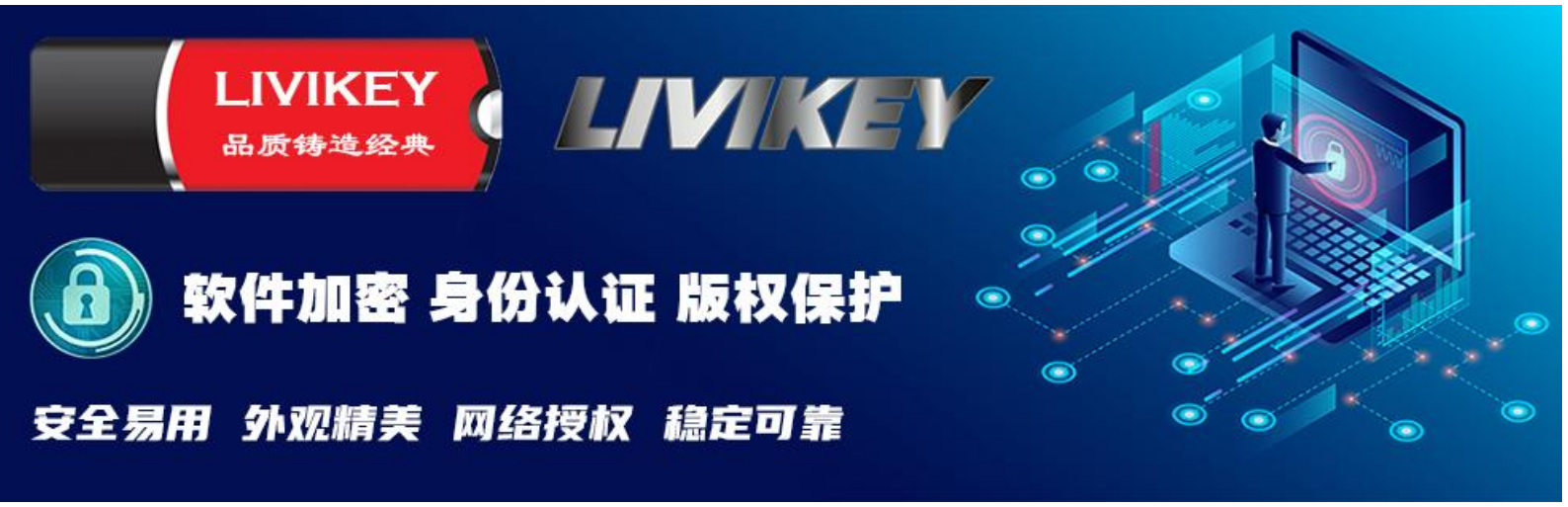
ViKey 加密锁 API 开发手册



上海恺迹信息科技有限公司

www.ivikey.com

软件开发协议

ViKey 加密锁的所有产品，包括但不限于：开发工具包，磁盘，光盘，硬件设备和文档，以及未来的所有定单都受本协议的制约。

1. 许可使用

您可以将本软件合并、连接到您的计算机程序中，但其目的只是保护该程序。您可以以存档为目的复制合理数量的拷贝。

2. 禁止使用

除在条款 1 中特别允许的之外，不得复制、反向工程、反汇编、反编译、修改、增加、改进软件、硬件和产品的其它部分。禁止对软件和产品的任何部分进行反向工程，或企图推导软件的源代码。禁止使用产品中的磁性或光学介质来传递、存储非本产品的原始程序或由我方提供的产品升级的任何数据。禁止将软件放在服务器上传播。

3. 有限担保

ViKey 加密锁保证在自产品交给您之日起的 12 个月内，在正常的使用情况下，硬件和软件存储介质没有重大的工艺和材料上的缺陷。

4. 修理限度

当根据本协议提出索赔时，我方唯一的责任就是免费进行替换或维修。我方对更换后的任何产品部件都享有所有权。

保修索赔单必须在担保期内写好，在发生故障 14 天内连同令人信服的证据交给我方。当将产品返还给我方或我方的授权代理商时，须预付运费和保险。

除了在本协议中保证的担保之外，我方不再提供特别的或隐含的担保，也不再对本协议中所描述的产品负责，包括它们的质量，性能和对某一特定目的的适应性。

5. 责任限度

不管因为什么原因，不管是因合同中的规定还是由于刑事的原因，包括疏忽的原因，而使您及任何一方受到了损失，由我方产品所造成的损失或该产品是起诉的原因或与起诉有间接关系，我方对您及任何一方所承担的全部责任不超出您购买该产品所支付的货款。在任何情况下，我方对于由于您不履行责任所导致的损失，或对于数据、利润、储蓄或其它的后续的和偶然的损失，即使我方被建议有这种损失的可能性，或您根据第 3 方的索赔而提出的任何索赔均不负责任。

