optiPoint 410/420 Familie

Gesicherte Sprachübertragung in SIP 7

Grundlagen

SIEMENS

Global network of innovation

Inhalt

1 VoIP mit der optiPoint 410/420 Telefonfamilie

Der Hauptvorteil von Voice over IP ist in erster Linie die Kostenersparnis. Mehrere Faktoren kommen dabei zum tragen:

- Die niedrigen Verbindungskosten, bei reiner IP-Netzverbindung
- Ein Netz für Daten und Telefonie, daher niedrigere Wartungskosten
- Firmen mit mehreren Standorten führen interne Gespräche über ihr Intranet
- Endgeräte können innerhalb des LANs den Standort wechseln und behalten dabei die Rufnummer

1.1 Prinzip VoIP

Voice over IP verwendet unterschiedliche Protokolle. Die Signalisierung erfolgt über das Session Initiation Protokoll (SIP – RFC 3261). Andere Protokolle kontrollieren Vorgänge wie:

- Die Vermittlung von Gesprächen
- Den Rufauf- und abbau
- Die Aushandlung von Parametern wie z. B. Codec oder Bandbreite

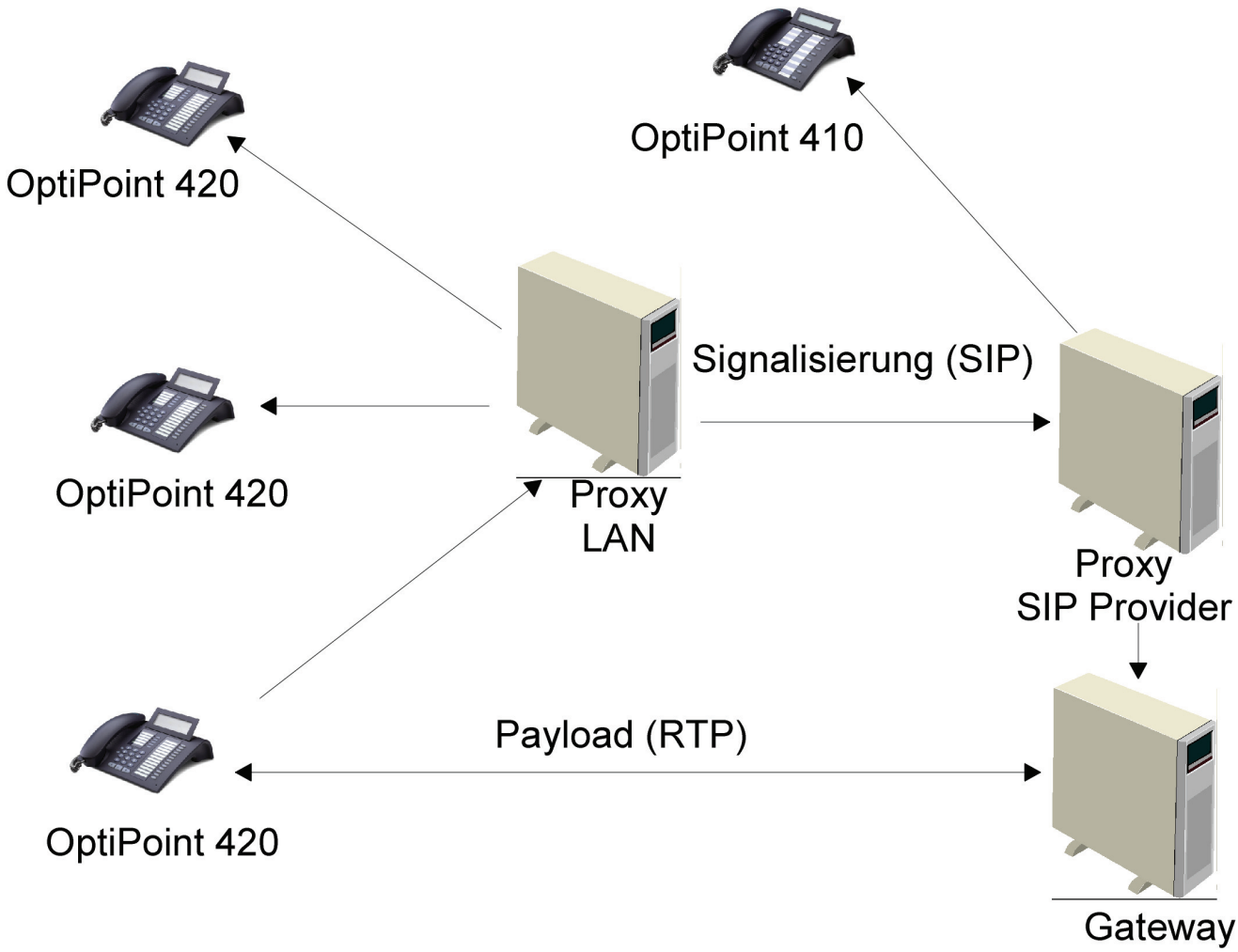
Um die Eigenschaften des Mediendatenstroms zu bestimmen wird zusätzlich das **S**ession **D**escription **P**rotocol (SDP) verwendet.

Für die Sprachübertragung ist das Realtime Transport Protocol (RTP) zuständig. Mit diesem Protokoll ist es der Empfängerstation möglich, Pakete trotz unzuverlässiger Übertragung über UDP einen Paketverlust festzustellen und die ursprüngliche Reihenfolge der Daten wiederherzustellen.

Im Prinzip kann VoIP bereits innerhalb einer eingerichteten IP-Infrastruktur mit IP-Telefonen funktionieren. Damit die einzelnen Teilnehmer über Direct-IP-Calls Verbindung aufnehmen können, müssen sie die IP-Adresse des gewünschten Gesprächspartners kennen. Solbald eine Adressänderung (z. B. bei DHCP) erfolgt, funktioniert dieses Verfahren nicht mehr problemlos, weil die Teilnehmer laufend ihre Adressbucheinträge ändern müssten. Das Einbinden weiterer Teilnehmer ist ebenfalls mit diesem Aufwand verbunden.

