
HNSciCloud Rhea Documentation

Release 1.0

HNSciCloud Consortium

Oct 18, 2018

Contents

1	Introduction	1
2	Getting Started	5
3	Researchers	17
4	Administrators	45
5	Data Coordinators	69
6	Support	75
7	Platform Services	77
8	Platform Training	79
9	Consortium	85

1.1 HNSciCloud

Ten of Europe's leading public research organisations launched the [Helix Nebula Science Cloud \(HNSciCloud\) Pre-Commercial Procurement \(PCP\)](#) to establish a European cloud platform that will support the high-performance, data-intensive scientific use-cases of this "Buyers Group" and of the research sector at large.

This PCP calls for the design and implementation of an innovative Infrastructure as a Service (IaaS) solutions for compute, storage, network connectivity, Federated Identity Management, and Service Payment Models, to augment and to enhance the science community's existing systems.

The RHEA Group consortium's HNSciCloud design builds on a solid foundation of engineering expertise, existing open source software and commercial services:

- RHEA System Engineering & Cyber Security expertise
- SixSq's Nuvla, a SlipStream-based hybrid-cloud management service
- Cyfronet's OneData for Data Management
- Exoscale IaaS Cloud Service (CloudStack)

The R&D innovations will be incorporated into these services as part of our commercial offerings, with minimum intrusion into the buyers' infrastructure, including:

- Peta-scale data management solution, portable to any cloud
- Flexible container management
- Authentication with users' home credentials

The Pilot (Phase 3) exercises the Consortium's platform at a larger scale and allows for continued evolution of the platform to meet the needs of scientific computing.

1.2 Functional Overview

The cloud platform coming from the design phase provides cost-effective access to large-scale compute and storage resources. The solution brings together two commercial services, an authentication infrastructure that supports federated identities, and a data management infrastructure that can handle the large data sets of the Buyers Group.

Authentication (KeyCloak) Federates external identity providers, allowing users to use their “home” credentials to access the platform.

Orchestration (Nuvla) Allows users to manage the full lifecycle of cloud applications with a high degree of automation.

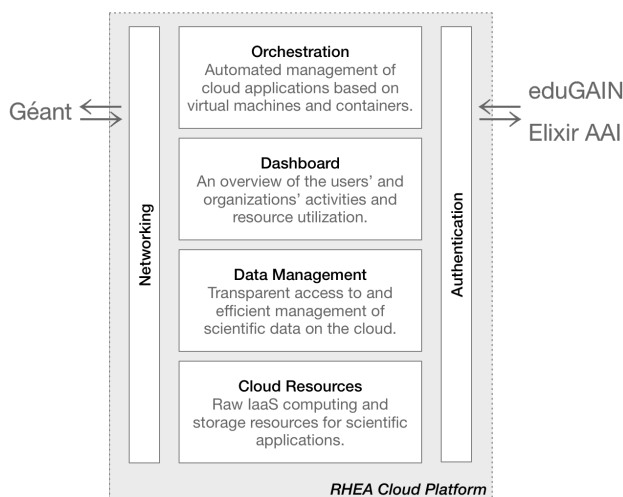
Data Management (Onedata) Allows users to access and to manage large datasets hosted in hybrid cloud infrastructures and/or at a Buyers Group organization with minimum intrusion.

Networking (peering with GEANT) Allows access to all the platform services from anywhere with enhanced access from sites connected to GEANT.

Cloud Resources (Exoscale) IaaS and storage services accessible from the hybrid cloud platform.

Dashboard (Exoscale, Nuvla) Provides an overview of users’ current activities.

All the components are based on open source software released under the Apache 2.0 license, which allows liberal academic and commercial reuse.



The integration of these key components was demonstrated during the prototype phase. The pilot phase concentrates on validating the platform at scale.

1.3 Actors

The primary users of the hybrid cloud platform will be researchers who want to analyze large datasets. However, there are many other actors involved to make the platform useful. To be as exact as possible when describing interactions with the platform, we have identified the full set of actors:

Researcher A person from a Buyers Group organization who analyzes scientific data by deploying instances of cloud applications (defined by Application Developers) for himself.

Application Operator A person from a Buyers Group organization who deploys and manages instances of cloud applications (defined by Application Developers) for others.

Data Service Operator A person from a Buyers Group organization or the Consortium who is responsible for deploying and maintaining the data services specific to an organization, project, or experiment.

Application Developer A person from a Buyers Group organization, Consortium or other organization who develops generalized software or services for use by others that use the platform's services, including data sets maintained by a Buyers Group organization. Defines (scalable) applications on the platform that can be deployed by a Researcher or Application Operator.

Data Coordinator A person from a Buyers Group organization who is responsible for managing the data (publishing, replicating, validating, archiving, etc.) for a specific organization, project, or experiment.

Account Coordinator A person from a Buyers Group organization who is responsible for managing the accounts (including credentials and quotas), monitoring resource utilization, and tracking costs.

Platform User or User A Researcher, Application Operator, Data Service Operator, Application Developer, Data Coordinator, Account Coordinator.

Broker Service Provider The organization that provides the cloud application management and brokering services for the platform, i.e. Nuvla.

Service Provider A “broker service provider” or “IaaS service provider”.

Consortium The organizations that together provide the hybrid cloud platform for HNSciCloud.

1.4 Scope and Coverage

This documentation covers the essentials for learning about and getting started with the HNSciCloud hybrid cloud platform from the RHEA collaboration. It contains only information specific to the platform as a whole. Documentation for the individual services that comprise the platform are available elsewhere and may need to be consulted for anything other than simple use cases. Links to that documentation are provided in the *Platform Services* section.

In order to start using the system, you will need to setup and configure your accounts. This section describes how to do this and then how to ensure that everything is working correctly.

2.1 Exoscale Account

Exoscale provides the IaaS computing resources for the RHEA Cloud Platform. The Frankfurt and Geneva Exoscale regions are connected to Géant and can provide high bandwidth to academic sites also connected to Géant. To use the platform, you must have an Exoscale account.

2.1.1 Buyers Group Account

The RHEA Consortium has already defined “organizations” within Exoscale to allow the administrator of each Buyers Group tenant to manage its users. Contact your administrator directly to obtain an account. If you don’t know your administrator, you can contact the Support HelpDesk (support@sixsq.com).

2.1.2 Voucher Redemption


You may have been given a voucher for HNSciCloud credit on Exoscale. You can either create a new account using the voucher or add the credit to an existing account.

Create Account

To redeem an Exoscale voucher and **create a new account**, open the provided voucher link within a web browser. A typical link looks like:

`https://portal.exoscale.com/register?coupon=XXXXXXX`


1. Enter the email address and password you wish to use. Accept the terms and hit sign up.

 **EXOSCALE**

☒ I agree to Exoscale's [privacy licence agreement](#)

[I already have an account](#)

2. A validation email has been sent. Check out your mailbox and click on the verification link.

 **EXOSCALE**

Activation email sent

Almost done! Please check your email and click on the [verification link we just sent you.](#)

You'll be able to login once your account is verified, which should only take a couple of minutes.

3. Choose “for team projects” and fill your details. Choose your [Exoscale](#) organization name and submit:

Thank you for registering with Exoscale
You are just one step away from starting your first instance!

☒ For team projects
 ☐ For personal projects

Your full name *

My name

Company or team name *

My Company

Address *

My address

City *

My City

Zip Code *

1234

Country *

Switzerland

☒ I agree to Exoscale's [terms of service](#).

You are registering with the following coupon:

Exoscale1234567890

4. You're in and you may now spawn new instances.

Credit an Existing Account

If you **already have an Exoscale account**, you can add the voucher credit to it. Simply make sure that you are logged into the [Exoscale portal](#) and then visit the link:

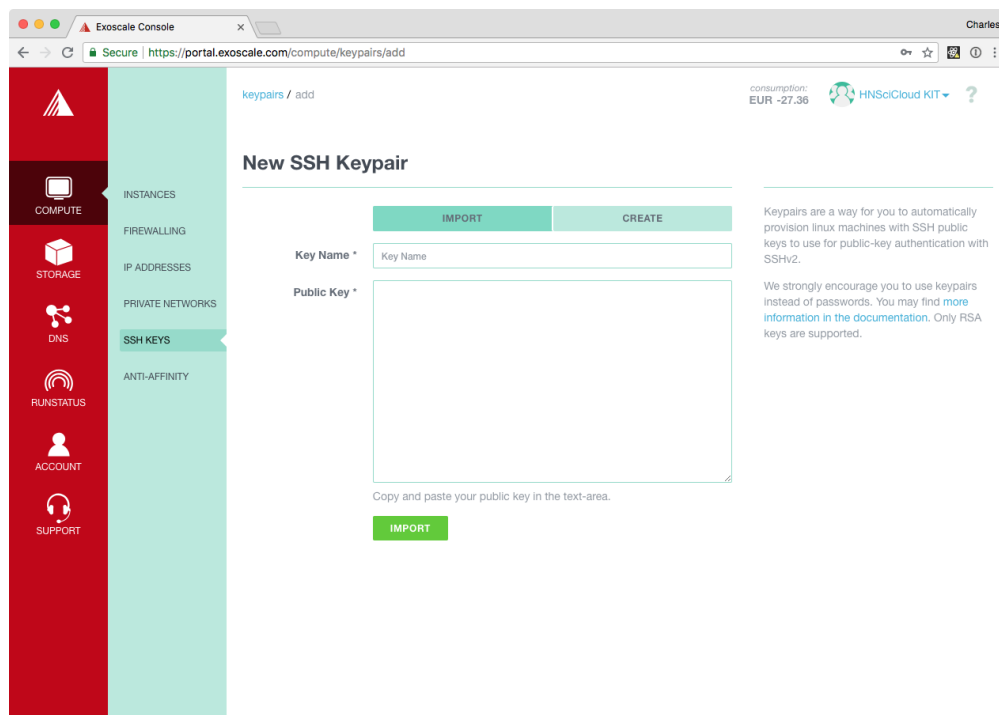
<https://portal.exoscale.com/account/funds/redeem-coupon>

Just enter the code and the amount of the voucher will be credited to your account.

2.1.3 SSH Configuration

It is very strongly recommended that you **use SSH keys to access your running virtual machines**.

To add your public SSH key to your account, navigate to the “Compute” tab and then the “SSH Keys” panel in the [Exoscale portal](#). From here, click on the “ADD” button to upload your public SSH key. You should see a dialog similar to the following screenshot.



Provide a descriptive name for the key, paste your **public** key into the textbox, and then click on “IMPORT”. After the import, click on the “Set as default” link below the key to make it the default.

You can also use this interface to create a new SSH key pair. If you do this, be sure to save the generated private key and configure your laptop to use this key.

2.1.4 GPUs and Large Flavors

To request access to the Exoscale GPU instance flavor, just submit a support ticket to support@sixsq.com. You can do the same if you need access to the “Mega” or “Titan” flavors.

If you have registered using a voucher, please specify that it’s related to the HNSciCloud project in order to speedup the request.

2.2 Nuvla Account

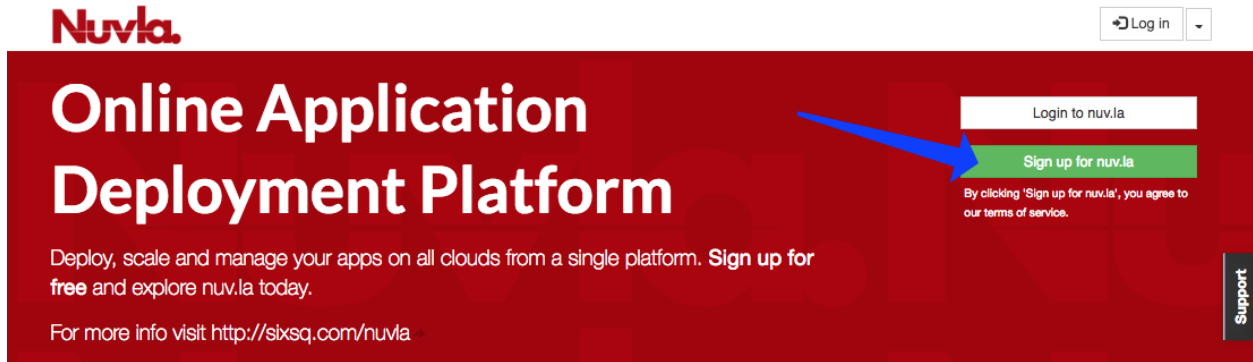
Orchestration features are implemented by [Nuvla](#). Use of Nuvla is entirely optional, although some automated deployment of systems (e.g. SLURM) will not be available otherwise.

2.2.1 Registration

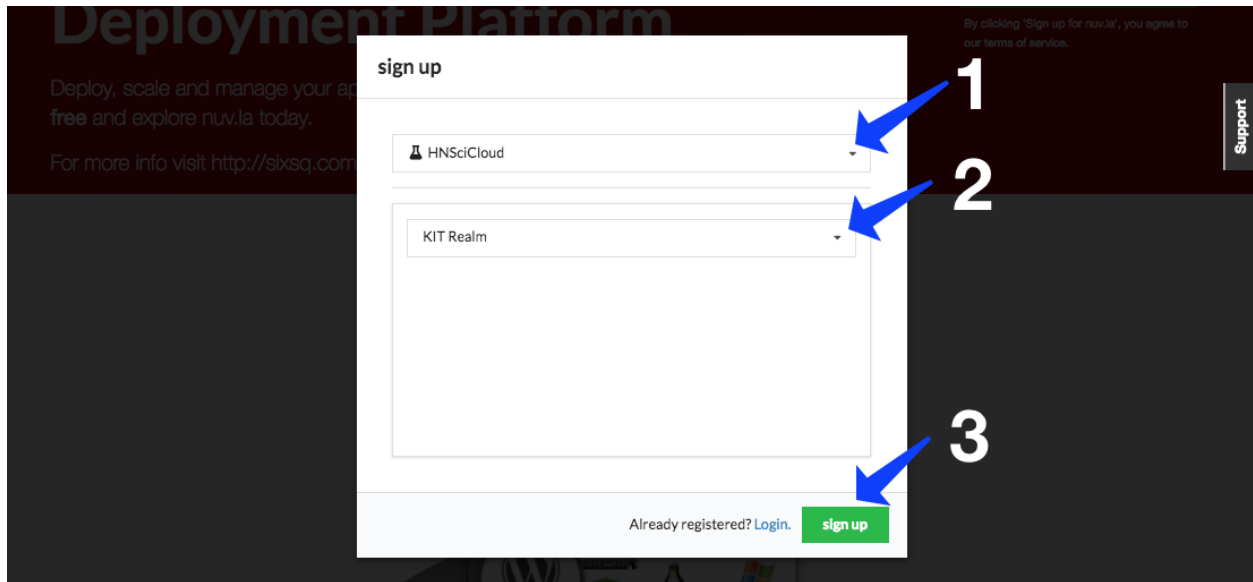
New users may create their accounts by registering with Nuvla with their institutional credentials through the eduGAIN and Elixir AAI identity federations.

The full procedure to activate an account in Nuvla using your institutional credentials is as follows:

1. Click on the login button which will then take you to a page to select your login method.

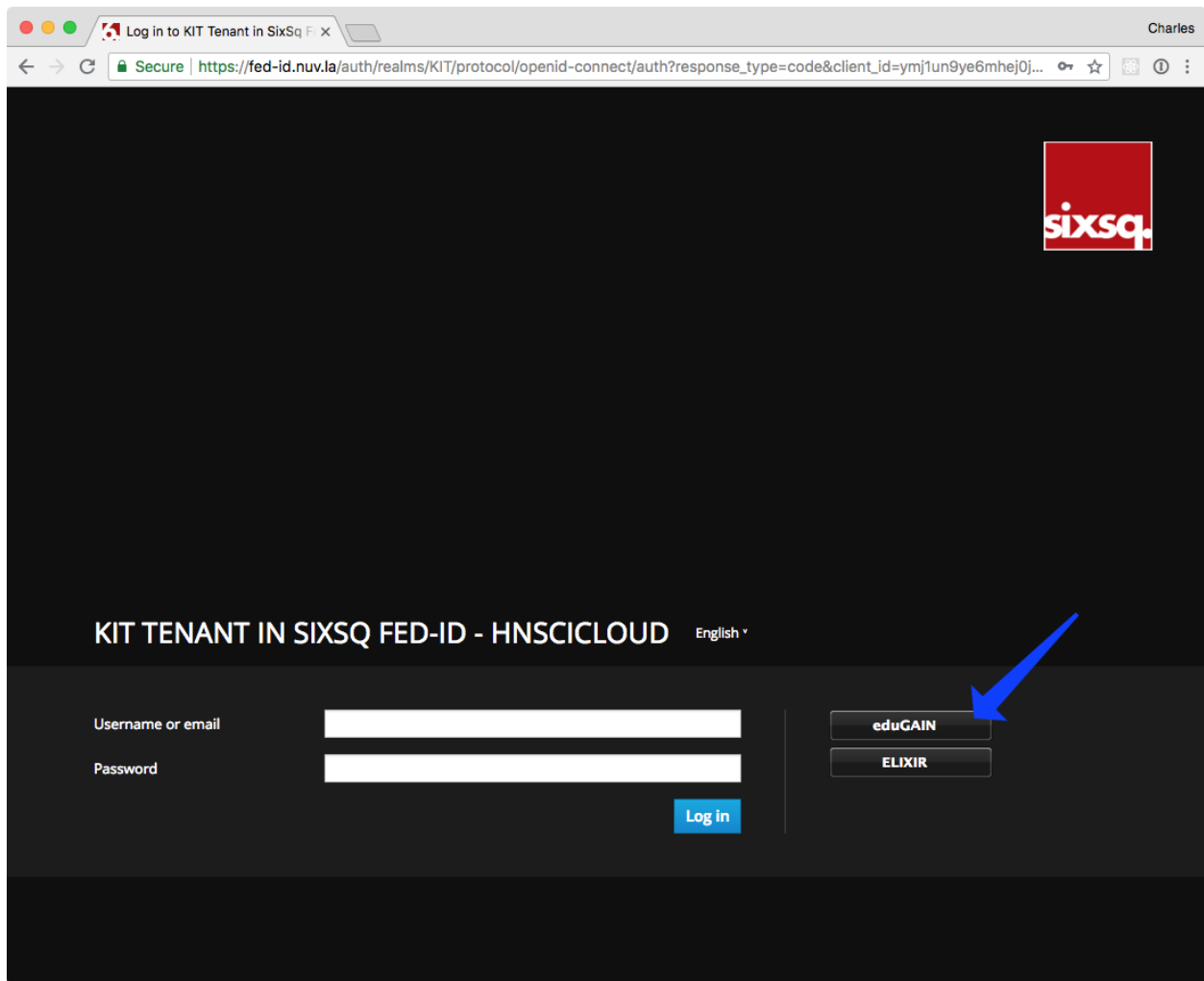


2. In this dialog, HNSciCloud should select “HNSciCloud” and then select their realm (or tenant) as shown in the figure below. Then click on the “sign up” button.



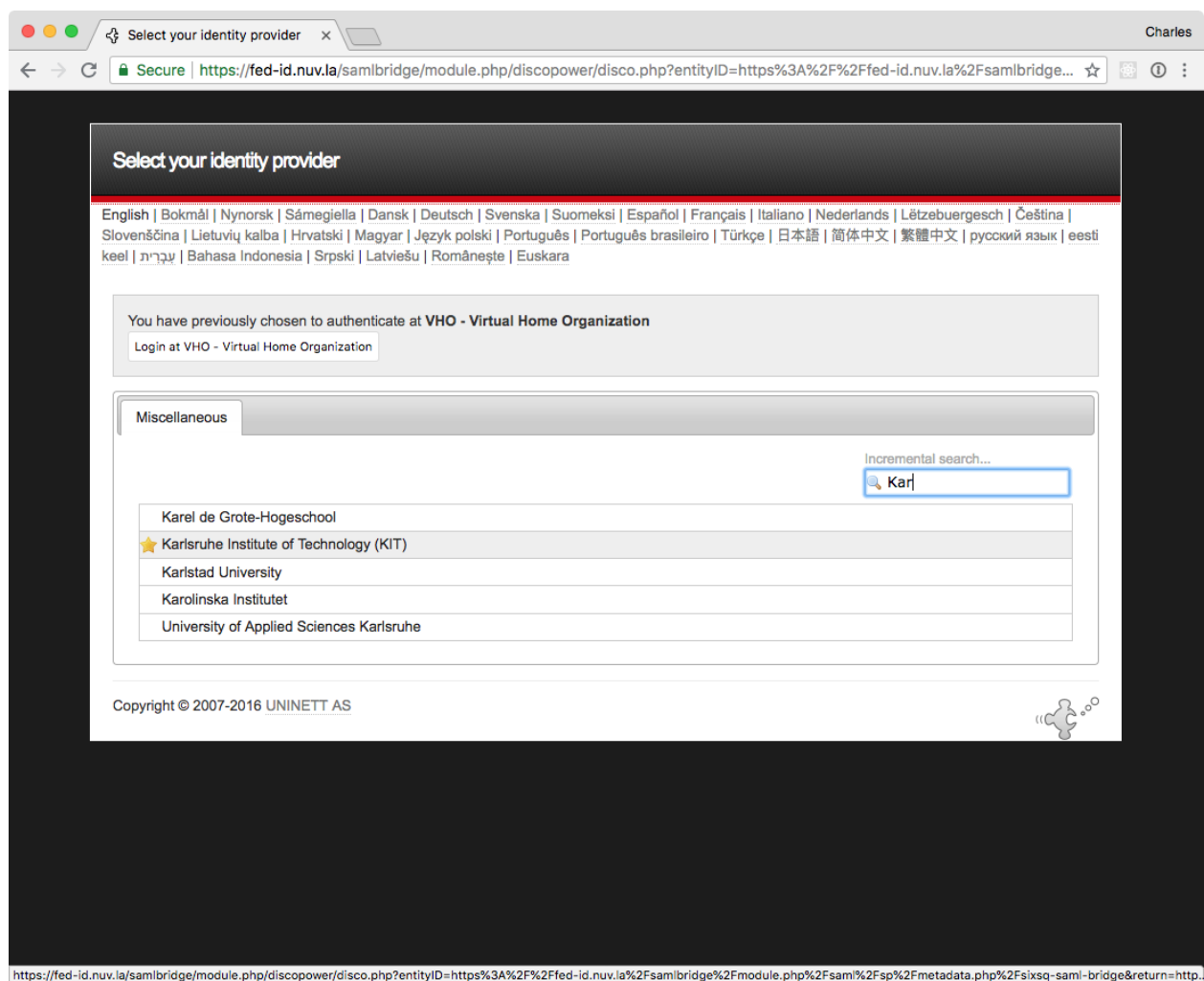
This will redirect users to their respective login realm in SixSq’s [Federated Identity Portal](#). This portal is SixSq’s authentication and authorization infrastructure (AAI) and it uses [Keycloak](#) and [simpleSAMLphp](#) underneath in order to make the authentication bridge between client applications (like Nuvla) and identity federations like eduGAIN and ELIXIR AAI (using SAML2.0).

3. Users shall then select which identity federation they want to use, either eduGAIN or ELIXIR.



4. For both eduGAIN and ELIXIR, users will then be presented with a comprehensive list of identity providers and a search field.

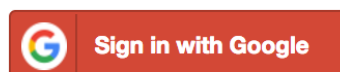
eduGAIN:



ELIXIR:



Log in using ...



or

your institutional account

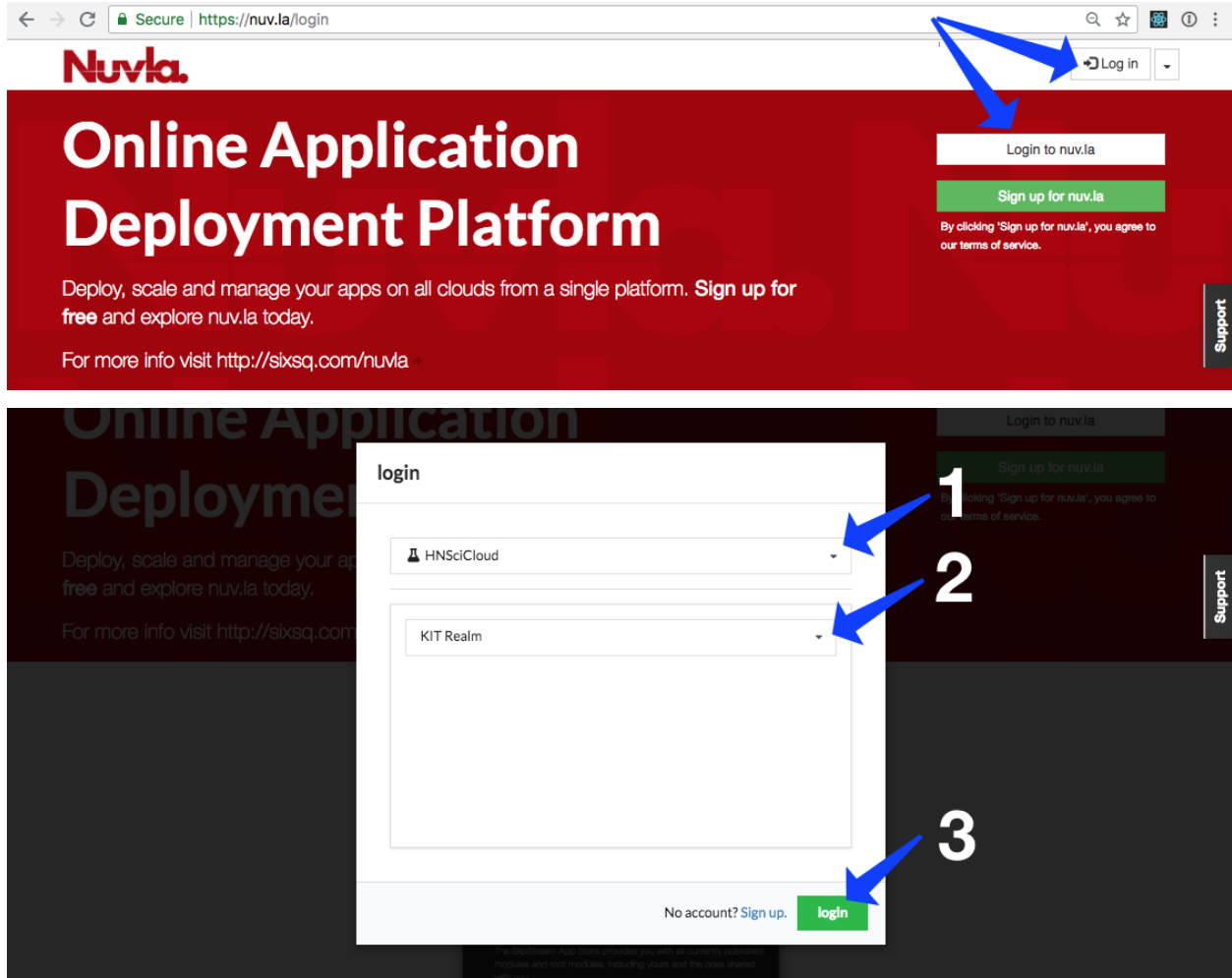
Aalborg University
Aalto University
Aarhus Basic Health Care College
Aarhus School of Marine and Technical Engineering
Aarhus University
ACOnet staff
Alexander Fleming Biomedical Sciences Research Center
Alexander Stuart test IdP (vin)

Upon selection of the identity provider, users will be redirected to their institute's login page.

