



VDI with
UDS Enterprise 3.5
and AWS

www.udsenderprise.com



Index

Introduction.....	2
UDS Enterprise on Amazon Web Services	3
Where do I start?	3
Deploying UDS Servers on AWS.....	4
▪ User creation in IAM module.....	4
▪ Creating a Bucket and Role	7
▪ Importing UDS servers.....	10
▪ Creating UDS servers	12
▪ Configuring UDS servers	18
▪ Creating base machines or templates on AWS	21
UDS Enterprise Administration.....	25
AWS service provider integration	25
▪ Creating base services	31
Creating a Service Pool	34
About Virtual Cable	39



Introduction

Amazon Web Services (AWS) is a proprietary platform from Amazon that offers cloud services. Among some of its advanced features, there is the ability to run virtual machines, virtual applications, databases, backups, and many other tasks. It integrates countless cloud services that are needed to develop, test, deploy, and manage virtual machines (VMs).

This **VDI Guide with UDS Enterprise & AWS** will help you understand how to deploy and configure UDS Enterprise components on that platform. This document shows, through real examples, how to create the necessary resources so that UDS Enterprise can deploy virtual desktops on AWS.



UDS Enterprise on Amazon Web Services

Before carrying out the integration, it is recommended to invest some time in learning about the different configurable parts of UDS Enterprise (for more information visit our [website](#). In the [Documentation](#) section you will find the UDS Enterprise Installation, Administration and User Manual). One of them is the Service Providers, an element of great importance for the configuration of AWS in UDS Enterprise.

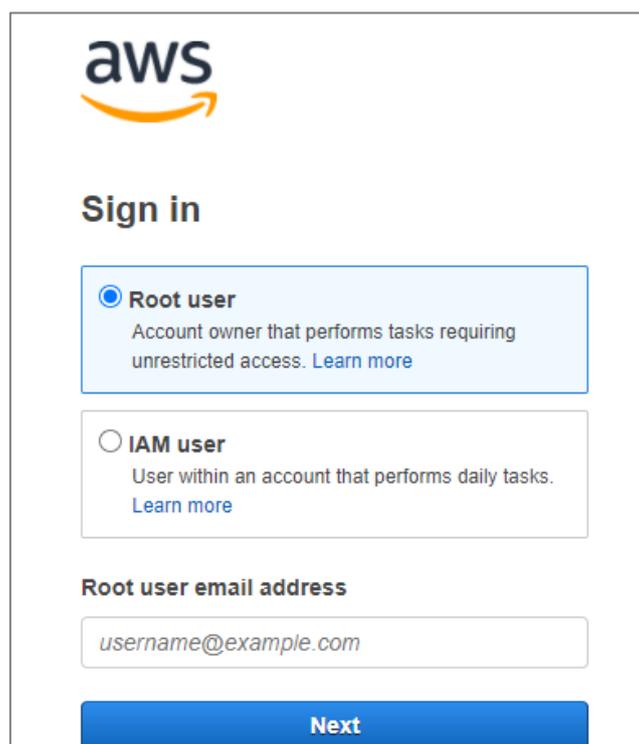
UDS Enterprise will enable the deployment of self-generated virtual desktops and virtual application sessions on the AWS platform. UDS components (Server, Tunnel, and Database) can be hosted in the AWS environment itself or can also be hosted on any other on-premise virtualization platform with connectivity to the AWS environment.

To import, install, and configure UDS Enterprise within an AWS environment, you must request its specific components for this environment (UDS-Server, UDS-Tunnel, and Database) and a serial number (Free/Evaluation/Enterprise) to Virtual Cable.

You must have a valid subscription on AWS with administration permissions on which to deploy UDS Enterprise components, virtual desktops, or Windows/Linux application servers.

Where do I start?

First, you must have an account with administrator privileges (you can use the “Root” account or an IAM account with permissions) on the AWS platform. If you already have it, Log in to the [portal](#).



The screenshot shows the AWS Sign in page. At the top is the AWS logo. Below it is the heading "Sign in". There are two radio button options: "Root user" (selected) and "IAM user". The "Root user" option includes the text "Account owner that performs tasks requiring unrestricted access. [Learn more](#)". The "IAM user" option includes the text "User within an account that performs daily tasks. [Learn more](#)". Below these options is a text input field labeled "Root user email address" containing the placeholder text "username@example.com". At the bottom is a blue "Next" button.



If you already have an active UDS environment (on an on-premises virtual platform or another cloud platform) and you want to integrate it with AWS, you will need to make the necessary configurations at the network level so that there is communication between the UDS servers and the AWS environment. In this case you can go directly to the “**UDS Enterprise Administration**” section.

If you want to host UDS components within the AWS environment, the Virtual Cable team will provide you with those components in a specific format and you will need to perform a series of tasks to import them.

Deploying UDS Servers on AWS

The following is an example of how to deploy the servers that make up the UDS environment on an AWS platform. This guide details the steps to locate and create the UDS-Server component. The same tasks must be performed for the UDS-Tunnel server and the database.

If the version of UDS to be installed is Enterprise, it will be necessary to upload the database server to the platform. If you’re using the UDS Free Edition or UDS Evaluation Edition you don’t need to deploy a database server. You can activate a local database included in the UDS-Server instead, although this configuration will not allow you to upgrade the environment.

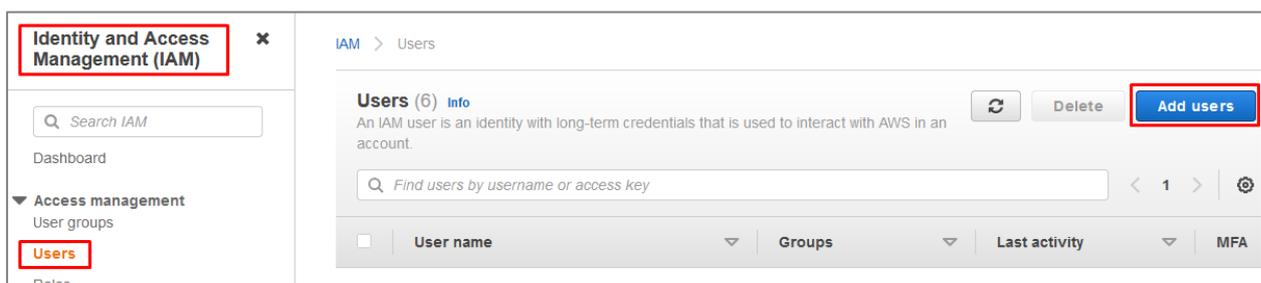
UDS servers will be provided by the Virtual Cable team in disk image format (. ova).

- **User creation in IAM module**

To import the UDS components you will need to have a user account (where you have the “**Access Key ID**” and the “**Secret Access Key**” available) with permissions, within the AWS IAM module. The required permissions will be: “**IAMFullAccess**”, “**AmazonEC2FullAccess**” and “**AmazonS3FullAccess**”.

If you do not have any previously created users or want to configure a specific one to be used by UDS (recommended), the following procedure will be performed:

Access the IAM module in your AWS environment (where have to have full permissions), within the menu select “**Access management**“, the “**Users**“ section, and “**Add users**“:





VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

In the new user creation wizard, indicate a name and select “*Access key – programmatic access*”:

Add user 1 2 3 4 5

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name* UDS-import

[+ Add another user](#)

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type*

- Access key - Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- Password - AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Required [Cancel](#) [Next: Permissions](#)

In the next step of the wizard, configure the necessary permissions that the user must have. You can create a group with specific permissions or assign them directly.

The permissions that the user must have will be: “*IAMFullAccess*”, “*AmazonEC2FullAccess*” and “*AmazonS3FullAccess*”.

