# The Consequences of Trust in Shared Secret Schemes[1]

Gustavus J. Simmons
P.O. Box 365
Sandia Park, New Mexico 87047

*Abstract.* By accepting a shared secret or shared control scheme specified by an access structure $\Gamma$, an issuing authority also implicitly accepts all of the access structures that can be realized as a result of trust relations that may exist among some of the participants in $\Gamma$. An algorithm is presented here that makes it possible to fully analyze the consequences of trust to such schemes.

## 1 Introduction

At Eurocrypt '90 Ingemarsson and Simmons [1] described a protocol that made it possible for a group of participants to set up a shared secret scheme without the assistance of a trusted issuing authority to generate and issue the shares. To accomplish this end, they first showed how the participants could jointly set up a unanimous consent scheme in which each of them held an equal and essential share. Each participant could then, acting as his own issuing authority, set up a private secondary shared secret scheme to share his personal share in the unanimous consent scheme with the other participants -- according to the trust he had in them to faithfully represent his interests. This two step protocol made it possible for the participants to jointly set up a shared secret scheme which accurately reflected their trust (or lack of trust) in the other participants.

The Ingemarsson-Simmons protocol has an obvious generalization: Given any shared secret or shared control scheme, $\Gamma$, what other schemes can be reached from $\Gamma$ as a consequence of trust relations that may exist among some of the participants in $\Gamma$? The answer to this question is of crucial concern to an issuing authority setting up a shared secret scheme since he has no way of knowing or of controlling the trust relations that may exist among the participants to whom he issues shares. In other words, anyone can share anything they know with anyone they trust. As the present author has shown [2,3] trust defines a partial ordering and a lattice $\mathcal{L}_n$ on the monotone access structures, $\Gamma_i$, for n participants in which $\Gamma_1 > \Gamma_2$ if $\Gamma_2$ can be "reached" from $\Gamma_1$ as a consequence of some set of trust relations between participants in $\Gamma_1$. Monotonicity says that control, as represented by an access structure

[1]Work supported by the ISS '90 Foundation

T. Helleseth (Ed.): Advances in Cryptology - EUROCRYPT '93, LNCS 765, pp. 448-452, 1994.

$\Gamma_i$, can never be strengthened by trust, but it can be weakened. Clear as this notion is, it is unfortunately infeasible to use, even for a modest number of participants. For example, there are 7579 monotone access structures on 5 participants, i.e. 7579 irreducible and distinct ways of entrusting a secret to subsets of 5 participants. Each of these 7579 structures is a possible trust relation by which a participant in a scheme involving 6 participants might conceivably be willing to entrust his share in a shared control scheme to subsets of the other participants. Consequently, given any access structure, $\Gamma$, on 6 participants, there are potentially $(7579)^6 \approx 1.9 \times 10^{23}$ possible sets of trust relations to be considered to determine which other structures could be reached from $\Gamma$. On the other hand there are only on the order of three and a half million access structures in all on 6 participants. As a result, until now there has been no hope of exploring $\mathcal{L}_6$. On the other hand, schemes involving 6 or more participants are very likely to occur in practice.

A system designer wishing to set up a shared control scheme, $\Gamma$, needs to know which access structures are "reachable" from the scheme he has chosen as reflecting the control (and risk -- of unauthorized use) he is willing to accept. If $\Gamma$ is the access structure he has chosen, there is a sublattice (in $\mathcal{L}_n$) of states that can be reached from $\Gamma$. If one or more of the access structures in this sublattice represent unacceptably weak control according to the designer's objectives (as will always be the case since $\Phi$ -- representing no control at all -- is the lowest element in the lattice itself and in all of the sublattices generated by the different access structures), he must examine the family of trust relations that carry the acceptable scheme into an unacceptable one, and decide whether in his judgement this family of trust relations are sufficiently likely to occur to make the risk of the unacceptable scheme occurring be unacceptable to him or not. If not, the original scheme represents an acceptable risk, while if so, another choice must be made for the initial scheme. Given two access structures $\Gamma_1$ and $\Gamma_2$, where $\Gamma_1 > \Gamma_2$, it is a simple write down procedure to write out the full family of trust relations that carry the first scheme into the second. While the number of terms in this family may grow as the product of the number of terms in the two schemes, the formal manipulation to do the write down operation is simple logical set manipulation of symbols. It is this utility for the analysis and design of shared control schemes that makes $\mathcal{L}_n$ so important. Unfortunately, as already been indicated, this has only been an intellectual exercise until now because of the total infeasibilty of actually calculating in whole, or even in any large part, interesting sized $\mathcal{L}_n$; i.e. for any $n > 5$.

## 2 The Algorithm

The object of this note is to describe an elegant, but simple, way to construct both the lattice $\mathcal{L}_n$ and the sublattices of $\mathcal{L}_n$ generated by specific access structures -- up to sizes that probably exceed our ability

to meaningfully specify shared control schemes. Even more important from a design standpoint, given any specified access structure, $\Gamma$, this technique will generate and exhibit only the sublattice reachable from $\Gamma$ in $\mathcal{L}_n$ . This means that a designer will generate and see only those structures that he needs to consider, but he will see all of these.

