Implementation of a Key Exchange Protocol Using Real Quadratic Fields

Extended Abstract

Renate Scheidler
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba
Canada R3T 2N2

Johannes A. Buchmann
FB-10 Informatik
Universität des Saarlandes
6600 Saarbrücken
West Germany

Hugh C. Williams
Department of Computer Science
University of Manitoba
Winnipeg, Manitoba
Canada R3T 2N2

I.B. Damgard (Ed.): Advances in Cryptology - EUROCRYPT '90, LNCS 473, pp. 98-109, 1991.

Implementation of a Key Exchange Protocol Using Real Quadratic Fields

Extended Abstract

1. Introduction

In [1] Buchmann and Williams introduced a key exchange protocol which is based on the Diffie-Hellman protocol (see [2]). However, instead of employing arithmetic in the multiplicative group F^* of a finite field F (or any finite Abelian group G), it uses a finite subset of an infinite Abelian group which itself is not a subgroup, namely the set of reduced principal ideals in a real quadratic field. As the authors presented the scheme and its security without analyzing its actual implementation, we will here discuss the algorithms required for implementing the protocol.

Let $D \in \mathbf{Z}_+$ be a squarefree integer, $K = \mathbf{Q} + \mathbf{Q}\sqrt{D}$ the *real quadratic number field* generated by $\sqrt{D}$, and $\mathbf{O} = \mathbf{Z} + \mathbf{Z}\frac{\sigma - 1 + \sqrt{D}}{\sigma}$ the *maximal real quadratic order* in K, where $\sigma = \begin{cases} 1 & \text{if} \quad D \equiv 2, 3 \pmod 4 \\ 2 & \text{if} \quad D \equiv 1 \pmod 4 \end{cases}$.

A subset $\mathbf{a}$ of $\mathbf{O}$ is called an *ideal* in $\mathbf{O}$ if both $\mathbf{a} + \mathbf{a}$ and $\mathbf{O}\cdot\mathbf{a}$ are subsets of $\mathbf{a}$. An ideal is said to be *primitive* if it has no rational prime divisors. Each primitive ideal $\mathbf{a}$ in $\mathbf{O}$ has a representation

$$\mathbf{a} = \left[\frac{Q}{\sigma}, \frac{P + \sqrt{D}}{\sigma} \right] = \mathbf{Z}\frac{Q}{\sigma} + \mathbf{Z}\frac{P + \sqrt{D}}{\sigma},$$

where $P, Q \in \mathbf{Z}$, Q is a divisor of $D - P^2$ (see [5]). Let $\Delta = \frac{4}{\sigma^2} D$ denote the *discriminant* of K, set $d = \lfloor \sqrt{D} \rfloor$.

A *principal ideal* $\mathbf{a}$ of $\mathbf{O}$ is an ideal of the form $\mathbf{a} = \frac{1}{\alpha} \mathbf{O}$, $\alpha \in K - \{0\}$. Denote by $\mathbf{P}$ the set of primitive principal ideals in $\mathbf{O}$. An ideal $\mathbf{a} = \frac{1}{\alpha} \mathbf{O} \in \mathbf{P}$ is *reduced* if and only if α is a *minimum* in $\mathbf{O}$, i.e. if $\alpha > 0$ and there exists no $\beta \in \mathbf{O} - \{0\}$ such that $|\beta| < \alpha$ and $|\beta'| < \alpha$. Since the set $\{\log \alpha \mid \alpha \text{ is a minimum in } \mathbf{O}\}$ is discrete in the real numbers $\mathbf{R}$, the minima in $\mathbf{O}$ can be arranged in a sequence $(\alpha_j)_{j \in \mathbf{Z}}$ such that $\alpha_j < \alpha_{j+1}$ for all $j \in \mathbf{Z}$. If we define $\mathbf{a}_j = \frac{1}{\alpha_j} \mathbf{O}$ for all $j \in \mathbf{Z}$, then the set $\mathfrak{R}$ consisting of all reduced ideals in $\mathbf{P}$ is finite and can be written as $\mathfrak{R} = \{\mathbf{a}_1, \ldots, \mathbf{a}_l\}$ where $l \in \mathbf{Z}_+$.

