CryptoTimeStamp:

TSP Time Stamp Server Based upon the CryptoServer 2000

Technical Whitepaper

Document-Number:	
Document-Version:	1.00.00
State from:	26.2.2003
Release state:	released
Author (s):	Dipl.- Inf. Rainer Herbertz
Customer:	

Office Aachen
Germanusstraße 4
D-52080 Aachen

Telephone: ++49 / 241 / 1696 200

Telex: ++49 / 241 / 1696 222

Internet: www.utimaco.de

E-Mail: info.sh@aachen.utimaco.de

Table of Contents

1 Introduction

The CryptoTimeStamp is a TSP time-stamp server compliant to RFC 3161. The overall–system of the CryptoTimeStamp consists of the components communication processor and Utimaco's hardware security module CryptoServer 2000. The timer and the signature function are implemented within the sealed and sensorically protected security module. The signature key is internally generated within the security module and certified by an external PKI afterwards.

For the system administration a command line tool exists, which can be executed upon any Windows or Linux PCs and which gains access via TCP/IP to the CryptoTimeStamp. All administrative commands must be signed by one or two security officers (SOs). For that purpose, the security officers use a signature chip cards, which are addressed by the command line tool. This is done by a PIN-PAD, linked to a PC.

The time setting is done manual via an administrative function. Optionally, an automatic adjustment occurs by the Network Time Protocol via an external NTP server.

By the communication processor, a TCP/IP interface is provided as external host interface, by which time-stamp requests can be sent to the time-stamp server according to RFC 3161, and time-stamps can be received. In addition, a software library with C function can be used for calculating hash values and verifying time-stamps.

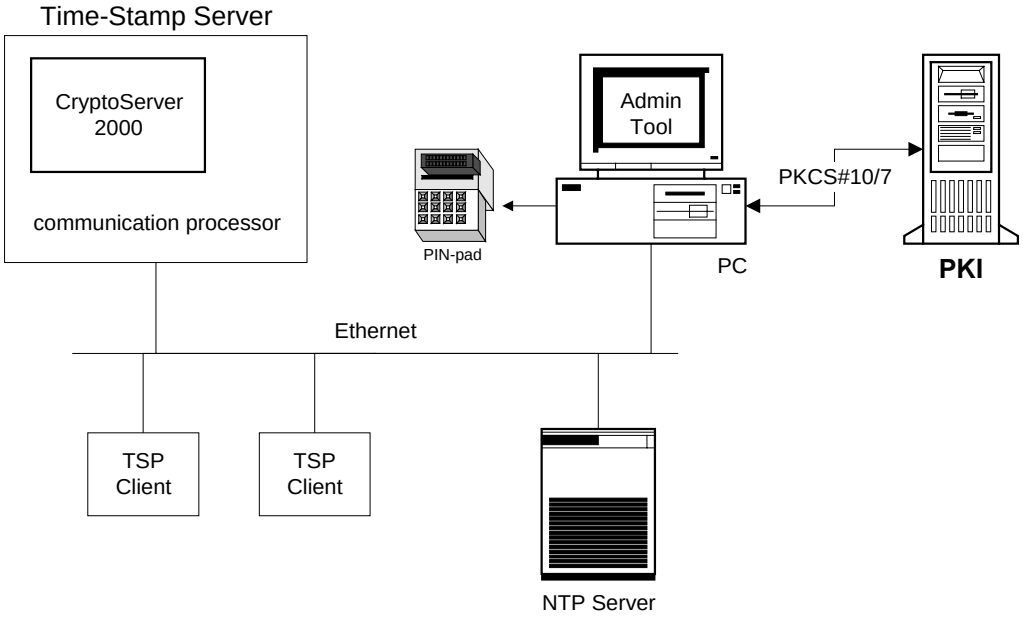

Figure 1.1: Physical units (devices) in a CryptoTimeStamp system

2 Time-stamp Generation

The time-stamp server CryptoTimeStamp can be addressed via TCP/IP by a client. It receives time-stamp requests on port 318 according to RFC 3161 (Time-Stamp Protocol, TSP). Thereby the socket based protocol is used. The time-stamp server always responds either with a complete time-stamp or an „Error Response“. A „Polling Response“ or a „Partial message Response” is never dispensed. If demanded in the time-stamp request, the respond contains the certificate of the time-stamp key.

The time-stamp server can contain various signature keys. Maximal one key is marked as „active“. This key is used for generating the time-stamp. If no active key is available, an error code is returned.

