Voice2Web security - Asterisk with TLS support

Technical report

Jan Staněk
Research & Development center(RDC) for Mobile Applications
April 2009

Introduction

Asterisk PBX used in Voice2Web project was configured to support secure TLS connections in the previous phase of V2W security project, but this set up was not properly tested because the only available hardware SIP phone snom360 was not able to connect to the Asterisk. Free softphones declaring that they have TLS support were also unable to connect to the Asterisk server. For this reason I decided to try another hardware phone and analyze the connection more deeply, to find out whether the failure is on the side of the Asterisk server or on the side of the snom phone. I had the opportunity to use Aastra57i as the other hardware phone.

Asterisk, snom360 and TLS

How to configure Asterisk PBX for TLS support was written in detail in the previous report so lets have a look at snom360 configuration. According to the manual, the phone just needs to upload the certificate and set transport mode to TLS and port to 5061. These steps are quite easy and straightforward. After I changed the configuration accordingly to these requirements and connected the phone, it freezed, totally. It stopped reacting to any input (both hardware buttons and web interface) and the only way was hardware restart.
I decided to analyze the traffic between snom and Asterisk, if there was any. I found out that there were a few packets for initialization of the TLS connection before the snom stopped responding again. I used the ssldump application to gain more information.

```
New TCP connection #1: r4al26.net.upc.cz(1975) <-> bolek.feld.cvut.cz(5061)
1 1  0.0097 (0.0097)  C>S  Handshake
      ClientHello
        Version 3.1
        cipher suites
        TLS_RSA_WITH_RC4_128_MD5
        TLS_RSA_WITH_RC4_128_SHA
        TLS_RSA_WITH_NULL_MD5
        TLS_RSA_WITH_NULL_SHA
        TLS_DH_anon_WITH_3DES_EDE_CBC_SHA
        TLS_DH_anon_WITH_RC4_128_MD5
        TLS_RSA_WITH_DES_CBC_SHA
        TLS_RSA_EXPORT1024_WITH_RC4_56_SHA
        TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA
        TLS_DH_anon_WITH_DES_CBC_SHA
        compression methods
                  NULL
1 2  0.0103 (0.0005)  S>C  Handshake
      ServerHello
        Version 3.1
        session_id[32]=
          b5 07 3a 0a 6d fa 66 da 92 6a b3 50 be e3 00 6a
          90 9f 8b e8 21 b3 8c 88 06 ac 96 ab 29 d4 e6 4d
```

```
        cipherSuite         TLS_RSA_WITH_RC4_128_MD5
        compressionMethod                   NULL
1 3  0.0103 (0.0000)  S>C  Handshake
      Certificate
1 4  0.0103 (0.0000)  S>C  Handshake
      ServerHelloDone
1    39.0951 (39.0847)  C>S  TCP RST
```

We can see that the phone started the communication with the server, they have done the first steps in the handshake, agreed on the cipher that will be used and when the server finished the initial sequence, snom stopped working.
I was not able to figure out the reason for this failure, but it sure looks like the problem is on the side of the snom phone. Old firmware might be the problem, but even though I tried, I was not able to upgrade it to a newer version than 4.5.

Asterisk, Aastra and TLS

The configuration of Aastra57i phone is a bit more complicated than the configuration of snom360, but if one has some time and the manual, it is not that painful. I will not waste time describing the configuration completely since I had borrowed the Aastra phone for the testing only and do not have it available anymore. The important fact is, that after the proper configuration, Aastra57i phone contacted the Asterisk server and created the connection using TLS without any problems.

Conclusion

Because the test with Aastra57i phone was successful, we proved that the Asterisk server used in the V2W project is configured properly and is able to handle communication with TLS support so we can now make calls with ciphered control SIP stream. The next step will be testing the SRTP so the calls will be ciphered completely - both control and data streams.