This taxonomy helps describe and categorize the escrow mechanisms of complete systems along with various design options.

A Taxonomy for Key Escrow Encryption Systems

Dorothy E. Denning and Dennis K. Branstad

A *key escrow encryption system* (or *escrowed encryption system*) is an encryption system with a backup decryption capability that allows authorized persons—users, officers of an organization, and government officials—under certain prescribed conditions, to decrypt ciphertext with the help of information supplied by one or more trusted parties holding special data recovery keys. The data recovery keys are not normally the same as those used to encrypt and decrypt the data, but rather provide a means of determining the data encryption/decryption keys. The term *key escrow* is used to refer to the safeguarding of these data recovery keys. Other terms used include *key archive*, *key backup*, and *data recovery* system.

This article presents a taxonomy for key escrow encryption systems, providing a structure for describing and categorizing the escrow mechanisms of complete systems as well as various design options. Table 1 applies the taxonomy to several key escrow products or proposals. The sidebar, "Glossary and Sources," identifies key terms, commercial products, and proposed systems.

Components

An escrowed encryption system can be divided logically into three main components:

User Security Component (USC). The USC is a hardware device or software program that provides data encryption and decryption capabilities as well as support for the key escrow function. This support can include attaching a *data recovery field* (DRF) to encrypted data. The DRF may be part of the normal key distribution mechanism.

Key Escrow Component (KEC). The KEC, which is operated by *key escrow agents,* manages the storage and release or use of data recovery keys. It may be part of a public key certificate management system or part of a general key management infrastructure.

Data Recovery Component (DRC). The DRC consists of the algorithms, protocols, and equipment needed to obtain the plaintext from the ciphertext plus information contained in the DRF and provided by the KEC. It is active only as needed to perform a specific authorized data recovery.

These logical components are highly interrelated, and the design choices for one affect the others. Figure 1 shows the interaction of the components. A USC encrypts plaintext data with a key K and attaches a DRF to the ciphertext. The DRC recovers the plaintext using information contained in the DRF plus information provided by the KEC.

Each of these components is described in the following sections.

User Security Component

The USC encrypts and decrypts data and performs functions that support the data recovery process. It is characterized by the following:

• **Application Domain.** A USC can support one or both of the following:
- *Communications.* This includes phone calls, electronic mail, and other types of connections. Emergency decryption is used by law enforcement in conjunction with court-authorized interception of communications, also known as wiretaps.
- *Stored data.* Stored data can be simple data files or more general objects. Emergency decryption is used either by the owners of the data to recover lost or damaged keys, or by law enforcement officials to decrypt computer files seized under a court order.

• **Data Encryption Algorithm.** The following attributes are particularly relevant to escrowed encryption:
- *Name and mode of operation.* Mode of operation can affect exportability, so, for example, triple encryption modes may not be allowed under a general export license.
- *Key length.* This can also affect exportability.
- *Classification.* An algorithm may be classified or unclassified; if unclassified, it may be proprietary or public.

• **Stored Identifiers and Keys.** The USC stores identifiers and keys used for emergency decryption:
- *Identifiers.* These can include a user or USC identifier, identifiers for keys, and identifiers for the KEC or escrow agents.
- *Keys.* These can include keys unique to the USC, keys belonging to its user, or global system keys used by the KEC. They can be public or private. Copies of the keys or their private counterparts may be held in escrow.

• **DRF and Mechanism.** When data are encrypted with a key K, the USC must bind the ciphertext and K to one or more data recovery keys, normally by attaching a DRF to the encrypted data. The binding is characterized by:
- *Whose data recovery keys.* K can be bound to keys held by the escrow agents of the sender, the receiver, or both. The choice affects data recovery.
- *Role in key distribution.* The DRF and binding mechanism can be integrated with the protocols used to transmit K to the intended recipient. In that case, the sender must transmit a valid DRF in order for the intended recipient to acquire the key.
- *Contents of DRF.* Normally, the DRF contains K encrypted under one or more data recovery keys (e.g., a product key, the public key of the sender or receiver, or a master public key of the KEC).

