

ZKBioOnline SDK For Windows

接口开发手册



熵基科技股份有限公司

<http://www.zkteco.com>

版权声明

熵基科技股份有限公司版权所有©,保留一切权利。

非经本公司书面许可,任何单位和个人不得擅自摘抄、复制本书的部分或全部,并不得以任何形式传播。

1. 目录

版权声明

1. 目录

2. 前言

3. 技术规格

- 3.1 平台支持
- 3.2 浏览器支持
- 3.3 硬件支持
- 3.4 技术规格
- 3.5 快速集成
 - 3.5.1 安装
 - 3.5.2 流程图

5. 接口说明

- 5.1 websocket协议
 - 5.1.1 通信数据格式定义
 - 5.1.2 获取服务信息 (info 请求)
 - 5.1.3 打开指纹设备 (open 请求)
 - 5.1.4 采集登记指纹 (register 请求)
 - 5.1.5 取消登记采集 (cancelregister 请求)
 - 5.1.6 关闭指纹设备 (close 请求)
 - 5.1.7 1:1指纹比对 (verify 请求)
 - 5.1.8 oncapture事件
 - 5.1.9 onenroll事件
 - 5.1.10 错误码
- 5.2 http/https 协议
 - 5.2.1 返回值数据格式定义
 - 5.2.2 获取系统信息
 - 5.2.3 开始采集
 - 5.2.4 取消采集
 - 5.2.5 获取图像
 - 5.2.6 获取模板
 - 5.2.7 1:1比对指纹
 - 5.2.8 错误码
- 5.3 后台比对库(matchdll.dll)
 - 5.3.1 Java Native API
 - 5.3.1.1 NativeToProcess
 - 5.3.1.2 NativeToSetThreshold
 - 5.3.2 C API
 - 5.3.2.1 SetThreshold
 - 5.3.2.2 Verify
 - 5.3.2.3 process

2. 前言

欢迎使用熵基科技ZKBioOnline SDK, 在使用前请仔细阅读本文档，以便于您能更快地掌握并使用。

3. 技术规格

3.1 平台支持

Windows XP SP3及以上系统

3.2 浏览器支持

高版本谷歌、火狐，IE10及以上IE浏览器等(注意：edge浏览器不支持websocket协议接口)

