

HANDS ON MATERI KOMPUTASI DI AWS

Pada modul ini akan dibahas mengenai layanan yang berkaitan dengan komputasi di AWS. Bagian pertama pada modul ini membahas deksripsi pengertian dan penjelasan lebih dalam mengenai layanan komputasi di AWS yaitu Amazon EC2, Beanstalk, Lambda, dan ECS. Selanjutnya, pada bagian kedua dijelaskan salah satu penerapan layanan komputasi di AWS yaitu Amazon EC2. Penjelasan tersebut meliputi langkah-langkah pengerjaan untuk menjalankan virtual mesin dan pengaturan sumber daya komputasi dengan sistem operasi Linux dan Windows. Kemudian, pada bagian terakhir modul ini terdapat latihan soal mengenai materi yang telah dibahas dalam modul.

DESKRIPSI

a. Amazon EC2 (Amazon Elastic Compute Cloud)

Amazon EC2 adalah sebuah *web service* yang disediakan oleh Amazon untuk membuat dan menjalankan mesin virtual pada *cloud*. Amazon EC2 menghilangkan kebutuhan untuk menyediakan perangkat keras sehingga pengguna dapat membuat aplikasi secara lebih cepat. Amazon EC2 menyediakan beberapa pilihan sistem operasi dan software packages seperti sistem operasi Linux dan Microsoft Windows Server. Amazon EC2 memungkinkan pengguna untuk memilih spesifikasi sumber daya komputasi sesuai dengan kebutuhan sistem operasi dan aplikasi yang akan digunakan. Sumber daya komputasi tersebut meliputi konfigurasi *memory*, CPU, *instance storage*, dan *boot partition size*. Setelah memilih spesifikasi sumber daya komputasi, pengguna akan dibebankan dengan biaya sewa yang sesuai pada spesifikasi dan kapasitas sumber daya komputasi yang telah dipilih.

b. Beanstalk (AWS Elastic Beanstalk)

Beanstalk merupakan salah satu layanan yang diberikan oleh AWS untuk memudahkan penerapan dan pembesaran sekala akses aplikasi web dan *web service* pada AWS. Beanstalk dapat mendukung aplikasi web dan *web service* yang dikembangkan menggunakan Java, .NET, PHP, Node.js, Python, Ruby, Go, dan Docker. Aplikasi dan *web service* tersebut berjalan pada server yang sudah dikenal seperti Apache, Nginx, Passenger, dan IIS. Setelah pengguna mengupload kode aplikasi web atau *web service* menuju AWS, kemudian Beanstalk akan secara otomatis menangani penyediaan kapasitas (*capacity provisioning*), keseimbangan beban (*load balancing*), penyebaran (*auto-scaling*), dan pemantauan kesehatan aplikasi (*application health*).

monitoring). Beanstalk dapat menangani maksimal beban kerja (*workload*) untuk meminimalkan biaya sewa. Tidak terdapat biaya tambahan untuk menggunakan layanan Beanstalk, pengguna hanya membayar biaya sewa sumber daya AWS yang diperlukan untuk menyimpan dan menjalankan aplikasi web atau *web service*.

c. Lambda

Lambda merupakan layanan AWS yang memungkinkan pengguna dapat menjalankan kode atau program tanpa menyediakan dan mengelola server. Pengguna hanya perlu membuat program dan mengupload program tersebut ke Lambda. Lambda akan menangani seluruh sumber daya komputasi yang dibutuhkan oleh program. Lambda dapat menjalankan hampir semua jenis aplikasi atau layanan backend tanpa pengelolaan sumber daya komputasi. Pengguna dapat mentrigger secara otomatis program yang telah dibuat dari layanan AWS lain atau menggilnya langsung dari aplikasi web atau mobile. Lambda tidak mengambil biaya sewa apabila program tidak berjalan. Pengguna hanya akan dikenakan biaya sewa apabila program berjalan sesuai dengan waktu komputasi program yang digunakan.

d. ECS (Amazon Elastic Container Service)

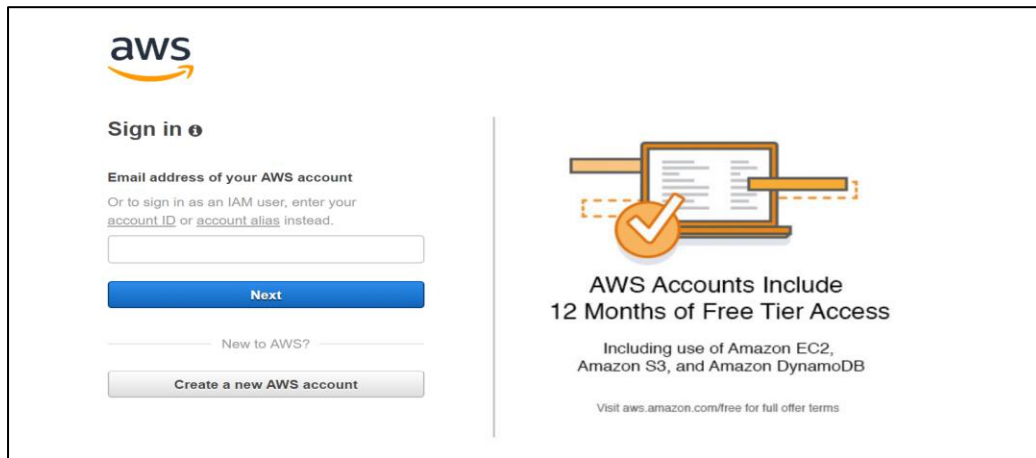
ECS adalah sebuah layanan kontainer pada AWS yang mendukung kontainer docker untuk memudahkan pengguna menjalankan dan menyatukan aplikasi dengan performa serta skalabilitas yang tinggi. ECS memungkinkan pengguna mengembangkan aplikasi dalam kontainer mulai dari aplikasi long-running dan microservices hingga aplikasi batch jobs dan machine learning. ECS menghilangkan kebutuhan pengguna untuk instal perangkat lunak kontainer secara manual serta mengatur penjadwalan kontainer pada beberapa mesin virtual. ECS memiliki fitur yaitu AWS Fargate yang dapat digunakan untuk mengelola kontainer tanpa harus menyiapkan atau mengelola server. Fargate membuat pengguna tidak lagi harus mengatur jenis dan kapasitas mesin virtual pada Amazon EC2 untuk menjalankan beberapa kontainer.

LANGKAH Pengerjaan

Penjelasan mengenai langkah-langkah pengerjaan untuk menjalankan mesin virtual dan pengaturan sumber daya komputasi terbagi menjadi dua yaitu dengan menggunakan sistem operasi Linux dan Windows.

