TR A6196

Securing Elcom-90 with TLS

Elcom WG
Convener: Ove Grande

May 2008

TECHNICAL REPORT

SINTEF Energy Research

Address: NO-7465 Trondheim, NORWAY
Reception: Sem Sælands vei 11
Telephone: +47 73 59 72 00
Telefax: +47 73 59 72 50

www.energy.sintef.no

Enterprise No.:
NO 939 350 675 MVA

SUBJECT/TASK (title)

Securing Elcom-90 with TLS

CONTRIBUTOR(S)

Tormod Lund, ABB AS

CLIENTS(S)

Statnett SF

TR NO.	DATE	CLIENT'S REF.	PROJECT NO.
TR A6196	2008-05-20	Anders Larsen	11X051

ELECTRONIC FILE CODE	RESPONSIBLE (NAME, SIGN.)	CLASSIFICATION
050926155953	Ove Grande	Unrestricted

ISBN N0.	REPORT TYPE	RESEARCH DIRECTOR (NAME, SIGN)	COPIES	PAGES
82-594-2907-1		Petter Støa	10	11

DIVISION	LOCATION	LOCAL FAX
Energy Systems	Sem Sælandsveg 11, Trondheim	+47 73597250

RESULT (summary)

This document is one of a series of technical reports which form the complete ELCOM-90 documentation. This is version .01 of the report. Future updates and new versions will NOT be published only to list of references. New versions will only be submitted when technical changes are made. Please see SINTEF's homepage at: http://www.sintef.no/ELCOM-90. From here you can download the latest version of all relevant documents as pdf-files for free.

This document describes how to secure Elcom traffic using Transport Layer Security, TLS. This is conceptually similar to accessing web pages using https, which is encapsulating the http protocol in TLS. With Elcom/TLS the Elcom protocol is encapsulated in TLS records, which provide for endpoint authentication, encryption and message integrity.

KEYWORDS

SELECTED BY AUTHOR(S)		
	ELCOM	Communication Protocol
	Security	TLS

TABLE OF CONTENTS

Page

1 INTRODUCTION

This document describes how to secure Elcom traffic using Transport Layer Security, TLS. This is conceptually similar to accessing web pages using https, which is encapsulating the http protocol in TLS. With Elcom/TLS the Elcom protocol is encapsulated in TLS records, which provide for endpoint authentication, encryption and message integrity.

2 REFERENCES

2.1 ELCOM-90 documentation

This document is one of a series of technical reports which form the complete ELCOM-90 documentation. Below you will find the numbers and titles for all the associated technical reports. New versions may be submitted when technical changes are made.
Please see SINTEF's homepage at: http://www.sintef.no//ELCOM-90. From here you can download the latest version of all relevant documents as pdf-files for free.

[1] TR 3701: **ELCOM-90 Application Programming Interface Specification**

[2] TR 3702: **ELCOM-90 Application Service Element. Service Definition**

[3] TR 3703: **ELCOM-90 Application Service Element. Protocol Specification**

[4] TR 3704: **ELCOM-90 Presentation Programming Interface Specification**

[5] TR 3705: **ELCOM-90 Presentation Service Definition**

[6] TR 3706: **ELCOM-90 Presentation Protocol Specification**

[7] TR 3825: **ELCOM-90 User Element Conventions**

[8] TR A3933: **ELCOM-90 Local Conventions**

[9] TR A4687: **PONG. The ELCOM net-watch procedure for TCP/IP networks**

[10] TR A4124: **ELCOM-90 Application Service Element, User's manual.**

[11] TR A6197: **Implementation of TLS security in the Elcom-90 reference version** (Internal document; not generally available).

2.2 Other documents

[12] STF90 A04075: **Secure Use of the ELCOM-90 Protocol**
SINTEF ICT Norway/Martin Gilje Jaatun, 2004-10-27 (Internal Document)

[13] **Employing TLS for the ELCOM-90 Protocol** (Draft Memo).
Martin Gilje Jaatun, 2004-10-31 (Internal Document)

[14] IEC Committee draft 57/754/CD: **Data and Communication Security – Part 3: Profiles including TCP/IP.**
IEC TC57 WG15. 2005-05-06.

[15] RFC 2246: **The TLS Protocol Version 1.0**
T. Dierks and C. Allen, January 1999.

[16] RFC 3268: **Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security (TLS)**
P. Chown, June 2002.

[17] RFC 3280: **Internet Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile**
R. Housley et. al. April 2002

2.3 Web Sites

[18] http://www.stunnel.org – Stunnel home page.

3 DEFINITIONS AND ABBREVIATIONS

For a longer list of Elcom-specific Definitions and Abbreviations, see [7]. The following are terms used in this document.

Initiator: In this document, a software system that initiates a communication link using the Elcom-90 protocol, i.e. the client, in client/server terminology.

Responder: In this document, a software system that accepts communication links using the Elcom-90 protocol, i.e. the server, in client/server terminology.

SSL: Secure Sockets Layer; the predecessor of TLS, originally specified by Netscape. The specification exists in versions 1-3, although version 1 is not in use.

