T/3498TL/2778/9

27 January 1997

Copy No.: ____

CLOUD COVER
CONFIDENTIALITY KEY INFRASTRUCTURE
PART 1: ARCHITECTURE & CONCEPT OF OPERATION

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 vi

I. INTRODUCTION 1

II. MODEL 2
- **A. User Environment** 2
- **B. Domains** 3
- **C. CKI Components** 3
- **D. Supporting Functions** 4

III. CKI CRYPTOGRAPHIC FUNCTIONS AND ASSOCIATED KEYS 5
- **A. Send and Receive Public / Private Keys** 5
- **B. Seed Keys** 6
- **C. Interoperability Keys** 7

IV. DISTRIBUTION OF KEYS AND ASSOCIATED PARAMETERS 10
- **A. Use of X.509 Certificates** 10
- **B. Distribution of Certificates** 11
- **C. Distribution of Seed Keys** 11
- **D. Distribution of Interoperability Keys** 12
- **E. Key Recovery** 12
- **F. Protection of Key Distribution** 12
- **G. Start up Conditions** 13

V. REVOCATION 15
- **A. Revocation of Send & Receive Certificates** 15
- **B. Blacklisted Users and External Seed Key Revocation** 15
- **C. Revocation of CMA Keys and CMA Compromise** 16

VI. IDENTIFICATION 17
- **A. Key identifiers** 17
- **B. Naming** 17
- **C. Domain Name Constraints** 18

REFERENCES

[HMG] Securing Electronic Mail within HMG - Part I: Infrastructure and Protocol, Draft C, T/3113TL/2776/11 21 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 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 793] "Transmission Control Protocol", J. Postel, 09/01/1981

[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

CKI Architecture Definitions

The following terms and associated concepts are described in the body of this specification. Each term is described in the given section:

a. Certificate Management Authority (CMA) (§II.C)

b. CKI certificate (§IV.A)

c. CKI User Agent (CKI UA) (§II.C)

d. CMA certificate (§IV.A)

e. Domain (§II.B)

f. Domain certificate (§IV.A)

g. Domain public / private key (§III.C)

h. External (§II.B)

i. Interoperability key (§III.C)

j. Local (§II.B)

k. Name constraints (§VI.C)

l. Receive certificate (§IV.A)

m. Receive public / private key (§III.A)

n. Recipient (§II.A)

o. Revoked user list (§V.B)

p. Seed key (§III.B)

q. Seed key identifier (§VI.A)

r. Send certificate (§IV.A)

s. Send public / private key (§III.A)

t. Sender (§II.A)

u. Shared secret key (§II.A)

v. Top Level Certificate Management Authority (TLCMA) (§II.C)

X.509 Authentication Framework Definitions

The following terms are defined in the X.509 Authentication Framework [X.509]:

a. Certification Authority (CA)

b. Certificate

c. CA Certificate

d. Certificate Revocation List (CRL)

X.500 Directory Definitions

The following term is defined in the Directory standard [X.500]:

a. Distinguished name

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 document specifies the architecture and concept of operation for a Confidential Key Infrastructure (CKI).

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

3. This specification is part 1 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 (eg Internet PKI as defined in [PKI-1] and [PKI-3]).

