KTM Logistics Web Server & Domain Setup Documentation

Overview

This document describes the process of setting up a production-ready web server for **ktm-logistics.com** and **www.ktm-logistics.com**, including DNS configuration, Apache VirtualHost setup, and SSL/TLS certificate installation via Let's Encrypt (Certbot). It also outlines standard practices and recommendations for long-term management.

Objectives

- Host `ktm-logistics.com` on an Apache2 web server running Ubuntu.
- Enable `www.ktm-logistics.com` (subdomain) to point to the same server.
- Secure the domain with HTTPS using Let's Encrypt SSL certificates.
- Ensure proper DNS resolution and redirect traffic consistently.

Steps Performed

1. Domain & DNS Configuration

- **Registrar**: The domain `ktm-logistics.com` was registered.
- **DNS Management**: Amazon Route 53 was used as the DNS hosting provider.
- **Records Configured**:
- **A Record**: `ktm-logistics.com → 51.21.222.101` (public IP of EC2 server).
- **CNAME Record**: `www.ktm-logistics.com → ktm-logistics.com`.
- Verified with:

```
dig +short www.ktm-logistics.com
```

Result:

```
ktm-logistics.com.
51.21.222.101
```

✅Confirms that `www` resolves correctly to the same IP.

Why: - The **A record** makes the root domain point to the server. - The **CNAME record** ensures `www` resolves to the root domain, avoiding duplicate management of IP addresses.

---

2. Apache VirtualHost Configuration

- Created a VirtualHost for port **80 (HTTP)** and **443 (HTTPS)**.
- Config file `/etc/apache2/sites-available/ktm-logistics.conf`:

```
<VirtualHost *:80>
    ServerName ktm-logistics.com
    ServerAlias www.ktm-logistics.com
    DocumentRoot /var/www/html

    <Directory /var/www/html>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/ktm_error.log
    CustomLog ${APACHE_LOG_DIR}/ktm_access.log combined
</VirtualHost>
```

- Enabled the site and reloaded Apache:

```
sudo a2ensite ktm-logistics.conf
sudo systemctl reload apache2
```

Why: - `ServerName` ensures Apache knows which domain to serve. - `ServerAlias` allows both `ktm-logistics.com` and `www.ktm-logistics.com` to work. - A dedicated VirtualHost file improves maintainability.

---

3. SSL Certificate Installation (Certbot)

- Installed **Certbot** with Apache plugin:

```
sudo apt install certbot python3-certbot-apache -y
```

- Issued certificate for both domains:

```
sudo certbot --apache -d ktm-logistics.com -d www.ktm-logistics.com
```

- Certbot created SSL-enabled VirtualHost `/etc/apache2/sites-available/ktm-logistics-le-ssl.conf`:

```
<IfModule mod_ssl.c>
<VirtualHost *:443>
    ServerName ktm-logistics.com
    ServerAlias www.ktm-logistics.com
    DocumentRoot /var/www/html

    <Directory /var/www/html>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    SSLCertificateFile /etc/letsencrypt/live/ktm-logistics.com/
fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/ktm-logistics.com/
privkey.pem
    Include /etc/letsencrypt/options-ssl-apache.conf
</VirtualHost>
</IfModule>
```

- Verified:

```
curl -I https://ktm-logistics.com
curl -I https://www.ktm-logistics.com
```

Why: - HTTPS is required for security, SEO ranking, and modern browser compliance. - Certbot automates renewal every 60 days (cron/systemd job created).

---

4. Redirects (Optional Best Practice)

To avoid duplicate content issues, redirect either `www` → root domain or root → `www`. Example (redirect all traffic to non-www):

```
<VirtualHost *:80>
    ServerName www.ktm-logistics.com
    Redirect permanent / https://ktm-logistics.com/
</VirtualHost>
```

Why: - Ensures a single canonical domain. - Improves SEO and avoids certificate confusion.

Standard Best Practices

1. **DNS Management**
2. Always use **A record** for the root domain.
3. Use **CNAME** for subdomains to simplify updates.
4. Keep TTL short (300s) during testing, then increase for stability.
5. **Apache Configuration**
6. One site per `.conf` file inside `/etc/apache2/sites-available/`.
7. Use `ServerAlias` for variations of the domain.
8. Store website files in `/var/www/<sitename>` instead of `/var/www/html` (cleaner for multiple sites).
9. **SSL/TLS**
10. Always use HTTPS (redirect HTTP → HTTPS).
11. Use Let's Encrypt for free auto-renewal.
12. Monitor certificate expiration with:

```
certbot certificates
```

13. **Security Hardening**
14. Disable directory listing (`Options -Indexes`).
15. Enable a firewall (UFW) to allow only ports `80` and `443`.
16. Regularly update Apache & Ubuntu security patches.
17. **Monitoring & Logging**
18. Monitor logs at `/var/log/apache2/`.
19. Use monitoring tools (CloudWatch, UptimeRobot, etc.).

Next Steps (Action Items)

- ✅Keep DNS records (`A` and `CNAME`) in Route 53.
- ✅Use Certbot auto-renew (`systemctl status certbot.timer`).
- 🔄Decide on **canonical domain** (with or without `www`) and add proper redirects.
- 📶Enable **UFW firewall**:

```
sudo ufw allow 'Apache Full'
sudo ufw enable
```

- ☣️Move site files to `/var/www/ktm-logistics` for better organization.
- 🆚Set up uptime monitoring and log rotation.

Conclusion

The domain `ktm-logistics.com` and its subdomain `www.ktm-logistics.com` are now: - Properly resolving through Route 53 DNS. - Hosted on Apache with VirtualHost configuration. - Secured with Let's Encrypt SSL certificates. - Ready for production with minor enhancements recommended (canonical redirects, monitoring, and firewall).

This setup follows industry best practices and ensures security, scalability, and maintainability.