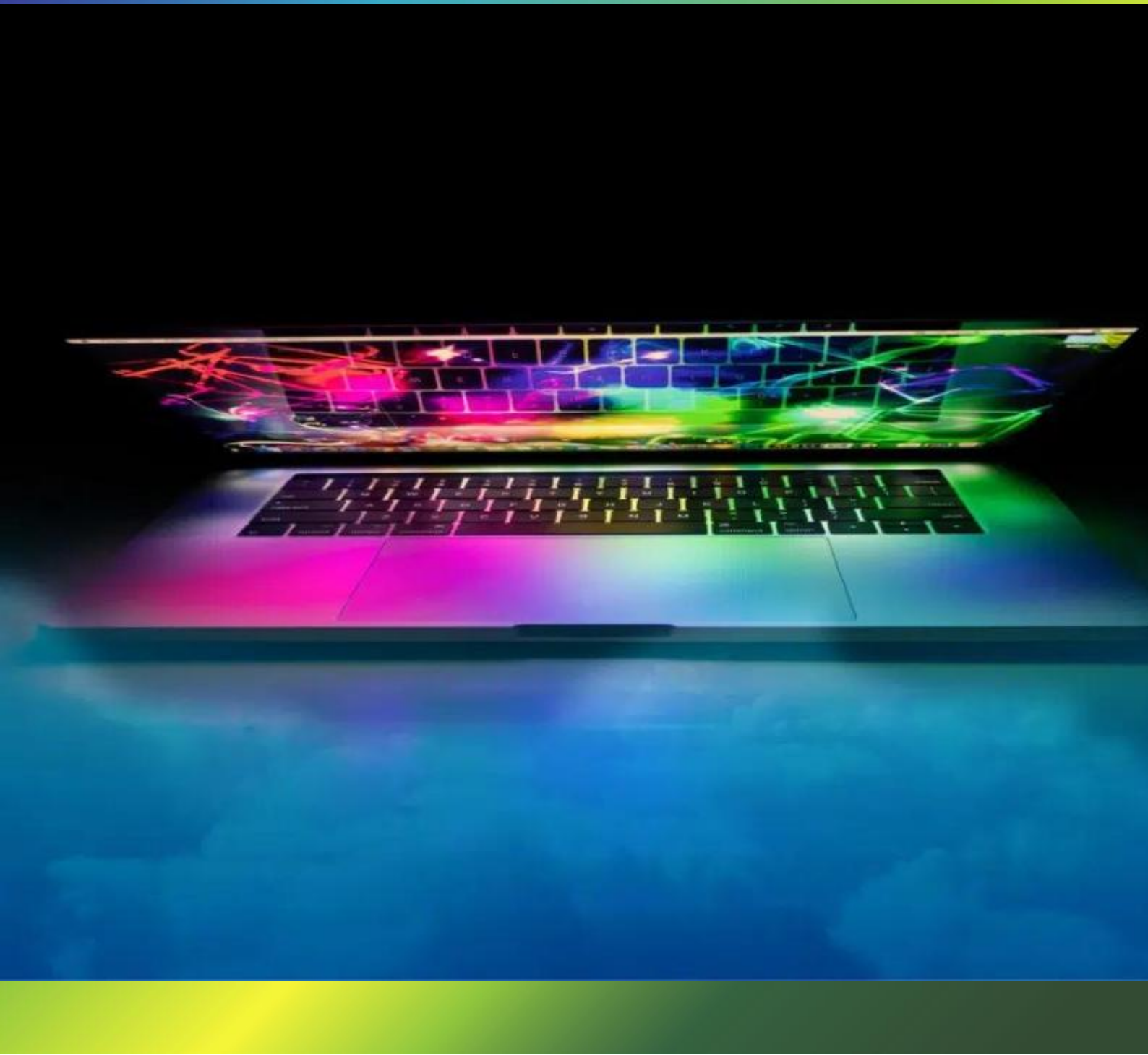




Automatic Deployment of RDS Servers and Monitoring



Index

1. Introduction	4
2. Configuring RDS Servers with RDS-Automation	6
2.1. Installation Requirements.....	6
2.1.1. Operating System Requirements	6
2.1.2. Recommended Hardware Requirements.....	6
2.1.3. Network Requirements.....	7
2.1.4. Software Requirements	7
2.1.5. Required Permissions and Access	8
2.1.6. Other Additional Requirements	8
2.2. Installing and Configuring the RDS-Automation Agent	9
2.2.1. Installation Path.....	9
2.2.2. Domain Configuration.....	10
2.2.3. Integration with the UDS Enterprise Platform	11
2.2.4. Additional Configuration.....	13
2.3. Files Generated During Installation	15
2.4. Process Flow and Execution of the RDS-Automation Agent	17
2.5. Logging and Execution Tracking System	21
3. Configuring and Operating UDS Monitor	22
3.1. Elements Required for Initial Configuration.....	22
3.2. Preparing the RDS Template with the RDS-Automation Agent	23
3.3. UDS Monitor Configuration Wizard	23
3.3.1. Access to the UDS Server	23
3.3.2. Selecting the Server Group, Provider, and Base Service	25
3.3.3. Configuring Monitoring and Scaling Policies.....	26
3.3.4. Configuring the Database and Logging System	29
3.3.5. Generating the Configuration Parameters File	30
3.4. Activating the UDS Monitor Container.....	31
3.5. Configuring the rds_monitor.ini File	34
3.6. Starting the Monitoring Service.....	35
3.7. UDS Monitor Operation	36
3.7.1. Automatic Creation of the Service Pool.....	36

3.7.2.	Initial Server Deployment	37
3.7.3.	Automatic Server Registration.....	38
3.7.4.	Monitoring and Automatic Server Expansion	38
3.7.5.	Automatic Replacement of Cache Servers	39
3.7.6.	Publishing New Versions.....	40
3.7.7.	Deleting a Service Pool.....	42
3.7.8.	Graphical Monitoring of RDS Servers	43
3.8.	Modifying Values in the rds_monitor.ini File	48
3.8.1.	Modifying the Number of Initial Servers	49
4.	Troubleshooting Common Issues.....	55
	About Virtual Cable.....	56

1. Introduction

This document describes in detail how two new features introduced in UDS Enterprise version 4.0 work. These features provide advanced automation capabilities for the dynamic deployment of application servers or RDS sessions, as well as continuous performance monitoring. Their main purpose is to enable the platform to automatically adjust available capacity according to users' actual resource consumption, thereby ensuring a more efficient, scalable service that adapts to demand.

These capabilities represent a significant advance in the automated management of RDS environments with UDS Enterprise. They reduce manual intervention, optimize resource use, and ensure that the system can respond proactively to increased loads or temporary capacity requirements. The solution is therefore organized into two main components, each with clearly differentiated yet complementary responsibilities:

- RDS Server Configuration

The RDS-Automation Agent comprises the components responsible for automating all tasks related to adding, configuring, and managing RDS servers in UDS Enterprise. The agent acts as an intermediary between the platform and the servers added to the environment, making the onboarding process fully automatic and standardized.

Its functions include preparing the server, applying specific settings, integrating it with UDS Enterprise services, and verifying that it is correctly registered and operational. The agent enables the platform to expand its capacity quickly and without manual intervention, while maintaining consistency across all deployed servers.

The agent can be used independently in environments where RDS servers are monitored with UDS Monitor and in environments without such monitoring. In the latter case, a manual task must be performed on the selected virtualization platform to create new servers. This involves manually cloning the template as many times as needed to provide the required number of RDS servers in the UDS Enterprise server group.

- Utility Machine with UDS Monitor:

The Utility Machine is a central component of the advanced automation process. It includes UDS Monitor software running in a dedicated Docker container. This service plays a key role in dynamic resource management by combining automated deployment with monitoring and analysis capabilities.

UDS Monitor can automatically deploy new RDS servers in a UDS Enterprise server group from a template hosted on a virtualization platform supported by UDS. This ensures that all generated servers have a consistent configuration aligned with the environment's requirements. It also continuously monitors resource consumption on existing servers, such as CPU, memory, and the number of active sessions, and evaluates whether these values exceed the thresholds defined in the scaling policies configured by the administrator.

When the system load reaches a level that could compromise performance or service availability, UDS Monitor automatically starts creating new RDS servers, ensuring that the platform always has enough capacity to meet demand. The process is transparent to end users and requires no manual intervention, helping to improve operational efficiency and system responsiveness.

2. Configuring RDS Servers with RDS-Automation

2.1. Installation Requirements

To ensure correct installation and stable operation of the RDS-Automation Agent, the machine on which it runs must meet a number of technical prerequisites. These requirements cover the operating system, hardware, network, required software, and minimum permissions. Meeting them is essential to ensure that the agent can correctly deploy, configure, and register RDS servers on the platform and to prevent failures during the automation process.

2.1.1. Operating System Requirements

The automation agent is designed to run exclusively on Windows Server operating systems that fully support Remote Desktop Services roles and features. The supported versions are:

- Windows Server 2016
- Windows Server 2019
- Windows Server 2022
- Windows Server 2025

These versions provide the APIs, services, and components required to install, configure, and operate the RDS environment, as well as compatibility with the modules used by the script.

2.1.2. Recommended Hardware Requirements

RDS server performance and correct agent execution depend directly on the resources assigned to the machine. To provide users with a suitable experience and allow the automation process to run without limitations, the following minimum and recommended requirements apply:

- **Processor: 64-bit CPU running at 1.4 GHz or faster and capable of supporting multi-user workloads.**
- **RAM:**
 - Minimum: 2 GB per concurrent user.
 - Recommended: 4 GB or more per concurrent user, especially in environments with resource-intensive applications.
- **Disk space:**
 - Minimum: 32 GB for the operating system and essential components.

- Recommended: 45 GB or more to accommodate updates, temporary storage, logs, and future expansion.
- **Network adapter: Gigabit Ethernet interface (1 Gbps), required to ensure suitable performance when communicating with the domain, UDS, and other dependent services.**

These values ensure that the server can support both the user workload and the agent's automated tasks without performance degradation.

2.1.3. Network Requirements

Correct operation of the agent depends on stable connectivity and proper name resolution within the environment. The machine must meet the following requirements:

- **Domain connectivity: The machine must be able to communicate with the domain that the RDS server will join, including resolving domain controllers and associated services.**
- **Connectivity to UDS servers: The machine must be able to communicate with the UDS server over TCP ports 443 and 43910.**
- **IP addressing through DHCP: The machine must obtain its IP configuration from an operational DHCP server, ensuring an assignment consistent with the infrastructure.**
- **DNS resolution: The server must have DNS configured correctly to resolve internal and external names as required. Incorrect resolution can prevent the server from joining the domain or installing roles.**

A correctly configured network prevents failures when joining the domain, installing roles, and communicating with UDS Enterprise.

2.1.4. Software Requirements

To run the automation script and install roles and features, the machine must have the following components:

- **.NET Framework 4.5 or later, required to run the libraries and modules used by the script.**
- **PowerShell 5.1 or later, required to run the commands and functions that form part of the automation process.**

Both components are essential to ensure compatibility with the agent's functions and the correct execution of automated processes. They are included by default in most supported operating systems.

2.1.5. Required Permissions and Access

The account used to run the script must have sufficient privileges to make system changes. The following permissions are required:

- **Local administrator privileges, required to install roles, modify settings, and perform elevated tasks.**
- **Write access to the Windows registry, because the script may need to create or modify configuration keys.**
- **Permission to install Windows Server roles and features, including Remote Desktop Services components.**
- **Internet access (optional), if additional components or updates must be downloaded during installation.**

Without these permissions, the automation process may fail or remain incomplete.

2.1.6. Other Additional Requirements

The following supplementary requirements must be checked before installation begins:

- **Windows Firewall configured to allow RDP traffic (port 3389 by default), which is essential for remote access and RDS server operation.**
- **GPU compatibility, if graphics acceleration is required. This requires suitable hardware with installed and up-to-date drivers.**
- **The machine must not be joined to a domain before the script is run, because the automation process automatically joins it to the appropriate domain. Joining it beforehand could cause conflicts or prevent the process from completing correctly.**
- **A configured and accessible UDS environment is required, together with administrator credentials for an authenticator. That environment must contain at least one “Managed” server group in which the new RDS servers configured by the agent will be registered.**

Checking all these requirements in advance is essential for correct installation and successful deployment of the automation agent. A thorough check prevents errors during script execution and ensures that the RDS server is properly configured and fully operational.

2.2. Installing and Configuring the RDS-Automation Agent

To begin installing the RDS-Automation Agent, download the latest version of the installer (UDSRDSAAutomationSetup-x.x.x-xxxxxxx.exe) from the official repository:

[RDS-Automation Installation Repository](#)

Name	Last modified	Size
 Parent Directory		-
 RDS-Monitor/	2026-05-27 13:24	-
 UDSRDSAAutomationSetup-4.0.0-20250911.exe	2026-05-27 13:21	228K

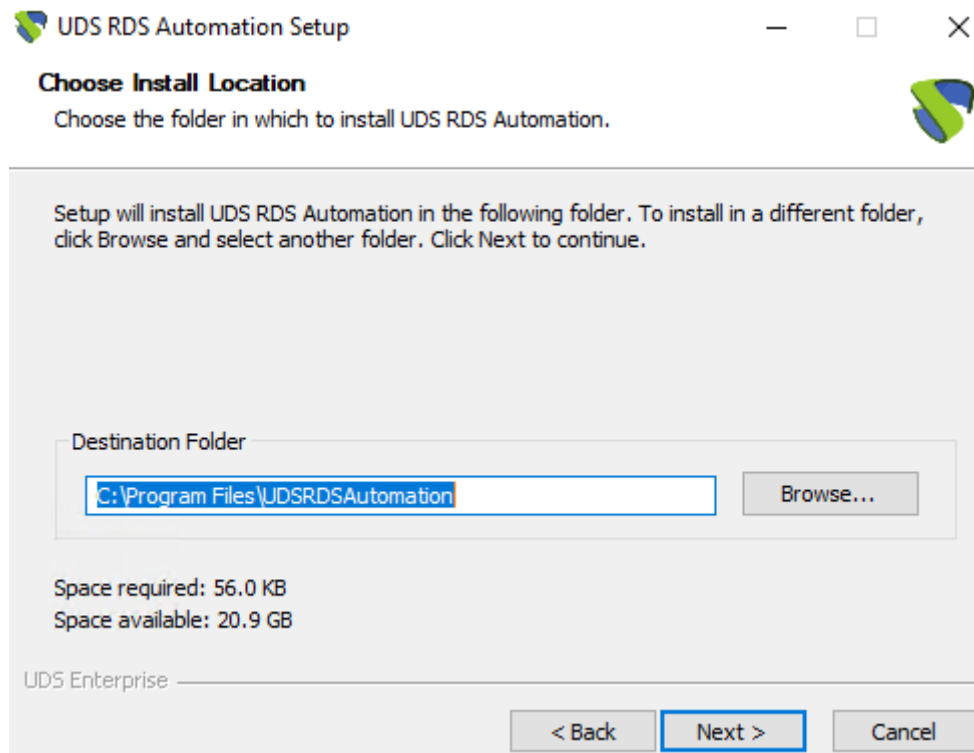
After downloading the installer, run it with administrator privileges to ensure that the wizard can correctly perform all required operating-system operations.

The installation wizard guides the administrator through a series of steps that collect all the information required to automate the subsequent configuration and registration of new RDS servers in specific server groups within the UDS Enterprise platform.

2.2.1. Installation Path

In the first step, the wizard asks where the automation agent components should be installed. By default, installation takes place in the standard Windows “Program Files” directory, although the administrator may select any other available path on the system.

The selected path contains all files required for agent operation, including automation scripts, configuration files, logs, and auxiliary utilities related to the automated deployment process.



