CHERRY eHealth Terminal ST-1506

Konfiguration VPN Client

KNOWLEDGE BASE

64410192-02, DE, Jun 2025 (KB Konfiguration VPN Client)

Inhaltsverzeichnis

1 „VPN Client“ des ST-1506

Das Cherry eHealth Terminal ST-1506 unterstützt seit der Firmwareversion 3.0.0 den Aufbau von VPN-Netzwerkverbindung bzw. VPN-Tunnel nach IPSec/IKEv2 (https://de.wikipedia.org/wiki/IPsec). Seit der Firmwareversion 4.0.47 ist eine VPN-Netzwerkverbindung auch über WireGuard möglich (https://de.wikipedia.org/wiki/WireGuard).

Dieses Dokument dient als Leitfaden für die Konfiguration des ST-1506 ab Firmwareversion 4.0.47 und bietet eine Beispielimplementierung eines IPSec VPN-Gateways.

2 Funktionsbeschreibung

Der **VPN Client** im ST-1506 unterstütz für IPSec zwei verschiedene Authentifizierungsmethoden, entweder EAP-MSCHAPv2, Authentisierung mit Benutzername und Passwort, oder EAP-TLS, Authentisierung mit Client Zertifikat und Private Key. Für WireGuard wird die Standard Authentifizierungsmethoden über Pre-shared key (PSK) unterstützt. Die Konfiguration des VPN-Clients findet über die Remote Schnittstelle des ST-1506 statt und wird in Kapitel 3 beschrieben.

Wird der VPN-Client im ST-1506 aktiviert, so wird sofort eine Verbindung zum VPN-Gateway aufgebaut und die Schnittstelle zum Konnektor über den VPN Kanal geroutet. Somit ist die Konnektor Schnittstelle nicht mehr aus dem lokalen Netzwerk erreichbar.

3 Konfiguration VPN Client

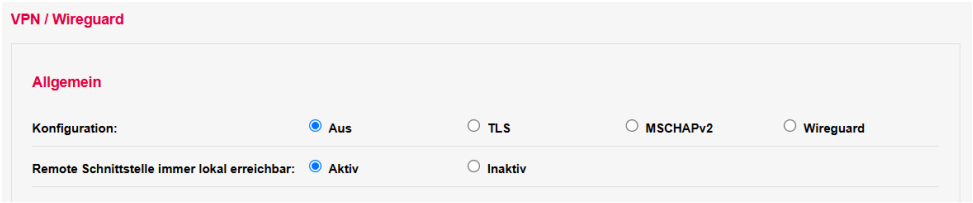
Im Folgenden wird die Konfiguration des VPN-Clients anhand des Browserinterface des ST-1506 beschrieben.

3.1 Allgemeine Konfiguration

Unter der Allgemeinen Konfiguration können folgende Elemente konfiguriert werden:

Konfiguration: Auswahl der zu verwendenden Authentisierung oder deaktivieren des VPN-Clients (Zustände: Aus, TLS, MSCHAPv2, Wireguard)

Remote Schnittstelle immer lokal erreichbar: Auswahl ob bei aktivem VPN auch lokal auf die Remote Schnittstelle zugegriffen werden kann (Standard aktiv)



3.2 Konfiguration für IPSec VPN Server

Unter VPN Server können die folgenden Elemente für IPSec konfiguriert werden:

Server IP Adresse: IP-Adresse des VPN-Gateways

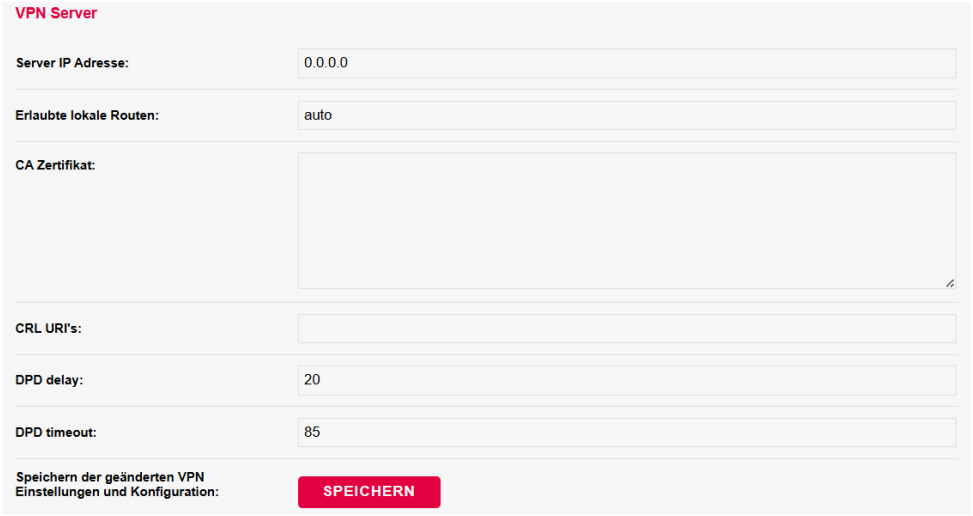
Erlaubte lokale Routen: Liste an lokalen IP-Adressen oder Routen, die nicht über das VPN geroutet werden. Für selbständige automatische Bearbeitung muss der Wert „auto" in das Feld eingetragen werden. (als Standard ist der Wert „auto" gesetzt)

CA Zertifikat: CA-Zertifikat des VPN-Gateways in base64-Konvertierung. Dieses Zertifikat muss self signed sein, da keine Chain unterstützt wird. Bei RSA Zertifikaten muss mindestens eine Schlüssellänge von 4096 bit verwendet werden.
Beispiel:
-----BEGIN CERTIFICATE-----
MIICOjCCAcCgAwIBAgIEKA/juTAKBggqhkjOPQQDAjBIMQswCQYDVQQGEwJB
...
iMDMoDbUbdW/qzUnhceJWzHCoDye8i4uMa9cNU2Deh2yxmBnyNYbqLMA==
-----END CERTIFICATE-----

CRL URI's: optional kann eine oder mehrere CRL's verwendet werden.
(Format: https://address.com/filename, http://address.com/filename, ...
Die verwendeten Adressen müssen korrekt und erreichbar sein für MSCHAP
(Setting "revocation=ifuri")

DPD delay: Intervall in Sekunden in dem Dead Peer Detection (DPD) Pakete gesendet werden (default 20 Sekunden).

DPD timeout: Timeout Wert in Sekunden nachdem die VPN-Verbindung beendet wird, wenn keine Antwort auf die DPD-Pakete erfolgt (default 85 Sekunden)

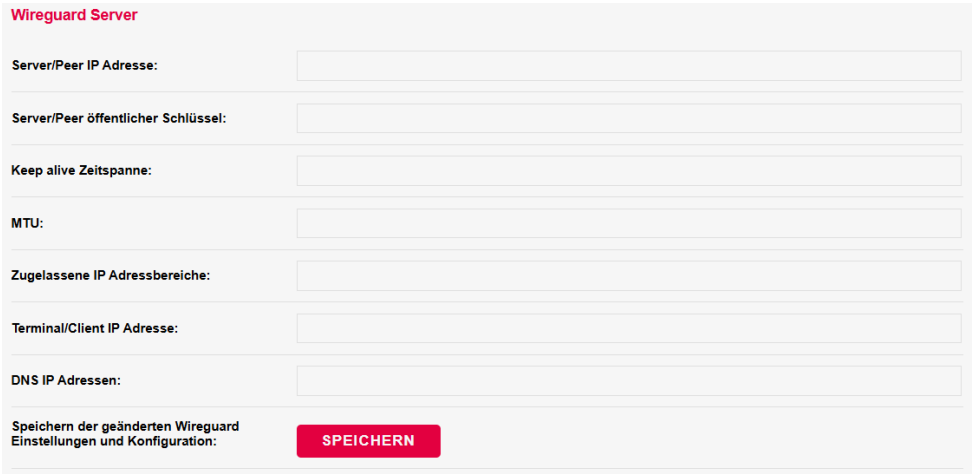


Durch Betätigen des Speicher Buttons werden die vorgenommenen Einstellungen gespeichert.