- When successfully authenticating with the identity provider, the user will then be redirected back to Nuvla.

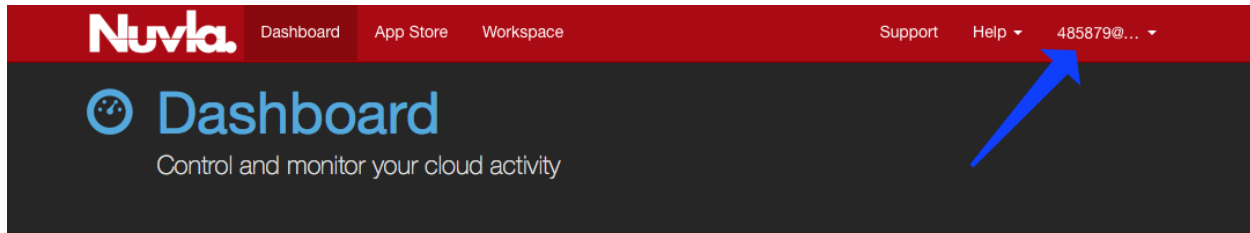
Warning: Currently there is no “success” message when you are sent back to Nuvla. When you arrive back on Nuvla, just login via one of the “Login” buttons.

To login, click on one of the login buttons, select the authentication and tenant (realm), and then click the green “login” button.

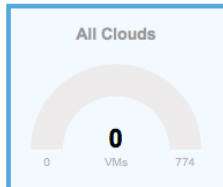


6. Depending on how recently you authenticated with your identity provider, you may be requested to authenticate again or simply logged in automatically. When you login, you will normally be redirected to the Nuvla dashboard.

Note: The first time you login you will be redirected to the App Store and offered the chance for a tutorial. This will not appear on subsequent visits.



Usage

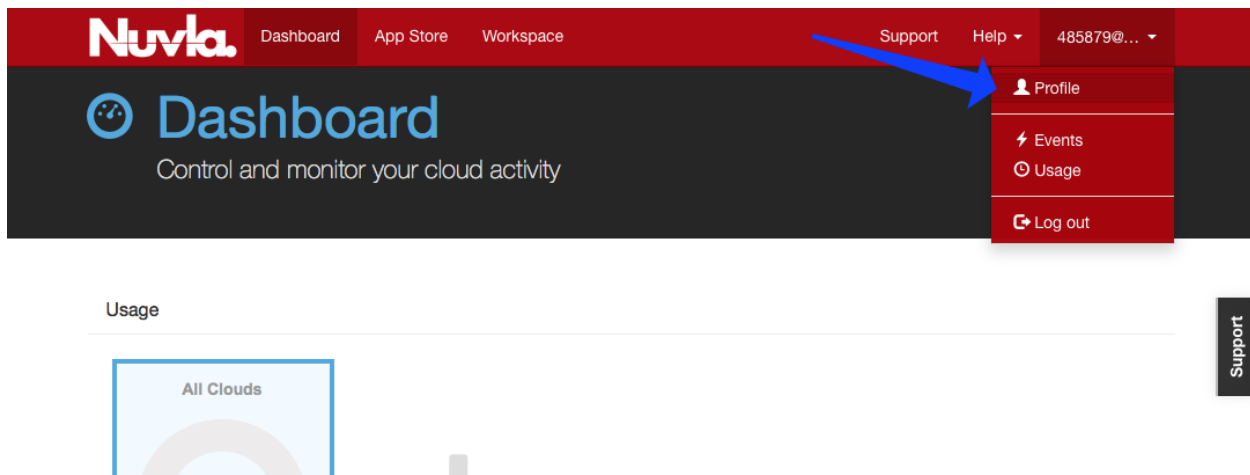


Support

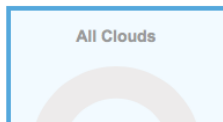
7. Users that are an **ACCOUNT MANAGER** must send an email to support@sixsq.com asking *admin* rights to the tenant, which shall be granted by SixSq, in SixSq's [Federated Identity Portal](#), where the account managers can then manage users, groups and roles (as described in [here](#)).
8. **STANDARD USERS** may want to contact the account manager for their realm so that the manager can assign roles to them or add them to a group. (This configuration may or may not be done automatically.)

2.2.2 Account Configuration

To use Nuvla to provision the data management services or cloud applications on the IaaS cloud infrastructures, you must configure your Nuvla account. To access your user profile, click on “Profile” link under your username.



Usage



Support

To update your user profile, click on the “Edit...” on the right side below the page header.

Remote Machine Access

To allow you have remote access to the (Linux) virtual machines that you deploy, you should provide a public SSH key. Once this key has been added to your profile, Nuvla will automatically configure all deployed virtual machines with this key, giving you ‘root’ access to your deployed machines. The instructions for creating an SSH key pair and configuring your profile can be found in the [Remote Machine Access](#) section of the SlipStream documentation.

This documentation also describes the installation of a “Remote Desktop Connection” client for accessing Windows machines.

Cloud Credentials

In order to be granted access to the Exoscale cloud credentials, technical users must contact their account managers, asking for a specific user role (**can_deploy**) to be given to them, as described in *Cloud Provider Configuration*.

2.3 Onedata Account

The *Onedata* services allow organizations and their users to manage their data sets on the cloud. These services are fully integrated with the federated identity management services of HNSciCloud: eduGAIN and Elixir AAI. Onedata automatically creates an account if necessary when logging in with your federated identity, so no explicit registration is required.

This section contains howtos for common tasks that will be carried out by researchers.

3.1 Cloud Resources

Exoscale provides the IaaS cloud resources for the RHEA platform. These resources can be provisioned directly via:

- [Exoscale Portal](#)
- [Libcloud](#)
- [Terraform](#)

A range of other services, like the private network features, are also available and may help to integrate your computing resources and applications with the cloud.

Follow the links to the Exoscale documentation [Exoscale](#) for complete information on these topics.

3.1.1 Virtual Machine Lifecycle

The full lifecycle for Virtual Machines can be handled through the [Exoscale portal](#). Log into the portal to view and control your resources.

Security Groups

Your account will initially be configured with an empty default security group. This essentially creates a firewall for your virtual machines that blocks access on all ports. This is a secure but not very useful default.

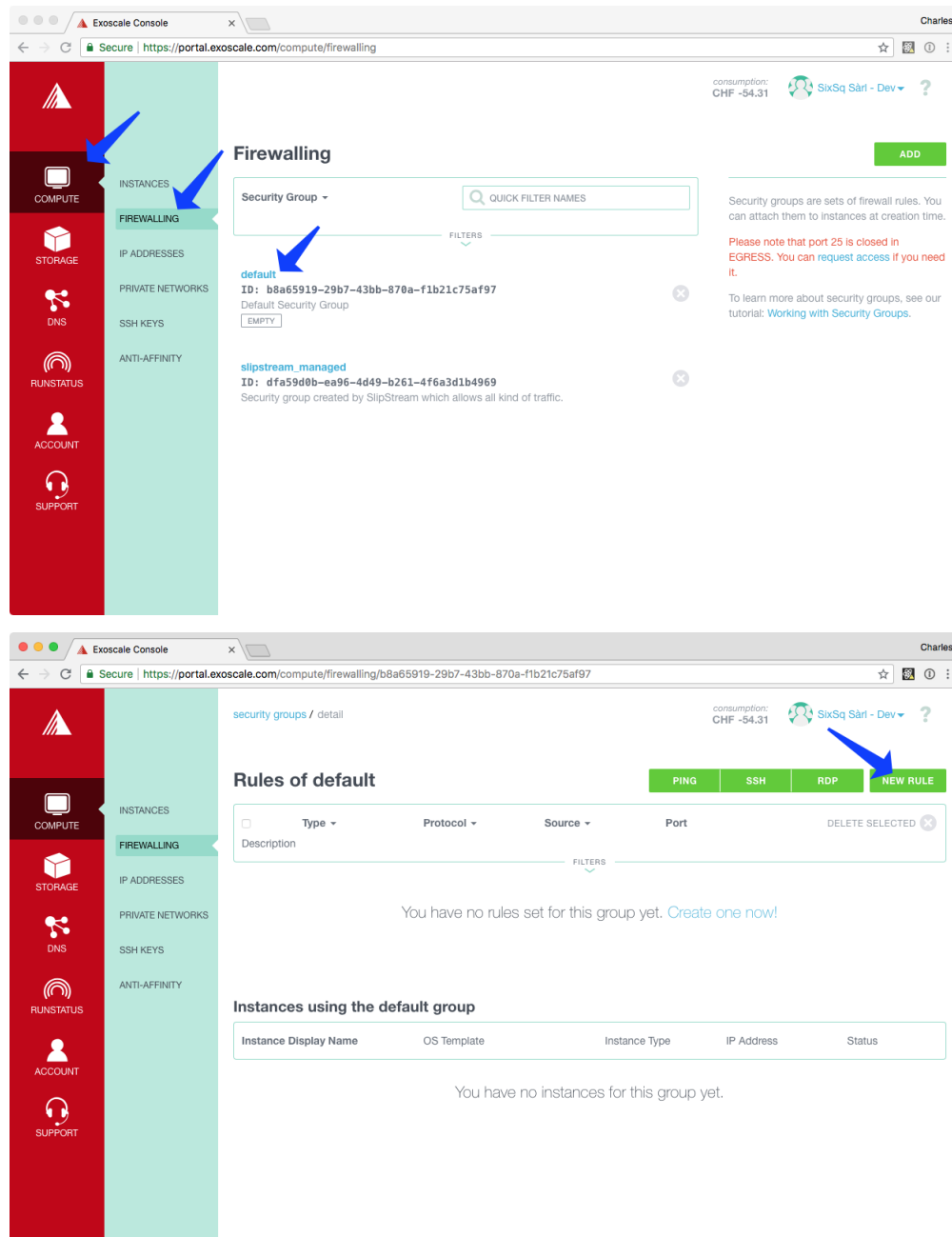
At least allow access to the SSH port to the security group so that you can log into your virtual machines. The following steps will allow input connections on port 22 (SSH):

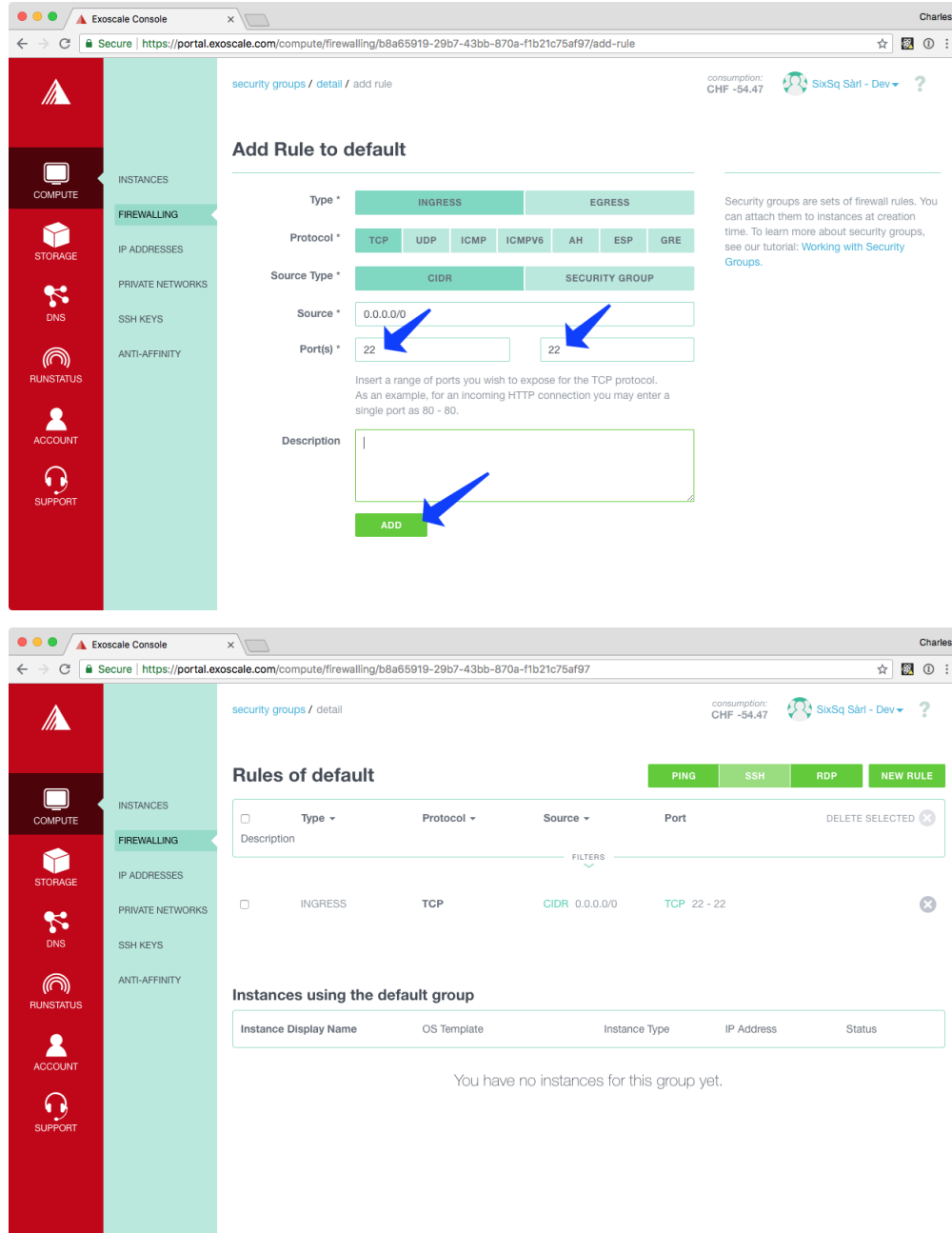
- Navigate to the “COMPUTE” and then “FIREWALLING” panel.
- Click on “default”.

- Click on “NEW RULE”.
- Add port range 22-22.
- Click “ADD”.

At the end of this, you should see a single rule allowing inbound access on port 22 (SSH). Any changes that you make to the security group are applied in real time to the machines that use the group.

Note: This could be accomplished more quickly by using the “SSH” button on the security group page. Using the dialog allows you to see the process for other, less common ports.





Starting Virtual Machines

Once you have logged into the [Exoscale portal](https://portal.exoscale.com), you can start new virtual machine instances by doing the following:

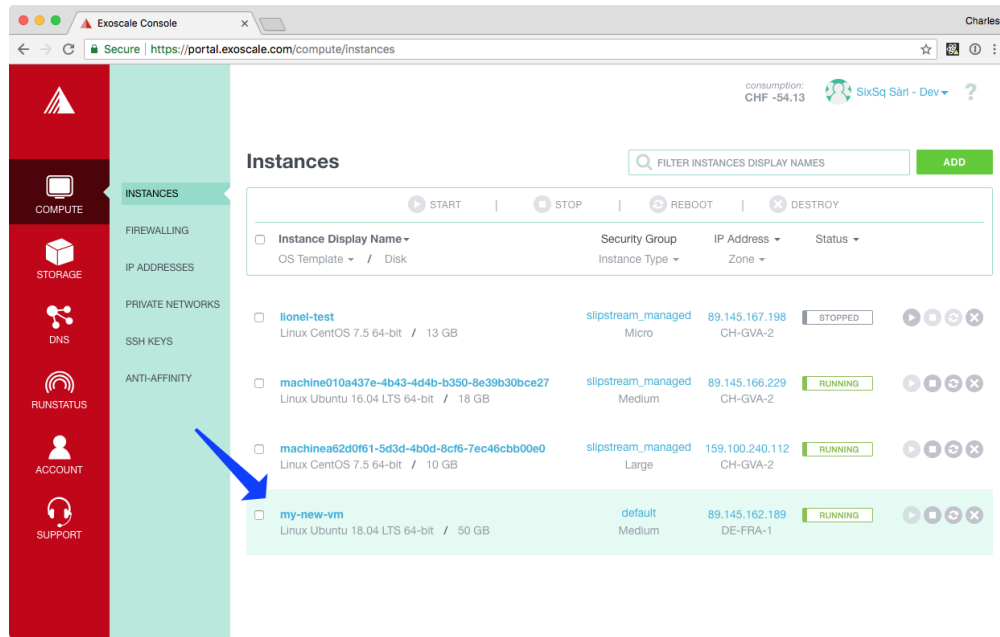
- Navigate to the “COMPUTE” tab and then the “INSTANCES” panel,
- Click the “ADD” button,
- Fill in the form with the VM characteristics, and then
- Click on the “CREATE” button.

You will then see your new machine in the list of virtual machine instances.

Note: Be sure that you have imported an SSH public key, so that you can access your instance via SSH.

The first screenshot shows the 'Instances' page in the Exoscale Console. The left sidebar has a red 'COMPUTE' button highlighted with a blue arrow. The main content area shows a table of instances with columns for Instance Display Name, OS Template, Disk, Security Group, Instance Type, IP Address, Zone, and Status. The table lists three instances: 'lionel-test' (Linux CentOS 7.5 64-bit, 13 GB, Micro, 89.145.167.198, CH-GVA-2, STOPPED), 'machine010a437e-4b43-4d4b-b350-8e39b30bce27' (Linux Ubuntu 16.04 LTS 64-bit, 18 GB, Medium, 89.145.166.229, CH-GVA-2, RUNNING), and 'machinea62d0f61-5d3d-4b0d-8cf6-7ec46cb00e0' (Linux CentOS 7.5 64-bit, 10 GB, Large, 159.100.240.112, CH-GVA-2, RUNNING). A blue arrow points to the 'ADD' button in the top right corner.

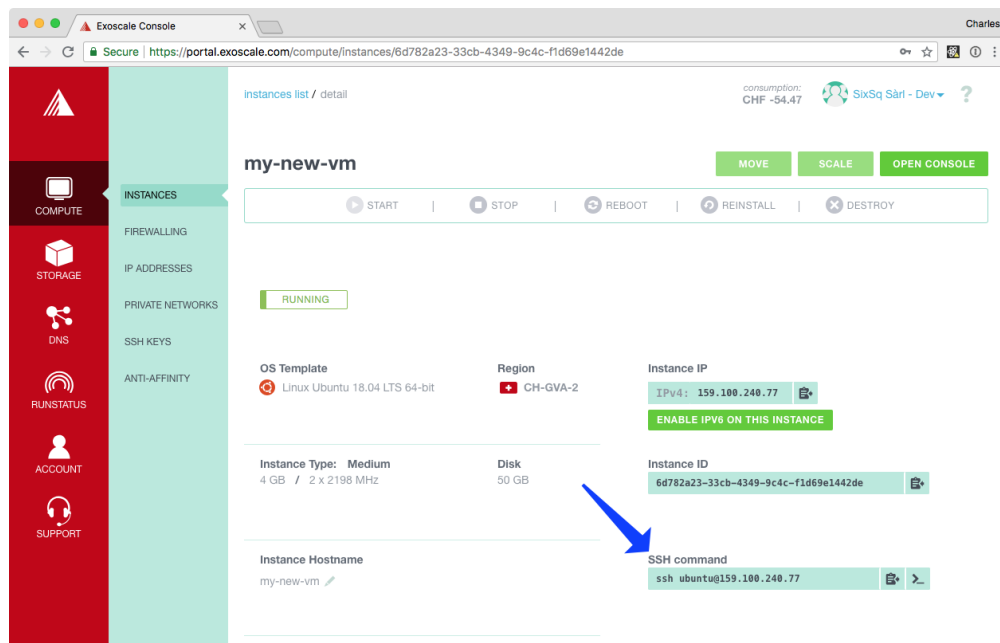
The second screenshot shows the 'New Instance' page. The left sidebar has a red 'COMPUTE' button highlighted with a blue arrow. The main content area shows the 'New Instance' form. The 'Hostname' field is 'my-new-vm'. The 'Zone' is 'CH-GVA-2'. The 'Template' is 'Linux Ubuntu 18.04 LTS 64-bit'. The 'Security Groups' section shows 'default' with a red warning icon and a blue arrow pointing to the 'CREATE' button. The 'Your instance:' summary shows 'my-new-vm', 'DE-FRA-1', 'Linux Ubuntu 18.04 LTS 64-bit', '2 x 2198Mhz', '4 GB RAM', '50 GB', 'Keypair: cal', and 'Security Groups: default'.



Exoscale supports a variety of Linux operating systems and Windows. Exoscale has four geographic regions: Geneva, Zurich, Vienna, and Frankfurt. The **Geneva** and **Frankfurt** regions are the primary ones for the HNSciCloud project.

Accessing Virtual Machines

You can follow the deployment progress of your machine from the list of instances. You can get the details for a particular machine by clicking on the machine name in the list. You should see a page similar to the following screenshot.



The command to use to access the machine can be found on this page. The command contains both the correct username and the machine's IP address.

From the terminal, you should be able to do the following to access the machine:

```

~> ssh ubuntu@159.100.240.77
Warning: Permanently added '159.100.240.77' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04 LTS (GNU/Linux 4.15.0-20-generic x86_64)

...

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@my-new-vm:~$

```

Either you will be able to log directly into a “root” account or you can use the command `sudo su -` to access it. With the “root” account you have full control to configure the machine with the software you need.

Terminating Virtual Machines

Either from the Virtual Machine detail page or from the instance list, you can control the state of the machine. With the action button “destroy” you can terminate the machine and free any resources.



The destroyed machine will disappear from the list of instances after the resources have been freed.

Warning: When you destroy a machine, the configuration of the machine and all data are lost.

You can stop and restart the machine as well. A stopped machine will retain its configuration and data. You will only be charged for the storage capacity of a stopped machine.

3.1.2 Virtual Private Network

Using a Virtual Private Network (VPN) makes sense in many cases where you want to integrate cloud resources into your local services seamlessly. One such example would be extending a local batch cluster with cloud resources.

To achieve that you may make use of the [private network](#) feature. Exoscale provides an [example using OpenBSD](#), which can be ported to other operating systems easily.

3.2 Data Access

To facilitate a seamless access to the Buyers’ distributed data sets, the project uses [Onedata](#) global data management system.

Onedata infrastructure, spanning multiple clouds, consists of two components - Onezone and Oneprovider, and user level Oneclient tool for the data access. Each Buyer Group has a single Onezone instance and per cloud instance of Oneprovider. Oneproviders attach to the Cloud’s data stores and connect to Onezone, with the latter being the main entry point and enabling the global authentication and metadata management.

At this time, it is assumed that the distributed Onedata infrastructure is already deployed by the Buyer’s Group [Data Coordinators](#) and the endpoints of Onezone and Oneproviders are available.

3.2.1 Using Nuvla to provision data access client

Users access their data on VMs using POSIX. The data becomes available on the VMs via mounting the required data sources with the help of [Oneclient tool](#). When deploying VMs via [Nuvla](#) service, users should use or, when building their own components, inherit from *oneclient- $\langle OS \rangle$* components, which are available at <https://nuv.la/module/HNSciCloud/onedata>.

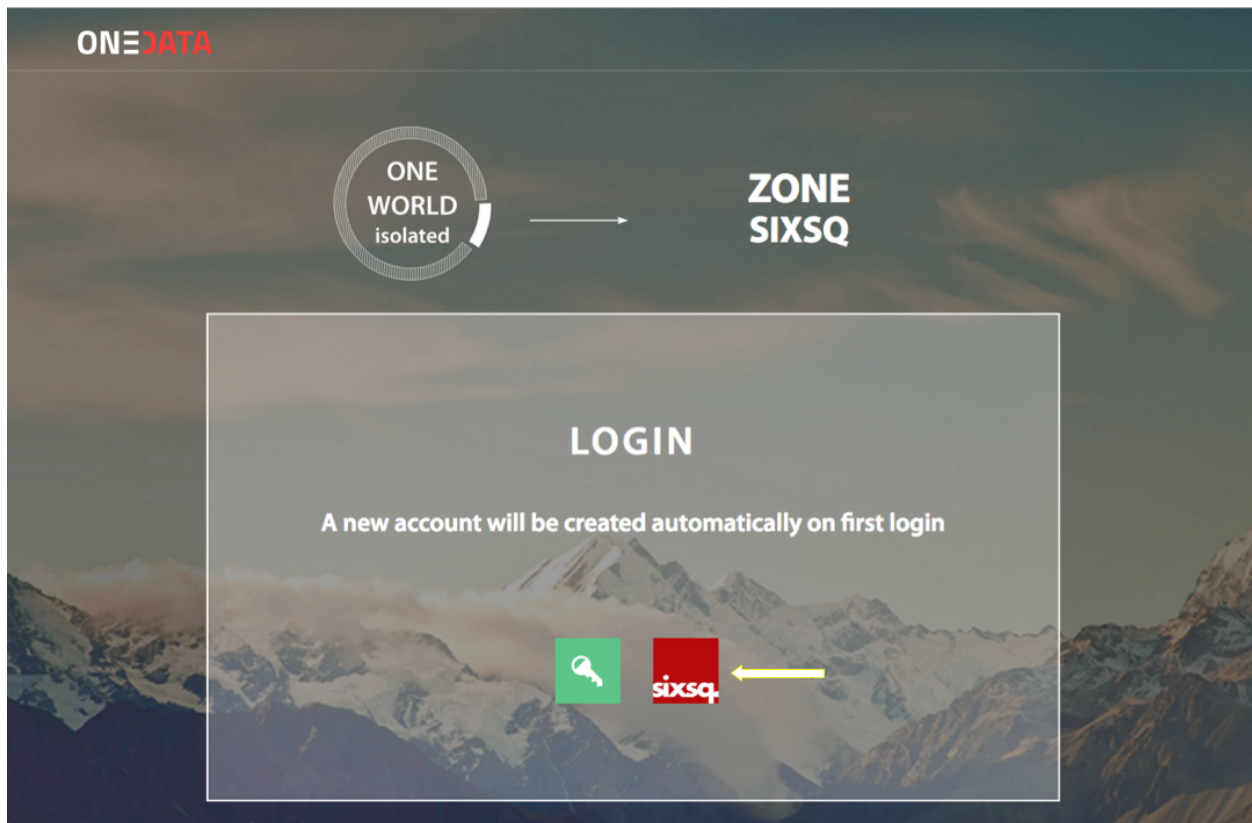
At the moment, for *oneclient* to mount the data volumes on VMs and enable POSIX access to the data, users need to provide it with a data access token and the Cloud local Oneprovider endpoint.

