

CSC- Bioweeek 2018

Using cPouta for cloud computing

8.2.2018

Kimmo Mattila, Shubham Kapoor, Ari-Matti Saren
(Jukka Nousiainen)



*CSC - Finnish research, education and public administration
ICT knowledge centre*

Course Introduction

WHAT EXACTLY IS IT?

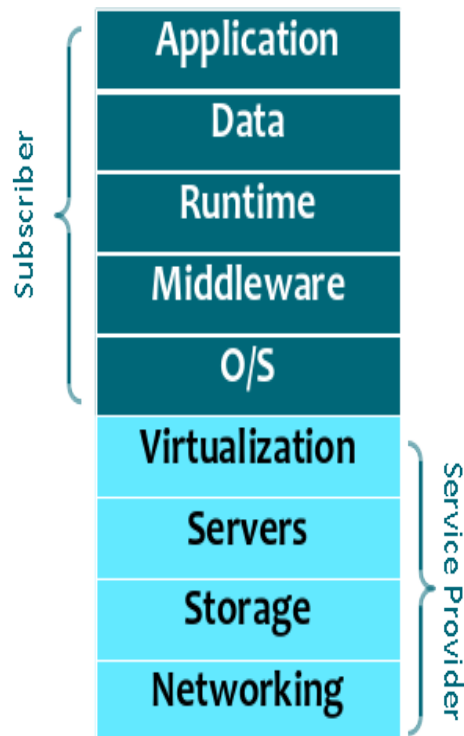
"This course will give a practical introduction to using CSC's cloud services"

- cPouta Infrastructure as a Service (IaaS) Cloud
- Allows running Virtual Machines (VMs) on CSC's Data Center infrastructure
- Grants users full control of applications and operating system
- On the flipside, users must manage and secure their VMs

Cloud computing? Myriads-as-a-Service

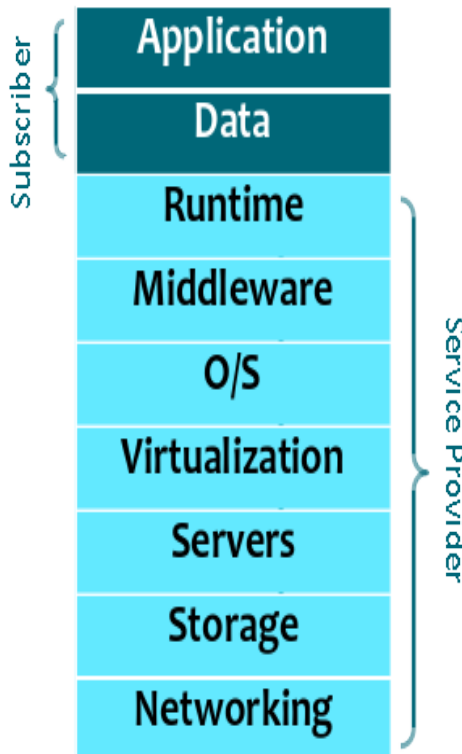


Infrastructure as a Service (IaaS)



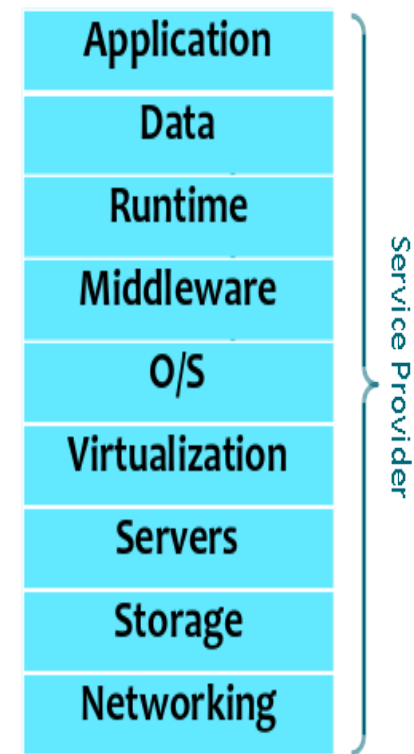
For example:
pouta.csc.fi, Amazon EC2, Microsoft Azure, Google Compute Engine

Platform as a Service (PaaS)



For example:
notebooks.csc.fi, Google AppEngine, Heroku, Amazon Elastic Beanstalk