In der IP-Telefonie mit dem SIP-Protokoll übernehmen Proxys die Aufgabe der Vermittlungsstelle. Ein Proxy kann ein SIP-Server-Programm enthalten, das ankommende SIP-Pakete an das Telefon des angerufenen Teilnehmers oder ggf. an einen anderen Proxy weiterleitet.

Funktionen wie Konferenzen, Voice-Mail oder Anrufweiterleitung werden ebenfalls zentral über den Proxy verwaltet. Da der Rufaufbau ebenfalls über den Proxy erfolgt, können dort Daten zur Zeiterfassung und Gebührenberechnung gespeichert werden.



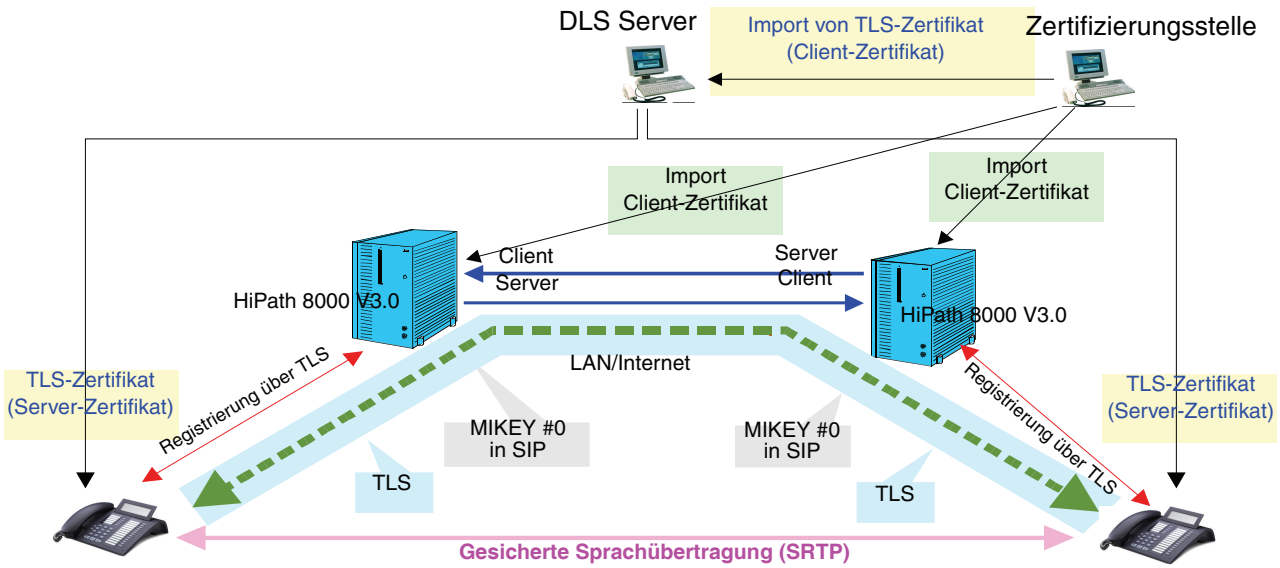
In einem Netz für VoIP werden in der Regel mehr als ein Proxy verwendet. Zum Beispiel ist ein Proxy für die interne Telefonie und ein Proxy für die Telefonate ins Festnetz zuständig. Die Verbindung zwischen VoIP und der klassischen Telefonie wird über ein Gateway hergestellt.

1.2 Unterschied in der Sicherheit

PSTN: Wollte man Telefongespräche über das herkömmliche Öffentliche Netz abhören oder stören, ist ein physischer Zugriff auf die Leitung oder Vermittlungsstellen notwendig. Ein spezielles Wissen als Voraussetzung, um die Signalisierung zu entschlüsseln, ist ein zusätzliches Hindernis.

VOIP: Für die Sprachübertragung im IP-Netz (Intranet und Internet) werden in der Regel standardisierte und offene Protokolle verwendet. Einem potentiellen Angreifer stehen dafür etliche, leicht zu beschaffende Programme zur Verfügung, mit denen es ihm möglich ist, den Datenverkehr mitzuschneiden und die verwendeten Protokolle zu entschlüsseln.

1.3 Lösungsweg für gesicherte Signalisierung und Sprachübertragung



(1) Das Einrichten der TLS-Zertifikate erfolgt bei der Erstinstallation der Telefone (DLS Plug & Play)

(2) Die HiPath 8000 erhält ein Client- und ein Server-Zertifikat von der Zertifizierungsstelle.

(3) Die Telefone registrieren sich über TLS an der Hipath 8000 und überprüfen anhand des Server CA Zertifikats die Authentizität der HiPath 8000

(4) Weitergabe eines Schlüssels innerhalb der verschlüsselten Signalisierungsmeldung

(5) Auf Grundlage des ausgetauschten Schlüssels wird die Sprache verschlüsselt

(6) Die HiPath 8000 Server überprüfen gegenseitig die Authentizität

1.3.1 Signalisierung

1.3.2 Gesicherte und ungesicherte Signalisierung im → SDP

Wenn bei eingeschalteter Verschlüsselung eine INVITE-Nachricht für den Rufaufbau geschickt wird, so wird sowohl eine → RTP- als auch eine → SRTP-Session geöffnet, da nicht von vornherein klar ist, ob eine Gegenstation eine gesicherte Verbindung aufbauen kann. Für dieses Angebot müssen zwei Port-Nummern zur Verfügung stehen. Ein Port wird im Fall der gesicherten, der andere im Fall der ungesicherten Übertragung verwendet. Sobald entschieden ist, welche Übertragung verwendet wird, wird einer der beiden Ports geschlossen.