3.3 Konfiguration für Wireguard Server

Unter Wireguard Server können die folgenden Elemente konfiguriert werden:

Server/Peer IP Adresse:	IP-Adresse oder DNS-Name des WireGuard Server; WireGuard .conf Datenfeld: Endpoint
Server/Peer öffentlicher Schlüssel:	öffentlicher Schlüssel des WireGuard Servers WireGuard .conf Datenfeld: PublicKey
Keep alive Zeitspanne:	Wert in Sekunden, um die Zeitspanne festzulegen, die der VPN-Kanal ohne Datenverkehr aufrechterhalten bleibt. WireGuard .conf Datenfeld: PersistentKeepalive
MTU:	Die Maximum Transmission Unit (MTU) beschreibt die maximale Paketgröße eines Protokolls auf Layer 3. Für WireGuard wird standardmäßig ein MTU von 1420 verwendet. WireGuard .conf Datenfeld: MTU
Zugelassene IP Adressbereiche: **(maximal 8 Einträge)**	Liste von zugelassenen IP-Adressbereichen, die durch den WireGuard VPN-Tunnel geroutet werden sollen. WireGuard .conf Datenfeld: AllowedIPs
Terminal/Client IP Adresse:	IP-Adresse des Terminals im VPN-Netzwerk. WireGuard .conf Datenfeld: Address
DNS IP Adressen:	Liste von DNS-Adressen, welche verwendet werden sollen. WireGuard .conf Datenfeld: DNS



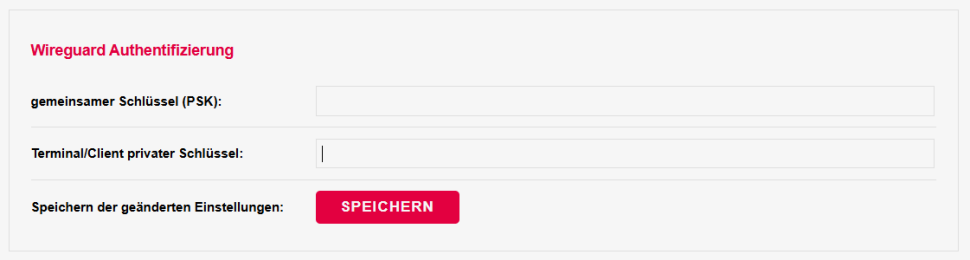
Durch Betätigen des Speicher Buttons werden die vorgenommenen Einstellungen gespeichert.

3.4 Konfiguration für Wireguard Authentifizierung

Unter Wireguard Authentifizierung können die folgenden Elemente konfiguriert werden:

gemeinsamer Schlüssel (PSK): [optional] Pre-shared key (PSK) des WireGuard Servers. WireGuard .conf Datenfeld: PresharedKey

Terminal/Client privater Schlüssel: Private Key des WireGuard Client im Terminal. WireGuard .conf Datenfeld: PrivateKey



Durch Betätigen des Speicher Buttons werden die vorgenommenen Einstellungen gespeichert.