Software as a Service (SaaS)



For example:
notebooks.csc.fi, **chipster.csc.fi**, Google Web Apps, Dropbox, Spotify

Taito.csc.fi cluster vs. cPouta



	Taito cluster	cPouta IaaS cloud
Operating system	Same for all: CSC's cluster OS	Chosen by the user
Software installation	Done by cluster administrators, customers can only install software to their own directories, no administrative rights	Installed by the user, the user has admin rights
User accounts	Managed by CSC's user administrator	Managed by the user
Security e.g. software patches	CSC administrators manage the common software and the OS	User has more responsibility: e.g. patching of running machines
Running jobs	Hundreds of computing cores available, but you may have to queue before your job starts.	The full capacity of the VM (1 – 48 cores are immediately available)
Environment changes	Changes to software happen.	The user can decide on versions.
Snapshot of the environment	Not possible	Can save as a Virtual Machine image

When to use cPouta in stead of Taito cluster (SaaS ?)

- In most cases the Taito cluster is both easier and more efficient platform for bioscience data analysis.
- However in some cases using your own virtual machine in cPouta can be a better or only solution for doing your analysis:
 - Analysis requires a longer uninterrupted running time that what is possible in Taito. (In Taito the maximum run time is 14 days).
 - The tool needs services that are blocked from normal users in Taito. E.g. users own MySQL database or a web server.
 - The tool is not compatible with Taito environment or installing it requires root level access.
 - External servers or users need direct access to your data or services.

VM flavors available in cPouta

Virtual machine flavors:

- **Standard flavors:** 1-6 Cores, memory: 1-16 GB
- **HPC flavors:** 1-48 cores, memory: 3,8 –240 GB
- **I/O flavors:** 2-16 cores, memory: 10-80 GB,
SSD Disk: 70-700 GB
- **GPU flavors:** 4-56 cores, 1-4 GPUs. Memory: 120-480 GB

Creating virtual resources in cPouta - User Interfaces



- Web User Interface - <https://pouta.csc.fi>
 - Suitable for administering individual VMs, keys, images, volumes...
 - The only UI to support Haka federated login

Today's
MO



- CLI tools
 - Suitable for more elaborate resource provisioning and possibly some lightweight (scripted) software integrations

