

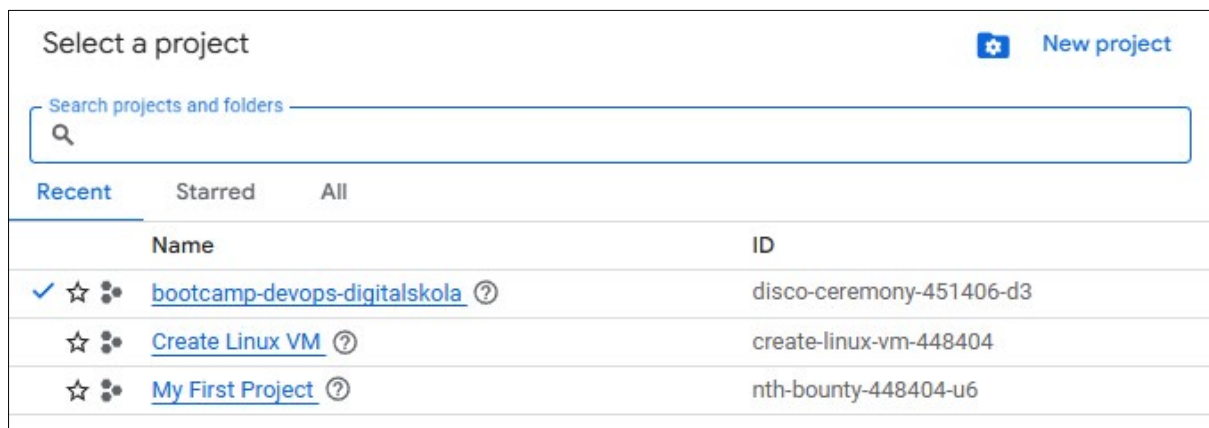
Project Sesi 5 Bootcamp DevOps Engineer Digital Skola

1. Setup 2 Server pada virtualbox
2. Setting 2 server tersebut bisa terhubung menggunakan ssh tanpa password
3. Buat 1 file dengan nama server1.txt di server1
4. Buat cronjob untuk automation kirim file server1.txt dari server1 ke server2 setiap hari senin jam 11 malam.

1. Setup 2 Server pada virtualbox / gcp

Langkah 1: Login ke Google Cloud Platform

- 1) Buka Google Cloud Console.
- 2) Login dengan akun Google.
- 3) Pilih proyek dimana server akan dibuat. Jika belum memiliki proyek atau ingin membuat proyek baru, dapat dilakukan dengan mengklik **Select a project > New Project**.



Langkah 2: Buka Halaman Compute Engine

- 1) Setelah masuk ke dashboard GCP, cari **Compute Engine** di menu navigasi.
- 2) Klik **VM instances**.

Langkah 3: Buat Instance VM

- 1) Klik **Create Instance**.
- 2) Isi detail instance:
 - **Name:** Beri nama untuk instance, yaitu **server1**.
 - **Region:** Pilih region asia-southeast1 (Singapore) agar minim latency.
 - **Zone:** Any. Biarkan GCP yang memilihkan Zone dalam Regionnya.

Google Cloud

bootcamp-devops-digitalskola

Compute Engine

VM instances

CREATE INSTANCE

IMPORT VM

REFRESH

Overview

Virtual machines

VM instances

Instance templates

Sole-tenant nodes

INSTANCES

OBSERVABILITY

INSTANCE SCHEDULES

VM instances

Filter

Enter property name or value

Status

Name

Zone

Recommendations

In use by

Internal

Machine configuration

Name *

server1

Region *

asia-southeast1 (Singapore)