Next versions of the platform will be extended to contain the auto-discovery of the Cloud local Oneprovider as well as auto-generation of the data access token.

Obtaining Onedata data Access Token

The following steps are required to get the data access token

- obtain the Buyer's Group Onezone endpoint
- in Onezone, authenticate via FedIdP



- when logged in Onezone web user interface, press *Access Tokens* in the top menu and press *Create new access token* button, then click on the *copy* pictogram

ONEDATA

GO TO YOUR FILES

DATA SPACE MANAGEMENT

+

Create new space

JOIN

Join a space

You can also become a provider yourself and support your own space.

GROUP MANAGEMENT

JOIN

Join a group

ACCESS TOKENS

Generate new token to get access to Onedata using command line client or REST API

MDAxNWxvY2F0aW9uIG9uZXpvcnM

+

Create new access token

AUTHENTICATION SETTINGS

USER ALIAS

MANAGE ACCOUNT

First

1

2

3

Get

To store at least one

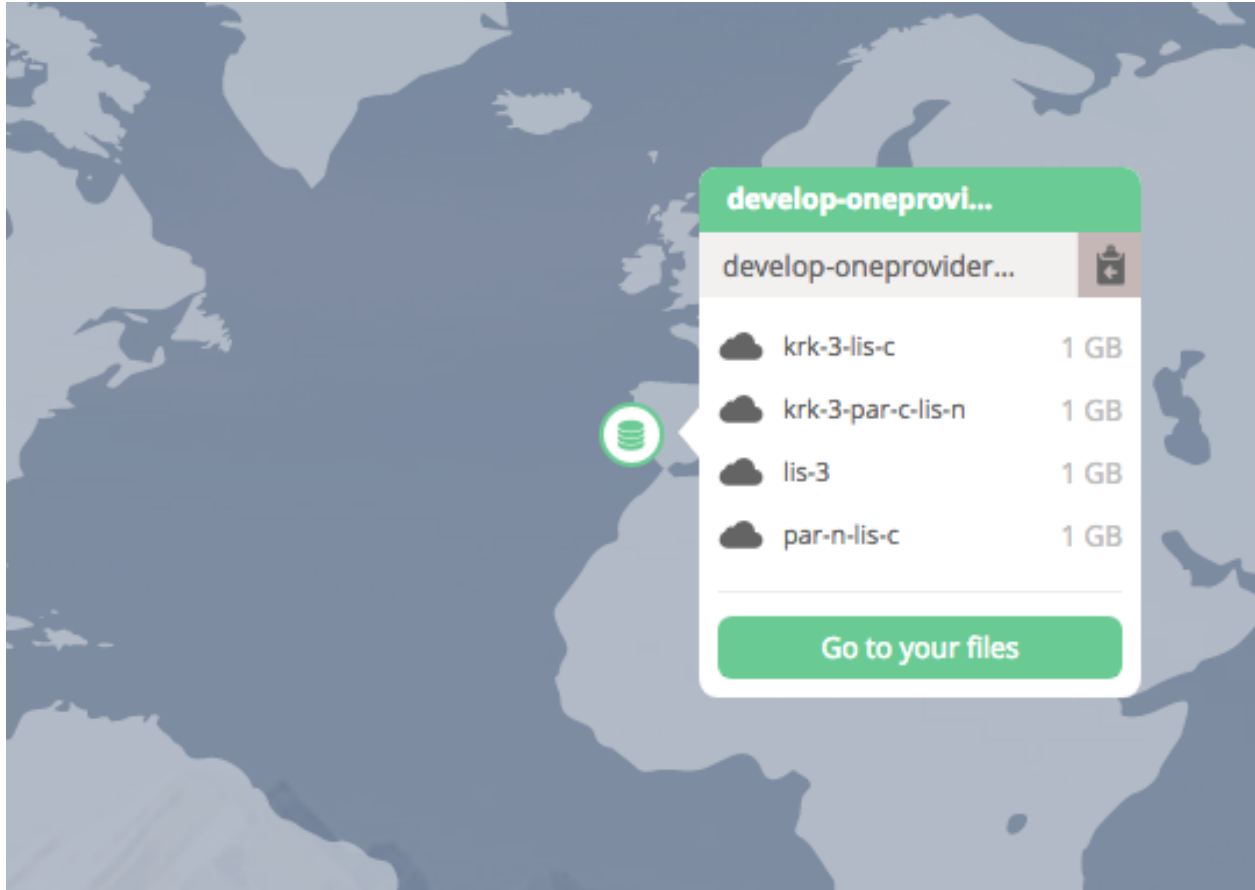
Chapter 3. Researchers

The copied data access token should be used with the **access-token** parameter in the *oneclient* component deployment.

Obtaining Oneprovider endpoint

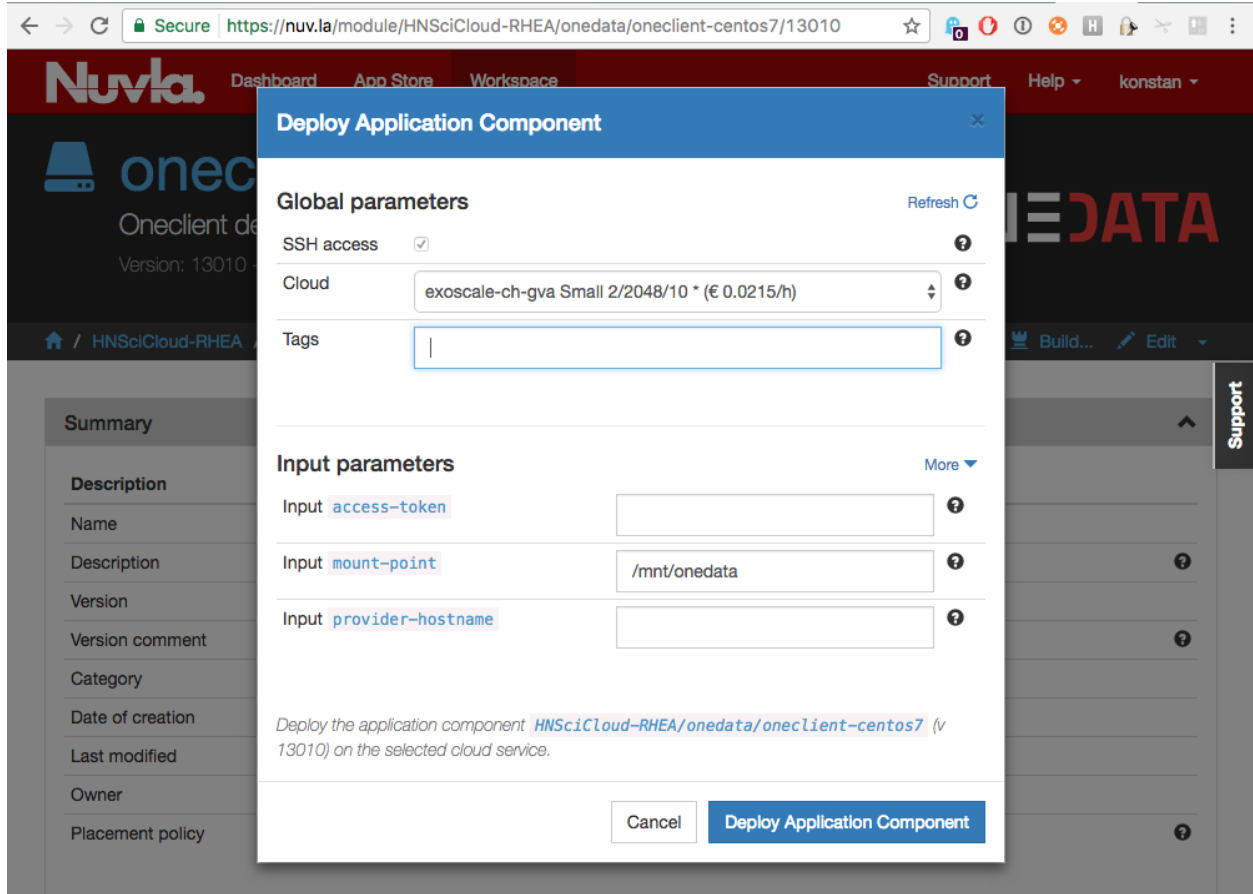
Check with your Data Manger for the IP/DNS name of the Oneprovider endpoint on the Cloud of your choice.

Oneprovider IP can also be easily copied from the Onezone ‘world’ view by clicking on the icon of selected One-provider instance and pressing *copy* button in the top right corner of the popup:



Provision VM for data access

Here it is explained on the example of CentOS 7 image. Go to <https://nuv.la/module/HNSciCloud/onedata/oneclient-centos7> and click on *Deploy*.



Select the Cloud on which you want to deploy the client via *Cloud* drop-down. Provide **access-token** and **provider-hostname** parameters. Optionally change the default mount point of the spaces in **mount-point** parameter. Then, click *Deploy Application Component* button.

3.2.2 Accessing data on VM

After the successful deployment, user should expect the spaces (supported by the selected provider) with the files available for POSIX access under **mount-point**. Example

```
# ls -l /mnt/onedata/
space-s3
space-gluster
# ls -l /mnt/onedata/space-s3
test.txt
#
```

3.2.3 External documentation

More information on how to

- access, manage and share your data;
- create groups of users with fine grained access rights;
- share and collaborate on your data

can be found in Onedata [User quickstart](#) and the [User guide](#).

3.3 Orchestration

This section contains information on orchestration tasks that make use of the Nuvla service.

3.3.1 Application Lifecycle

Jargon

Nuvla has some jargon associated with it, that it would be helpful to know.

Image A reference to a virtual machine image in the supported clouds. Usually something like Ubuntu or CentOS.

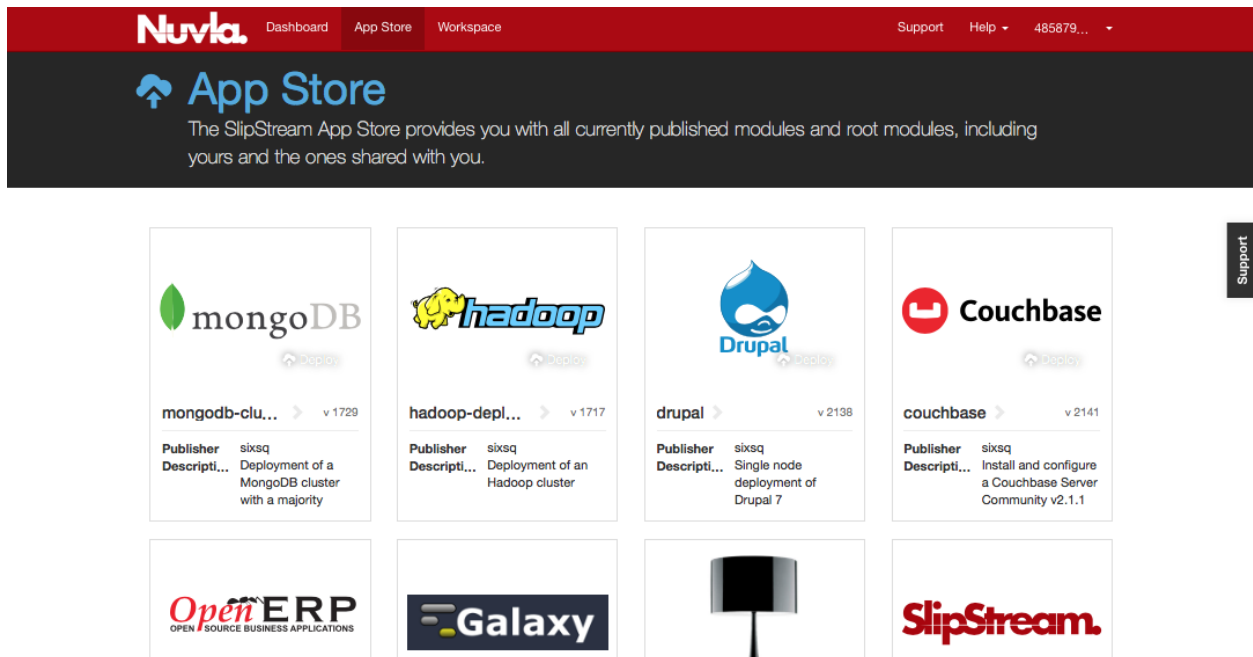
Component A single machine service that references an image and defines a set of recipes to install and configure the necessary software. This is something like R-Studio or Wordpress.

Application A service that consists of multiple machines that is managed as a unit. Examples of these are things like a SLURM batch cluster or a Docker Swarm cluster.

Generically, all these are called “modules”.

App Store

The first place to look for interesting components (single virtual machine services) and applications (multiple machine services) is the [Nuvla App Store](#).



Within the [Nuvla Workspace](#), there are other applications of interest:

- `examples/images`: Minimal distributions of common operating systems. Usually used as the basis for other components.

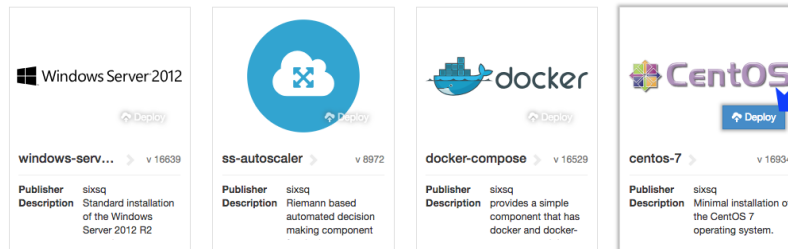
- **apps:** Curated list of applications that can be used as examples for your own applications.
- **HNSciCloud:** [This](#) workspace contains several prearranged components and applications to facilitate the testing and evaluation process, including for example:
 - **HNSciCloud/Benchmarking:** Both generic and HNSciCloud-specific benchmarks for evaluating the system. Relevant for Test Cases 2.2, 5.1 and 11.4.3.
 - **HNSciCloud/Images:** A subset of `examples/images`, containing only the HNSciCloud specific operating systems.
 - **HNSciCloud/VMProvisioningPersonalisation:** An App for testing the provisioning and contextualization of a VM, according to Test Case 2.5.
 - **HNSciCloud/S3EndpointTest-Exoscale_OTC:** An App for testing S3 in both Exoscale and OTC, according to Test Case 2.3.
 - **HNSciCloud/HDF5_IO:** An App for verifying HDF5 compliance with the VMs' local storage in the cloud, according to Test Case 4.1.

Other application definitions will appear over time. If you have specific needs, contact SixSq support to request new ones.

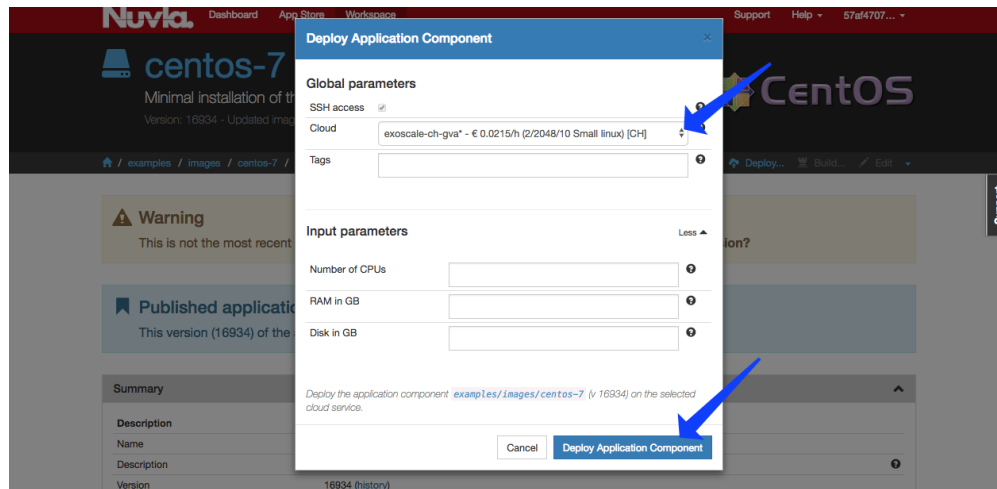
Deploying an Image

The simplest thing to deploy is an image.

Look through the App Store and find the Ubuntu or CentOS image then click on the “deploy” button.



In the deployment dialog, choose the cloud/region you want to use. You can also provide tags or change the requested resources by clicking on “More”. When you’re ready, click on the “Deploy Application Component” button.



Follow the progress from the dashboard or deployment detail page.

The screenshot shows the Nuvla Dashboard interface. At the top, there's a navigation bar with 'Dashboard', 'App Store', 'Workspace', 'Support', and 'Help'. The main header says 'Dashboard' and 'Control and monitor your cloud activity'. Below this, there's a 'Usage' section with three gauges for 'All Clouds', 'exoscale-ch-gva', and 'exoscale-de-fra'. The 'All Clouds' gauge shows 1 VMs out of 1732. The 'exoscale-ch-gva' gauge shows 1 VMs out of 20. The 'exoscale-de-fra' gauge shows 0 VMs out of 20. There's a '+ Add a new cloud' button. Below the usage section, there's a 'Deployments' section with a 'Virtual Machines' tab. It shows a table of deployments with columns: ID, Application / Component, Service URL, State, VMs, Start Time [UTC], Clouds, Tags, and User. The first row shows a deployment with ID 'b108be18', Application / Component 'centos-7 16934', Service URL 'ssh://root@185.19.29.12', State 'Ready', 1 VM, Start Time '2018-08-23 14:27:09', Clouds 'exoscale-ch-gva', and User '57af4707-286a-4859-83d1-...'. There are blue arrows pointing to the 'ID' and 'Service URL' columns with labels 'deployment details' and 'service page' respectively. At the bottom, there's a footer with copyright information and a 'Powered by SlipStream' logo.

Once the machine is in the “Running” state, you can log into the machine via SSH. This can be done via the “Service URL” or manually from a terminal.

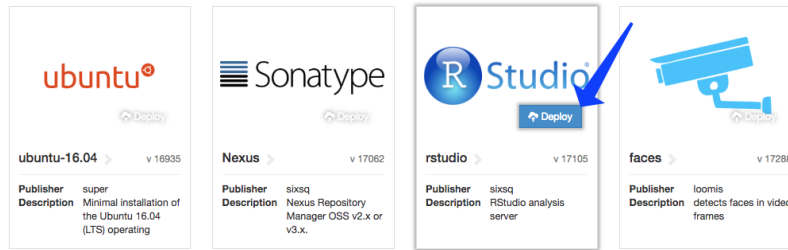
Terminate the machine by clicking on the “cross” icon in the Dashboard or the “Terminate” action on the deployment detail page.

This screenshot is a zoomed-in view of the deployment table from the previous image. It shows the same table with columns: ID, Application / Component, Service URL, State, VMs, Start Time [UTC], Clouds, Tags, and User. The first row is highlighted. A blue arrow points to a small 'x' icon (cross icon) at the end of the first row, indicating the terminate action.

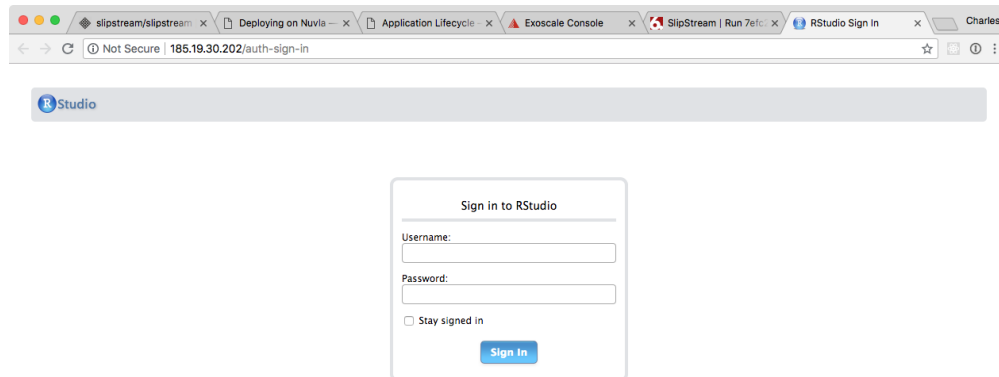
Deploying a Component

We will continue by deploying a component. The example we will use is an R-Studio server. R-Studio is a web application that provides easy use of the R statistical language.

Find the R-Studio application in the App Store. Click on “deploy” and follow the same process as before. You can follow the status as you did for the image deployment.



While waiting for the component to start, you might want to look at the [component definition](#) to see how a component is defined. Once it is in the “Running” state, you can bring up the R-Studio interface by clicking on the “Service URL”.

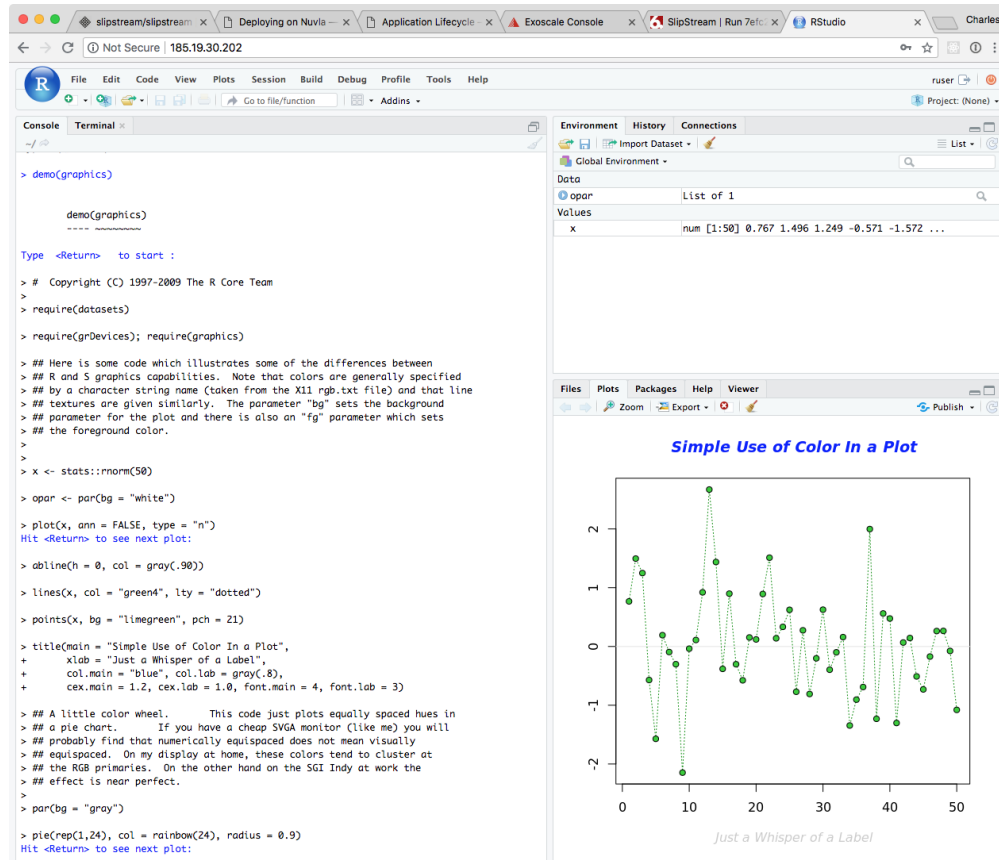


Each instance gets its own randomly generated password. To find it, look at the deployment detail page, open the “Parameters” section, and then find the values for “rstudio-user” and “rstudio-password”.

The screenshot shows the Nuvla dashboard with a table of machine configuration details. A blue circle highlights the 'machine:ram_gb' field and the 'machine:studio_pwd' field.

Name	Value
machine:abort	
machine:cloudservice	exoscale-ch-gva
machine:complete	false
machine:cpu.nb	
machine:disk.GB	
machine:disk.attach.size	
machine:disk.attached.device	
machine:disk.detach.device	
machine:exoscale-ch-gva.disk	10
machine:exoscale-ch-gva.instance.type	Small
machine:exoscale-ch-gva.security.groups	slipstream_managed
machine:extra.disk.volatile	
machine:hostname	185.19.30.202
machine:image.id	Linux Ubuntu 16.04 LTS 64-bit
machine:image.platform	ubuntu
machine:instanceid	e31aa94f-e41c-491f-a039-1b1976484bb6
machine:is.orchestrator	false
machine:network	Public
machine:pre.scale.done	false
machine:ram_gb	
machine:studio_pwd	g2ITNx0ZYQI=
machine:studio_user	ruser
machine:scale.done	false
machine:scale.state	operational
machine:service.offer	service-offer/64c4468c-8cb9-42a4-809a-3245dbcd2bc
machine:state.custom	RStudio Ready!
machine:url.service	http://185.19.30.202 ➔
machine:url.ssh	ssh://root@185.19.30.202 ➔
machine:vmstate	running

Use the username and password to log in. You can see if it is working by trying the `demo (graphics)` command in the R console.



You can terminate the machine on the Dashboard page or the deployment detail page, as before.

Deploying an Application

For SlipStream an “application” consists of a deployment with multiple virtual machines. To assist with the lifecycle management of applications, Nuvla will deploy one “orchestrator” machine per cloud.

To deploy, the example application:

- Ensure that you are logged into Nuvla.
- Navigate to the [Docker Swarm](#) application or choose this application from the App Store.
- Click on the *Deploy* action.
- Click on the *Deploy Application* button in the dialog.

You should not need to change anything in the deployment dialog, although, you may add tags to your deployment.