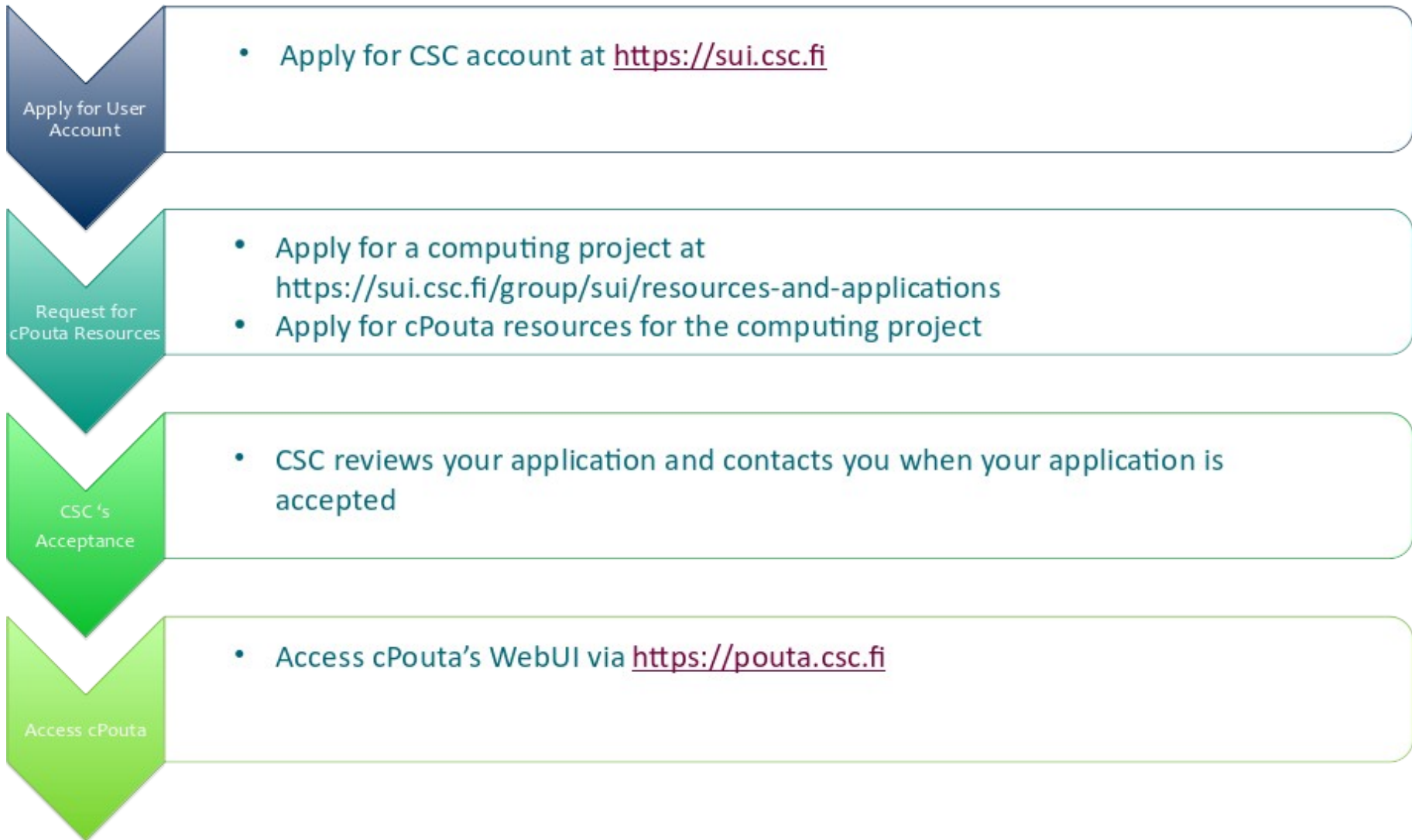


- Programming APIs
 - Suitable for building very large systems and stacks
 - Support from individual services (compute, storage) to full-fledged orchestration

Simple procedure for launching a Virtual machine in cPouta

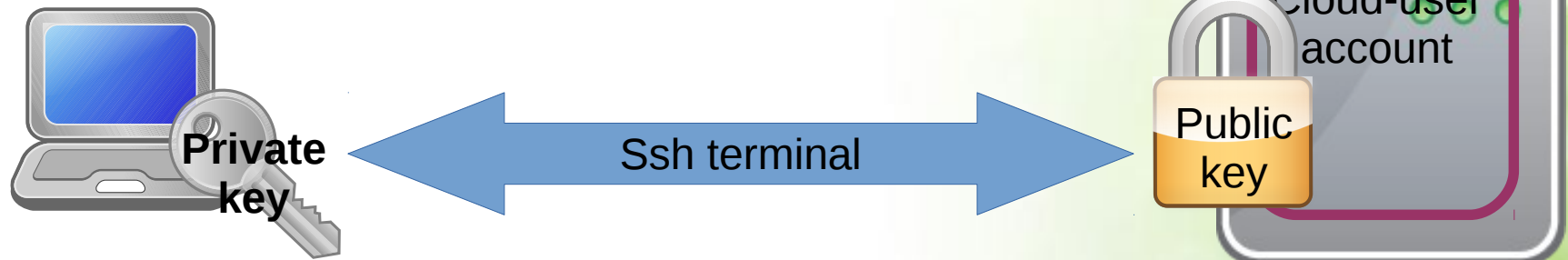
- 0. Get access to cPouta service**
- 1. Generate an ssh key pair and import it to cPouta interface**
- 2. Define and launch a Virtual Machine**
- 3. Associate an IP address to your server**
- 4. Create and apply security groups to enable remote access**
- 5. Create and attach a volume to store your data**
- 6. Install software you need**

