canarie

Technical Guide

How to Use DAIR Cloud Resources: Linux/Windows/GPU

Table of Contents

Security and Account Defaults

> Using SSH Keys to Secure Access to Your VMs 5

> Setup Security Groups / Firewalls 11

> Creating Default Accounts for Your VMs 14

The Basics

> Creating an Instance or Virtual Machine (VM) 17

> Logging into an Instance 29

> GPU Specific Access over VNC 36

> Creating User Accounts 46

Advanced Topics (Linux OS instances only)

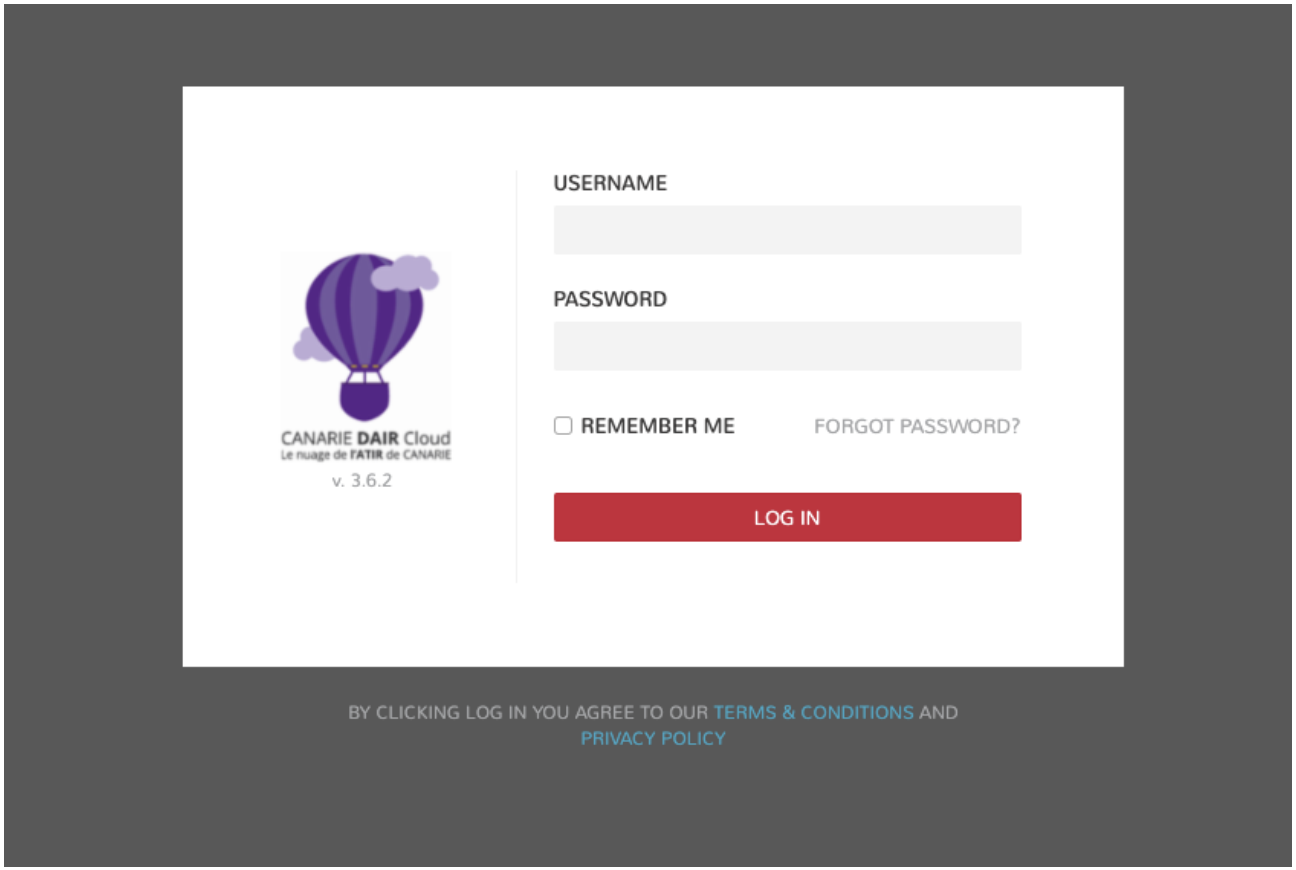
> Morpheus CLI 49

Security and Account Defaults:
Creating and Using Your SSH Key
(SSH key logins are required in DAIR)

Web Login to the DAIR Cloud:

> https://cloud.canarie.ca/login/auth

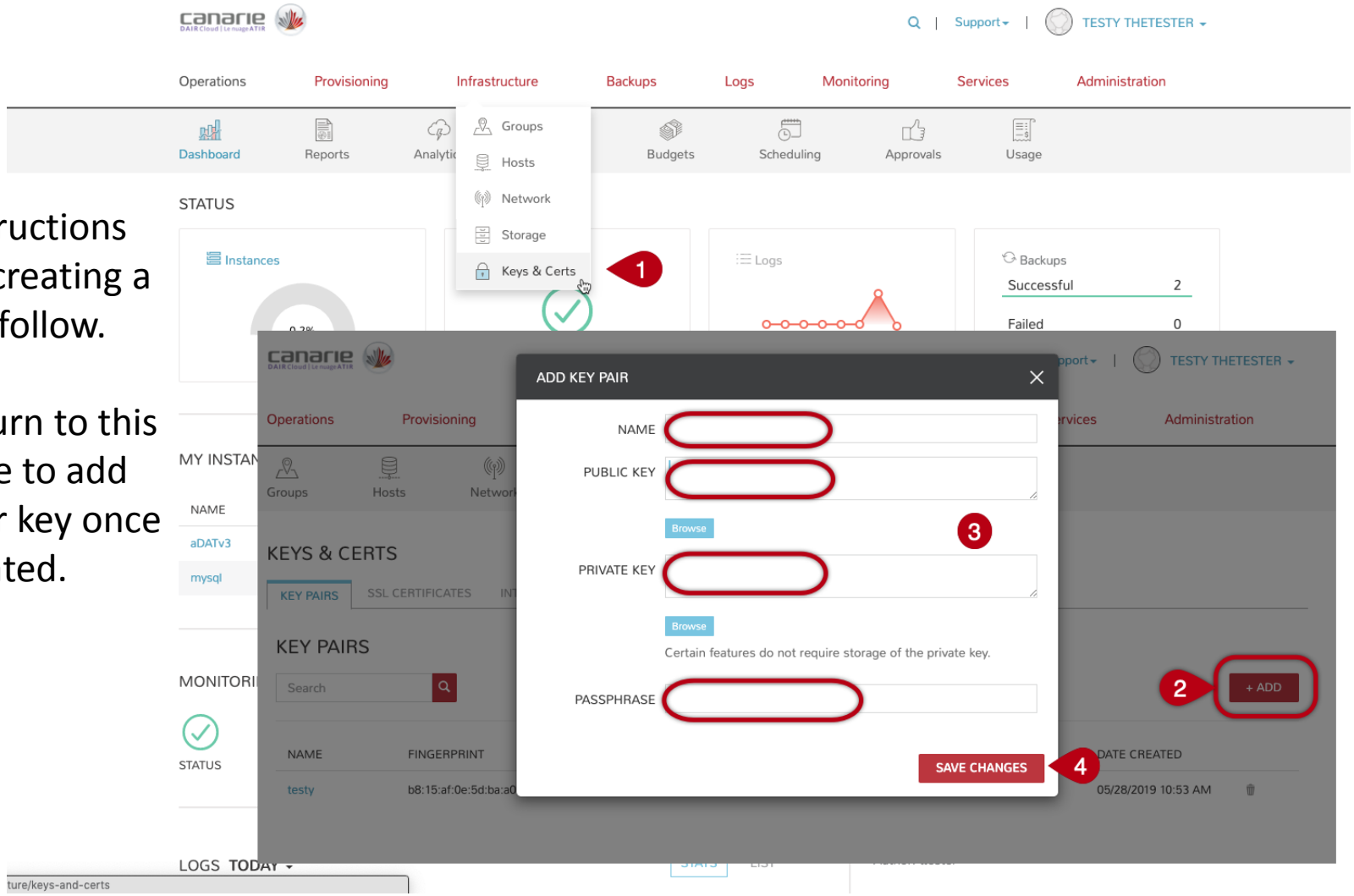
> Use credentials emailed to you



Adding your SSH key to DAIR

Instructions for creating a key follow.

Return to this page to add your key once created.



How to Create an SSH Key (Mac or Linux)

1. On MacOS
   - Open a terminal and go to Step 3
2. On Windows 10
   - Add the Linux subsystem for windows (see https://docs.microsoft.com/en-us/windows/wsl/install-win10)
   - Install Ubuntu or CentOS from the app store
   - Open Linux command shell window from Start menu
3. On the command prompt run this command:

   ```
   $ ssh-keygen -t rsa -m PEM -C "your email address"
   ```

   - Follow the directions and name your Key something memorable
   - Add a passphrase if you want a more secure key, just don't forget it!
4. This will create 2 files; "Name"(private key) and "name.pub"(public key)
5. Run the "cat" command on each file so you can copy and paste keys into your key-pair fields (Step 3 on previous slide)

How to Create an SSH Key (Win-8.x or earlier)

> Download “Putty” tool, the MSI complete installer, or:

> Download putty.exe and puttygen.exe

- https://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html

Package files

You probably want one of these. They include versions of all the PuTTY utilities.

(Not sure whether you want the 32-bit or the 64-bit version? Read the FAQ entry.)

MSI (‘Windows Installer’)

32-bit:	putty-0.71-installer.msi	(or by FTP)	(signature)
64-bit:	putty-64bit-0.71-installer.msi	(or by FTP)	(signature)

Unix source archive

.tar.gz:	putty-0.71.tar.gz	(or by FTP)	(signature)

Alternative binary files

The installer packages above will provide versions of all of these (except PuTTYtel), but you can download standalone binar

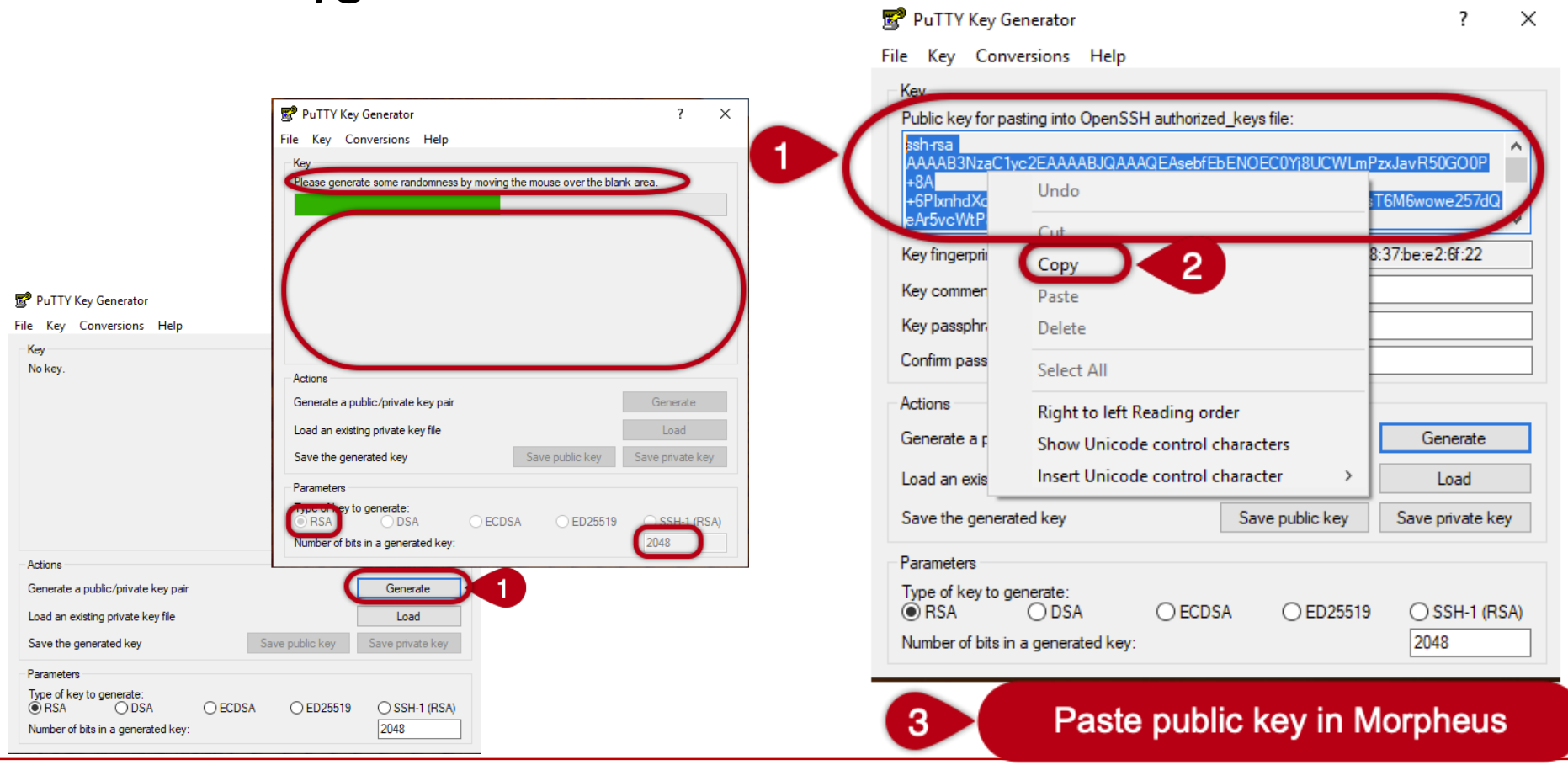
(Not sure whether you want the 32-bit or the 64-bit version? Read the FAQ entry.)

putty.exe (the SSH and Telnet client itself)			
32-bit:	putty.exe	(or by FTP)	(signature)
64-bit:	putty.exe	(or by FTP)	(signature)
pscp.exe (an SCP client, i.e. command-line secure file copy)			
32-bit:	pscp.exe	(or by FTP)	(signature)
64-bit:	pscp.exe	(or by FTP)	(signature)
psftp.exe (an SFTP client, i.e. general file transfer sessions much like FTP)			
32-bit:	psftp.exe	(or by FTP)	(signature)
64-bit:	psftp.exe	(or by FTP)	(signature)
puttytel.exe (a Telnet-only client)			
32-bit:	puttytel.exe	(or by FTP)	(signature)
64-bit:	puttytel.exe	(or by FTP)	(signature)
plink.exe (a command-line interface to the PuTTY back ends)			
32-bit:	plink.exe	(or by FTP)	(signature)
64-bit:	plink.exe	(or by FTP)	(signature)
pageant.exe (an SSH authentication agent for PuTTY, PSCP, PSFTP, and Plink)			
32-bit:	pageant.exe	(or by FTP)	(signature)
64-bit:	pageant.exe	(or by FTP)	(signature)
puttygen.exe (a RSA and DSA key generation utility)			
32-bit:	puttygen.exe	(or by FTP)	(signature)
64-bit:	puttygen.exe	(or by FTP)	(signature)

Create an SSH Key with Puttygen (Cont'd)

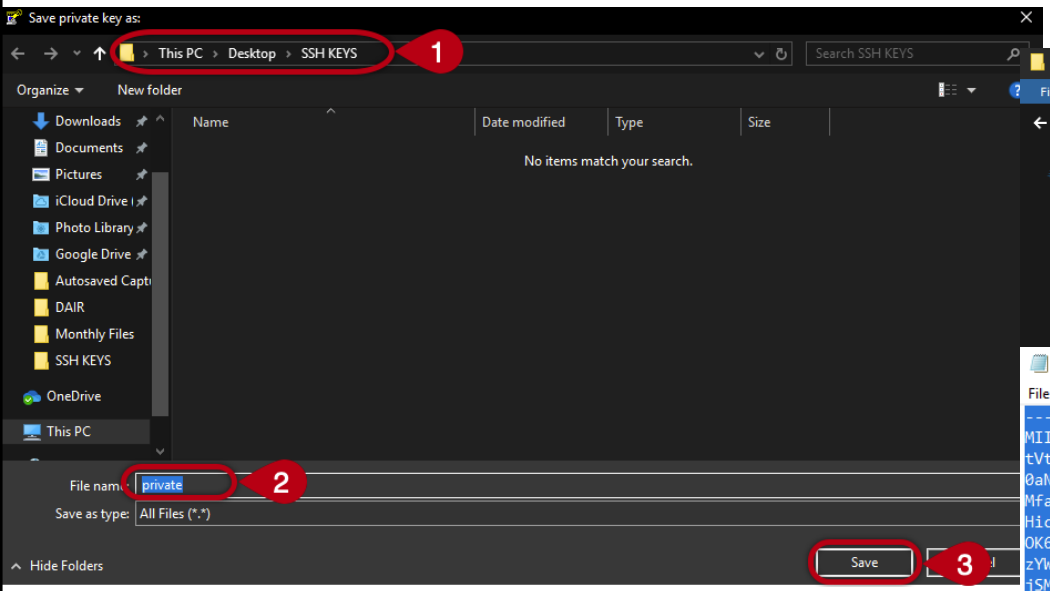
On Windows 8.1 and earlier versions:

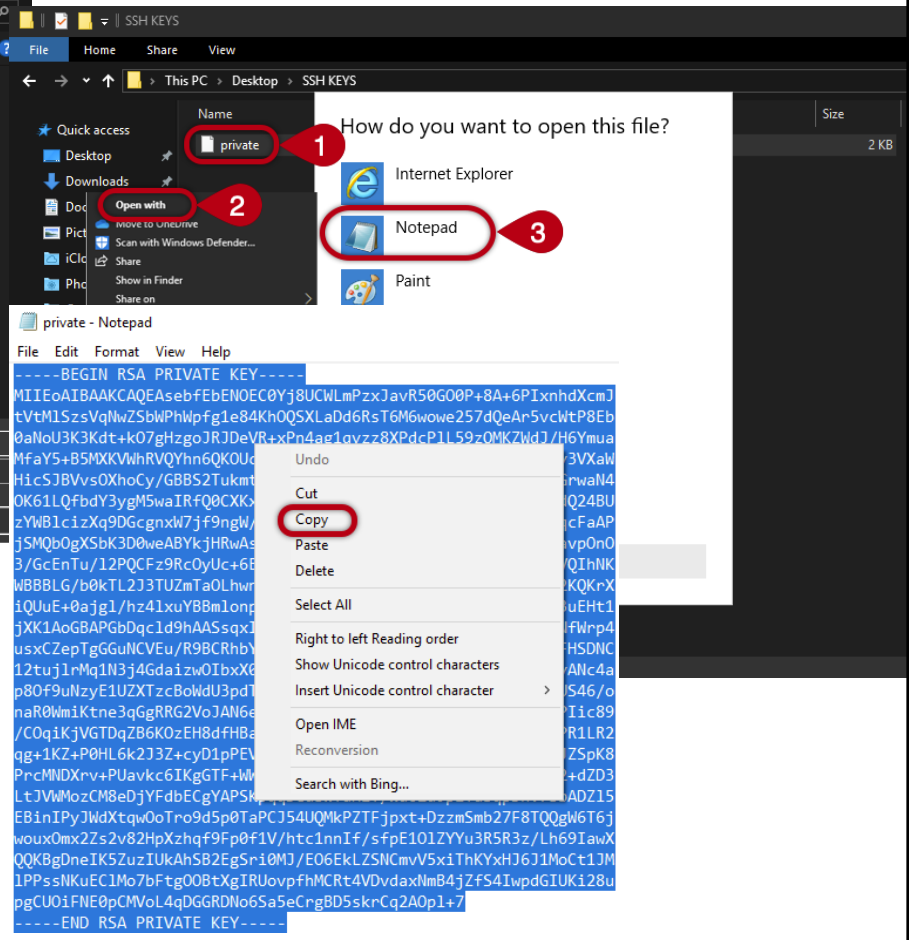
> Start Puttygen – Generate and copy public key



Create an SSH Key with Puttygen (Cont'd)

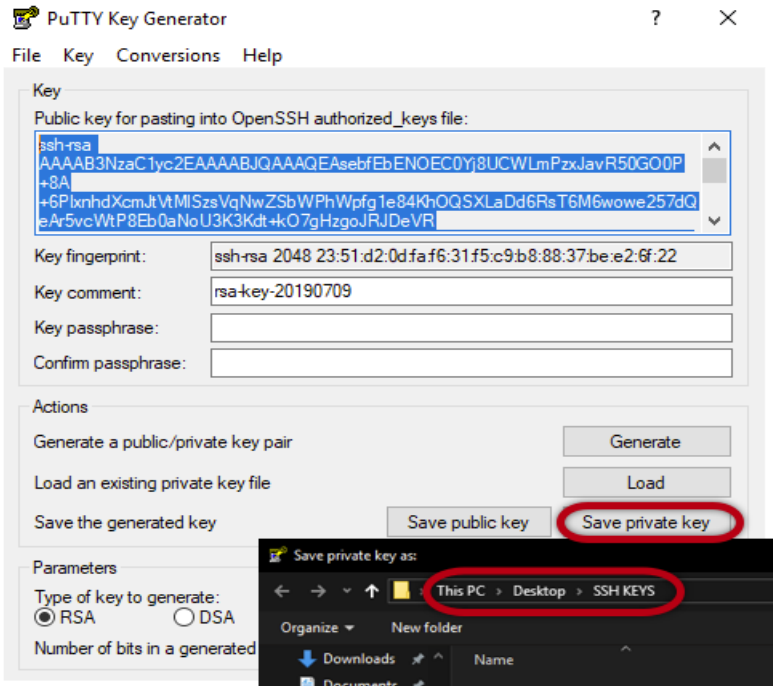
> Export your private key…

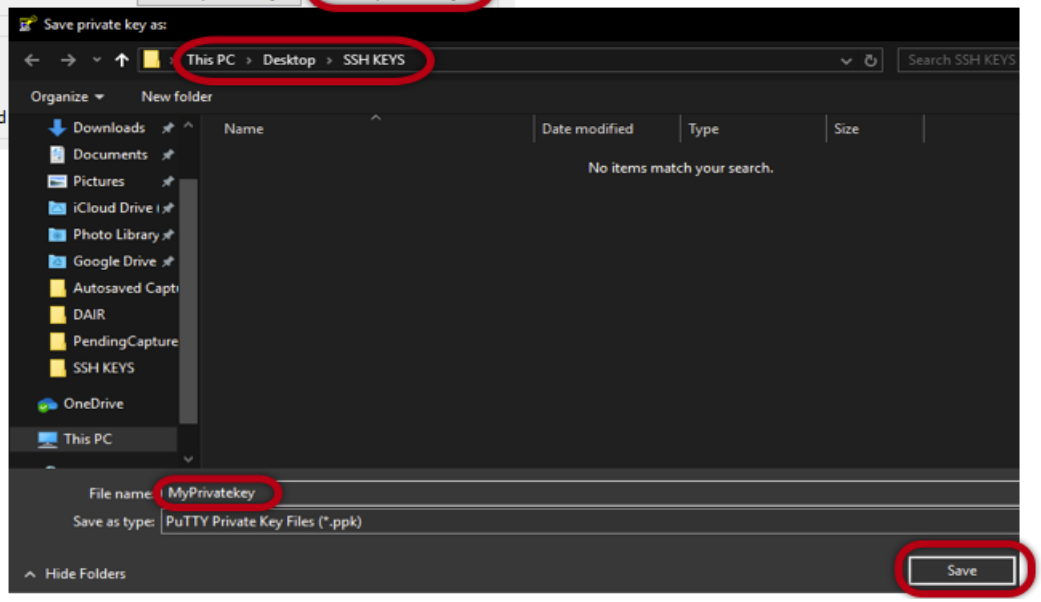




> Copy private key and paste in Morpheus (see Slide 5)

Create an SSH Key with Puttygen (Cont'd)





> Save your Private Key on your local Windows machine for use in Putty

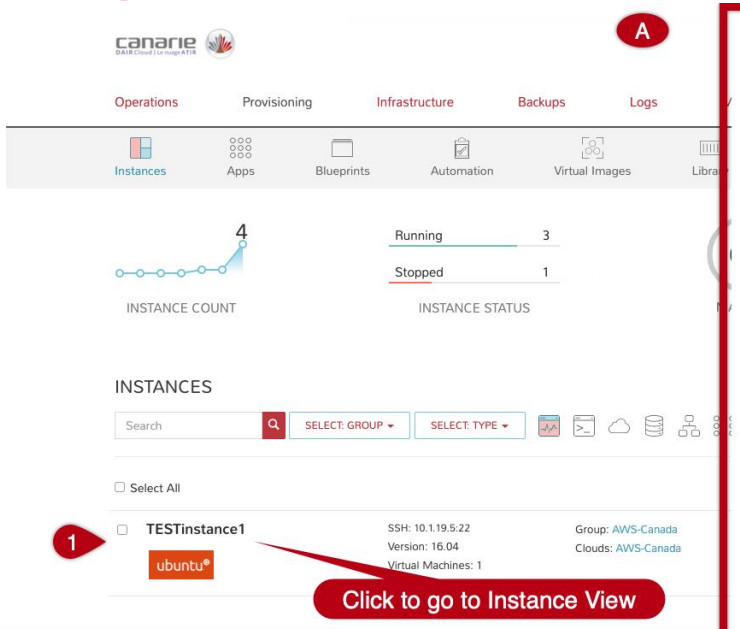
Once complete, return to page 5 for instructions on entering your key.

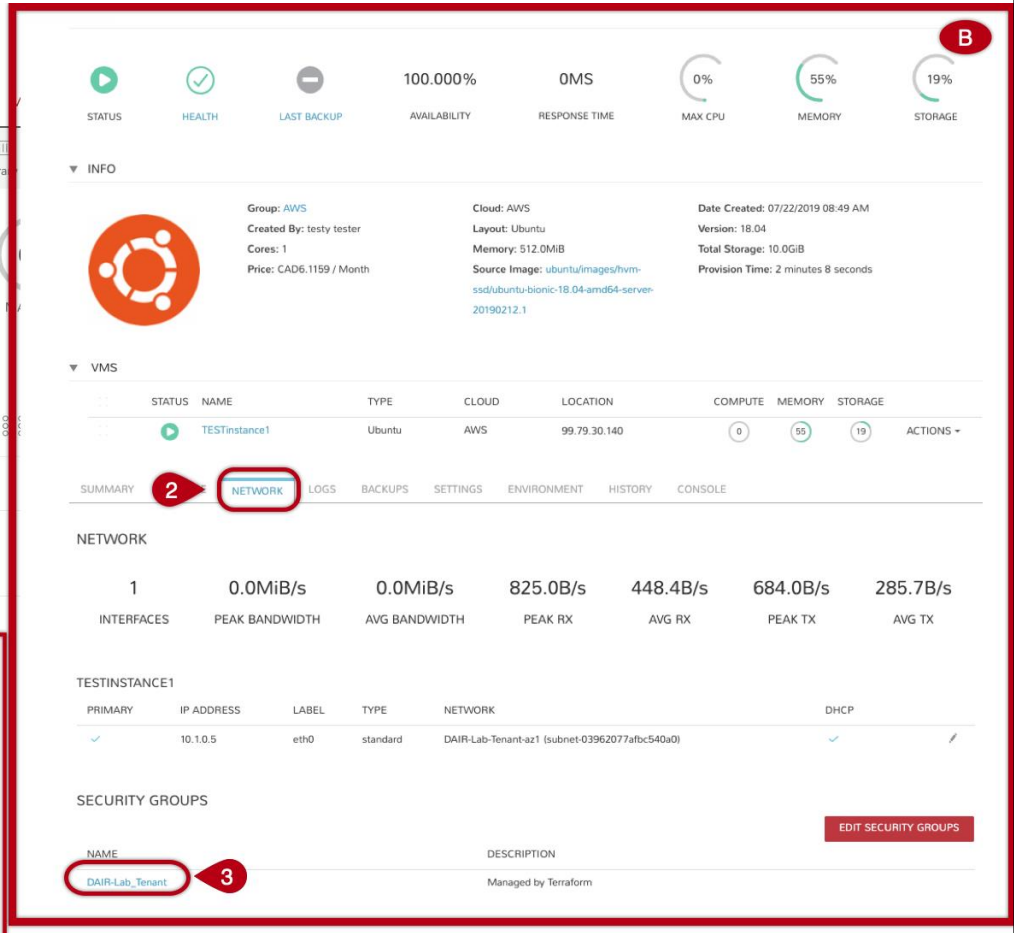
Security and Account Defaults:
Setup Security Groups / Firewalls

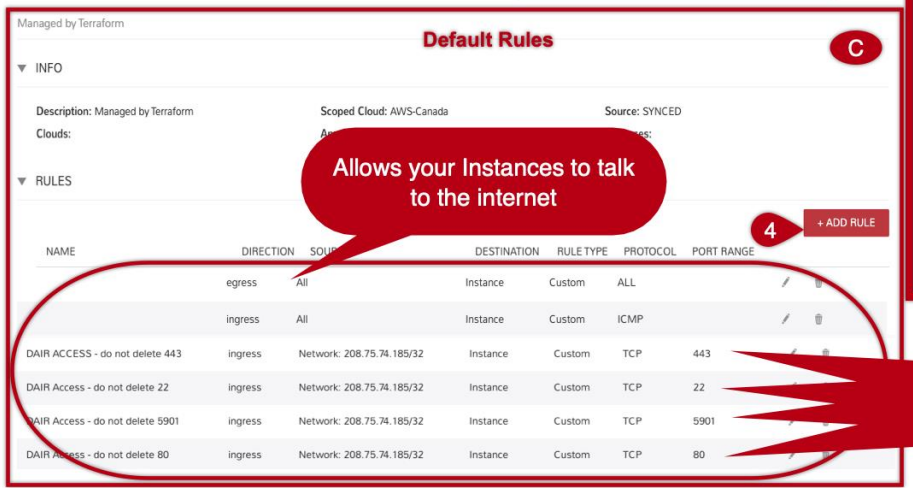
Avoid Getting Hacked!

