Task 2 - Cloud Server

Cloud Server using Microsoft Azure

Student: Philipp Moritzer
Student ID: 21170004

Module: Unix System
Professor: 이길흥
Due Date: 6th June 2021
Date of assignment: 11th May 2021

Task: The task of the following summary consists of selecting and registering to a free-tier Service of either Google Cloud Platform, Windows Azure or Amazon Webservice. Using this platform the goal is to install a linux operating system, access the system and use it. While using the system there should be done some practice with commands learnt at the class. (For example: Shell Programming, Java Programming, C Programming)

Microsoft Azure

Microsoft Azure is a Cloud Computing Service provided by Microsoft. It provides Software as a Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS). The Applications, Platforms and Infrastructure is run on Microsoft managed data centers. In this special occasion Microsoft Azure is used to build a Cloud Linux Server running Ubuntu. The goal of this project is to create an accessible Linux (Ubuntu) Virtual Machine. The Virtual Machine is supposed to be accessed by Telnet or SSH. After creating the Machine it should be usable and there will be some minor practice done with commands learnt at the class.

Setting up the Virtual Machine

To setup the Virtual Machine, the website https://portal.azure.com/ is accessed and a free account created. The next step is to create the Virtual Machine using the Wizard provided by the platform (see Figure 1 and Figure 2).

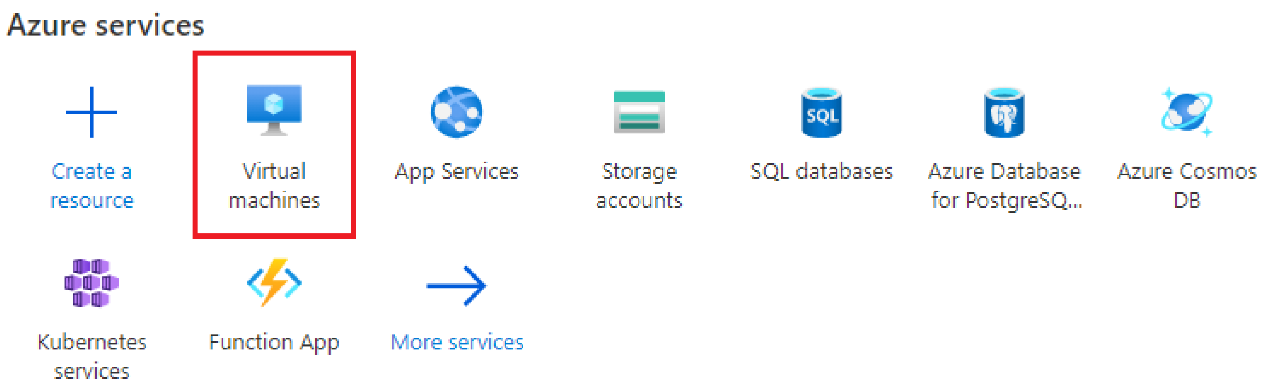


Figure 1: Main Page of the Azure Portal

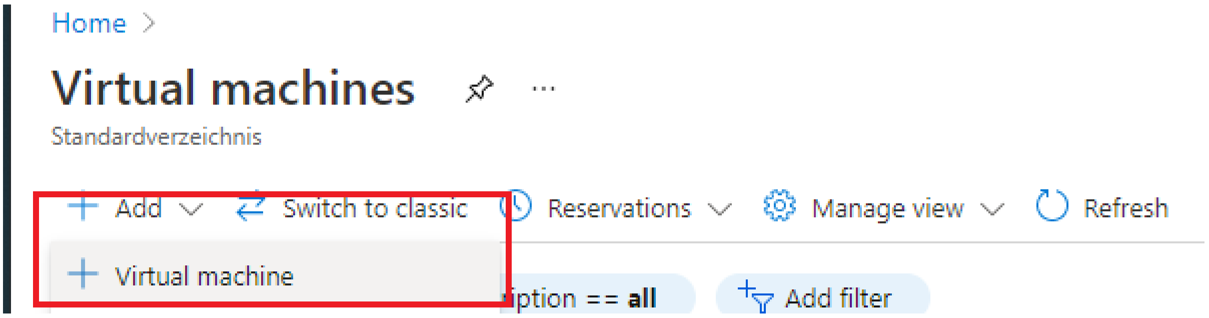
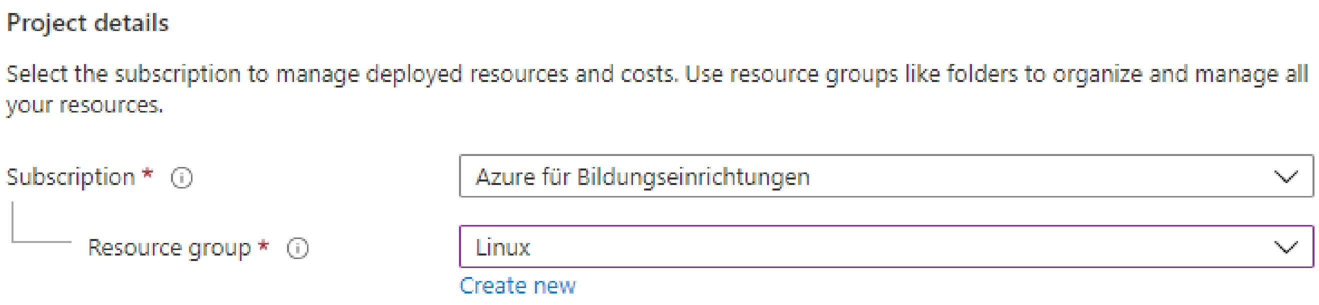


Figure 2: Add Virtual Machine in the VM Overview Page

After initalizing the Setup Wizard the Virtual Machine's properties have to be set.