0. Getting access to cPouta Services - MinEdu funded track



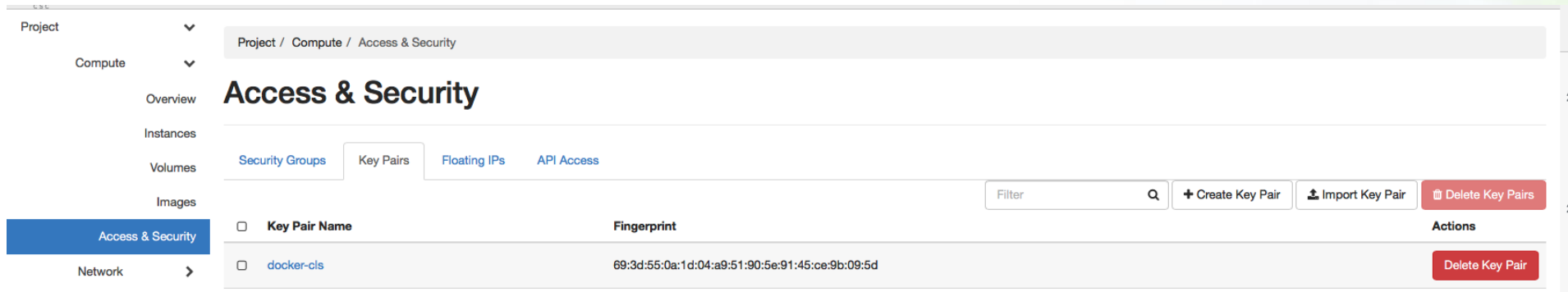
1. ssh keys

- The default and recommended way to authenticate your Virtual Machine is using ssh key pairs
- A key pair consists of two files: encrypted private key is installed in your local computer (“key”) and public key (“lock”) is inserted to a new virtual machine in the launching process
- Key pairs can be generated with ssh command or in cPouta WWW interface
- Private key should be protected with password
- Same key pair can be used for several VMs



Creating a key pair

Navigate to
Compute>Access and Security>Key Pairs



Project / Compute / Access & Security

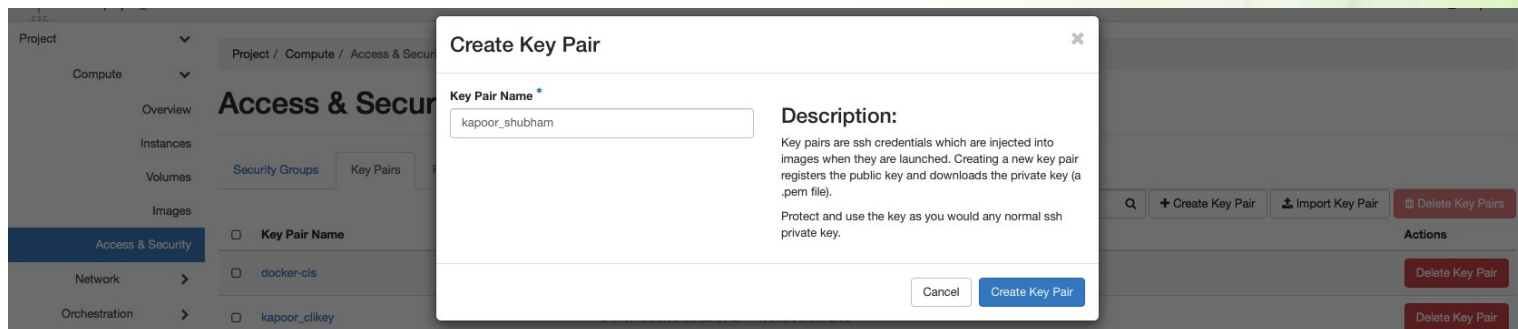
Access & Security

Security Groups | **Key Pairs** | Floating IPs | API Access

Filter + Create Key Pair Import Key Pair Delete Key Pairs

<input type="checkbox"/>	Key Pair Name	Fingerprint	Actions
<input type="checkbox"/>	docker-cls	69:3d:55:0a:1d:04:a9:51:90:5e:91:45:ce:9b:09:5d	Delete Key Pair

Click on create Key Pair, name key as **lastname_firstname**



Project / Compute / Access & Security

Access & Security

Security Groups | Key Pairs

Create Key Pair

Key Pair Name

Description:
Key pairs are ssh credentials which are injected into images when they are launched. Creating a new key pair registers the public key and downloads the private key (a .pem file).
Protect and use the key as you would any normal ssh private key.

