

# Lab: Amazon EC2

INF 551

Wensheng Wu

# Sign up for AWS



# Use credit card

- Make sure you use a credit card
- Do not use a debit card
  - Seems that Amazon requires additional approval for using debit card

# Choose Amazon Linux AMI (not the first one: "2 AMI")

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review


## Step 1: Choose an Amazon Machine Image (AMI)

### Quick Start

My AMIs

AWS Marketplace

Community AMIs

☒ Free tier only 

[Cancel and Exit](#)

1 to 19 of 19 AMIs



**Amazon Linux 2 AMI (HVM), SSD Volume Type** - ami-0d8f6eb4f641ef691 (64-bit x86) / ami-0f378490dca16e3f4 (64-bit Arm)

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)  
64-bit (Arm)



**Amazon Linux**  
Free tier eligible

**Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type** - ami-02f706d959cedf892

The Amazon Linux AMI is an EBS-backed, AWS-supported image. The default image includes AWS command line tools, Python, Ruby, Perl, and Java. The repositories include Docker, PHP, MySQL, PostgreSQL, and other packages.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

Select

64-bit (x86)



**Red Hat**

Free tier eligible

**Red Hat Enterprise Linux 8 (HVM), SSD Volume Type** - ami-0520e698dd500b1d1 (64-bit x86) / ami-0099847d600887c9f (64-bit Arm)

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Select

64-bit (x86)  
64-bit (Arm)

Choose this one

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

## Step 2: Choose an Instance Type

Filter by: All instance types Current generation Show/Hide Columns

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	General purpose	t2.xlarge	4	16	EBS only	-	Moderate	Yes

Cancel

Previous

Review and Launch

Next: Configure Instance Details

Then press "Review and Launch"

Create a key pair and save it in a folder where  
you will be accessing AWS from

## Select an existing key pair or create a new key pair



A key pair consists of a **public key** that AWS stores, and a **private key file** that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.

Note: The selected key pair will be added to the set of keys authorized for this instance. [Learn more about removing existing key pairs from a public AMI.](#)

Create a new key pair

**Key pair name**

inf551-fa19

Download Key Pair



You have to download the **private key file** (\*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel

Launch Instances

## Connect To Your Instance



I would like to connect with

- ☐ A standalone SSH client ⓘ
- ☐ EC2 Instance Connect (browser-based SSH connection) ⓘ
- ☐ 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 (inf55x-fa19.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 inf55x-fa19.pem
```

4. Connect to your instance using its Public DNS:

```
ec2-52-15-153-77.us-east-2.compute.amazonaws.com
```

Example:

```
ssh -i "inf55x-fa19.pem" ec2-user@ec2-52-15-153-77.us-east-2.compute.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.

You may need to set up this too  
If you are using Windows;  
Or use Cygwin (see later)