Zone *

Any

General purpose

Compute optimized

Memory optimized

Storage optimized

GPUs

Machine types for common workloads, optimized for cost and flexibility

Series	Description	vCPUs	Memory	CPU Platform
C4	Consistently high performance	2 - 192	4 - 1,488 GB	Intel Emerald Rapids
C4A	Arm-based consistently high performance	1 - 72	2 - 576 GB	Google Axion
N4	Flexible & cost-optimized	2 - 80	4 - 640 GB	Intel Emerald Rapids
C3	Consistently high performance	4 - 192	8 - 1,536 GB	Intel Sapphire Rapids
C3D	Consistently high performance	4 - 360	8 - 2,880 GB	AMD Genoa
E2	Low cost, day-to-day computing	0.25 - 32	1 - 128 GB	Intel Broadwell
N2	Balanced price & performance	2 - 128	2 - 864 GB	Intel Cascade Lake
N2D	Balanced price & performance	2 - 224	2 - 896 GB	AMD Milan
T2A	Scale-out workloads	1 - 48	4 - 192 GB	Ampere Altra
T2D	Scale-out workloads	1 - 60	4 - 240 GB	AMD Milan
N1	Balanced price & performance	0.25 - 96	0.6 - 624 GB	Intel Haswell

Langkah 4: Pilih Machine Type


Di bagian **Machine type**, pilih **e2-micro** (2 vCPU, 1 GB memory). Ini adalah tipe mesin yang paling murah dan cocok untuk beban kerja ringan.

Machine type
Choose a machine type with preset amounts of vCPUs and memory that suit most workloads. Or, you can create a custom machine for your workload's particular needs. [Learn more](#)

PRESET

CUSTOM

e2-micro (2 vCPU, 1 core, 1 GB memory)




vCPU
0.25-2 vCPU (1 shared core)

Memory
1 GB

Langkah 5: Pilih Boot Disk

- 1) Di bagian **Boot disk**, klik **Change**.
- 2) Pilih **Ubuntu** sebagai OS.
- 3) Pilih versi Ubuntu yang Anda inginkan, misalnya **Ubuntu 20.04 LTS**.
- 4) Pilih **Balanced persistent disk** (atau boleh juga Standard persistent disk) sebagai tipe disk.
- 5) Sesuaikan ukuran disk sesuai kebutuhan, tetapi untuk biaya terendah, gunakan ukuran minimal (10 GB).

Operating system and storage

Name	server1
Type	New balanced persistent disk
Size	10 GB
Snapshot schedule ?	default-schedule-1
License type ?	Free
Image	 Ubuntu 20.04 LTS

CHANGE

Langkah 6: Konfigurasi Jaringan

- 1) Di bagian **Firewall**, **Allow HTTP traffic** dan **Allow HTTPS traffic** tidak perlu dicentang karena pada projek ini tidak membutuhkan akses web.
- 2) Network tag dapat ditambahkan untuk memudahkan manajemen. Saya beri tag **server-ubuntu**.
- 3) Untuk network interface biarkan menggunakan vpc default (bawaan gcp).

Networking

Firewall ?

Add tags and firewall rules to allow specific network traffic from the Internet

☐ Allow HTTP traffic

☐ Allow HTTPS traffic

☐ Allow Load Balancer Health Checks

Network tags

server-ubuntu ?

Hostname ?

Set a custom hostname for this instance or leave it default. Choice is permanent

Network interfaces ?

Network interface is permanent

Edit network interface

Network *
default ?

Subnetwork *
default IPv4 (10.148.0.0/20) ?

To use IPv6, you need an IPv6 subnet range. [LEARN MORE](#)

Network interface card
—

IP stack type

- ☒ IPv4 (single-stack)
- ☐ IPv4 and IPv6 (dual-stack)
- ☐ IPv6 (single-stack)

Primary internal IPv4 address
Ephemeral (Automatic) ?

Alias IP ranges

+ ADD IP RANGE

External IPv4 address
Ephemeral ?

Network Service Tier

- ☒ Premium ?
- ☐ Standard (asia-southeast1) ?
200 GB / mo free in every region

Langkah 7: Buat Instance

- 1) Dapat dilihat etimasi biaya perbulan untuk server1 adalah \$8.64. Tidak masalah karena untuk pengguna baru GCP masih diberi credit \$300.
- 2) Setelah semua konfigurasi selesai, klik **Create**.