Project Details

The section Project Details defines the cost and resource management of the Virtual Machine.



Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

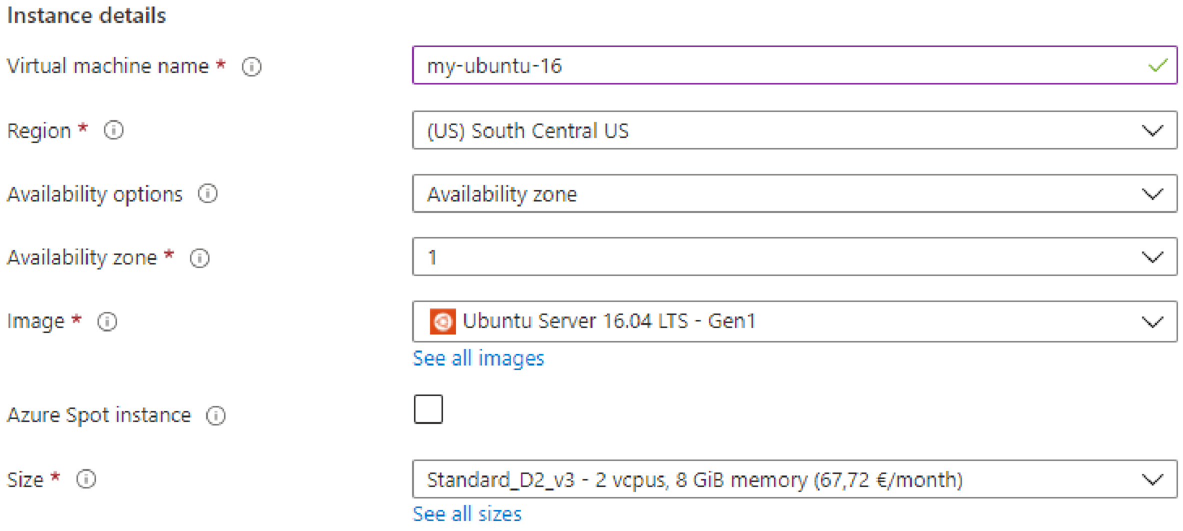
Subscription *	Azure für Bildungseinrichtungen
Resource group *	Linux Create new

Figure 3: Setting up Project Details

- Subscription: Azure for Educational Institutions
    - Defines that we use the free Azure Service provided for Educational Institutions.
- Resource Group: Linux
    - Assign a group to manage multiple Virtual Machines in one's Account.

Instance Details

In the instance details the properties of this certain Virtual Machine will be set.



Instance details

Virtual machine name *	my-ubuntu-16 ✓
Region *	(US) South Central US
Availability options	Availability zone
Availability zone *	1
Image *	Ubuntu Server 16.04 LTS - Gen1 See all images
Azure Spot instance	□
Size *	Standard_D2_v3 - 2 vcpus, 8 GiB memory (67,72 €/month) See all sizes

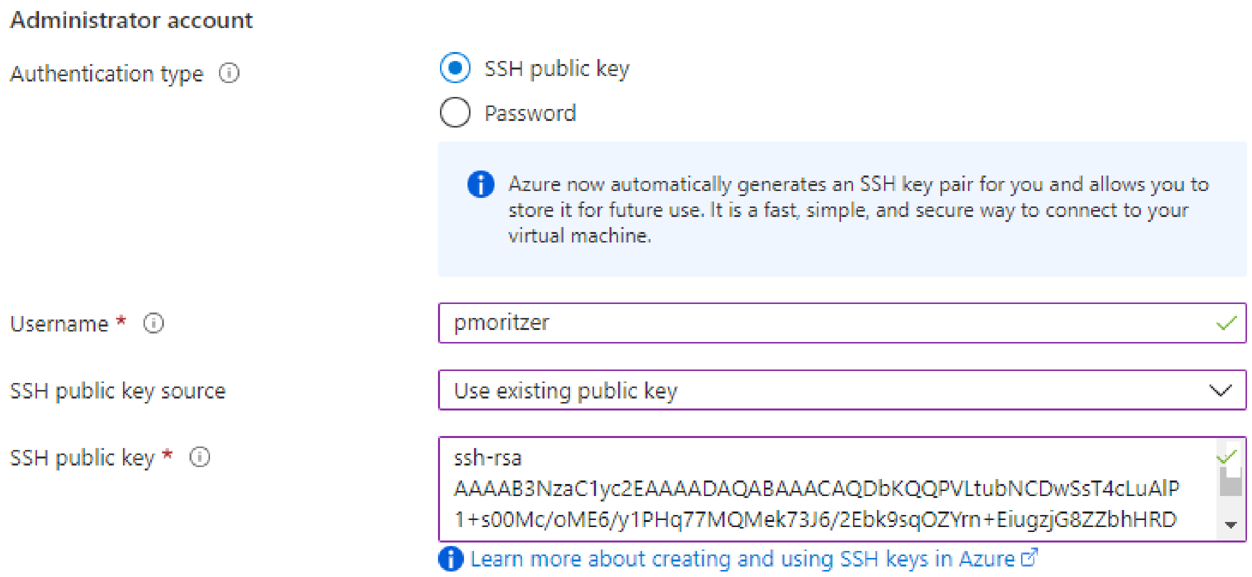
Figure 4: Setting up Instance Details

- Virtual Machine Name: my-ubuntu-16
    - The name of the Virtual Machine in the Azure Portal.
- Region: (US) South Central US
    - The region in which the Server will be located. It will be left on South Central US (default).
- Availability Options: Availability zone
    - Availability zone options means that the resources will be physically separated within an Azure region.
- Availability Zone: 1
    - Set to default since it is not important in this step.
- Image: Ubuntu 16.04 LTS - Gen1

  - Defines that the Vitual Machine will run Ubuntu 16.04 Latest Stable Relase. If a Windows Virtual Machine is needed it is also possible to choose this here or any other Version of a listed Operating System.
- Azure Spot instance: Not selected
  - Makes the user pay for workload instead of time. It is not needed here since the Virtual Machine will be set on a free account.
- Size: Standard_D2_v3 - 2 vcpus, 8 GiB memory (67,72 €/month)
  - There will be 2 virtual CPUs and 8 Gigabytes of Memory assigned to this Virtual Machine. More Power is more expensive but the free budget is used here so it is not changed.

