

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

Key Features

☐☐ Automated Container Management

- Automatic creation of an MinIO 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 MinIO 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

1🗪 Environment Preparation

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

2🗪 Module Installation

- Upload and activate the **WHMCS Docker MinIO 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 **MinIO instances** through WHMCS easy, automated, and flexible! ☑

 Go to MinIO

 User manual

 Status: **running**

 CPU usage: **1 CPU**
99.98%

 Memory usage: **122.3MiB / 1GiB**
11.94% 88.06%

 Disk usage: **128K / 974M**
99%

MINIO

Web Console: <https://console.1741969814.d01-test.uuq.pl/>

 REST API: <https://1741969814.d01-test.uuq.pl> 

 Username: **befDzX** 

 Password:  

 [Change Administrator Password](#)

 Version: **RELEASE.2025-02-28T09-55-16Z**

 Administrator: **befDzX**

 Users:

API Connection status

✓ API Connection OK

Container

[Refresh](#) [Log](#)

Status	Running
Name	1741969814.d01-test.uuq.pl (f36febde549a)
CPU usage	<div style="width: 99.93%;"><div style="width: 99.93%;"></div></div> 99.93%
Memory usage	172.3MiB / 1GiB <div style="width: 16.83%;"><div style="width: 16.83%;"></div></div> 16.83% <div style="width: 83.17%;"><div style="width: 83.17%;"></div></div> 83.17%
Disk IO	422kB / 5.28MB
Disk mounted	128K/974M <div style="width: 99%;"><div style="width: 99%;"></div></div> 99%
Disk file	35M
Network IO	163kB / 13.2MB

App

[Refresh](#)

Version	RELEASE.2025-02-28T09-55-16Z
Users	

Metric Statistics

Metric	Enabled	Current Usage	Last Update
Traffic IN (GB)	✓	1.21 GB	56 minutes ago
Traffic OUT (GB)	✓	0.16 GB	56 minutes ago

[Refresh Now](#)

What is MinIO

Docker MinIO module **WHMCS**

[Order now](#) | [Download](#) | [FAQ](#) | [n8n](#)

MinIO is an open-source object storage solution that is fully compatible with Amazon S3. It is designed for storing large amounts of unstructured data such as files, images, videos, backups, and logs. MinIO provides a high-performance alternative to traditional storage solutions, offering scalability, security, and ease of use.

Key Features of MinIO:

1. **Amazon S3 Compatibility**

MinIO is designed to be fully compatible with the Amazon S3 API. This means that applications that interact with S3 can seamlessly work with MinIO without any changes to the codebase. It provides the same object storage functionality that developers are used to with S3, making it easy to migrate or integrate with existing systems.

2. **High Performance**

MinIO is built for high-performance applications. It is optimized to handle large-scale data and provide fast throughput, making it suitable for storing and serving massive datasets. Its low-latency and high throughput capabilities make it ideal for data-intensive applications like AI/ML workloads, big data storage, and video streaming.

3. **Scalability**

MinIO can scale easily from a single server to a distributed cluster. It supports high availability and can handle petabytes of data across a large number of nodes. Whether you're starting with a small deployment or expanding to a global system, MinIO provides the scalability you need to grow without major reconfiguration.

4. **Security**

Security is a top priority in MinIO. It supports end-to-end encryption, both in transit (using TLS) and at rest (using AES-256 encryption). Access control policies and fine-grained permissions help ensure that only authorized users can access your data. MinIO also supports bucket versioning and data redundancy to protect against data loss.

5. **Simplicity**

MinIO is designed to be simple to deploy and manage. It can be run as a single binary, making it easy to install and configure. It supports Docker and Kubernetes, allowing you to deploy it in containerized environments or cloud-native architectures. Additionally, MinIO's web interface provides a user-friendly way to manage and monitor your storage.

6. **Use Cases**

MinIO is used in a variety of scenarios, including:

- **Cloud Storage:** Build your own private cloud storage solution.
- **Big Data:** Store and manage large datasets for analytics and processing.
- **Backup:** Create reliable, cost-effective backup solutions.
- **AI/ML:** Store training datasets and models for machine learning applications.
- **Media and Streaming:** Serve high-resolution images, videos, and audio files.

Why Choose MinIO?

MinIO stands out for its simplicity, performance, and compatibility with existing tools and applications. It's an ideal choice for businesses and developers looking to build scalable, secure, and cost-effective object storage solutions. MinIO's open-source nature makes it an affordable alternative to proprietary cloud storage services like Amazon S3, while still offering many of the same features and capabilities.

Whether you're building a new application or migrating an existing system, MinIO offers a powerful and flexible storage solution that can meet your needs as they evolve.

Changelog

Docker MinIO module **WHMCS**

[Order now](#) | [Download](#) | [FAQ](#) | [n8n](#)

v1.0 Released 17-03-2025

First version

Installation and configuration guide

Basic concepts and requirements

Docker MinIO 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 MinIO module** is part of the **WHMCS Docker module series** developed by **PUQcloud**. This module enables service providers to offer **MinIO 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 MinIO 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-MinIO/php82/PUQ_WHMCS-Docker-MinIO-latest.zip
```

PHP 8.1

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

PHP 7.4

