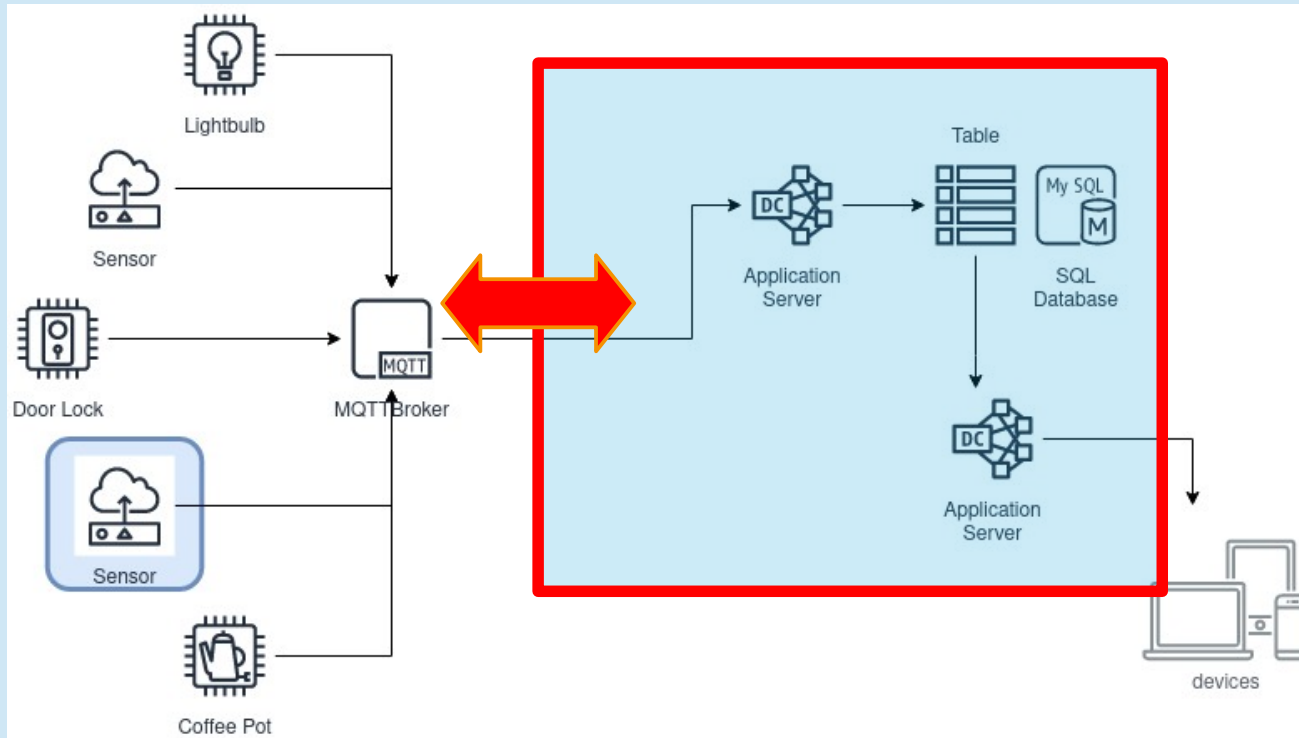


# LAB Exercises

## Prepare VM on the Stratus Cloud

**2024**

# IoT LAB – Cloud Services

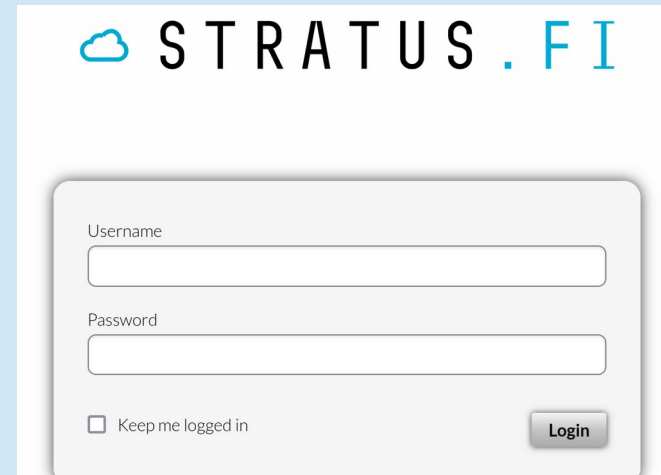


# IoT LAB – today's tasks

- Prepare virtual machine(s)
- Construct IP connectivity
- Install InfluxDB
- Install Grafana

