media5 corporation

Enabling Live Communications at the Edge of IP Networks

Configuration Notes 293

Enabling Security Features in Firmware DGW v2.0

June 22, 2011

Proprietary

Table of Contents

Scope

This document describes the steps required to configure a Mediatrix unit loaded with the DGW v2.0 firmware for secure SIP signalling and secure media (SRTP) operation. This is not a complete key-exchange, TLS or general security tutorial. For more information on those topics, please see the links section.

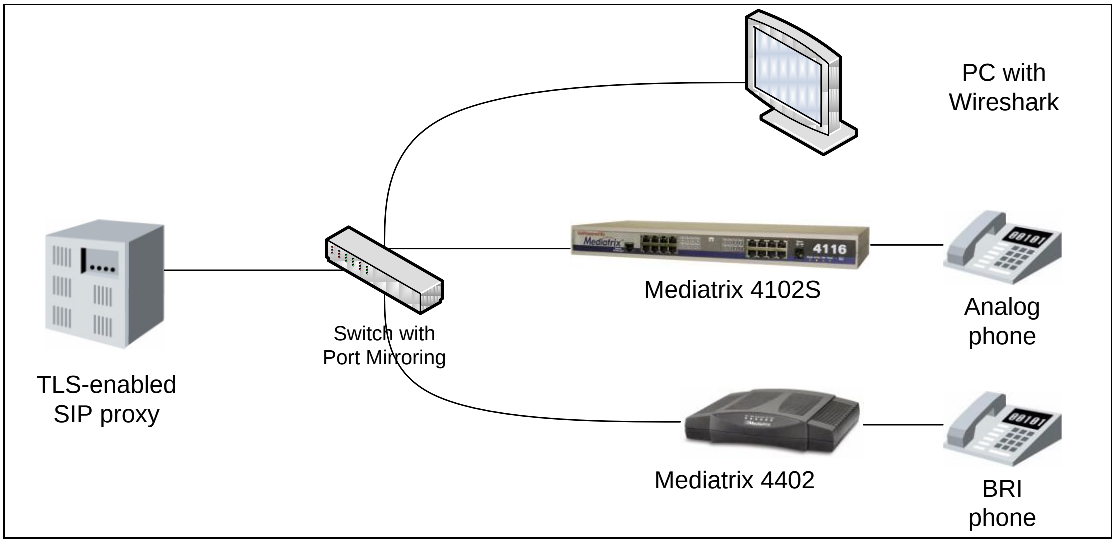
Acronyms and Definitions

RTP	Real Time Protocol
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SRTP	Secure Real Time Protocol
SSL	Secure Socket Layer
TLS	Transport Layer Security
Wireshark	Network sniffing and capture tool

Setup Description

In this scenario, the endpoints used are a Mediatrix 41XX and a Mediatrix 4402 BRI Gateway units. Both Mediatrix units must be loaded with Dgw v2.0. If the Mediatrix 41XX device is loaded with SIP v5.0, please refer to the *Technical Bulletin – Upgrading from SIP 5.0 to DGW 2.0* to update your firmware, as this is outside the scope of this document. We will use the freely available openSIPS (www.opensips.org) as the SIP proxy and configure it for TLS operation.

Note: The Mediatrix 4102**S** model supports DGW v2.0, while the regular 4102 model does not. All analog gateways with port densities starting with 4 (4104) and higher support DGW v2.0, as well as all the digital gateways 440X and 3XXX models.



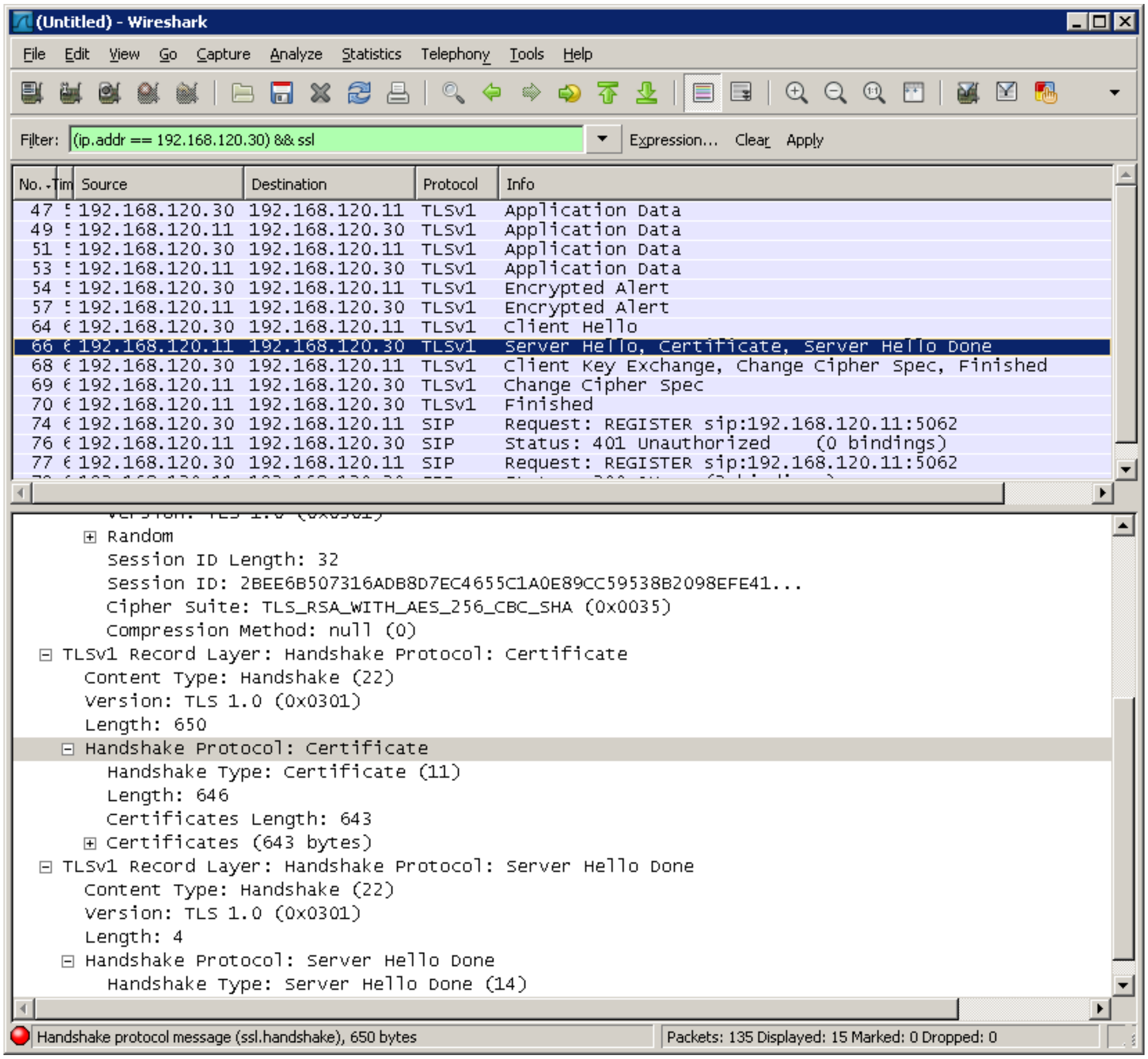
Basics of Security Exchanges

At the level at which we are working, establishing a TLS connection is fairly straightforward. In practise, at a lower level, there are quite a lot of additional complications to guard against various possible attacks.

This is the overall exchange in order to build the TLS link and bring it “up”:

- The client (Mediatrix) initially connects to the server on a configured TCP port (16000 is the default source port, the destination port is the configured SIP proxy port).
- The client sends a “Client Hello” message with the supported TLS/SSL protocol version, cipher specifications and compression algorithms.
- The server replies with a “Server Hello” message with the selected cipher and the server certificate.
- The client verifies the server certificate (validations are configured via the TlsCertificateValidation variable).
- The client generates a secret and encrypts it with the server’s public key. This encrypted secret is then sent to the server.
- The client and the server use the secret to create the same symmetric encryption key.
- The client and the server switch to encrypted communication by using the previously agreed cipher and the key just established.

This brief exchange can be seen in the following Wireshark capture.



(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Expression... Clear Apply

Filter: (ip.addr == 192.168.120.30) && ssl

```
No. -Tim Source          Destination     Protocol Info
 47  5   192.168.120.30  192.168.120.11  TLSv1    Application Data
 49  5   192.168.120.11  192.168.120.30  TLSv1    Application Data
 51  5   192.168.120.30  192.168.120.11  TLSv1    Application Data
 53  5   192.168.120.11  192.168.120.30  TLSv1    Application Data
 54  5   192.168.120.30  192.168.120.11  TLSv1    Encrypted Alert
 57  5   192.168.120.11  192.168.120.30  TLSv1    Encrypted Alert
 64  6   192.168.120.30  192.168.120.11  TLSv1    Client Hello
 66  6   192.168.120.11  192.168.120.30  TLSv1    Server Hello, Certificate, Server Hello Done
 68  6   192.168.120.30  192.168.120.11  TLSv1    Client Key Exchange, Change Cipher Spec, Finished
 69  6   192.168.120.11  192.168.120.30  TLSv1    Change Cipher Spec
 70  6   192.168.120.11  192.168.120.30  TLSv1    Finished
 74  6   192.168.120.30  192.168.120.11  SIP      Request: REGISTER sip:192.168.120.11:5062
 76  6   192.168.120.11  192.168.120.30  SIP      Status: 401 Unauthorized    (0 bindings)
 77  6   192.168.120.30  192.168.120.11  SIP      Request: REGISTER sip:192.168.120.11:5062
```

```
        Version: TLS 1.0 (0x0301)
      ⊞ Random
        Session ID Length: 32
        Session ID: 2BEE6B507316ADB8D7EC4655C1A0E89CC59538B2098EFE41...
        Cipher Suite: TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
        Compression Method: null (0)
  ⊟ TLSv1 Record Layer: Handshake Protocol: Certificate
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 650
    ⊟ Handshake Protocol: Certificate
        Handshake Type: Certificate (11)
        Length: 646
        Certificates Length: 643
      ⊞ Certificates (643 bytes)
  ⊟ TLSv1 Record Layer: Handshake Protocol: Server Hello Done
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 4
    ⊟ Handshake Protocol: Server Hello Done
        Handshake Type: Server Hello Done (14)
```

Handshake protocol message (ssl.handshake), 650 bytes

Packets: 135 Displayed: 15 Marked: 0 Dropped: 0

When obtaining the server certificates during the early negotiation, the following information will be checked by the client:

- the server signature,

- the CA (certification authority) who signed the certificate,
- validate that the server identified in the certificate is the same as the one that presented it,
- the expiration date of the certificate.