2.2.2. Domain Configuration

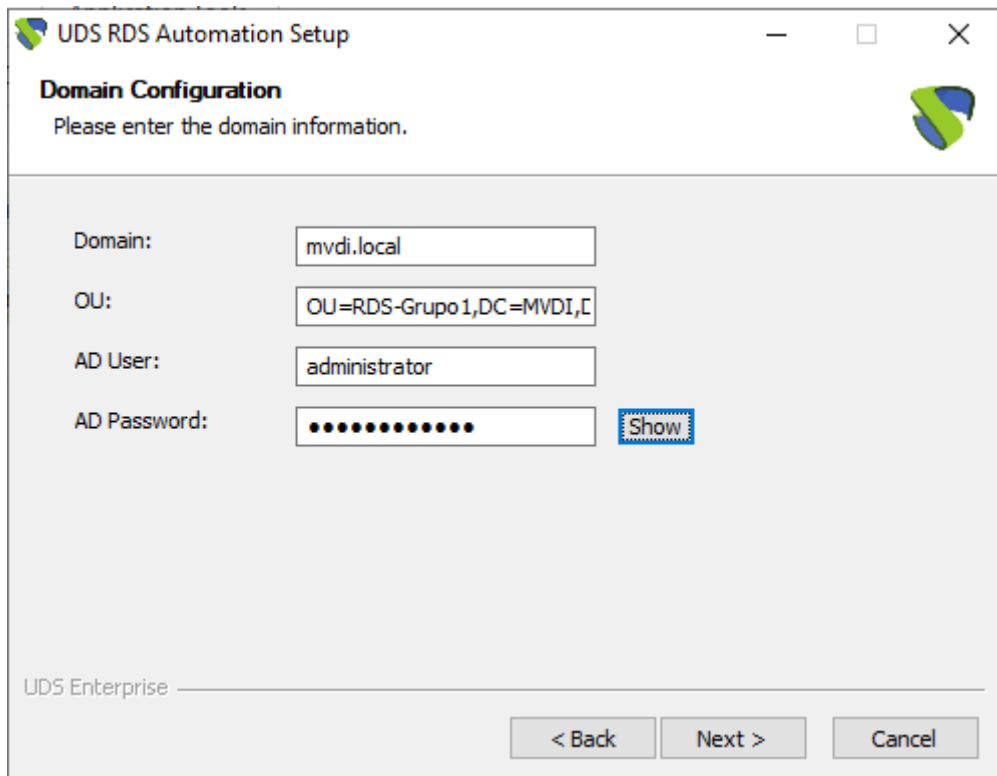
In the next step, enter the details of the Active Directory domain that the new RDS servers generated by the automation process will join automatically.

The required configuration includes:

- Domain name, which the server must be able to resolve.
- User account with sufficient permission to join computers to the domain.
- Password associated with that account.

The wizard also lets you optionally specify an Organizational Unit (OU) in which the new RDS servers will be created and registered automatically. If no OU is specified, the servers are registered in the default OU. This supports structured organization of computers in Active Directory and centralized application of specific Group Policy Objects (GPOs).

Correct configuration of this section is essential to ensure that new servers are integrated automatically into the corporate infrastructure.

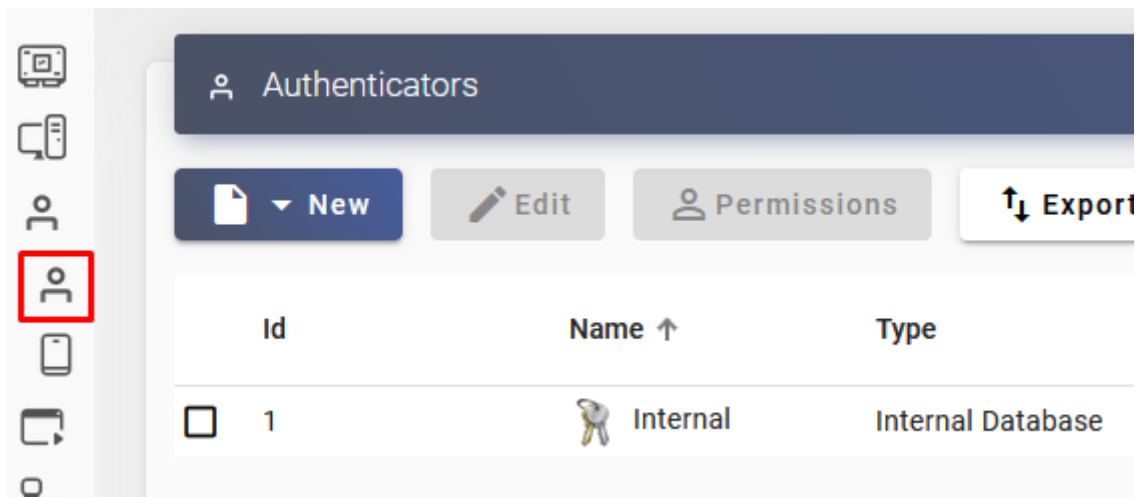


2.2.3. Integration with the UDS Enterprise Platform

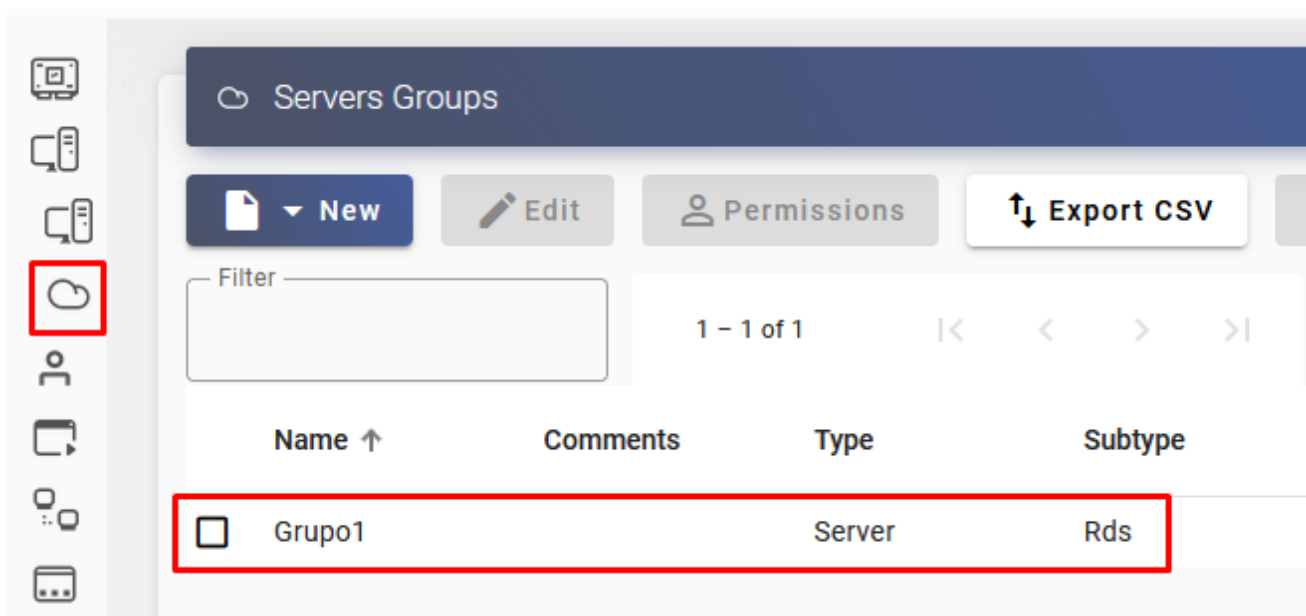
Next, specify the connection and authentication parameters for the UDS Enterprise platform where the generated RDS servers will be registered automatically.

The following parameters are required:

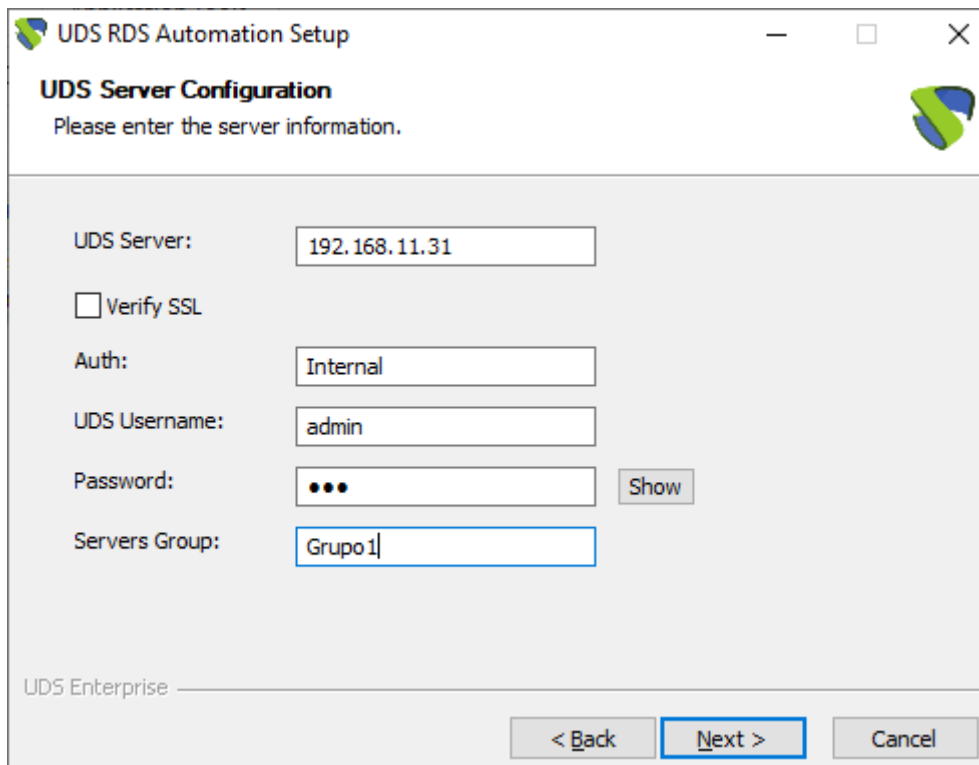
- **UDS Server:** Internal IP address or FQDN of the UDS Enterprise server.
- **Auth:** Exact name of the authenticator previously configured in UDS Enterprise administration that contains the user with the administrative privileges required for automatic server registration. Any authenticator type may be used, provided that the selected user belongs to it and has administrative permission.



- **UDS Username:** User with administrator permissions who belongs to the authenticator specified above.
- **Password:** Password for the specified administrator account.
- **Server Group:** Exact name of the server group previously created in UDS Enterprise to which the new RDS servers will be added automatically. It must be created as "Managed" - "RDS Server".



The authenticator, user, and server group names must exactly match those defined in UDS Enterprise, including capitalization, spaces, and special characters. Any discrepancy may prevent automatic server registration.



The screenshot shows a Windows-style window titled "UDS RDS Automation Setup". Inside, the "UDS Server Configuration" section is active, with the instruction "Please enter the server information." Below this, there are several input fields: "UDS Server:" with the value "192.168.11.31", "Verify SSL" (unchecked checkbox), "Auth:" with the value "Internal", "UDS Username:" with the value "admin", "Password:" (masked with dots) with a "Show" button, and "Servers Group:" with the value "Grupo1". At the bottom, there is a "UDS Enterprise" label and three buttons: "< Back", "Next >" (highlighted with a blue border), and "Cancel".

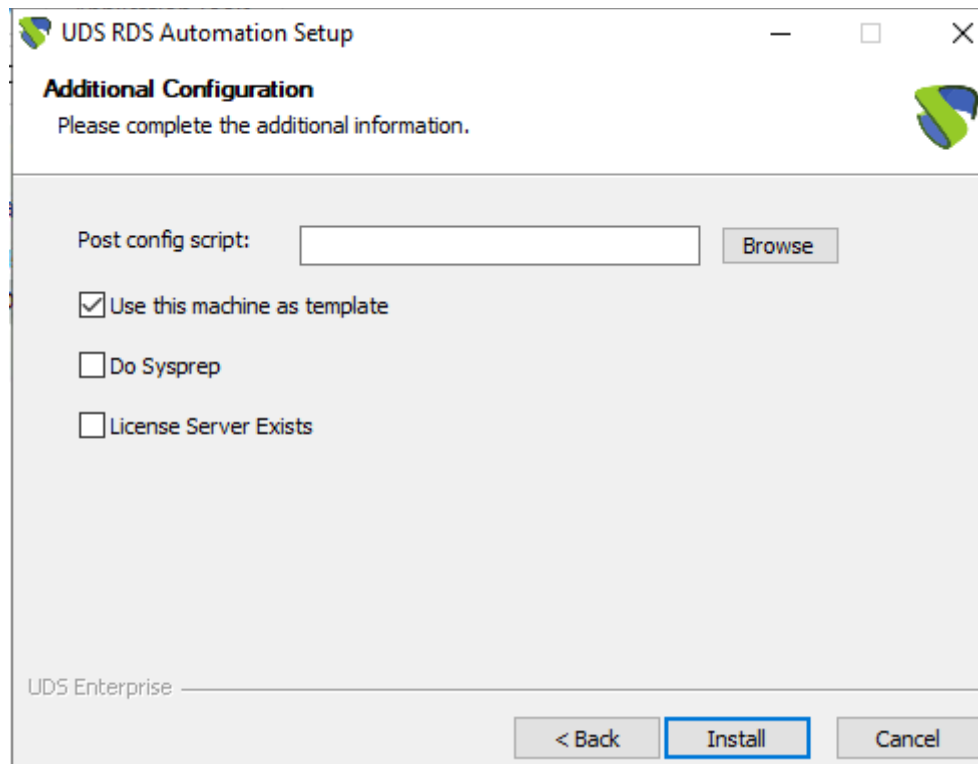
After all details have been entered correctly, the wizard stores this configuration in the internal file subsequently used by the main automation script.

2.2.4. Additional Configuration

The final wizard step provides advanced options for preparing the machine and configuring RDS licensing.

- **Post config script:** Specifies an existing script that will run after all configuration has been applied to the new server.
- **Use this machine as template:** When selected (enabled by default), the script is not applied to this machine, allowing it to be used as a template. The script runs only on clones created either manually or automatically by the UDS Monitor component. The template machine is identified by recording its MAC address so that the configuration script is not run on it.
- **Do Sysprep:** When enabled, Sysprep runs on the new servers. This is not required in most cases. All information required by Sysprep must be specified in the `unattend.xml` file located in the agent installation directory. For example, the new administrator password is defined in that file.
- **License Server Exists:** An RDS license server may also be configured optionally. In most cases, this configuration is applied through the organization's own GPOs. If this feature is enabled, the wizard requests additional information:

- FQDN of the RDS license server.
- Licensing mode.



After all the preceding steps are complete and installation is confirmed, the wizard automatically generates the config.json file, which contains all the configuration required to run the automated process.

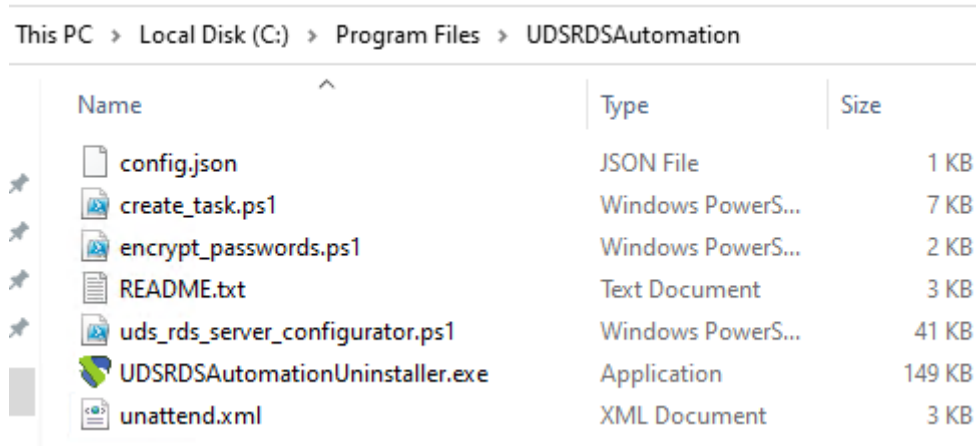
The installer then runs an auxiliary PowerShell script that creates a scheduled task in Windows. This task ensures that the main script runs automatically after each system restart, allowing the process to continue after operations that require the server to be restarted.

