

# Set Up an Email Server on an Azure VM

## Step 1: Prepare Azure VM and No-IP Subdomain

### 1. Provision an Azure Ubuntu Server 22.04 VM:

- Example: VM Name: emailserver-vm(NSG allows 22,80,443,587)
- Public IP: (your actual assigned public IP)

Virtual machines    Get started

+ Create    ↺ Switch to classic    ⌚ Reservations    ⚙️ Manage view    ↻ Refresh    ⬇️ Export to CSV    🔗 Open query    |    ⋮    📊 Group by none

🔔 You are viewing a new version of Browse experience. Click here to access the old experience.

🔍 Filter for any field...    Subscription equals all    Type equals all    Resource Group equals all    Location equals all    + Add filter

<input type="checkbox"/>	Name ↑	Subscription	Resource Group	Location	Status	Operating syst...	Size	Public IP ad
<input type="checkbox"/>	🔗 emailserver-vm	⋮ Azure subscript...	emailserver-rg	Central India	Running	Linux	Standard_B1s	98.70.31.239

### 2. Set up a free No-IP subdomain:

- Register/login at [noip.com](https://noip.com)
- Create hostname: yourtestmail.ddns.net
- Point the IPv4 value to your Azure VM public IP

noip

English    ?    🛒    🔔    uananth.1703@gmail.com

Dashboard

DDNS & Remote Access

DNS Records

DDNS Keys

Dynamic Update Client

Update Clients

Device Configuration Assistant

Managed DNS

Domains

Email

Server Monitoring

SSL Certificates

Records

My No-IP > Records

Create Hostname

Free Dynamic DNS: 1/1

Upgrade to Enhanced Dynamic DNS

Free Dynamic DNS accounts are limited to 1 Hostname, currently there is 1 Hostname in your account.

Upgrade Now

ddns.net 1 record

No-IP Owned

Add Record

Name	Type	Content	TTL	Last Update
yourtestmail	A	98.70.31.239	60	Sep 14, 2025 23:47:27

Feedback

my@f59bcd64 2025-09-12T16:23:53Z web03

The screenshot shows the ddns.net web interface. At the top, it says 'ddns.net 1 record' and 'No-IP Owned'. There is an 'Add Record' button. Below is a table with columns: Name, Type, Content, TTL, and Last Update. The table is currently empty. Below the table, there is a form to add a new record. The 'Type' dropdown is set to 'A'. The 'Host' field contains 'yourtestmail'. The 'Content' field contains '98.70.31.239'. The 'TTL' dropdown is set to '60 seconds'. There are checkboxes for 'Wildcard' and 'Enable Dynamic DNS', both of which are unchecked. At the bottom right, there are 'Cancel' and 'Save' buttons. A 'Feedback' button is also visible on the right side.

## Step 2: Install and Configure Postfix

SSH into your VM:

```
bash
```

```
ssh azureuser@<your-azure-vm-public-ip>
```

Install Postfix and mailutils:

```
bash
```

```
sudo apt update && sudo apt install postfix mailutils -y
```

- During configuration choose: "Internet Site"
- Set system mail name: yourtestmail.ddns.net

Edit Postfix config:

```
bash
```

```
sudo nano /etc/postfix/main.cf
```

Set (replace values if your VM's real hostname is different):

```
text
```

```
myhostname = yourtestmail.ddns.net
```

```
mydomain = ddns.net
```

```
myorigin = /etc/mailname
```

```
inet_interfaces = all
```

```
inet_protocols = all
```

