The Cost of Reducing Key-Storage Requirements in Secure Networks

Chris Mitchell
Hewlett-Packard Ltd., Filton Road, Stoke Gifford, Bristol BS12 6QZ, U.K.

and

Fred Piper
Royal Holloway and Bedford New College, Univ. of London, Egham Hill, Egham, Surrey TW20 OEX, U.K.

In large secure networks where each node needs to have the capability to communicate securely with every other node, the key storage requirement can become a significant problem. Various authors have suggested methods for using combinatorial and algebraic techniques to ease this storage problem. However, the use of such schemes can result in an unacceptable reduction of security. Therefore in this paper we list some formal requirements for a key distribution scheme and show that one of the proposed schemes fails to satisfy them.

Keywords: Key management, Cryptography, Network security, Key distribution.

Chris Mitchell is Project Manager for Communication and Information Theory at the Bristol Research Centre of Hewlett-Packard Laboratories. He received his B.Sc. (1975) and Ph.D. (1979) degrees in Mathematics from the University of London. Between 1979 and 1985 he worked on the design and implementation of encryption systems at Racal Comsec Ltd., latterly as Chief Mathematician. His research interests are in cryptography, data security, and a number of topics within applied combinatorial mathematics. He is a member of the British Computer Society, the Institution of Electrical Engineers and the London Mathematical Society, and a Fellow of the Institute of Mathematics and its Applications.

1. Introduction

In a large network which requires the capability of secure end to end communication between every pair of users, key management can become a very significant problem. We consider here the situation where conventional ("symmetric" or "private key") cryptographic techniques are to be used. In such a network every pair of users requires a common secret key to secure their communications, and means need to be provided for the generation, distribution and storage of such keys.

More formally, suppose N terminals are connected in a network, and each pair of terminals needs to have the capability for secure communications. Thus each pair of terminals needs to have in common a secret key. Unless a reduction scheme is adopted, this requires each terminal to store a unique key for use with each other terminal; this entails the storage of $(N-1)L$ bits at each terminal, where L is the number of bits in each key. As Jansen has pointed out in a recent paper [3], if $N = 10{,}000$ and $L = 150$, then each terminal requires a storage capacity of 1.5 megabits. Combinatorial and algebraic techniques have been proposed [1–3] for reducing this rather large storage requirement. Their basic idea involves distributing sets of "subkeys" to each user, which can then be combined to provide keys between every pair of users.

In general, the introduction of such a scheme is likely to have security implications, and it is ab-

Fred Piper is currently Professor of Mathematics at Royal Holloway and Bedford New College, University of London, having previously been Professor of Mathematics at Westfield College, University of London since 1975. He received his B.Sc. and Ph.D. (1964) degrees in Mathematics from Imperial College, University of London. His research interests are in geometry, combinatorics and cryptology. He is the co-author of two books on cryptography: "Cipher Systems" (1982) and "Secure Speech Communications" (1986).

solutely crucial that they are fully investigated so that the reduction in key storage does not result in an unacceptable loss of security. In particular, if two or more users pool the subkeys that they have, then it could become possible for them to deduce the keys being used by other pairs of terminals in the network.

In the next section we discuss some general requirements for subkey systems of this type and then examine Jansen's proposed scheme in this light. Our conclusions are that for this scheme the reduction in security level is undesirable. It is important to be aware of the fact that other schemes can be constructed which do not have all these drawbacks. Research is continuing in this important area which will provide a variety of such schemes, the selection of which can be tailored to fit the requirements of individual networks.

2. General Requirements

In order to state the requirements for a subkey system we need to discuss in a little more detail what we mean by such a system. Basically, we assume that each network node is provided with a set of subkeys, together with a list of instructions on how to combine these subkeys in order to obtain the key to be used for securing communications with any other node. For the purposes of this discussion it is reasonable to assume that these lists and the function used to combine the subkeys are publicly known, and it is just the values of the subkeys themselves which are kept secret.

From this it should be clear that we need a method for producing the lists, together with a set of criteria which must be satisfied by the chosen lists. We now consider these selection criteria. Before proceeding note that although this model fits precisely with Jansen's scheme [3], the Blom model [1,2] is a little more complex in that it assumes the existence of an algebraic structure operating on the set of subkeys; we do not consider that type of system further here.

First and foremost, the set of subkeys used to make up the key for securing communications between network nodes A and B should never be equal to the set of subkeys required to construct the key to be used by nodes C and D unless $A = C$ and $B = D$, i.e. every pair of users should have a unique key. This is not a completely precise statement unless we make clear whether or not the key to be used by node A to secure traffic sent to node B is to be distinct from that used by B to secure traffic to A. In our general discussion we only need provide for one key to be used by nodes A and B since, if necessary, this single key could be "split in half" to provide a key for traffic in each direction.

Note that we have not assumed that the set of subkeys used by node A to construct the key for securing traffic to node B is equal to the set of subkeys held in common by A and B. However, it is clearly necessary for the subkeys used to be contained within the common subset.

Secondly, it is normally crucial that the key to be used by A and B for secure communications is not known by any other network node. In a subkey system such as described above, this implies that the complete set of subkeys used by A and B to construct their common key is known to no other node. If we denote the set of subkeys held at A by (A), then a necessary condition for this to be true is that:

$$(A) \cap (B) \subset (C) \text{ iff } A = C \text{ or } B = C.$$

A third potential threat might arise from the fact that two nodes could pool their subkeys in an attempt to deduce some of the other keys being used on the network. Then a necessary condition for their attack to be prevented is that:

$$(A) \cap (B) \subset (C) \cup (D)$$

iff at least one of $A = C$, $A = D$, $B = C$, $B = D$ holds. It is then straightforward to generalise this condition to ensure that the system is resistant to attacks by up to w nodes pooling their subkeys, and this topic will be discussed further in a future paper.