The following screenshot shows how to assign them directly, using the option “*Attach existing policies directly*”. Using the policy finder, you check “*IAMFullAccess*”, “*AmazonEC2FullAccess*” and “*AmazonS3FullAccess*”:

Add user 1 2 3 4 5

Set permissions

[Add user to group](#) [Copy permissions from existing user](#) [Attach existing policies directly](#)

[Create policy](#) [Refresh](#)

Filter policies IAMFullAccess Showing 1 result

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	IAMFullAccess	AWS managed	Permissions policy (2)



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Add user 1 2 3 4 5

▼ Set permissions

Add user to group

Copy permissions from existing user

Attach existing policies directly

Create policy

Filter policies Showing 1 result

Policy name	Type	Used as
<input checked="" type="checkbox"/> AmazonEC2FullAccess	AWS managed	Permissions policy (4)

Add user 1 2 3 4 5

▼ Set permissions

Add user to group

Copy permissions from existing user

Attach existing policies directly

Create policy

Filter policies Showing 1 result

Policy name	Type	Used as
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS managed	Permissions policy (2)

Follow the user creation wizard and check that all the data is correct:

Add user 1 2 3 4 5

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name UDS-import

AWS access type Programmatic access - with an access key

Permissions boundary Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AmazonEC2FullAccess
Managed policy	AmazonS3FullAccess
Managed policy	IAMFullAccess

[Cancel](#) [Previous](#) [Create user](#)



Proceed to create the new user with the assigned permissions. At this point it is very important that you copy the user's data: "**Access key ID**" and "**Secret Access key**" (especially the latter, since once the wizard window is closed this data will no longer be available, although it will be possible to generate a new "**Secret Access key**" if necessary).

Add user

1 2 3 4 5

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://950472154737.signin.aws.amazon.com/console>

[Download .csv](#)

User	Access key ID	Secret access key
UDS-import	AKIA52TEODZYTR4LKMU6	89/YAxHFy /Ou38VZbipgyh98VRaDkeO xjwTCoMPb Hide

[Close](#)

■ Creating a Bucket and Role

Once you have a user and the connection data, you must create a new bucket from the "**S3**" module of the AWS environment so that it can later be modified to contain the UDS components.

Access Amazon S3, go to "**Buckets**" and click on "**Create bucket**":

Amazon S3

Buckets

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

Amazon S3

Account snapshot

Last updated: Sep 17, 2021 by Storage Lens. Metrics are generated every 24 hours. [Learn more](#)

Buckets (2) Info

Buckets are containers for data stored in S3. [Learn more](#)

[Refresh](#) [Copy ARN](#) [Empty](#) [Delete](#) [Create bucket](#)



In the wizard, indicate a name, select your region and leave the rest of the options by default:

Amazon S3 > Create bucket

Create bucket [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name

Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Object Ownership [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and granted using access control list ownership determines who can specify access to objects.

<input checked="" type="radio"/> ACLs disabled (recommended) All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.	<input type="radio"/> ACLs enabled Objects in this bucket can be owned by other accounts. Access to this bucket and its objects is specified using ACLs.
--	--



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Create the bucket that will host the UDS servers:

The screenshot shows the Amazon S3 console interface. On the left sidebar, the 'Buckets' menu item is highlighted with a red box. The main content area displays the 'Buckets (3)' section, which includes a search bar containing 'bucket-uds'. Below the search bar, there is a table with the following data:

Name	AWS Region	Access
bucket-uds	EU (Frankfurt) eu-central-1	Bucket and objects not public

Now you have an application that will allow you to create a role and modify the bucket previously created in the AWS environment, with all the necessary configurations and permissions to import the UDS components.

Download the following application:

https://images.udsenderprise.com/files/AWS/UDS_Import/setup_vmimport_role.zip

Extract the.zip file and execute it by command line on a computer with Windows OS with the following parameters:

- **-a** : Access key of the user with permissions indicated in the previous point.
- **-s** : Secret Access Key of the user.
- **-b** : Name of the **"bucket"** that will be created in the AWS environment (S3) and that will serve to store the UDS servers.
- **-n** : Name of the role to be created in the AWS Environment (IAM) to allow the import of UDS servers.

```
C:\Windows\system32\cmd.exe
C:\>
C:\>setup_vmimport_role.exe -a AKIA52TE0DZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbipgyh98VRaDkeOxjwTCoMPb
-b bucket-uds -n rol-uds
```



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Once the command is executed, you can see how the role has been created in the IAM module of the AWS environment:

The screenshot shows the AWS IAM console interface. On the left, the 'Identity and Access Management (IAM)' sidebar is visible, with the 'Roles' option highlighted in a red box. The main content area displays the 'Roles (9)' page, with a search bar containing 'rol-uds'. Below the search bar, a table lists the role details:

<input type="checkbox"/>	Role name	Trusted entities
<input type="checkbox"/>	rol-uds	AWS Service: vmie

■ Importing UDS servers

To import the UDS components, you must have their images in .ova format:

The screenshot shows a file explorer window with the following items:

- Parent Directory
- UDS-Dbserver-aws.3.5.0.ova
- UDS-Server-aws.3.5.0.ova
- UDS-Tunnel-aws.3.5.0.ova

Once downloaded, you should run an application that will upload the UDS component to the indicated bucket of the AWS environment.

Download the following application:

https://images.udsenderprise.com/files/AWS/UDS_Import/import_uds_appliance.zip

Extract the .zip file and execute it by command line on a computer with Windows OS with the following parameters:

- **-a** : Access key of the user with permissions indicated in the previous point.
- **-s** : Secret Access Key of the user.
- **-b** : Name of the bucket that will be created in the AWS environment (S3) and that will serve to store the UDS servers.
- **-n** : Name of the role to be created in the AWS Environment (IAM) to allow the import of UDS servers.
- **-f** : Path of the UDS component to import (you can also indicate cloud repositories here, for example: **-f** <https://images.udsenderprise.com/3.5/stable/aws/UDS-Server-aws.3.5.0.ova>).



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

```
C:\Windows\system32\cmd.exe
C:\>import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbipgyh98VRaDkeOxjwTCoMPb
-b bucket-uds -n rol-uds -f "F:\UDS Enterprise 3.0\UDS-Server-aws.3.5.0.ova"
```

Once executed, wait for it to be uploaded:

```
C:\Windows\system32\cmd.exe - import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbip...
C:\>import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbipgyh98VRaDkeOxjwTCoMPb
-b bucket-uds -n rol-uds -f "F:\UDS Enterprise 3.0\UDS-Server-aws.3.5.0.ova"
Uploading UDS-Server-aws.3.5.0.ova [=====] 4%
```

And the machine is imported:

```
C:\Windows\system32\cmd.exe - import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbip...
C:\>import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbipgyh98VRaDkeOxjwTCoMPb
-b bucket-uds -n rol-uds -f "F:\UDS Enterprise 3.0\UDS-Server-aws.3.5.0.ova"
Uploading UDS-Server-aws.3.5.0.ova [=====] 100%
Task ID: import-ami-083b5445a3457c364
Importing [=====] 19% \ (State: converting)
```

This last phase of the process can take several minutes. At this point the server imported into the bucket is converted and generates an AMI.

Once the process is finished, you'll have the UDS server as an AMI:

```
C:\Windows\system32\cmd.exe
C:\>import_uds_appliance.exe -a AKIA52TEODZYTR4LKMU6 -s 89/YAxHFy/Ou38VZbipgyh98VRaDkeOxjwTCoMPb
-b bucket-uds -n rol-uds -f "F:\UDS Enterprise 3.0\UDS-Server-aws.3.5.0.ova"
Uploading UDS-Server-aws.3.5.0.ova [=====] 100%
Task ID: import-ami-083b5445a3457c364
Importing [=====] 51% / (State: preparing ami)
Importing [=====] 51% - (State: preparing ami)
Importing [=====] 51% \ (State: preparing ami)
Importing [=====] 51% | (State: preparing ami)
Importing [=====] 51% / (State: preparing ami)
Importing [=====] 51% - (State: preparing ami)
Importing [=====] 57% \ (State: preparing ami)
Importing [=====] 58% | (State: preparing ami)

File "UDS-Server-aws.3.5.0.ova" deleted from s3 bucket bucket-uds
AMI ID: ami-0ee93b81cde8a4262
AMI name set to "UDS-Server-aws.3.5.0.ova"
Done
C:\>
```



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

It will be available in the “**EC2**” dashboard of the AWS environment:

Amazon Machine Images (AMIs) (1) Info

Owned by me Search

AMI ID = ami-0ee93b81cde8a4262 X Clear filters

<input type="checkbox"/>	Name	AMI ID	AMI name
<input type="checkbox"/>	UDS-Server-aws.3.5.0.ova	ami-0ee93b81cde8a4262	import-ami-083

Select an AMI.

