Open Copy Protection System

Philips Research
Proposal to Broadcast Protection Discussion Group

Version 1.4, May 7, 2002

Principle Authors: Michael A. Epstein, Michael S. Pasieka
Auditor: Dr –Ing Christof Paar (christof@ece.wpi.edu)
Philips Research
345 Scarborough Road
Briarcliff Manor NY 10510
Voice: 914/945-6239
Fax: 914/9456141
Email: Michael.Epstein@philips.com

TABLE OF CONTENTS

1 A DETAILED SUMMARY OF THE PROPOSAL 5

1.1 TERMINOLOGY 5
1.2 SCOPE 5
1.3 SOURCE AND SINK DEVICES 6
1.4 THEORY OF OPERATION 6
1.4.1 NOTATION 6
1.4.2 OCPS INITIAL CONDITIONS AND DEVICE STORAGE ASSUMPTIONS 6
1.4.3 OCPS BLOCK CIPHER 7
1.4.4 OCPS PROTOCOL OVERVIEW 7
1.4.5 OCPS PROTOCOL DETAILED DESCRIPTION 7
1.4.6 OCPS COPY CONTROL INFORMATION 10

2 KEY MANAGEMENT 10

2.1 KEY DISTRIBUTION 10
2.1.1 KEY GENERATION AND ELGAMAL PARAMETERS 11
2.1.2 TRUST AUTHORITY KEY PROTECTION 11
2.1.3 KEY DISTRIBUTION 12
2.2 KEY REVOCATION 13
2.2.1 CRITERIA FOR KEY REVOCATION 14
2.2.2 KEY REVOCATION PROCESS 14

3 IMPLEMENTATION 1

3.1 GENERAL SOFTWARE AND HARDWARE IMPLEMENTATION 15
3.1.1 PUBLIC KEY CERTIFICATE DEFINITION 15
3.1.2 REVOCATION NOTICE DEFINITION 16
3.1.3 PROTOCOL IMPLEMENTATION 16
3.1.4 STORAGE REQUIREMENTS 17

4 ROBUSTNESS OF EACH CRYPTOGRAPHIC ALGORITHM 18

4.1 ATTACK ON THE OCPS BLOCK CIPHER 18
4.2 SECURITY OF THE INITIAL KEY ESTABLISHMENT 18
4.3 SECURITY OF THE KEY DERIVATION PROTOCOL 19

5 ERROR PROPAGATION CHARACTERISTICS OF THE ENCRYPTION ALGORITHM 19

5.1 SINGLE CIPHERTEXT ERROR IN THE BLOCK CIPHER 19

6 RENEWABILITY 19

6.1 REVOCATION IS RENEWABILITY 19

7 RESISTANCE TO OBSOLESCENCE 20

7.1 RESISTANCE TO IMPROVED ATTACKS 20
7.2 RESISTANCE TO IMPROVED COMPUTER SPEED 20

8 MAINTENANCE COMPLEXITY 20

8.1 MAINTENANCE OF THE SYSTEM 20

9 APPLICABILITY TO DIFFERENT DIGITAL INTERFACES 21

10 AVAILABILITY FOR US IMPORT/EXPORT 21

11 LICENSING TERMS 21

12 BLOCK CIPHER MODE 21

13 CIRCUMVENTION DEVICES 22

13.1 DEFEATING CIRCUMVENTION 22

14 AMENDMENTS NEEDED TO INTERFACE STANDARDS 22

14.1 SPECIFICS RELATING TO THE IEEE 1394 BUS. 22

15 VIEW OF SUBMITTER REGARDING STANDARDIZATION OF COPY PROTECTION 1

15.1 THE LONG TERM VIEW ON COPY PROTECTION 24

16 OTHER INFORMATION 25

APPENDIX A REFERENCES 26

1 A Detailed Summary of the Proposal

We present the following proposal to the Broadcast Protection Discussion Group (BPDG) as an approved digital output protection technology.

We name our system the Open Copy Protection System (OCPS – pronounced *octopus*). Our intent is to present an open system where all of the components are drawn from commonly available standards.

1.1 Terminology

APS	Analog Protection System
CBC	Cipher Block Chaining
CE	Consumer Electronics including such other products used to receive consumer content including products used in connection with personal computers
OCCI	OCPS Copy Control Information
CEA	Consumer Electronics Association
CGMS	Copy Generation Management System (Information)
CFI	Call for Information
DES	Digital Encryption Standard
DSTB	Digital Set Top Box
DT	Digital Tape
DTV	Digital Television
DVR	Digital Video Recorder
DVD	Digital Video Disc
DVDP	Digital Video Disc Player
IEEE	The Institute of Electrical and Electronics Engineers
IP	Intellectual Property
MPAA	Motion Picture Association of America
MPEG	Motion Picture Expert Group
OCPS	Open Copy Protection System
PC	Personal Computer
TA	Trust Authority

1.2 Scope

The scope of this specification covers the copy protection of content in a limited context. We cover the causal attacker. A casual attacker is an ordinary consumer that is motivated to misuse content against accepted rules including redistribution over the Internet . We also cover the attacker who wants to create cloned devices.

We include coverage of the casual attacker by creating a specification that is immune to the two well-known types of attacks: a hardware black box and a software patch to the system.

We prevent two types of advanced attacks. The first type of attack is the breaking of cryptographic keys. In this type of attack, the attacker must break a private key used in the initial key exchange and all subsequent random keys used during the session or break each session key. The second type of attack is a cloning attack. This attack is difficult due to the security of the private key in hardware. Additionally, a revocation mechanism is specified so that known attack devices may be remotely deactivated.

1.3 Source and Sink Devices

We assert that our methods are interface independent, however we limit it's use by license to specific network interfaces. We define a source device as a product that sends content using OCPS to a sink device. A sink device is a product that receives content from an OCPS link. Trust Authority

In our protocol, we use the concept of a Trust Authority (TA). This is a set of secure trusted third parties set up to issue certificates. We assume the TA controls the long life private key. The TA shall sign two types of certificates. The first type is a certificate containing the public keys of each device. The second type is a certificate containing revocation notifications.

1.4 Theory of Operation

1.4.1 Notation

We describe the use of private and public keys, random numbers and certificates. In the case of either a public or private key, the notation is the capital **K** subscripted by the type and owner of the key. For example, $\mathbf{K_{PubSource}}$ is the public key of a source device. Random numbers are denoted as **R** subscripted by the device, which created the random number. For example, $\mathbf{R_{Source}}$ is a random number created on a source device. As for certificates, we use **Cert** subscripted by the signing entity and parenthetically enclosing the data signed. For example, $\mathbf{Cert_{FooBar}(K_{PubSink})}$ is a certificate signed by FooBar containing a public key of a sink device. We denote an encryption as **E** and enclose in brackets the key used to encrypt and enclose in parentheses the data encrypted. For example, $\mathbf{E\{K_{PubSink}\}(R_{Source})}$ is a random number generated on a source device and encrypted using the public key of a sink device. Finally, $\mathbf{H(R_{Source}, K_{RandSource})}$ denotes the hashing function, **H**, operating on the concatenation of $\mathbf{R_{Source}}$ and $\mathbf{K_{RandSource}}$.