II. MODEL

A. User Environment

9. The environment of the user of the CKI is illustrated in figure 1:

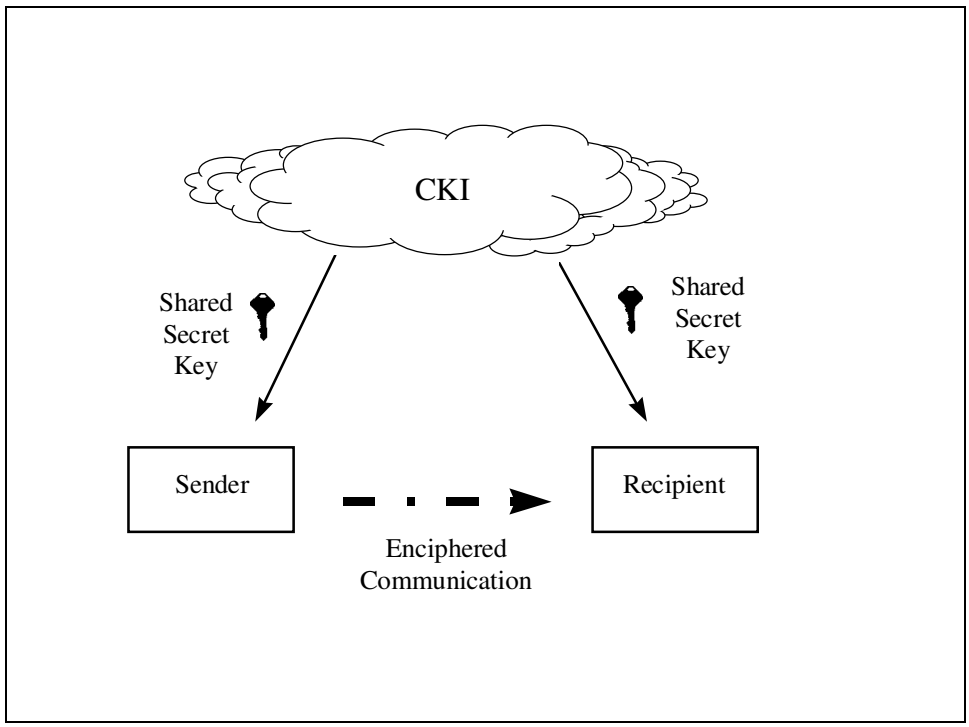

Figure 1 CKI User Environment

10. The CKI provides a *shared secret key* to two communicating entities a *sender* and *recipient*.

11. The *shared secret key* is for encipherment of data between the *sender* and *recipient* to provide confidentiality, either directly or indirectly (see note below). A separate key should be used for the encipherment of communications in the opposite direction.

> **Note:** When the CKI is used to support Secure Electronic Mail within HMG [HMG] the shared secret key is used as a token key which enciphers a data (enciphering) key which in turn enciphers the data being communicated.

12. The CKI places no restrictions on the form of the communications between the *sender* and *recipient*. It can be used to protect both store and forward messaging and direct peer to peer communications. Furthermore, the protocols used within the CKI are designed to operate over a variety of messaging and peer to peer communication services.

13. The keys provided by the CKI may be used to secure CKI key management protocol exchanges as well as other security management and non-security applications.

B. Domains

14. Users are grouped together by *domains* for the purposes of CKI key management.

15. Users in the same *domain* generally belong to the same community (eg government department), come under the responsibility of a common security authority and operate under a common security policy.

16. An identified user only belongs to one domain.

Note: a person may only belong to several domains if it operates under different identities.

17. Aspects of the CKI which relate to operation within the *domain* are called *local*.

18. Aspects of the CKI which relate to operation between *domains* are called *external*.

C. CKI Components

19. The components of the CKI are illustrated in figure 2:

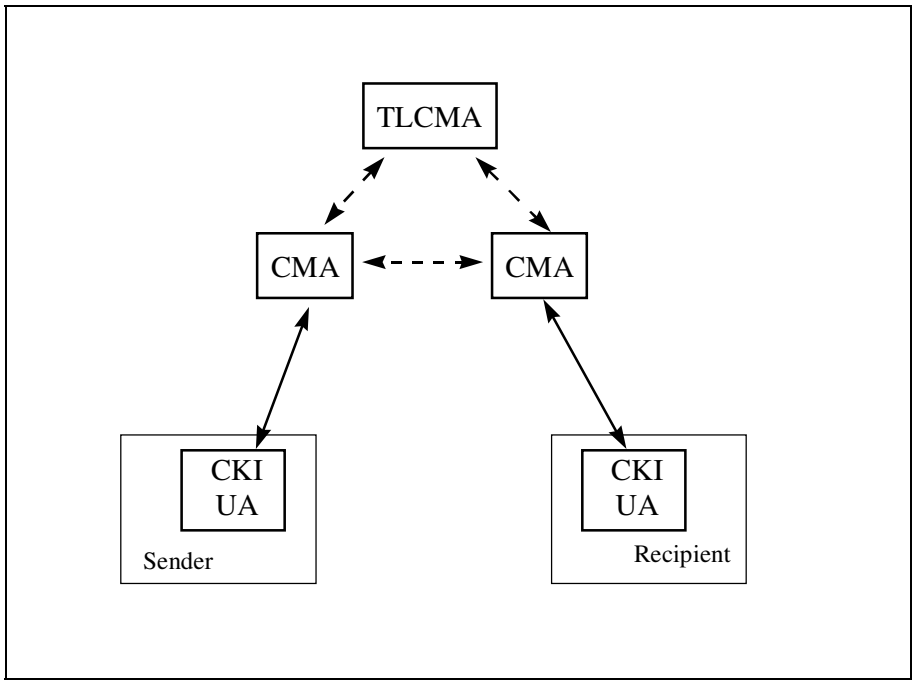

Figure 2 CKI Components

20. The local cryptographic and protocol functions required to support the CKI are performed by the *CKI User Agent* (CKI UA) within the sender and recipient systems.

21. Functions required for the management of keys for a domain of users are supported by a *Certificate Management Authority* (CMA).

22. There is one, and only one, identifiable CMA entity per domain.

Note: a CMA may be implemented by more than one computer system, (eg as a distributed system or a system with back up) provided it is identifiable as a single entity.

