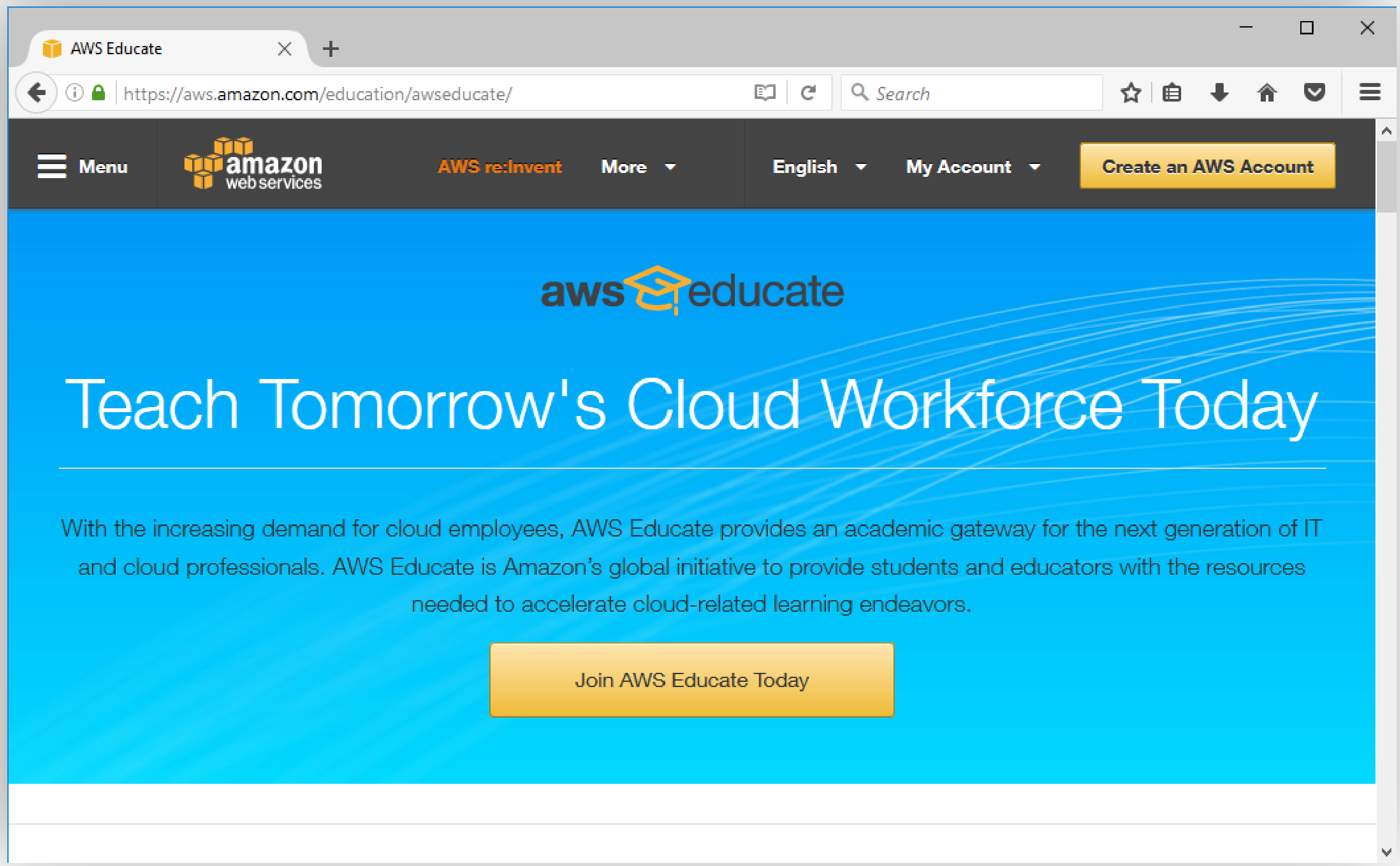
- Launch VM instances through AWS
- Allow network traffic (ping, ssh)
- Transfer files
- Download, install, and run Hadoop

# Launch VM instances

Submit your application to Amazon

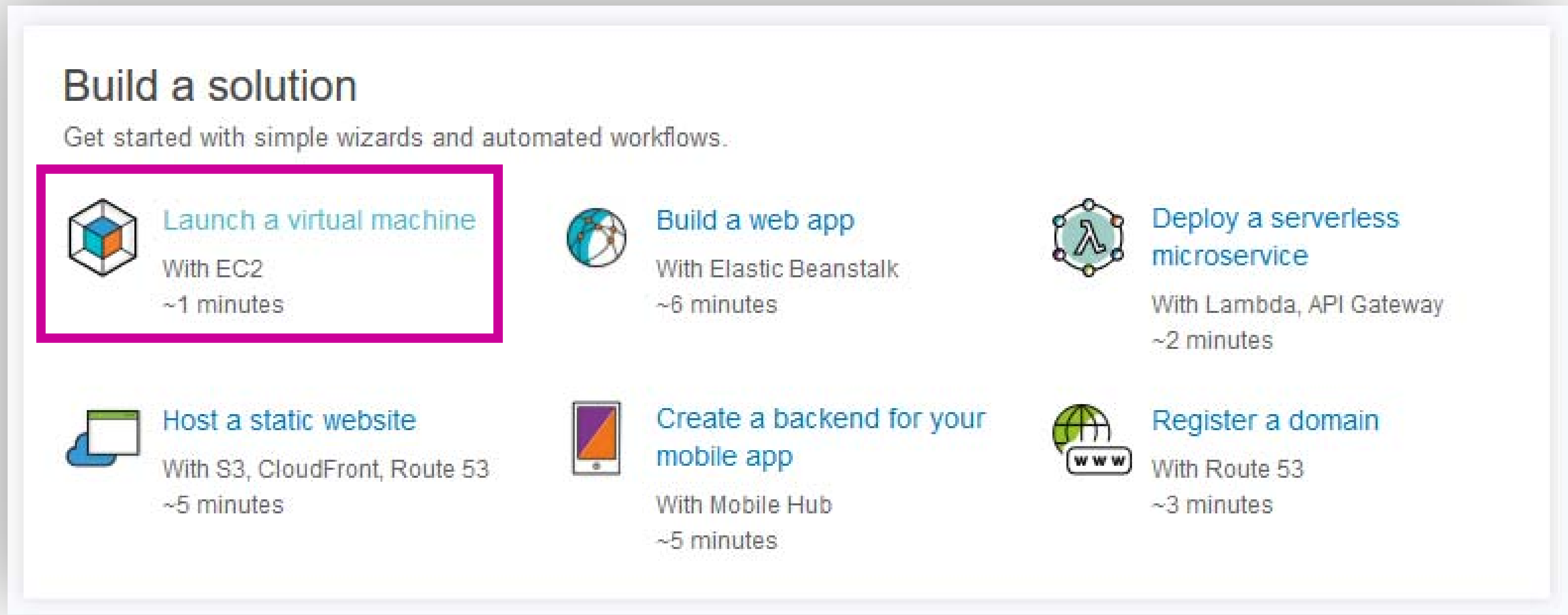


Teach Tomorrow's Cloud Workforce Today

With the increasing demand for cloud employees, AWS Educate provides an academic gateway for the next generation of IT and cloud professionals. AWS Educate is Amazon's global initiative to provide students and educators with the resources needed to accelerate cloud-related learning endeavors.

# Launch VM instances

Launch a virtual machine



## Build a solution

Get started with simple wizards and automated workflows.

Launch a virtual machine
With EC2
~1 minutes

Build a web app
With Elastic Beanstalk
~6 minutes

Deploy a serverless microservice
With Lambda, API Gateway
~2 minutes

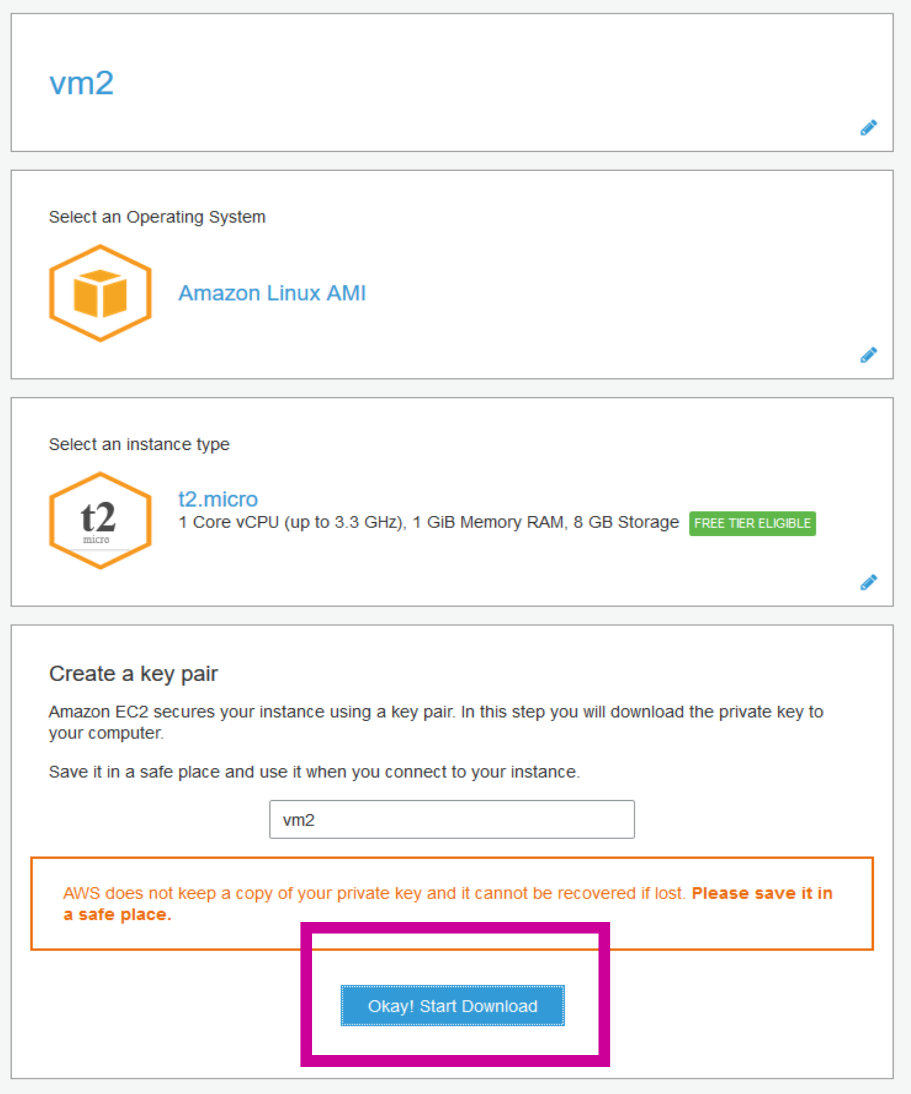
Host a static website
With S3, CloudFront, Route 53
~5 minutes

Create a backend for your mobile app
With Mobile Hub
~5 minutes

Register a domain
With Route 53
~3 minutes

# Launch VM instances

Provide necessary information
Create and download the key pair



vm2

Select an Operating System

Amazon Linux AMI

Select an instance type

t2.micro
1 Core vCPU (up to 3.3 GHz), 1 GiB Memory RAM, 8 GB Storage FREE TIER ELIGIBLE

Create a key pair

Amazon EC2 secures your instance using a key pair. In this step you will download the private key to your computer.

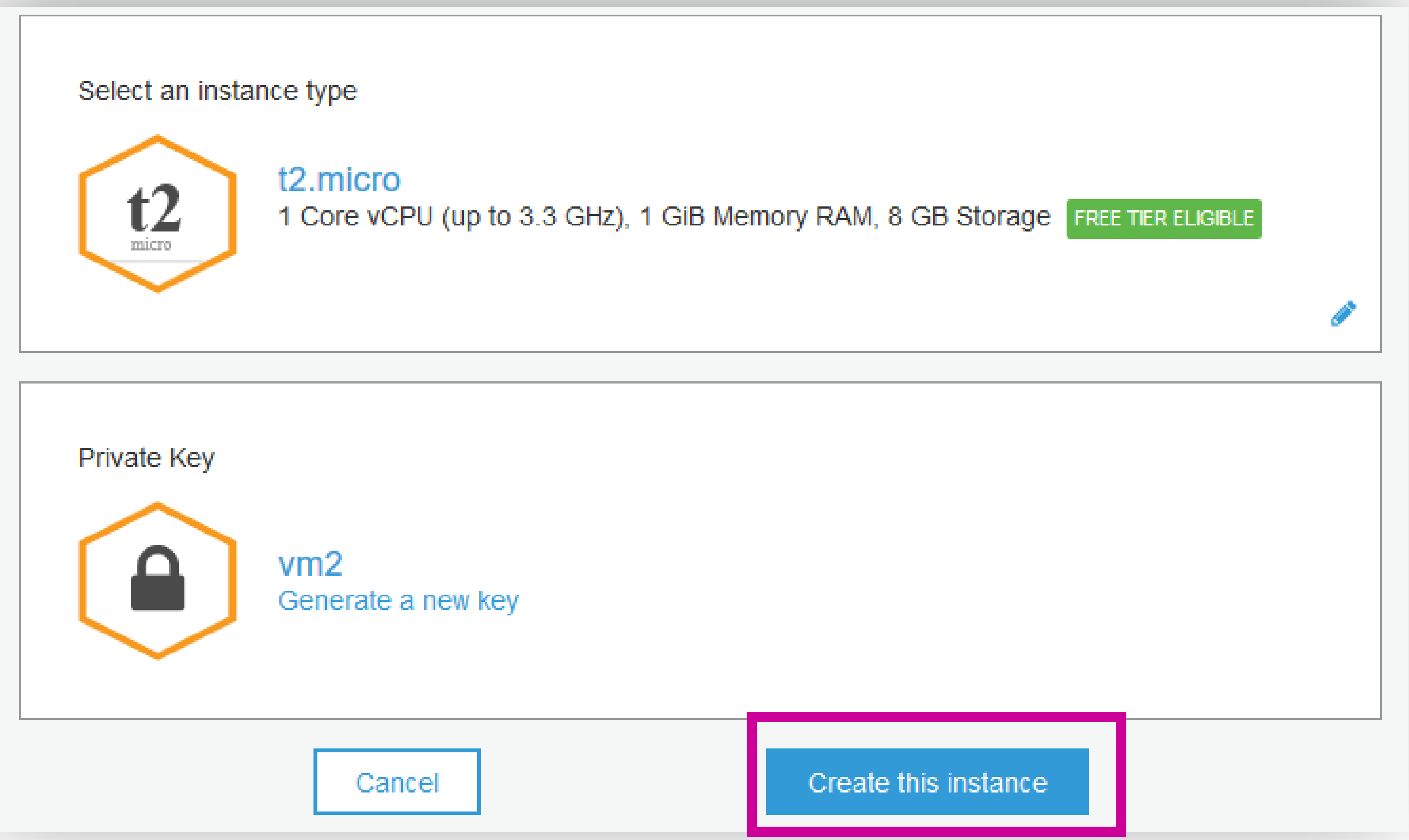
Save it in a safe place and use it when you connect to your instance.

vm2

AWS does not keep a copy of your private key and it cannot be recovered if lost. **Please save it in a safe place.**