1.4.2 OCPS Initial Conditions and Device Storage Assumptions

The OCPS protocol assumes some initial conditions. We specify that each device shall securely store a unique private key, either $\mathbf{K_{PrvSource}}$ for a source device or $\mathbf{K_{PrvSink}}$ for a sink device, and the public key of the TA, $\mathbf{K_{PubTA}}$. Additionally, a X.509 certificate [X.509] shall be stored containing the corresponding public key and digitally signed by a

TA, either $\mathbf{Cert_{TA}(K_{PubSource})}$ for a source device or $\mathbf{Cert_{TA}(K_{PubSink})}$ for a sink device. We use these in the authentication phase of the protocol. Finally, each device shall be able to securely store a number of public keys that have been revoked. See Sections 2.2.2 and 3.1.4 for further details.

We also assume that a compliant source device must inform OCPS of the nature of the material to be transferred. The nature of material that can be transferred is defined in table 1 which describes the OCCI codes accommodated by the OCPS protocol.

1.4.3 OCPS Block Cipher

We specify the OCPS block cipher as the Digital Encryption Standard (DES) in Cipher Block Chaining (CBC) mode. See Sections 3.1.3.2 and 12 for further details.

1.4.4 OCPS Protocol Overview

The OCPS protocol has five phases. The first phase is the authentication phase where the source and sink devices authenticate each other. The second phase is the key exchange phase where random numbers and key material are created, encrypted and exchanged. The third phase is the key generation phase where the key material is used to create a session key. The fourth phase is the information transmission stage where the content is securely transferred between the source and sink devices. Finally, the fifth phase is the key update phase, which is periodically executed, while phase four is executed. The period of phase five is defined in Section 3.1.3.4.

1.4.5 OCPS Protocol Detailed Description

Referring to Figure 1, the first phase of the protocol is authentication and is initiated by the source device sending $\mathbf{Cert_{TA}(K_{PubSource})}$ to the sink. The key $\mathbf{K_{PubSource}}$ is an ElGamal public key generated as specified in Section 2.1.1. The sink device then verifies the certificate using $\mathbf{K_{PubTA}}$. The sink device then checks that the public key

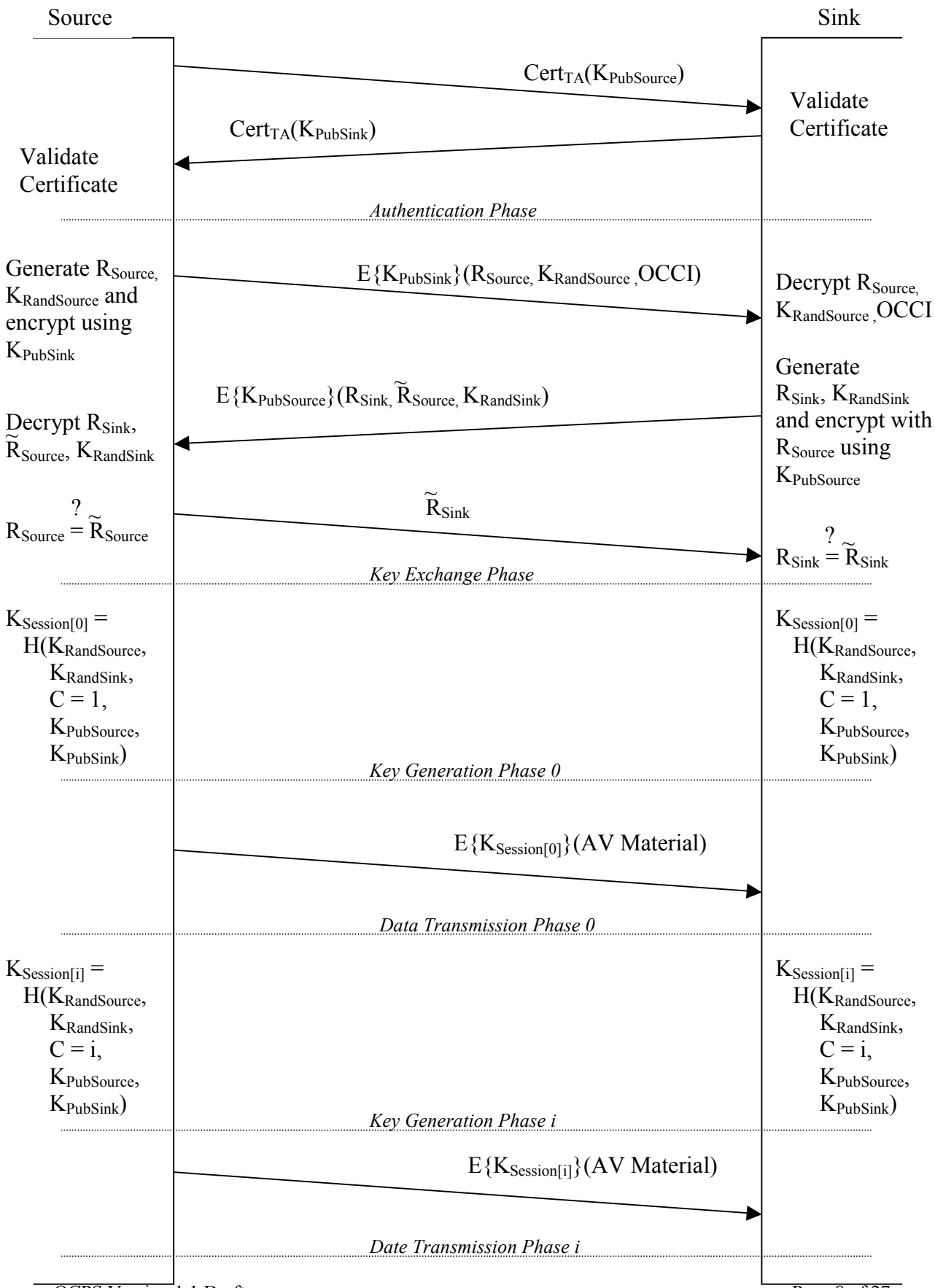


Figure 1 OCPS Key Agreement Protocol

$\mathbf{K_{PubSource}}$ is not in the revocation list stored in the sink. If the certificate is legitimate and the public key is not revoked then the public key of the source is accepted.

In response to the acceptance of the public key of the source, the sink device sends $\mathbf{Cert_{TA}(K_{PubSink})}$ to the source device. The key $\mathbf{K_{PubSink}}$ is an ElGamal public key generated in an analogous manner to $\mathbf{K_{PubSource}}$. The source device then verifies the certificate using $\mathbf{K_{PubTA}}$. The source device then checks that the public key is not in the revocation list stored on the source. If the certificate is legitimate and the public key is not revoked then the public key of the sink is accepted. This completes the authentication phase.