Monthly estimate
\$8.64
That's about \$0.01 hourly
Pay for what you use: no upfront costs and per second billing

Item	Monthly estimate
2 vCPU + 1 GB memory	\$7.54
10 GB balanced persistent disk	\$1.10
Snapshot schedule	Cost varies
Total	\$8.64

[Compute Engine pricing](#)

[^ LESS](#)

CREATE **CANCEL** [↔ EQUIVALENT CODE](#)

- 3) Tunggu beberapa saat hingga instance Anda selesai dibuat.

Running data protection operation for "server1"
bootcamp-devops-digitalskola

4 minutes ago

Create VM instance "server1" and its boot disk "server1"
bootcamp-devops-digitalskola

4 minutes ago

Filter Enter property name or value							
<input type="checkbox"/> Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input checked="" type="checkbox"/>	server1	asia-southeast1-b			10.148.0.2 (nic0)	34.87.172.125 (nic0)	SSH ▾ ⋮

Langkah 8: Buat Instance server2

- 1) Untuk membuat server2 yang spesifikasinya sama dengan server1, dapat dilakukan dengan klik **Create Similar**.
- 2) Tampilan detail instance seperti awal akan muncul namun sudah sama persis dengan yang sudah diconfig hingga **Langkah 6**. Jadi, hanya perlu ubah Nama menjadi **server2**.
- 3) Lalu, klik **Create**.



Langkah 9: Akses VM

- 1) Setelah instance berjalan, **server1** dan **server2** dapat dilihat di daftar VM instances.
- 2) Klik **SSH** untuk mengakses VM melalui browser atau gunakan SSH client dengan IP eksternal yang diberikan.

VM instances

Filter

Enter property name or value

<input type="checkbox"/>	Status	Name ↑	Zone	Recommendations	In use by	Internal IP	External IP	Connect
<input type="checkbox"/>	✓	server1	asia-southeast1-b			10.148.0.2 (nic0)	34.87.172.125 (nic0)	SSH <div>▼</div> <div>⋮</div>
<input type="checkbox"/>	✓	server2	asia-southeast1-b			10.148.0.3 (nic0)	35.240.169.222 (nic0)	SSH <div>▼</div> <div>⋮</div>

- 3) Pada project ini yang akan digunakan adalah SSH client untuk mengakses **server1** dan **server2** melalui local machine (laptop). Untuk itu, SSH Authentication perlu dilakukan terlebih dahulu seperti berikut:
 - a. Pastikan bahwa firewall mengizinkan koneksi SSH (port 22).
 - b. Di laptop melalui terminal (di sini memakai Ubuntu dengan WSL) SSH key pair dibuat dengan command **ssh-keygen -t rsa -f ~/.ssh/admin_key -C ubuntu_admin**.
 - c. Command tersebut akan menghasilkan dua file: **admin_key** (private key) dan **admin_key.pub** (public key). (note: jika opsi **-f** tidak dipakai maka nama file default yang dibuat adalah **id_rsa**)

```
kaisenberga@MENOMEN:~$ sudo su -
[sudo] password for kaisenberga:
root@MENOMEN:~# ssh-keygen -t rsa -f ~/.ssh/admin_key -C ubuntu_admin
```

```
root@MENOMEN:~# ls -la .ssh
total 24
drwx----- 2 root root 4096 Feb 20 13:02 .
drwx----- 6 root root 4096 Feb 20 11:48 ..
-rw----- 1 root root 2602 Feb 20 13:02 admin_key
-rw-r--r-- 1 root root 566 Feb 20 13:02 admin_key.pub
```