Cancel Create Key Pair

Filter + Create Key Pair Import Key Pair Delete Key Pairs

<input type="checkbox"/>	Key Pair Name	Actions
<input type="checkbox"/>	docker-cls	Delete Key Pair
<input type="checkbox"/>	kapoor_clikey	Delete Key Pair

Storing a Key pair

Linux and Mac OS X



- Create .ssh directory in ~ if its not there already

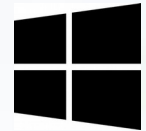
```
mkdir -p .ssh
chmod 700 .ssh
```
- Move key pair to .ssh directory

```
cd .ssh
mv ../Downloads/yourkey.pem .
```
- Make key unreadable by others

```
chmod 600 yourkey.pem
```
- Protect key with passphrase (Optional)

```
ssh-keygen -p -f yourkey.pem
```

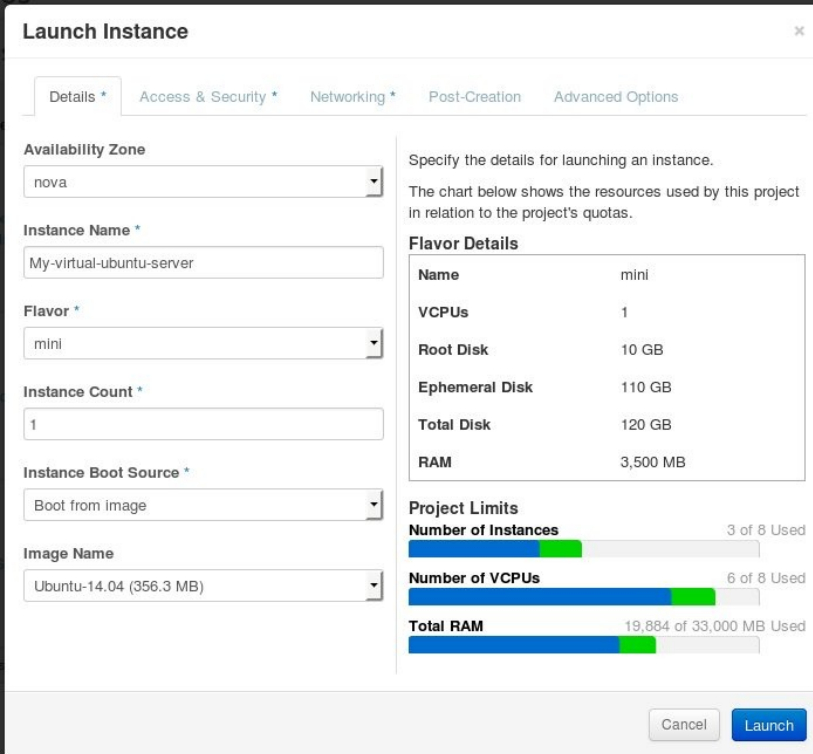
Windows



- Download Putty and Puttygen tools if you don't have them
 - <https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>
 - CertUtil -hashfile putty-0.70-installer.msi SHA256
 - 822da752a69f399377614561125f51a9220a7aa0f4e8afcf1b0e5cd341c0b676
- Load your **private key (yourkey.pem)** into **puttygen** and change it to .ppk format
- Open Putty, load .ppk file under **Connection | SSH | Auth | Private key file for authentication**
 - Provide user name **cloud-user**
 - Provide password which you added to Puttygen (Optional)

2. Launching Virtual machine

- Launching a VM can be done with cPouta web interface: <https://pouta.csc.fi>
- Most important settings:
 - **Flavor:** Size of the machine (cores, memory)
 - **Instance Boot Source:**
 - Boot from image or snapshot
 - What kind of operating system will be used
 - Typically a new Centos or Ubuntu version is used
 - **Key pair to be used**
(you can't add key later on!)
- When everything is ready, press: **Launch**



Launch Instance

Details * Access & Security * Networking * Post-Creation Advanced Options

Availability Zone
nova

Instance Name *
My-virtual-ubuntu-server

Flavor *
mini

Instance Count *
1

Instance Boot Source *
Boot from image

Image Name
Ubuntu-14.04 (356.3 MB)

Specify the details for launching an instance.
The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	mini
VCPUs	1
Root Disk	10 GB
Ephemeral Disk	110 GB
Total Disk	120 GB
RAM	3,500 MB

Project Limits

Number of Instances	3 of 8 Used
Number of VCPUs	6 of 8 Used
Total RAM	19,884 of 33,000 MB Used