Administrator Account

In this section it is determined how to access the server.



Administrator account

Authentication type ⓘ ● SSH public key ○ Password

Azure now automatically generates an SSH key pair for you and allows you to store it for future use. It is a fast, simple, and secure way to connect to your virtual machine.

Username * ⓘ: pmoritzer ✓

SSH public key source: Use existing public key

SSH public key * ⓘ:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAACAQDbKQQPVLtubNCDwSsT4cLuAlP
1+s00Mc/oME6/y1PHq77MQMek73J6/2Ebk9sqOZYrn+EiugzjG8ZZbhHRD
```

Figure 5: Setting up Administrator account

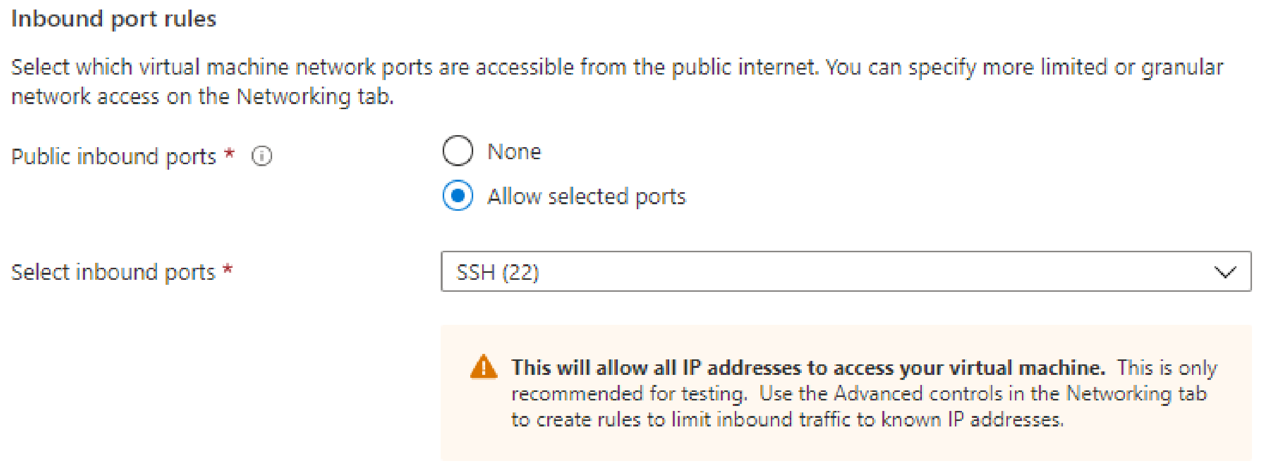
- Username: pmoritzer
  - The username to access the server, using student's id
- SSH public key source: Use existing public key
  - Generating a Public Key using the ssh-gen Windows Command-Line Tool
- SSH public key: <ssh key>

To generate the Public SSH-Key following command has to be used. The result will be safed to a id_rsa.pub-File which content has to be inserted into the SSH public key field.