# Stratus.FI

- OpenNebula virtualization platform
- Open to all students from FI
- Pre-configured virtual machines
- Login to management the same as login to KYPO PCs
- <https://stratus.fi.muni.cz/>



The screenshot shows the login interface for Stratus.FI. At the top, there is a logo consisting of a blue cloud icon followed by the text "STRATUS.FI" in a black, sans-serif font. Below the logo is a light gray rounded rectangle containing the login form. The form has two input fields: "Username" and "Password". Below the password field is a checkbox labeled "Keep me logged in". To the right of the checkbox is a gray button with the text "Login" in white.

# Stratus.FI

- Two ways to log into VM:
  - PKI
  - **password**

# Using Stratus.FI with PKI

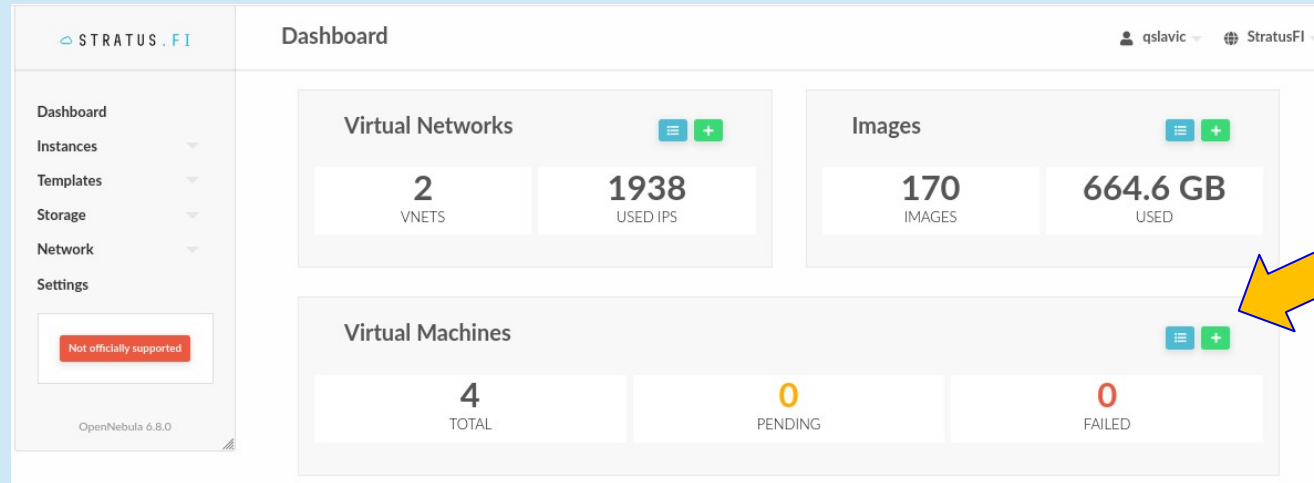
- Prepare ssh keys
- Linux
  - `ssh-keygen -b 4096 -t rsa -f Stratus`
  - PassPhrase preferably empty
  - Will generate files **Stratus** and **Stratus.pub**
- Windows
  - Puttygen

# Using Stratus.FI with PKI

- Save your public key to Stratus
  - Login
  - Click your name → Settings → Auth → Public SSH Key
  - You can add new key if you already have one installed
- Create virtual machine

# Create virtual machine

- Use template 1307 - Debian12[CVTFI]
  - Instantiate as persistent
  - Assign a name
  - Click Create
- Don't use excessive resources



The screenshot shows the StratusFI dashboard with the following data:

Category	Sub-category	Value
Virtual Networks	VNETS	2
	USED IPS	1938
Images	IMAGES	170
	USED	664.6 GB
Virtual Machines	TOTAL	4
	PENDING	0
	FAILED	0

A yellow arrow points to the 'Virtual Machines' section of the dashboard.