23. Key management information required for operation of the CKI between domains (eg interoperability key - see § III.C) is established either:

a. By peer to peer protocol exchanges between the CMA's for each domain;

b. From a *Top Level Certificate Management Authority* (TLCMA) which supports the key management of the top level keys for a set of interoperating CMAs.

Note: a TLCMA need not be present if approach (a) is used.

24. A CMA is a certification authority for public keys used in the CKI.

D. Supporting Functions

25. As illustrated by figure 3 the CKI can make use of two supporting infrastructures:

26. A directory (eg as defined in [X.500]) can be used to distribute certified public keys between CKI components (see § IV.A). The CKI, however, does not depend on the use of a directory for the distribution of certificates.

27. An Authentication Key Infrastructure (AKI) can be used to provide certified keys for signing CKI certificates and protecting the CKI key management protocol exchanges. The AKI and CKI together form the Public Key Infrastructure (PKI).

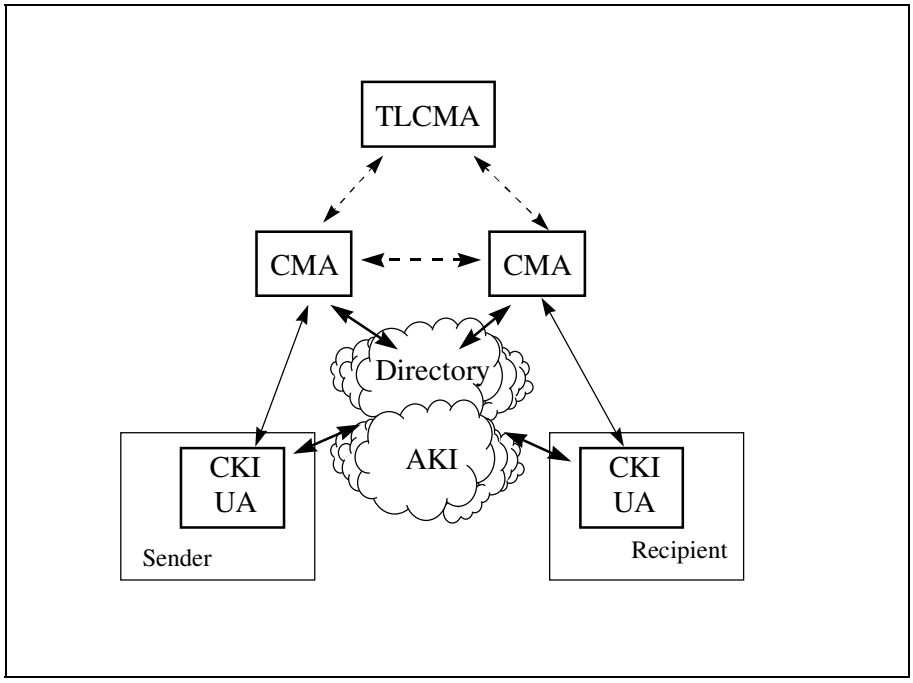

Figure 3 Supporting Functions

III. CKI CRYPTOGRAPHIC FUNCTIONS AND ASSOCIATED KEYS

A. Send and Receive Public / Private Keys

28. The basis of the CKI key generation process is the Diffie-Hellman algorithm as defined in [DH76].

29. The CKI uses separate Diffie-Hellman public / private key pairs for the generation of the shared secret keys for a given direction between two parties as follows:

a. *Send private key*: Assigned to the sender and used by the sender's CKI UA to generate the shared secret key;

b. *Send public key*: Assigned to the sender and used by the recipient's CKI UA to generate the shared secret key;

c. *Receive public key*: Assigned to the recipient and used by the sender's CKI UA to generate the shared secret key;

d. *Receive private key*: Assigned to the recipient and used by the recipient's CKI UA to generate the shared secret key.

30. The key generation process also depends on the use of a base (g) and modulus (N) common to the sender and recipient.

31. The generation of a shared secret key K_{shared} from a *receive / send public key* (K_{pub}) and a *send / receive private key* (K_{priv}) with the base and modulus (g , N) can be described mathematically as the function f1:

$K_{shared} = f1(K_{pub}, K_{priv}, g , N)$

32. This function is illustrated in figure 4:

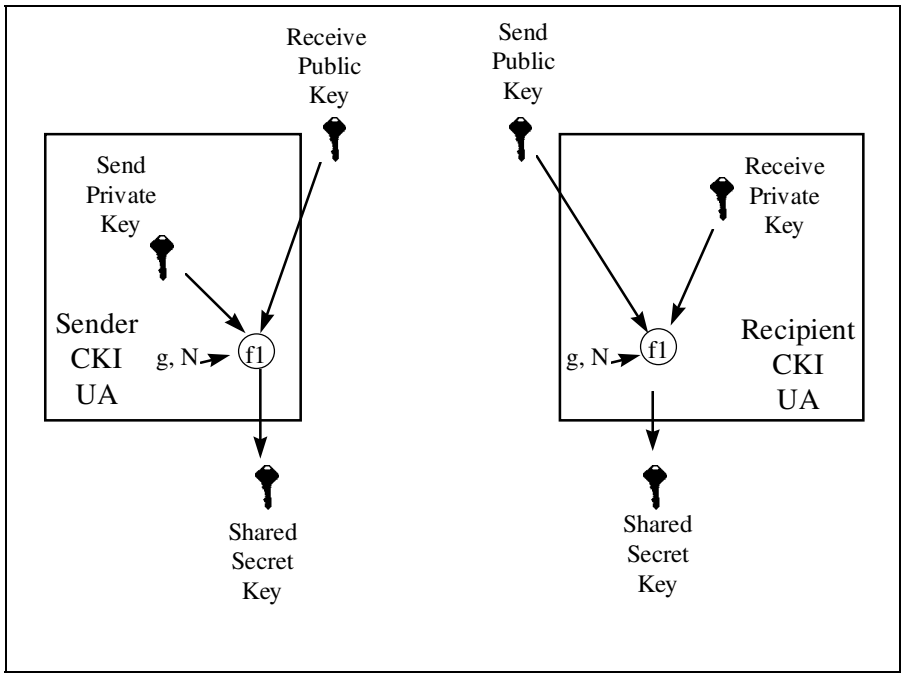

Figure 4 Diffie-Hellman public / private keys

B. Seed Keys

33. The send and receive private keys are generated by the CKI UA from *seed keys* provided by its CMA.

34. The CKI UA uses a datestamp to generate a private key from the seed key. The same *datestamp* value is used by the CMA in the generation of the public key. This *datestamp* is also used in the validity period of the public key certificate (see §IV.A).

35. Several private keys can be generated from the same *seed key* by using different *datestamp* values.

36. A CKI UA uses a local *seed key* for the generation of:

a. send private keys to protect communication to users in the local and external domains, and

b. receive private keys to protect communications from users in the local domain.

37. A CKI UA uses an *external seed key* for each remote domain for the generation of receive private keys to protect communications from users in that external domain.

38. The generation of a (send or receive) private key (K_{priv}) using a (local or external) seed key (K_{seed}) and a *datestamp* (T) can be described mathematically as the function f2:

$K_{priv} = f2(K_{seed}, T)$

39. The generation of a shared secret key in CKI UA is illustrated in figure 5:

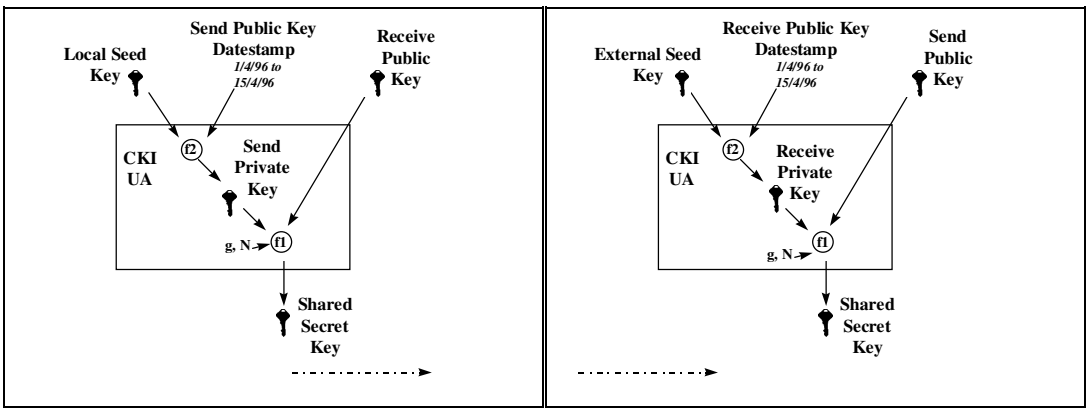

Figure 5 Key generation functions using a seed key - Interdomain

> **Note:** for communication within a domain the recipient would use the *local seed key* for the generation of its receive private key.

40. The functions to generate the send private key (f1) and the shared secret key (f2) can be implemented as a single function with the *seed key*, *datestamp* and receive / send public key as inputs. In mathematical terms f1 and f2 can be combined into a single function f3:

$K_{shared} = f3(K_{pub}, K_{seed}, T, g, N)$

C. Interoperability Keys

41. The seed keys and public keys are generated using domain *interoperability keys* held by a CMA.

42. A seed key is a function of the *interoperability key*, the user's distinguished name (see § VI.B) and a seed key identifier.

43. A send / receive public key is a function of the *interoperability key*, the user's distinguished name, the associated seed key identifier and a datestamp.

44. Local seed keys, send public keys and receive public keys for local communications are generated using a *local interoperability key* which is specific (internal) to a domain.

