Secret Selling of Secrets with Several Buyers

Arto Salomaa and Lila Santean
Mathematics Department
University of Turku
20500 Turku, Finland

June 7, 2010

Abstract

We present a simple **ANDOS** protocol for the case of more than one buyer. The protocol uses bits left invariant when a one-way function is applied to a binary number.

1 Introduction

S is a seller of secrets who has listed a number of questions and offers to sell the answer to any of them at a huge price which we assume to be the same for each of the secrets. The secrets could be of political importance, for instance, concerning the whereabouts of a sought-after terrorist or concerning the contents of a secret pact between two countries. A buyer **B** wants to buy a secret but does not want to disclose which one. For instance, **B** might be an agent of a country. Disclosing the ignorance of the country concerning a specific matter might be delicate or even dangerous and might, in fact, induce a new secret for **S** to sell.

The abbreviation **ANDOS** (**A**ll or **N**othing **D**isclosure of **Secrets**) is used in [1] for protocols dealing with the situation described above. Although this is not important, we may assume that the secrets are factorizations of products of two large primes. Indeed, if the original secrets are encrypted using **RSA** (with a different **RSA** system for each secret), then a particular secret can be read whenever the factorization of the corresponding modulus becomes known.

More background material is contained in [3]. Apart from being of interest on their own right, **ANDOS** protocols can be used as building blocks in more sophisticated protocols. Of special interest among such protocols are the protocols for secret balloting systems when elections are carried over a computer network. Among the conditions to be met, [2], are the secrecy of votes and exclusion of illegal votes, as well as the possibility of each voter to check that his/her (hereinafter her) vote is taken into account and also to cancel her vote.

2 One buyer

Assume that $s_1, \ldots, s_k$ are secrets possesed by **S**, each of them containing n bits. For each s_i, **S** has publicized what the secret is about. **B** has decided to buy the secret s_j. **S** should transfer it to her without learning the index j. The following is an obvious first try for a protocol.

Step 1. **S** tells **B** a one-way function f mapping n-bit numbers into n-bit numbers but keeps the inverse f^{-1} to herself.

Step 2. **B** chooses k random n-bit numbers $x_1, \ldots, x_k$ and tells **S** the k-tuple $(y_1, \ldots, y_k)$, where

$$y_i = \begin{cases} x_i & \text{if } i \neq j, \\ f(x_i) & \text{if } i = j. \end{cases}$$

Step 3. **S** tells **B** the sequence of numbers

$$s_i \oplus f^{-1}(y_i), i = 1, \ldots, k.$$

(Here $\oplus$ denotes bitwise addition, also called **XOR**.)

Step 4. **B** is able to compute s_j since she knows $x_j = f^{-1}(y_j)$.

Clearly, **S** has no way of distinguishing the exceptional value y_j and, hence, does not learn which secret **B** wants. On the other hand, if **B** is an active cheater (that is, deviates from the protocol), she can present several or all of the numbers y_i to **S** in the form $f(x_i)$.

In the next protocol **B** has no way of cheating but if **S** is an active cheater, she can learn which secret **B** wants. Thus, the situation is reverse to that encountered in the previous protocol.

For an injection f mapping n-bit numbers into n-bit numbers and an n-bit number x, we say that an index $i, 1 \leq i \leq n$, is a *fixed bit index* (FBI) with respect to the pair (x, f) if the i'th bit in x equals the i'th bit in $f(x)$. Clearly, i is FBI with respect to (x, f) iff i is FBI with respect to $(f(x), f^{-1})$. If f has a reasonably random behaviour (like the customarily considered encryption functions) then, for a random x, roughly $n/2$ indices are FBI's with respect to (x, f).

We are now ready for the protocol.

Step 1. **S** tells **B** a one-way function f but keeps the inverse f^{-1} to herself. She also tells **B** k random n-bit numbers $x_1, \ldots, x_k$.

Step 2. **B** (who wants to buy s_j) tells **S** all FBI's with respect to (x_j, f).