You will again be redirected to the Nuvla Dashboard at the end of the process. This application will run a Docker Swarm cluster with one Master and one Worker by default; you can change the number of workers in the deployment dialog if you want. This will also take a few minutes to complete.

If you have time, you can log into the master node with SSH and run a container on this swarm. You might want to try to run the nginx webserver:

```

# deploy nginx
docker service create --name my-swarm-nginx -d -p 8080:80 nginx

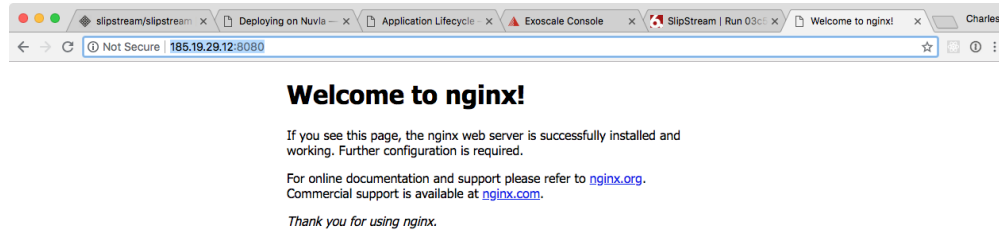
```

(continues on next page)

(continued from previous page)

```
# show status
docker service ps my-swarm-nginx
```

Using the IP address of the master or any worker should show the nginx welcome page.



You can use the same methods to terminate the Swarm cluster when you are finished with it.

3.3.2 IMPORTANT - Scaling Guidelines

In the past, some scalability issues have been observed when provisioning big deployment through Nuvla.

While this issue is being worked on, technical users are kindly asked to respect the following deployment guidelines in Nuvla.

Deploying Scalable Applications

When deploying a scalable application, users should **not scale up or down by more than 100 VMs at the time**. Also, users should **keep the total number of VMs per application deployment under 300**.

Deploying Single Components

For single component deployment, **each user is able, for now, to deploy up to 500 instances**.

3.3.3 Start a Linux VM

The main interface for managing virtual machine deployments is [Nuvla](#). The procedure to deploy standard Linux-based VMs is straightforward:

- Navigate to the component/image you want to deploy,
- Click on the “deploy” button,
- Choose the cloud/offer to use in the deployment dialog, and
- Provision the image by clicking the “deploy” button.

You will then be taken to the dashboard where you can follow the progress of your deployment. Detailed documentation can be found on the SlipStream documentation website, specifically the [deployment documentation](#).

Once the deployment reaches the “ready” state, you can log into your virtual machine via SSH. Be sure that you have configured your Nuvla account with your public SSH key.

The common “native” images that are supported across clouds can be found in the `examples/images` project within the Workspace. In general, Ubuntu 14/16, Debian 7/8, and CentOS 7 are well supported across clouds.

Other Linux distributions may be supported as well. Either you can ask SixSq (through support) to extend the list of available images or create your own [native image](#) component.

3.3.4 Start a Windows VM

The main interface for managing deployments is [Nuvla](#). Deploying Windows VMs is straightforward and the same as for Linux VMs:

- Navigate to the component/image you want to deploy,
- Click on the “deploy” button,
- Choose the cloud/offer to use in the deployment dialog, and
- Provision the image by clicking the “deploy” button.

You will then be taken to the dashboard where you can follow the progress of your deployment. Detailed documentation can be found on the SlipStream documentation website, specifically the [deployment documentation](#).

Once the deployment reaches the “ready” state, you can log into your virtual machine via Microsoft Remote Desktop.

The supported Windows images can be found in the `examples/images` project within the Workspace.

3.3.5 Deploy Docker Containers

[Docker](#) is a convenient system for defining, packaging, and deploying applications. It uses container technologies to allow portability between operating systems and to achieve fast start-up times. By default it will use images from the [Docker Hub](#), an open registry of containers.

On typical IaaS cloud infrastructures, you must first deploy a virtual machine, install Docker ([Docker Compose](#)), and then start your container. SixSq provides a Docker Compose recipe on [Nuvla](#) to make the installation of software and the deployment of containers easy.

To launch the Docker Compose virtual machine, find the [Docker Compose Component](#) either in the App Store or the Workspace (or by clicking the link!).

- Click on “Deploy”
- Choose the cloud you want to use, and
- Wait for the virtual machine to start.

Once the machine is ready, you can log into the machine via SSH using the Service URL link or manually. Once you are on the machine, you can then use Docker and Docker Compose as you normally would.

For example, a simple “Hello World” example:

```
$ docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
b04784fba78d: Pull complete
Digest: sha256:f3b3b28a45160805bb16542c9531888519430e9e6d6ffc09d72261b0d26ff74f
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

...
```

This will show you a message and indicate that the installation is working correctly. It will also provide some pointers for doing more complicated (and useful) tasks.

For example, you can try something similar from the [Docker Compose Getting Started](#) page. Following the instructions there, you can deploy a simple web application that provides a welcome message with a counter of how many times it has been loaded.

```
~/composetest$ docker-compose up
Creating network "composetest_default" with the default driver
Building web
Step 1/5 : FROM python:3.4-alpine
3.4-alpine: Pulling from library/python
acb474fa8956: Pull complete

...

redis_1 | 1:M 26 Jun 07:32:59.588 * The server is now ready to accept connections on
↪port 6379
web_1 | * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
web_1 | * Restarting with stat
web_1 | * Debugger is active!
web_1 | * Debugger PIN: 241-972-540
```

Note: Instead of using `localhost` or `0.0.0.0`, you will need to use the **IP address of the virtual machine** (with the 5000 port!) from your remote browser. If everything worked correctly, you should see a message like “Hello World! I have been seen 2 times.”.

From here you might want to look at the entries [Deploy Docker Swarm](#) or [Deploy a Kubernetes Cluster](#).

3.3.6 Deploy Docker Swarm

Running Docker in “[Swarm](#)” mode allows you to deploy and control a cluster of machines running the Docker Engine. The Docker Swarm component in [Nuvla](#) automates the installation, configuration, and deployment of a swarm.

The [Docker Swarm component](#) can be found in the Workspace or App Store on Nuvla (or by clicking the link!). After finding the component, you can deploy a swarm by:

- Clicking on “deploy”,
- Choosing whether the deployment is [scalable](#),
- Choosing the number of masters and workers, and
- Launching the swarm by clicking on “Deploy” in the deployment dialog.

Once the deployment is ready, you can check the status of the swarm with:

```
$ docker info

...

Swarm: active
NodeID: ra0mpoiy6pnv7j6xnlbav3e12
Is Manager: true
ClusterID: 8gux86j2845pzs9uuwftgqnfu
Managers: 1
Nodes: 4
```

(continues on next page)

(continued from previous page)

...

If you are logged into a manager, you should see that the “Is Manager” flag is true. Here a 5 node deployment was started with 1 manager and 4 workers.

You can get more details about the swarm by looking at the node status from a manager machine:

```
$ docker node ls
ID                                HOSTNAME                                STATUS  _
↪ AVAILABILITY  MANAGER STATUS
o476smd7wm3s97rin0u6nn84q        worker2310ef996-af16-4815-8ba7-c88d630d95f4  Ready  _
↪ Active
ra0mpoiy6pnv7j6xnlbav3e12 *      master1310ef996-af16-4815-8ba7-c88d630d95f4  Ready  _
↪ Active      Leader
vl3c4ypaguwglypr5qko6zkgu        worker3310ef996-af16-4815-8ba7-c88d630d95f4  Ready  _
↪ Active
vvz3beena6t4200bhag331d5b        worker1310ef996-af16-4815-8ba7-c88d630d95f4  Ready  _
↪ Active
```

You should see a listing like the one above, if everything has worked correctly. The cluster is ready to be used.

To deploy a service to the swarm, you can follow the [docker swarm service](#) tutorial. From the manager node, you can deploy, list, inspect and remove the services as follows:

```
$ docker service create --replicas 1 --name helloworld alpine ping docker.com
gzhyxwm0jp4ddesv56g9gcv7

$ docker service inspect --pretty gzhyxwm0jp4d

ID:          gzhyxwm0jp4ddesv56g9gcv7
Name:        helloworld
Service Mode: Replicated
  Replicas: 1
Placement:
UpdateConfig:
  Parallelism: 1
  On failure: pause
  Max failure ratio: 0
ContainerSpec:
  Image: _
↪ alpine:latest@sha256:b09306f2dfa3c9b626006b2f1ceeeaa6fcbfac6037d18e9d0f1d407260cb0880
  Args:      ping docker.com
Resources:
Endpoint Mode: vip

$ docker service ps gzhyxwm0jp4d
ID            NAME           IMAGE           NODE                                _
↪ DESIRED STATE  CURRENT STATE  ERROR  PORTS
x9twsry8knc  helloworld.1  alpine:latest  master1310ef996-af16-4815-8ba7-
↪ c88d630d95f4  Running      Running about a minute ago

$ docker service rm gzhyxwm0jp4d
gzhyxwm0jp4d
```

See the Docker [Swarm](#) documentation for scaling and other management actions for your Docker applications.

3.3.7 Deploy a Kubernetes Cluster

[Kubernetes](#) is an open-source system for automating deployment, scaling, and management of containerized applications. With [Nuvla](#), you can deploy your own Kubernetes cluster automatically.

The [Kubernetes component](#) can be found in the Workspace or App Store on Nuvla (or by clicking the link!). After finding the component, you can deploy a cluster by:

- Clicking on “deploy”,
- Choosing whether the deployment is [scalable](#),
- Choosing the number of workers, and
- Launching the cluster by clicking on “Deploy” in the deployment dialog.

Once the cluster is ready, you can SSH into the master node. From here, list all the nodes and check their current status with:

```
$ kubectl get nodes
NAME                STATUS    AGE
159.100.240.160     Ready    2m
159.100.240.193     Ready    2m
159.100.240.64      Ready    2m
```

You should see output similar to the above if all the nodes have been correctly deployed and configured.

After checking the status of the cluster, you can then use the `kubectl` command to deploy and control services on the Kubernetes cluster.

If you are unfamiliar with Kubernetes, you can follow the tutorials and look at the reference documentation on the [Kubernetes](#) website.

For a simple test deployment of [nginx](#) (a web server), create a file `nginx.json` with the following contents:

```
{
  "kind": "Pod",
  "apiVersion": "v1",
  "metadata": {
    "name": "nginx",
    "namespace": "default",
    "labels": {
      "name": "nginx"
    }
  },
  "spec": {
    "containers": [{
      "name": "nginx",
      "image": "nginx",
      "ports": [{"hostPort": 80, "containerPort": 80}]
    }]
  }
}
```

You can then deploy this web server with the command:

```
$ kubectl create -f nginx.json
pod "nginx" created
```

You can then discover what node is running the webserver with:

```
$ kubectl describe pod nginx

$ kubectl describe pod nginx | grep Node:
Node:                159.100.240.64/159.100.240.64
```

In this case the server is running on the node 159.100.240.64. You can point a browser to this node and then you should see the nginx welcome page.

Afterwards, you can run the command:

```
$ kubectl delete pod nginx
pod "nginx" deleted
```

to remove the nginx deployment from the cluster.

3.3.8 Create Components and Applications

SlipStream (and Nuvla) allow you to define portable cloud components and applications. For SlipStream “components” are single virtual machine services and “applications” are multiple virtual machine services make up of one or more components.

To ensure that the components and applications are portable, SlipStream uses an application model that described customized images as a set of deltas (recipes) that describe the changes from “native images”.

In other words, components define a template, including resources characteristics such as default CPU, RAM and disk. They also define recipes (aka scripts) that are executed as part of the deployment, to transform the image into a fully configured VM. The native images are normally minimal operating system images that are have consistent behavior across clouds.

The SlipStream tutorial contains an [extensive section](#) on how to create your own components and applications. Please refer to that tutorial. For cases where quick start up times are important, you can also [build images](#) on clouds that support it.

3.3.9 Create API Key/Secret

Nuvla supports the use of generated API key/secret pairs for accessing the service. Compared to other authentication methods, they provide more control over access granted to clients accessing Nuvla via the API and command line.

The API key/secret pairs have the following advantages over using other authentication mechanisms:

- Many independent pairs can be created allowing fine-grained control over programmatic access by clients.
- API key/secret pairs created with a predefined lifetime (time-to-live, TTL), disallow access after the TTL has expired.
- Long-lived clients can use API key/secret pairs with an unlimited lifetime to simplify credential management.
- **Any API key/secret can be revoked at any time and independently of any other credentials.**

The internal process for handling authentication when using API key/secrets is the following:

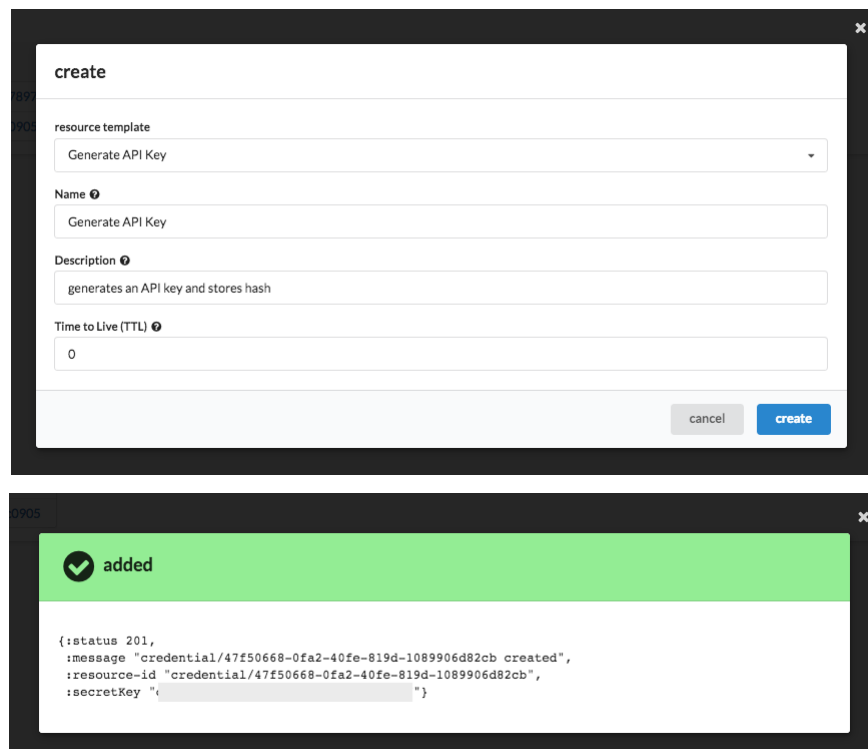
1. Create an API key/secret pair, saving the secret provided in the response.
2. Use the API key/secret to authenticate against the Nuvla service.
3. The server responds with a time-limited session cookie that must be used with subsequent requests to the server.
4. When the session cookie expires, the client must use the API key/secret pair to re-authenticate with the server.

While the API key/secret can be revoked, the session cookie is an **irrevocable access token** with limited lifetime. Consequently, after an API key/secret has been revoked, there is a window of time where active session cookies will still allow access. **The maximum lifetime of a session cookie is fixed at 24 hours.**

Creating an API Key/Secret

The easiest method to create an API Key/Secret is via the newer browser interface to Nuvla. The procedure to do this is:

1. Navigate to the [Nuvla WebUI](#).
2. Ensure that you are logged in. Click on the login button in the upper right-hand corner if you are not.
3. Navigate to the search panel for [Credentials](#).
4. Click on the “search” button.
5. Click on the “add” button.
6. In the dialog, select “Generate API Key” if it isn’t already selected.
7. Change the values for “name”, “description”, and “TTL” if desired.
8. Click on “create”.
9. Note the secret provided in the dialog. You will not be able to recover this secret later. Your API key is the “resource-id” field and the secret the “secretKey”.



The top screenshot shows a 'create' dialog box with the following fields:

- resource template: Generate API Key
- Name: Generate API Key
- Description: generates an API key and stores hash
- Time to Live (TTL): 0

The bottom screenshot shows a confirmation message 'added' with a green checkmark. Below the message is a JSON response:

```
{:status 201,
 :message "credential/47f50668-0fa2-40fe-819d-1089906d82cb created",
 :resource-id "credential/47f50668-0fa2-40fe-819d-1089906d82cb",
 :secretKey "..."}
```

Using the API Key/Secret

You can use the API key/secret to log in via the [REST API](#), [Python API](#), [Clojure API](#), and [Libcloud driver](#).

Revoking an API Key/Secret

When logged into Nuvla via the API, revoking an API key/secret corresponds to deleting the credential. This can be accomplished by doing the following:

```
$ ss-curl \
-X DELETE \
https://nuv.la/api/credential/05797630-c1e2-488b-96cd-2e44acc8e286
```

Once the credential is deleted/revoked, it can no longer be used to authenticate with Nuvla.

3.3.10 Benchmark

Nuvla provides a benchmarking infrastructure, which can be used by any authenticated user.

The benchmark entries themselves comply with an open schema where users can publish benchmarking data in a consistent and flexible way. The published benchmark record requires an associated Service Offer attribute, where all the instance's resources are described, and a Credentials attribute, associated with the user running/publishing the benchmark. The remaining message attributes are optional, letting users publish any sort of metrics as long as they belong to a predefined namespace (as described in the official [SlipStream API Documentation](#)).

The benchmarks can then be used to select the best performing clouds and service offers over time, continuously.

To illustrate this feature and build our own knowledge base, we publish benchmarks resulting from our continuous monitoring system. All four regions of the Exoscale cloud are covered, including notably the Geneva and Frankfurt regions used by HNSciCloud.

The published benchmarks are obtained by running the *'Unixbench'* suite to measure CPU performance through fast synthetic benchmarks like *Whetstone* and *Dhrystone*, every hours.

As an example, the following image shows the CPU performances on both Exoscale data centers over a period of 12 hours. In the image it is possible to evaluate the consistency of the CPU performance by plotting the average benchmark scores plus two edge (low and high) percentiles, which provide a general view of the CPU MIPS range. The shorter the range, the more consistent the CPU performance is.



For more details on the benchmark resource, including usage examples, refer to the [benchmark API documentation](#).

3.3.11 Batch Clusters

High Performance Computing (HPC) involves the execution of a CPU-intensive application with a particular set of input parameters and input data sets. Because of the large CPU requirements, these applications normally use the Message Passing Interface (MPI) to create a high-performance platform from a sizable number of discrete machines.

These types of applications are naturally job or task-based and historically have been run on batch systems such as [Slurm](#), or [HTCondor](#). These systems can be run within cloud infrastructures, although they generally lead to a significant amount of incidental complexity and service management overheads.

An example SlipStream application for a [Slurm cluster](#) is provided. This example deploys a fully functioning Slurm cluster with the following characteristics:

- One master node and multiple workers (two by default).
- The `/home` area exported by the master to all of the workers.
- A `/scratch` area with local scratch space on all nodes.
- A single, default SLURM partition containing all nodes.
- Common software packages (e.g. `g++`, `OpenMPI`) are installed.
- The root account on the master can access all workers via SSH.
- Parallel SSH has been installed to facilitate cluster management.

The cluster can be used from the `tuser` account and managed through the `root` account on the master node.

Starting the Cluster

To deploy the cluster, navigate to the “slurm-cluster” application within Nuvla and click the `Deploy...` action. You can choose which cloud infrastructure to use, the number of workers, and their size from the deployment dialog.

You can also deploy this application from the command line using the [SlipStream client](#). Before using any of the SlipStream commands, you will need to authenticate with Nuvla using the `ss-login` command:

```
$ ss-login --username=<username> --password=<password>
```

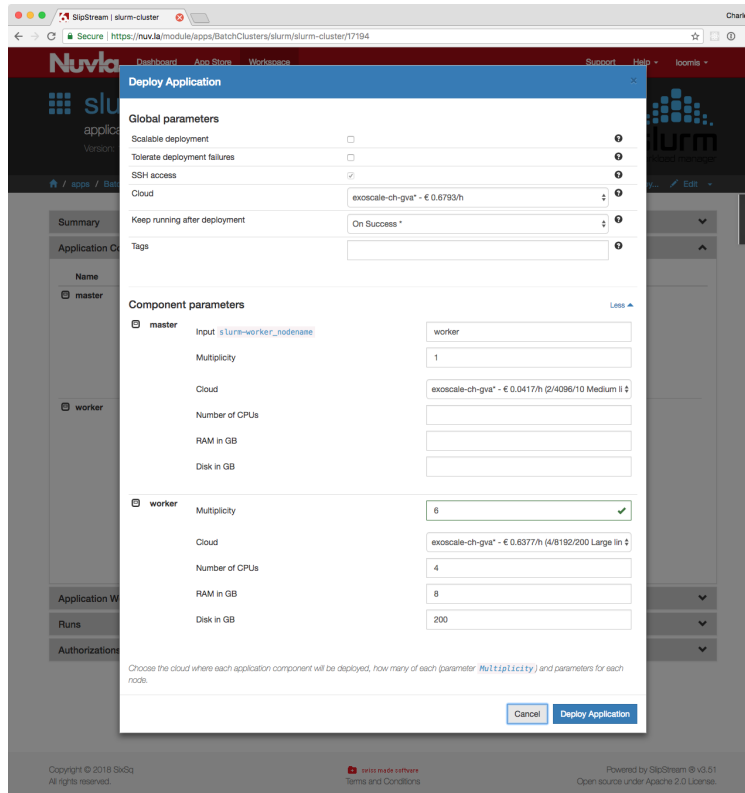
On success, this will exit with a return code of 0 and store an authentication token locally for the SlipStream client commands. You can use the `ss-logout` command to delete this authentication token.

You can now start a SLURM cluster with the `ss-execute` command:

```
$ ss-execute \
  --parameters="worker:multiplicity=4,
               worker:instance.type=Huge" \
  --keep-running=on-success \
  --wait=20 \
  --final-states=Done,Cancelled,Aborted,Ready \
  apps/BatchClusters/slurm/slurm-cluster

::: Waiting 20 min for Run https://nuv.la/run/1a90f7df-a8db-4fa8-b2d2-463afa296c5a to
↪reach Done,Cancelled,Aborted,Ready
[2018-19-24-13:19:59 UTC] State: Initializing
[2018-20-24-13:20:20 UTC] State: Initializing
[2018-20-24-13:20:50 UTC] State: Initializing
[2018-21-24-13:21:21 UTC] State: Initializing
[2018-21-24-13:21:51 UTC] State: Provisioning
[2018-22-24-13:22:21 UTC] State: Provisioning
```

(continues on next page)



(continued from previous page)

```
[2018-22-24-13:22:52 UTC] State: Provisioning
[2018-23-24-13:23:22 UTC] State: Provisioning
[2018-23-24-13:23:53 UTC] State: Provisioning
[2018-24-24-13:24:23 UTC] State: Executing
[2018-24-24-13:24:54 UTC] State: Executing
[2018-25-24-13:25:24 UTC] State: Executing
[2018-25-24-13:25:55 UTC] State: Executing
[2018-26-24-13:26:25 UTC] State: Executing
[2018-26-24-13:26:56 UTC] State: Executing
[2018-27-24-13:27:26 UTC] State: Executing
[2018-27-24-13:27:57 UTC] State: Executing
[2018-28-24-13:28:27 UTC] State: Executing
[2018-28-24-13:28:57 UTC] State: Executing
[2018-29-24-13:29:28 UTC] State: Executing
[2018-29-24-13:29:58 UTC] State: SendingReports
[2018-30-24-13:30:29 UTC] State: SendingReports
OK - State: Ready. Run: https://nuv.la/run/1a90f7df-a8db-4fa8-b2d2-463afa296c5a
```

With the given options, the SLURM cluster will contain 4 workers and 1 master. Each of the workers will be of the “Huge” flavor. The command will wait until the cluster reaches one of the given final states. It will also provide you with the deployment (“run”) identifier (the UUID in the “[https://nuv.la/run/...](https://nuv.la/run/)” URL) that can be used to terminate the cluster.

The example shows how to change the number of worker nodes in the cluster with the **worker:multiplicity** parameter. You can also specify the flavor (instance type) of the machine with the **worker:instance.type** parameter. Supported values are: Micro, Tiny, Small, Medium, Large, Extra-large, Huge, Mega, Titan, GPU-small, and GPU-large. Access to the GPU and larger machines must be requested through support. You can also specify the disk size with **worker:disk** and/or **master:disk**.

Use the `--help` option to find out how to set other options for the `ss-execute` command or the SLURM application description for other parameters.

Accessing the Cluster

Once the deployment is in the “Ready” state, you can log into the master node to use the cluster. You can find the IP address for the master node from Nuvla in the deployment details page, or you can get the IP address **after the deployment is ready** with the command:

```
$ ss-get --run=1a90f7df-a8db-4fa8-b2d2-463afa296c5a master.1:hostname
159.100.244.254
```

replacing the run ID with the one for your deployment. The SSH key from your user profile will have been added to the `root` and `tuser` accounts.

Managing the Cluster

The SLURM cluster will have been deployed with common software packages and a batch queue ready to run jobs. Nonetheless, you may want to adjust the node or SLURM configurations. You might want to consult the [SLURM Documentation](#) or [Administrator Quick Start](#) for managing SLURM.

The `root` account on the master node can be used to manage the cluster. To facilitate this, parallel SSH has been installed and the `root` account can access all workers via SSH. Two files have been created in `/root` that list all hosts in the cluster (`hosts-cluster`) and all workers (`hosts-workers`).

From the `root` account on the master, you can, for example, install the package “bc” all nodes with the command:

```
$ parallel-ssh --hosts=hosts-cluster apt install -y bc
[1] 13:58:40 [SUCCESS] worker-1
[2] 13:58:40 [SUCCESS] worker-2
[3] 13:58:40 [SUCCESS] master
```

The command also allows you to see or capture the output from each command. There is also a `parallel-scp` command for distributing files around the cluster.

Running Jobs

Generally, you will want to run your jobs from a non-privileged account. The account `tuser` has been preconfigured for this. You might want to consult the [SLURM Documentation](#) or [User Quick Start](#) for information on using SLURM for running your applications.

The entire `/home` area is exported via NFS to all workers. Consequently, all user accounts have a shared NFS home area across the cluster. Data and/or executables uploaded to the master node will be visible across the cluster.

There is also a `/scratch` area on each node that provides local scratch disk space. Keep in that this scratch space:

- Is fully accessible to all users. Create subdirectories with more restricted permissions if you want to isolate your files from other users.
- Resides on the root partition. Be careful not to fill this space completely as it could have negative consequences for the operating system.

