

Setting up n8n workflow

Docker n8n module **WHMCS**

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

Overview

The **Docker n8n 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

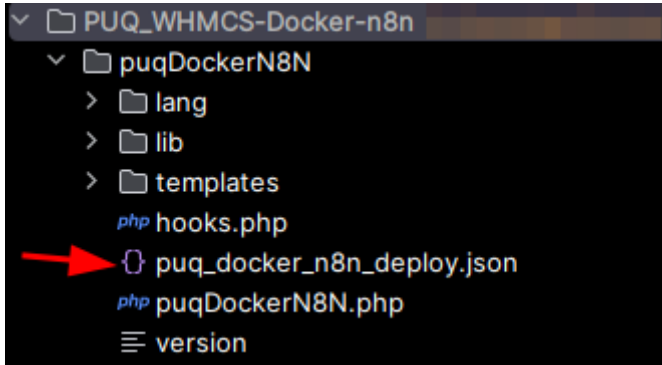
You have two options:

Option 1: Use the Latest Version from the n8n Marketplace

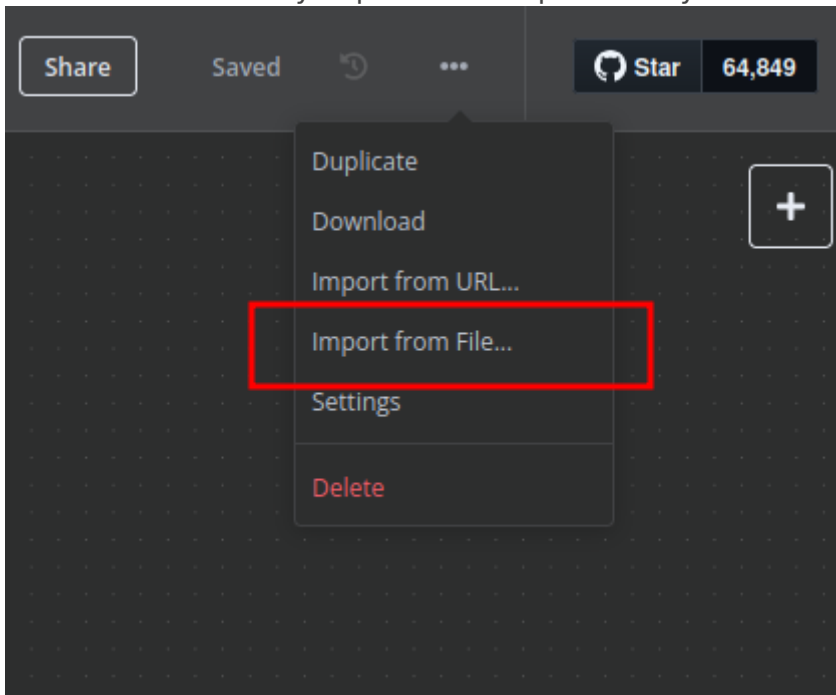
- The latest workflow templates for our modules are available on the official n8n marketplace.
- Visit our profile to access all available templates: [PUQcloud on n8n](#)

Option 2: Manual Installation

- Each module version comes with a workflow template file.



