T/3502TL/2778/9

27 January 1997

Copy No.: ____

CLOUD COVER
CONFIDENTIALITY KEY INFRASTRUCTURE
PART 5: MAPPING OF THE CKI KEY MANAGEMENT PROTOCOL ONTO COMMUNICATION AND MESSAGING PROTOCOLS

ISSUE 0.A

This document and its content **shall** only be used for the purpose for which it was issued. The copyright of this document is reserved and is vested in the Crown

©1997 Crown Copyright.

FOREWORD

This paper is issued by the Communications-Electronics Security Group (CESG) of Government Communications Headquarters as part of its responsibility to advise HMG on Electronic Information Systems Security (Infosec).

It suggests an architecture for a public key infrastructure (PKI) to support confidentiality between communicating systems. The paper forms part of a suite of documents which collectively provide advice on the implementation of a PKI, and the use of the services enabled by such an infrastructure (eg electronic mail). The architecture as described in the paper is an initial attempt at defining a PKI, and CESG will take into account any comments on its feasibility.

Technical correspondence in connection with this document should be addressed to:

Communications-Electronics Security Group (X27)
Government Communications Headquarters
PO Box 144
Cheltenham GL52 5UE
United Kingdom

AMENDMENT RECORD		
Issue	**Date**	**Description**
Initial draft	6 December 1996	
0.A	27 January 1997	Reformat & release for internal review

CONTENTS

FOREWORD ii
CONTENTS iv
REFERENCES v
DEFINITIONS vii

I. INTRODUCTION 1

II. GENERAL 2

III. MAPPING TO ELECTRONIC MAIL / MESSAGING 3
- **A. General** 3
- **B. Internet Electronic Mail** 3
- **C. X.400 Messages** 3

IV. MAPPING TO PEER TO PEER COMMUNICATION SERVICES 4
- **A. General** 4
- **B. Socket Based Management Protocol** 4
- **C. OSI Connection Oriented Transport Service (COTS)** 6

V. MAPPING TO HTTP 7
- **A. General** 7
- **B. MIME Content-Type for PKI** 7
- **C. Procedures for use of HTTP** 7

REFERENCES

[HMG] Securing Electronic Mail within HMG - Part I: Infrastructure and Protocol, Draft C, T/3113TL/2776/11 21st March 1996

[DH76] New Directions in Cryptography, IEEE Trans. In Information Theory IT-22 (1976) pages 644-655 W. Diffie and M. Hellman

[RHC] A proposed Architecture for Trusted Third Party Services, N. Jefferies, C. Mitchell, M. Walker, Information Security Group, Royal Holloway

[PKI-1] Internet Public Key Infrastructure Part I: X.509 Certificate and CRL Profile, June 1996, Internet Draft

[PKI-3] Internet Public Key Infrastructure Part III: Certificate Management Protocols, November 1996, Internet Draft

[RFC 793] "Transmission Control Protocol", J. Postel, 09/01/1981

[RFC 822] "Standard for the format of ARPA Internet text messages", D. Crocker, 08/13/1982

[RFC 1521] "MIME (Multipurpose Internet Mail Extensions) Part One: Mechanisms for Specifying and Describing the Format of Internet Message Bodies", N. Borenstein, N. Freed, 09/23/1993

[RFC 1945] "Hypertext Transfer Protocol -- HTTP/1.0" T. Berners-Lee, R. Fielding, H. Frystyk May 1996 (Information)

[X.214] ITU-T X.214 (95) | ISO/IEC 8072:1996 Information technology - Open systems interconnection - Transport service definition

[X.420] ITU-T X.420 (to be published) | ISO 10021-7 Information technology - Message Handling Systems (MHS) - Interpersonal Messaging System

Note: this is equivalent to X.420 (92) plus implementor's guide version 8.

[X.500] ITU-T Recommendation X.500 to X.525 (1993) | ISO/IEC 9594:1994, Information technology – Open Systems Interconnection – The Directory

[X.509DAM] Final Text of Draft Amendments DAM 4 to ISO/IEC 9594-2, DAM 2 to ISO/IEC 9594-6, DAM 1 to ISO/IEC 9594-7, and DAM 1 to ISO/IEC 9594-8 on Certificate Extensions ISO/IEC JTC 1/SC 21/WG 4 and ITU-T Q15/7 Collaborative Editing Meeting on the Directory, Geneva, April 1996 - Final draft 30th June 1996

[X.509TC] Technical Corrigenda to Rec. X.500 | ISO/IEC 9594 resulting from Defect Reports 9594/128