Unlike the standard `/tmp` directory, the `/scratch` area is not subject to the operating system’s periodic clean up. You must manually remove files to free disk space.

Stopping the Cluster

When your calculations have completed, you can release the resources assigned to the cluster by either clicking the `Terminate` action from the deployment detail page in the web application or using the command line:

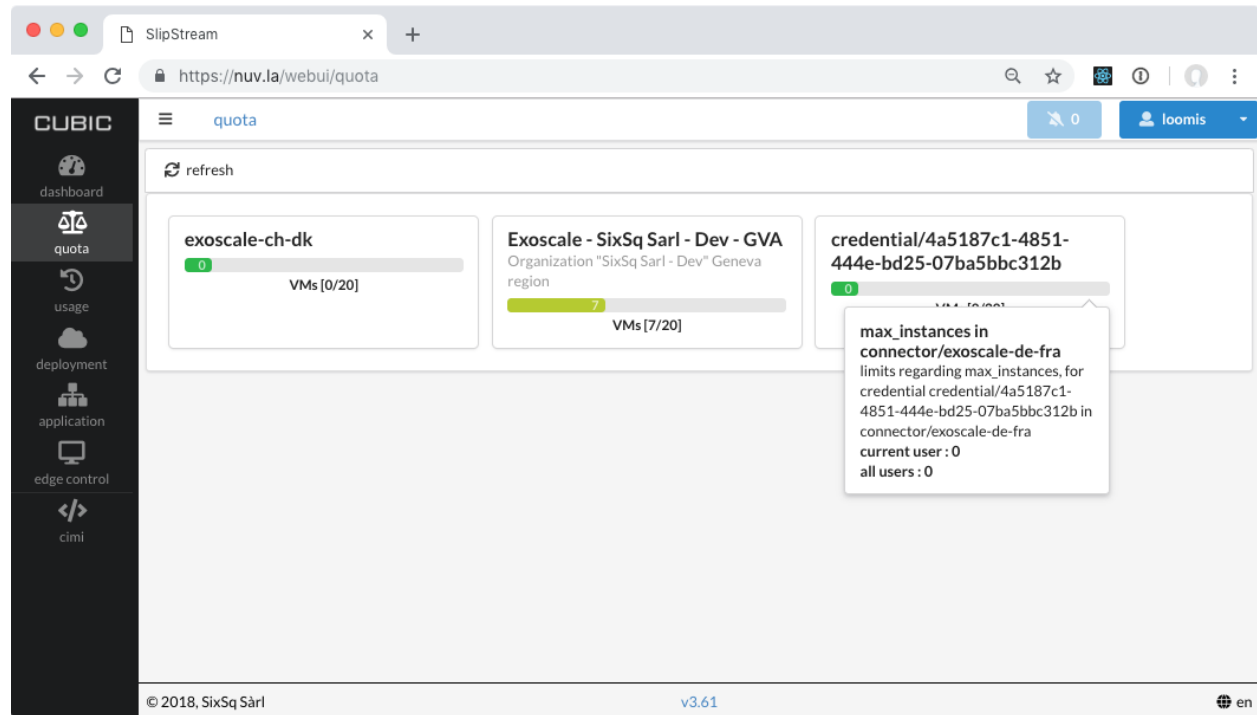
```
$ ss-terminate 98f42dca-98e8-4265-875e-90ddf81d6fca
```

The command line will wait for the full termination of the run.

Warning: All the resources, including local storage, will be released. Be sure to copy your results off the master node to your preferred persistent storage.

3.4 Quota

Nuvla protects users from overuse by applying a quota mechanism. A user's current quotas and quota status can be found on the “[Quota](#)” page of the new browser interface. Hovering over a Quota indicator will provide more details.



When deploying new workloads, Nuvla checks if the requested deployment will exceed the quota. Only if this is not the case, will the deployment be accepted.

Note: Users can request changes to their quotas by asking their group or organization managers, who will then authorize it and pass it along to support.

More details on this feature and how to access it from the API can be found in the [Quota API documentation](#).

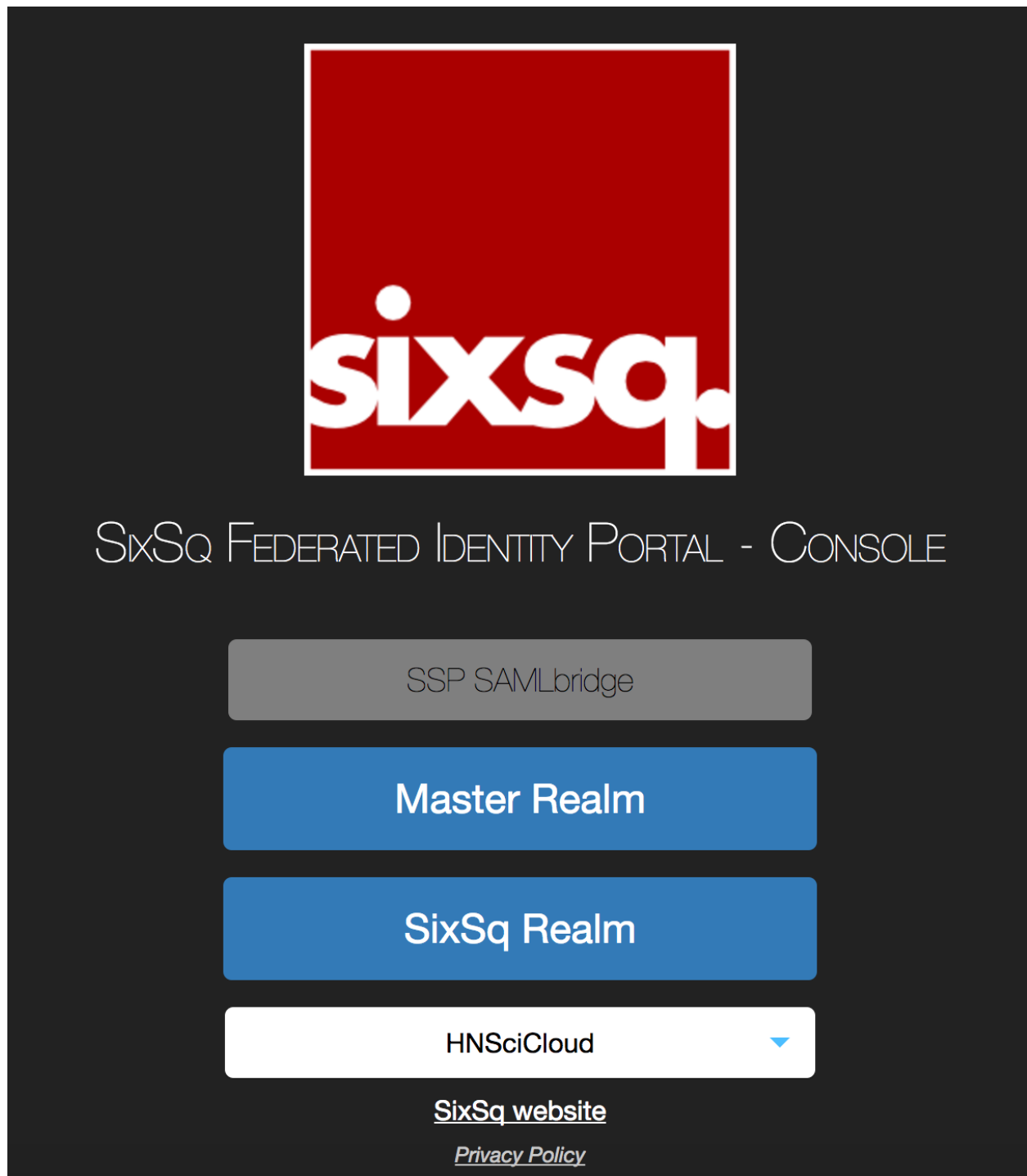
CHAPTER 4

Administrators

This section contains howtos for common tasks that will be carried out by account administrators.

These account administrators are supposed to have followed the [Account Activation](#) steps, and already have been granted the *admin* privileges to their tenant/realm in [SixSq's Federated Identity Portal](#).

To access the account management portal in Keycloak, users need to select their tenant (same as realm) in [SixSq's Federated Identity Portal](#) and login.



4.1 Cloud Provider Configuration

Cloud accounts for each organization (tenant) have been created on the three cloud infrastructures that provide resources to the Rhea Consortium's hybrid cloud platform—Exoscale. An organization's users share access to these accounts via Nuvla. The organization's (tenant's) administrator decides who has access to these credentials.

4.1.1 Cloud Accounts

The general configuration of the cloud accounts follow a hierarchical approach, as shown in the diagram below.

For Exoscale, there is a top-level organization that owns and manages all the Buyers Group tenants. On each tenant then, the respective organization administrator is also given ownership.

With this setup, it is ensured that all the cloud accounts will be automatically setup in Nuvla, given that users have the necessary rights to provision resources.

4.1.2 Granting Access

To grant access to the shared credentials and to allow users to deploy applications on the clouds, each account manager should:

1. Login to [SixSq's Federated Identity Portal](#)
2. Select the users (or groups of users) who need provisioning access, and assign them with the role **can_deploy** (which has already been created).

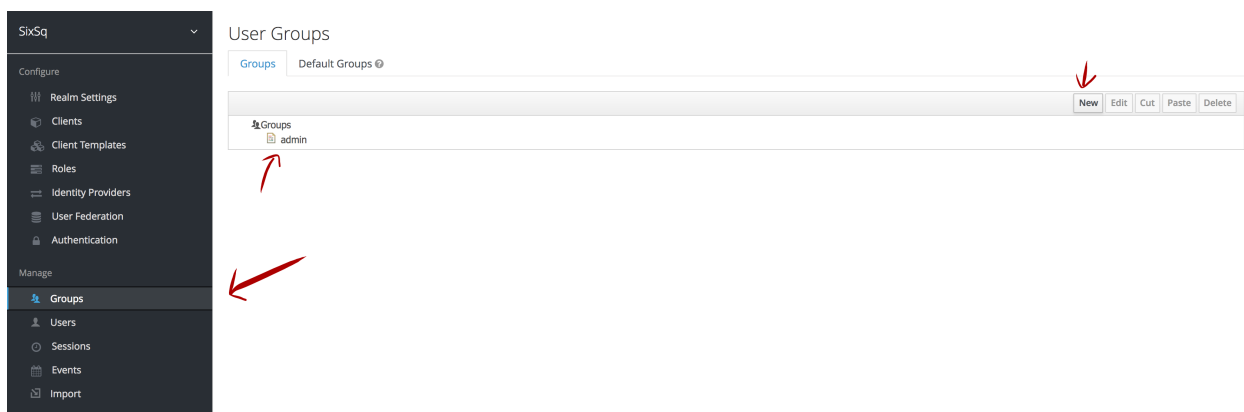
Once this is done, the affected users will automatically get access to the cloud credentials for Exoscale in Nuvla. The assignment of groups or roles is done during the login process, so users may have to logout and login again to have access to new groups or roles.

4.2 Manage Groups

To simplify, groups are to be considered the same as sub-tenants, as they provide the required isolation level.

Groups can be managed through the Keycloak's UI within [SixSq's Federated Identity Portal](#).

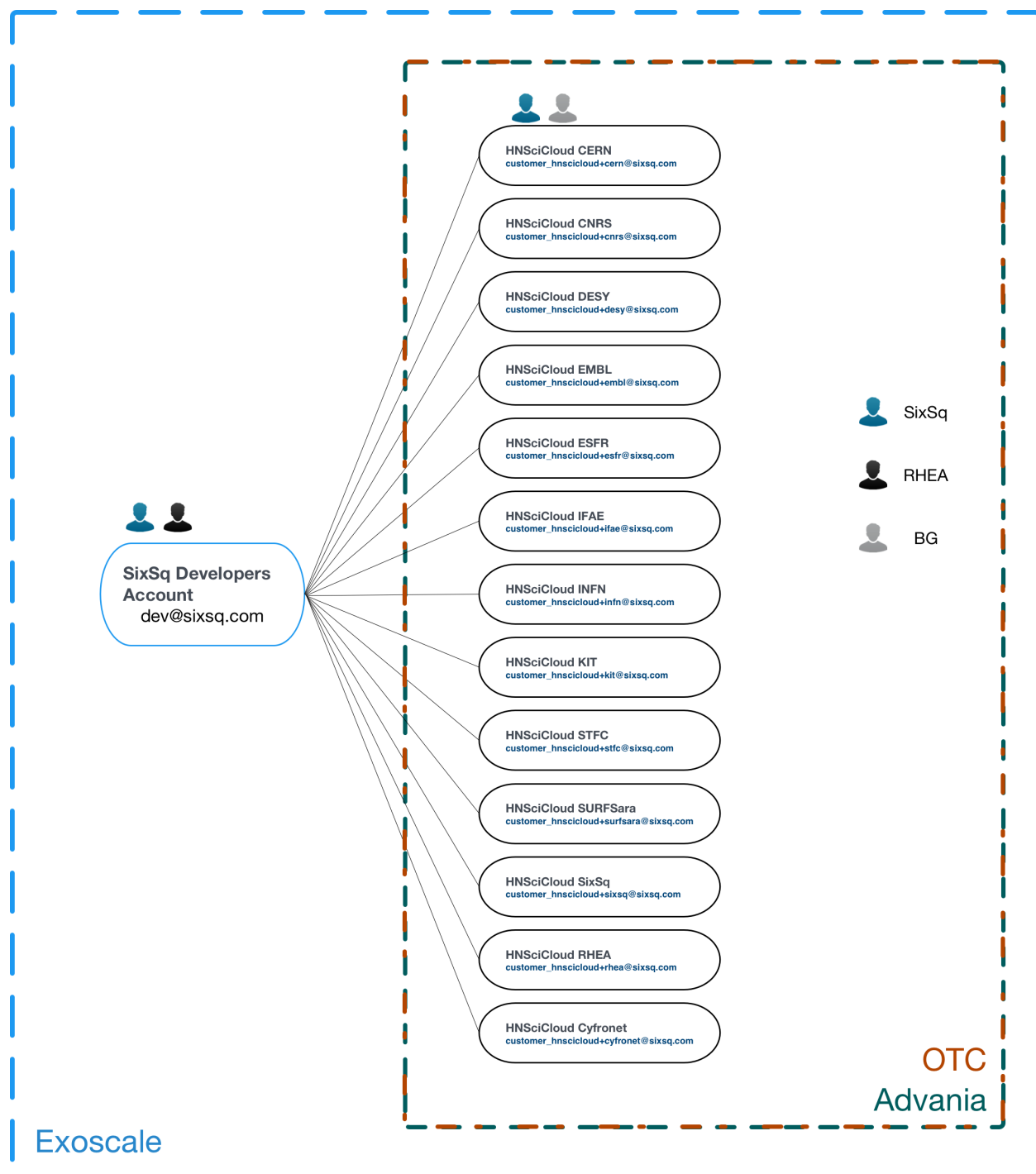
Upon registration in Nuvla, account managers will be given (if requested) *admin* privileges in their respective tenant. Tenant account managers belong to the **admin** group in Keycloak, and have full management privileges within that realm.



Below follows a comprehensive list of actions that can be performed group-wise.

4.2.1 Create New Group

Groups can be nested within each other which allows account managers to define hierarchies. Here we refer to **top level group** and **subgroup** as being the main group and the ones within it, respectively.



The screenshot shows the SixSq user management interface. On the left is a sidebar with navigation options: Configure (Realm Settings, Clients, Client Templates, Roles, Identity Providers, User Federation, Authentication) and Manage (Groups, Users, Sessions, Events, Import). The 'Users' section is selected. The main area displays the user '9ad67dc3aded28dac61e0f5b78b76109acd69c07@elixir-europe.org'. The 'Role Mappings' tab is active, showing a list of available roles (can_deploy, data-manager, one-data-special-role, personalRole) and assigned roles (offline_access, uma_authorization). The effective roles are listed as offline_access, uma_authorization, and view_accounting.

Top Level Group

Without selecting any group (as shown on the figure above), click *New* and enter your group name.

The screenshot shows the 'Create group' form in the SixSq interface. The 'Name' field is highlighted with a red oval and contains the text 'test'. Below the field are 'Save' and 'Cancel' buttons.

Upon creation, a new page will show up with the group settings, as shown below:

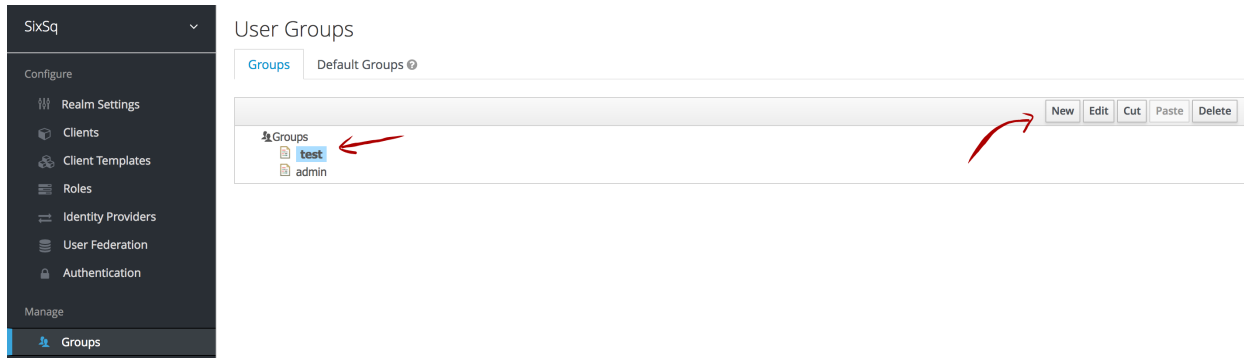
The screenshot shows the 'Test' group settings page in the SixSq interface. The 'Role Mappings' tab is active, showing a list of available roles (offline_access, uma_authorization) and assigned roles. The effective roles are listed as offline_access and uma_authorization. The 'Client Roles' section shows a dropdown menu with 'realm-management' selected, displaying a list of available roles (create-client, impersonation, manage-authorization, manage-clients, manage-events) and assigned roles.

Here is where the account managers can rename the group, assign custom attributes, see the group members (empty

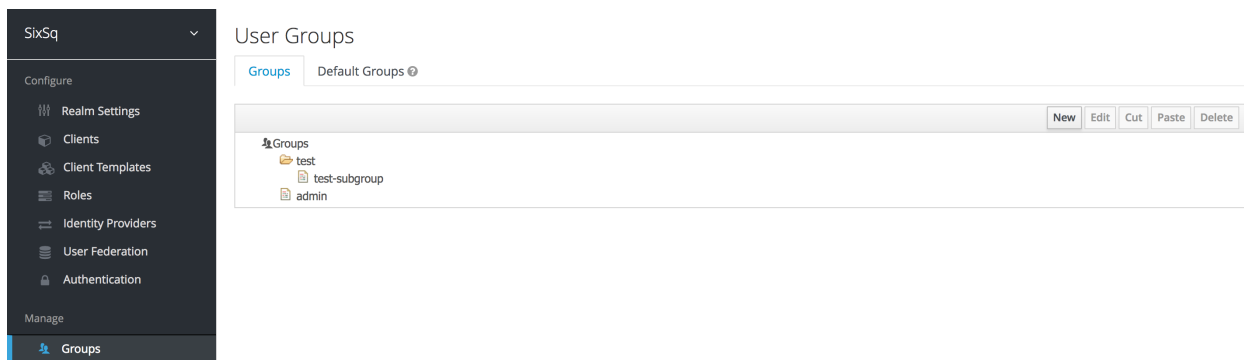
by default), and map the group roles.

Subgroup

To create a group that belongs to a top level one, the creation process is exactly the same as above, but the account manager **has** to select the respective top level group before clicking on *New*:



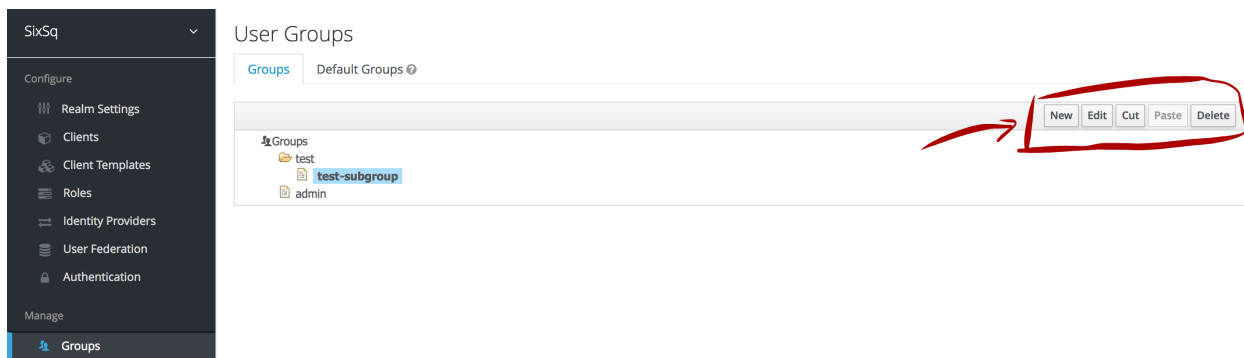
Once created:



NOTE: please note that subgroups will by default inherit the role mappings from the parent top level groups

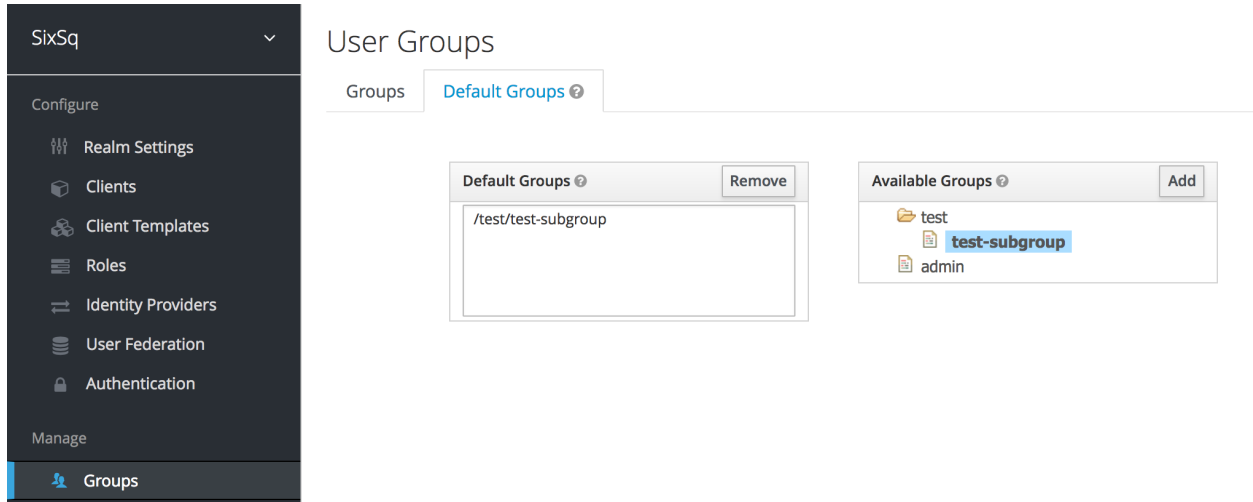
4.2.2 Edit, Cut, Paste and Delete Groups

The button panel on the right corner of the groups' interface allows all the basic group operations:



4.2.3 Default Groups

“Set of groups that new users will automatically join”



As the above figure shows, new users will now automatically be assigned to the subgroup *test-subgroup*.

4.3 Managing Roles

Roles can be managed through the Keycloak's UI within SixSq's Federated Identity Portal.

There are different type of roles and actions that can be performed to manage them:

4.3.1 Realm Roles

Roles that are applied on the tenant level.

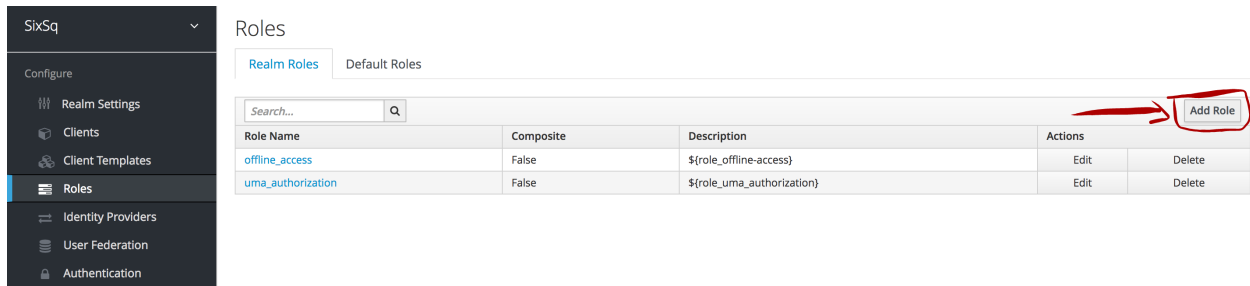


Default Roles

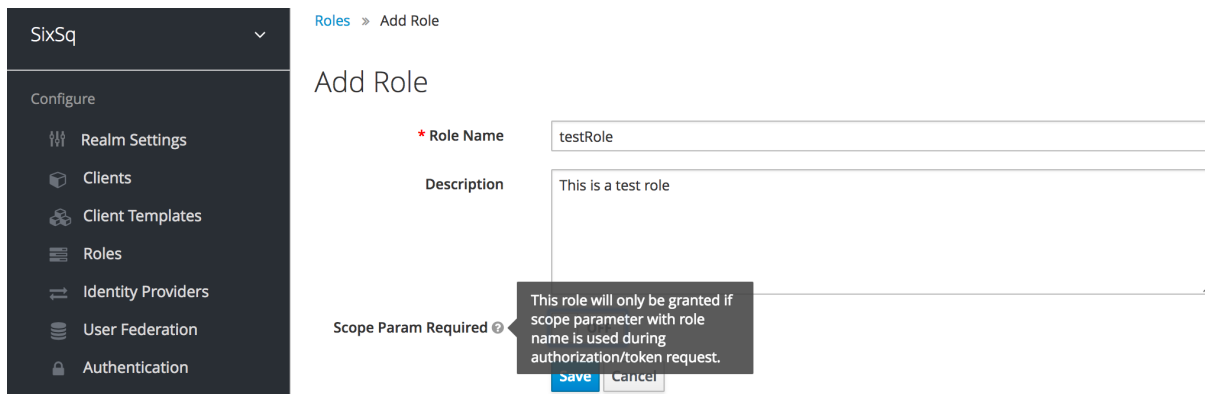
If selected, default roles will automatically be assigned to new users.

