Colored Burau Matrices, E-multiplication, and the Algebraic Eraser™ Key Agreement Protocol

SecureRF Corporation
100 Beard Sawmill Road
Suite 350
Shelton, CT 06484

203-227-3151
info@SecureRF.com
www.SecureRF.com

Finite Fields: An example.

In mathematics, the concept of a field emerges as an abstraction of the real numbers: a field is a set F, where it is possible to add, subtract, multiply, and divide elements in F and obtain new elements in F. These operations are assumed to behave in familiar ways: for example, in the case of addition there is some element in the set F, denoted 0, which behaves like the number zero in that for any $a \in F$

$$a + 0 = 0 + a,$$

and given two elements $a, b \in F$,

$$a + b = b + a.$$

To get a sense of what working with a field is like we move away from this abstract concept, and consider the set of integers $\boldsymbol{F_7} = \{0, 1, 2, 3, 4, 5, 6\}$. In order to define the operations in $\boldsymbol{F_7}$, we need to use the notation mod p, which was introduced by C. F. Gauss in the late 18th century. Given a prime number p, we say two integers, a, b are equal *mod* p, denoted

$$a = b \bmod p,$$

if b is the remainder when we divide a by p. For example,

$$11 = 4 \bmod 7, \quad 29 = 1 \bmod 7.$$

Using this notation the set $\boldsymbol{F_7}$ becomes as field when we define the following operations: given $a, b \in \boldsymbol{F_7}$

$$a \oplus b = (a + b) \bmod 7, \quad a \otimes b = a \cdot b \bmod 7.$$

For example,

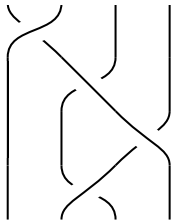
$$3 \oplus 6 = 2, \quad 2 \oplus 5 = 0$$

$$3 \otimes 5 = 1, \quad 2 \otimes 6 = 5.$$

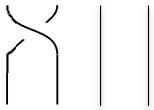
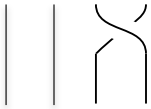
The finite field $\boldsymbol{F_7}$ can be used in many of the same ways real numbers can be: matrices can have entries in $\boldsymbol{F_7}$, polynomials can have coefficients in $\boldsymbol{F_7}$, and elements in $\boldsymbol{F_7}$ can be plugged into such polynomials.

From Braids to Matrices and E-multiplication.

In order to illustrate the core operation of AE based cryptography, we will work with the 4 stranded braid group, B_4 (see *An Introduction to the Mathematics of Braids*). Working with B_N, while technically more challenging, is analogous. Each braid $\beta \in B_4$, say

$\beta =$ 

is the product of single crossing braids

$b_1 =$  $b_2 =$  $b_3 =$ 

and their inverses. In this case, $\beta = b_1^{-1}b_2b_3b_2^{-1}$. Each of the single crossing braids, b_i (or its inverse), is associated with the permutation of the set {1,2,3,4}, denoted σ_i, defined by

$$\sigma_i: i \to i+1, \qquad i+1 \to i, \qquad j \to j, \text{ if } j \neq i, i+1.$$

Next we associate with each b_i, a matrix whose entries are polynomials in the variables $\{t_1, t_2, t_3, t_4\}$, and a method of hybrid matrix multiplication of matrix/ permutation pairs associated to the single crossing braids that allows us to work with braids in a cryptographic setting. The matrices associated with each b_i is denoted $CB(b_i)$ and is called the colored Burau matrix of b_i. The matrices are defined as follows:

$$CB(b_1) = \begin{pmatrix} -t_1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$CB(b_2) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ t_2 & -t_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad CB(b_3) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Since any permutation of the set $\{1,2,3,4\}$ can be made to act on the variables $\{t_1, t_2, t_3, t_4\}$, it can act on any polynomial in these variables and any matrix with such polynomial entries. With these observations in mind we can define the product of the colored Burau/permutation pairs denoted $\circ$