- d. Buka isi file **admin_key.pub** dengan command **cat .ssh/admin_key.pub** lalu copy.

```
root@MENOMEN:~# cat .ssh/admin_key.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQDqMnQ4YWYcCBbzzd5LSJj1C5BKX/t4yR
fuoy5cs1kVZ3ZQZc89LJ6A/uQQ5o5o+9Lj/GTpD4KRCrnr4bjCrEeJq6z3ewzNGYFNSH8N
I1NFcpIXUktfbzmpcgrA4tXfBNQfTa7vu9NQsWKKRx/hfxQftQayGE139df0Lkz0DgsD06
65mArN+4XX9AgNV3306Jp380bvnGbykIK1vq/pC+TL4c4in9mzzz1JKmUJFK2kIVqVp0+K
mQLoz4CMRfBLYmgSDHElnA0tHLrFlDiJc jV9xnFUVRZFNKyOWEJXL5Epc5U0LdZ0A9d+LE
Y07NFwKvftcMRFCpn0k+11YDZFH7VHG016bpBWb6do0MwsFklt84Vomu8j6BLIZmxW0luT
U9sysa/1r8QNni08YZ7rjJfT90Nd7LJwkfsOowxhULKrZ0u7fY6gjEn6x/QP4at+OagcEZ
nOHsK/MatDxCcJ2/IZtBGC5t2lEpQbkFx+3ePi6kfjEjf+kh8ySASN47o85zc= ubuntu_
admin
```

- e. Pada VM Intances pilih **server1**
- f. Klik **Edit** (tombol pensil).
- g. Scroll ke bagian **SSH Keys** dan paste admin_key.pub ke dalamnya.
- h. Simpan perubahan.

← server1
EDIT
RESET

SSH Keys

These keys allow access only to this instance, unlike project-wide SSH keys. [Learn more](#)

☐ Block project-wide SSH keys
When checked, project-wide SSH keys cannot access this instance. [Learn more](#)

SSH key 1 *

DxCcJ2/IZtBGC5t2lEpQbkFx+3ePi6kfjEjf+kh8ySASN47o85zc= ubuntu_admin

Enter public SSH key

+ ADD ITEM

- i. Ulangi langkah e hingga h untuk **server2**.
- j. Gunakan private key untuk mengakses **server1** dan **server2** melalui SSH dengan command: `ssh -i .ssh/admin_key ubuntu_admin@[EXTERNAL_IP]` di dua terminal terpisah.

```
root@MENOMEN:~# ssh -i .ssh/admin_key ubuntu_admin@34.87.172.125
```

```
ubuntu_admin@server1:~$
```

```
root@MENOMEN:~# ssh -i .ssh/admin_key ubuntu_admin@35.240.169.222
```

```
ubuntu_admin@server2:~$
```

2. Setting 2 server tersebut bisa terhubung menggunakan ssh tanpa password

Untuk menghubungkan dua server menggunakan ssh tanpa password adalah sebagai berikut:

SSH server1 ke server2

- 1) Pada server1, generate ssh key dengan command **ssh-keygen -t rsa -f .ssh/to_server2_key**.
- 2) Command tersebut akan menghasilkan dua file: **to_server2_key** (private key) dan **to_server2_key.pub** (public key) di direktori **.ssh/**.

```
ubuntu_admin@server1:~$ ssh-keygen -t rsa -f .ssh/to_server2_key
```

```
ubuntu_admin@server1:~$ cd .ssh
ubuntu_admin@server1:~/.ssh$ ls -la
total 24
drwx----- 2 ubuntu_admin ubuntu_admin 4096 Feb 21 03:02 .
drwxr-xr-x 4 ubuntu_admin ubuntu_admin 4096 Feb 20 07:22 ..
-rw----- 1 ubuntu_admin ubuntu_admin  584 Feb 20 06:03 authorized_keys
-rw-r--r-- 1 ubuntu_admin ubuntu_admin  222 Feb 21 02:45 known_hosts
-rw----- 1 ubuntu_admin ubuntu_admin 2610 Feb 21 03:02 to_server2_key
-rw-r--r-- 1 ubuntu_admin ubuntu_admin  574 Feb 21 03:02 to_server2_key.pub
```