3. The Jansen System

In this system [3], the storage requirement is considerably reduced by dividing the set of all terminals into a number of subsets, then dividing each subset into smaller subsets, and so on, producing an "n-level" scheme, given that the partitioning process taken place $n - 1$ times. The assumption is that all subsets of a set have the same size. To complete the scheme, n sets of subkeys are introduced, one for each level of the scheme. Each user is supplied with a different selection of these subkeys, and the key to be used by a pair of terminals is made up from the composition of subkeys held in common by this pair of terminals.

More formally, having chosen the number of levels for the scheme, the degree of the partitioning at each level needs to be decided; let r_i represent the number of subsets of a set at level i of the partitioning process. Then each terminal is given a label which consists of an n-tuple $(a_1, a_2, \ldots, a_n)$, where $1 \leqslant a_i \leqslant r_i$ $(1 \leqslant i \leqslant n)$, and this implies that N (the total number of terminals) equals the product $r_1 \cdot r_2 \cdots r_n$. For every i $(1 \leqslant i \leqslant n)$ a set of r_i^2 subkeys is defined, $K_i(s, t)$ say $(1 \leqslant s, t \leqslant r_i)$, such that, given $s < t$, $K_i(s, t)$ is distinct from the $K_i(t, s)$. Each terminal $(a_1, a_2, \ldots, a_n)$ is then supplied with keys $K_i(a_i, m)$ and $K_i(m, a_i)$ where $1 \leqslant m \leqslant r_i$ and $1 \leqslant i \leqslant n$.

We now see that each terminal needs to retain $2(r_1 + r_2 + \ldots + r_n) - n$ keys instead of $r_1 \cdot r_2 \cdots r_n - 1$. The key to be used by terminal $A = (a_1, a_2, \ldots, a_n)$ to communicate with terminal $B = (b_1, b_2, \ldots, b_n)$ is then a composition of the subkeys $K_1(a_1, b_1), K_2(a_2, b_2), \ldots, K_n(a_n, b_n)$. It is not hard to see that this provides for a unique key to be used by every pair of users (indeed by every ordered pair of users), thus meeting the first criterion above.

Now consider the key to be used by terminal $A^* = (b_1, a_2, a_3, \ldots, a_n)$ to communicate with terminal $B^* = (a_1, b_2, b_3, \ldots, b_n)$. By our definition above this is then made up from the composition of the subkeys: $K_1(b_1, a_1)$, $K_2(a_2, b_2), \ldots, K_n(a_n, b_n)$, which is certainly distinct from the key used to communicate between A and B since we assumed that $K_i(s, t)$ is always distinct from $K_i(t, s)$. However, it is clear from the definitions above that $K_1(b_1, a_1)$ is known to both terminals A and B, and hence A and B both have sufficient information to deduce the key used by A^* to communicate with B^*. Hence our second requirement that the key used by one pair of terminals is known to no other terminal in the network is not met by the proposed system. This can be seen as the cost of adopting an otherwise attractive scheme for reducing key storage requirements.

To complete this discussion we show how the formal description above relates to the example in Jansen's paper [3]. In this example: $n = 2$, $r_1 = r_2 = 4$ and hence $N = 4 \times 4 = 16$. Using the language of [3] we must write g_{tu} for $K_1(t, u)$, s_{tu} for $K_2(t, u)$, and assign label T_i to terminal (v, w) iff $i = 4(v - 1) + w$. Then the key used by terminal T_1 (i.e. terminal (1, 1)) to communicate with terminal T_{14} (i.e. (4, 2)) is made from subkeys $K_1(1, 4)$ and $K_2(1, 2)$, i.e. g_{14} and s_{12}. Similarly the key used by terminal $T_2 = (1, 2)$ to communicate with $T_{13} = (4, 1)$ is made from the subkeys $K_1(1, 4) = g_{14}$ and $K_2(2, 1) = s_{21}$. Jansen points out that these keys are distinct, but although this is true, as we have already observed, both g_{14} and s_{21} are also known to T_1 and T_{14}.

Conclusions

In this paper we have drawn attention to the fact that, although reductions in key storage requirements are clearly desirable, they should not be undertaken at the expense of security. We have identified criteria at least some of which must be satisfied in order to produce a subkey system of satisfactory security level. Although these are not the only possible criteria, it is clear that some requirements are fundamental. It is certainly true that systems can be constructed which meet a stringent set of criteria and still offer the potential for considerable savings in key storage space. In general, the more stringent the set of criteria met by the system, the less is the potential for saving storage space. In a future paper we will describe a range of possible solutions, giving the system designer a choice of security level with corresponding levels of storage saving.

References

[1] R. Blom: Non-Public Key Distribution. *Advances in Cryptology: Proceedings of Crypto 82*, Plenum Press (New York, 1983) 231–236.

[2] R. Blom: An Optimal Class of Symmetric Key Generation Systems. *Advances in Cryptology: Proceedings of Eurocrypt 84*, Springer-Verlag (Berlin). Lecture Notes in Computer Science 209 (1985) 335–338.

[3] C.J.A. Jansen: On the Key Storage Requirements for Secure Terminals. *Computers and Security* 5 (1986) 145–149.