$$\{CB(b_1), \sigma_1\} \circ \{CB(b_2), \sigma_2\} =$$

$$\left\{ \begin{pmatrix} -t_1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \sigma_1 \right\} \circ \left\{ \begin{pmatrix} 1 & 0 & 0 & 0 \\ t_2 & -t_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \sigma_2 \right\} =$$

$$\left\{ \begin{pmatrix} -t_1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot {}^{\sigma_1}\begin{pmatrix} 1 & 0 & 0 & 0 \\ t_2 & -t_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \sigma_1 \cdot \sigma_2 \right\}$$

where

$${}^{\sigma_1}\begin{pmatrix} 1 & 0 & 0 & 0 \\ t_2 & -t_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

indicates the permutation σ_1 acting on the matrix,

$${}^{\sigma_1}\begin{pmatrix} 1 & 0 & 0 & 0 \\ t_2 & -t_2 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ t_1 & -t_1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and the dot notation · indicates matrix multiplication and permutation composition as needed. The above multiplication of colored Burau matrices/permutation pairs mimics the multiplication introduces for braids and gives a sense of the complex structure the braid group has. The appropriate colored Burau matrices for inverses of the single crossing braids arise from the above product formula and are given as follows:

$$CB(b_1^{-1}) = \begin{pmatrix} -\frac{1}{t_2} & \frac{1}{t_2} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad CB(b_2^{-1}) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & -\frac{1}{t_3} & \frac{1}{t_3} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

$$CB(b_3^{-1}) = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & -\frac{1}{t_4} & \frac{1}{t_4} \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Given any braid expressed as a product of the single crossing braids and their inverses,

$$\beta = b_{i_1}^{\epsilon_1} b_{i_2}^{\epsilon_2} \cdots b_{i_k}^{\epsilon_k}$$

where $i_1, i_2, \dots, i_k \in \{1,2,3\}$, and $\epsilon_1, \epsilon_2, \dots, \epsilon_k \in \{+1, -1\}$, can be associated with product of colored Burau/permutation pairs in the natural way: letting the permutation associated to β be denoted by σ_β,

$$\left(CB(\beta), \sigma_\beta\right) = \left(CB\left(b_{i_1}^{\epsilon_1}\right), \sigma_{i_1}\right) \circ \left(CB\left(b_{i_2}^{\epsilon_2}\right), \sigma_{i_2}\right) \circ \cdots \circ \left(CB\left(b_{i_k}^{\epsilon_k}\right), \sigma_{i_k}\right).$$

The entries in the matrix $CB(\beta)$ become very large and complex as the braid grows longer, thus an additional tool is needed to move to cryptographic applications.

When a collection of four non-zero field elements, termed $t-$ values, are specified

$$t-\text{values} = \{\tau_1, \tau_2, \tau_3, \tau_4\} \subset \square_7\square$$

a polynomial $f(t_1, t_{2,} t_3, t_4)$ in 4 variables and coefficients in $\square_7$, the notation,

$$f(t_1, t_2, \dots, t_N) \downarrow_{t-\text{values}},$$

is referred to as applying the $t-$ values, is given by

$$f(t_1, t_2, \dots, t_N) \downarrow_{t-\text{values}} = f(\tau_1, \tau_2, \dots, \tau_N).$$

A matrix with polynomial entries can be evaluated at the $t-$ values by simply plugging the $t-$ values into the polynomial entries. With all these components in place we can now define e-multiplication.