1.3.2.1 Beispiel für eine INVITE-Anweisung

```
: MON MAR 26 08:20:15 2007
Packet Out--->10.23.22.102
INVITE sip:15192@10.23.22.102;transport=tls SIP/2.0
Via: SIP/2.0/TLS 10.23.15.16:45678;branch=z9hG4bKe3a34755c27ac111f
Route: <sip:15192@10.23.22.102;lr;transport=tls>
Max-Forwards: 70
From: 498972115182
<sip:498972115182@10.23.22.102>tag=c518793c05;epid=SC268060
To: <sip:15192@10.23.22.102;transport=tls>
Call-ID: 83a48296d0489f71
CSeq: 1 INVITE
Allow: INVITE,ACK,CANCEL,BYE,REFER,NOTIFY,UPDATE
Allow-Events: talk, hold
Contact: 498972115182 <sip:498972115182@10.23.15.16:45678;transport=tls>
Min-SE: 90
Session-Expires: 3600
Supported: timer, replaces, 100rel
User-Agent: optiPoint 410_420/V7 V7 R0.1.0 M5T SIP Stack/4.0.24.24
X-Siemens-Call-Type: ST-secure
Content-Type: application/sdp
Content-Length: 803
v=0
o=MxSIP 0 1669068321 IN IP4 10.23.15.16
s=SIP Call
c=IN IP4 10.23.15.16
t=0 0
m=audio 5004 RTP/AVP 0 8 9 18 4 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:9 G722/8000
a=rtpmap:18 G729/8000
a=rtpmap:4 G723/8000
a=rtpmap:101 telephone-event/8000
a=silenceSupp:off - - - -
a=fmtp:4 annexa=no
a=fmtp:18 annexb=no
a=fmtp:101 0-15
a=ptime:20
m=audio 5006 RTP/SAVP 0 8 9 18 4 101
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:9 G722/8000
a=rtpmap:18 G729/8000
```

```
a=rtpmap:4 G723/8000
a=rtpmap:101 telephone-event/8000
a=silenceSupp:off - - - -
a=fmtp:4 annexa=no
a=fmtp:18 annexb=no
a=fmtp:101 0-15
a=key-mgmt:mikey
AQAVgPw6XtcCAAAAAAAAAAAAABQgfunAAAAAAUBAAVtaWtleQsA08MZMhHrhAAKFIWOc5QnOgp
H21Bz0KeGkfcqKNnOAQAAAAAAAAVAAAAEM2ttTMuxkbS9RrKOcYOpdsAAA==
a=ptime:20
```

1.3.3 Gesicherte Signalisierung

Der bereits vorher verfügbaren TLS-Unterstützung wurden zwei wesentliche Erweiterungen hinzugefügt, die die gesicherte SIP-Signalsierung noch erhöht.

1.3.3.1 Zertifikatsprüfung für TLS

Wenn eine TLS-Verbindung eingerichtet wurde, wird das übertragene Server-Zertifikat überprüft – sofern das Telefon für diese Prüfung konfiguriert wurde → Seite 11.

Verhalten im Fall einer fehlerhaften Zertifikatsprüfung:

- es ist keine Registrierung möglich
- die Wiederherstellungs-Strategie ist ähnlich anderer Fehler-Behandlungen
- ein Protokoll kennzeichnet das Problem als Fehler.
- es wird eine Fehlermeldung ausgegeben

1.3.3.2 Überwachung der TLS-Verbindung

Das Telefon überwacht die gesicherte TLS-Verbindung zum SIP-Server, indem es die Antwort des Servers auf die Meldungen, die periodisch vom Telefon an den Server gesandt werden, untersucht.

1.3.3.3 TLS-Zertifikate

Zertifizierungsstellen-Zertifikat (CA-Zertifikat) mit dem auch die 8k Zertifikate ausgestellt wurden → Seite 8 und → Seite 10 werden über DLS an die Telefone gesandt.

1.3.4 Gesicherte Sprachübertragung

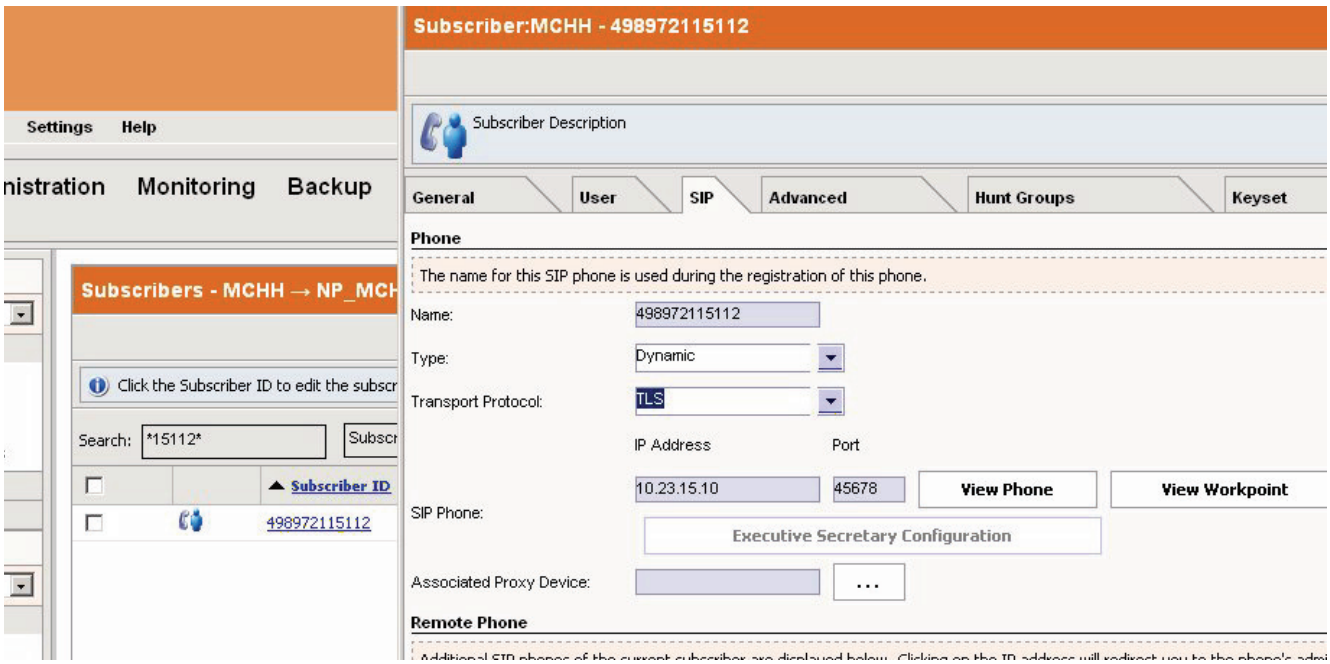
Für die nicht gesicherten Anrufe wird → RTP als Transportprotokoll verwendet. Für verschlüsselte Anrufe wird stattdessen → SRTP (mit dem dazugehörenden SRTCP Kontrollprotokoll) verwendet. Das gewährleistet eine fundierte Sicherheit, die sich aus Verschlüsselung und Integritäts-Schutz zusammensetzt, die Sprachübertragung im IP-Netzwerk schützt, wenn Empfänger- und Sendegerät im SIP-Mode arbeiten.