If any of these steps fails, your TLS link will not go "up". For those familiar with HTTPS, this is essentially the same procedure but using a SIP server/proxy instead of a HTTPS server.

Installation

Installing a TLS-Enabled Server/Proxy

Using two Mediatrix gateways connected back-to-back using a SIP trunk would be sufficient to demonstrate the use of the new security features. However, we prefer to demonstrate the configuration of the units and test scenarios in a more real-world environment by using a separate TLS-enabled SIP proxy. For this purpose, we have chosen openSIPS as it is free and easy to configure for basic use.

For more information on setting up openSIPS, please refer to the openSIPS installation documentation at www.opensips.org/docs. Otherwise skip this section.

Please note that (at the moment of this writing) by default opensips is configured to keep the TLS links up for a period of 2 minutes. We have made a small code modification that allows the links to stay up for 120 minutes. See the annex for more information.

About Certificates

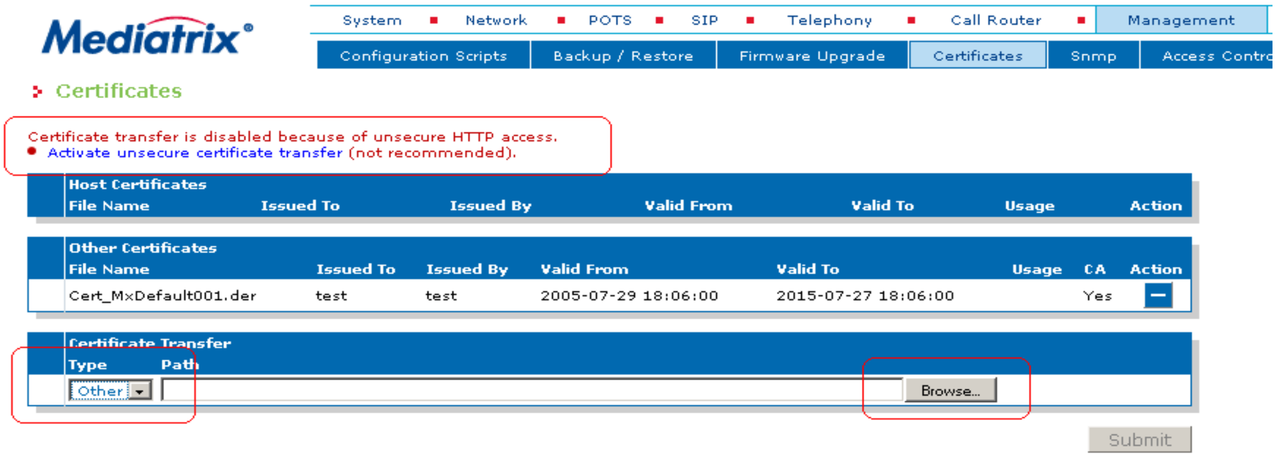
In order to enable TLS on the Mediatrix units, you will need at least a CA certificate that will validate that the certificate presented by the server is valid. This certificate must be uploaded to the Mediatrix units. In order to use the Wireshark features that are described later, a copy of the SIP server certificate containing its private key (this will be used to decrypt the TLS) will also be needed. The certificates need to be in ITU X.509 format.

For certificate creation, we recommend the FAQ page from the openssl project:

http://www.openssl.org/support/faq.html#USER3

Installing Certificates on the Mediatrix Unit

1) Navigate to the *Management-> Certificates* section.
2) Activate **unsecured certificate transfer**.
3) Select the certificate type **Other**, then click *Browse*. A pop-up explorer window appears and allows you to browse your local file system to locate the server's CA certificate file (usually with a .crt extension), using format X.509.
4) When the certificate is loaded, the required services must be restarted. This can be done by following the provided link at the top of the web page.



Certificate transfer is disabled because of unsecure HTTP access.

- Activate unsecure certificate transfer (not recommended).

Host Certificates

File Name	Issued To	Issued By	Valid From	Valid To	Usage	Action

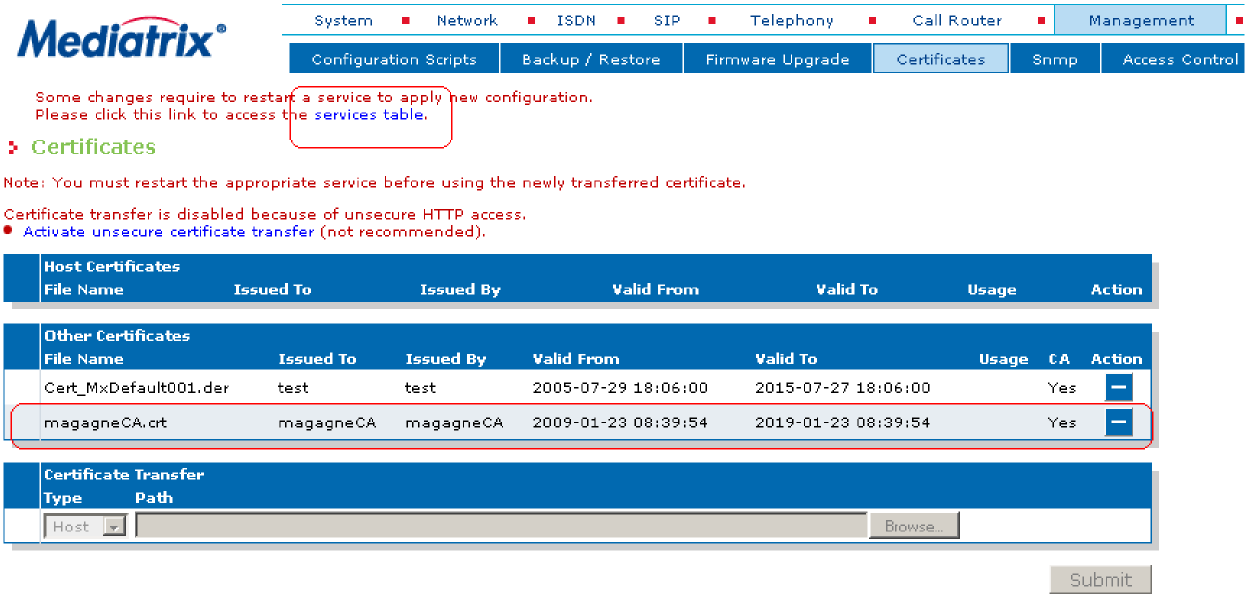
Other Certificates

File Name	Issued To	Issued By	Valid From	Valid To	Usage	CA	Action
Cert_MxDefault001.der	test	test	2005-07-29 18:06:00	2015-07-27 18:06:00		Yes	−

Certificate Transfer

Type	Path
Other	Browse...

Submit



Some changes require to restart a service to apply new configuration.
Please click this link to access the services table.

Certificates

Note: You must restart the appropriate service before using the newly transferred certificate.

Certificate transfer is disabled because of unsecure HTTP access.
- Activate unsecure certificate transfer (not recommended).

Host Certificates

File Name	Issued To	Issued By	Valid From	Valid To	Usage	Action

Other Certificates

File Name	Issued To	Issued By	Valid From	Valid To	Usage	CA	Action
Cert_MxDefault001.der	test	test	2005-07-29 18:06:00	2015-07-27 18:06:00		Yes	−
magagneCA.crt	magagneCA	magagneCA	2009-01-23 08:39:54	2019-01-23 08:39:54		Yes	−

Certificate Transfer

Type	Path
Host	

It is important to know the distinction between a “*Host*” and “*Other*” certificate.

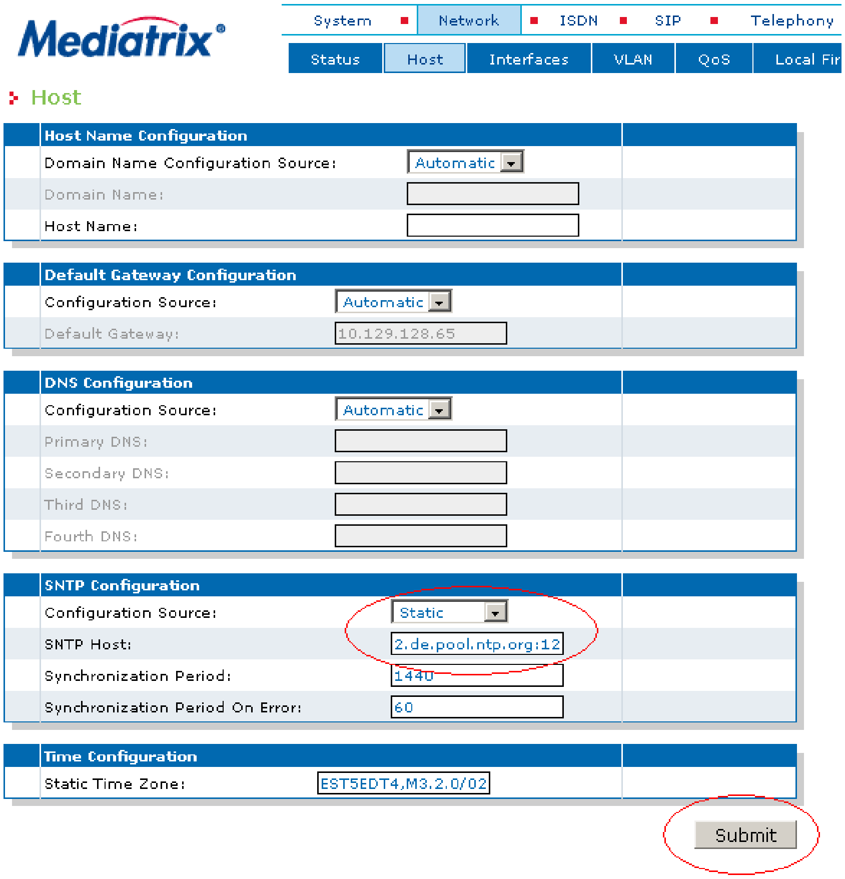
An “*Other*” certificate is simply a CA certificate used to validate the certificate of the server to which the Mediatrix unit is trying to connect.

A “Host” certificate is a server certificate that is required if the Mediatrix unit acts as a TLS server and presents its certificate to other clients. An example of this would be two Mediatrix gateways with no SIP proxy in the middle. At least one of the units will require a Host certificate. If only one unit has a Host certificate, the calls will be allowed in only one direction (Unit 1 calls Unit 2). For bi-directional calls, both Mediatrix units would require a Host certificate.