```
$ ssh-keygen -m PEM -t rsa -b 4096
```

Inbound Port Rules

Inbound Port rules decide on which port the virtual machine can be accessed. Here only the SSH Standard Port 22 will be used.



Inbound port rules

Select which virtual machine network ports are accessible from the public internet. You can specify more limited or granular network access on the Networking tab.

Public inbound ports * ⓘ ○ None ● Allow selected ports

Select inbound ports * SSH (22)

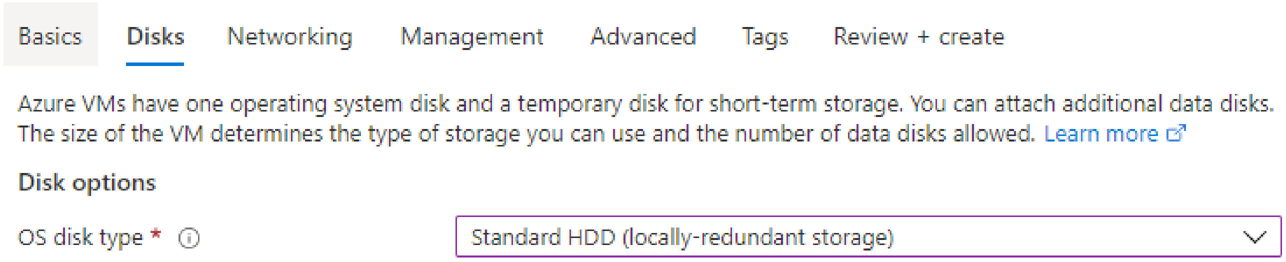
⚠ **This will allow all IP addresses to access your virtual machine.** This is only recommended for testing. Use the Advanced controls in the Networking tab to create rules to limit inbound traffic to known IP addresses.

Figure 6: Setting up Inbound Port Rules

- Public inbound ports: Allow selected ports
  - Only selected ports are allowed
- Select inbound ports: SSH(22)
  - Only the standard SSH Port (22) is allowed to access the server

Disks - Disk Type

The disk type will be set to Standard HDD to save running costs:



Azure VMs have one operating system disk and a temporary disk for short-term storage. You can attach additional data disks. The size of the VM determines the type of storage you can use and the number of data disks allowed. Learn more

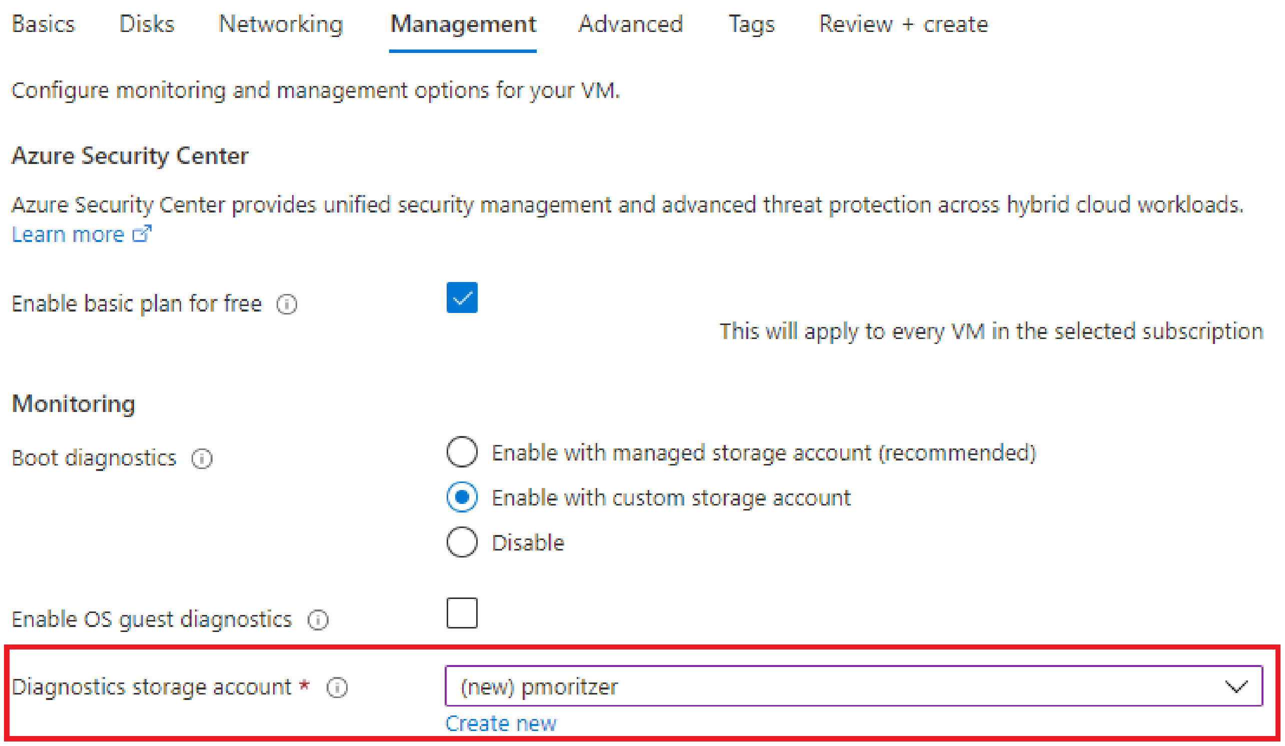
Disk options

OS disk type * ⓘ Standard HDD (locally-redundant storage)

Figure 7: Setting up Disks - Disk Type

Management - Monitoring

In the management Tab a monitoring account will be set up. The name is the same as the user id:



Configure monitoring and management options for your VM.

Azure Security Center

Azure Security Center provides unified security management and advanced threat protection across hybrid cloud workloads.
Learn more

Enable basic plan for free ⓘ ☑

This will apply to every VM in the selected subscription

Monitoring

Boot diagnostics ⓘ
○ Enable with managed storage account (recommended)
● Enable with custom storage account
○ Disable

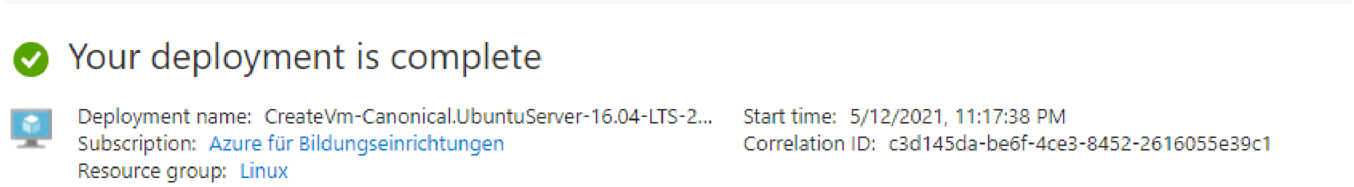
Enable OS guest diagnostics ⓘ □

Diagnostics storage account * ⓘ (new) pmoritzer
Create new

Figure 8: Setting up Management Account

Finishing the Setup

After Validating a click on "Review + create" creates the Virtual Machine and it is ready to be accessed.



Your deployment is complete

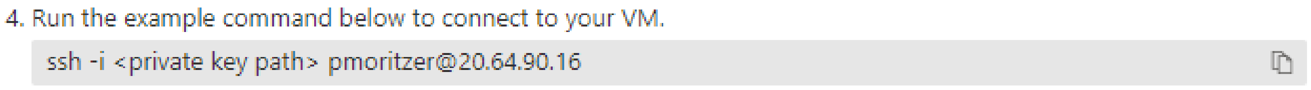
Deployment name: CreateVm-Canonical.UbuntuServer-16.04-LTS-2...
Subscription: Azure für Bildungseinrichtungen
Resource group: Linux

Start time: 5/12/2021, 11:17:38 PM
Correlation ID: c3d145da-be6f-4ce3-8452-2616055e39c1

Figure 9: Finishing the Setup

Accessing the Virtual Machine

To connect to the Virtual Machine the Azure Portal provides connection instructions. This time how to connect with SSH:

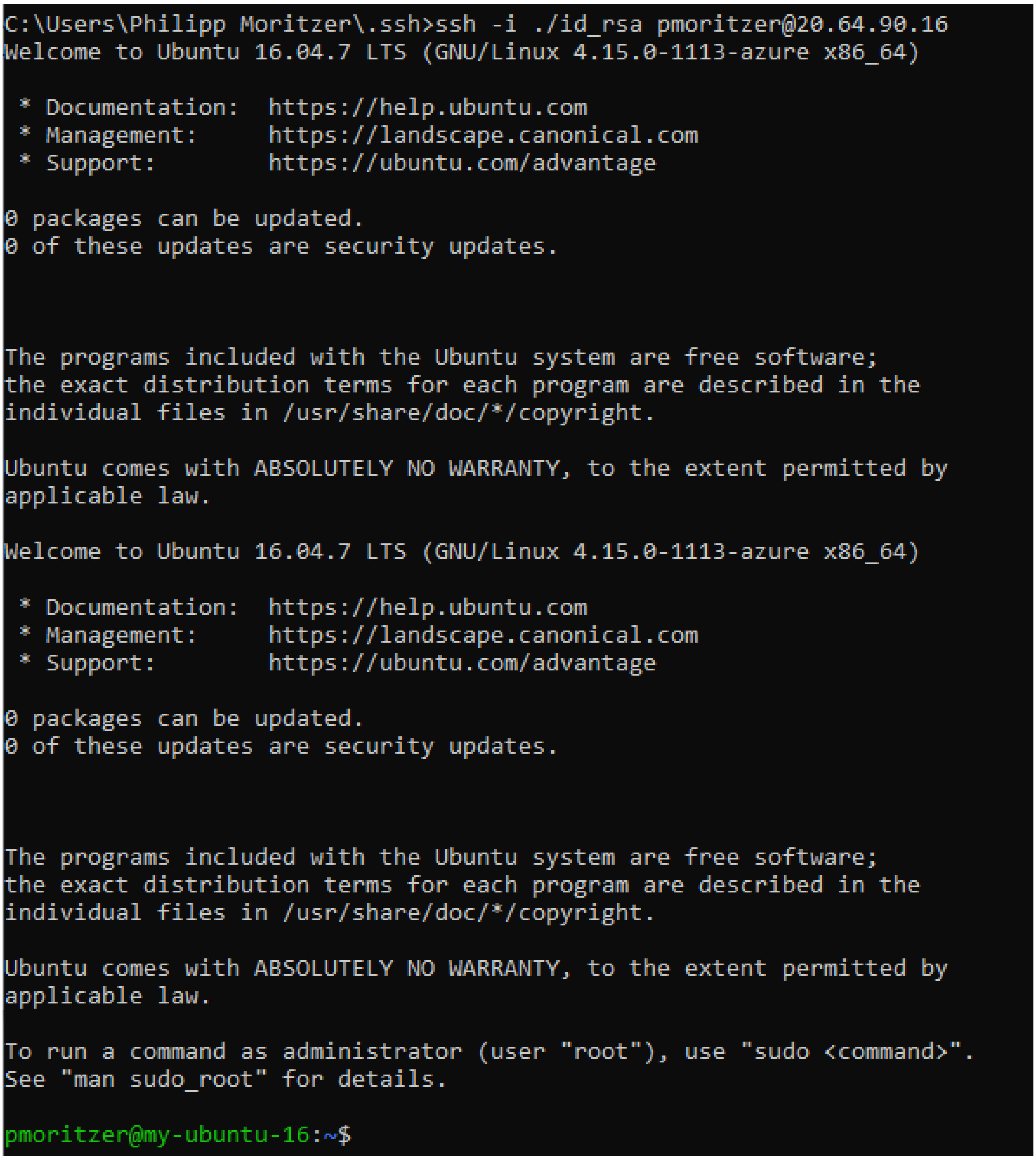


4. Run the example command below to connect to your VM.