This import process must be repeated with the rest of UDS components:

Amazon Machine Images (AMIs) (8) Info

Owned by me Search

<input type="checkbox"/>	Name	AMI ID
<input type="checkbox"/>	UDS-Tunnel-aws.3.5.0.ova	ami-0612f4a63612
<input type="checkbox"/>	UDS-Dbserver-aws.3.5.0.ova	ami-071b823da11
<input type="checkbox"/>	UDS-Server-aws.3.5.0.ova	ami-0ee93b81cde8

■ Creating UDS servers

The next step in deploying the UDS components is to create the virtual instance that will contain the UDS servers, based on the AMIs imported in the previous step.

Within the “**EC2**” module of the AWS environment, access the “**AMIs**” section, select the UDS component and click on “**Launch instance from image**”.



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Spot Requests
Savings Plans
Reserved Instances **New**
Dedicated Hosts
Capacity Reservations
▼ **Images**
 AMIs **New**
 AMI Catalog

Amazon Machine Images (AMIs) (1/8) Info

Owned by me ▼ Search

<input type="checkbox"/>	Name	AMI ID
<input type="checkbox"/>	UDS-Tunnel-aws.3.5.0.ova	ami-0612f4a6
<input type="checkbox"/>	UDS-Dbserver-aws.3.5.0.ova	ami-071b823c
<input checked="" type="checkbox"/>	UDS-Server-aws.3.5.0.ova	ami-0612f4a6
<input type="checkbox"/>	xUbuntu20-UDS-I	
<input type="checkbox"/>	UDS-Tunnel-aws.3.5.0.ova	

Launch instance from image
Copy AMI
Edit AMI permissions

In the instance creation wizard, you must choose the appropriate type for each UDS component. The minimum requirements for all components are (DB server, UDS-Server and UDS-Tunnel): 2vCPUs and 2 GB of RAM.

Step 2: Choose an Instance Type

Amazon EC2 provides a wide selection of instance types optimized to fit different workloads. Instances are virtual servers that can run applications. They have varying amounts of vCPUs, memory, storage, and networking capacity, and give you the flexibility to choose the right amount of resources for your applications. [Learn more](#) about instance types and their computing needs.

Filter by: **All instance families** ▼ **Current generation** ▼

Currently selected: t2.medium (- ECUs, 2 vCPUs, 2.3 GHz, -, 4 GiB)

	Family	Type	vCPUs	Memory (GiB)
<input type="checkbox"/>	t2	t2.nano	1	0.5
<input type="checkbox"/>	t2	t2.micro Free tier eligible	1	1
<input type="checkbox"/>	t2	t2.small	1	2
<input checked="" type="checkbox"/>	t2	t2.medium	2	4



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

In the next step of the wizard, configure the details of the instance. You must indicate a valid network and subnet at least, to allow communication with other elements, and assign a public IP to have external access:

1. Choose AMI | 2. Choose Instance Type | **3. Configure Instance** | 4. Add Storage | 5. Add Tags | 6. Configure Security Group | 7. Review

Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances ⓘ [Launch into Auto Scaling Group](#) ⓘ

Purchasing option ⓘ Request Spot instances

Network ⓘ [Create new VPC](#)

Subnet ⓘ [Create new subnet](#)
247 IP Addresses available

Auto-assign Public IP ⓘ

Hostname type ⓘ

DNS Hostname ⓘ Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

NOTE:

The database server will not need to have a public IP.

In step 4 of the wizard, indicate the type of storage:

Step 4: Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type ⓘ	Device ⓘ	Snapshot ⓘ	Size (GiB) ⓘ	Volume Type ⓘ	IOPS ⓘ	Throughput (MB/s) ⓘ	Delete on Termination ⓘ	Encr
Root	/dev/sda1	snap-0175971180179588f	<input type="text" value="8"/>	General Purpose SSD (gp2)	100 / 3000	N/A	<input type="checkbox"/>	Not E

[Add New Volume](#)

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

In step 6 of the wizard, create or select a “**Security Group**” with the necessary rules for each UDS component. Only UDS-Server and UDS-Tunnel servers will need input rules:

- **UDS server.** Port: 80/443 (for user and administrator access).

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a **new** security group
 Select an **existing** security group

Security group name:
Description:

Type	Protocol	Port Range	Source	Description
All TCP	TCP	0 - 65535	Custom 10.0.0.0/16	Internal Connectivity
HTTPS	TCP	443	Custom 0.0.0.0/0	Users Access

- **Tunnel server.** Port: 443 and 10443 (443 for tunneled connections and 10443 for HTML5 connections).

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a **new** security group
 Select an **existing** security group

Security group name:
Description:

Type	Protocol	Port Range	Source	Description
All traffic	All	0 - 65535	Custom 10.0.0.0/16	Internal Connectivity
Custom TCP	TCP	443	Custom 0.0.0.0/0	Tunnel Connection
Custom TCP	TCP	10443	Custom 0.0.0.0/0	Guacamole HTML5



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

▪ Observer:

Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: Create a **new** security group
 Select an **existing** security group

Security group name:

Description:

Type ⁱ	Protocol ⁱ	Port Range ⁱ	Source ⁱ	Description ⁱ
All traffic ^v	All	0 - 65535	Custom ^v 10.0.0.0/16	Internal Connectivity ^x

Check that all the data is correct and launch the instance:

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

▼ **AMI Details** [Edit AMI](#)

 **import-ami-083b5445a3457c364 - ami-0ee93b81cde8a4262**
AWS-VMImport service: Linux - Debian GNU/Linux 11 (bullseye) - 4.15.0-041500-generic
Root Device Type: ebs Virtualization type: hvm

▼ **Instance Type** [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.medium	-	2	4	EBS only	-	Low to Moderate

▼ **Security Groups** [Edit security groups](#)

Security group name: UDS-Server
Description: UDS Access Portal

Type ⁱ	Protocol ⁱ	Port Range ⁱ	Source ⁱ	Description ⁱ
HTTPS	TCP	443	0.0.0.0/0	Users Access

▶ **Instance Details** [Edit instance details](#)



Before, create or chose a “*Key pair*”:

Select an existing key pair or create a new key pair

A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance. Amazon EC2 supports ED25519 and RSA key pair types.

Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about [removing existing key pairs from a public AMI](#).

Choose an existing key pair ▼

Select a key pair

uds-server | RSA ▼

I acknowledge that I have access to the corresponding private key file, and that without this file, I won't be able to log into my instance.

Cancel Launch Instances

Once the instance is launched, you can access the “*Instances*” section of the “*EC2*” module of the AWS environment and visualize the creation of the UDS component:

The screenshot shows the AWS Management Console interface for the EC2 Instances page. The left sidebar contains navigation options like 'EC2 Dashboard', 'EC2 Global View', 'Events', 'Tags', 'Limits', and 'Instances'. The main content area displays a table of instances. The table has columns for 'Name', 'Instance ID', 'Instance state', and 'Instance type'. One instance is highlighted with a red box, indicating it is in a 'Pending' state.