- 3) Di direktori **.ssh/**, buka/baca file **to_server2_key.pub** dengan command **cat to_server2_key.pub** lalu copy.

```
ubuntu_admin@server1:~/.ssh$ cat to_server2_key.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDCHGOA56Fc1xoYS/8yViHXxFCgD1B6
iJl12RIJismb+A64EA7pJwgKltmG0mBbmE2xWhslykJJLVRpzyk7YVKruiLZDXwVwW3p
2xo1/3TWqet4FFJDACsohPXWsMiadeNOD3bucvtdBGdk5WGUMdudaWxuk5Lmn2GH/XAQ
GtpE1/1EjF9Hf9K+Cx5defyEu1NnQfxj1gVokspPIGTjrc+nMaRWLBRglMP1pNHnMrig
fQG810QkJTtaGLUpBAD0FzVTjrP/kuRze66w+JnKPV2eR+M9A9SRQgAzIJj6QwXzJteU
VePMszBKxAgKXgwfkF8K32SqM9ABdADxQXT3+YTe5eF7Ej1UC1euaN68CyNFeFoPI8xL
n9jNMRYH+eMwSM7fyYaONKEym7AI/GfA/pLaDaa4qD9DAeg1XurWoPTYNbFOI2GDIueH
Rz501lgFUyyXLRJn+7Rn5zdkymqXNW+XSSeatSoZp4cRstGRLoDriN3I/smWmYfH1sKS
cbrtiTk= ubuntu_admin@server1
```

- 4) Pada server2, di direktori **.ssh/**, buka file **authorized_keys** dengan tex editor nano menggunakan command **nano authorized_keys**. Di dalamnya sudah terdapat satu public key milik laptop, yaitu **admin_key.pub**. Aksi yang hanya perlu dilakukan adalah paste **to_server2_key.pub** di bawahnya. Lalu simpan.
- 5) Kembali ke server1, verifikasi bahwa sekarang server1 telah bisa terhubung menggunakan ssh ke server2 dengan menjalankan command **ssh -i to_server2_key ubuntu_admin@35.240.169.222**.
- 6) server1 telah berhasil terhubung ke server2 dengan ssh.


```

ubuntu_admin@server2:~$ cd .ssh
ubuntu_admin@server2:~/.ssh$ nano authorized_keys
ubuntu_admin@server2:~/.ssh$ cat authorized_keys
# Added by Google
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDQdMnQ4YWYcCBbzzd5lSJJ1C5BKX/t4
yRfuoy5cs1kVZ3ZQZc89LJ6A/uQQ5o5o+9Lj/GTpD4KRCrnr4bjCrEeJq6z3ewzNGYFN
SH8NI1NFcpIXUktfbzmpcgrA4tXfBNQfTa7vu9NQsWKKRx/hfxQftQayGE139df0lKz0
DgsD0665mArN+4XX9AgNV3306Jp380bvnGbykIK1vq/pC+TL4c4in9mzzz1JKmUJFK2k
IVqVpO+KmQLoz4CMRfBLYmgSDHElnA0tHLrFlDiJcjV9xnFUVRFNkyOWEJXL5Epc5UO
LdZ0A9d+lEY07NFwKvftcMRFCpn0k+11YDZFH7VHG016bpBwb6do0MWsFkl84Vomu8j
6BLIZmxW0luTU9sysa/1r8QNNi08YZ7rjJfT90Nd7LJwkfs0owxhULKrZ0u7fY6gjEn6
x/QP4at+0agcEZn0HsK/MatDxCcJ2/IZtBGC5t2lEpQbkFx+3ePi6kfjEjf+kh8ySASN
47o85zc= ubuntu_admin
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDCHGOA56FclxoYS/8yViHXxFCgD1B6
iJl12RIJismb+A64EA7pJwgKltmG0mBbmE2xWhslykJJLVRpzyk7YVKruiLZDXwVwW3p
2xo1/3TWqet4FFJDacsohPXWsMiadeNOD3bucvtdBGdk5WGUMdudaWxuk5Lmn2GH/XAQ
GtpE1/lEjF9Hf9K+Cx5defyEu1NnQfxj1gVokspIGTjrc+nMaRWLBrgLMP1pNHnMrig
fQG810QkJTtaGLUpBAD0FzVTjrP/kuRze66w+JnKPV2eR+M9A9SRQgAzIJj6QwXzJteU
VePMszBKxAgKXgwfkF8K32SqM9ABdADxQXT3+YTe5eF7EjLUC1euaN68CyNFeFoPI8xL
n9jNMRyH+eMwSM7fyYaONKEym7AI/GfA/pLaDaa4qD9DAegLXurWoPTyNbFOI2GDIueH
Rz50llgFuyyXLRJn+7Rn5zdkymqXNW+XSSeatSoZp4cRstGRL0DriN3I/smWmYfH1sKS
cbrtiTk= ubuntu_admin@server1