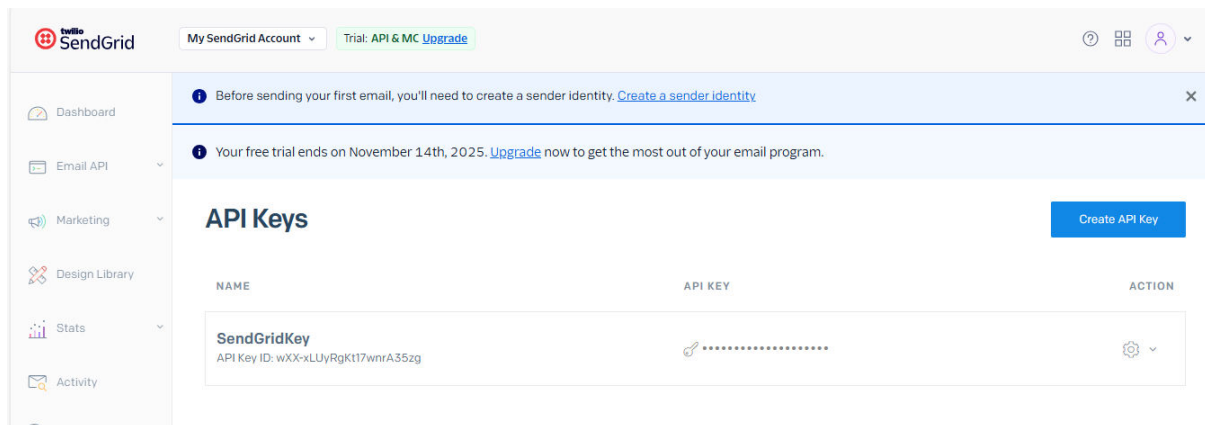
```
mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain
```

home\_mailbox = Maildir/

---

### Step 3: Create and Use SendGrid API Key

- Sign up at [SendGrid](#)
- Go to Settings > API Keys > Create API Key
- Name: AzureVMRelay, Permissions: Full Access (for SMTP)
- Save the generated API key (format: SG.xxxxxxx)



### Step 4: Configure Postfix to Use SendGrid as Relay

Add to /etc/postfix/main.cf:

text

relayhost = [smtp.sendgrid.net]:587

smtp\_sasl\_auth\_enable = yes

smtp\_sasl\_password\_maps = static:apikey:SG.xxxxxxx # (use your actual API key here)

smtp\_sasl\_security\_options = noanonymous

smtp\_tls\_security\_level = encrypt

smtp\_tls\_CAfile = /etc/ssl/certs/ca-certificates.crt

smtp\_use\_tls = yes

---

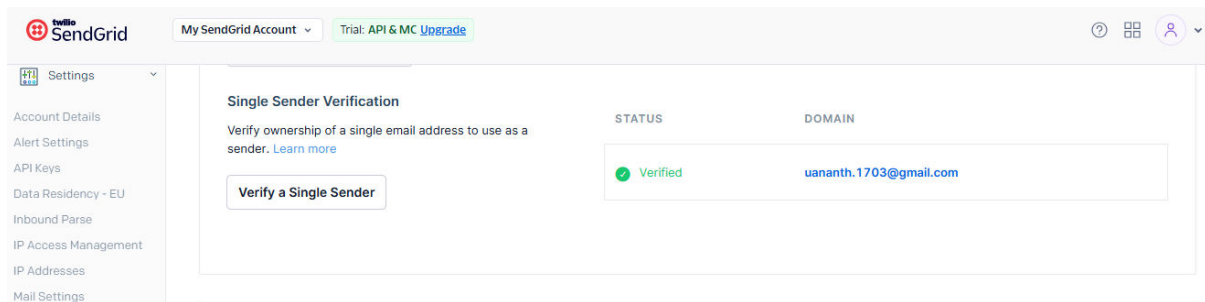
### Step 5: Verify Single Sender in SendGrid

1. In SendGrid, go to Settings > Sender Authentication > Single Sender Verification

2. Click "Get Started" and enter:

- Name: Ananth
- Email: [uananth.1703@gmail.com](mailto:uananth.1703@gmail.com)
- Fill in address details as per your info

3. Confirm verification by clicking the link sent to your Gmail inbox



---

## Step 6: Set Postfix to Always Use Verified Sender

Edit `/etc/postfix/sender_canonical`:

```
bash
```

```
sudo nano /etc/postfix/sender_canonical
```

Add:

```
text
```

```
azureuser@emailserver-vm uananth.1703@gmail.com
```

Create the lookup map:

```
bash
```

```
sudo postmap /etc/postfix/sender_canonical
```

In `/etc/postfix/main.cf` add or uncomment:

```
text
```

```
sender_canonical_maps = hash:/etc/postfix/sender_canonical
```

