Phase 1: VM Creation (Performed on GCP)

1. Create a VM

```
gcloud compute instances create freeradius-server \
   --machine-type=n1-standard-4 \
   --zone=us-central1-a \
   --boot-disk-size=20GB \
   --image-family=ubuntu-2004-lts \
   --image-project=ubuntu-os-cloud \
   --scopes=https://www.googleapis.com/auth/cloud-platform \
   --tags=http-server,https-server,radius-server
```

2. Allow SSH Traffic (if not already allowed)

```
gcloud compute firewall-rules create allow-ssh --allow=tcp:22 –source-ranges=0.0.0.0/0
```

3. Allow RADIUS Traffic

```
gcloud compute firewall-rules create allow-radius --allow=udp:1812,udp:1813 –target-tags=radius-server
```

Phase 2: VM Initialization

1. Install necessary modules:

```
brew install mysql
brew install freeradius-server
brew install freeradius-server-mysql
```

2. Secure the MySQL Installation as per requirements:

```
sudo mysql_secure_installation
```

3. Start the sql Service, FreeRADIUS and enable necessary modules

```
brew services start mysql
sudo radiusd -X
sudo ln -s /usr/local/etc/freeradius-server/mods-available/sql /usr/local/etc/freeradius-server/mods-enabled/sql
```

4. Configure the Default Authentication Method:

```
sudo nano /etc/freeradius/3.0/sites-enabled/default
```

Phase 3:Creating and configuring the user

1. Log in to the MySQL Server and update the root password

```
mysql>  sudo mysql -u root -p


mysql>  ALTER USER  'root'@'localhost'  IDENTIFIED WITH  mysql_native_password  BY 'password';

mysql>  FLUSH PRIVILEGES;
```

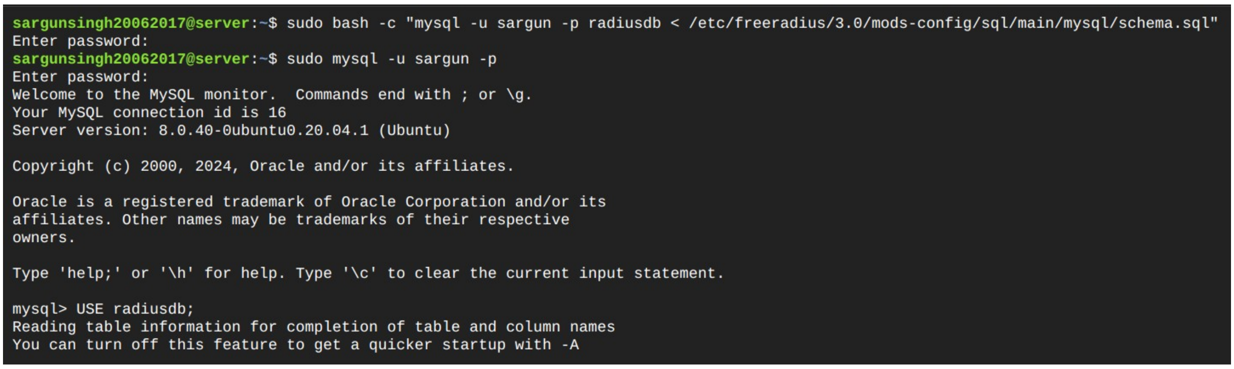
2. Create a new user (sargun) in this case and grant privelages

mysql> CREATE USER 'sargun'@'localhost' IDENTIFIED BY 'Free@Radius789';

mysql> GRANT ALL PRIVILEGES ON radiusdb.* TO 'sargun'@'localhost';

mysql> FLUSH PRIVILEGES;