[X.509] ITU-T X.509 (93) | ISO/IEC 9594-8: 1995 Information Technology – Open Systems Interconnection – The Directory: Authentication Framework

[X.511] ITU-T X.511 (93) | ISO/IEC 9594-3: 1995 Information Technology – Open Systems Interconnection – The Directory: Abstract Service Definition

[X.690] ITU-T X.690 (94) | ISO/IEC 8825-1:1995 Information Technology Information technology- ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)

DEFINITIONS

The following terms and associated concepts are described in the CKI Architecture and Concept of Operation (Part 1):

a. Certificate Management Authority (CMA)

b. CKI certificate

c. CKI User Agent (CKI UA)

d. CMA certificate

e. Domain

f. Domain certificate

g. Domain public / private key

h. External

i. Interoperability key

j. Local

k. Name constraints

l. Receive certificate

m. Receive public / private key

n. Recipient

o. Revoked user list

p. Seed key

q. Seed key identifier

r. Send certificate

s. Send public / private key

t. Sender

u. Shared secret key

v. Top Level Certificate Management Authority (TLCMA)

ABBREVIATIONS

AKI	Authentication Key Infrastructure
CA	Certification Authority
CKI	Confidentiality Key Infrastructure
CMA	Certificate Management Authority
CRL	Certificate Revocation List
PKI	Public Key Infrastructure
TLCMA	Top Level Certificate Management Authority
UA	User Agent

I. INTRODUCTION

1. This specification defines mappings of the Confidentiality Key Infrastructure (CKI) key management protocols (Part 2) onto underlying communications and messaging protocols.

2. The CKI uses asymmetric cryptographic techniques in the generation of a shared symmetric key for confidentiality.

3. This specification is Part 5 of a set of specifications for the CKI, which includes:

Part 1: Architecture and concept of operation for the CKI;

Part 2: CKI key management protocol;

Part 3: Profile for the use of X.509 certificates in support of the CKI;

Part 4: Schema for the use of an X.500 directory in support of the CKI;

Part 5: Mapping of the CKI key management protocol onto communication and messaging protocols.

4. The use of X.500 directories is an optional part of the CKI.

5. The CKI is based on the Diffie-Hellman key agreement mechanism [DH76] with support of trusted third party services [RHC].

6. The CKI was initially developed to support secure electronic mail within and between UK government departments [HMG]. However, it is designed to be applicable to a range of application and communication services, and can be used to support confidentiality for governmental, commercial or any other type of organisation.

7. The CKI supports the management of confidentiality keys. It forms part of a public key infrastructure which can also incorporate an infrastructure for the management of authentication keys (called the Authentication Key Infrastructure - AKI). The AKI can be used to provide certified keys for signing CKI certificates and protecting protocol exchanges required for the CKI.

8. The design of the CKI takes account of the ongoing development of standards for public key infrastructures as they exist at the time this specification was developed (e.g. Internet PKI as defined in [PKI-1] and [PKI-3]).

II. GENERAL

9. The communication and messaging protocols identified in this document are used to transport the **PKIMessage** syntax, as defined in Part 2, between CKI entities (CKI user agents, certificate management authorities, top level certificate management authority).

10. These protocols can also be used in support of the Internet Public Key Infrastructure (PKI) certificate management protocols [PKI-3].

11. When the Internet PKI is issued as an Internet RFC with status proposed standard, the Internet PKI protocol mappings may be used as an alternative to those described in this document.

12. The **PKIMessage** syntax is encoded using the Distinguished Encoding Rules as defined in [X.690].

III. MAPPING TO ELECTRONIC MAIL / MESSAGING

A. General

13. **`PKIMessage`** is transported as a binary file attached to an electronic mail / message.

14. The file name PKI.MSG may be used as the file name transferred in the message.

> **Note:** use of this file name in transfer does not preclude an alternative name being used locally to save PKI messages. Alternatively, a messaging user agent may directly process the PKI message without ever saving it to a file.

15. The electronic mail / messaging service shall provide the security services as required in Part 2 (authentication, integrity and confidentiality) using, for example, the security mechanisms described in [HMG].

B. Internet Electronic Mail

16. Files are transferred in Internet [RFC 822] electronic mail as an **`application/octet-stream`** MIME content type [RFC 1521].

C. X.400 Messages

17. Files are transferred in a X.400 message File Transfer body part as defined in [X.420].

IV. MAPPING TO PEER TO PEER COMMUNICATION SERVICES

A. General

18. This protocol is designed to run over connection-oriented, reliable transports, with all 8 bits in an octet being significant in the data stream.

19. On establishment of the underlying connection a secure bind operation is exchanged between the peer entities to provide peer entity authentication as described in Part 2.

