Equitability in Retroactive Data Confiscation versus Proactive Key Escrow

Yvo Desmedt*[1,2], Mike Burmester*[1,2], and Jennifer Seberry[3]

[1] Department of Computer Science, Florida State University, 206 Love Building, Tallahassee, FL 32306-4530, USA
desmedt@cs.fsu.edu, mikeb@dcs.rhbnc.ac.uk
[2] Information Security Group, Royal Holloway, University of London, Egham Surrey TW20 OEX, UK
[3] Centre for Computer Security Research, TITR, University of Wollongong Australia
Jennifer_seberry@uow.edu.au

Abstract. The British Regulations of Investigatory Powers (RIP) Act 2000 is one of the first modern bills for mandatory disclosure of protected data in a democratic country. In this paper we compare this bill from a technical point of view with the US key escrow proposal (EES) and its variants and then, more generally we compare the merits of data confiscation vs key escrow.

A major problem with key escrow is that once a private key is recovered it can be used to decipher ciphertexts which were sent well before a warrant was issued (or after its expiration). Several alternative key escrow systems have been proposed in the literature to address this issue. These are equitable, in the sense that the control of society over the individual and the control of the individual over society are fairly shared. We show that equitability is much easier to achieve with data confiscation than with key escrow. Consequently, although the RIP act was heavily criticized in the press and on the internet, it inherently maintains a better level of privacy than key escrow.

Finally we present some practical deniable decryption variants of popular public key systems.

Key words: RIP, key escrow, data confiscation.

1 Introduction

Key escrow was proposed as a mechanism to protect society from criminals who use encryption to block access to evidence of crime [5, 25, 11] (for a taxonomy of key escrow systems see [13]). While the US key escrow proposal (EES) [11] was never mandatory, the British Regulations of Investigatory Powers (RIP) Act 2000 [29] has been enacted and is now law. The RIP act has been heavily criticized in the press and over the internet (see e.g. [31]). Indeed some internet societies have considered boycotting Britain [23]. Moreover, the RIP act is being

* Research undertaken while visiting the University of Wollongong.

used as a test case by several other countries who are considering similar acts [27], so it could be used as an excuse by less democratic countries to weaken the privacy that encryption provides. A critical analysis is therefore crucial.

The RIP act differs in several respects from the EES. With RIP, it is the person to whom protected (encrypted) material is addressed who should disclose the material, if a warrant for this purpose has been issued [29]. Compliance may be achieved by "simply making a disclosure of the relevant information in an intelligent form" [29, section 50(1,a)], but "shall require the disclosure of the (decryption) key, if the disclosure can only be complied with the disclosure of the key itself" [29, section 51(1)].

Most of the efficient encryption systems currently used (e.g. the SSL [34]) employ a public key cryptosystem to distributed the session keys for symmetric encryption (such as triple DES). For these systems it would not be possible to comply with the disclosure notice unless the (symmetric) session keys are revealed. Therefore it is most likely that with such schemes, key disclosure will be required. However this is not the case with public key encryption schemes. For example, with the RSA encryption scheme [30] compliance can be achieved by simply disclosing the message(s) (the ciphertext must be the encryption of the message). For the ElGamal encryption scheme [16] disclosing the message is not sufficient: in this case the receiver must also, either reveal the key or prove that the ciphertext is the encryption of the message with the public key of the receiver (see Section 3.1).

It is important to note that with the RIP act, disclosure of the decryption key(s) (or the decrypted ciphertext(s)) may only be required *after* an investigation has started. This is in contrast to key escrow, for which shares of the decryption keys must be given to the escrow agencies *before* any investigation.

The RIP act clearly has some controversial aspects. For example, the penalty for *knowingly* failing to complywith a disclosure notice is "imprisonment for a term not exceeding two years ..." [29, section 53(5,a)], whereas tipping off (e.g. an employee who tips off his security manager) can lead "to imprisonment for a term not exceeding five years ..." [29, section 54(4,a)].