Cancel Launch

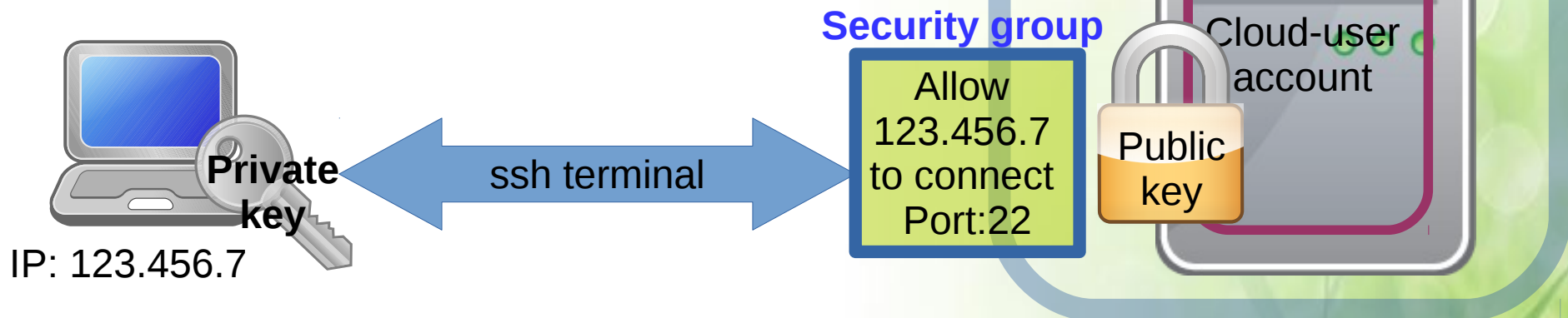
3. Associating IP address to your VM

- First the VM has just internal IP address that can't be used for connecting the VM.
- When the VM has been launched you can associate a public IP address for it: **Actions: Associate Floating IP**
- You have only a limited number of IP:s in use
- You can re-use old IP-addresses that are no longer associated to a VM

4. Opening VM to internet with security groups



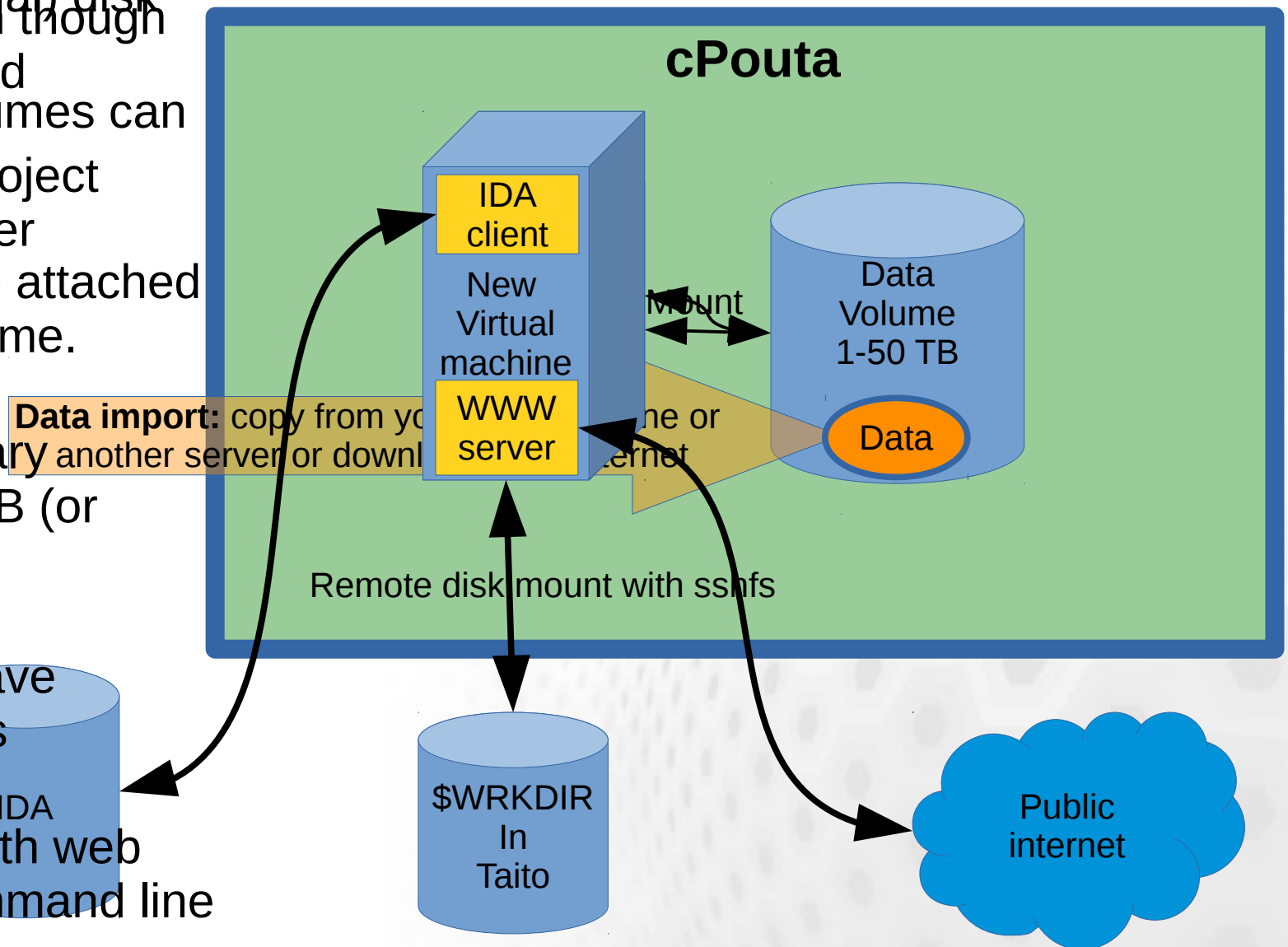
- By default the firewall of your VM blocks all incoming connections
- Security groups define combinations of ports and IP addresses for which the incoming connections are permitted
- Security groups are first created in the web interface and then applied to virtual machines
- One security group and include several “rules” and be used by several virtual machine
- One machine can use several security groups
- Typical case: allow connections from the IP address of your own computer to port 22 (ssh port).