3.3 硬件支持

- ZK4500
- ZK8500
- FS200
- FS300
- ZK6000A
- ZK7000A
- Live-10R
- Live-20R

3.4 技术规格

适用于B/S系统

协议	端口号
http	22001
https	22002
websocket	22003

3.5 快速集成

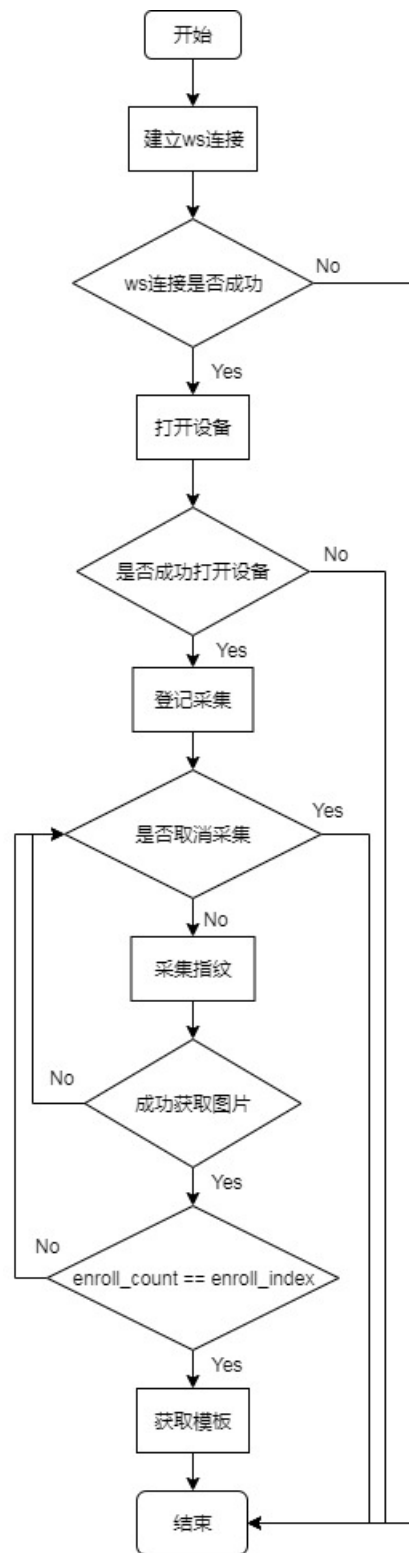
3.5.1 安装

- (1) 在客户端双击安装程序“setup.exe”安装sdk。
- (2) 按照安装向导完成安装，即可通过http/https/websocket协议访问。

注意：不同协议不支持混合使用

3.5.2 流程图

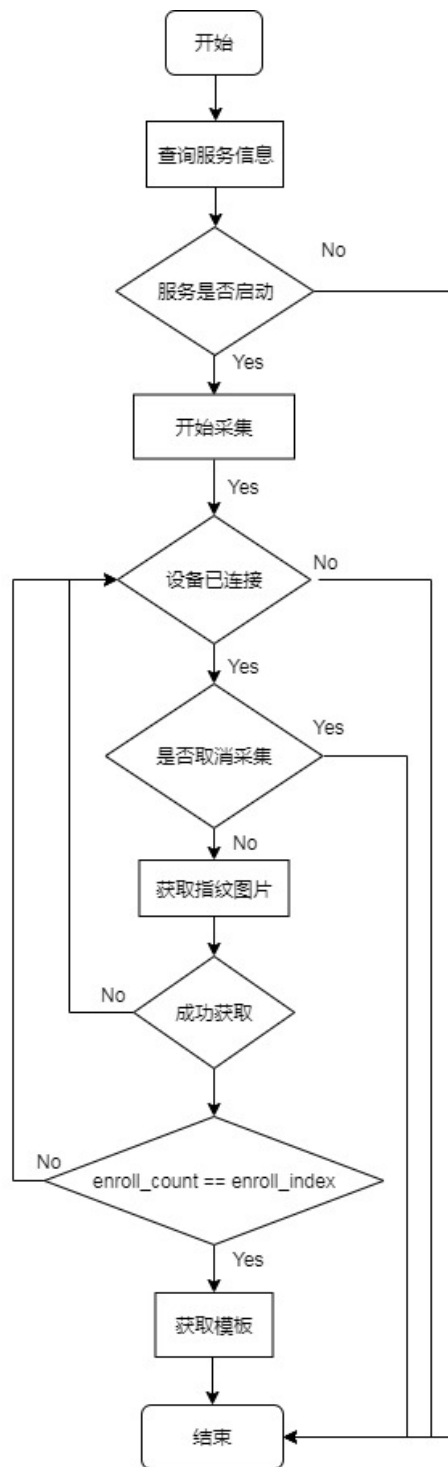
websocket协议的调用流程：



websocket协议 获取登记指纹的流程步骤:

- 建立websocket连接，通过websocket连接情况可以判断服务的运行情况。
- 调用 open 请求打开设备，根据 open 的应答消息判断是否成功打开设备，成功继续流程步骤，失败则结束。
- 调用 register 请求，准备开始采集指纹。需要提前结束登记采集流程时，可调用 cancelregister 请求结束登记指纹采集流程。
- 手指按压指纹仪采集指纹，采集成功触发 onenroll事件。
- 接收onenroll 事件返回的3条image消息（3次采集完成的指纹图片数据）和一条template消息（完成采集的指纹模板数据）。采集结束。

http/https协议调用流程：



http/https 协议获取登记指纹的流程步骤:

- 调用 /zkbioonline/info 方法查询服务信息，通过响应消息判断服务运行状态。
- 调用 /zkbioonline/fingerprint/beginCapture 方法开始采集登记指纹，通过响应消息判断服务是否可以正常采集，否则可以根据返回的错误码了解设备连接状态等情况。
- 在采集登记指纹的过程中，可以通过调用 /zkbioonline/fingerprint/cancelCapture 方法提前取消登记采集。
- 每次采集完成后，通过 /zkbioonline/fingerprint/getImage 方法获取采集的指纹图片，并根据登记索引enroll_index 是否等于3，来判断是否已经完成登记采集。
- 完成登记采集后，可以通过 /zkbioonline/fingerprint/getTemplate 方法获取采集的登记指纹特征，登记指纹采集完成。

5. 接口说明

5.1 websocket协议

5.1.1 通信数据格式定义

通信协议采用 websocket 时，后台服务在本地 22003端口监听。

建立连接示例代码如下：

```
try {
    var host = "ws://127.0.0.1:22003";
    var websocket = new WebSocket(host);
} catch (exception) {
    console.log("error.");
}
```

