

SOBioDecyptor (DLL interface)

Dll to decrypt the Biodata frames from StepOver Signature Device:

given elements:

- 1) Notray Private key
- 2) Encrypted AES key
- 3) Biodata Buffer

Exported functions:

+option1: Set the notary private key file

```
function LoadPrivateKeyFromFile(notaryKeyName:string):boolean; stdcall;
```

+option2: load the notary private key from buffer

```
function LoadPrivateKeyFromBuffer(notaryKeyContent:Pointer; KeyBufferSize : integer):boolean; stdcall;
```

+option3: load the decrypted aes from a binary stream (normal called skf-file)

```
function LoadDecryptedAESKey(keyFile:string) : boolean;stdcall;
```

+set the encrypted aes-key: in this case you need also the document hash

```
function SetEncryptedAESKey(DocHash:pointer; hashsize:integer;keybuffer:Pointer; keysize:Integer): boolean;stdcall;
```

```
function SetDecryptedAESKey(keybuffer:Pointer; keysize:Integer): boolean;stdcall;
```

//returns the Decrypted AEs key

```
function GetDecryptedAESKey(DataBuffer : Pointer; DataBufferLen : integer) : integer;
```

+decrypt biodata and save it to file

```
function DecryptBiodataToFile(ptrEncData:Pointer; dataLen:Integer; FileName:string):Boolean; stdcall;
```

+decrypt biodata frames

```
function DecryptBiodataFrames(ptrEncData:Pointer; nFrames:Integer):Boolean; stdcall;
```

+decrypt a single biodata frame

```
function DecryptBiodataFrame(ptrEncData : Pointer):boolean; stdcall;
```

+decrypt a biodata given as a string (done in the sopad.dll)

```
function DecryptBiodataString(strEncData : pchar):Boolean; stdcall;
```