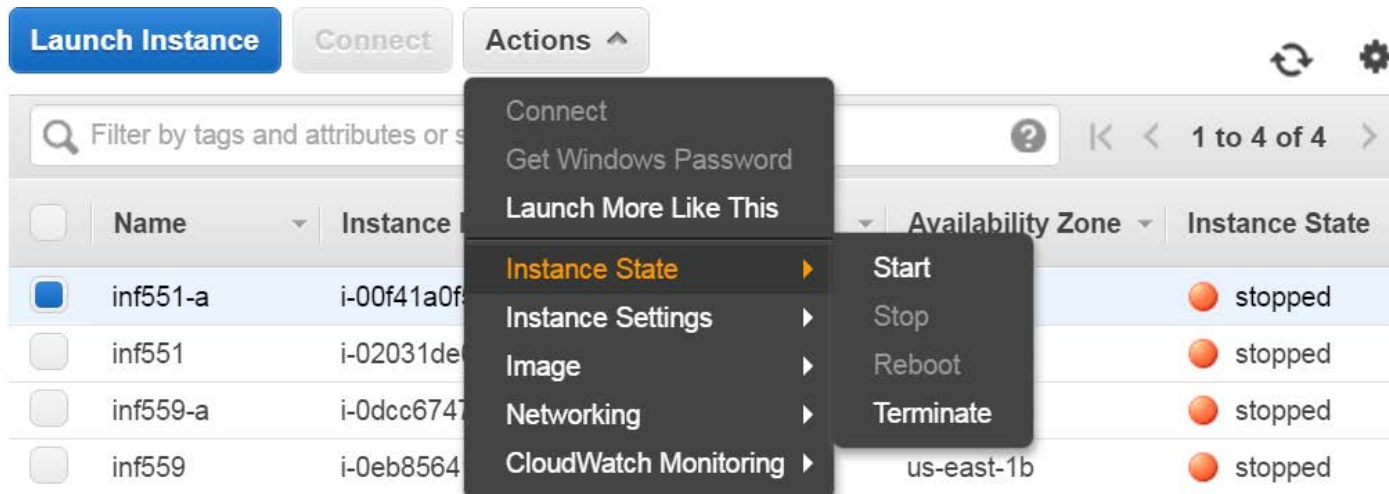
# Change permission of key file

- If you see the error message:
  - Load key "inf55x-fa19.pem": bad permissions
- Execute this:
  - `chmod 400 inf55x-fa19.pem`



# Start and stop instance

- Remember to stop the instance
  - When you are not using it
- Save energy and avoid bills



# Host address

- Host address of your instance may change
  - When you stop and restart the instance
- You need to ssh to the new host address

# Install SSH client

- Windows:
  - Option 1: Install Cygwin and choose openssh
  - Option 2: Install putty package
    - <http://tartarus.org/~simon/putty-snapshots/x86/putty-installer.msi>
- iOS
  - Mac OS comes with ssh client preinstalled

# Cygwin

# Cygwin

Get that Linux feeling - on Windows

- First install Cygwin ([www.cygwin.com](http://www.cygwin.com))

## What...

### ...is it?

Cygwin is:

- a large collection of GNU and Open Source tools which provide functionality similar to a [Linux distribution](#) on Windows.
- a DLL (cygwin1.dll) which provides substantial POSIX API functionality.

### ...isn't it?

Cygwin is not:

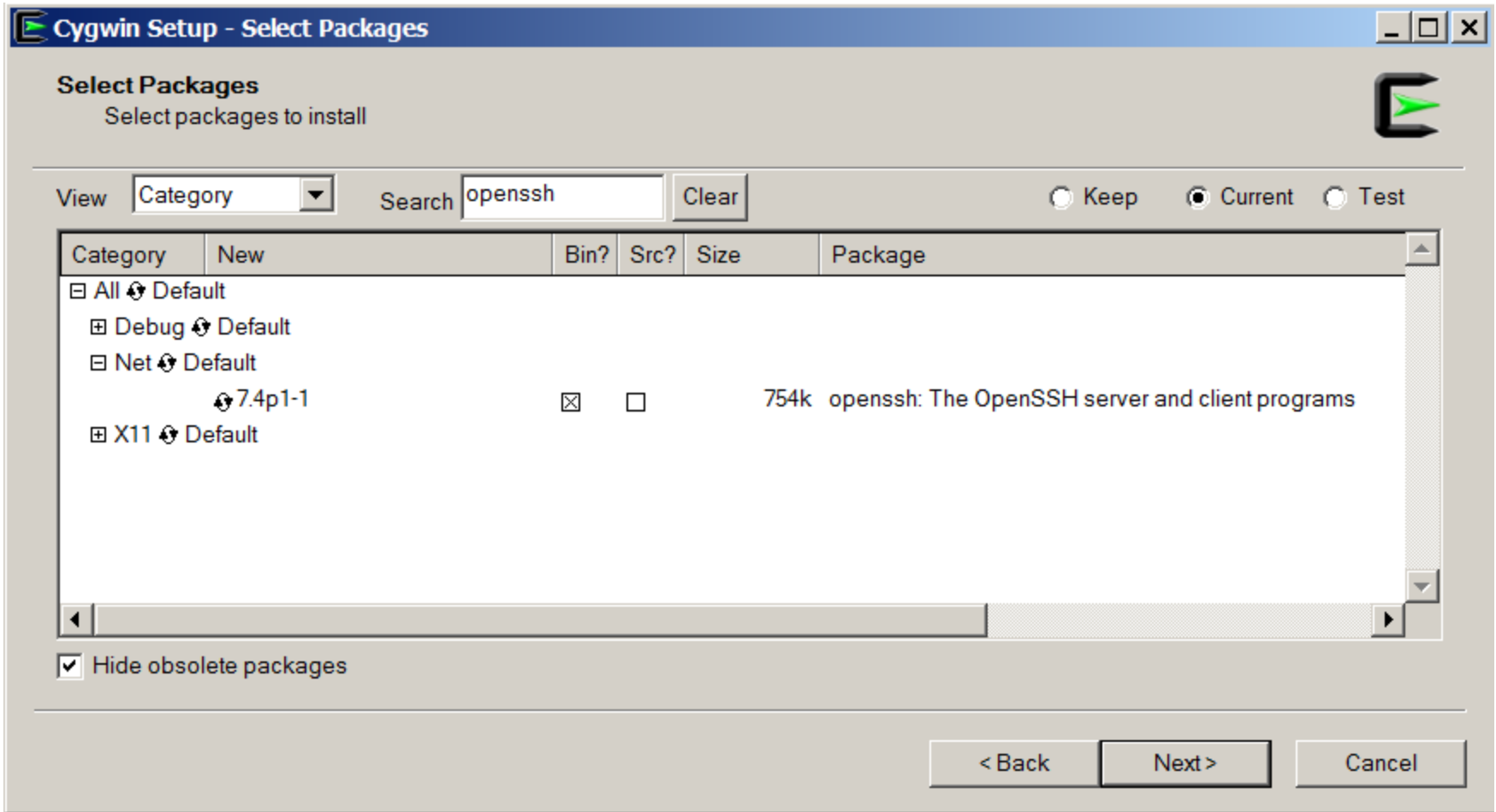
- a way to run native Linux apps on Windows. You must rebuild your application *from source* if you want it to run on Windows.
- a way to magically make native Windows apps aware of UNIX® functionality like signals, ptys, etc. Again, you need to build your apps *from source* if you want to take advantage of Cygwin functionality.

Choose either this (if your OS is 64bit) or this

## Current Cygwin DLL version

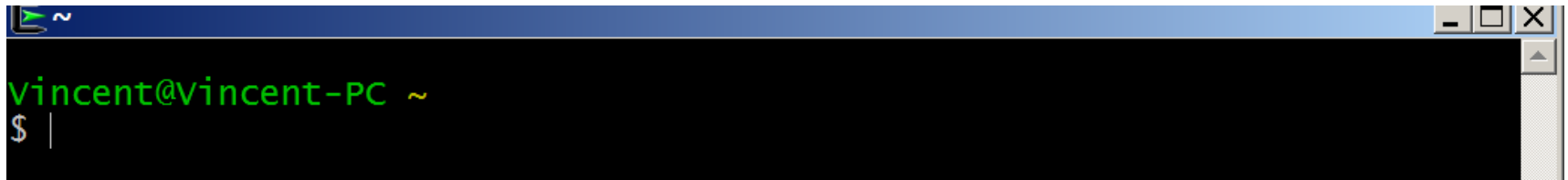
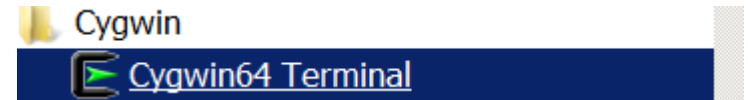
The most recent version of the Cygwin DLL is [2.6.1](#). Install it by running [setup-x86.exe](#) (32-bit installation) or [setup-x86\\_64.exe](#) (64-bit installation).

