

# Private Location Verification

Vadym Fedyukovych

September 27, 2018

## Abstract

We present location verification protocol and implementation that allows for location privacy. With a Schnorr-like protocol, we verify that location committed is close enough to another known location. Protocol was implemented with Crypto++ library.

## 1 Introduction

Location is considered as a point within a polygon. Known results and state of the art. SNARK-based and 'older' interactive proofs-based solutions. This is the last line.

### 1.1 Our contribution

## 2 Protocol

### 2.1 Definitions

### 2.2 Notations

Airdrop location  $(x_l, y_l)$  available in clear, node location  $(x_n, y_n)$  hidden (committed), acceptable maximum distance  $d$  from node to airdrop, setup: group description and parameters, group elements for making commitments, initial message, challenge and responses of a Schnorr-like protocol.

### 2.3 Interactive proof

Interactive argument system about integers is designed with a group of a hidden order  $q$ , that is, order of the group is not available to the Prover. Inequality (not far from) statement is converted into equality with 4-squares Lagrange theorem (Lipmaa). Schnorr proof was extended into a proof systems for polynomial relations with polynomials of higher degree in challenge for a number of applications [3, 2]. A comparable proof system for integers was introduced at [4].

## 2.4 Proof setup

Proof system for relations about integers is well described at Idemix documentation [1]

Multiplicative group of residue classes, RSA-like modulus.

## 2.5 Non-Intractive proof

## 2.6 Security properties

# 3 Implementation

We have this protocol implemented on top of Crypto++ library <sup>1</sup> serving as a bignumbers backend.

Producing four-squares witness [5] is a work in progress, and is not a part of the protocol reported. To facilitate larger proof-of-concept application, a temporary approximate solution was introduced producing four squares.

# 4 Discussion and Conclusion

## References

- [1] Jan Camenisch. Specification of the identity mixer cryptographic library RZ 3730 version 2.3.0, 2010.
- [2] Giovanni Di Crescenzo and Vadym Fedyukovich. Zero-knowledge proofs via polynomial representations. In *Proceedings of the 37th International Conference on Mathematical Foundations of Computer Science*, pages 335–347, 2012.
- [3] Vadym Fedyukovich. An argument for Hamiltonicity. Cryptology ePrint Archive, Report 2008/363, 2008.
- [4] Vadym Fedyukovich. Proving outcome of private statistical signal testing. In *Statistical Methods of Signal and Data Processing*, pages 172–175, 2010.
- [5] Paul Pollack and Enrique Trevino. Finding four squares in Lagrange’s theorem.

---

<sup>1</sup><https://cryptopp.com/>

Common input of Prover and Verifier is commitment  $s_U$  to node location (1), airdrop location  $(x_l, y_l, z_l)$  and threshold  $d^2$  (integers), parameters  $(N, g, g_x, g_y, g_z, g_r, \{h_j\})$ .

$$s_U = g_x^{x_n} g_y^{y_n} g_z^{z_n} g^r \pmod{N} \quad (1)$$

Private input of Prover is node location  $(x_n, y_n, z_n)$  (integers) and location commitment randomness  $r$ , four numbers  $\{a_j\}$  calculated according to (2). Statement being proved is

$$d^2 - ((x_n - x_l)^2 + (y_n - y_l)^2 + (z_n - z_l)^2) = \sum_{j=1}^4 a_j^2 \quad (2)$$

Protocol runs as follows:

1. Prover picks random  $\{\alpha_j\}, \eta, \gamma, \beta_x, \beta_y, \beta_z, \beta_r, \rho_0, \rho_1$ , produces  $f_0, f_1$ , sends  $b_0, b_1, t_a, s_a, t_n$ :

$$f_0 = -(\beta_x^2 + \beta_y^2 + \beta_z^2) - \sum_{j=1}^4 \alpha_j^2 \quad (3)$$

$$f_1 = -((x_n - x_l)\beta_x + (y_n - y_l)\beta_y + (z_n - z_l)\beta_z) - \sum_{j=1}^4 a_j \alpha_j \quad (4)$$

$$t_n = g_x^{\beta_x} g_y^{\beta_y} g_z^{\beta_z} g^{\beta_r}, \quad s_a = g^\gamma \prod_{j=1}^4 h_j^{\alpha_j}, \quad t_a = g^\eta \prod_{j=1}^4 h_j^{\alpha_j} \quad (5)$$

$$b_0 = g^{f_0} g_r^{\rho_0}, \quad b_1 = g^{f_1} g_r^{\rho_1} \pmod{N} \quad (6)$$