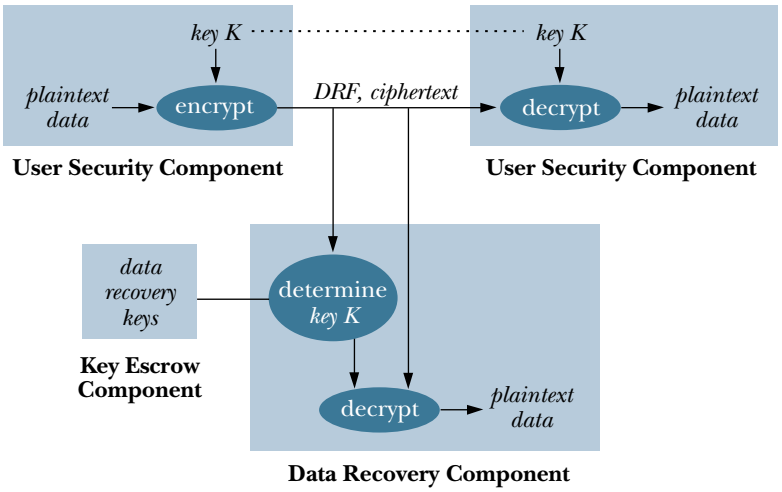

Figure 1. Key escrow encryption system

Table 1. Summary characteristics of key escrow encryption systems and approaches

Key Escrow System or Approach (* is commercial product)	User Security Component (USC)					Key Escrow Component (KEC)					Data Recovery Component (DRC)	
	App	Ene Alg	Keys	DRF	Imp	Role	Type	Esc Keys	Split	Service	Keys Req	Per
AT&T Crypto Backup	f,c	Ur	pub	pub	S		C	master	1,k/n	dec K	S	K
Bankers SecureKEES*	c	U	priv	pub,k	H		C	user	k/n	rel KU	S/R	S/R
Bell Atlantic Yaksha	c,f	U	priv	na		KMI	C	session	1	rel K		K
Blaze File Crypto	f	U	none	na	H		C	dir	1	dec file	S	K
Clipper Chip (EES)*	c	C	priv	priv	H		G	prod	2/2	rel KU/exp	S	S
Cylink Key Escrow	c,f	U	priv	pub,k		PKI	C	user	1,k/n	rel KU	S/R	S/R
Desmedt Traceable	c	U	priv	pub,k				user			R	R
Fortezza Card*	c,f	C	priv	pub	H		C	user	1	rel KU	R	R
Fortress KISS	c	U	priv	pub,k	H	PKI	G	master	2/2	dec KU	S/R	S/R
Kilian/Leighton F-safe	c	U	priv	pub,k				user	k/n	rel KU		
Leiberich TB-Clipper	c	C	priv	priv	H/c		G	prod	2/2	rel KU/tb	S	S
Leighton/Micali		U	priv	na				prod		rel KU/K	S/R	S/R
Lenstra/Winkler/Yacobi	c	U	priv	pub		PKI		user	k/n	rel KUV/tb	S/R	S/R
Lotus Notes Int'l*	c	U	pub	pub	S		G	master		dec partial K	S	K
Micali Fair Crypto	c	U	priv	pub,k		PKI		user	k/n	rel KU/tb	R	R
Micali Partial Escrow								partial				
Micali/Sidney Esc.		U	priv	priv				user	t/u/n	rel KU		
National CAKE	f,c	U	pub	pub	H		C	master	1	dec K	S	S
Nechvatal Public-Key	c		pub	pub				prod	n/n	rel KU	S	S
Nortel Entrust*	c,f	Ur	priv	pub,k	H,S	PKI	C	user	1	rel KU	S/R	S/R
PC Sec. Stoplock KE*	c,f	Ur	priv		S	KMI	C	system				
Royal Holloway TTPs	c	U	priv	priv,k		PKI	C	user		rel KU	S/R	S/R
RSA Secure*	f	Ur	pub	pub	S		C	master	k/n	dec K	S	K
Shamir Partial Escrow			priv					partial		rel		
TECSEC VEIL*	f	U	priv	priv,k	S	KMI	C	system	n/n	rel K	S/R	K
TESS w Key Escrow	c	U	priv	na	H	PKI	C	user	any	rel KU	S/R	S/R
ThresholdDecryption	c	U	priv	pub				user	k/n	th-dec K	S?	S?
TIS Comm. Key Esc.*	f,c	U	pub	pub			C	master	1	dec K	S	K
TIS Software Clipper	c	U	pub	pub	S			prod	2/2	rel KU	S	S

User Security Component (USC)
• App = application: c = communications; f = files and other stored objects.
• Enc Alg = data encryption algorithm: C = classified; U = unclassified; Ur = proprietary unclassified.
• Keys = stored keys used with key escrow function: priv = private keys and optionally public keys; pub = public keys only.
• DRF = encryption keys used to compute Data Recovery Field: priv = private keys (and, optionally, pubic keys); pub = public keys; k = DRF also used with key establishment/distribution; na = not applicable.
• Imp = implementation: H = some special hardware required; H/c = hardware with a clock; S = software with optional hardware.