The second phase of the protocol is the key exchange phase. This phase implements the modified Needham-Schroeder protocol as described in [Menezes, et al. 1]. The source device generates a 64 bit random number, $\mathbf{R_{Source}}$, 128 bits of true random key material, $\mathbf{K_{RandSource}}$, and the OCCI bits encrypts all values using the public key of the sink device creating $\mathbf{E\{K_{PubSink}\}(R_{Source}, K_{RandSource}}$,OCCI). This encrypted value is sent to the sink device, which decrypts the value using $\mathbf{K_{PrvSink}}$ thus retrieving $\mathbf{R_{Source}}$,$\mathbf{K_{RandSource}}$ and OCCI. OCCI will be used to govern the use the content transferred via OCPS and terminate the transfer if necessary.

In response, the sink device generates a 64 bit random number, $\mathbf{R_{Sink}}$, and 128 bits of true random key material, $\mathbf{K_{RandSink}}$, and encrypts both values and the random number $\mathbf{R_{Source}}$ using the public key of the source device creating $\mathbf{E\{K_{PubSource}\}(R_{Sink}, R_{Source}, K_{RandSink})}$. This encrypted value is sent to the source device, which decrypts the value using $\mathbf{K_{PrvSource}}$ and retrieving $\mathbf{R_{Sink}}$, $\mathbf{R_{Source}}$ and $\mathbf{K_{RandSink}}$.

At this point the time needed for $\mathbf{R_{Source}}$ to make the round trip is measured against an accepted maximum threshold of 1 milliseconds. If the round trip time is greater than the maximum time allowed and the OCCI forbids non-local transmission than the protocol is terminated by the source device.

The source then compares the received $\mathbf{R_{Source}}$ with the random number just sent. If they are equal the source sends the random value, $\mathbf{R_{Sink}}$ back to the sink. The sink then compares the received $\mathbf{R_{Sink}}$ with the random number just sent. This completes the key exchange phase.

At this point the time needed for $\mathbf{R_{Sink}}$ to make the round trip is measured against an accepted maximum threshold of 1 milliseconds. If the round trip time is greater than the maximum time allowed and the OCCI forbids non-local transmission than the protocol is terminated by the sink device.

The third phase of the protocol is the key generation phase. Both the source and sink create a 32 bit counter value **C** initialized to one. Both the source and the sink device create the session key $\mathbf{K_{Session[0]}}$ by computing $\mathbf{H(K_{RandSource}, K_{RandSink}, C, K_{PubSource}, K_{PubSink})}$.

The fourth phase is the information transmission stage. The session key $\mathbf{K_{Session[0]}}$ is used to encrypt the AV material using the OCPS block cipher. This encrypted material cannot be intercepted since an outsider does not know the key used. In addition, the sink device can only decrypt useful information, i.e., AV material, from the decrypted message during the current phase.

Periodically hereafter, the fifth phase, the key update phase shall execute. This assures the breaking of a single session key will cause only limited exposure. The key update phase simply requires that the counter **C** is incremented. Then, both the source and the sink device create the new session key $\mathbf{K_{Session[i]}}$ by computing $\mathbf{H(K_{RandSource}, K_{RandSink}, C, K_{PubSource}, K_{PubSink})}$.

1.4.6 OCPS Copy Control Information

The OCPS Copy Control Information (OCCI) is composed of 8 bits which are carried as additional data in the authentication phase. The assigned value of the OCCI bits are in table 1.

Value	Meaning
0x0	Unrestricted content
0x1	Unscreened content
0x2	Marked content
0x3 – 0x	Reserved

2 Key management

The key management system of this proposal consists of two sections: key distribution and key revocation. The key distribution section describes how keys are created, authenticated and distributed to Consumer Electronics (CE) manufacturers and CE devices. The key revocation section describes the manner in which public keys are revoked.

2.1 Key Distribution

The key distribution system relies first on a TA, which protects a private key, designated $\mathbf{K_{PrvTA}}$, with absolute certainty. The public key of the TA, designated $\mathbf{K_{PubTA}}$, can be well known without any loss of security. Indeed, this public key will be contained in each CE device using OCPS.

2.1.1 Key Generation and ElGamal Parameters

All public private key pairs used in this proposal will be generated as per Annex B of [ANSI X9.42]. In particular the key pair used by the TA should be carefully chosen since a successful attack on a TA private key would lead to a catastrophic failure of the entire copy protection system.

The TA will rely on the DSA Signature system as described in [FIPS 186]. The TA public key prime will be 1024 bits long as it is used only for signatures. All other public keys will be for the ElGamal encryption algorithms. Every party has one distinct ElGamal public/private key pair. The ElGamal prime modulus is limited to 512 bits by US export control. To be secure, this proposal requires that during manufacture, a different prime modulus is used for each device in the generation of its ElGamal public/private key pair. See Section 4.2 for a discussion of the security considerations.

This proposal uses an ElGamal algorithm as described in [Menezes et al. 2] with the difference that it is based on a discrete logarithm problem in a subgroup, analogous to the Diffie-Hellman algorithm described in [ANSI X9.42]. The use of a subgroup requires the following changes to the description in [Menezes et al. 2]:

Pertaining to Algorithm 8.17:

- p is a 512 bit prime generated according to [ANSI X9.42, Annex B.1].
- q is a prime and order of a subgroup of the multiplicative group of $GF(p)$. q has a bit length equal to 160. q is identical to the parameter in [ANSI X9.42].
- α is a generator of the subgroup with order q. α is analogous to the parameter g in [ANSI X9.42] and is selected as described in [ANSI X9.42, Annex B.2].
- the integer a is a statistically unique and unpredictable number in the interval $[2,(q-2)]$.

Pertaining to Algorithm 8.18:

- the integer k is a statistically unique and unpredictable number in the interval $[2,(q-2)]$.
- in the decryption process step 2(a), compute $\gamma^{(q-a)} \bmod p$. This value is identical to $\gamma^{(-a)} \bmod p$.

If the message m that is encrypted with the ElGamal scheme has less than 511 bits, the message must be placed in the most significant position of a 511 bit block and the remaining least significant bit positions must be padded with pseudo random bits. The pseudo random bits may not be derived from the message or other secret system parameters.

2.1.2 Trust Authority Key Protection

The private key, $\mathbf{K_{PrvTA}}$, of the TA shall be kept in a well protected site. The location of this site shall be kept confidential. The private key shall be permanently burned into a semiconductor device. This mechanism protects against accidental erasure and/or malicious replacement. Use of the key shall be allowed only by use of a secret sharing

system requiring the physical presence of three (3) of the officers of the TA. Physical tokens containing the secret share shall also be used to make use of the private key. Dedicated hardware, which is not connected to any external network, will sign certificates and revocation notices.