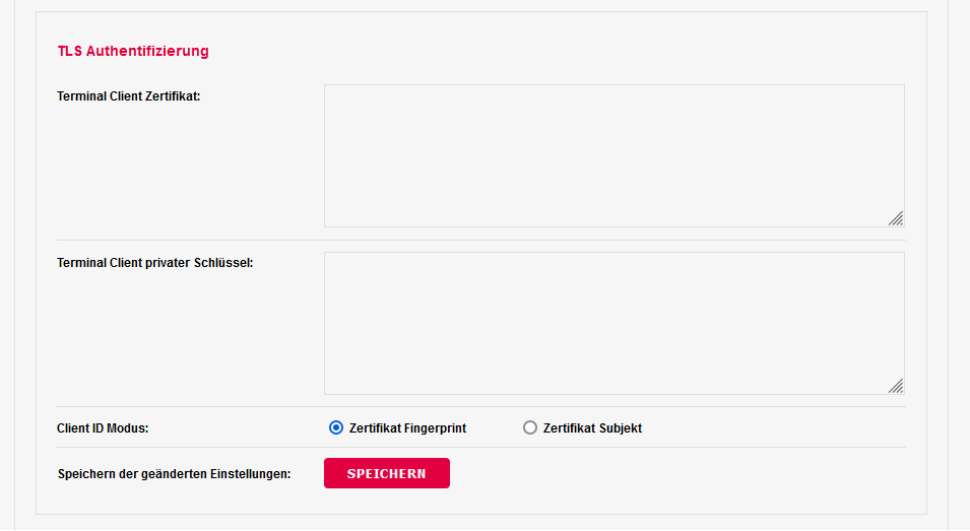
3.5 Konfiguration für IPSec TLS Authentifizierung

Unter TLS Authentifizierung können die folgenden Elemente dieses Modus konfiguriert werden:

Terminal Client Zertifikat: selbsterstelltes Client Zertifikat in base64 Konvertierung.

Terminal Client privater Schlüssel: Private Key des Client Zertifikates in base64 Konvertierung.

Client ID Modus: Auswahl der zu verwendenden Client ID. Bei TLS werden die Optionen Zertifikat Fingerprint oder Zertifikat Subjekt angeboten.



Durch Betätigen des Speicher Buttons werden die vorgenommenen Einstellungen gespeichert.

3.6 Konfiguration MSCHAPv2 Authentifizierung

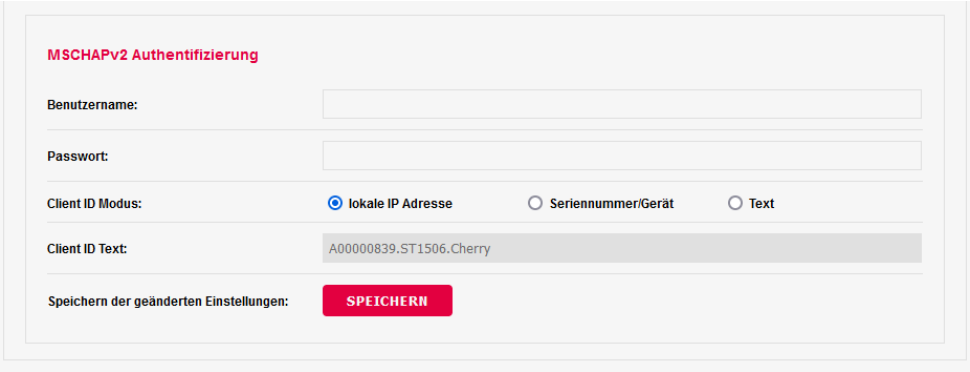
Unter MSCHAPv2 Authentifizierung können die folgenden Elemente dieses Modus konfiguriert werden:

Benutzername: Benutzername welcher für die Authentisierung verwendet werden soll.

Passwort: Passwort welches für die Authentisierung verwendet werden soll.

Client ID Modus: Auswahl der zu verwendenden Client ID. Bei MSCHAPv2 werden die Optionen lokale IP Adresse, Seriennummer/Gerät oder Text angeboten.

Client ID Text: Bei Auswahl des Client ID Modus „Text" kann hier eine selbst definierte Client ID vergeben werden. Als default Wert ist hier die ID hinterlegt, welche bei der Auswahl des Modus „Seriennummer/Gerät" verwendet wird.