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

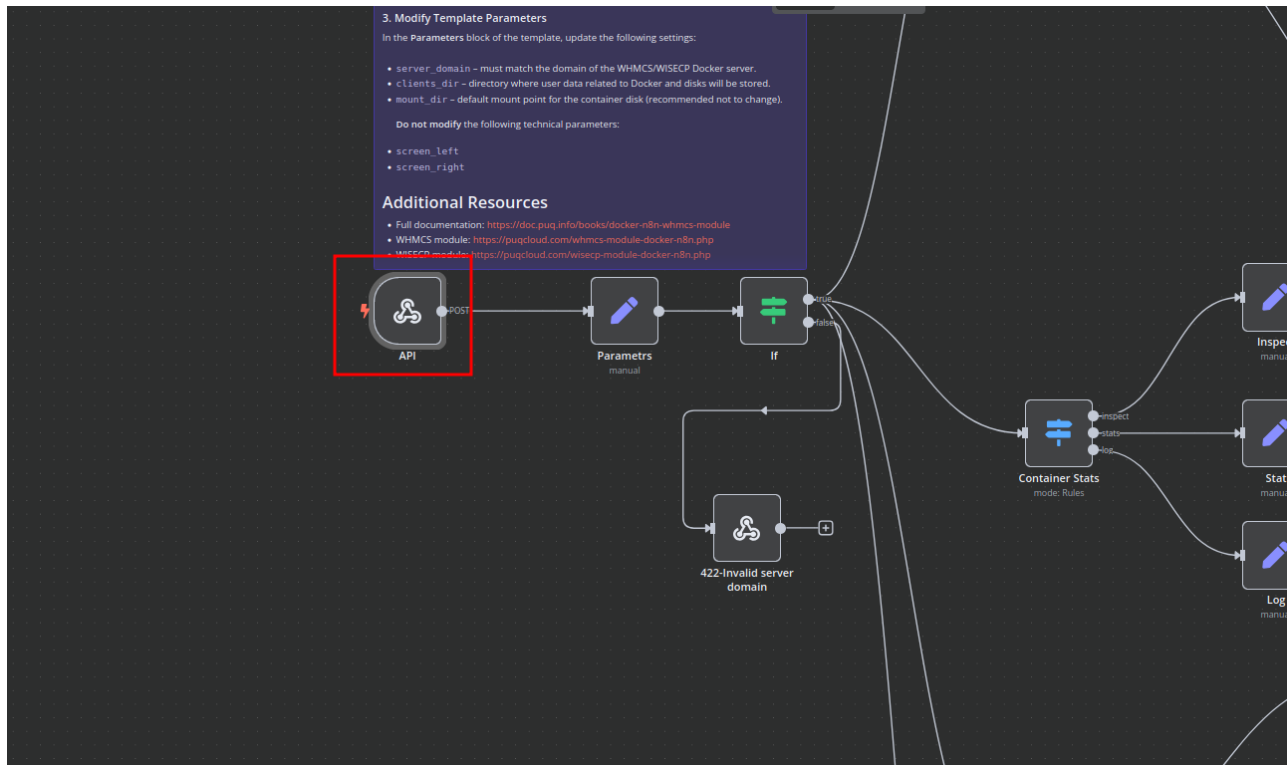


n8n Workflow API Backend Setup for WHMCS/WISECP

Configure API Webhook and SSH Access

- Create a **Basic Auth Credential**

for the Webhook API Block in n8n.



← Back to canvas



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](#)

When will this node trigger my flow?



API



Listen for test event

Parameters

Settings

Docs

Webhook URLs

Test URL

Production URL

POST

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

HTTP Methods

POST



Path

docker-n8n

Authentication

Basic Auth

Credential for Basic Auth

Incom API User



Incom API User

Basic Auth



+ Create new credential

Now you respond. [More details](#)