The entire TA site including officers needed to enable the use of $\mathbf{K_{PrvTA}}$ should be replicated at a different location, preferably on a different continent. The private key shall be the same for each location. Redundant sites guarantee the safety of $\mathbf{K_{PrvTA}}$ as well as providing multiple locations for access in case of power failures, catastrophic weather conditions and the like.

2.1.3 Key Distribution

As shown in Figure 2, a set of public and private key pairs is generated at each CE manufacturer according to the specifications in Section 2.1.1. For efficient use of the TA, a large set of such key pairs should be generated at one time. The CE manufacturer shall keep the private keys, $\mathbf{K_{PrvCE[1, i]}}$, of this set secret. The CE manufacturer will keep the private keys secret out of self-interest since they will be revoked if they become known as described in Section 2.2.1.

The public keys of this set, $\mathbf{K_{PubCE[1, i]}}$, are sent to the TA via an authenticated channel. The public keys do not have to be kept secret; it is sufficient for the TA to know with certainty that the public keys received are the ones that were sent. A signature scheme could be used to send the public keys via the Internet but a reliable courier could be used as well.

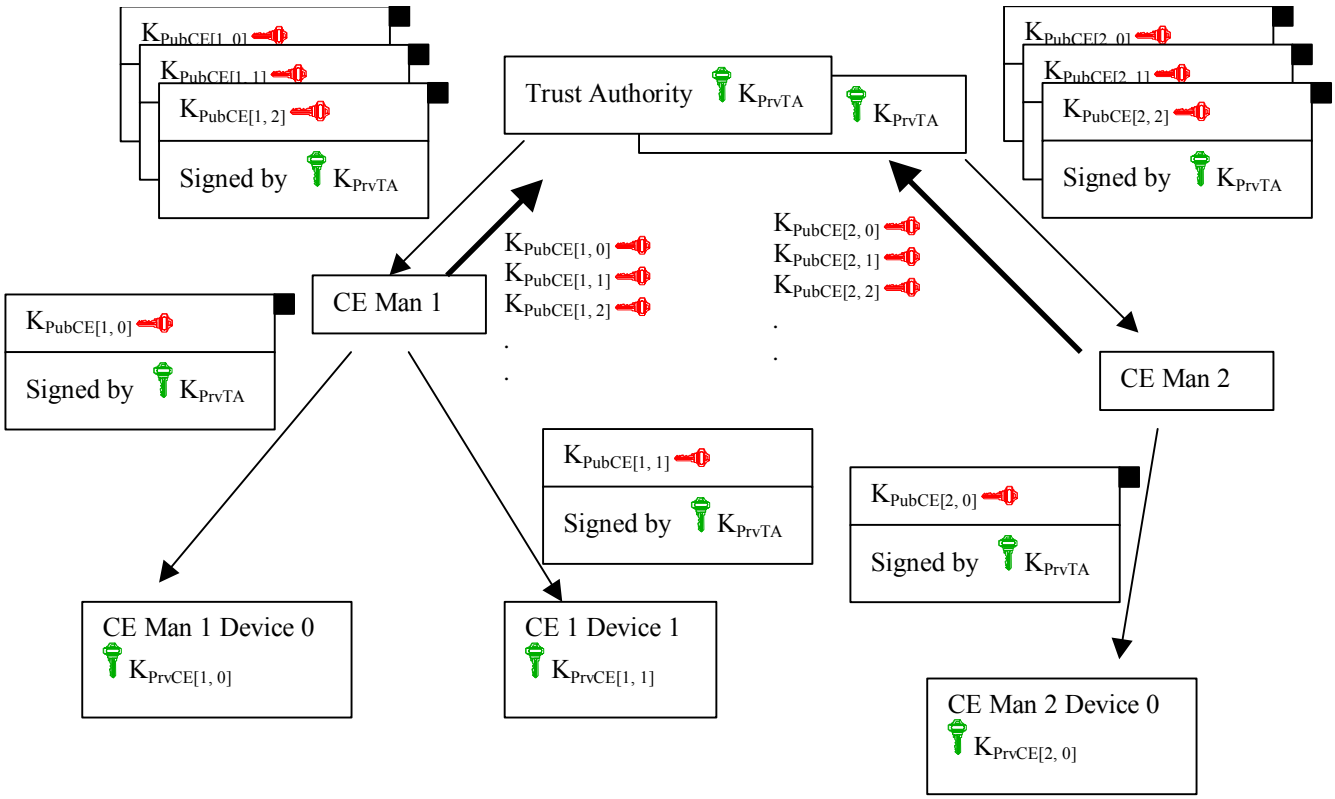


Figure 2 Key Distribution

Once the public keys of a CE manufacture are accepted as authentic, the TA will encapsulate each public key in a X.509 compliant certificate [X.509] signed with the private key of the TA, $\mathbf{K_{PrvTA}}$, as specified in Section 3.1.1. Care shall be taken to ensure that the private key of the TA is not at risk during this procedure. The signed certificates can then be sent over any channel to the CE manufacturer.

The CE manufacturer can then build devices securely storing a unique private key, $\mathbf{K_{PrvCE[1, x]}}$, and a certificate for the corresponding public key, $\mathbf{K_{PubCE[1, x]}}$ in each device. Note that the CE manufactures may then securely discard the certificate and the private key. In fact, keeping this information constitutes a liability for the CE manufacturer. Discarding the information relieves the manufacture of any tracking of keys. This completes the key distribution part of OCPS.

2.2 Key Revocation

Clearly, if a sufficiently able opponent attacks a CE device, the private key will be discovered. This private key can be used to clone the CE device while, at the same time, allowing improper use of content. In order to mitigate this potential attack, OCPS

provides a method for revoking public keys whose corresponding private key has been compromised.

2.2.1 Criteria for Key Revocation

Keys shall be revoked in only two circumstances:

1) The private key becomes public knowledge. Presentation of the actual private key to both the CE manufacture and the TA is required to prove this condition.

2) There are two or more CE devices that have the same public key even if the private key is not known. This shall be shown by the presentation of two CE devices that present identical public key certificates to both the CE manufacturer and the TA is required to prove this condition.

In either case, the public key will be revoked by means of a revocation certificate.

2.2.2 Key Revocation Process

When a private key of a single device manufactured by company **x**, $\mathbf{K_{PrvCE[x,\,i]}}$, is compromised, the proof of either criteria is formally presented to the both the TA and the CE manufacturer. The CE manufacturer shall be contractually required to respond to this presentation with an acknowledgement of the compromised public and private key pair. Contractual obligations shall specify under what circumstances the TA may revoke the public key, $\mathbf{K_{PubCE[x,\,i]}}$, corresponding to the compromised private key, without the acknowledgement of the manufacturer of the device.