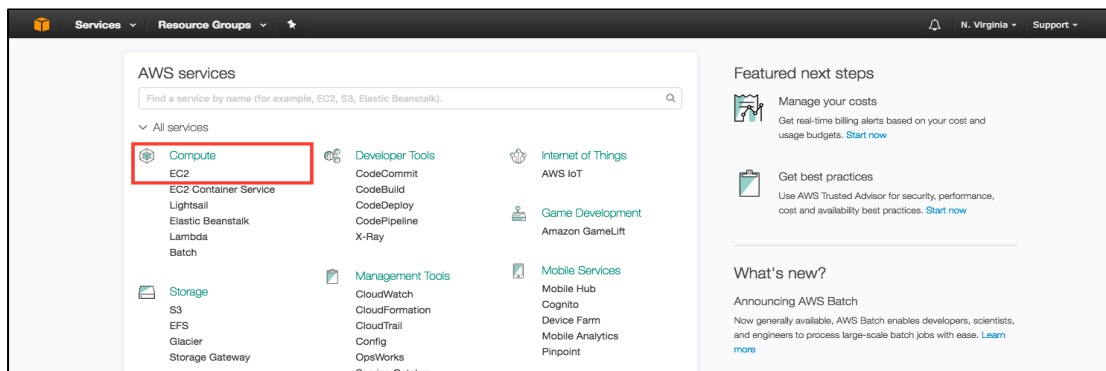
Launch Amazon EC2 Linux Instance

1. Sing up jika tidak memiliki akun AWS pada <https://portal.aws.amazon.com/billing/signup> , kemudian sign in jika sudah memiliki akun pada AWS.



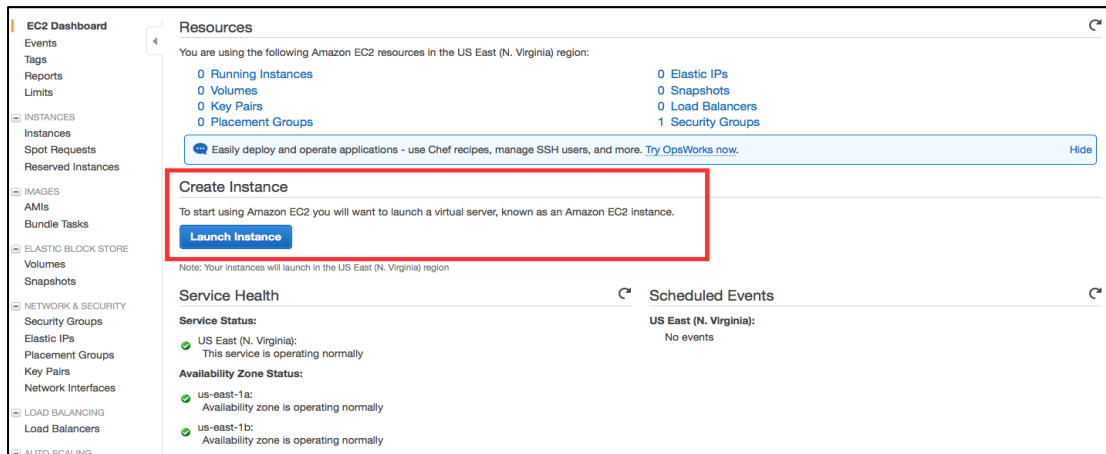
Gambar 1. Sign in AWS

2. Menjalankan Amazon EC2 Linux Instance
 - a. Pilih EC2 pada bagian AWS Services.



Gambar 2. EC2

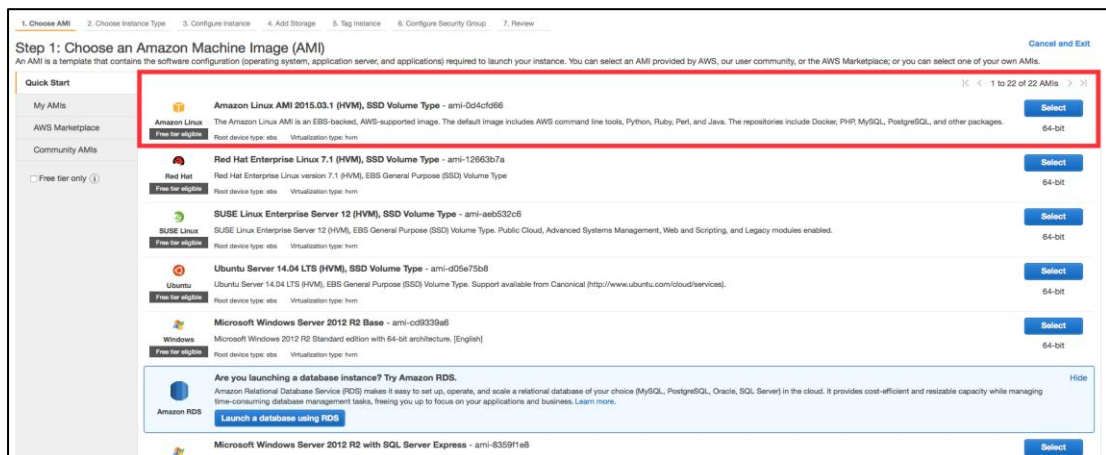
- b. Pilih **Launch Instance** untuk membuat dan mengkonfigurasi mesin virtual.



Gambar 3. Lauch Instance

3. Membuat dan Mengkonfigurasi Mesin Virtual Linux

- a. Langkah pertama yang dilakukan adalah memilih AMI (Amazon Machine Image). AMI merupakan template server yang telah dikonfigurasi untuk menjalankan mesin virtual. Setiap AMI berisi sistem operasi dan dapat juga berisi aplikasi server. Klik **Select** untuk memilih Amazon Linux AMI.



Gambar 4. Select

- b. Langkah selanjutnya adalah memilih spesifikasi sumber daya komputasi yang sesuai dengan kebutuhan yang diperlukan. Klik **Review and Launch** untuk melihat konfigurasi dan memilih

spesifikasi

yang

sesuai.

| | Family | Type | vCPUs | Memory (GiB) | Instance Storage (GiB) | EBS-Optimized Available | Network Performance |
|-------------------------------------|-----------------|---|-------|--------------|------------------------|-------------------------|---------------------|
| <input checked="" type="checkbox"/> | General purpose | t2.micro <small>Free tier eligible</small> | 1 | 1 | EBS only | - | Low to Moderate |
| <input type="checkbox"/> | General purpose | t2.small | 1 | 2 | EBS only | - | Low to Moderate |
| <input type="checkbox"/> | General purpose | t2.medium | 2 | 4 | EBS only | - | Low to Moderate |
| <input type="checkbox"/> | General purpose | t2.large | 2 | 8 | EBS only | - | Low to Moderate |
| <input type="checkbox"/> | General purpose | m4.large | 2 | 8 | EBS only | Yes | Moderate |
| <input type="checkbox"/> | General purpose | m4.xlarge | 4 | 16 | EBS only | Yes | High |
| <input type="checkbox"/> | General purpose | m4.2xlarge | 8 | 32 | EBS only | Yes | High |
| <input type="checkbox"/> | General purpose | m4.4xlarge | 16 | 64 | EBS only | Yes | High |
| <input type="checkbox"/> | General purpose | m4.10xlarge | 40 | 160 | EBS only | Yes | 10 Gigabit |
| <input type="checkbox"/> | General purpose | m3.medium | 1 | 3.75 | 1 x 4 (SSD) | - | Moderate |
| <input type="checkbox"/> | General purpose | m3.large | 2 | 7.5 | 1 x 32 (SSD) | - | Moderate |
| <input type="checkbox"/> | General purpose | m3.xlarge | 4 | 15 | 2 x 40 (SSD) | Yes | High |
| <input type="checkbox"/> | General purpose | m3.2xlarge | 8 | 30 | 2 x 80 (SSD) | Yes | High |

Cancel Previous **Review and Launch** Next: Configure Instance Details

Gambar 5. Review and Launch

- c. Langkah selanjutnya, klik **Launch** pada bagian bawah setelah memilih spesifikasi mesin virtual.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

Improve your instances' security. Your security group, launch-wizard-1, is open to the world.

Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

AMI Details

Amazon Linux AMI 2015.03.1 (HVM), SSD Volume Type - ami-9d4cfd66 [Edit AMI](#)

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

Instance Type

| Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GiB) | EBS-Optimized Available | Network Performance |
|---------------|----------|-------|--------------|------------------------|-------------------------|---------------------|
| t2.micro | Variable | 1 | 1 | EBS only | - | Low to Moderate |

[Edit instance type](#)

Security Groups

[Edit security groups](#)

Security group name: launch-wizard-1
Description: launch-wizard-1 created 2015-09-11T13:35:57.265-07:00

| Type | Protocol | Port Range | Source |
|------|----------|------------|-----------|
| SSH | TCP | 22 | 0.0.0.0/0 |

Instance Details

[Edit instance details](#)

Storage

[Edit storage](#)

Tags

[Edit tags](#)

Cancel Previous **Launch**

Gambar 6. Launch

- d. Langkah selanjutnya adalah memilih key pair (kunci), Pilih **Create a new key pair** dan berikan nama key pair tersebut. Kemudian klik **Download Key Pair** dan jalankan dengan klik **Launch Instance**.

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

MyKeyPair

Download Key Pair

...

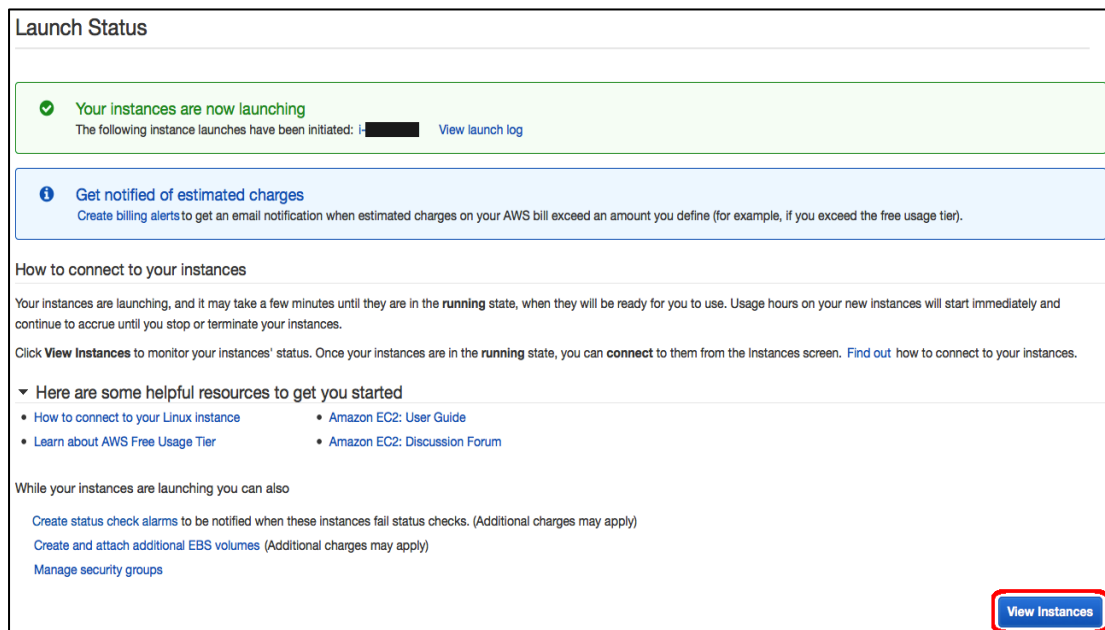
You have to download the **private key file** (*.pem file) before you can continue. **Store it in a secure and accessible location.** You will not be able to download the file again after it's created.

Cancel

Launch Instances

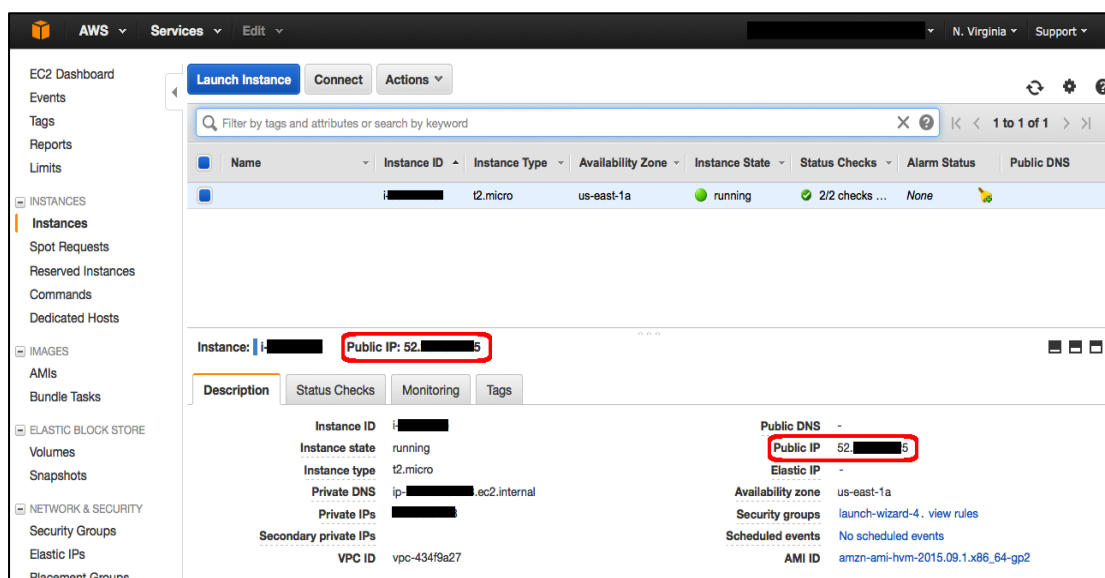
Gambar 7. Key Pair

- e. Selanjutnya, klik **View Instance** untuk melihat status mesin virtual yang telah dijalankan.



Gambar 8. View Instances

- f. Selanjutnya, dalam beberapa saat status pada Instance State akan berubah menjadi “running” yang menandakan mesin virtual telah berhasil berjalan dan akan muncul alamat IP Public dari mesin virtual tersebut. IP Public dapat digunakan untuk menghubungkan mesin virtual dengan menggunakan SSH.