```
ssh -i <private key path> pmoritzer@20.64.90.16
```

Figure 10: SSH Connection instruction

The connection using this command will be successful by providing the private key for the connection:



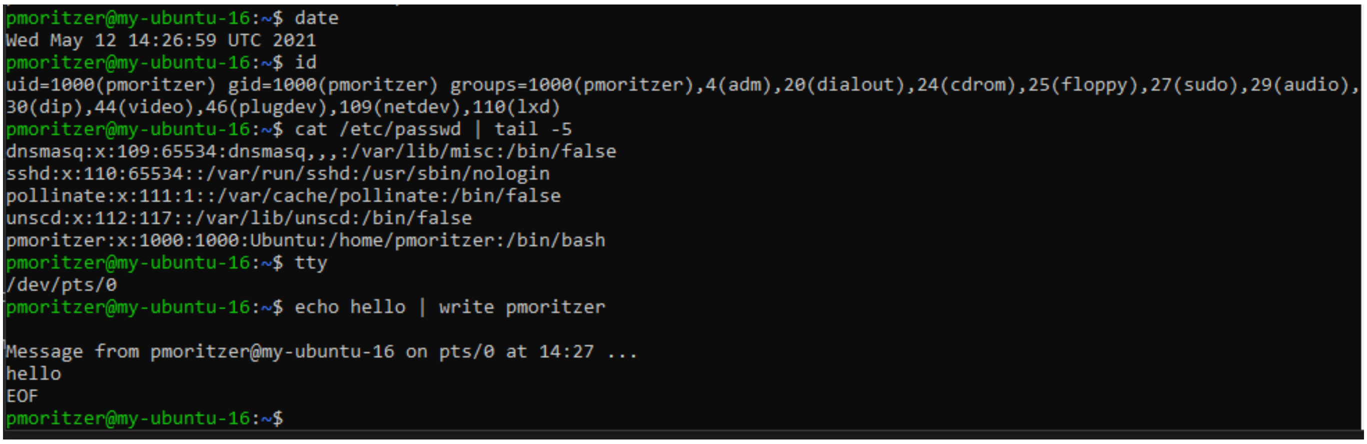
```
C:\Users\Philipp Moritzer\.ssh>ssh -i ./id_rsa pmoritzer@20.64.90.16
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.15.0-1113-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 of these updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.15.0-1113-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 of these updates are security updates.