Durch Betätigen des Speicher Buttons werden die vorgenommenen Einstellungen gespeichert.

4 Fehlermeldungen WireGuard Client

Meldung	Bedeutung
Fehler 1	DNS Dienst nicht verfügbar
Fehler 2	DNS Fehler - Server Name unbekannt
Fehler 3	interner Fehler
Fehler 4	Route nicht konfigurierbar
Fehler 5	Konflikt mit Terminal SICCT Port Nummer
Fehler 6	Konflikt mit PIN-Pad Port Nummer
Fehler 7	Keine Verbindung mit Server vorhanden
Fehler 8	Kein privater Schlüssel für Terminal gesetzt
Fehler 9	Keine zugelassenen IP Adressbereiche gesetzt
Fehler 10	Keine VPN IP Adresse für Terminal gesetzt
Fehler 11	Kein öffentlicher Schlüssel für Server gesetzt
Fehler 12	Keine Server Adresse gesetzt

5 Beispielkonfigurationen für IPSec nach strongSwan

5.1 VPN-Client ST-1506

Nachfolgend ist eine äquivalente swanctl.conf Konfiguration des ST-1506 dargestellt:
(Authentifizierung über Username/Passwort bei EAP-Mschap, bzw Client Zertifikat bei EAP-TLS)

```
connections {
  connect_with_xeap_tls {
    local_addrs  = 0.0.0.0
    remote_addrs = 10.2.0.64  # setable Server address
    dpd_delay=30 # setable
    dpd_timeout=150 # setable

    unique=replace
    send_cert=always

    vips=0.0.0.0

    local {
      auth = eap-tls
      certs = /usr/share/keys/ipsec/ipsec-peer.rsa4k.pem  # Client Zertifikat
      id = "CN=ST1506-IPSEC-RSA4K-PEER-DEBUG, ST=Vienna, C=AT"
    }
    remote {
      auth = eap-tls
    }
    children {
      xtunnel {
        local_ts = 0.0.0.0-169.253.255.255, 169.255.0.0-254.255.255.255  # restricted from FW v4.0.0,
0.0.0.0 v3.0.0
        remote_ts = 0.0.0.0/0
        start_action=start
        dpd_action=start
        close_action=trap
        esp_proposals = aes256gcm64-sha256-sha384-sha512, aes256-sha256-sha384-sha512
        rekey_time = 24h
      }
    }
    # ike2
    version = 2
    send_certreq = yes
    proposals = aes256gcm64-sha256-sha384-sha512-ecp384-ecp521-ecp384bp-ecp512bp-modp4096,
aes256-sha256-sha384-sha512-ecp384-ecp521-ecp384bp-ecp512bp-modp4096
    rekey_time = 8h
  }

  connect_with_xeap_mschapv2 {
```

```
    local_addrs  = 0.0.0.0
    remote_addrs = 10.2.0.64  # setable
    dpd_delay=30  # setable
    dpd_timeout=150  # setable

    unique=replace
    send_cert=never

    # ike2
    version = 2
    send_certreq = yes
    proposals = aes256gcm64-sha256-sha384-sha512-ecp384-ecp521-ecp384bp-ecp512bp-modp4096,
aes256-sha256-sha384-sha512-ecp384-ecp521-ecp384bp-ecp512bp-modp4096
    rekey_time = 8h

    vips=0.0.0.0

    local {
      auth = eap-mschapv2
      eap_id = theobroma  # setable
      id = A12345678.ST1506.Cherry  # setable from FW v4.0.0, otherwise empty !
    }
    remote {
     auth = pubkey
     revocation=ifuri
    }
    children {
      xtunnel_chap {
        local_ts = 0.0.0.0-169.253.255.255, 169.255.0.0-254.255.255.255 # restricted from FW v4.0.0,
0.0.0.0 otherwise
        remote_ts = 0.0.0.0/0
        start_action=start
        dpd_action=start
        close_action=trap
        esp_proposals = aes256gcm64-sha256-sha384-sha512,aes256-sha256-sha384-sha512
        rekey_time = 24h
      }
    }
  }
}
```