Name	Instance ID	Instance state	Instance type
xUbuntu20-UDS-PLT	i-0bb367e400acefd8a	Stopped	t2.small
-	i-0f49b4268988dc06a	Pending	t2.medium



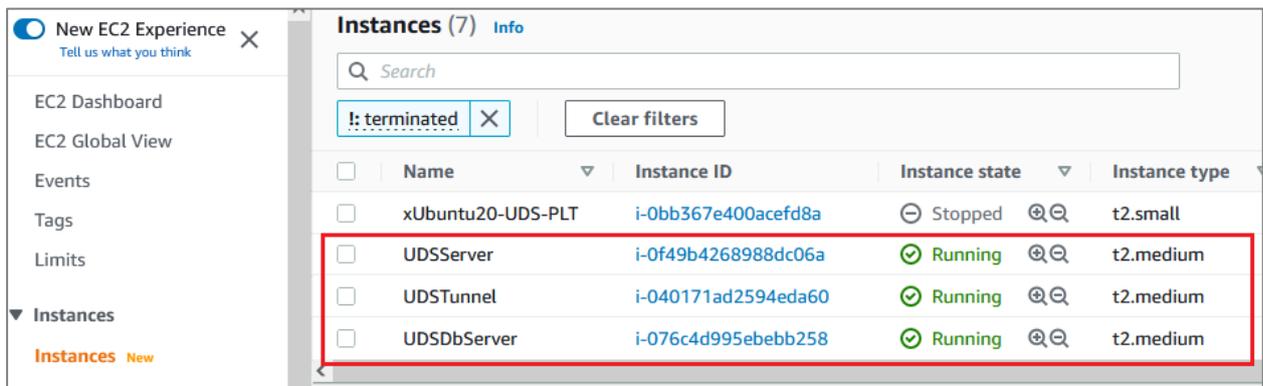
VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Once started, you can indicate a descriptive name for the server:



Repeat the process with all UDS servers:



▪ Configuring UDS servers

Once you have all the UDS components instantiated, proceed to their configuration.

This configuration example is based on a virtual machine deployed on the same network as the UDS servers to have direct connectivity with them.

○ Database configuration

If you are using the database provided by the Virtual Cable team, it will already be pre-configured and you will only have to verify that you have IP connectivity (by default the network is configured by DHCP).

The default server credentials are:

- **User:** root
- **Password:** UDS

This server has a ready-to-use DB instance with UDS Enterprise with the following data:

- **Instance Name:** UDS
- **User:** UDS
- **Password:** UDS



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

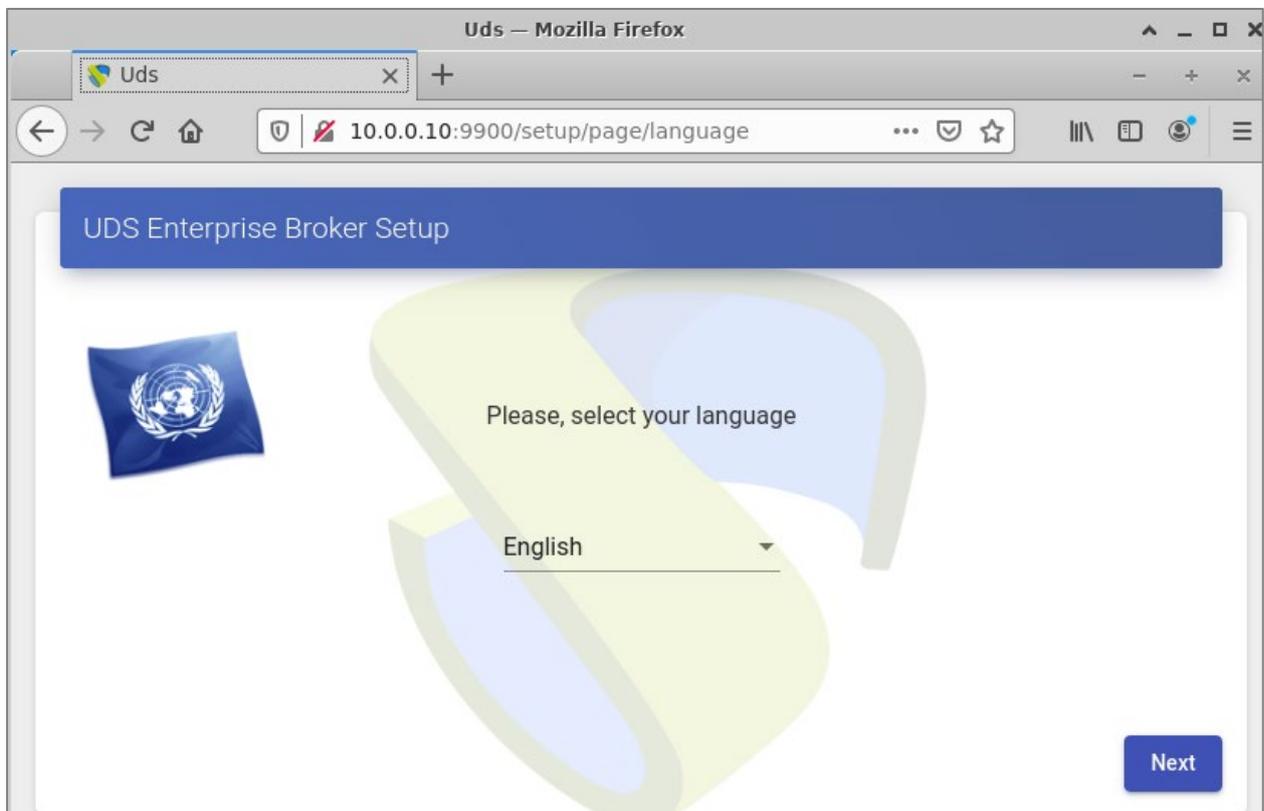
By default, the server has its network configuration via DHCP. It is advisable to always use static addressing in all UDS components.

- o UDS Server configuration

The UDS-Server component is the main element of the UDS environment. Before accessing this configuration wizard, you need to confirm that the server has an IP address via DHCP assigned.

Once you know the IP address assigned to the server (selecting the instance, in the section “*Private IPv4 addresses*”), access via browser to the IP address of the UDS server with port 9900.

`http://IP_Server:9900`



Here, indicate all the necessary data (IP data, serial to activate the subscription, credentials, etc.) to configure the server.

For more information on UDS server configuration, go to the UDS Enterprise installation, administration, and user manual available in the [Documentation](#) section of the website.

NOTE:

During the wizard configuration procedure, you will be asked for the configuration data of **the database server**. In the case of using an external server, you must indicate the data **of the database server** configured previously (IP address, instance, username and password).