# Launch VM instances

Create the instance



Select an instance type

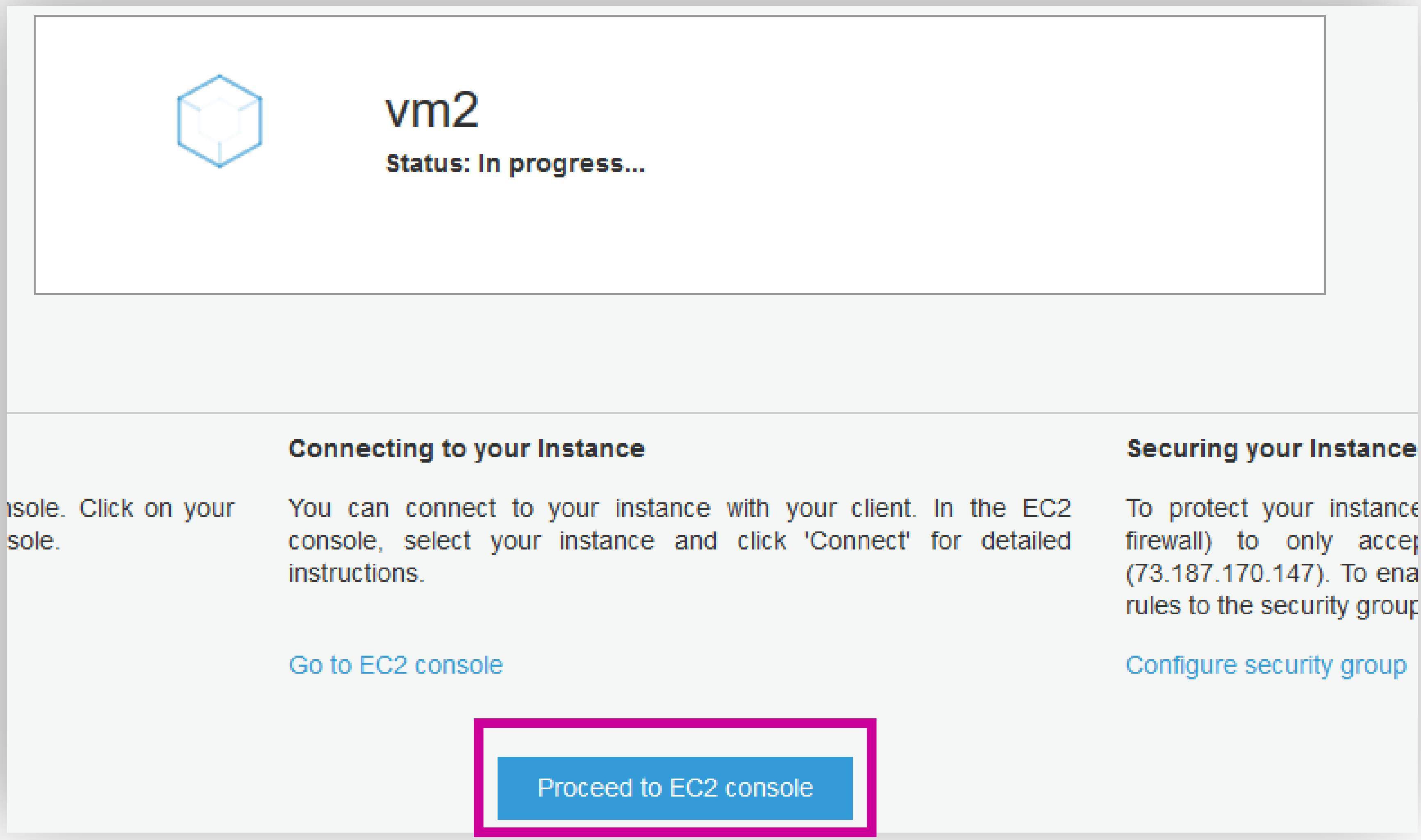
t2.micro
1 Core vCPU (up to 3.3 GHz), 1 GiB Memory RAM, 8 GB Storage FREE TIER ELIGIBLE

Private Key

vm2
Generate a new key

# Launch VM instances

Creating…
You can go ahead proceed to the console



vm2

**Status: In progress...**

nsole. Click on your sole.

**Connecting to your Instance**

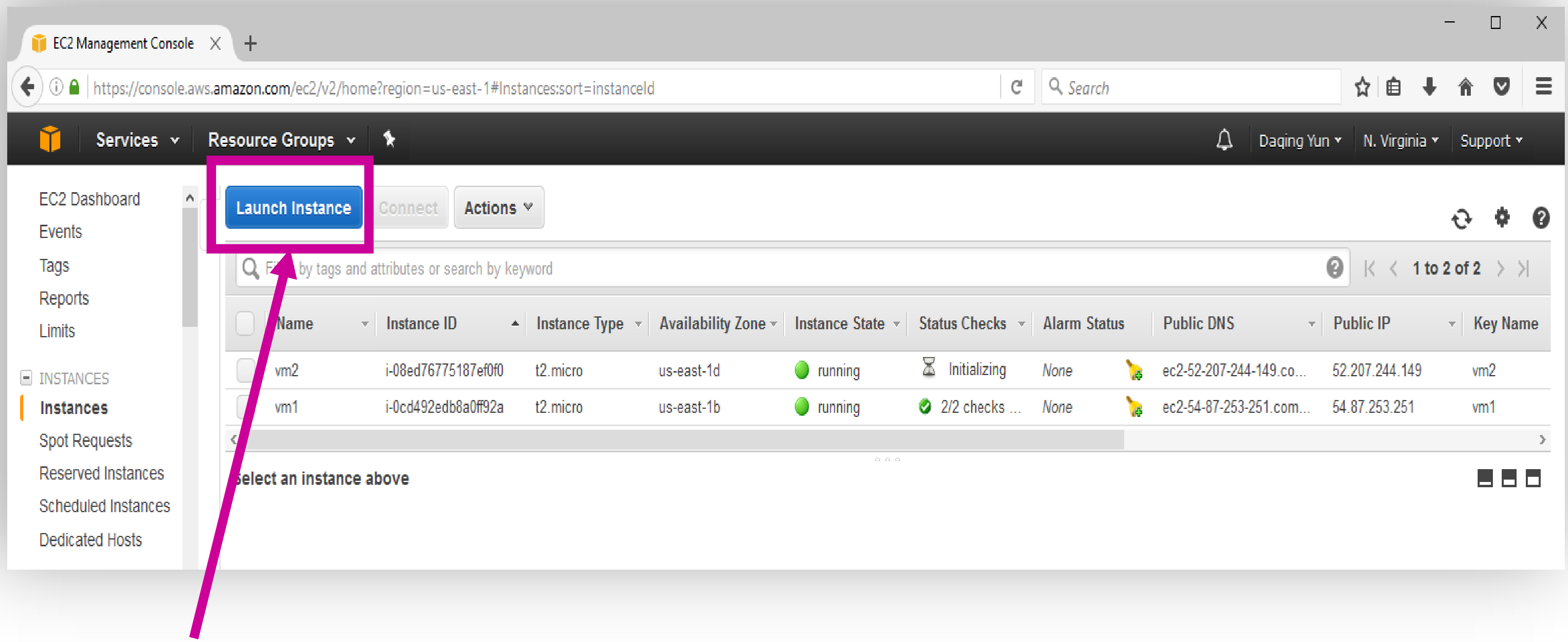
You can connect to your instance with your client. In the EC2 console, select your instance and click 'Connect' for detailed instructions.

**Securing your Instance**

To protect your instance firewall) to only accep (73.187.170.147). To ena rules to the security group

# Launch VM instances

The console is like…

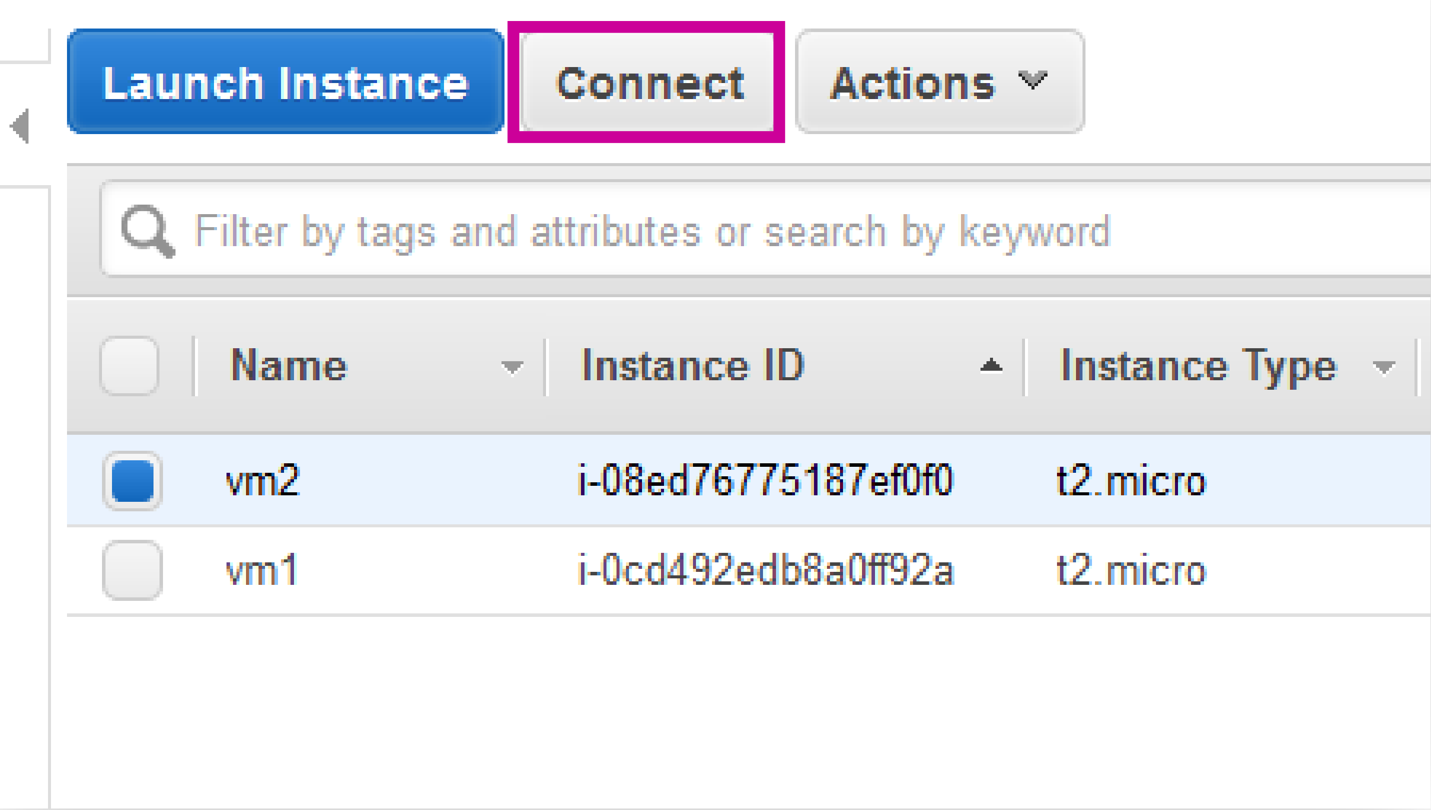


| Name | Instance ID | Instance Type | Availability Zone | Instance State | Status Checks | Alarm Status | Public DNS | Public IP | Key Name |
|---|---|---|---|---|---|---|---|---|---|
| vm2 | i-08ed76775187ef0f0 | t2.micro | us-east-1d | running | Initializing | None | ec2-52-207-244-149.co... | 52.207.244.149 | vm2 |
| vm1 | i-0cd492edb8a0ff92a | t2.micro | us-east-1b | running | 2/2 checks ... | None | ec2-54-87-253-251.com... | 54.87.253.251 | vm1 |

You can keep launching…

# Allow Network Traffic

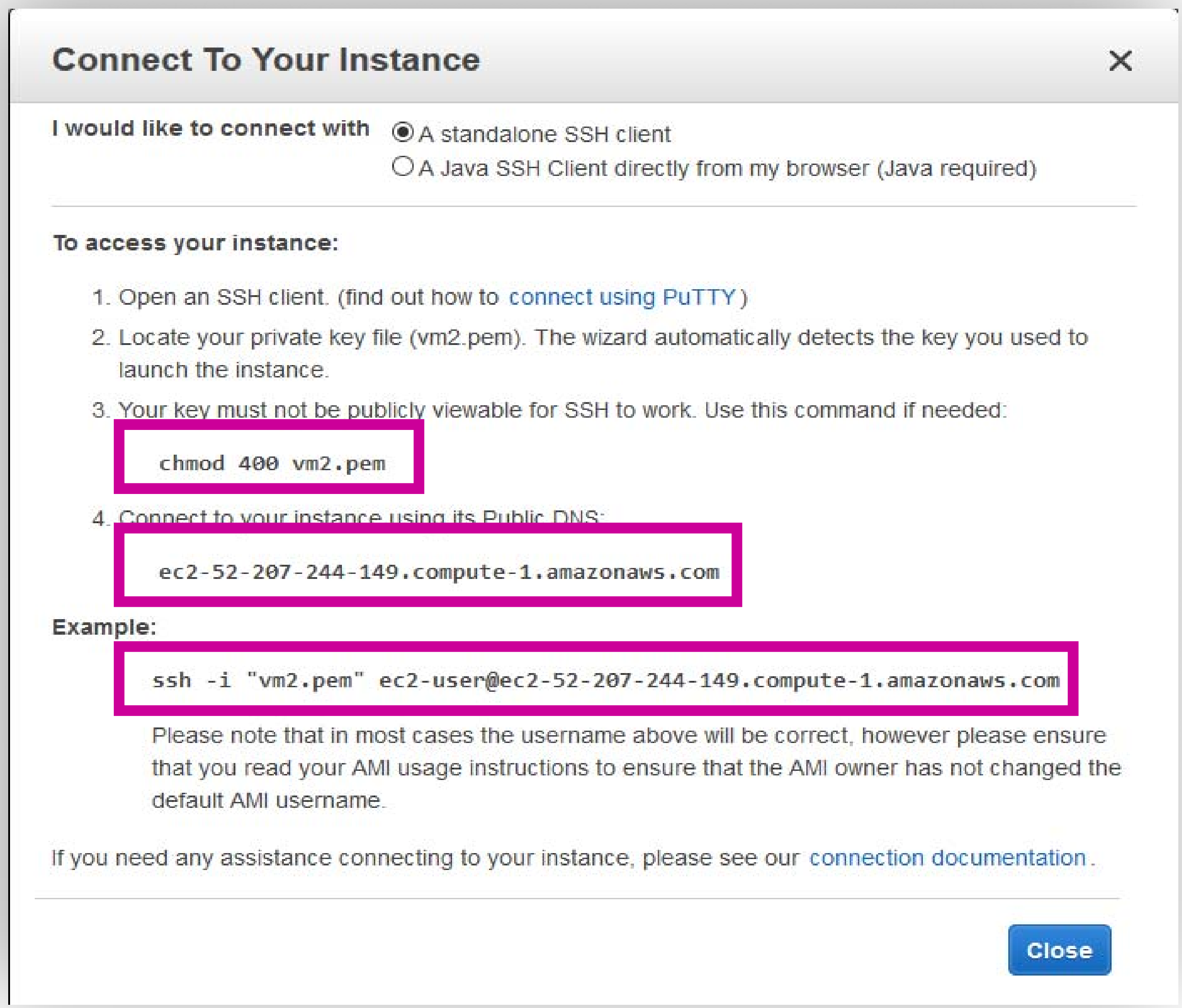
Try to connect…



| □ | Name | Instance ID | Instance Type |
|---|---|---|---|
| ■ | vm2 | i-08ed76775187ef0f0 | t2.micro |
| □ | vm1 | i-0cd492edb8a0ff92a | t2.micro |

# Allow Network Traffic

Once you click connect, you will see instructions. If you are using Mac, you can use Terminal in /Application to connect



I would like to connect with ● A standalone SSH client ○ A Java SSH Client directly from my browser (Java required)

**To access your instance:**

1. Open an SSH client. (find out how to connect using PuTTY)
2. Locate your private key file (vm2.pem). The wizard automatically detects the key you used to launch the instance.
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:

```
chmod 400 vm2.pem
```

4. Connect to your instance using its Public DNS:

```
ec2-52-207-244-149.compute-1.amazonaws.com
```

**Example:**

```
ssh -i "vm2.pem" ec2-user@ec2-52-207-244-149.compute-1.amazonaws.com
```

Please note that in most cases the username above will be correct, however please ensure that you read your AMI usage instructions to ensure that the AMI owner has not changed the default AMI username.