Define an (exponential) *distance* between two ideals $\mathbf{a}, \mathbf{b} \in \mathfrak{R}$ as follows:

$\lambda(\mathbf{a}, \mathbf{b}) = \alpha$ where $\alpha \in K^{>0}$ is such that $\mathbf{b} = \frac{1}{\alpha} \mathbf{a}$ and $|\log \alpha|$ is minimal.

(The logarithm of this distance function is exactly the distance as defined in [1] and [4].) Similarly, let the distance between an ideal $\mathbf{a} \in \mathfrak{R}$ and a positive real number x be

$\lambda(\mathbf{a}, x) = \frac{e^x}{\alpha}$ where $\alpha \in K^{>0}$ is such that $\mathbf{a} = \frac{1}{\alpha} \mathbf{O}$ and $|x - \log \alpha|$ is minimal.

Throughout our protocol the inequalities $\eta^{-\frac{1}{4}} < \lambda(\mathbf{a}, \mathbf{b}), \lambda(\mathbf{a}, x) < \eta^{\frac{1}{4}}$ will be satisfied for all $\mathbf{a}, \mathbf{b} \in \mathfrak{R}$, $x \in \mathbf{R}_+$, where η is the *fundamental unit* of K.

<u>Lemma 1</u>: Let $\mathbf{b} \in \mathfrak{R}$ and write $\mathbf{b} = \mathbf{b}_j$, $\mathbf{b}_k = \left[\frac{Q_{k-1}}{\sigma}, \frac{P_{k-1} + \sqrt{D}}{\sigma} \right]$ for $k \geq j$. Then the following is true:

a) $\mathbf{b}_k \in \mathfrak{R}$ and $0 < P_k \leq d$, $0 < Q_k \leq 2d$ for $k \geq j$,

b) $1 + \frac{1}{\sqrt{\Delta}} < \lambda(\mathbf{b}_{j+1}, \mathbf{b}_j) < \sqrt{\Delta}$,

c) $\lambda(\mathbf{b}_{j+2}, \mathbf{b}_j) > 2$,

d) If $\mathbf{b} = \frac{1}{\beta}\mathcal{O}$, $\beta \in K_{>0}$, then $\lambda(\mathbf{b}, x) = \frac{e^x}{\beta}$,

e) $\lambda(\mathbf{b}_k, \mathbf{b}_j) = \frac{\lambda(\mathbf{b}_k, x)}{\lambda(\mathbf{b}_j, x)}$ for any $x \in \mathbf{R}_+$, $k \geq j$.

Since principal ideal generators and distances are generally irrational numbers, we need to use approximations in our protocol. Denote by $\mathbf{a}(x)$ the reduced ideal *closest* to $x \in \mathbf{R}_+$, i.e. $|\log \lambda(\mathbf{a}(x), x)| < |\log \lambda(\mathbf{b}, x)|$ for any $\mathbf{b} \in \Re$, $\mathbf{b} \neq \mathbf{a}$, and by $\hat{\mathbf{a}}(x)$ the ideal actually computed by our algorithm. Define $\mathbf{a}_+(x)$ to be the reduced ideal such that its distance to x is maximal and < 1. Similarly, $\lambda(\mathbf{a}_-(x), x) > 1$ and minimal. Let $\lambda_1(x) = \lambda(\mathbf{a}(x), x)$, $\lambda_2(x) = \lambda(\hat{\mathbf{a}}(x), x)$. Denote by $\hat{\lambda}(\mathbf{a}, x)$ the approximation of $\lambda(\mathbf{a}, x)$ computed by our algorithm; write $\hat{\lambda}(\mathbf{a}, x) = \frac{M(\mathbf{a}, x)}{2^p}$ where $M(\mathbf{a}, x) \in \mathbf{Z}_+$ and $p \in \mathbf{Z}_+$ is a *precision constant* to be determined later. $\hat{\lambda}_1(x)$, $M_1(x)$, $\hat{\lambda}_2(x)$, $M_2(x)$ are defined analogously to $\hat{\lambda}(x)$ and $M(x)$ with respect to $\lambda_1(x)$ and $\lambda_2(x)$. Set