# Make sure you select "openssh"



# Start Cygwin

- Once installed, look for Cygwin program folder in your list of programs
- Select "Cygwin64 Terminal"
  - This starts a bash command line window like below
  - Note by default your home directory ~ is located in c:\cygwin64\home\<your user id>



# Cygwin64 Terminal

- A Linux bash shell
  - Note it uses forward slashes
  - E.g., `cd usc/551/551-sp17`
- But it accepts Windows style path (if quoted)
  - E.g., `cd "c:\cygwin64"`

# Log onto EC2 from Cygwin

- `ssh -i <your identify file.pem> ec2-user@ec2-[your ec2 instance ip].compute-1.amazonaws.com`
  - Replace ssh above with sftp for file transfer
- Note: pem file is used here, no need to convert it to ppk file as in Putty



# Log into EC2 instance via Cygwin

```
ec2-user@ip-172-31-52-194:~  
vincent@Vincent-PC ~/usc/551/551-fa16/Amazon  
$ ssh -i "inf551.pem" ec2-user@ec2-54-224-125-6.compute-1.amazonaws.com  
Last login: Wed Jan  4 01:20:44 2017 from cpe-174-108-65-35.carolina.res  
.rr.com  
  
  _|  _|  _|  
  _|  (  _|  _|  
  _|  \  _|  _|  
Amazon Linux AMI  
  
https://aws.amazon.com/amazon-linux-ami/2016.03-release-notes/  
35 package(s) needed for security, out of 109 available  
Run "sudo yum update" to apply all updates.  
Amazon Linux version 2016.09 is available.  
[ec2-user@ip-172-31-52-194 ~]$ ls  
apache-cassandra-2.2.8  download  local-out  temp  
apache-hive-2.1.0-bin  hadoop-2.7.3  metastore_db  
derby.log              inf551      spark-2.0.1-bin-hadoop2.7  
[ec2-user@ip-172-31-52-194 ~]$ |
```

# First time log in...

```
Vincent@Vincent-PC ~/usc/551/551-fa16/Amazon
$ ssh -i "inf551.pem" ec2-user@ec2-54-173-96-53.compute-1.amazonaws.com
The authenticity of host 'ec2-54-173-96-53.compute-1.amazonaws.com (54.173.96.53)' can't be established.
ECDSA key fingerprint is SHA256:jY9qPXiec94tsH/A2pVN0v1Pb9qkyUG1b9hCbHtcMfo.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-54-173-96-53.compute-1.amazonaws.com,54.173.96.53' (ECDSA) to the list of known hosts.
Last login: Thu Jan  5 01:34:30 2017 from cpe-174-108-65-35.carolina.res.rr.com

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


https://aws.amazon.com/amazon-linux-ami/2016.03-release-notes/
38 package(s) needed for security, out of 112 available
Run "sudo yum update" to apply all updates.
Amazon Linux version 2016.09 is available.
[ec2-user@ip-172-31-52-194 ~]$
```

# Connect to instance using Putty

- Instructions on how to connect from Windows using PuTTY
  - [https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html?icmpid=docs\\_ec2\\_console](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html?icmpid=docs_ec2_console)

# Convert key to Putty format

- Converting Your Private Key Using PuTTYgen
  - [https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html?icmpid=docs\\_ec2\\_console](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/putty.html?icmpid=docs_ec2_console)

PuTTY Key Generator

?