6. 协议终止

当您不能遵守本协议所规定的条款时，将终止您的许可和本协议。但条款 2, 3 ， 4 ， 5 将继续有效。

产品列表

ViKey 系列加密锁	
ViKeyAPP	实用型加密锁
ViKeySTD	标准型加密锁
ViKeyNET	网络型加密锁
ViKeyPRO	专业型加密锁
ViKeyWEB	身份认证型加密锁
ViKeyTime	时间型加密锁
更多产品信息，请登录 www.ivikey.com 网站或直接联系我们 TEL:18917081416	

1、ViKey 加密锁概述

ViKey 加密锁是一款可以支持软件保护应用和身份认证应用的大容量增强型加密锁，最多提供 4096 字节超大容量存储空间，免驱动的 USB 设备。由于该加密锁使用双用户对加密锁权限进行管理，即是普通用户和管理员，管理员拥有加密锁的全部权限(一般这个管理员密码由软件开发商或软件销售商掌握)，普通用户只能执行匿名权限操作、对指定普通用户权限的内容进行读写的操作；对于受保护的软件，通过 ViKey 加密锁使软件更安全，可以保护该软件不被非法复制和非授权访问或使用。当使用 ViKey 加密锁加密保护您的软件后，启动所加密保护的程序时，此时若 ViKey 加密锁不存在或对某个应用模块的访问已超过预先设定的次数，程序会发出错误信息，从而终止，这就达到了加密保护软件的目的。

ViKey 加密锁还可以应用在各种安全系统身份认证领域，包括网站系统、OA 办公系统、信息查询系统等。通过 ViKey 加密锁的使用替换传统的用户名和密码，保证了系统的登录安全。ViKey 加密锁提供多种 API 接口调用方式。

- **LIB 静态库方式** 该方式好处是开发后编译的文件无须附带其它文件，也无须向系统注册相关文件。操作方法是在开发软件时把 ViKey.LIB 及 ViKey.H 文件包含到工程中，在程序中调用各个函数，最后编译工程就可以了。
- **DLL 动态库方式** 该方式是大多数编程序语言所支持的一种开发方式，软件开发者只需在程序中进行声明或引用并调用相应的函数即可，程序在运行时必须保证该 ViKey.DLL 文件在系统目录下或与 EXE 文件在同一目录下。
- **COM 组件方式** 该方式也是绝大多数编程语言所支持的开发方式 COM 组件是提供了所有的应用程序 API ，在运行时必须保证该 ViKey.DLL 组件存在并已注册。
- **Active 控件方式** 该方式也是绝大多数编程语言所支持的开发方式控件是提供了所有的应用程序 API，在运行时必须保证该 ViKeyActiveX.DLL 控件存在并已注册。

2、产品专用术语介绍

2.1 硬件 ID

硬件 ID 为 32 位 bit 长度的整数，是由加密锁，加密芯片在制作过程中固化在加密芯片中的一组硬件序列号。不会产生相同的编号，全球唯一。硬件 ID 是每个加密狗的唯一标识，相当于人的身份证号。该编号可以用作跟踪各个加密设备使用情况，如防代理商窜货等功能。也可以用来作为加密算法中的一个加密条件，增强产品的安全性。

2.2 软件 ID

软件 ID 为 32 位 bit 长度的整数，主要是提供给使用加密锁的软件开发厂商，为了区分不同软件产品使用的不同硬件 ID 的加密锁的一种 ID 标识。例如一个软件需要多把加密狗都可以使用，可以利用用该软件绑定加密狗的软件 ID，判断加密狗是否为指定软件 ID，若加密狗为指定的软件 ID，则允许软件正常运行，否则退出。