$$G = 1 + \frac{1}{15(d+1)}, \qquad \gamma = \lceil G^{-1} 2^p \rceil, \qquad \chi = 1 + \frac{1}{2^{p-1}}.$$

The protocol can be outlined as follows: Two communication partners A and B agree publicly on a small number $c \in \mathbf{R}_+$ and an initial ideal $\hat{\mathbf{a}}(c)$ with approximate distance $M_2(c)$ from c. A secretly chooses $a \in \{1, \dots, d\}$, computes $\hat{\mathbf{a}}(ac)$ and $M_2(ac)$ from $\hat{\mathbf{a}}(c)$ and $M_2(c)$, and sends both to B. Similarly, B secretly chooses $b \in \{1, \dots, d\}$, calculates $\hat{\mathbf{a}}(bc)$ and $M_2(bc)$, and transmits both to A. Now both communication partners are able to determine an ideal $\hat{\mathbf{a}}(abc)$. Although this ideal need not be the same for A and B (due to their different approximation errors in the computation), a little additional work will enable them to agree on a common ideal which is the secret key.

As pointed out in [1], we expect $l = |\mathfrak{R}| >> D^{\frac{1}{2} - \varepsilon}$ for arbitrary ε if D is chosen correctly and sufficiently large. This shows that an exhaustive search attack is infeasible. The authors conjecture that breaking the protocol enables one to factor. In [1] it is proved that solving the *discrete logarithm problem* for reduced principal ideals in real quadratic orders - given $\mathfrak{a} \in \mathfrak{R}$ find $\lambda(\mathfrak{a}, x)$ - in polynomial time implies being able to both break the scheme and factor D in polynomial time.

Throughout the protocol we will assume $M(\mathfrak{a}, x) \geq \gamma$ for all $\mathfrak{a} \in \mathfrak{R}$ and $x \in \mathbf{R}_+$. Any number $\theta \in K$ is approximated by $\hat{\theta} \in \mathbf{Q}$ such that $\chi^{-1}\theta \leq \hat{\theta} \leq \chi\theta$.

2. The Algorithms

For our protocol we need to perform arithmetic in both $\mathbf{P}$ and $\mathfrak{R}$. Our first algorithm enables us to compute any reduced ideal $\mathfrak{a}_k$ from a given reduced ideal $\mathfrak{a}_j$ by simply going through $\mathfrak{R}$ "step by step".

Algorithm 1 (*Neighbouring in* $\mathfrak{R}$): *Input:* $\mathfrak{a}_j \in \mathfrak{R}$.

Output: The neighbours $\mathfrak{a}_{j+1}, \mathfrak{a}_{j-1} \in \mathfrak{R}$ and ψ_+, ψ_- such that $\mathfrak{a}_{j\pm 1} = \psi_\pm \mathfrak{a}_j$.