```
wget http://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-MinIO/php74/PUQ_WHMCS-Docker-MinIO-latest.zip
```

All versions are available via link:

https://download.puqcloud.com/WHMCS/servers/PUQ_WHMCS-Docker-MinIO/

2. Unzip the archive with the module.

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

3. Copy and Replace "puqDockerMinIO" from "PUQ_WHMCS-Docker-MinIO" to "WHMCS_WEB_DIR/modules/servers/"

Preparing Docker Server

Docker MinIO 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>

Setting up n8n workflow

Docker MinIO module **WHMCS**

[Order now](#) | [Download](#) | [FAQ](#) | [n8n](#)

Overview

The **Docker MinIO 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:

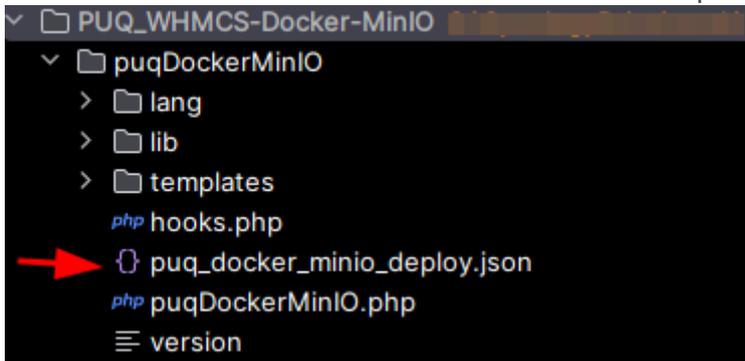
Option 1: Use the Latest Version from the n8n Marketplace

- The latest workflow templates for our modules are available on the official n8n marketplace.

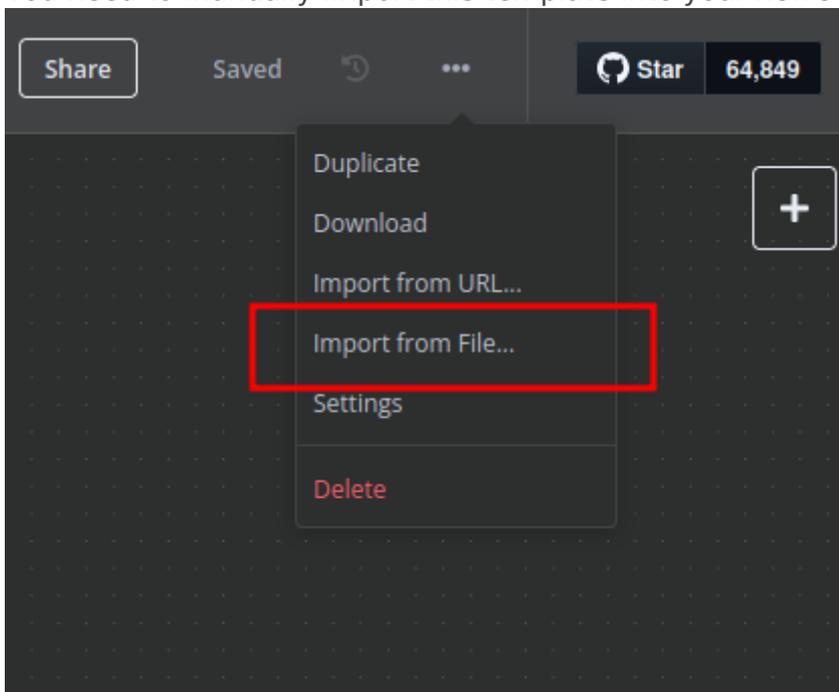
- Visit our profile to access all available templates: [PUQcloud on n8n](#)

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