Step 3. **S** tells **B** the numbers

$$s_i \oplus f^{-1}(y_i), i = 1, \ldots, k,$$

where y_i is obtained from x_i by replacing all bits whose indices are not in the FBI set of **Step 2** with their complements.

Step 4. Since $f^{-1}(y_j) = x_j$, **B** is able to compute s_j.

The buyer **B** cannot cheat to get two secrets since the numbers x_j are chosen by **S**. On the other hand, **S** can find j by computing FBI's with respect to each pair (x_i, f) and comparing them with the set of **Step 2**.

A more sophisticated protocol can be used to prevent both **S** and **B** from cheating. **B** commits herself to a specific action, that is, specifies which secret she wants to buy. The commitment is "locked in a box" using a one-way function, but in the course of the protocol **B** has to convince **S** that she is acting according to the commitment. This should be done without disclosing information about the action itself—a typical case of a minimum disclosure proof. Details of such a protocol are hinted at in [1].

3 Two buyers

The difficulties met in the preceding section can be overcome in a simple way in the case of two buyers **B** and **C** who want to buy secrets s_j and $s_{j'}$, respectively. The idea is that the buyers have individual one-way functions and each of them operates on numbers provided by the other.

Step 1. **S** tells **B** and **C** individually one-way functions f and g but keeps the inverses to herself.

Step 2. **B** tells **C** (respectively **C** tells **B**) k random n-bit numbers $x_1, \ldots, x_k$ (respectively $x'_1, \ldots, x'_k$).

Step 3. **B** tells **C** (respectively **C** tells **B**) the set FBI_B of FBI's with respect to (x'_j, f) (respectively the set FBI_C of FBI's with respect to $(x_{j'}, g)$).

Step 4. **B** (respectively **C**) tells **S** the numbers $y_1, \ldots, y_k$ (respectively $y'_1, \ldots,$ y'_k), where y_i results from x_i by replacing every bit whose index is not in FBI_C with its complement (respectively y'_i results from x'_i by replacing every bit whose index is not in FBI_B with its complement).

Step 5. **S** tells to **B** (respectively **C**) the numbers

$$s_i \oplus f^{-1}(y'_i)(\text{respectively } s_i \oplus g^{-1}(y_i)), i = 1, \ldots, k.$$

Step 6. **B** (respectively **C**) is able to compute s_j (respectively s'_j) since she knows $x'_j = f^{-1}(y'_j)$ (respectively $x_{j'} = g^{-1}(y_{j'})$).

As before, **B** and **C** learn the secret they want. **S** does not learn anything about the choices, and neither do **B** and **C** learn more than one secret or the choice of the other. A coalition between **B** and **C** renders this protocol to the first protocol considered in **Section 2** and, thus, **B** and **C** learn all secrets. A coalition between **S** and one of the buyers reveals which secret the other buyer wants.

Let us consider a simple example. **RSA** is used to construct the one-way functions needed.

Example. Choose $k = 8, n = 12$. Assume that **S** has the following eight 12-bit secrets for sale:

$$\begin{array}{rcrcl} s_1 & = & 1990 & = & 011111000110 \\ s_2 & = & 471 & = & 000111010111 \\ s_3 & = & 3860 & = & 111100010100 \\ s_4 & = & 1487 & = & 010111001111 \\ s_5 & = & 2235 & = & 100010111011 \\ s_6 & = & 3751 & = & 111010100111 \\ s_7 & = & 2546 & = & 100111110010 \\ s_8 & = & 4043 & = & 111111001011. \end{array}$$

Step 1. **S** tells **B** (respectively **C**) the function f (respectively g) based on $n_1 = 7387$ (respectively $n_2 = 2747$) which is the product of the primes $p_1 = 83, q_1 = 89$ (respectively $p_2 = 67, q_2 = 41$). The encryption and decryption moduli are $d_1 = 777, e_1 = 5145$ (respectively $d_2 = 2261, e_2 = 1421$).