Options

No properties

Add option



API credential
Basic Auth

Save

Connection

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

Sharing

Details

User

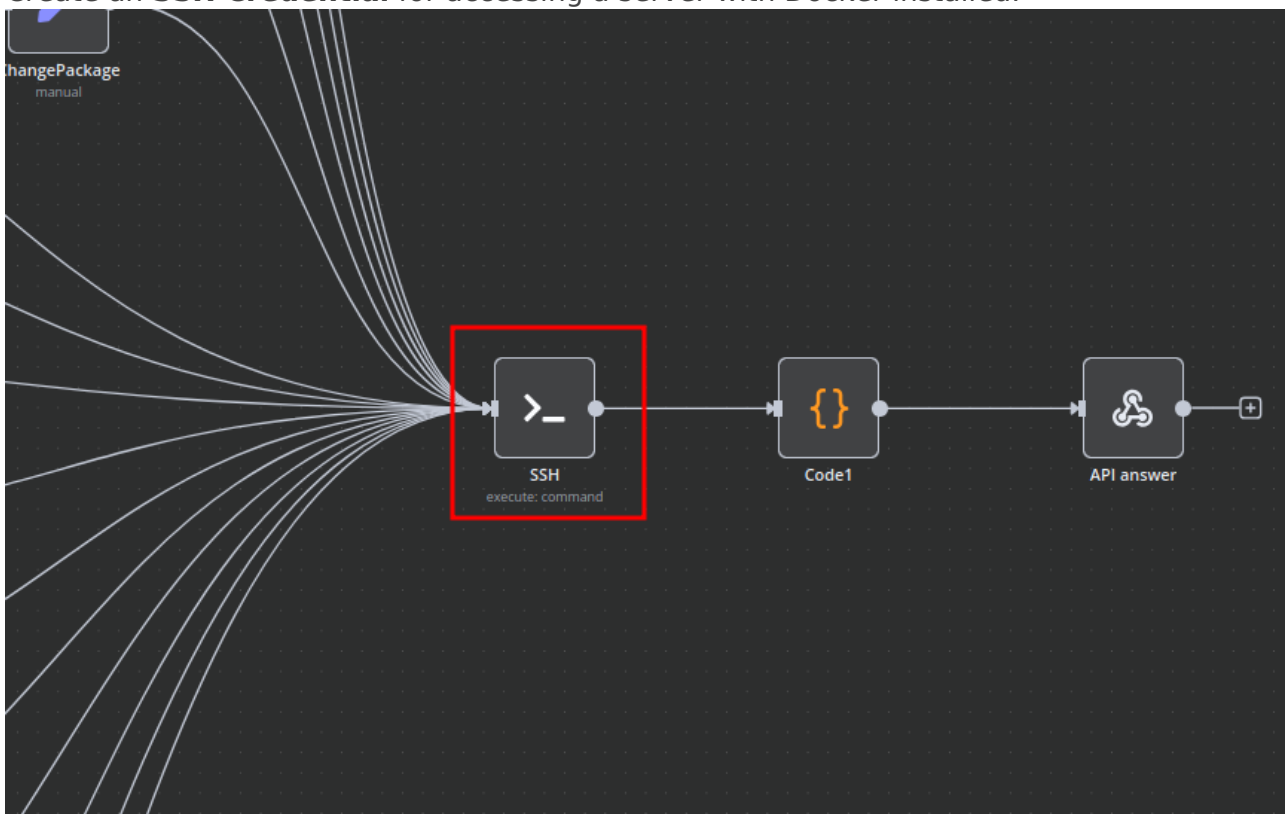
test

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 /

>

d01-test.uuq.pl-puq

SSH Password

×

Connection

Sharing

Details

✔ Connection tested successfully

Retry

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

Host *

d01-test.uuq.pl

Port *

22

Username

puq

Password

.....

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

Modify Template Parameters

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

3. Modify Template Parameters

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

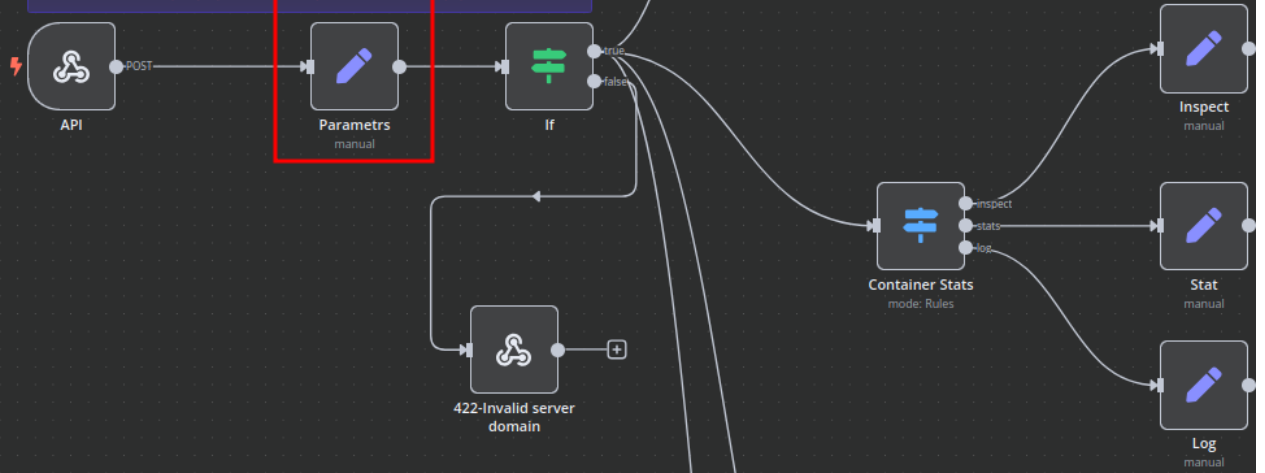
- `server_domain` - must match the domain of the WHMCS/WISECP Docker server.
- `clients_dir` - directory where user data related to Docker and disks will be stored.
- `mount_dir` - default mount point for the container disk (recommended not to change).