Although the RIP act may be somewhat controversial, the idea of confiscating data is worth comparing with the concept of key escrow. There are several good reasons for this. The most important one being that numerous papers have already been published on key escrow (see e.g. [20, 22, 2, 19, 7, 1]), and most of its problems have been addressed. Furthermore, "key recovery" has often been used as a synonym for key escrow and there are also several papers published on this topic (e.g. in [9]). As far as we know there are no scientific papers on data confiscation.

We do not claim that our study of data confiscation covers all aspects. Indeed, the idea of key escrow is by now (at least) 10 years old [5] and it has taken many years to reach the present state of knowledge. Obviously current research on key escrow will facilitate research on data confiscation, but the fact that these notions are quite different may imply new problems, still to be discovered.

We focus on a particular problem of key escrow that has received some attention. This problem has to do with the fact that, once the key has been recovered, it can be used to decipher ciphertexts sent well before the warrant was issued and/or well after the warrant has expired. Several researchers have pointed out this problem. Lenstra-Winkler-Yacobi [22] state that:

> the key is supposed to be "returned" (!) at the expiration of the warrant, but non-compliance with this or other Dept. of Justice procedures explicitly "shall not provide the basis for any motion to suppress or other objection to the introduction of electronic surveillance evidence lawfully acquired" [14].

It is no surprise therefore that alternative key escrow systems have been proposed to address, to a certain extent, this issue (see e.g. [25, 22, 19, 7, 1]).

The strongest model proposed so far guarantees time-limited decryption even when the escrow agencies are taken over unlawfully. For example, encrypted data (ciphertexts) of law abiding citizens in a democratic society is protected even if at a later date a dictatorship takes full control of the escrow agencies. Burmester-Desmedt-Seberry have proposed a scheme with this property [7] and give credit to Gus Simmons [33] for having first observed the problem. The scheme in [7] however needs some interaction: with each public key updating, the receiver must send this key to the sender (the escrow agencies do not need to do this).

In this paper we examine how to achieve such a time limitation when working with data confiscation. We focus on proven security. This means that we shall not consider the following popular scheme in which a public key encryption scheme is used as a transport mechanism to distribute session keys of a symmetric encryption scheme (such as triple DES or IDEA) [24]. For this scheme, compliance with a disclosure notice is achieved by simply disclosing the session key, without having to reveal anything else. The problem with this scheme is that this is not proven secure. For example, given a ciphertext, it may be possible for the parties involved to disclose a spurious key and a spurious message which produce the same ciphertext. The one-time pad allows such deniable encryption [8]. Other deniable cryptosystems have been studied in [8]. The question whether this is possible for DES (or triple DES) is not known.

Our paper is organized as follows. In Section 2 we discuss attacks by different parties in the context of data confiscation. Since deniable encryption undermines data confiscation, we analyze it in Section 2.1. In Section 3 we discuss the issues to guarantee equitable data confiscation. Solutions are discussed in Section 4. Before we end, in Section 5 we describe new methods to obtain practical deniable encryption.

2 Attacks on data confiscation

In this section we discuss some possible types of attack by different parties, and in particular those attacks which to a large extent affect the privacy of the citizens.

2.1 Attacks by citizens

There are several ways to bypass data confiscation. As in the case of key escrow, data confiscation can be bypassed by using, for example, information hiding techniques (see e.g., [28]). However this approach does not not scale well and indeed most private information is sent by parties who may not (or cannot) use information hiding technologies. It is therefore important to analyze data confiscation for the case when such technologies are not used.

Deniable encryption was proposed by Canetti-Dwork-Naor-Ostrovsky [8], and makes it possible to open a ciphertext in different ways. This kind of encryption can therefore be used to undermine data confiscation. The simplest example of a deniable encryption scheme is the one-time pad. For this scheme, intercepted encrypted data can be opened to produce any cleartext (for an appropriate key).