Note that by default it is not possible to upload a Host certificate without first clicking on **Activate unsecured certificate transfer**. This is because the certificate upload will be done in clear text, which means **the private key will be susceptible to interception!**

Important: Media5 recommends uploading Host certificates from a PC that is connected directly to the gateway.

Warning: Since certificates have a start date and expiry date, the use of NTP (Network Time Protocol) is now **mandatory** on the Mediatrix units when using the security features. To setup the NTP server, go to the *Network-> Host* section and configure your NTP server accordingly.



Host

Host Name Configuration	
Domain Name Configuration Source:	Automatic
Domain Name:	
Host Name:	

Default Gateway Configuration	
Configuration Source:	Automatic
Default Gateway:	10.129.128.65

DNS Configuration	
Configuration Source:	Automatic
Primary DNS:	
Secondary DNS:	
Third DNS:	
Fourth DNS:	

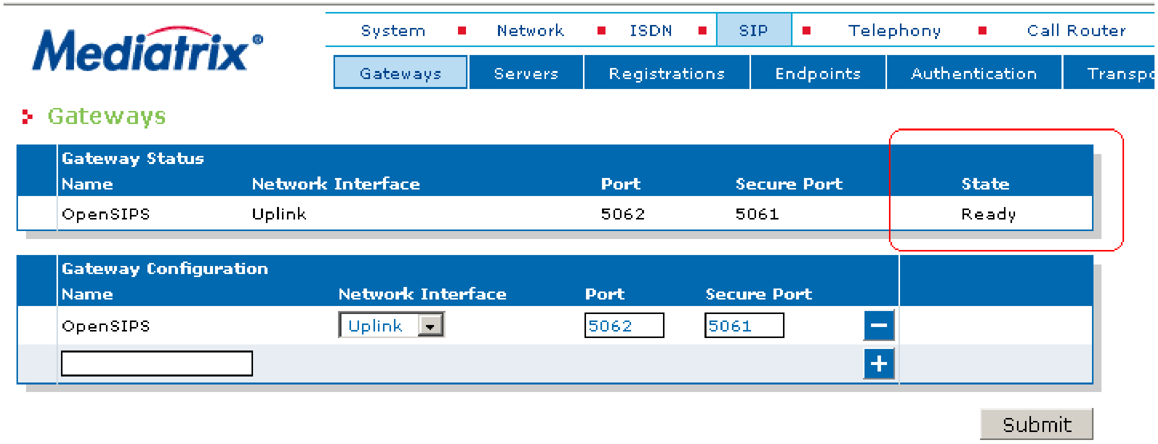
SNTP Configuration	
Configuration Source:	Static
SNTP Host:	2.de.pool.ntp.org:12
Synchronization Period:	1440
Synchronization Period On Error:	60

Time Configuration	
Static Time Zone:	EST5EDT4,M3.2.0/02

Mediatrix Configuration

SIP Gateway Configuration

Before using TLS, the SIP gateway needs to be properly configured. To do that, go to the *SIP-> Gateways* section.

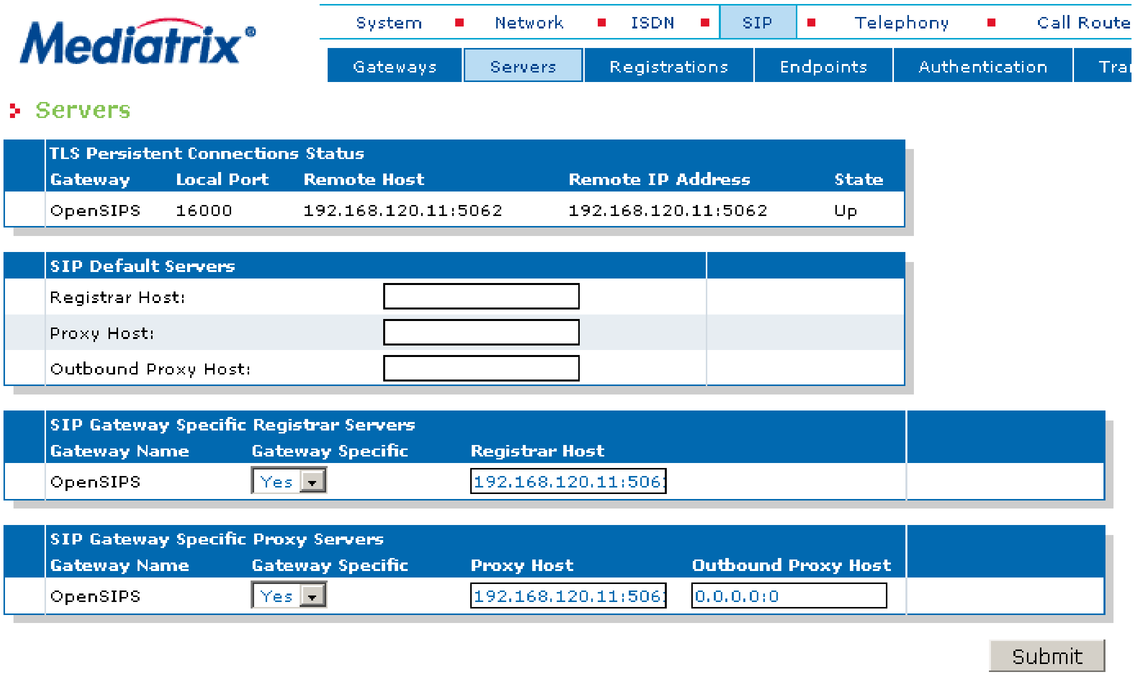


Gateways

Gateway Status				
Name	Network Interface	Port	Secure Port	State
OpenSIPS	Uplink	5062	5061	Ready

Gateway Configuration			
Name	Network Interface	Port	Secure Port
OpenSIPS	Uplink	5062	5061

In this example, the gateway called “OpenSIPS” is listening on port 5062. To configure the gateway, click the **Servers** tab.



Servers

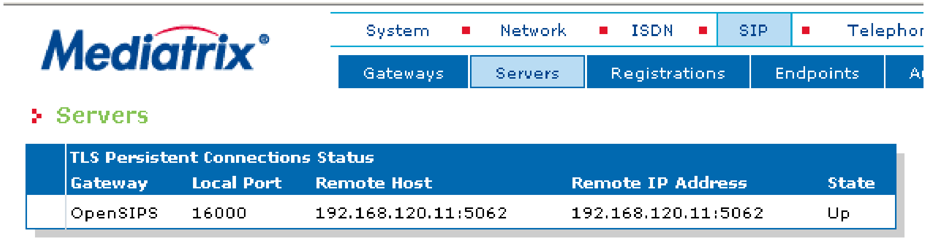
TLS Persistent Connections Status				
Gateway	Local Port	Remote Host	Remote IP Address	State
OpenSIPS	16000	192.168.120.11:5062	192.168.120.11:5062	Up

SIP Default Servers	
Registrar Host:	
Proxy Host:	
Outbound Proxy Host:	

SIP Gateway Specific Registrar Servers		
Gateway Name	Gateway Specific	Registrar Host
OpenSIPS	Yes	192.168.120.11:506

SIP Gateway Specific Proxy Servers			
Gateway Name	Gateway Specific	Proxy Host	Outbound Proxy Host
OpenSIPS	Yes	192.168.120.11:506	0.0.0.0:0

For settings that are gateway-specific, use the *Gateway Specific* sections. In the previous example, the settings are valid only for the “OpenSIPS” gateway. Both the SIP Registrar and SIP Proxy are configured to 192.168.120.11 on port 5062.



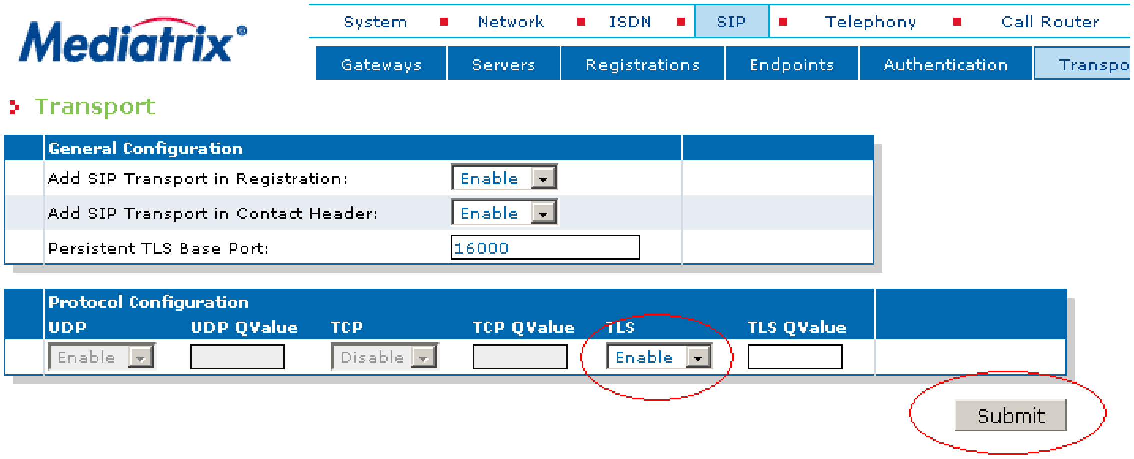
Servers

TLS Persistent Connections Status				
Gateway	Local Port	Remote Host	Remote IP Address	State
OpenSIPS	16000	192.168.120.11:5062	192.168.120.11:5062	Up

Enabling Secure Signalling (TLS)

The Mediatrix unit does not support mixing TLS and non-TLS links. This means that it is not possible to configure separate gateways (*SIP-> Gateways*) using secure and non-secure links. Once TLS is enabled, it is enabled for all configured gateways.

Go to the *SIP-> Transport* tab and simply enable TLS, click *Submit* and follow the link to start the appropriate service. Please notice the configuration field for the previously discussed port 16000.