Key Escrow Component (KEC)
• Role = integration of key escrow into key management infrastructure: KMI = integrated with key management infrastructure; PKI = component of public key infrastructure administered by certificate authorities.
• Type = type of system: C = keys held by commercial or private sector escrow agents; G = keys held by government.
• Esc Keys = keys stored in escrow: dir = file encryption key used with entire directory; master = escrow agent master key; partial = part of user or application key; prod = product unique key; session = session key; system = keys managed by system; user = user key.
• Split = splitting of keys with escrow agents: n/n = n out of n needed for decryption; k/n = k out of n needed using threshold techniques; t/u/n = allows t to conspire and compromise key and n-u to withhold.
• Service = service provided to DRC: dec K = decrypt data encryption key K; rel K = release K from escrow; thd-dec K = use threshold decryption; dec KU = decrypt user or product key; rel KU = release KU from escrow; rel KUV = release keys used by pair of users U and V; tb = time-bounded keys released; exp = keys released with expiration date.

Data Recovery Component (DRC)
• Keys Req = keys required for decrypting data: S = keys associated with the sender or the sender's USC; R = keys associated with the receiver or the receiver's USC; S/R = keys associated with either sender or receiver.
• Per = frequency with which DRC must interact with KEC to get keys: K = once per session/file key; S = once per sender; R = once per receiver.

Blanks in table denote open or unspecified elements.

In some cases, only some of the bits of K may be made available through the DRF so the remaining bits must be determined through brute force. The DRF also contains information identifying the data recovery keys, the KEC or key escrow agents, the encryption algorithm and mode, or the DRF creation method. The entire DRF may be encrypted under a *family key* associated with the DRC in order to protect identifiers transmitted in the DRF. Single-key or public-key cryptography can be used. The length of the DRF can affect the suitability of a particular scheme to certain applications (e.g., radio communications) where error rates are high.

– *Transmission and frequency.* Normally, the DRF precedes the ciphertext in a message or file header. With open connections, it may be retransmitted at regular intervals.

– *Validation.* The DRF may include an *escrow authenticator* verified by the receiver to determine the integrity of the DRF. Alternatively, if public keys are used to create the DRF, the receiver could recompute the DRF and compare the result with the DRF received.

• **Interoperability.** A USC may be designed to interoperate only with correctly functioning USCs and not with USCs that have been tampered with or that do not support key escrow.

• **Implementation.** A USC may be implemented in hardware, software, firmware, or some combination thereof. Hardware is generally more secure and less vulnerable to modification than software. If classified algorithms are used, they must be implemented in tamper-resistant hardware. Hardware implementations may include special-purpose cryptoprocessors, random number generators, and/or a high-integrity clock. Products that implement a USC are sometimes called *escrowed encryption products* (or devices). They have also been called *escrow-enhanced* or *escrow-enabled products.*

• **Assurance.** The USC may provide assurance that users cannot circumvent or disable the key escrow mechanisms or other features. A USC that can be used or modified to "cheat" is called a *rogue* USC. The possibility of rogue USCs is strongly dependent on the data recovery mechanism and implementation. We distinguish between *single rogues,* which can interoperate with non-rogues, and *dual rogues,* which interoperate only with other rogues. Single rogues present the greatest threat to emergency data recovery because they require no collaboration on the part of the receiver.