20. Each protocol exchange is linked to the bind using a nonce which is derived from the bind authentication exchange as described in Part 2.

21. The underlying network or transport protocol shall provide:

a. connectionless integrity or connection integrity without recovery

b. connectionless confidentiality or connection confidentiality

Note: When combined with the secure bind, nonce and underlying security service this provides the security services required in Part 2.

22. Separate connections should be used to carry:

a. request / response exchanges (from CKI UA to CMA, CMA to TLCMA and CMA to CMA), and

b. announcements (from CMA to CKI UA and TLCMA to CMA).

B. Socket Based Management Protocol

23. This protocol is as defined for the Internet PKI [PKI-3]. It supports operation over any socket based implementation of a transport protocol including a TCP [RFC 793] bytestream.

24. The following “simple” socket based protocol is to be used for transport of PKI messages. This protocol is suitable for cases where an end entity (sender or recipient) or a CMA initiates a transaction and can poll to pick up the results.

25. If a transaction is initiated by a CMA then an end entity must either supply a listener process or be supplied with a polling reference (see below) in order to allow it to pick up the PKI message from the PKI management component.

26. The protocol basically assumes a listener process (on an CMA) which can accept PKI messages on a well defined port (port number to be specified - see note below). Typically an initiator binds to this port and submits the initial PKI message for a given transaction ID. The responder replies with a PKI message and/or with a reference number to be used later when polling for the actual PKI message response.

Note: When a port number is allocated to the Internet PKI [PKI-3] this should be used for the CKI. In the interim it is recommended that a port number greater than 5000 is used (1-1023 are for Internet managed

port numbers, 1024-5000 are used for clients). Also, port numbers for other protocols registered in RFC 1700 should be avoided.

27. If a number of PKI response messages are to be produced for a given request (say if some part of the request is handled more quickly than another) then a new polling reference is also returned.

28. When the final PKI response message has been picked up by the initiator then no new polling reference is supplied.

29. The initiator of a transaction sends a "socket PKI message" to the recipient. The recipient responds with a similar message.

30. A "socket PKI message" consists of: length (32-bits), flag (8-bits), value (defined in table1)

31. The length field contains the number of octets of the remainder of the message (i.e. number of octets of "value" plus one).

32. The flag and value fields are described in table 1.

Message name	flag	value	comment
msgReq	'00'H	DER-encoded PKI message	PKI message from initiator
pollRep	'01'H	polling reference (32-bits)	poll response where no PKI message response ready; use polling reference value for later polling
pollReq	'02'H	polling reference (32 bits)	request for a PKI message response to initial message
negPollRep	'03'H	`00'H	no further polling responses (i.e., transaction complete)
partialMsgRep	'04'H	next polling reference (32-bits), DER encoded PKI message	partial response to initial message plus new polling reference to use to get next part of response
finalMsgRep	'05'H	DER encoded PKI message	final (and possibly sole) response to initial message
errorMsgRep	'06'H	human readable error message	produced when an error is detected (e.g., a polling reference is received which doesn't exist or is finished with)

Table 1. Management Protocol Field Definition.

33. Where a **`PKIConfirm`** message is to be transported (always from the initiator to the responder) then a msgReq message is sent and a negPollRep is returned.

34. The sequence of messages which can occur is then:

    a. end entity sends msgReq and receives one of pollRep, negPollRep, partialMsgRep or finalMsgRep in response.

    b. end entity sends pollReq message and receives one of negPollRep, partialMsgRep, finalMsgRep or ErrorMsgRep in response.

C. OSI Connection Oriented Transport Service (COTS)

35. The OSI Transport Service [X.214] is used. No special use of T-Connect is made. Each **`PKIMessage`** PDU is mapped directly onto T-Data.

V. MAPPING TO HTTP

A. General

36. This mapping only supports the transport of request / response exchanges of PKI Messages. One way announce exchanges are not supported.

37. The PKI Message is carried as a MIME [RFC 1521] content type using the Hyper Text Transfer Protocol (HTTP) as defined in [RFC 1945].

B. MIME Content-Type for PKI

38. Pending the definition of a Mime content-type for PKI Messages in [PKI-3] the following content type may be used. It should be noted that this content type has not been registered.

39. MIME media type name: application

40. MIME subtype name: PKI

41. There are no required or optional MIME parameters.

C. Procedures for use of HTTP

42. The client posts a request to the server in the HTTP protocol. The POST includes an Entity-Body of the application/PKI content with the request.

43. The server replies to this request with status 200 (OK) and includes another application/PKI content with the response.