通信数据格式采用 JSON ，各字段区分大小写，定义说明如下：

请求的数据格式：

功能	描述
module	模块名称
msgid	自定义消息id
function	方法
parameter	参数，具体格式参考具体接口说明
data	返回消息，具体格式参考具体接口说明

应答的数据格式：

功能	描述
module	模块名称
msgid	消息id，应答的消息id和请求的消息id一致
function	方法
ret	0成功，其它失败，具体请参考错误码
data	存放返回的数据，具体格式参考具体接口说明

5.1.2 获取服务信息（info 请求）

请求消息的字段：

功能	获取服务信息
module	common
msgid	自定义消息id
function	info
parameter	无参数，非必要字段

请求消息的示例如下：

```
{
  "module"      : "common",
  "msgid"       : "123456789",
  "function"    : "info",
  "parameter"   : ""
}
```

应答消息的字段：

功能	获取服务信息
module	common
msgid	自定义消息id
function	info
ret	0成功，其它失败，具体请参考错误码
data.now	当前请求时间
data.server_version	sdk版本
data.start	sdk服务启动的时间
data.biometric.engversion	算法版本
data.biometric.enroll_count	登记采集需要采集次数
data.biometric.type	生物识别类型

应答消息的示例如下：

```
{
  "data": {
    "biometric": {
      "engversion": "10.0",
      "enroll_count": 3,
      "type": "fingerprint"
    },
    "now": "2019-11-12 10:53:35",
    "server_version": "5.2.14",
    "start": "2019-11-12 10:51:00"
  },
  "function": "info",
}
```



```
"module": "common",
"msgid": "123456789",
"ret": 0
}
```

5.1.3 打开指纹设备（open 请求）

打开指纹，默认进入指纹采集状态，每次采集都会触发oncapture事件，oncapture事件具体参考5.1.8。

请求消息的字段：

功能	打开指纹设备
module	fingerprint
msgid	自定义消息id
function	open
parameter	无参数，非必要字段

请求消息的示例如下：

```
{
  "module"    : "fingerprint",
  "msgid"     : "123456789",
  "function"  : "open",
  "parameter" : ""
}
```

应答消息的字段：

功能	打开指纹设备
module	fingerprint
msgid	自定义消息id
function	open
ret	0成功，其它失败，具体请参考错误码
data.SN	设备序列号

应答消息的示例如下：

```
{
  "data": {
    "SN": "3832175200303"
  },
  "function": "open",
  "module"   : "fingerprint",
  "msgid"    : "123456789",
  "ret"      : 0
}
```

5.1.4 采集登记指纹（ register 请求）

采集登记指纹需要采集三次指纹，每次采集完成触发onenroll事件，onenroll事件具体参考5.1.9。采集登记指纹完成后会进入指纹采集状态。

请求消息的字段：

功能	采集登记指纹
module	fingerprint
msgid	自定义消息id
function	register
parameter	无参数，非必要字段

请求消息的示例如下：

```
{
  "module"      : "fingerprint",
  "msgid"       : "123456789",
  "function"    : "register",
  "parameter"   : ""
}
```

应答消息的字段：

功能	采集登记指纹
module	fingerprint
msgid	自定义消息id
function	register
ret	0成功，其它失败，具体请参考错误码
data	没有返回数据时，返回空

应答消息的示例如下：

```
{
  "data"        : "",
  "function"    : "register",
  "module"      : "fingerprint",
  "msgid"       : "123456789",
  "ret"         : 0
}
```

5.1.5 取消登记采集（ cancelregister 请求）

该方法用于取消采集登记指纹，即将onenroll事件状态重置为oncapture事件状态。

请求消息的字段：

功能	取消登记采集
module	fingerprint
msgid	自定义消息id
function	cancelregister
parameter	无参数，非必要字段

请求消息的示例如下：

```
{
  "module" : "fingerprint",
  "msgid" : "123456789",
  "function" : "cancelregister",
  "parameter": ""
}
```

应答消息的字段：

功能	取消登记采集
module	fingerprint
msgid	自定义消息id
function	cancelregister
ret	0成功，其它失败，具体请参考错误码
data	没有返回数据时，返回空

应答消息的示例如下：

```
{
  "data" : "",
  "function": "cancelregister",
  "module" : "fingerprint",
  "msgid" : "123456789",
  "ret" : 0
}
```

5.1.6 关闭指纹设备 (close 请求)

请求消息的字段：

功能	关闭指纹设备
module	fingerprint
msgid	自定义消息id
function	close
parameter	无参数，非必要字段

请求消息的示例如下：

```
{
  "module"    : "fingerprint",
  "msgid"     : "123456789",
  "function"  : "close",
  "parameter" : ""
}
```