Step 2. **B** tells **C** eight 12-bit numbers $x_i, 1 \leq i \leq 8$:

$$\begin{array}{rcrcl} x_1 & = & 743 & = & 001011100111 \\ x_2 & = & 1988 & = & 011111000100 \\ x_3 & = & 4001 & = & 111110100001 \\ x_4 & = & 2942 & = & 101101111110 \\ x_5 & = & 3421 & = & 110101011101 \\ x_6 & = & 2210 & = & 100010100010 \\ x_7 & = & 2306 & = & 100100000010 \\ x_8 & = & 912 & = & 001110010000. \end{array}$$

C tells **B** eight 12-bit numbers $x_i', 1 \leq i \leq 8$:

$$\begin{array}{rcrcl} x_1' & = & 1708 & = & 011010101100 \\ x_2' & = & 711 & = & 001011000111 \\ x_3' & = & 1969 & = & 011110110001 \\ x_4' & = & 3112 & = & 110000101000 \\ x_5' & = & 4014 & = & 111110101110 \\ x_6' & = & 2308 & = & 100100000100 \\ x_7' & = & 2212 & = & 100010100100 \\ x_8' & = & 222 & = & 000011011110. \end{array}$$

Step 3. **B** wants to buy the secret s_7. Therefore she computes

$$f(x_7') = {x_7'}^{e_1} (\text{mod } n_1) = 2212^{5145} (\text{mod } 7387) = 5928.$$

Comparing the binary representations of x'_7 and $f(x'_7)$,

$$\begin{aligned} 2212 &= 0100010100100 \\ 5928 &= 1011100101000 \end{aligned}$$

B tells **C** the set $\mathrm{FBI}_B = \{0, 1, 4, 5, 6\}$ of FBI's with respect to (x'_7, f).

C wants to buy the secret s_2. Therefore she computes

$$g(x_2) = x_2^{e_2} (\bmod\ n_2) = 1988^{1421} (\bmod\ 2747) = 1660.$$

Comparing the binary representations of x_2 and $g(x_2)$,

$$\begin{aligned} 1988 &= 11111000100 \\ 1660 &= 11001111100 \end{aligned}$$

C tells **B** the set $\mathrm{FBI}_C = \{0, 1, 2, 6, 9, 10\}$ of FBI's with respect to (x_2, g).

Step 4. **B** tells **S** the numbers $y_i, 1 \le i \le 8$, where y_i results from x_i by replacing every bit whose index is not in the set $\{0, 1, 2, 6, 9, 10\}$ (that is every bit whose index is in the set $\{3, 4, 5, 7, 8\}$) with its complement:

$$\begin{aligned} y_1 &= 001101011111 &= 863 \\ y_2 &= 011001111100 &= \mathbf{1660} \\ y_3 &= 111000011001 &= 3609 \\ y_4 &= 101011000110 &= 2758 \\ y_5 &= 110011100101 &= 3301 \\ y_6 &= 100100011010 &= 2330 \\ y_7 &= 100010111010 &= 2234 \\ y_8 &= 001000101000 &= 552. \end{aligned}$$

C tells **S** the numbers $y'_i, 1 \le i \le 8$, where y'_i results from x'_i by replacing every bit whose index is not in the set $\{0, 1, 4, 5, 6\}$ (that is every bit whose index is in the set $\{2, 3, 7, 8, 9, 10, 11, 12\}$) with its complement:

$$\begin{aligned} y'_1 &= 1100100100000 &= 6432 \\ y'_2 &= 1110101001011 &= 7499 \\ y'_3 &= 1100000111101 &= 6205 \\ y'_4 &= 1001110100100 &= 5028 \\ y'_5 &= 1000000100010 &= 4130 \\ y'_6 &= 1011010001000 &= 5768 \\ y'_7 &= 1011100101000 &= \mathbf{5928} \\ y'_8 &= 1111101010010 &= 8018. \end{aligned}$$