2.1 Verification

A verification of the time-stamp via CryptoTimeStamp is not designated. The verification must be accomplished externally by suitable software.

3 Administration of the CryptoTimeStamp

The „CryptoServer LAN 2000 Operating Manual“ describes in depth the administration of the communicating computer (start-up and shut down, network address configuration etc.), so that this will therefore not be outlined in this document. This section outlines just the administration tasks specific for the time stamp functionality inside the hardware security module.

The administration of the time-stamp server is provided by an authorized user. It is important to distinguish between two user groups:

User group	Task
Security Officers „Hardware Security Module“ (HSM-SO)	Administration of the security module (e.g. Firmware Update), authorization awarding to TS-SO’s.
Security Officers „time-stamp server“ (TS-SO)	Signature key generation / activation, import of certificates, time setting, etc.

For system administration the Security Officers make use of a command line tool running on a Windows or Linux PC and having access to the time-stamp server over TCP/IP. All administration commands are to be signed by one or two Security Officers. The Security Officers hold therefore signature smart cards activated by the command line tool over a PIN pad connected to the PC. HSM-SO decides in authorization awarding to TS-SO’s on the necessity of having one or two TS-SO’s signatures (two persons rule) for the execution of time-stamp server’s administration commands.

3.1 Initialization of the CryptoTimeStamp

The CryptoTimeStamp security module contains primarily merely the firmware and a public key of a HSM-SO (delivery status). For HSM-SO there is also a smart card as delivery condition. Following steps are therefore necessary for installation of the time-stamp server:

1. Smart cards certification for one or several TS-SO’s.
2. TS-SO’s registration into the security module by using HSM-SO.
3. Internal time setting by TS-SO.
4. Policy Setting.
5. Generation of a signature key and a certification request according to PKCS#10 by TS-SO.
6. Certification of the key by an external PKI.
7. Import of PKI´s CA certificate into the security module of the time-stamp server by TS-SO.
8. Import of the certificate for the signature key into the security module of the time-stamp server by TS-SO.
9. Activation of the signature key by TS-SO.

The time-stamp server is subsequently operational.

3.2 Administrative Functions of the Commando Line Tool

The command line tool provide following functions for the time-stamp server administration:

- Generation of a signature key.

  TS-SO can indicate the key size (currently max. 2048 Bit). The generated key is stored in the security module and receives the “inactive“ status.
- Generation of the certification request according to PKCS#10.

  TS-SO can choose a „Distinguished Name“ for the certificate. The request is written in a file.
- Import of a CA certificate.

  The certificate has to be available as a X509v3 format and binary file. The input and verification of a Hash Value is additionally necessary in order to guarantee the certification integrity.
- Import of a certificate to a signature key.

  The certificate has to be available either as a X509v3 or PKCS#7 format as a binary file. The import is only allowed when the security module contains an appropriate signature key. The certificate is to be verified using the CA certificate. If the verification fails, the import is denied.
- Activation of the signature key.

  The activation of a signature key is only possible when a certificate is available to this key. The key receive the “active“ status, all the others stored in the same security module the „inactive“ status.
- Internal time setting.

  Manual time resetting (by TS-SO).
- Policy Setting.

  An object identifier that defines the timestamp policy of the provider is configured into the Timestamp server. This object identifier is included in every Timestamp response.
- Display of all keys and certificates stored in the security module.

  The state of the keys and certificates is displayed, too.
- Deletion of the keys and certificates stored in the security module.

  Once a signature key is deleted, the appertaining certificate (if available) is deleted as well. Once a certificate is deleted, the status of the appertaining key is set to „inactive“.
- Output of the administration logfiles.

  The Logfile (see chapter 5) is saved in a separate file
- Export of a certificate to a signature key.

  The certificate is binary filed as a X509v3 format.

Depending on the configuration of the time-stamp server, further functions are available for the signature keys back up. Please see section 4.2 for a detailed description .

- Backup of a signature key pair.

  The encrypted signature key is exported from the security module and stored within a file. A private data format of the security module is used for that purpose.
- Restore of a signature key pair.

  The signature key is read from the file and imported into the security module. The file must be backup-derived.

3.3 Required Equipment

To use the administration tool a special PIN pad that is delivered together with the time-stamp server must be connected to the PC via the serial line.

It is not possible to use arbitrary smart cards for security officer authentication. The smartcards must be obtained by Utimaco or generated with Utimaco's PKI.