It is clear that data confiscation is only effective when the encryption is *not* deniable, in other words when the encryption scheme corresponds to a commitment scheme.

2.2 Active attacks by law enforcement agencies

Active attacks by law enforcement agencies may seem unrealistic. However, the RIP act clearly stipulates the involvement of the Secret Intelligence Service, the GCHQ, and the Defence Intelligence (see e.g. [29, section 6]). Their mission may not force them to limit themselves to passive attacks. Protection against active attacks is therefore important.

To attack time-limited data confiscation, law enforcement agencies can use malleable attacks [15]. Let us consider such a type of attack in more detail. Suppose that an agency wants to get hold of the plaintext M_1 of a ciphertext C_1, sent before the time-limited warrant was granted, and suppose that a party (e.g. an insider) is willing to help the law enforcement agency. For this purpose the party can send a ciphertext C_2 whose plaintext M_2 leaks some information about M_1. A particular case of this attack is a replay attack, for which $C_2 = C_1$.

3 Requirements

Although it is often common in modern cryptography to give formal models, we will see that many of the models we need already exists. We therefore discuss the requirements in a more informal way.

3.1 An introduction

It is obvious that for a public key encryption scheme which is also a commitment scheme, the sender can comply with disclosure by simply revealing the message and the randomness used.

If no randomness is used, as for example with RSA, then disclosure of the message it is sufficient. If randomness is used, as for example with ElGamal, then the receiver does not know the randomness (if the discrete logarithm is hard). However in this case it is still possible for the receiver to comply with the disclosure notice without revealing any private keys. For this purpose the receiver must use an interactive zero-knowledge proof [18] to prove that the ciphertext is the encryption of the plaintext with the public key of the receiver (a proof of knowledge of the discrete logarithm [10] can be used for ElGamal).

Note that the RIP act states that [29, Section 50(5)]:

> It shall not be necessary, for the purpose of complying with the requirement, for the person given notice to make a disclosure of any keys in addition to those the disclosure of which is, alone, sufficient to enable the person to whom they are disclosed to obtain access to the information and to put it into an intelligible form.

So with public key encryption schemes the receiver can comply with the RIP act without having revealing the decryption keys.

If public keys are used to distribute the session keys of a symmetric encryption scheme (as in SSL), then disclosure is complied by simply revealing the session keys.

3.2 Undeniable data confiscation

Revealing the randomness used, or proving that the message and ciphertext are properly linked with the public key of the receiver, is only an effective means of disclosure from the point of view of enforcement agency if the encryption scheme is not deniable. We call such schemes, *undeniable*. As mentioned earlier, undeniable encryption schemes are also commitment schemes. For the sake of completeness and to avoid any ambiguity (due to the definition of blobs in [6]) we give the definition below.

Definition 1. Let C be the ciphertext and E be the encryption function with k the receiver's public key. The encryption system E is *undeniable* if, for all ciphertexts C, for all plaintexts m, m', and for all random choices r, r',

$$C = E_k(m, r) = E_k(m', r') \text{ implies } m = m'.$$

If no randomness is used, as in the case of RSA, we take the randomness to be the empty string. So the definition is sufficiently general. As we know RSA does not offer the security of schemes that use randomness [17].

3.3 Sender and receiver coercibility

Canetti-Dwork-Naor-Ostrovsky[8] observed that both the sender and the receiver can be coerced into revealing the message. Since this is the goal of data confiscation, we define the following.

Definition 2. An undeniable encryption scheme is *receiver coercible* if, given a ciphertext $C = E_k(m, r)$, with k the public key of the receiver, the receiver can produce the randomness r and the plaintext m. An undeniable encryption scheme which does not have this property is only *sender coercible*.

Although receiver coercibility is an important property for fair data confiscation, a weaker form allows the receiver to *prove* that the revealed message is the correct one.

Definition 3. An undeniable encryption scheme is *receiver coercible with proof* if, given $C = E_k(m, r)$ the receiver can prove in zero-knowledge that the revealed plaintext m is correct.