2.3 用户密码

用户密码是保障普通用户权限的一个密码。如果校验的密码不正确，加密锁的操作权限只能进行各种匿名权限操作。如果校验正确，则加密锁将把权限提升用户权限。用户权限可以修改用户密码、读写数据及各种种匿名权限操作等。

2.4 管理员密码

管理员密码是保障管理员权限的一个密码，由于这个密码权限非常高级，一般应掌握在开发商核心管理人员手中。如果校验的密码不正确，加密锁的操作权限只能进行各种匿名权限操作。如果校验正确，则加密锁将把权限提升管理员权限，可以进行全部操作，如设置产品名称、软件 ID、修改普通用户密码，设置密码校验次数、读写数据等。

2.5 数据存储空间

ViKey 加密狗除了一些基本属性（软件 ID，计数器模块）之外，还提供了数据存储空间。这些数据存储空间前半部分为用户数据存储区，后半部分为管理员数据存储区。例如 ViKeyAPP 实用型加密狗共有 128 字节存储空间，其中前 64 字节为用户数据储存区，后 64 字节为管理员数据储存区，VIKeySTD 标准型加密狗拥有 2048 字节存储空间，其中前 1024 字节为用户数据储存区，后 1024 字节为管理员数据储存区。

2.6 读写权限

匿名用户拥有加密锁基本属性的读权限，例如硬件 ID(HID)、软件 ID(SID)等。

普通用户拥有普通用户数据区的读写权限、管理员数据区的读权限。其他的属性只有读权限。

管理员拥有 ViKey 加密狗的全部数据区和属性的读写权限。

3、调用流程图

API 函数调用的流程图如图 3-1 函数调用总体流程图，具体的各个操作请参考相应的例子程序。

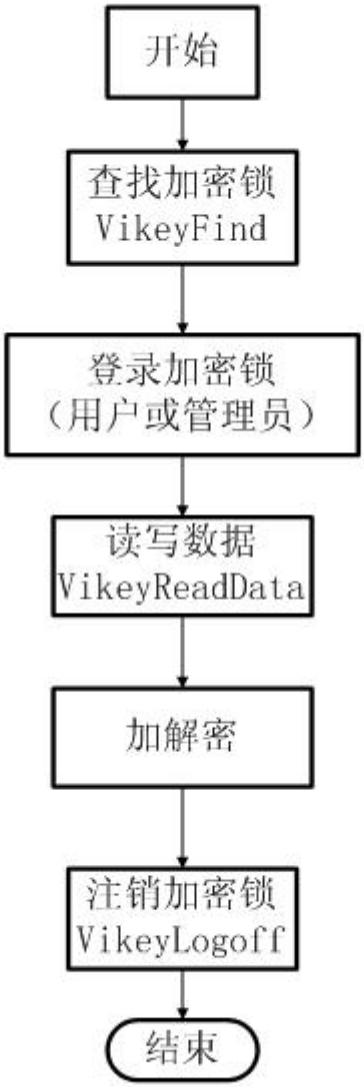


图 3-1 函数调用总体流程图

4、产品出厂默认设置

- 用户密码: “111111” 每个密码为 8 位
- 管理员密码: “000000” 每个密码为 8 位
- 用户密码尝试次数：“10” 范围 0~255
- 管理员密码尝试次数：“10”范围 0~255
- 软件 ID：“1234ABCD” 长度为 8 位
- ViKeyWeb 的 MD5Key: “FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF”。 32 字节
- 存储空间的内容全为 0；
- ViKeyTime 加密锁的时间默认是北京时间

5、API 接口函数说明

5.1 ViKeyFind 查找加密锁

函数原型	DWORD VikeyFind(DWORD* pdwCount);
功能	查找计算机上的 ViKey 加密锁
输入参数	无
输出参数	pdwCount 查找到 ViKey 加密锁的数量
权限类别	匿名
返回值	返回 0 表示成功 返回 VIKEY_ERROR_NO_VIKEY 没有找到 ViKey 加密锁

5.2 VikeyGetHID 获取加密狗的硬件 ID

函数原型	DWORD VikeyGetHID(WORD Index, DWORD *pdwHID);
功能	获取加密锁的硬件 ID（HID）
输入参数	Index ViKey 操作句柄
输出参数	pdwHID 硬件 ID
权限类别	匿名
返回值	返回 0 表示成功