Algorithm: $\mathfrak{a}_{j+1}$ is obtained by computing one iteration in the continued fraction expansion of the irrational number $\dfrac{P_{j-1} + \sqrt{D}}{Q_{j-1}}$. The algorithm for $\mathfrak{a}_{j-1}$ is the inverse of the algorithm for $\mathfrak{a}_{j+1}$. In particular:

$$q_{j-1}=\left\lfloor\frac{P_{j-1}+d}{Q_{j-1}}\right\rfloor,\quad P_j=q_{j-1}Q_{j-1}-P_{j-1},\quad Q_j=\frac{D-P_j^2}{Q_{j-1}},\qquad \psi_+=\frac{\sqrt{D}-P_j}{Q_j},$$

$$Q_{j-2}=\frac{D-P_{j-1}^2}{Q_{j-1}},\quad q_{j-2}=\left\lfloor\frac{P_{j-1}+d}{Q_{j-2}}\right\rfloor,\ P_{j-2}=q_{j-2}\,Q_{j-2}-P_{j-1},\quad \psi_-=\frac{\sqrt{D}+P_{j-1}}{Q_{j-2}}.$$

<u>Algorithm 2</u> (*Multiplication in* **P**): *Input:* **a**, **a**' $\in$ **P**.

Output: $U \in \mathbf{Z}_{\geq 0}$, **c** $\in$ **P** such that **aa**' = U**c**.

Algorithm: See [3], [4].

<u>**Lemma 2**</u>: If **a** = $\mathbf{a}_s$, **a** = $\mathbf{a}_t$ such that $\mathbf{a}_{s-1}, \mathbf{a}_{t-1} \in \mathfrak{R}$, then Algorithm 2 performs O(log D) arithmetic operations on numbers of input size O(log D).

Proof: By Lemma 1 all input numbers are polynomially bounded in D. The algorithm performs a fixed number of arithmetic operations plus two applications of the Extended Euclidean Algorithm which has complexity O(log D). ♦

<u>Algorithm 3</u> (*Reduction in* **P**): *Input:* $\mathbf{c} = \left[\frac{Q}{\sigma}, \frac{P+\sqrt{D}}{\sigma}\right] \in$ **P**.

Output: $\mathbf{b} \in \mathfrak{R}$, $G, B \in \mathbf{Z}_{\geq 0}$ such that $\theta = \frac{G+B\sqrt{D}}{Q}$ and $\mathbf{b} = \theta\mathbf{c}$.

Algorithm: The algorithm is very similar to Algorithm 1 and uses again the continued fraction expansion of $\frac{P+\sqrt{D}}{Q}$ (see [3]).

<u>Lemma 3</u>: If $\mathbf{c} = \frac{1}{U}\mathbf{a}_s\mathbf{a}_t$ where $\mathbf{a}_s$, $\mathbf{a}_t$ are as in Lemma 2, then Algorithm 3 performs O(log D) arithmetic operations on numbers of input size O(log D).

Proof: By [5], Algorithm 2, and Lemma 1, the maximun number of iterations is O(log D). The bound on the input size follows from Lemma 1 and results in [4]. ♦

<u>Algorithm 4</u>: *Input:* $\hat{\mathbf{a}}(x), \hat{\mathbf{a}}(y) \in \mathfrak{R}$, $M_2(x)$, $M_2(y)$ for $x, y \in \mathbf{R}_+$.

Output: $\hat{\mathbf{a}}(x+y) \in \mathfrak{R}$, $M_2(x+y)$.

Algorithm: First use Algorithm 2 to compute $U \in \mathbf{Z}$, $\mathbf{c} = \left[\frac{Q}{\sigma}, \frac{P+\sqrt{D}}{\sigma}\right] \in \mathbf{P}$ such that $(U)\mathbf{c} = \hat{\mathbf{a}}(x)\hat{\mathbf{a}}(y)$. Then compute $\mathbf{b} = \left[\frac{Q'}{\sigma}, \frac{P'+\sqrt{D}}{\sigma}\right] \in \mathfrak{R}$ and $G, B \in \mathbf{Z}_{\geq 0}$ such that $\mathbf{b} = \theta\mathbf{c}$, $\theta = \frac{G+B\sqrt{D}}{Q}$ using Algorithm 3. Finally apply Algorithm 1 to $\mathbf{b}$ a certain number of times to obtain $\hat{\mathbf{a}}(x+y) = \zeta\mathbf{b} = \frac{\zeta\theta}{U}\hat{\mathbf{a}}(x)\hat{\mathbf{a}}(y)$. Set

$$M_2(x+y) = \left\lceil \frac{\hat{\zeta}\hat{\theta} M_2(x) M_2(y)}{2PU} \right\rceil,$$

where $\hat{\zeta}$, $\hat{\theta}$ are rational approximations to ζ, θ, respectively.