Zur Information wird im Folgenden die cipher selection dargestellt, die der Strongswan Server beim Verbindungsaufbau sieht:
#11[CFG] received proposals:
IKE:AES_GCM_8_256/PRF_HMAC_SHA2_256/PRF_HMAC_SHA2_384/PRF_HMAC_SHA2_512/ECP_384/ECP_521/ECP_384_BP/ECP_512_BP/MODP_4096,
IKE:AES_CBC_256/HMAC_SHA2_256_128/HMAC_SHA2_384_192/HMAC_SHA2_512_256/PRF_HMAC_SHA2_256/PRF_HMAC_SHA2_384/PRF_HMAC_SHA2_512/ECP_384/ECP_521/ECP_384_BP/ECP_512_BP/MODP_4096

10[CFG] received supported signature hash algorithms: sha256 sha384 sha512 identity

5.2 VPN-Gateway

Nachfolgend ist eine Ipsec.conf Konfiguration eines VPN Server (strongSwan 5.9.1) dargestellt:

```
# basic cipher setup
conn cipher-setup
  # enforce IKEv2
  keyexchange=ikev2
  keyingtries=%forever
  lifetime=12h
  ikelifetime=24h
  # configure dead peer detection (DPD)
  dpddelay=45
  dpdtimeout=80
  dpdaction=restart

# basic network setup
conn network-setup
  left=%defaultroute
  leftid=%any
  leftsourceip=172.16.0.0/24
  # allow connection from anyone
  right=%any
  # ASSIGN the peer an ip address within the given range
  # ! do not use Link local address range here !
  rightsourceip=172.18.0.0/24

conn authenticate-with-eap-tls
  # use basic cipher and network configuration
  also=cipher-setup
  also=network-setup
  auto=add
  # authenticate using eap-tls
  rightauth=eap-tls
  leftauth=eap-tls
  authby=pubkey
  leftcert=/etc/ipsec.d/certs/ipsec-site.secp384r1.pem
  leftid="CN=ST1506-IPSEC-SECP384R1-SITE-DEBUG, ST=Vienna, C=AT"
  rightid=%any
  #or rightid="CN=*,ST=Vienna,C=AT"


conn authenticate-with-eap-mschapv2
  # use basic cipher and network configuration
  also=cipher-setup
  also=network-setup
```

```
auto=add
## authenticate using eap-mschap
rightauth=eap-mschapv2
leftauth=pubkey
leftcert=/etc/ipsec.d/certs/ipsec-site-revoked-1.secp384r1.pem
leftid="CN=ST1506-IPSEC-SECP384R1-SITE-DEBUG, ST=Vienna, C=AT"
rightid=%any
#oder rightid="*.ST1506.Cherry"  # v4.0.0: client id modus: Seriennummer
eap_identity=%identity
```

6 Kontakt

Bitte halten Sie bei Anfragen an den Technischen Support folgende Informationen bereit:

- Artikel- und Serien-Nr. des CHERRY eHealth-Kartenterminals
- Firmware-Version des CHERRY eHealth-Kartenterminals
- Name und Version verwendeter Software
- Bezeichnung und Hersteller Ihres Systems (Konnektor, Verwaltungssoftware)
- Betriebssystem und ggf. installierte Version eines Service Packs

Cherry Digital Health GmbH
Rosental 7, c/o Mindspace
81677 München

Internet: www.cherry.de

Telefon: +49 (0) 9643 2061-100*

*zum Ortstarif aus dem deutschen Festnetz, abweichende Preise für Anrufe aus Mobilfunknetzen möglich