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

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

pmoritzer@my-ubuntu-16:~$
```

Figure 11: Establishing an SSH Connection

Using the Virtual Machine

Basic Commands

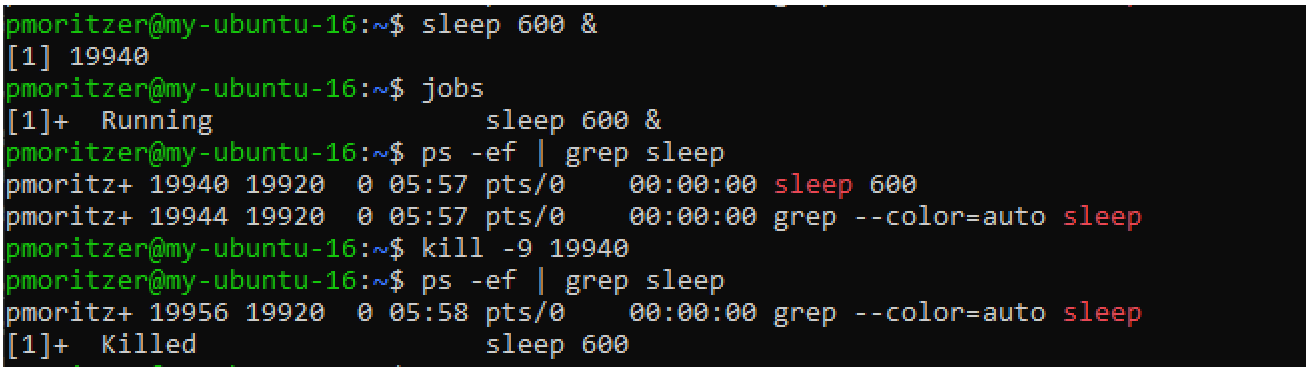
Using simple Unix commands to show that the server is working:



```
pmoritzer@my-ubuntu-16:~$ date
Wed May 12 14:26:59 UTC 2021
pmoritzer@my-ubuntu-16:~$ id
uid=1000(pmoritzer) gid=1000(pmoritzer) groups=1000(pmoritzer),4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),29(audio),
30(dip),44(video),46(plugdev),109(netdev),110(lxd)
pmoritzer@my-ubuntu-16:~$ cat /etc/passwd | tail -5
dnsmasq:x:109:65534:dnsmasq,,,:/var/lib/misc:/bin/false
sshd:x:110:65534::/var/run/sshd:/usr/sbin/nologin
pollinate:x:111:1::/var/cache/pollinate:/bin/false
unscd:x:112:117::/var/lib/unscd:/bin/false
pmoritzer:x:1000:1000:Ubuntu:/home/pmoritzer:/bin/bash
pmoritzer@my-ubuntu-16:~$ tty
/dev/pts/0
pmoritzer@my-ubuntu-16:~$ echo hello | write pmoritzer

Message from pmoritzer@my-ubuntu-16 on pts/0 at 14:27 ...
hello
EOF
pmoritzer@my-ubuntu-16:~$
```

Figure 12: Trying out simple unix commands

```
$ sleep 600 & # start a 600 seconds sleep progress in background
$ jobs  # lists all the jobs
$ ps -ef | grep sleep # lists all the processes containing 'sleep'
$ kill -9 19940 # terminates sleep process
```



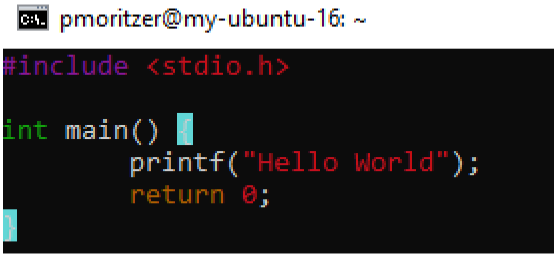
```
pmoritzer@my-ubuntu-16:~$ sleep 600 &
[1] 19940
pmoritzer@my-ubuntu-16:~$ jobs
[1]+  Running                 sleep 600 &
pmoritzer@my-ubuntu-16:~$ ps -ef | grep sleep
pmoritz+ 19940 19920  0 05:57 pts/0    00:00:00 sleep 600
pmoritz+ 19944 19920  0 05:57 pts/0    00:00:00 grep --color=auto sleep
pmoritzer@my-ubuntu-16:~$ kill -9 19940
pmoritzer@my-ubuntu-16:~$ ps -ef | grep sleep
pmoritz+ 19956 19920  0 05:58 pts/0    00:00:00 grep --color=auto sleep
[1]+  Killed                  sleep 600
```

Figure 13: Trying out job commands

C Programming

Writing a simple 'Hello World' program using C Programming Language:

```
$ sudo apt install gcc # installs gcc compiler for c language
$ vi hello_world.c
```

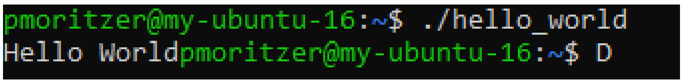


pmoritzer@my-ubuntu-16: ~

```
#include <stdio.h>

int main() {
        printf("Hello World");
        return 0;
}
```

Figure 14: Hello World C-Program

```
$ gcc -o hello_world hello_world.c # compiles program with output name hello_world
$ ./hello_world # execute hello world program
```



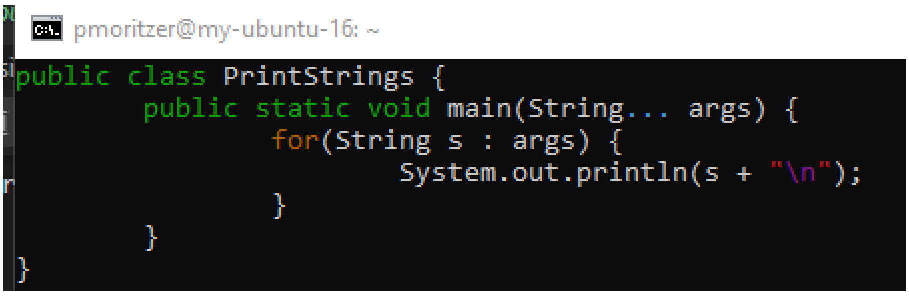
```
pmoritzer@my-ubuntu-16:~$ ./hello_world
Hello Worldpmoritzer@my-ubuntu-16:~$ D
```

Figure 15: Running Hello World C-Program

Java Programming

Writing a simple Java Programm that prints an array of input strings:

```
$ sudo apt install openjdk-8-jdk-headless # install JDK for Java Compiling and Runnings
$ vi PrintStrings.java
```

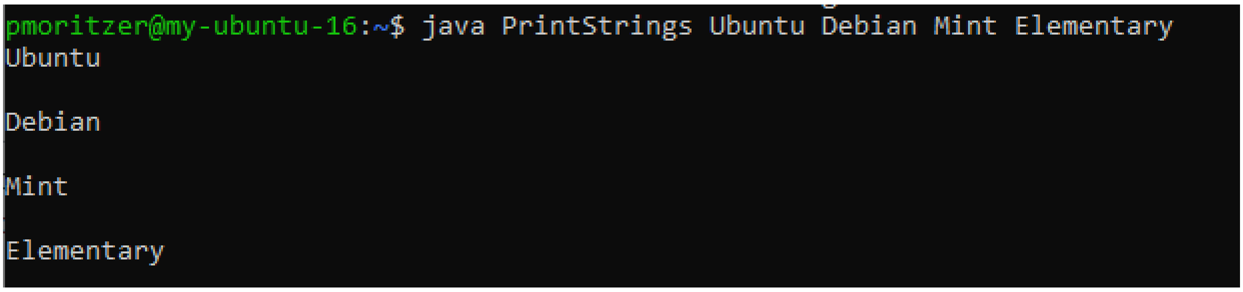


pmoritzer@my-ubuntu-16: ~

```
public class PrintStrings {
        public static void main(String... args) {
                for(String s : args) {
                        System.out.println(s + "\n");
                }
        }
}
```

Figure 16: Printing Argument Java Program

```
$ javac PrintStrings.java # compiles PrintStrings.java - program
$ java PrintStrings Ubuntu Debian Mint Elementary # Executes compiled Java Program with arguments
```



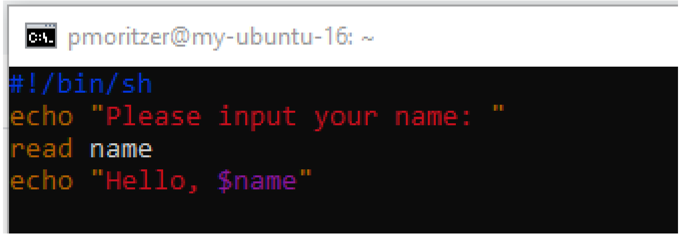
```
pmoritzer@my-ubuntu-16:~$ java PrintStrings Ubuntu Debian Mint Elementary
Ubuntu

Debian

Mint

Elementary
```

Figure 17: Running Printing Argument Java Program

Shell Programming

Writing a simple shell programm that asks the user for the name and greets the user:

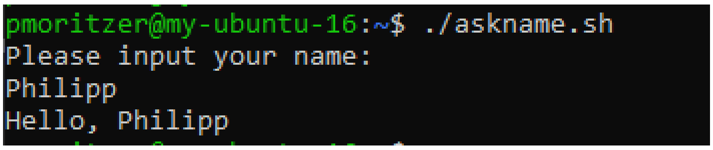
```
$ vi askname.sh
```



```
#!/bin/sh
echo "Please input your name: "
read name
echo "Hello, $name"
```

Figure 18: Ask name Shell Script

