You Zhou kecen.zhou@gmail.com

AuthInfo Leakage (Implemented)

- **Threat Description**

  In this phase, assumption made upon group server that it will never be compromised will be weakened to a degree that its passwords field holding $< \text{ID}, \text{SHA256(passwords)} >$ is at the risk of leaking out, and when it happens, dictionary attacks are almost definitely followed. Considering most users are bad passwords choosers, by online hash cracking engine like Hash Hunters, naive passwords such as "baseball" and "basketball" can be figured out immediately. Worse, as the number of users climbs up, the chances of having two users with the same passwords become concerning, meaning that by cracking one, attackers actually crack down two or more.

- **Mechanisms**

  By concatenating a random

T2 File Key Leakage

- **Threat Description**

  A per group file key list is stored in the group server, but the list is stored unencrypted. Attackers are assumed to be very efficient to steal the encrypted files from file servers, so the associated file keys are the final defend against them. Once the group server is compromised, the confidentiality of files would be no longer ensured.

- **Mechanism**

  The group server will generate a per group master key to encrypt the file key list object, and the master key will be exported to each member of the group in binary format. Every time before a download request, group members need to import the binary file to the group server first so that the restored master key can decrypt the file key object.

Threat 3: File Key Non-Renewal

- **Threat Description**

  In the current setting, files live forever with the file keys used to upload them. However, the longer the file key stays unchanged, the more likely it would be learned by attackers.

- **Mechanism**

  Since we generate a new file key every time a group member is revoked, the key list object tends to hold a huge number of keys as the time goes. For renewal process, generating a new key to substitute all keys existing is impractical because encrypted files need to be decrypted first with their associated old keys and then encrypted with the new key – it is time costly. Hence, the renewal mechanism will only target files requested most often, since the more often they are requested, the more features of them can be learned by attackers.

Threat 4: DoS Attack

- **Threat Description**

  Attackers can easily spawn huge number of threads requesting handshaking process or other operations repeatedly to paralyze the servers. Legitimate clients in that scenario will not be able to access to or receive reply from servers.

- **Mechanism**

  Servers will put a certain limit on the amount of requests can be sent from each IP address per unit time. For specific operations: 1.handshaking requests will be only allowed to do twice per login. 2.upload/download the same files will only be allowed to do twice per day. 3.group creation can only be requested three times per day. 4.deleting groups that are not existing for three times will result immediate disconnection.