Once this phase is complete, no further administrator intervention is required on the server, except to install applications that must be present on the template machine and consequently on the new deployed servers.

NOTE: As with any other template used in a UDS Enterprise environment, the template should be powered off when deployments or clones are created, whether manually or through the UDS Monitor machine.

2.3. Files Generated During Installation

The RDS-Automation script installation directory contains various files required for the automation system to operate.



This PC > Local Disk (C:) > Program Files > UDSRDSAAutomation			
Name	Type	Size	
config.json	JSON File	1 KB	
create_task.ps1	Windows PowerS...	7 KB	
encrypt_passwords.ps1	Windows PowerS...	2 KB	
README.txt	Text Document	3 KB	
uds_rds_server_configurator.ps1	Windows PowerS...	41 KB	
UDSRDSAAutomationUninstaller.exe	Application	149 KB	
unattend.xml	XML Document	3 KB	

The most relevant files include:

- uds_rds_server_configurator.ps1: Main script responsible for executing the entire automation workflow.
- config.json: File containing all configuration parameters used by the process.

The config.json file can be reviewed manually with any text editor to verify that all parameters configured during installation are correct or to modify them.

If passwords are changed manually, the encrypt_passwords script must then be run to automatically encrypt the credentials stored in the configuration file.

This mechanism protects the sensitive information used during the automated process.

```

config.json - Notepad
File Edit Format View Help
{
  "domain": "mvgi.local",
  "ou": "OU=RDS-Grupo1,DC=MVDI,DC=local",
  "adUser": "administrator",
  "adPassword": "____76492d1116743f0423413b16050a",
  "udsServer": "192.168.11.31",
  "checkSSL": false,
  "auth": "Internal",
  "udsUsername": "admin",
  "udsPassword": "____76492d1116743f0423413b16050",
  "restUrl": "https://192.168.11.31/uds/rest/",
  "serversGroup": "Grupo1",
  "postRunScript": "",
  "isTemplate": true,
  "doSysprep": false,
  "useLicenseServer": false,
  "licenseServer": "",
  "licenseMode": ""
}

```

- unattend.xml: Used during Sysprep to automate multiple initial Windows operating-system settings.

This file is used only when Sysprep was explicitly enabled for the machine during installation.

All settings in this file are fully customizable and can be adapted to the specific requirements of each environment.

The file includes settings for preparing the operating system automatically, including:

- Retaining device installations while the system is generalized.
- Automatically setting the time zone to Romance Standard Time, corresponding to Central European time.

The file also automates aspects of the initial Windows experience:

- Configuring the local administrator password.
- Skipping the Out-of-Box Experience (OOBE).
- Hiding the End-User License Agreement (EULA).

These settings reduce manual intervention and accelerate the automated deployment process.

During the system's first sign-in, the unattend.xml file automatically runs the `uds_rds_server_configurator.ps1` script.

This setting is critical and strictly required to ensure that the automated process continues after Sysprep and system restarts.

Without this automation, the process could not continue correctly with the subsequent configuration phases.

The file also allows initial system customization parameters to be defined, including:

- Owner name.
- Registered organization.
- Initial computer name.
- Workgroup.

All these values are fully customizable and can be changed to meet the environment's requirements.

In general, unattend.xml fully automates the initial preparation of the operating system, eliminating manual tasks and ensuring consistent configuration across all deployed servers.

Although the supplied file can be used directly as a functional reference, it can also be extended or modified to add further settings, provided that the main script continues to run automatically when Sysprep finishes.

2.4. Process Flow and Execution of the RDS-Automation Agent

After the agent has been installed on the template, it runs whenever a clone of that machine is created, either manually from the virtualization platform or automatically by the UDS Monitor component. Clones should always be created while the machine is powered off. The agent follows the workflow below.

Initial Preparation

This initial phase performs the prerequisite checks and operations required to ensure that the environment is in a valid state before the automated process continues.

The main actions include:

- Running Sysprep if it was previously configured.
- Checking for changes to the machine's MAC address to prevent potential conflicts caused by cloning.

This stage ensures that the system is ready to begin integration into the corporate infrastructure.

Joining the Domain

In this phase, the server automatically joins the Active Directory domain specified during agent installation, using the supplied credentials.

The script generates and assigns a random name to the RDS server to prevent naming conflicts within the environment.

Installing RDS Roles

After the server has joined the domain and completed the required restart, the script automatically installs all roles and features required for RDS operation.

The installation includes the core RDS components and all dependencies required to ensure that the environment is operational.

The process is carried out using automated PowerShell commands and may require additional restarts, depending on the state of the operating system.

Configuring the RDS Deployment

After the roles are installed, the system automatically configures the RDS environment.

This phase applies parameters related to:

- General deployment configuration.
- Association with the RDS license server.
- Licensing mode configuration.

Creating the RDS Collection

The automated process creates an RDS collection named UDSENTERPRISE. UDS Enterprise subsequently uses this collection to publish and manage remote sessions or vApps.

The collection is automatically configured with the parameters required for integration with the UDS Enterprise platform.

Installing the UDS RDS Server Component

After the RDS role has been fully configured and all required restarts have completed, the “UDS RDS Server” component is downloaded and installed automatically. The script first attempts to download it from the official Internet repository or, if no Internet connection is available, directly from the UDS Server machine.

This component establishes communication between the RDS server and the UDS Enterprise platform, enabling centralized management of sessions, users, and published resources.

The script also automatically configures and publishes the UDS RDS Server in the collection created earlier.

Automatically Registering the RDS Server in UDS Enterprise

After the RDS environment and the UDS RDS Server component have been configured, the server is automatically registered in the server group previously specified in UDS Enterprise.

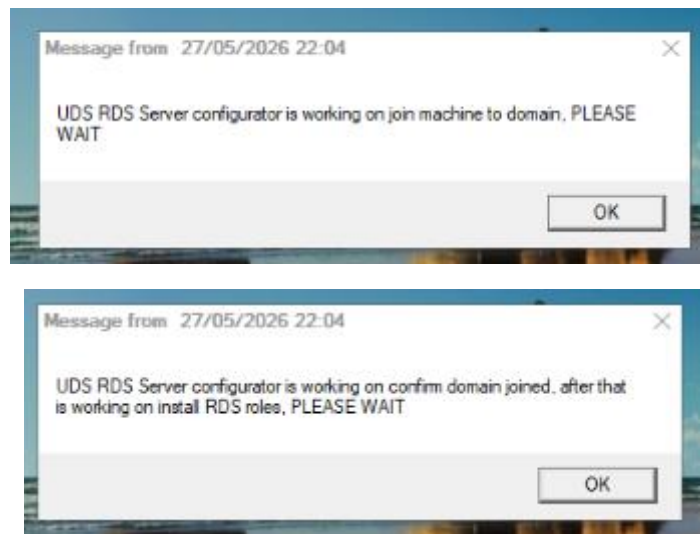
Completing the Process

The final phase of the automated process performs cleanup and final system validation tasks.

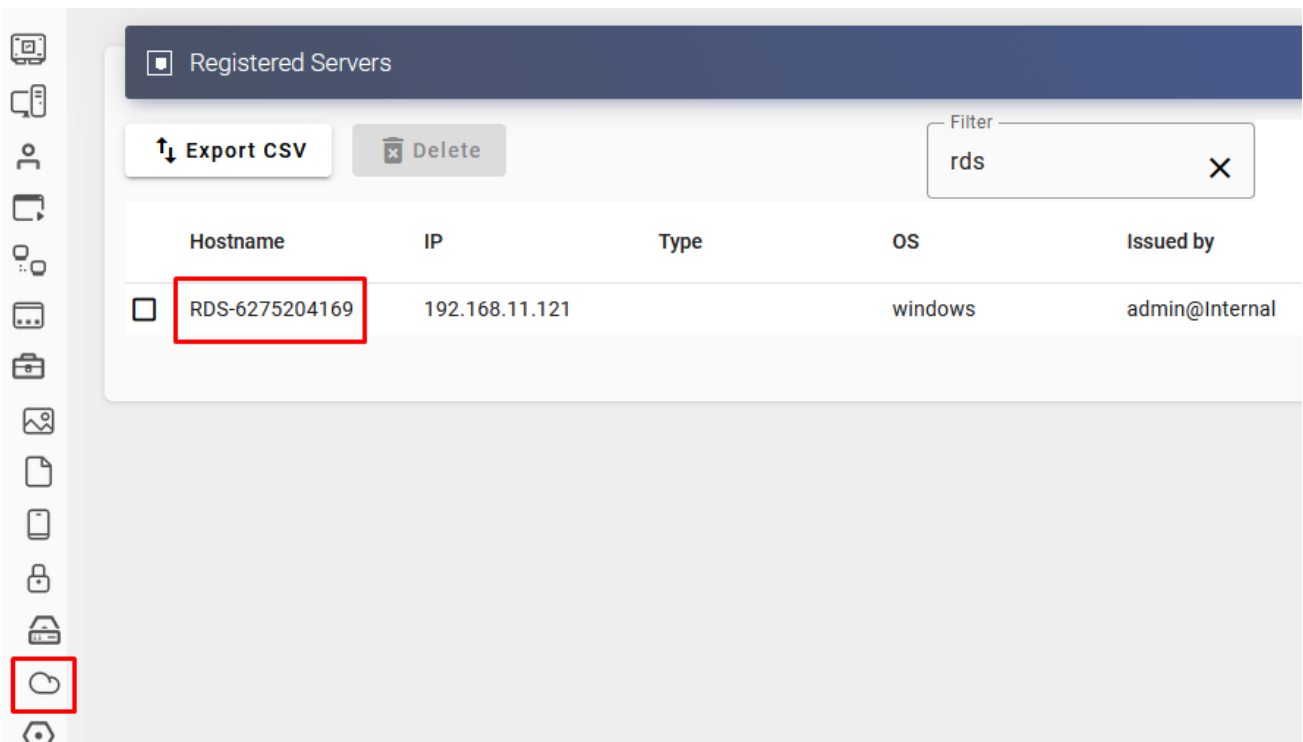
These tasks include:

- Deleting temporary files.
- Removing temporary entries from the Windows registry.
- Performing a final check of installed services and components.
- Performing a final server restart to apply all settings completely.

Throughout the process, the server console displays the tasks being performed. The server also restarts several times:



After the process is complete, the server is fully operational and ready to provide services in the UDS Enterprise environment. The registered server token appears under "Tools" - "Tokens" - "Servers":



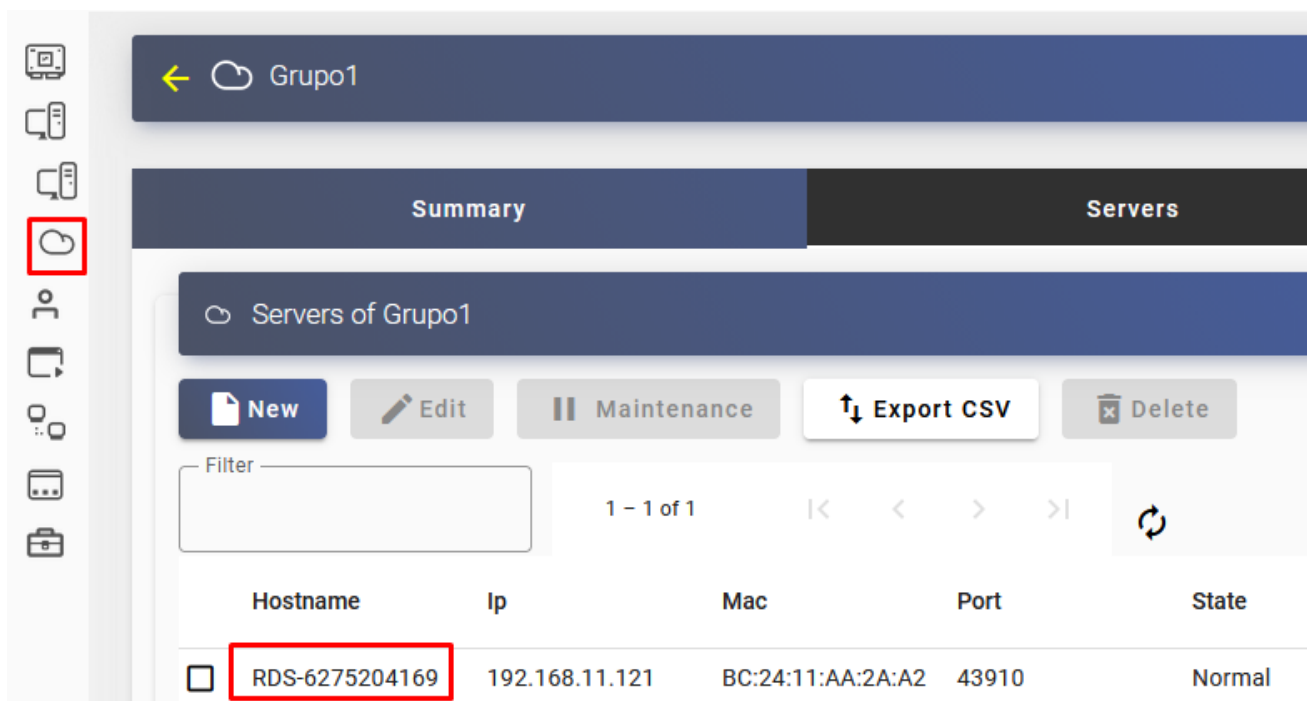
Registered Servers

Export CSV Delete

Filter rds X

	Hostname	IP	Type	OS	Issued by
☐	RDS-6275204169	192.168.11.121		windows	admin@Internal

The server is also created and available in the server group specified in the agent configuration:



← Grupo1

Summary Servers

Servers of Grupo1

New Edit Maintenance Export CSV Delete

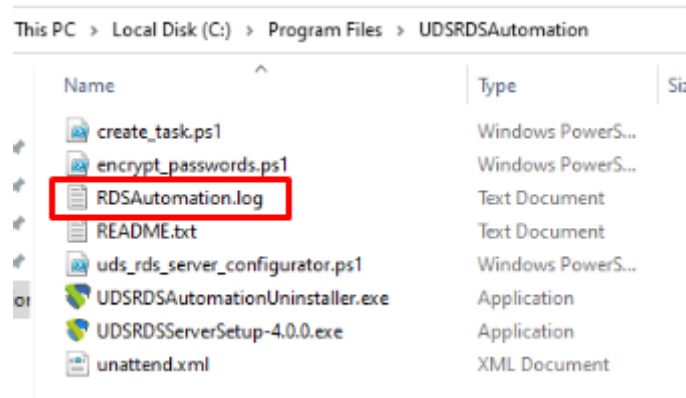
Filter

1 - 1 of 1

	Hostname	Ip	Mac	Port	State
☐	RDS-6275204169	192.168.11.121	BC:24:11:AA:2A:A2	43910	Normal

2.5. Logging and Execution Tracking System

The RDS-Automation Agent includes a log file that records all operations performed during the automation process in detail. The file is stored in the agent installation directory and is named RDSAutomation.log.



This file provides complete tracking of every configuration phase performed on the server, supporting both auditing and troubleshooting.

Logs are mainly classified into two categories:

- **INFO: Informational messages indicating process progress, actions performed, and operations completed successfully.**
- **ERROR: Messages generated when a failure occurs while running commands or performing critical configuration or validation tasks.**

If an error is detected, the process stops automatically and records the event in the log file, making it possible to identify the exact point at which the issue occurred.

One of the main advantages of this mechanism is that, after an error, the administrator can correct the machine configuration or manually modify config.json and then restart the server to resume the automated process.