```
$ chmod u+x askname.sh
$ ./askname.sh
```



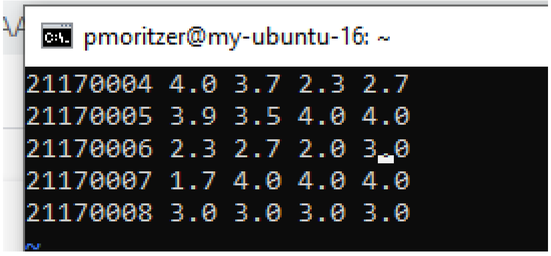
```
pmoritzer@my-ubuntu-16:~$ ./askname.sh
Please input your name:
Philipp
Hello, Philipp
```

Figure 19: Running Ask name Shell Script

AWK

Using the AWK language to print out the average grades from students:

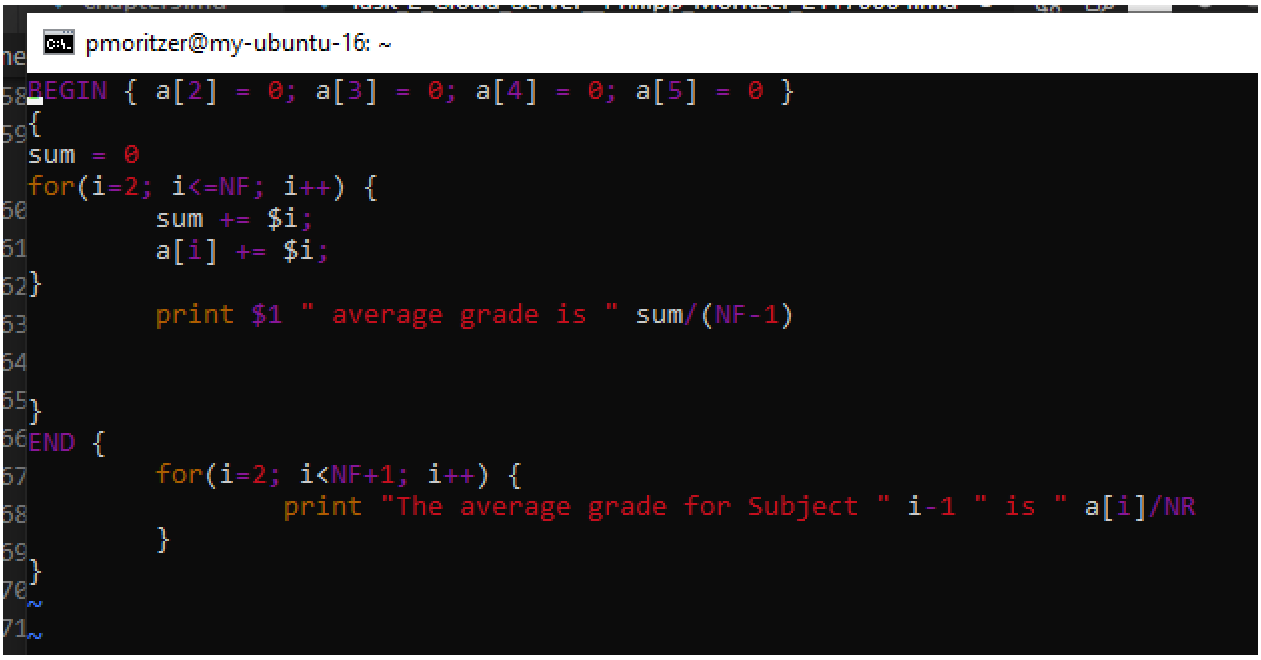
```
vi students.txt
```



```
21170004 4.0 3.7 2.3 2.7
21170005 3.9 3.5 4.0 4.0
21170006 2.3 2.7 2.0 3.0
21170007 1.7 4.0 4.0 4.0
21170008 3.0 3.0 3.0 3.0
```

Figure 20: Setting up students.txt data source

```
vi average.awk
```

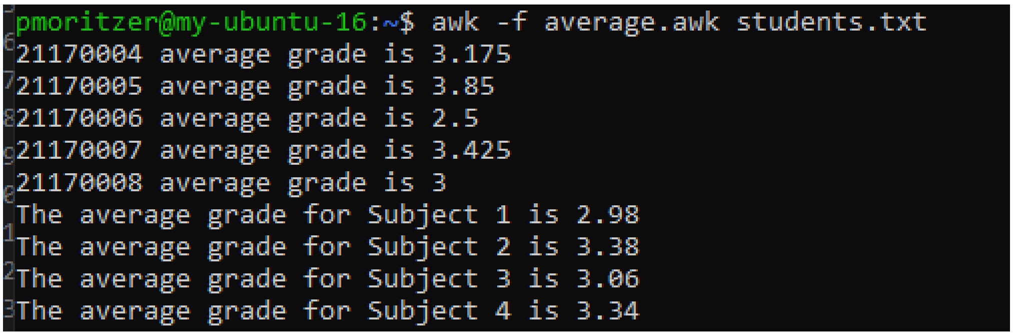


```
BEGIN { a[2] = 0; a[3] = 0; a[4] = 0; a[5] = 0 }
{
sum = 0
for(i=2; i<=NF; i++) {
        sum += $i;
        a[i] += $i;
}
        print $1 " average grade is " sum/(NF-1)


}
END {
        for(i=2; i<NF+1; i++) {
                print "The average grade for Subject " i-1 " is " a[i]/NR
        }
}
~
~
```

Figure 21: AWK Script to determine averages

```
awk -f average.awk students.txt
```

Output:



```
pmoritzer@my-ubuntu-16:~$ awk -f average.awk students.txt
21170004 average grade is 3.175
21170005 average grade is 3.85
21170006 average grade is 2.5
21170007 average grade is 3.425
21170008 average grade is 3
The average grade for Subject 1 is 2.98
The average grade for Subject 2 is 3.38
The average grade for Subject 3 is 3.06
The average grade for Subject 4 is 3.34
```

Figure 22: Running awk script with the data source