Reload Postfix:

```
bash
```

```
sudo systemctl reload postfix
```

---

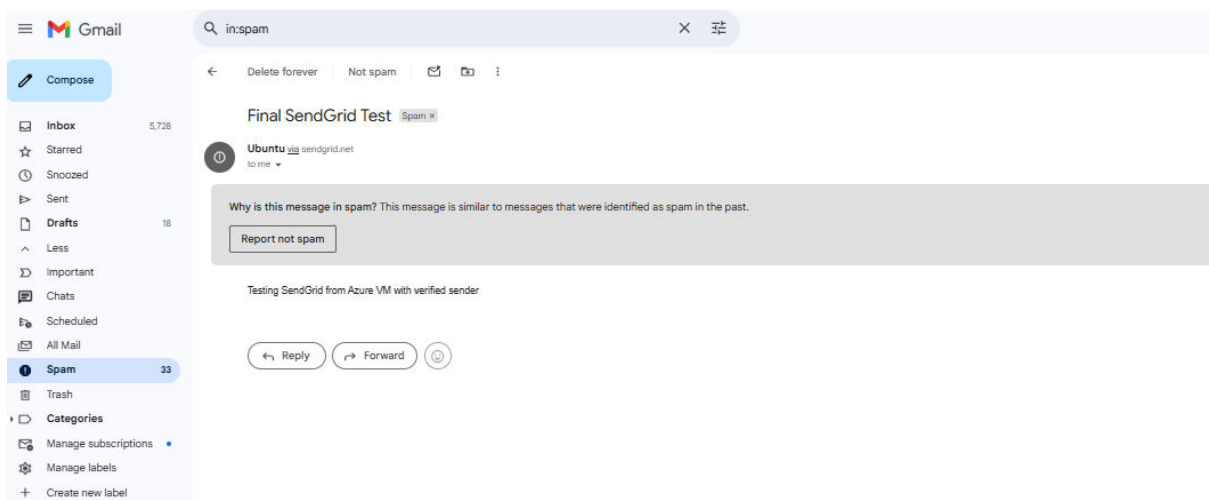
## Step 7: Send and Verify Test Mail

Send:

bash

```
echo "Testing SendGrid from Azure VM with verified sender" | mail -s "Final SendGrid Test" uananth.1703@gmail.com
```

Check your Gmail inbox (including Spam) for the message.

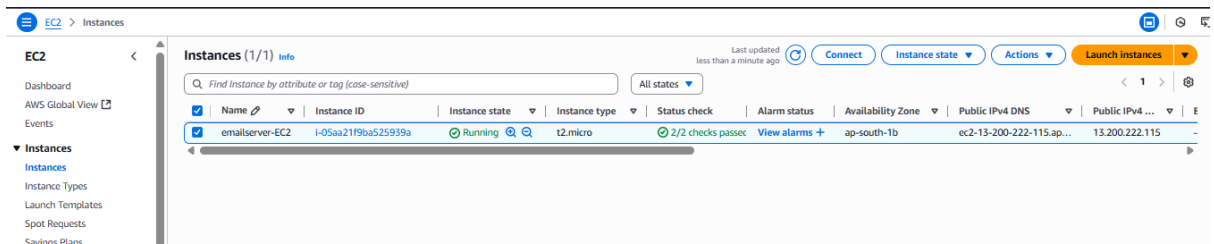


# Complete Email Server Setup on AWS Ubuntu with SendGrid Relay

## Step 1: Prepare AWS EC2 Ubuntu Server and No-IP Subdomain

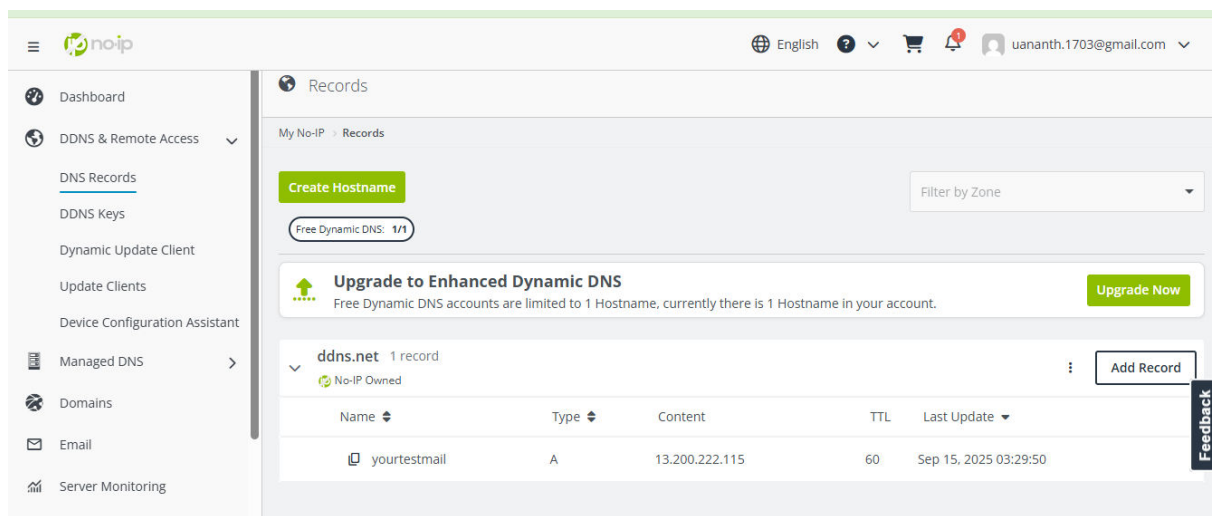
1. Launch an AWS EC2 instance with:

- Ubuntu Server 22.04
- Public Elastic IP (e.g. 13.200.222.115)
- Security group allows inbound ports 22 (SSH), 80 (HTTP), 587 (SMTP submission)



## 2. Register and create a free No-IP hostname:

- Hostname: yourtestmail.ddns.net
- Point this hostname to your AWS Elastic IP



## Step 2: Install and Configure Postfix

SSH into the EC2 instance:

```
bash
```

```
ssh -i "keypair.pem" ubuntu@13.200.222.115
```

Install Postfix and mailutils:

```
bash
```

```
sudo apt update && sudo apt install postfix mailutils -y
```

- Choose "Internet Site" when prompted for mail configuration
- Set system mail name as:

```
text
```

```
yourtestmail.ddns.net
```

Edit Postfix config:

bash

sudo nano /etc/postfix/main.cf

Set these essential parameters (modify if your hostname differs):

text

myhostname = yourtestmail.ddns.net

mydomain = ddns.net

myorigin = /etc/mailname

inet\_interfaces = all

inet\_protocols = all

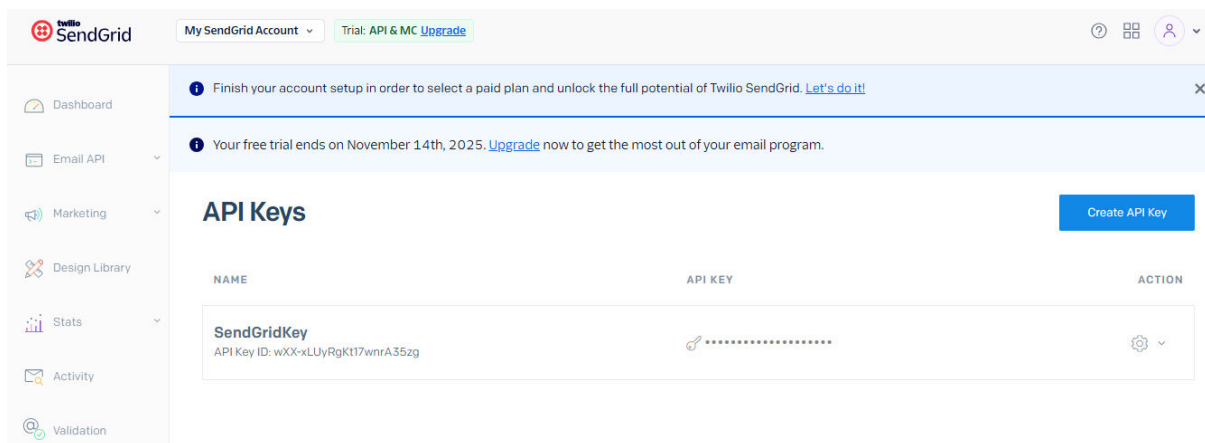
mydestination = \$myhostname, localhost.\$mydomain, localhost, \$mydomain

home\_mailbox = Maildir/

---

### Step 3: Create SendGrid API Key

- Register/login at [SendGrid](#)
- Navigate: Settings > API Keys > Create API Key
- Name it e.g. AWSRelay, set permissions as Full Access
- Copy your API key (format: SG.xxxxxxxx...)