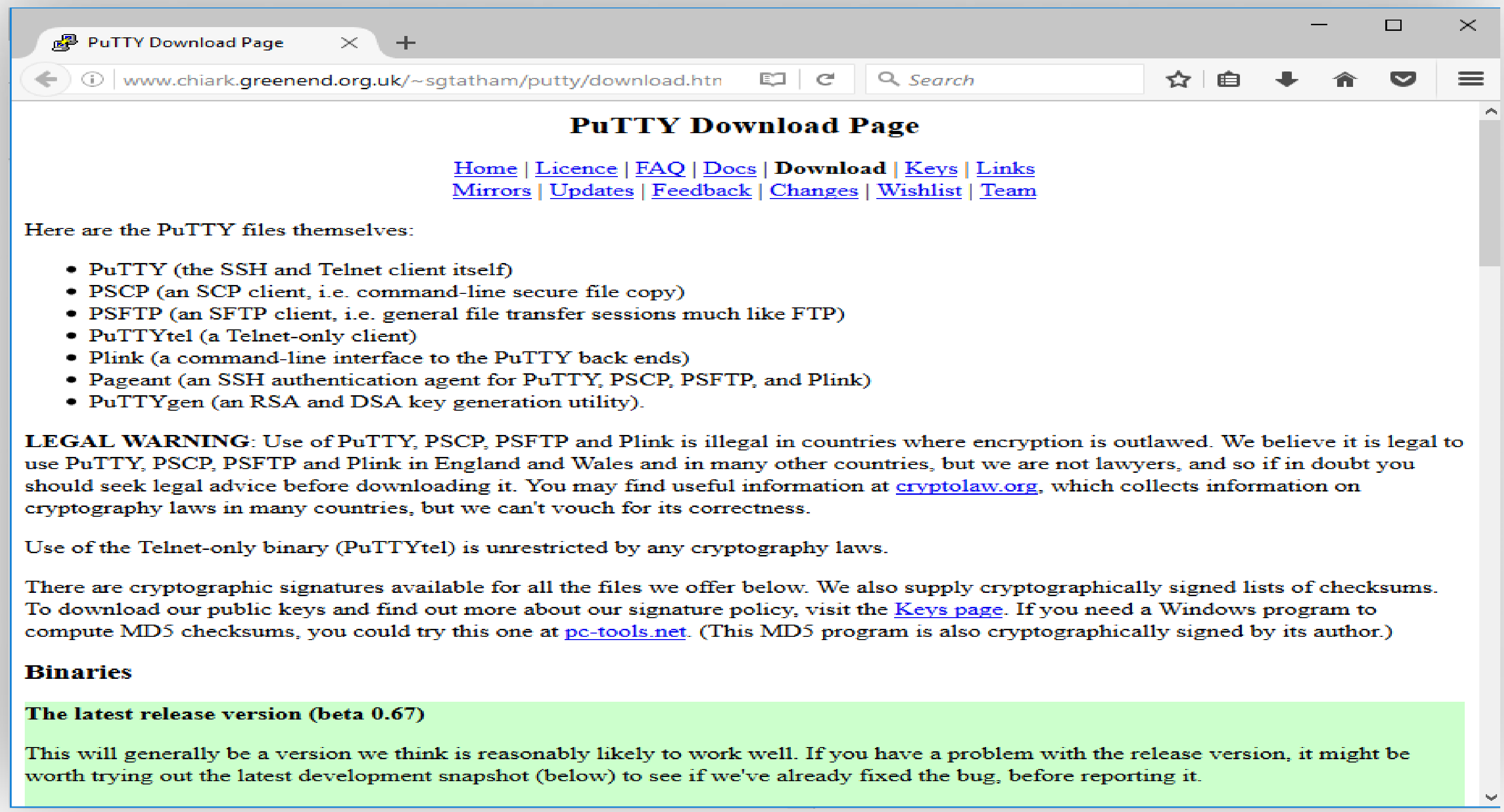
If you need any assistance connecting to your instance, please see our connection documentation.

# Allow Network Traffic

Let us use putty to connect from Windows 10

Download and install putty.exe
https://the.earth.li/~sgtatham/putty/latest/x86/putty.exe

Also download puttygen.exe
https://the.earth.li/~sgtatham/putty/latest/x86/puttygen.exe



**PuTTY Download Page**

Home | Licence | FAQ | Docs | **Download** | Keys | Links
Mirrors | Updates | Feedback | Changes | Wishlist | Team

Here are the PuTTY files themselves:

- PuTTY (the SSH and Telnet client itself)
- PSCP (an SCP client, i.e. command-line secure file copy)
- PSFTP (an SFTP client, i.e. general file transfer sessions much like FTP)
- PuTTYtel (a Telnet-only client)
- Plink (a command-line interface to the PuTTY back ends)
- Pageant (an SSH authentication agent for PuTTY, PSCP, PSFTP, and Plink)
- PuTTYgen (an RSA and DSA key generation utility).

**LEGAL WARNING**: Use of PuTTY, PSCP, PSFTP and Plink is illegal in countries where encryption is outlawed. We believe it is legal to use PuTTY, PSCP, PSFTP and Plink in England and Wales and in many other countries, but we are not lawyers, and so if in doubt you should seek legal advice before downloading it. You may find useful information at cryptolaw.org, which collects information on cryptography laws in many countries, but we can't vouch for its correctness.

Use of the Telnet-only binary (PuTTYtel) is unrestricted by any cryptography laws.

There are cryptographic signatures available for all the files we offer below. We also supply cryptographically signed lists of checksums. To download our public keys and find out more about our signature policy, visit the Keys page. If you need a Windows program to compute MD5 checksums, you could try this one at pc-tools.net. (This MD5 program is also cryptographically signed by its author.)

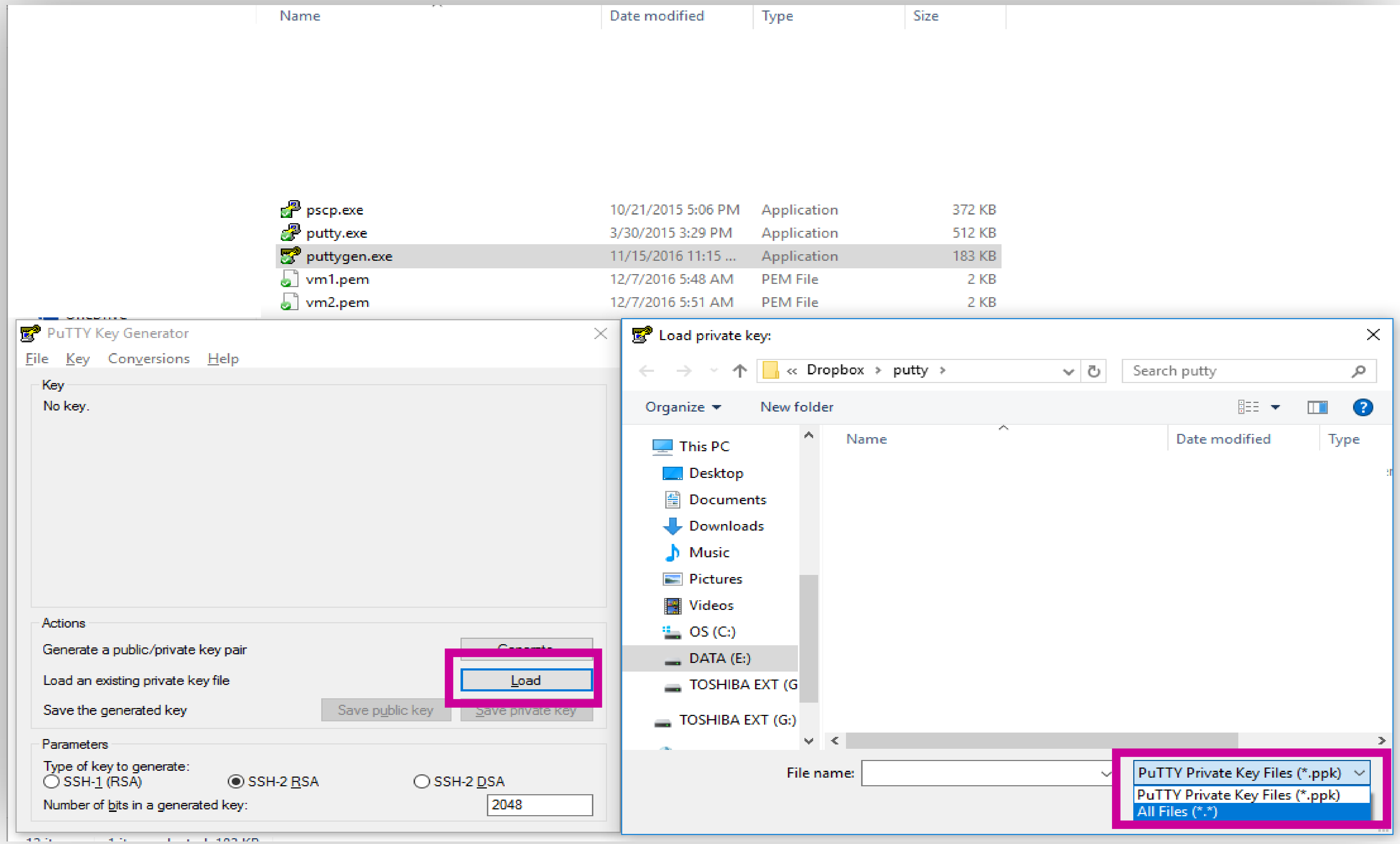
**Binaries**

**The latest release version (beta 0.67)**

This will generally be a version we think is reasonably likely to work well. If you have a problem with the release version, it might be worth trying out the latest development snapshot (below) to see if we've already fixed the bug, before reporting it.

# Allow Network Traffic

PuTTY does not natively support the private key format (.pem) generated by Amazon EC2. Let’s convert it using puttygen.exe



| Name | Date modified | Type | Size |
|---|---|---|---|
| pscp.exe | 10/21/2015 5:06 PM | Application | 372 KB |
| putty.exe | 3/30/2015 3:29 PM | Application | 512 KB |
| puttygen.exe | 11/15/2016 11:15 ... | Application | 183 KB |
| vm1.pem | 12/7/2016 5:48 AM | PEM File | 2 KB |
| vm2.pem | 12/7/2016 5:51 AM | PEM File | 2 KB |

PuTTY Key Generator

Key

No key.

Actions

Generate a public/private key pair

Load an existing private key file

Save the generated key

Parameters

Type of key to generate:

○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048

Load private key:

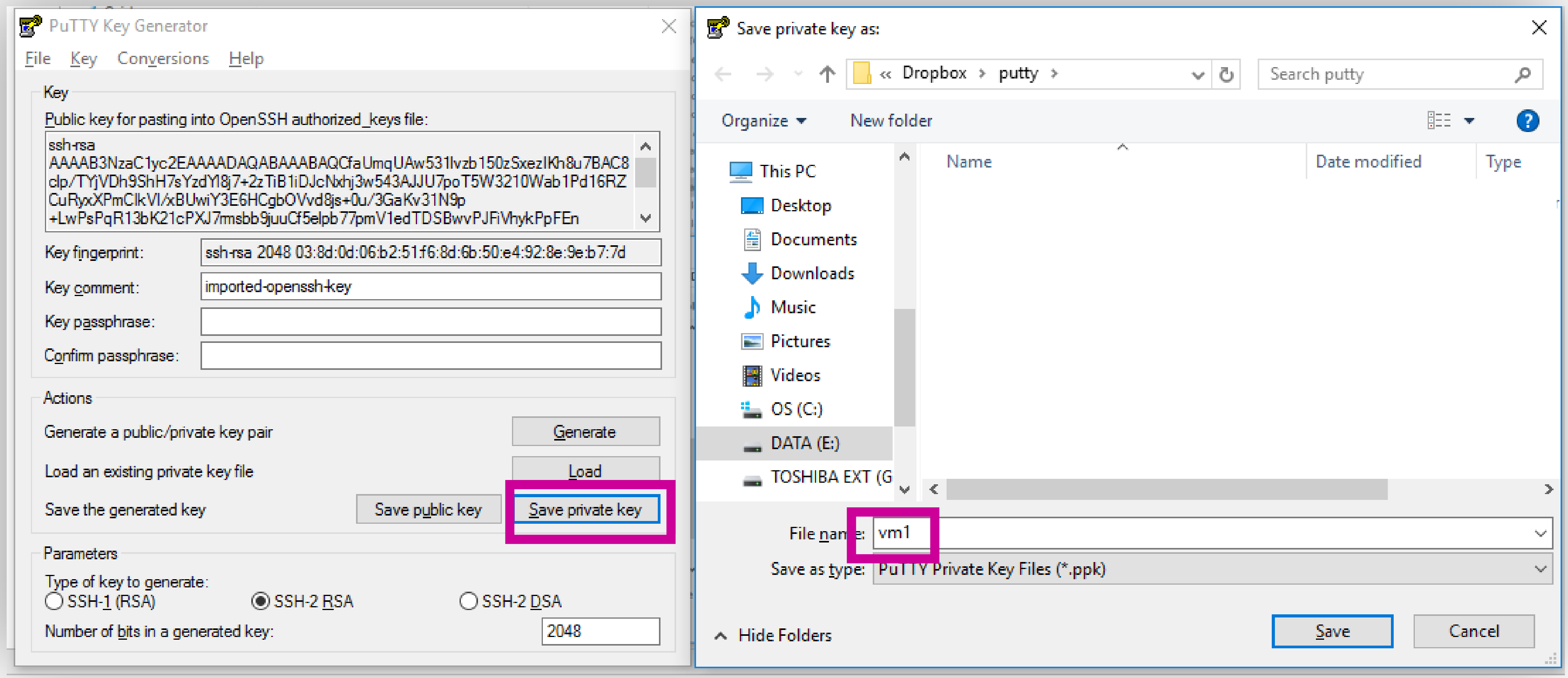
File name:

PuTTY Private Key Files (*.ppk)

PuTTY Private Key Files (*.ppk)
All Files (*.*)

# Allow Network Traffic

Save private key, specify the file name with .ppk file extension



Key

Public key for pasting into OpenSSH authorized_keys file:

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQCfaUmqUAw531lvzb150zSxezIKh8u7BAC8
clp/TYjVDh9ShH7sYzdYl8j7+2zTiB1iDJcNxhj3w543AJJU7poT5W3210Wab1Pd16RZ
CuRyxXPmClkVI/xBUwiY3E6HCgbOVvd8js+0u/3GaKv31N9p
+LwPsPqR13bK21cPXJ7msbb9juuCf5elpb77pmV1edTDSBwvPJFiVhykPpFEn
```

Key fingerprint: ssh-rsa 2048 03:8d:0d:06:b2:51:f6:8d:6b:50:e4:92:8e:9e:b7:7d

Key comment: imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair

Load an existing private key file

Save the generated key

Parameters

Type of key to generate:

○ SSH-1 (RSA) ● SSH-2 RSA ○ SSH-2 DSA

Number of bits in a generated key: 2048

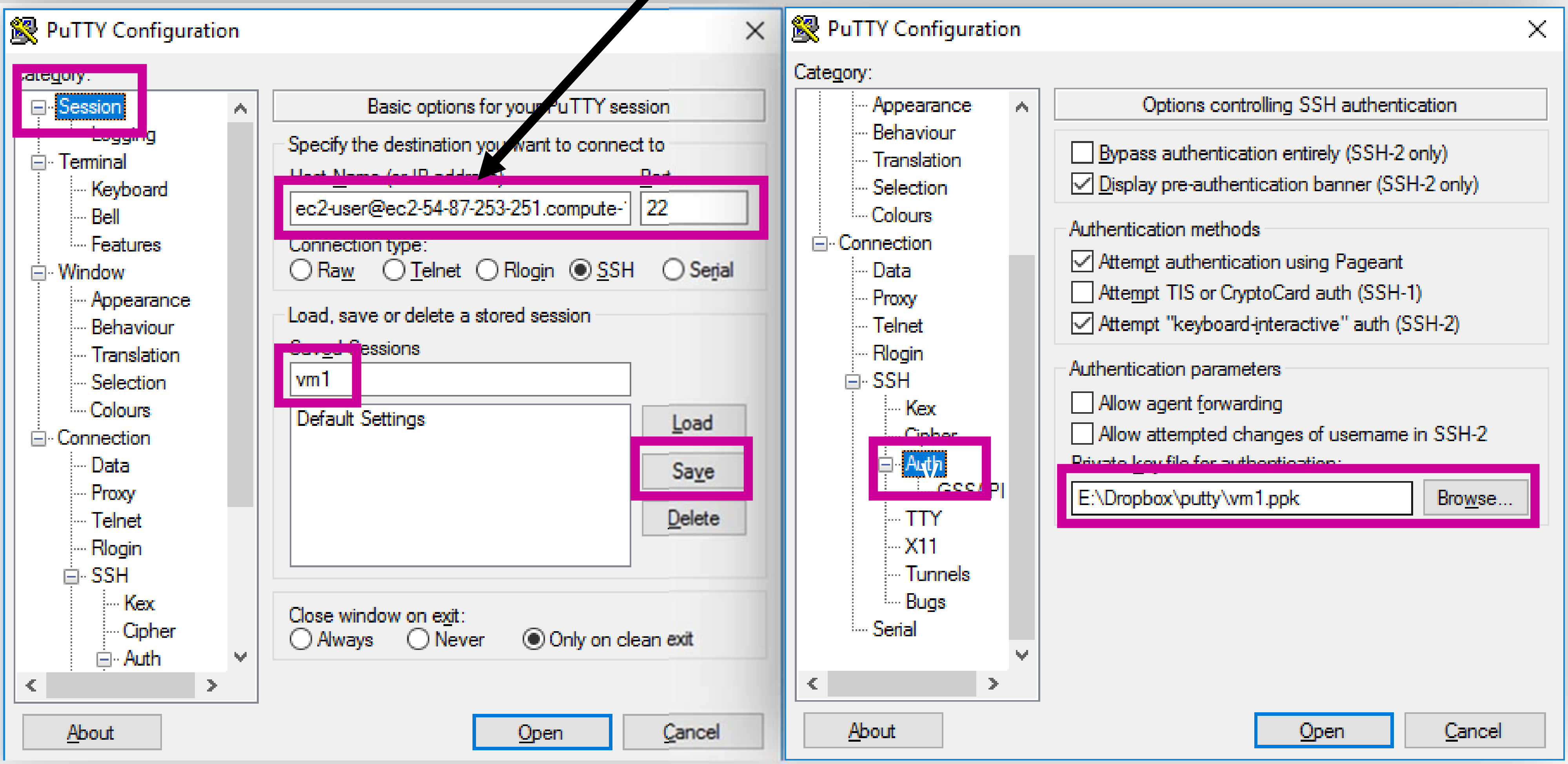
File name: vm1

Save as type: PuTTY Private Key Files (*.ppk)

# Allow Network Traffic

Let us try to connect vm1

```
ec2-user@ec2-54-87-253-251.compute-1.amazonaws.com
```



PuTTY Configuration

Category:

Basic options for your PuTTY session

Specify the destination you want to connect to

Host Name (or IP address): ec2-user@ec2-54-87-253-251.compute-

Port: 22

Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: vm1

Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

PuTTY Configuration

Category:

Options controlling SSH authentication

- □ Bypass authentication entirely (SSH-2 only)
- ☑ Display pre-authentication banner (SSH-2 only)

Authentication methods

- ☑ Attempt authentication using Pageant
- □ Attempt TIS or CryptoCard auth (SSH-1)
- ☑ Attempt "keyboard-interactive" auth (SSH-2)

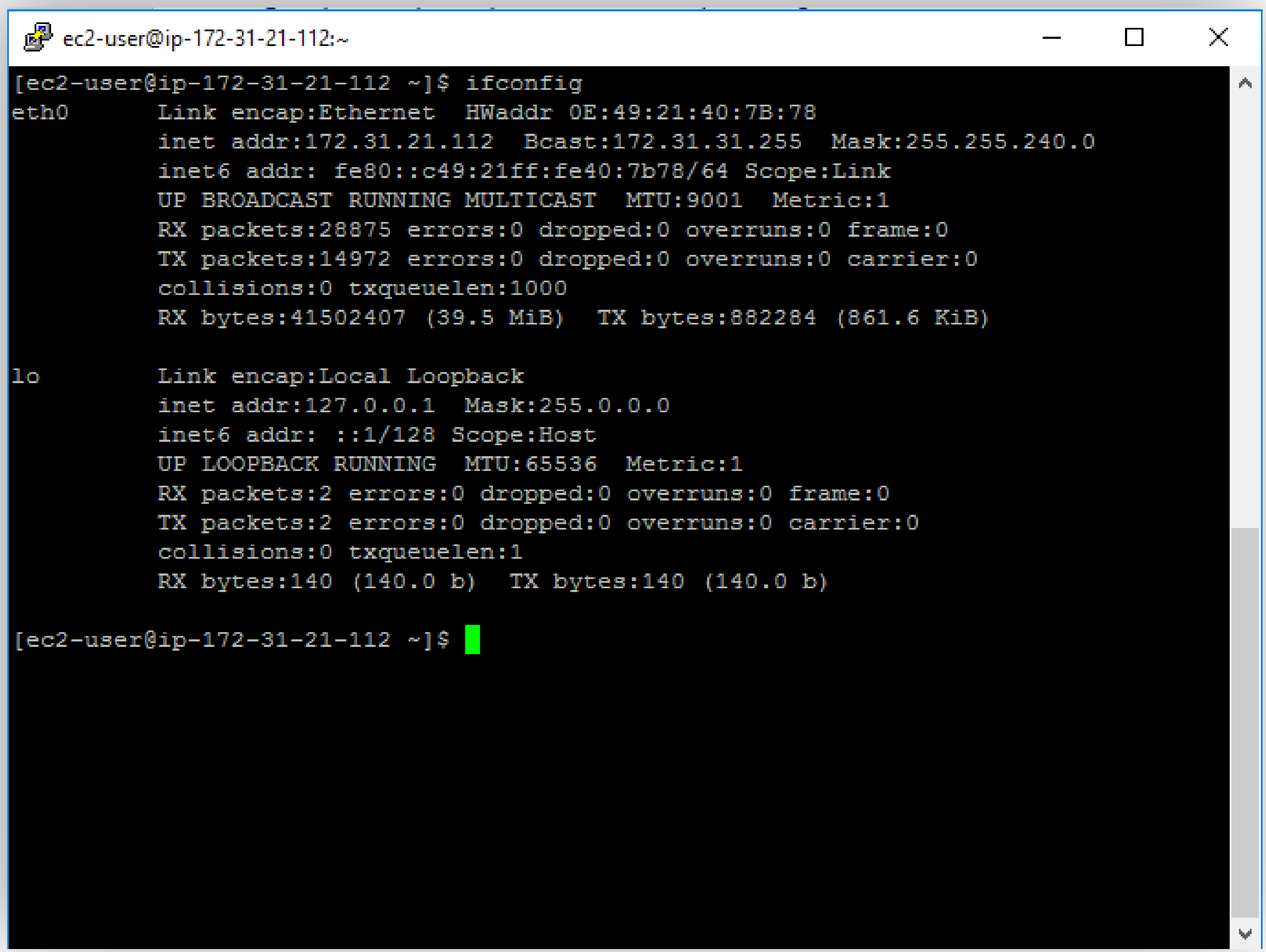
Authentication parameters

- □ Allow agent forwarding
- □ Allow attempted changes of username in SSH-2

Private key file for authentication: E:\Dropbox\putty\vm1.ppk

# Allow Network Traffic

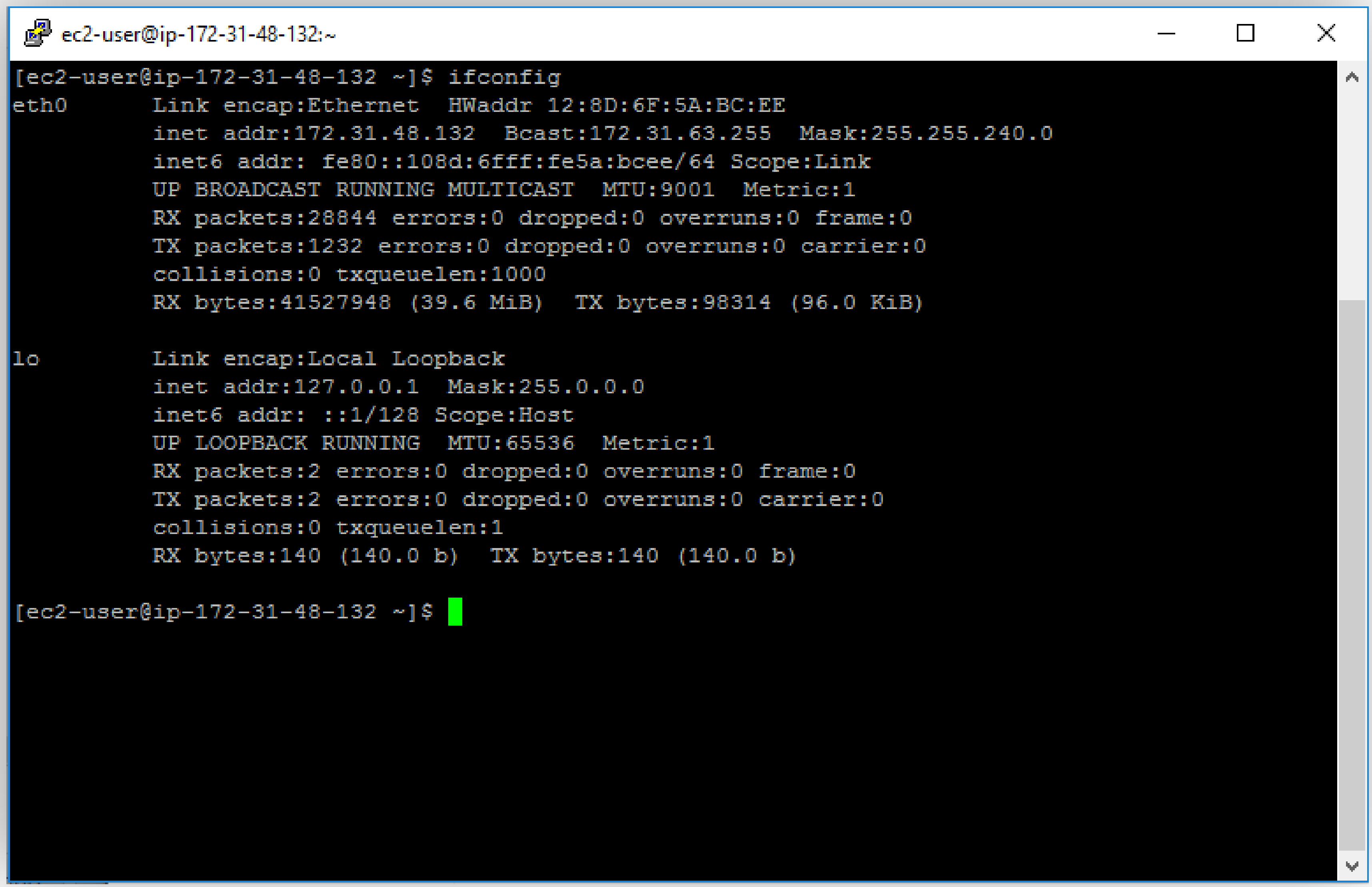
Now you have remotely logged into vm1
Let us issue a command `ifconfig`



```
[ec2-user@ip-172-31-21-112 ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 0E:49:21:40:7B:78
          inet addr:172.31.21.112  Bcast:172.31.31.255  Mask:255.255.240.0
          inet6 addr: fe80::c49:21ff:fe40:7b78/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:28875 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14972 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:41502407 (39.5 MiB)  TX bytes:882284 (861.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:140 (140.0 b)  TX bytes:140 (140.0 b)

[ec2-user@ip-172-31-21-112 ~]$
```

# Allow Network Traffic

Do the same to connect vm2



```
[ec2-user@ip-172-31-48-132 ~]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 12:8D:6F:5A:BC:EE
          inet addr:172.31.48.132  Bcast:172.31.63.255  Mask:255.255.240.0
          inet6 addr: fe80::108d:6fff:fe5a:bcee/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:28844 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1232 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:41527948 (39.6 MiB)  TX bytes:98314 (96.0 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:140 (140.0 b)  TX bytes:140 (140.0 b)