mysql> EXIT;



```
sargunsingh20062017@server:~$ sudo bash -c "mysql -u sargun -p radiusdb < /etc/freeradius/3.0/mods-config/sql/main/mysql/schema.sql"
Enter password:
sargunsingh20062017@server:~$ sudo mysql -u sargun -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.40-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> USE radiusdb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

Phase 4: Using a default schema and updating the table

1. Importing the schema:
sudo bash -c "mysql -u sargun -p radiusdb </usr/local/etc/freeradius-server/mods-config/sql/main/mysql/schema.sql"

2. Login as the user
sudo mysql -u sargun -p

3. **Inserting Values:**

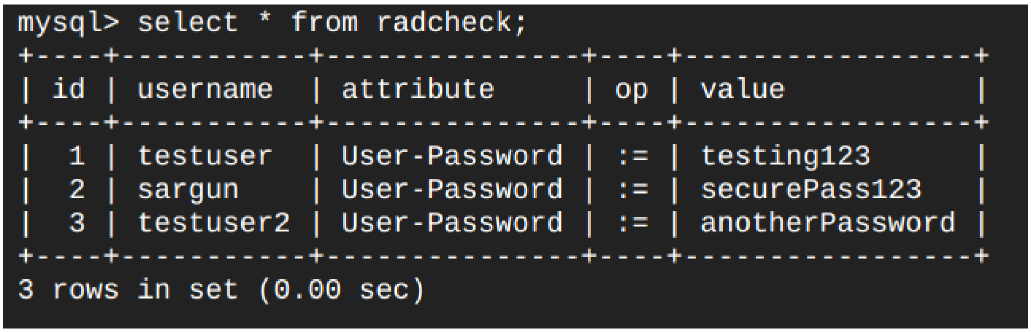
mysql> INSERT INTO radcheck (UserName, Attribute, op, Value) VALUES ('testuser', 'User-Password', ':=', 'testing123');

mysql> INSERT INTO radcheck (UserName, Attribute, op, Value) VALUES ('sargun', 'User-Password', ':=', 'securePass123');

mysql> INSERT INTO radcheck (UserName, Attribute, op, Value) VALUES ('testuser2', 'User-Password', ':=', 'anotherPassword');

mysql> SELECT * FROM radcheck;

mysql> EXIT;



```
mysql> select * from radcheck;
+----+-----------+---------------+----+-----------------+
| id | username  | attribute     | op | value           |
+----+-----------+---------------+----+-----------------+
|  1 | testuser  | User-Password | := | testing123      |
|  2 | sargun    | User-Password | := | securePass123   |
|  3 | testuser2 | User-Password | := | anotherPassword |
+----+-----------+---------------+----+-----------------+
3 rows in set (0.00 sec)
```

Phase 5: Modiying files for running a simple FreeRADIUS server

File 1: /usr/local/etc/freeradius-server/mods-available/sql:

1. Driver Changed:

•Original: driver = "rlm_sql_null" (or potentially rlm_sql_sqlite)
•Changed to: driver = "rlm_sql_mysql"
•Purpose: To enable the MySQL driver for database interaction.

2. Dialect changed:

•Original: dialect = "sqlite"
•Changed to : dialect = "mysql"

3.TLS Configuration Removed:

•Original: A tls block was present within the mysql section
•Changed to: The entire tls block was removed or commented out.
•Purpose: To disable TLS for the MySQL connection, as it wasn't necessary for a basic setup

4.MySQL Connection Details:

```
server = "localhost"
port = 3306
login = "sargun"
password = "Free@Radius789"
radius_db = "radiusdb"
```

5. Set read_clients property equal to yes

```
read_clients = yes
```

File 2: /usr/local/etc/freeradius-server/sites-enabled/default

authorize Section Module
Original: -sql
Changed to: sql

File 3: /usr/local/etc/freeradius-server/clients.conf
Shared Secret:

- Changed to: secret = testing123
- Purpose: To set a shared secret for the localhost client, matching the secret used in the radtest command.

Phase 6: Testing the FreeRadius server with different users but same keys in files

1. Start the server

```
sudo systemctl start freeradius
```

2. Run the radtest command

```
radtest  <username>  <password>  <server-ip>  <port>  <shared-secret>
```

- **<username>** : The username for testing
- **<password>** : The password for the user
- **<server-ip>** : The RADIUS server is running locally
- **<port>** : The default RADIUS authentication port.
- **<shared-secret> :** The shared secret between the RADIUS client (radtest) and the server

3. Testing Cases

Case 1: radtest testuser testing123 127.0.0.1 1812 testing123

Case 2: radtest testuser2 anotherPassword 127.0.0.1 1812 testing123

Case 3: radtest sargun securePass123 127.0.0.1 1812 testing123

Case 1 & 2 Output

Case 1:

```
Sent Access-Request Id 153 from 0.0.0.0:40057 to 127.0.0.1:1812 length 78
User-Name = "testuser"
User-Password = "testing123"
NAS-IP-Address = 10.128.0.5
NAS-Port = 1812
Cleartext-Password = "testing123"
Received Access-Reject Id 153 from 127.0.0.1:1812 to 127.0.0.1:40057 length 38
Message-Authenticator = 0x7bc698590bbc6c7aad73032d669a71d0
(0) -: Expected Access-Accept got Access-Reject
```

Case 2: Sent Access-Request Id 47 from 0.0.0.0:33643 to 127.0.0.1:1812 length 79

User-Name = "testuser2"

User-Password = "anotherPassword"

NAS-IP-Address = 10.128.0.5

NAS-Port = 1812

Cleartext-Password = "anotherPassword"

Received Access-Reject Id 47 from 127.0.0.1:1812 to 127.0.0.1:33643 length 38

Message-Authenticator = 0xfca6fa2a6bccd67f052f80ec1f19705a

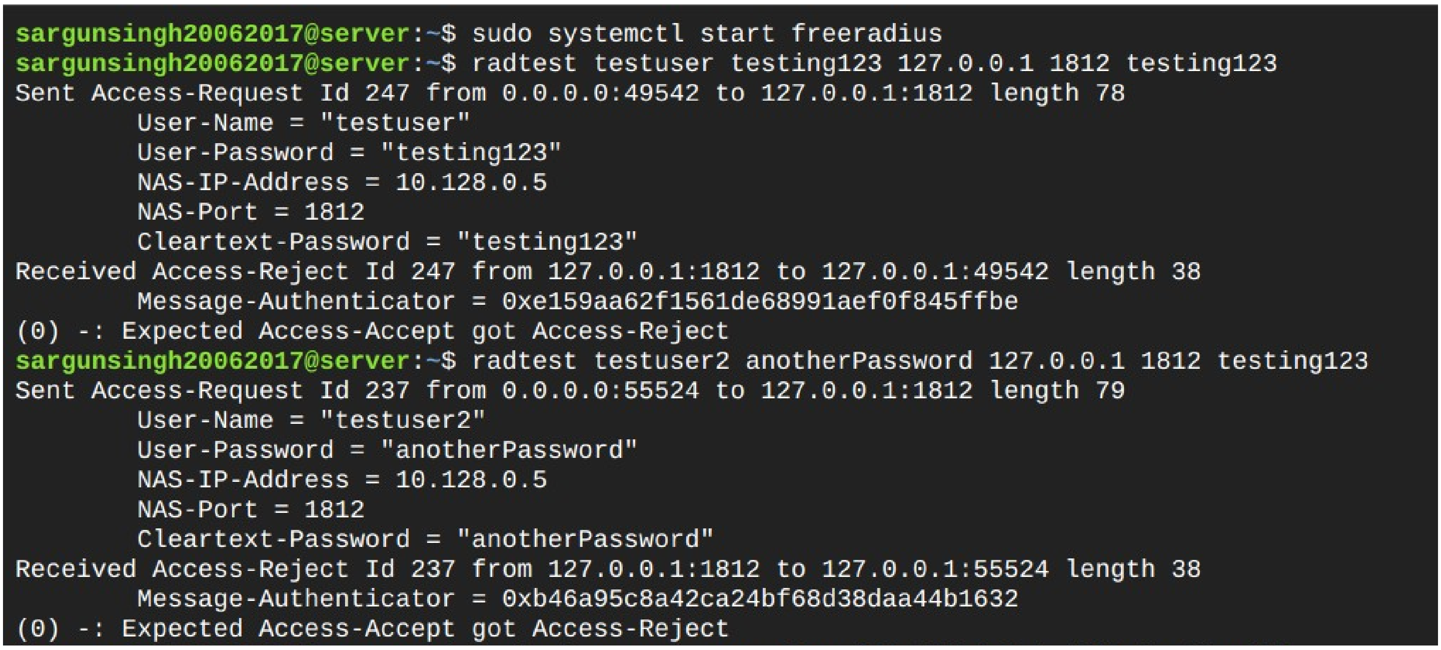
(0) -: Expected Access-Accept got Access-Reject

Why It Happens

- The RADIUS server is **configured for sargun**, and the database credentials are set for this user (login = "sargun", password = "Free@Radius789").
- The **shared secret** testing123 configured in /etc/freeradius/3.0/clients.conf is only valid for requests that come from a client identified as sargun.
  - When a request from testuser arrives, the server cannot authenticate it because testuser isn't configured in the RADIUS database.
  - **No match = Access-Reject**: The server replies with an Access-Reject message.

Technical Aspect

- When a request is sent by radtest for testuser, the server attempts to authenticate by checking the radcheck table in the MySQL database. If it doesn’t find a matching entry for testuser (or the password doesn't match), it sends an Access-Reject.
- The RADIUS server uses the shared secret (testing123) to verify that the request is coming from a trusted source. If the client’s shared secret does not match what's in clients.conf, the server rejects the request regardless of user credentials.