<u>Lemma 4</u>: If $\hat{\mathbf{a}}(x) = \mathbf{a}_s$, $\hat{\mathbf{a}}(y) = \mathbf{a}_t$ such that $\mathbf{a}_{s-1}, \mathbf{a}_{t-1} \in \mathfrak{R}$, then Algorithm 4 performs O(log D) arithmetic operations on inputs of size O(log D).

Proof:By Lemma 2, computing $\mathbf{c}$ takes O(log D) arithmetic operations on inputs of size O(log D). By Lemma 3, the same is true for the computation of $\mathbf{b}$. From Lemma 1 it can be proved that, in obtaining $\hat{\mathbf{a}}(x+y)$ from $\mathbf{b}$, all numbers involved are polynomially bounded in D and $\hat{\mathbf{a}}(x+y)$ can be obtained from $\mathbf{b}$ in O(log D) iterations. ♦

Both communication partners can determine the key by using the following algorithm which is based on the idea of a standard exponentiation method:

Algorithm 5: *Input:* $\hat{\mathfrak{a}}(x) \in \Re$ for $x \in \mathbf{R}_+$, $M_2(x)$, $y \in \mathbf{Z}_+$.

Output: $\hat{\mathfrak{a}}(xy)$, $M_2(xy)$.

Algorithm: 1) Determine the binary decomposition $y = \sum_{i=0}^{l} b_i\, 2^{l-i}$ of y, $b_i \in \{0,1\}$, $b_0 = 1$.

2) Set $\hat{\mathfrak{a}}(z_0) = \hat{\mathfrak{a}}(x)$.

3) for $i = 1$ to l do

a) Compute $\hat{\mathfrak{a}}(2z_{i-1})$, $M_2(2z_{i-1})$ using Algorithm 4.

Set $\hat{\mathfrak{a}}(z_i) := \hat{\mathfrak{a}}(2z_{i-1})$, $M_2(z_i) := M_2(2z_{i-1})$.

b) if $b_i = 1$ then compute $\hat{\mathfrak{a}}(z_i+x)$, $M_2(z_i+x)$ using Algorithm 4.

Set $\hat{\mathfrak{a}}(z_i) := \hat{\mathfrak{a}}(z_i+x)$, $M_2(z_i) := M_2(z_i+x)$.

4) Set $\hat{\mathfrak{a}}(xy) := \hat{\mathfrak{a}}(z_l)$, $M_2(xy) = M_2(z_l)$.

Lemma 5: If $\hat{\mathfrak{a}}(x) = \mathbf{a}_s$ such that $\mathbf{a}_{s-1} \in \Re$ and y is polynomially bounded in D, then Algorithm 5 performs $O((\log D)^2)$ arithmetic operations on inputs of size $O(\log D)$.

Proof: For each iteration, steps 3a and 3b each perform $O(\log D)$ operations on numbers of input size $O(\log D)$ by Lemma 4. So the number of operations needed for step 3 is $O(l \log D) = O((\log D)^2)$. ♦

3. The Protocol

Algorithm 6 (*Initial values*): *Input*: $r \in \{2,\ldots, d\}$.

Output: $\mathfrak{a} \in \mathfrak{R}$, $M \in \mathbf{Z}_+$, such that the ideal $\mathfrak{a}$ and its distance M can be used as initial values for the protocol.