The scheduled task created during installation remains active until the main script successfully completes all configuration phases. It is deleted from the system automatically only after the entire process has finished successfully.

```
INFO - 2026-05-27 20:46:28 - Obtaining RDS servers
INFO - 2026-05-27 20:46:28 - Obtaining RDS servers in the group Grupo1
INFO - 2026-05-27 20:46:28 - Adding RDS server RDS-6275204169 the group of servers Grupo1
INFO STATE - 2026-05-27 20:46:28 - Change of State Machine to: AllFinished
INFO - 2026-05-27 20:46:28 - All tasks completed successfully, exiting...
INFO - 2026-05-27 20:46:28 - Last state: AllFinished
INFO TASK - 2026-05-27 20:46:28 - Delete scheduled task

END - 2026-05-27 20:46:29 - RDS Automation program End
```

3. Configuring and Operating UDS Monitor

After the RDS-Automation Agent has been configured and the RDS server template has been prepared correctly, UDS Monitor can be deployed and configured on the Utility Machine. As described in the introduction, UDS Monitor provides monitoring and dynamic autoscaling capabilities for a specific server group within the UDS Enterprise infrastructure.

This component continuously monitors the status and load of the RDS servers in the group, evaluates resource-consumption metrics, and automatically expands capacity when thresholds previously configured by the administrator are reached.

In addition to monitoring, UDS Monitor automatically deploys new RDS servers from a template previously prepared with RDS-Automation. This integration ensures that all generated servers have a consistent configuration fully aligned with the policies defined in the environment.

Together, RDS-Automation and UDS Monitor create a fully automated RDS infrastructure that can adapt dynamically to user demand without manual intervention.

3.1. Elements Required for Initial Configuration

Before configuring UDS Monitor, ensure that the following components have been prepared:

- A Windows Server template that meets all requirements for installing the RDS-Automation Agent.
- RDS-Automation Agent installer (UDSRDSAAutomationSetup-x.x.x-xxxxxxx.exe).

[RDS-Automation Repository](#)

- UDS Monitor configuration wizard, available for Windows or Linux.

[UDS Monitor Wizard Repository](#)

- A Utility Machine on which the UDS Monitor Docker container will run. Download the appropriate format for the virtualization environment where it will be deployed:

[VMware Utility](#)

[Hyper-V Utility](#)

[Azure Utility](#)

[Xen/XCP-ng Utility](#)

[RAW Utility](#)

[QCOW2 Utility](#)

[AWS Utility](#)

NOTE: A Utility Machine with its own UDS Monitor configuration must be deployed for each server group.

3.2. Preparing the RDS Template with the RDS-Automation Agent

The first step is to install and configure the RDS-Automation Agent on the Windows Server template by following the procedure detailed in the previous section.

After this process has completed successfully, the template is ready to be used as the cloning source in the autoscaling system managed by UDS Monitor.

All servers generated automatically from this template inherit the behavior configured through RDS-Automation, including:

- Automatic joining of the corporate domain.
- Installation and configuration of the RDS role, including a collection.
- Automatic installation of the UDS RDS Server component.
- Automatic registration in UDS Enterprise.
- Automatic integration into the corresponding server group in UDS Enterprise.

3.3. UDS Monitor Configuration Wizard

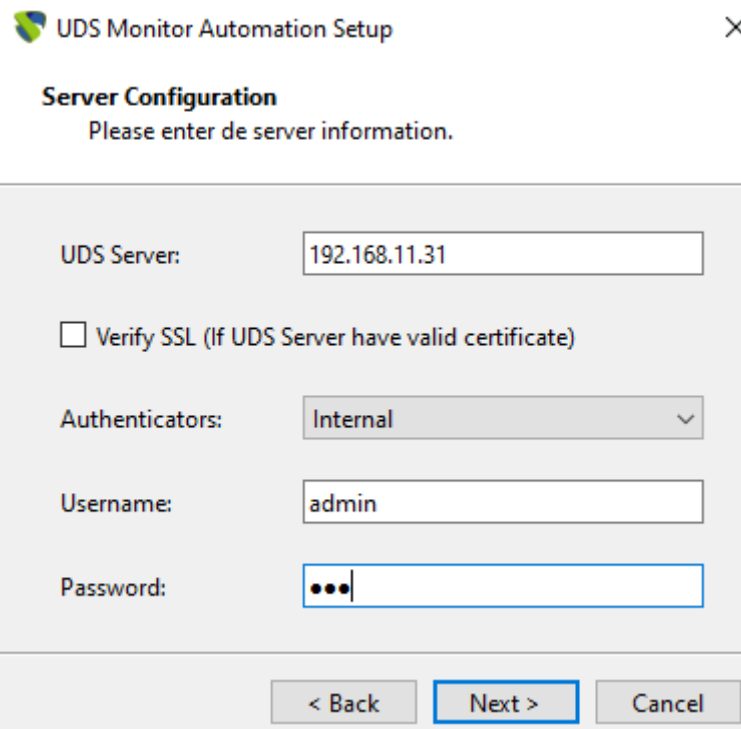
After preparing the template, run the UDS Monitor configuration wizard (uds-monitor-wizard.exe on Windows). The wizard can run on the template itself or on any other Windows or Linux machine with a direct TCP 443 connection to the UDS Enterprise server.

The wizard's main purpose is to generate the configuration file subsequently used by the UDS Monitor component to perform monitoring and autoscaling.

Wizard steps:

3.3.1. Access to the UDS Server

The first wizard step requests the parameters required to connect UDS Monitor to the UDS server.



UDS Monitor Automation Setup [Close]

Server Configuration
Please enter de server information.

UDS Server:

☐ Verify SSL (If UDS Server have valid certificate)

Authenticators:

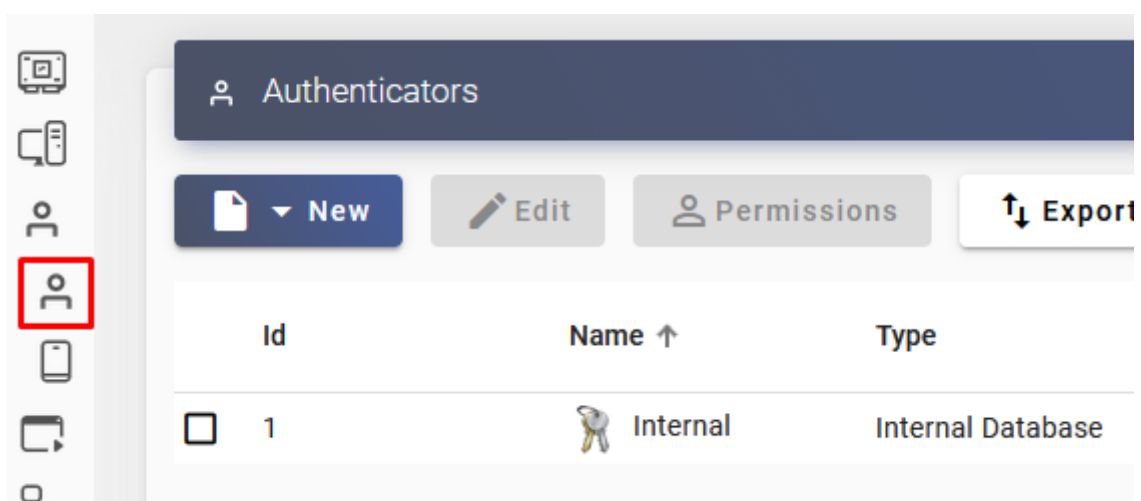
Username:

Password:

< Back Next > Cancel

The required parameters are:

- **UDS Server:** Internal IP address or FQDN of the UDS Enterprise server. If certificate verification is enabled, specify the name and ensure that the UDS server has a valid, trusted certificate.
- **Authenticators:** An authenticator previously configured in UDS Enterprise administration that contains a user with administrative privileges. Any authenticator type may be used, provided that the selected user belongs to it and has administrative permission.



Authenticators		
New Edit Permissions Export		
Id	Name ↑	Type
☐ 1	 Internal	Internal Database

- **UDS Username:** User with administrator permissions who belongs to the authenticator specified above.

- **Password:** Password for the specified administrator account.

NOTE: If the Authenticator field is not populated automatically, there may be no authenticator on the UDS server, TCP port 443 may not be reachable, or there may be a name-resolution or certificate problem if SSL verification is enabled.

UDS Monitor uses these credentials to perform automatic operations on the platform, such as registering servers and creating Service Pools.

3.3.2. Selecting the Server Group, Provider, and Base Service

The next wizard step configures parameters for the server group to be monitored, including the virtualization infrastructure that hosts it, the template machine on which its servers are based, and the initial behavior of the automated environment.

The screenshot shows a dialog box titled "UDS Monitor Automation Setup" with a close button (X) in the top right corner. Below the title bar, the section is labeled "Servers Group Configuration" with a subtitle "Please enter the server group configuration details." The configuration area contains four fields: "Servers Group:" with a dropdown menu showing "Grupo1"; "Service Provider:" with a dropdown menu showing "Proxmox"; "Base Service:" with a dropdown menu showing "W2022RDS-Template"; and "Server limit to create:" with a spinner box set to "4". To the right of the "Server limit to create:" field is the "Initial servers to create:" field with a spinner box set to "2". At the bottom right of the dialog are three buttons: "< Back", "Next >" (which is highlighted with a blue border), and "Cancel".

Specify the following:

- **The server group in which new RDS servers will be registered automatically through RDS-Automation and which will be monitored. This must be the same group specified on the template when the RDS-Automation Agent was configured.**

This item must already exist in UDS administration so that it appears in the drop-down list and can be selected.

- **The service provider used by UDS Enterprise: the virtualization platform on which RDS servers will be generated automatically and where the template with the installed and configured RDS-Automation Agent is located.**

This item must already exist in UDS administration so that it appears in the drop-down list and can be selected.

- **The base service within the service provider, previously created in UDS, to which the template machine with the installed and configured RDS-Automation Agent is assigned.**

This item must already exist in UDS administration so that it appears in the drop-down list and can be selected.

- **Maximum number of servers that may be created in this server group.**
- **Initial number of servers deployed when the group is created. The system always attempts to maintain at least this number of servers.**

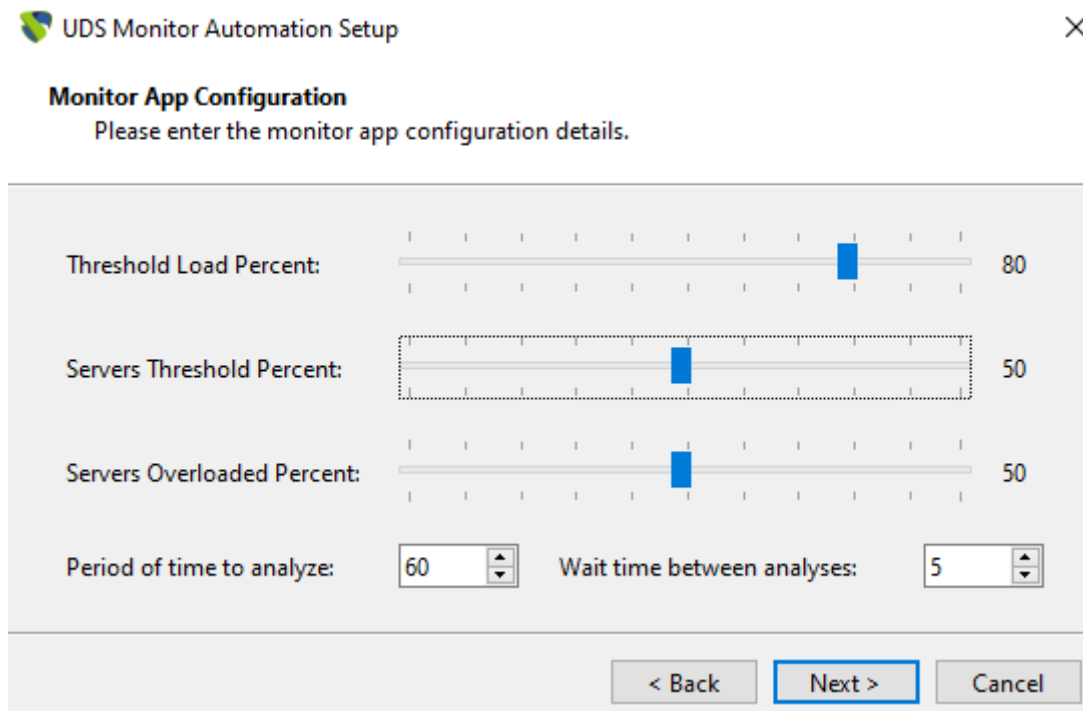
3.3.3. Configuring Monitoring and Scaling Policies

One of UDS Monitor's most important capabilities is continuously analyzing the status and load of the managed RDS server group to determine automatically when the available infrastructure must be expanded.

The next step uses a set of interrelated monitoring and autoscaling parameters. Changing any one of these values can directly affect the overall behavior of the automatic scaling system. Define them according to the infrastructure's expected behavior and the users' usual workload.

UDS Monitor analyzes the environment in periodic monitoring cycles, during which it collects server usage metrics. From this analysis, the system determines whether any servers are overloaded and whether the percentage of affected servers warrants the automatic deployment of new resources.

The available parameters are:



UDS Monitor Automation Setup

Monitor App Configuration
Please enter the monitor app configuration details.

Threshold Load Percent: 80

Servers Threshold Percent: 50

Servers Overloaded Percent: 50

Period of time to analyze: 60 Wait time between analyses: 5

< Back Next > Cancel

- **Threshold Load Percent**

Defines the load percentage above which an individual server is temporarily considered overloaded during a given analysis.

For example, if this value is set to 80%, any server that exceeds 80% during a monitoring cycle is provisionally marked as heavily loaded.

The load percentage is based on internal system metrics such as CPU usage, memory consumption, and the number of active sessions, with predefined weights of 30%, 60%, and 10%, respectively.

The system uses these metrics to calculate an overall utilization value representing the server's general load state.

Exceeding this threshold in a single analysis does not automatically mean that the server is definitively considered overloaded. The value acts only as an initial indicator in each monitoring cycle and is subsequently evaluated together with the other configured parameters.

- **Servers Threshold Percent**

Defines the minimum percentage of analyses in which a server must have exceeded the Threshold Load Percent before it is ultimately considered overloaded.

Its main purpose is to prevent unnecessary scaling caused by temporary spikes or isolated periods of high load.

This parameter depends directly on the values configured for Period of time to analyze and Wait time between analyses.

To determine the total number of analyses performed during a complete monitoring cycle, UDS Monitor divides the Period of time to analyze by the Wait time between analyses.

For example, if Period of time to analyze is 60 minutes and Wait time between analyses is 5 minutes, the system performs $60 / 5 = 12$ analyses. In other words, 12 analysis cycles run during the complete configured period.

If Servers Threshold Percent is set to 50%, the server must have exceeded the load threshold in at least 50% of the completed analyses before it is definitively considered overloaded.

In the preceding example, $12 \times 0.5 = 6$. The server must therefore have been detected as overloaded in at least 6 of the 12 completed analyses.

This mechanism prevents UDS Monitor from reacting to brief increases or temporary load fluctuations that do not represent sustained server saturation.

- **Servers Overloaded Percent**

Defines the minimum percentage of overloaded servers required for UDS Monitor to determine that the overall infrastructure must be expanded automatically.

After identifying the servers that are definitively considered overloaded according to the preceding parameter, the system calculates the total percentage of affected servers in the monitored group.

For example, if this parameter is set to 50%, UDS Monitor starts a scaling process only when at least half of the active servers are overloaded.

This mechanism bases scaling decisions on the overall behavior of the environment rather than on individual or isolated situations.

- **Period of Time to Analyze**

Defines the total time interval used by UDS Monitor to collect and analyze statistical information about server load.

During this period, the system stores the results obtained from each periodic analysis.

The total number of analyses performed during this interval depends directly on the value configured for Wait time between analyses.

This parameter therefore defines the complete time window over which UDS Monitor evaluates server behavior before deciding whether a sustained overload condition actually exists.

A longer analysis period produces more stable statistics that are less sensitive to temporary fluctuations, while a shorter period allows the system to respond more quickly to changes in load.

- **Wait Time Between Analyses**

Defines the amount of time between each monitoring cycle performed by UDS Monitor.

This value determines how often the system collects new utilization metrics and reevaluates the load status of the monitored servers.