```
sargunsingh20062017@server:~$ sudo systemctl start freeradius
sargunsingh20062017@server:~$ radtest testuser testing123 127.0.0.1 1812 testing123
Sent Access-Request Id 247 from 0.0.0.0:49542 to 127.0.0.1:1812 length 78
        User-Name = "testuser"
        User-Password = "testing123"
        NAS-IP-Address = 10.128.0.5
        NAS-Port = 1812
        Cleartext-Password = "testing123"
Received Access-Reject Id 247 from 127.0.0.1:1812 to 127.0.0.1:49542 length 38
        Message-Authenticator = 0xe159aa62f1561de68991aef0f845ffbe
(0) -: Expected Access-Accept got Access-Reject
sargunsingh20062017@server:~$ radtest testuser2 anotherPassword 127.0.0.1 1812 testing123
Sent Access-Request Id 237 from 0.0.0.0:55524 to 127.0.0.1:1812 length 79
        User-Name = "testuser2"
        User-Password = "anotherPassword"
        NAS-IP-Address = 10.128.0.5
        NAS-Port = 1812
        Cleartext-Password = "anotherPassword"
Received Access-Reject Id 237 from 127.0.0.1:1812 to 127.0.0.1:55524 length 38
        Message-Authenticator = 0xb46a95c8a42ca24bf68d38daa44b1632
(0) -: Expected Access-Accept got Access-Reject
```

Case 3 Output

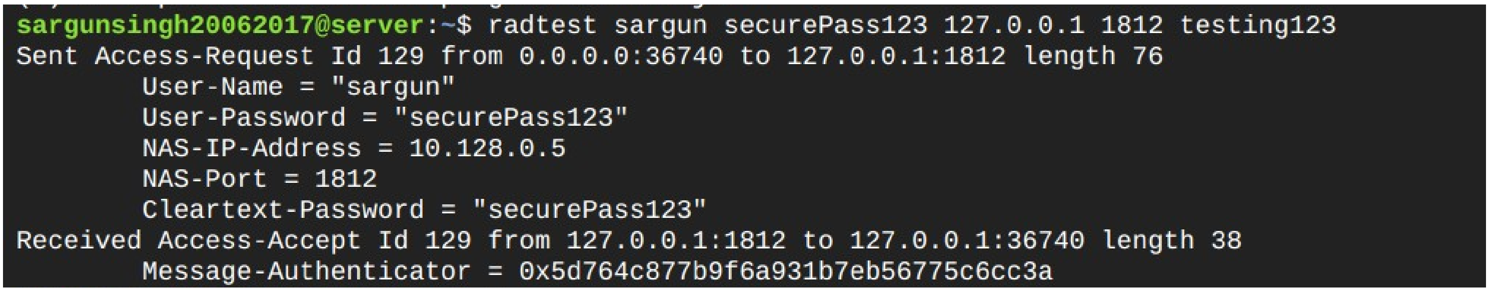
```
Sent Access-Request Id 208 from 0.0.0.0:44884 to 127.0.0.1:1812 length 76
User-Name = "sargun"
User-Password = "securePass123"
NAS-IP-Address = 10.128.0.5
NAS-Port = 1812
Cleartext-Password = "securePass123"
Received Access-Accept Id 208 from 127.0.0.1:1812 to 127.0.0.1:44884 length 38
Message-Authenticator = 0x5aba46d7f3768943ea011127e5a58b70
```

Why It Happens:

- The RADIUS server is **specifically configured for sargun**. The database entry for sargun exists in radcheck, and the password (securePass123) matches the stored value.
- The **shared secret** testing123 is correct for the server-client authentication, so the server recognizes the request as valid.
- **Successful Authentication**: Because the username (sargun) exists in the radcheck table, the password matches, and the shared secret is correct, the server sends an Access-Accept message.

Technical Aspect:

- When the radtest command sends a request with the username sargun, the server first checks the radcheck table in the MySQL database. It finds sargun with the correct password (securePass123) and authenticates the user.
- **Shared Secret Verification**: The server then checks that the shared secret (testing123) in clients.conf matches the one used by radtest. Since the secrets match, the server accepts the request, returning an Access-Accept response.



```
sargunsingh20062017@server:~$ radtest sargun securePass123 127.0.0.1 1812 testing123
Sent Access-Request Id 129 from 0.0.0.0:36740 to 127.0.0.1:1812 length 76
        User-Name = "sargun"
        User-Password = "securePass123"
        NAS-IP-Address = 10.128.0.5
        NAS-Port = 1812
        Cleartext-Password = "securePass123"
Received Access-Accept Id 129 from 127.0.0.1:1812 to 127.0.0.1:36740 length 38
        Message-Authenticator = 0x5d764c877b9f6a931b7eb56775c6cc3a
```