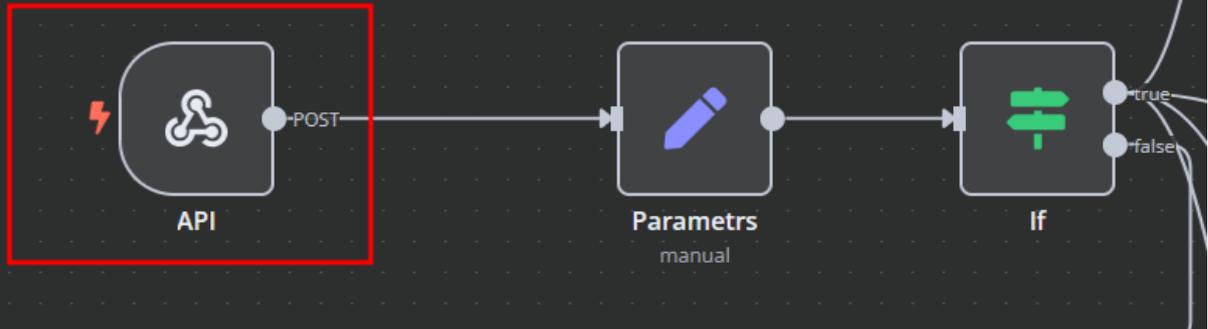
Configure API Webhook and SSH Access

- Create a **Basic Auth Credential** for the Webhook API Block in n8n.

• screen_right

Additional Resources

- Full documentation: <https://doc.puq.info/books/docker-minio-whmcs-module>
- WHMCS module: <https://puqcloud.com/whmcs-module-docker-minio.php>



Pull in events from Webhook

[Listen for test event](#)

Once you've finished building your workflow, run it without having to click this button by using the production webhook URL. [More info](#)

API [Listen for test event](#)

Parameters **Settings** [Docs](#)

Webhook URLs

Test URL **Production URL**

POST `https://n8n.puqcloud.com/webhook-test/docker-minio`

HTTP Methods

POST

Path

`docker-minio`

Authentication

Basic Auth

Credential for Basic Auth

`Incom API User`

- Incom API User**
Basic Auth
- Unnamed credential**
Basic Auth
- + Create new credential**

No properties

Add option

API credential
Basic Auth

Save

Connection

Need help filling out these fields? [Open docs](#)

Sharing

Details

User

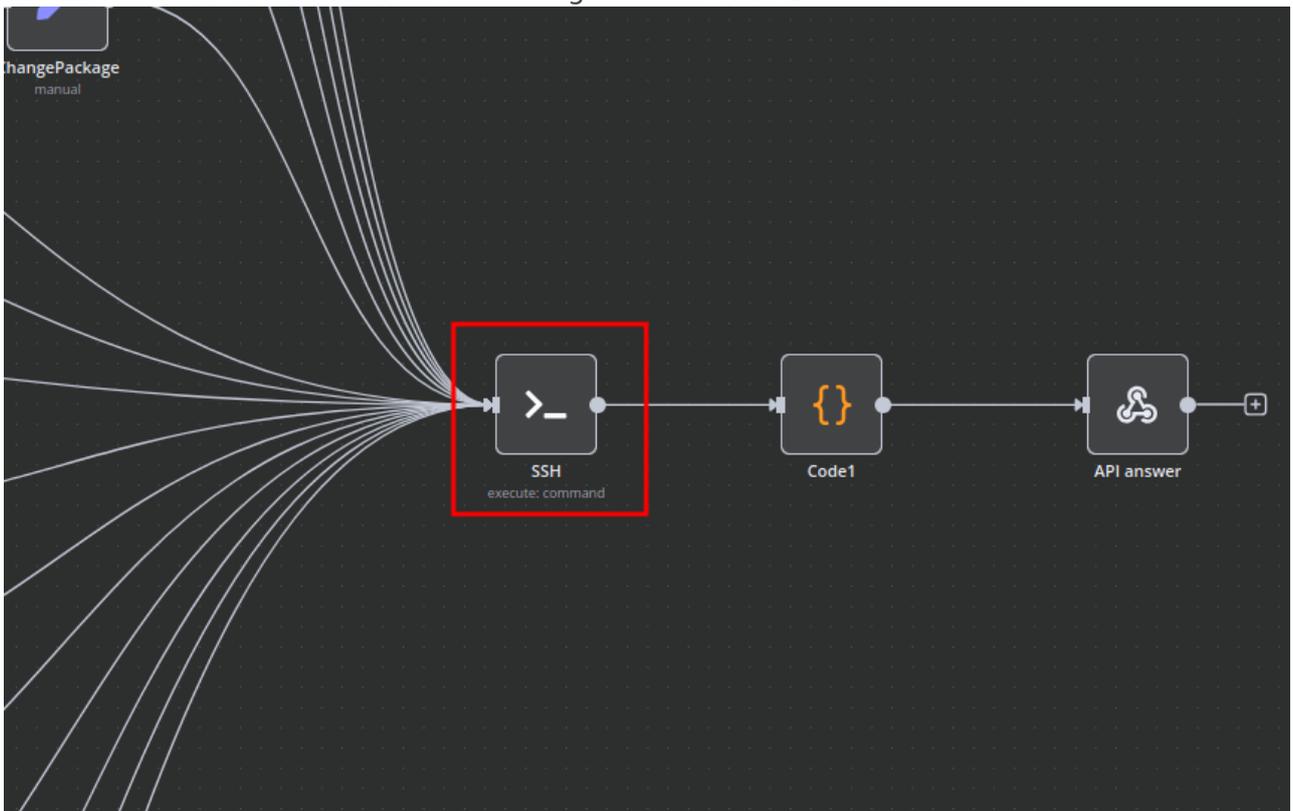
