Rahel Arnold **Maurizio Pasquinelli** **Tim Bachmann**

Project: Secure Team Chat

Abstract: Secure Team Chat is an application extending BACnet. It allows creating new users as well as following and unfollowing others. Moreover, creating team chats, inviting users to them and exchanging messages within such a group are available features.

Github: redez-sem-hs20/groups/04-secureTeams

Project Idea

In Secure Scuttlebutt private chats between two persons and small group chats with less than 8 participants are possible. In this project, the idea was to implement a first group chat, called channel, in BACnet. A chat has always one owner, which can invite other users to his chat group. Within a channel, the messages are encrypted.

Components

The application uses user-specific logs to store interactions. The content of those events can either be cleartext (readable for everyone) or cyphertext (readable for a specific set of recipients only). We introduced channels, an abstraction for a set of members (and owners) that share a common key for the encryption of messages within this specific group. Lastly and for end-user convenience, we store an alias file locally that maps a self-chosen name to a corresponding public identifier. So there is no further requirement of always having to insert public keys into each command. Additionally, these IDs can therefore be represented as those individually set aliases in the logs too.

Deployment / Integration

The events and the append-only log files are provided by the BACnet code. The synchronization of the log files is achieved by reading all user logs that the current user actually follows. All events that are not yet in the log file of the current user are wrapped in a clear text event with type `log/sync`. If the event is already a sync event, it is first unwrapped.
The program is called with the command `./user.py (alias) <command> [options]`, where the alias is the human readable name of a user. The implemented commands include `create`, `log`, `message`. The command `create` will create a new user including all data like private and public key. The command `log` will display the contents of the log file as a list of formatted messages. The `message` command will send a new message to a specified channel. A complete list of all commands can be found in the README.md file in the GitHub repository.

Further Work

Up to now, Secure Team Chat only allows inviting users to a group. In the future, it might be helpful to implement commands which enable the owner to pass on his role, close a channel, throw out participants or add the possibility for users to leave a group on their own.
For extremely secure messages, an option for self-destruction after some time might be useful.