Create Roles

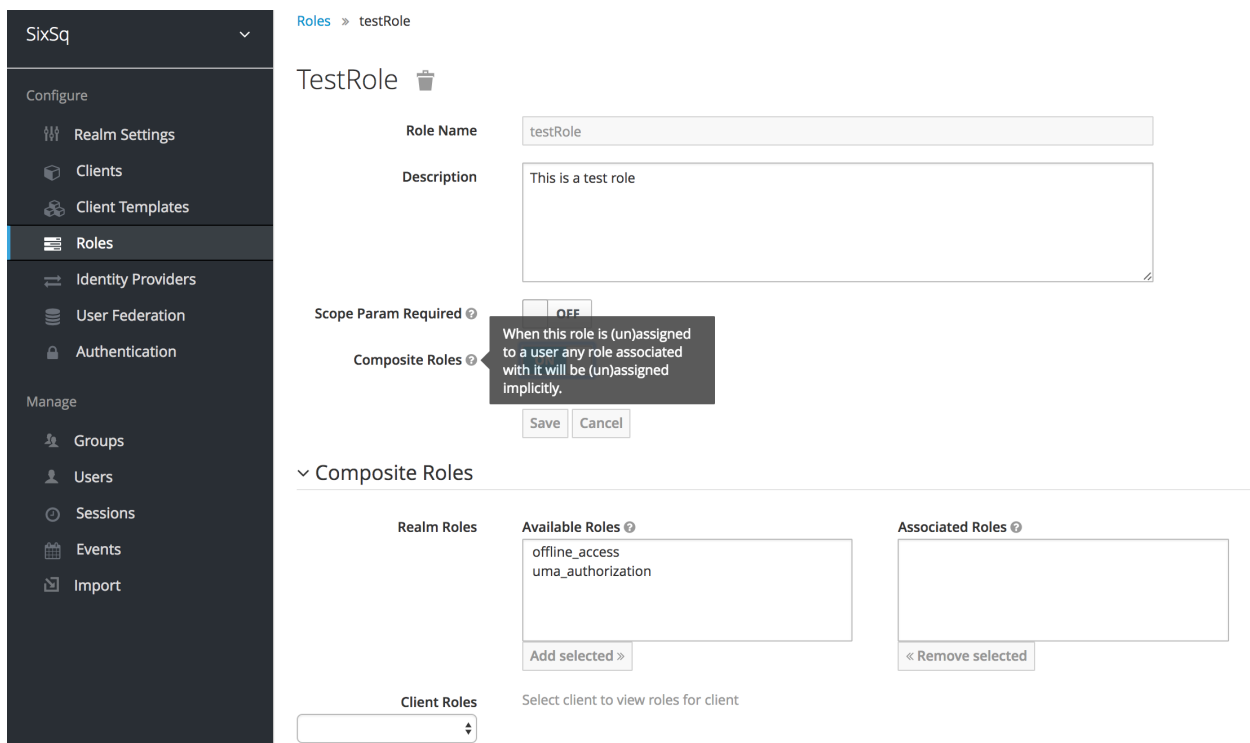
To create a new role, simply click on *Add Role*:



This will open a new form where the account manager can define the role's name, description and whether the role will only be granted if scope parameter with role name is used during authentication/token request:



Once created, account managers will then also have the option to assign composite roles:



NOTE: by default, new rules do not become “Default Roles” for that realm.

Edit and Delete Roles

To edit, simply click on the role name (from the list of roles). To delete, once inside the role edition page, click on the bin icon next to the role name:

The screenshot shows the 'Roles' management page in the SixSq interface. The left sidebar contains a 'Configure' section with 'Roles' highlighted. The main content area shows a list of roles under the 'Realm Roles' tab. The roles listed are 'offline_access', 'testRole', and 'uma_authorization'. The 'testRole' is selected, and a 'Delete Role' dialog box is open, asking for confirmation to delete the role. A red arrow points to the 'Delete' button in the dialog.

Role Name	Composite	Description	Actions
offline_access	False	\${role_offline-access}	Edit Delete
testRole	False	This is a test role	Edit Delete
uma_authorization	False	\${role_uma_authorization}	Edit Delete

4.3.2 Client Roles

Keycloak clients are trusted browser apps and web services in a realm. These clients can request a login.

Account managers can also define client specific roles. IT IS NOT RECOMMENDED that account managers change the roles of already existing clients, as the default tenant clients (and respective Client Templates) are not configured to propagate the user client roles (which are defined on the *Clients* section under the *Scope* tab, for each client).

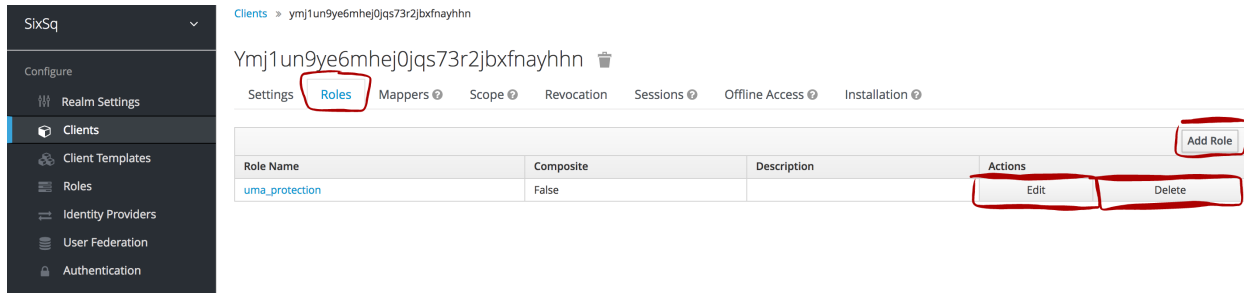
Manage Client Roles

To manage client roles, account manager should first select the desired client from the list

The screenshot shows the 'Clients' management page in the SixSq interface. The left sidebar contains a 'Configure' section with 'Clients' highlighted. The main content area shows a list of clients under the 'Clients' tab. The clients listed are 'account', 'admin-cli', 'broker', 'OneZoneClient', 'realm-management', 'security-admin-console', and 'ymj1un9ye6mhej0q573r2jbxfnayhnh'. The 'account' client is selected.

Client ID	Enabled	Base URL	Actions
account	True	/auth/realms/SixSq/account	Edit Export Delete
admin-cli	True	Not defined	Edit Export Delete
broker	True	Not defined	Edit Export Delete
OneZoneClient	True	Not defined	Edit Export Delete
realm-management	True	Not defined	Edit Export Delete
security-admin-console	True	/auth/admin/SixSq/console/index.html	Edit Export Delete
ymj1un9ye6mhej0q573r2jbxfnayhnh	True	Not defined	Edit Export Delete

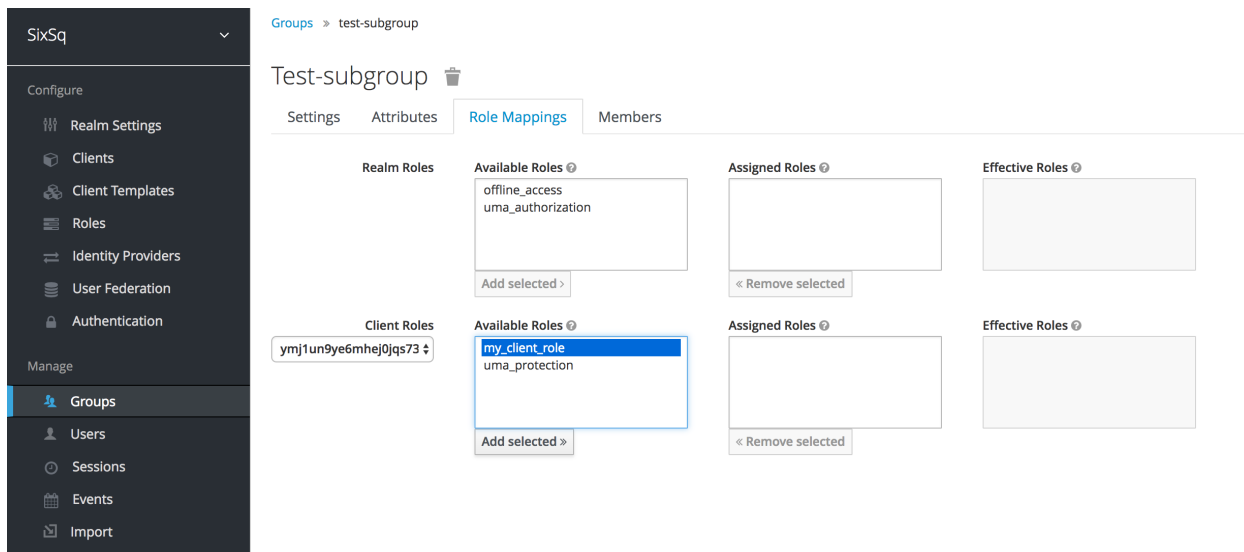
and then click on the *Roles* tab. Here, the account managers will get a list of the client roles and the chance to add new ones as well as edit and delete existing ones:



To add and modify client roles, the interface is exactly the same as stated above for Realm roles.

4.3.3 Mapping Realm and Client Roles to Groups

The instructions on how to map a role to a group can be found in [here](#). Once in the group page, switch to the “Role Mappings” tab and select the desired roles, as shown below.



4.4 Account Management API

It is also possible to manage accounts in Keycloak through its REST API: <http://www.keycloak.org/docs-api/3.1/rest-api/index.html>

4.5 Blacklisting Users

One of the desired user management features for the HNSciCloud project is the ability to block users. This is an action that can be carried by account managers, at any time, removing the user’s access to all client applications in the respective tenant.

There are several ways an account manager can reduce the privileges or even block a user:

4.5.1 Deny User Access to Specific Resources

In some cases, the account manager might only want to grant or remove privileges from a certain user, which will later translate on a forbidden access to a specific resource.

Assuming that all the ACLs have been defined, the account manager should use the roles and groups available in Keycloak to manage the users' access to the deployed resources.

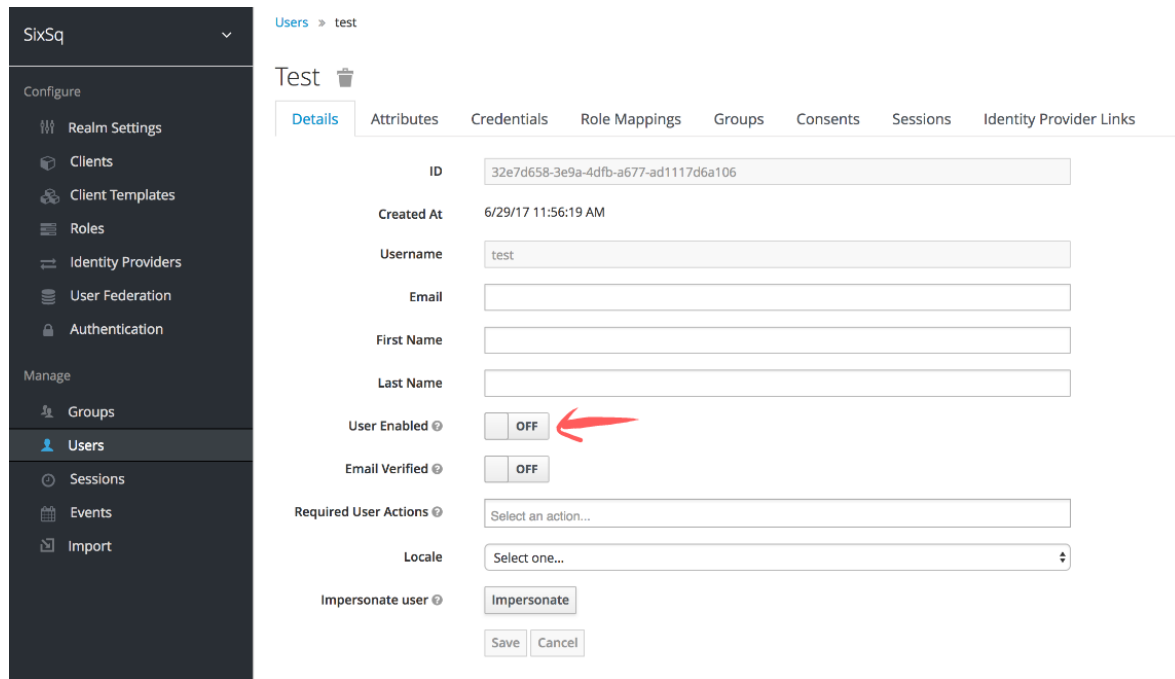
These actions are documented in the [Manage Groups](#) and [Managing Roles](#) sections.

4.5.2 Blocking a Local User

On the occasional scenario where there are local user accounts setup through Keycloak, the account manager can block these users through 3 different ways:

Disable the account

Users with disabled accounts cannot login. The account manager should go the user's view and toggle the *User Enabled* button:



The screenshot shows the Keycloak user management interface. On the left is a sidebar with a menu including 'SixSq', 'Configure', 'Manage', and 'Users'. The 'Users' menu item is selected. The main area displays the details for a user named 'Test'. The 'User Enabled' toggle is currently set to 'OFF', and a red arrow points to it. Other fields include ID, Created At, Username, Email, First Name, Last Name, Email Verified, Required User Actions, Locale, and an Impersonate user button. At the bottom are 'Save' and 'Cancel' buttons.

Change the password

An obvious way to block a user is to simply disable his/her credentials or even change the password:

Remove the user

Finally, a definite and also obvious way to block a user, is simply delete the account:

Authentication Manage	32e7d658-3e9a-4dfb-a67...	test				Edit	Impersonate	Delete
	3e0447e3-c283-4302-914...	test10				Edit	Impersonate	Delete
	10417582-57a3-4bd4-932...	test11				Edit	Impersonate	Delete

4.5.3 Blocking an External User

In most cases, users will be registering into Keycloak (and consequently into the corresponding client applications and services) through external IdPs. These kind of accounts do not have an associated password and therefore it is recommended to have a different approach when blocking users.

To block a user from an external IdP, account manager should add their unique usernames into a **BLACKLISTED_USERNAMES** list, which is crosschecked against every user during login. If a user is blacklisted, he'll be able to register into Keycloak but his/her login will be interrupted and a "blacklisted" message will be shown.

To blacklist a user, account manager should first find the respective username from the users list. Let's assume we want to block user `foo@bar.com` `http://myidp.com/login!etc`. Account managers should then do:

1. go to *Authentication*
2. select in which federation should the user be blacklisted
3. click on *Config* from the *blacklist* script.

This will open a script where account managers can then find a global variable (a list) called *BLACKLISTED_USERNAMES*. The username to block shall then be put into that list, as a string, like shown below:

Then just click **Save**, and that's it. For any other users to be blocked, the procedure is the same, just appending their username to the list above.

Warning: Blocking a user does not revoke nor destroy his/her current session. Blocking actions only affect users on their next login.

4.6 Whitelisting Users

The ability to only permit login to a predefined set of users is also possible available in Keycloak.

As for [blacklisting](#), the account manager can also define a list of usernames that are allowed to login through IdPs into Keycloak and respective clients.

By default new users are not entitled with anything else than the roles to manage and view their own user account, but obviously this can be changed by the account manager by either setting default groups and roles or assigning these to users individually.

4.6.1 Whitelisting a Local User

Local user registration is not enabled in the current SixSq Fed-ID portal, thus new local users have to be manually added by the account manager by creating new users through the “Users” section in the Keycloak UI.

Users

Search...

Q

View all users

Unlock users

Add user

ID	Username	Email	Last Name	First Name	Actions
----	----------	-------	-----------	------------	---------

4.6.2 Allowing an External User

By default, whitelisting for external users is disabled.

In cases where the account manager wants to restrict the login process to a set of known users coming from either eduGAIN or ELIXIR, he/she needs to enable the whitelisting capability in the authentication flows in Keycloak:

Auth Type	Requirement	Requirement	Requirement	Actions
Script (custom_attr_validator_edugain)	<input checked="" type="radio"/> REQUIRED	<input type="radio"/> OPTIONAL	<input type="radio"/> DISABLED	Actions
Script (whitelist)	<input type="radio"/> REQUIRED	<input type="radio"/> OPTIONAL	<input checked="" type="radio"/> DISABLED	Actions
Script (blacklist)	<input checked="" type="radio"/> REQUIRED	<input type="radio"/> OPTIONAL	<input type="radio"/> DISABLED	Actions

PLEASE NOTE: blacklisting has priority over whitelisting, so if a user is both whitelisted and blacklisted, he/she will not be allowed to login and be considered blacklisted anyway.

Once enabled, account managers can add users to their whitelist in a similar way as it is done for the blacklist. Let's assume we want to add user `foo@bar.com` to the whitelist:

1. go to *Authentication*
2. select in which federation should the user be blacklisted
3. click on *Config* from the *whitelist* script.

Authentication

Flows Bindings Required Actions Password Policy OTP Policy

EduGAIN - First Broker Login With Attribute Check ⓘ

New Copy Delete Add execution Add flow

Auth Type	Requirement				
Script (custom_attr_validator_edugain)	<input checked="" type="radio"/> REQUIRED <input type="radio"/> OPTIONAL <input type="radio"/> DISABLED				Actions ▾
Script (whitelist)	<input checked="" type="radio"/> REQUIRED <input type="radio"/> OPTIONAL <input type="radio"/> DISABLED				Actions ▾
Script (blacklist)	<input checked="" type="radio"/> REQUIRED <input type="radio"/> OPTIONAL <input type="radio"/> DISABLED				Delete Config

This will open a script where account managers can then find a global variable (a list) called `WHITELISTED_USERNAMES`. The username to allow shall then be put into that list, as a string, like shown below:

Authentication Flows » EduGAIN - First Broker Login With Attribute Check » whitelist

Whitelist

ID: 8f697958-59f1-44b7-ad22-10c964ca60e2

Alias ⓘ: whitelist

Script Name ⓘ: whitelist

Script Description ⓘ: List of users who are allowed to login

Script Source ⓘ

```

1  /* to whitelist users, add their usernames to the following list. ex:
2     WHITELISTED_USERNAMES = ['usernameAB', 'user123']
3     */
4  WHITELISTED_USERNAMES = ['foo@bar.comhttp://myidp.com/login!etc']
5
6  /*
7   * Template for JavaScript based authenticator's.
8   * See org.keycloak.authentication.authenticators.browser
9   * .ScriptBasedAuthenticatorFactory
10  */
11 // import enum for error lookup
12 AuthenticationFlowError = Java.type("org.keycloak.authentication
13     .AuthenticationFlowError");
14 AuthenticationFlowError =
15     Java.type("org.keycloak.authentication.AuthenticationFlowError");
16 Response = Java.type("javax.ws.rs.core.Response");
17 MediaType = Java.type("javax.ws.rs.core.MediaType");
18
19 /**
20  * An example authenticate function.
21  *
22  * The following variables are available for convenience:
23  * user - current user {@see org.keycloak.models.UserModel}
24  * realm - current realm {@see org.keycloak.models.RealmModel}

```

Then just click **Save**, and that's it. That user can now login, provided the username is not blacklisted.

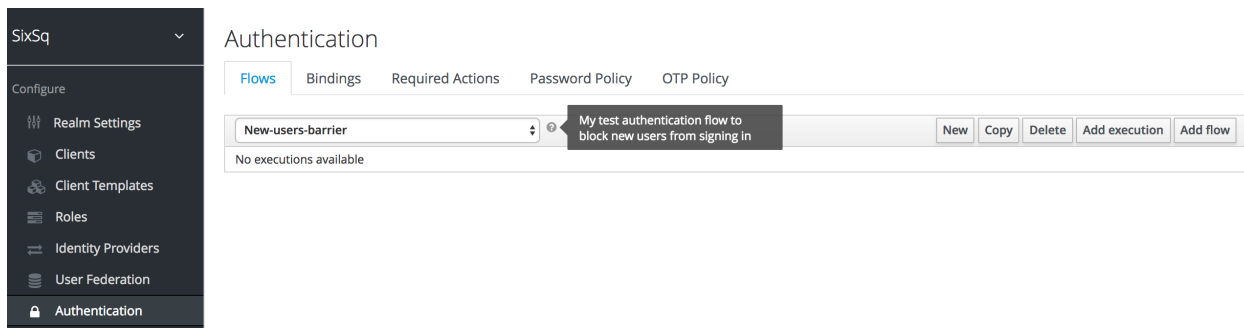
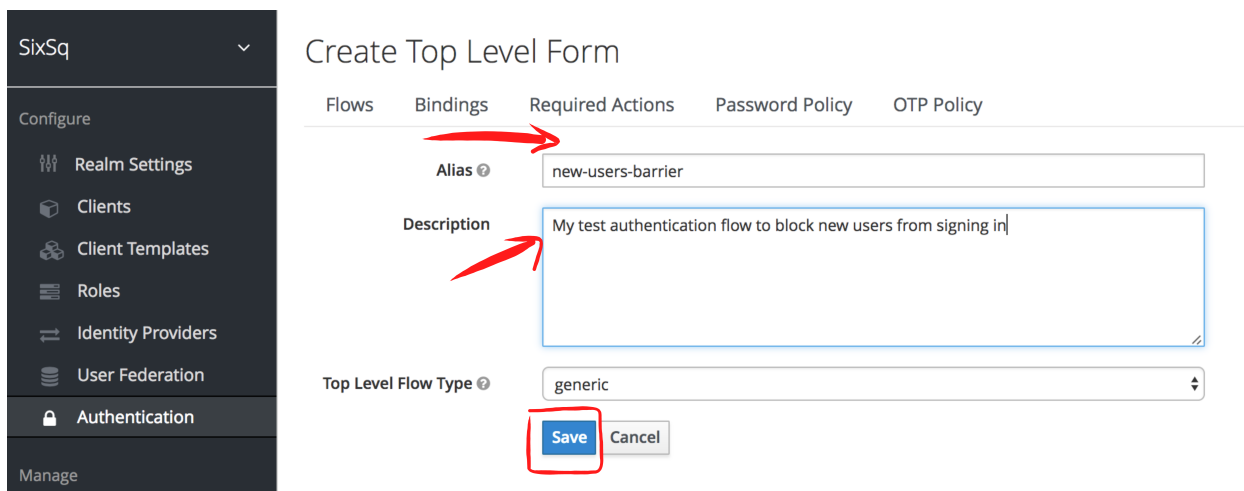
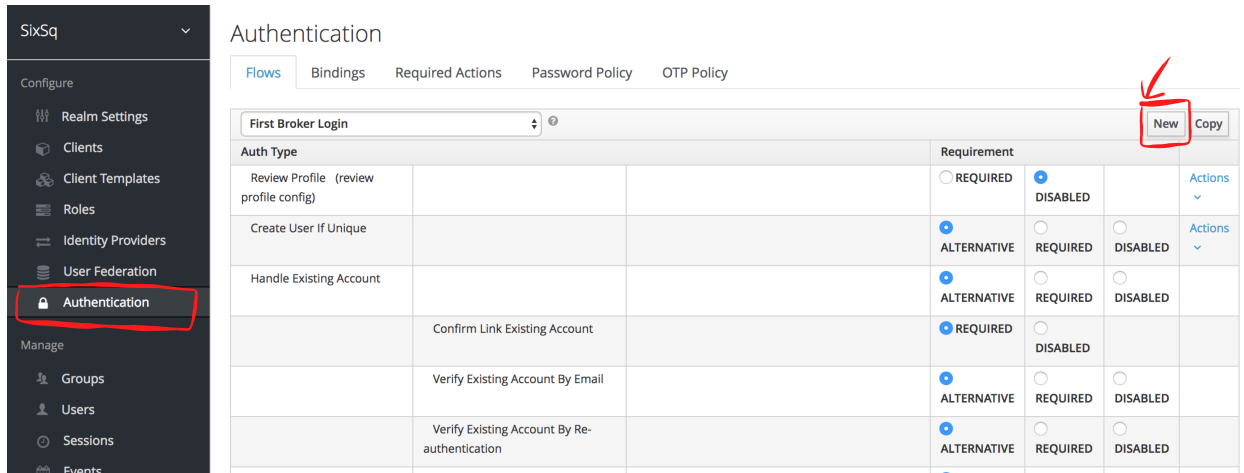
4.7 Blocking New User Logins

A useful user management feature is the ability to freeze the authentication process for one's tenant, making it impossible for new users to sign in into Nuvla.

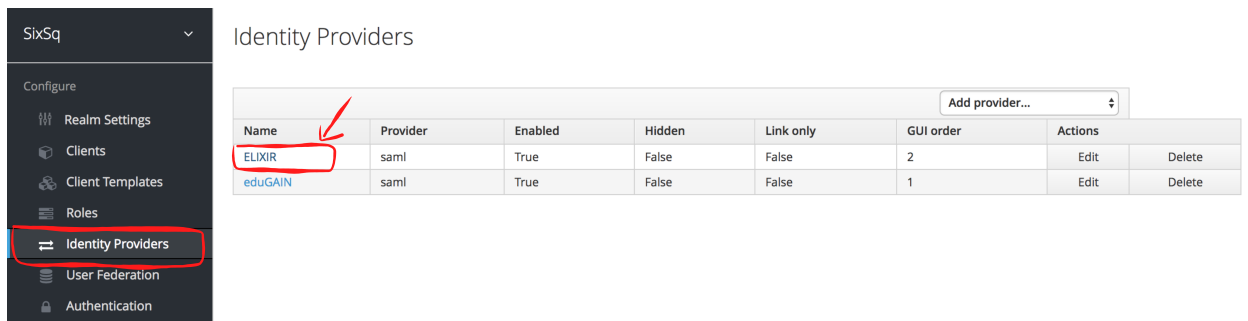
This is a different process from blacklisting users, as this will completely block any unregistered users from signing in, while letting existing users authenticate normally.