E-multiplication is an operation that inputs two ordered pairs,

$$(M, \sigma_0),\ (\beta, \sigma_\beta),$$

where M is a 4x4 matrix with entries in $\square_7$, σ_0 is a permutation of the set {1,2,3,4}, $\beta \in B_N$, and σ_β is the permutation associated to β. The output of e-multiplication, denoted,

$$(M', \sigma') = (M, \sigma_0) \star (\beta, \sigma_\beta),$$

is another ordered pair, (M', σ'), where again M' is a 4x4 matrix with entries in $\square_7$, σ' is a permutation of the set {1,2,3,4}. The definition of e-multiplication when the braid $\beta = b_i^{\pm 1}$ is given by

$$(M, \sigma_0) \star \left(b_i^{\pm 1}, \sigma_{b_i^{\pm 1}}\right) = \left(M \cdot^{\sigma_0} \left(CB\left(b_i^{\pm 1}\right)\right) \downarrow_{t-values},\ \sigma_0 \cdot \sigma_{b_i^{\pm 1}}\right).$$

In the general case, when $\beta = b_{i_1}^{\epsilon_1} b_{i_2}^{\epsilon_2} \cdots b_{i_k}^{\epsilon_k}$, (where $\epsilon_i \in \{+1, -1\}$) the e-multiplication is executed iteratively:

$$(M, \sigma_0) \star (\beta, \sigma_\beta) = \left(\left((M, \sigma_0) \star \left(b_{i_1}^{\epsilon_1}, \sigma_{b_{i_1}}\right)\right) \star \left(b_{i_2}^{\epsilon_2}, \sigma_{b_{i_2}}\right)\right) \star \cdots \star \left(b_{i_k}^{\epsilon_k}, \sigma_{b_{i_k}}\right).$$

In order to get a real sense of how e-multiplication works an example is in order. Returning to the illustrative case of $N = 4$, and using the finite field $\boldsymbol{F_5} = \{0, 1, 2, 3, 4\}$, we begin by choosing $t - \text{values} = \{3, 2, 3, 4\}$, a braid

$$\beta = b_3 b_3 b_2 b_1 b_3^{-1} b_2^{-1} b_1 b_1,$$

whose permutation is given by

$$\sigma_\beta = \sigma_3\ \sigma_3\ \sigma_2\sigma_1\ \sigma_3^{-1}\sigma_2^{-1}\sigma_1\sigma_1,$$

and the matrix/permutation pair

$$(M, \sigma_0) = \left(\begin{pmatrix} 1 & 2 & 4 & 1 \\ 0 & 1 & 2 & 0 \\ 0 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (2\ 3\ 1\ 4) \right).$$

The iterative computation of $(M, \sigma_0) \star (\beta, \sigma_\beta)$ begins with e-multiplication

$$\left(\begin{pmatrix} 1 & 2 & 4 & 1 \\ 0 & 1 & 2 & 0 \\ 0 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (2\ 3\ 1\ 4) \right) \star \left\{ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & t_3 & -t_3 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (1\ 2\ 4\ 3) \right\}$$

$$= \left(\begin{pmatrix} 1 & 2 & 4 & 1 \\ 0 & 1 & 2 & 0 \\ 0 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & t_1 & -t_1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \downarrow_{\{3,2,3,4\}}, (2\ 3\ 4\ 1) \right)$$

$$= \left(\begin{pmatrix} 1 & 2 & 4 & 1 \\ 0 & 1 & 2 & 0 \\ 0 & 3 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 3 & -3 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (2\ 3\ 4\ 1) \right)$$

$$= \left(\begin{pmatrix} 1 & 4 & 3 & 0 \\ 0 & 2 & 4 & 2 \\ 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (2\ 3\ 4\ 1) \right),$$

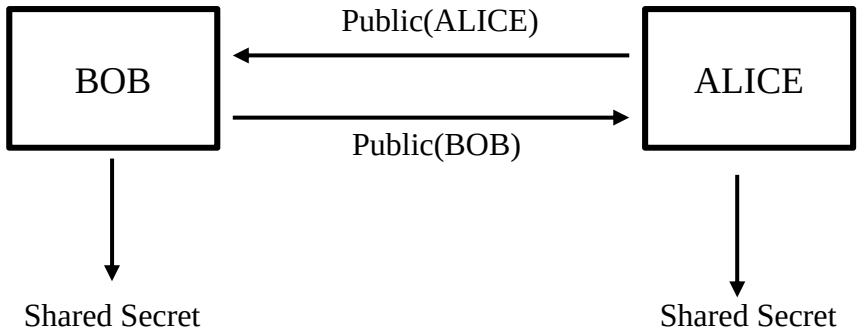
where all the arithmetic is performed in the finite field $\boldsymbol{F_5}$, thus for example $-3 = 2 \bmod 5$.