[ec2-user@ip-172-31-48-132 ~]$
```

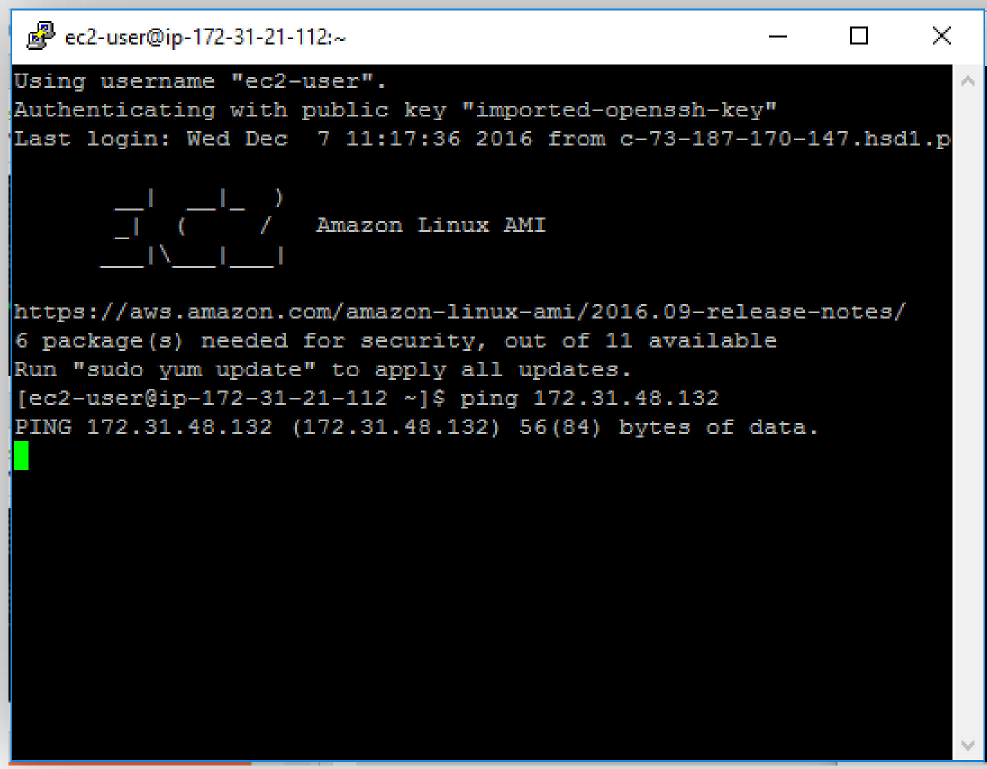
# Allow Network Traffic

Let us see if vm1 and vm2 can talk to each other

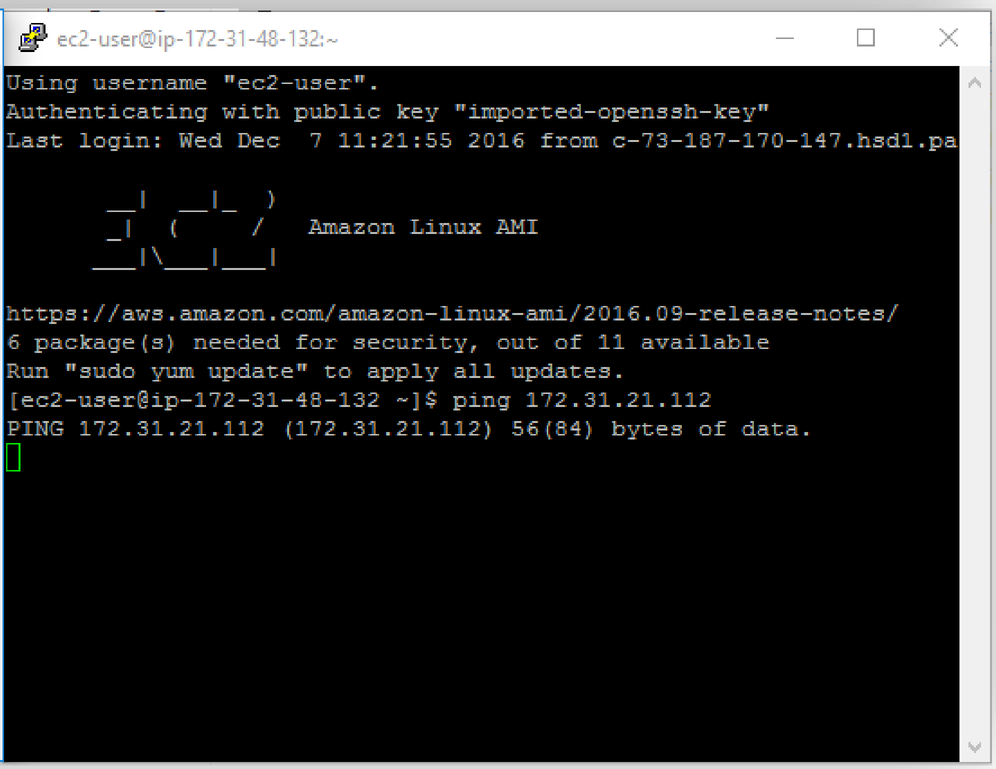
```
ping 172.31.48.132
ping 172.31.21.112
```

Cannot ping each other…

Use `ctrl + c` to kill the ping commands



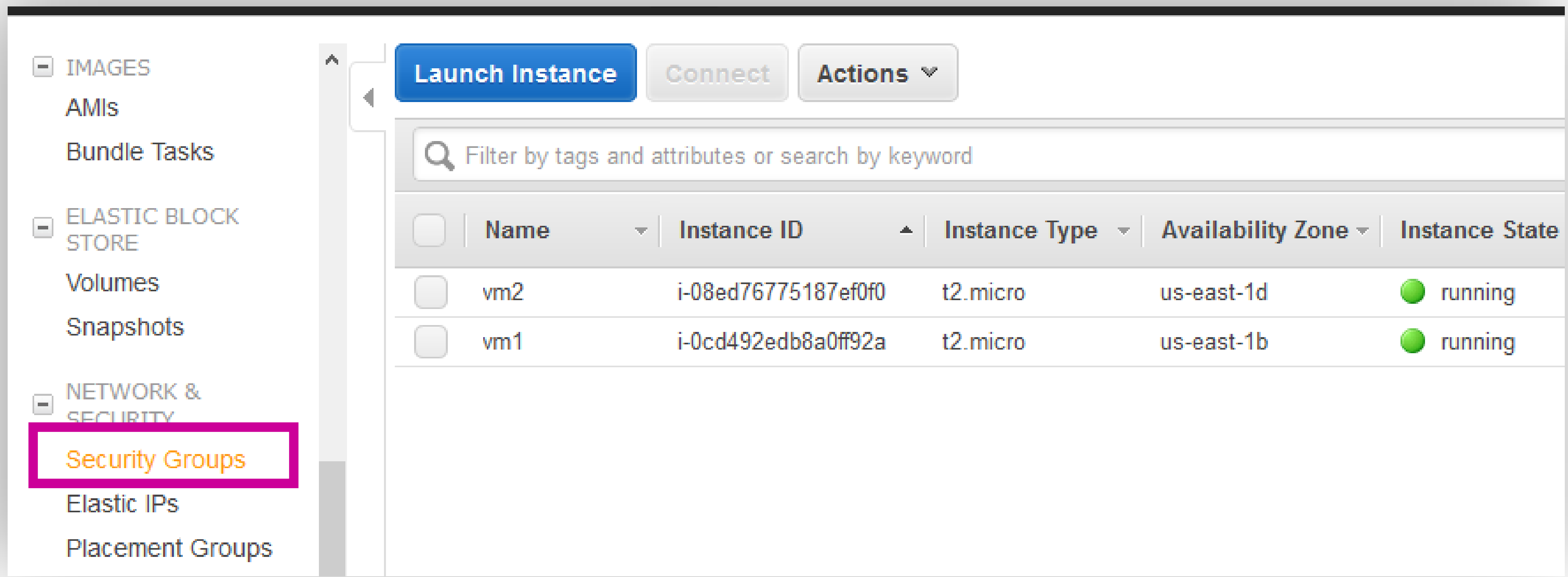
```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Wed Dec  7 11:17:36 2016 from c-73-187-170-147.hsd1.p

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
6 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-112 ~]$ ping 172.31.48.132
PING 172.31.48.132 (172.31.48.132) 56(84) bytes of data.
```



```
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Wed Dec  7 11:21:55 2016 from c-73-187-170-147.hsd1.pa

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
6 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-48-132 ~]$ ping 172.31.21.112
PING 172.31.21.112 (172.31.21.112) 56(84) bytes of data.
```

# Allow Network Traffic

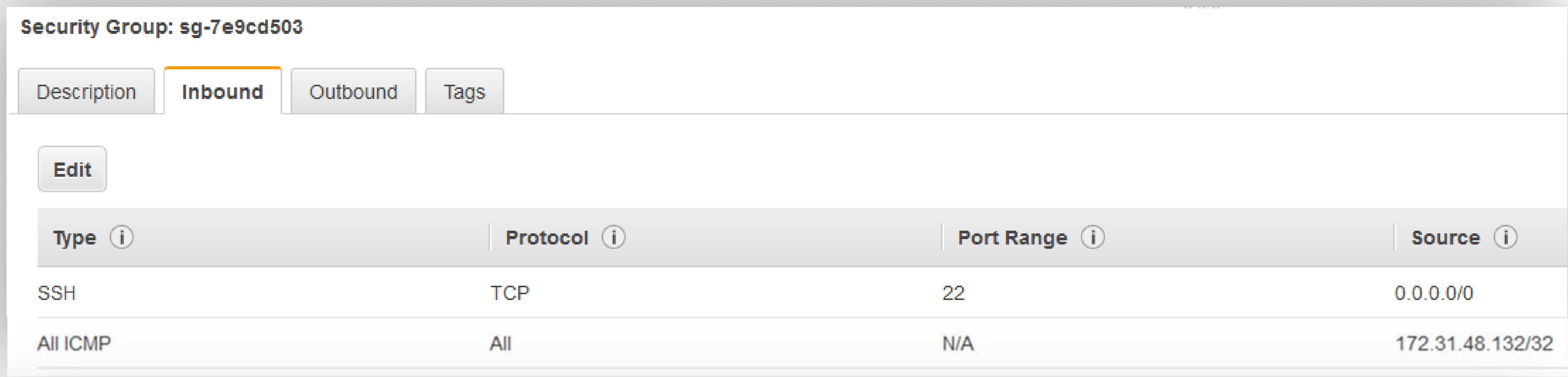
This’s because you need to add appropriate security group for the VMs to allow network traffic



| □ | Name | Instance ID | Instance Type | Availability Zone | Instance State |
|---|---|---|---|---|---|
| □ | vm2 | i-08ed76775187ef0f0 | t2.micro | us-east-1d | ● running |
| □ | vm1 | i-0cd492edb8a0ff92a | t2.micro | us-east-1b | ● running |

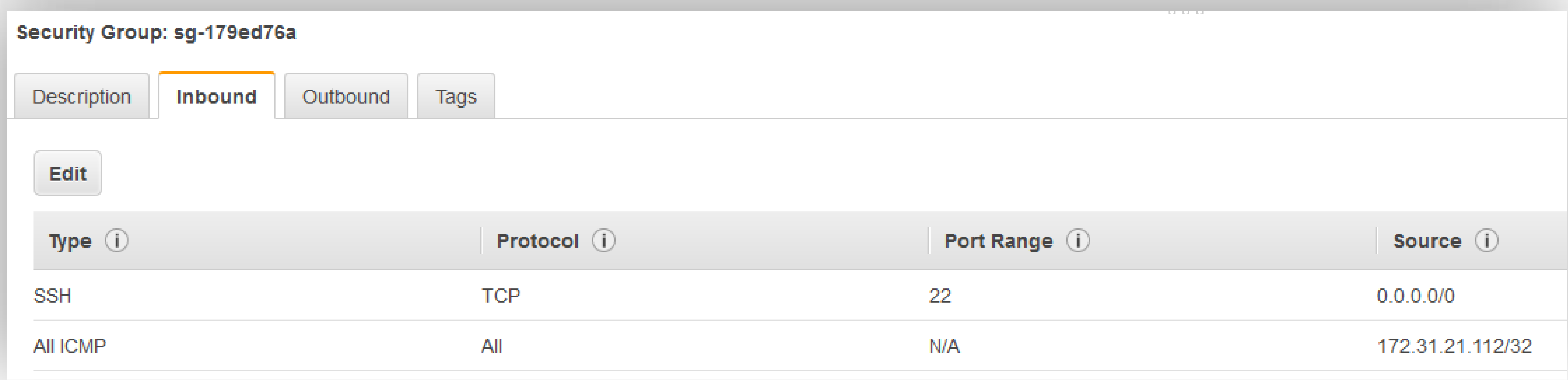
# Allow Network Traffic

Let us allow network traffic for ICMP echo and SSH remote login, and all TCP traffic (not shown)



Security Group: sg-7e9cd503

| Type | Protocol | Port Range | Source |
|---|---|---|---|
| SSH | TCP | 22 | 0.0.0.0/0 |
| All ICMP | All | N/A | 172.31.48.132/32 |

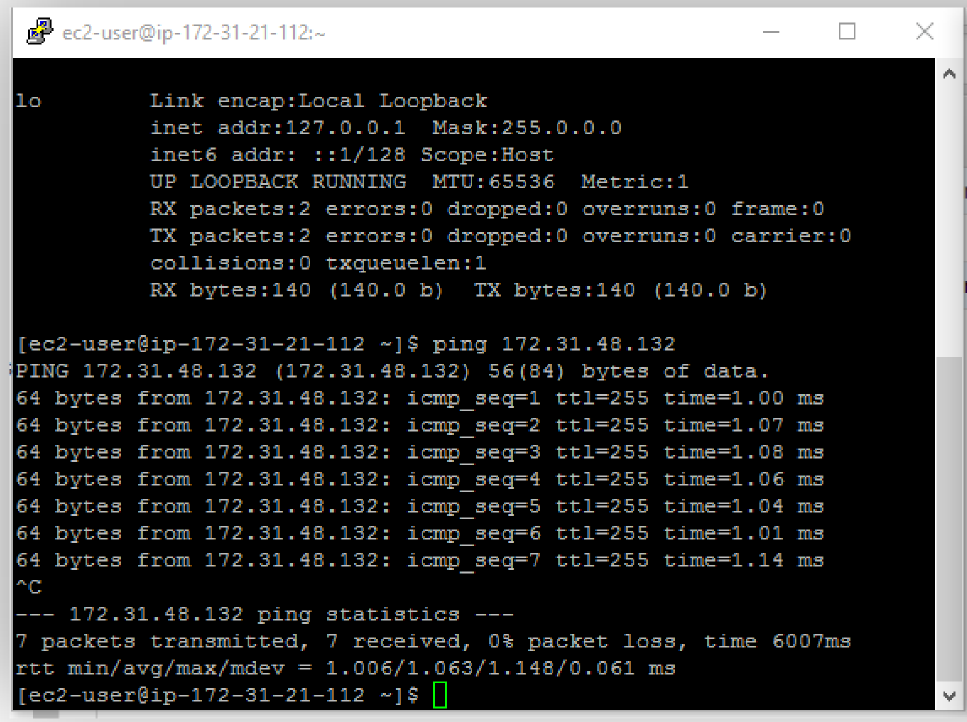


Security Group: sg-179ed76a

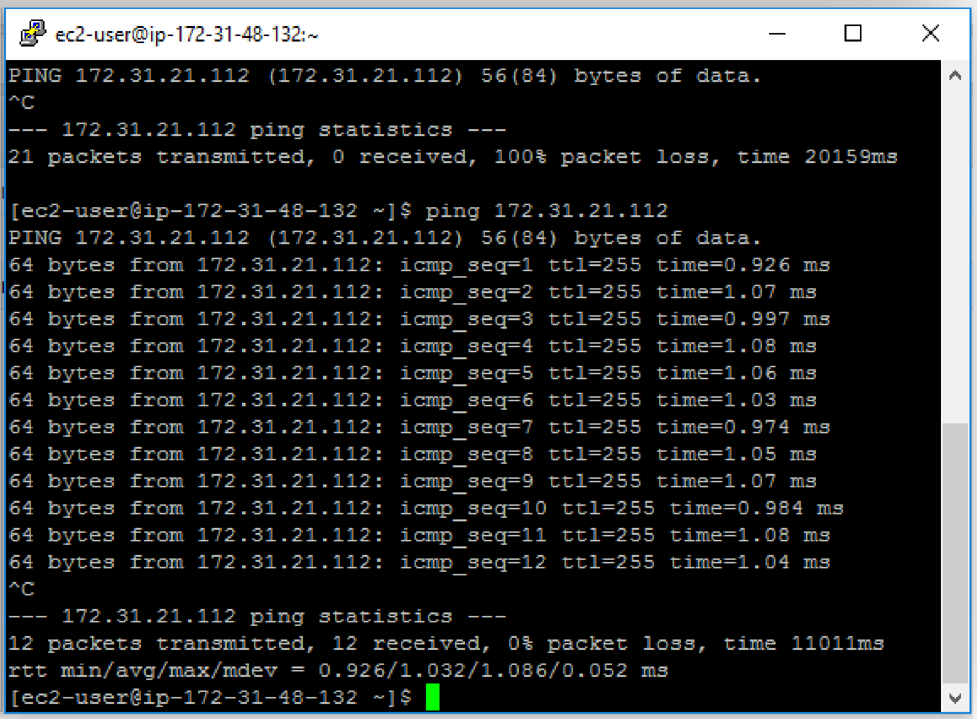
| Type | Protocol | Port Range | Source |
|---|---|---|---|
| SSH | TCP | 22 | 0.0.0.0/0 |
| All ICMP | All | N/A | 172.31.21.112/32 |

# Allow Network Traffic

Now you can ping each other



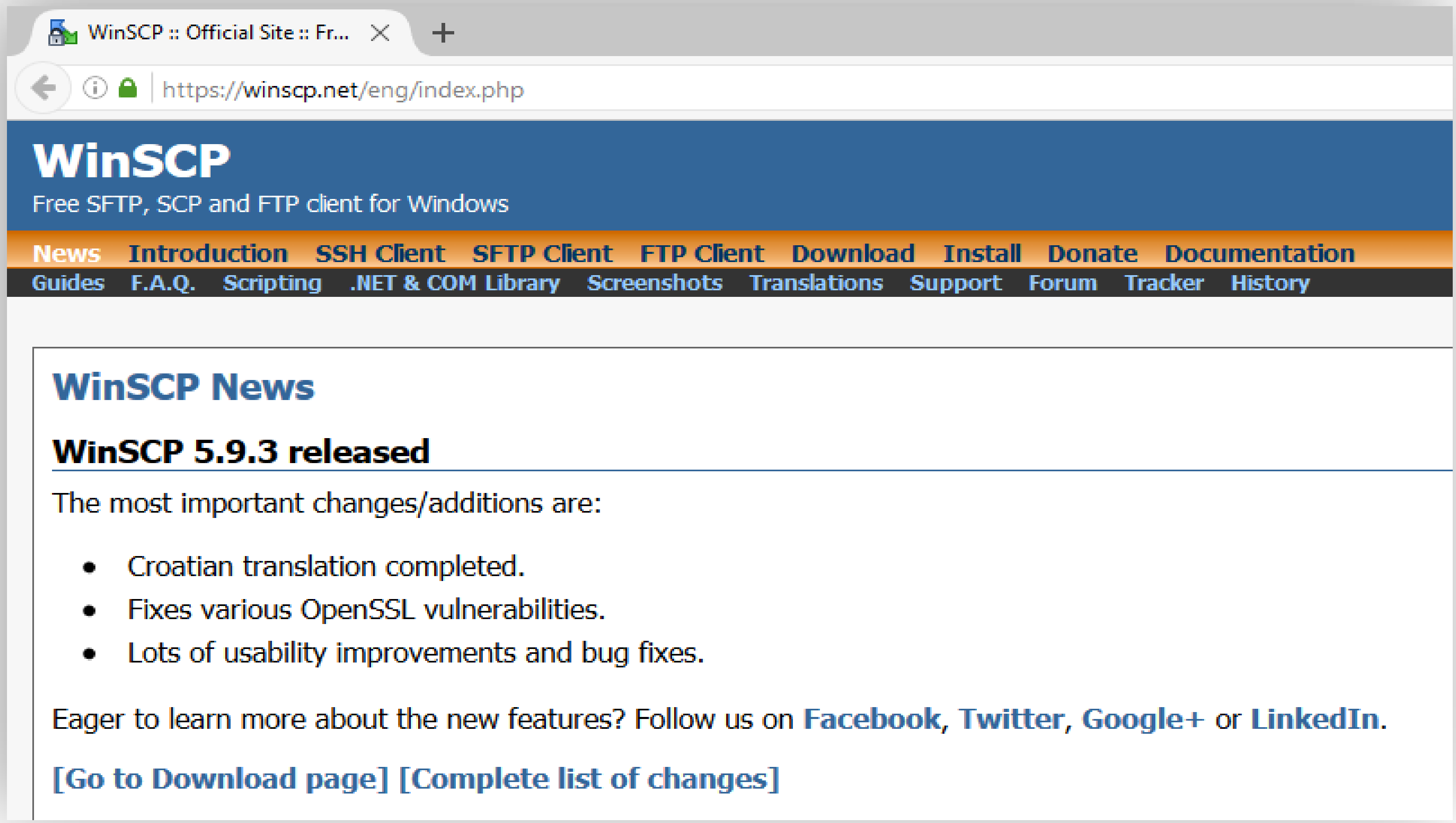
```
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:140 (140.0 b)  TX bytes:140 (140.0 b)