# Create virtual machine

The screenshot shows the 'Create Virtual Machine' page in the Stratus.FI web interface. The page title is 'Create Virtual Machine' and the user is 'qslavic'. The interface includes a sidebar with navigation options: Dashboard, Instances, Templates, Storage, Network, and Settings. A red box in the sidebar indicates 'Not officially supported' for OpenNebula 6.8.0. The main content area shows a list of templates, with 'Debian 12 [CVTFI]' selected. Below the list, there are options to 'Instantiate as persistent', 'VM name' (with a placeholder 'Your VM Name'), 'Number of instances' (set to 1), and 'Start on hold'. The 'Create' button is highlighted with a yellow arrow.

STRATUS.FI

## Create Virtual Machine

qslavic StratusFI

← Reset Create

You selected the following Template: **Debian 12 [CVTFI]**

ID	Name	Owner	Group	Registration time
1428	IoT_Lab_Master	qslavic	users	10/11/2023 14:00:18
1427	PV284_Net_Test	qslavic	users	10/11/2023 13:32:09
1426	PV284_Stratus_MASter_Copy	qslavic	users	10/11/2023 13:27:12
1369	PA195_Riak	xdohnal	pa195	02/10/2023 09:11:36
1343	PA195-hadoop-single	xdohnal	pa195	22/09/2023 14:31:38
1322	PA220 - home assignments	xdohnal	users	14/09/2023 21:23:50
1317	PA195-Hadoop	bar	users	06/09/2023 19:44:08
<b>1307</b>	<b>Debian 12 [CVTFI]</b>	<b>oneadmin</b>	<b>oneadmin</b>	<b>12/06/2023 09:39:22</b>
1301	Alpine Linux 3.18.0	xvaclav	users	25/05/2023 12:45:26
1286	Fedora Linux 38 [CVTFI]	oneadmin	oneadmin	18/04/2023 16:56:20

10 Show 12 entries

Previous 1 2 3 4 5 ... 8 Next

Instantiate as persistent

VM name

Number of instances

Start on hold

**Debian 12 [CVTFI]**

# Using Stratus.FI with password

- Generate DES encrypted password
- Create the VM
- Install the root password
- Log-in via VNC
- Create an user
- Log-in via ssh and newly created user

# Generate DES encrypted password

- Linux
  - `mkpasswd -m des your_password`
  - `mkpasswd` is part of the `whois` package
- Windows
  - Use some on-line Python interpreter, e.g., <https://www.programiz.com/python-programming/online-compiler/>
    - `import crypt`
    - `hashed_password = crypt.crypt("your_password", "salt")`
    - `print(hashed_password)`

# Using Stratus.FI with password

- Open the VM → Conf
- → Update Configuration
- → Context → Custom vars

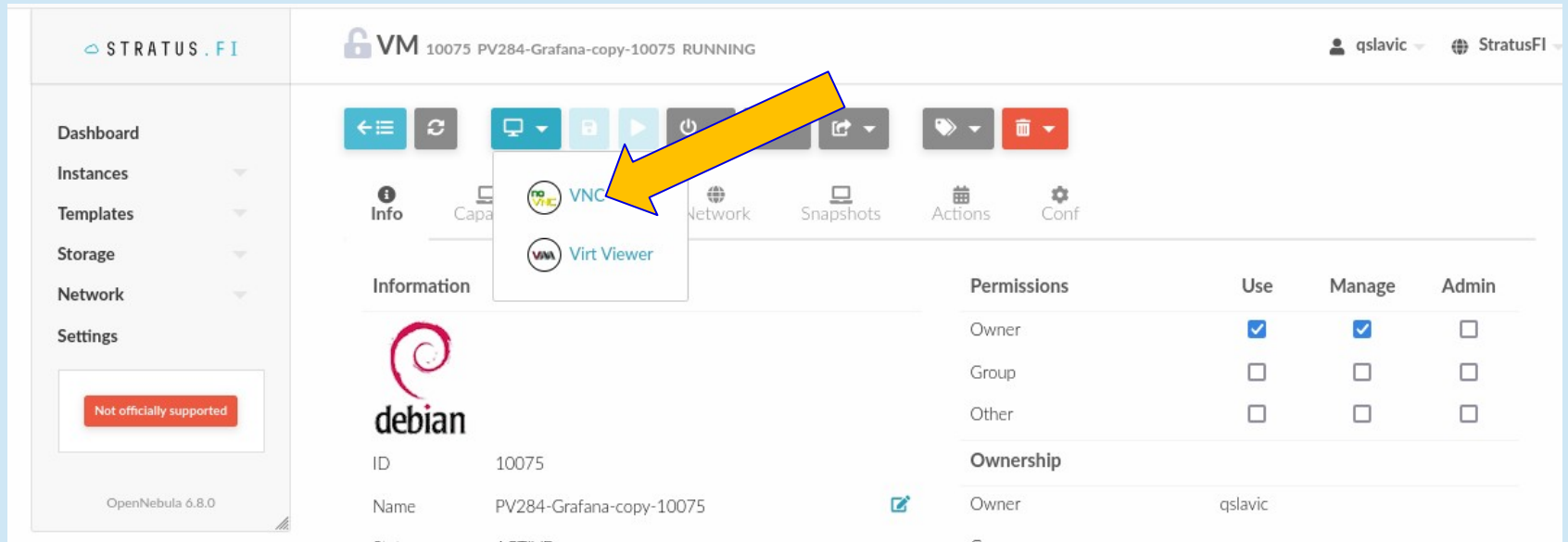
The screenshot shows the 'Update VM Configuration' interface for a VM named '12 [CVTFI]-9913'. The user is logged in as 'oujezsky' and the interface is for 'StratusFI'. The 'Update' button is highlighted with a yellow arrow. The 'Context' tab is selected, and the 'Custom vars' section is active. The 'Custom vars' section contains two entries:

Name	Value
CRYPTED_PASSWORD	<input type="text"/>
DISK_ID	2

Each entry has a '+' button below it to add more variables. A second yellow arrow points to the 'CRYPTED\_PASSWORD' value input field.

# Create a local user

- `useradd -m your_username`
- `passwd your_username`



The screenshot shows the OpenNebula web interface for a VM. The VM is identified as 'VM 10075 PV284-Grafana-copy-10075 RUNNING'. A yellow arrow points to the 'VNC' button in the top toolbar, which has opened a dropdown menu with 'VNC' and 'Virt Viewer' options. Below the toolbar, there are sections for 'Information', 'Permissions', and 'Ownership'. The 'Information' section shows the OS as 'debian' and the ID as '10075'. The 'Permissions' table shows that the user 'qslavic' has 'Use' and 'Manage' permissions. The 'Ownership' section shows the owner is 'qslavic'.

Permissions	Use	Manage	Admin
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Group	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Other	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

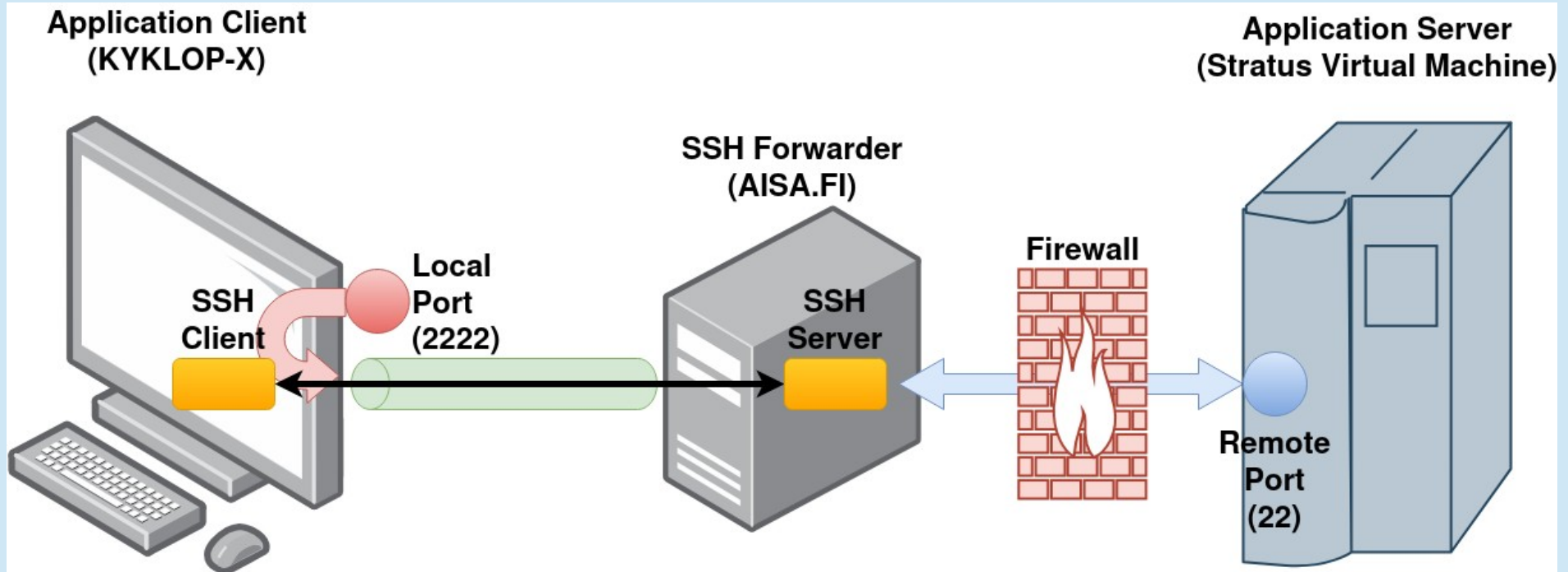
  

Ownership	Owner
Owner	qslavic

# Using Stratus.FI

- Now you can access the console from Stratus management interface
- Access is allowed from FI, not from KYPO LAB
- Similar situation is expected in real life as well
- Two way to overcome the fence:
  - SSH port forwarding
  - OpenVPN

# SSH port forwarding



# SSH port forwarding

- Mainly for web interface to your servers
- Example Linux
  - `ssh -L 2222:Your_VM_IP_addr:22`
  - `-L 3000:Your_VM_IP_addr:3000`
  - `-L 8086:Your_VM_IP_addr:8086 yourlogin@aisa.fi.muni.cz`
  - `ssh -i Stratus -p 2222 root@127.0.0.1`
  - Web browser `http://localhost:3000`



# SSH port forwarding – Windows Client

- Set host aisa.fi.muni.cz
- Side pannel → connection → SSH → Tunnel
- *Source port* **2222**
- *Destination* *Your\_VM\_IP\_addr:22* → **Click add**
- *Source port* 3000
- *Destination* *Your\_VM\_IP\_addr:3000* → **Click add**
- *Source port* 8086
- *Destination* *Your\_VM\_IP\_addr:8086* → **Click add**
- Save the configuration
- Open a new putty window, **host = 127.0.0.1, port = 2222**

