Theory used by libgfshare

Simon McVittie

23rd April 2006

Introduction

libgfshare implements Shamir secret sharing [SHAMIR] over the field $GF(2^8)$, instead of $GF(p)$ for a prime p as suggested by Shamir's paper. This document aims to prove the security and integrity of this scheme.

Note that while I believe this document to be correct, I accept no responsibility for loss or damage caused by relying on the correctness of my proof.

Definitions

Let F be a field with multiplicative identity 1 and additive identity 0.

If $A = \{(a_1, b_1), \cdots, (a_n, b_n)\}$, with the a_i distinct nonzero elements of F and the b_i elements of F, indexed by $I = \{1, \cdots, n\}$, then define

$$P_A(x) = \sum_{j \in I} b_j \prod_{k \in I, k \neq j} (x - a_k)(a_j - a_k)^{-1}$$

a polynomial of degree at most $n - 1$. (By distinctness of the a_i, the inverses required exist.) This is the Lagrange interpolating polynomial for the points in A.

Lemma 1

Let $a_1, \cdots, a_t \in F$ be distinct and nonzero; let $b_1, \cdots, b_{t-1}, c \in F$ be arbitrary. Then there exists $b_t \in F$ such that if $A = \{(a_1, b_1), \cdots, (a_t, b_t)\}$ then $P_A(0) = c$.

Proof

Let $I = \{1, \cdots, t\}$. We have

$$P_A(0) = \sum_{j \in I} b_j \prod_{k \in I, k \neq j} -a_k (a_j - a_k)^{-1} = \sum_{j \in I} y_j \prod_{k \in I, k \neq j} a_k (a_k - a_j)^{-1}$$

Let

$$b_t = \left[c + \sum_{j \in I, j \neq t} b_j \prod_{k \in I, k \neq j} a_k (a_j - a_k)^{-1} \right] \left[\prod_{k \in I, k \neq t} a_k^{-1} (a_k - a_t) \right]$$

Then

$$\begin{aligned} P_A(0) &= \sum_{j \in I, j \neq t} b_j \prod_{k \in I, k \neq j} a_k (a_k - a_j)^{-1} + b_t \prod_{k \in I, k \neq t} a_k (a_k - a_t)^{-1} \\ &= \sum_{j \in I, j \neq t} b_j \prod_{k \in I, k \neq j} a_k (a_k - a_j)^{-1} - \sum_{j \in I, j \neq t} b_j \prod_{k \in I, k \neq j} a_k (a_k - a_j)^{-1} + c \\ &= c \end{aligned}$$

as required.

Lemma 2

For any $x_1, \cdots, x_t$ distinct and nonzero elements of F, and any $y_1, \cdots, y_t, u$ arbitrary elements of F, let

$$X = \{(x_1, y_1), \cdots, (x_t, y_t)\}$$

and

$$U = \{(x_1, y_1), \cdots, (x_{t-1}, y_{t-1}), (u, P_X(u))\}$$

Then $P_X = P_U, i.e. P_X(x) = P_U(x)$ for all $x \in F$.

Proof

Let $S_{a,b} = \{(x_1, y_1), \cdots, (x_{t-1}, y_{t-1}), (a, b)\}$. Then

$$P_{S_{a,b}}(x) = \sum_{j<t} y_j (x-a)(x_j-a)^{-1} \prod_{k\neq j, k<t} (x-x_k)(x_j-x_k)^{-1} + b \prod_{k<t} (x-x_k)(a-x_k)^{-1}$$