- Security group rules for SSH connections **must not have a CIDR of 0.0.0.0/0**
  - The CIDR value specifies the allowable source IP address range for computers connecting to an instance
  - All-zeroes allows any computer in the world access and makes your instance highly vulnerable to attacks
- You must know the public IP address of the computer(s) you would like to permit access to your VMs in DAIR
  - You can determine your IP by searching Google for “what is my IP” on each machine you want to give access to
  - For example, to enable only a single IP address, make the CIDR **aa.bb.cc.dd/32** (where aa.bb.cc.dd is your public IP numeric value)
- Create a new rule for each IP address that you wish to grant SSH access (see details on next slide)

Configure/Edit your Firewall Rules

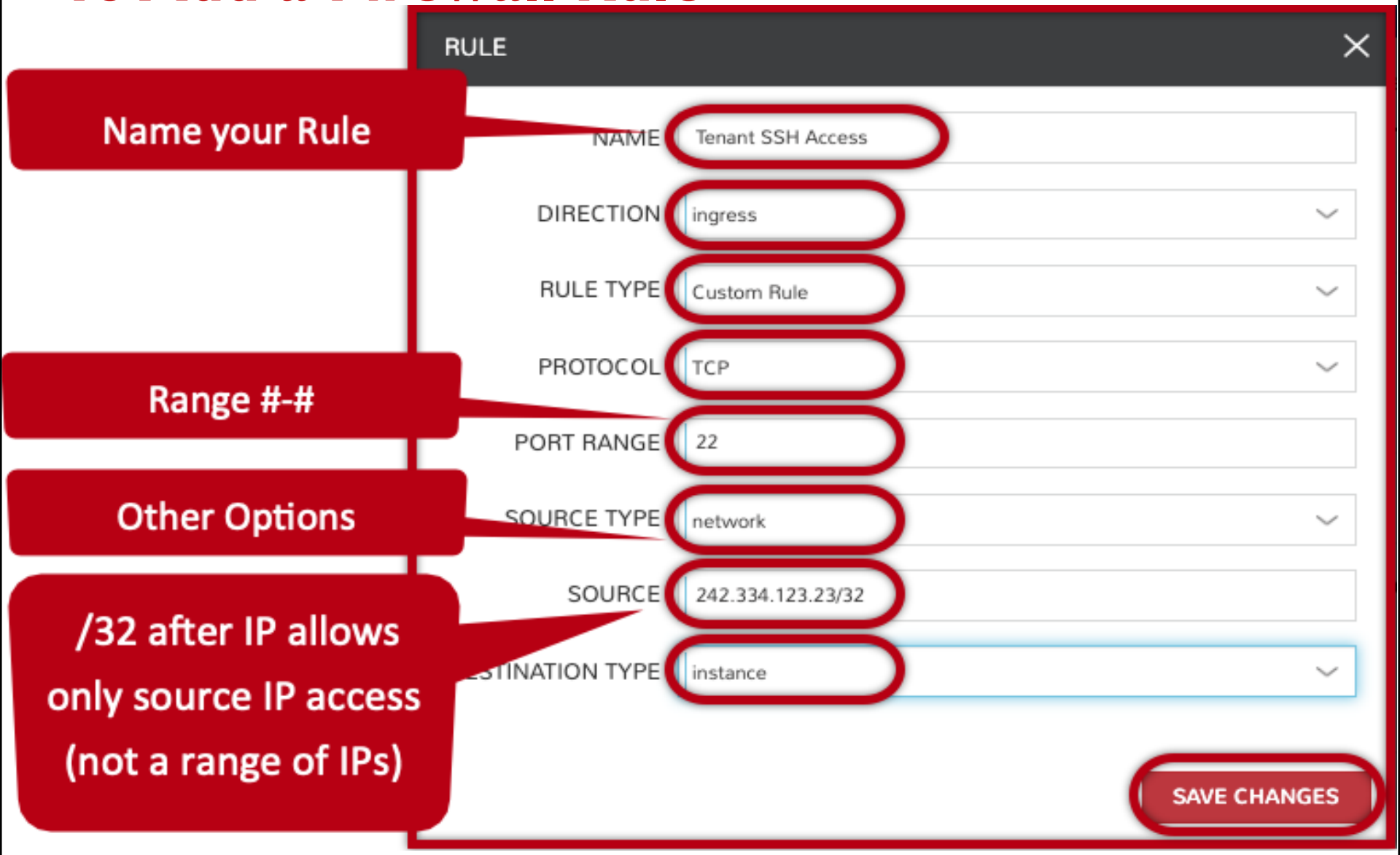




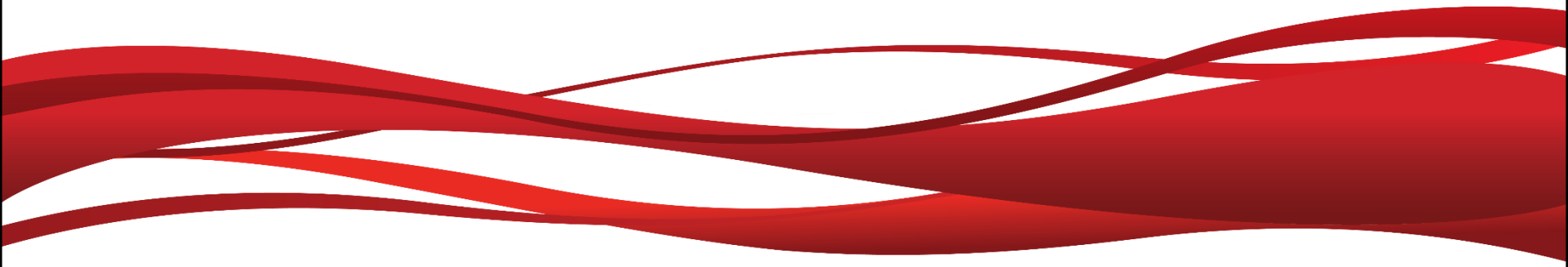


22 & 5901 Allows Console to work
443 & 80 is for agent updates

To Add a Firewall Rule