2. Verifier chooses and sends his challenge  $c$
3. Prover produces and sends responses

$$X_n = cx_n + \beta_x, \quad Y_n = cy_n + \beta_y, \quad Z_n = cz_n + \beta_z, \quad R = cr + \beta_r \quad (7)$$

$$A_j = ca_j + \alpha_j, \quad R_a = c\gamma + \eta, \quad R_d = c\rho_1 + \rho_0 \quad (8)$$

4. Verifier accepts if

$$g_x^{X_n} g_y^{Y_n} g_z^{Z_n} g^R s_U^{-c} = t_n, \quad g^{R_a} \left( \prod_{j=1}^4 h_j^{A_j} \right) s_a^{-c} = t_a \quad (9)$$

$$g^{c^2 d^2 - ((X_n - cx_l)^2 + (Y_n - cy_l)^2 + (Z_n - cz_l)^2) - (A_1^2 + A_2^2 + A_3^2 + A_4^2)} g_r^{R_d} = b_1^c b_0 \pmod{N} \quad (10)$$

Figure 1: Private location verification protocol

Input of Prover is location commitment  $s_U$  (1), location  $(x_n, y_n, z_n)$  and random  $r$  to open this commitment, airdrop location  $(x_l, y_l, z_l)$ , threshold  $d^2$ , parameters  $(N, g, g_x, g_y, g_z, g_r, h_j)$ .

Non-interactive proof is produced as follows:

1. Prover calculates  $a_1 \dots a_4$  from locations and threshold, picks random  $\{\alpha_j\}, \eta, \gamma, \beta_x, \beta_y, \beta_z, \beta_r, \rho_0, \rho_1$ , produces  $t_n, s_a, t_a, b_0, b_1$ :

$$t_n = g_x^{\beta_x} g_y^{\beta_y} g_z^{\beta_z} g_r^{\beta_r}, \quad s_a = g^\gamma \left( \prod_{j=1}^4 h_j^{a_j} \right), \quad t_a = g^\eta \left( \prod_{j=1}^4 h_j^{\alpha_j} \right) \pmod{N} \quad (11)$$

$$\tilde{f}_0 = \beta_x^2 + \beta_y^2 + \beta_z^2 + \alpha_1^2 + \alpha_2^2 + \alpha_3^2 + \alpha_4^2 \quad (12)$$

$$\tilde{f}_1 = (x_n - x_l)\beta_x + (y_n - y_l)\beta_y + (z_n - z_l)\beta_z + a_1\alpha_1 + a_2\alpha_2 + a_3\alpha_3 + a_4\alpha_4 \quad (13)$$

$$b_0 = g^{\tilde{f}_0} g_r^{\rho_0}, \quad b_1 = g^{2\tilde{f}_1} g_r^{\rho_1} \pmod{N} \quad (14)$$

2. Prover produces his challenge with a hash function:

$$c = H(t_n || s_a || t_a || b_1 || b_0 || s_U) \quad (15)$$

3. Prover produces responses:

$$\begin{aligned} X_n &= -cx_n + \beta_x, \quad Y_n = -cy_n + \beta_y, \quad Z_n = -cz_n + \beta_z, \quad R = -cr + \beta_r \\ A_j &= -ca_j + \alpha_j, \quad R_a = -c\gamma + \eta, \quad R_d = -c\rho_1 + \rho_0 \end{aligned} \quad (16)$$

Non-interactive proof is  $(c, X_n, Y_n, Z_n, R, \{A_j\}, R_a, R_d, s_a, b_1)$ .

Proof verification:

$$\begin{aligned} F_d &= ((X_n + cx_l)^2 + (Y_n + cy_l)^2 + (Z_n + cz_l)^2) + (A_1^2 + A_2^2 + A_3^2 + A_4^2) - c^2 d^2 \\ &H(g_x^{X_n} g_y^{Y_n} g_z^{Z_n} g^R s_U^c || s_a || g^{R_a} \left( \prod_{j=1}^4 h_j^{A_j} \right) s_a^c || b_1 || g^{F_d} g_r^{R_d} b_1^c || s_U) = c \end{aligned} \quad (17)$$

Figure 2: Non-interactive location proof generation and verification