Hence if we let $d_{i,j} = x_i - x_j$ and $e_i = u - x_i$ (both of which are necessarily nonzero, by distinctness of the x_i and u) we have

$$P_X(u) = \sum_{j<t} y_j e_t d_{j,t}^{-1} \prod_{k\neq j, k<t} e_k d_{j,k}^{-1} + y_t \prod_{k<t} e_k d_{t,k}^{-1}$$

and if we also let $f_i = x - x_i$,

$$P_U(x) = \sum_{j<t} y_j (u-x) e_j^{-1} \prod_{k\neq j, k<t} f_k d_{j,k}^{-1} + P_X(u) \prod_{k<t} f_k e_k^{-1}$$

$$P_U(x) = \sum_{j<t} y_j (u-x) e_j^{-1} \prod_{k\neq j, k<t} f_k d_{j,k}^{-1} + \left\{\prod_{k<t} f_k e_k^{-1}\right\} \left\{\sum_{j<t} y_j e_t d_{j,t}^{-1} \prod_{l\neq j, l<t} e_k d_{j,l}^{-1} + y_t \prod_{l<t} e_l d_{t,l}^{-1}\right\}$$

Expanding,

$$\begin{aligned} P_U(x) &= \textstyle\sum_{j<t} y_j (u-x) e_j^{-1} \left\{\prod_{k\neq j, k<t} f_k d_{j,k}^{-1}\right\} \\ &\quad + \textstyle\sum_{j<t} y_j e_t d_{j,t}^{-1} \left\{\prod_{l\neq j, l<t} e_k d_{j,l}^{-1}\right\} \left\{\prod_{k<t} f_k e_k^{-1}\right\} \\ &\quad + y_t \left\{\textstyle\prod_{k<t} e_k d_{t,k}^{-1} f_k e_k^{-1}\right\} \end{aligned}$$

$$P_U(x) = \sum_{j<t} y_j \left[(u-x) e_j^{-1} \left\{\prod_{k\neq j, k<t} f_k d_{j,k}^{-1}\right\} + e_t d_{j,t}^{-1} f_j e_j^{-1} \left\{\prod_{k\neq j, k<t} e_k d_{j,k}^{-1} f_k e_k^{-1}\right\}\right] + y_t \prod_{k<t} d_{t,k}^{-1} f_k$$

$$= \sum_{j<t} \left[y_j \prod_{k\neq j, k<t} f_k d_{j,k}^{-1}\right] \left[(u-x) e_j^{-1} + e_t e_j^{-1} d_{j,t}^{-1} f_j\right] + y_t \prod_{k<t} d_{t,k}^{-1} f_k$$

Now

$$\begin{aligned} (u-x) e_j^{-1} + e_t e_j^{-1} d_{j,t}^{-1} f_j &= (e_j^{-1} d_{j,t}^{-1}) \left[(u-x) d_{j,t} + e_t f_j\right] \\ &= (e_j^{-1} d_{j,t}^{-1}) \left[(u-x)(x_j - x_t) + (u-x_t)(x-x_j)\right] \end{aligned}$$

$$= (e_j^{-1} d_{j,t}^{-1})(x - x_t)(u - x_j)$$
$$= d_{j,t}^{-1} f_t$$

Hence

$$P_U(x) = \sum_{j<t} \left[y_j \prod_{k \neq j, k<t} f_k d_{j,k}^{-1} \right] \left[f_t d_{j,t}^{-1} \right] + y_t \prod_{k<t} d_{t,k}^{-1} f_k = P_X(x)$$

as required.

Construction

Let s be the number of "shares" and t be the required threshold to recover the shared secret (i.e. we construct a "t of s" share).

Given a secret $f \in F$ we may construct a Lagrange interpolating polynomial P_X of degree no more than $t-1$, with $P_X(0) = f$, as follows:

- choose distinct nonzero $x_1, \cdots, x_s \in F$

- choose arbitrary (and unpredictable) $y_1, \cdots, y_{t-1} \in F$

- use Lemma 1 to select y_t such that $X = \{(x_1, y_1), \cdots, (x_t, y_t)\}$ has the desired intercept f

To obtain additional shares, calculate $y_{t+1} = P_X(x_{t+1}), \cdots, y_s = P_X(x_s)$.