FileKeyConversionsHelp

Key

Public key for pasting into OpenSSH authorized\_keys file:

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDNy  
+3U4p7FuPieVAA5Wn57ignInfg7ueTHDSKDFc6ggQCEBOyCM0DfH7NImNqdLQkG0  
mQ8+jG7TwIJVLDXa5Em6x6l8xYjA4qD8BoS/CxIXkknioW1+MF0V3CzrBnipq4dXZMBz  
AI9tPkQrA9K/jJe2uBSUGEiBnnwuF2N6DwWKuSO/37TrU55r9o2yR0I1HFUFN3Q4W4v  
3pGPRFhKNMo3C7sJbpBIXxAXAkCFIQDFs0bFz9INgubaWq7nLSsVwn84dt0aw5GY

Key fingerprint:ssh-rsa 2048 be:e8:b0:c0:3e:2a:f0:7e:cd:c9:35:11:6d:46:85:93

Key comment:imported-openssh-key

Key passphrase:

Confirm passphrase:

Actions

Generate a public/private key pair

Generate

Load an existing private key file

Load

Save the generated key

Save public key

Save private key

Parameters

Type of key to generate:

☒ RSA

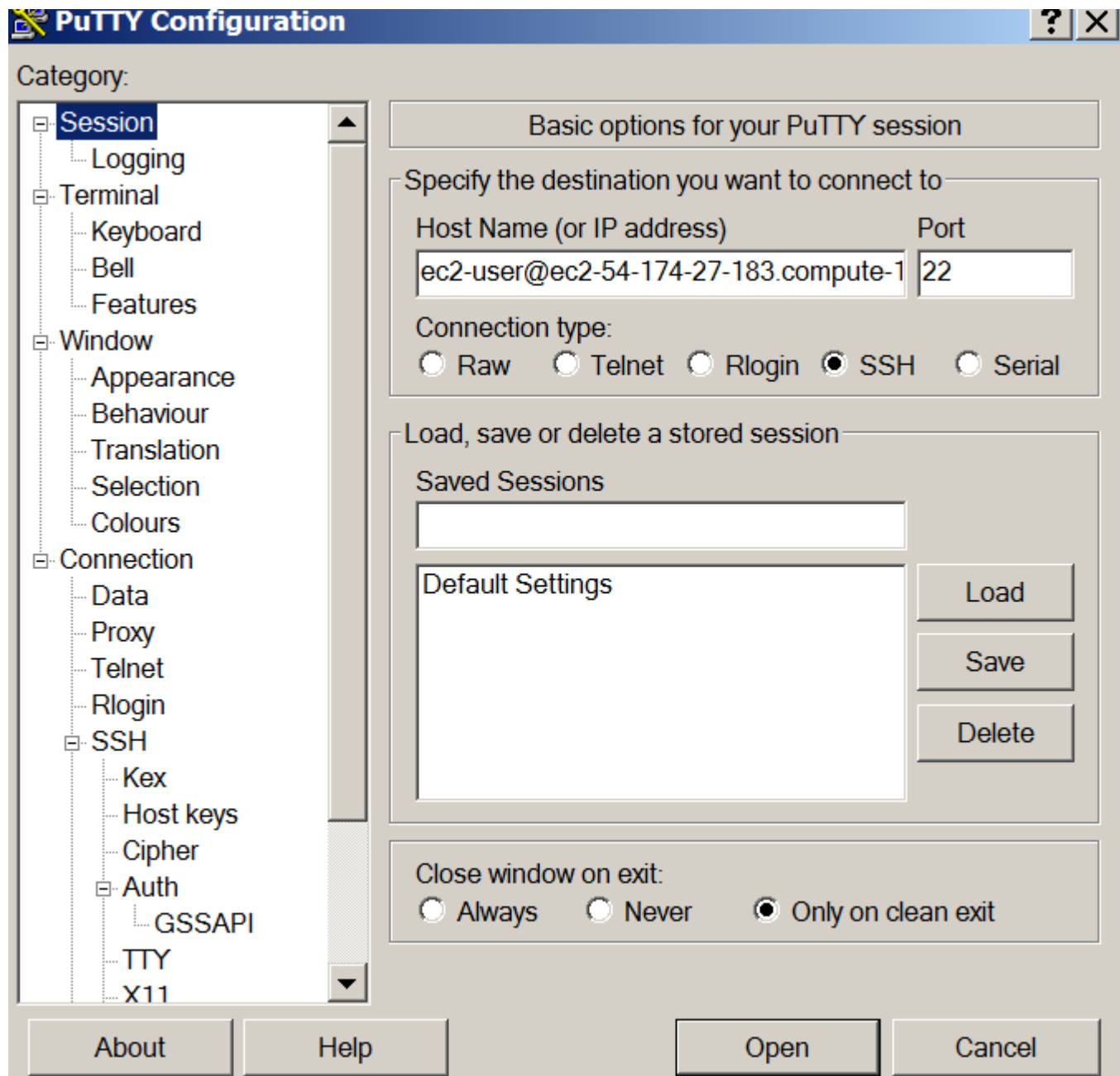
☐ DSA

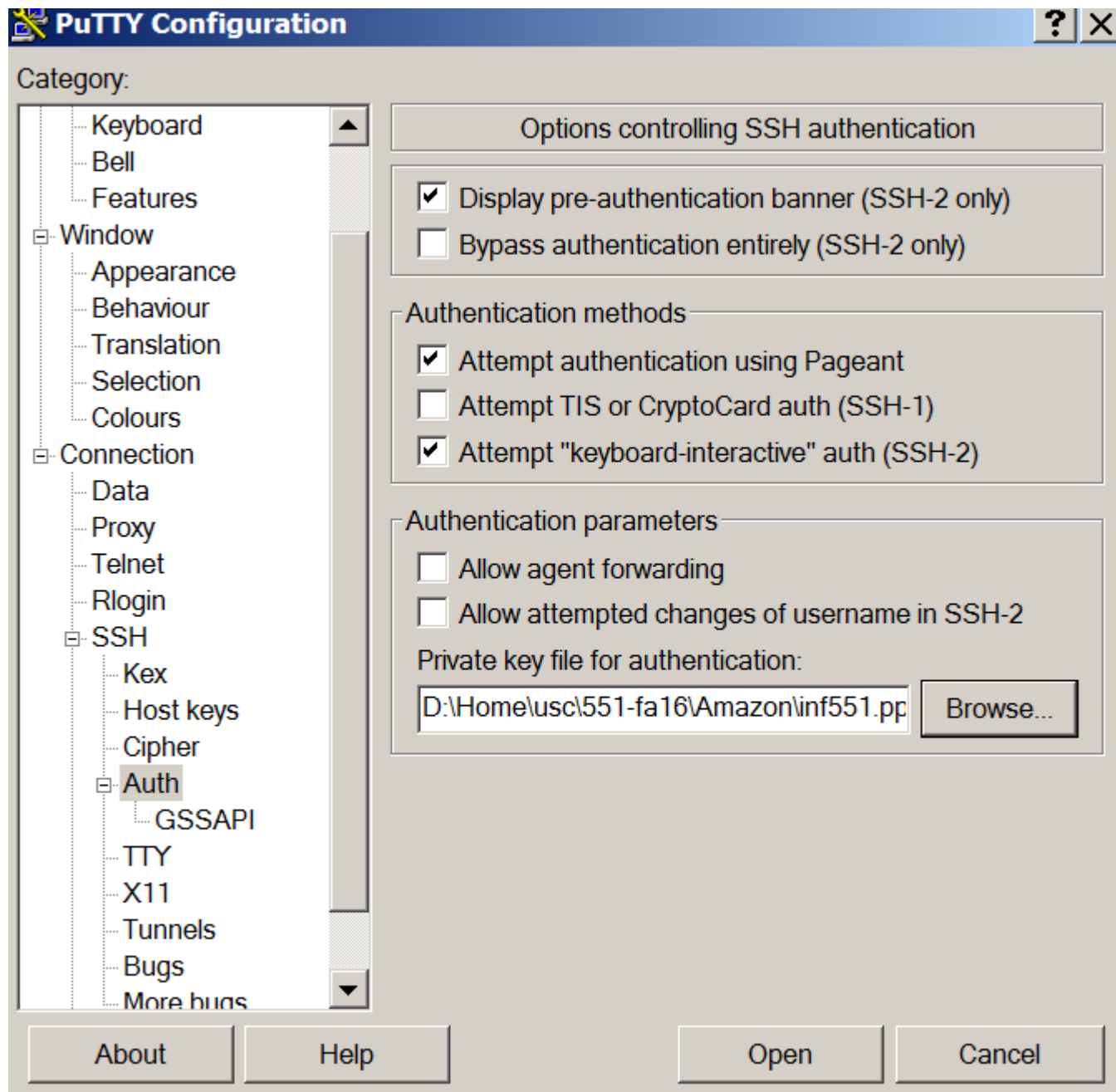
☐ ECDSA

☐ ED25519

☐ SSH-1 (RSA)