Each account manager can achieve this within his/her own tenant by following these steps:

1. on the *Authentication* panel of the Keycloak portal, create a new Authentication Flow

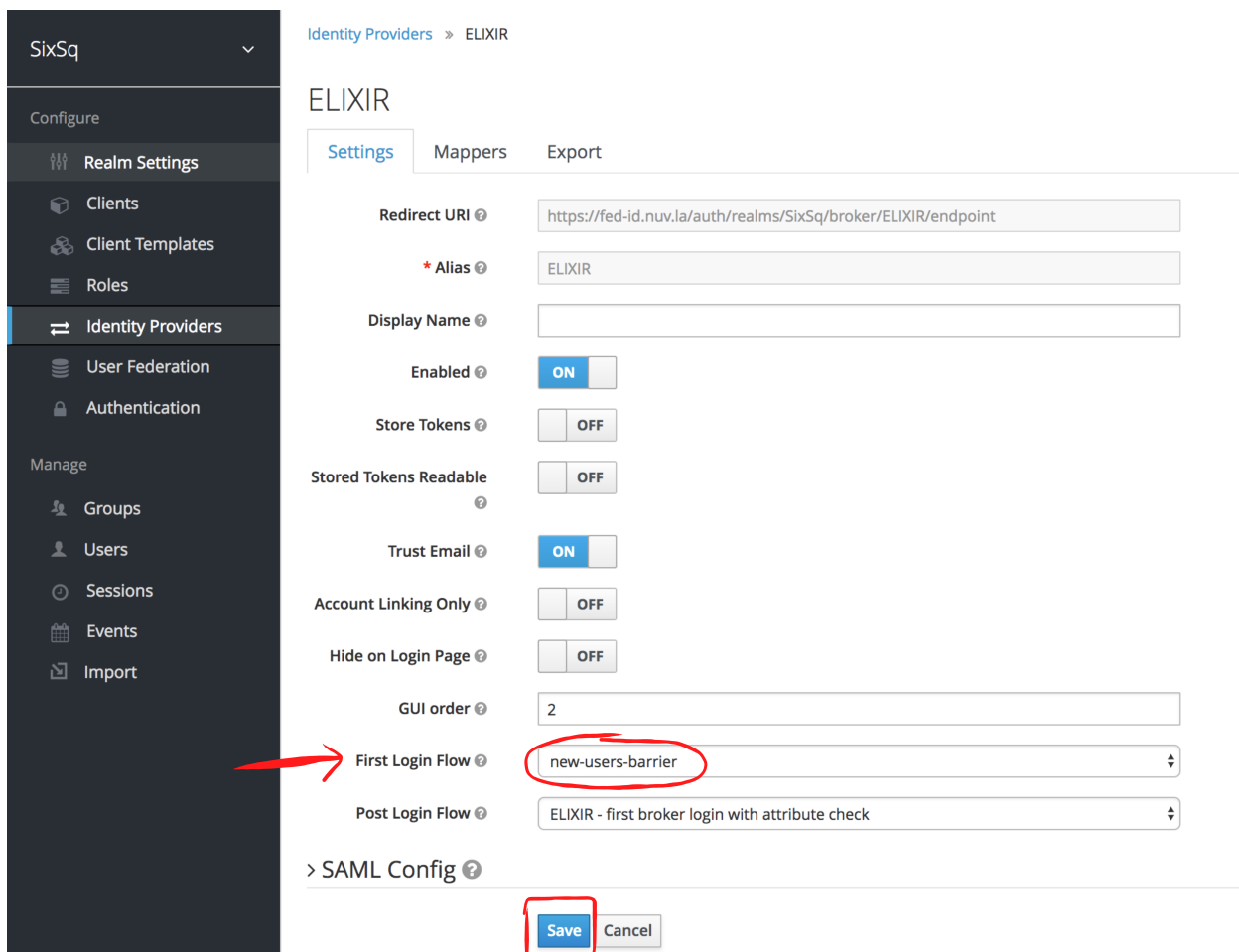


2. do not add any other executions or flows under this new flow. Leaving it empty will basically tell Keycloak not to do anything when summoned
3. since for HNSciCloud user are signing in from external Identity Providers, go to the *Identity Providers* panel in Keycloak and chose one from the list



Name	Provider	Enabled	Hidden	Link only	GUI order	Actions
ELIXIR	saml	True	False	False	2	Edit Delete
eduGAIN	saml	True	False	False	1	Edit Delete

4. for the *First Login Flow* configuration parameter, select the new Authentication Flow you've created above



Identity Providers » ELIXIR

ELIXIR

Settings Mappers Export

Redirect URI

* Alias

Display Name

Enabled ☒

Store Tokens ☐

Stored Tokens Readable ☐

Trust Email ☒

Account Linking Only ☐

Hide on Login Page ☐

GUI order

First Login Flow

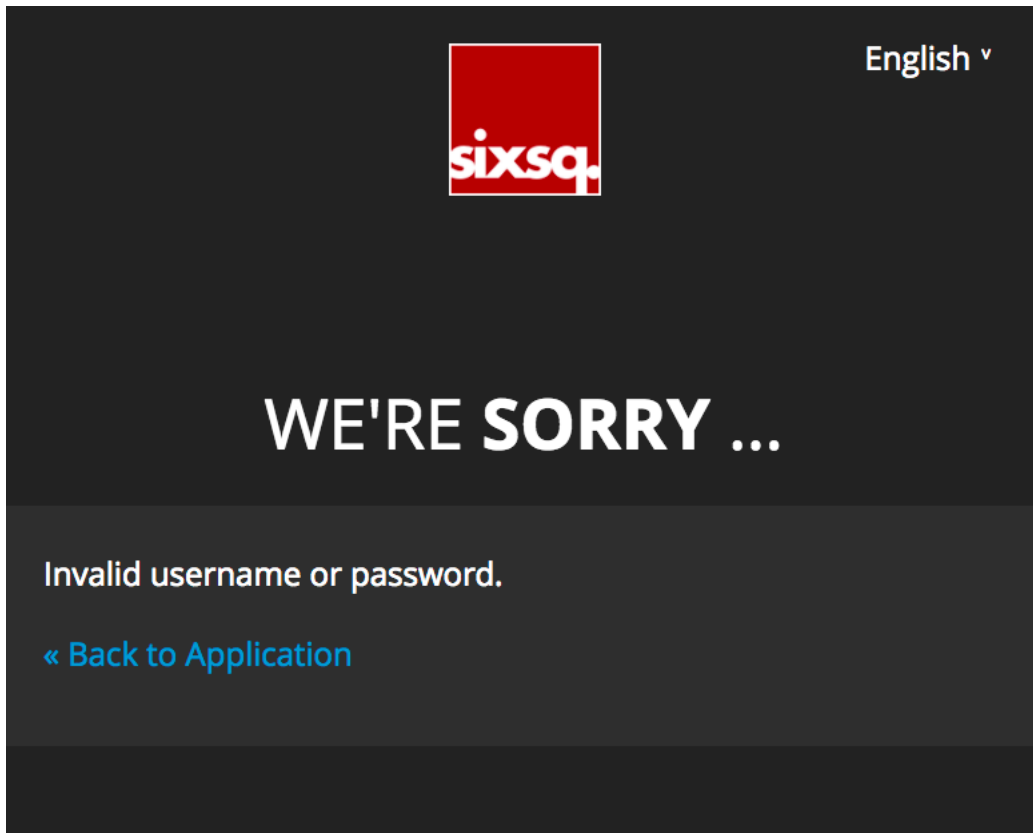
Post Login Flow

> SAML Config

5. repeat steps 3 and 4 for all the Identity Providers in the tenant

And that's it!

From this point on, new user signing in to [Nuvla](#) (or even to Keycloak directly) will get an “Invalid” login (see below) and will not be registered.



To stop this blockage, just revert the First Login Flow to what it was before, re-doing steps 3 and 4 above, but this time moving the *First Login Flow* back to **first broker login**.

4.8 Monitoring

Nuvla provides a dedicated API resource for [monitoring cloud activities](#).

Refreshed continuously (~every minute), data gets collected from all the configured clouds. The monitored data is then assembled to include for each virtual machines, the following:

- vCPU
- RAM
- Disk
- Service offer

This information is collected for both cloud resources provisioned via Nuvla and provisioned directly to the different clouds (e.g. API, portal, CLI), by-passing Nuvla. This means that Nuvla provides a unified global view over all compute resources deployed, independently of it being used as the deployment engine or not.

Further, the monitoring resource inherits the basic CIMI functionality, which means it contains specific and precise ACLs, such that only specific users, groups and organization members with the appropriate rights have access to this information.

This resource is key for other new features, such as the [Accounting](#) feature.

The REST resource providing this functionality is called *virtual-machine* and can be used, for example, to query the following information:

- Count all virtual machines
- Count all virtual machines belonging to a give user (requires privileged rights)
- Count all virtual machines launched directly with the cloud, by-passing Nuvla
- Simulate what virtual machines a given user sees (requires privileged rights)
- Group currently running virtual machines, by billing period (e.g. billed by minute vs hour)
- List all virtual machines belonging to a given deployment
- Count of running virtual machines, grouped by cloud

From the list above, *privileged rights* include administration rights (super user), as well as group and organization owners.

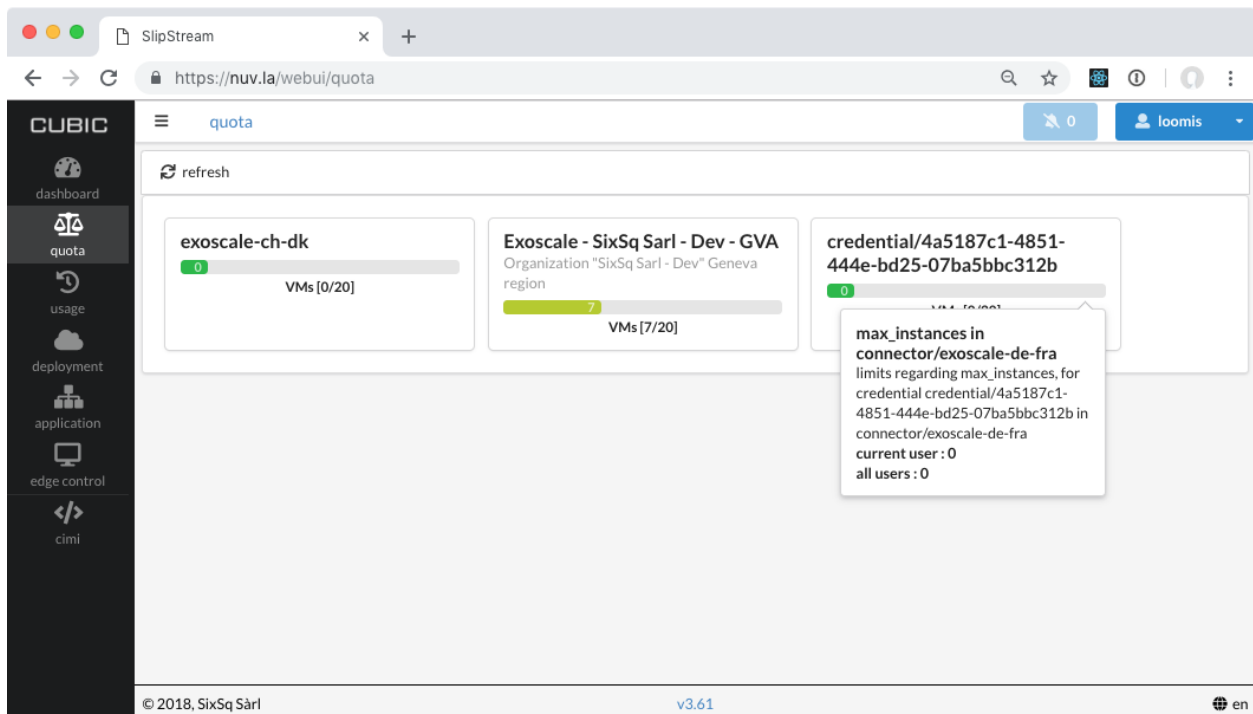
4.9 Quota

Nuvla protects users from overuse by applying a quota mechanism. The same feature gives administrators a powerful tool to control usage on the platform. The feature takes the form of a Quota resource that provides fine grain control over how many resources can be consumed, by users, groups, etc.

When deploying new resources, Nuvla checks if the requested deployment will result in the user exceeding her quota. Only if this is not the case, will the deployment be accepted. (If more than one quota applies, then they must all pass for the request to be accepted.)

Note: The current usage when evaluating a quota will include all workloads: those deployed through Nuvla and those deployed directly on the cloud. Nuvla, however, can only block deployment requests that pass through Nuvla.

The quotas and quota status can be found on the “Quota” page of the new browser interface. Hovering over a Quota indicator will provide more details.



The organization’s administrator can find more details on this feature and how to access Quota resources from the API in the [Quota API documentation](#).

Because a malformed quota can have a severe impact on a large number of users, all quota changes or additions must be handled through Nuvla [Support](#) and requested by an organization’s administrator.

Note: Quotas defined on the Exoscale cloud are periodically queried and then updated on Nuvla. As for all quotas, these can be seen in the browser interface and through the API.

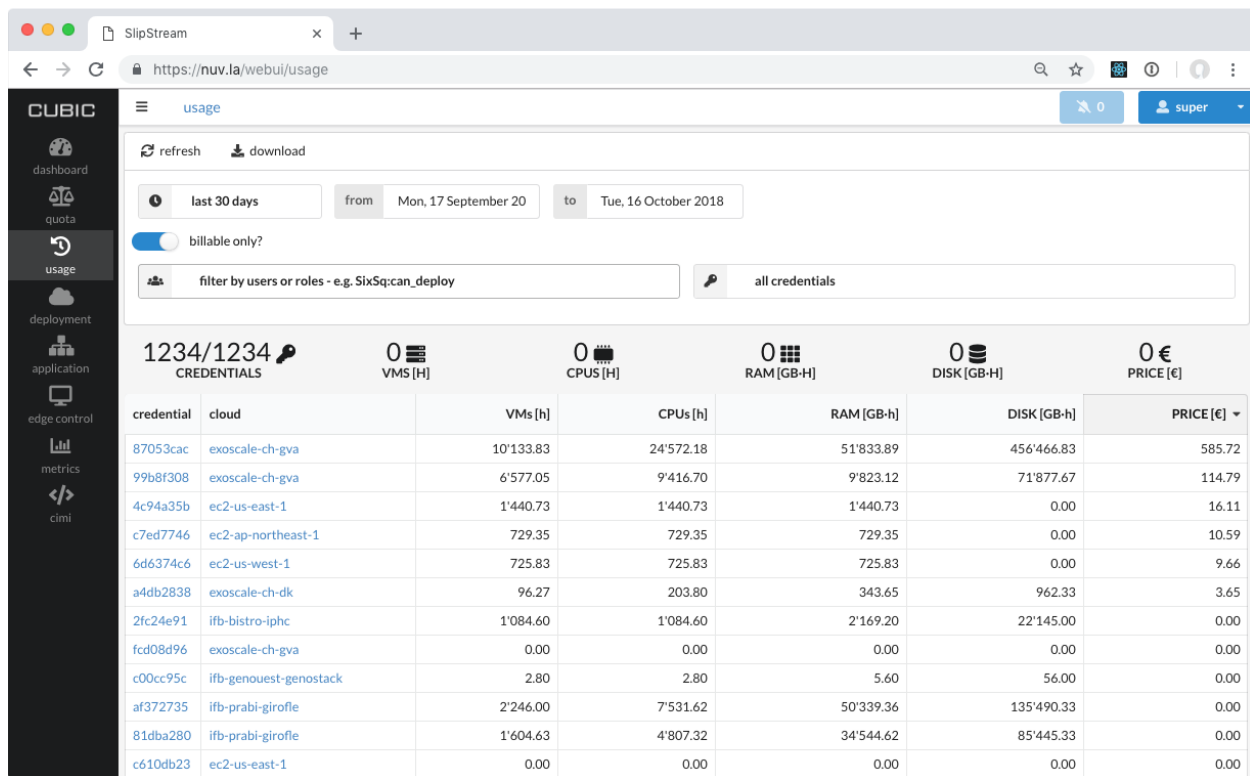
4.10 Accounting

4.10.1 Browser Dashboard

Nuvla provides a convenient dashboard to show your resource usage. If you are a tenant administrator, you will also be able to see the usage for your tenant. To access the dashboard, go to the new [Nuvla browser interface](#) and login with your federated identity.

Once logged in you can click on the “Usage” icon on the left or go directly to the URL for the [usage dashboard](#).

When you first visit the page, the usage information will be collected and shown for all of your credentials for the last 30 days. The page should look similar to the following screenshot.

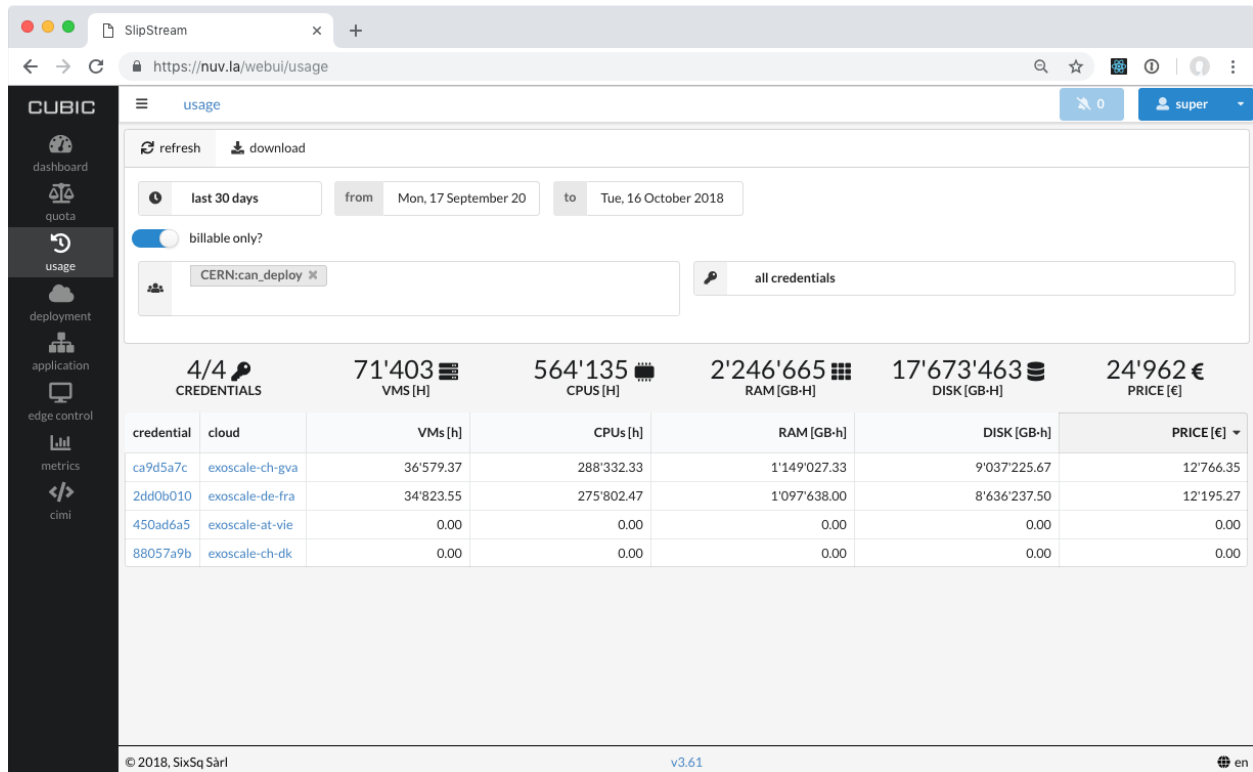


The screenshot shows the Nuvla usage dashboard in a web browser. The dashboard has a sidebar with navigation links: dashboard, quota, usage (selected), deployment, application, edge control, metrics, and cimi. The main content area shows a table of usage data for the last 30 days, from Mon, 17 September 20 to Tue, 16 October 2018. The table has columns for credential, cloud, VMs [H], CPUs [H], RAM [GB-H], DISK [GB-H], and PRICE [€]. The table lists 15 credentials and their usage data.

credential	cloud	VMs [H]	CPUs [H]	RAM [GB-H]	DISK [GB-H]	PRICE [€]
87053cac	exoscale-ch-gva	10'133.83	24'572.18	51'833.89	456'466.83	585.72
99b8f308	exoscale-ch-gva	6'577.05	9'416.70	9'823.12	71'877.67	114.79
4c94a35b	ec2-us-east-1	1'440.73	1'440.73	1'440.73	0.00	16.11
c7ed7746	ec2-ap-northeast-1	729.35	729.35	729.35	0.00	10.59
6d6374c6	ec2-us-west-1	725.83	725.83	725.83	0.00	9.66
a4db2838	exoscale-ch-dk	96.27	203.80	343.65	962.33	3.65
2fc24e91	ifb-bistro-iphc	1'084.60	1'084.60	2'169.20	22'145.00	0.00
fc08d96	exoscale-ch-gva	0.00	0.00	0.00	0.00	0.00
c00cc95c	ifb-genouest-genostack	2.80	2.80	5.60	56.00	0.00
af372735	ifb-prabi-giroffe	2'246.00	7'531.62	50'339.36	135'490.33	0.00
81dba280	ifb-prabi-giroffe	1'604.63	4'807.32	34'544.62	85'445.33	0.00
c610db23	ec2-us-east-1	0.00	0.00	0.00	0.00	0.00

Your dashboard will only include your credentials, or your tenant’s credentials, if you are a tenant manager.

You can use the filter to select the usage for certain users or roles. For the HNSciCloud tenants, the role takes the form “tenant:can_deploy”, for example, the role for the SixSq tenant is “SixSq:can_deploy”. After filtering and clicking the “refresh” button, the dashboard should look like this screenshot.



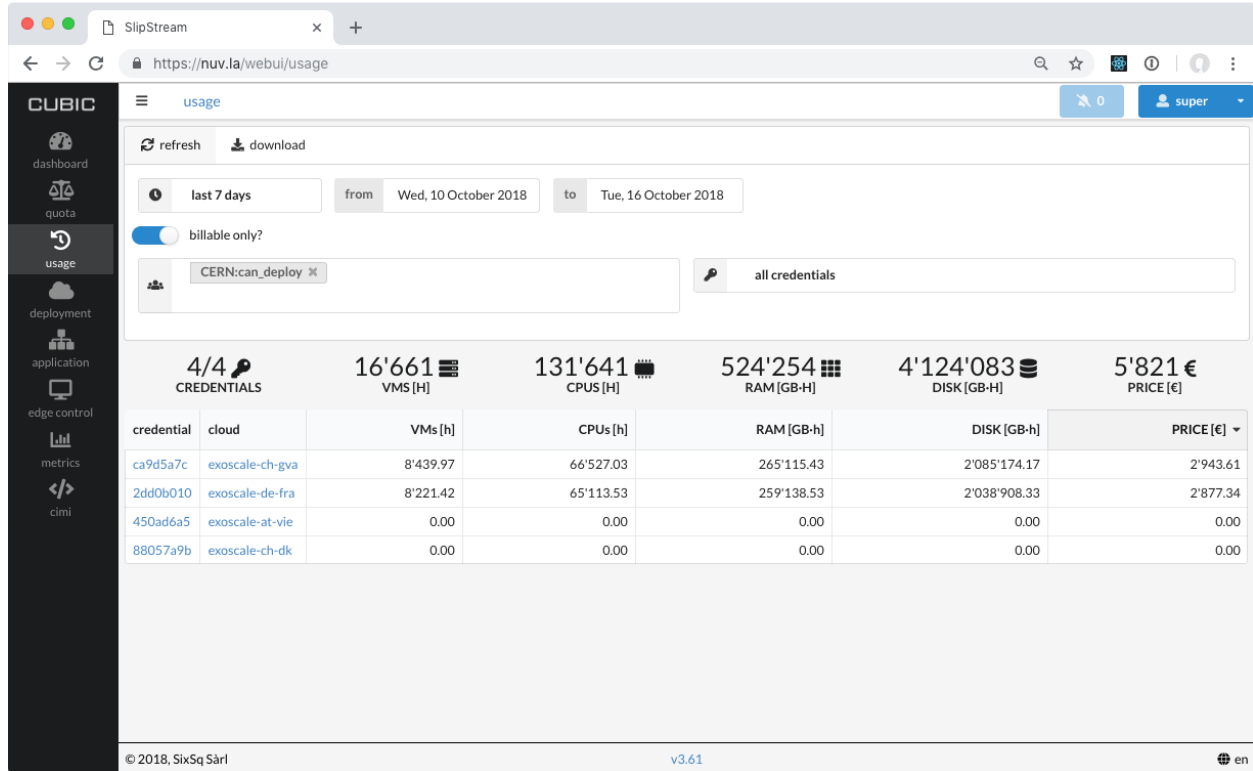
You may not see a difference in the filtering if you only have access to the accounting information for a single group.

You can also filter by specific users. However, this will only work for virtual machines that have been deployed through Nuvla. Virtual machines deployed directly on the cloud can only be attributed to a tenant, not individual users.

You can also change the time period for the dashboard. You can select one of the predefined time periods:

- Today
- Yesterday
- Last 7 days
- Last 30 days

or set specific “from/to” dates via the date picker widget. Again, click the “refresh” button to update the displayed information. This screenshot shows the usage information for the CERN tenant over the last 7 days.



Note that the “from” date always starts just after midnight and the “to” date goes to just before midnight.

The “billable only?” toggle limits the results to virtual machine states that incur charges. You can turn this off to include information for virtual machines in, for example, the “stopped” state. Note that the pricing information in this case, will be inaccurate.

The table is interactive and you can click on the column headers to reorder the results by the values in the column.

4.10.2 REST API

The usage dashboard is driven entirely by the Nuvla API. You can also use this API directly to obtain custom views of the usage information.

Nuvla provides a new [dedicated API](#) resource for monitoring and accounting information. The CIMI resources of interest are:

- **VirtualMachine:** Provides information on all active virtual machine. Together, these resources show the global current state of the hybrid cloud infrastructure.
- **Metering:** These are snapshots of the “virtual-machine” resources that are take every minute. These provide historical usage information and are the basis of the usage dashboard.
- **StorageBucket:** These provide information on the usage of S3 resources on the cloud. This information is not include in the browser interface, but can be obtained from the API.

The new feature regularly snapshots the global state of deployed resources provided by the [Monitoring](#) resource, to build a historical view of usage. In the process, it also pulls other valuable information such as pricing from the corresponding service offers.

This functionality, coupled with the extensive CIMI query and filtering capabilities, provides together a powerful, yet simple, way to extract accounting information from Nuvla.

The [dedicated API](#) documentation provides several examples of queries in order to extract useful information, such as pricing or vCPUs, for a given group and time period. This feature can also be used to plot trends, trigger alerts and much more.

This section contains howtos for common tasks that will be carried out by data coordinators.