# InfluxDB

- Follow the instructions on the web  
<https://docs.influxdata.com/influxdb/v2/install/>
- Or go directly to the download page  
<https://www.influxdata.com/downloads/>
- If the gpg package is missing, install it via ***apt-get install gpg***
- Don't forget to start the influxdb: ***sudo service influxdb start***
- And check it's running: ***sudo service influxdb status***

# Grafana

- Follow the instructions on the web:  
<https://grafana.com/docs/grafana/latest/setup-grafana/installation/debian/>
- Get started with Grafana and InfluxDB:  
<https://grafana.com/docs/grafana/latest/getting-started/get-started-grafana-influxdb/>
- Login to Grafana: <https://127.0.0.1:3000>
- Default credentials: admin/admin

# Create a database (bucket) in InfluxDB

- Open the InfluxDB UI: <http://localhost:8086>
- Create a Bucket (InfluxDB v2 uses "buckets" instead of databases):
  - Log in to the InfluxDB UI.
  - Go to Data > Buckets > Click Create Bucket.
  - Enter a name for the bucket. > Click Create to save the bucket.
- Set Up Authorization (Token):
  - Go to Data > Tokens.
  - Either create a new token or use the default one to authenticate your API calls.
  - Copy the token, you'll need it for data interactions via the InfluxDB API or InfluxDB CLI.

# Add InfluxDB as a Data Source in Grafana

- Log in to Grafana: Go to your Grafana instance (e.g., <http://localhost:3000>).
- Add Data Source:
  - In the left sidebar, click on Configuration > Data sources.
  - Click Add data source.
  - Select InfluxDB from the list.
- Configure InfluxDB Connection:
  - URL: Enter <http://localhost:8086>
  - Database Type: Choose InfluxQL or Flux based on your setup. InfluxDB v2 typically uses Flux as the query language.
  - Token: Paste the InfluxDB token you used to authenticate.
  - Organization: Enter the name of your InfluxDB organization.
  - Bucket: Select the bucket name where the sensor data is stored.
- Save & Test: Click Save & Test to ensure the connection is successful.

**Now you have the cloud SW installed.  
Next lab exercise we connect the RockPi  
as data source.**