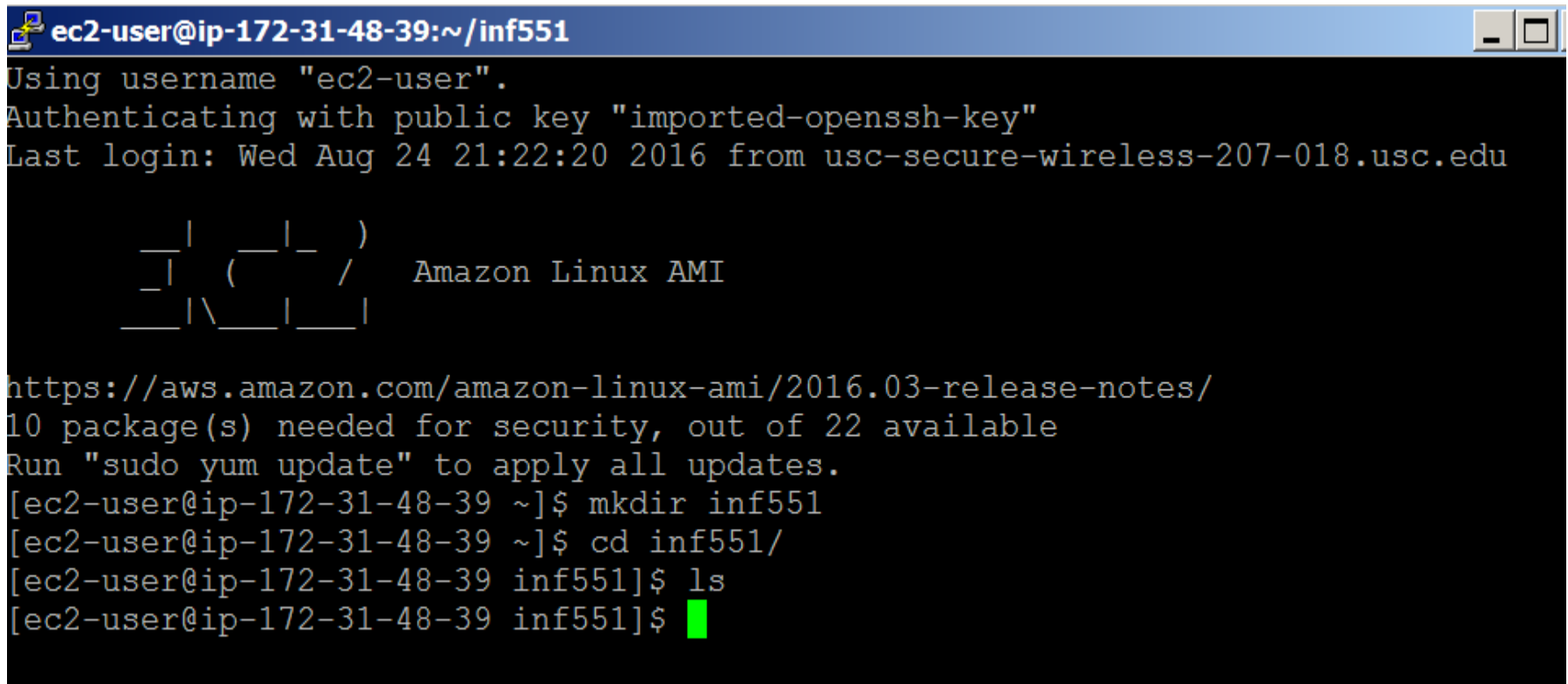
Number of bits in a generated key:2048





# Connected

- Create a directory called inf551



```
ec2-user@ip-172-31-48-39:~/inf551
Using username "ec2-user".
Authenticating with public key "imported-openssh-key"
Last login: Wed Aug 24 21:22:20 2016 from usc-secure-wireless-207-018.usc.edu

  ____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|
  ____|  (____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|
  ____|  \____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|  ____|

Amazon Linux AMI

https://aws.amazon.com/amazon-linux-ami/2016.03-release-notes/
10 package(s) needed for security, out of 22 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-48-39 ~]$ mkdir inf551
[ec2-user@ip-172-31-48-39 ~]$ cd inf551/
[ec2-user@ip-172-31-48-39 inf551]$ ls
[ec2-user@ip-172-31-48-39 inf551]$
```



# Submission to blackboard

- A screenshot like previous slide, showing
  - You are successfully connected
  - You have created a directory called inf551

# Extra

- Update pre-installed packages
  - `sudo yum update`

```
curl.x86_64 0:7.40.0-8.59.amzn1
dracut.noarch 0:004-409.31.amzn1
kernel-tools.x86_64 0:4.4.16-27.56.amzn1
libcurl.x86_64 0:7.40.0-8.59.amzn1