VDI with UDS Enterprise 3.5 & Amazon Web Services

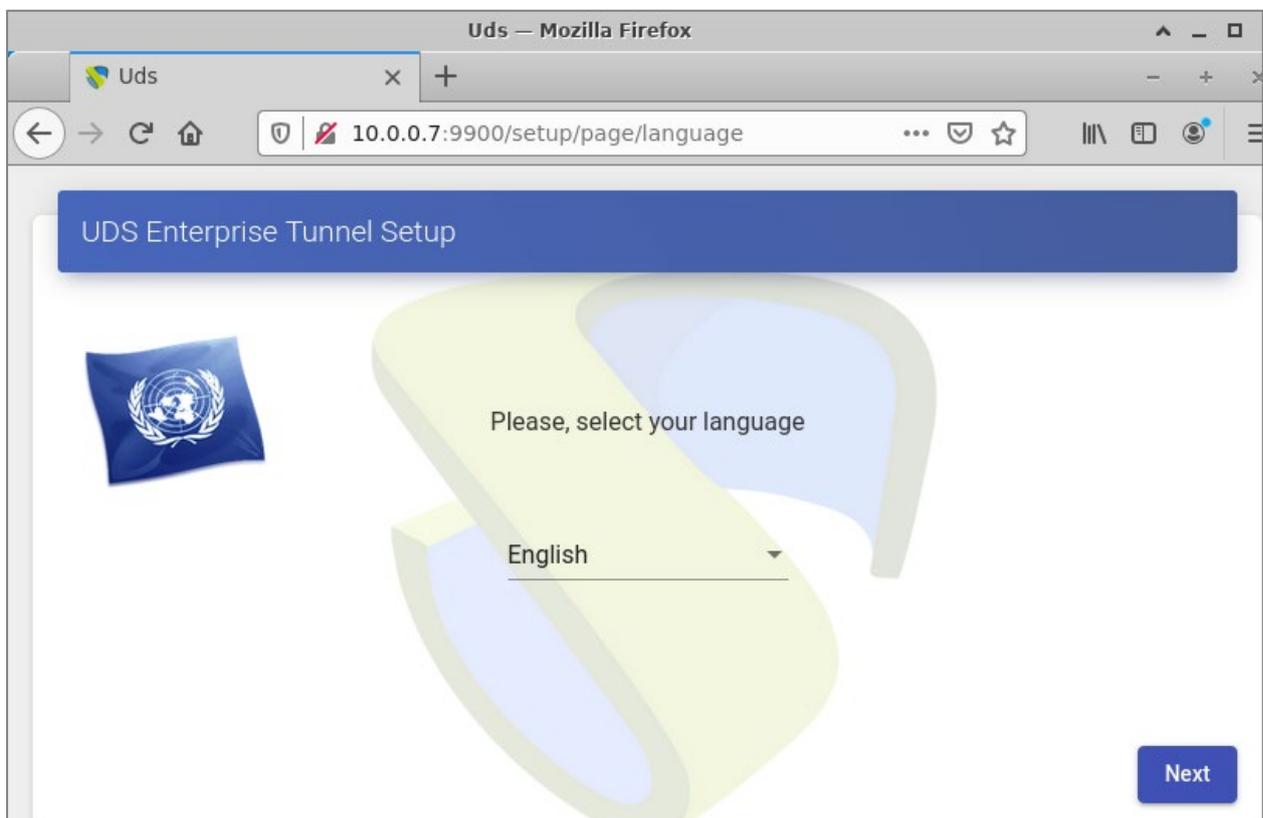
www.udsenderprise.com

- o UDS Tunnel configuration

The UDS-Tunnel component is the element that will provide you with secure access to virtual desktops over the Internet. It will also be responsible for establishing the HTML5 connection (HTML5 Transport for desktops and vApps). It has a configuration wizard accessible via web browser. Before accessing this configuration wizard, you need to confirm that the server has an IP address assigned.

Once you know the IP address assigned to the server (selecting the instance, in the section “*Private IPv4 addresses*”), access via browser to the IP address of the UDS- Server with port 9900

`http://IP_Tunnel:9900`



Here you can indicate all the necessary data (IP data, credentials, certificates, etc.) to configure the server.

For more information on configuring the UDS-Tunnel server, go to the UDS Enterprise installation, administration, and user manual in the [Documentation](#) section of our website.

NOTE:

During the configuration procedure of the wizard the system will ask you for the connection data of the UDS server.



- **Creating base machines or templates on AWS**

For UDS to deploy virtual desktops on the AWS platform, it is necessary to have a base machine or template on which the new desktops self-generated by UDS will be based. This base machine can be deployed in different ways. Among them, it is possible to import an existing template on another platform (using the same applications that you have used to import the different UDS components) or rely on the preconfigured machines (AMIs), offered by the AWS environment itself.

If you choose to import a template, it is important that you make sure that it will have a unique access mode (SSH or RDP type), in order to access it once it is hosted on the AWS platform (this platform does not have a console to be able to manage, configure and modify the machines).

Another important point to keep in mind is the network configuration. It is necessary that it is configured to take IP address via DHCP.

- **Base machine access and configuration**

Once the base machine or template has been deployed and it is accessible via RDP, for example, you must install all the software that you need to have available on the virtual desktops deployed by UDS, perform the optimization configurations of the template, which are very important for the good performance of the machines (disable unnecessary services, optimization of the start time, etc...) and finally perform the installation of the UDS Actor.

NOTE:

For more information on installing the UDS Actor, have a look at the UDS Enterprise Installation, Administration and User Manual in the [Documentation](#) section of the UDS Enterprise website



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

During the configuration of the UDS Actor you must indicate in the connection data against UDS Server the local DNS address/name or IP or public DNS depending on the type of deployment (in the case of using IP addresses instead of names you have to make sure that these addresses are not dynamic and change when the servers are turned on/off).

UDS Actor Configuration Tool

UDS Server **Advanced**

SSL Validation: Ignore certificate

UDS Server: 10.0.0.10

Authenticator: Administration

Username: udsadmin

Password:

Register with UDS | Test configuration | Close

NOTE:

In order to visualize the configuration of the UDS Actor in an Ubuntu O.S. through RDP, execute the following command from a console:

Ubuntu 18: `sudo QT_X11_NO_MITSHM=1 /usr/sbin/UDSActorConfig`

Ubuntu 20: `xhost + && sudo QT_X11_NO_MITSHM=1 /usr/sbin/UDSActorConfig`



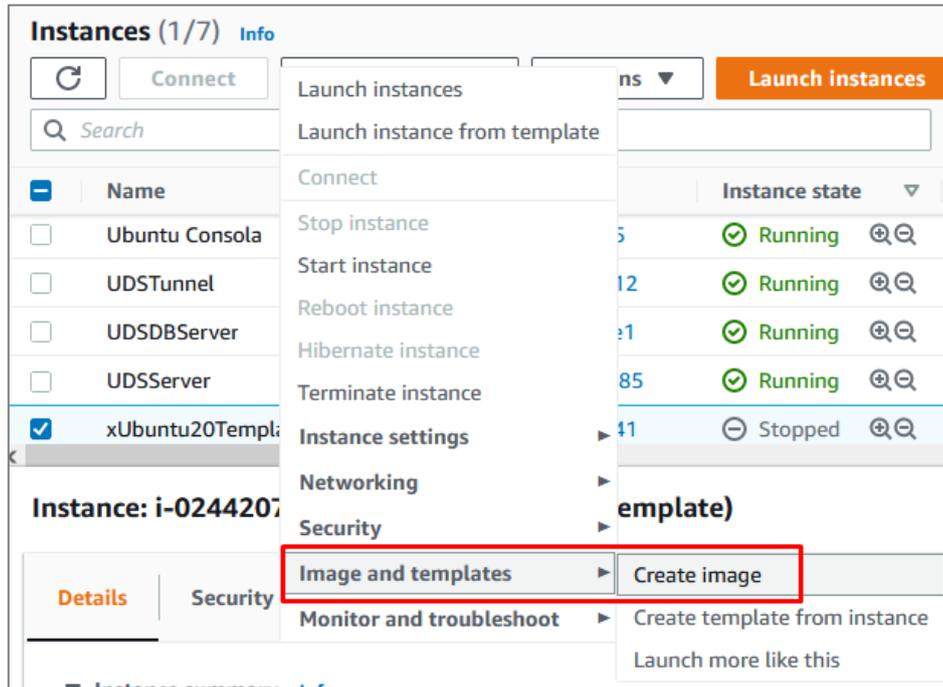
VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

o AMI Creation

Once the UDS Actor has been configured and installed, **you will be able to turn off the base machine or template** and create the AMI that you will use in UDS to generate the virtual desktops to which the users will make the connection.

After turning off the base machine or template, select it, click on “*Actions*”, “*Image and templates*” and “*Create image*”:



In the image creation wizard, indicate a descriptive name for the AMI, (it will be the one that you can visualize from the UDS administration console) and mark the “*Delete on termination*” option. Click on “*Create Image*” to generate the AMI:



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Create image [Info](#)

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID
i-024420780cc630741 (xUbuntu20Template)

Image name
xUbuntu20-UDS-IMG (xubuntu-borra)

Maximum 127 characters. Can't be modified after creation.

Image description - optional
Image description

Maximum 255 characters

No reboot
 Enable

Instance volumes

Volume type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination
EBS	/dev/s...	Create new snapshot fr...	20	EBS General Purpose SS...	100		<input checked="" type="checkbox"/> Enable

[Add volume](#)

NOTE:

To prevent orphaned volumes from being left on the platform, you must select the “Delete on termination” option.