Key Escrow Component

The KEC is responsible for storing all data recovery keys and for assisting the DRC by providing required data or services. It has the following elements:

• **Role in Key Management Infrastructure.** The KEC could be a component of the *key management infrastructure,* which could be a *single-key infrastructure* (e.g., a *key distribution center*) or a *key infrastructure.* With the latter, the agents could serve as the *public-key certi*

• **Escrow Agents.** The escrow agents, also called *trusted parties,* are responsible for operating the KEC. They may be registered with a *key escrow center* that coordinates their operation or serves as a point of contact for the USC or DRC. Escrow agents are characterized by:

– *Type of agents.* Escrow agents may be entities in the government or private sector. The former could restrict use of their services to government agencies. The latter, which are used with what are called *commercial* or *private key escrow systems,* could be internal to an organization or to independent companies offering commercial services, or to trusted third parties.

– *Identifiability.* This includes name and location.

– *Accessibility.* This is determined by the location of the escrow agents (e.g., local or foreign) and their hours of operation (e.g., 24 hours a day, 7 days a week).

– *Security.* This refers to how well the KEC protects against compromise, loss, or abuse of escrowed keys. It includes *reliability* and *resiliency,* which is a measure of the "trust" required of the escrow agents for protecting the escrowed keys from compromise and for enabling data recovery.

– *Accountability.* This assures identification of an escrow agent that sabotages data recovery or that releases keys to unauthorized parties or releases them under unauthorized circumstances.

– *Liability.* This characterizes the liability of the escrow agents in case keys are compromised or become unavailable. Escrow agents might be *bonded* to protect against liability.

– *Certified/licensed.* This indicates whether the escrow agents are certified and licensed with a government. To qualify for a license, escrow agents may be required to meet specified conditions. Use of certified agents may affect exportability.

• **Data Recovery Keys.** With escrowed encryption, all encrypted data are bound to escrowed data recovery keys that enable access to the data encryption keys. The data recovery keys are characterized by:

– *Granularity of keys.* Options include:

a. *Data encryption keys.* This includes session keys, network keys, and file keys. A key distribution center could generate, escrow, and distribute such keys.

b. *Product keys.* These are unique to a USC.

c. *User keys.* Normally, these would be public-private-key pairs used to establish data encryption keys. The KEC might serve as the user's public-key certificate authority, issuing a certificate for the user's public key.

d. *Master keys.* These keys are associated with

the KEC and used by multiple USCs.
- *Splitting of keys (secret sharing, threshold schemes).* A data recovery key can be split into multiple key components, with each component held by a separate agent. Keys can be split so that all n escrow agents are needed to restore a given key or so that any "k out of n" for some k, where n is the number of agents, suffices. They can be split using a *general monotone access structure,* allowing for the specification of arbitrary subsets of escrow agents that can work together to restore a key.
- *Who generates and distributes keys.* Keys can be generated by the KEC, the USC, or a combination of both. If generated by the USC, the keys may be split and escrowed using *verifiable secret sharing* schemes so that the escrow agents can check the validity of their individual components without knowing the original key. Keys may be generated jointly so a user cannot hide a "shadow key" in an escrowed key and thereby circumvent the key escrow mechanism.
- *Time of escrow.* Keys could be escrowed during product manufacture, system or product initialization, or user registration. If a user's private key (of a public-private key pair) is escrowed, it could be escrowed when the corresponding public key goes into the public-key infrastructure and a certificate is issued. A USC might send encrypted data only to users with public-key certificates signed by approved escrow agents.
- *Key update.* Some systems may allow data recovery keys to be changed. Such updates could be performed on request or on a regular basis.
- *Complete or partial.* A portion of a key could be escrowed instead of the complete key. In this case, the unescrowed portion of the key would be determined through a brute force attack when it is needed for data recovery.
- *Storage of keys.* This could be off-line (e.g., on floppy disks stored in safes or smartcards) or on-line.

• **Data Recovery Services.** The KEC provides services, including release of information, to the DRC characterized by:
- *Authorization procedures.* The procedures under which people operating or using the DRC can use the services of the KEC may include establishing proof of identity and legal authority to access the data to be decrypted.
- *Services provided.* There are several possible options:
  a. *Release data recovery keys.* This approach is normally used when the data recovery keys are session keys or user or product keys (master keys are not released). The keys might be released with an expiration date, after which they are automatically destroyed.
  b. *Release derived keys.* The KEC releases derivatives of data recovery keys, such as *time-bounded keys* that enable decryption only of data encrypted during a specific period of time.
  c. *Decrypt key.* This approach is normally used when master data recovery keys are used to encrypt data encryption keys (or user keys) in the DRF so the KEC need not release the master keys to the DRC.
  d. *Perform threshold decryption.* Each escrow agent provides a "piece" of a decryption to the DRC, which combines the results to get the plaintext.
- *Transmission of data* to and from the DRC, either manually or electronically.