The next step is to e-multiply this output with $\{CB\{b_3), \sigma_3\}$ and then proceed to the end of the braid β. The final output will be

$$\left(\begin{pmatrix} 4 & 1 & 4 & 4 \\ 3 & 4 & 0 & 4 \\ 4 & 2 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{pmatrix}, (1\ 4\ 2\ 3) \right).$$

The AE Key Agreement Protocol.

The AEKAP is a protocol that enables two users, Alice and Bob, to evaluate a *shared secret*, κ, using their own private key and the public key of the other user.



At a high level the basic structure of the protocol is as follows. Each user, say Alice, will generate matrix with finite field entries, M_A, together with a braid β_A which contained in part of the braid group assigned to Alice. This pair, (M_A, β_A) is Alice's *private key*;

$$\text{Private(ALICE)} = (M_A, \beta_A).$$

Likewise, Bob will generate a matrix M_B and a braid β_B, and the pair (M_B, β_B) and again, this pair is Bob's *private key*;

$$\text{Private(BOB)} = (M_B, \beta_B).$$

Using the notation, id for the identity permutation (i.e., the permutation that keeps everything fixed), and letting σ_{β_A} denote the permutation associated with β_A, Alice the evaluates her public key by computing the e-multiplication,

$$\text{Public(ALICE)} = (M_A, \text{id}) \star (\beta_A, \sigma_{\beta_A}).$$

Similarly, letting σ_{β_B} denote the permutation associated with $\beta_{B,}$ Bob evaluates his public key by computing the e-multiplication,

$$\text{Public(BOB)} = (M_B, \text{id}) \star (\beta_B, \sigma_{\beta_B}).$$

Once Bob and Alice exchange their respective pubic keys, they can both evaluate the same shared secret κ: Alice evaluates

$$(M_A, \text{id}) \cdot \text{Public(BOB)} \star (\beta_A, \sigma_{\beta_A}).$$

and Bob evaluates

$$(M_B, \text{id}) \cdot \text{Public(ALICE)} \star (\beta_B, \sigma_{\beta_B}).$$

The above e-multiplication outputs are the identical (though they look quite different), due to the specifics of each users private key. Though the details are beyond the scope of this paper there are two key features to understand. Both matrices M_A and M_B are chosen to themselves be polynomials over the finite field of a fixed matrix of a special type. Thus $M_A \cdot M_B = M_B \cdot M_A$. Furthermore Alice and Bob are each assigned parts of the braid group whose elements commute with each other, and hence multiplying the braid β_A by β_B is the same as multiplying β_B by β_A. In summary, both Alice and Bob arrive at the shared secret

$$\kappa = (M_A, \mathrm{id}) \cdot \mathrm{Public(BOB)} \star \left(\beta_A, \sigma_{\beta_A}\right) = (M_B, \mathrm{id}) \cdot \mathrm{Public(ALICE)} \star \left(\beta_B, \sigma_{\beta_B}\right).$$

Shared secrets can be used in many ways including basic authentication and encryption. The process of evaluating the shared secret can be executed very efficiently, and users can generate new Private/Public key pairs at will. As a concluding remark, if there is a need to increase security of the key agreement protocol, the size of either Alice's and Bob's Private keys would be increased. The time required to evaluate of the shared secret κ would increase linearly as the key size increases linearly. This is one of many unique features of the method.