---

### Step 4: Configure Postfix to Use SendGrid

In /etc/postfix/main.cf append:

text

```
relayhost = [smtp.sendgrid.net]:587
```

```
smtp_sasl_auth_enable = yes
```

```
smtp_sasl_password_maps = static:apikey:SG.xxxxxxxx... # Replace with your API key exactly
```

```
smtp_sasl_security_options = noanonymous
```

```
smtp_tls_security_level = encrypt
```

```
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
```

```
smtp_use_tls = yes
```

Save and reload Postfix:

```
bash
```

```
sudo systemctl reload postfix
```

---

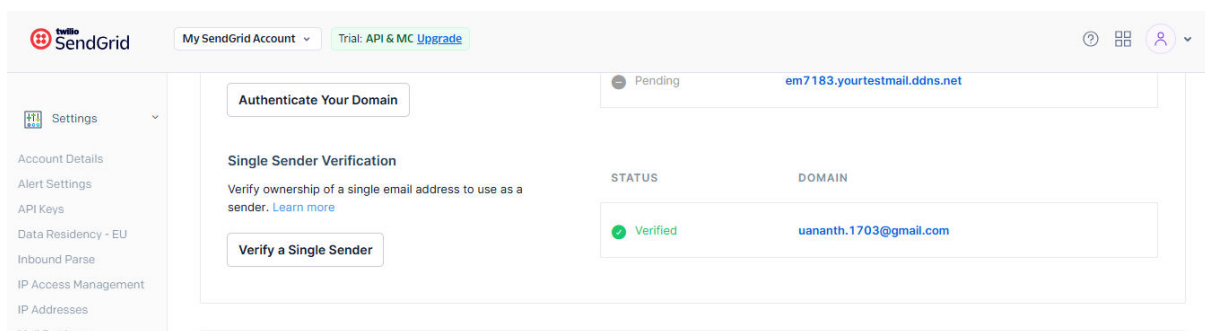
## Step 5: Verify Single Sender on SendGrid

- Go to SendGrid dashboard > Settings > Sender Authentication > Single Sender Verification
- Click "Get Started," enter your sending email, e.g.:

text

uananth.1703@gmail.com

- Fill in required contact details.
- Click the verification link sent to your email inbox.



---

## Step 6: Configure Postfix Sender Rewriting

Create or edit sender canonical map:

```
bash
```



```
sudo nano /etc/postfix/sender_canonical
```

Add this line replacing with your actual hostname:

text

```
ubuntu@ip-172-31-8-96.ap-south-1.compute.internal uananth.1703@gmail.com
```

Save and generate the database:

bash

```
sudo postmap /etc/postfix/sender_canonical
```

Add the following to /etc/postfix/main.cf if not present:

text

```
sender_canonical_maps = hash:/etc/postfix/sender_canonical
```

Reload Postfix to apply all changes:

bash

```
sudo systemctl reload postfix
```

---

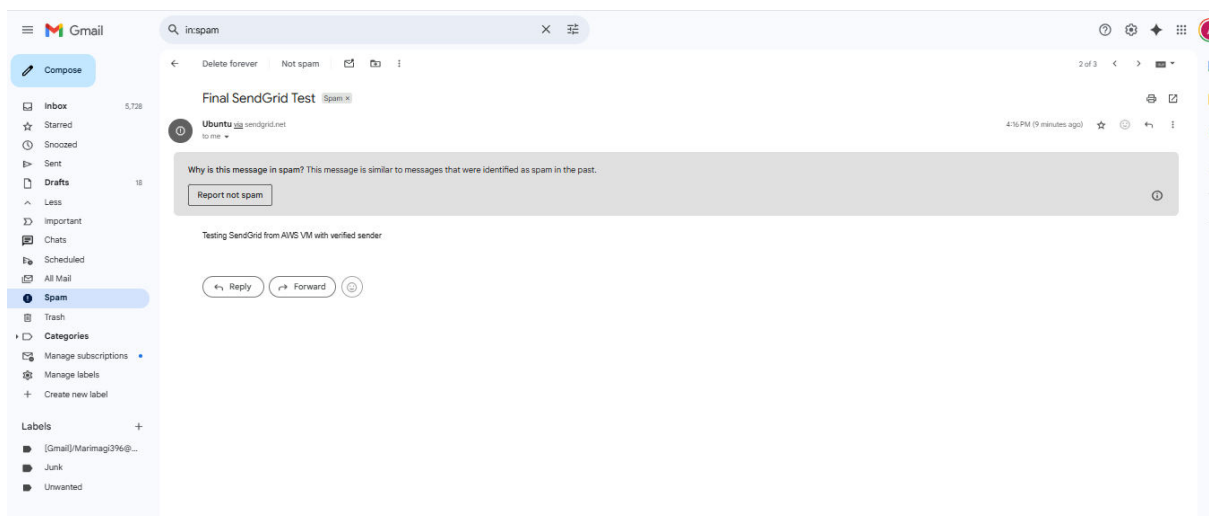
## Step 7: Send Test Email

Send a test email command:

bash

```
echo "Testing final SendGrid mail relay setup on AWS" | mail -s "SendGrid Relay Test" uananth.1703@gmail.com
```

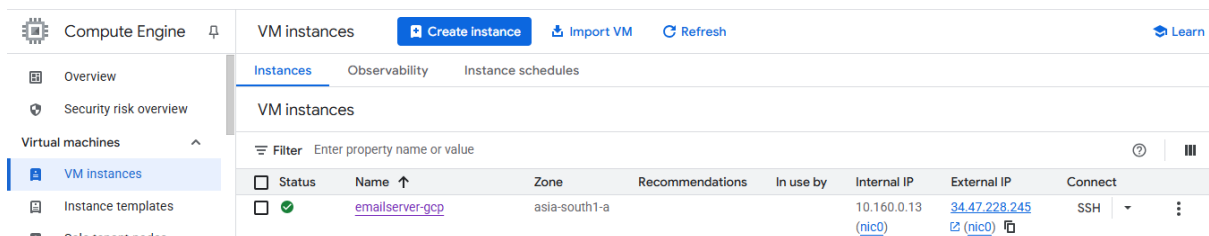
Check your Gmail inbox (including Spam folder) to verify receipt.



# Complete Email Server Setup on GCP Ubuntu VM Using SendGrid SMTP Relay (With SASL and Sender Canonical Mapping)

## Step 1: Prepare Google Cloud Compute Engine VM and No-IP Subdomain

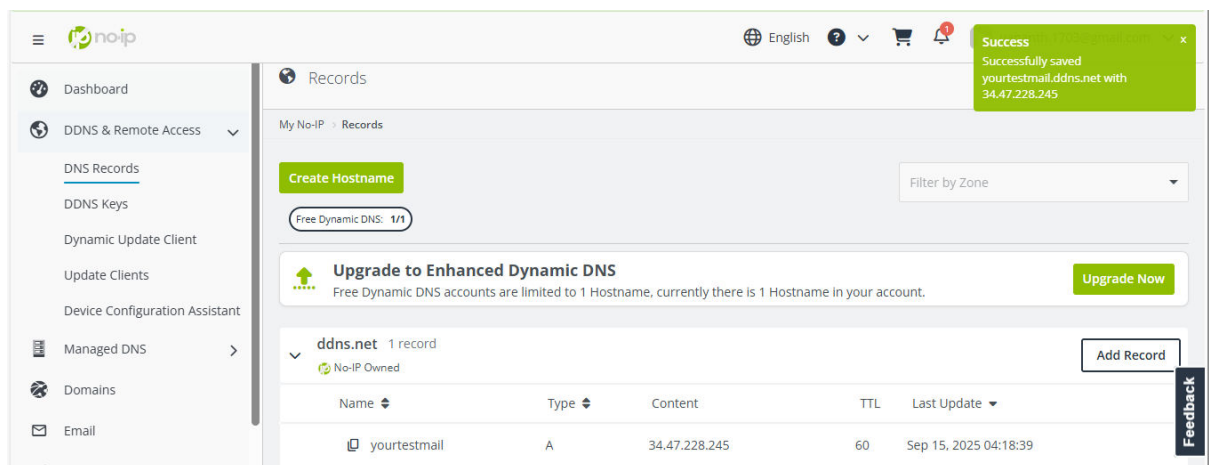
1. Create a GCP Compute Engine VM (Ubuntu 22.04 recommended).
2. Allocate a static external IP to your VM.
3. Configure GCP firewall to allow inbound ports:
  - 22 (SSH)
  - 80 (HTTP)
  - 587 (SMTP submission)



The screenshot shows the Google Cloud Platform interface for VM instances. The left sidebar has a menu with 'Compute Engine' selected, and 'VM instances' is highlighted under 'Virtual machines'. The main content area shows a table of VM instances. One instance named 'emailserver-gcp' is listed with a status of 'Running' (green checkmark), located in the 'asia-south1-a' zone. It has an internal IP of 10.160.0.13 and an external IP of 34.47.228.245. The 'Connect' column shows an SSH button and a 'nic0' link.

