LOGZILLA DOCUMENTATION

Using HTTPS

Enable HTTPS on the LogZilla web interface with self-signed or CA-signed certificates, including key generation, installation, and UI configuration

Administration · Generated June 11, 2026 · logzilla.ai/docs/administration/using-https

HTTPS provides encrypted communication for the LogZilla web interface, ensuring secure access to dashboards, configuration, and administrative functions. This guide covers certificate generation, installation, configuration, and the related web-UI settings.

Certificate Requirements

LogZilla supports both self-signed certificates and certificates from trusted Certificate Authorities (CAs). For production environments, CA-signed certificates are recommended for better security and user trust.

All examples in this guide use the following directory on the LogZilla host to store TLS key and certificate files:

```
/tmp/logzilla-certs
```

You can use a different directory if preferred; just adjust the paths in the commands accordingly.

Self-Signed Certificate Generation

Generate a self-signed certificate in a temporary directory:

```
mkdir -p /tmp/logzilla-certs
cd /tmp/logzilla-certs

openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
  -keyout logzilla.key \
  -out logzilla.crt
```

A prompt for certificate details will appear:

```
Country Name (2 letter code) [AU]: US
State or Province Name (full name) [Some-State]: California
Locality Name (eg, city) []: San Francisco
Organization Name (eg, company) [Internet Widgits Pty Ltd]: Company Name
Organizational Unit Name (eg, section) []: IT Department
Common Name (e.g. server FQDN or name) []: logzilla.company.com
Email Address []: admin@company.com
```

`.` and Enter can be pressed to leave any field blank if not used or not needed.

Important: The Common Name must match the hostname or IP address used to access the LogZilla web interface (for example, `logzilla.company.com` or the server's IP).

Modern browsers also validate the Subject Alternative Name (SAN) extension; ensure the hostname you use to access LogZilla is present there.

Certificate Installation

Using Self-Signed Certificates

After generating the certificate in `/tmp/logzilla-certs`, enable HTTPS:

```
logzilla https --set /tmp/logzilla-certs/logzilla.key /tmp/logzilla-certs/logzilla.crt
```

Using CA-Signed Certificates

To use a certificate obtained from a trusted Certificate Authority:

Copy the key and certificate to a directory (for example, `/tmp/logzilla-certs`):

```
mkdir -p /tmp/logzilla-certs
cp /path/to/logzilla-ca.key /tmp/logzilla-certs/
cp /path/to/logzilla-ca.crt /tmp/logzilla-certs/
```

Enable HTTPS using those files:

```
logzilla https --set /tmp/logzilla-certs/logzilla-ca.key /tmp/logzilla-certs/logzilla-ca.crt
```

The filenames are not important as long as the correct paths are used in the `logzilla https --set` command.

HTTPS Configuration Options

Enable HTTPS

Activate HTTPS for the primary web interface:

```
logzilla https --set /tmp/logzilla-certs/<keyfile>.key /tmp/logzilla-certs/<certificate>.crt
logzilla config HTTPS_PORT 443
logzilla restart -c front
```

Disable HTTPS

Return to HTTP-only access:

```
logzilla config HTTPS_PORT 0
logzilla restart -c front
```

Force HTTPS Redirects

Redirect all HTTP requests to HTTPS automatically:

```
logzilla config FORCE_HTTPS true
```

Disable forced HTTPS redirects:

```
logzilla config FORCE_HTTPS false
```

Verification

After enabling HTTPS, verify the configuration by accessing the LogZilla web interface using `https://` instead of `http://`. The browser should show a secure connection indicator.

For self-signed certificates, browsers will display a security warning that can be safely bypassed for internal use.

Certificate Renewal

Self-signed certificates expire after the specified validity period (365 days in the example above). For production environments, implement a certificate renewal process or use automated certificate management tools like Let's Encrypt.

Troubleshooting

Common Issues

- **Certificate path errors**: Ensure certificate files exist at the paths configured for `logzilla https --set` (for example, under `/tmp/logzilla-certs`) and are readable by LogZilla.
- **Common Name mismatch**: Verify the certificate Common Name matches the hostname or IP address used to access LogZilla.
- **Port conflicts**: Ensure port 443 (or the chosen HTTPS port) is available for HTTPS traffic.

Checking Certificate Details

View certificate information:

```
openssl x509 -in /tmp/logzilla-certs/<certificate>.crt -text -noout
```

Testing HTTPS Connection

Test the HTTPS configuration:

```
openssl s_client -connect hostname:443 -servername hostname
```

Replace `hostname` with the actual host or IP used to reach the LogZilla web interface.

Web Interface Port Settings

(Settings → System Settings → Front)

The **Front** section of System Settings controls which ports the LogZilla web UI and API listen on, and how HTTP/HTTPS are used.

Http Port

UI and API HTTP port

The TCP port used for **unencrypted** HTTP access to the primary LogZilla UI and API (for example, `80`).

- Set to `0` to disable HTTP on the primary UI.
- Common values: `80` (default) or another port (for example, to use LogZilla with an external reverse proxy).

Https Port

UI and API HTTPS port

The TCP port used for **encrypted** HTTPS access to the primary LogZilla UI and API.

- Set to `0` to disable HTTPS on the primary UI.
- Typical production value: `443`.
- Must correspond to the port set with `logzilla config HTTPS_PORT`.

Http Port Ui2

UI and API HTTP port for UI2 (experimental)

HTTP port for the **UI2** interface (new UI, version 2) and its API.

- Leave at the default (for example, `8080`) unless explicitly using the UI2 frontend.
- Set to `0` to disable HTTP access for UI2.

Https Port Ui2

UI and API HTTPS port for UI2 (experimental)

HTTPS port for the **UI2** interface and its API.

- Default example: `8889`.
- Set to `0` to disable HTTPS for UI2.
- If UI2 is exposed over HTTPS, ensure the certificate configured via `logzilla https --set` is valid for the host/port combination you will use.

Force Https

Use HTTPS instead of HTTP

Controls whether HTTP requests are automatically redirected to HTTPS.

- **On** —- All HTTP requests to the UI/API are redirected to the configured HTTPS port. This corresponds to `logzilla config FORCE_HTTPS true` and is recommended for production deployments.
- **Off** —- Both HTTP and HTTPS are allowed. Users can access the UI via either scheme.

Note that forcing HTTPS only has an effect if an HTTPS port is configured and reachable.

Http Client Body Buffer Size Mb

If client body request is above this size, it will be buffered on the disk, which can make processing slower. This affects mostly Kinesis receiver requests.

Maximum size (in megabytes) that LogZilla will buffer **in memory** for the HTTP request body. When a request body exceeds this size:

- The body is temporarily buffered to disk instead of memory.
- This reduces memory usage but can slow down very large requests.

Guidelines:

- Default: `50` MB (suitable for most installations).
- Increase this value if large payloads are frequently sent and there is sufficient memory.
- Decrease this value if memory pressure is a concern and slower large-request handling is acceptable (for example, some Kinesis or bulk ingestion endpoints).

Changes to these settings usually require a front-end restart:

```
logzilla restart -c front
```

to fully apply.