Step 5. **S** tells **B** the numbers $s_i \oplus f^{-1}(y'_i), 1 \leq i \leq 8$ (recall that $f^{-1}(y') = y'^{d_1} (\text{mod } n_1) = y'^{777} (\text{mod } 7387)$):

$$
\begin{array}{rcccccc}
s_1 & = & 1990 & = & 0011111000110 & & \\
f^{-1}(y'_1) & = & 5897 & = & \underline{1011100001001} & & \\
s_1 \oplus f^{-1}(y'_1) & = & & & 1000011001111 & = & 4303 \\
\\
s_2 & = & 471 & = & 0000111010111 & & \\
f^{-1}(y'_2) & = & 5546 & = & \underline{1010110101010} & & \\
s_2 \oplus f^{-1}(y'_2) & = & & & 1010001111101 & = & 5245 \\
\\
s_3 & = & 3860 & = & 0111100010100 & & \\
f^{-1}(y'_3) & = & 4161 & = & \underline{1000001000001} & & \\
s_3 \oplus f^{-1}(y'_3) & = & & & 1111101010101 & = & 8021 \\
\\
s_4 & = & 1487 & = & 0010111001111 & & \\
f^{-1}(y'_4) & = & 4345 & = & \underline{1000011111001} & & \\
s_4 \oplus f^{-1}(y'_4) & = & & & 1010100110110 & = & 5430 \\
\\
s_5 & = & 2235 & = & 0100010111011 & & \\
f^{-1}(y'_5) & = & 6070 & = & \underline{1011110110110} & & \\
s_5 \oplus f^{-1}(y'_5) & = & & & 1111100001101 & = & 7949 \\
\\
s_6 & = & 3751 & = & 0111010100111 & & \\
f^{-1}(y'_6) & = & 2660 & = & \underline{0101001100100} & & \\
s_6 \oplus f^{-1}(y'_6) & = & & & 0010011000011 & = & 1219 \\
\\
s_7 & = & 2546 & = & 0100111110010 & & \\
f^{-1}(y'_7) & = & 2212 & = & \underline{0100010100100} & & \\
s_7 \oplus f^{-1}(y'_7) & = & & & 0000101010110 & = & \mathbf{342} \\
\\
s_8 & = & 4043 & = & 0111111001011 & & \\
f^{-1}(y'_8) & = & 1469 & = & \underline{0010110111101} & & \\
s_8 \oplus f^{-1}(y'_8) & = & & & 0101001110110 & = & 2678.
\end{array}
$$

S tells **C** the numbers $s_i \oplus g^{-1}(y_i), 1 \leq i \leq 8$ ($g^{-1}(y) = y^{d_2} (\text{mod } n_2) = y^{2261} (\text{mod } 2747)$).

$$
\begin{array}{rclcl}
s_1 & = & 1990 & = & 011111000110 \\
g^{-1}(y_1) & = & 576 & = & \underline{001001000000} \\
s_1 \oplus g^{-1}(y_1) & = & & & 010110000110 \quad = \quad 1414 \\
s_2 & = & 471 & = & 000111010111 \\
g^{-1}(y_2) & = & 1988 & = & \underline{011111000100} \\
s_2 \oplus g^{-1}(y_2) & = & & & 011000010011 \quad = \quad \mathbf{1555} \\
\\
s_3 & = & 3860 & = & 111100010100 \\
g^{-1}(y_3) & = & 1477 & = & \underline{010111000101} \\
s_3 \oplus g^{-1}(y_3) & = & & & 101011010001 \quad = \quad 2769 \\
\\
s_4 & = & 1487 & = & 010111001111 \\
g^{-1}(y_4) & = & 2162 & = & \underline{100001110010} \\
s_4 \oplus g^{-1}(y_4) & = & & & 110110111101 \quad = \quad 3517 \\
\\
s_5 & = & 2235 & = & 100010111011 \\
g^{-1}(y_5) & = & 677 & = & \underline{001010100101} \\
s_5 \oplus g^{-1}(y_5) & = & & & 101000011110 \quad = \quad 2590 \\
\\
s_6 & = & 3751 & = & 111010100111 \\
g^{-1}(y_6) & = & 581 & = & \underline{001001000101} \\
s_6 \oplus g^{-1}(y_6) & = & & & 110011100010 \quad = \quad 3298 \\
\\
s_7 & = & 2546 & = & 100111110010 \\
g^{-1}(y_7) & = & 840 & = & \underline{001101001000} \\
s_7 \oplus g^{-1}(y_7) & = & & & 101010111010 \quad = \quad 2746 \\
\\
s_8 & = & 4043 & = & 111111001011 \\
g^{-1}(y_8) & = & 473 & = & \underline{000111011001} \\
s_8 \oplus g^{-1}(y_8) & = & & & 111000010010 \quad = \quad 3602.
\end{array}
$$