应答消息的字段：

功能	关闭指纹设备
module	fingerprint
msgid	自定义消息id
function	close
ret	0成功，其它失败，具体请参考错误码
data	没有返回数据时，返回空

应答消息的示例如下：

```
{
  "data"      : "",
  "function": "close",
  "module"    : "fingerprint",
  "msgid"     : "123456789",
  "ret"       : 0
}
```

5.1.7 1:1指纹比对 (verify 请求)

该方法是对传入的两个指纹模板做1:1指纹比对，返回一个比对分数值(0~100), 用户可根据自己需求设置阈值，判断是否比对成功(一般默认大于35则比对成功)。

请求消息的字段：

功能	1:1指纹比对
module	fingerprint
msgid	自定义消息id
function	verify
parameter.template1	参与比对的模板1
parameter.template2	参与比对的模板2

请求消息的示例如下：

```
{
  "module"    : "fingerprint",
  "msgid"     : "123456789",
  "function"  : "verify",
  "parameter": {
    "template1": "****",
    "template2": "****"
  }
}
```

应答消息的字段：

功能	1:1指纹比对
module	fingerprint
msgid	自定义消息id
function	verify
ret	0成功，其它失败，具体请参考错误码
data.grade	比对分数值（范围：0~100，一般默认大于35则比对成功）

应答消息的示例如下：

```
{
  "data": {
    "grade": 100
  },
  "function": "verify",
  "module"   : "fingerprint",
  "msgid"    : "123456789",
  "ret"      : 0
}
```

5.1.8 oncapture事件

每次采集指纹，触发oncapture事件触发后，会返回2条数据，即当次采集的指纹图片数据和模板数据。

oncapture事件触发返回的 image 应答消息字段：

功能	oncapture事件
module	fingerprint
datatype	image
function	oncapture
ret	0成功，其它失败，具体请参考错误码
data.biotype	生物识别类型，指纹识别类型为0
data.enroll_index	采集指纹图片的的序号（弃用，仅onenroll事件用于显示登记采集显示登记索引）
data.width	图片宽
data.height	图片高
data.quality	图片质量
data.jpg_base64	jpg格式指纹像数据（base64 String）
data.raw_base64	原始指纹图像数据（base64 String）

示例如下：

```
{
  "data": {
    "biotype": 0,
    "enroll_index": 1,
    "height": 375,
    "jpg_base64": "****",
    "quality": 71,
    "raw_base64": "****",
    "width": 288
  },
  "datatype" : "image",
  "function" : "oncapture",
  "module"   : "fingerprint",
  "ret"      : 0
}
```

oncapture事件触发返回的 template 应答消息字段：

功能	oncapture事件
module	fingerprint
datatype	template
function	oncapture
ret	0成功，其它失败，具体请参考错误码
data.biotype	生物识别类型，指纹识别类型为0
data.template	指纹模板数据（base64 string）
data.templateLen	模板数据长度

示例如下：

```
{
  "data": {
    "biotype": 0,
    "template": "TbBTUzIXAAAE8"
    "templateLen": 1267,
  },
  "datatype" : "template",
  "function" : "oncapture",
  "module"   : "fingerprint",
  "ret"      : 0
}
```

5.1.9 onenroll事件

onenroll事件触发后，前面2次返回图片数据，第三次返回图片数据和模板数据。（注：并在触发三次后，重置为指纹采集状态）

onenroll事件触发返回的 image 应答消息字段：

功能	onenroll事件
module	fingerprint
datatype	image
function	onenroll
ret	0成功，其它失败，具体请参考错误码
data.biotype	生物识别类型，指纹识别类型为0
data.enroll_index	登记索引（三次采集的图片数据索引对应分别为1,2,3）
data.width	图片宽
data.height	图片高
data.quality	图片质量
data.jpg_base64	jpg格式指纹图像数据（base64 String）
data.raw_base64	原始指纹图像数据（base64 String）

示例如下：

```
{
  "data": {
    "biotype": 0,
    "enroll_index": 1,
    "height": 375,
    "jpg_base64": "/9j/4AAQSkZJ",
    "quality": 71,
    "raw_base64": "8/T29PL08/Px9",
    "width": 288
  },
  "datatype": "image",
  "function": "onenroll",
  "module": "fingerprint",
  "ret": 0
}
```

onenroll事件触发返回的 template 应答消息字段：

功能	onenroll事件
module	fingerprint
datatype	template
function	onenroll
ret	0成功，其它失败，具体请参考错误码
data.biotype	生物识别类型，指纹识别类型为0
data.template	指纹特征模板（base64 string）
data.templateLen	模板数据长度