5.3 VikeyGetType 获取加密狗型号

函数原型	DWORD VikeyGetType(WORD Index, VikeyType *pType)
功能	获取 ViKey 加密狗的类型
输入参数	WORD Index　　ViKey 操作句柄
输出参数	VikeyType *pType　　返回 ViKey 加密狗的类型 0: ViKeyAPP　实用型加密狗 1: ViKeySTD　标准型加密狗 2: ViKeyNET　网络型加密狗 3: ViKeyPRO　专业型加密狗 4: ViKeyWEB　身份认证型加密狗 5: ViKeyTIME　时间型加密狗
权限类别	用户权限
返回值	返回 0 表示成功

5.4 VikeyGetLevel 获取加密狗当前权限

函数原型	DWORD VikeyGetLevel(WORD Index, BYTE *pLevel);
功能	获取 ViKey 加密狗的当前的权限
输入参数	WORD Index　　ViKey 操作句柄
输出参数	BYTE *pLevel　　权限返回的指针 0　//匿名权限 1　//用户权限 2　//管理员权限
权限类别	匿名
返回值	返回 0 表示成功

5.5 VikeySetPtroductName 设置加密狗设备名称

函数原型	DWORD VikeySetPtroductName(WORD Index, WCHAR szName[16]);
功能	设置 ViKey 加密狗的设备名称 ViKeyAPP、ViKeyWEB 不支持此项功能
输入参数	WORD Index ViKey 操作句柄
输出参数	WCHAR szName[16] 最大为 16 个宽字符，UniCode 编码格式，每个宽字符为两个字节
权限类别	管理员权限
返回值	返回 0 表示成功

5.6 VikeyGetPtroductName 获取加密狗设备名称

函数原型	DWORD VikeyGetPtroductName(WORD Index, WCHAR szName[16]);
功能	获取 ViKey 加密狗的设备名称 ViKeyAPP、ViKeyWEB 不支持此项功能
输入参数	WORD Index ViKey 操作句柄 WCHAR szName[16] 最大为 16 个宽字符，UniCode 编码格式，每个宽字符为两个字节
输出参数	PVIKEYTIME pTime 返回加密锁的设备时间
权限类别	用户权限
返回值	返回 0 表示成功

5.7 VikeyUserLogon 用户权限登录

函数原型	DWORD VikeyUserLogon(WORD Index, WORD PassWord1, WORD PassWord2);
功能	以用户权限登录 ViKey 加密锁，若登陆成功，ViKey 加密狗则为用户权限
输入参数	WORD Index ViKey 操作句柄 WORD wPassWord1 用户密码 1 WORD wPassWord2 用户密码 2
输出参数	无
权限类别	匿名
返回值	返回 0 表示成功

5.8 VikeyAdminLogon 管理员权限登录

函数原型	DWORD VikeyAdminLogon(WORD Index, WORD PassWord1, WORD PassWord2);
功能	以管理员权限登录 ViKey 加密锁，若登陆成功，ViKey 加密狗则为管理员权限
输入参数	WORD Index ViKey 操作句柄 WORD wPassWord1 管理员密码 1 WORD wPassWord2 管理员密码 2
输出参数	无
权限类别	匿名
返回值	返回 0 表示成功

5.9 VikeyLogoff 注销登录

函数原型	DWORD VikeyLogoff(WORD Index);
功能	注销已经登录的加密锁，注销过后的加密锁为匿名权限
输入参数	WORD Index ViKey 操作句柄
输出参数	无
权限类别	匿名
返回值	返回 0 表示成功

5.10 VikeySetUserPassWordAttempt 设置用户密码尝试次数

函数原型	DWORD VikeySetUserPassWordAttempt(WORD Index, BYTE cAttempt);
功能	设置用户密码的尝试次数，如果使用错误密码登陆，该次数则被减一，当尝试次数变为零时，ViKey 加密狗的用户权限将被锁定，锁定后的加密狗，即便使用正确的用户密码登陆，也无法登陆成功，这种情况下需要用管理员权限，重新设置一下该次数。该功能类似于常用的银行卡的登陆密码，只有三次错误密码登陆的机会，否则银行卡将被锁定。
输入参数	WORD Index ViKey 操作句柄 BYTE cAttempt 密码尝试次数 0：不做限制 非 0：限制为指定次数
输出参数	无
权限类别	管理员
返回值	返回 0 表示成功