Once a public key is designated to be revoked, a revocation notice $\mathbf{Cert_{TA}(K_{PubCE[x,\,i]},\ REVOKED)}$ is created. The date and time are as specified in Section 3.1.2. Care shall be taken to ensure that the private key of the TA is not at risk during this procedure. The signed revocation notice can then be sent over any channel to the CE devices in the field. Suggested methods are inclusion in commercial DVD disks and broadcasting over terrestrial and cable channels. The key revocation process is shown in Figure 3.

Any sink device will react to revocation notices when they are received. Any source device will broadcast the revocation notices to all devices on the bus when received from:

1) a real time source.

2) storage media upon insertion of the media. Note that power on with the media already inserted shall be considered a new insertion.

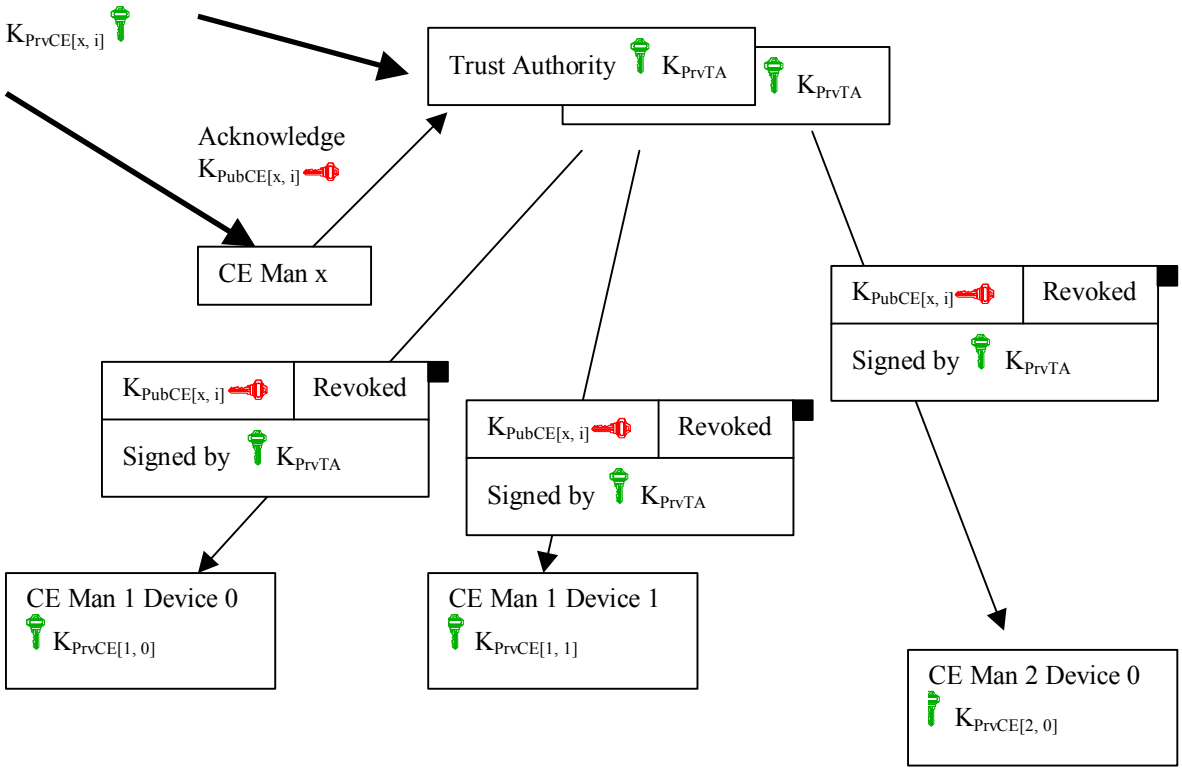


Figure 3 Key Revocation

3 Implementation

3.1 General Software and Hardware Implementation

3.1.1 Public Key Certificate Definition

An X.509 certificate as specified in [X.509] shall be stored within each device. Referring to [X.509], the following attributes will hold:

1) Version = V3
2) SerialNumber = Sequence from TA
3) Signature = DSA as specified in [FIPS 186]
4) Issuer = TA official name
5) Validity = null (the certificate is valid unless revoked)
6) Subject = CE manufacturer official name
7) SubjectPublicKeyInfo = {DSA as specified in [FIPS 186], bit string of the public key of the device}

No option fields will be used for these certificates.

In the signature created by the TA, the following attributes will hold:

1) AlgorithmIdentifier = DSA as specified in [FIPS 186]
2) Encrypted = bit string of the signature

3.1.2 Revocation Notice Definition

For a revocation notice, an X.509 certificate as specified in [X.509] shall be used. Referring to [X.509], the following attributes will hold:

1) Version = V3
2) SerialNumber = Sequence from TA
3) Signature = DSA as specified in [FIPS 186]
4) Issuer = TA official name
5) Validity = {(UTCTime[Time of creation], null} Note that in order to allow for multiple TA systems the choice of the type of time used will be UTCTime.
6) Subject = CE manufacturer official name
7) SubjectPublicKeyInfo = {DSA as specified in [FIPS 186], bit sting of the public key of the device being revoked}
8) Extension = Marked as critical. Value contains the ASCII representation of "REVOKED"

No other optional fields will be used for the certificate.

In the signature created by the TA, the following attributes will hold:

1) AlgorithmIdentifier = DSA as specified in [FIPS 186]
2) Encrypted = bit string of the signature

3.1.3 Protocol Implementation

Given the OCPS four phase protocol described above, we present the methods implementing the protocol. Where appropriate, we discuss the minimum parameters that should be used to ensure the security of the system.

Phase one specifies the exchanging of certificates, the use of a public key to validate the certificates and a lookup of the public key in a revocation list. The exchanged certificates shall be as specified in Section 3.1.1.

3.1.3.1 Hash Method

The third phase of the OCPS protocol specifies a hash. The hash method shall be the SHA-1 hashing method as specified in [FIPS 180-1].

3.1.3.2 Block Cipher

The fourth phase uses DES as the block cipher as specified in [FIPS 46-2] in CBC mode as specified in [FIPS 81]. The Initialization Vector (IV) shall always be zero.

3.1.3.3 Random Number Generator

The random number generator shall be as in Appendix 3 of [FIPS 186].

3.1.3.4 Session Key Period

The period between session key updates shall be 10 seconds.

3.1.4 Storage Requirements

Each sink or source device will contain the public key of the master certificate authority. The maximum size of this key shall be as specified in Section 2.1.1. This public key will be used to verify certificates that contain public keys and revocation notices. As part of the robustness of the device, the public key of any certificate authority will be in immutable storage such as ROM. Other comparable methods may be used to securely embed this public key in software. This prevents the replacement of the public key by the public key of an adversary.