Note that the ElGamal scheme is an example of an undeniable encryption scheme which is receiver coercible with proof, but it is *not* receiver coercible (if the discrete logarithm is hard).

We are now in a position to discuss equitable data confiscation.

3.4 Equitable data confiscation

Since we do not have escrow agencies with data confiscation, it may seem that one should not be concerned with the possibility of unlawful government action. This would make our analysis much simpler.

However, this impression is wrong. After a coup all senders of messages may be forced to reveal the plaintext-randomness pairs (m, r) of ciphertexts $C = E_k(m, r)$ sent long before the coup. A scheme that protects against such an attack is called *equitable*.

A weaker form of equitability requires that the revealing of randomness does not leak anything additional about an unrevealed plaintext m of a ciphertext $C = E_k(m, r)$. We call this *weak equitability*.

4 Solutions

We now consider two solutions.

Solution 1. *A probabilistic public key encryption scheme which is,*

(i) *secure against chosen-ciphertext attacks, and*
(ii) *a commitment scheme,*

is an undeniable data confiscation scheme which is sender equitable against active attacks by the law enforcment agency. The receiver is coercible with proof.

This follows immediately from the definitions, e.g. of chosen-ciphertext attack. For equitability, we assume that it is legal for all the senders to erase the randomness used after a certain time limit.

Examples of such schemes are the Cramer-Shoup cryptosystem [12] (an adaptation of the ElGamal system [16]) and the Okamoto-Uchiyama cryptosystem modified to get semantic security [26, p. 311]. For another example see [35]. The security of some these schemes is proven only in the Random Oracle model [3].

Solution 2. *The Goldwasser-Micali encryption scheme [17] is an undeniable data confiscation scheme which is sender/receiver coercible, with sender equitability.*

In this case the trapdoor (the factorization of n) can be used to find the randomness the sender used. However, equitability is restricted to the passive case, when the law enforcement agency does not coerce the receiver. Indeed the receiver knows the trapdoor that is needed to compute the randomness. So we only have weak equitability for the receiver.

5 New deniable encryption schemes

Before we conclude we discuss some new ways to achieve deniable encryption.

A deniable encryption scheme [8] allows the sender to open the ciphertext as any message. Canetti-Dwork-Naor-Ostrovsky's work focused on the case when the message space is $\{0, 1\}$. A weaker definition of deniability would allow the sender to be able to deny having sent a specific message by opening a different message, not just any message. If the message space is $\{0, 1\}$, then of course the two definitions are identical. In many practical encryption schemes the cardinality of the message space is larger than two.

5.1 A heuristic scheme

Our scheme is a variant of the RSA encryption scheme. Our technique can be viewed as a subliminal data transmission [32].

Set-up: Alice chooses 4 different large primes p_1, p_2, p_3 and p_4. She then computes $n_1 = p_1 * p_2$, $n_2 = p_3 * p_4$, $n = p_1 * p_2 * p_3 * p_4$, $\lambda(n_1)$ (the Carmichael function[1] of n [21]), $\lambda(n_2)$. Alice then chooses $e_1 \in_R Z_{\lambda(n_1)}$ and $e_2 \in_R Z_{\lambda(n_2)}$. Since $\lambda(n) = \mathrm{lcm}(\lambda(n_1), \lambda(n_2))$ she can compute the unique e modulo $\lambda(n)$ such that:

$$e = e_1 \bmod \lambda(n_1)$$
$$e = e_2 \bmod \lambda(n_2)$$

Alice publishes (e, n) as her public key and gives (e_1, n_1) and (e_2, n_2) to her friends.

Encryption:
For Bob who is not a friend of Alice: Bob uses the normal RSA encryption. The scheme is undeniable for Bob.
For Carol who is a friend of Alice: Carol uses the key (e_1, n_1) instead of (e, n).

