Secrect Sharing Progress

Joan Ngure

Report

How does making secret shares short work?
Distribution Scheme:

1. Choose a random encryption key K. Encrypt the secret S using the encryption function ENC under the key K, let $E = ENC_x(S)$.
2. Using Information dispersal algorithm (IDA) partition the encrypted file E into n fragments, $E_1, E_2, \cdots, E_n$
3. Using perfect secret sharing scheme (PSS) generate n shares for the key K, denoted $K_1, K_2, \cdots, K_n$.
4. Send to each participant $P_i, i = 1, 2, \cdots, n$ the share $S_i = (E_i, K_i)$. The portion K_i is privately transmitted to P_i (e.g. using encryption or any other secure way).

Reconstruction Scheme:

1. Collect from m participants $P_{ij}, j = 1, 2, ., ., m$ their shares $S_i, = (E_{ij}, K_{ij})$.
2. Using IDA reconstruct E out of the collected values $E_{ij}, j = 1, 2, \cdots, m$.
3. Using PSS recover the key K out of $K_{ij}, j = 1, 2, \cdots, m$.
4. Decrypt E using K to recover the secret S.

Example: Scheme is (3,5).
We have our secret S and using our key K we encrypt it. We can use for example symmetric key encryption scheme which is length preserving. In this example since the secret is a bit short I will assume $|K| = |S|$ and $E = S \oplus K$.

Let S = 110010011100011
and K = 010100110011000
then E = 100110101111011

Dividing E into 3 bits:
E_1 = 10011
E_2 = 01011

$E_3 = 11011$
We can convert the bits in decimal and choose a prime larger than 2^5 say 37.
$E_1 = 19$
$E_2 = 11$
$E_3 = 27$
The polynomial will be in the form:
$E(x) = E_1 + E_2x + E_3x^2 = 19 + 11x + 27x^2$
$E(1) = 57 \mod 37 = 20$
$E(2) = 1$
$E(3) = 36$
$E(4) = 14$
$E(5) = 9$
We use the same procedure to share the key. If 3 people come together, they can reconstruct the secret and the key. The encryption scheme is well defined and known to the participants. They know after reconstruction the secret is of the form $E_1E_2E_3$ and the numbers are then converted to binary. The key is in the form $K_1K_2K_3$ and it is used to decrypt the message.