Transport

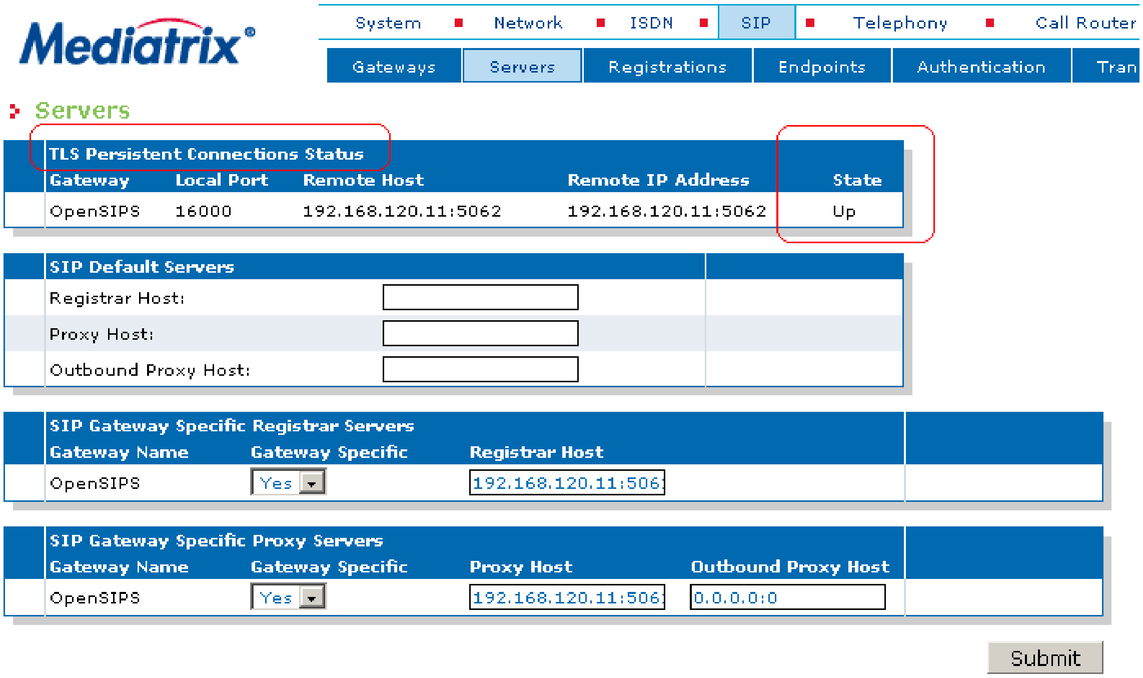
General Configuration	
Add SIP Transport in Registration:	Enable
Add SIP Transport in Contact Header:	Enable
Persistent TLS Base Port:	16000

Protocol Configuration

UDP	UDP QValue	TCP	TCP QValue	TLS	TLS QValue
Enable		Disable		Enable	

If the TLS link is established, the "Ready" LED on the Mediatrix unit turns on steady green. The status of the TLS link can also be found in the web page and in the syslog.

Warning: This "Ready" LED is available only on the 41XX and 440X models. The 3XXX models have no such LED.



Servers

TLS Persistent Connections Status

Gateway	Local Port	Remote Host	Remote IP Address	State
OpenSIPS	16000	192.168.120.11:5062	192.168.120.11:5062	Up

SIP Default Servers	
Registrar Host:	
Proxy Host:	
Outbound Proxy Host:	

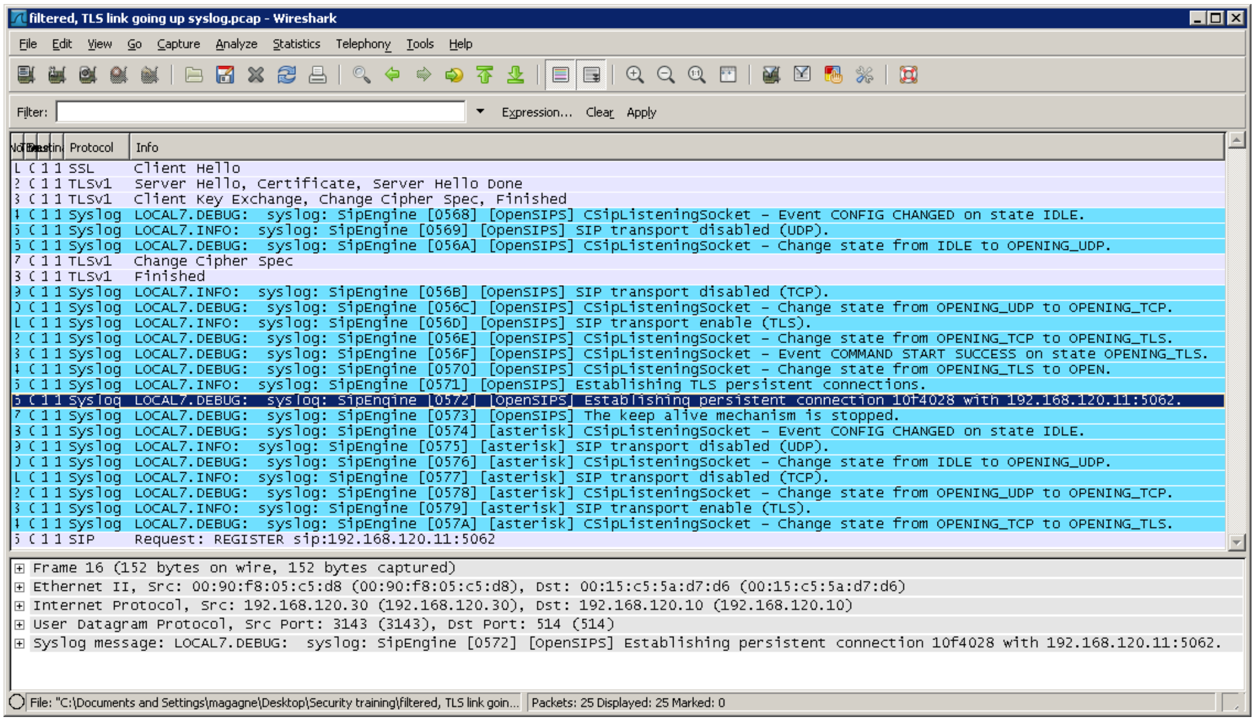
SIP Gateway Specific Registrar Servers

Gateway Name	Gateway Specific	Registrar Host
OpenSIPS	Yes	192.168.120.11:506

SIP Gateway Specific Proxy Servers

Gateway Name	Gateway Specific	Proxy Host	Outbound Proxy Host
OpenSIPS	Yes	192.168.120.11:506	0.0.0.0:0

A syslog message will be sent saying "establishing persistent connection"


filtered, TLS link going up syslog.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
No. Time Source Destination Protocol Info

```
1 ( 1 1 SSL     Client Hello
2 ( 1 1 TLSv1   Server Hello, Certificate, Server Hello Done
3 ( 1 1 TLSv1   Client Key Exchange, Change Cipher Spec, Finished
4 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0568] [OpenSIPS] CSipListeningSocket - Event CONFIG CHANGED on state IDLE.
5 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [0569] [OpenSIPS] SIP transport disabled (UDP).
6 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [056A] [OpenSIPS] CSipListeningSocket - Change state from IDLE to OPENING_UDP.
7 ( 1 1 TLSv1   Change Cipher Spec
8 ( 1 1 TLSv1   Finished
9 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [056B] [OpenSIPS] SIP transport disabled (TCP).
0 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [056C] [OpenSIPS] CSipListeningSocket - Change state from OPENING_UDP to OPENING_TCP.
1 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [056D] [OpenSIPS] SIP transport enable (TLS).
2 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [056E] [OpenSIPS] CSipListeningSocket - Change state from OPENING_TCP to OPENING_TLS.
3 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [056F] [OpenSIPS] CSipListeningSocket - Event COMMAND START SUCCESS on state OPENING_TLS.
4 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0570] [OpenSIPS] CSipListeningSocket - Change state from OPENING_TLS to OPEN.
5 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [0571] [OpenSIPS] Establishing TLS persistent connections.
6 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0572] [OpenSIPS] Establishing persistent connection 10f4028 with 192.168.120.11:5062.
7 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0573] [OpenSIPS] The keep alive mechanism is stopped.
8 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0574] [asterisk] CSipListeningSocket - Event CONFIG CHANGED on state IDLE.
9 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [0575] [asterisk] SIP transport disabled (UDP).
0 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0576] [asterisk] CSipListeningSocket - Change state from IDLE to OPENING_UDP.
1 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [0577] [asterisk] SIP transport disabled (TCP).
2 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [0578] [asterisk] CSipListeningSocket - Change state from OPENING_UDP to OPENING_TCP.
3 ( 1 1 Syslog  LOCAL7.INFO:  syslog: SipEngine [0579] [asterisk] SIP transport enable (TLS).
4 ( 1 1 Syslog  LOCAL7.DEBUG:  syslog: SipEngine [057A] [asterisk] CSipListeningSocket - Change state from OPENING_TCP to OPENING_TLS.
5 ( 1 1 SIP     Request: REGISTER sip:192.168.120.11:5062
```

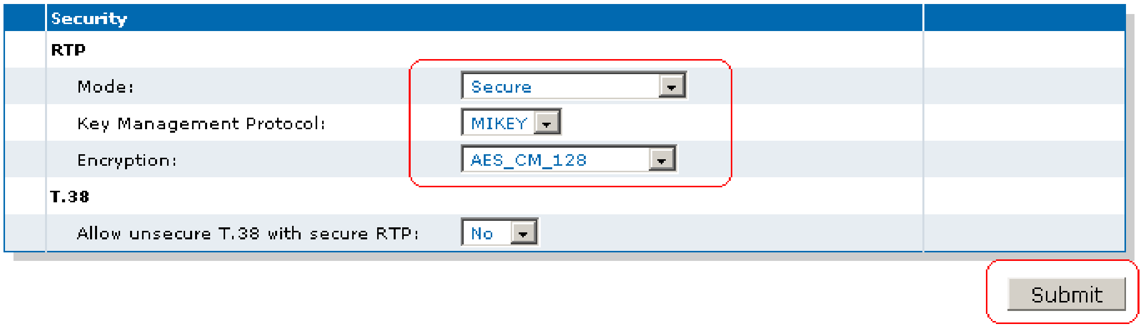
```
⊞ Frame 16 (152 bytes on wire, 152 bytes captured)
⊞ Ethernet II, Src: 00:90:f8:05:c5:d8 (00:90:f8:05:c5:d8), Dst: 00:15:c5:5a:d7:d6 (00:15:c5:5a:d7:d6)
⊞ Internet Protocol, Src: 192.168.120.30 (192.168.120.30), Dst: 192.168.120.10 (192.168.120.10)
⊞ User Datagram Protocol, Src Port: 3143 (3143), Dst Port: 514 (514)
⊞ Syslog message: LOCAL7.DEBUG:  syslog: SipEngine [0572] [OpenSIPS] Establishing persistent connection 10f4028 with 192.168.120.11:5062.
```

File: "C:\Documents and Settings\magagne\Desktop\Security training\filtered, TLS link goin... Packets: 25 Displayed: 25 Marked: 0

Enabling Secure Media (SRTP)

Now that encrypted signalling is configured, the media streams can also be encrypted and secured. Without encryption, RTP is still vulnerable to interception.