Step 6. **B** learns the secret s_7 by computing the bitwise addition of x'_7 and the 7th number received from **S**, that is:

$$
\begin{array}{rclcl}
x'_7 & = & 2212 & = & 100010100100 \\
 & & 342 & = & \underline{000101010110} \\
 & & & & 100111110010 \quad = \quad \mathbf{2546}.
\end{array}
$$

As **C** wants to buy the secret s_2 she computes the bitwise addition between x_2 and the 2nd number received from **S**, that is:

$$\begin{array}{rcrcl} x_2 & = & 1988 & = & 11111000100 \\ & & 1555 & = & \underline{11000010011} \\ & & & & 00111010111 \; = \; \mathbf{471}. \end{array}$$

4 More than two buyers

We have observed that in case of many buyers the main difficulty is due to coalitions. However, if there are at least three buyers, it seems that one honest buyer is enough to make the cheating of the other buyers impossible. So no honest majority is needed. Let us see how this works.

We assume that there are three buyers **A**, **B**, **C** and describe the protocol from **A**'s point of view. **A** wants the secret s_j.

Step 1. **S** tells **A** two one-way functions f_A^B and f_A^C.

Step 2. **B** (respectively **C**) tells **A** k random n-bit numbers $x_1^{BA}, \ldots, x_k^{BA}$ (respectively $x_1^{CA}, \ldots, x_k^{CA}$).

Step 3. **A** tells **B** (respectively **C**) the set FBI_A^B (respectively FBI_A^C) of FBI's with respect to the pair (x_j^{BA}, f_A^B) (respectively the pair (x_j^{CA}, f_A^C)).

Step 4. **B** (respectively **C**) tells **S** the numbers y_i^{BA} obtained from x_i^{BA} (respectively y_i^{CA} obtained from x_i^{CA}), $i = 1, \ldots, k$, by replacing every bit whose index is not in FBI_A^B (respectively FBI_A^C) with its complement.

Step 5. **S** tells **A** the numbers

$$s_i \oplus (f_A^B)^{-1}(y_i^{BA}) \oplus (f_A^C)^{-1}(y_i^{CA}), i = 1, \ldots, k.$$

Step 6. **A** is able to compute s_j since she knows $x_j^{BA} = (f_A^B)^{-1}(y_i^{BA})$ and $x_j^{CA} = (f_A^C)^{-1}(y_i^{CA})$.

Analogous parts should be stated for **B** and **C** to complete the protocol. Thus, **S** gives both of them two one-way functions, both of them receive numbers from the other two buyers, etc. The protocol works in exactly the same way for $t > 3$ buyers. Each of the buyers gets $t-1$ one-way functions from the seller, as well as sets of numbers from all of their fellow buyers.

It is clear that each of the buyers gets the secret she wants. It is also clear that if all buyers are in coalition, they learn all the secrets. However, no coalition of $t-1$ (or less) dishonest buyers can gain much because every bit in the sequences sent to them by **S** depends on a bit provided by the honest buyer.

5 Conclusion

In case of more than one buyer, complicated arguments working with minimum disclosure proofs can be avoided. It seems that coalitions between buyers do not help if at least one of the buyers is honest.

Similar ideas can be used for other cryptographic protocols as well. We hope to return to this matter in the near future.

References

[1] G.Brassard, C.Crepeau and J.-M Robert. All-or-nothing disclosure of secrets. *Springer Lecture Notes in Computer Science* **263**(1987) 234-38.

[2] H.Nurmi and A.Salomaa. A cryptographic approach to the secret ballot. *Behavioral Science*, to appear.

[3] A.Salomaa. *Public-Key Cryptography. EATCS Monographs in Theoretical Computer Science*, Springer-Verlag, in print.