4 Keys and Certificate Database

The security module of the CryptoTimeStamp contains a database, storing following data:

- Signature keys.

  It is possible to store several hundred signature keys with different key length. Maximum one of these keys is “active”, the others are “inactive”. The active key serves generating time stamps. The signature key is generated in the security module and are normally impossible to export (exception: see section 4.2).

- Certificates.

  The corresponding certificate is stored for each key. The certificates are generated by an external PKI and imported by an administration tool.

- One CA Certificate.

  PKI´s CA Certificate is imported with administration tool. It serves for the verification of signature keys certificates.

- HSM-SO’s Public Key.

  This key serves to HSM-SO authentication and imported while manufacturing the security module. It can be changed later with CryptoServer 2000 administration tool.

- TS-SO’s Public Key.

  This key serves TS-SO‘s authentication and is brought in by HSM-SO.

- Backup Key.

  It concerns a 24 byte DES key for encrypting signature keys for backup (see section 4.2).

4.1 Change of the Signature Key

The capacity of the time-stamp server to memorize several signature keys could serve for an online changing procedure. For that purpose an new signature key can be anytime generated, which is initially marked as „inactive“. After the key certification and certificate import, it can be anytime later switched over to the new key by activating the new key during operating time.

4.2 Recovery

In the standard version of the CryptoTimeStamp no signature key export is provided. If the content of the internal data bank is deleted (e.g. caused by an exploit on the security module or by hardware failure), the signature key are irrecoverable. In that case the time-stamp server has to be re-initialized. If so, a new signature key is to generate and certify. The old key has to be disabled based on the PKI (admission into CRL).

A signature key backup can be generated through an additional function in the time-stamp server, taking place as follows:

The time-stamp server generates an internal 24 bytes DES backup key. This key is decomposed into two XOR halves and written on two special smart cards, which are further assigned to two Security Officers. The writing of the key parts to the smart cards takes place over a PIN pad connected directly to a security module (so that nobody can spy on them either

within the network or on PC). The smart cards are secured with a PIN code. The signature keys are subsequently to export from the time-stamp server in an encrypted form and to secure.

In the case of re-initialization, the two halves of the backup key can be imported from the smart cards into the security module, which is again taking place within a PIN pad connected to the security module. TS-SO's can subsequently import the secured signature keys over the command line administration tool into the time-stamp server. Upon importing all required certificates, the time-stamp server is operational again.

The same method applies in the case of signature key distribution on several time-stamp servers, whereas all of them are working with the same key. The operation of several time-stamp servers makes particularly sense in order to achieve safeguarding against failure or to increase performance at high flow rate.

The backup function provided for signature keys is designed just for the key recover in a time-stamp server, not for an external use of the key (no standard format).

5 Logging

The time-stamp server runs an administration-logfile within the security module, in which all administrative operations according to paragraph 3.2 are stored combined with time and date. The logfile has a fixed size and is recorded rotary. That means, if the logfile is full, the eldest entry at a time will be deleted for clearing space for a new entry. The logfile can be exported via the command line tool from the security module and should be saved constantly.

6 The Internal Clock of the CryptoTimeStamp

The clock which the CryptoTimeStamp uses for generating the time-stamp, is located within the security module and is so protected against manipulation by sensors. For clock adjustment there exist two mechanisms within the time-stamp server:

For balancing inexactness of the internal clock, the time-stamp server is able to process an automatic time correction. Therefor, the time-stamp server features the functionality of an NTP client. Via configuration, a network address of an NTP server can be set, that can provide the current time for the time-stamp server. In doing so the clock can be put forward and back for three seconds a day to the maximum. This is sufficient for balancing possible inexactnesses of the internal clock. Via this interface it is not possible to adjust the clock for more than three seconds, for preventing manipulation (e.g. affection of a false NTP server) by an attacker.

Besides the clock can be adjusted manually by the TS-SO's at any time, using the command line tool. In doing so there is no limitation referring to the adjustment of time. This method is to be used especially when initiating the time-stamp server. Therefore the TS-SO's are responsible for the correct time within the time-stamp server.

The internal clock always contains UTC time (Universal Time Coordinate). Also the time-stamps contain UTC time. This is to be regarded when adjusting the clock.

For excluding a hardware failure of the internal clock, resp. realizing it at an early stage, it is recommended to implement a monitoring tool upon a host, which prompts the time-stamp server-time in regular intervals and proves the plausibility (e. g. by comparison with the host's time).