1) Go to the *Telephony -> CODECs* page and enable secure RTP by changing the *Mode* to **Secure**.
2) Choose a Key Management Protocol. The Mediatrix unit supports both MIKEY and SDES.
3) Choose the encryption algorithm. Currently the Mediatrix unit supports AES with 128 bits. The choice "NULL" will not encrypt the RTP. This should be selected only for debugging purposes.
4) Click *Submit*.



Security

RTP

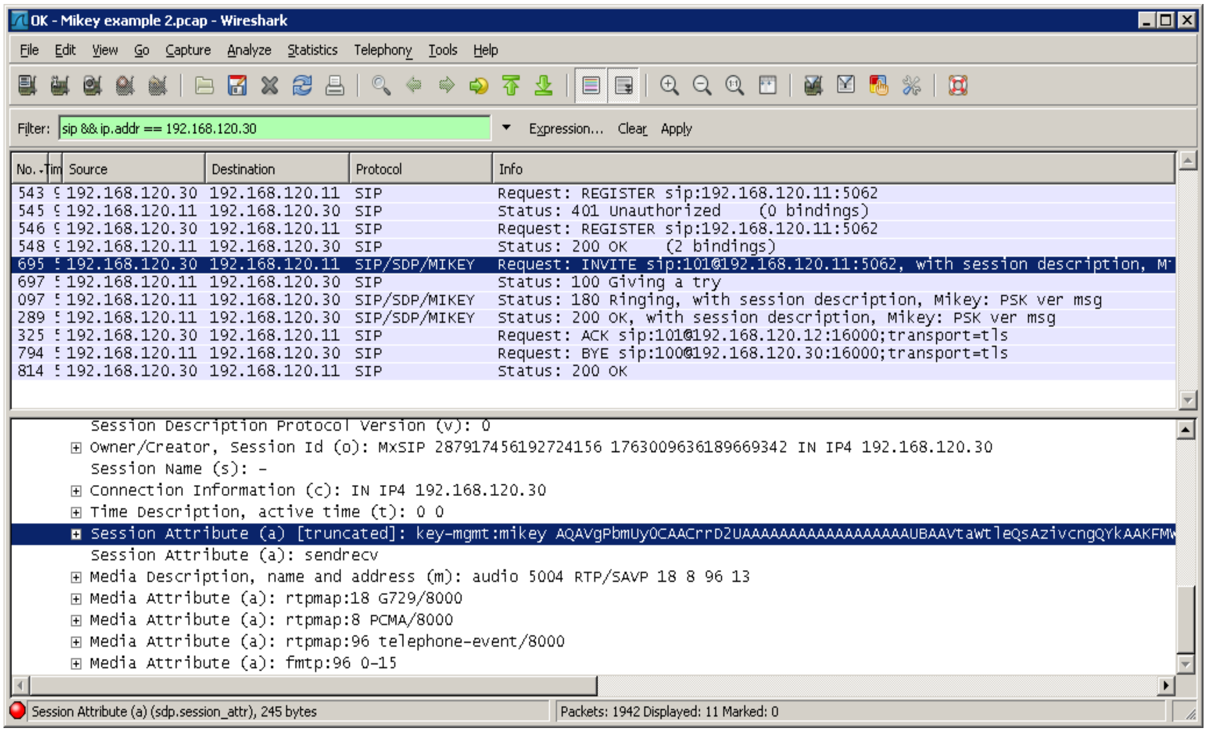
Mode:	Secure
Key Management Protocol:	MIKEY
Encryption:	AES_CM_128

T.38

Allow unsecure T.38 with secure RTP:	No

Note: T.38 packets will never be encrypted. Setting "Allow unsecured T.38 with secure RTP" will allow using T.38, otherwise it would be rejected.

In the following Wireshark trace, the Mikey parameters are sent in the INVITE SDP.



OK - Mikey example 2.pcap - Wireshark

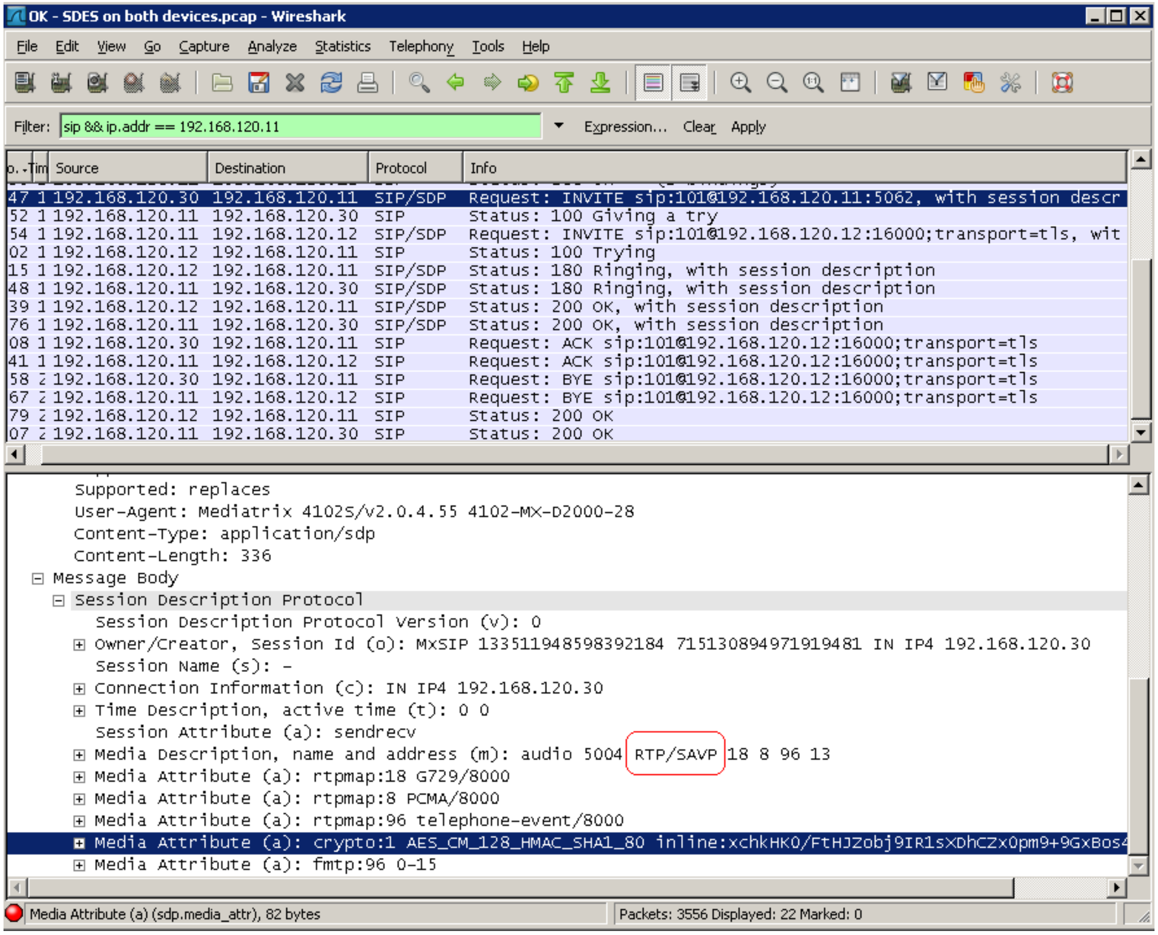
Filter: sip && ip.addr == 192.168.120.30

```
No. Tim Source         Destination    Protocol        Info
543 9 192.168.120.30 192.168.120.11 SIP             Request: REGISTER sip:192.168.120.11:5062
545 9 192.168.120.11 192.168.120.30 SIP             Status: 401 Unauthorized    (0 bindings)
546 9 192.168.120.30 192.168.120.11 SIP             Request: REGISTER sip:192.168.120.11:5062
548 9 192.168.120.11 192.168.120.30 SIP             Status: 200 OK    (2 bindings)
695 5 192.168.120.30 192.168.120.11 SIP/SDP/MIKEY   Request: INVITE sip:101@192.168.120.11:5062, with session description, M
697 5 192.168.120.11 192.168.120.30 SIP             Status: 100 Giving a try
097 5 192.168.120.11 192.168.120.30 SIP/SDP/MIKEY   Status: 180 Ringing, with session description, Mikey: PSK ver msg
289 5 192.168.120.11 192.168.120.30 SIP/SDP/MIKEY   Status: 200 OK, with session description, Mikey: PSK ver msg
325 5 192.168.120.30 192.168.120.11 SIP             Request: ACK sip:101@192.168.120.12:16000;transport=tls
794 5 192.168.120.11 192.168.120.30 SIP             Request: BYE sip:100@192.168.120.30:16000;transport=tls
814 5 192.168.120.30 192.168.120.11 SIP             Status: 200 OK
```

```
      Session Description Protocol Version (v): 0
    ⊞ Owner/Creator, Session Id (o): MxSIP 287917456192724156 1763009636189669342 IN IP4 192.168.120.30
      Session Name (s): -
    ⊞ Connection Information (c): IN IP4 192.168.120.30
    ⊞ Time Description, active time (t): 0 0
    ⊞ Session Attribute (a) [truncated]: key-mgmt:mikey AQAVgPbmUy0CAACrrD2UAAAAAAAAAAAAAAAAAAUBAAVtaWtleQsAzivcngQYkAAKFMW
      Session Attribute (a): sendrecv
    ⊞ Media Description, name and address (m): audio 5004 RTP/SAVP 18 8 96 13
    ⊞ Media Attribute (a): rtpmap:18 G729/8000
    ⊞ Media Attribute (a): rtpmap:8 PCMA/8000
    ⊞ Media Attribute (a): rtpmap:96 telephone-event/8000
    ⊞ Media Attribute (a): fmtp:96 0-15
```

Session Attribute (a) (sdp.session_attr), 245 bytes | Packets: 1942 Displayed: 11 Marked: 0

Enabling SDES instead of Mickey, the INVITE will be slightly different. SDES parameters will be added to the SDP Media Attributes instead of the Session Attributes.