45. The external seed keys and receive public keys external communications are generated using an *external interoperability key* which is shared with a CMA for an external domain.

46. Both the send and receive public keys are generated by the sender's CMA.

47. The same *external interoperability key* is used to protect communications in either direction between the domains.

48. *External interoperability keys* are either:

a. generated and passed to the CMAs by the Top Level CMA (TLCMA), or

b. generated by the CMAs themselves using the Diffie Hellman algorithm [DH76] with a *domain public key* for the external domain and a *domain private key* for the local domain.

49. The generation of a local / external seed key (K_{seed}) from a *local / external interoperability key* ($K_{interop}$), the user's name (DN) and a seed key identifier (KId_{seed}) can be described mathematically as the function f4:

$K_{seed} = f4\ (K_{interop}\ ,\ DN\ ,\ KId_{seed})$

50. Similarly, the generation of a send / receive public key (K_{pub}) from a *local / external interoperability key* ($K_{interop}$), the user's distinguished name (DN), a seed key identifier (KId_{seed}) and a datestamp (T) can be described mathematically as the function f5:

$K_{pub} = f5\ (K_{interop}\ ,\ DN\ ,\ KId_{seed}\ ,\ T)$

51. These functions and the keys used in generating a shared secret for the sender and recipient key can be illustrated in figures 6 and 7:

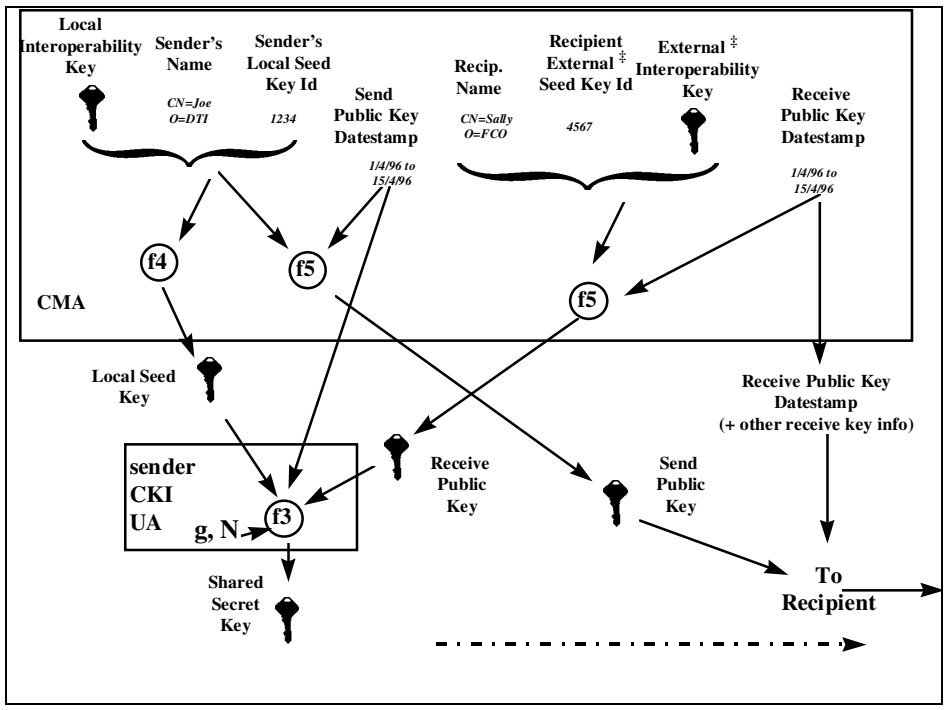

Figure 6 CKI UA and CMA key generation functions - Sender

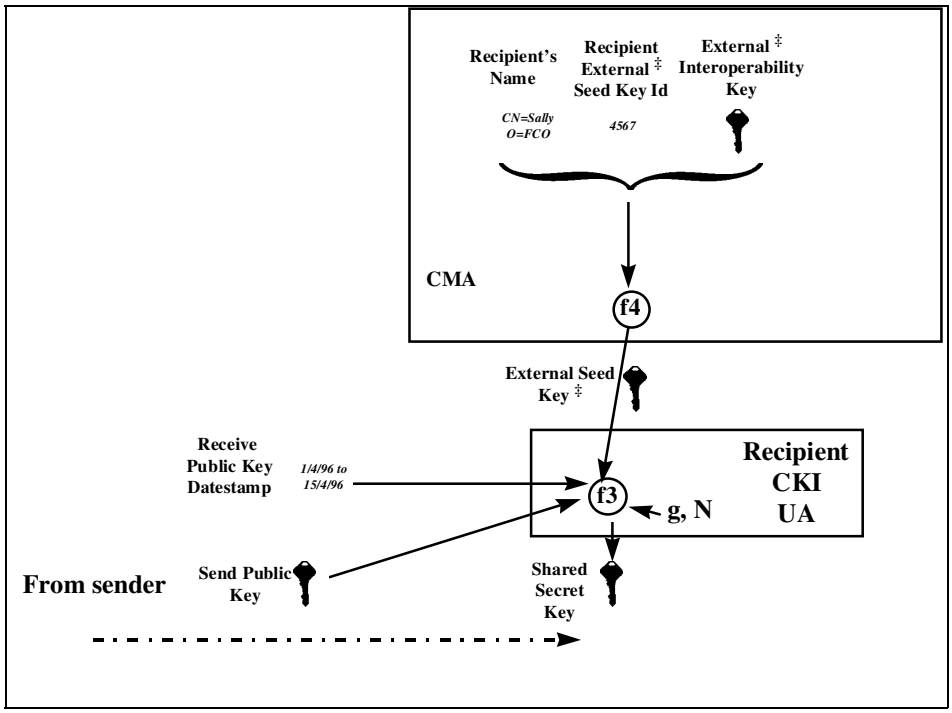

Figure 7 CKI UA and CMA key generation functions - Recipient

Note: ‡ for communication within a domain *local interoperability keys* and associated local seed keys are used, instead an *external interoperability key* and external seed key.

IV. DISTRIBUTION OF KEYS AND ASSOCIATED PARAMETERS

A. Use of X.509 Certificates

52. The send or receive public keys are distributed in the form of a public key certificate as defined in [X.509] and [X.509DAM] signed by the CMA which generated the public key.

53. Four types of public key may be carried in the certificate. These certificates are referred to as follows:

a. Send public key - *send certificate*

b. Receive public key - *receive certificate*

c. Domain public key - *domain certificate*

d. The CMA's public key used for signing certificates - *CMA certificate*

Note: a *CMA certificate* is a form of *CA certificate*. *CMA certificates* are produced by the Authentication Key Infrastructure.

54. *Send, receive* and *domain certificates* are referred to collectively as *CKI certificates*.

55. Both *send* and *receive certificates* are created by the sender's CMA.

56. The start and end dates in the certificate validity period is also used as the datestamp for input into the key generation function (see §III.B).

57. The base and modulus to be used in the generation of the shared secret key is placed in the send certificate, either explicitly or by reference using an object identifier.

58. The seed key identifier used in generating the public key is placed in the *send* and *receive certificates*.

59. An identifier for the CMA's certification public key is placed in the *send, receive* and *domain certificates* so that, in the case of a CMA's certification public key being revoked, the effected "user" certificates can be identified (see §V.C on revocation of certification keys).

60. CMA certificates contain the name constraints (see §VI.C) for user's in the CMA's domain.

61. If the validity of a *CKI certificate* is uncertain (eg a currently valid certificate revocation list is not available - see §V) then the certificate should be taken as being invalid.

62. The profile for use of X.509 certificates is specified in part 3.

B. Distribution of Certificates

63. Send and receive certificates are distributed to the sender from its CMA either:

a. Using the CKI key management protocol (Part 2), or

b. Via a directory (Part 4).

64. The send and receive certificates are distributed from the sender to the recipient in the user to user communication protocol along with the enciphered data (eg as part of the header for the protected message).

Note: The CKI does not preclude alternative communication paths being used.

The receive certificate is passed from the sender to the recipient to provide the seed key identifier and datestamp required for creating the recipient's private key.

65. A domain certificate (and the domain private key) can be distributed from the TLCMA to the CMA for that domain.

66. CMA certificates are distributed from the TLCMA to CMAs and from CMAs to their users either:

a. Using the CKI key management protocol (see part 2), or

b. Via a directory (Part 4), or

c. By some other means (eg using an Authentication Key Infrastructure)

67. A CKI UA may pre-load and/or cache certificates.

68. A CKI UA is responsible for checking the validity of the certificates it uses (see §V on revocation).

C. Distribution of Seed Keys

69. The local seed key is loaded by means outside the scope of the CKI specifications (eg manually).

70. External seed keys are loaded from the CMA into a recipient CKI UA using the CKI key management protocol (Part 2).

71. When requesting an external seed key, the CKI UA provides the certificate for the associated receive public key which contains the information needed to generate the seed key (i.e. name and seed key identifier, see §VI.A).

72. In its response a CMA, as well as providing the external seed key, passes the certificate for the external CMA .

D. Distribution of Interoperability Keys

73. Interoperability keys, and other parameters which control the operation of a domain, are either:

a. passed from a TLCMA to the CMAs using protocol exchanges with the TLCMA, or

b. generated by the CMAs themselves using peer CMA to CMA protocol exchanges to carry their domain certificates.

Note: the CKI key management protocol is defined in part 2.