[ec2-user@ip-172-31-21-112 ~]$ ping 172.31.48.132
PING 172.31.48.132 (172.31.48.132) 56(84) bytes of data.
64 bytes from 172.31.48.132: icmp_seq=1 ttl=255 time=1.00 ms
64 bytes from 172.31.48.132: icmp_seq=2 ttl=255 time=1.07 ms
64 bytes from 172.31.48.132: icmp_seq=3 ttl=255 time=1.08 ms
64 bytes from 172.31.48.132: icmp_seq=4 ttl=255 time=1.06 ms
64 bytes from 172.31.48.132: icmp_seq=5 ttl=255 time=1.04 ms
64 bytes from 172.31.48.132: icmp_seq=6 ttl=255 time=1.01 ms
64 bytes from 172.31.48.132: icmp_seq=7 ttl=255 time=1.14 ms
^C
--- 172.31.48.132 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6007ms
rtt min/avg/max/mdev = 1.006/1.063/1.148/0.061 ms
[ec2-user@ip-172-31-21-112 ~]$ 
```



```
PING 172.31.21.112 (172.31.21.112) 56(84) bytes of data.
^C
--- 172.31.21.112 ping statistics ---
21 packets transmitted, 0 received, 100% packet loss, time 20159ms

[ec2-user@ip-172-31-48-132 ~]$ ping 172.31.21.112
PING 172.31.21.112 (172.31.21.112) 56(84) bytes of data.
64 bytes from 172.31.21.112: icmp_seq=1 ttl=255 time=0.926 ms
64 bytes from 172.31.21.112: icmp_seq=2 ttl=255 time=1.07 ms
64 bytes from 172.31.21.112: icmp_seq=3 ttl=255 time=0.997 ms
64 bytes from 172.31.21.112: icmp_seq=4 ttl=255 time=1.08 ms
64 bytes from 172.31.21.112: icmp_seq=5 ttl=255 time=1.06 ms
64 bytes from 172.31.21.112: icmp_seq=6 ttl=255 time=1.03 ms
64 bytes from 172.31.21.112: icmp_seq=7 ttl=255 time=0.974 ms
64 bytes from 172.31.21.112: icmp_seq=8 ttl=255 time=1.05 ms
64 bytes from 172.31.21.112: icmp_seq=9 ttl=255 time=1.07 ms
64 bytes from 172.31.21.112: icmp_seq=10 ttl=255 time=0.984 ms
64 bytes from 172.31.21.112: icmp_seq=11 ttl=255 time=1.08 ms
64 bytes from 172.31.21.112: icmp_seq=12 ttl=255 time=1.04 ms
^C
--- 172.31.21.112 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 11011ms
rtt min/avg/max/mdev = 0.926/1.032/1.086/0.052 ms
[ec2-user@ip-172-31-48-132 ~]$ 
```

# Transfer Files

Let us using **WinSCP**
https://winscp.net/eng/download.php
https://winscp.net/download/WinSCP-5.9.3-Setup.exe



WinSCP :: Official Site :: Fr...
https://winscp.net/eng/index.php
WinSCP
Free SFTP, SCP and FTP client for Windows
News Introduction SSH Client SFTP Client FTP Client Download Install Donate Documentation
Guides F.A.Q. Scripting .NET & COM Library Screenshots Translations Support Forum Tracker History

WinSCP News

**WinSCP 5.9.3 released**

The most important changes/additions are:

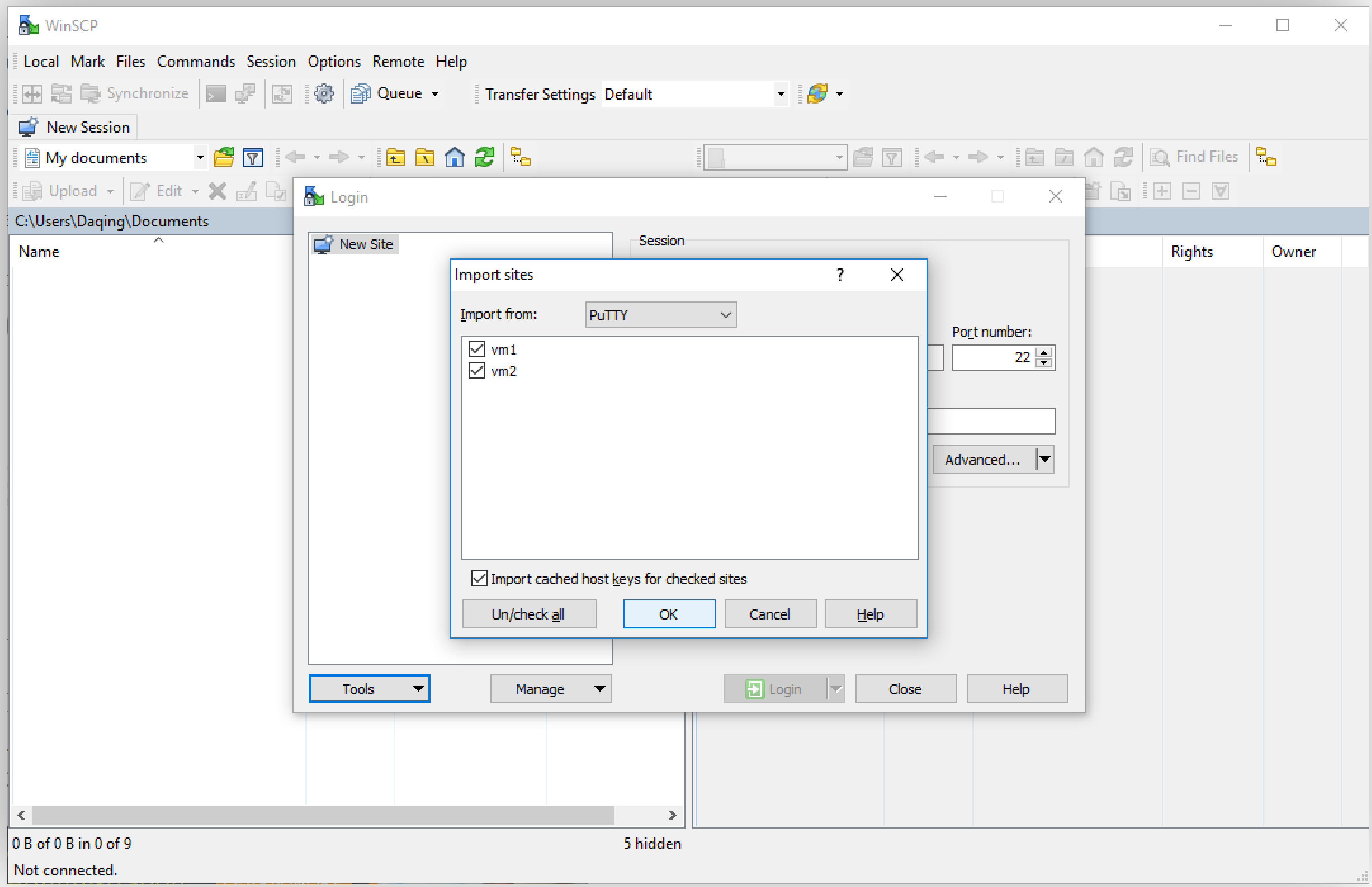
- Croatian translation completed.
- Fixes various OpenSSL vulnerabilities.
- Lots of usability improvements and bug fixes.

Eager to learn more about the new features? Follow us on Facebook, Twitter, Google+ or LinkedIn.

[Go to Download page] [Complete list of changes]

# Transfer Files

Tools → Import Sites → OK



C:\Users\Daqing\Documents

Import sites

Import from: PuTTY

☑ vm1
☑ vm2

☑ Import cached host keys for checked sites

Port number: 22

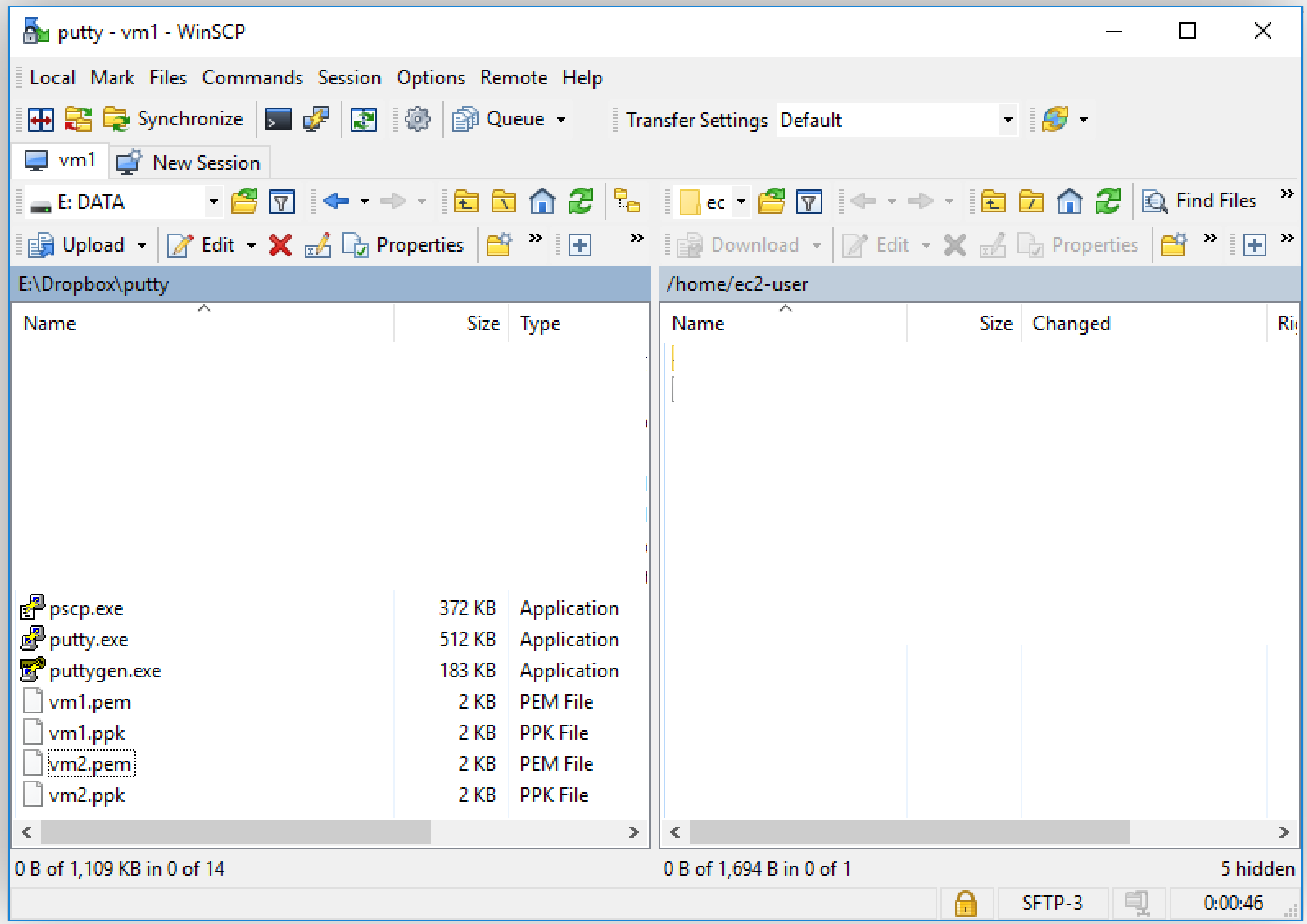
0 B of 0 B in 0 of 9

5 hidden

Not connected.

# Transfer Files

Upload the key vm2.pem to VM vm1



E:\Dropbox\putty

| Name | Size | Type |
|---|---|---|
| pscp.exe | 372 KB | Application |
| putty.exe | 512 KB | Application |
| puttygen.exe | 183 KB | Application |
| vm1.pem | 2 KB | PEM File |
| vm1.ppk | 2 KB | PPK File |
| vm2.pem | 2 KB | PEM File |
| vm2.ppk | 2 KB | PPK File |

/home/ec2-user

| Name | Size | Changed | Ri |
|---|---|---|---|

0 B of 1,109 KB in 0 of 14

0 B of 1,694 B in 0 of 1

5 hidden

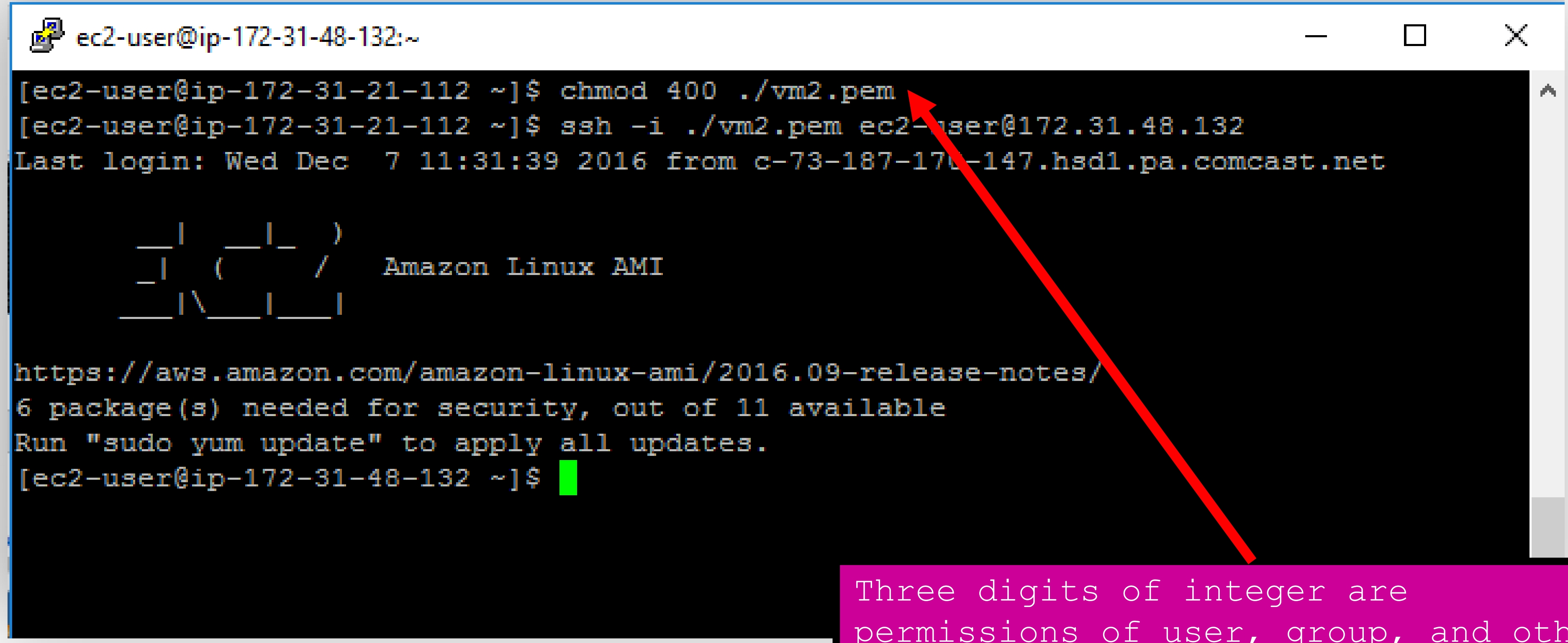
SFTP-3

0:00:46

# Remote Access

Log into vm2 from vm1