Status	Name	Zone	Recommendations	In use by	Internal IP	External IP	Connect
Running	emailserver-gcp	asia-south1-a			10.160.0.13	34.47.228.245	SSH

4. Create a free No-IP subdomain (e.g., yourtestmail.ddns.net) pointing to your VM's static IP.



The screenshot shows the No-IP DDNS management interface. A green success message at the top right states 'Successfully saved yourtestmail.ddns.net with 34.47.228.245'. The left sidebar has a menu with 'Dashboard', 'DDNS & Remote Access', 'Managed DNS', 'Domains', and 'Email'. The 'DDNS & Remote Access' section is expanded, showing 'DNS Records', 'DDNS Keys', 'Dynamic Update Client', 'Update Clients', and 'Device Configuration Assistant'. The 'DNS Records' section shows a table of records for the domain 'ddns.net'. One record is listed: 'yourtestmail' with type 'A' and content '34.47.228.245'. The table also shows TTL (60) and Last Update (Sep 15, 2025 04:18:39). A 'Feedback' button is visible on the right side of the table.

Name	Type	Content	TTL	Last Update
yourtestmail	A	34.47.228.245	60	Sep 15, 2025 04:18:39

## Step 2: Install Postfix and Required SASL Packages

SSH into your GCP VM:

```
bash
```

```
ssh ubuntu@<your-gcp-vm-public-ip>
```

Install required packages:

```
bash
```

```
sudo apt update
```

```
sudo apt install postfix mailutils libsasl2-2 sasl2-bin libsasl2-modules ca-certificates libsasl2-modules-db -y
```

- During Postfix install, choose Internet Site.
- Set system mail name as yourtestmail.ddns.net.

---

## Step 3: Configure SASL for Postfix

Create SASL config for Postfix:

```
bash
```

```
sudo mkdir -p /etc/postfix/sasl
```

```
sudo nano /etc/postfix/sasl/smtpd.conf
```

Add:

```
text
```

```
pwcheck_method: saslauthd
```

```
mech_list: plain login
```

Save and exit.

Restart SASL daemon:

```
bash
```

```
sudo systemctl restart saslauthd
```

```
sudo systemctl enable saslauthd
```

---

## Step 4: Configure Postfix (/etc/postfix/main.cf)

Edit main Postfix config:

```
bash
```

```
sudo nano /etc/postfix/main.cf
```

Ensure the following settings are present and correct (adjust hostname):

```
text
```

```
myhostname = emailserver-gcp.asia-south1-a.c.compact-flash-471109-h5.internal
```

```
mydomain = ddns.net
```

```
myorigin = /etc/mailname
```

```
inet_interfaces = all
```

```
inet_protocols = all
```

```
mydestination = $myhostname, yourtestmail.ddns.net, localhost.$mydomain, localhost,  
$mydomain
```

```
home_mailbox = Maildir/
```

```
relayhost = [smtp.sendgrid.net]:587
```

```
smtp_sasl_auth_enable = yes
```

```
smtp_sasl_password_maps = static:apikey:SG.xxxxxxxx-your-api-key-here
```

```
smtp_sasl_security_options = noanonymous
```