返回指纹模板数据格式如下：

```
{
  "data": {
    "biotype": 0,
    "template": "TbBTUzIxAAAE8"
    "templateLen": 1267
  },
  "datatype": "template",
  "function": "onenroll",
  "module": "fingerprint",
  "ret": 0
}
```

5.1.10 错误码

错误码如下表所示

错误码	说明
0	成功
-10001	失败
-10002	服务未初始化
-10003	无效参数或者参数格式错误
-10004	模式错误
-10005	接口不支持
-10006	初始化指纹库失败
-10007	打开指纹采集设备失败
-10008	非法指纹模板
-10009	假手指
-10010	继续按指纹采集
-10011	指纹或者指静脉采集失败
-10012	导出模板失败
-10013	正在使用websocket连接
-10014	未打开设备或者webs连接标识不一致
-10015	异常跳出
-10016	登记时与上一次指纹不同
-10017	该指纹设备已绑定webs连接
-10018	该指纹设备状态异常

5.2 http/https 协议

5.2.1 返回值数据格式定义

返回值数据格式：json

```
{
  "ret":0,
  "data": "",
  "error": ""
}
```

功能	描述
ret	0成功，其它失败，具体请参考错误码
error	错误描述
data	保存返回的数据

5.2.2 获取系统信息

请求方法：

```
HTTP GET

"http://127.0.0.1:22001/zkbioonline/info"
或
"https://127.0.0.1:22002/zkbioonline/info"
```

返回的消息字段：

参数名	参数类型	描述
ret	int	0成功，其它失败，具体请参考错误码
error	string	错误描述
data.now	string	当前请求时间
data.server_version	string	sdk版本
data.start	string	sdk服务启动的时间
data.biometric.engversion	string	算法版本
data.biometric.enroll_count	int	登记采集需要采集次数
data.biometric.type	string	生物识别类型

返回消息示例如下：

```
{
```

```
"ret":0,
"error":"","
"data":
{
    "server_version":"5.0.1",
    "start":"2015-10-24 18:00",
    "now":"2015-10-24 18:00",
    "biometric": {
        "type":"fingerprint",
        "engversion":"10.0",
        "enroll_count":3
    }
}
```

5.2.3 开始采集

请求方法：

HTTP GET

"http://127.0.0.1:22001/zkbioonline/fingerprint/beginCapture?type=1"

或

"https://127.0.0.1:22002/zkbioonline/fingerprint/beginCapture?type=1"

参数：

参数名	参数类型	描述
type	int	1表示采集登记指纹(采集同一手指3次指纹),不是1则表示采集比对指纹

成功时返回：

```
{
    "ret":0,
    "data":"","
    "error":""
}
```

5.2.4 取消采集

请求方法：

HTTP GET

"http://127.0.0.1:22001/zkbioonline/fingerprint/cancelCapture"

或

"https://127.0.0.1:22002/zkbioonline/fingerprint/cancelCapture"

成功时返回：

```
{
  "ret":0,
  "data": "",
  "error":""
}
```

5.2.5 获取图像

请求方法：

HTTP GET

"http://127.0.0.1:22001/zkbioonline/fingerprint/getImage"

或

"https://127.0.0.1:22002/zkbioonline/fingerprint/getImage"

参数名	参数类型	描述
ret	int	0成功，其它失败，具体请参考错误码
error	string	错误描述
data.biotype	int	生物识别类型，0表示指纹
data.enroll_index	int	登记索引(用于登记采集时的登记采集次数索引，当enroll_index等于enroll_count时表示登记采集完成,可以调用gettemplate获取模板)
data.width	int	图片宽度
data.height	int	图片高度
data.quality	int	图片质量
data.jpg_base64	string	jpg格式指纹图像数据 (base64 String)
data.raw_base64	string	原始指纹图像数据 (base64 String)

返回的消息示例如下：

```
{
  "ret":0,
  "error": "",
  "data":
  {
    "biotype":0,
    "enroll_index":0,
    "quality":100,
    "width":280,
    "height":360,
    "jpg_base64": "****",
    "raw_base64": "****",
  }
}
```

5.2.6 获取模板

请求方法：

HTTP GET

"http://127.0.0.1:22001/zkbioonline/fingerprint/getTemplate"