TLS: Transport Layer Security. Public specification RFC2246 based on SSL V3.0. TLS V1.0 is often referred to as SSL V3.1.

PDU: Protocol Data Unit.

DES: Data Encryption Standard. This is a legacy block cipher encryption algorithm.

AES: Advanced Encryption Standard. This is a new block cipher encryption algorithm, also known as Rijndael.

CA: Certificate authority. In its simplest form, this is a piece of software that can be used to issue valid certificates for use e.g. by TLS. In a more general form, this also deals with the administrative issues of ensuring that certificates are issued to properly identified entities and distributed in a secure fashion. Commercial entities such as Verisign function as CAs in this sense.

4 OVERVIEW

As stated in the introduction, Elcom/TLS is conceptually simple. The Elcom protocol is run unchanged in a TLS 'tunnel'. This can be visualized with a network layer diagram.

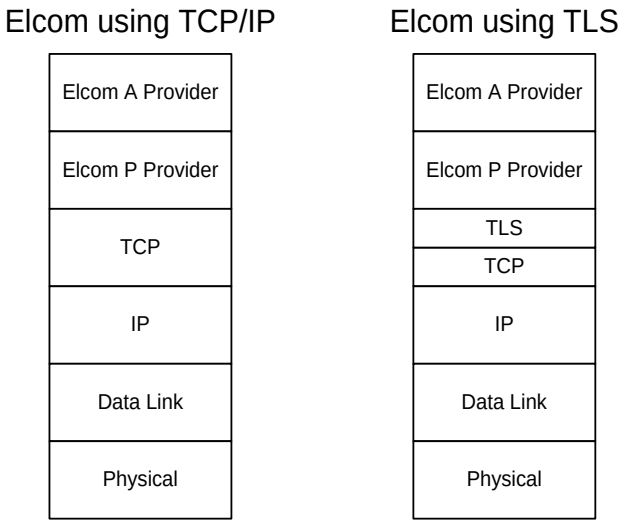

Figure 1 Elcom over TCP/IP and TLS

4.1 Protection Scope

Elcom/TLS as specified here provides protection against network attacks targeted at the Elcom protocol, specifically:

- Partner authentication protects against spoofed Elcom partners and man-in-the-middle attacks;
- Message integrity features protect against tampering and replay attacks;
- Encryption protects against data disclosure through eavesdropping.

Elcom/TLS **only** protects against network attacks, and should be employed as part of a defence in depth strategy, where other measures are used to provide sufficient protection and hence integrity for the host systems and the involved applications.

Further considerations are given in [12], together with some alternatives to Elcom/TLS.

Elcom/TLS as specified here secures Elcom over TCP/IP only. Conceptually, TLS may run over any connection-oriented data stream, and as such might be able to secure Elcom over X.25. This is not being considered, however, as the use of X.25 is not considered essential with today's Elcom installations.

4.2 Implementation Scenarios

Implementing Elcom/TLS can be achieved using different approaches:

- Using existing transparent tunnel software, such as stunnel [18]. This provides a low-cost option, with some authentication limitations (see the chapter on authentication below).
- Integrating TLS with the protocol provider software. This allows certificate authentication to be tied to the connect data in the Connect Request PDUs, and would facilitate security configuration at the user element level.
- Constructing a customized tunnel program. This would leave existing Elcom software unchanged, but still provide integrated authentication by decoding the Connect Request PDU.

This document proposes that the selected implementation approach is a local issue, and that different implementations should be interoperable, since the basic TLS and Elcom protocols remain the same. Although the different approaches give different levels of authentication, this affects the local system only, as long as certificates are properly deployed.

4.3 Coexistence with existing protocol

Elcom/TLS should coexist with Elcom over TCP/IP. To this end, different TCP/IP port numbers should be used for encrypted an unencrypted traffic, and a system should be able to deal with both simultaneously. It is suggested that it should be possible to disable unencrypted traffic if not needed, but this may be also achieved by a properly configured firewall.

5 USING ELCOM/TLS

5.1 Addressing

Elcom/TLS uses the same address format as Elcom over TCP/IP, as specified in [7]. As stipulated above, it is expected that a distinct TCP/IP port number is used for receiving TLS traffic.

5.2 Connection Establishment

With Elcom/TLS, the Elcom Initiator is also the TLS client, and will perform the initial connection and send out the TLS ClientHello to begin the TLS handshake. Upon completion of the handshake, Elcom PDUs should be sent as TLS application data, starting with the Connect Request PDU.

5.3 Connection Closure

TLS provides closure alerts to facilitate secure connection closure. All implementations must send a closure alert to prior to closing the connection, and must respond to incoming closure alerts. An implementation may close the connection after sending a closure alert, without waiting for a reply. If a connection is closed without the reception of a closure alert, that session should not be resumed (see session caching below).

5.4 Session Caching

Elcom/TLS implementations may optionally support the TLS Session Caching facility to speed up the connection handshake process when there are multiple or repeated connections to the same remote partner. If used, is recommended that the session cache is set to expire at a configurable interval, typically around 24 hours.

