CryptoKnight

Features

Server

The server manages the licenses by creating new licenses for registered users. The core features that are provided by the server application are:

- register new users
- save user data to an encrypted (DPAPI) file
- load data form encrypted file
- generate license keys per user with preset limitations
- reads max. user acceptance from the AppSettings, which are encrypted (DPAPI)
- offers client based connections via TCP/IP sockets
- provide plugin based extensions for client hosts
- offers license encrypted plugin transmition

Client

The client application connects to the server and requests plugins. The core features that are provided by the server application are:

- plugins for plain text encryption and decryption
- license based authentication system
- secured sandboxing of the loaded plugins
- rich feature based WPF UI / UX

Plugins

The plugins offer state of the art encryption and decryption algorithms such as:

- Aes :D
- DES
- RC2
- Rijndael
- TripleDES

Communication Sequence

Prerequisite

- server registers new users and generates license keys
- server discovers available plugins

Server / Client Communication

C >> S

```
LoginMessage: Contains the user information and license key
```

C << S

```
LoginResponseMessage: Server replies with an encrypted FileKey, which contains the
password for the plugin and the expiration date until the request is valid
```

C << S

```
PluginResponse: In case of an successful login, the server side discovered plugins
are transmitted to the authenticated client.
```

Encryption / Decryption

The FileKey which consisting of the plugin password and expiration date is encrypted using the license key. The PluginResponse is encrypted by an one-time generated password and contains the encrypted plugin. This authentication / encryption combination is used to prevent message spoofing.