Do not modify the following technical parameters:

- `screen_left`
- `screen_right`

Additional Resources


- Full documentation: <https://doc.puq.info/books/docker-n8n-whmcs-module>
- WHMCS module: <https://puqcloud.com/whmcs-module-docker-n8n.php>
- WISECP module: <https://puqcloud.com/wiseep-module-docker-n8n.php>




 **Parameters**

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).

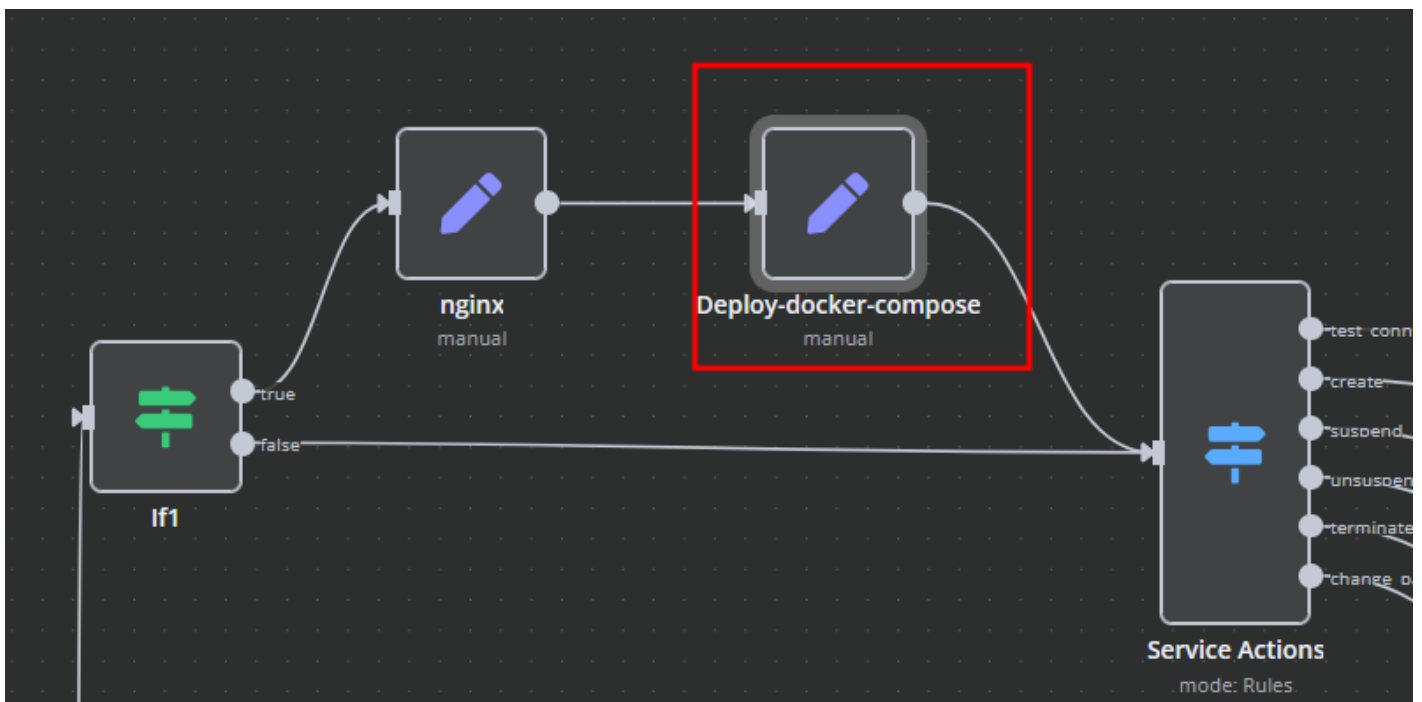
Do not modify the following technical parameters:

- `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](#)

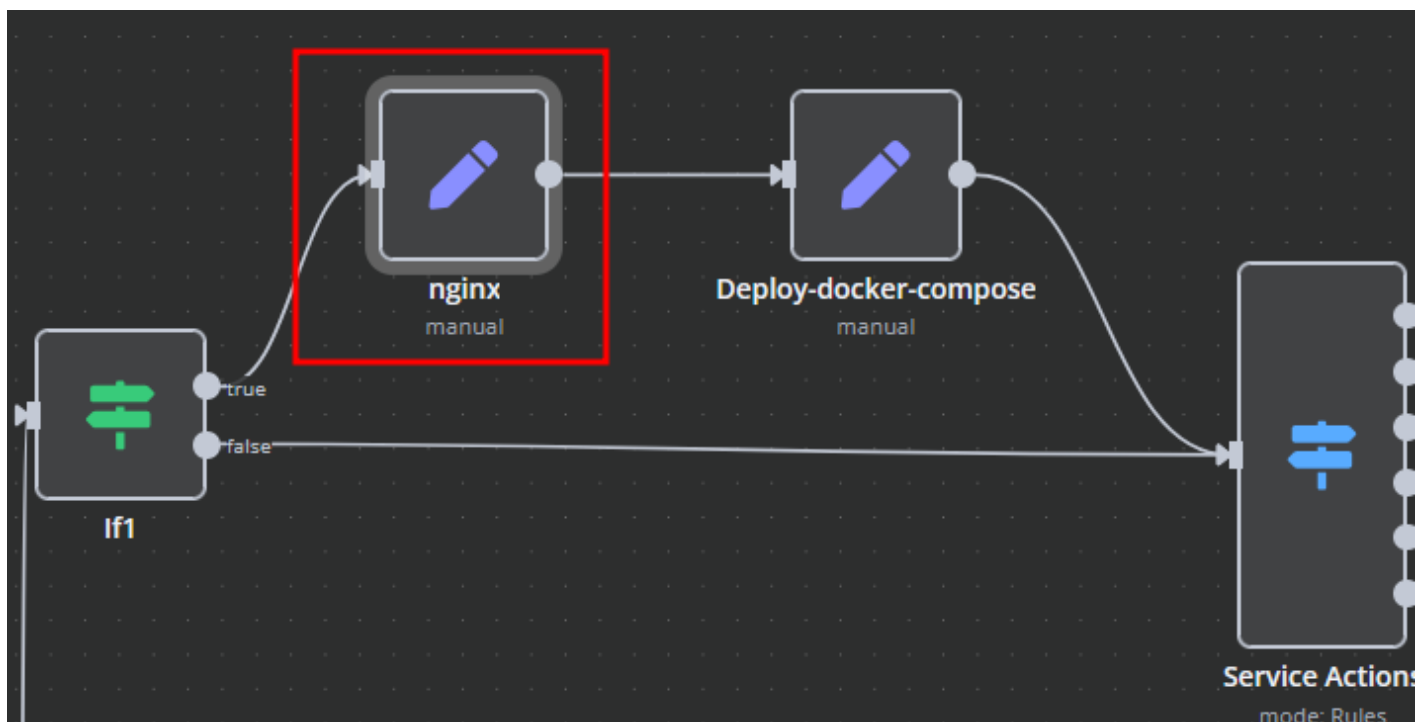
```
version: "3"
services:
  n8n-{{ $('API').item.json.body.domain }}:
    image: n8nio/n8n
    restart: unless-stopped
    container_name: {{ $('API').item.json.body.domain }}
    environment:
      - VIRTUAL_HOST={{ $('API').item.json.body.domain }}
      - LETSENCRYPT_HOST={{ $('API').item.json.body.domain }}
      - WEBHOOK_URL=https://{{ $('API').item.json.body.domain }}
    volumes:
      - {{ $('Params').item.json.mount_dir }}/{{ $('API').item.json.body.domain }}:/home/