test

Password

.....

Enterprise plan users can pull in credentials from external vaults. [More info](#)

- 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

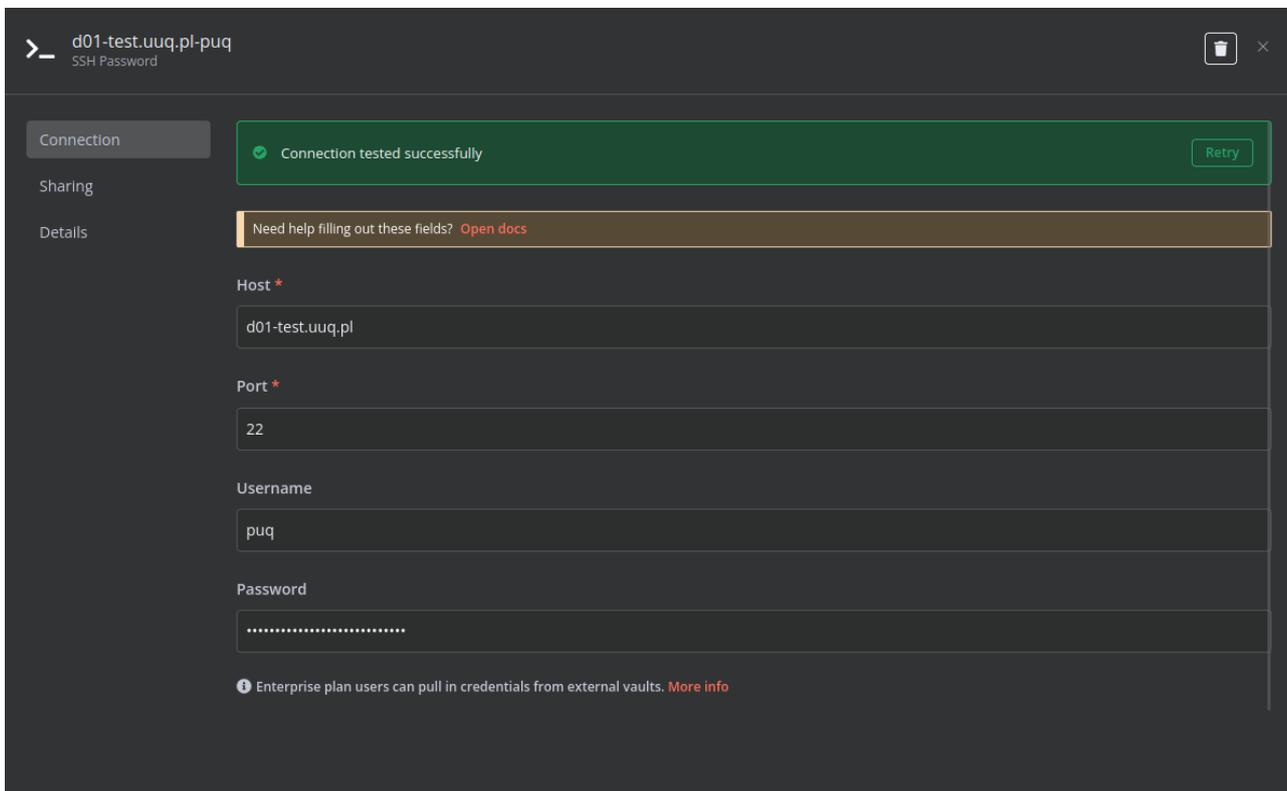
Execute

Command

fx {{ \$json.sh }}

Working Directory

fx /

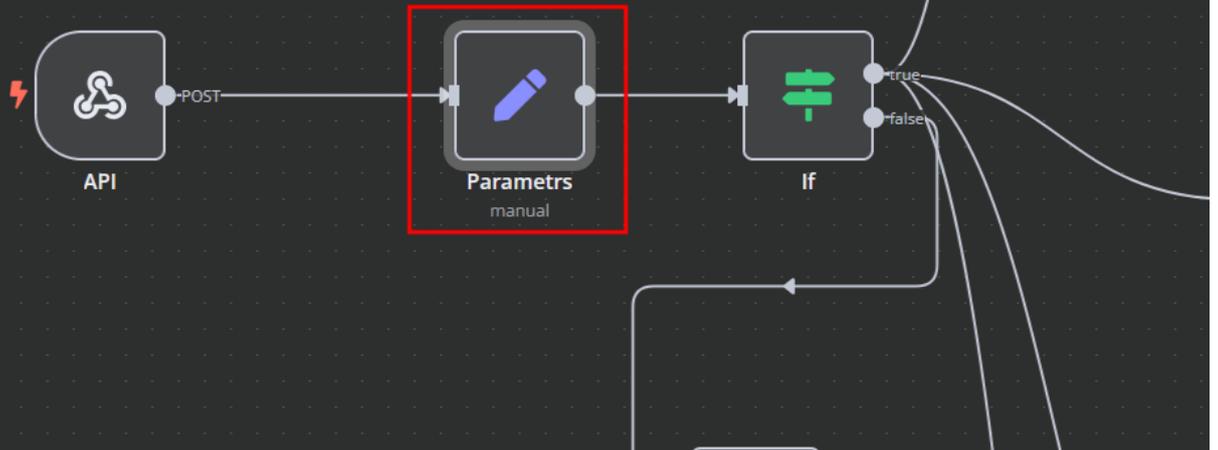


Modify Template Parameters

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

Additional Resources

- Full documentation: <https://doc.puq.info/books/docker-minio-whmcs-module>
- WHMCS module: <https://puqcloud.com/whmcs-module-docker-minio.php>



 **Parameters** 🚧 Test step

Parameters Settings Docs [↗](#)

Mode
Manual Mapping ▼

Fields to Set

server_domain
A String ▼
d01-test.uuq.pl
[empty]

clients_dir
A String ▼
/opt/docker/clients
[empty]

mount_dir
A String ▼
/mnt
[empty]

screen_left
A String ▼
{{
[empty]

screen_right
A String ▼
}}
[empty]

- `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).

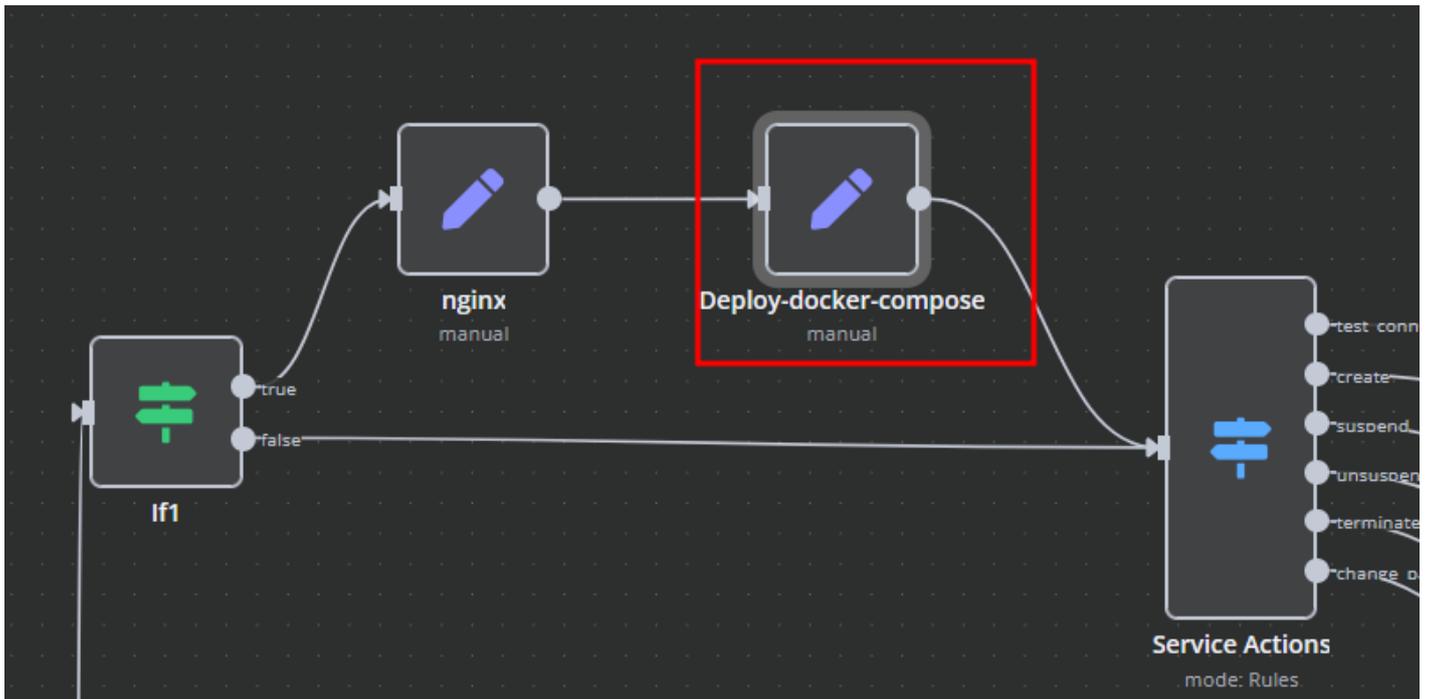
Do not modify the following technical parameters:

- `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](#)