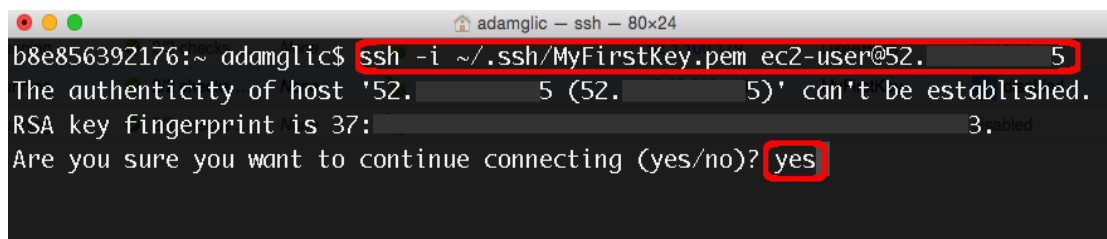


Gambar 9. IP Public

4. Menghubungkan Mesin Virtual dengan SSH

- Langkah pertama yang dilakukan adalah pastikan permission key pair yang telah didownload tidak public. Ubah permission dengan **chmod 400**.

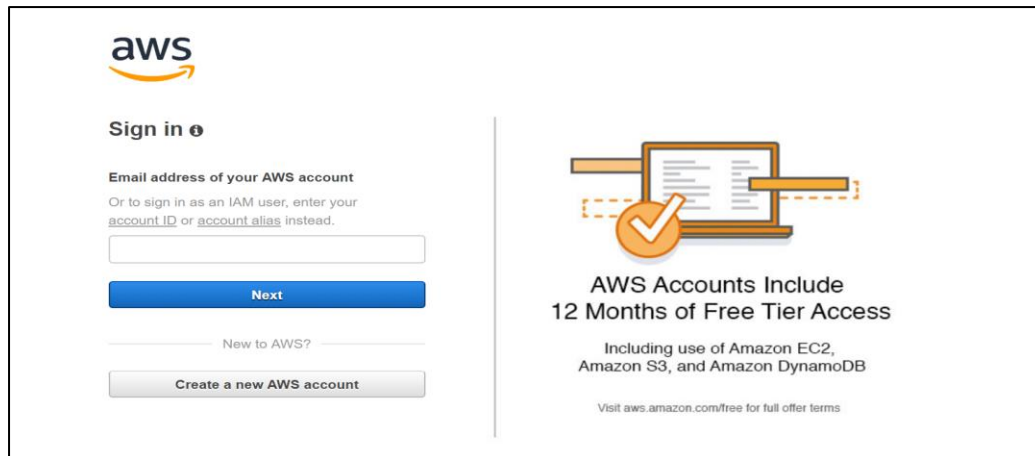
Gambar 10. chmod 400



- c. Respon apabila berhasil terhubung mesin virtual AWS Linux di cloud dengan SSH.

Gambar 12. Respon Berhasil

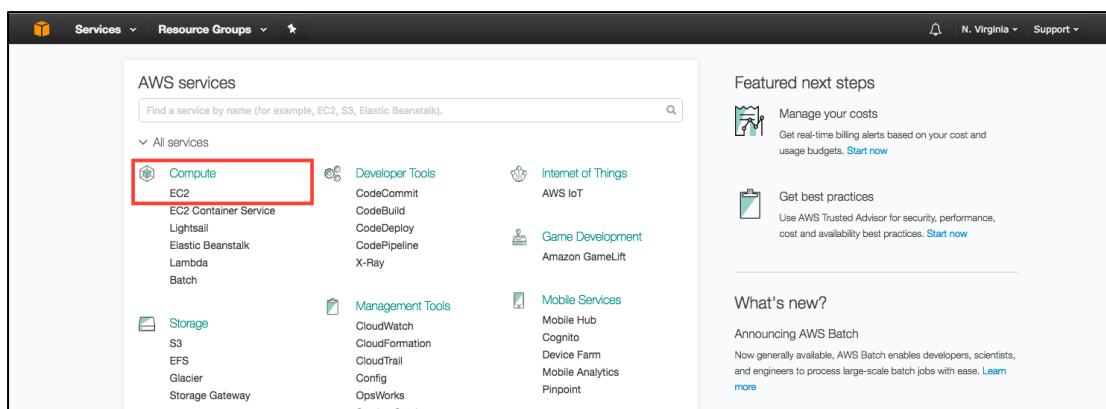
1. Sing up jika tidak memiliki akun AWS pada <https://portal.aws.amazon.com/billing/signup> , kemudian sign in jika sudah memiliki akun pada AWS.



Gambar 13. Sign in AWS

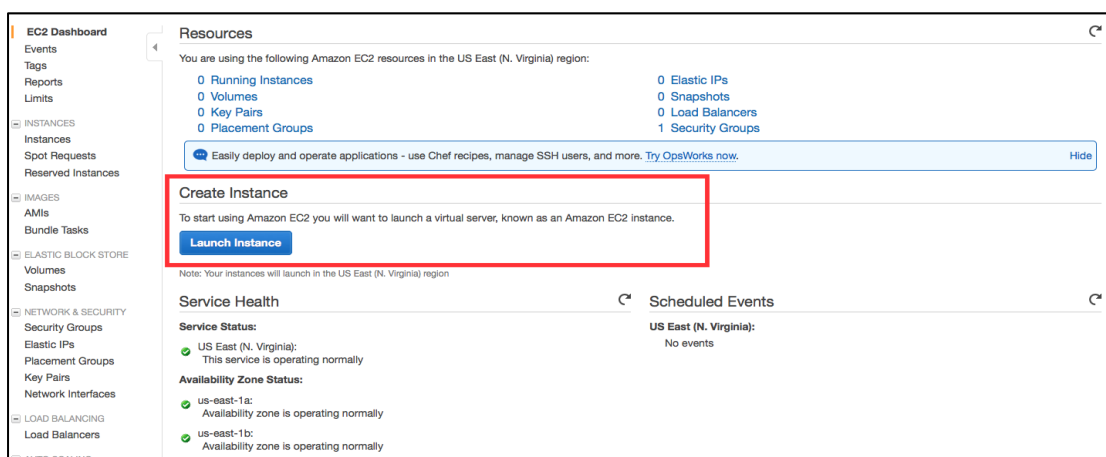
2. Menjalankan Amazon EC2 Windows Instance

a. Pilih EC2 pada bagian AWS Services.



Gambar 14. EC2

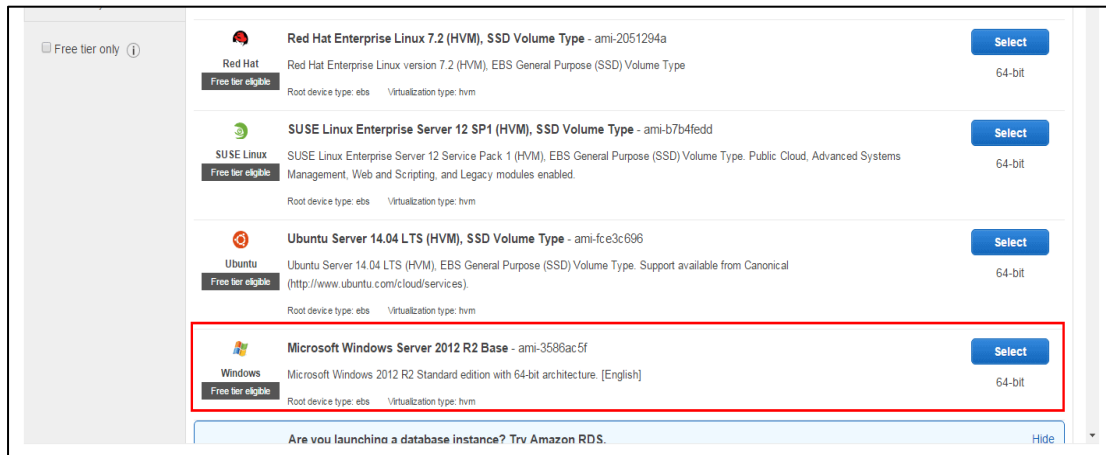
b. Pilih **Launch Instance** untuk membuat dan mengkonfigurasi mesin virtual.