Als Basis-Leistungsmerkmal wird eine Schlüsselverwaltung nach Aufwand (pro Anruf) zur Verfügung gestellt, wobei → MIKEY mit einem NULL-Verschlüsselungsalgorithmus zur Übertragung des Schlüssels verwendet wird (informelle Bezeichnung: „MIKEY Option 0“). MIKEY wird im Telefon erzeugt und über die Signalisierung an das Zieltefon gesandt.

1.4 Konfiguration

1.4.1 HiPath 8000

In der HiPath 8000 muss bei „Connect info“ der Transport-Typ auf TLS pro Benutzer (BGL) eingestellt sein. Die Einstellung erfolgt über den Assistant (Common Management Portal):



1.5 Zertifikate

1.5.1 Erstellen und installieren der TLS-Zertifikate für die SIP-Signalisierung

Wenn TLS als Transport-Protokoll für die SIP-Signalisierung zwischen dem HiPath 8000 Server und einem SIP-Telefon (oder SIP-Endgerät) verwendet wird, wird ein Authentifizierungs-Handshake während des TLS-Verbindungsaufbaus ausgeführt.. Dieser Handshake erfordert ein Zertifikat mit einem öffentlichen Schlüssel und einen entsprechenden privaten (geheimen) Schlüssel auf dem HiPath 8000-Server. Das SIP-Telefon (SIP-Endgerät) benötigt ein entsprechendes Client-Zertifikat einer Stammzertifizierungsstelle (root CA), das eingerichtet wurde, um das Server-Zertifikat zu überprüfen, das es vom HiPath 8000 Server während des TLS-Handshakes erhält..

Die nachfolgenden Schritte beschreiben detailiert, wie man ein TLS-Zertifikats-Paar erstellt – zum Beispiel ein X.509 Server-Zertifikat und das entsprechende Client-Zertifikat einer Stammzertifizierungsstelle (root CA) – und wie man das TLS-Server-Zertifikat auf dem HiPath 8000 Server installiert..

DLS wird verwendet, um das entspechenden „root CA“ Client-Zertifikat auf Siemens SIP-Telefonen zu installieren, die TLS als Tansportprotokoll verwenden. Das gleiche „root CA“ Client-Zertifikat wird von allen SIP-Telefonen verwendet, um z. B. ein X.509 Server-Zertifikat zu prüfen, das auf dem HiPath 8000 Server installiert ist.

Über die OpenSSL Zertifikatsschnittstelle (Zertifizierungsdienst) auf dem HiPath 8000 Server wird ein Zertifikats-Paar erstellt.. Eine ausführliche Beschreibung der „OpenSSL“-Befehle, die in dieser Anwendung verwendet werden, finden Sie unter http://www.openssl.org/docs/apps/openssl.html)

1.5.1.1 Schritt 1: Erstellen der Stammzertifizierungsstelle

1. Erstellen Sie einen neuen Schlüssels (gespeichert in **`rootkey.pem`**) und eine Zertifikatsanforderung (gespeichert in **`rootreq.pem`**):

   ```
   openssl req -newkey rsa:1024 -sha1 -keyout rootkey.pem -out root-
   req.pem -config root.cnf -nodes
   ```

2. Signieren und speichern Sie die Zertifikatsanforderung in der Datei **`rootcert.pem`**:

   ```
   openssl x509 -req -in rootreq.pem -sha1 -extfile root.cnf -extensions
   certificate_extensions -signkey rootkey.pem -out rootcert.pem -days
   3650
   ```

3. Speichern Sie das Root-Zertifikat und die Root-Schlüssel zusammen in eine Datei:

   ```
   cat rootcert.pem rootkey.pem > root.pem
   ```

4. Prüfen Sie das neue Zertifikat, dass Sie gerade erstellt haben:

   ```
   openssl x509  -subject -issuer -noout -in root.pem
   ```

1.5.1.2 Schritt 2: Server-Zertifizierungsstelle erstellen und mit der Stammzertifizierungsstelle signieren

1. Erstellen Sie die Dateien **`serverCAkey.pem`** und **`serverCAreq.pem`**:

   ```
   openssl req -newkey rsa:1024 -sha1 -keyout serverCAkey.pem -out server-
   CAreq.pem -config serverCA.cnf -nodes
   ```

2. Generieren Sie die Dateien **`serverCAcert.pem`** und **`root.srl`**:

   ```
   openssl x509 -req -in serverCAreq.pem -sha1 -extfile serverCA.cnf -ex-
   tensions certificate_extensions -CA root.pem -CAkey root.pem -CAcreate-
   serial -out serverCAcert.pem -days 3650
   ```

3. Verbinden Sie **`serverCAcert.pem`**, **`serverCAkey.pem`**, und **`rootcert.pem`** zur Datei **`serverCA.pem`**:

```
cat serverCAcert.pem serverCAkey.pem rootcert.pem > serverCA.pem
```

4. Überprüfen Sie das gerade erstellte Zertifikat:

```
openssl x509  -subject -issuer -noout -in serverCA.pem
```

1.5.1.3 Schritt 3: Server-Zertifikat erstellen und mit der Server-Zertifizierungsstelle signieren

1. Erstellen Sie die Dateien **`serverreq.pem`** und **`serverkey.pem`**:

```
openssl req -newkey rsa:1024 -sha1 -keyout serverkey.pem -out server-
req.pem -config server.cnf -reqexts req_extensions -nodes
```

2. Erstellen Sie die Dateien **`servercert.pem`** und **`serverCA.slr`**:

```
openssl x509 -req -in serverreq.pem -sha1 -extfile server.cnf -extensi-
ons certificate_extensions -CA serverCA.pem -CAkey serverCA.pem -CAcre-
ateserial -out servercert.pem -days 3650
```