Alternate construction, as used in libgfshare

In libgfshare the construction used is as follows:

- construct a polynomial P by choosing arbitrary and unpredictable coefficients of $x, \cdots, x^{t-1}$ from F, and setting the coefficient of x^0 to f: this therefore has the desired intercept f

- choose distinct nonzero $x_1, \cdots, x_s \in F$ and evaluate $y_1 = P(x_1), \cdots, y_s = P(x_s)$

Proof of equivalence in a finite field F

Suppose F is finite, as is the case in libgfshare, and that in each construction, arbitrary choices are made from among all possible values in F.

In the alternate construction, given $x_1, \cdots, x_t, f$ we choose a polynomial $P(x) = f + m_1 x + \cdots + m_{t-1}x^{t-1}$ by choosing arbitrary coefficients $m_1, \cdots, m_{t-1} \in F$, i.e. choosing arbitrarily from among the $|F|^{t-1}$ distinct polynomials of degree no more than $t-1$ with intercept f.

In the first construction, given $x_1, \cdots, x_t, f$ we obtain a polynomial by choosing arbitrary $y_1, \cdots, y_{t-1} \in F$. The polynomials chosen are necessarily distinct since no polynomial can pass through both (x_i, p) and (x_i, q) for any $p \neq q$, so by choosing each y_i from among the $|F|$ elements of F, we choose arbitrarily from a set of $|F|^{t-1}$ distinct polynomials whose intercepts are all f.

Since there are only $|F|^{t-1}$ such polynomials, each construction chooses arbitrarily from among the same set, and by the pigeonhole principle there exists a bijective mapping between sets of arbitrary y values in the first construction and sets of arbitrary coefficients in the second.

Theorem: With at least t pieces the secret is recoverable

Let $B \subset \{(x_1, y_1), \cdots, (x_s, y_s)\}$ with $|B| = t$. Then $P_B(0) = c$.

Further, if $B' \subset \{(x_1, y_1), \cdots, (x_s, y_s)\}$ with $|B'| > t$, then for every subset B of B' with $|B| = t$, $P_B(0) = f$.

Proof

The second part is trivially implied by the first.

Recall that $X = \{(x_1, y_1), \cdots, (x_t, y_t)\}$ and that $P_X(0) = f$. If $B = X$ the result is true. If not, repeatedly apply Lemma 2 to replace an element of X not in B with an element of B not in X, preserving the value of $P(0)$.

Theorem: With fewer than t pieces no information is gained

Let $C \subset \{(x_1, y_1), \cdots, (x_s, y_s)\}$ with $|C| < t$. Then for each $d \in F$, there exists $D \supset C$, $|D| = t$, such that $d = P_D(0)$.

(In other words, any $d \in F$ remains a possible value for the secret, so an attacker with fewer than t shares has gained no information.)

Proof

Let a_i, b_i be such that $C = \{(a_1, b_1), \cdots, (a_n, b_n)\}$, some $n < t$. Choose arbitrary $a_{n+1}, \cdots, a_t$ and arbitrary $b_{n+1}, \cdots, b_{t-1}$. Let b_t be chosen by applying Lemma 1 with $c := d$. Then by choice of b_t, $P_C(0) = d$ as required.

Implementation in $GF(2^8)$

The program `test_gfshare_isfield`, compiled and run by `make check`, demonstrates that the calculations done by libgfshare are indeed performed in a field.

Attacks not addressed

This document has not addressed the following:

- Attacks based on the use of a predictable or partially predictable pseudorandom number generator might be possible.

- In the implementation used in libgfshare, the field F is the field of byte values, with addition being bitwise exclusive-or, and multiplication as usual; each byte of the secret is shared separately by applying this algorithm separately. This means that when a secret file is shared, the length in bytes of each share equals the length in bytes of the secret. If the length of the secret is itself secret, it should be padded to some standard length before sharing.

References

[SHAMIR] Adi Shamir, "How to share a secret", Communications of the ACM, 22(1), pp612–613, 1979. Available at http://www.cs.tau.ac.il/~bchor/Shamir.html

Copyright and disclaimer