Gambar 15. Launch Instance

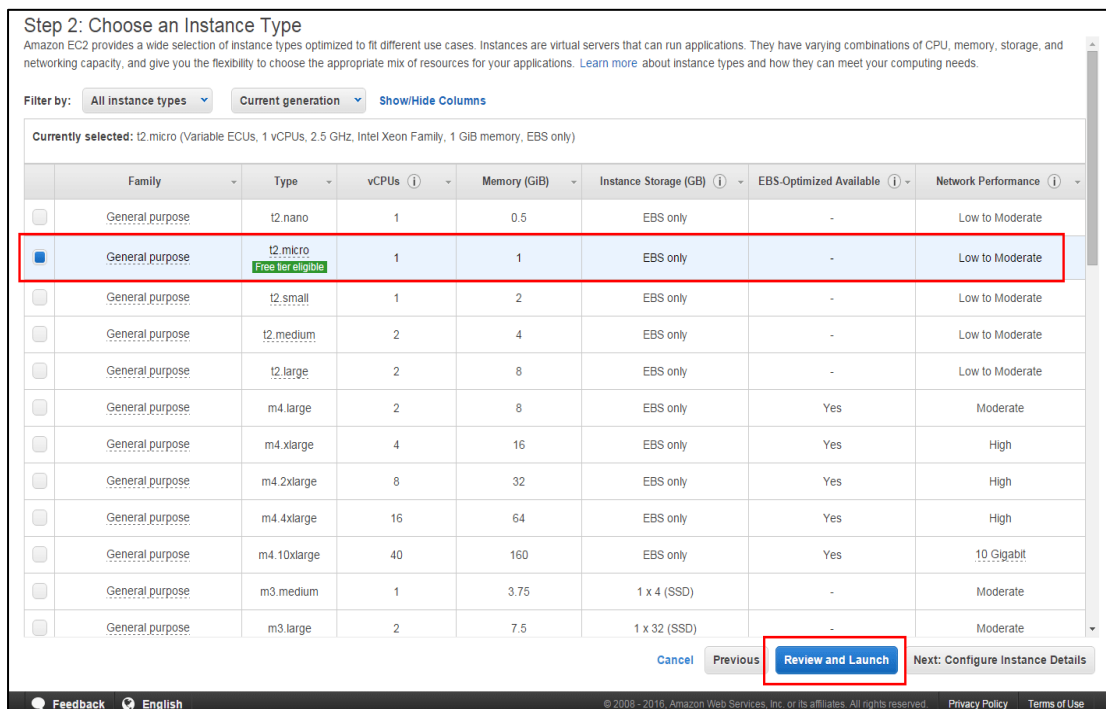
3. Membuat dan Mengkonfigurasi Mesin Virtual Windows

- a. Langkah pertama yang dilakukan adalah memilih AMI (Amazon Machine Image). Klik **Select** untuk memilih Microsoft Windows Server 2012 R2 Base AMI.



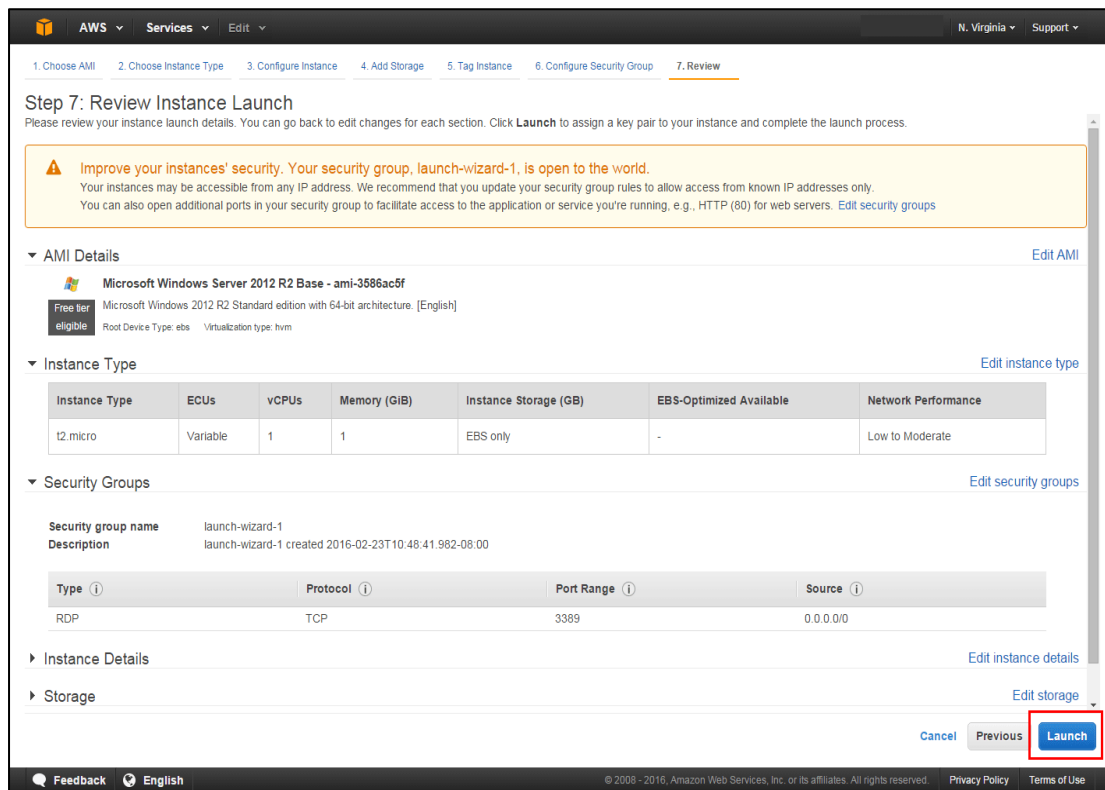
Gambar 16. Select

- b. Langkah selanjutnya adalah memilih spesifikasi sumber daya komputasi yang sesuai dengan kebutuhan yang diperlukan. Klik **Review and Launch** untuk melihat konfigurasi dan memilih spesifikasi yang sesuai.