node/.n8n
  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.





Test step

ParametersSettingsDocs

Mode

Manual Mapping

Fields to Set

mainString

=

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_locationString

=

Custom header

Custom header

Drag input fields here or Add Field

Include Other Input Fields

Options

No properties

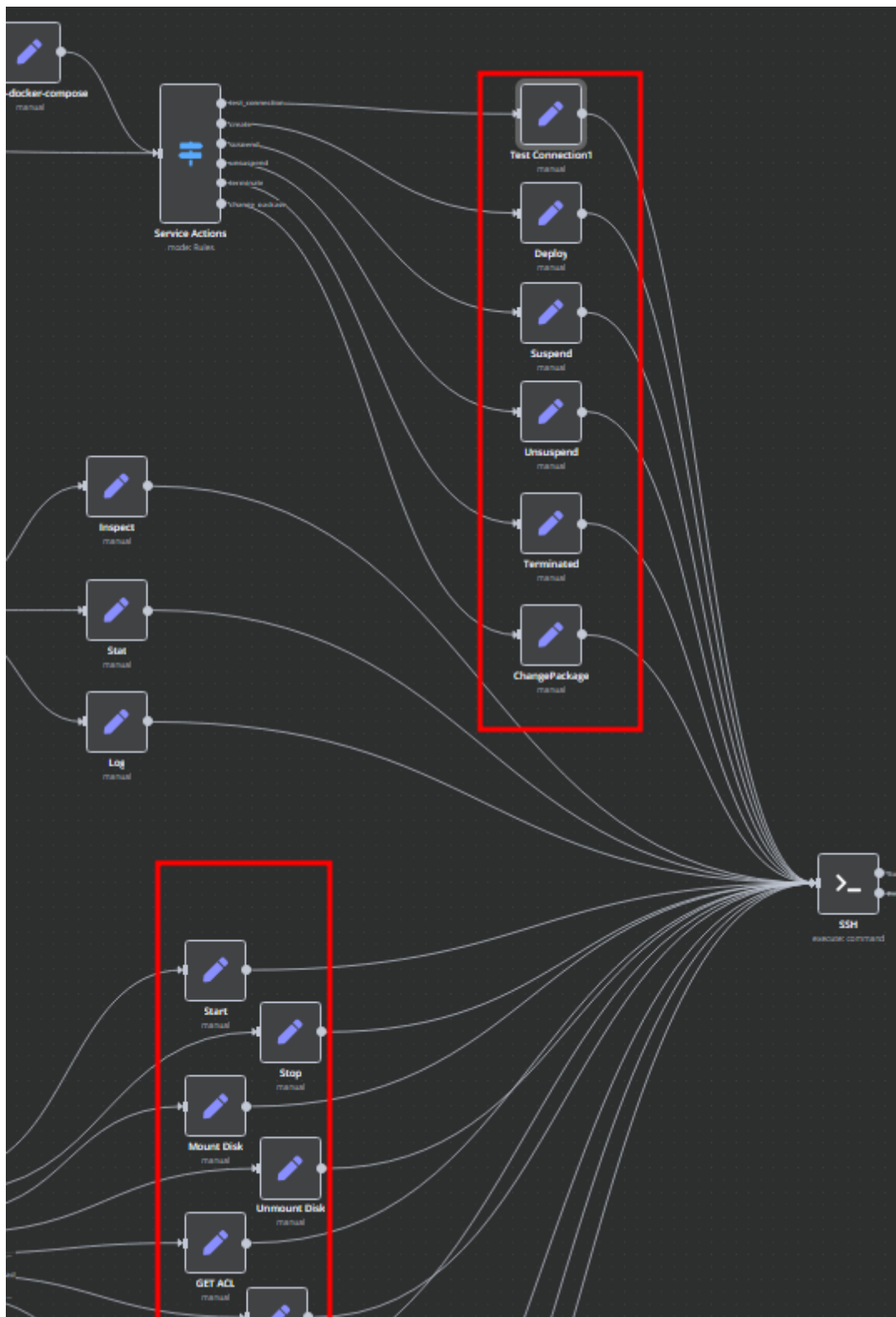
Add option

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.



Revision #8

Created 5 March 2025 04:12:13 by Ruslan

Updated 21 March 2025 13:39:44 by Yuliia Noha