[1] $\lambda(n)$ is the least positive integer for which we have $b^{\lambda(n)} = 1 \bmod n$ for all $b \in Z_n^*$. If $n = pq$, the product of two different odd primes, then $\lambda(n) = \mathrm{lcm}(p-1, q-1)$.

Let M_1 be the message that Carol wants to encrypt. Carol first computes $C_1 = M_1^{e_1} \bmod n_1$. Then she chooses a C_2 at random from Z_{n_2} and uses the Chinese Remainder Theorem to combine C_1 and C_2 uniquely into a $C \bmod n$. This is the ciphertext she sends to Alice. If Alice is coerced, she produces the unique message M which is such that $C = M^e \bmod n$, which looks likely as random.

Observe that the RSA cryptosystem is not a proven secure. To get semantic security (in the random Oracle model [3]) we can use the technique in [4]. Also, with this protocol the effective bandwidth for Carol is reduced. Can our techniques be used to make high-bandwidth proven secure deniable encryption?

6 Conclusion

Although data confiscation may not be constitutional in countries that protect citizens against self incrimination, we have shown that it clearly has some privacy advantages over key escrow.

In this paper we focused on the time-limited properties of data confiscation and observed that deniable encryption prevents undeniable data confiscation. This paper also opens several new research problems, in particular:

1. Since the US key escrow proposal is not mandatory while the British RIP act is, it is worth studying the properties of key confiscation in greater details. In particular what other advantages/disadvantages does data confiscation have over key escrow?
2. The question on how to obtain equitability relative to the receiver for a scheme which is also receiver coercible seems hard to address. A trivial, but unacceptable solution would be for the sender to destroy his/her secret key. Another trivial solution would be to update the public key on a regular basis, but such a solution is too impractical.

Disclaimer

The authors have focused on technical aspects of privacy. It is not the goal of this paper to endorse the British Regulations of Investigatory Powers Act 2000.

References

1. M. Abe. A key escrow scheme with time-limited monitoring for one-way communication. In E. Dawson, A. Clark, and C. Boyd, editors, *Information Security and Privacy, 5th Australian Conference, ACISP 2000*, Lecture Notes in Computer Science 1841, Springer 2000, 163–177.
2. M. Bellare and S. Goldwasser. Verifiable partial key escrow. In *Proceedings of the 4th ACM Conference on Computer and Communications Security*, April 1997.
3. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *First ACM Conference on Computer and Communications Security*, 1993, 62–73