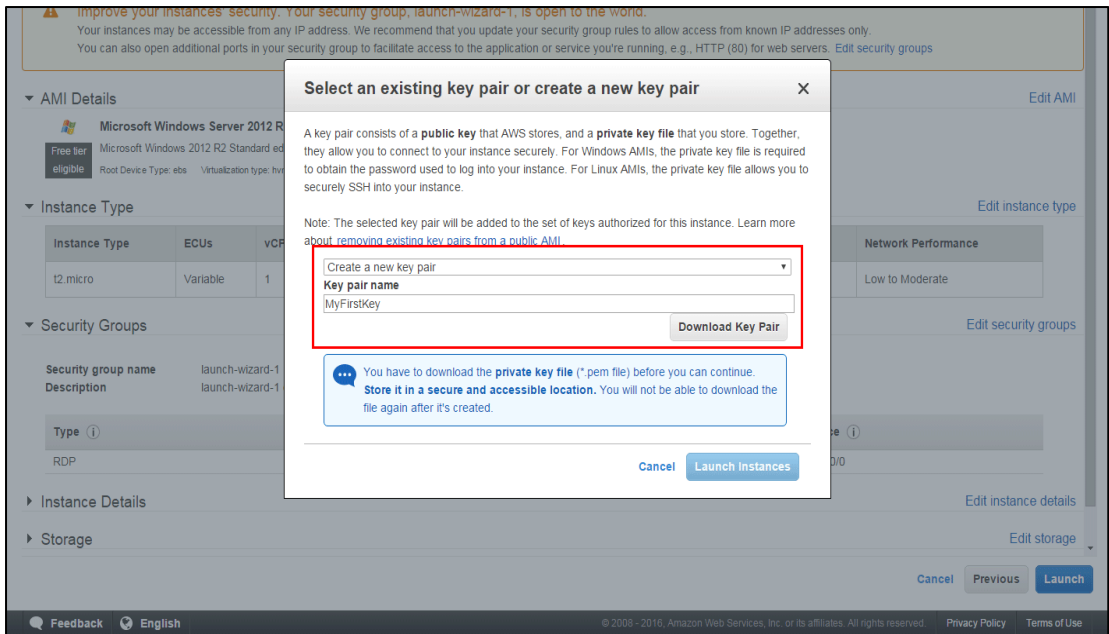
Gambar 17. Review and Launch

- c. Langkah selanjutnya, klik **Launch** pada bagian bawah setelah memilih spesifikasi mesin virtual.



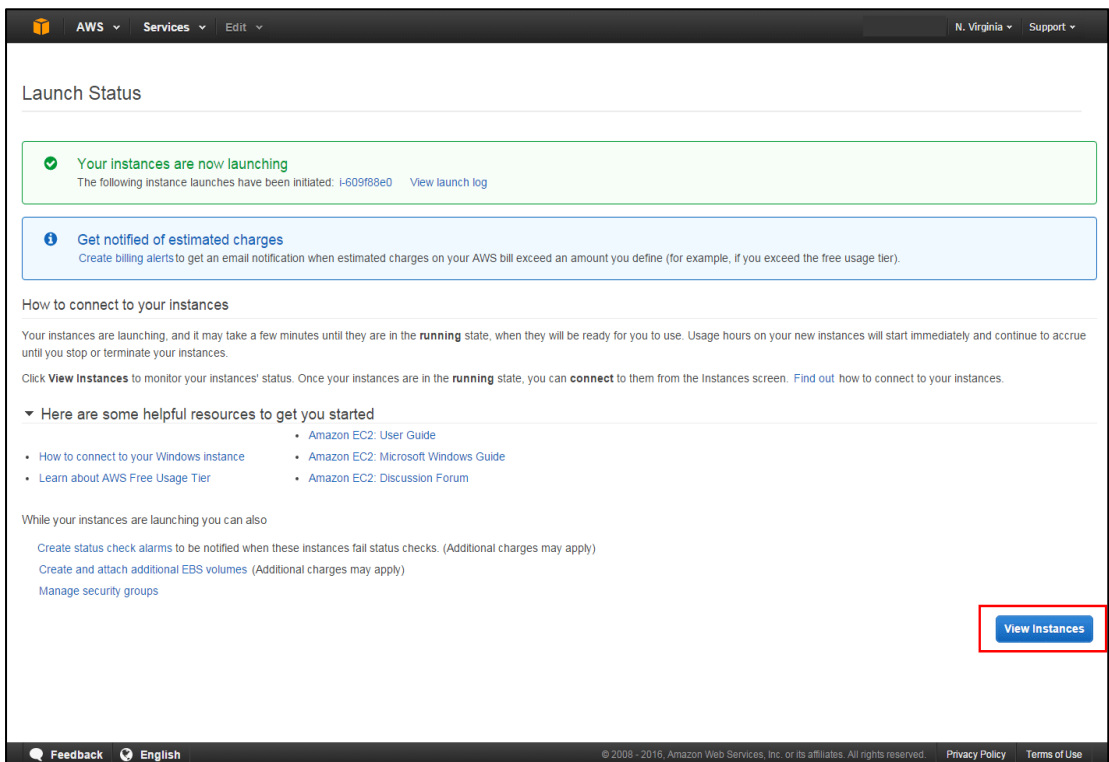
Gambar 18. Launch

- d. Langkah selanjutnya adalah memilih key pair (kunci), Pilih **Create a new key pair** dan berikan nama key pair tersebut. Kemudian klik **Download Key Pair** dan jalankan dengan klik **Launch Instance**.



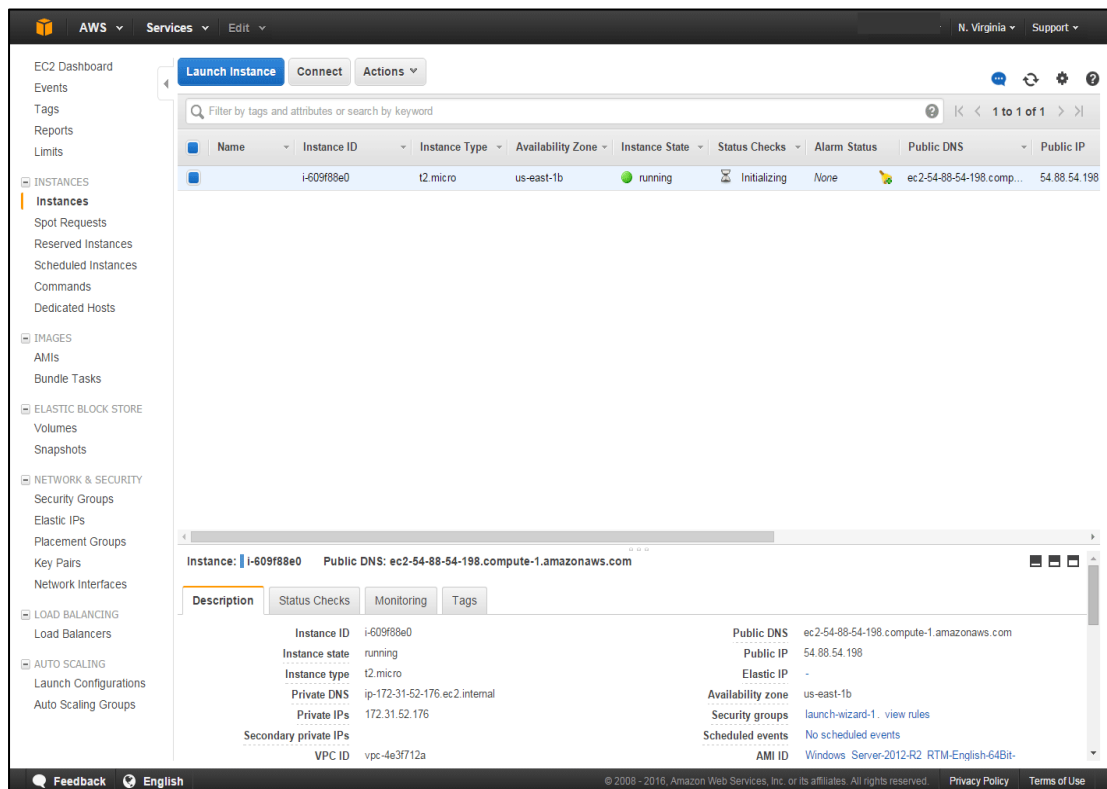
Gambar 19. Key Pair

- e. Selanjutnya, klik **View Instance** untuk melihat status mesin virtual yang telah dijalankan.