3. Verbinden Sie die Dateien **`servercert.pem`**, **`serverkey.pem`**, **`servercert.pem`** and **`rootcert.pem`** zur Datei **`server.pem`**:

```
cat servercert.pem serverkey.pem serverCAcert.pem rootcert.pem > ser-
ver.pem
```

4. Überprüfen Sie das gerade erstellte Zertifikat:

```
openssl x509 -subject -issuer -noout -in server.pem
```

1.5.1.4 Schritt 4: Client-Zertifikat erstellen und mit der Stamm-Zerfifizierungsstelle signieren

1. Generieren Sie die Dateien **`clientreq.pem`** und **`clientkey.pem`**:

```
openssl req -newkey rsa:1024 -sha1 -keyout clientkey.pem -out clien-
treq.pem -config client.cnf -reqexts req_extensions -nodes
```

2. Generieren Sie die Dateien **`root.srl`** und **`clientcert.pem`**:

```
openssl x509 -req -in clientreq.pem -sha1 -extfile client.cnf -extensi-
ons certificate_extensions -CA root.pem -CAkey root.pem -CAcreateseri-
al -out clientcert.pem -days 3650
```

3. Verbinden Sie die Dateien **`clientcert.pem`**, **`clientkey.pem`**, und **`rootcert.pem`** zur Datei **`client.pem`**:

```
cat clientcert.pem clientkey.pem rootcert.pem > client.pem
```

4. Überprüfen Sie das gerade erstellte Zertifikat:

```
openssl x509  -subject -issuer -noout -in client.pem
```

5. Erstellen Sie die Datei **dh1024.pem**:

```
openssl dhparam -check -text -5 1024 -out dh1024.pem
```

1.5.1.5 Schritt 5: Schreiben der Dateien in die entsprechenden Verzeichnisse auf dem Hipath 8000 Server

1. Die Dateien **client.pem** und **server.pem** werden im Verzeichnis **/usr/local/ssl/private** gespeichert:

```
cp client.pem server.pem /usr/local/ssl/private
```

2. Die Dateien **root.pem** und **dh1024.pem** werden im Verzeichnis **/usr/local/ssl/certs** gespeichert:

```
cp root.pem dh1024.pem /usr/local/ssl/certs
```

3. After the files are stored, bring each node to state 3 and back to state 4 to restart the TTUD. Nachdem die Dateien gespeichert sind, muss jeder Knoten von Status 3 auf Status 4 zurückgesetzt werden, um TTUD neu zu starten.

1.5.1.6 Schritt 6: Installation auf SIP-Telefone, die TLS verwenden.

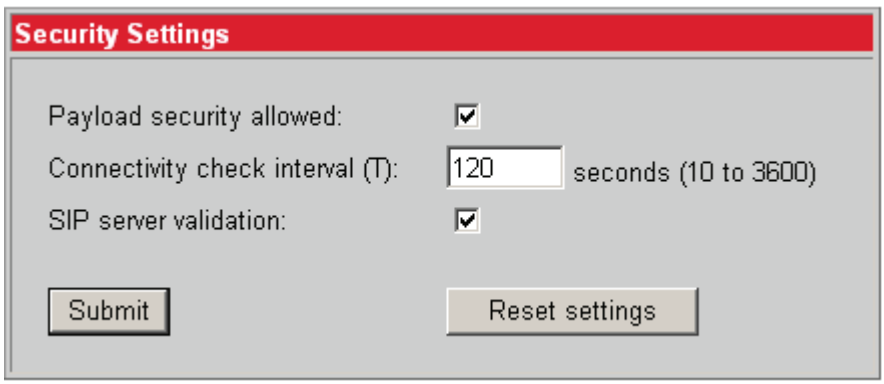
Kopieren Sie das Zertifikat **root.pem** per DLS auf die Telefone.

Das Löschen der Zertifikate erfolgt entweder durch zurücksetzen des Telefones auf Werkseinstellungen oder aber per DLS. Im DLS muss dazu das Zertifikat entfernt, dann das leere Zertifikat aktiviert und danach gesichert und damit zum Telefon gesendet werden.

1.5.2 optiPoint 410/420 S SIP 7

1.5.2.1 Security Settings

Rufen Sie im Web-Interface-Menü des entsprechenden Telefons den Dialog „Security“ auf. Sie erhalten folgendes Fenster:



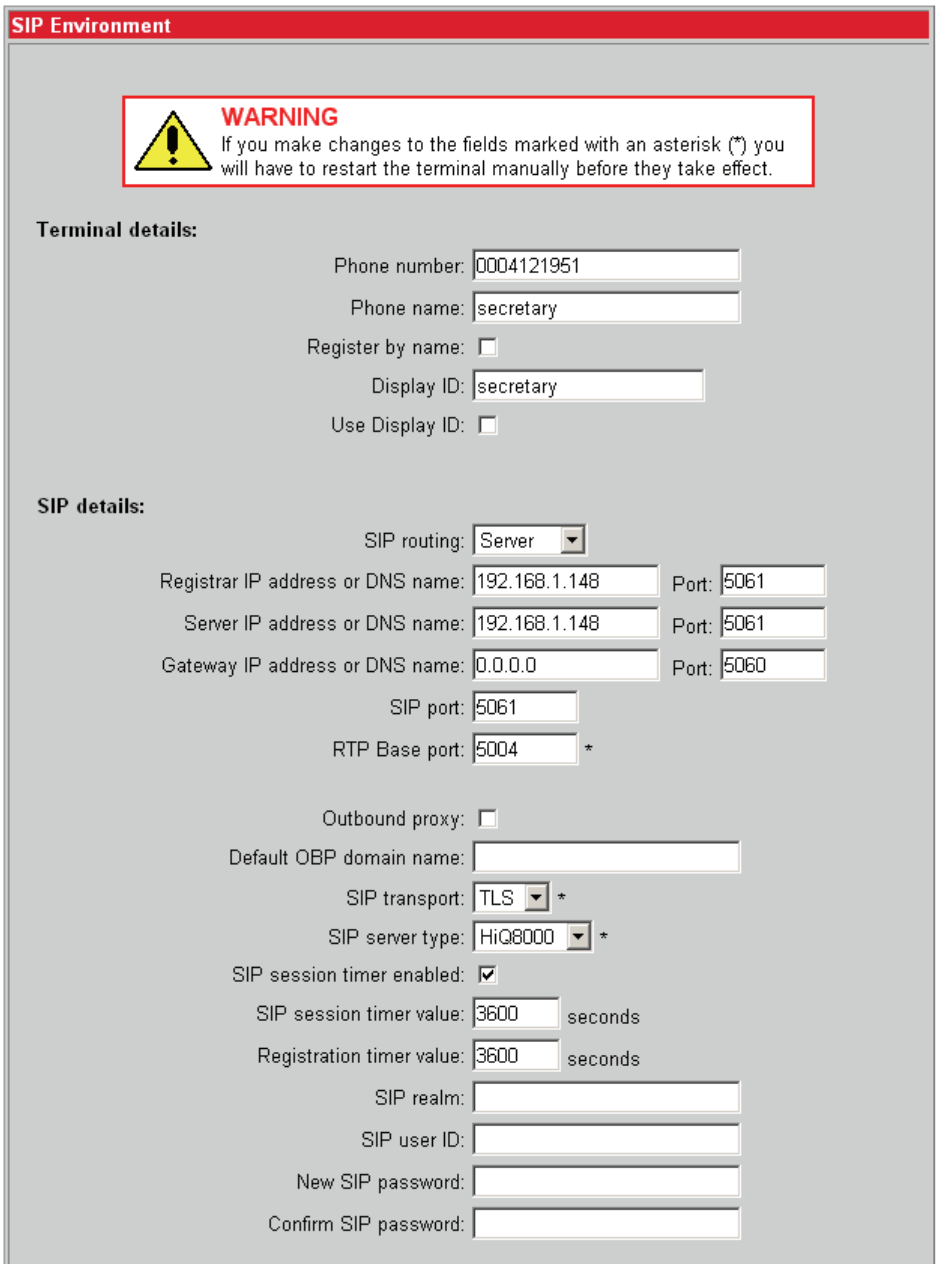
Markieren Sie „Payload security allowed“ als eingeschaltet, damit Sie immer eine gesicherte Verbindung aufbauen, wenn es möglich ist.

Um die TLS-Verbindung zu kontrollieren, muss das Feld „Connectivity check interval (T)“ einen Wert Größer 0 enthalten. Die Voreinstellung ist 120. Wenn Sie die Kontrollfunktion ausschalten wollen, setzen Sie den Wert auf 0. Das Wiedereinschalten erfolgt mit der Eingabe eines Werts größer 0. Nach Aus- bzw. Einschalten wird ein Neustart erforderlich. Wenn Sie nur einen Wert, der größer 0 ist, ändern muss das Telefon nicht neu gestartet werden.

Haben Sie „SIP server validation“ eingeschaltet, dann wird bei der Registrierung von der Hipath 8000 ein Server-Zertifikat verlangt und auf Gültigkeit überprüft → Seite 5.

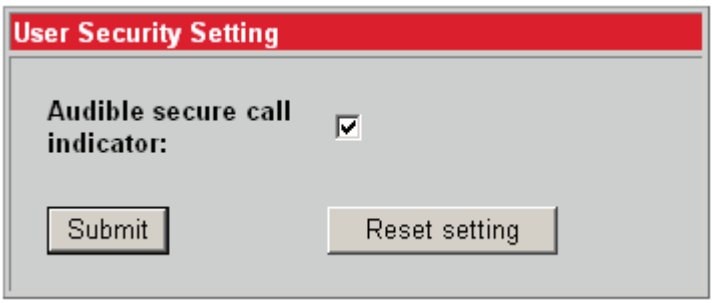
1.5.2.2 SIP Environment

Rufen Sie im Web-Interface-Menü des entsprechenden Telefons den Dialog „Sip Environment“ auf. Sie erhalten folgendes Fenster:



Die Port-Adresse bei Registrar und Server muss 5061 lauten. SIP transport muss auf TLS eingestellt werden

1.5.2.3 User Security Setting



Ist die Option markiert, so wird ein Aufmerksamkeitston ausgegeben, wenn **keine** gesicherte Verbindung aufgebaut wurde.

1.6 Telefonieren mit/ohne gesicheter Verbindung

Beim Versuch, eine gesicherte Verbindung aufzubauen können 4 Zustände vorkommen:

	Telefon 1	Telefon 2	Protokoll
1	Security ein	Security ein	SRTP
2	Security ein	Security aus oder nicht möglich	RTP
3	Security aus oder nicht möglich[1]	Security eingeschaltet	RTP
4	Security aus oder nicht möglich	Security ausg oder nicht möglich	RTP

1 Das Telefon hat noch keinen SIP 7 Status

1.6.1 Verbindung mit einem Teilnehmer

Wenn Sie einen Teilnehmer über eine gesicherte Verbindung anrufen, erhalten Sie im Telefondisplay ein blinkendes „Schloss-Symbol“. Ist die akustische Signalisierung für gesicherte Verbindung eingeschaltet, so hören Sie einen kurzen Signalton, wenn die Verbindung nicht gesichert ist.

1.6.2 Konferenz

Befinden Sie sich in einer gesicherten Verbindung zu einem Gesprächspartner und leiten eine Rückfrage ein, wird der aktuelle Partner ins Halten gelegt. Die neue Verbindung aus der Rückfrage kann sowohl gesichert als auch ungesichert sein. Werden jetzt alle drei Teilnehmer zu einer Konferenz zusammengeschaltet, so bleibt die erste Verbindung gesichert. Sobald die Verbindung zu einem Partner nicht sicher ist, so gilt die ganze Konferenz als nicht sicher.

2 Protokolle

2.1 Session Initiation Protocol (SIP)

Das Session Initiation Protocol wird in der IP-Telefonie ausschließlich als Signalisierungsprotokoll verwendet.

SIP ist ein textbasiertes und damit auch leicht lesbares Protokoll . Jedes SIP-Paket beginnt in der ersten Zeile mit einer Anfrage oder einem Antwortcode und ist folgendermaßen aufgebaut:

```
INVITE sip:kremer@siemens.net SIP/2.0
```

An erster Stelle steht immer die Anfragemethode, die von einer URL gefolgt wird. In der Regel werden URLs des SIP-Schemas verwendet (ähnlich wie eine eMail-Adresse). Abgeschlossen wird die Zeile mit der Protokollkennung „SIP"die mit einem „/“ von der Versionsnummer, aktuell 2.0, getrennt ist.

