Setting up a secure communication (SSL/TLS)

The D2000 system can be configured to ensure that communication between the server and clients takes place through a secure encrypted communication channel. Security is implemented by **Transport Layer Security** (TLS v1.3).

The following steps are required to enable secure communication:

1. For the server, it is necessary to obtain/generate the encryption key and certificate. The certificate has to be distributed to the client processes.

The key and certificate can be generated, for example, using the **openssl** utility (https://slproweb.com/products/Win32OpenSSL.html).

Generating an encryption key

```
openssl genrsa -out server.key 4096
```

Generating a certificate signing request

```
openssl req -new -key server.key -out server.csr
```

Generating a self-signed certificate

```
openssl x509 -req -days 730 -in server.csr -signkey server.key -out server.crt
```

2. Setting up TLS support in the kernel registers

```
HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<instalacia>\cfg_<aplikacia>\TLS_Server\TLS_CertFile = c:\<path>\server.crt
HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<instalacia>\cfg_<aplikacia>\TLS_Server\TLS_KeyFile = c:\<path>\server.key
HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<instalacia>\cfg_<aplikacia>\TLS_Server\TLS_RequiredLevel = <level>
```

Setting the required security level of the connecting client <level>:

- **None** - kernel allows the client to connect with/without security
- **TLSNoPeerAuth** - kernel allows connection only from a client who communicates securely (currently, no client certificate is used or verified)

3. Setting up TLS support in the registers for clients

```
 HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<installation>\cfg_<application>\TLS_Client\TLS_TrustedCerts = c:
\<path>\server.crt
 HKEY_LOCAL_MACHINE\SOFTWARE\Ipesoft\<installation>\cfg_<application>\TLS_Client\TLS_RequiredLevel = <level>
```

TLS_TrustedCerts: the path to the server certificate. It is also possible to enter multiple certificates separated by a semicolon (;). This is applicable for redundant systems or in a certificate exchange process when both an old and a new server certificate can be configured.

TLS_RequiredLevel: the required security level of the connecting client:

- **None** - the client will connect to the kernel no matter if the kernel supports secure communication or not
- **TLSNoPeerAuth** - the client will only connect to the kernel supporting secure communication (but the kernel does not need to be verified by a certificate, i.e. its certificate is not compared with the *TLS_TrustedCerts* list)
- **TLSPeerAuth** - the client will only connect to the kernel ensuring secure communication whose certificate is in the *TLS_TrustedCerts* list

4. To use TLS, the client must **also start with /C<*application_name*> parameter** in addition to the usual parameters (/S, /RD or /RF)

The reason is to already know the name of the application before connecting to the application server and loading the parameters from the TLS registers (see point 3).
The alternative is to set the *DefaultApplication* parameter in the registry.
Note: we recommend setting the *DefaultApplication* in the registry so that it is not necessary to enter the parameter */C<application_name>* not only in all shortcuts on the desktop, but also when starting applications manually.

A client connecting to a server using TLS will write this in the log. If certificate verification is also required and the certificate is correctly verified, the word VERIFIED is in the log:

[2022-09-23 07:48:11.289]I CLIENT - Connecting to D2000 Server [localhost] TCP/IP|TLS...
[2022-09-23 07:48:11.348]I CLIENT - Connection established to D2000 Kernel V22.00.074 s380 [TCP/IP localhost:3119][TLSv1.3 VERIFIED]. ConnectionSqId = 1

The kernel accepting the client via TLS also writes this information in the log:

[2022-09-23 07:48:10.598]I BACKEND - RegistrateProces request from DispPC.HIP V22.00.074 s380 [TCP/IP 127.0.0.1:50481][TLSv1.3]. ClientName: DispPC.HIP
[2022-09-23 07:48:10.600]I SERVER - RegistrateProces OK. ClientName: DispPC.HIP Id: 7652 ComputerName: PC1PHUM1v SAS: 0

Change of keys and certificates

The D2000 Server reads the TLS configuration each time the client is connected, so it is possible to change the configuration of the D2000 Server (including the change of files with a certificate and private key) during the D2000 Server runtime without any restart of the D2000 Server.

Blog

You can read the blog about setting up TLS:

- Security - Configuring TLS (in 15 minutes).

Related pages:

D2000 system processes
Start parameters of processes