• **Safeguards for Escrowed Keys.** The KEC employs safeguards to protect against compromise or loss of keys. These can include a combination of technical, procedural, and legal safeguards. Examples are auditing, separation of duties, split knowledge, two-person control, physical security, cryptography, redundancy, computer security, trusted systems, independent testing and validation, certification, accreditation, configuration management, and laws with penalties for misuse.

Data Recovery Component

The DRC supports recovery of plaintext from encrypted data using information supplied by the KEC and in the DRF. It is characterized by:

• **Capabilities.** These include:
- *Timely decryption.*
- *Real-time decryption of intercepted communications.*
- *Post-processing.* The DRC can decrypt communications previously intercepted and recorded.
- *Transparency.* Decryption is possible without the knowledge of the parties involved.
- *Independence.* Once the keys are obtained, the DRC can decrypt using its own resources, that is, independently of the KEC.

• **Data Encryption Key Recovery.** To decrypt data, the DRC must acquire the data encryption key K in the following ways:
- *Access through sender or receiver.* A critical factor is whether k can be recovered using data recovery keys associated with the sender, the receiver, or either party. If access is possible only through keys held by the sender's escrow agents, the DRC must obtain key escrow data for all parties transmitting messages to a particular user, possibly precluding real-time decryption, especially if the parties are in different countries and using different escrow agents. Likewise, if access is possible only through keys held by the receiver's escrow agents, real-time decryption of all messages transmitted from a particular user may be impossible. If data recovery is possible using keys held by either set of escrow agents, the DRC can decrypt intercepted communications both to and from a particular USC in real time once the key used by that USC is obtained. A system may provide this capability for two-way simultaneous communications

(e.g., phone calls) by requiring that the same K be used for both ends of the conversation.
– *Frequency of interaction with KEC.* The DRC may be required to interact with the KEC once per data encryption key or once per USC or user. The former requires an on-line connection between the DRC and the KEC to support real-time decryption of communications when the session key changes per conversation.
– *Need for brute force.* If the escrow agents return partial keys to the DRC, the DRC must use brute force to determine the remaining bits.

• **Safeguards on Decryption.** The DRC can use technical, procedural, and legal safeguards to control what can be decrypted. For example, data recovery may be restricted to a particular time period (e.g., as authorized by a court order). These safeguards supplement restrictions imposed by the KEC in its release of keys. Authentication mechanisms could be used to prevent the DRC from using the keys it acquires to create and substitute bogus messages.

Acknowledgments
We wish to thank Matt Blaze, Yvo Desmedt, Carl Ellison, Ravi Ganesan, Carmi Gressel, Hans-Joachim Knobloch, David Maher, Silvio Micali, Edward Scheidt, Greg Shanton, and Peer Wichmann for helpful comments on an earlier version of this taxonomy.

About the Authors:
DOROTHY E. DENNING is a professor of Computer Science at Georgetown University. **Author's Present Address:** Georgetown University, Department of Computer Science, 225 Reiss Science Building, Washington, DC 20057; email: denning@cs.georgetown.edu

DENNIS K. BRANSTAD is Director of cryptographic technologies at Trusted Information Systems, Inc. **Author's Present Address:** Trusted Information Systems, 3060 Washington Road, Glenwood, MD 21738; email: dbranstad@tis.com

Glossary and Sources

AT&T Crypto Backup. This is a proprietary design for a commercial system that backs up document keys through an escrowed master key. See David P. Maher, "Crypto Backup and Key Escrow," in this issue of *Communications of the ACM.*

Bankers Trust Secure Key Escrow Encryption System (SecureKEES). Employees of a corporation register their encryption devices (e.g., smartcards) and private encryption keys with one or more commercial escrow agents selected by the corporation. See SecureKEES product literature, CertCo, Bankers Trust Co.

Bell Atlantic Yaksha System. An on-line key security server generates and distributes session keys and file keys using a variant of the RSA algorithm. The server transmits the keys to authorized parties for data recovery purposes. See Ravi Ganesan, "The Yaksha Security System," in this issue of *Communications of the ACM.*

Blaze's Smartcard-Based Key Escrow File System. This is a prototype smartcard-based key escrow system for use with the Cryptographic File System. A user escrows a file encryption key on a smartcard entrusted with an escrow agent. See Matt Blaze, "Key Management in an Encrypting File System," AT&T Bell Laboratories.