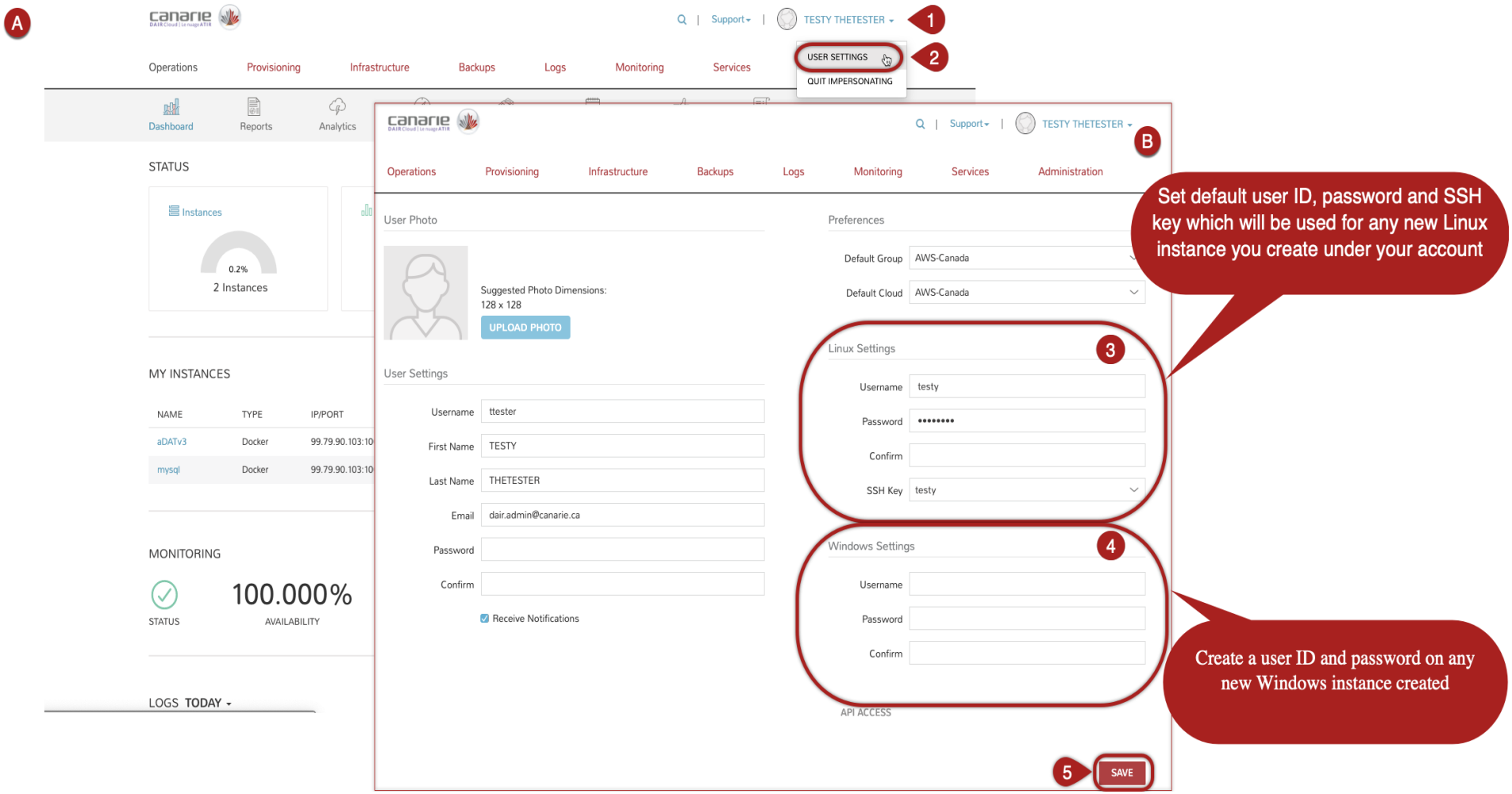


Security and Account Defaults:
Creating Default Accounts for Your VMs

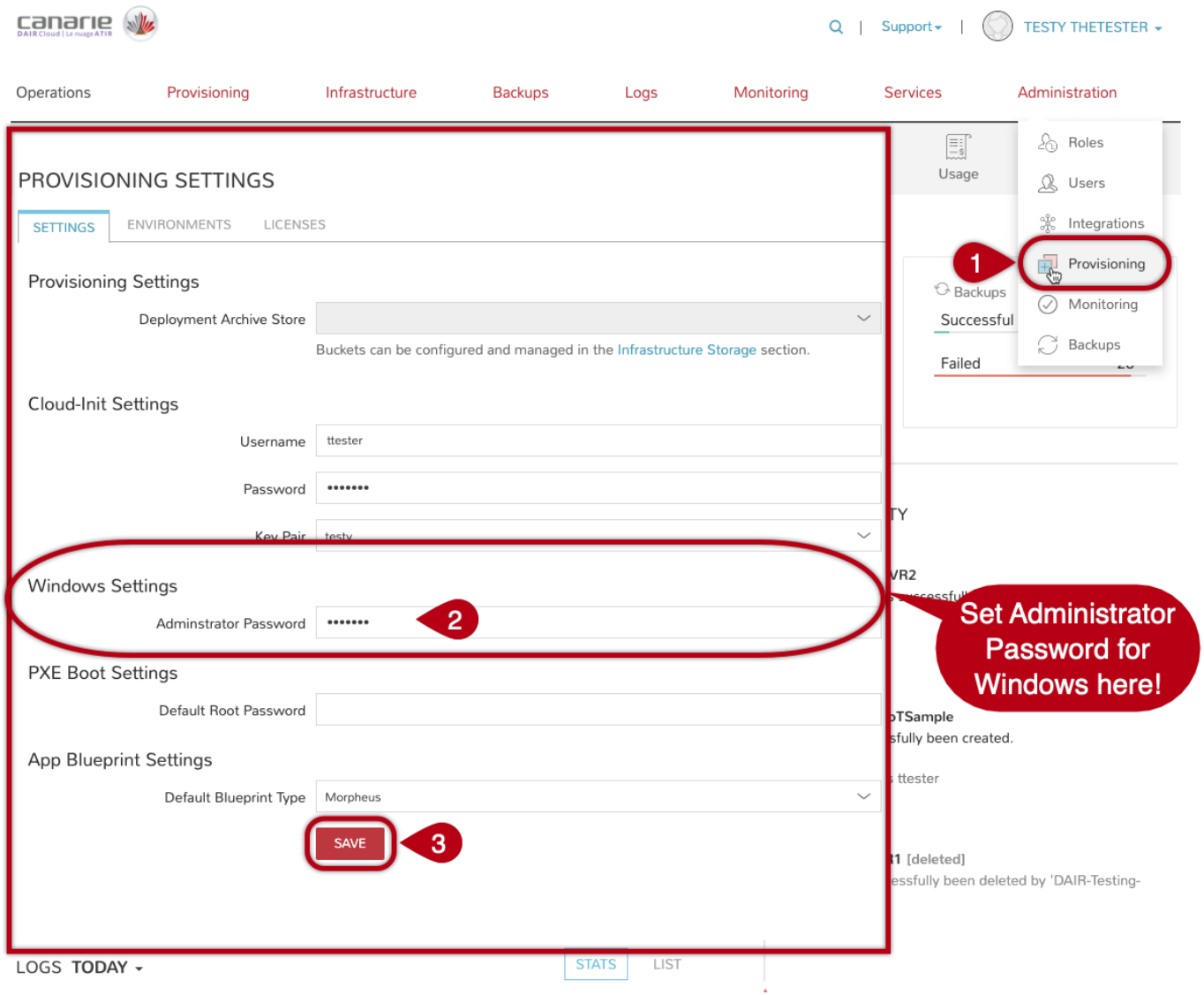


Configure Your Tenant Account

This must be done before deploying an instance

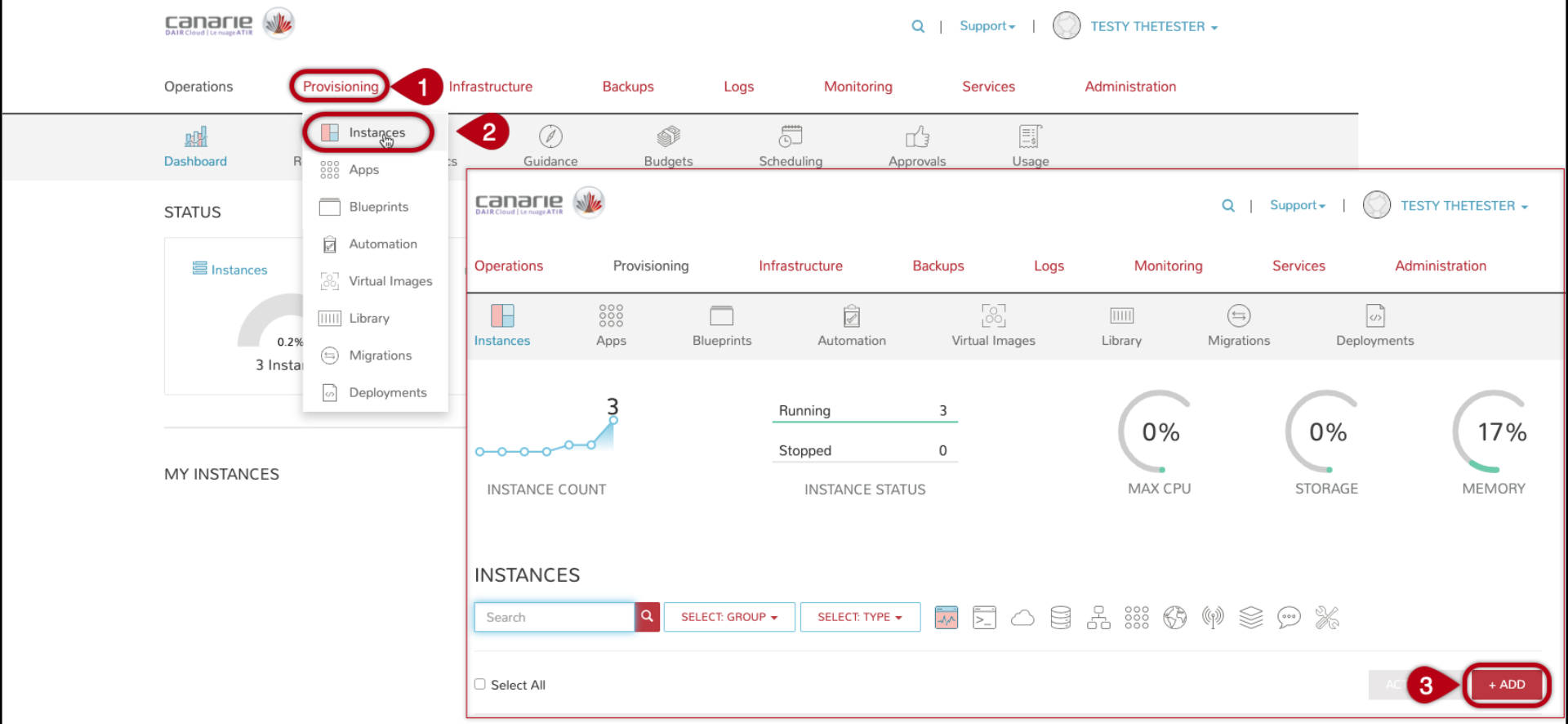


Configure Your Tenant Account for successful Provisioning



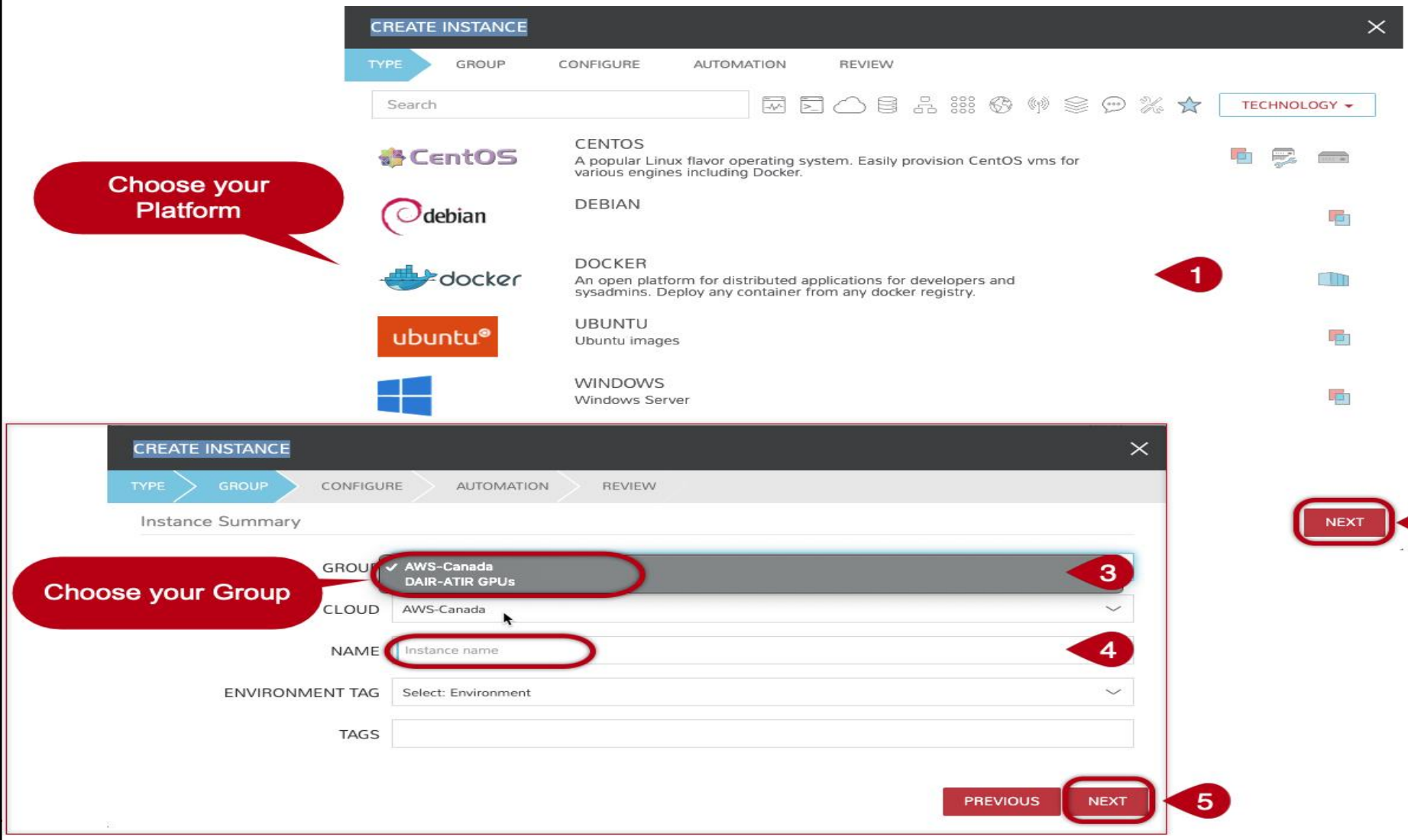
The Basics:
Creating an Instance or Virtual Machine (VM)

Creating your First Instance

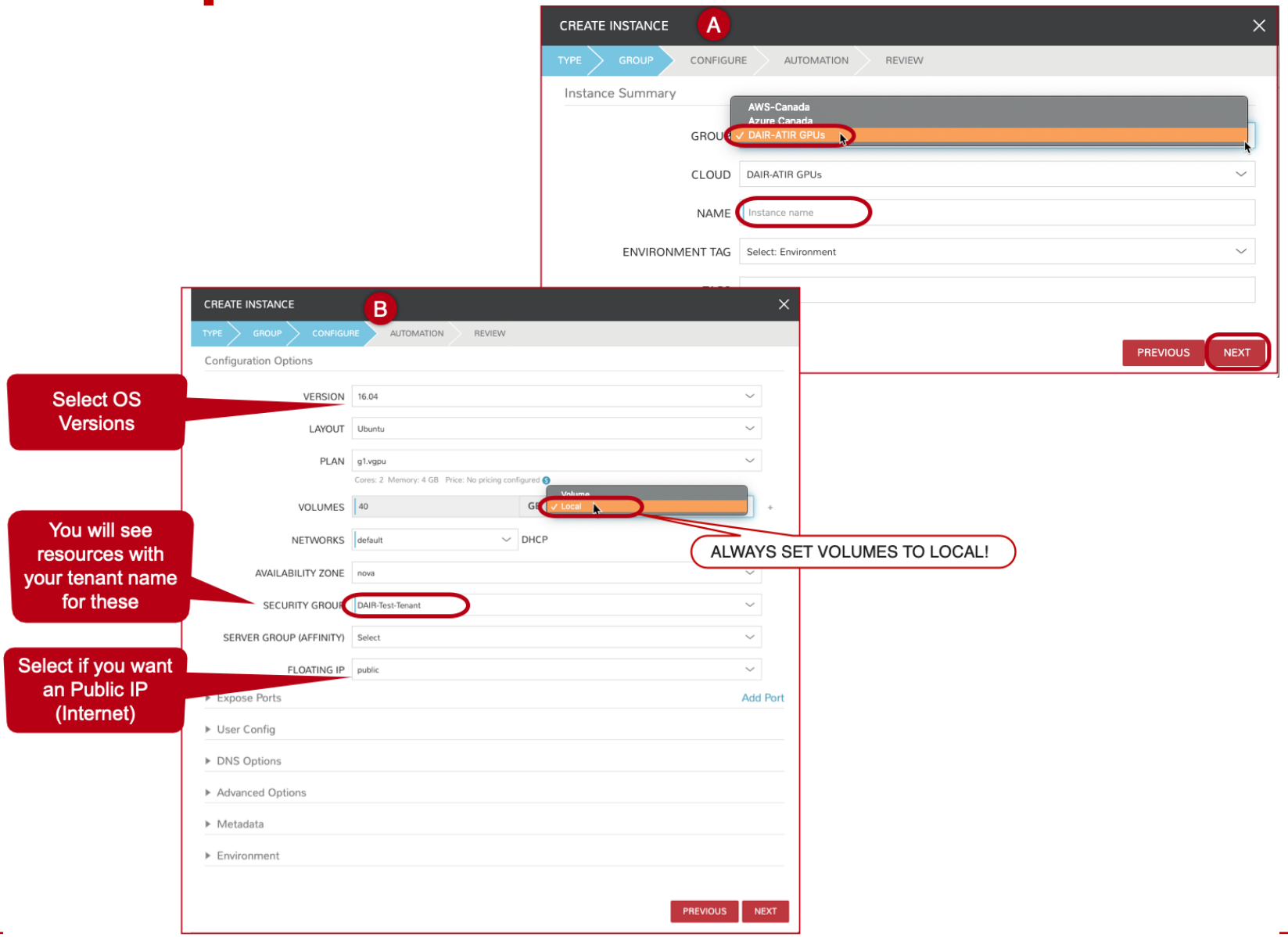


Creating an Instance: Platform and Cloud

New instances may be provisioned based on OS type.

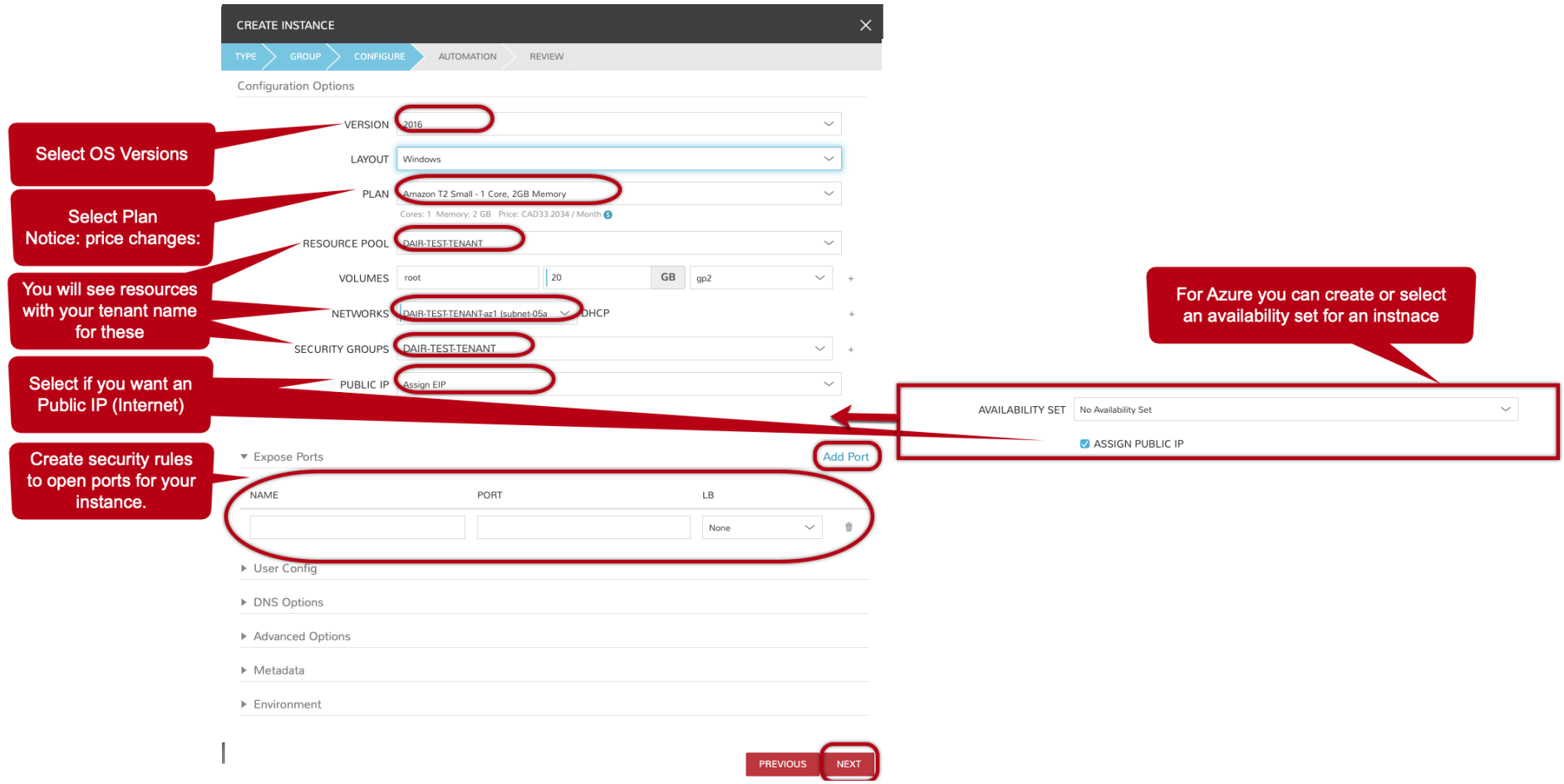


GPU Specific Details



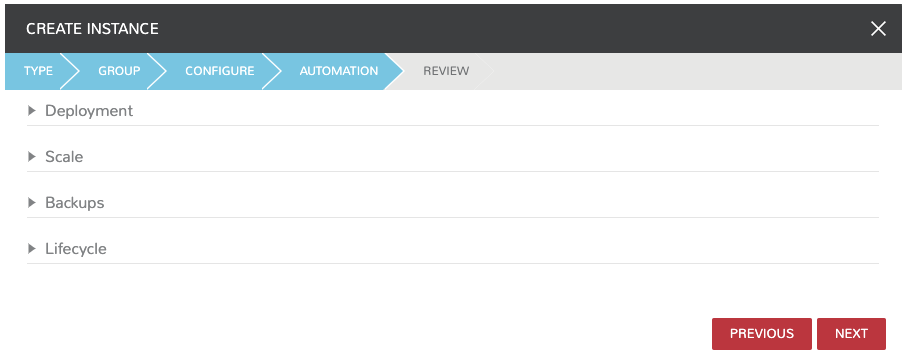
Creating a Windows Instance:

Provision configuration options for your instance.

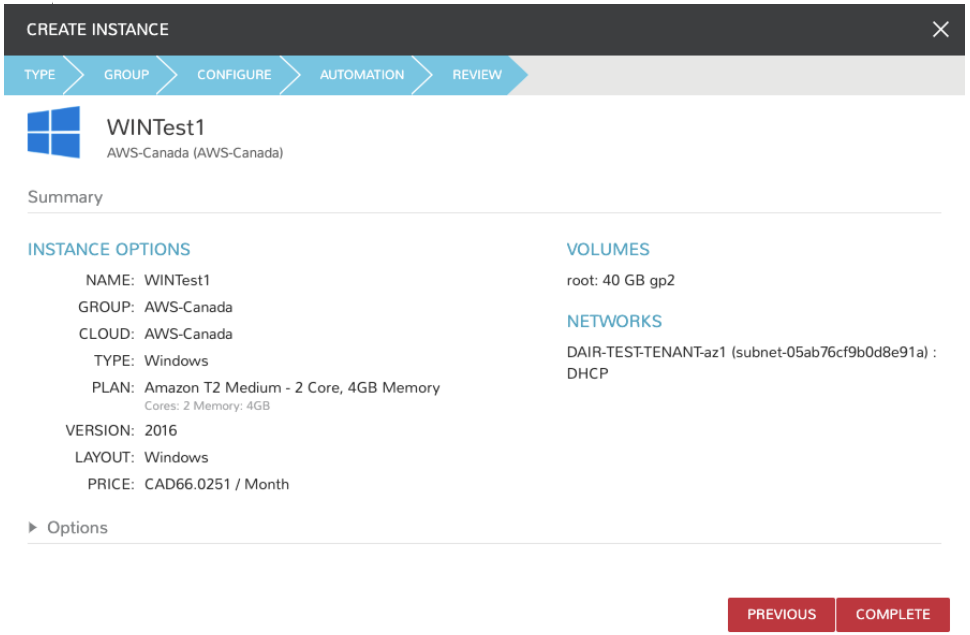


Creating a Windows Instance:

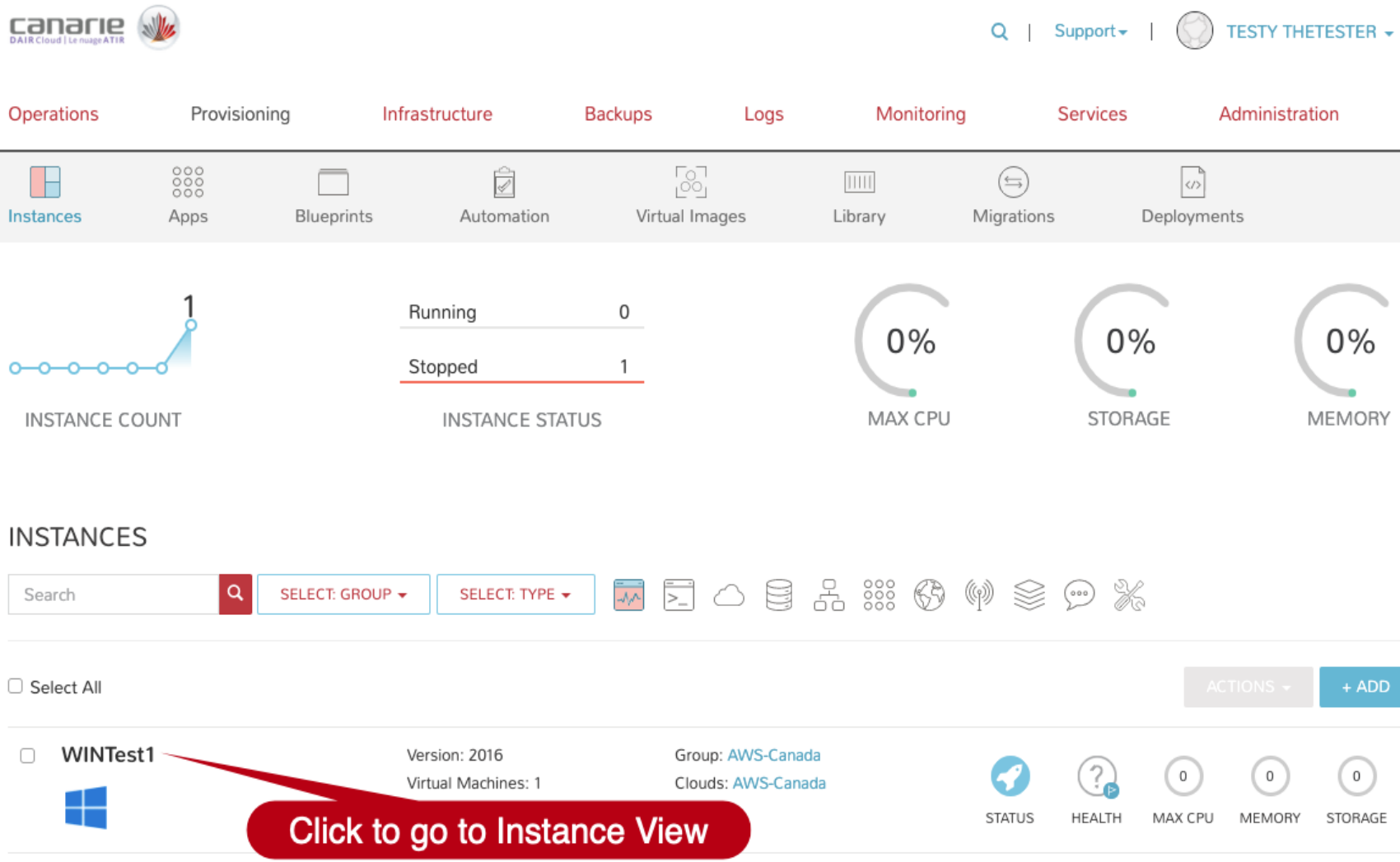
Some advanced configuration options:



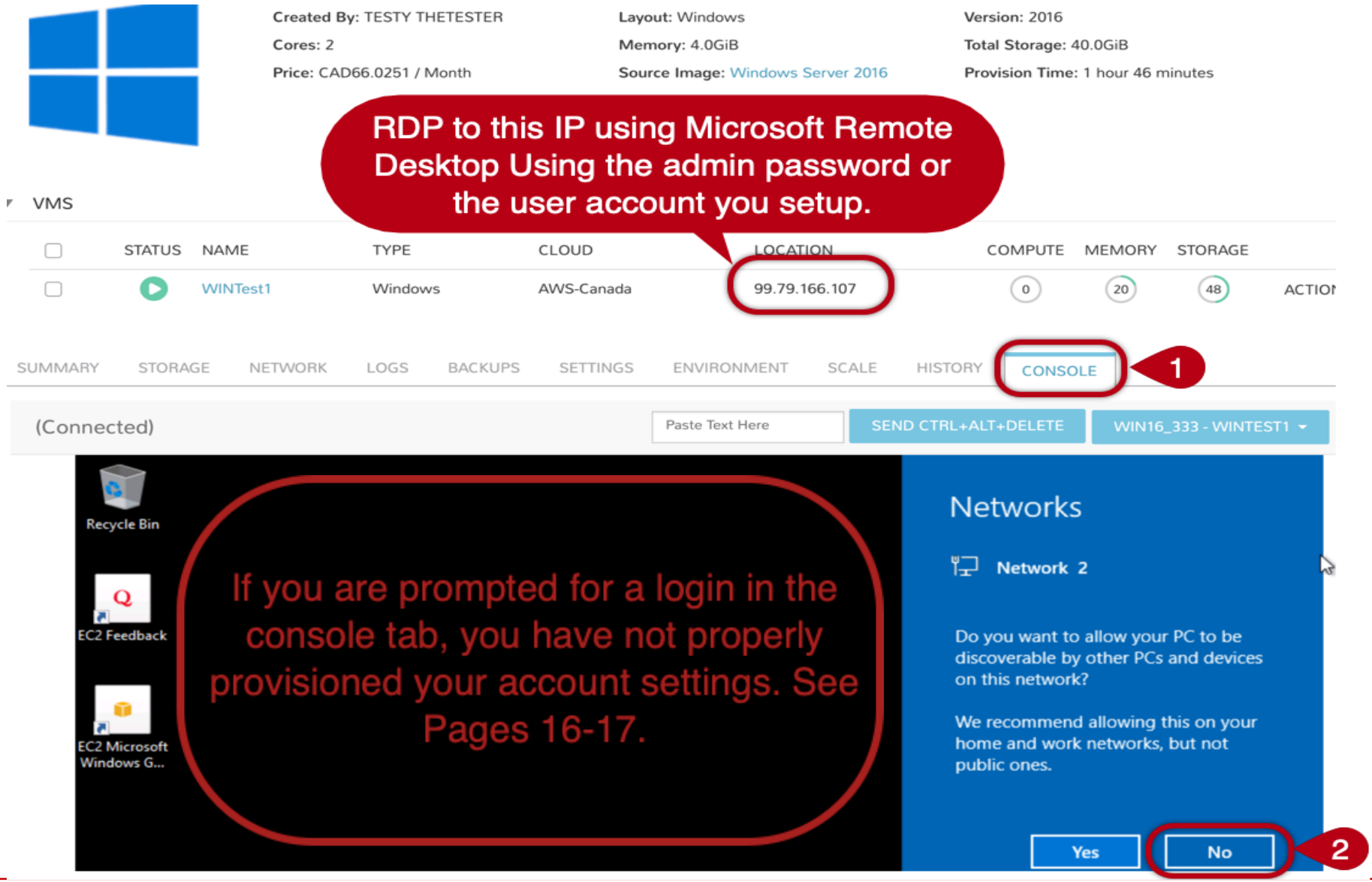
> Review and Complete



Creating a Windows Instance:

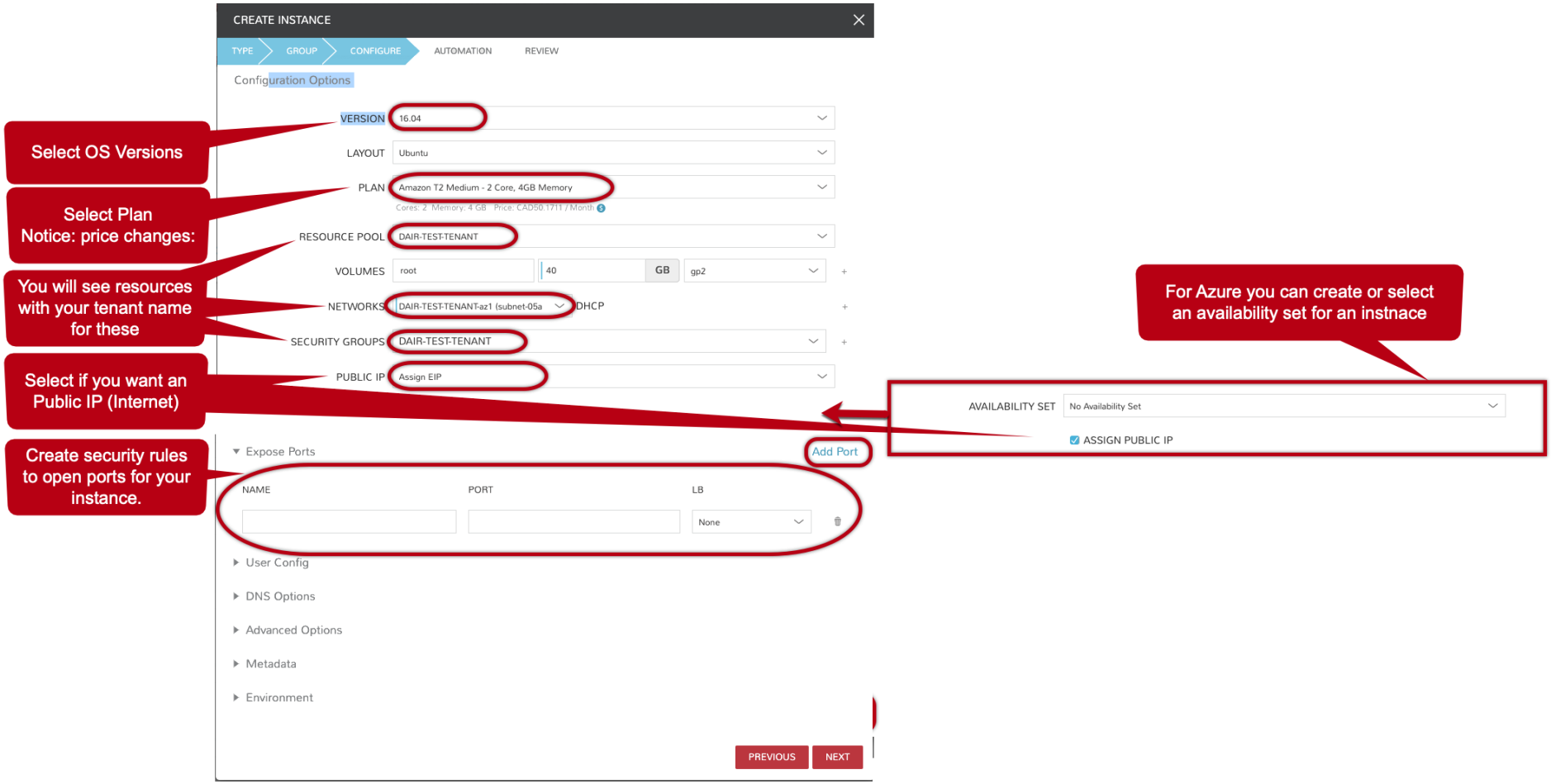


Creating a Windows Instance:



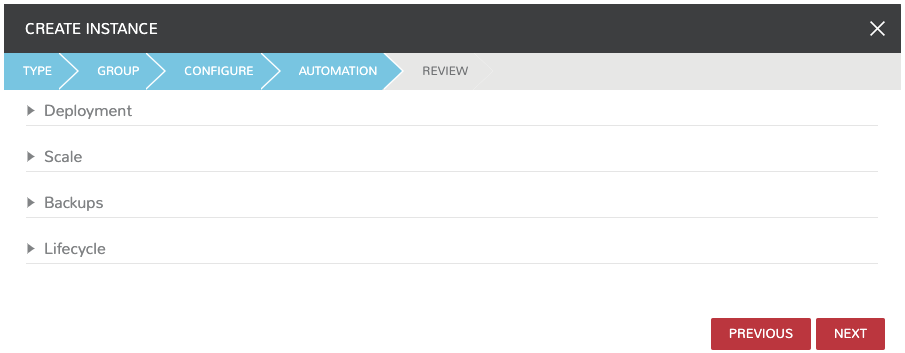
Creating a Linux Instance:

> Provision configuration options for your instance.

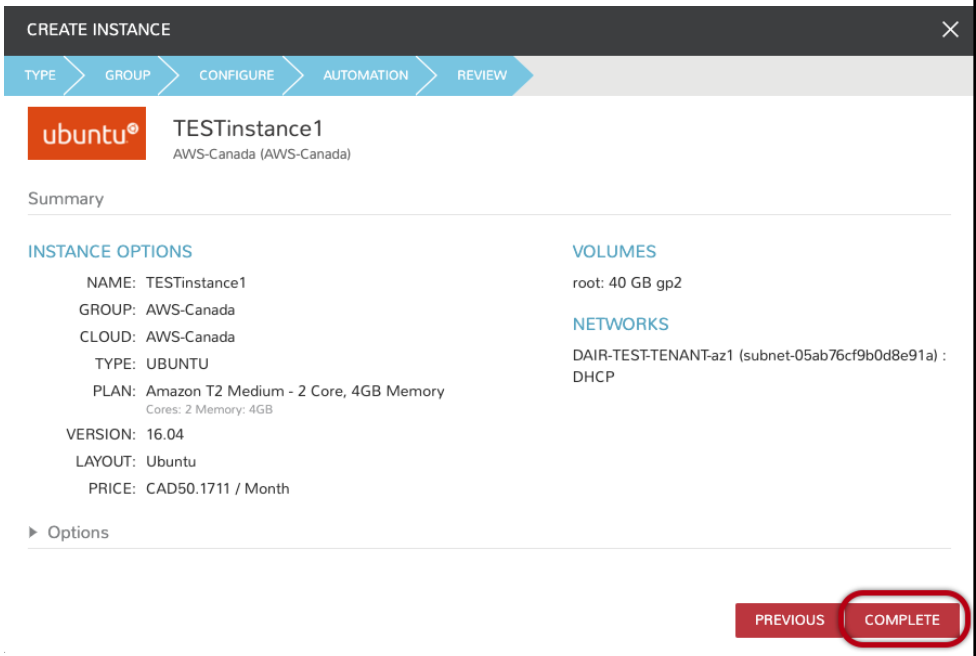


Creating a Linux Instance:

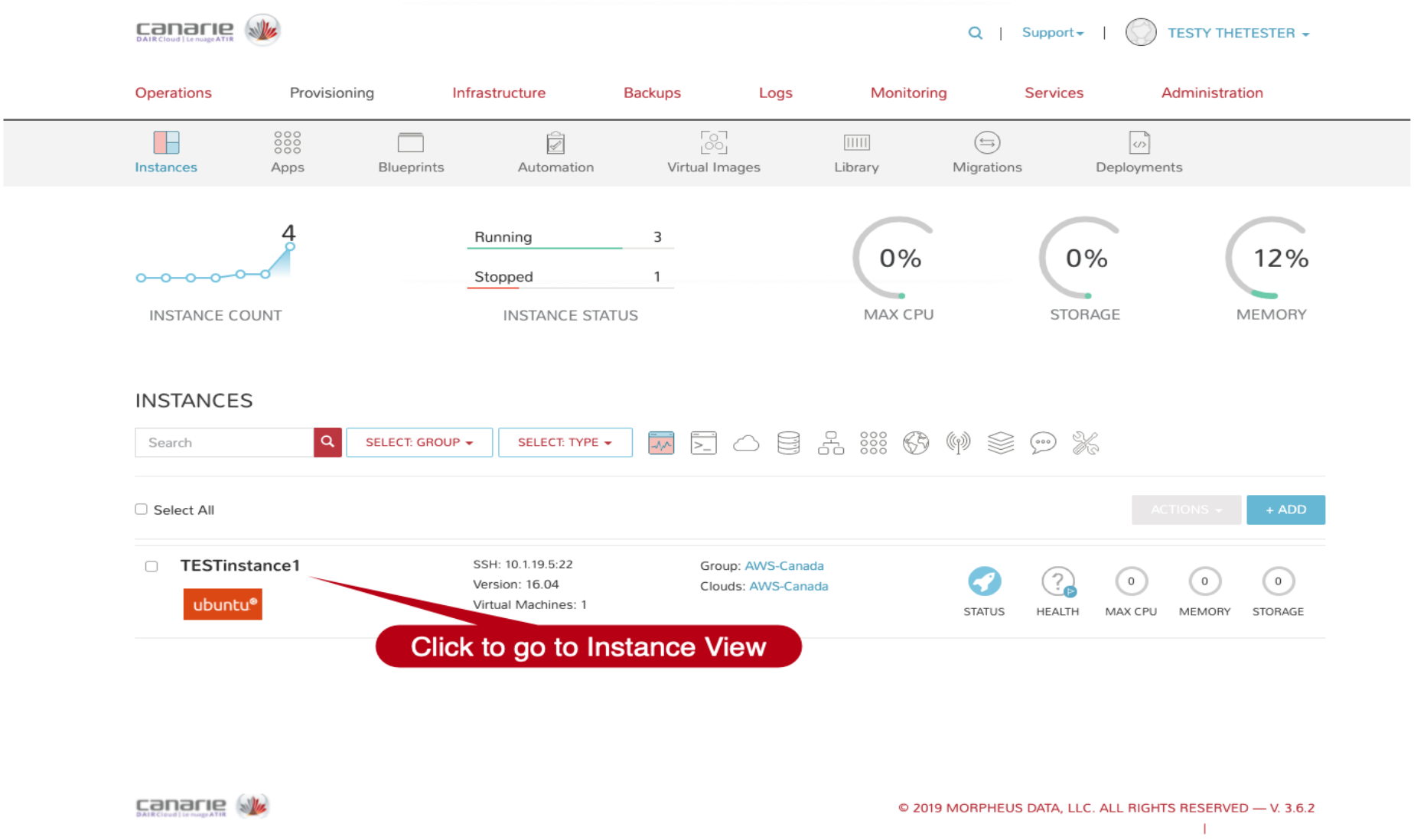
Some advanced configuration options:



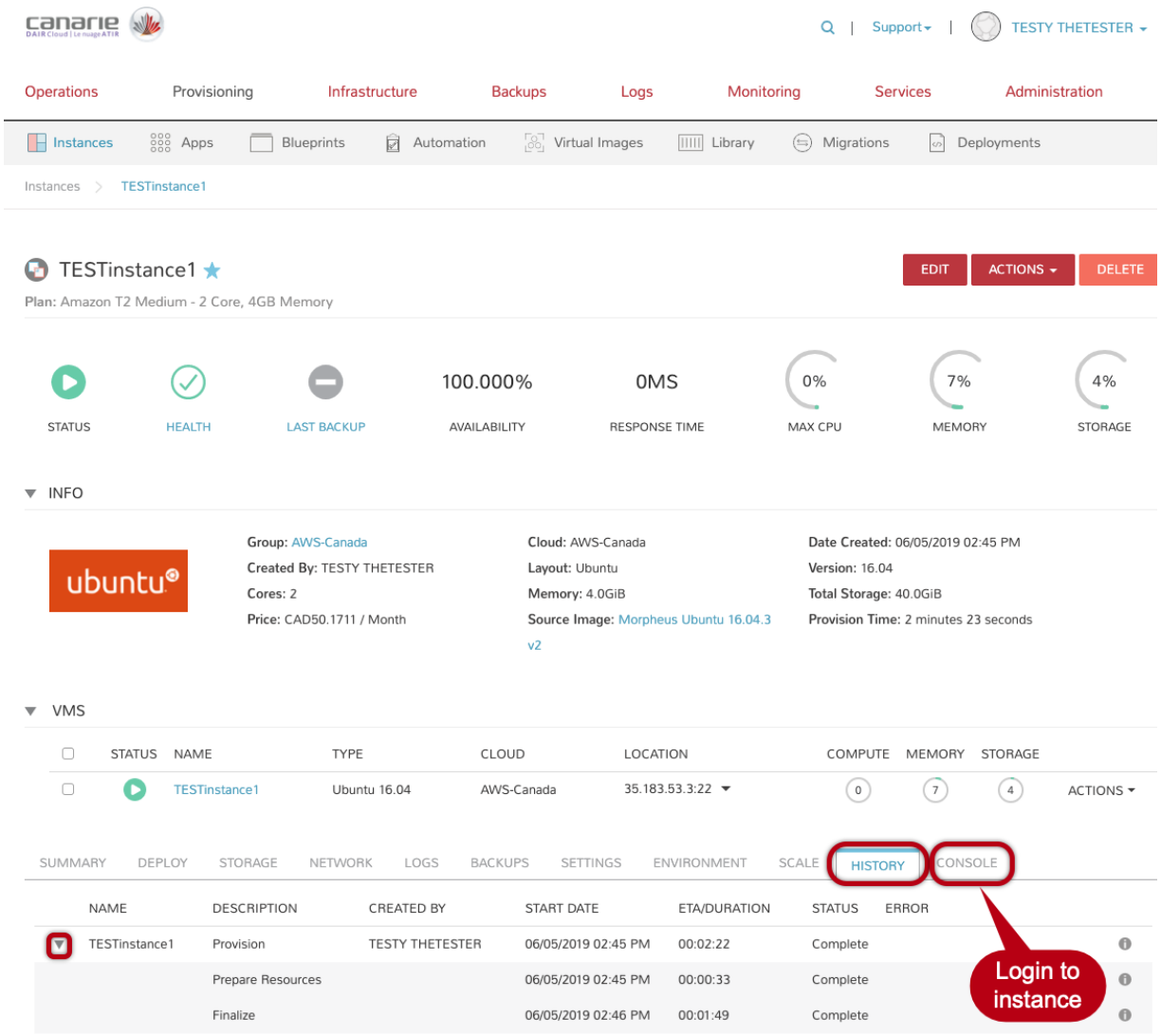
> Review and Complete



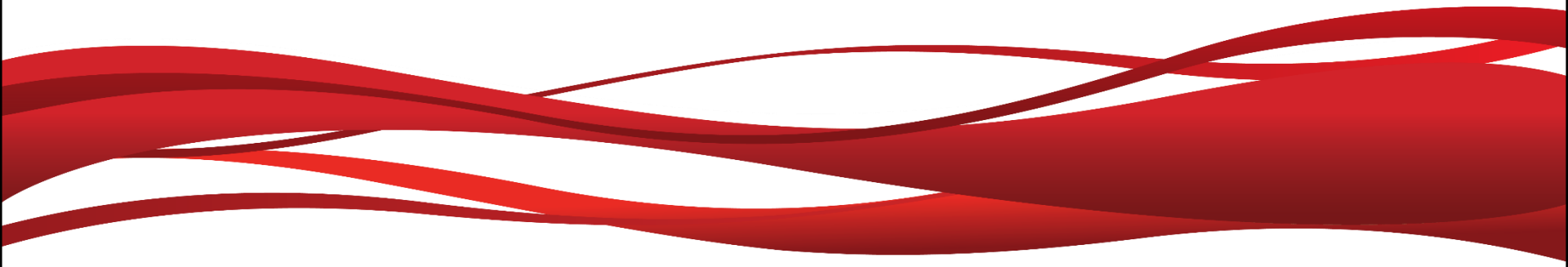
Creating a Linux Instance:



Creating a Linux Instance:



The Basics:
Logging into an Instance

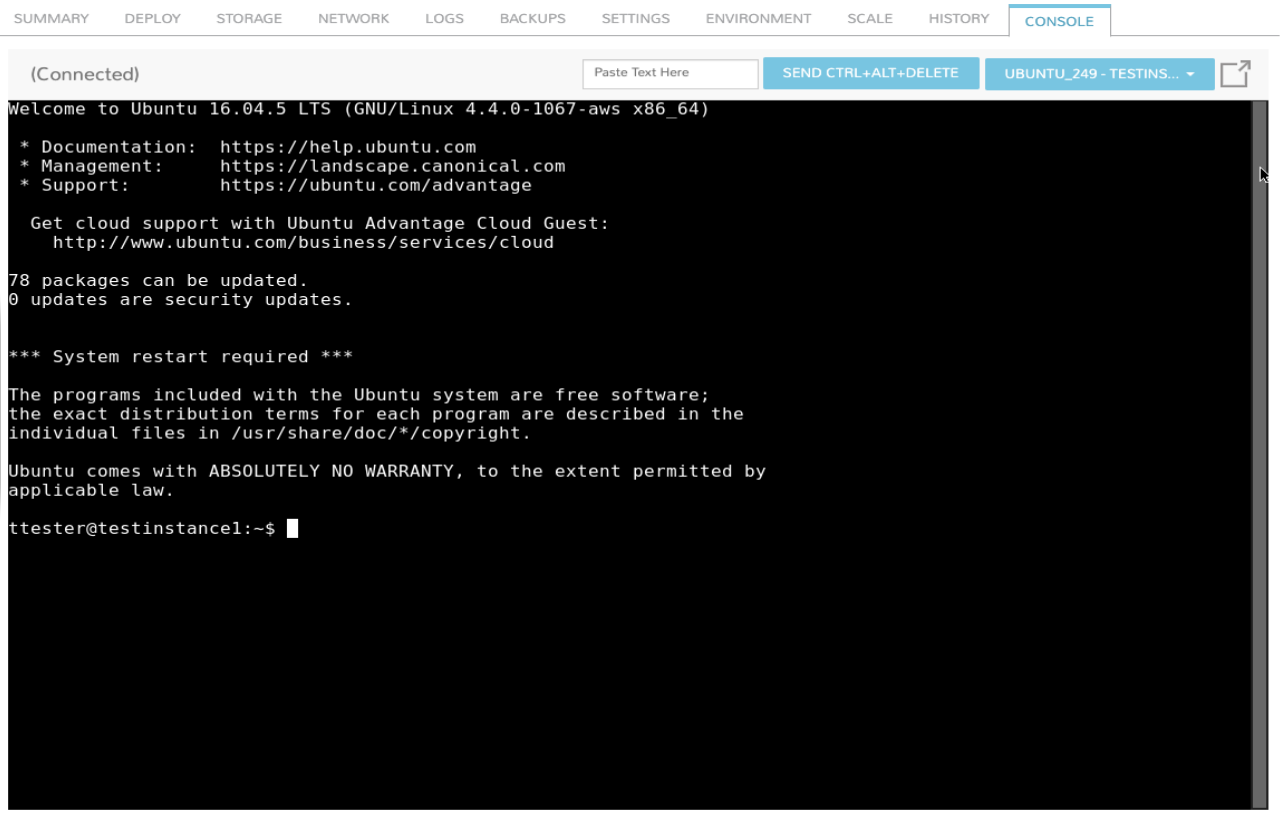


Login to Your Linux Instance – Option 1

> Use the built-in console (web browser)

> You are auto logged in with the account you setup

If you are prompted for a login in the console tab, you have not properly provisioned your account settings. See Pages 16-17.

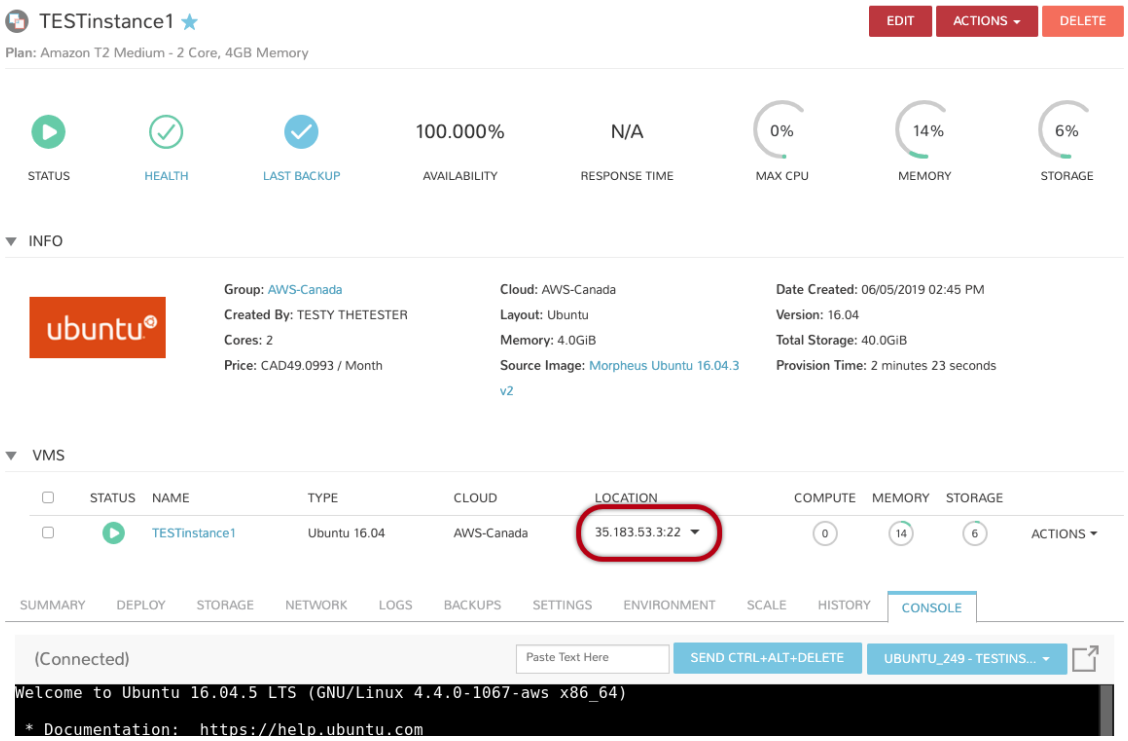


Login to Your Linux Instance – Option 2

- Use your SSH client application
- Mac & Windows (with Linux subsystem)

```
ssh -i /pathtoyourkey/yourkey username@XXX.XXX.XXX.XXX
```

- Your IP is listed under “Location”

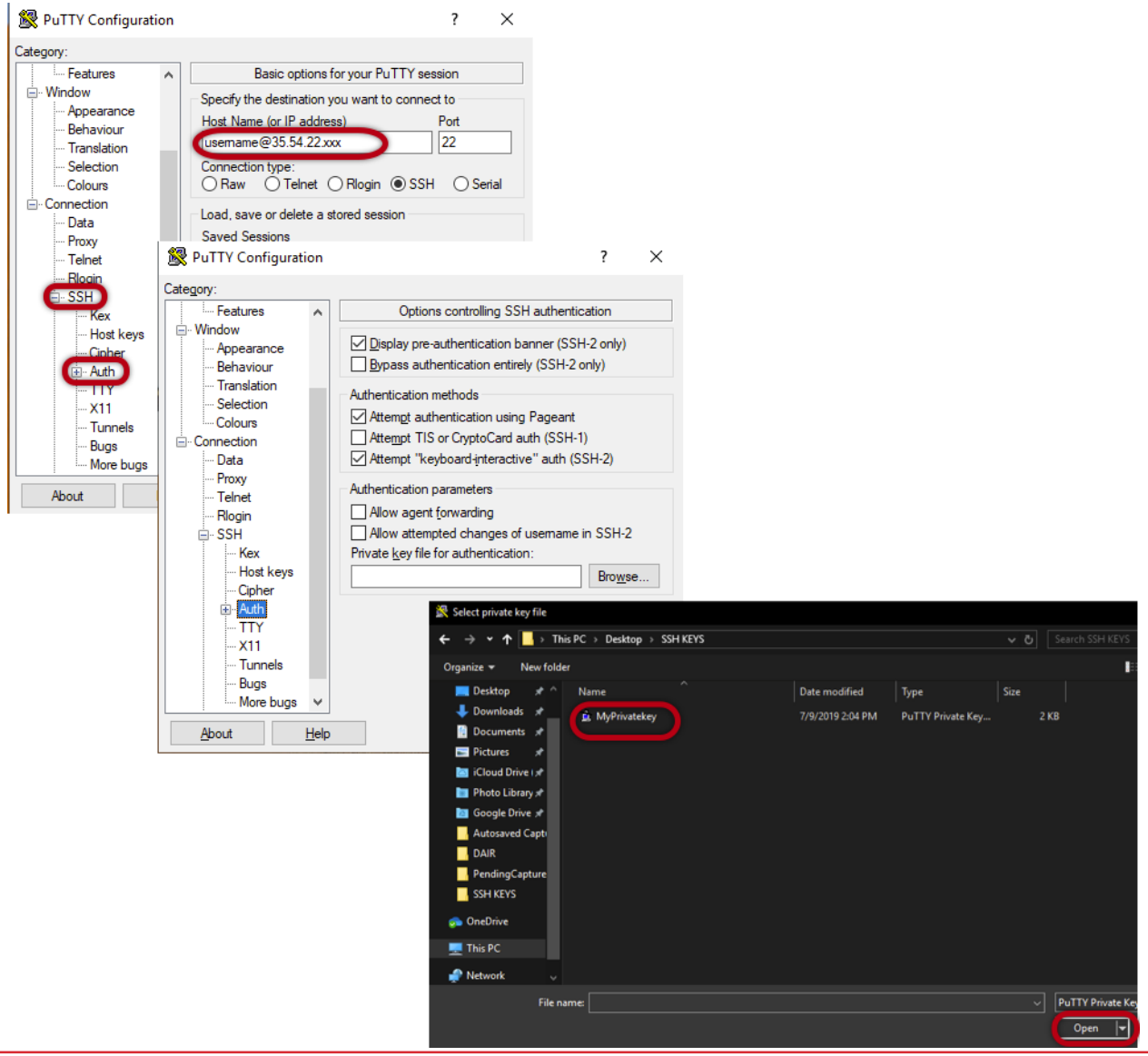


Login to Your Linux Instance – Option 3

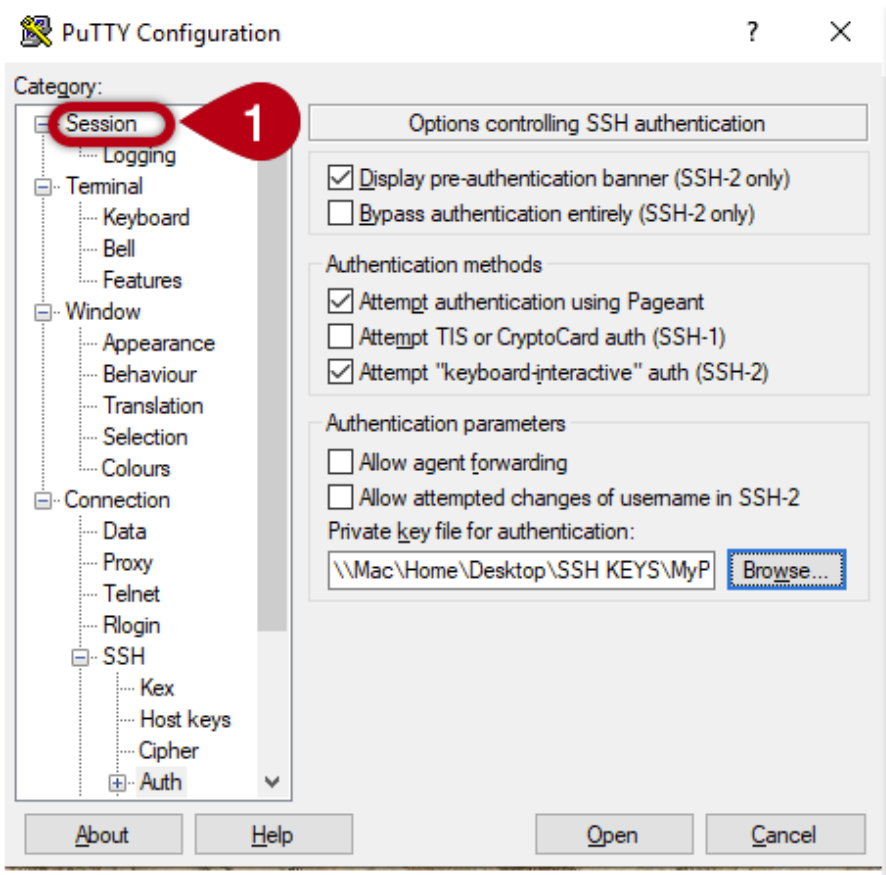
On your Windows machine:

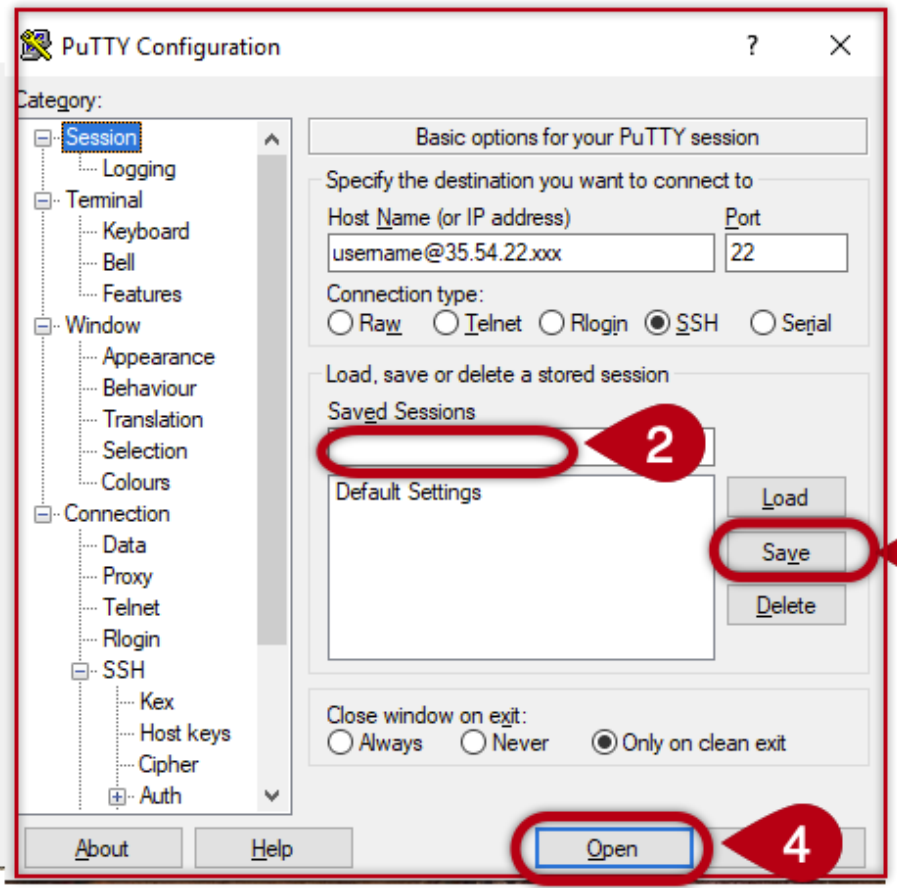
- Locate and run putty.exe (previously installed, see Slide-7)
- Use the IP (see previous slide) and Username you defined in your Morpheus default user settings (see Slide-16) to create a new “putty” session as shown in the following slides.

Login to Your Linux Instance – Option 3



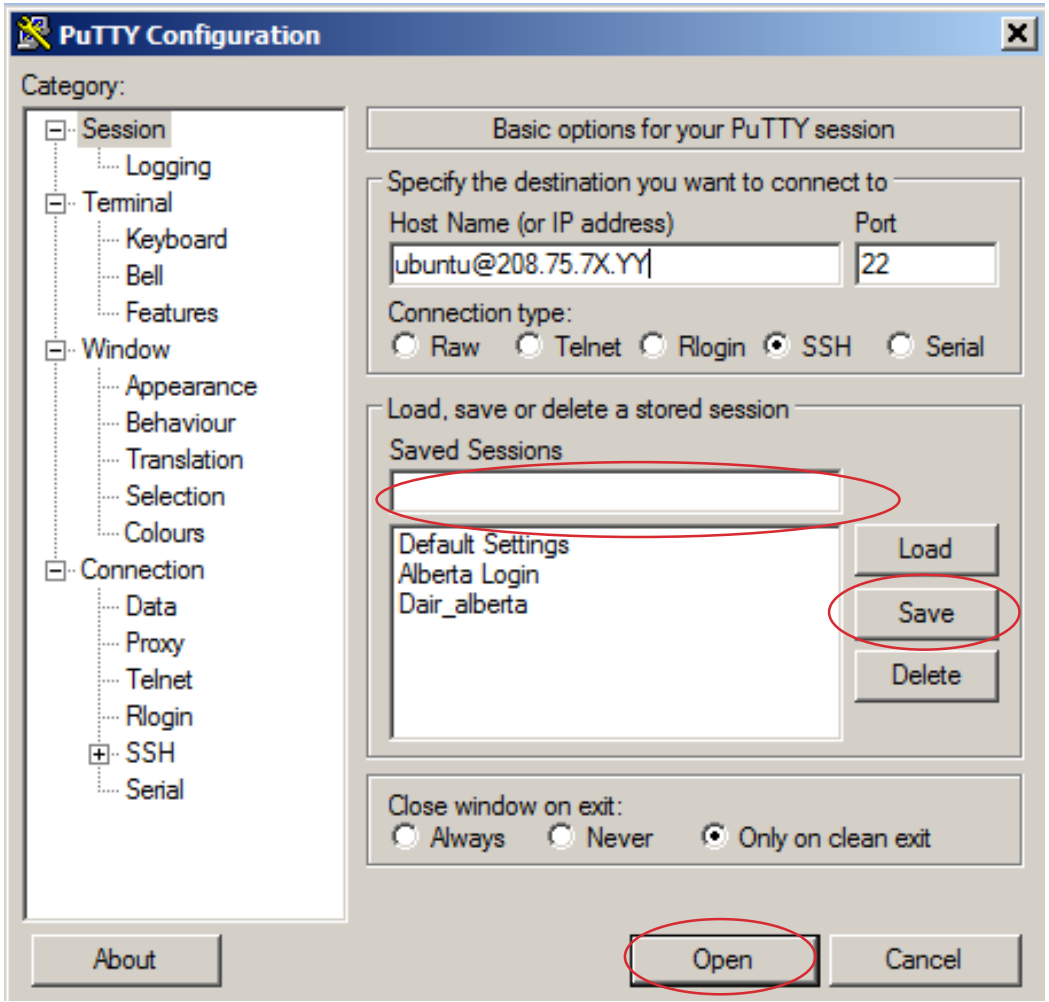
Login to Your Linux Instance – Option 3





Login to Your Linux Instance – Option 3

- Click “Session”
- Enter a “Saved Session” name
- Click “Save”
- Finally “Open” to start an SSH session to your server



The Basics:
Secure VNC to XWindows for Linux GPU Based Instances

Login: VNC to XWindows (MAC or Linux)

- Setup an SSH tunnel to your instance

https://github.com/apenwarr/sshuttle

- OSX process
  - Change uname to your username

```
ssh -i ~/yourkey -p 22 uname@208.75.7x.xxx -L 5901:127.0.0.1:5901 sleep 2h
```

- This terminal will be unusable until you terminate the tunnel

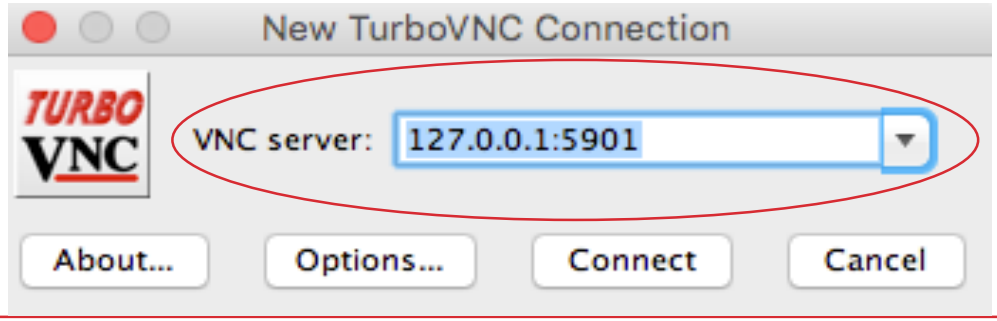
Login: VNC to XWindows (MAC or Linux)

- Use the username and password you set up in the Morpheus account setup
- Install TurboVNC
- Run TurboVNC viewer
- Connect as illustrated

```
ssh-I /{downloadsDirectory}/{Keypair}.pem
Ubuntu@208.75.7x.yy
```

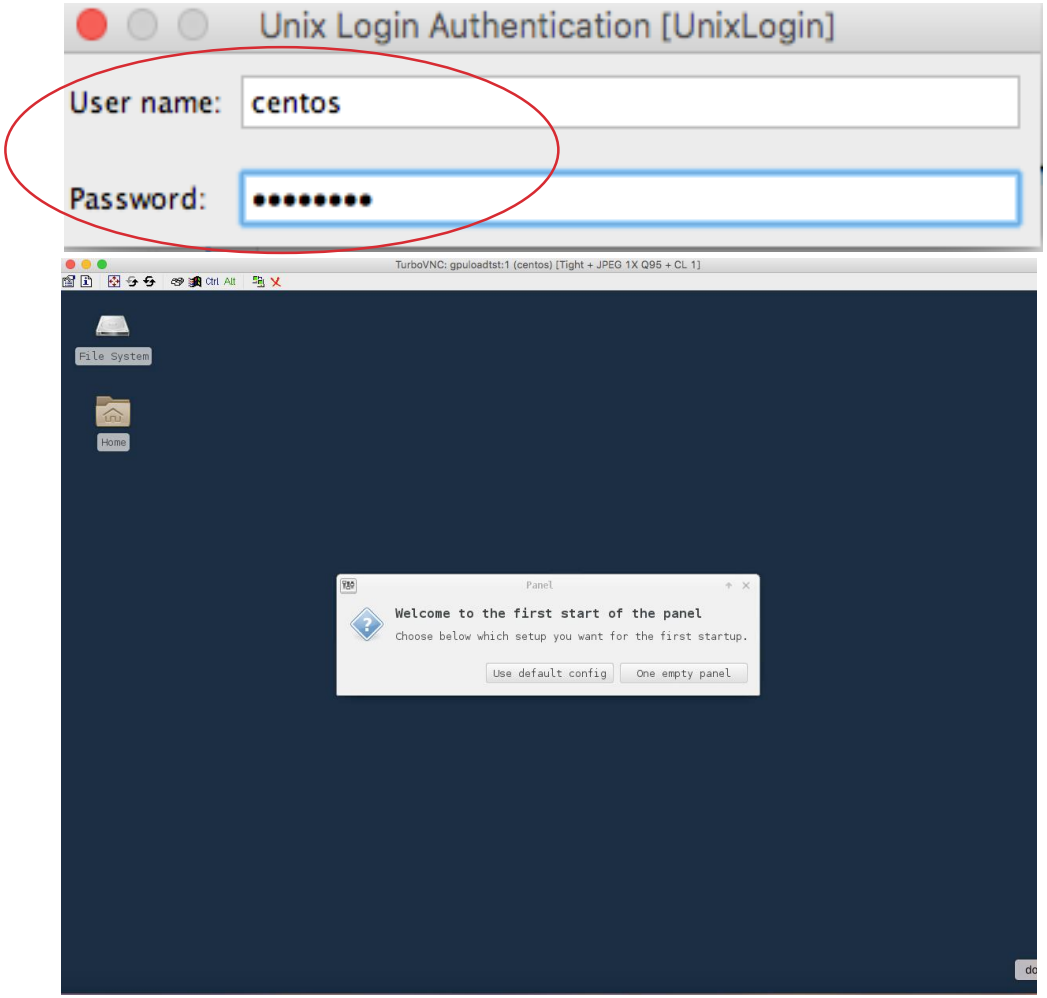
```
sudo passwd ubuntu
New Password:
Re-enter New Password:
```

https://sourceforge.net/projects/turbovnc/files/



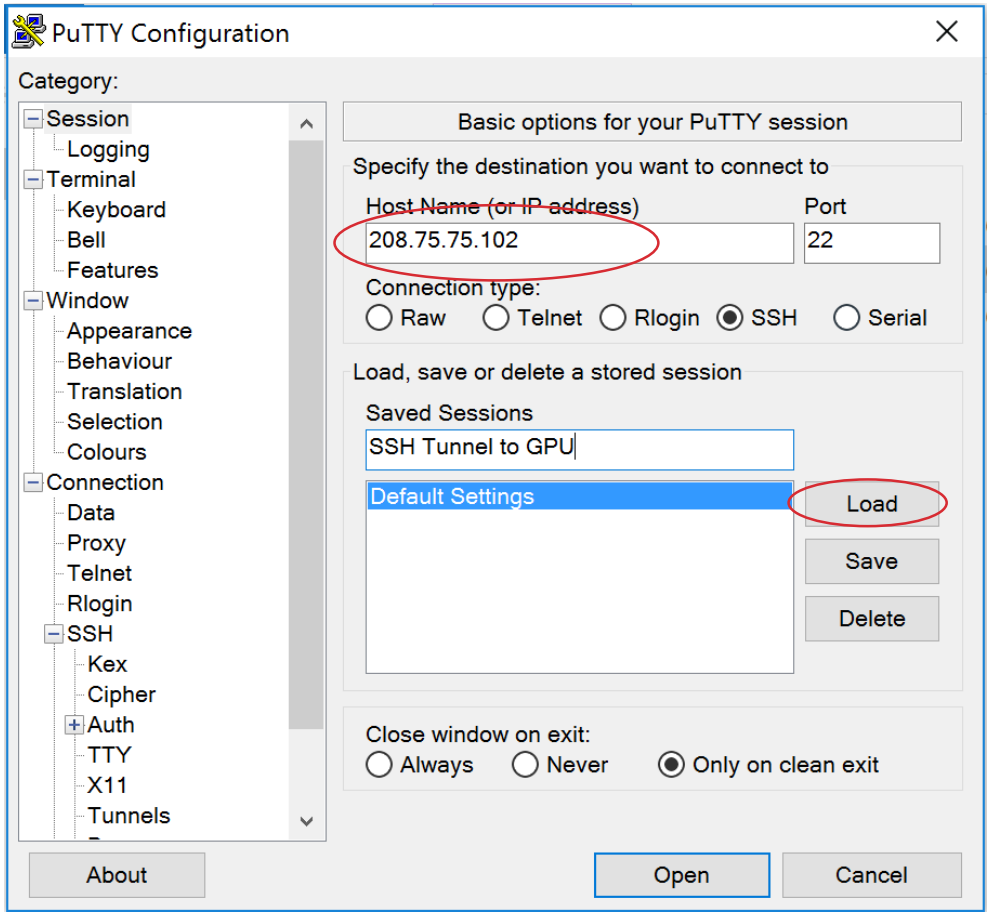
Login: VNC to XWindows (MAC or Linux)