Algorithm: Set $\mathfrak{a} = \hat{\mathfrak{a}}(c) = \mathcal{O}$, $M = M_2(c) = \lceil 2^p r \rceil$, where $c = \log r$. Then $M \geq 2^{p+1} > \gamma$. Since $1 + \frac{1}{\sqrt{\Delta}} < r = \lambda_2(c) < \sqrt{\Delta}$, we have $\mathfrak{a} = \mathfrak{a}_-(c)$.

In order to find a unique key ideal, all approximation errors $\rho_2(x) = \dfrac{\hat{\lambda}_2(x)}{\lambda_2(x)}$ $(x \in \mathbf{R}_+)$ in Algorithms 4, 5, and 6 must be close to 1, i. e. p must be sufficiently large.

Theorem 1: Let $a, b \in \{1,\ldots,d\}$, $\hat{\mathfrak{a}}(c)$, $M_2(c)$ as in Algorithm 6. Let $\hat{\mathfrak{a}}(abc)$ be computed by applying Algorithm 5 first to $\hat{\mathfrak{a}}(c)$, $M_2(c)$, and b to obtain $\hat{\mathfrak{a}}(bc)$ and $M_2(bc)$, then to $\hat{\mathfrak{a}}(bc)$, $M_2(bc)$, and a to obtain $\hat{\mathfrak{a}}(abc)$ and $M_2(abc)$. If $2^p \geq 1280d(d^2-1)$, then $\hat{\mathfrak{a}}(abc) \in \{\mathfrak{a}_-(abc), \mathfrak{a}_+(abc)\}$ and $M_2(abc) \geq \gamma$.

The uniqueness of the key ideal is guaranteed by the following Lemma:

Lemma 6: Let $p, a, b, c, \hat{\mathfrak{a}}(c), M_2(c)$ be as in Theorem 1. Set $x = abc$.

If $\lambda_1(x) > G^2$ or $\lambda_1(x) < G^{-2}$ then $\hat{\mathfrak{a}}(x) = \mathfrak{a}_-(x)$.

If $G^{-2} \leq \lambda_1(x) \leq G^2$ then $\mathfrak{a}(x)$ can be determined from $\hat{\mathfrak{a}}(x)$.

Proof: Omit the argument x for brevity. If $\lambda_1 > G^2$ or $\lambda_1 < G^{-2}$ then $\hat{\lambda}_2 > G$ and hence $\lambda_2 = \dfrac{\hat{\lambda}_2}{\rho_2} > 1$, so $\hat{\mathfrak{a}} = \mathfrak{a}_-$.

If $G^{-2} \leq \lambda_1 \leq G^2$, then by Theorem 1 $\hat{\mathbf{a}} \in \{\mathbf{a}_+, \mathbf{a}_-\}$, so $\mathbf{a} = \hat{\mathbf{a}}$ or $\mathbf{a}$ is one of the neighbours of $\hat{\mathbf{a}}$. From Theorem 1 it can be proved that $G^{-1} \leq \rho_2 \leq G$ and hence $G^{-3} \leq \hat{\lambda}_1 < \frac{1 + 2^{-p}}{1 - G^3 2^p} G^3$. So both communication partners can determine an ideal $\mathbf{b}$ which is either $\hat{\mathbf{a}}$ or a neighbour of $\hat{\mathbf{a}}$ such that $G^{-3} \leq \hat{\lambda}(\mathbf{b}, abc) < \frac{1 + 2^{-p}}{1 - G^3 2^p} G^3$. Then it can be shown that $\frac{1}{1 + \frac{1}{\sqrt{\Delta}}} < \lambda(\hat{\mathbf{a}}, \mathbf{b}) < 1 + \frac{1}{\sqrt{\Delta}}$ therefore by Lemma 1: $\hat{\mathbf{a}} = \mathbf{a}$. ♦

We are now equipped to set up the protocol. We assume $2^p \geq 1280d(d^2 - 1)$.