5.11 VikeySetAdminPassWordAttempt 设置管理员密码尝试次数

函数原型	DWORD VikeySetAdminPassWordAttempt(WORD Index, BYTE cAttempt);

功能	设置管理员密码的尝试次数， 如果使用错误密码登陆，该次数则被减一，当尝试次数变为零时，ViKey 加密狗的管理员将被锁定，锁定后的加密狗，即便使用正确的管理员密码登陆，也无法登陆成功，这种情况下只能退还给我们，才能被解锁。该功能类似于常用的银行卡的登陆密码，只有三次错误密码登陆的机会，否则银行卡将被锁定。
输入参数	WORD Index ViKey 操作句柄 BYTE cAttempt 密码尝试次数 0：不做限制 非 0：限制为指定次数
输出参数	无
权限类别	管理员
返回值	返回 0 表示成功

5.12 VikeyGetUserPassWordAttempt 获取用户密码尝试次数

函数原型	DWORD VikeyGetUserPassWordAttempt(WORD Index, BYTE *pcCurrentAttempt, BYTE *pcMaxAttempt)
功能	获取用户密码的尝试次数
输入参数	WORD Index ViKey 操作句柄
输出参数	BYTE *pcCurrentAttempt 当前密码尝试次数 0：已经被锁定，不能成功登陆 非 0：当前剩余的密码尝试次数 BYTE *pcMaxAttempt 密码尝试最大次数 0：不做限制 非 0：最大次数
权限类别	匿名
返回值	返回 0 表示成功

5.13 VikeyGetAdminPassWordAttempt 获取管理员密码尝试次数

函数原型	DWORD VikeyGetAdminPassWordAttempt(WORD Index,

	BYTE *pcCurrentAttempt, BYTE *pcMaxAttempt)
功能	获取管理员密码的尝试次数
输入参数	WORD Index ViKey 操作句柄
输出参数	BYTE *pcCurrentAttempt 当前密码尝试次数 0：已经被锁定，不能成功登陆 非 0：当前剩余的密码尝试次数 BYTE *pcMaxAttempt 密码尝试最大次数 0：不做限制 非 0：最大次数
权限类别	匿名
返回值	返回 0 表示成功

5.14 VikeySetNewPassword 设置新密码

函数原型	DWORD VikeySetNewPassword(WORD Index, DWORD PasswordFactor, WORD* pwUserPw1, WORD* pwUserPw2, WORD* pwAdminPw1, WORD* pwAdminPw2);
功能	产生用户密码和管理员密码
输入参数	WORD Index ViKey 操作句柄 DWORD PasswordFactor 密码因子
输入\输出参数	当 PasswordFactor 密码因子为 0 时，新密码分别为 pwUserPw1，pwUserPw2，pwAdminPw1，pwAdminPw2 输入的指定密码。 当 PasswordFactor 密码因子为非 0 时，加密锁会根据这个密码因子自动生成一组特定的密码，pwUserPw1，pwUserPw2，pwAdminPw1，pwAdminPw2 为返回的新密码。 WORD* pwUserPw1 用户密码 1 WORD* pwUserPw2 用户密码 2 WORD* pwAdminPw1 管理员密码 1 WORD* pwAdminPw2 管理员密码 2
权限类别	管理员

返回值	返回 0 表示成功

5.15 VikeySetSoftID 设置软件 ID

函数原型	DWORD VikeySetSoftID(WORD Index, DWORD dwSoftID);
功能	设置 ViKey 加密锁的软件 ID
输入参数	WORD Index ViKey 操作句柄 DWORD dwSoftID 软件 ID ,32bit 位的整数
输出参数	无
权限类别	管理员
返回值	返回 0 表示成功

5.16 VikeyGetSoftID 获取软件 ID

函数原型	DWORD VikeyGetSoftID(WORD Index, DWORD* pdwSoftID);
功能	获取 ViKey 加密锁的软件 ID
输入参数	WORD Index ViKey 操作句柄
输出参数	DWORD* pdwSoftID 软件 ID
权限类别	匿名
返回值	返回 0 表示成功

5.17 ViKeySetUpdateTag 设置更新标签

函数原型	DWORD ViKeySetUpdateTag(WORD Index, DWORD dwUpdateTag);
功能	设置更新标签
输入参数	WORD Index ViKey 操作句柄 DWORD dwUpdateTag 更新标签