Folgende SIP-Methoden sind z. B. in RFC 3261 spezifiziert:

Methode	Beschreibung
REGISTER	Anmelden eines Benutzers beim SIP-Server
INVITE	Signalisierung eines Anrufes
ACK	Bestätigung des Anrufenden zu einer Invite-Anfrage
CANCEL	Abbruch einer Invite-Anfrage
BYE	Beenden eines Anrufes
OPTIONS	Abfrage der Eigenschaften und Fähigkeiten.

Das Protokoll kann durch Einfügen neuer Anfragen beliebig erweitert werden.

Ein Antwortpaket auf eine SIP-Anfrage beginnt wie folgt:

SIP/2.0 200 OK

Am Zeilenanfang stehen die Protokollkennung und die Versionsnummer, der eine dreistellige numerische Antwortkennung mit einem beschreibenden Text folgt. Die Antwortcodes sind in mehrere Kategorien eingeteilt, wie in folgender Tabelle dargestellt.

Code	Beschreibung	Beispiele
1xx	Provisorische Antworten	100 Trying, 180 Ringing
2xx	Anfrage erfolgreich ausgeführt	200 OK
3xx	Umleitung	301 Moved Permanently, 302 Moved Temporarily
4xx	Fehler bei der Durchführung der Anfrage	401 Unauthorized, 404 Not Found
5xx	Serverseitiger Fehler	500 Internal Error, 501 Not Implemented
6xx	Globaler Fehler	600 Busy Everywhere, 606 Not Acceptable

Nach der ersten Zeile folgt eine beliebige Anzahl von Headerzeilen. Wichtige Header sind:

From Dieses Feld enthält einen Namen und eine URL, die den Absender eindeutig identifizieren soll. Beispiel für einen From-Header:

```
From: "Wolfgang Kremer" <sip:1122339@freenet.de>;tag=2CA772EF
```

To Der To-Header identifiziert den logischen Empfänger der Anfrage.

Contact Contact gibt eine SIP- bzw. SIPS-URI an, unter der die Gegenstelle für weitere Anfragen erreicht werden kann.

```
Contact: "Wolfgang Kremer" <sip:1122339@@92.128.41.222;trans-
port=udp>;q=1.0;expires=3600
```

Call-Id Identifiziert mit den From- und To-Tags eindeutig einen Dialog. Ein Beispiel eines Call-Id-Header:

```
Call-ID: 1234567890@92.128.41.222
```

CSeq Der CSeq-Header besteht aus einer Zahl und der Methode aus der Anfragezeile. Er dient dazu, Transaktionen innerhalb eines Dialogs zu ordnen. Ein Beispiel eines CSeq-Headers ist:

```
CSeq: 6061 BYE
```

Authorisierungsheader SIP verwendet den im RFC 2617 standardisierten Digest-Authentifizierungsmechanismus, der auch im HTTP-Protokoll verwendet wird.

2.1.0.1 Dialog

Dialog bedeutet in SIP einen Austausch von Nachrichten zwischen zwei SIP-Endpunkten. Ein Dialog wird durch die Call-ID, From- und To-Tags auf jeder Seite eindeutig identifiziert. Ein Dialog wird durch eine positive Antwort auf eine Invite-Anfrage erstellt und durch eine BYE-Anfrage wieder beendet.

2.1.0.2 Transaktion

Eine Transaktion wird durch eine Anfrage und ihre Antworten definiert, in der die Rollen des Clients und des Servers klar festgelegt sind.

- **Client** stellt eine Anfrage
- **Server** liefert die Antwort auf die Anfrage

Der Host, der eine Anfrage an einen anderen Host sendet, wird für die Dauer der Transaktion **User Agent Client** oder kurz **UAC** genannt.

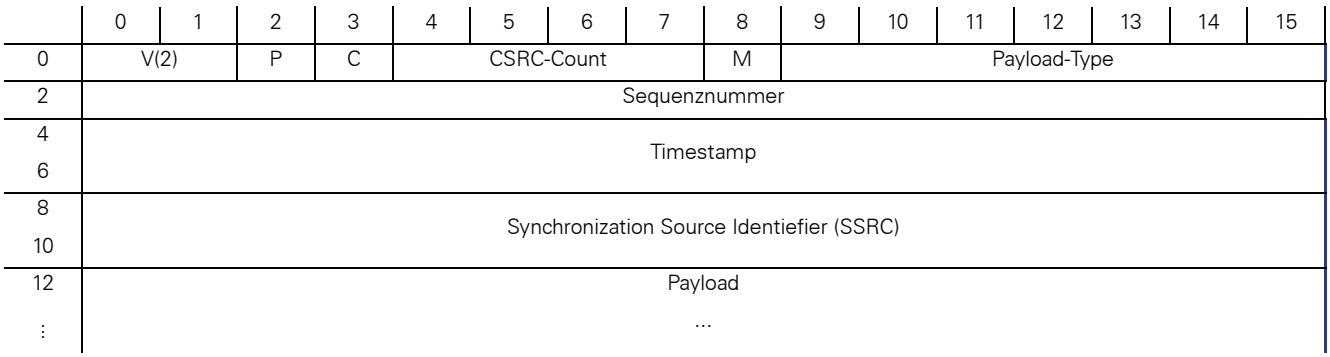
Der Host, der die Anfrage entgegennimmt und sie beantwortet, heißt **User Agent Server** oder kurz **UAS**.

2.2 SDP

Das **S**ession **D**escription **P**rotocol wird von dem Signalisierungsprotokoll SIP dazu verwendet, die Eigenschaften des Mediendatenstroms wie den verwendeten Codec und Bitrate oder auch ein Transportprotokoll und die Zieladresse mit den verwendeten Ports auszuhandeln und festzulegen. SDP-Bodies sind bei VoIP in SIP-Paketen eingebettet.