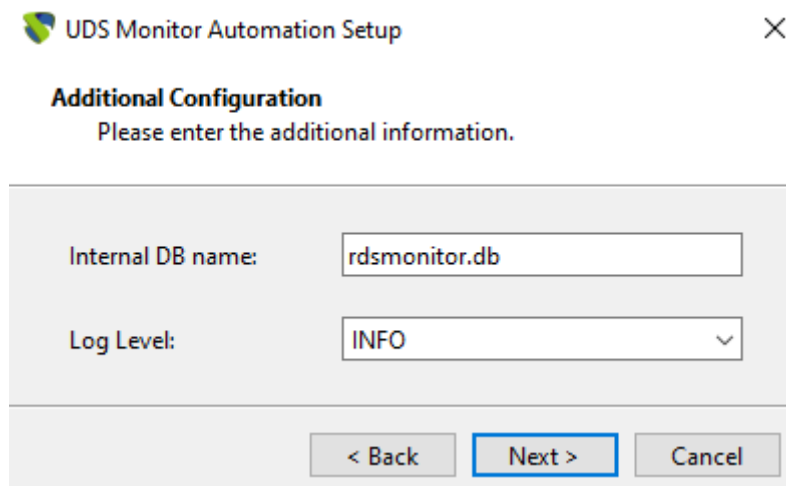
For example, if this value is set to 5 minutes, UDS Monitor checks the status of the servers every 5 minutes throughout the configured analysis period.

A higher analysis frequency detects overload conditions more quickly, but also increases the number of monitoring operations performed by the system.

Conversely, longer intervals reduce the analysis frequency and make the system more conservative in response to temporary load fluctuations.

3.3.4. Configuring the Database and Logging System

In the final wizard step, specify the name of the internal database used by UDS Monitor and the level of logging detail.



The screenshot shows a window titled "UDS Monitor Automation Setup" with a close button (X) in the top right corner. Below the title bar, the text "Additional Configuration" is displayed, followed by the instruction "Please enter the additional information." The main area of the window contains two configuration fields: "Internal DB name:" with a text input field containing "rdsmonitor.db", and "Log Level:" with a dropdown menu currently set to "INFO". At the bottom of the window, there are three buttons: "< Back", "Next >" (which is highlighted with a blue border), and "Cancel".

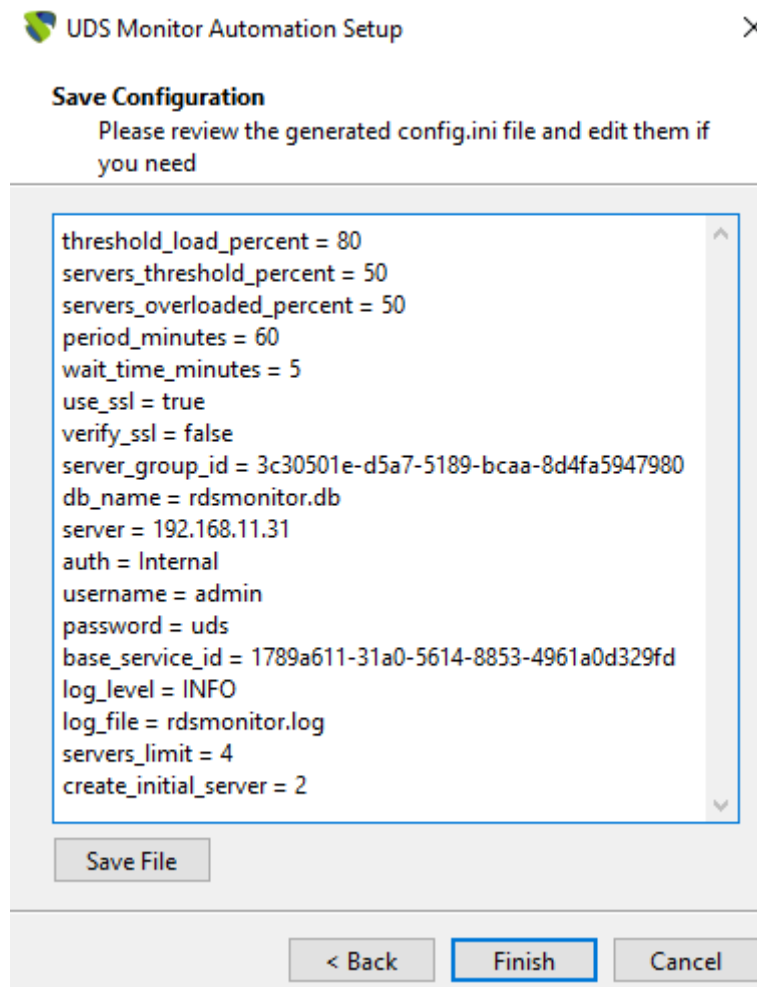
The proposed default database name is normally retained unless the environment has specific requirements.

The logging level defines how much information is recorded while the monitoring service runs, supporting subsequent auditing, tracking, and troubleshooting.

NOTE: The INFO level is recommended. Other levels, such as ERROR, do not record the events performed by the server.

3.3.5. Generating the Configuration Parameters File

After all settings are complete, the wizard displays the contents of the configuration file, rds_monitor.ini, and allows it to be saved for later use.



```
File Edit Format View Help
threshold_load_percent = 80
servers_threshold_percent = 50
servers_overloaded_percent = 50
period_minutes = 60
wait_time_minutes = 5
use_ssl = true
verify_ssl = false
server_group_id = 3c30501e-d5a7-5189-bcaa-8d4fa5947980
db_name = rdsmonitor.db
server = 192.168.11.31
auth = Internal
username = admin
password = __8c0885a97d406ce308b3f09c1e6be4f00c48d062c4a6
base_service_id = 1789a611-31a0-5614-8853-4961a0d329fd
log_level = INFO
log_file = rdsmonitor.log
servers_limit = 4
create_initial_server = 2
```

NOTE: If the template machine was used to run the UDS Monitor configuration wizard (uds-monitor-wizard.exe), save rds_monitor.ini outside the machine so that it remains available later. Then leave the template powered off so that clones can subsequently be generated in a consistent state.

3.4. Activating the UDS Monitor Container

After preparing the template and generating rds_monitor.ini with the UDS Monitor configuration wizard (uds-monitor-wizard.exe), connect to the Utility Machine deployed on the selected virtualization platform, preferably through SSH, and enable the UDS Monitor container.

The machine's default credentials are:

- User: root
- Password: uds

NOTE: The default password can be changed with the passwd command.

The machine obtains its network configuration through DHCP. Assigning a static IP address is recommended. To do so, modify /etc/network/interfaces and restart the machine to confirm that the changes have been applied.

```
GNU nano 7.2 /etc/network/interfaces *
# This file describes the network interfaces available on your s
# and how to activate them. For more information, see interfaces

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
    address 192.168.11.103
    netmask 255.255.255.0
    gateway 192.168.11.1

allow-hotplug enp1s0
iface enp1s0 inet dhcp

allow-hotplug ens32
```

After signing in to the machine and configuring a static IP address, enable the container that runs UDS Monitor with the following command:

rdsmonitor-tool enable

```
root@utility-500:~# rdsmonitor-tool enable
```

After this command is run, the system automatically deploys the Docker container for the monitoring service.

```
root@utility-500:~# rdsmonitor-tool enable
4d2ae1e5026f62d75a0d354a6be8df1f07b5a69b1f4ed8540404ecb5bf04d471
2026-06-25 13:24:13,733 INFO Container 'rds-monitor' created and started successfully.
2026-06-25 13:24:13,733 WARNING Configuration file not found at /etc/utility/rds_monitor.ini. Create it (setup w
izard), then the monitor will pick it up on the next start.
2026-06-25 13:24:13,733 WARNING After creating the config file, restart the container with: monitor-tool restart
root@utility-500:~#
```

The tool indicates that the UDS Monitor configuration file, `rds_monitor.ini`, must be supplied. This file was generated earlier with `uds-monitor-wizard.exe`.

Check the container status with the following command:

docker ps

```
root@utility-500:~# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
1e79cbabe86b	virtualcableuds/rds-monitor:5.0-20260625	"python -m monitor_r..."	About a minute ago	Up About a minute		rds-monitor

```
root@utility-500:~#
```

The `rdsmonitor-tool` utility provides commands that simplify management of the Docker container used by UDS Monitor on the Utility Machine.

These commands support routine management, maintenance, and updating of the monitoring service without requiring direct interaction with Docker.

```
root@utility-500:~# rdsmonitor-tool -h
usage: rdsmonitor-tool [-h] {enable,start,stop,restart,disable,status,update,create-config,logs,versions,add-servers}
positional arguments:
  {enable,start,stop,restart,disable,status,update,create-config,logs,versions,add-servers}
  enable                Check config, paths and enable
  start                 Start the existing container (run enable first to create it)
  stop                  Stop the container (restarts on reboot unless disabled)
  restart               Restart the container
  disable               Stop and remove the container
  status                Show the container status
  update                Update to a specific --version (lists available versions if omitted)
  create-config          Create a default config file
  logs                  Follow the container logs
  versions               List available image versions from Docker Hub
  add-servers            Add N servers to the existing pool
options:
  -h, --help            show this help message and exit
```

The command syntax is:

`rdsmonitor-tool [arguments]`

enable

Enables and deploys the Docker container for UDS Monitor.

start

Starts server creation and monitoring.

stop

Stops the monitoring service.

restart

Restarts the UDS Monitor container.

This is particularly useful after changes to `rds_monitor.ini`, other configuration changes, or environment-related updates.

disable

Stops and disables the UDS Monitor container.

status

Displays the current status of the Docker container used by UDS Monitor.

update

Updates the container. The version parameter must be supplied with the target version number.

create-config

Creates an example configuration file (rds_monitor.ini).

logs

Displays the monitoring service's system logs.

versions

Displays the available UDS Monitor container versions.

add-servers

Adds new servers to the server group.

3.5. Configuring the rds_monitor.ini File

During the initial deployment of the UDS Monitor container with rdsmonitor-tool enable, the tool requests the configuration to be provided through rds_monitor.ini. It automatically creates the following path on the machine:

/etc/utility/

Copy the rds_monitor.ini file previously generated by the UDS Monitor configuration wizard to this location.

After copying the file, edit it and apply any required changes:

```
GNU nano 8.4 /etc/utility/rds_monitor.ini
threshold_load_percent = 80
servers_threshold_percent = 50
servers_overloaded_percent = 50
period_minutes = 60
wait_time_minutes = 5
use_ssl = true
verify_ssl = false
server_group_id = 3c30501e-d5a7-5189-bcaa-8d4fa5947980
db_name = rdsmonitor.db
server = 192.168.11.31
auth = Internal
username = admin
password = __8c0885a97d406ca558cec8bc175dbdea1350864fe48ee8b89425449bb4ca
base_service_id = 1789a611-31a0-5614-8853-4961a0d329fd
log_level = INFO
log_file = rdsmonitor.log
servers_limit = 4
create_initial_server = 2
enable_rdsmonitor_web = true
```

NOTE: The specified administrator password cannot be changed because it is encrypted. Modifying the server group to monitor or the base service is also not recommended because they are specified by ID. To change any of these three values, run the UDS Monitor configuration wizard again to generate a new file, or copy in the new values.

This file contains all configuration used by UDS Monitor, including:

- UDS Enterprise connection parameters.
- Virtualization provider configuration and the base service to use.
- Server group to monitor.
- Monitoring and scaling policies.
- Automatic deployment configuration.
- Login parameters and general behavior.

Correct placement and reading of this file are essential for system operation.

3.6. Starting the Monitoring Service

After copying the configuration file, start the monitoring service. This also launches the initial creation of the number of RDS servers specified in the file.

Start monitoring with the following command: `rdsmonitor-tool start`

```
root@utility-500:~# rdsmonitor-tool start
rds-monitor
2026-07-14 16:40:20,657 INFO Container 'rds-monitor' started successfully.
root@utility-500:~#
```

On the first run, the system detects that the specified minimum number of servers does not exist and begins creating them. By default, it checks every 5 minutes whether the servers are available, performing a maximum of 12 checks over a total of 60 minutes.

In parallel, the system creates the Service Pool required to manage the servers and their publication. This is explained in more detail below:

☐	 UDSRDS-202606021646	Locked	0	0	0%
--------------------------	---	--------	---	---	----

When the servers have been created and the specified initial number is available, the initial deployment phase ends. UDS Monitor continues active monitoring and waits until the configured automatic growth policies require new servers. The `rdsmonitor-tool` logs command displays these tasks in real time:

```
2026-06-02 17:36:30,789 - monitor.workers.initial - INFO - Server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 now has 2 servers.
2026-06-02 17:36:30,789 - monitor.workers.initial - INFO - Initial worker completed successfully
.
2026-06-02 17:36:30,789 - monitor.workers.monitor - INFO - Starting data collection process.
2026-06-02 17:36:30,885 - monitor.workers.expander - INFO - Server group Grupo1(3c30501e-d5a7-5189-bcaa-8d4fa5947980) does not need to be expanded. Waiting for next check...
2026-06-02 17:41:30,952 - monitor.workers.expander - INFO - Server group Grupo1(3c30501e-d5a7-5189-bcaa-8d4fa5947980) does not need to be expanded. Waiting for next check...
```

3.7. UDS Monitor Operation

After the monitoring service has started correctly, UDS Monitor automatically begins initialization and performs the tasks described below:

3.7.1. Automatic Creation of the Service Pool

The first operation performed by UDS Monitor is the automatic creation of the corresponding Service Pool in UDS Enterprise. It is created in a special state named "Locked":

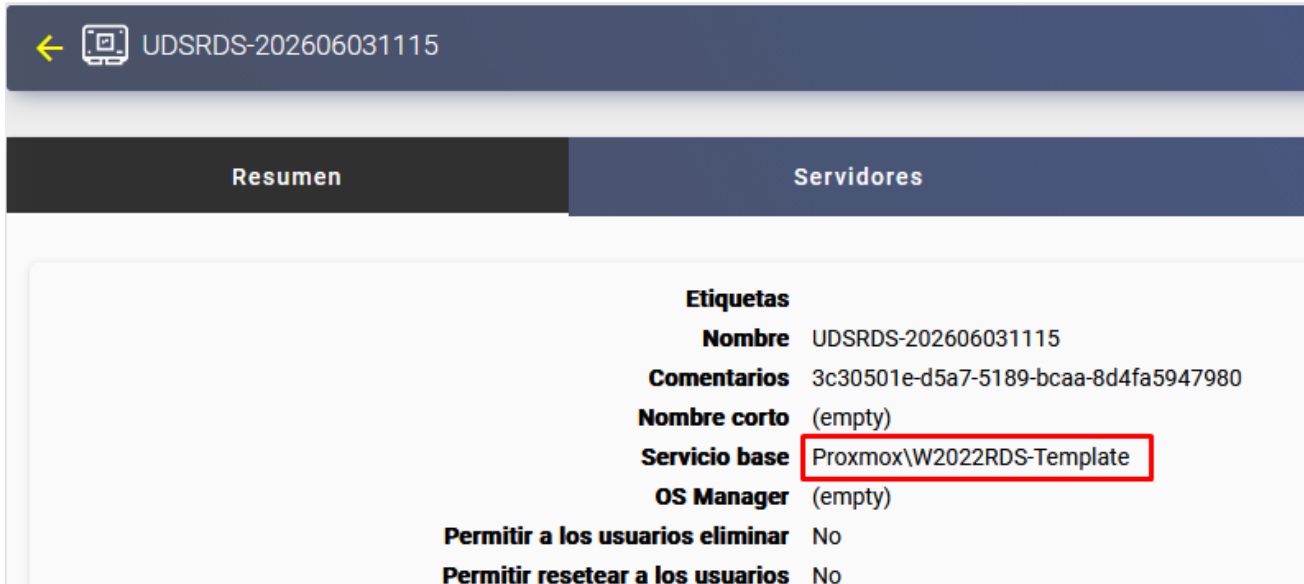
Service Pools						
New Edit Permissions Export CSV Delete						
Name ↑	Status	User services	In Preparation	Usage	Visible	
☐  UDSRDS-202606021646	Locked	0	0	0%	no	

This Service Pool is used internally to manage dynamically deployed servers and maintain automated control of the infrastructure. It is therefore not visible to users and cannot have transports.

When the Service Pool is created, its publication is also created on the virtualization platform. A clone named UDSP, UDS-Pub, or similar is generated, depending on the platform. When the publication reaches a valid state, the system begins creating the initial RDS servers specified in `rds_monitor.ini`.

NOTE: If an error occurs during this initial publication, investigate its cause and then create another publication by opening the pool in the Locked state, selecting the Publications tab, and clicking New.

To identify the created pool when several pools exist, display the Parent Service column. Alternatively, open the pool and view its base service on the Summary tab. It corresponds to the service specified in rds_monitor.ini on the UDS Monitor machine.



UDSRDS-202606031115

Resumen **Servidores**

Etiquetas

Nombre UDSRDS-202606031115

Comentarios 3c30501e-d5a7-5189-bcaa-8d4fa5947980

Nombre corto (empty)

Servicio base Proxmox\W2022RDS-Template

OS Manager (empty)

Permitir a los usuarios eliminar No

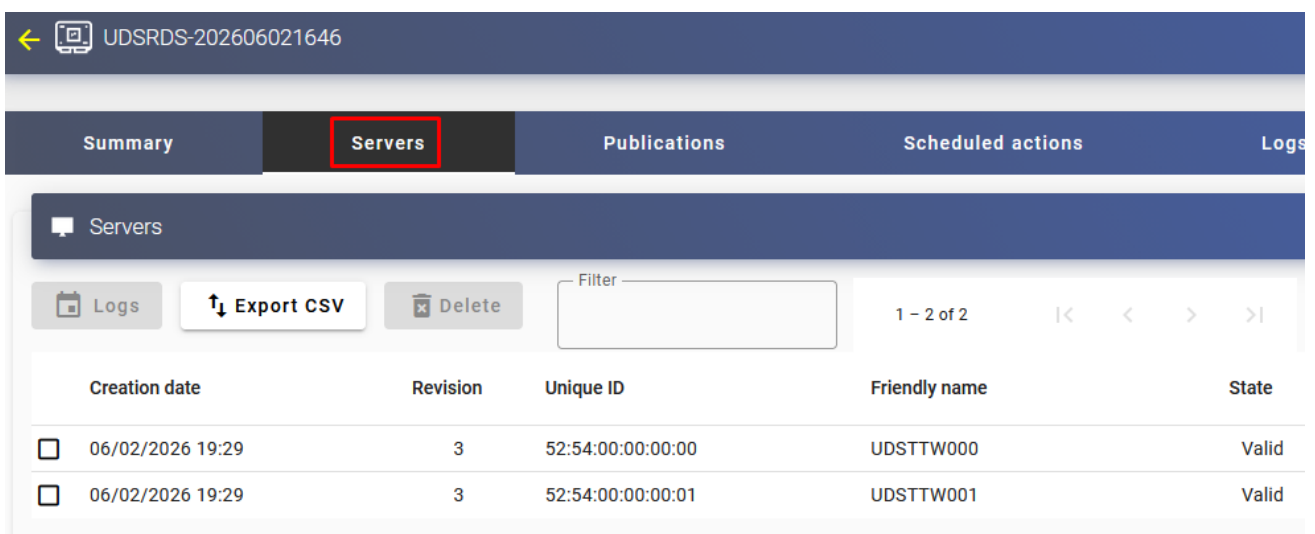
Permitir resetear a los usuarios No

3.7.2. Initial Server Deployment

The system then automatically deploys the initial number of servers previously defined in rds_monitor.ini.

Each server is cloned from the configured template and automatically runs the complete configuration process defined through RDS-Automation.

The Servers tab of the Service Pool in the Locked state displays the servers:



UDSRDS-202606021646

Summary **Servers** **Publications** **Scheduled actions** **Logs**

Servers

Logs **Export CSV** **Delete** **Filter**

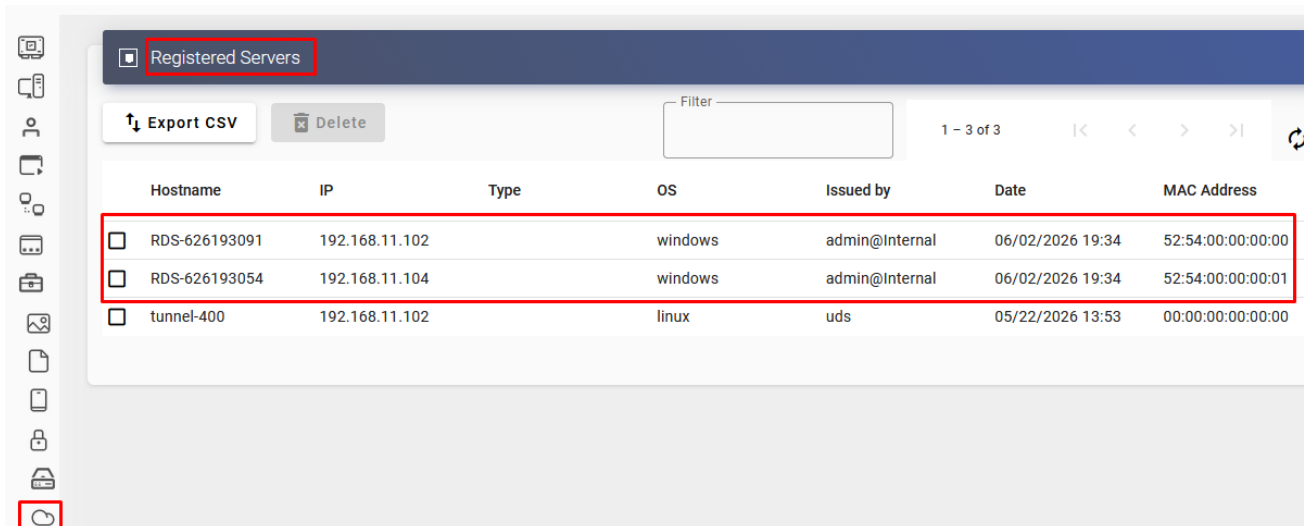
1 - 2 of 2

	Creation date	Revision	Unique ID	Friendly name	State
☐	06/02/2026 19:29	3	52:54:00:00:00:00	UDSTTW000	Valid
☐	06/02/2026 19:29	3	52:54:00:00:00:01	UDSTTW001	Valid

This process may take some time because the servers must be generated and configured. They appear on the virtualization platform.

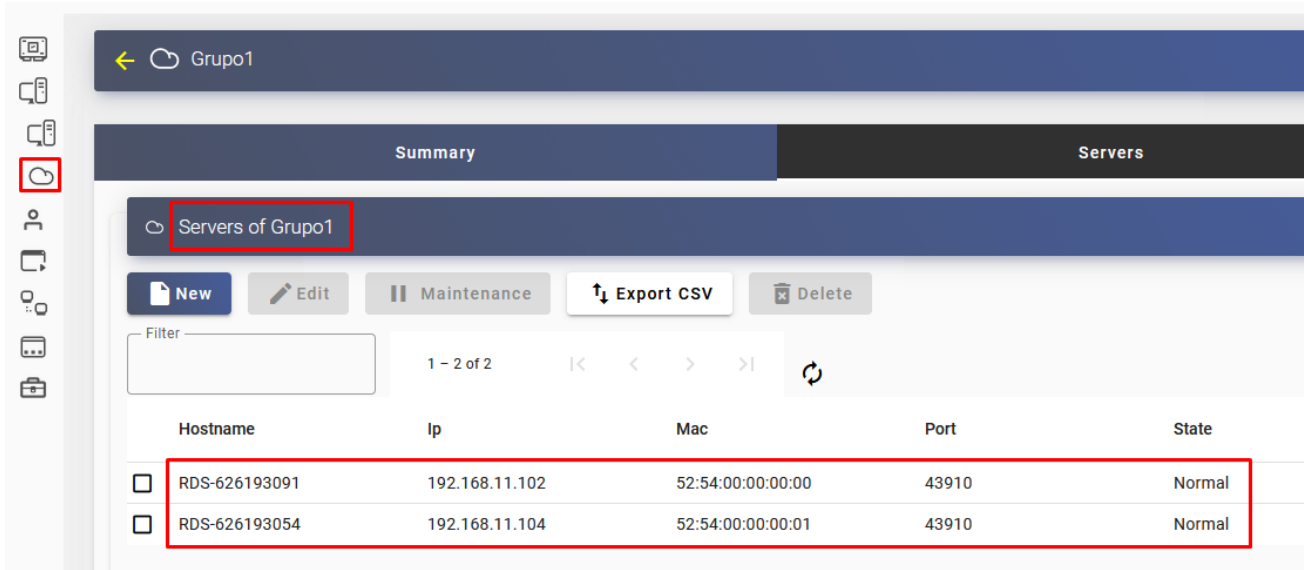
3.7.3. Automatic Server Registration

After each server has been configured by the RDS-Automation Agent, it is registered automatically. Its token is registered first and is visible under “Tools” - “Tokens” - “Servers”:



Hostname	IP	Type	OS	Issued by	Date	MAC Address
☐ RDS-626193091	192.168.11.102		windows	admin@Internal	06/02/2026 19:34	52:54:00:00:00:00
☐ RDS-626193054	192.168.11.104		windows	admin@Internal	06/02/2026 19:34	52:54:00:00:00:01
☐ tunnel-400	192.168.11.102		linux	uds	05/22/2026 13:53	00:00:00:00:00:00

After the tokens are registered, the servers are automatically added to the group specified in the RDS-Automation Agent. To view the servers in a group, select “Services” - “Servers” and open the relevant group.



Hostname	Ip	Mac	Port	State
☐ RDS-626193091	192.168.11.102	52:54:00:00:00:00	43910	Normal
☐ RDS-626193054	192.168.11.104	52:54:00:00:00:01	43910	Normal

If the servers are registered in both locations, the initial deployment was successful and all processes operated correctly.

3.7.4. Monitoring and Automatic Server Expansion

As explained earlier, UDS Monitor monitors server load according to the configured policies. The log shows when the system determines that the number of servers must be increased.

- The system indicates that scaling is unnecessary because the requirements have not been met:

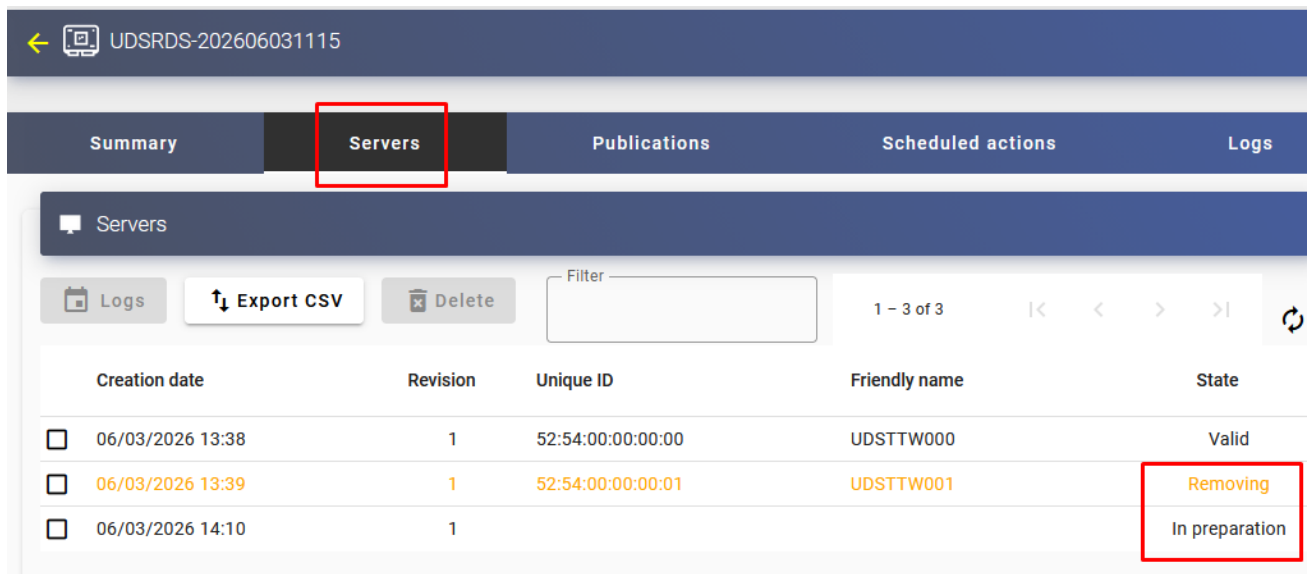
```
2026-06-03 11:55:17,616 - monitor.workers.expander - INFO - Server group Grupo1(3c30501e-d5a7-5189-bcaa-8d4fa5947980) does not need to be expanded. Waiting for next check...
2026-06-03 12:00:17,706 - monitor.workers.expander - INFO - Server group Grupo1(3c30501e-d5a7-5189-bcaa-8d4fa5947980) does not need to be expanded. Waiting for next check...
```

- The system detects that the number of servers must be increased:

```
2026-06-03 21:29:55,247 - monitor.workers.expander - INFO - Server group Grupo1(3c30501e-d5a7-5189-bcaa-8d4fa5947980) needs to be expanded.
2026-06-03 21:29:55,247 - monitor.workers.expander - INFO - Increasing servers for group 3c30501e-d5a7-5189-bcaa-8d4fa5947980.
```

3.7.5. Automatic Replacement of Cache Servers

Within the Service Pool in the Locked state automatically generated by UDS Monitor, the system continuously monitors the cache servers associated with that pool on its Servers tab. If one is deleted manually, the system automatically replaces it and restores the number of servers to the configured initial value:



UDSRDS-202606031115

Summary **Servers** Publications Scheduled actions Logs

Servers

Logs Export CSV Delete Filter 1 - 3 of 3

	Creation date	Revision	Unique ID	Friendly name	State
☐	06/03/2026 13:38	1	52:54:00:00:00:00	UDSTTW000	Valid
☐	06/03/2026 13:39	1	52:54:00:00:00:01	UDSTTW001	Removing
☐	06/03/2026 14:10	1			In preparation

To check how many servers the pool must have available, review the Summary tab. Initially, this value always matches the value in the UDS Monitor rds_monitor.ini configuration file. However, if the system has detected that more servers are required, the value may change and no longer match the file.

UDSRDS-202606031115

Summary Servers Publications

Tags

Name UDSRDS-202606031115

Comments 3c30501e-d5a7-5189-bcaa-8d4fa5947980

Short name (empty)

Base service Proxmox\W2022RDS-Template

OS Manager (empty)

Allow removal by users No

Allow reset by users No

Ignores unused No

Visible No

Associated Image  -----

Pool group  Default

Calendar access denied text (empty)

Custom launch message text (empty)

Enable custom launch message No

Initial available services 2

Services to keep in cache 0

Services to keep in L2 cache 0

Maximum number of services to provide 2

Whenever a server is deleted from this section, its virtual machine is removed from the virtualization platform, it is removed from the server group, and its token is deleted in a cascading operation.

3.7.6. Publishing New Versions

A common task is updating the template machine's operating system, installing new applications, or performing other maintenance that makes existing deployed servers obsolete. To transfer these template changes to the RDS servers, create a new publication of the service.

Publishing in UDS Enterprise generates new servers based on the template and automatically removes the old servers. To create a publication, open the Service Pool, select Publications, and click New:

←

UDSRDS-202606031115

Summary

Servers

Publications

Scheduled

Publications

New

Cancel

Changelog

Export CSV

Filter

Revision

Publish date

State

Reason

1

06/03/2026 13:15

Valid

A new publication machine, cloned from the template, is generated on the hypervisor platform. Wait for its state to change from Preparing to Valid:

←

UDSRDS-202606031115

Resumen

Servidores

Publicaciones

Acciones pro

Publicaciones

Nuevo

Cancelar

Registro de cambios

Exportar a CSV

Filtro

1 - 2 of 2

⏪

⏴

⏵

⏩

↺

Revisión

Fecha de publicación

Estado

Razón

1

03/06/2026 13:15

Eliminando

2

03/06/2026 22:53

Válido

All old servers, which can be identified in the Revision column, change to the Deleting state while new servers with the current revision are created in parallel:

←

UDSRDS-202606031115

Resumen

Servidores

Publicaciones

Acciones programadas

Logs

Servers

Logs

Exportar a CSV

Borrar

Filtro

1 - 4 of 4

<

<

>

>

↺

	Fecha de creación	Revisión	Unique ID	Nombre descriptivo	Estado
☐	03/06/2026 17:59	1	52:54:00:00:00:06	UDSTTW006	Eliminando
☐	03/06/2026 18:07	1	52:54:00:00:00:07	UDSTTW007	Eliminando
☐	03/06/2026 23:05	2	52:54:00:00:00:00	UDSTTW000	En preparación
☐	03/06/2026 23:05	2	52:54:00:00:00:01	UDSTTW001	En preparación

NOTE: Any server with an active user session or application is deleted as well. Perform publication when no users are connected. You can place either the service or the servers in the server group into maintenance mode to prevent new connections.

3.7.7. Deleting a Service Pool

If the Service Pool in the Locked state generated automatically by the rds-monitor monitoring service must be deleted, or is deleted by mistake, it can be removed at any time by selecting it and clicking Delete:

Service Pools

New

Edit

Permissions

Export CSV

Delete

Name ↑	Status	User selected
☒  UDSRDS-202608311242	Locked	

This action deletes all existing generated servers from both UDS Enterprise administration, including the Service Pool, server tokens, and server group, and from the virtualization platform that manages them.

After the pool is deleted, the monitoring service logs show errors because the monitored pool no longer exists:

```
2026-08-31 12:41:50,315 - monitor.api - WARNING - API call failed: HTTP Error: code=0, message="Service pool for server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 not found"
2026-08-31 12:41:55,340 - monitor.api - WARNING - API call failed: HTTP Error: code=0, message="Service pool for server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 not found"
2026-08-31 12:42:00,366 - monitor.api - WARNING - API call failed: HTTP Error: code=0, message="Service pool for server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 not found"
```

To force the system to generate the Service Pool and servers again, restart the rds-monitor monitoring container or restart the server:

```
root@utility-500:~# rdsmonitor-tool restart
rds-monitor
2026-08-31 12:42:23,499 INFO Container 'rds-monitor' restarted successfully.
root@utility-500:~#
```

NOTE: When the pool is created again, the values in `rds_monitor.ini` are applied, including the `create_initial_server` parameter.

3.7.8. Graphical Monitoring of RDS Servers

In addition to the monitoring and autoscaling capabilities described above, UDS Monitor provides a web interface that graphically displays the behavior and load of the RDS servers managed by the system.

This portal provides real-time access to metrics collected by the agents installed on the RDS servers. It offers a detailed view of infrastructure status and makes it possible to understand precisely the criteria UDS Monitor uses to determine when available capacity must be expanded automatically.

The information displayed in this interface is valuable for supervision, scaling-policy validation, and analysis of user behavior within the environment.

Access the monitoring web console by using the IP address or DNS name of the Utility Machine and adding `/rdsmonitor` to the URL.

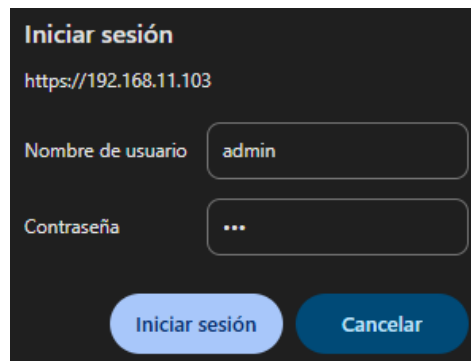
`https://UTILITY_MACHINE_IP/rdsmonitor`

```
https://192.168.11.103/rdsmonitor/
```

After the connection is established, the system requests authentication. Use the following default credentials:

User: admin

Password: The password configured on the Utility Machine.



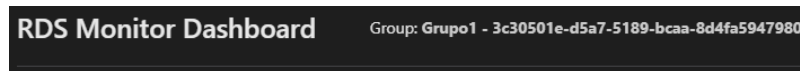
Iniciar sesión

<https://192.168.11.103>

Nombre de usuario

Contraseña

After successful authentication, the main monitoring dashboard appears.



The charts displayed in the portal use data collected directly by UDS Monitor during each monitoring cycle.

Data-collection frequency is determined by Wait time between analyses, which defines the interval between samples collected by the system. The total amount of information displayed in the charts depends on Period of time to analyze, which defines the complete time window used to present historical data. Both values are specified in `rds_monitor.ini`.

The display therefore reflects the same information that UDS Monitor uses internally to evaluate server load and decide whether new resources must be deployed.

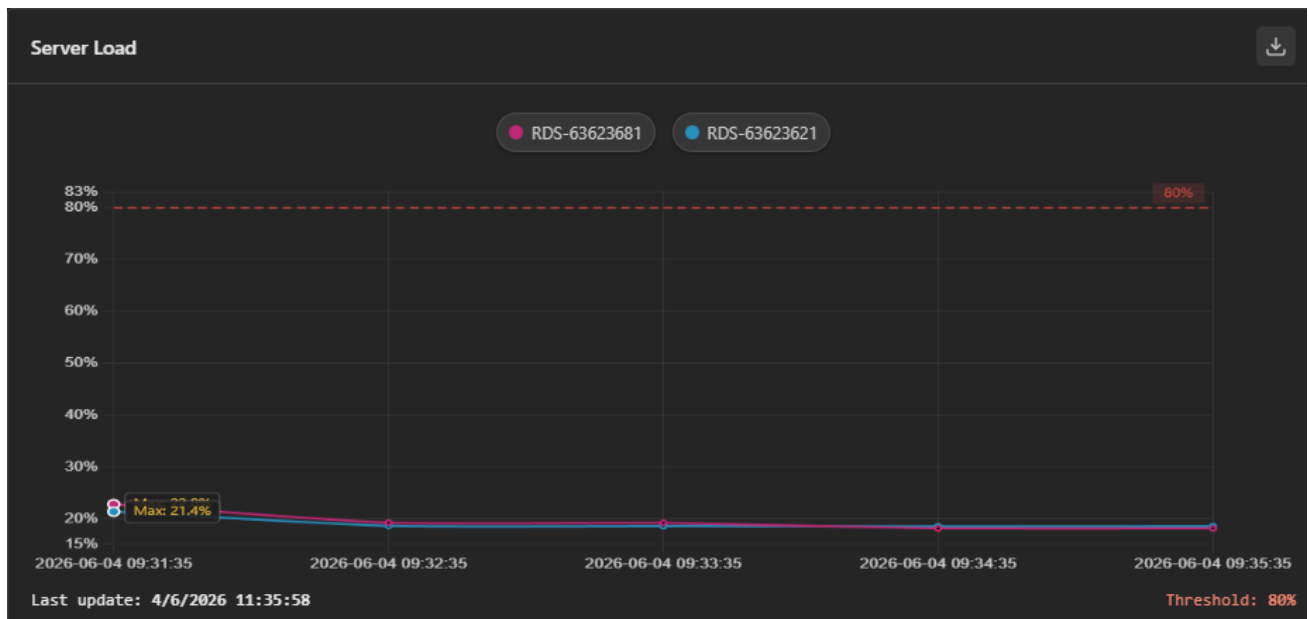
Each monitored server is represented by a separate line, making it easy to compare the load across all members of the group.

The following metrics are available:

- **Server Load: The load level calculated for each monitored RDS server. By default, it is based on CPU usage, memory usage, and number of users, weighted at 30%, 60%, and 10%, respectively.**

The chart includes a dashed reference line representing the Threshold Load Percent configured in `rds_monitor.ini`.

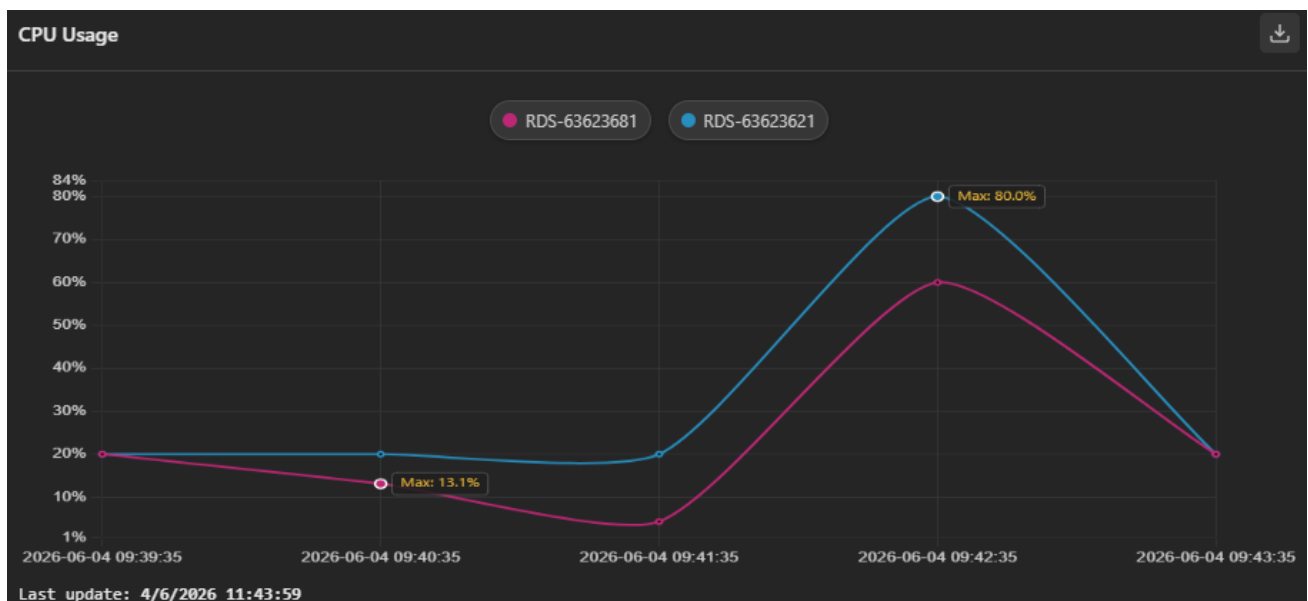
When a server's line repeatedly exceeds this threshold, the server begins to count as potentially overloaded in the automatic scaling algorithm.



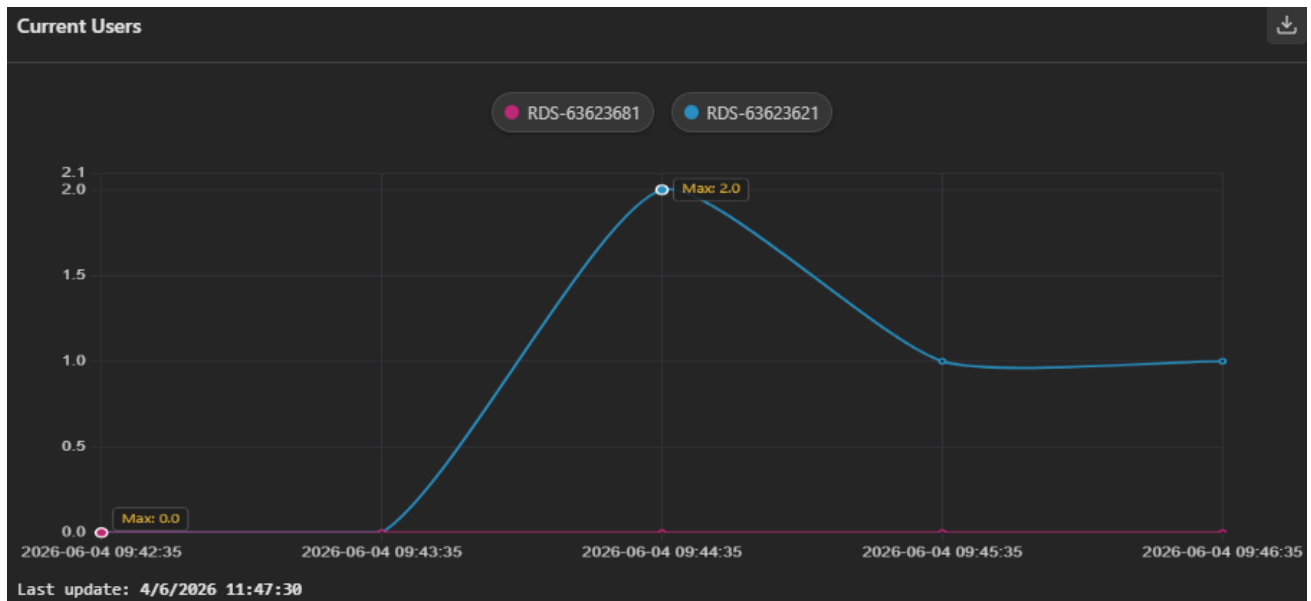
- **Memory Usage:** Displays RAM consumption for each monitored server.



- **CPU Usage:** Displays CPU consumption for each monitored server.

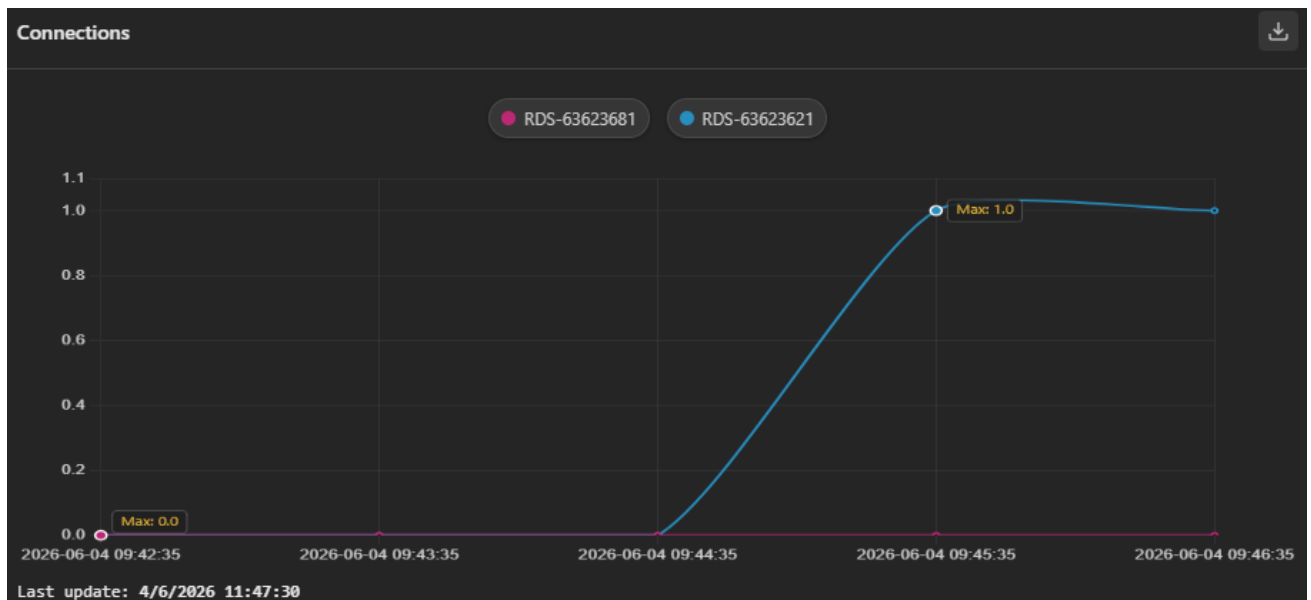


- **Current Users:** Displays the number of users with an active session on each RDS server.



- **Connections:** Represents the total number of connections established on each server.

This value does not necessarily match the number of connected users, because one user may maintain several simultaneous connections, for example through desktop sessions and application sessions.



3.8. Modifying Values in the rds_monitor.ini File

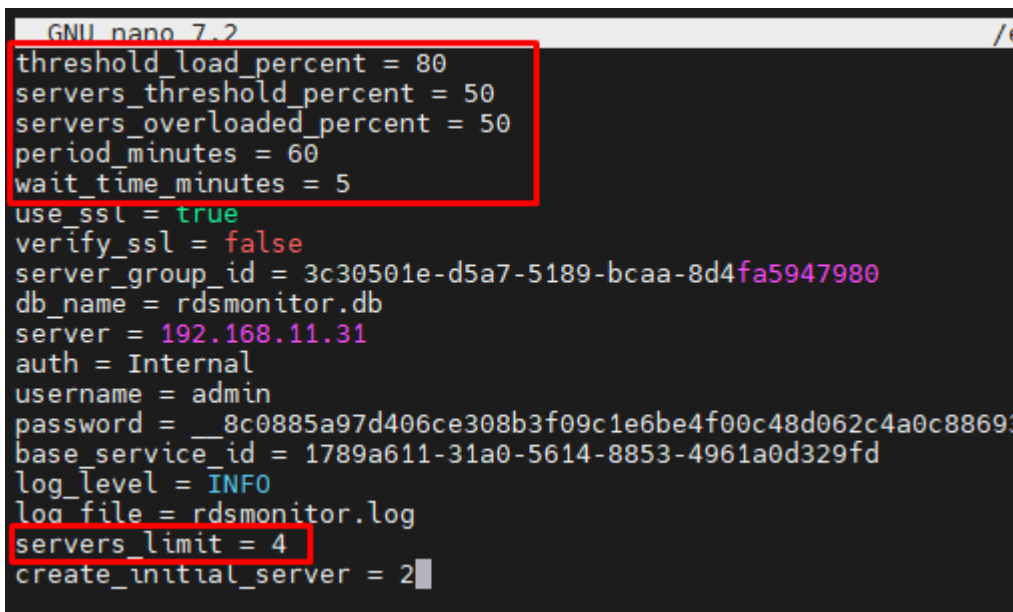
After UDS Monitor has been deployed and started, certain configuration parameters may need to be changed to adapt system behavior to new operational requirements or infrastructure changes.

Common situations include:

- Adjusting the load thresholds used by the autoscaling system.
- Changing the maximum number of permitted servers.
- Changing the monitoring and analysis periods.
- Changing parameters related to the virtualization platform or UDS Enterprise.

All these settings are stored in `rds_monitor.ini` at the following path on the Utility Machine:

`/etc/utility/rds_monitor.ini`