或

"https://127.0.0.1:22002/zkbioonline/fingerprint/getTemplate"

参数名	参数类型	描述
ret	int	0成功，其它失败，具体请参考错误码
error	string	错误描述
data.biotype	int	生物识别类型，0表示指纹
data.template	string	指纹模板数据（base64 string）
data.length	int	模板数据长度

返回的消息示例如下：

```
{
  "ret":0,
  "error": "",
  "data":
  {
    "biotype":0,
    "length":1478,
    "template":""
  }
}
```

5.2.7 1:1比对指纹

请求方法：

HTTP POST

"http://127.0.0.1:22001/zkbioonline/fingerprint/verify"

或

"https://127.0.0.1:22002/zkbioonline/fingerprint/verify"

POST 的body数据格式为json，数据的字段说明：

参数名	参数类型	描述
reg	String	登记模板 (base64 String)
ver	String	比对模板 (base64 String)

POST数据格式为json字符串，示例如下：

```
{
  "reg": "TVFTUjIyAAAEehAFBQUHcc7...",
  "ver": "TVFTUjIyAAAEehAFBQUHcc7..."
}
```

返回的消息

参数名	参数类型	描述
ret	int	0成功，其它失败，具体请参考错误码
error	String	错误描述
score	int	比对分数， > 35表示匹配

返回的消息示例如下：

```
{
  "ret": 0,
  "error": "",
  "score": 100
}
```

5.2.8 错误码

错误值	描述
0	操作成功
4	正在操作,请稍后再试
-1	操作失败
-2	不支持操作,请确认是否URL错误
-3	参数错误(暂未使用)
-5	无效数据格式
-6	无效模板
-7	比对异常
-2000	调用错误,目前仅支持HTTP Get
-2001	连接采集器失败
-2002	加载算法库失败
-2003	提取模板失败
-2004	采集图像失败
-2005	正在采集, 请按手指或者取消采集
-2006	请按手指
-2007	分配内存失败(暂未使用)
-2008	取消失败(暂未使用)
-2009	疑似假指纹或者手指按太快

5.3 后台比对库(matchdll.dll)

5.3.1 Java Native API

类名：zk.jni.JavaToBiokey

备注：类名必须为 zk.jni.JavaToBiokey，不能改成其他，否则会出现查找不到类库的情况。

5.3.1.1 NativeToProcess

功能：比对 9.0、10.0 算法指纹模板(Base64)

方法接口：

```
public native boolean NativeToProcess(String ARegTemplate, String AVerTemplate)
```

参数：

参数名	参数类型	描述
ARegTemplate	String	登记模板(Base64 string)
AVerTemplate	String	比对模板(Base64 string)

返回值:

比对成功返回 True , 失败返回 False。

5.3.1.2 NativeToSetThreshold

功能 : 设置比对指纹模板的阈值

方法接口 :

```
public native void NativeToSetThreshold(int AThreshold, int AOneToOneThreshold)
```

参数 :

参数名	参数类型	描述
AThreshold	int	阈值
AOneToOneThreshold	int	设定Biokey低速指纹1 : 1比对的识别阈值分数(1-100), 默认为10, 值越大, 误判率越低同时拒绝率变大。

5.3.2 C API

5.3.2.1 SetThreshold

功能 : 设置比对指纹模板的阈值

方法接口 :

```
BOOL __stdcall SetThreshold(int nThreshold, int nOneToOneThreshold)
```

参数 :

参数名	参数类型	描述
AThreshold	int	阈值
AOneToOneThreshold	int	设定Biokey低速指纹1：1比对的识别阈值分数(1-100)，默认为10，值越大，误判率越低同时拒绝率变大。

返回值：
成功返回TRUE;失败返回FALSE

5.3.2.2 Verify

功能：比对 9.0、10.0 算法指纹模板(Base64)

方法接口：

```
int __stdcall verify(const char *szRegTemplate, const char *szVerTemplate);
```

参数：

参数名	参数类型	描述
szRegTemplate	char*	登记模板(Base64 string)
szVerTemplate	char*	比对模板(Base64 string)

返回值
比对分数

5.3.2.3 process

功能：比对 9.0、10.0 算法指纹模板(Base64)

方法接口：

```
BOOL __stdcall process(const char *szRegTemplate, const char *szVerTemplate);
```

参数：

参数名	参数类型	描述
szRegTemplate	char*	登记模板(Base64 string)
szVerTemplate	char*	比对模板(Base64 string)

返回值：
成功返回TRUE；失败返回FALSE