The Clipper/Capstone Chips. These tamper-resistant chips implement the Escrowed Encryption Standard (EES), which uses the classified Skipjack algorithm. Unique data recovery keys, programmed onto each chip, are split between two government agencies and restricted to government use. See Dorothy E. Denning and Miles Smid, "Key Escrowing Today," *IEEE Communications,* Vol. 32, No. 9, Sept. 1994, pp. 58–68.

Cylink Key Escrow. This proposal uses Diffie-Hellman techniques for integrating key escrow services into a public-key infrastructure. See Jim Omura, "Alternatives to RSA Using Diffie-Hellman with DSS," white paper, Cylink, Sept. 1995.

Desmedt Traceable Ciphertexts. This proposal binds the DRF to ciphertext in such a way that the identity of the receiver can be determined if the receiver can determine the session key. See Yvo Desmedt, "Securing Traceability of Ciphertexts—Towards a Secure Software Key Escrow System," *Proceedings of Eurocrypt '95,* Saint-Malo, France, May 21–25, 1995, pp. 147–157.

Fortezza Card. This commercially available PC card contains a Capstone chip. A user's public-private encryption keys are stored on the card and escrowed with the user's public-key certificate authority.

Fortress KISS: Keep the Invaders (of Privacy) Socially Sane. This proposed system uses tamper-resistant encryption chips and escrow agent master data recovery keys. See Carmi Gressel, Ran Granot, and Itai Dror, "International Cryptographic Communication Without Key Escrow. KISS: Keep the Invaders (of Privacy) Socially Sane," International Cryptography Institute 1995: Global Challenges.

Kilian and Leighton Failsafe Key Escrow. With this proposal, a user's keys are generated jointly by the user and key escrow agents so the user cannot circumvent key escrow. See Joseph Kilian and Tom Leighton, "Fair Cryptosystems, Revisited," *Proceedings of CRYPTO 95,* pp. 208–221.

Leiberich Time-Bounded Clipper with a Clock. This proposed enhancement to Clipper offers time-bounded data recovery through a clock and date-dependent device unique keys. Otto Leiberich, private communication, June 1994.

Leighton and Micali Key Escrow with Key Agreement. With this proposal, each user has an escrowed private key. Any two users can compute a shared secret key

from their own private key and the identifier of the other. See Tom Leighton and Silvio Micali, "Secret-Key Agreement without Public-Key Cryptography," *Proceedings of Crypto 93,* pp. 208–221.

Lenstra, Winkler, and Yacobi Key Escrow with Warrant Bounds. This proposal allows the escrow agents to release keys that restrict decryption to the communications of a particular user or pair of users during a specific time interval. See Arjen K. Lenstra, Peter Winkler, and Yacov Yacobi, "A Key Escrow System with Warrant Bounds," *Proceedings of Crypto 95,* pp. 197–207.

Lotus Notes International Edition (Differential Workfactor Cryptography). Data are encrypted with 64-bit keys, 24 of which are encrypted under a public key of the government and transmitted with the data. The government can obtain the remaining 40 bits through brute force. See Lotus Backgrounder, "Differential Workfactor Cryptography," Lotus Development Corp., 1996.

Micali and Sidney Resilient Clipper-Like Key Escrow. This proposal allows keys to be split so recovery is possible even if some of the escrow agents compromise or fail to produce their key components. See Silvio Micali and Ray Sidney, "A Simple Method for Generating and Sharing Pseudo-Random Functions, with Applications to Clipper-Like Key Escrow Systems," *Proceedings of Crypto 95,* pp. 185–196.

Micali Fair Public Key Cryptosystems. Verifiable secret sharing techniques are proposed whereby users generate, split, and escrow their private keys with escrow agents of their choice as a prerequisite to putting their public keys in the public key infrastructure. See Silvio Micali, "Fair Cryptosystems," MIT/LCS/TR-579.c, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Mass., Aug. 1994.