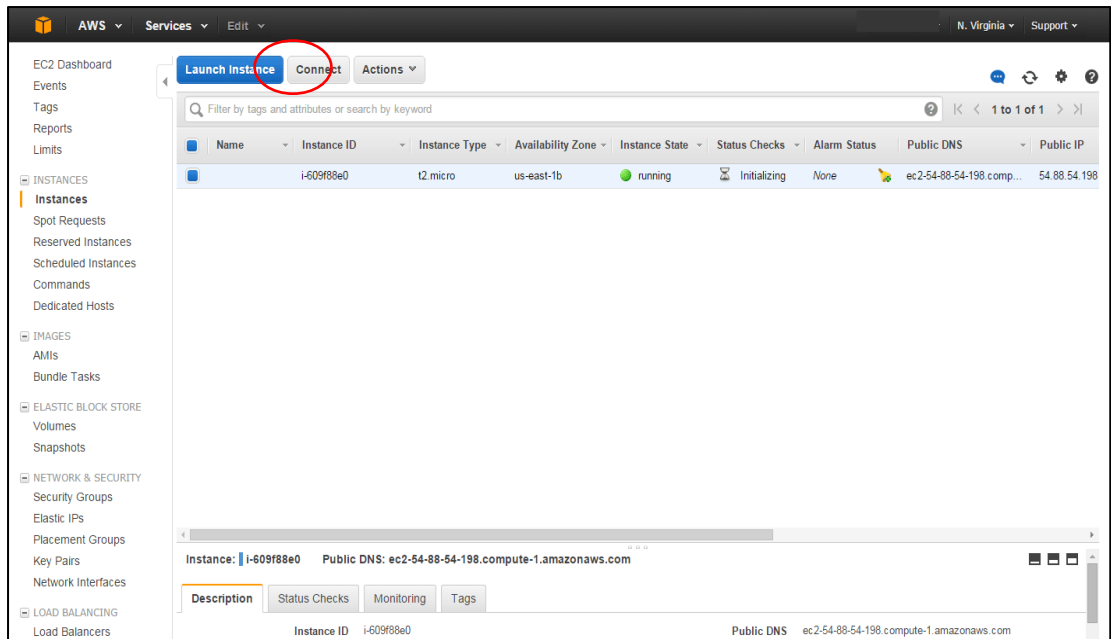
Gambar 20. View Instances

- f. Selanjutnya, dalam beberapa saat status pada Instance State akan berubah menjadi “running” yang menandakan mesin virtual telah berhasil berjalan.



Gambar 21. Mesin Virtual Berjalan

4. Menghubungkan Mesin Virtual Dengan RDP (Remote Desktop Protocol) client.
- a. Langkah pertama yang dilakukan adalah pastikan RDP client telah terinstall (https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/connecting_to_windows_instance.html#rdp-prereqs), kemudian klik **connect** pada mesin virtual windows.



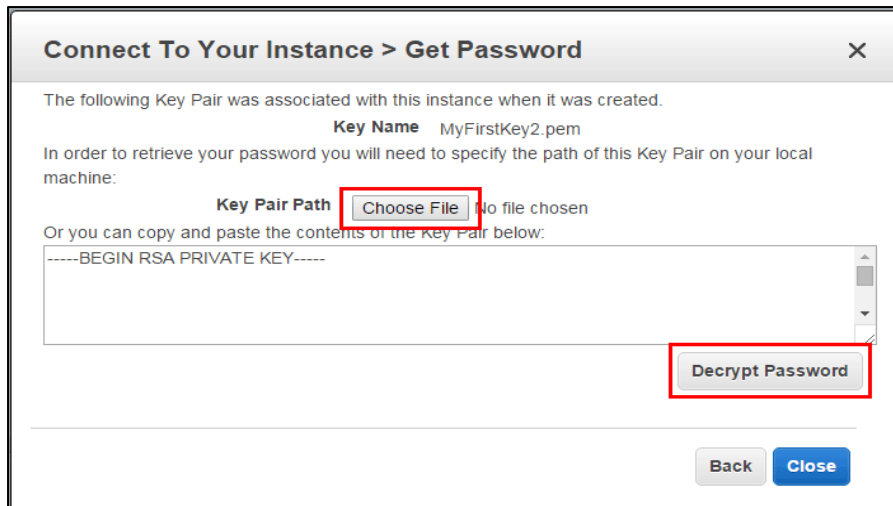
Gambar 10. Connect

- b. Selanjutnya, isikan username dan password.



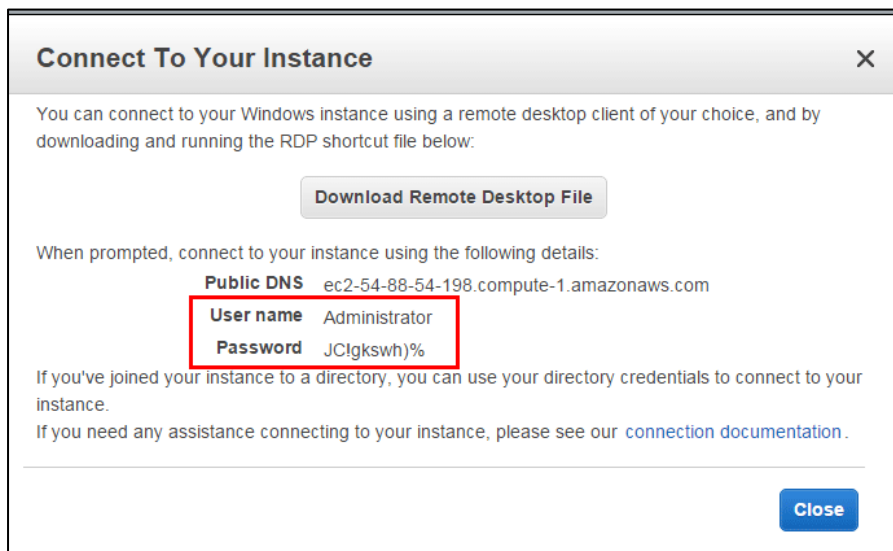
Gambar 22. Username dan Password

- c. Kemudian masukkan file key pair yang telah didownload sebelumnya kemudian klik **Decrypt Password**.



