SCHLEIFENBAUER
LIVING FOR THE POWER TO DELIVER

About the firmware version 2.60 update

Last Modified: February 18th, 2020

What's included in 2.60 (changes since 2.58)?

New features:

- TLS encryption supporting the following private keys, RSA1024 and Elliptic Curve prime256v1 and secp384r1.

Notes:

- See following pages for more technical info.
- This version can only be installed after 2.58 is installed. When installing from an older version than 2.58 first upgrade to 2.58. Check the release notes of 2.58 for installation instructions.
- Documentation for SPDM 2.50 improves the descriptions used in 2.44, no functionality has been changed.
- The SNMP MIB file is unchanged since 2.44 and can be used for all later versions.

Installing this firmware

Installation of firmware is preferred by using the Schleifenbauer Product Service Tool (SPST version 1.1.7). There is a specific *SPST user manual* available on the Schleifenbauer website.

Notes:

- Uploading 2.xx bin files in the Gateway's web interface is not possible!
- While updating or restarting the devices, power distribution will not be interrupted.

Key Generation:

Certificates with a public and private key can be uploaded for with the web interface. The certificates are used for the for the "Authentication" phase of the handshake. These certificates can be generated in 3 steps:
1. Generate a private key
2. Generate a CSR (Certificate Signing Request) with the private key.
3. Create a self-signed certificate with the CSR.

The following private keys will be supported:
1. RSA (1024 bit)
2. Elliptic Curve (secp256r1/ prime256v1 and secp384r1)

TLS Cipher Suites Supported:

The TLS cipher suites are listed in the following format (example):

TLS_DHE_RSA_WITH_AES_256_GCM_SHA384

1. TLS prefix
2. Key Exchange algorithm (DHE) (Excluded for static keys)
3. Authentication algorithm (RSA)
4. Encryption algorithm (AES)
5. Encryption strength (256)
6. Encryption mode (GCM)
7. MAC (SHA284)

The certificate uploaded is used for the "Authentication algorithm" (RSA or Elliptic Curve). Then other options (Key Exchange, Encryption etc) are selected from what the client lists as supported in the "Client Hello". The following cipher suites are supported:

Static Key with RSA:
MBEDTLS_TLS_RSA_WITH_AES_128_GCM_SHA256
MBEDTLS_TLS_RSA_WITH_AES_128_CBC_SHA256
MBEDTLS_TLS_RSA_WITH_AES_256_CBC_SHA256
MBEDTLS_TLS_RSA_WITH_AES_128_CBC_SHA
MBEDTLS_TLS_RSA_WITH_AES_256_CBC_SHA

Ephemeral Elliptic-curve Diffie-Hellman with elliptic curve:
MBEDTLS_TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
MBEDTLS_TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
MBEDTLS_TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
MBEDTLS_TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256

Elliptic-curve Diffie-Hellman with elliptic curve:
MBEDTLS_TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
MBEDTLS_TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA
MBEDTLS_TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
MBEDTLS_TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256

Note on self-signed certificates:

At the moment only self-signed certificates are supported.

Certificate Generation:

OpenSSL can be used to generate the private keys and certificates. We can provide scripts for this purpose if requested.