libevent.x86_64 0:2.0.21-4.19.amzn1
ntp.x86_64 0:4.2.6p5-41.32.amzn1
ntpd.x86_64 0:4.2.6p5-41.32.amzn1
openssl.x86_64 1:1.0.1k-15.93.amzn1
python27.x86_64 0:2.7.10-4.122.amzn1
python27-boto.noarch 0:2.42.0-1.1.amzn1
python27-botocore.noarch 0:1.4.46-1.58.amzn1
python27-devel.x86_64 0:2.7.10-4.122.amzn1
python27-libs.x86_64 0:2.7.10-4.122.amzn1
tzdata.noarch 0:2016f-1.63.amzn1
tzdata-java.noarch 0:2016f-1.63.amzn1
vim-common.x86_64 2:7.4.1967-1.42.amzn1
vim-enhanced.x86_64 2:7.4.1967-1.42.amzn1
vim-filesystem.x86_64 2:7.4.1967-1.42.amzn1
vim-minimal.x86_64 2:7.4.1967-1.42.amzn1
wget.x86_64 0:1.18-1.18.amzn1
```

Complete!

# Get familiar with the instance

- It has the following preinstalled
  - nano (text editor)
  - vi
  - python
  - curl
  - perl
  - ssh
  - wget
  - java

# sftp/psftp

- sftp: secure file transfer
  - psftp: putty version of sftp
- Tutorials that may be useful
  - <https://kb.iu.edu/d/akqg>
  - <https://www.digitalocean.com/community/tutorials/how-to-use-sftp-to-securely-transfer-files-with-a-remote-server>

# Python tutorials

- Learn Python - Free Interactive Python Tutorial  
– <http://www.learnpython.org/>
- The Python Tutorial — Python 3.5.2 documentation
- Google's Python Class | Python Education | Google Developers