Each sink or source device will contain storage for a public key encapsulated in an X.509 certificate as specified in [X.509]. The maximum number of bits of this key shall be as specified in Section 2.1.1. The public key will be unique for each AV device. This certificate is not required to be secured in any way.

Each sink or source device will have a secure way of storing a unique private key that corresponds to the unique public key of the AV device. The maximum number of bits of this key shall be as specified in Section 2.1.1. This key shall be secured in a robust manner as compromise of this key represents an effective attack on the copy protection system.

Each CE sink or source device is required to keep a revocation list of not less than 100 keys along with the corresponding date and time of the revocation. Only the public key and the date and time on the revocation notice is stored rather than the entire certificate. The maximum number of bits of each revoked key shall be as specified in Section 2.1.1. The maximum size of the date and time will be as specified for UTCTime in [X.509].

A source shall recognize a revocation notice, verify the signature of the notice, store the notice locally and transfer the notice to a connected sink device. A sink device shall respond to a source device transfer of a revocation notice, verify the notice and store the notice.

Storage of the notices shall first fill empty space in the revocation list. Subsequently, the date and time are examined to see if a revocation is more recent that the oldest revocation on the list. If the revocation notice is more current than the oldest revocation on the list

than the oldest revocation notice is removed from the list and the new revoked public is stored along with the data and time of the revocation notice. In the case of a notice that contains the same public key as a notice already in the stored revocation list, the notice with the newer date shall replace the older notice.

4 Robustness of Each Cryptographic Algorithm

4.1 Attack on the OCPS Block Cipher

The OCPS block cipher, DES, is the best studied symmetric algorithm in the public domain. The best theoretical attack against DES is known as linear cryptanalysis [Matsui]. Linear cryptanalysis requires 2^{43} known plaintext-ciphertext pairs generated under one key. However, this attack is not applicable to the OCPS key agreement protocol because of the periodic change of the DES encryption key.

The best known practical attack against DES is based on an exhaustive key search. In 1998, the first actual implementation of a key search machine has been reported [Gilmore]. The current state of the art is a machine that can find an individual session key in 4.5 days on average using custom hardware costing approximately US $250,000. The periodic change of the DES encryption key in the OCPS protocol results in 720 keys that have to be found to retrieve a two hour movie. Application of the key search machine [Gilmore] would yield an average search time of approximately 3240 days (about 9 years). A shorter key search time can only be achieved at considerably higher costs.

4.2 Security of the Initial Key Establishment

The protocol for key establishment is based on a modified Needham-Schroeder protocol as described in [Menezes, et al 1]. The protocol provides mutual authentication of Source and Sink. Under the assumption that an attacker cannot break the ElGamal encryption scheme, an attacker cannot learn any information about the partial keys $\mathbf{K_{RandSource}}$ and $\mathbf{K_{RandSink}}$ and thus about the session keys. Neither can an attacker trick Source or Sink in even starting an encryption or decryption process, respectively, with any of the messages during the initial key establishment phase.

One potential attack is against the certificate exchange and verification phase. In particular, an attacker capable of generating false certificates will be able to establish a communication session and successfully decrypt AV material. However, this would require breaking of the DSA algorithm with 1024 bit modulus and 160 bit subgroup. The best known attacks are either the index calculus method against the 1024 bit modulus, or one of the square root attacks (Pollard's rho method or one of its parallized derivatives) [Menezes, et al 3]. Both attacks require 2^{80} operations which is not feasible with current technology, neither with supercomputers nor special-purpose hardware. Similarly, attacking the hash function, SHA-1, by finding a collision with an incorrect public key as a hash input would require about 2^{80} operations.

Another potential point of attack is against one of the individual ElGamal encryption algorithms. This proposal uses ElGamal with 512 bit modulus and a 160 bit subgroup. The best known attack against this set-up is the index calculus attack against the modulus. The best reported implementation of this attack was able to break a discrete logarithm problem (in a prime field) with approximately 282 bit (85 decimal digits) [Weber]. Although it appears to be principally possible to run a successful index-calculus attack against a 512 bit modulus, such an attack will require massive computational resources, presuming several months of computations, and expert knowledge in algorithmic number theory. This assumption is based on the resources and efforts used against the RSA130 challenge in 1996 [Lenstra], which poses a computationally somewhat smaller, but comparable problem. It seems highly unlikely that a casual hacker will be able to succeed with such an attack. Moreover, this document specifically dictates different ElGamal moduli for every device. As a consequence, breaking of one device would not lead to a catastrophic system failure. If a hacker would start cloning devices based on the single ElGamal broken key, this behavior will most likely be detected and will result in a revocation of the ElGamal public key. Note that breaking of another ElGamal key will require the same massive effort.

4.3 Security of the Key Derivation Protocol

The key derivation protocol is based on the one specified in the X9.42 ANSI draft standard Subsection 7.7. All attacks against the session key derivation protocol have to exploit the SHA-1 hash function. There are no weaknesses known against this hash function. The most attractive attack would try to first recover one of the session keys (see Section. 4.1) and from there recovering the initial keys $\mathbf{K}_{\mathbf{RandSource}}$ and $\mathbf{K}_{\mathbf{RandSink}}$. However, this would mean an inverting of the SHA-1 hash process, a method for which is completely unknown.

5 Error propagation Characteristics of the Encryption Algorithm

5.1 Single Ciphertext Error in the Block Cipher

The block cipher used in this proposal is DES in Cipher Block Chaining (CBC) mode (see section 12). The effect of an error in the ciphertext for a block cipher in CBC mode is well known [FIPS 81]. In the case of DES, for each single bit error in the ciphertext, one block (64 bits) will be garbled. In addition, the following block will have a single bit error in the same location in the block as the ciphertext error. All other blocks will remain unaffected since CBC mode is self-recovering.

6 Renewability

6.1 Revocation is Renewability

Fundamental to any real copy protection scheme is a secret embedded in the hardware of the consumer electronic device. Without such a secret, a man in the middle attack is unavoidable. Once this concession is made, the algorithms and key lengths chosen will be long lived and should be resistant to obsolescence as indicated in Section 7.

The only part of the system that can be renewed in an inexpensive consumer product is the expulsion of compromised product. In this case, we are concerned that an attacker can expose the private key of a certified consumer device. Once a key is known, the protocols can be attacked and the content copied directly from the serial link. Therefore, this system provides a method of revoking exposed keys. Since every device contains a unique key, only the consumer device that has been attacked is disabled. Such an approach renews the existing devices by providing a system that can still be relied upon for the transfer of content. This system therefore provides the ultimate in renewability: the continued use of the consumer device without any interaction required by the consumer.

7 Resistance to Obsolescence

7.1 Resistance to Improved Attacks

All of the cryptographic algorithms used in this proposal are used by banking and other crucial industries. As such, the very best cryptographers have attacked them without any significant results. Each of these algorithms is based on well-studied mathematical problems such that real breakthroughs are unlikely. There is no better metric than these principles when it comes to expecting an algorithm to resist future attacks.