输出参数	无
权限类别	管理员
返回值	返回 0 表示成功

5.18 ViKeyGetUpdateTag 获取更新标签

函数原型	DWORD ViKeyGetUpdateTag(WORD Index, DWORD* pdwUpdateTag);
功能	获取 ViKey 加密锁更新标签
输入参数	WORD Index ViKey 操作句柄
输出参数	DWORD* pdwUpdateTag 更新标签
权限类别	用户权限
返回值	返回 0 表示成功

5.19 ViKeySetVersionFlag 设置版本标志

函数原型	DWORD ViKeySetVersionFlag(WORD Index, DWORD dwVersionFlag);
功能	设置版本标志
输入参数	WORD Index ViKey 操作句柄 DWORD dwVersionFlag 版本标志
输出参数	无
权限类别	管理员权限
返回值	返回 0 表示成功

5.20 ViKeyGetVersionFlag 获取版本标志

函数原型	DWORD ViKeyGetVersionFlag(WORD Index, DWORD* pdwVersionFlag);
功能	获取版本标志
输入参数	WORD Index ViKey 操作句柄

输出参数	DWORD* pdwVersionFlag 版本标志
权限类别	用户权限
返回值	返回 0 表示成功

5.21 VikeyReadData 读取数据

函数原型	DWORD VikeyReadData(WORD Index, WORD Addr, WORD Length, BYTE * buffer);
功能	从 ViKey 中读取数据
输入参数	WORD Index ViKey 操作句柄 WORD Addr 地址 WORD Length 长度（单位字节）
输出参数	BYTE * buffer 数据
权限类别	用户权限
返回值	返回 0 表示成功

5.22 VikeyWriteData 写入数据

函数原型	DWORD VikeyWriteData(WORD Index, WORD Addr, WORD Length, BYTE * buffer);
功能	写入数据到 ViKey 加密锁
输入参数	WORD Index ViKey 操作句柄 WORD Addr 地址 WORD Length 长度（单位字节） BYTE * buffer 数据
输出参数	无
权限类别	用户权限

返回值	返回 0 表示成功

5.23 ViKeyRandom 获取随机数

函数原型	DWORD ViKeyRandom(WORD Index, WORD* pwRandom1, WORD* pwRandom2, WORD* pwRandom3, WORD* pwRandom4);
功能	从 ViKey 加密锁中获取四个随机数
输入参数	WORD Index ViKey 操作句柄
输出参数	WORD* pwRandom1 随机数 1 WORD* pwRandom2 随机数 2 WORD* pwRandom3 随机数 3 WORD* pwRandom4 随机数 4
权限类别	用户权限
返回值	返回 0 表示成功

5.24 VikeySeed 获取种子数

函数原型	DWORD VikeySeed(WORD Index, DWORD dwSeed, WORD* pwData1, WORD* pwData2, WORD* pwData3, WORD* pwData4);
功能	通过种子 dwSeed 获取四个数
输入参数	WORD Index ViKey 操作句柄 DWORD dwSeed 种子因子
输出参数	WORD* pwData1 数值 1 WORD* pwData2 数值 2 WORD* pwData3 数值 3 WORD* pwData4 数值 4

权限类别	用户权限
返回值	返回 0 表示成功

5.25 ViKeyDecraseModule 递减 Module

函数原型	DWORD ViKeyDecraseModule(WORD Index, WORD wModuleIndex);
功能	将指定计数器中的数值减一
输入参数	WORD Index ViKey 操作句柄 WORD wModuleIndex 计数器的序号
输出参数	无
权限类别	用户权限
返回值	返回 0 表示成功

5.26 ViKeyGetModule 获取 Module 的值

函数原型	DWORD ViKeyGetModule(WORD Index, WORD ModuleIndex, WORD* pwValue);
功能	获取 ViKey 加密狗中计数器的值
输入参数	WORD Index ViKey 操作句柄 WORD wModuleIndex 计数器的序号
输出参数	WORD* pwValue 计数器的值
权限类别	用户权限
返回值	返回 0 表示成功

5.27 ViKeySetModule 设置 Module

函数原型	DWORD ViKeySetModule(WORD Index, WORD wModuleIndex, WORD wValue, WORD wMode);