4. M. Bellare and P. Rogaway. Optimal Asymmetric Encryption. In A. De Santis, editor, *Advances in Cryptology — Eurorypt '94*, Lecture Notes in Computer Science #950, Springer 1995, 92–111
5. T. Beth. Zur Sicherheit der Informationstechnik. *Informatik-Spektrum*, 13, 1990, 204–215. (In German)
6. G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, 37(2), 1998, 156–189
7. M. Burmester, Y. Desmedt, and J. Seberry. Equitable key escrow with limited time span. In K. Ohta and D. Pei, editors, *Advances in Cryptology — Asiacrypt '98, Proceedings* Lecture Notes in Computer Science #1514, Springer 1998, 380–391
8. R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky. Deniable encryption. In B. S. Kaliski, editor, *Advances in Cryptology — Crypto '97, Proceedings*, Lecture Notes in Computer Science #1294, Springer 1997, 90–104
9. Key recovery alliance (KRA) technology papers. Special Issue of Computer & Security, 2000, **19**(1).
10. D. Chaum and J.-H. Evertse and J. van de Graaf and R. Peralta. In A. Odlyzko, editor, *Advances in Cryptology, Proc. of Crypto '86*, Lecture Notes in Computer Science #263, Springer-Verlag 1987, 200–212
11. A proposed federal information processing standard for an escrowed encryption standard (EES). Federal Register, July 30, 1993.
12. R. Cramer ans V. Shoup. A Practical Public Key Cryptosystem Provably Secure against Adaptive Chosen Ciphertetx Attack. In H. Krawczyk, editor, *Advances in Cryptology — Crypto '98*, Lecture Notes in Computer Science #1462, Springer 1998, 13–25.
13. D. E. Denning and D. K. Branstad. A taxonomy of key escrow encryption systems. *Commun. ACM*, 39(3), 1996, 34–40
14. Department of Justice Briefing Re Escrowed Encryption Standard, Department of Commerce, Washington D.C., February 4, 1994.
15. D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *Proceedings of the Twenty third annual ACM Symp. Theory of Computing, STOC*, 1991, 542–552
16. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inform. Theory*, 1985, 31, 469–472
17. S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2), 1984, 270–299
18. S. Goldwasser and S. Micali and C. Rackoff. The Knowledge Complexity of Interactive Proof Systems. *Siam J. Comput.*, **18**(1), 1989, 186–208
19. J. He and E. Dawson. A new key escrow cryptosystem. In E. Dawson and. J. Golic, editor, *Cryptography Policy and Algorithms, Proceedings*, Lecture Notes in Computer Science #1029, Springer 1996, 105–114
20. J. Kilian and T. Leighton. Failsafe key escrow, revisited. In D. Coppersmith, editor, *Advances in Cryptology — Crypto '95, Proceedings*, Lecture Notes in Computer Science# 963, Springer 1995, 208–221
21. D. E. Knuth. *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms.* Addison-Wesley, Reading, MA, 1981.
22. A. K. Lenstra, P. Winkler, and Y. Yacobi. A key escrow system with warrant bounds. In D. Coppersmith, editor, *Advances in Cryptology — Crypto '95, Proceedings*, Lecture Notes in Computer Science #963, Springer 1995, 197–207
23. C. D. Marsan. Internet organization opposes new u.k. wiretapping law. http://www.cnn.com/2000/TECH/computing/08/04/wiretap.flap.idg/index.html
24. A. Menezes. P.C van Oorscot and S.A. Vanstone. Handbook of applied cryptography. *CRC Press*, 1997

25. S. Micali. Fair public-key cryptosystems. In E. F. Brickell, editor, *Advances in Cryptology — Crypto '92, Proceedings*, Lecture Notes in Computer Science 740, Springer 1993, 113–138
26. T. Okamoto and S. Uchiyama. A new Public-Key Cryptosystem as Secure as Factoring. In K. Nyberg, editor, *Advances in Cryptology — Eurocrypt '98*, Lecture Notes in Computer Science #1403, Springer 1998, 308–318.
27. R. Perera. Dutch secret service accused of e-mail snooping. http://www.cnn.com/2000/TECH/computing/08/02/netherlands.email.idg/index.htm
28. A. Pfitzmann, editor. *Information Hiding, Third International Workshop, Proceedings* Lecture Notes in Computer Science #1768, Springer 1999.
29. Regulation of Investigatory Powers Act 2000. http://www.homeoffice.gov.uk/ripa/ripact.htm.
30. R. L. Rivest and A. Shamir and L. Adleman. A method for obtaining digital signatures and public key cryptosystems. *Commun. ACM*, 1978 **21**, 294–299
31. L. Rohde. U.K. E-mail Snooping Bill passed. http://www.cnn.com/2000/TECH/computing/07/28/uk.surveillance.idg/index.html
32. G. J. Simmons. The prisoners' problem and the subliminal channel. In D. Chaum, editor, *Advances in Cryptology. Proc. of Crypto 83*, pp. 51–67. Plenum Press N.Y., 1984. Santa Barbara, California, August 1983.
33. G. J. Simmons, observation made at the Workshop on Key Escrow, June 22–24, 1994.
34. SSL vs 3.0, http://home.netscape.com/eng/ssl3/
35. Y. Tsiounis and M. Yung. The security of ElGamal based encryption. In H. Imai and Y. Zheng, editors, *Public Key Cryptography, First International Workshop on Practice and Theory in Public Key Cryptography, PKC'98*, Springer 1998, 117–134