7.2 Resistance to Improved Computer Speed

Assuming that no better methods then those known today and described in Section 4 are developed to attack the cryptographic algorithms, the only remaining threat is the increase of computational resources of an attacker. In each case the amount of resources applied to attacking the system will be a function of the size of the keys used. In our case we are limited in key size by cost and the ability to export the complete system. Nevertheless, the key sizes chosen here are acceptable given our targeted adversaries. Of course, as export controls are eased larger key sizes can be applied to this system thus strengthening the system on an as needed basis.

8 Maintenance Complexity

8.1 Maintenance of the System

The only required maintenance of the system is generation of revocation notices for cause as detailed in Section 2.2. Interested parties such as the Motion Picture Association of America (MPAA) will most likely generate proof of such violation. The only cost of the system will be the generation of the revocation notices. Note that no database of keys is maintained other than a list of public keys for revoked devices. Distribution of the notices will be borne by the owners of material in the form of space on DVD disks or the purchase of bandwidth on broadcast systems. All of such cost are amortized over the entire installed base and would therefore be reasonably contained.

There is also the requirement for consumer devices to accept legitimate revocation notices. Since revocation notices will distributed along with other material and absorbed over time, the consumer will bear no additional direct costs.

9 Applicability to Different Digital Interfaces

The system as specified will operate with any generic sink or source terminal on a digital bus. There are no restrictions on the devices save that they be compliant with this system.

10 Availability for US Import/Export

The following cryptographic functions are used in this proposal.

1) DES (56 bits)
2) DSA (1024 bit signature verification)
3) ElGamal Encrypt/Decrypt (512 bits)
4) SHA-1

DES is exportable under a mass market license. All others are exportable under current US export laws in binary form (e.g. within an IC) without a license.

11 Licensing Terms

The licensing term will be as per described in the OCPS license agreement.

12 Block Cipher Mode

The OCPS block cipher will consist of the DES block cipher in CBC mode. The CBC mode uses the previous ciphertext to XOR with the current plaintext prior to encryption. The CBC flow is shown in Figure 4.

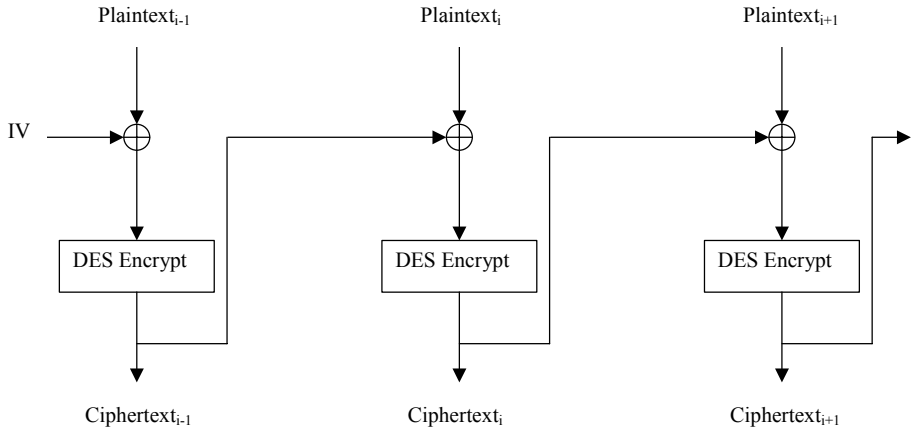


Figure 4 CBC Stream Cipher

13 Circumvention Devices

13.1 Defeating Circumvention

Building a circumvention device would require either an unexpected attack against a trusted cryptographic algorithm or the cloning of an existing device. The former is an unlikely event as detailed in Section 4. Therefore, we will restrict the discussion to a circumvention device that is built using a key acquired from an existing device.

The OCPS protocol used in this proposal requires an authenticated public key and knowledge of the corresponding private key. Without the certified public key, any compliant device will refuse to communicate with the circumvention device. Without the private key, the circumvention device will not be able to decrypt the material. Therefore, the circumvention device will have to acquire a set of keys to communicate with a compliant device.

Acquiring the keys will be difficult but not impossible. However, whenever a significant number of circumvention devices are on the market with the same compromised key, that key will scheduled for revocation. This key will then be revoked as specified in Section 2.2. The circumvention device using this key will then become useless as the revocation notices spread across the existing base of compliant devices. Such a strategy devalues all circumvention devices, as it becomes clear to the consumer market that such devices are unreliable.

14 Amendments Needed to Interface Standards

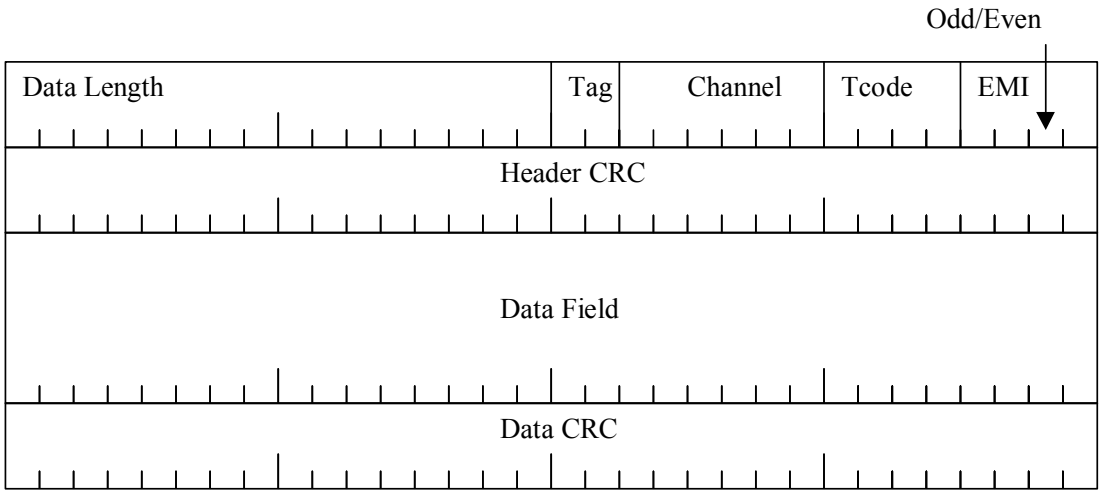
14.1 Specifics Relating to the IEEE 1394 Bus.

The application of this proposal to the IEEE 1394 bus [IEEE 1394] requires some detailed explanation of packet structures and protocol modifications. Actual modification of the standard is not needed.

All key exchange messages will be asynchronous write packets. The 1394 write address used for certificates is beyond the scope of this proposal. The key generation phase of the OCPS will initially generate two session keys $\mathbf{K}_{\mathbf{Session[0]}}$ and $\mathbf{K}_{\mathbf{Session[1]}}$.

