

Java 接口（用于数据库加密）

V1.01

第一章 总体说明

1.1 SDK 包版本说明

版本	需求	说明
V1.0	一期需求 对应 2.1 ~ 2.17 章节	都是对应一期需求。 不同点是： V1.0 内置 log4j-api V1.13 依赖外部 logback-classic(1.2.11) 日志系统
V1.13		
V1.14	二期需求 新增接口：加密同时计算 MAC，验证 MAC 并解密。 对应 2.18 ~ 2.21 章节 ISM4Mac 接口新增 checkDigest 方法，实现 mac 值对比。对应 2.17 章节。	依赖外部 logback-classic(1.2.11) 日志系统

1.2 SDK 包结构说明

- doc 文档
- sample 示例代码
- jar
 - sydapi-database-1.x.xxxxxx.jar 接口 API
 - dependencies 所有依赖
 - groovy-all-2.4.15 通信协议支持
 - proto-1.0-20220905 通信协议支持
 - sydapi4j-1.0.20220906.jar 加密机通信包

第二章 接口说明

2.1. 设置日志配置

/******

功能:

设置日志相应的配置。

函数原型:

```
static void SYD_SetLogConfig (  
    String pcLogPath,  
    int    iLogLevel);
```

返回值: 无

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

pcLogPath:	输入。日志存储路径, 以 / 或 \\ 结尾表示目录
iLogLevel:	输入。日志级别设置
	0: 不记录日志
	1: 记录网络层错误日志
	2: 记录接口错误日志及 网络层错误日志
	3: 记录正确日志及错误日志

2.2. 建立与签名服务器的连接

/******

功能:

设置配置, 后续创建对象时建立到指定地址和端口号的签名服务器的 TCP/IP 连接。

◆长连接使用 SydApiBuilder 类辅助创建, 通常只用创建一个单例的对象。

函数原型:

```
- SydApiBuilder (  
    String[] pcIpList,  
    int[]    iPortList,  
    int      iConnectTimeout,  
    int      iDealTimeout  
);
```

返回值: SydApiBuilder 的对象。

异常：IllegalArgumentException 异常对象，参数配置错误。

参数说明：

pcIpList: 输入。签名服务器的 IP 地址，**（可输入多个地址，用于主备，高可用。默认将第一个 IP 地址及端口号当主加密设，其余为备用签名服务器）**

iPortList: 输入。签名服务器的端口号，**（可输入多个端口号，与 pcIpList 参数一一对应，数量相等，用于主备，高可用）。**

iConnectTimeOut: 输入。连接超时，单位毫秒。

iDealTimeOut: 输入。交易接口超时设定，单位毫秒（不超过 10ms）。
（该句柄支持数据重发，降低丢包概率；支持主机可选；支持超时时间设置，支持高可用，用于新增接口调用）

- build 方法（多线程安全）

SydApi build()

返回值：SydApi 的对象。如果采用短连接，交易结束，要调用 SydApi 的方法 SYD_Disconnect_Ex 进行释放连接。

异常：SydApiException 异常对象，retCode 属性为错误码。

参数：无

◆短连接使用 SydApi 下的 SYD_Short_Connect_Ex 方法。

函数原型：

```
SYD_Short_Connect_Ex(  
    String[] pcIpList,  
    int[] iPortList,  
    int iConnectTimeOut,  
    int iDealTimeOut  
)
```

参数说明：

pcIpList: 输入。签名服务器的 IP 地址，**（可输入多个地址，用于主备，高可用。默认将第一个 IP 地址及端口号当主加密设，其余为备用签名服务器）**

iPortList: 输入。签名服务器的端口号，**（可输入多个端口号，与 pcIpList 参数一一对应，数量相等，用于主备，高可用）。**

iConnectTimeOut: 输入。连接超时，单位毫秒。

iDealTimeOut: 输入。交易接口超时设定，单位毫秒（不超过 10ms）。
（该句柄支持数据重发，降低丢包概率；支持主机可选；支持超时时间设置，支持高可用，用于新增接口调用）

2.3. 断开与签名服务器的连接

/******

功能:

关闭与签名服务器的 TCP/IP 连接

函数原型:

```
void SYD_Disconnect_Ex();
```

返回值: 无

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

无

2.4. 短数据 hash 接口

/******

功能:

对指定数据进行 SM3 哈希处理。

函数原型:

```
String SYD_SM3_Hash_ShortData(  
    byte[] pcData  
);
```

返回值: 哈希值 (固定长度 32 字节) 转 HEX 字符串格式

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

pcData: 输入。原始数据。

2.5. 长数据 hash 接口

/******

功能:

对指定数据进行 SM3 哈希处理。

函数原型:

```
LongDataReturn<String,String> SYD_SM3_Hash_LongData(  
    byte[] pcData,  
    int    iPkgNum,  
    String pcProcData  
);
```

返回值: LongDataReturn<String,String>对象。属性如下:

procData: 过程数据 (包序号为 1 或 2 时有值) 字符串格式

data: 哈希值 (固定长度 32 字节) 转 HEX 字符串格式

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明：

pcData: 输入。 要加密的数据。
iPkgNum: 输入。 包序号： 0 仅一包 pcHash 为输出 hash 结果
1 第一包 输出过程数据 pcProcData
2 中间包 pcProcData 为输入上一包返回的过程数据输出本包返回的过程数据
3 最后一包 pcHash 为输出 hash 结果
pcProcData: 输入。过程数据 （包序号为 0 或 1 时传入 null，包序号为 2 或 3 时需要值）

2.6. 短数据加解密接口

/******

功能：

对指定数据进行 SM4 加解密处理。

函数原型：

```
byte[] SYD_SM4_ShortData(  
    int    iKeyIndex,  
    byte[] pcData,  
    int    iFlag  
);
```

返回值：报文明文或密文。

异常：SydApiException 异常对象，retCode 属性为错误码。

参数说明：

iKeyIndex: 输入。会话密钥索引值
pcData: 输入。 报文密文或明文。
iFlag: 输入。加解密标识, 0:表示 ECB 解密; 1:表示 ECB 加密

2.7. 批量数据加解密接口

/******

功能：

对指定批量数据进行 SM4 加解密处理。

函数原型：

```
List<byte[]> SYD_SM4_BatchData(  
    int[]    iKeyIndex,  
    List<byte[]> pcData,  
    int    iFlag  
);
```

返回值：List<byte[]>对象，List 的长度与输入参数 pcData 的长度相同。

参数说明：

iKeyIndex: 输入。 会话密钥索引值数组

pcData: 输入。 要加密的报文明文或密文数组。
iFlag: 输入。 加解密标识, 0:表示 ECB 解密; 1:表示 ECB 加密

2.8. 长数据加解密接口

/******

功能:

对数据进行 SM4 加解密处理。

函数原型:

```
byte[] SYD_SM4_LongData(  
    int    iKeyIndex,  
    byte[] pcData,  
    int    iPkgNum,  
    int    iFlag  
);
```

返回值: 输出数据;

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex: 输入。会话密钥索引值
pcData: 输入。要加密的数据。(数据的长度第一包以及中间包时需为 16 的整数倍)
iPkgNum: 输入。包序号
 0: 仅一包
 1: 第一包
 2: 中间包
 3: 结束包
iFlag: 输入。加解密标识, 0:表示 ECB 解密; 1:表示 ECB 加密

2.9. 批量数据 MAC 计算接口

/******

功能:

对批量数据进行 SM4 MAC 计算。

函数原型:

```
String[] SYD_SM4Mac_BatchData(  
    int[]    iKeyIndex,  
    List<byte[]> pcData  
);
```

返回值: String[] 数组, 长度等于 pcData List 的长度。

String[] 数组中的每个值为 mac 值 (固定长度 16 字节) 的 HEX 字符串。

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值数组
pcData:	输入。	要做 MAC 的数据组

2.10. 批量数据 MAC 校验接口

/******

功能:

对批量数据进行 SM4 MAC 校验。有一条数据失败则抛出异常。

函数原型:

```
boolean[] SYD_SM4Mac_BatchData(  
    int[]    iKeyIndex,  
    List<byte[]> pcData,  
    String[] pcMac  
);
```

返回值: boolean[] 数组长度等于 pcData List 的长度。

boolean[] 数组中的每个值为对应索引 pcData 和 pcMac 的校验结果。

true 表示校验成功, false 表示校验失败。

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值数组
pcData:	输入。	要做 MAC 的数据组
pcMac:	输入。	输入的 mac 值数组组, 每个 ma 从 (固定长度 16 字节) 的 HEX 字符串。

2.11. 长数据 MAC 计算接口

/******

功能:

对数据进行 SM4 MAC 计算。

函数原型:

```
LongDataReturn<String, String> SYD_SM4Mac_LongData(  
    int    iKeyIndex,  
    byte[] pcData,  
    int    iPkgNum,  
    String pcProcData  
);
```

返回值: LongDataReturn<String, String> 的对象

属性 procData 过程数据 (包序号为 1 或 2 时有值) 字符串

属性 data 输出的 mac 值 (固定长度 16 字节) HEX 字符串

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值
------------	-----	------------------

pcData:	输入。	要做 MAC 的数据
iPkgNum:	输入。 包序号:	0 仅一包 pcHash 为输出 hash 结果 1 第一包 输出过程数据 pcProcData 2 中间包 pcProcData 为输入上一包返回的过程数据输出本包返回的过程数据 3 最后一包 pcHash 为输出 hash 结果

2.12. 长数据 MAC 校验接口

/*****

功能:

对数据进行 SM4 MAC 校验处理。

函数原型:

```
LongDataReturn<String, Boolean>SYD_SM4Mac_LongData(
    int    iKeyIndex,
    byte[] pcData,
    int    iPkgNum,
    String pcProcData,
    String pcMac
);
```

返回值: LongDataReturn<String, Boolean> 的对象

属性 procData 过程数据 (包序号为 1 或 2 时有值) 字符串

属性 data 输出的 mac 校验结果。Boolean 类型, true 表示校验成功, false 表示校验失败。

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值
pcData:	输入。	要做 MAC 的数据
iPkgNum:	输入。 包序号:	0 仅一包 pcHash 为输出 hash 结果 1 第一包 输出过程数据 pcProcData 2 中间包 pcProcData 为输入上一包返回的过程数据输出本包返回的过程数据 3 最后一包 pcHash 为输出 hash 结果
pcProcData:	输入。	过程数据 (包序号为 1 或 2 时有值)
pcMac:	输入。	输出的 mac 值 (固定长度 16 字节) HEX 字符串

2.13. 短数据 MAC 计算接口

/*****

功能:

对数据进行 SM4 MAC 计算。

函数原型:

```
String SYD_SM4Mac_ShortData(  
    int    iKeyIndex,  
    byte[] pcData  
);
```

返回值: mac 值(固定长度 16 字节)转 HEX 字符串格式

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值
pcData:	输入。	计算 mac 所需要的数据。

2.14. 短数据 MAC 验证接口

/*****

功能:

对数据进行 SM4 MAC 校验。

函数原型:

```
boolean SYD_SM4Mac_ShortData(  
    int    iKeyIndex,  
    byte[] pcData,  
    String pcMac,  
);
```

返回值: 为 true 表示校验成功, false 表示校验失败。

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iKeyIndex:	输入。	计算 mac 所需要的密钥索引值
pcData:	输入。	计算 mac 所需要的数据。
pcMac:	输入。	mac 值(固定长度 16 字节)HEX 字符串

2.15. initSm3/update/digest 长数据操作

功能:

可以对长数据接口输入 0 到 4064 任意长度的 pcData。

```
ISM3 initSM3();
```

返回值: ISM3 对象 用于调用 update/digest 方法

```
Void update(byte[] pcData)
```

入参: pcData 要加密的数据

```
String digest();
```

返回值: 返回的结果

2.16. initSm4/update/digest 长数据操作

功能:

可以对长数据接口输入 0 到 4064 任意长度的 pcData (解密输入数据长度范围是 16-4064)。

```
ISM4 initSM4(int iKeyindex,  
              int ende);
```

入参: int iKeyindex 密钥索引值
 int ende 加解密标识: 0 解密, 1 加密

返回值: ISM4 对象 用于调用 update/finish 方法

```
Byte[] update(byte[] pcData)
```

入参: pcData 要加密/解密的数据

返回: byte[] 中间数据

```
Byte[] finish();
```

返回值: 返回的加解密的结果

2.17. initSm4Mac/update/digest 长数据操作

可以对长数据接口输入 0 到 4064 任意长度的 pcData

```
ISM4Mac initSm4Mac(int iKeyindex);
```

入参: int iKeyindex 密钥索引值
 可用 setPcMac 方法对 Mac 进行赋值

返回值: ISM4Mac 对象 用于调用 update/digest/checkDigest 方法

```
pcMac 有值: Byte[] update(byte[] pcData)
```

入参: pcData 要做 Mac 的数据组

pcMac 无值: Byte[] update(byte[] pcData)

入参: pcData 要做 Mac 的数据组

digest();

返回值: 返回结果数据

2.18. 短数据加密并计算 MAC

/*****

功能:

对数据进行 SM4 加密, 后对密文 MAC 计算。

函数原型:

```
DataAndMac SYD_SM4_EncryptAndMac_ShortData(  
    int iEncKeyIndex,  
    int iMacKeyIndex,  
    byte[] pcData  
);
```

返回值: DataAndMac

- data 密文

- mac 值(固定长度 16 字节)转 HEX 字符串格式

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iEncKeyIndex: 输入。 加密计算所需要的密钥索引值

iMacKeyIndex: 输入。 计算 mac 所需要的密钥索引值

pcData: 输入。 计算 mac 所需要的数据。

2.19. 短数据校验 MAC 并解密

/*****

功能:

短数据先校验 MAC, 后解密

函数原型:

```
byte[] SYD_SM4_CheckMacAndDecrypt_ShortData(  
    int iEncKeyIndex,  
    int iMacKeyIndex,  
    byte[] pcData,  
    String mac  
);
```

返回值: DataAndMacCheck

- data 密文

- macCheck mac 校验结果

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iEncKeyIndex: 输入。 加密计算所需要的密钥索引值

iMacKeyIndex: 输入。 计算 mac 所需要的密钥索引值

pcData: 输入。 数据。

2.20. 批量数据加密并计算 MAC

/*****

功能:

对批量数据进行 SM4 加密, 后对密文 MAC 计算。

函数原型:

```
List<DataAndMac> SYD_SM4_EncryptAndMac_BatchData(  
    int[] iEncKeyIndex,  
    int[] iMacKeyIndex,  
    List<byte[]> pcData  
);
```

返回值: List<DataAndMac>

- data 密文
- mac 值(固定长度 16 字节)转 HEX 字符串格式

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

iEncKeyIndex: 输入。 加密计算所需要的密钥索引值

iMacKeyIndex: 输入。 计算 mac 所需要的密钥索引值

pcData: 输入。 数据。

2.21. 批量数据校验 MAC 并解密

/*****

功能:

批量数据先校验 MAC , 后解密

函数原型:

```
List<DataAndMacCheck> SYD_SM4_CheckMacAndDecrypt_BatchData(  
    int[] iEncKeyIndex,  
    int[] iMacKeyIndex,  
    List<DataAndMac> pcData  
);
```

返回值: DataAndMacCheck

- data 密文
- macCheck mac 校验结果

异常: SydApiException 异常对象, retCode 属性为错误码。

参数说明:

- iEncKeyIndex: 输入。 加密计算所需要的密钥索引值
- iMacKeyIndex: 输入。 计算 mac 所需要的密钥索引值
- pcData: 输入。 密文数据和 MAC 对比值。

附录 错误码

错误码	说明
-1	网络连接失败
-2	网络发送失败
-3	网络接收失败
15	数据格式错误
33	密钥不存在
80	输入数据长度不正确
90	MAC 验证不通过