```

```

ubuntu_admin@server1:~/.ssh$ ssh -i to_server2_key ubuntu_admin@35.240.169.222

```

```

ubuntu_admin@server2:~$

```

SSH server2 ke server1

- 1) Pada server2, generate ssh key dengan command **ssh-keygen -t rsa -f .ssh/to_server1_key**.
- 2) Command tersebut akan menghasilkan dua file: **to_server1_key** (private key) dan **to_server1_key.pub** (public key) di direktori **.ssh/**.
- 3) Di direktori **.ssh/**, buka/baca file **to_server1_key.pub** dengan command **cat to_server1_key.pub** lalu copy.
- 4) Pada server1, di direktori **.ssh/**, buka file **authorized_keys** dengan tex editor nano menggunakan command **nano authorized_keys**. Paste **to_server1_key.pub** lalu simpan.

```

ubuntu_admin@server2:~$ ssh-keygen -t rsa -f .ssh/to_server1_key

```

```

ubuntu_admin@server2:~$ cd .ssh
ubuntu_admin@server2:~/.ssh$ ls -la
total 20
drwx----- 2 ubuntu_admin ubuntu_admin 4096 Feb 21 03:12 .
drwxr-xr-x 5 ubuntu_admin ubuntu_admin 4096 Feb 21 02:43 ..
-rw----- 1 ubuntu_admin ubuntu_admin 1159 Feb 21 03:06 authorized_keys
-rw----- 1 ubuntu_admin ubuntu_admin 2610 Feb 21 03:11 to_server1_key
-rw-r--r-- 1 ubuntu_admin ubuntu_admin 574 Feb 21 03:11 to_server1_key.pub

