XMPP Server Setup on Debian EC2

`xmpp.drshn.me`

Goal

Create a lightweight, self-hosted XMPP server on AWS EC2 to allow private, encrypted communication using clients like Gajim or Monal.

1 Initial Setup

1.1 Install Prosody

```
sudo apt update
sudo apt install prosody certbot python3-certbot-nginx
```

Listing 1: Update and install Prosody, Certbot, and Nginx plugin

1.2 Set Domain DNS (A record)

- `xmpp.drshn.me` → EC2 Public IP

1.3 Enable SSL with Let's Encrypt

```
sudo certbot certonly --standalone -d xmpp.drshn.me
```

Listing 2: Obtain a standalone certificate

1.4 Install Certificate for Prosody

```
sudo install -o prosody -g prosody -m 640 \\
  /etc/letsencrypt/live/xmpp.drshn.me/fullchain.pem \\
  /etc/prosody/certs/xmpp.drshn.me.crt

sudo install -o prosody -g prosody -m 640 \\
  /etc/letsencrypt/live/xmpp.drshn.me/privkey.pem \\
  /etc/prosody/certs/xmpp.drshn.me.key
```

Listing 3: Copy certificate and key into Prosody's directory

2 Prosody Configuration

Virtual Host Configuration

Path: `/etc/prosody/conf.avail/xmpp.drshn.me.cfg.lua`

```
VirtualHost "xmpp.drshn.me"
    enabled = true
    ssl = {
        key = "/etc/prosody/certs/xmpp.drshn.me.key";
        certificate = "/etc/prosody/certs/xmpp.drshn.me.crt";
    }
    allow_registration = false
    modules_enabled = {
        "mam";        -- Message Archive Management
        "carbons";    -- Message Carbons
        "pep";        -- Personal Eventing Protocol
    }
```

Listing 4: VirtualHost settings

Enable Configuration

```
sudo ln -s \\
  /etc/prosody/conf.avail/xmpp.drshn.me.cfg.lua \\
  /etc/prosody/conf.d/xmpp.drshn.me.cfg.lua
sudo systemctl restart prosody
```

Listing 5: Activate site and restart Prosody

3 Permissions Fix (Let Prosody Read SSL Certs)

Create a script to add Prosody to the `ssl-cert` group and fix permissions.

Script: `ssl_grp.sh`

```
sudo usermod -aG ssl-cert prosody
sudo chgrp ssl-cert /etc/letsencrypt/archive/xmpp.drshn.me/privkey1.pem
sudo chmod 640   /etc/letsencrypt/archive/xmpp.drshn.me/privkey1.pem
sudo chmod 750   /etc/letsencrypt /etc/letsencrypt/archive \\
              /etc/letsencrypt/archive/xmpp.drshn.me
sudo chmod 750   /etc/letsencrypt/live /etc/letsencrypt/live/xmpp.drshn
    .me
```

Listing 6: Add user to ssl-cert group and adjust permissions

Run

```
chmod +x ssl_grp.sh
./ssl_grp.sh
```

Listing 7: Make and execute the script

4 Certificate Debugging

```
openssl s_client -connect xmpp.drshn.me:5222 -starttls xmpp
```

Listing 8: Test TLS handshake on port 5222

Expect: `Verify return code:  0 (ok)`

5 User Management

Add a User

```
sudo prosodyctl adduser talk@xmpp.drshn.me
```

Listing 9: Create a new XMPP account

6 Client Testing

From Arch Linux (Gajim)

- Add account: `talk@xmpp.drshn.me`
- Enable OMEMO in chat settings

From macOS (Monal)

- Install: `brew install -cask monal`
- Add account and test encrypted chats

7 Final Checklist

- [x] Prosody running with proper certs
- [x] TLS works on port 5222
- [x] OMEMO encryption enabled
- [x] MAM (message archive) working
- [x] DNS record resolves to correct IP

8 Future Enhancements

- Allow self-registration via web or curl (build `/register` API)
- Add `mod_smacks` for stream resumption
- Push notifications (XEP-0357)
- Host public room or anonymous gateway