```
version: "3"

services:
  {{ $('API').item.json.body.domain }}:
    image: minio/minio
    restart: unless-stopped
    container_name: {{ $('API').item.json.body.domain }}
    command: server /data --console-address ":9001"
    environment:
      MINIO_ROOT_USER: {{ $('API').item.json.body.username }}
      MINIO_ROOT_PASSWORD: {{ $('API').item.json.body.password }}
      MINIO_BROWSER_REDIRECT_URL: https://console.{{ $('API').item.json.body.domain }}
      LETSENCRYPT_HOST: {{ $('API').item.json.body.domain }},console.{{
$('API').item.json.body.domain }}
      VIRTUAL_HOST_MULTIPORTS: |-
        {{ $('API').item.json.body.domain }}:
          "/":
            port: 9000
        console.{{ $('API').item.json.body.domain }}:
          "/":
            port: 9001
    volumes:
      - "{{ $('Parameters').item.json.mount_dir }}/{{ $('API').item.json.body.domain }}/data:/
data"
    networks:
      - nginx-proxy_web
    mem_limit: "{{ $('API').item.json.body.ram }}"G"
    cpus: "{{ $('API').item.json.body.cpu }}"

networks:
  nginx-proxy_web:
    external: true
```

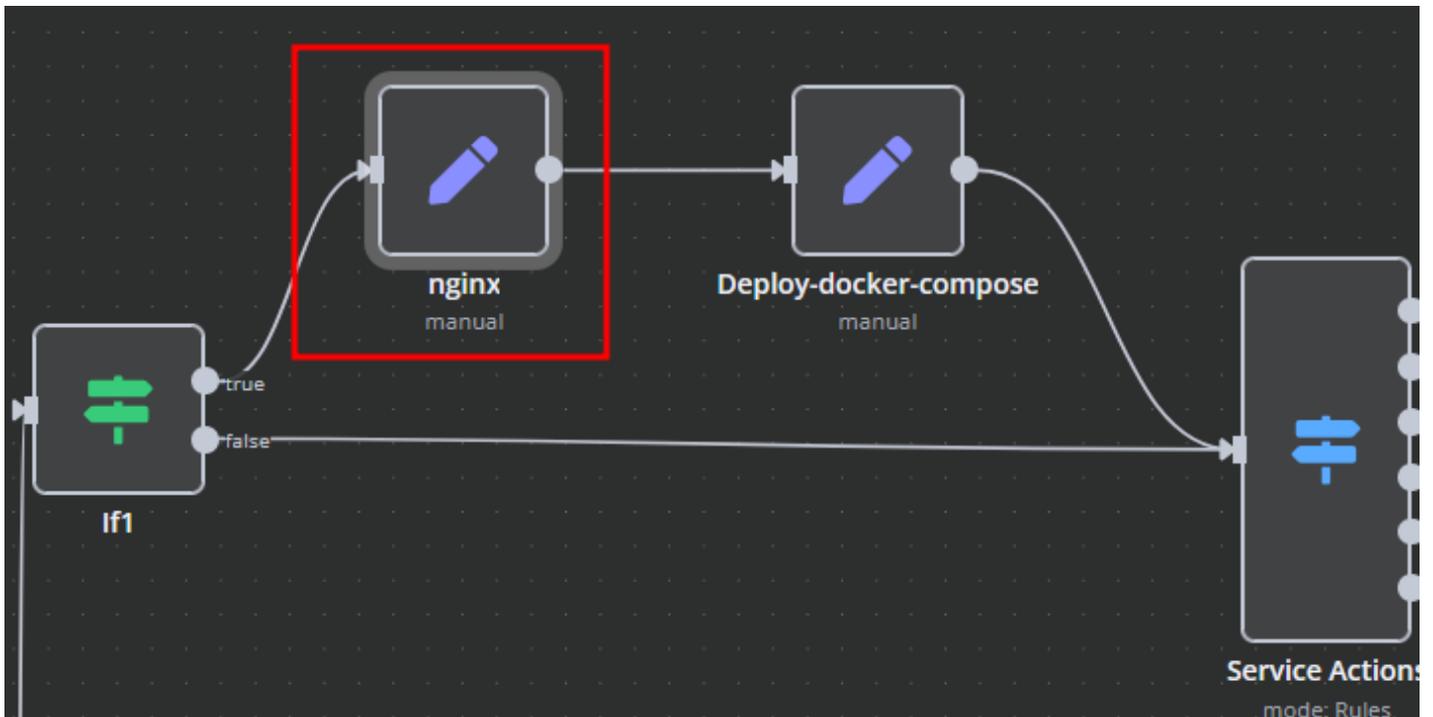
nginx

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.

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



Mode

Manual Mapping

Fields to Set

main

A String

```
= ignore_invalid_headers off;
client_max_body_size 0;
proxy_buffering off;
proxy_request_buffering off;
```

ignore_invalid_headers off; client_max_body_size 0; proxy_buffering off; proxy_request_buffering off;

main_location

A String

```
= # Custom header
proxy_set_header Host $http_host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
```

Custom header proxy_set_header Host \$http_host; proxy_set_header X-Real-IP \$remote_addr; proxy_set_head...

console

A String

```
= ignore_invalid_headers off;
client_max_body_size 0;
proxy_buffering off;
proxy_request_buffering off;
```

ignore_invalid_headers off; client_max_body_size 0; proxy_buffering off; proxy_request_buffering off;

console_location

A String

