Shamir Secret Sharing Algorithm

Approach: At first we need to choose a secret key that will be shared among multiple parties. Then after that we need to choose parameters of the Shamir Secret Sharing Algorithm which includes the number of parties that we want to distribute the secret to and the minimum number of shares required to reconstruct the secret. Then after that we need to generate a random polynomial of degree where the constant term is the secret After that we need to evaluate the polynomial at different points After evaluating the share, we will Distribute the shares . Each party will have a unique share, once that is done, we will Reconstruct the secret , now to reconstruct the secret we will require a minimum number of shares called threshold below that it will not be possible to obtain the secret with that threshold or more than that only it will be possible to Reconstruct the Original message. For that we will use Lagrange Interpolation to compute the polynomial that generates shares.

Let us Consider that we need we need to encode a secret message **S.**

1. Now for that we will divide the secret into N parts : **S1,S2,S3,S4…Sn**.
2. After dividing it a number k is chosen by the user to decrypt the parts and choose the original message .
3. It is chosen in such a way that if we know less than **K** parts then we will not be able to find the secret. The secret can be reconstructed with **(K-1)** parts or less than that.
4. If we know **K** or more parts from **S1,S2,S3** we can compute the secret key easily.

**Example:**

* For the given two points, (x1, y1) and (x2, y2) we can find a linear polynomial **ax + by = c**.
* Similarly, for the given three points, we can find a quadratic polynomial  **ax2 + bx + cy = d**.

So, the idea is to build a polynomial with the degree **(K – 1)** such that the constant term is the secret code and the remaining numbers are random and this constant term can be found by using any K points out of N points generated from this polynomial by using Lagrange’s Basis Polynomial.

