A One Round Protocol for Tripartite Diffie–Hellman

Antoine Joux

SCSSI, 18, rue du Dr. Zamenhoff
F-92131 Issy-les-Mx Cedex, France
Antoine.Joux@ens.fr

Abstract. In this paper, we propose a three participants variation of the Diffie–Hellman protocol. This variation is based on the Weil and Tate pairings on elliptic curves, which were first used in cryptography as cryptanalytic tools for reducing the discrete logarithm problem on some elliptic curves to the discrete logarithm problem in a finite field.

1 Introduction

Since its discovery in 1976, the Diffie–Hellman protocol has become one of the most famous and largely used cryptographic primitive. In its basic version, it is an efficient solution to the problem of creating a common secret between two participants. Since this protocol is also used as a building block in many complex cryptographic protocols, finding a generalization of Diffie–Hellman would give a new tool and might lead to new and more efficient protocols.

In this paper, we show that the Weil and Tate pairings can be used to build a tripartite generalization of the Diffie–Hellman protocol. These pairings were first used in cryptography as cryptanalytic tools to reduce the complexity of the discrete logarithm problem on some "weak" elliptic curves. Of course, the problem of setting a common key between more than two participants has already been addressed (see the protocol for conference keying in [1]). However, all the known techniques require at least two round of communication. In some protocols having these two rounds can be somewhat cumbersome, and a single round would be much preferable. To give an example, exchanging an email message key with a two round Diffie–Hellman protocol would require both participants to be connected at the same time, which is a very undesirable property for a key exchange protocol. For this reason, we believe that the one round tripartite Diffie–Hellman presented here is a real improvement over conference keying even though the computational cost will be somewhat higher.

2 The Discrete Logarithm Problem on Weak Elliptic Curve

The discrete logarithm problem on elliptic curves is now playing an increasingly important role in cryptography. When elliptic curve cryptosystems where first

W. Bosma (Ed.): ANTS-IV, LNCS 1838, pp. 385–393, 2000.

proposed in [9], computing the number of points of a given curve was a challenging task, since the Schoof, Elkies and Atkin algorithm was not yet mature (for a survey of this algorithm see [6]). For this reason and also to simplify the addition formulas, the idea of using special curves quickly arose. However, it was shown later on that some of these special cases are not good enough. Today, three weak special cases have been identified. In one of them, the discrete logarithm problem becomes easy (i.e. polynomial time) as was shown in [11,10]. This easiest case happens when the number of points of the elliptic curve over $\mathbb{F}_p$ is exactly p. In the two other cases, the discrete logarithm problem on the elliptic curve is transformed into a discrete logarithm problem in a small extension of the field of definition of the elliptic curve. These two reductions are called the Menezes, Okamoto, Vanstone (MOV) reduction [8] and the Frey, Rück (FR) reduction [3]. A survey of these reductions was published at Eurocrypt'99 [4], and gave a comparison of these two reductions. The conclusion was the FR reduction can be applied to more curves than the MOV reduction and moreover that it can be computed faster than the MOV reduction. Thus for all practical usage, the authors recommend the FR reduction. However, they claim that the computation of the FR and MOV reduction may be a heavy load. We will show that in fact this is not the case and that these reductions can be turned from cryptanalytic to cryptographic tools.

Pairings on Elliptic Curve

The MOV and FR reductions are both based on a bilinear pairing, in the MOV case it is the Weil pairing and in the FR case it is (a variation of) the Tate pairing. In the sequel, we describe these pairings for an elliptic curve E defined over $\mathbb{F}_p$. In order to define these pairings, we first need to introduce the function field and the divisors of the elliptic curve. Very informally, the function field $K(E)$ of E is the set of rational map in x and y modulo the equation of E (e.g. $y^2 - x^3 - ax - b$). A divisor D is an element of the free group generated by the points on E, i.e. it can be written as a finite formal sum: $D = \sum_i a_i(P_i)$, where the P_i are points on E and the a_i are integers. In the sequel, we will only consider divisors of degree 0, i.e. such that $\sum_i a_i = 0$.

Given any function f in $K(E)$, we can build a degree 0 divisor $div(f)$ from the zeros and poles of f simply by forming the formal sum of the zeroes (with multiplicity) minus the formal sum of the poles (with multiplicity). Any divisor $D = div(f)$ will be called a principal divisor. In the reverse direction, testing whether a degree 0 divisor $D = \sum_i a_i(P_i)$ is principal or not, can be done by evaluating $\sum a_i P_i$ on E. The result will be the point at infinity if and only if D is principal.

Given a function f in $K(E)$ and a point P of E, f can be evaluated at P by substituting the coordinates of P for x and y in any rational map representing f. The function f can also be evaluates at a divisor $D = \sum_i a_i(P_i)$, using the following definition:

$$f(D) = \prod_i f(P_i)^{a_i}.$$