Figure 5 Isochronous Packet Odd/Even Bit Placement

Once the key exchange is complete a channel will be opened for isochronous packets to



transfer the encrypted data. The encryption key used will be the current session key $\mathbf{K_{Session[i]}}$. The packets in the isochronous stream will contain an odd/even bit in the EMI bits of the isochronous packet as in Figure 5. When the odd/even bit changes state a new encryption key, $\mathbf{K_{Session[i+1]}}$, will be used on the following packet. At that time, a new encryption key, $\mathbf{K_{Session[i+2]}}$, will be computed to prepare for the next such change as shown in Figure 6.

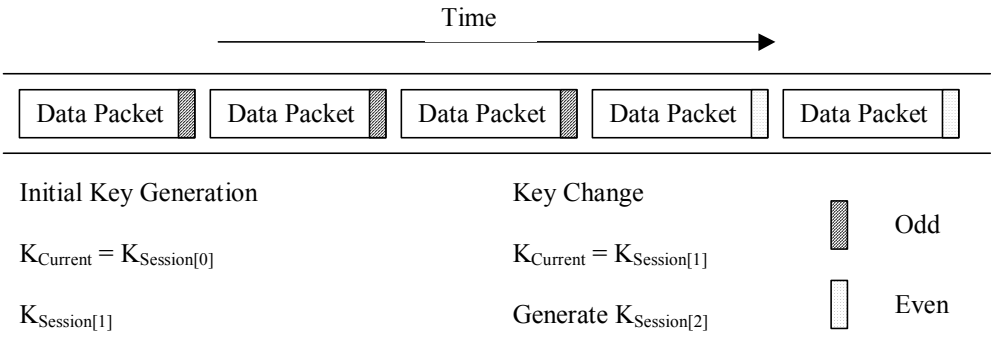


Figure 6 Session Key Change

All revocation notices will be asynchronous write packets. A second 1394 address, the exact specification of which is beyond the scope of this proposal, will be used for the revocation notices.

15 View of Submitter Regarding Standardization of Copy Protection

15.1 The Long Term View on Copy Protection

This proposal accomplishes rigorous protection for copyright holders across a serial link. While the specifics target the 1394 serial interface, the protocol and system architecture applies equally well to any bi-directional interface (i.e. USB, etc.). Such a system is not a complete copy protection solution. To be complete a copy protection solution must specify conditions for a number of areas within a consumer electronics architecture and ensure that they are an integrated whole. In addition, the solution should include a rigorous specification of the level of adversary to be defeated.

This proposal has only taken in the scope of an attacker that is an ordinary consumer without specialized knowledge or equipment. Even so, some effort should be expended to protect against more organized attacks when it is feasible. Given these pre-conditions the following areas in consumer electronics architecture must be protected:

1) The transmission of material between sinks and sources.
2) The movement of material within the source or sink.
3) The storage of material on media, such as disks or tapes.

This proposal addresses the only the first item: the protection of material between sources and sinks. Even in this area, the protection only extends to digital transmission and not to analog transmission methods such as NTSC or RGB. Up to this point in time, the protection method of NTSC has been the Macrovision method. This method is still

adequate. But inexpensive circumvention devices are a problem for Macrovision. Both this problem and the difficulties with storage can be addressed by the application of watermarking technology. Such a solution places a non-removable mark in the material and establishes the rules for use. On compliant devices such a system will protect analog transmission by allowing the display of watermarked materials but not allowing copying.

Watermarks may also protect the storage of material. Should a non-compliant device generate a recording, a compliant device will reject such a recording as unplayable. Such a system is imperfect in that non-compliant devices can ignore the watermark, much the same way that the Macrovision system is defeated today. However this would still serve the purpose of keeping many ordinary consumers from making illegal copies.

Protecting material inside a consumer device is beyond the scope of the attackers we wish to protect against. Opening up a consumer device and extracting key material requires skills beyond the ordinary consumer. However, the construction of devices without easy access to internal busses is a reasonable precaution. Any other types of precautions would either quickly be proven useless or increase the costs of consumer devices by an unreasonable amount. The exceptions to such precautions are means required to protect keys within consumer devices as specified in Section 3.1.4. In this case, the use of specially constructed secure processing chips would be a reasonable approach.

In summation, a complete copy protection system is outside the scope of this proposal. This proposal should work well with other standards to provide a complete solution.

16 Other Information

No other information at this time.

Appendix A References

[ANSI X.509] International Telecommunications Union, *Recommendation X.509: The Directory – Authentication Framework.*, 1988.

[ANSI X9.42] American Bankers Association, *Public Key Cryptography for The Financial Service Industry: Agreement of Symmetric Keys on Using Diffie-Hellman and MQV Algorithms*, Working Draft, 1998.

[FIPS 46-2] Federal Information Processing Standards, Data Encryption Standard (DES), FIPS Publication 46-2, http://www.itl.nist.gov/div897/pubs/fip46-2.htm, 1993.

[FIPS 81] Federal Information Processing Standards, DES Modes of Operation, FIPS Publication 81, http://www.itl.nist.gov/div897/pubs/fip81.htm, 1980.

[FIPS 180-1] Federal Information Processing Standards, Secure Hash Standard, FIPS Publication 180-1, http://www.itl.nist.gov/div897/pubs/fip180-1.htm, 1995.

[FIPS 186] Federal Information Processing Standards, Digital Signature Standard (DSS), FIPS Publication 186, http://www.itl.nist.gov/div897/pubs/fip186.htm, 1995.

[Gilmore] Electronic Frontier Foundation, *Cracking Des: Secrets of Encryption Research, Wiretap Politics & Chip Design*, J. Gilmore, Editor, O'Reilly & Associates, ISBN: 1565925203.

[IEEE 1394] IEEE Standard for a High Performance Serial Bus, IEEE Std 1394-1995, IEEE, August 30, 1996, ISBN 1-55937-583-3

[Lenstra] A. K. Lenstra, "Posting to sci.crypt", April 11, 1996.

[Matsui] M. Matsui, *Linear Cryptanalysis for DES Cipher,* Advances in Cryptography – EUROCRYPT '93 Proceedings, Springer-Verlag 1994, pp. 386-397.

[Menezes, et al 1] A. Menezes, P van Oorschot, S Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997, p. 508 and note 12.39.

[Menezes, et al 2] A. Menezes, P van Oorschot, S Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997, pp. 294-296.

[Menezes, et al 3] A. Menezes, P van Oorschot, S Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997, pp. 106-107.

[Philips] Philips Semiconductors, *FAMEX Backgrounder* http://www.semiconductors.philips.com/identification/products/famexbg/famexbgp3.stm, Philips Electronics N.V.

[Weber] D. Weber, *Computer Discrete Logarithms with Quadratic Number Rings*, Advances in Cryptography - Eurocrypt '98, Lecture Notes in Computer Science, Springer Verlag.