When a verifier wants to compute the user's public key from the certificate (I_u, C), following formula applies: $PK_u = C + h(I_u, C)Y$. Note that the equation used to compute σ is exactly Schnorr's signing equation. The only difference from Schnorr's signature scheme is the construction of the point C. Here, this point contains an additive component that the user provides. This is necessary to guarantee that only the user knows his secret key.

7 (t, n) Threshold Scheme for Implicit Certificates

In this section, we incorporate the distributed Schnorr signature scheme into a (t, n) threshold scheme for implicit certificates in the same way as was done in Section 6. In such a scheme, n players $P_1, ..., P_n$, called the *shareholders*, represent a CA with public key PK_0. A group of t shareholders together can reconstruct SK_0 and issue an implicit certificate. Any coalition of less than t shareholders does not have any information about SK_0.

Our scheme consists of three steps. First, the shareholders representing the CA have to generate a key pair. Everybody will know the value of PK_0, while only a coalition of at least t shareholders shall be able to recover SK_0 or issue certificates. Second, the shareholders issue a certificate to a user. Finally, the user verifies if the certificate is valid.

In Section 8, we will give a proof that the presented scheme is as secure as the Schnorr signature scheme. This means that if an adversary could forge an implicit certificate *and* know the corresponding private key, he could also forge a Schnorr signature.

7.1 Key Generation Protocol

We would like to generate a random shared secret SK_0 such that each shareholder P_i who follows the protocol holds a share s_i in this key. Moreover, a coalition of less than t players cannot get any information about SK_0.

This situation corresponds exactly to the generation of a shared secret, as described in Section 2.4. Using the notation introduced in Section 2.4, the situation is as follows:

$$(\alpha_1, ..., \alpha_n) \stackrel{(t,n)}{\longleftrightarrow} (SK_0 | PK_0, b_i G, H_0), \;\; i \in \{1, ..., t-1\}.$$

7.2 Certificate Issuing Protocol and Public Key Reconstruction

Suppose a subset $H_1 \subseteq H_0$ wants to issue an implicit certificate.

1. The user selects a random number c_u and sends $V_u = c_u G$ to the shareholders. V_u is called the public request value of the user.

2. If $|H_1| < t$, stop. Otherwise, H_1 generates a random shared secret as shown in Section 2.4. Let the public output be

$$(\beta_1, ..., \beta_n) \stackrel{(t,n)}{\longleftrightarrow} (e | V, c_i G, H_2), \;\; i \in \{1, ..., t-1\}.$$