<u>**Protocol**</u>:

The two communication partners Alice and Bob perform the following steps:

1) Both Alice and Bob agree on D and a small positive integer r. They compute $\mathbf{a} = \hat{\mathbf{a}}(c)$, $\mathrm{M} = \mathrm{M}_2(c) \geq \gamma$ using Algorithm 6 where $c = \log r$. D, $\mathbf{a}$, and M can be made public.

2) Alice secretly chooses $a \in \{1, \ldots, d\}$ and from $\mathbf{a}$, M computes $\hat{\mathbf{a}}(ac)$, $\mathrm{M}_2(ac) \geq \gamma$ using Algorithm 5. She sends both to Bob.

3) Bob secretly chooses $b \in \{1, \ldots, d\}$ and from $\mathbf{a}$, M computes $\hat{\mathbf{a}}(bc)$, $\mathrm{M}_2(bc) \geq \gamma$ using Algorithm 5. He sends both to Alice.

4) From $\hat{\mathbf{a}}(ac)$, $\mathrm{M}_2(ac)$, and b, Bob computes $\hat{\mathbf{a}}(abc)$ and its two neighbours as well as their approximate distances (i.e. M values) using Algorithms 5 and 1. If he finds among these an ideal $\mathbf{b}$ such that $\frac{2^p}{G^3} \leq \mathrm{M}(\mathbf{b}, abc) < \frac{(1 + 2^p)G^3}{1 - 2^p G^3}$, then $\mathbf{b} = \mathbf{a}(abc)$. In this case he sends

'0' back to Alice. If he cannot find such an ideal, then by Lemma 6 he can compute $\mathbf{a}_{-}(abc)$. In this case he sends '1' to Alice.

5) From $\hat{\mathbf{a}}(bc)$, $M_2(bc)$, and a, Alice computes $\hat{\mathbf{a}}(abc)$, $M_2(abc)$ using Algorithm 5. If she received '0' from Bob, then she computes the neighbours of $\hat{\mathbf{a}}(abc)$ and their M values and attempts to compute $\mathbf{a}(abc)$. If successful, she sends '0' back to Bob. The common key is then $\mathbf{a}(abc)$. Otherwise the ideal $\hat{\mathbf{a}}(abc)$ she computed is $\mathbf{a}_{-}(abc)$. In this case she sends '1' to Bob. If Alice received '1' from Bob, then he was unable to determine $\mathbf{a}(abc)$, so we must have $\lambda_1(abc) < G^{-2}$ or $\lambda_1(abc) > G^2$ by Lemma 6, in which case the ideal $\hat{\mathbf{a}}(abc)$ computed by Alice is $\mathbf{a}_{-}(abc)$. This is then the key. In this case she sends '1' back to Bob.

6) If Bob receives the same bit he sent, then the ideal he computed in step 4 is the key. The only other possibility is that he sent '0' and received '1'. In this case Alice was unable to determine $\mathbf{a}(abc)$. The key is then the ideal $\hat{\mathbf{a}}(abc) = \mathbf{a}_{-}(abc)$ initially computed by Bob.

References:

[1] J. A. Buchmann, H. C. Williams, *A key exchange system based on real quadratic fields*, extended abstract, to appear in: Proceedings of CRYPTO '89.

[2] W. Diffie, M. Hellman, *New directions in cryptography*, IEEE Trans. Inform. Theory, vol. 22, 1976.

[3] R. A. Mollin, H. C. Williams, *Computation of the class number of a real quadratic field*, to appear in: Advances in the Theory of Computation and Computational Mathematics (1987).

[4] A. J. Stephens, H. C. Williams, *Some computational results on a problem concerning powerful numbers*, Math. of Comp. vol. 50, no. 182, April 1988.

[5] H. C. Williams, M. C. Wunderlich, *On the parallel generation of the residues for the continued fraction factoring algorithm*, Math. of Comp. vol. 48, no. 177, January 1987.