Micali Guaranteed Partial Key-Escrow. Under this proposal, the private keys of users are partially escrowed. The escrow agents verify that the bits in their possession are correct and that only a relatively small number of bits are unescrowed. See Silvio Micali, "Guaranteed Partial Key-Escrow," MIT/LCS/TM-537, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Mass., 1995.

National Semiconductor CAKE. This proposal combines a Trusted Information System (TIS) Commercial Key Escrow (CKE) with National's PersonaCard. See W.B. Sweet, "Commercial Automated Key Escrow (CAKE): An Exportable Strong Encryption Proposal," National Semiconductor, iPower Business Unit, June 4, 1995.

Nechvatal Public-Key Based Key Escrow System. This proposal uses Diffie-Hellman public-key techniques for escrowing keys and for data recovery. See James Nechvatal, "A Public-Key Based Key Escrow System," *Journal of Systems and Software,* to appear Oct. 1996.

Nortel Entrust. This commercial product archives user's private encryption keys as part of the certificate authority function and public-key infrastructure support. See Warwick Ford, "Entrust Technical Overview," White Paper, Nortel Secure Networks, Oct. 1994.

PC Security Stoplock KE. This commercial product integrates private key escrow into the key management infrastructure. See *Stoplock Press,* PC Security, Ltd., Marlow, Buckinghamshire, UK, Issue 3, Nov. 1995.

Royal Holloway Trusted Third Party (TTP) Services. This proposed architecture for a public key infrastructure requires that the trusted TTPs associated with pairs of communicating users share parameters and a secret key. See Nigel Jefferies, Chris Mitchell, and Michael Walker, "A Proposed Architecture for Trusted Third Party Services," Royal Holloway, University of London, 1995.

RSA Secure. This file encryption product provides data recovery through an escrowed master public key, which can be split among up to eight trustees using a threshold scheme. See RSA Secure, product literature from RSA Data Security, Inc.

Shamir Partial Key Escrow. This is a proposal to escrow all but 48 bits of a long (256-bit) key. The 48 bits, generated randomly for each session or file, are determined by brute force during data recovery. See Adi Shamir, "Partial Key Escrow: A New Approach to Software Key Escrow," The Weizmann Institute, presentation at NIST Key Escrow Standards meeting, Sept. 15, 1995.

TECSEC VEIL. This commercial product provides file (and object) encryption. Private key escrow is built into the key management infrastructure. See Edward M. Scheidt and Jon L. Roberts, "Private Escrow Key Management," TECSEC Inc., Vienna, Va. See also TECSEC VEIL, product literature.

TESS with Key Escrow. The Exponential Security System supports a general access structure for key escrow. The DRC obtains a particular session key by participating in the key establishment protocol and acquiring the sender's or receiver's private key. See Thomas Beth, Hans-Joachim Knobloch, and Marcus Otten, "Verifiable Secret Sharing for Monotone Access Structures," *Proceedings of the 1st ACM Conf. on Communication and Computer Security,* 1993; Thomas Beth, Hans-Joachim Knobloch, Marcus Otten, Gustavus J. Simmons, and Peer Wichmann, "Towards Acceptable Key Escrow Systems," *Proceedings of the 2nd ACM Conference on Communication and Computer Security,* 1994, pp. 51–58.

Threshold Decryption. With threshold decryption, a secret key can be shared by a group of escrow agents in such a way that through collaboration of the agents, information can be decrypted without the agents releasing their individual key components. See Yvo Desmedt, Yair Frankel, and Moti Yung, "A Scientific Statement on the Clipper Chip Technology and Alternatives," 1993.

TIS Commercial Key Escrow. This is a commercial key escrow system for stored data and file transfers. Data recovery is enabled through master keys held by a Data Recovery Center. See Stephen T. Walker, Stephen B. Lipner, Carl M. Ellison, and David M. Balenson, "Commercial Key Recovery," in this issue of *Communications of the ACM.*

TIS Software Key Escrow Paralleling Clipper. This proposed design is similar to that of Clipper, except that it uses software rather than hardware and public key cryptography for data recovery. See Stephen T. Walker, Stephen B. Lipner, Carl M. Ellison, and David M. Balenson, "Commercial Key Recovery," in this issue of *Communications of the ACM.*

For more detailed descriptions of these systems, see also Dorothy E. Denning's "Descriptions of Key Escrow Systems" at http://www.cosc.georgetown.edu/ndenning/crypto/appendix.html