```
chmod 400 ./vm2.pem
ssh -i ./vm2.pem ec2-user@172.31.48.132
```



```
[ec2-user@ip-172-31-21-112 ~]$ chmod 400 ./vm2.pem
[ec2-user@ip-172-31-21-112 ~]$ ssh -i ./vm2.pem ec2-user@172.31.48.132
Last login: Wed Dec  7 11:31:39 2016 from c-73-187-170-147.hsd1.pa.comcast.net

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
6 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-48-132 ~]$
```

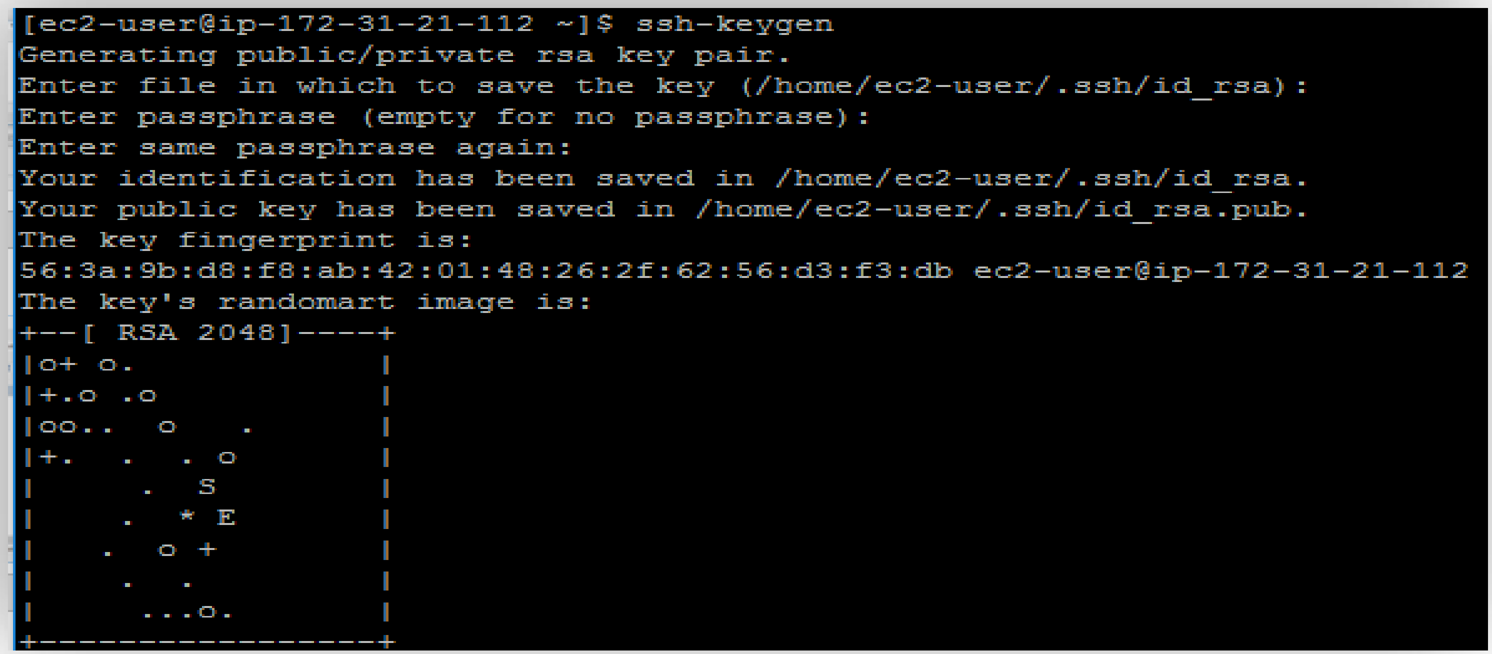
```
Three digits of integer are
permissions of user, group, and others
Each digit can be: 4, 2, 1, 0
4: read; 2: write; 1: execute;
0: no permission
Each digit is a combination of the
numbers 4, 2, 1, and 0
```

# Remote Access

## Log in “without” specifying a key/password

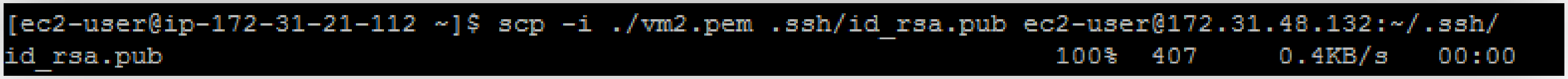
Remember we’ve uploaded vm2.pem to vm1, and we are able to log into vm2 from vm1

Now at vm1, generate public/private key pairs using **ssh-keygen**



```
[ec2-user@ip-172-31-21-112 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ec2-user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ec2-user/.ssh/id_rsa.
Your public key has been saved in /home/ec2-user/.ssh/id_rsa.pub.
The key fingerprint is:
56:3a:9b:d8:f8:ab:42:01:48:26:2f:62:56:d3:f3:db ec2-user@ip-172-31-21-112
The key's randomart image is:
+--[ RSA 2048]----+
|o+ o.            |
|+.o .o           |
|oo..  o   .      |
|+.  .  . o       |
|     .  S        |
|    .  * E       |
|   .  o +        |
|    .  .         |
|     ...o.       |
+-----------------+
```

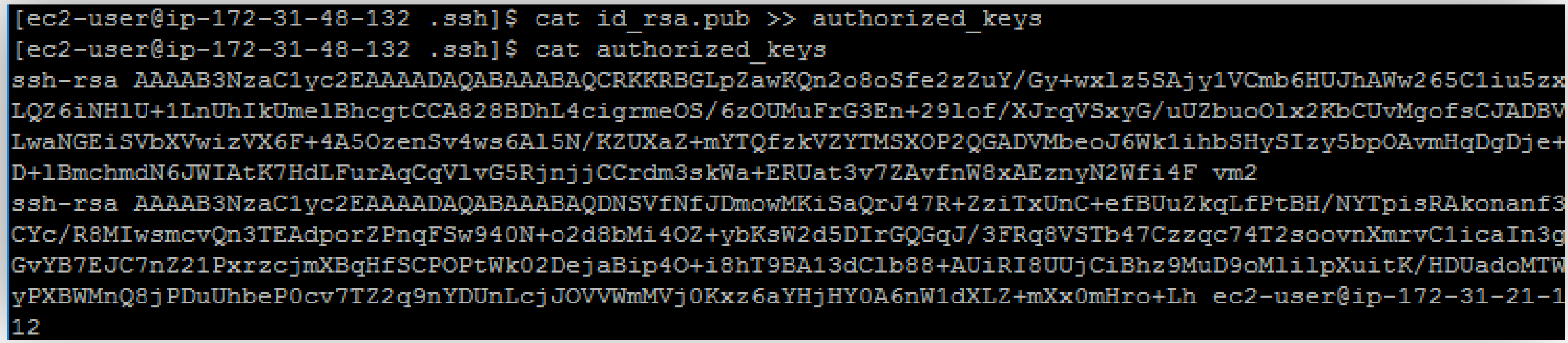
Copy vm1’s public key to vm2 using **scp** command



```
[ec2-user@ip-172-31-21-112 ~]$ scp -i ./vm2.pem .ssh/id_rsa.pub ec2-user@172.31.48.132:~/.ssh/
id_rsa.pub                                                          100%  407     0.4KB/s   00:00
```

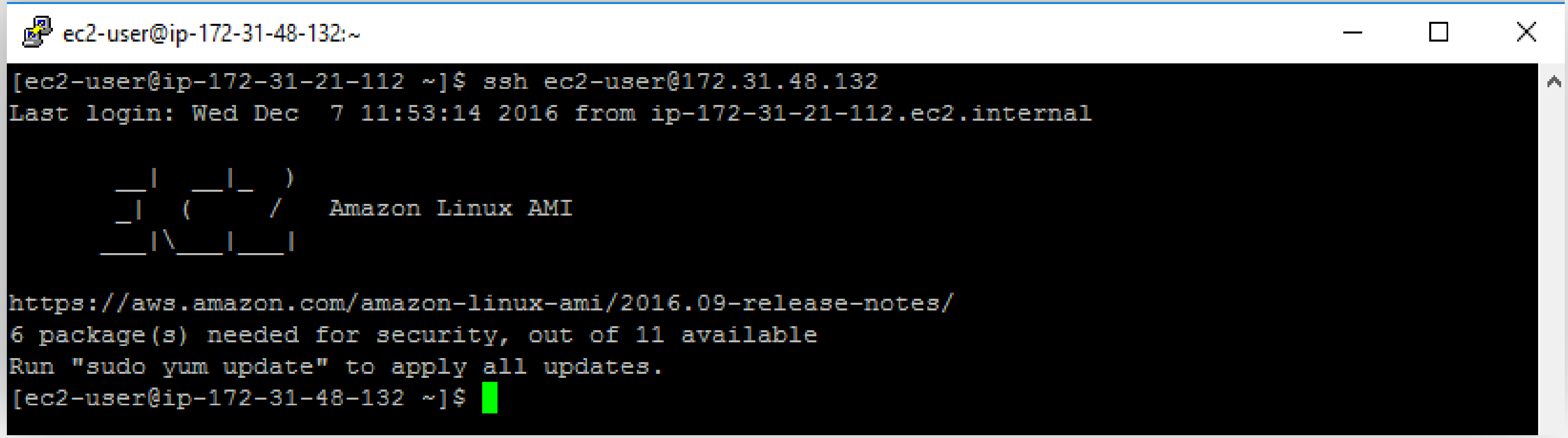
# Remote Access

Now at vm2, put vm1’s public key into **authorized_keys**



```
[ec2-user@ip-172-31-48-132 .ssh]$ cat id_rsa.pub >> authorized_keys
[ec2-user@ip-172-31-48-132 .ssh]$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCRKKRBGLpZawKQn2o8oSfe2zZuY/Gy+wxlz5SAjy1VCmb6HUJhAWw265C1iu5zx
LQZ6iNH1U+1LnUhIkUmelBhcgtCCA828BDhL4cigrmeOS/6zOUMuFrG3En+29lof/XJrqVSxyG/uUZbuoOlx2KbCUvMgofsCJADBV
LwaNGEiSVbXVwizVX6F+4A5OzenSv4ws6Al5N/KZUXaZ+mYTQfzkVZYTMSXOP2QGADVMbeoJ6Wk1ihbSHySIzy5bpOAvmHqDgDje+
D+lBmchmdN6JWIAtK7HdLFurAqCqVlvG5RjnjjCCrdm3skWa+ERUat3v7ZAvfnW8xAEznyN2Wfi4F vm2
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDNSVfNfJDmowMKiSaQrJ47R+ZziTxUnC+efBUuZkqLfPtBH/NYTpisRAkonanf3
CYc/R8MIwsmcvQn3TEAdporZPnqFSw940N+o2d8bMi4OZ+ybKsW2d5DIrGQGqJ/3FRq8VSTb47Czzqc74T2soovnXmrvC1icaIn3g
GvYB7EJC7nZ21PxrzcjmXBqHfSCPOPtWk02DejaBip4O+i8hT9BA13dClb88+AUiRI8UUjCiBhz9MuD9oM1ilpXuitK/HDUadoMTW
yPXBWMnQ8jPDuUhbeP0cv7TZ2q9nYDUnLcjJOVVWmMVj0Kxz6aYHjHY0A6nW1dXLZ+mXx0mHro+Lh ec2-user@ip-172-31-21-1
12
```

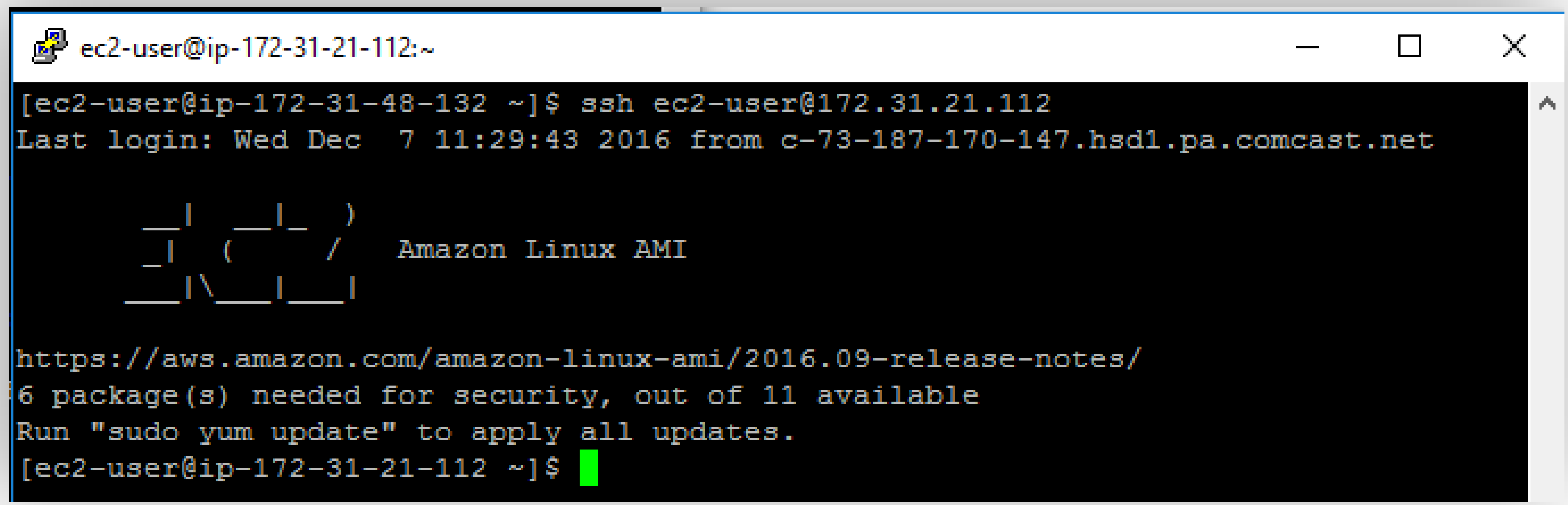
Now at vm1, log into vm2 “without” specifying a key