Given a lattice L on a set S, and a subset $A \subset S$, define the closure of A , denoted by $A^c$, to be the set of all elements in S that dominate at least one element of A in L. We adopt the convention that $x > x$, so that $A \subset A^c$. The complement of $A^c$ in S is denoted by $S \backslash A^c$. Define two sets, $\lfloor A \rfloor$ and $\lceil A \rceil$, which we will call by the names usually given these quantities in numerical analysis; the floor of A and the ceiling of A respectively. As will be apparent in a moment this nomenclature has an intuitive appeal in the setting in which it is used here. $\lfloor A \rfloor$ is the maximal set of independent elements in L of members of $A^c$, i.e. a maximal independent subset of $A^c$ such that every element in $A^c$ dominates at least one element in $\lfloor A \rfloor$. Another way of defining $\lfloor A \rfloor$ is that it is the set consisting of the lowest element in all maximal length chains in the sublattice on $A^c$ in L. Similarly, $\lceil A \rceil$ is the minimal set of independent elements in L on members of $S \backslash A^c$, i.e. the set containing highest element in all maximal length chains in the sublattice on the elements of $S \backslash A^c$ in L. In a very natural sense $\lfloor A \rfloor$ and $\lceil A \rceil$ define the boundaries of a proper partition of L induced by the set A: no element in $\lceil A \rceil$ dominates any element in $\lfloor A \rfloor$ and every element in $\lfloor A \rfloor$ dominates at least one element in $\lceil A \rceil$. You can see now why I said the floor and ceiling nomenclature was so natural. $A^c$ and $S \backslash A^c$ properly partition L as defined above. $\lfloor A \rfloor$ is the floor of $A^c$ in the sense that it is the subset of $A^c$ that lies below everything else in the sublattice on $A^c$. Similarly, $\lceil A \rceil$ is the ceiling of $S \backslash A^c$ in the sense that it is the subset of $S \backslash A^c$ that lies above everything else in the sublattice on $S \backslash A^c$ in L.

If we take $\mathbf{L}_n$ to be the lattice defined on the $2^n$ subsets of n elements partially ordered by the usual ordering of set inclusion, then $\mathbf{L}_n$ is isomorphic to the binary n-dimensional hypercube $\mathbf{H}_n$. The first result is that the family of all possible floor functions in $\mathbf{H}_n$ is precisely the family of all monotone access structures on n participants. The following theorem is the substance of this note.

**Theorem:**

In the lattice, $\mathcal{L}_n$, of monotone access structures on n participants, $\Gamma_1$ directly dominates $\Gamma_2$ if and only if

$$\Gamma_2 = (\Gamma_1 \backslash S_x) \cup x \qquad x \in \lceil \Gamma_1 \rceil \text{ and } S_x = (y \mid y \in \Gamma_1 \,,\, y > x \text{ in } \mathcal{L}_n).$$

$\lceil \Gamma_1 \rceil$ is easy to calculate. In 1991 Jackson, Martin and Simmons [4] proved that every shared secret scheme has a perfect realization by a

geometrical scheme. As an essential step in their proof they defined a quantity $\Gamma^*$ which was computed from $\Gamma$ by interchanging the operations of $\times$ and +. $\lceil\Gamma\rceil$ in $H_n$ is simply the term by term set complement (in S) of $\Gamma^*$.

Example:

$$\Gamma = AB + ACD$$

$$\Gamma^* = A + BC + BD$$

$$\lceil\Gamma\rceil = AC + AD + BCD.$$

Using the theorem, we can now write down with no difficulty the access structures directly dominated by $\Gamma$ in $\mathcal{L}_4$ : AB + AC, AB + AD and AB + ACD + BCD .The reader may ask, Why doesn't AB +CD appear in this list since the other two pairs of elements from the term ACD do occur paired with the term AD in schemes directly dominated by $\Gamma$ (besides the fact that the theorem doesn't generate this structure)? The answer is that the structure AB + ACD + BCD which is directly dominated by $\Gamma$ in turn directly dominates AB + CD as can easily be verified by applying the theorem to AB + ACD + BCD. The beauty of the theorem is that it is constructive -- using the constructive characterization for $\lceil\Gamma\rceil$ given above. Nice as it is to have a practical means of directly constructing $\mathcal{L}_n$ (one merely starts by applying the theorem to the unanimous consent scheme on n participants, so that no recursion on lower order lattices is needed or involved) the practical merit of the result is that we can start with an arbitrary access structure, $\Gamma$, and by applying the theorem iteratively construct precisely the sublattice of $\mathcal{L}_n$ generated by $\Gamma$, and then by using simple write down rules mechanically express all of the families of trust relations that can weaken $\Gamma$ to reach these schemes.

## Conclusion

The algorithm described here makes it easy to directly calculate either the lattice $\mathcal{L}_n$ of monotone access structures on n participants or the sublattice of $\mathcal{L}_n$ generated by a particular access structure, $\Gamma$, for values of n that probably exceed our ability to meaningfully specify shared control scheme in practical situations.

## References

1. I. Ingemarsson, G. J. Simmons: A Protocol to Set Up Shared Secret Schemes Without the Assistance of a Mutually Trusted Party, Lecture Notes in Computer Science 473; Advances in Cryptology: Proceedings of Eurocrypt '90, I. Damgard, Ed. Aarhus, Denmark, May 21-24 ,1990, Springer-Verlag, Berlin, 1991, pp. 266-282

2. G. J. Simmons: Geometric Shared Secret and/or Shared Control Schemes, Lecture Notes in Computer Science 537; Advances in Cryptology: Proceedings of Crypto '90, S. A. Vanstone, Ed. Santa Barbara, CA, August 11-15, 1990, Springer-Verlag, Berlin 1991, pp. 216-241

3. G. J. Simmons: An Introduction to the Mathematics of Trust in Security Protocols, Proceedings of the IEEE Computer Security Foundations Workshop VI, June 15-17 1993, Franconia NH, IEEE Computer Society 1993, pp. 121-127

4 G. J. Simmons, W. A. Jackson and K. Martin: The Geometry of Shared Secret Schemes, Bulletin of the Institute of Combinatorics and its Applications (ICA), Vol. 1, No. 1, 1991 pp. 71-88