```
GNU nano 7.2 /etc/utility/rds_monitor.ini
threshold_load_percent = 80
servers_threshold_percent = 50
servers_overloaded_percent = 50
period_minutes = 60
wait_time_minutes = 5
use_ssl = true
verify_ssl = false
server_group_id = 3c30501e-d5a7-5189-bcaa-8d4fa5947980
db_name = rdsmonitor.db
server = 192.168.11.31
auth = Internal
username = admin
password = __8c0885a97d406ce308b3f09c1e6be4f00c48d062c4a0c88693
base_service_id = 1789a611-31a0-5614-8853-4961a0d329fd
log_level = INFO
log_file = rdsmonitor.log
servers_limit = 4
create_initial_server = 2
```

To ensure that changes are applied correctly, stop the monitoring service first.

After stopping the service, edit `rds_monitor.ini` directly with any text editor. Alternatively, generate a new configuration file with the UDS Monitor configuration wizard and replace the existing file.

The parameters commonly changed are those related to monitoring policies, autoscaling, and environment capacity.

Some changes may require additional actions before they take effect. Depending on the parameter, restarting the monitoring service may be sufficient, or the entire UDS Monitor container may need to be restarted.

Use the following command to restart the container:

`rdsmonitor-tool restart`

```
root@utility-400:~# rdsmonitor-tool --restart  
rds-monitor  
2026-06-04 13:47:37,541 INFO Container 'rds-monitor' restarted successfully.  
root@utility-400:~# █
```

After making the changes, start the monitoring service again so that UDS Monitor begins using the new configuration.

Some parameters support dynamic changes and can be updated directly in the configuration file without stopping the service. For example, new `threshold_load_percent` values are used automatically in subsequent monitoring cycles.

Not all parameters in `rds_monitor.ini` behave in the same way.

Some values take effect only at specific points in the Service Pool lifecycle. In these cases, changing the configuration file alone does not immediately affect the deployed infrastructure.

A representative example is `create_initial_server`, whose behavior is described in detail in the following section.

3.8.1. Modifying the Number of Initial Servers

The `create_initial_server` parameter defines the minimum number of RDS servers that must exist in the Service Pool managed by UDS Monitor.

Although it is configured in `rds_monitor.ini`, its behavior differs from that of other monitoring and autoscaling parameters.

UDS Monitor evaluates this value only in the following situations:

- When the Service Pool is created.
- When the system detects that the total number of servers managed by the Service Pool is zero.
- When changed through the `rdsmonitor-tool add-servers` command.

Consequently, changing the parameter directly in the configuration file does not immediately create or delete additional servers.

The configured minimum number of servers can be viewed on the Summary page of the corresponding Service Pool. The Initial Servers value represents the minimum number of servers that UDS Monitor attempts to keep available.

←  UDSRDS-202606031115

Resumen
Servidores

Etiquetas

Nombre

UDSRDS-202606031115

Comentarios

3c30501e-d5a7-5189-bcaa-8d4fa5947980

Nombre corto

(empty)

Servicio base

Proxmox\W2022RDS-Template

OS Manager

(empty)

Permitir a los usuarios eliminar

No

Permitir resetear a los usuarios

No

Ignora no usadas

No

Visible

No

Imagen asociada

 _____

Grupo de Pool

 Por defecto

Texto de acceso denegado al calendario

(empty)

Texto del mensaje en inicio de servicio

(empty)

Habilitar mensaje en inicio de servicio

No

Servicios disponibles inicialmente

2

Servicios para mantener en la caché

0

Servicios para mantener en la caché L2

0

Número máximo de servicios a proveer

2

Mostrar transportes

No

Contadores

(empty)

This value may increase automatically when the system detects a need to scale because of the environment's load. It can also be reduced by deleting servers through their tokens.

Deleting a server directly from the UDS Enterprise Servers section does not have the intended effect, because UDS Monitor detects the missing resource and automatically recreates it.

Therefore, any controlled reduction in the number of servers must be performed by deleting the corresponding tokens.

Deleting a token does not necessarily delete the associated server immediately.

The following conditions must be met for the process to complete correctly:

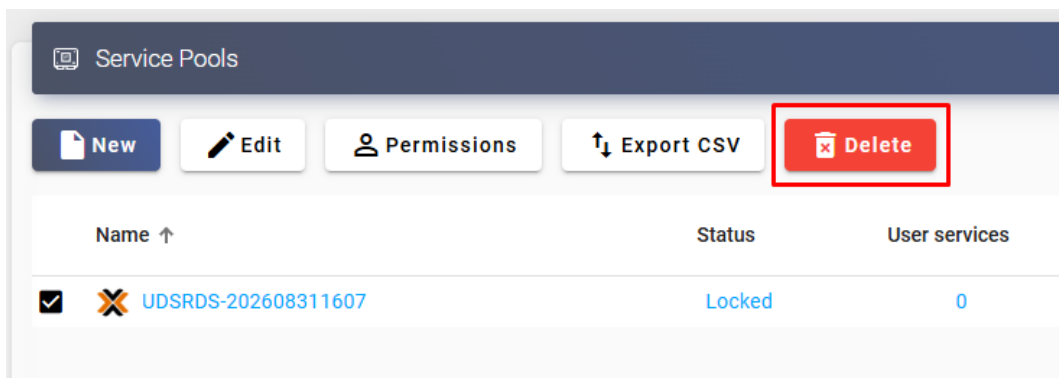
- The server must have existed for more than one hour.
- The monitoring service must be active.

If the token is deleted less than one hour after the server was created, UDS Monitor keeps the resource active until the minimum period has elapsed.

After this period, the server is automatically deleted from both the Service Pool and the corresponding virtualization platform.

- **Example 1 - Changing the Initial Servers of an Existing Pool by Deleting the Service Pool**

A simple way to change the initial servers is to select and delete the resource pool in the Locked state.



When the pool is deleted, the system removes all generated servers. Edit the UDS Monitor configuration file to specify a new initial server count, then generate the Service Pool again with `rdsmonitor-tool restart`.

- **Example 2 - Changing the Initial Servers of an Existing Pool by Modifying the Value Directly in the rds-monitor Container**

Instead of deleting every server in the pool, change the value directly in the UDS Monitor container. Run `rdsmonitor-tool add-servers` and specify the number of servers to add:

```
root@utility-500:~# rdsmonitor-tool add-servers 2
2026-09-01 16:20:36,307 - monitor.logsetup - INFO - Logging at level INFO (/monitor/src/logs/rdsmonitor.log).
root@utility-500:~#
```

This procedure creates two new servers and also changes the pool's initial server value.

- **Example 3 - Changing the Initial Servers of an Existing Pool When the Total Number of Servers Is Zero**

Delete the tokens associated with all servers belonging to the Service Pool.

Registered Servers

Export CSV

Delete

Filter

rds

1 - 2 of

	Hostname	IP	Type	OS	Issued by
☒	RDS-656135489	192.168.11.108		windows	admin@Internal
☒	RDS-656133115	192.168.11.102		windows	admin@Internal

Delete servers token - USE WITH EXTREME CAUTION!!!

Are you sure do you want to delete the following items?
Token issued by admin@Internal from 192.168.11.102, Token issued by admin@Internal from 192.168.11.108

Yes

No

This causes the pool's servers to be deleted automatically, provided that they have been active for more than one hour. They are also deleted from the virtualization platform:

UDSRDS-202606031115

Summary

Servers

Publications

Scheduled actions

Logs

Servers

Logs

Export CSV

Delete

Filter

1 - 2 of 2

<<

<

>

>>

Refresh

	Creation date	Revision	Unique ID	Friendly name	State
☐	06/05/2026 13:30	2	52:54:00:00:00:00	UDSTTW000	Removing
☐	06/05/2026 13:53	2	52:54:00:00:00:01	UDSTTW001	Removing

After they have been deleted, refresh the Summary tab. Initial Servers is displayed as 0:

Custom launch message text	(empty)
Enable custom launch message	No
Initial available services	0
Services to keep in cache	0
Services to keep in L2 cache	0
Maximum number of services to provide	1
Show transports	No

The monitoring service also reports that it has deleted them:

```
- monitor.workers.sync - INFO - Deleting one user service from pool because it has no associated server in server group.
- monitor.workers.sync - INFO - Deleting one user service from pool because it has no associated server in server group.
- monitor.workers.sync - INFO - Deleting one token from tokens because it has no used in any server group.
- monitor.workers.sync - INFO - Deleting one token from tokens because it has no used in any server group.
```

Stop the monitoring service and edit rds_monitor.ini with the new initial server value:

```
GNU nano 7.2 /etc/rdsmonitor config/config.ini
threshold_load_percent = 10
servers_threshold_percent = 50
servers_overloaded_percent = 10
period_minutes = 2
wait_time_minutes = 1
use_ssl = true
verify_ssl = false
server_group_id = 3c30501e-d5a7-5189-bcaa-8d4fa5947980
db_name = rdsmonitor.db
server = 192.168.11.31
auth = Internal
username = admin
password = __8c0885a97d406ce308b3f09c1e6be4f00c48d062c4a0c8869
base_service_id = 1789a611-31a0-5614-8853-4961a0d329fd
log_level = INFO
log_file = rdsmonitor.log
servers_limit = 4
create_initial_server = 3
```

After changing create_initial_server to the final value, start the monitoring service again:

```
root@utility-400:~# docker exec -it rds-monitor python3 -u /monitor/src/monitor_rds.py
2026-06-05 14:14:25,889 - monitor.monitor_web.server - INFO - RDS Monitor Web is enabled: True
2026-06-05 14:14:25,890 - __main__ - INFO - Running initial worker to check server group and create servers if needed.
2026-06-05 14:14:25,923 - monitor.workers.initial - INFO - Server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 has no servers, creating new service pool.
2026-06-05 14:14:25,942 - monitor.workers.initial - INFO - Service pool already exists for server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980. Increasing cache.
2026-06-05 14:14:25,962 - monitor.workers.initial - INFO - Waiting for servers to be created in group 3c30501e-d5a7-5189-bcaa-8d4fa5947980... (Attempt 1/12)
2026-06-05 14:14:25,973 - monitor.workers.initial - INFO - Server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 still has no servers, waiting 5 minutes...
```

As shown in the screenshot, the system detects that there are no servers and creates the number specified in rds_monitor.ini:


UDSRDS-202606031115

Summary
Servers
Publications
Scheduled actions
Logs

Servers


Logs


Export CSV


Delete

Filter

1 - 5 of 5

	Creation date	Revision	Unique ID	Friendly name	State
☐	06/05/2026 13:30	2	52:54:00:00:00:00	UDSTTW000	Removed
☐	06/05/2026 13:53	2	52:54:00:00:00:01	UDSTTW001	Removed
☐	06/05/2026 16:14	2	52:54:00:00:00:02	UDSTTW002	Valid
☐	06/05/2026 16:14	2	52:54:00:00:00:03	UDSTTW003	Valid
☐	06/05/2026 16:15	2	52:54:00:00:00:04	UDSTTW004	Valid

After the servers have been created and registered in the server group, they are available for use.

4. Troubleshooting Common Issues

■ RDS-Automation Agent

If a server is deleted from a server group and it was not created through the UDS Monitor machine, it is NOT deleted from the virtualization platform. It remains available to be added again or placed in another server group.

If a server token is deleted, the server is also removed from the server group in which it is registered. If it does not belong to any group, it is removed completely from UDS.

If a server is deleted manually from the virtualization platform, its token should also be deleted to prevent obsolete records.

If a token is deleted manually from UDS administration, the now-unused server should also be deleted from the virtualization platform.

To move an RDS server to another server group, remove it from its current group and add it to the new group.

■ UDS Monitor

When a server group contains no servers, the following monitoring-system message is normal:

```
2026-06-08 09:48:42,579 - monitor.workers.expander - ERROR - Error during data collection for group
3c30501e-d5a7-5189-bcaa-8d4fa5947980: Server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 has no ser
vers
Traceback (most recent call last):
  File "/monitor/src/monitor/workers/expander.py", line 58, in run
    server_group, overload = await asyncio.gather(server_group_task, is_overload_task)
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/monitor/src/monitor/analyze.py", line 50, in is_server_group_overload
    raise EmptyServerGroupError(f"Server group {server_group_id} has no servers")
monitor.analyze.EmptyServerGroupError: Server group 3c30501e-d5a7-5189-bcaa-8d4fa5947980 has no ser
vers
^C2026-06-08 09:49:24,776 - monitor.workers.expander - INFO - SyncWorker run task cancelled cleanly
```

If the token of a server belonging to a pool created automatically by UDS Monitor is deleted but the server remains, check that the server was created more than one hour ago. It is deleted automatically after that period.

If monitoring is run in interactive mode and no events appear, the logging level may be set to ERROR, in which case only error events are recorded. The INFO level is recommended to display all events.

```
root@utility-500:~# docker exec -it rds-monitor python3 -u /monitor/src/monitor_rds.py
2026-06-16 13:27:28,203 - monitor.logsetup - ERROR - Logging at level ERROR (/monitor/src/logs/rdsmonitor.log).
2026-06-16 13:27:28,204 - monitor.logsetup - ERROR - Level ERROR active: INFO/DEBUG will not be shown.
```

About Virtual Cable

Virtual Cable specializes in digital workplace transformation. The company develops, supports, and markets UDS Enterprise. Its team of experts has designed VDI solutions tailored to each industry to provide a unique user experience fully adapted to the needs of every user profile. Virtual Cable's professionals have more than 30 years of experience in IT and software development and more than 15 years in virtualization technologies. Millions of Windows and Linux virtual desktops are deployed with UDS Enterprise worldwide every day.