```

```
ubuntu_admin@server2:~/.ssh$ cat to_server1_key.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC+eC3KdJpxE3KMZRJ7ob06VFLayW0LH
HvV1U8c/JDfQN8fN9KoRbvuhRXX80h791AYTwwA7tLVMyKsJ2kHy4hijtTBs2cfWnW01P
nFUiRj6X5TLMqx+48Q5q90p+cvh/cvsS2wMk9tCo4w9nRB8n9KcoAt5L/5gy6Zt0ly2ug
CjVPDIrAe58+YigUSnDD6WDsGmhxT6PP0kUEpvjxDGjSeakEUXbAnn/qf54JXiBP8tCq6
/aDLRINjTljtfIzM/gu4jV9NzfNMjsOMyrPalmpnnrMqVtQhLqp06uvBpvhEfFWWmkTMR
YG81LL2P0Dq0bMBR6jMf8NS0f7EhZHIP6NkXUm5XZ45S48yYKLT0VSRGwVLGunAbDMLer
D+Hh4rkAOcu+amwnqVgLD57LCuPdPRqdRmyY/FB+k20SiIT240zyJYXzVWSTqwARAH2W3
hTHSk1FzMDKKT1VC5hMjNwk1b3lpV1hXRU0z0QcmIQ11e1jSUGLRsfY0/rbNMJ2gcW1c=
ubuntu_admin@server2
```

```
ubuntu_admin@server1:~/.ssh$ nano authorized_keys
ubuntu_admin@server1:~/.ssh$ cat authorized_keys
# Added by Google
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC+eC3KdJpxE3KMZRJ7ob06VFLayW0LH
Rfuoy5cs1kVZ3ZQZc89LJ6A/uQQ5o5o+9Lj/GTPd4KRCrnr4bjCrEeJq6z3ewzNGYFNSh
8NI1NFcpIXUktfbzmpcgrA4tXfBNQfTa7vu9NQsWKKRr/hfxQftQayGE139df0Lkz0Dgs
D0665mArN+4XX9AgNV3306Jp380bvnGbykIK1vq/pC+TL4c4in9mzzz1JKmUJFK2kIVqV
p0+KmQLoz4CMRfBLYmgSDHElnA0tHLrFLdiJcjV9xnFUVrZFNkyOWEJXL5Epc5UOLdZ0A
9d+LEyO7NFwKvftcMRFCpn0k+11YDZFh7VHG016bpBwb6do0MwsFklt84Vomu8j6BLIZm
xW0luTU9sysa/1r8QNni08YZ7rjJfT90Nd7LJwKfs0owxhULKrZ0u7fY6gJen6x/QP4at
+0agcEZn0HsK/MatDxCcJ2/IZtBGC5t2lEpQbkFx+3ePi6kfjEjf+kh8ySASN47o85zc=
ubuntu_admin
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGC+eC3KdJpxE3KMZRJ7ob06VFLayW0LH
HvV1U8c/JDfQN8fN9KoRbvuhRXX80h791AYTwwA7tLVMyKsJ2kHy4hijtTBs2cfWnW01P
nFUiRj6X5TLMqx+48Q5q90p+cvh/cvsS2wMk9tCo4w9nRB8n9KcoAt5L/5gy6Zt0ly2ug
CjVPDIrAe58+YigUSnDD6WDsGmhxT6PP0kUEpvjxDGjSeakEUXbAnn/qf54JXiBP8tCq6
/aDLRINjTljtfIzM/gu4jV9NzfNMjsOMyrPalmpnnrMqVtQhLqp06uvBpvhEfFWWmkTMR
YG81LL2P0Dq0bMBR6jMf8NS0f7EhZHIP6NkXUm5XZ45S48yYKLT0VSRGwVLGunAbDMLer
D+Hh4rkAOcu+amwnqVgLD57LCuPdPRqdRmyY/FB+k20SiIT240zyJYXzVWSTqwARAH2W3
hTHSk1FzMDKKT1VC5hMjNwk1b3lpV1hXRU0z0QcmIQ11e1jSUGLRsfY0/rbNMJ2gcW1c=
ubuntu_admin@server2
```

- 5) Kembali ke server2, verifikasi bahwa sekarang server2 telah bisa terhubung menggunakan ssh ke server1 dengan menjalankan command **ssh -i to_server1_key ubuntu_admin@34.87.172.125**.
- 6) Server2 telah berhasil terhubung ke server1 dengan ssh.

```
ubuntu_admin@server2:~/.ssh$ ssh -i to_server1_key ubuntu_admin@34.87.172.125
```

```
ubuntu_admin@server1:~$
```

3. Buat 1 file dengan nama server1.txt di server1

Pada server2, buat direktori **Kirim > masuk ke direktori Kirim > buat file dengan command touch server1.txt.**

```
ubuntu_admin@server1:~$ mkdir Kirim
ubuntu_admin@server1:~$ cd Kirim/
ubuntu_admin@server1:~/Kirim$ touch server1.txt
```

```
ubuntu_admin@server1:~/Kirim$ ls
server1.txt
```

Lalu, untuk persiapan folder untuk menerima dengan nama **terima** di **server2**. Pembuatan direktori-direktori ini hanya untuk terlihat rapih.

```
ubuntu_admin@server2:~$ mkdir terima
ubuntu_admin@server2:~$ ls
terima
```

4. Buat cronjob untuk automation kirim file **server1.txt** dari **server1** ke **server2** setiap hari senin jam 11 malam