After finishing the creation of the image, you will have it available in the AMIs section and you can rename it with a descriptive name:

Savings Plans
Reserved Instances **New**
Dedicated Hosts
Capacity Reservations

▼ Images
AMIs **New**
AMI Catalog
▼ Elastic Block Store

Amazon Machine Images (AMIs) (1/5) [Info](#)

Owned by me

Name	AMI ID	AMI name
UDS-Dbserver-aws.3.5.0.ova	ami-01fd17e380c723c3c	import-ami-0881dc14e038fbf95
UDS-Tunnel-aws.3.5.0.ova	ami-029f27757281c849f	import-ami-09ee01f2e2572fd6c
UDS-Server-aws.3.5.0.ova	ami-07e4f6954f66b8222	import-ami-0cbf1ae32401db173
Xubuntu20-UDS-Template.ova	ami-0b76ac0810e052be9	import-ami-0b3fe26cc2d9a9c13
<input checked="" type="checkbox"/> xUbuntu20-UDS-IMG	ami-0ed18ffa7c80938d3	xUbuntu20-UDS-IMG (xubuntu-borra)

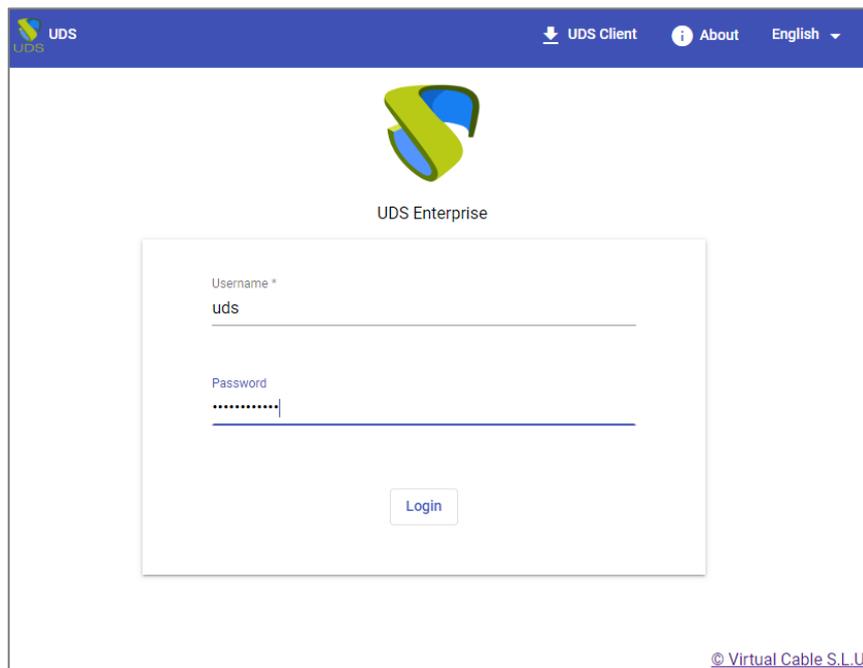
Once you have the image (AMI), you can access the UDS administration to continue with the process of configuration and deployment of VDI services.



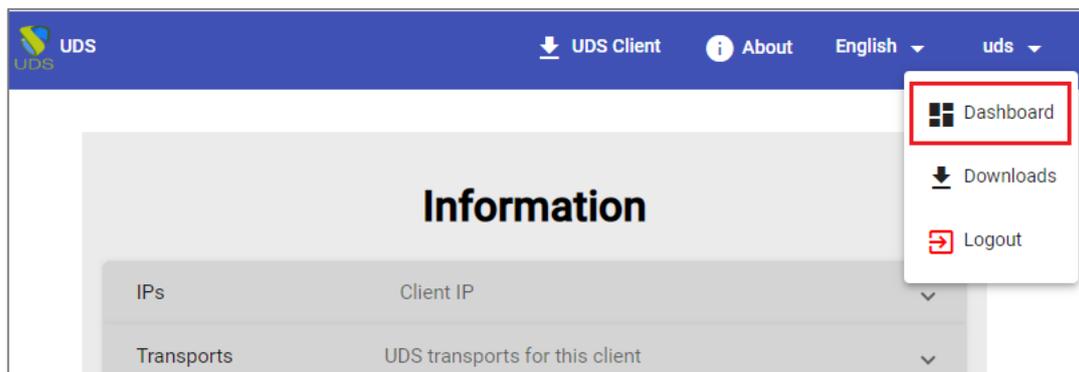
UDS Enterprise Administration

AWS service provider integration

To perform the integration of AWS as a UDS Enterprise service provider, enter the UDS administration. Access via web browser the IP address or name of the UDS Server component and validate with an administrator user (in the first access, use the system administrator user indicated in the UDS-Server configuration wizard).



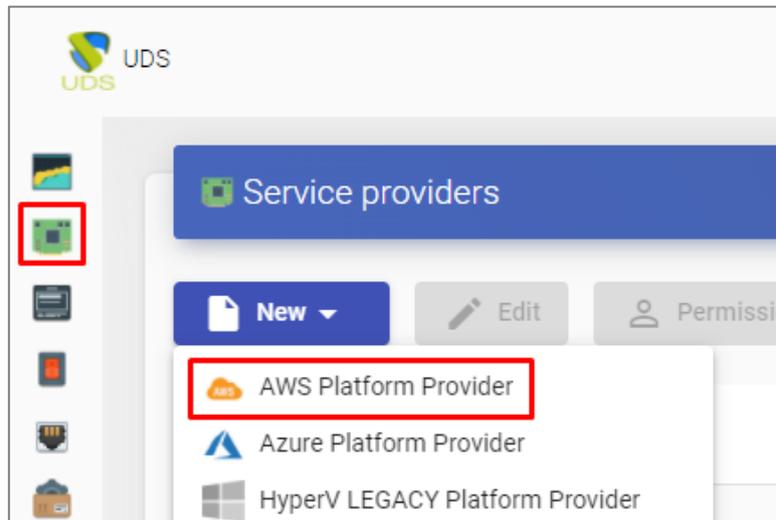
Once validated in the UDS login portal, access the **“Dashboard”** from the user menu.





VDI with UDS Enterprise 3.5 & Amazon Web Services

Within the UDS administration, access the “*Services*” menu and click on “*New*” to register a new “*Service provider*”. Select “*AWS Platform Provider*”:



In order for UDS to connect to the AWS platform and be able to automatically deploy virtual desktops, it will be necessary to indicate the following data:

New provider

Main Advanced

Tags
Tags for this element

Name *
Amazon Web Services

Comments
Comments for this element

Access Key ID *
Obtained from user created on AWS IAM for UDS Enterprise

Secret Access Key *
Obtained from user created on AWS IAM for UDS Enterprise - Keys



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

- **Main:** Enter a descriptive name, and the connection data with the AWS account. To obtain an “*Access Key ID*” and a “*Secret Access Key*” you will need to create a new user (or use an existing one) in the IAM module of the AWS console.

NOTE:

It is possible to use the same user that you have used to import the UDS machines, as long as you have all the data. In this example we are going to create a new user.

To create a new user, access the IAM module by selecting the menu “*Users*” and clicking on “*Add users*”:

The screenshot shows the AWS IAM console interface. On the left, the 'Identity and Access Management (IAM)' sidebar is visible, with the 'Users' option under 'Access management' highlighted with a red box. The main content area shows the 'Users (5)' page, with the 'Add users' button in the top right corner highlighted with a red box. The page includes a search bar, a table of users, and various action buttons like 'Delete' and 'Add users'.

In the new user creation wizard, indicate a name and in “*Select AWS credential type*”, choose the option “*Access key – Programmatic access*”:

The screenshot shows the 'Add user' wizard in the AWS IAM console. The first step is 'Set user details', where the 'User name*' field contains 'uds' and is highlighted with a red box. Below it is a '+ Add another user' button. The second step is 'Select AWS access type', where the 'Access key - Programmatic access' option is selected and highlighted with a red box. The description for this option states: 'Enables an access key ID and secret access key for the AWS API, CLI, SDK, and other development tools.' The 'Password - AWS Management Console access' option is unselected.