74. The other parameters established by the above protocol exchanges are:

a. The base and modulus to be used for communications sent from the CMA's domain. In the case of peer CMA to CMA exchanges (see §IV.D above) this information is loaded by means outside the CKI protocol.

b. The name constraints for users in the local and external domains (see §VI.C),

c. An initial list of revoked users in the local and external domains (see §V.A),

d. A "my" and "your" reference for the interoperability key (see §VI.A),

e. The CMA certificate for the external CMA; this certificate contains the name constraints for users in that domain as well as the CMA public key.

E. Key Recovery

75. Optionally, a CMA may support the on-line recovery of the private key pair associated with a certified public key for use by an authorised law enforcement agency.

76. Access to on-line key recovery is supported using the CKI key management protocol (Part 2).

F. Protection of Key Distribution

77. Public keys are always distributed in the form of a certificate which has its own inherent protection.

78. The CKI key management protocol exchanges generally require authentication and integrity protection. This is provided by the underlying messaging or communications service using, for example, the secure messaging service for which the CKI was initially developed [HMG].

79. Additional confidentiality protection is required for the distribution of:

a. Interoperability keys sent from the TLCMA to a CMA,

b. Domain private keys sent from the TLCMA to a CMA,

c. External seed keys sent from a CMA to a CKI UA,

d. Private keys passed from a CMA to an authorised law enforcement agency for key recovery.

80. This confidentiality is achieved by either or both of:

a. Protection applied directly to the key (i.e. outside the scope of the CKI specifications).

b. Confidentiality applied as part of the underlying messaging or communications service using, for example, the secure messaging service [HMG].

Note: the mechanism used to directly protect keys is currently outside the scope of the CKI specifications.

81. The keys used by the TLCMA and the CMA for confidentiality of exchanges should not be the same as those used for normal (ie not CKI key management protocol) communications.

82. A CKI UA's local seed key may be used in the protection of keys sent from the CMA.

G. Start up Conditions

83. The following identifies the keys required to start up the operation of the CKI.

84. These keys are established by means outside the scope of the CKI specifications. Those keys marked with a ‡ are provided by the AKI. Other keys may be loaded, for example, manually.

85. Some keys are only relevant to specific modes of use of the CKI.

86. The following start up keys need to be established in the TLCMA. These keys are only required if a TLCMA is used to establish interoperability keys:

a. Keys used to protect CKI key management protocol exchanges with CMAs.

(i) If secure electronic mail [HMG] is used to protect these exchanges this requires:

- A TLCMA local seed key and send certificate,
- An interoperability key to generate receive certificates for CMAs known to the TLCMA,

Note: for the purposes of securing communications between the TLCMA and CMAs, the CMAs form a domain with the TLCMA acting also as the top level domain CMA.

- Keys to authenticate exchanges ‡.

(ii) If interoperability keys are directly protected:

- The key encrypting keys used to protect interoperability keys.

87. The following start up keys need to be established in each CMA:

a. The private key of the CMA‡ used to sign the CKI certificates and its associated CMA certificate ‡.

b. CA public key(s) ‡ to validate CMA certificates from external domains.

c. Keys used to protect CKI management protocol exchanges.

(i) If secure electronic mail [HMG] is used to protect these exchanges this requires:

- If a TLCMA is used to establish interoperability keys, seed key needed to receive data from the TLCMA and public key‡ needed to validate CKI certificates from the TLCMA.
- Keys to authenticate exchanges ‡.

(ii) If a peer to peer exchange is used to establish the interoperability key, CMA's domain private key and domain certificate.

Note: this information can be provided by a TLCMA, if present.

(iii) If interoperability keys are directly protected, the key encrypting keys used to protect interoperability keys.

(iv) If external seed keys are directly protected, the key encrypting keys used to protect external seed keys.

88. The following start up keys need to be established in each CKI UA:

a. The local seed key.

b. The CMA Certificate ‡ for the user's CMA.

c. CA public key(s) ‡ to validate external CMA certificates.

d. Keys used to protect CKI management protocol exchanges.

(i) If secure electronic mail [HMG] is used to protect these exchanges this requires:

- Keys required to authenticate exchanges ‡.

(ii) If external seed keys are directly protected, the key encrypting key used to protect external seed keys sent from the CMA.

V. REVOCATION

A. Revocation of Send & Receive Certificates

89. The serial numbers of revoked receive certificates are listed in a certificate revocation list (CRL) as defined in [X.509].

> **Note:** CRLs may include send certificates but this is not considered essential for the operation of the CKI. Confidentiality is seen as primarily a concern of the sender to ensure that the recipient's key has not been compromised. Once a message has been sent it is too late to stop a breach of confidentiality if the sender's key has already been compromised. The sender may be informed of a compromise of his own keys without the use of CRLs.

90. CRLs are distributed by a CMA to all CKI UAs in its domain using the CKI key management protocol (Part 2).

> **Note:** CRLs may be sent to external (recipient domain) CMAs for forwarding to users in their domains. However, this is not considered essential for the operation of the CKI for reasons similar to those given in the note for para 90 above.