OK - SDES on both devices.pcap - Wireshark

Filter: sip && ip.addr == 192.168.120.11

```
o. Tim Source         Destination    Protocol   Info
47 1 192.168.120.30 192.168.120.11 SIP/SDP    Request: INVITE sip:101@192.168.120.11:5062, with session descr
52 1 192.168.120.11 192.168.120.30 SIP        Status: 100 Giving a try
54 1 192.168.120.11 192.168.120.12 SIP/SDP    Request: INVITE sip:101@192.168.120.12:16000;transport=tls, wit
02 1 192.168.120.12 192.168.120.11 SIP        Status: 100 Trying
15 1 192.168.120.12 192.168.120.11 SIP/SDP    Status: 180 Ringing, with session description
48 1 192.168.120.11 192.168.120.30 SIP/SDP    Status: 180 Ringing, with session description
39 1 192.168.120.12 192.168.120.11 SIP/SDP    Status: 200 OK, with session description
76 1 192.168.120.11 192.168.120.30 SIP/SDP    Status: 200 OK, with session description
08 1 192.168.120.30 192.168.120.11 SIP        Request: ACK sip:101@192.168.120.12:16000;transport=tls
41 1 192.168.120.11 192.168.120.12 SIP        Request: ACK sip:101@192.168.120.12:16000;transport=tls
58 2 192.168.120.30 192.168.120.11 SIP        Request: BYE sip:101@192.168.120.12:16000;transport=tls
67 2 192.168.120.11 192.168.120.12 SIP        Request: BYE sip:101@192.168.120.12:16000;transport=tls
79 2 192.168.120.12 192.168.120.11 SIP        Status: 200 OK
07 2 192.168.120.11 192.168.120.30 SIP        Status: 200 OK
```

```
      Supported: replaces
      User-Agent: Mediatrix 4102S/v2.0.4.55 4102-MX-D2000-28
      Content-Type: application/sdp
      Content-Length: 336
  ⊟ Message Body
    ⊟ Session Description Protocol
        Session Description Protocol Version (v): 0
      ⊞ Owner/Creator, Session Id (o): MxSIP 133511948598392184 715130894971919481 IN IP4 192.168.120.30
        Session Name (s): -
      ⊞ Connection Information (c): IN IP4 192.168.120.30
      ⊞ Time Description, active time (t): 0 0
        Session Attribute (a): sendrecv
      ⊞ Media Description, name and address (m): audio 5004 RTP/SAVP 18 8 96 13
      ⊞ Media Attribute (a): rtpmap:18 G729/8000
      ⊞ Media Attribute (a): rtpmap:8 PCMA/8000
      ⊞ Media Attribute (a): rtpmap:96 telephone-event/8000
      ⊞ Media Attribute (a): crypto:1 AES_CM_128_HMAC_SHA1_80 inline:xchkHK0/FtHJZobj9IR1sXDhCZx0pm9+9GxBos4
      ⊞ Media Attribute (a): fmtp:96 0-15
```

Media Attribute (a) (sdp.media_attr), 82 bytes | Packets: 3556 Displayed: 22 Marked: 0

The *RTP/SAVP* is a flag which states that the endpoint is attempting to initiate a secure media connection. Seethe text in red in the above example.

Troubleshooting

To troubleshoot when using security, Wireshark must be configured for TLS sniffing.

The following are a few examples of issues that may be encountered while configuring TLS.

Enabling TLS Debugging on Wireshark

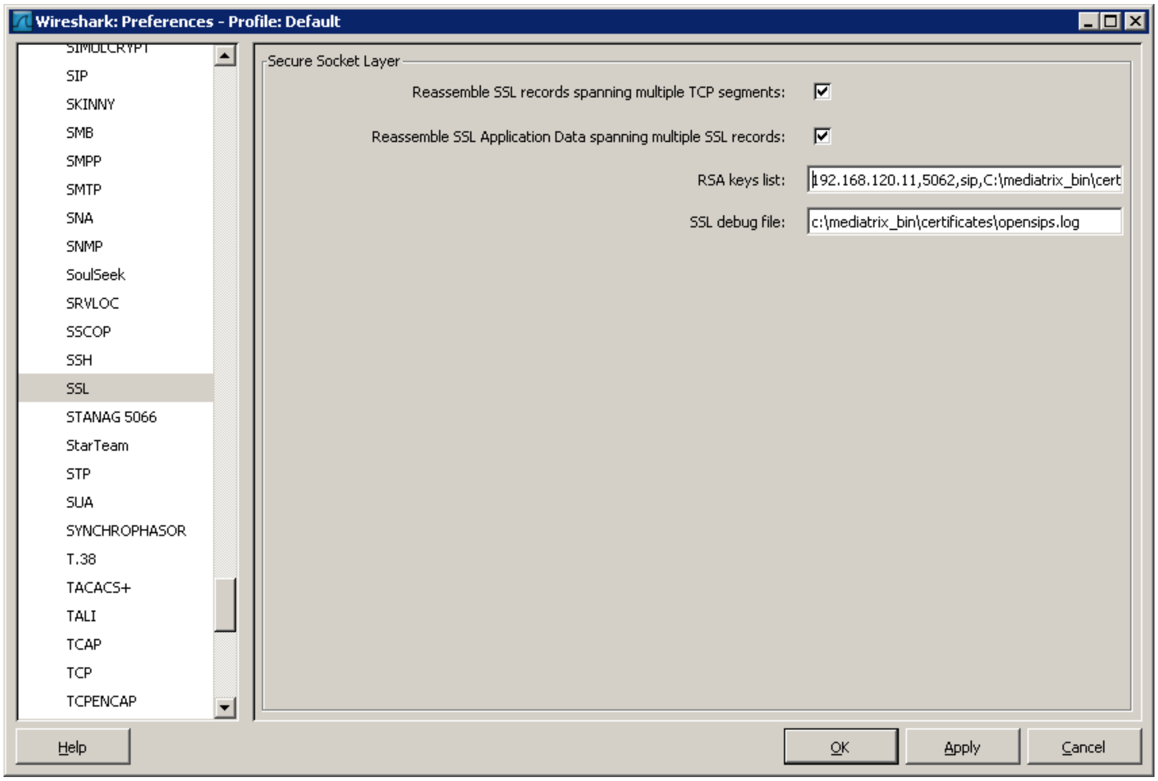
Once the TLS link is up, it is no longer possible to read the SIP packets as they are TLS-encrypted. To debug TLS, Wireshark needs to be configured to decrypt them. For this step, the public keys associated with the server certificate are needed.

1) In the *Edit/Preferences* dialog, select the Protocols/SSL node and fill the RSA key list. The field specifies the binding between an IP address, a port, a protocol and a RSA decryption key. Enter the IP address of the server, the SIP port and the path to the file containing the server private key. Several such bindings may be specified by separating them with a semi-colon “;”.

   Example: The server is located at 192.168.120.11 and listens on port 5062

   192.168.120.11,5062,sip,C: \certificates\192.168.120.11.key

   When having difficulty decrypting SIP packets, the “SSL debug file” may be used to determine what is going wrong.



Reassemble SSL records spanning multiple TCP segments: ☑

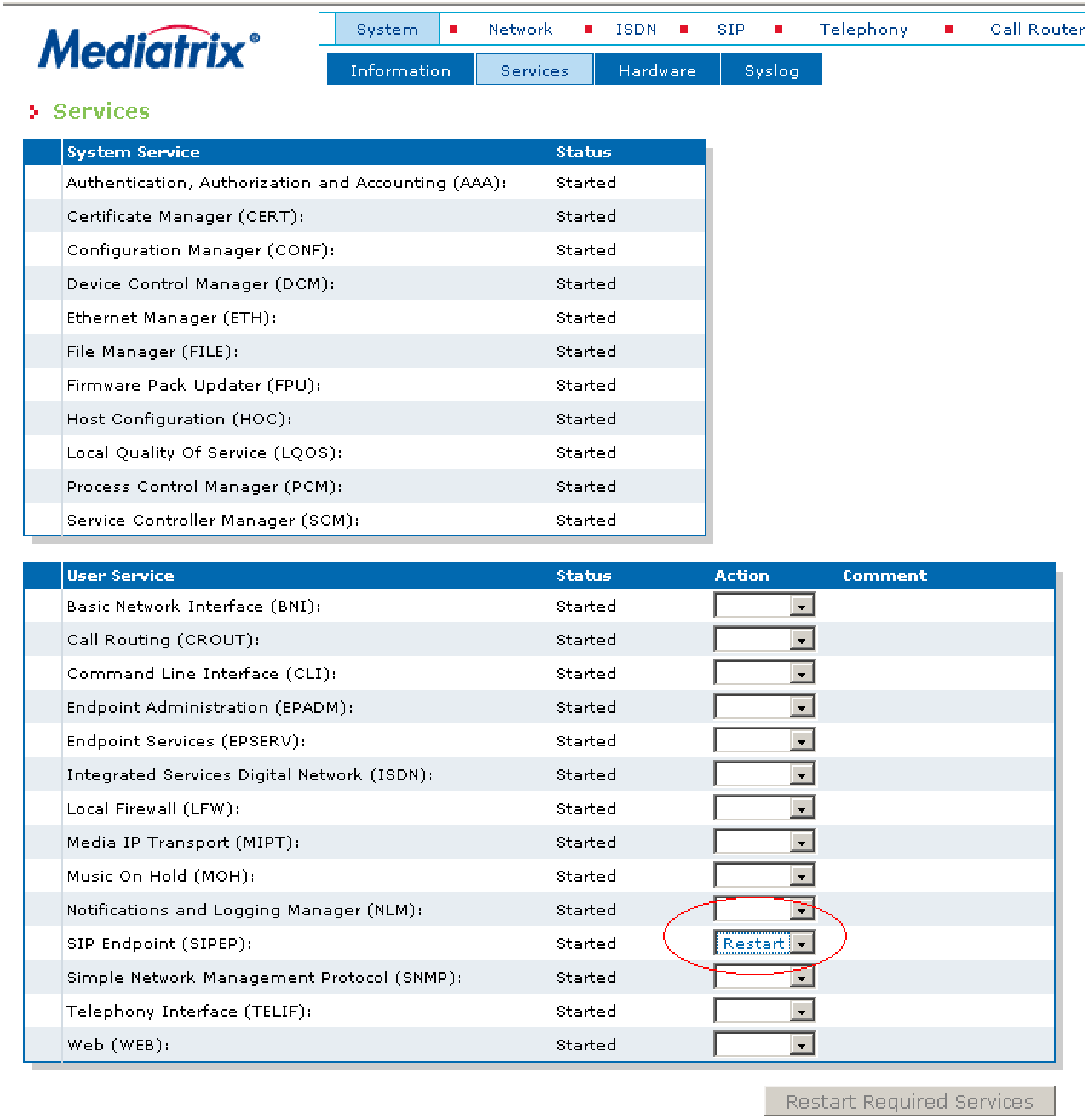
Reassemble SSL Application Data spanning multiple SSL records: ☑

RSA keys list: 192.168.120.11,5062,sip,C:\mediatrix_bin\cert

SSL debug file: c:\mediatrix_bin\certificates\opensips.log

2) Start the Wireshark capture.