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

In the next step of the new user creation wizard, select the permissions. You can assign the user to a group with the permission assigned: “*AmazonEC2FullAccess*” or directly assign this permission to the user, as shown in the following screenshot:

Add user (1, 2, 3, 4, 5)

▼ Set permissions

Add user to group Copy permissions from existing user Attach existing policies directly

Create policy ↻

Filter policies Showing 1 result

	Policy name	Type	Used as
<input checked="" type="checkbox"/>	AmazonEC2FullAccess	AWS managed	Permissions policy (3)

In step 3, if necessary, you can add tags for the user.

Add user (1, 2, 3, 4, 5)

Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key	Value (optional)	Remove
<input type="text" value="Add new key"/>	<input type="text"/>	<input type="button" value="Remove"/>

You can add 50 more tags.



Finally, check that all the data are correct and proceed to the creation of the new user:

Add user

1 2 3 **4** 5

Review

Review your choices. After you create the user, you can view and download the autogenerated password and access key.

User details

User name	uds
AWS access type	Programmatic access - with an access key
Permissions boundary	Permissions boundary is not set

Permissions summary

The following policies will be attached to the user shown above.

Type	Name
Managed policy	AmazonEC2FullAccess

Tags

No tags were added.

[Cancel](#) [Previous](#) [Create user](#)

Add user

1 2 3 4 **5**

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://950472154737.signin.aws.amazon.com/console>

[Download .csv](#)

	User	Access key ID	Secret access key
▶	✓ uds	AKIA52TEODZY3OHFILVM	***** Show



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Once created, you already have the “*Access key ID*” and the “*Secret Access key*” to add it in the configuration of the service provider.

	User	Access key ID	Secret access key
▶	✓ uds	AKIA52TEODZY3OHFILVM	K7hCH+NdFQuhuOab52/k9+r67SRhqXkwyg+ZmkHf Hide

Keep in mind that the “*Secret Access key*” cannot be consulted again, although you can generate a new one.

New provider

Main Advanced

Tags
Tags for this element

Name *
Amazon Web Services

Comments
Comments for this element

Access Key ID *
AKIA52TEODZY3OHFILVM

Secret Access Key *
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

[Test](#) [Discard & close](#) [Save](#)

- **Advanced:** Indicate the concurrency of creation and deletion, the timeout of the connection, if necessary, a proxy server (for communication between the UDS-Server and AWS) and the region of your EC2 environment.

New provider

Main Advanced

Creation concurrency *
30

Removal concurrency *
15

Timeout *
30

Proxy
Proxy used for connection to AWS (use PROTOCOL://host:port, i.e. http://host:port)

Default region *
eu-central-1

[Test](#) [Discard & close](#) [Save](#)



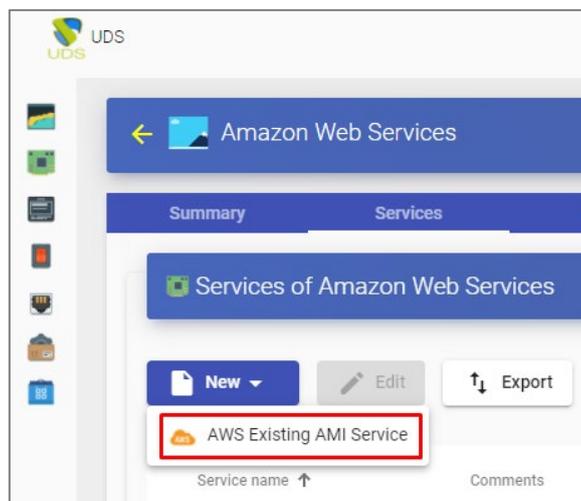
Perform a connection test with the service provider to confirm the correct integration and save.

NOTE:

Even if the test is not correct, you can save the provider and thus not lose the indicated data.

- **Creating base services**

When you have a valid “*Service provider*” connected to the AWS platform, you can create services based on “Amazon **Machine Images**” (AMIs). To do this, access the provider (with double click or right button – “*Detail*”) and in the “*Services*” tab click on “*New*” – “*AWS Existing AMI Service*”.





VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

To create a “*AWS Existing AMI Service*” base service, indicate the following data:

- Main:
 - **Name:** Descriptive name of the base service.
 - **Region:** Location of the Amazon EC2 environment to work on.
 - **AMI:** Base machine image or template to deploy virtual desktops (with the UDS Actor installed and configured).
 - **Instance type:** Number of resources that the virtual desktops automatically deployed by UDS will have (in this list all the types of available machines will be shown. Therefore, you have to make sure that the type chosen is the right one for the service to be deployed).
 - **Key pair:** A set of security credentials that is used to prove your identity when you connect to an Amazon EC2 instance.
 - **Machine Names:** Root name that the virtual desktops generated by UDS will have.
 - **Name Length:** Number of digits of the counter for UDS machines. These digits will be joined to the “*machine names*” to form the DNS name of the virtual desktops (with 1 digit you can create 9 machines, with 2, 99, with 3, 999, etc).

New service

Main Network

Tags
Tags for this element

Name *
xUbuntu20

Comments
Comments for this element

Region *
eu-central-1

AMI *
xUbuntu20-UDS-IMG (xUbuntu-borra)

Instance type *
t2.micro (1 cpus, 1024 MB, i386,x86_64, 2.5 GHz)

Key pair *
UDSServers (c8:e8:4f:f8:4a:bb:c7:21:79:d6:14:ec:de:38:76:22:14:be:0c:d6)

Machine Names *
xUbuntu-

Name Length *
3

Discard & close Save



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

- Network:
 - **VPC:** Existing virtual network of the AWS environment and to which the virtual desktops will be connected.
 - **Subnetwork:** Existing subnet to which the virtual desktops will be connected.
 - **Security Group:** A security group that will be assigned to the virtual desktops.

New service

Main **Network**

VPC *
VPC-10-16 (10.0.0.0/16) ▼

Subnetwork *
Public subnet/subnet-062bacaefd3fa0088 (10.0.0.0/24 on eu-central-1c) ▼

Security groups *
VDIs (Escritorios generados por UDS) ▼

Save and you will already have a valid base service to automatically deploy virtual desktops on AWS:

Amazon Web Services

Summary **Services** Usage Logs

Services of Amazon Web Services

New Edit Export Delete Filter

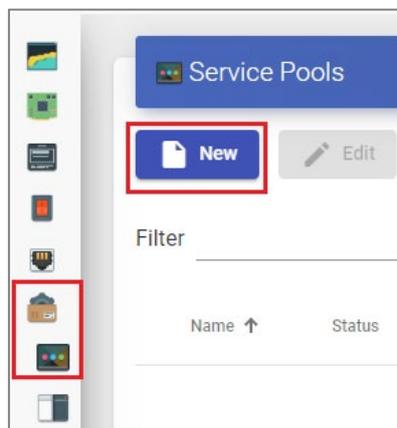
Service name ↑	Comments	Type	Serv
<input type="checkbox"/> xUbuntu20		AWS Existing AMI Service	0



Creating a Service Pool

Before proceeding to create a services pool (to publish virtual desktops), it will be necessary to have at least one **“Authenticator”** with user groups (to validate and be able to assign the service to users), an **“OS Manager”** (to indicate the O.S. and persistence policy of the generated desktops) and a **“Transport”** (to make the connection to the desktop) previously configured. To see more details of how to configure these elements you can access the Installation, Administration and User Manual of UDS Enterprise in the [Documentation](#) section of our website.

When you have the elements mentioned above (**“Authenticator”**, **“OS Manager”** and **“Transport”**), you can create **“Service Pools”**. Access the **“Pools”** section, open the **“Service Pools”** tab and click on **“New”**.



In the **“Main”** tab indicate the name of the service (this name will be visible to users) and select the base service created previously (in this case of the AWS silver form and the xUbuntu20 base service) and an existing **“OS Manager”** (in this example one will be used for Linux O.S. and non-persistent type).

New service Pool

< Main Display Advanced >

Tags
Tags for this element

Name *
Desktop xUbuntu

Short name
Short name for user service visualization

Comments
Comments for this element

Base service
Amazon Web Services\xUbuntu20

OS Manager
Linux no persistente

Publish on creation
 Yes

Discard & close Save



VDI with UDS Enterprise 3.5 & Amazon Web Services

The parameters of the "Advanced" and "Display" tabs can be left by default. In the "Availability" tab indicate the initial desktops that UDS will generate and the ones in cache.