- Login to XWindows
- Enter username and the password you just setup for Ubuntu or CentOS
- You are logged into XWindows
- Select use default config
- You now have access to XWindows and can run GPU GUI applications



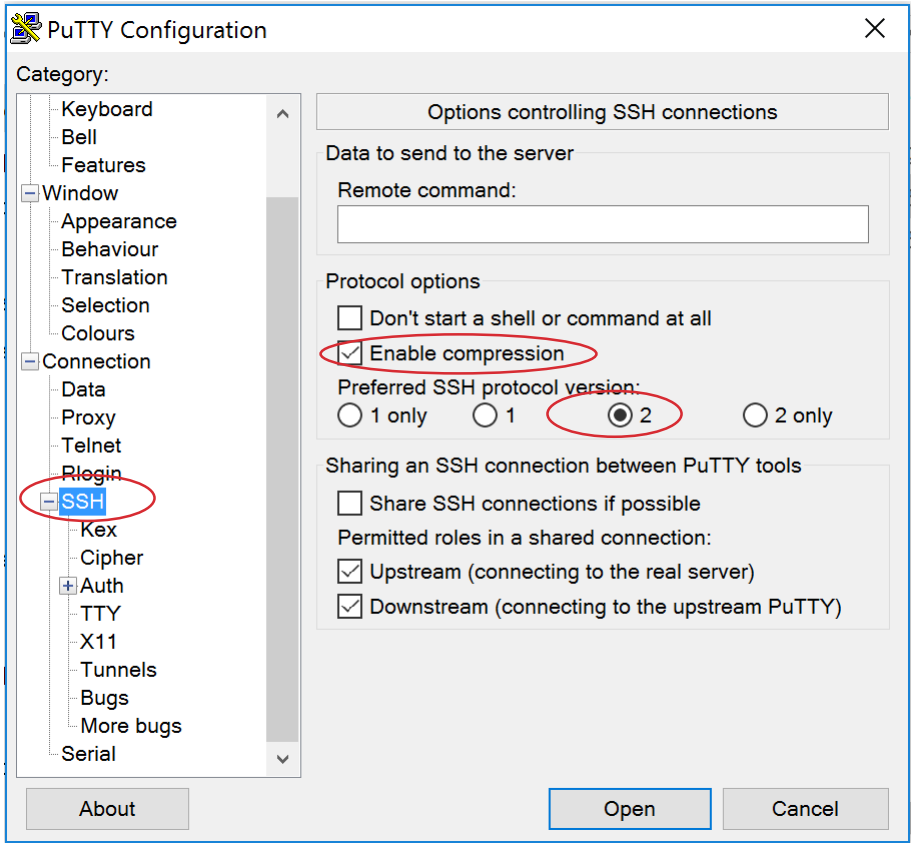
Login: VNC to XWindows (Windows)

- Locate and run putty.exe
  - See Slide-7 if putty is not installed
- Load your previous saved Putty session
- Edit Hostname to match your IP



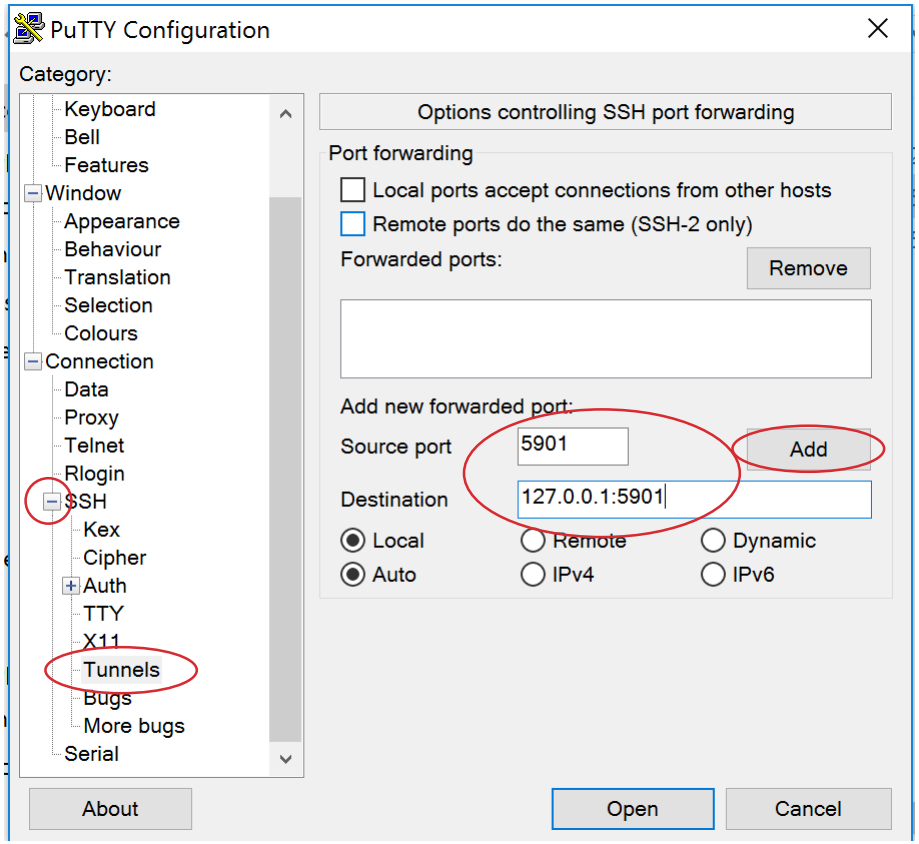
Login: VNC to XWindows (Windows)

- Click “SSH”
- Enable compression
- Set SSH Protocol Version to “2”



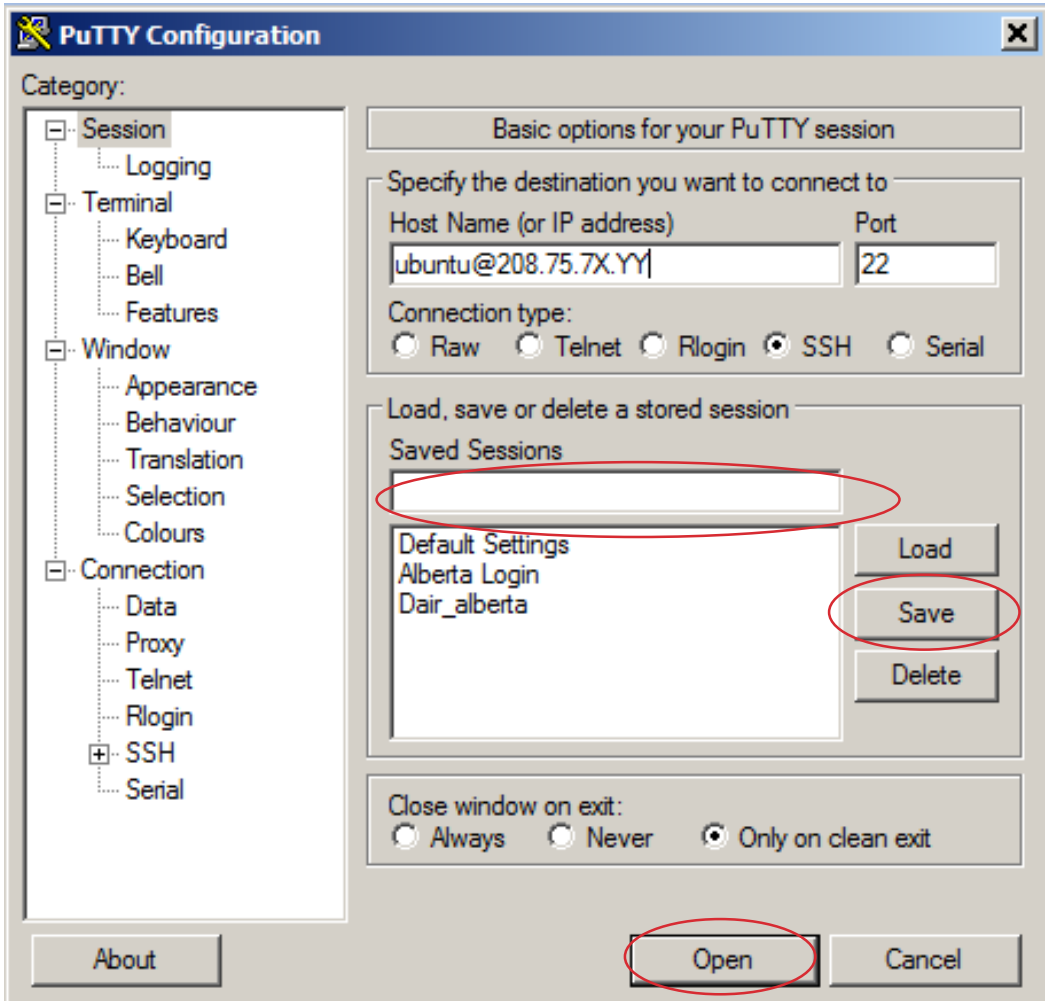
Login: VNC to XWindows (Windows)

- Open “SSH”
- Click “Tunnels”
- Set up new forwarding port as seen here
- Click “Add”



Login: VNC to XWindows (Windows)

- Click “Session”
- Enter a “Saved Session” name
- Click “Save”
- Finally, “Open” to start SSH session with an established tunnel for Secure VNC access



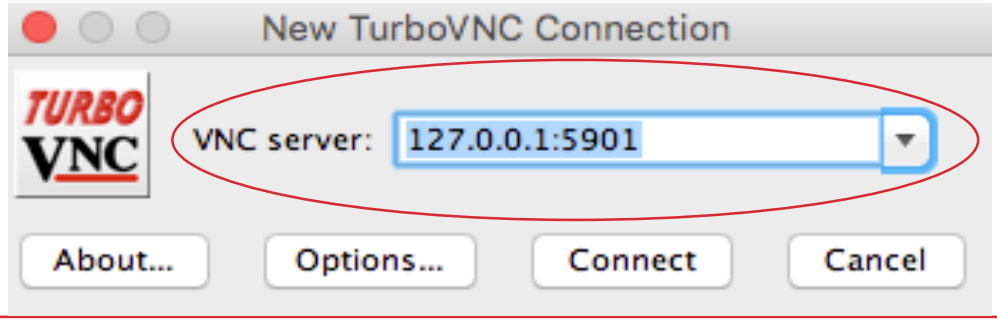
Login: VNC to XWindows (Windows)

- From the putty terminal
- Add password to ubuntu user
  - For CentOS, use “CentOS” instead of “Ubuntu”
- Install TurboVNC
- Run TurboVNC viewer
- Connect as illustrated

```
sudo passwd ubuntu
New Password:
Re-enter New Password:
```

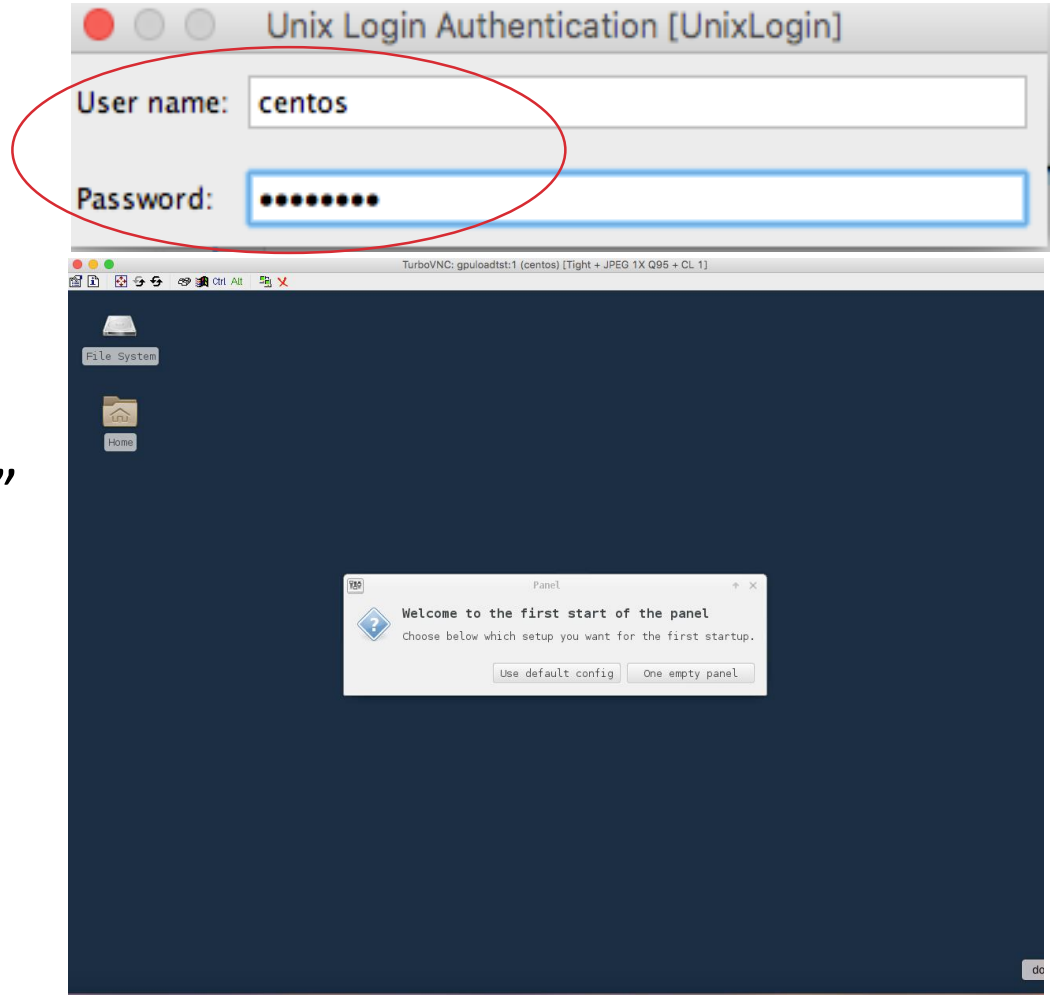
Turbo VNC download location:

https://sourceforge.net/projects/turbovnc/files/



Login: VNC to XWindows (Windows)

- Login to XWindows
- Enter username and the password you just setup for Ubuntu or CentOS
- You are logged into XWindows.
- Select “Use default config”
- You now have access to XWindows and can run GPU GUI applications.



The Basics:
Creating User Accounts

Create a User Account on Your Linux VM

Connect to your Linux instance using SSH:
https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstancesLinux.html

Use the adduser command to add a new user account to an EC2 instance (replace new_user with the new account name). The following example creates an associated group, home directory, and an entry in the */etc/*passwd file of the instance:

```
$ sudo adduser new_user
```

Note: If you add the new_user to an Ubuntu instance, include the_—disabled-password_option to avoid adding a password to the new account:

```
$ sudo adduser —disabled-password new_user
```

Change the security context to the new_user account so that folders and files you create will have the correct permissions:

```
$ sudo su - new_user
```

Note: When you run the sudo su - new_user command, the name at the top of the command shell prompt changes to reflect the new user account context of your shell session.

Create a .ssh directory in the new_user home directory:

```
$ mkdir .ssh
```

Use the chmod command to change the .ssh directory's permissions to 700. Changing the permissions restricts access so that only the new_user can read, write, or open the .ssh directory.

```
$ chmod 700 .ssh
```

Use the touch command to create the authorized_keys file in the .ssh directory:

```
$ touch .ssh/authorized_keys
```

Use the chmod command to change the .ssh/authorized_keys file permissions to 600. Changing the file permissions restricts read or write access to the new_user.

```
$ chmod 600 .ssh/authorized_keys
```

open their public key in a text editor and edit with vi or other Linux text edit and insert the private key then save the file.

```
$ vi .ssh/authorized_keys
```

Advanced Topics:
Morpheus Command Line

Using Morpheus CLI Commands

- Anything you perform through the Dashboard may be performed through CLI commands or the Morpheus API
- Documentation can be found here:
  - CLI:
    - https://docs.morpheusdata.com/en/3.6.2/cli/gettingStarted.html
  - API:
    - https://docs.morpheusdata.com/en/3.6.2/api/intro.html

canarie

canarie.ca | @canarie_inc