3) Restart the SIPEP service on the Mediatrix unit or simply reboot the unit. This will enable the TLS renegotiation.



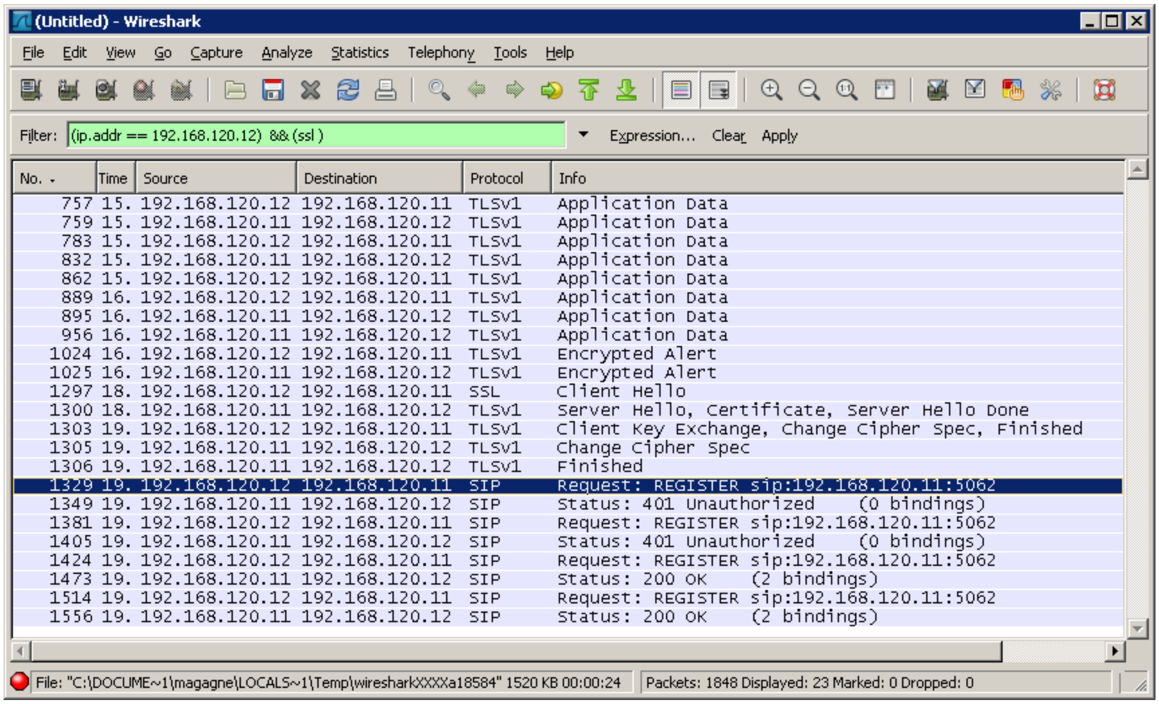
Services

System Service	Status
Authentication, Authorization and Accounting (AAA):	Started
Certificate Manager (CERT):	Started
Configuration Manager (CONF):	Started
Device Control Manager (DCM):	Started
Ethernet Manager (ETH):	Started
File Manager (FILE):	Started
Firmware Pack Updater (FPU):	Started
Host Configuration (HOC):	Started
Local Quality Of Service (LQOS):	Started
Process Control Manager (PCM):	Started
Service Controller Manager (SCM):	Started

User Service	Status	Action	Comment
Basic Network Interface (BNI):	Started		
Call Routing (CROUT):	Started		
Command Line Interface (CLI):	Started		
Endpoint Administration (EPADM):	Started		
Endpoint Services (EPSERV):	Started		
Integrated Services Digital Network (ISDN):	Started		
Local Firewall (LFW):	Started		
Media IP Transport (MIPT):	Started		
Music On Hold (MOH):	Started		
Notifications and Logging Manager (NLM):	Started		
SIP Endpoint (SIPEP):	Started	Restart	
Simple Network Management Protocol (SNMP):	Started		
Telephony Interface (TELIF):	Started		
Web (WEB):	Started		

4) When the unit is rebooted and the “Ready” LED is lit on the Mediatrix unit, stop the packet capture.

5) Using the “ssl” filter in the capture should show the SIP packets between the two endpoints.


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Expression... Clear Apply

Filter: (ip.addr == 192.168.120.12) && (ssl)

```
No. -  Time Source         Destination     Protocol Info
  757 15. 192.168.120.12 192.168.120.11 TLSv1    Application Data
  759 15. 192.168.120.11 192.168.120.12 TLSv1    Application Data
  783 15. 192.168.120.12 192.168.120.11 TLSv1    Application Data
  832 15. 192.168.120.11 192.168.120.12 TLSv1    Application Data
  862 15. 192.168.120.12 192.168.120.11 TLSv1    Application Data
  889 16. 192.168.120.12 192.168.120.11 TLSv1    Application Data
  895 16. 192.168.120.11 192.168.120.12 TLSv1    Application Data
  956 16. 192.168.120.11 192.168.120.12 TLSv1    Application Data
 1024 16. 192.168.120.12 192.168.120.11 TLSv1    Encrypted Alert
 1025 16. 192.168.120.11 192.168.120.12 TLSv1    Encrypted Alert
 1297 18. 192.168.120.12 192.168.120.11 SSL      Client Hello
 1300 18. 192.168.120.11 192.168.120.12 TLSv1    Server Hello, Certificate, Server Hello Done
 1303 19. 192.168.120.12 192.168.120.11 TLSv1    Client Key Exchange, Change Cipher Spec, Finished
 1305 19. 192.168.120.11 192.168.120.12 TLSv1    Change Cipher Spec
 1306 19. 192.168.120.11 192.168.120.12 TLSv1    Finished
 1329 19. 192.168.120.12 192.168.120.11 SIP      Request: REGISTER sip:192.168.120.11:5062
 1349 19. 192.168.120.11 192.168.120.12 SIP      Status: 401 Unauthorized    (0 bindings)
 1381 19. 192.168.120.12 192.168.120.11 SIP      Request: REGISTER sip:192.168.120.11:5062
 1405 19. 192.168.120.11 192.168.120.12 SIP      Status: 401 Unauthorized    (0 bindings)
 1424 19. 192.168.120.12 192.168.120.11 SIP      Request: REGISTER sip:192.168.120.11:5062
 1473 19. 192.168.120.11 192.168.120.12 SIP      Status: 200 OK     (2 bindings)
 1514 19. 192.168.120.12 192.168.120.11 SIP      Request: REGISTER sip:192.168.120.11:5062
 1556 19. 192.168.120.11 192.168.120.12 SIP      Status: 200 OK     (2 bindings)
```

File: "C:\DOCUME~1\magagne\LOCALS~1\Temp\wiresharkXXXXa18584" 1520 KB 00:00:24 Packets: 1848 Displayed: 23 Marked: 0 Dropped: 0

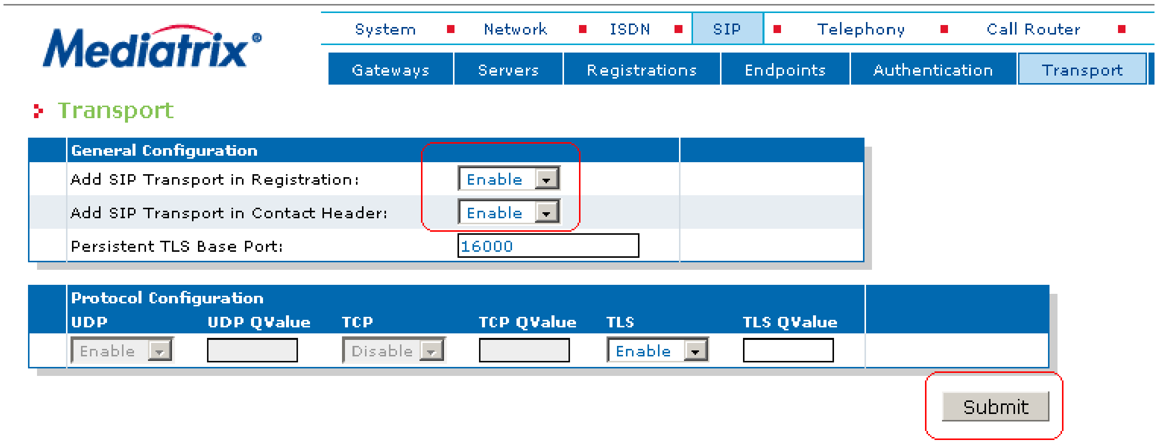
REGISTER Messages Not Being Answered

In the first example, TLS is enabled on one of the Mediatrix gateways and not on the second gateway.

The REGISTER requests from the second gateway are not being answered. This is because the proxy is expecting the SIP message to be SSL encapsulated. Simply restart the Wireshark capture and enable TLS on the second gateway. Restart the required services

Server Internal Error (or Similar Messages)

Some servers/proxies will require Interop variables to be enabled. For example, the default openSIPS installation requires adding the SIP transport field in the registration and contact headers. To do so, set the following variables to **Enable**.



General Configuration

Add SIP Transport in Registration:	Enable
Add SIP Transport in Contact Header:	Enable
Persistent TLS Base Port:	16000

Protocol Configuration

UDP	UDP QValue	TCP	TCP QValue	TLS	TLS QValue
Enable		Disable		Enable	

Below is a SIP Register message from one endpoint (192.168.120.30) that has the TLS transport in the Contact Header disabled and also a SIP Register message from the other endpoint that has the Contact Header enabled (192.168.120.12).