5.5 Key Renegotiation

TLS provides a facility with which the encryption parameters, such as the session key, may be renegotiated without restarting a connection. For Elcom/TLS, the ability to initiate a renegotiation is optional; the ability to respond to it is not.

Renegotiation should be triggered by the connection being established for a certain amount of time and/or the transmittal of a certain amount of data, with these amounts being configurable. If the other part of the link initiates a renegotiation, the trigger conditions for the local implementation should be reset as if the renegotiation was started from here.

5.6 Protocol Constraints

5.6.1 Protocol Versions

TLS is based on the Netscape SSL protocol, which is commonly used in versions 2 and 3. TLS V1.0 is often referred to as SSL V 3.1. The protocol version used is determined in the handshake process. Due to known security issues, Elcom/TLS should not use protocol versions prior to SSL V3.0, i.e. SSL V3.0 and TLS V1.0 (SSL V3.1) should be supported.

5.6.2 Cipher Strength

TLS supports several different algorithms for key exchange, block encryption and message hashing, collectively known as cipher suites. Also, this support is in an open-ended fashion, with additional cipher suites specified in e.g. RFC 3268 [16].

For Elcom/TLS the following cipher suites should not be used, and explicitly disallowed by the responder:

- Any cipher suite with a key exchange algorithm of NULL or DH_anon (meaning no authentication).
- Any cipher suite with a block cipher of NULL, meaning no encryption.

It is also strongly recommended that ‘weak’ encryption algorithms are avoided, in particular the ‘exportable’ cipher suites, which use a 40-bit key for the block ciphers. Also avoid RSA keys with a length < 1024 for certificates. Use of standard DES (DES_CBC) is also discouraged.

A recommended default algorithm can be something like TLS_DHE_RSA_WITH_AES_128_CBC_SHA, but note that the key exchange algorithm selected (here DHE_RSA) will depend on the certificate used (see [15]).

6 PARTNER AUTHENTICATION

Elcom/TLS implementations should use certificates for authentication in both directions, i.e.:

- Anonymous key exchange shall be explicitly disallowed by both initiator and responder
- The responder shall always issue a certificate request to the initiator

6.1 Certificates and Certificate Authorities

Elcom/TLS uses certificates for partner authentication and key exchange. These should be issued by a trusted Certificate Authority (CA). We can distinguish between two types of CAs:

- Private CA, i.e. one (or more) of the communicating parties act as a CA, and that this CA issues only certificates for Elcom communication within a certain area.
- Public CA, i.e. certificates are bought from a third party such as Verisign, which issues certificates for varied purposes.

An Elcom/TLS implementation must support certificates from more than one CA at a time. Support for chained certificates, where a partner certificate is not signed by a root CA, but by an intermediate, is optional for Elcom/TLS, and should be verified between communication partners before applied.

6.2 Levels of Authentication

Depending on the implementation used, and the needs in a particular scenario, three levels of authentication may be used with Elcom/TLS. These are accumulative.

1. All certificates are validated against the configured CAs for an installation, as well as validated with respect to expiration date. If a private CA is used, this may be sufficient authentication, as it proves the partner to be a trusted Elcom partner.
2. Certificates are further verified as being within a list of specific certificates. This is useful if using certificates from a public CA, and implementing TLS with stunnel or similar software.
3. Certificates are verified against configured certificates for specific partners, based on the content of the Connect Request PDU. This requires a TLS implementation that understands Elcom Connect Request PDUs.

6.3 Certificate Revocation

An Elcom/TLS implementation must support the use of certificate revocation lists according to RFC 3280 [17], to allow a CA to revoke certificates that should no longer be valid. Retrieval and installation of these is a local issue, and it is permissible to force a communication restart to activate these.

6.4 Certificate Expiration

An Elcom/TLS implementation should check for expired certificates during initial handshake and key renegotiation.

7 LOGGING AND ERROR HANDLING

7.1 Logging

An Elcom/TLS implementation should have a logging facility that support persistent storage of security-related events in a protected log file, ie. The log events should not be lost at restart, and the file should be protected from unauthorized modification (and possibly inspection).

7.2 Error Handling

Depending on the implementation method, specific Elcom Result codes may be returned from ACONC:

Implementation	Type of Error	Result code	Comment
Tunnel	Any TLS error	30	As the tunnel is transparent, no specific information can be provided to the user element.
Integrated	Certificate Rejected By Responder	20	The initiator certificate was rejected by the remote responder. The certificate may still be valid, but the canonical name does not match what the responder expected for this initiator.
	Responder Certificate Mismatch	21	The responder certificate does not match what is configured for the responder in question, although the certificate may still be valid.
	TLS Unavailable	22	The local system is unable to handle TLS communications, e.g. due to a configuration error.
	TLS Error	23	Some TLS related error prevented this connection from succeeding.

Note: Of these errors, only error 20, Certificate Rejected, is communicated between systems; the others are generated locally in one system, and are thus not normative.