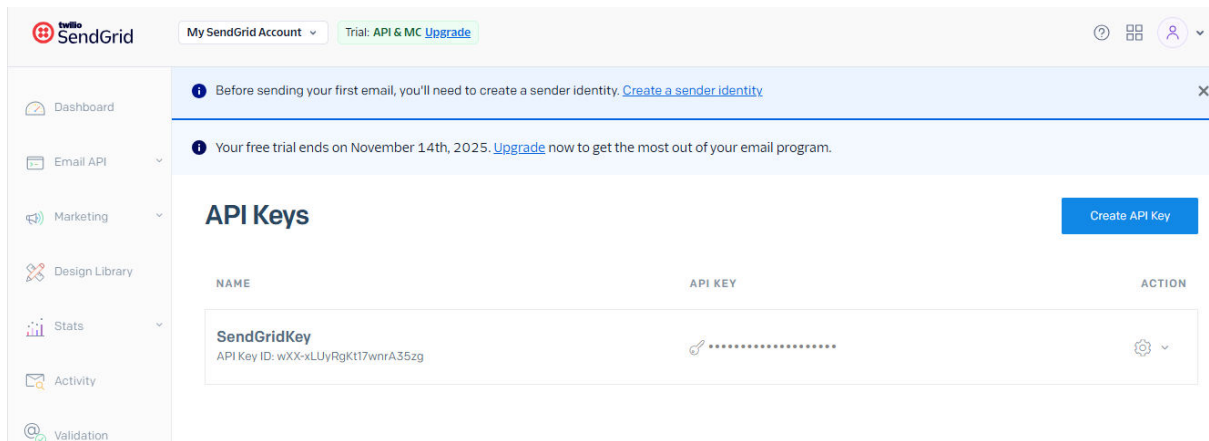
```
smtp_tls_security_level = encrypt
```

```
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
```

```
smtp_use_tls = yes
```

```
sender_canonical_maps = hash:/etc/postfix/sender_canonical
```

Replace SG.xxxxxxxx-your-api-key-here with your actual SendGrid API key.



---

## Step 5: Setup Sender Canonical Mapping

Create or edit `/etc/postfix/sender_canonical` to rewrite sender address:

```
bash
```

```
sudo nano /etc/postfix/sender_canonical
```

Add this line:

```
text
```

```
uananth_1703@emailserver-gcp.asia-south1-a.c.compact-flash-471109-h5.internal  
uananth.1703@gmail.com
```

Save and exit.

Generate hash map:

```
bash
```

```
sudo postmap /etc/postfix/sender_canonical
```

---

## Step 6: Reload Postfix to Apply Changes

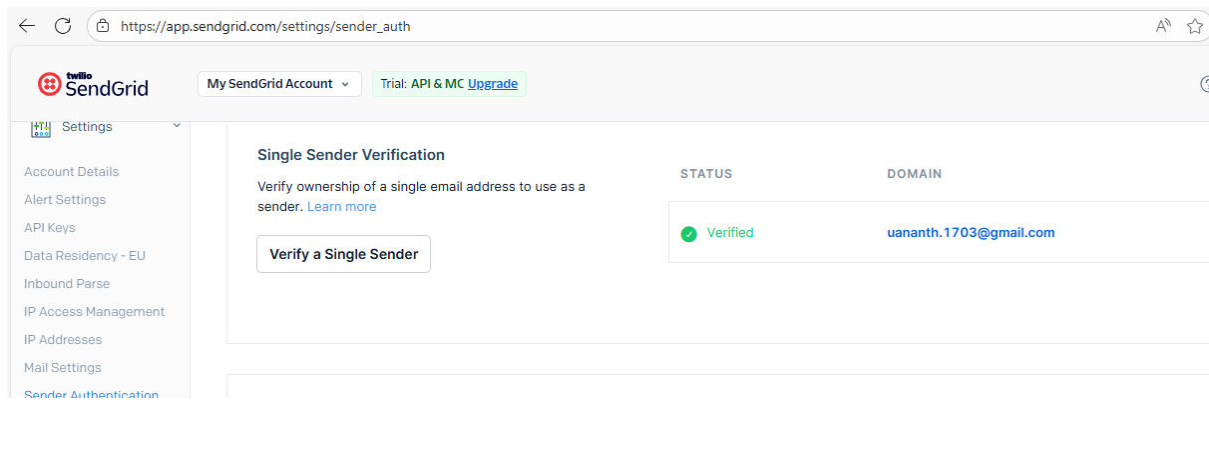
```
bash
```

```
sudo systemctl reload postfix
```

---

## Step 7: Verify Single Sender on SendGrid

- Verify your sender email (uananth.1703@gmail.com) as a Single Sender Verification in the SendGrid dashboard under Settings > Sender Authentication.
- Click verification link sent to your Gmail inbox.



## Step 8: Test Sending Email

Send a test mail from your VM:

```
bash
```

```
echo "Final successful SendGrid mail relay test" | mail -s "SendGrid GCP Test" uananth.1703@gmail.com
```

Check your Gmail inbox (including spam/junk folder) for email delivery.