Untuk membuat cronjob untuk automation kirim file **server1.txt** dari **server1** ke **server2** adalah sebagai berikut:

- 1) Membuat script untuk dijalankan/dieksekusi yaitu **transfer_file.sh**. Ketik command **nano transfer_file.sh** lalu buat perintah **scp -i ~/.ssh/to_server2_key /home/ubuntu_admin/kirim/server1.txt ubuntu_admin@35.240.169.222:/home/ubuntu_admin/terima**. Command tersebut akan melakukan secure copy (scp) dari **server1** ke **server2** dengan menggunakan ssh-key. Opsi **-i ~/.ssh/to_server2_key** harus dipakai karena nama ssh-key yang dipakai adalah custom. Jika nama default yang dipakai (yaitu **id_rsa**) maka tidak perlu opsi **-i**.

```
ubuntu_admin@server1:~/kirim$ nano transfer_file.sh
```

```
ubuntu_admin@server1:~/kirim$ cat transfer_file.sh
#!/bin/bash

scp -i ~/.ssh/to_server2_key /home/ubuntu_admin/kirim/server1.txt ubuntu_admin@
35.240.169.222:/home/ubuntu_admin/terima/
```

- 2) Tambahkan izin execute (x) dengan command **chmod +x transfer_file.sh** sehingga file tersebut dapat dijalankan/dieksekusi dan melakukan perintah yang telah diketik didalamnya.

```
ubuntu_admin@server1:~/kirim$ chmod +x transfer_file.sh
ubuntu_admin@server1:~/kirim$ ls -la
total 12
drwxrwxr-x 2 ubuntu_admin ubuntu_admin 4096 Feb 21 09:35 .
drwxr-xr-x 6 ubuntu_admin ubuntu_admin 4096 Feb 21 09:25 ..
-rw-rw-r-- 1 ubuntu_admin ubuntu_admin   0 Feb 21 09:25 server1.txt
-rwxrwxr-x 1 ubuntu_admin ubuntu_admin 134 Feb 21 09:35 transfer_file.sh
```

- 3) Untuk mempersingkat waktu membuktikan cronjob dapat berjalan dengan benar, cronjob dibuat untuk waktu dekat. Command **date** untuk melihat waktu sekarang. Lalu command **crontab -e** untuk mengedit file cronjob dan membuat perintah automasi yaitu **43 9 * * * /home/ubuntu_admin/kirim/transfer_file.sh** lalu save. Perintah tersebut akan melakukan eksekusi file script **transfer_file.sh** pada pukul 09:43 setiap hari. Sehingga cronjob akan berjalan sekitar 1 menit lagi dari sekarang.

```
ubuntu_admin@server1:~/kirim$ date
Fri Feb 21 09:41:50 UTC 2025
```

```
ubuntu_admin@server1:~/ kirim$ crontab -e
```

```
43 9 * * * /home/ubuntu_admin/kirim/transfer_file.sh
```

- 4) Pada sever2, setelah waktu yang ditentukan, lakukan command **ls -la terima/** untuk melihat list di dalam direktori **terima**. Terlihat bahwa file **server1.txt** telah ada tepat pada pukul 09:43.

```
ubuntu_admin@server2:~$ ls -la terima/
total 8
drwxrwxr-x 2 ubuntu_admin ubuntu_admin 4096 Feb 21 09:43 .
drwxr-xr-x 6 ubuntu_admin ubuntu_admin 4096 Feb 21 09:29 ..
-rw-rw-r-- 1 ubuntu_admin ubuntu_admin    0 Feb 21 09:43 server1.txt
```

- 5) Untuk melakukan perintah sesuai dengan soal, cronjob yang dibuat adalah **0 23 * * 1 /home/ubuntu_admin/kirim/transfer_file.sh**. Cronjob tersebut akan melakukan automation kirim file **server1.txt** dari **server1** ke **server2** setiap hari senin jam 11 malam atau jam 23:00.

```
0 23 * * 1 /home/ubuntu_admin/kirim/transfer_file.sh
```