Register from 192.168.120.30

```
REGISTER sip:192.168.120.11:5062 SIP/2.0
Via: SIP/2.0/TLS 192.168.120.30:16000;branch=z9hG4bK840120998b9b8d813.231a4bf34e2eaa130
Max-Forwards: 70
From: <sip:100@192.168.120.11:5062>;tag=2ce647ee6c
To: <sip:100@192.168.120.11:5062>
```

```
Call-ID: a1b5ddebef59717a
CSeq: 151405030 REGISTER
Authorization: Digest
username="100",realm="192.168.120.11",nonce="4a5e430d000006aba1954cd956e94d0dc440d94d977f8d3
a",uri="sip:192.168.120.11:5062",response="c42b06827c08018c8c34cd0696269193"
Contact: <sip:100@192.168.120.30:16000> (NO TRANSPORT METHOD IN HEADER)
User-Agent: Mediatrix 4102S/v2.0.4.55 4102-MX-D2000-28
Content-Length: 0
```

Invite from 192.168.120.12

```
INVITE sip:100@192.168.120.11:5062 SIP/2.0
Via: SIP/2.0/TLS 192.168.120.12:16000;branch=z9hG4bK6a4e16b7b478eae83.051f66c579acf19dd
Max-Forwards: 70
From: <sip:101@192.168.120.11:5062>;tag=e437e6cd75
To: <sip:100@192.168.120.11:5062>
Call-ID: 822ebcc6433a7565
CSeq: 1793624545 INVITE
Allow: INVITE, ACK, BYE, CANCEL, REFER, NOTIFY, UPDATE
Contact: <sip:101@192.168.120.12:16000;transport=tls>
Min-SE: 1800
Session-Expires: 3600
Supported: timer
Supported: replaces
User-Agent: Mediatrix 4402 plus/v2.0.4.55 44XX-MX-D2000-36
Content-Type: application/sdp
Content-Length: 300
```

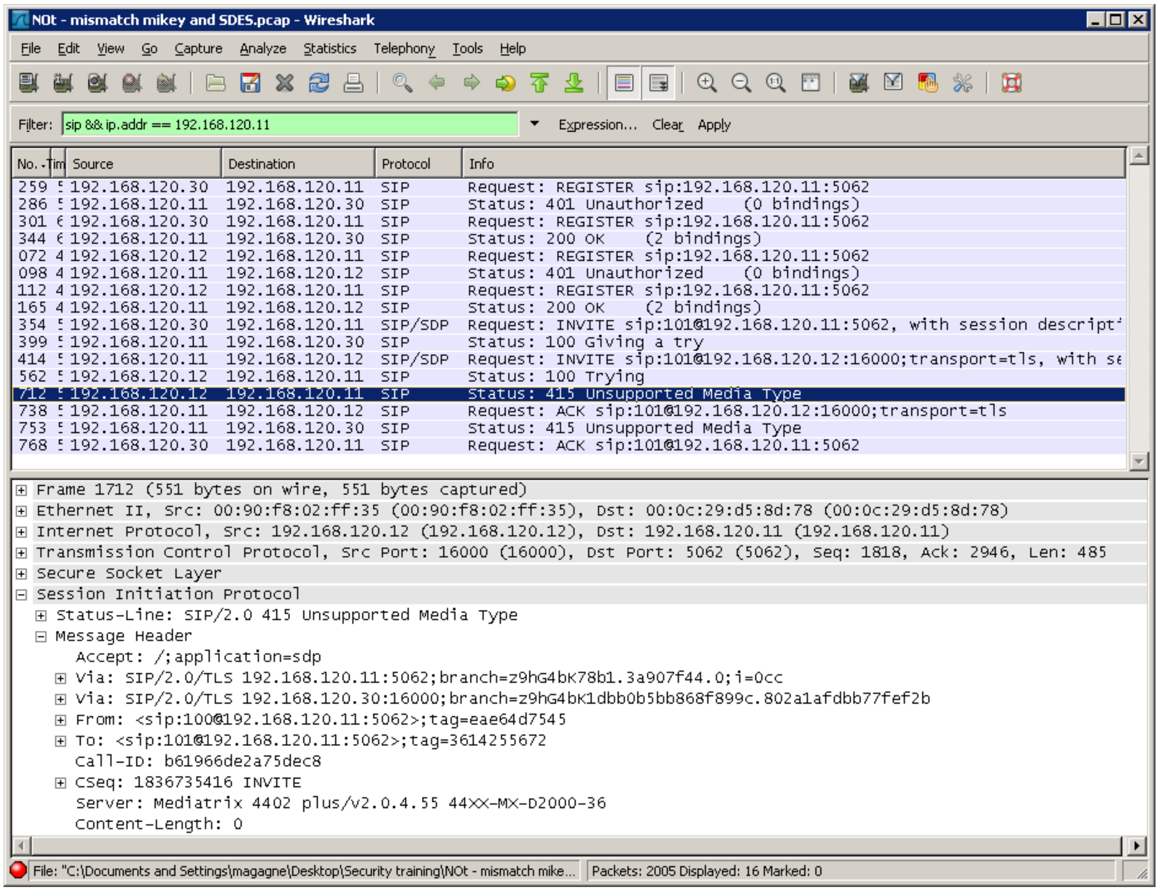
Here is the REGISTER for a subsequent working call with the Interop variable enabled.

```
REGISTER sip:192.168.120.11:5062 SIP/2.0
Via: SIP/2.0/TLS 192.168.120.30:16000;branch=z9hG4bK7006d85fe396d7632.499de40d84094b998
Max-Forwards: 70
From: <sip:100@192.168.120.11:5062>;tag=586e1a152b
To: <sip:100@192.168.120.11:5062>
Call-ID: 83216656d213ac84
CSeq: 1378359313 REGISTER
Authorization: Digest
username="100",realm="192.168.120.11",nonce="4a669cc80000000b95e8ef28fe0ce518d557d54ad3cc655
a",uri="sip:192.168.120.11:5062",response="214ede0b4eda7b7e6d6d2e03eb013755"
Contact: <sip:100@192.168.120.30:16000;transport=tls>
User-Agent: Mediatrix 4102S/v2.0.4.55 4102-MX-D2000-28
Content-Length: 0
```

Mikey and SDES Mismatch

It is strongly recommended to select only one single key management protocol. In the following example, SDES is configured on endpoint 1 (192.168.120.30) and Mikey on endpoint 2 (192.168.120.12).

The gateway 192.168.120.12 will return a SIP 415 Unsupported Media because it is not configured for SDES management.



```
Filter: sip && ip.addr == 192.168.120.11

No. -Tim Source          Destination     Protocol  Info
259 ! 192.168.120.30  192.168.120.11  SIP       Request: REGISTER sip:192.168.120.11:5062
286 ! 192.168.120.11  192.168.120.30  SIP       Status: 401 Unauthorized    (0 bindings)
301 ( 192.168.120.30  192.168.120.11  SIP       Request: REGISTER sip:192.168.120.11:5062
344 ( 192.168.120.11  192.168.120.30  SIP       Status: 200 OK    (2 bindings)
072 4 192.168.120.12  192.168.120.11  SIP       Request: REGISTER sip:192.168.120.11:5062
098 4 192.168.120.11  192.168.120.12  SIP       Status: 401 Unauthorized    (0 bindings)
112 4 192.168.120.12  192.168.120.11  SIP       Request: REGISTER sip:192.168.120.11:5062
165 4 192.168.120.11  192.168.120.12  SIP       Status: 200 OK    (2 bindings)
354 ! 192.168.120.30  192.168.120.11  SIP/SDP   Request: INVITE sip:101@192.168.120.11:5062, with session descript
399 ! 192.168.120.11  192.168.120.30  SIP       Status: 100 Giving a try
414 ! 192.168.120.11  192.168.120.12  SIP/SDP   Request: INVITE sip:101@192.168.120.12:16000;transport=tls, with se
562 ! 192.168.120.12  192.168.120.11  SIP       Status: 100 Trying
712 ! 192.168.120.12  192.168.120.11  SIP       Status: 415 Unsupported Media Type
738 ! 192.168.120.11  192.168.120.12  SIP       Request: ACK sip:101@192.168.120.12:16000;transport=tls
753 ! 192.168.120.11  192.168.120.30  SIP       Status: 415 Unsupported Media Type
768 ! 192.168.120.30  192.168.120.11  SIP       Request: ACK sip:101@192.168.120.11:5062

⊞ Frame 1712 (551 bytes on wire, 551 bytes captured)
⊞ Ethernet II, Src: 00:90:f8:02:ff:35 (00:90:f8:02:ff:35), Dst: 00:0c:29:d5:8d:78 (00:0c:29:d5:8d:78)
⊞ Internet Protocol, Src: 192.168.120.12 (192.168.120.12), Dst: 192.168.120.11 (192.168.120.11)
⊞ Transmission Control Protocol, Src Port: 16000 (16000), Dst Port: 5062 (5062), Seq: 1818, Ack: 2946, Len: 485
⊞ Secure Socket Layer
⊟ Session Initiation Protocol
  ⊞ Status-Line: SIP/2.0 415 Unsupported Media Type
  ⊟ Message Header
      Accept: /;application=sdp
    ⊞ Via: SIP/2.0/TLS 192.168.120.11:5062;branch=z9hG4bK78b1.3a907f44.0;i=0cc
    ⊞ Via: SIP/2.0/TLS 192.168.120.30:16000;branch=z9hG4bK1dbb0b5bb868f899c.802a1afdbb77fef2b
    ⊞ From: <sip:100@192.168.120.11:5062>;tag=eae64d7545
    ⊞ To: <sip:101@192.168.120.11:5062>;tag=3614255672
      Call-ID: b61966de2a75dec8
    ⊞ CSeq: 1836735416 INVITE
      Server: Mediatrix 4402 plus/v2.0.4.55 44XX-MX-D2000-36
      Content-Length: 0

File: "C:\Documents and Settings\magagne\Desktop\Security training\NOt - mismatch mike...   Packets: 2005 Displayed: 16 Marked: 0
```

The following Syslog message should also be seen:

```
syslog: SdpTools [D3A2] Received the wrong key management protocol. Secure stream disabled.
```

Annexes

Mediatrix Support Portal

http://www.media5corp.com/en/support-portal

Mediatrix Download Portal

http://www.media5corp.com/downloads

SSL and Certificates Information

http://www.openssl.org

http://en.wikipedia.org/wiki/X.509 (see links section)

Mikey Information

http://tools.ietf.org/html/rfc3830

SDES Information

http://tools.ietf.org/html/rfc4568

OpenSIPS Configuration Notes

tcp_conn.h:

```
#define TCP_CHILD_TIMEOUT pour 0 (avoid response delays)
#define DEFAULT_TCP_CONNECTION_LIFETIME pour 12000 (avoid connection drops after 2 minutes
of inactivity)
```

opensips.cfg:

```
disable_tls = no
listen = tls:192.168.120.11:5062
tls_verify_server = 0
tls_verify_client = 0
tls_require_client_certificate = 0
tls_method = TLSv1
tls_certificate = "/home/user/opensips/etc/opensips/cert.pem"
tls_private_key = "/home/user/opensips/etc/opensips/privkey.pem"
#tls_ca_list    = "/home/user/opensips/etc/opensips/tls/user/user-calist.pem"
```