91. CRLs may also be distributed via the directory. Revoked certificates should be removed from the directory. (See part 3)

92. It is recommended that certificate validity periods are relatively low (eg a few days or weeks). Thus, any administrative changes need not necessarily be reflected in CKI CRLs.

93. If the validity of a *CKI certificate* is uncertain (eg a currently valid certificate revocation list is not available) then the certificate should be taken as being invalid.

94. If CKI Certificates are obtained in real time from the CMA (i.e. CKI certificates are not cached in the CKI UA nor obtained through a general purpose directory) then it is not necessary to use CRLs to check the validity of the certificates.

B. Blacklisted Users and External Seed Key Revocation

95. A *revoked user list* contains the names of users from an external domain who have been blacklisted, or whose external seed keys have been revoked, for example, due to compromise. This list is passed from the TLCMA or between CMAs using the CKI key management protocol (Part 2).

> **Note:** a user requiring CKI keys for different purposes (eg electronic mail, web communications) may use different names for each purpose. In such a case, only the user's seed key created for a particular name would be revoked and this may not be all the seed keys of an actual user entity.

96. An initial *revoked user list* is provided when a new interoperability key is established and an updated *revoked user list* is sent whenever a user is added or removed from the list. The full list is sent on each protocol exchange.

97. Unexpired receive public key certificates for users in the *revoked user list* shall be revoked (i.e. included in the CRL) and no new certificates created for these users until they are no longer on the revoked user list. In addition, any current seed key identifier for that revoked user shall not be re-used.

98. When a user is removed from the *revoked user list*, new certificates can be created for that user using a new seed key identifier.

99. The handling of blacklisted local users and the revocation of local seed keys is outside the scope of this specification.

C. Revocation of CMA Keys and CMA Compromise

100. If a CMA's certification public key is revoked all certificates signed by that CMA are revoked.

101. A CRL containing a CA certificate (authority) revocation list can be used to inform CKI UAs of authorities from external domains whose public keys have been compromised.

102. In the case of a TLCMA being used the authority revocation list is issued by the TLCMA to all CMAs for onward distribution to its users.

103. The authority key identifier in a CKI certificate is used to identify the certificates impacted by the revocation of a CMA's public key.

VI. IDENTIFICATION

A. Key identifiers

104. When an external interoperability key is established (either through peer exchange or with a TLCMA) each CMA establishes two references to the interoperability key which are used as a "my" and "your" reference. Each CMA's "my" reference is unique within its domain.

105. An external *seed key identifier* is an octet string containing a concatenation of:

a. the sender's domain "my" external interoperability key reference (which equals the recipients "your" reference), followed by

b. the recipient's domain "my" external interoperability key reference (which equals the sender's "your" reference), followed by

c. a unique identifier selected by the sender's CMA.

106. The resulting *seed key identifier* is unique within both the sender's and recipients domain, but may not necessarily be globally unique.

107. The structure of a local *seed key identifier* is a local matter.

108. The *seed key identifier* is used in generating the seed key and associated public key (see §III.C) and is carried in the certificate for the public key (see §IV.D).

109. Only one *seed key identifier* should be used with a given user name at any one time except where rekeying is taking place, for example, for recovery after a user's seed key is compromised.

110. This key identifier does not uniquely identify the send or receive public key.

> **Note:** if required the certificate serial number or a hash of the public key can be used for this purpose.

B. Naming

111. CKI Users and CMAs are identified using a Distinguished Name as defined in [X.500].

112. The name of the CMA is also used to uniquely identify a domain.

113. It is recommended that a CMA is named separately from the CA which signs authentication certificates for a user domain, and that they use separate public keys.

114. If a CKI user (sender or recipient) requires that different keys are provided for different purposes (eg electronic mail, web communications) then different names should be used.

C. Domain Name Constraints

115. The user naming space for a CMA's domain is defined by *name constraints* carried in its CMA certificate using the certificate extension field defined in [X.509DAM].

116. *Name constraints* specify the set of naming subtrees permitted and excluded from the CMA's domain. A subtree can either name:

a. an organisational body (eg department, section or group), to indicate that all members of that organisation are permitted / excluded, or

b. a single user

117. The excluded subtrees takes precedence. Thus, the excluded subtrees can be used to specify those users, or organisational units which are excluded from a larger (permitted) organisational body.

118. If excluded subtrees only are specified then the CMA's user domain is taken to encompass all those names outside the excluded subtrees.

119. This information is used by a CMA to identify the external CMA responsible for a given user name.

120. If a receive certificate is requested for that user which is identified as being within the *name constraints* of more than one CMA then that request will fail.

121. A CMA may also take local action when an interoperability key is established with an external CMA which has overlapping *name constraints* (eg pass a message to the system security manager to resolve the conflict).