2.3 RTP

Das in RFC 3550 standardisierte **R**eal-Time **T**ransport **P**rotocol ist bei SIP das Standardprotokoll, um die Sprachdaten in Echtzeit zu übertragen. Bei der Übertragung von Sprache kann auf Zuverlässigkeit verzichtet werden, deshalb wird auch als Transportprotokoll UDP verwendet. Hintergrund ist, dass verlorene Pakete bei einer nicht zu hohen Paketverlustrate zu weniger Qualitätseinbußen führen als Verzögerungen oder Jitter, die durch wiederholte Übertragungsversuche entstehen würde.



2.4 SRTP

Secure RTP ist ein in RFC 3711 standardisiertes RTP-Profil, mit der Zielsetzung, die Vertraulichkeit der übertragenen Daten herzustellen und deren Integrität zu sichern, ohne den Overhead übermäßig zu erhöhen.

Ein SRTP-Paket ist wie in nachfolgender Tabelle aufgebaut und enthält die gleichen Header wie ein normales RTP-Paket. Der Unterschied besteht darin, dass der Payload verschlüsselt und dieser mitsamt des Headers authentifiziert werden kann. Zusätzlich kann sich am Ende des Pakets noch ein Master Key Identifier, welcher einen Master-Key und eine Signatur zur Authentifizierung befinden.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	V(2)		P	C	CSRC-Count				M	Payload-Type						
2	Sequenznummer															
4 6	Timestamp															
8 10	Synchronization Source Identiefier (SSRC)															
12 ⋮	Payload ...															
	SRTP Master Key Identifier (MKI) (Optional und beliebig lang)															
	Authentifizierungs-Tag (Optional und beliebig lang)															

2.5 TLS

Das in RFC 2246 spezifizierte TLS-Protokoll (Transport Layer Security) bietet einen Kommunikationskanal, in dem die Daten verschlüsselt übertragen werden und die Integrität der Daten absichert sind.

Im ersten Schritt sendet der Client ein **Client Hello-Paket**, das die unterstützten Kombinationen von Verschlüsselungsalgoritmen mit Modi, Schlüssellänge und Algorithmen enthält. Der Server antwortet auf dieses Paket mit einem **Server Hello**, in dem eine der angebotenen Algorithmenkombinationen ausgewählt und optional das Zertifikat des Servers übertragen werden kann, das vom Client überprüft wird. Der serverseitige Teil des Schlüsseltauschs kann implizit durch die Übertragung des Zertifikats oder explizit durch eine spezielle Nachricht erfolgen. Je nach verwendeter Schlüsseltauschmethode wird der öffentliche RSA-Schlüssel oder die öffentlichen **Diffie-Hellman-Parameter** gesendet. Zusätzlich kann der Server auch eine Authentifizierung des Clients mit einem Zertifikat anfordern. Nach dem optionalen Senden eines Client-Zertifikats und der Bestätigung dessen Authentizität durch den Server erfolgt der clientseitige Schlüsseltausch, der folgendermaßen erfolgt:

- Senden eines zufällig generierten Premaster Keys, der mit dem öffentlichen Schlüssel des Servers verschlüsselt ist.

- Senden des öffentlichen Diffie-Hellman-Parameters des Clients. Dieser Schritt kann auch implizit mit dem Senden des Client-Zertifikats erfolgen.

Da TLS im Allgemeinen als sicheres Protokoll akzeptiert ist und bereits in vielen alltäglichen Situationen, wie z.B. dem HTTP-Datenverkehr beim Online-Banking, verwendet wird, kann davon ausgegangen werden, dass ein über TLS geführter Dialog im Sinne der Vertraulichkeit und Integrität abgesichert ist.

2.6 MIKEY

Um den Austausch von Schlüsseln und den dazugehörenden Sicherheitsprametern zu unterstützen muss das MIKEY-Protokoll der Signalisierung beim Rufaufbau hinzugefügt werden. MIKEY RFC 3830 wird in SDP-Meldungen eingefügt, die wiederum in SIP-Meldungen eingebettet sind. Zur Zeit wird nur die MIKEY Option 0 unterstützt (Schlüssel in Klartext).

Sobald MIKEY (Option 0) verwendet wird – der Schlüssel selbst wird unverschlüsselt übertragen – ist ein Schlüsseltausch nur erlaubt, wenn die SIP-Signalisierung über die sichere TLS-Verbindung erfolgt. Es findet keine Schlüsselaustausch vor, wenn:

- SIP über TCP oder über UDP erfolgt,
- Die Einstellung „Gesicherte Sprachübertragung“ ausgeschaltet ist,

In diesen Fällen liegt auch kein gesicherter Anruf vor.

A Abkürzungen

Diese Liste enthält die in diesem Handbuch verwendeten Abkürzungen.

Abkürzung	Definition
CA	Certificate Authority – Zertifizierungsstelle
DHCP	Dynamic Host Configuration Protocol.
DLS	Deployment and Licensing Service
DNS	Domain name server
EAP	Extensible Authentication Protocol
EAPOL	Extensible Authentication Protocol Over LAN
FTP	File Transfer Protocol.
IAS	Internet Authentication Service
IETF	Internet Engineering Task Force; Internet standards body
IIIS	Internet Information Server
IP	Internet Protocoll
PEAP	Protected Extensible Authentication Protocol
PKI	Public Key Infrastructure
RFC	Request For Comments; A IETF Protocol Specification
TAP	Techniker ArbeitsPlatz (Meist ein speziell mit Soft- und Hardware ausgestattetes Notebook des Technikers.)
TLS	Transport Layer Security
TTLS	Tunneled Transport Layer Security
VID	Virtual LAN ID (0-4095)
VLAN	Virtual LAN

www.siemens.com/enterprise

Die Informationen in diesem Dokument enthalten lediglich allgemeine Beschreibungen bzw. Leistungsmerkmale, welche im konkreten Anwendungsfall nicht immer in der beschriebenen Form zutreffen bzw. welche sich durch Weiterentwicklung der Produkte ändern können. Die gewünschten Leistungsmerkmale sind nur dann verbindlich, wenn sie bei Vertragsschluss ausdrücklich vereinbart werden. Die verwendeten Marken sind Eigentum der Siemens Enterprise Communications GmbH & Co. KG bzw. der jeweiligen Inhaber..

Hofmannstr. 51 • D-81359 München
Bestell-Nr.: A31003-J4270-T100-1-18