```
= # Custom header
proxy_set_header Host $http_host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;
proxy_set_header X-Real-IP-Proxy true;
```

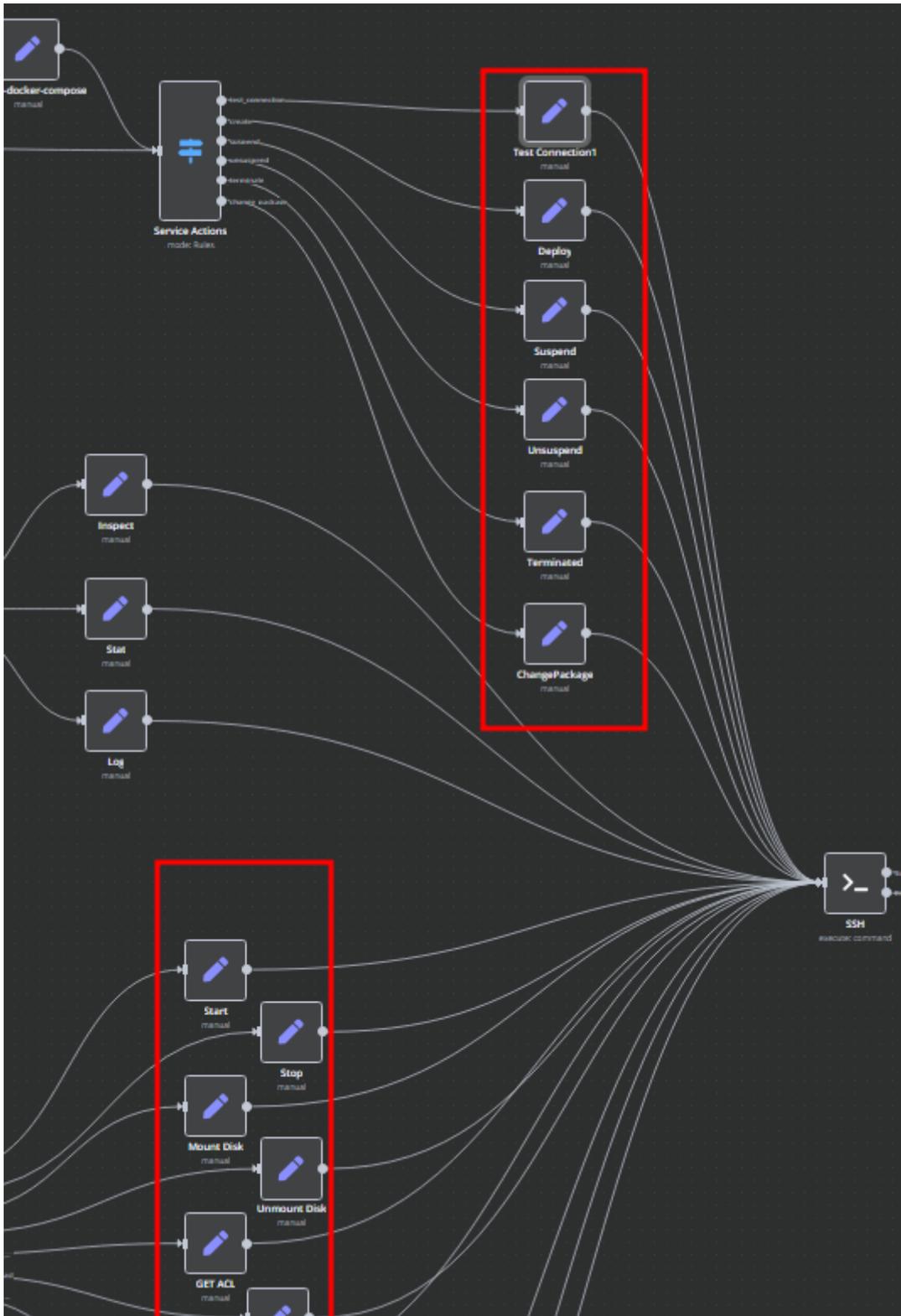
Custom header proxy_set_header Host \$http_host; proxy_set_header X-Real-IP \$remote_addr; proxy_set_head...

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.



Add server

Docker MinIO module **WHMCS**

[Order now](#) | [Download](#) | [FAQ](#) | [n8n](#)

Add a new server to the system WHMCS.

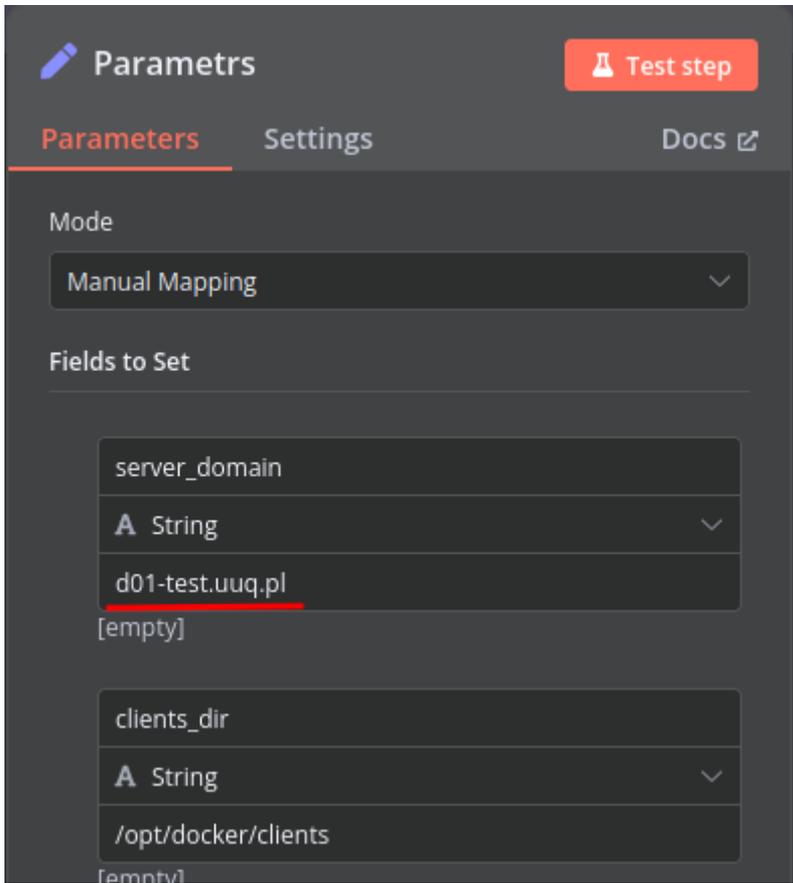
System Settings->Servers->Add New Server

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

Name	<input type="text" value="d01-test.uuq.pl"/>
Hostname	<input type="text" value="d01-test.uuq.pl"/>
IP Address	<input type="text"/>
Assigned IP Addresses (One per line)	<input type="text"/>
Monthly Cost	<input type="text" value="0.00"/>
Datacenter/NOC	<input type="text"/>
Maximum No. of Accounts	<input type="text" value="200"/>
Server Status Address	<input type="text"/> To display this server on the server status page, enter the full path to the server status folder (required to be uploaded to each server you want to monitor) - eg. https://www.example.com/status/
Enable/Disable	<input type="checkbox"/> Check to disable this server

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.



The screenshot shows the 'Parameters' configuration page in n8n. At the top, there is a 'Test step' button. Below it, there are tabs for 'Parameters' (selected), 'Settings', and 'Docs'. The 'Mode' is set to 'Manual Mapping'. Under 'Fields to Set', there are two parameter entries:

- server_domain**: Type 'String', value 'd01-test.uuq.pl' (underlined in red).
- clients_dir**: Type 'String', value '/opt/docker/clients'.

In the **Server Details** section, select the "**PUQ Docker MinIO**" 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 MinIO <input type="button" value="Test Connection"/>
	✓ Connection successful. Some values have been auto-filled.
Username	<input type="text" value="minio"/>
Password	<input type="password" value="...."/>
Access Hash	<input type="text" value="https://n8n.puqcloud.com/webhook/docker-minio"/>
Secure	<input checked="" type="checkbox"/> Check to use SSL Mode for Connections

Webhook URLs

Test URL

Production URL

POST

https://n8n.puqcloud.com/webhook/docker-minio

HTTP Methods

POST

Path

docker-minio

Authentication

Basic Auth

Credential for Basic Auth

Incom API User

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 MinIO 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 MinIO"** module

Edit Product

The screenshot shows the 'Edit Product' interface for the 'PUQ Docker MinIO' module. The 'Module Settings' tab is active. The configuration includes:

- Module Name:** PUQ Docker MinIO
- Server Group:** None
- License key:** [Redacted] success: 2025-04-09T17:58:30+02:00
- Disk space:** 1 GB Ex: 1
- CPU:** 1 Ex: 0.1
- RAM:** 1 GB Ex: 0.1
- Link to instruction:** https://puq.info/
- Client Area:** A link to the instruction will be reflected in the client area.
- Main domain:** [Empty field]
- Subdomain:** {unixtime}
- Notification, used disk space X %:** 60
- Notification disk limit email template:** puqDockerMinIO Notification disk limit

At the bottom, there are two toggle switches for 'Metric Billing': 'Traffic IN (GB)' and 'Traffic OUT (GB)', both currently set to 'ON'.

- **License key** - A pre-purchased license key for the **"PUQ Docker MinIO"** 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.

Metric Billing

Docker MinIO 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/>

Products/Services

Edit Product

The screenshot shows the WHMCS Product Configuration interface. At the top, there are tabs for 'Details', 'Pricing', 'Module Settings', 'Custom Fields', 'Configurable Options', 'Upgrades', 'Free Domain', 'Cross-sells', 'Other', and 'Links'. The 'Pricing' tab is active. Below the tabs, there are several sections for configuring the product. At the bottom of the interface, there is a 'Metric Billing' section with two toggle switches: 'Traffic IN (GB)' and 'Traffic OUT (GB)'. Both switches are currently turned 'ON' and have a 'Configure Pricing' link next to them. A red box highlights this section. In the bottom right corner of the interface, there is a link that says 'Switch to Advanced Mode'.