```
[ec2-user@ip-172-31-21-112 ~]$ ssh ec2-user@172.31.48.132
Last login: Wed Dec  7 11:53:14 2016 from ip-172-31-21-112.ec2.internal

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
6 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-48-132 ~]$ 
```

# Remote Access

Do similar things to enable remote log into vm1 from vm2 “without” specifying a key



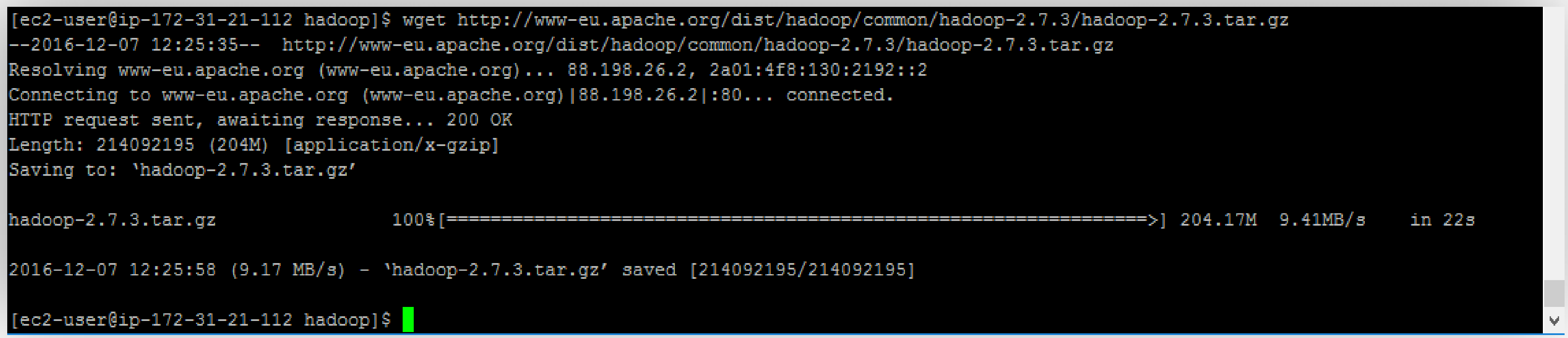
```
[ec2-user@ip-172-31-48-132 ~]$ ssh ec2-user@172.31.21.112
Last login: Wed Dec  7 11:29:43 2016 from c-73-187-170-147.hsd1.pa.comcast.net

       __|  __|_  )
       _|  (     /   Amazon Linux AMI
      ___|\___|___|

https://aws.amazon.com/amazon-linux-ami/2016.09-release-notes/
6 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-21-112 ~]$ 
```

# Hadoop

Download Hadoop, let us download the binary package
http://www-eu.apache.org/dist/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz

You can download it into your Windows first, and then transfer it to the virtual machine

Using **wget** command on the virtual machine:

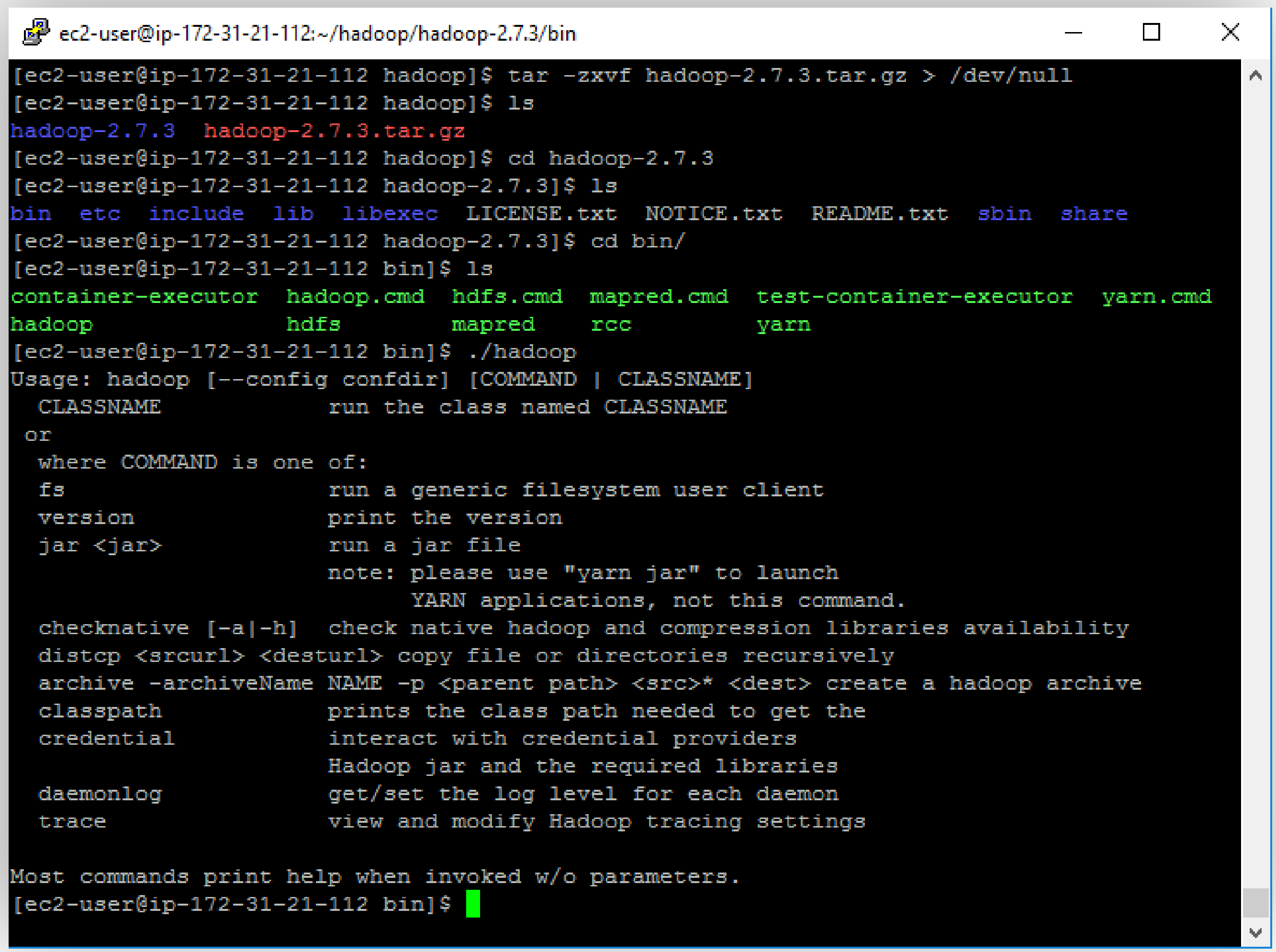


```
[ec2-user@ip-172-31-21-112 hadoop]$ wget http://www-eu.apache.org/dist/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz
--2016-12-07 12:25:35--  http://www-eu.apache.org/dist/hadoop/common/hadoop-2.7.3/hadoop-2.7.3.tar.gz
Resolving www-eu.apache.org (www-eu.apache.org)... 88.198.26.2, 2a01:4f8:130:2192::2
Connecting to www-eu.apache.org (www-eu.apache.org)|88.198.26.2|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 214092195 (204M) [application/x-gzip]
Saving to: 'hadoop-2.7.3.tar.gz'

hadoop-2.7.3.tar.gz                100%[================================================================>] 204.17M  9.41MB/s    in 22s

2016-12-07 12:25:58 (9.17 MB/s) - 'hadoop-2.7.3.tar.gz' saved [214092195/214092195]

[ec2-user@ip-172-31-21-112 hadoop]$
```

# Hadoop

## Extract the downloaded package, and run it

```
tar -zxvf hadoop-2.7.3.tar.gz
cd hadoop-2.7.3/bin
./hadoop
```



```
[ec2-user@ip-172-31-21-112 hadoop]$ tar -zxvf hadoop-2.7.3.tar.gz > /dev/null
[ec2-user@ip-172-31-21-112 hadoop]$ ls
hadoop-2.7.3  hadoop-2.7.3.tar.gz
[ec2-user@ip-172-31-21-112 hadoop]$ cd hadoop-2.7.3
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ ls
bin  etc  include  lib  libexec  LICENSE.txt  NOTICE.txt  README.txt  sbin  share
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ cd bin/
[ec2-user@ip-172-31-21-112 bin]$ ls
container-executor  hadoop.cmd  hdfs.cmd  mapred.cmd  test-container-executor  yarn.cmd
hadoop              hdfs        mapred    rcc         yarn
[ec2-user@ip-172-31-21-112 bin]$ ./hadoop
Usage: hadoop [--config confdir] [COMMAND | CLASSNAME]
  CLASSNAME            run the class named CLASSNAME
 or
  where COMMAND is one of:
  fs                   run a generic filesystem user client
  version              print the version
  jar <jar>            run a jar file
                       note: please use "yarn jar" to launch
                             YARN applications, not this command.
  checknative [-a|-h]  check native hadoop and compression libraries availability
  distcp <srcurl> <desturl> copy file or directories recursively
  archive -archiveName NAME -p <parent path> <src>* <dest> create a hadoop archive
  classpath            prints the class path needed to get the
  credential           interact with credential providers
                       Hadoop jar and the required libraries
  daemonlog            get/set the log level for each daemon
  trace                view and modify Hadoop tracing settings

Most commands print help when invoked w/o parameters.
[ec2-user@ip-172-31-21-112 bin]$
```

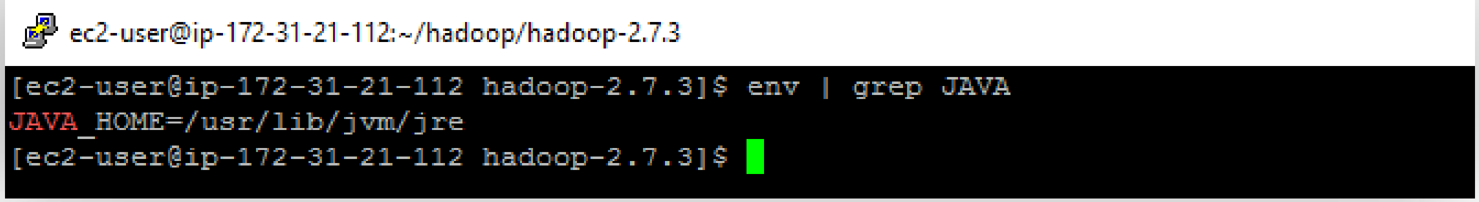
# Hadoop

## Setting up a Single Node Cluster

[http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/SingleCluster.html](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/SingleCluster.html)

## Make sure you have JAVA installed

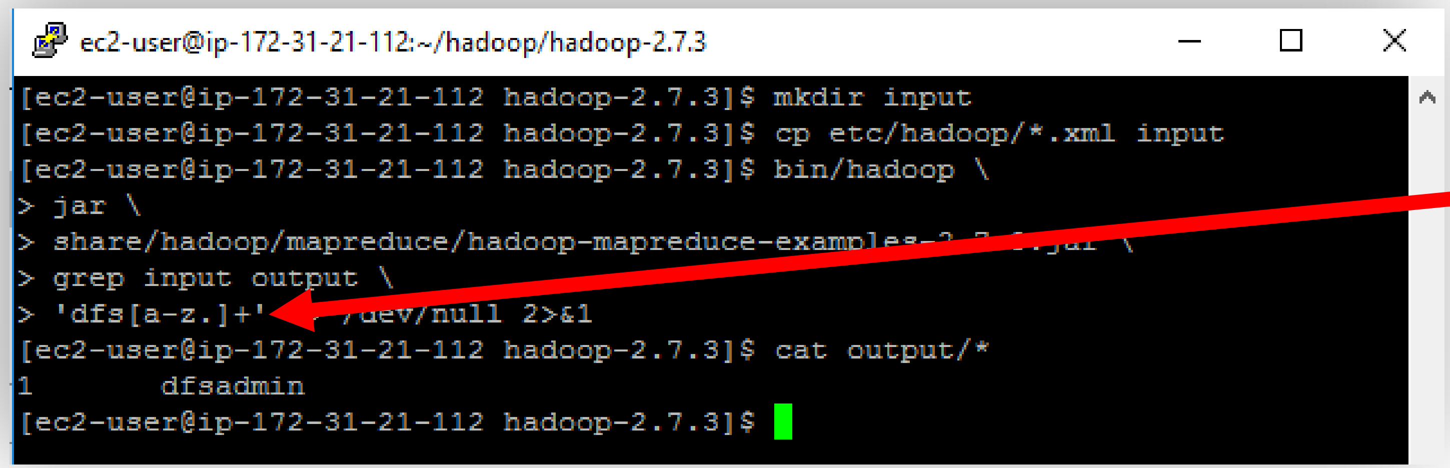
```
env | grep JAVA
```



```
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ env | grep JAVA
JAVA_HOME=/usr/lib/jvm/jre
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$
```

## Standalone operation

[http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/SingleCluster.html#Standalone_Operation](http://hadoop.apache.org/docs/current/hadoop-project-dist/hadoop-common/SingleCluster.html#Standalone_Operation)



```
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ mkdir input
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ cp etc/hadoop/*.xml input
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ bin/hadoop \
> jar \
> share/hadoop/mapreduce/hadoop-mapreduce-examples-2.7.3.jar \
> grep input output \
> 'dfs[a-z.]+' > /dev/null 2>&1
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$ cat output/*
1       dfsadmin
[ec2-user@ip-172-31-21-112 hadoop-2.7.3]$
```

**regular expression**