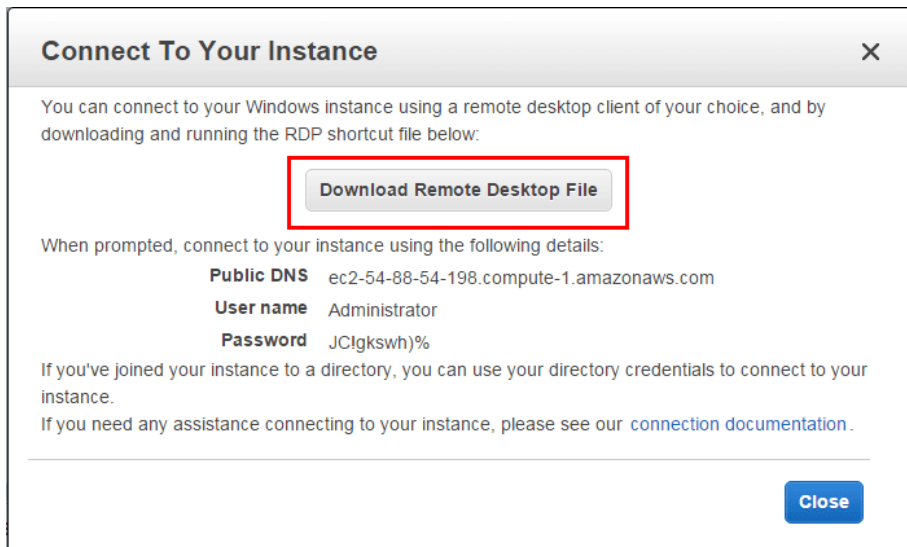
Gambar 23. Locate Keypair

- d. Hasil apabila key pair berhasil di decrypt.



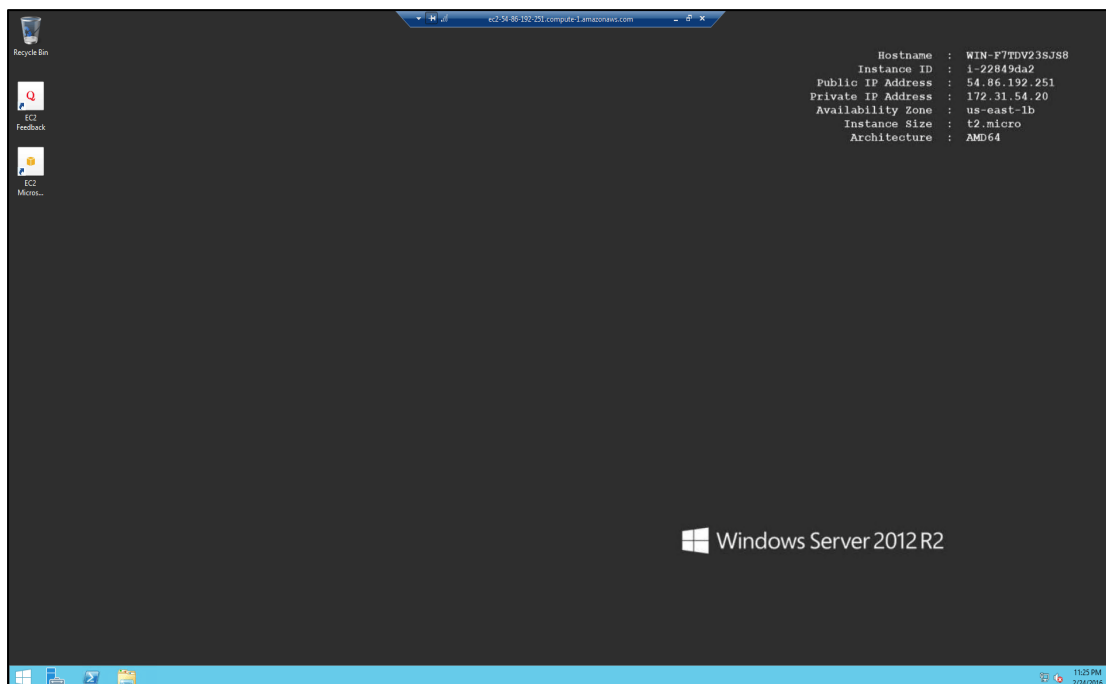
Gambar 13. Decrypt Keypair

- e. Kemudian klik **Download Remote Desktop File** dan buka file tersebut.



Gambar 14. Download File

- f. Mesin virtual windows berhasil berjalan.



Gambar 13. Mesin Virtual Windows Berjalan

SOAL LATIHAN

PT. XYZ merupakan sebuah perusahaan startup yang bergerak di bidang e-commerce. Mereka meminta anda untuk mengelola sebuah VPS yang mereka miliki. Anda diminta untuk melakukan instalasi dan konfigurasi portal web perusahaan menggunakan **WordPress** dengan skema sistem sebagai berikut.



Peran masing-masing layanan tersebut, antara lain :

1. **Client** berperan untuk mengirimkan request sesuai dengan definisi *endpoint* pada Tabel 1. Anda dapat memanfaatkan aplikasi pengirim request HTTP seperti [Postman](#).
2. **EC2 Webserver** yang berperan sebagai komputer virtual tempat Webserver dipasang. Terdapat beberapa jenis webserver yang dapat anda pakai antara lain : Apache2 Webserver, nginx atau lighttpd. Instance EC2 yang dipakai menggunakan sistem operasi Linux dengan distro bebas. Instance ini harus dapat diakses menggunakan alamat IP global yang disediakan oleh AWS.
3. **EC2 MySQL** yang berperan sebagai komputer virtual tempat server MySQL dipasang. Instance EC2 yang dipakai menggunakan sistem operasi Linux dengan distro bebas. Instance ini harus dapat diakses menggunakan IP dari instance **EC2 Webserver**.
4. Untuk langkah-langkah instalasi Wordpress dapat dilihat pada tutorial yang dicantumkan pada bagian referensi.

Catatan :

1. Anda dapat menggunakan instance EC2 berjenis Red Hat atau Amazon Linux yang merupakan turunan dari distro Red Hat/Fedora/Centos.
2. Pada beberapa distro, file konfigurasi MySQL/MariaDB dapat diakses pada lokasi **/etc/my.cnf**
3. Anda dapat mengubah/menambahkan konfigurasi

```
bind-address=0.0.0.0
```

pada file konfigurasi server database MySQL/MariaDB agar server tersebut dapat menerima permintaan koneksi dari beragam IP

4. Pada beberapa distro yang mengaktifkan fitur SELinux, webserver perlu dikonfigurasi agar diijinkan mengakses database secara remote. Cara mengkonfigurasinya adalah dengan mengetikkan perintah berikut sebagai root pada mesin yang akan mengakses database secara remote.

```
$ sudo setsebool -P httpd_can_network_connect_db 1
```

REFERENSI

1. Overview AWS Compute Services. Online at :
<https://docs.aws.amazon.com/whitepapers/latest/aws-overview/compute-services.html>
2. AWS EC2. Online at : <https://aws.amazon.com/ec2/>
3. AWS ECS. Online at : <https://aws.amazon.com/ecs/>
4. AWS Lambda. Online at : <https://aws.amazon.com/lambda/>
5. How to Configure WordPress to use a Remote Database. Online at :
<https://www.rosehosting.com/blog/how-to-configure-wordpress-to-use-a-remote-database/>