Configure Pricing



Traffic IN (GB)

Metric Type: Monthly

Metric Unit: GigaBytes

Pricing

Quantity Included

Scheme:

0.00

Per Unit  Total Volume  Graduated 

Price Per GB

PLN

EUR

UAH

USD

1.00

1.00

1.00

1.00

Close

Save

Email Template (puqDockerMinIO Welcome Email)

Docker MinIO 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:** puqDockerMinIO Welcome Email

Create New Email Template

Email Type

Product/Service

Unique Name

puqDockerMinIO Welcome Email

Cancel

Create

Subject:

MinIO 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 MinIO instance is in progress.

Within the next 4 minutes, you will be able to use your MinIO instance.

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

Here is the link to your MinIO server.

[https://console.{\\$service_domain}/](https://console.{$service_domain}/)

Thank you for choosing us.

{\$signature}

Subject: MinIO Order Information

File ▾ Edit ▾ View ▾ Insert ▾ Format ▾ Table ▾ Help ▾

Paragraph ▾ Verdana ▾ 11pt ▾ **B** *I* ~~S~~ U A ▾ **A** ▾    

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 **MinIO** instance is in progress.
Within the next 4 minutes, you will be able to use your **MinIO** instance.

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

Here is the link to your **MinIO** server.

[https://console.{ \\$service_domain }/](https://console.{ $service_domain }/)

Thank you for choosing us.

{ \$signature }

P 82 WORDS POWERED BY TINYMCE 

Email Template (puqDockerMinIO Update Email)

Docker MinIO 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:** puqDockerMinIO Update Email

Create New Email Template ×

Email Type

Product/Service ▾

Unique Name

puqDockerMinIO Update Email

Cancel

Create

Subject:

MinIO Update Information

Body:

Dear {\$client_name},

Your instance is currently being updated.

You will be able to use your MinIO server again within 3 minutes.

Here is the link to your MinIO server.

[https://console.{\\$service_domain}/](https://console.{$service_domain}/)

Thank you for choosing us.

{\$signature}

Subject: MinIO Update Information

File ▾ Edit ▾ View ▾ Insert ▾ Format ▾ Table ▾ Help ▾

Paragraph ▾ Verdana ▾ 11pt ▾ **B** *I* ~~S~~ U A ▾ **A** ▾    ▾  ▾

         ▾             

Dear **{\$client_name}**,

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

Here is the link to your **MinIO** server.

[**https://console.{\\$service_domain}/**](https://console.{$service_domain}/)

Thank you for choosing us.

{\$signature}

P 38 WORDS POWERED BY TINYMCE 

Email Template (puqDockerMinIO Notification disk limit)

Docker MinIO 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:** puqDockerMinIO Notification disk limit

Create New Email Template ×

Email Type

Product/Service ▾

Unique Name

puqDockerMinIO Notification disk limit

Cancel

Create

Subject:

Disk space usage `{%disk_used_percentage}%`

Body:

Dear `{%client_name%}`,

We want to inform you that your MinIO 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 ▾ **B** *I* ~~S~~ U A ▾ **A** ▾    

Dear **$\{\$client_name\}$** ,

We want to inform you that your **MinIO** 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 

Admin Area

Product Information

Docker MinIO 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.

Server: d01-test.uuq.pl (2/200 Accounts) | Domain: 1741969814.d01-test.uuq.pl | Username: befDzX | Password: GEpChGju | Status: Active

Recurring Amount: 0.00 | Next Due Date: N/A | Termination Date: | Billing Cycle: Free | Payment Method: PayPal | Promotion Code: None

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

API Connection status: ✔ API Connection OK

Container Metrics:

Status	Running
Name	1741969814.d01-test.uuq.pl (f36febde549a)
CPU usage	99.97%
Memory usage	120.5MiB / 1GiB (11.77% / 88.23%)
Disk IO	483KB / 12.7MB
Disk mounted	160K/974M (99%)
Disk file	37M
Network IO	172KB / 13.2MB

Product/Service: Docker MinIO 10 | Quantity: 1 | First Payment Amount: 0.00 | Recurring Amount: 0.00 | Next Due Date: N/A | Termination Date: | Billing Cycle: Free | Payment Method: PayPal | Promotion Code: None

Server: d01-test.uuq.pl (2/200 Accounts) | Domain: 1741969814.d01-test.uuq.pl | Username: befDzX | Password: GEpChGju | Status: Active

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

API Connection status: ✔ API Connection OK

Container Log:

```

INFO: Formatting 1st pool, 1 set(s), 1 drives per set.
INFO: WARNING: Host local has more than 0 drives of set. A host failure will result in data becoming unavailable.
MinIO Object Storage Server
Copyright: 2015-2025 MinIO, Inc.
License: GNU AGPLv3 - https://www.gnu.org/licenses/agpl-3.0.html
Version: RELEASE.2025-02-28T09-55-16Z (go1.23.6 linux/amd64)

API: http://172.26.0.4:9000 http://127.0.0.1:9000
WebUI: https://console.1741969814.d01-test.uuq.pl

Docs: https://docs.min.io
INFO:
You are running an older version of MinIO released 1 week before the latest release
Update: Run `mc admin update ALIAS`

```

App Metrics:

Version	RELEASE.2025-02-28T09-55-16Z
Users	

Metric Statistics

	Metric	Enabled	Current Usage	Last Update			
Metric Statistics	Traffic IN (GB)	✓	0.00 GB	4 hours ago			
	Traffic OUT (GB)	✓	0.01 GB	4 hours ago			
				Refresh Now			
Address	Reg Date	Name	Pricing	Status	Next Due Date		

Client Area

Home screen

Docker MinIO 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 MinIO"**: 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.


Go to MinIO


User manual

 Status:	running
 CPU usage:	1 CPU 99.63%
 Memory usage:	120.7MiB / 1GiB 11.79% 88.21000000000001%
 Disk usage:	128K / 974M 99%

MINIO

Web Console: <https://console.1741969814.d01-test.uuq.pl/>

 REST API:	https://1741969814.d01-test.uuq.pl	
 Username:	befDzX	
 Password:	<input type="password" value="....."/>	

 [Change Administrator Password](#)

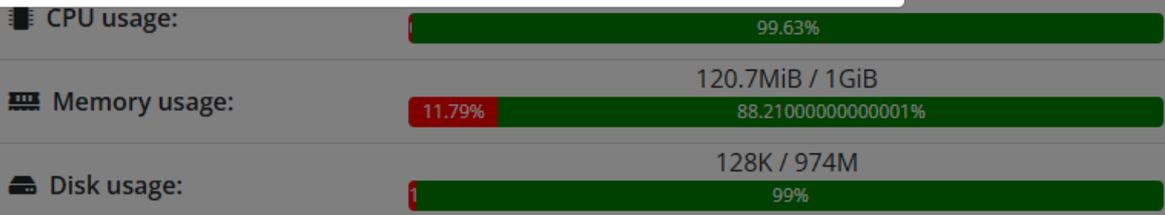
 Version:	RELEASE.2025-02-28T09-55-16Z
 Administrator:	befDzX
 Users:	

Change Administrator Password



Generate

Save



MINIO

Web Console: <https://console.1741969814.d01-test.uuq.pl/>

REST API: <https://1741969814.d01-test.uuq.pl>

Username: befDzX

Password:

Change Administrator Password

Version: RELEASE.2025-02-28T09-55-16Z

Administrator: befDzX

Users:

IP Access Control

Docker MinIO 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.

The screenshot shows the 'IP Access Control' configuration page in a WHMCS client area. On the left is a navigation menu with 'Overview', 'Information', 'Restrict by IP' (highlighted with a red box), and 'Reinstall'. The main content area has the title 'IP Access Control' and a note: 'If IP is not specified, access is not limited'. There are two sections for configuration: 'Web Console' with the URL 'https://console.1741969814.d01-test.uuq.pl' and 'REST API' with the URL 'https://1741969814.d01-test.uuq.pl'. Each section has a text input field labeled 'Enter allowed IPs, one per line'. At the bottom, there is a 'Save' button and a footer that reads 'Powered by WHMCompleteSolution'.

Reinstall

Docker MinIO 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**.



- ★ Overview
- i Information
- 🛡️ Restrict by IP
- ↻ Reinstall**
- 🔧 Actions
- ↑ Upgrade/Downgrade
- 🚫 Request Cancellation

You are in the area of reinstalling service.
You must be aware of what you will do here.
Reinstalling the service, completely remove all data.
To protect against accidental reinstallation.
Please enter the word: **reinstall** In capital letters.

↻ Reinstall

Metrics

Docker MinIO 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.



Docker MinIO 10
Docker MinIO

ACTIVE

Registration Date
Friday, March 7th, 2025

Recurring Amount
\$0.00

Billing Cycle
Free Account

Next Due Date
-

Payment Method
PayPal

[Manage](#) [Metrics](#)

This product has usage-based billing charges in addition to the base price. Usage metrics and their pricing information are displayed below.

Metric	Current Usage	Pricing	Last Update
Traffic IN (GB)	1.21 GB	\$1.00 / GB	2 hours ago
Traffic OUT (GB)	0.16 GB	\$1.00 / GB	2 hours ago