Using Volumes in the cPouta



- In cPouta VM:s have only small local (virtual) disk preserved even though the VM is closed
- Virtual data volumes can be created
- Volumes are project specific, not user specific
- Volumes can be attached one VM at the time.
- No backup!
- Volume sizes vary between 1-50 TB (or more)
- A project can have several volumes
- Management with web interface or command line client



Data import: copy from your own server or download from another server or download from the internet

6. Installing software

- So now there is a an empty virtual machine running
 - How do I add the software I need ?
- 1) **System level repository installation:**
 - Command as superuser - **sudo**
 - Centos and Scientific linux: **sudo yum**
 - Ubuntu: **sudo apt**
 - add missing libraries and linux commands and many applications too
 - 2) **Compile codes or download pre-compiled binaries.**
 - 3) Install Docker and use Docker images.
 - 4) Use Conda!

Lifecycle of your virtual machine

- **Active** – as name says. Consumes billing units regardless of the real usage
- **Shut off** – Not active, but still reserves the resources. Consumes still billing units.
- **Suspended** – Temporarily paused. Current state saved. Can be revoked. Consumes billing units
- **Shelved** – VM is shut off. State is saved and IP and volume information is preserved. Can be later on revoked (un-shelved). Does not use billing units.
- **Terminate** – Removes the Virtual Machine.

Security guidelines for cPouta

- You are responsible for the security of your VM
- Run **updates** regularly or preferable enable automatic updates. Some updates, such as a kernel upgrade, require a reboot of the virtual machine.
- Subscribe to security announcements for your OS
- Run a **restrictive firewall/security group**: open only the connections that are needed.
- Disable/Remove unneeded accounts and if possible use keys in stead of passwords for authentication.
- An ideal scenario might be 3 accounts - root (with ssh disabled), a user account for a sysadmin(key login only) and a user-level account for a service(login disabled).
- **Terminate or shelve VM:s** that are not in active use.

Managing a cPouta project

- A cPouta project contain a set of resources: cores, memory, storage, ip-addresses
- Project members can build one or several VM:s and volumes based on the granted resources.
- A default project contains: 8 cores, 32 GB memory, 1 TB disk space, 2 ip addresses.
- If needed you can ask for more resources for your project.
- When VM:s and Volumes are active they are consuming billing units (even if no one is using them).
- Project members can manage other members machines and volumes too
- Your CSC account can be a member in many cPouta projects.