5.1 Deploying Data Management Services

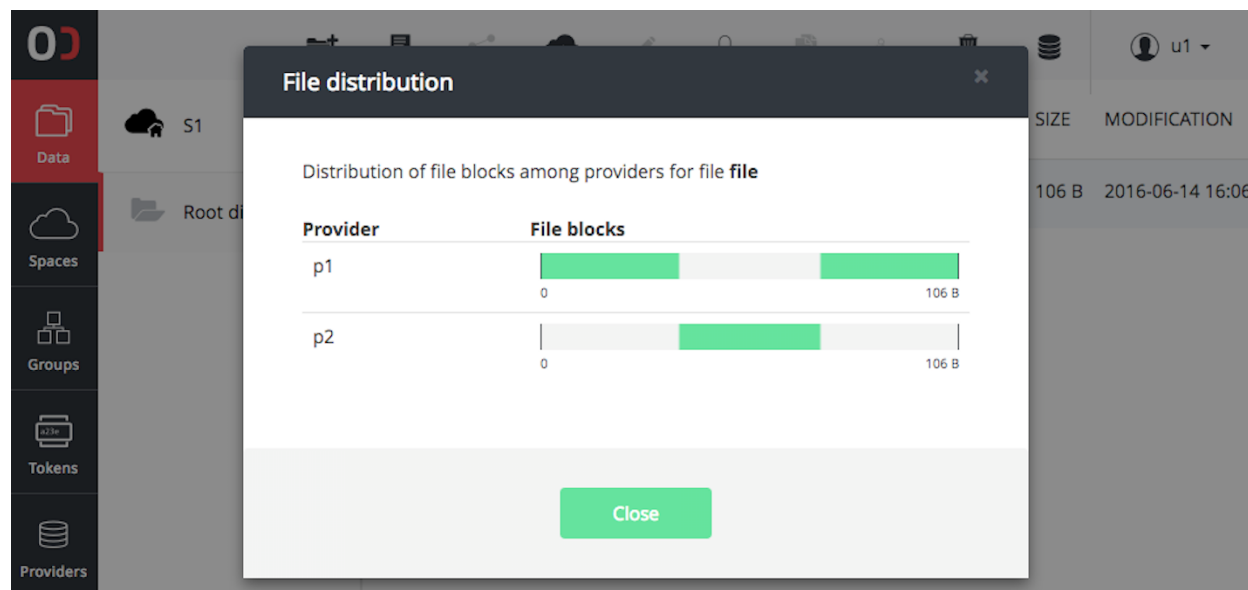
Data management services are based on Onedata technology. For general overview of Onedata and its core concepts including zones, providers and spaces please refer to the [official documentation](#).

Information on deploying these services is available from CYFRONET. If you need to do this, you can contact them through the Consortium's HelpDesk (support@sixsq.com) or directly.

5.2 Replicating Data

Onedata allows for full or partial replication of datasets between storage resources managed by Oneprovider instances. The replication can be performed at the level of entire spaces, directories, files or specific file blocks.

Onedata web interface provides visual information on the current replication of each file among the storage providers supporting the user space in which this file is located. Sample replication visualization is presented in the image below:



5.2.1 REST interface

For full control over transfer and replication users can directly invoke REST API of Oneprovider service. The documentation for this API can be found in the [official documentation](#).

5.3 Import and Export Data

5.3.1 POSIX

First of all, POSIX protocol can be used to import or export data to/from Onedata virtual filesystem using standard tools, such as 'cp' or 'rsync'. It is necessary to run the Oneclient command line tool on an access machine where the target data set is available (ingress) or where it should be exported to (egress). In case the storage managed by Onedata is available directly from the machine running Oneclient, this situation is detected automatically and the transfer between Onedata managed storage and external storage is performed directly and transparently without going via Oneprovider service, which is only called for metadata operations.

5.3.2 CDMI and REST

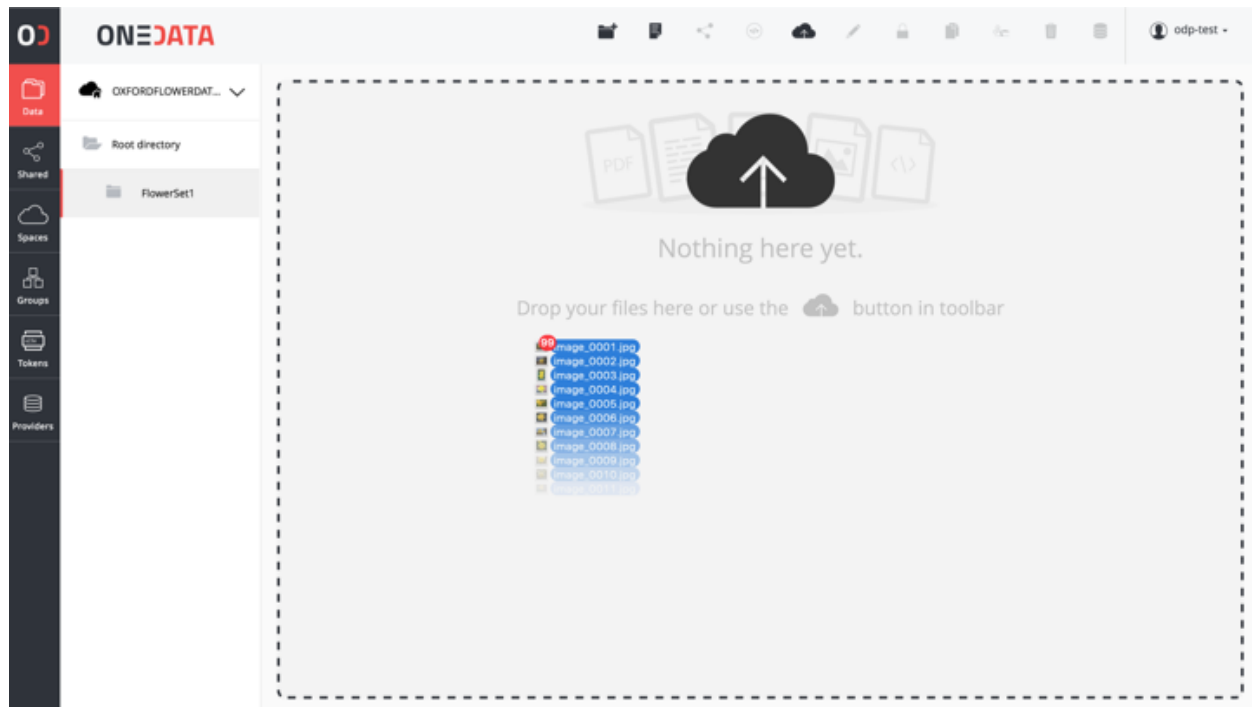
Furthermore, Onedata implements full CDMI v1.1.1 protocol specification, including data download/upload requests, and thus provides object storage system interface for all data in parallel to the POSIX protocol. This enables integration with custom user services such as portals.

For batch data transfer management, Onedata provides REST API giving programmatic access to data replication and transfer control and monitoring between the data sites.

5.3.3 GUI

Finally for small files or data sets, they can be uploaded and downloaded directly using the Web Graphical User Interface, available on all major browser and mobile devices. In order to import data using web browser, simply open

a directory within a space where the files should be uploaded, and move the files from your desktop or file browser to the Onedata Internet browser window. The upload progress will be displayed until all files are successfully uploaded.



5.3.4 Legacy data import

In use cases where there is a need to provision large legacy datasets, it is possible to configure Oneprovider service to expose such data set directly from the legacy storage without any data migration to another storage. Oneprovider service will run periodically synchronization of files on such storage, and will detect automatically new or updated files and will update its metadata database automatically. This option can be selected when adding new storage to the Oneprovider and has to be performed by Oneprovider administrators:

The screenshot shows the ONE DATA PROVIDER PANEL interface. On the left, the 'CLUSTERS' sidebar is active, showing a search bar and a list of items: ONEPROVIDER-DEMO, Nodes, Provider, Storages, and Spaces. The main panel is titled 'SPACES' and contains a 'Cancel supporting space' button. Below this is the 'ADD SUPPORT FOR A SPACE' form.

ADD SUPPORT FOR A SPACE

Storage: ExperimentOutput

Support token: KAJHSDLKJAHSDJKLHASJKLHkashd

Size: 100

Mount in root: ☒

Import storage data: ☒

IMPORT CONFIGURATION

Import strategy: Simple scan

Max depth (optional): Example: 3

UPDATE CONFIGURATION

Update strategy: Simple scan

Max depth (optional): Example: 3

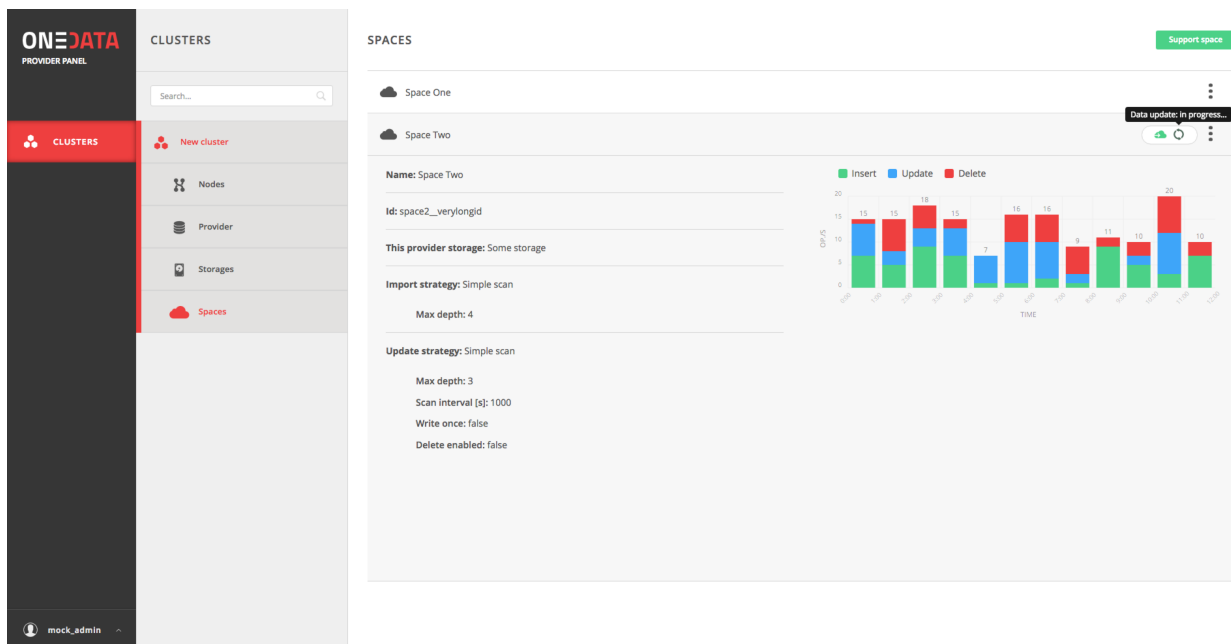
Scan interval [s]: 10

Write once: ☐

Delete enabled: ☐

Support space

Once the storage is configured for the legacy data import it will be continuously monitored for changes in the data collection (new files, modified files, deleted files) and basic statistics on the scan process will be displayed.

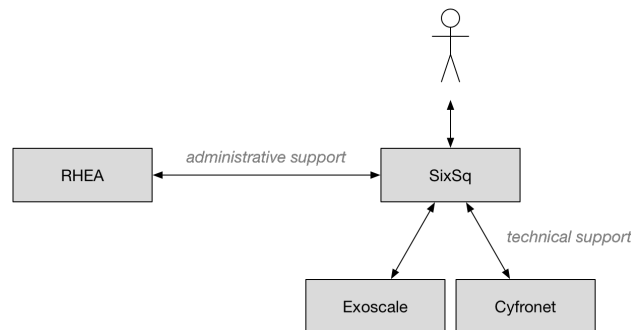


For more details, check the [official documentation](#).

Onedatify

Provisioning of legacy data using Onedata can be achieved in an easy way using a script called Onedatify, which automates the process of installing and configuring a Oneprovider instances, setting up the specified storage resources and automatically synchronizing the storage contents which are made immediately available via Onedata. Usage instructions are available [here](#).

All members of the consortium are committed to providing **timely, high-quality support** to users of the system. SixSq will act as a gateway, providing first level support itself and coordinating interactions with the other consortium members. The diagram provides an overview of the support infrastructure.



SixSq manages its support, including the support for HNSciCloud, through Freshdesk. The support services are offered in English, but may also be available in the local languages of the partners (e.g. French).

Users can submit support tickets via:

- Email (support@sixsq.com),
- [Freshdesk](#), and
- Phone (+41 22 544 17 33).

A knowledge base is integrated into the Freshdesk system. A [dedicated section of the knowledge base](#) is maintained for HNSciCloud.

The consortium provides 8x5 support with a 1-hour response time. Users can set the incident priority level when submitting their tickets to ensure that they are dealt with in the appropriate time frame. Escalation of urgent or unsolved issues is possible.

7.1 Exoscale

- [Exoscale Operational Status](#)
- [Exoscale Portal](#)
- [General Exoscale Information](#)
- [Exoscale User Documentation](#)
- [Exoscale API Documentation](#)

7.2 Onedata

- [Onezone Browser Interface](#) (obtain the URL from the data manager for your organization)
- [General Onedata Information](#)
- [Onedata User Documentation](#)
- [Onedata API Documentation](#)

7.3 Keycloak

- [Keycloak Portal](#)
- [General Keycloak Information](#)
- [Keycloak Documentation](#)

7.4 Nuvla (SlipStream)

- [Nuvla Operational Status](#)
- [Nuvla Browser Interface](#)
- [General Nuvla Information](#)
- [SlipStream User Documentation](#)
- [SlipStream API Documentation](#)
- [SlipStream Python API](#)
- [SlipStream Clojure API](#)
- [SlipStream Libcloud Driver Documentation](#)

The materials for previous demonstration and training events can be found in [older versions of the documentation](#).

8.1 GridKa Training

GridKa School, 28 August 2018, Karlsruhe, Germany

8.1.1 Introduction

Platform Overview

An overview of the platform's services and architecture can be found in the [Functional Overview](#) section.

Expected Results

In this training, you will learn how to deploy cloud applications with Nuvla, a cloud application management platform, and how to deploy virtual machines with the Exoscale cloud infrastructure.

After this training, you will have:

- Created an account on Exoscale,
- Created an account on Nuvla,
- Deployed virtual machines through the Exoscale web portal,
- Deployed simple and complex cloud applications with Nuvla's browser interface, and
- Viewed the quantity of resources you have used to complete the training.

Further details can be found in documentation for the [Platform Services](#).

8.1.2 Prerequisites

To follow the complete end-user tutorial, you must have the following:

- A laptop with access to the Internet, most likely via WiFi
- An SSH client installed on your laptop
 - Linux/Mac OS: default SSH client
 - Windows: use [Putty](#)
- An **OpenSSH** keypair
 - [Linux/Mac OS](#)
 - [Windows](#)
- An identity from an eduGAIN or Elixir AAI Identity Provider.

Note: As you are all from KIT, you should be able to log into the platform using the credentials from the KIT Shibboleth Identity Provider.

If you don't meet any of the prerequisites, consult the referenced documentation, ask your administrator, or chat with the tutor.

8.1.3 Nuvla & Cloud Accounts

To follow this training, you must have accounts on Nuvla and Exoscale. You will create both accounts and configure them so that they can work together.

Exoscale Account

Register with Exoscale to create an account there. You will have been given a voucher code that will allow you to create an account with 30€ of credit. This is more than enough to follow the training.

The account you create belongs to you. You are welcome to continue using the account (and any remaining credit) after the training. When the credit is gone, you can add more credit with a credit card.

Follow the [Voucher Redemption](#) instructions for creating an account with the voucher code you received.

Warning: On the last step, choose “**for personal projects**”, not “for team projects”.

The process requires email validation, so you will need access to your mail client. After the email validation, you should be able to log into the [Exoscale portal](#) with the email address and password you provided.

When you are logged into the Exoscale portal, do the [SSH Configuration](#) to allow you to access your virtual machines via SSH.

Nuvla Account

You will register with Nuvla using your federated identity from KIT. Follow the [Registration](#) instructions to create your account.

Note: When you log in for the first time, will be redirected to the App Store page of Nuvla and offered a tutorial. You can click away the tutorial dialog, as we'll not be using that here.

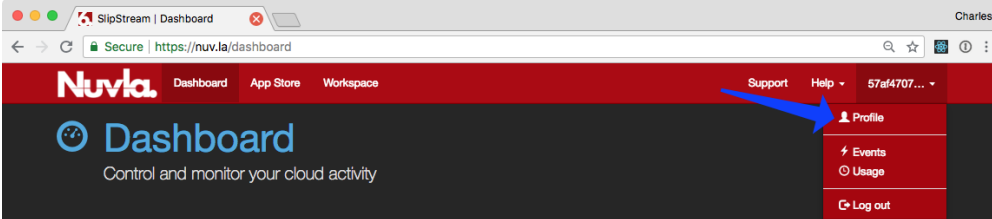
General Fields

There are three general user profile fields that you may want to change. Open your user profile, click on the “Edit” action, and then open the “General” section by clicking on the header.

- Change the default cloud to “exoscale-ch-gva” (or another cloud for which you will have credentials).
- Verify the “Keep running” parameter is set to “On Success”.
- Copy your SSH public key into the corresponding textbox.

Afterwards, be sure to click on the `Save` action!

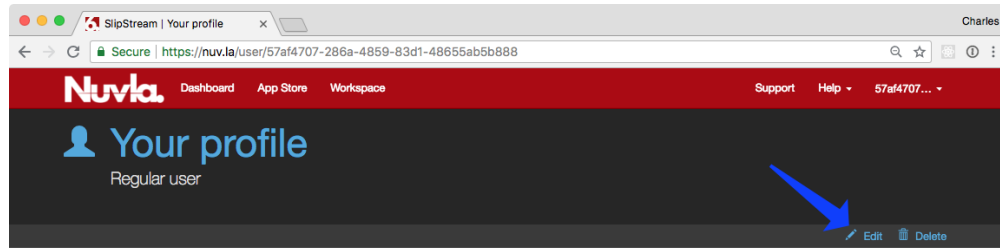
Warning: Be sure to copy the full contents of your public SSH key as a **single line of text!**



The screenshot shows the Nuvla Dashboard interface. The top navigation bar includes links for Dashboard, App Store, and Workspace. A user profile dropdown menu is open, showing options for Profile, Events, Usage, and Log out. A blue arrow points to the Profile option.

Below the dashboard header, the Usage section is visible, showing a gauge for All Clouds with a value of 0 VMs out of 1732. There is a button to Add a new cloud.

The Deployments section is also visible, with a tab for Virtual Machines. It includes a filter for Include inactive runs and a table with columns: ID, Application / Component, Service URL, State, VMs, Start Time [UTC], Clouds, Tags, and User. The table currently shows 0 found items.

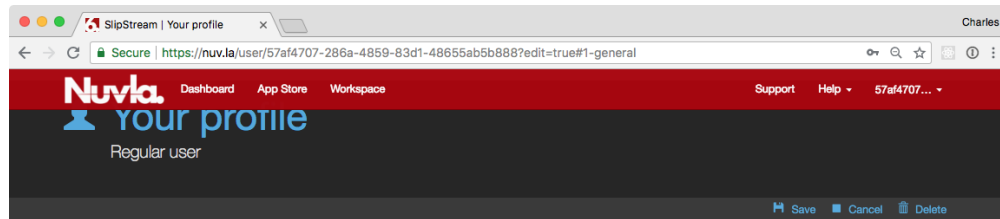


No clouds configured!

Please note that you will not be able to deploy anything until you configure the credentials for at least one cloud in the sections below. For example, you might want to **configure the cloud**, which is currently set up as default in your profile. For more details about how to solve this please refer to the [documentation](#).

Summary

Description	Value
Username	57af4707-286a-4859-83d1-48655ab5b888
First name	Charles
Last name	Loomis
Organization	SixSq
Email	cal@sixsq.com
Is administrator?	<input type="checkbox"/>
Roles	session/5c7f03de-4f20-417b-af6f-ff9e708eb9a2 USER ANON SixSq/sixsq SixSq:http://partner-switchaai.ch/sixsq.com
Date of creation	Wednesday, 22 August 2018, 08:17:46 UTC
State	ACTIVE



Summary

General

Description	Value
Default Cloud	exoscale-ch-gva
Keep running after deployment	On Success
Level of verbosity	0 - Actions
Execution timeout (in minutes)	30
SSH Public Key(s) (one per line)	ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAQEAwSw4WTABnlabGt62u1fXaw7zhMIOX/bm 1++m/cLPxgTIRJLor/K2QW10i0IE5/yOU/OQV8KGBKu+xemfkkdIMzXNQFaD+Gs T1GrbUayl5v0DiBVS24Hd8j8TmdzpldxWEYvAPyjurUsNGwn72XMTMhNIC2CvVq 205I2VjdOT575Wjymas/0/xvpuzYYFGVjCd5dJjNlwgpl/8a68Jld/MRaxGbkNkYGRq vU9SXQNaOIFYLqoak9PynEvBTA+MSn1HPvHY0EGqYGGEmIFTCM2chbKP8o1 jwnXgcAzW2Olvk9720kOOp7XVOldGIMqT6IX1UBJBq2IE/Fuw== loomis
Cloud usage email	never

Cloud Credentials

You must provide your cloud credentials to Nuvla, so that it can act on your behalf when provisioning cloud resources.

To learn where to find your Exoscale API key and secret and how to configure your user profile see the [main SlipStream documentation](#).

For this tutorial, add credentials for the **exoscale-ch-gva** and **exoscale-de-fra** regions. The API key and secret will be the same for all Exoscale regions.

8.1.4 Deploying on Exoscale

Run through the entire lifecycle for a virtual machine on Exoscale. To understand how to manage virtual machines on Exoscale, follow the instructions for the entire *Virtual Machine Lifecycle* in the core part of the documentation.

8.1.5 Deploying on Nuvla

Infrastructure-as-a-Service (IaaS) cloud infrastructures generally provide you only images with core operating system software. Nuvla allows you to manage both simple and complicated cloud applications and to automate the entire application lifecycle.

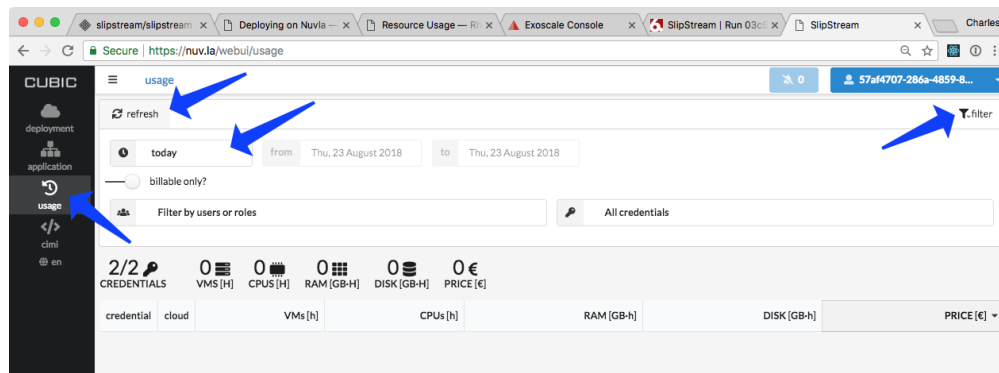
To understand how to manage full cloud applications with Nuvla, refer to the instructions for the entire *Application Lifecycle* to deploy a cloud image, cloud component, and a cloud application.

8.1.6 Resource Usage

Nuvla has the ability to track your current usage of resource and to maintain a history of this. To see what resources you've consumed during the tutorial, go to the address <https://nuv.la/webui>. This is a prototype of a new browser interface for Nuvla.

To see your consumption,

- Click on the “usage” icon on the left.
- Open the filter via the button in the upper right corner.
- Change the time period to “today”.
- Click on “refresh” to see your usage.



The result should look something like the following screenshot. Remember to shutdown applications and virtual machines that you're not using.

credential	cloud	VMs [h]	CPUs [h]	RAM [GB-h]	DISK [GB-h]	PRICE [€]
TOTAL	SELECTED	0.50	1.00	1.67	18.33	0.02
52cc6e9d	exoscale-ch-gva	0.50	1.00	1.67	18.33	0.02
d2b69a46	exoscale-de-fra	0.00	0.00	0.00	0.00	0.00

8.1.7 Going Forward

In this training, you will learn how to deploy cloud applications with Nuvla, a cloud application management platform, and how to deploy virtual machines with the Exoscale cloud infrastructure.

If everything's gone well, you will have completed the following tasks during the training:

- Created an account on Exoscale,
- Created an account on Nuvla,
- Deployed virtual machines through the Exoscale web portal,
- Deployed simple and complex cloud applications with Nuvla's browser interface, and
- Viewed the quantity of resources you have used to complete the training.

The accounts you have created are your own personal accounts, so you are welcome to continue using them.

Going forward, you may want to look through the documentation for the *Platform Services*. That documentation covers all the service details. If you can't find the information you need, feel free to contact support@sixsq.com.

Consortium

The consortium consists of four organizations that have experience working together to deliver computing solutions. Rhea leads the consortium consisting of software providers (SixSq and CYFRONET) and a cloud service provider (Exoscale).



Experienced leader of large frame contracts in the Space and Defence sectors, including software centric systems and software products. Majority shareholder of SixSq, RHEA has developed a solid relationship with the cloud specialist and supports spin-in of its technologies into the space and defence sectors. RHEA acts as the prime for the project and performs much of the system testing. Rhea is based in Belgium. [\[more info\]](#)



Responsible for technical coordination, SixSq brings its knowledge of cloud technologies and its innovations from the Nuvla cloud application management platform. SixSq is based in Geneva, Switzerland. [\[more info\]](#)



Provides the Onedata data management solution and support for it. Onedata is seamlessly integrated into the Nuvla/SlipStream solution, allowing for easy deployment of the platform. Academic Computer Centre CYFRONET AGH is based in Krakow, Poland. [\[more info\]](#)

Exoscale is the cloud computing platform for cloud native teams. Relying only on pure “as-a-service” components, Exoscale is built by DevOps for DevOps. Originally based in Switzerland, it provides cloud resources in Switzerland, Austria, and Germany to the Consortium’s hybrid cloud platform. [\[more info\]](#)