In this example, we have indicated UDS to automatically create 4 desktops and always have at least 2 available in cache.

The screenshot shows the 'New service Pool' configuration window with the 'Availability' tab selected. The parameters are as follows:

Parameter	Value
Initial available services	4
Services to keep in cache	2
Services to keep in L2 cache	0
Maximum number of services to provide	15

Buttons at the bottom: Discard & close, Save

By selecting the "Service Pool" and opening the "Publications" tab, check if the publication has been generated correctly. When it is in a "Valid" state, the system will start autogenerating the virtual desktops indicated in the cache parameters.

The screenshot shows the 'Desktop xUbuntu' interface with the 'Publications' tab selected. The 'Publications' section contains a table with the following data:

Revision	Publish date	State	Reason
1		Valid	

Buttons: New, Cancel, Changelog, Export



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

In the “Cache” tab you can see how the desktops start to be generated.

Creation date	Revision	Unique ID	IP	Friendly name	State	Cache level	Actor version
	1	0a:5e:48:93:bb:0a	unknown	xUbuntu-000	Waiting OS	1	3.5.0
	1	0a:bc:a2:76:2a:0a	10.0.0.35	xUbuntu-001	Valid	1	3.5.0
	1	0a:b0:79:67:ff:76	unknown	xUbuntu-002	Waiting OS	1	3.5.0
	1	0a:bd:82:83:8f:ee	unknown	xUbuntu-003	Waiting OS	1	3.5.0

In the AWS environment also see how virtual desktops are generated:

Name	Instance ID	Instance state	Instance type	Status check
xUbuntu20-UDS-PLT	i-0bb367e400acefd8a	Stopped	t2.small	-
UDS-DBServer	i-001de04eaceac33e5a	Running	t2.micro	2/2 checks passed
UDS-Server	i-0c016ec56d530e782	Running	t2.micro	2/2 checks passed
UDS-Tunnel	i-0de66d15c228d4f04	Running	t2.micro	2/2 checks passed
xUbuntu-002	i-0dc3e4f95549197cb	Running	t2.micro	2/2 checks passed
xUbuntu-003	i-09c55cb9f7ae7425e	Running	t2.micro	Initializing
xUbuntu-001	i-01b739d3547e01819	Running	t2.micro	2/2 checks passed
xUbuntu-000	i-0790c78d0b270a32a	Running	t2.micro	2/2 checks passed



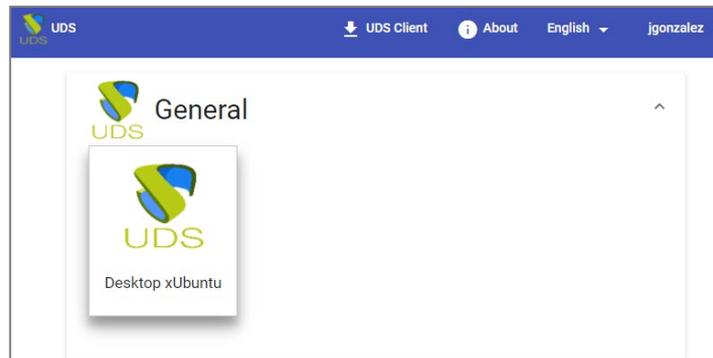
VDI with UDS Enterprise 3.5 & Amazon Web Services

Once the desktops are in a “Valid” state (i.e., the UDS Actor installed in the template has finished applying the necessary settings), they will be available for users to access.

The screenshot shows the UDS console interface for a desktop named 'Desktop xUbuntu'. The 'Cached services' section is active, displaying a table with the following data:

Creation date	Revision	Unique ID	IP	Friendly name	State	Cache level	Actor version
<input type="checkbox"/>	1	0a:5e:48:93:bb:0a	10.0.0.180	xUbuntu-000	Valid	1	3.5.0
<input type="checkbox"/>	1	0a:bc:a2:76:2a:0a	10.0.0.35	xUbuntu-001	Valid	1	3.5.0
<input type="checkbox"/>	1	0a:b0:79:67:ff:76	10.0.0.150	xUbuntu-002	Valid	1	3.5.0
<input type="checkbox"/>	1	0a:bd:82:83:8f:ee	10.0.0.66	xUbuntu-003	Valid	1	3.5.0

Access with a user to the services window (it is not possible to use the super-user administrator) and see the service available.



NOTE:

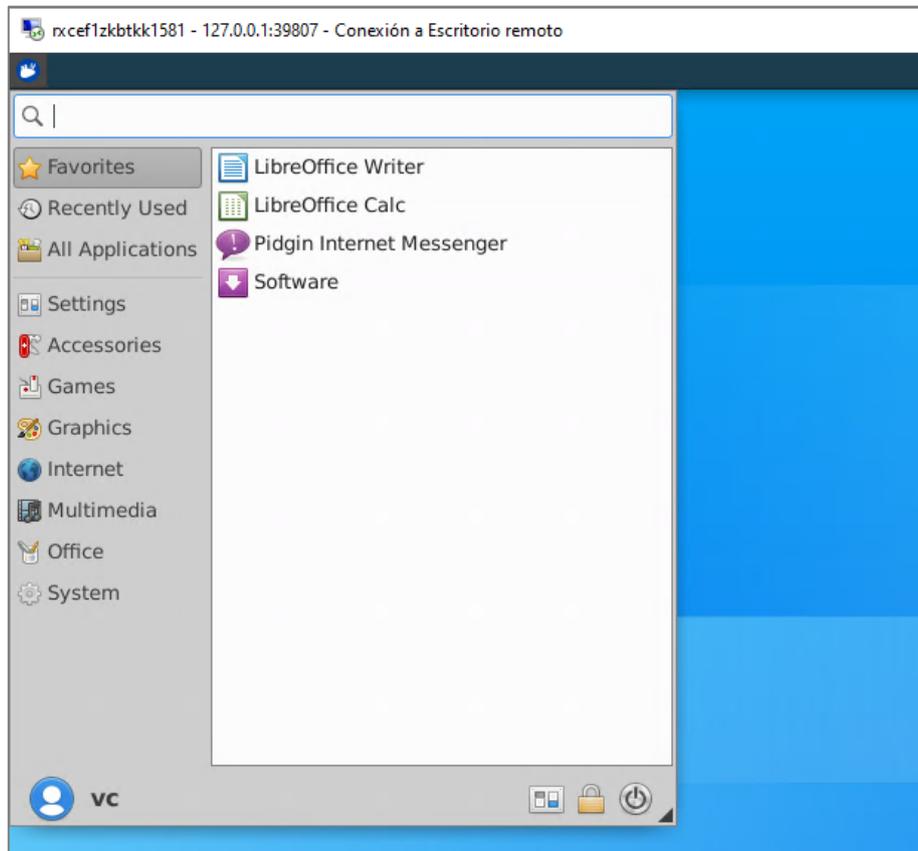
The “Service Pool” created must have a group of users (tab “Groups”) and a transport (tab “Transports”) assigned so that users can see it.



VDI with UDS Enterprise 3.5 & Amazon Web Services

www.udsenderprise.com

Access the service by clicking on the image (in this example an RDP type transport has been configured).



NOTE:

If you are outside the VPC network configured in AWS, it will be necessary to use a tunneled transport (as you can see in the screenshot of the connection example, it is connecting to 127.0.0.1 since the connection is made via Tunnel).



About Virtual Cable

Virtual Cable develops and markets UDS Enterprise through a subscription model by number of users, including support and updates.

The Virtual Cable team has more than 30 years of experience in IT and software development and more than 15 in virtualization technologies. Every day millions of Windows and Linux virtual desktops with UDS Enterprise are deployed worldwide.

For more information, visit www.udsenderprise.com or email us at info@udsenderprise.com

Should you have any technical question, please refer to support@udsenderprise.com

For information on UDS Enterprise subscription system, support services, SLAs and other details regarding the software see [this document](#).