功能	设置递减计数器的初始值和模式
输入参数	WORD Index ViKey 操作句柄 WORD wModuleIndex 计数器的序号 WORD wValue 计数器的初始值 WORD wMode 计数器的模式 （1:允许递减， 0：不允许递减）
输出参数	无
权限类别	用户权限
返回值	返回 0 表示成功

5.28 ViKeyCheckModule 检查 Module

函数原型	DWORD ViKeyCheckModule(WORD Index, WORD wModuleIndex, WORD *IsZero, WORD* CanDecrase);
功能	检查计数器的数值是否为零 模式是否允许可以递减
输入参数	WORD Index ViKey 操作句柄 WORD wModuleIndex 计数器的序号
输出参数	WORD *IsZero 是否为零 WORD* CanDecrase 是否允许递减
权限类别	匿名
返回值	返回 0 表示成功

5.29 VikeySetMaxClientCount 设置客户端最大链接数

函数原型	DWORD VikeySetMaxClientCount(WORD Index, WORD dwCount);
功能	设置可以客户端的最大链接数 该函数仅对有网络功能的加密狗(ViKeyNET, ViKeyPRO)有效
输入参数	WORD Index ViKey 操作句柄

	WORD dwCount	最大链接数
输出参数	无	
权限类别	管理员权限	
返回值	返回 0 表示成功	

5.30 VikeyGetMaxClientCount 获取客户端最大链接数

函数原型	DWORD VikeyGetMaxClientCount(WORD Index, WORD* pdwCount);	
功能	获取客户端的最大链接数 该函数仅对有网络功能的加密狗(ViKeyNET, ViKeyPRO)有效	
输入参数	WORD Index	ViKey 操作句柄
输出参数	WORD *pdwCount	最大链接数
权限类别	用户权限	
返回值	返回 0 表示成功	

5.31 VikeyMD5 哈希计算

函数原型	DWORD VikeyMD5(WORD Index, WORD length, BYTE * pText, BYTE* pResult);	
功能	计算输入内容的 MD5 哈希值	
输入参数	WORD Index WORD length BYTE * pText	ViKey 操作句柄 输入内容的字节数 输入的内容指针
输出参数	BYTE* pResult	MD5 哈希结果
权限类别	匿名	
返回值	返回 0 表示成功	

5.32 VikeySetMD5Key 设置 MD5Key

函数原型	DWORD VikeySetMD5Key(WORD Index, BYTE * pMD5key);
功能	设置 HMAC_MD5 算法中的的 Key
输入参数	WORD Index ViKey 操作句柄 BYTE * pMD5key Md5key 的指针，MD5 的 key 长度为 32 字节
输出参数	无
权限类别	管理员权限
返回值	返回 0 表示成功

5.33 VikeyHmacMD5 HMAC-Md5 哈希计算

函数原型	DWORD VikeyHmacMD5(WORD Index, WORD length, BYTE * pText, BYTE* pResult);
功能	计算 HMAC-Md5 的哈希值
输入参数	WORD Index ViKey 操作句柄 WORD length 数据长度 BYTE * pText 数据内容
输出参数	BYTE* pResult 返回结果(Md5 特征码)
权限类别	用户权限
返回值	返回 0 表示成功

5.34 VikeySHA1 SHA1 哈希计算

函数原型	DWORD VikeySHA1(WORD Index, WORD length, BYTE * pText, BYTE* pResult);
功能	计算输入内容的 SHA1 哈希值
输入参数	WORD Index ViKey 操作句柄 WORD length 输入内容的字节数

	BYTE * pText	输入的内容指针
输出参数	BYTE* pResult	SHA1 哈希结果
权限类别	匿名	
返回值	返回 0 表示成功	

5.35 VikeySetSHA1Key 设置 SHA1Key

函数原型	DWORD VikeySetSHA1Key(WORD Index, BYTE * pSHA1key);	
功能	设置 HMAC_SHA1 算法中的的 Key	
输入参数	WORD Index	ViKey 操作句柄
	BYTE * pSHA1key	SHA1key 的指针，SHA1 的 key 长度为 32 字节
输出参数	无	
权限类别	管理员权限	
返回值	返回 0 表示成功	

5.36 VikeyHmacSHA1 HMAC-SHA1 哈希计算

函数原型	DWORD VikeyHmacSHA1(WORD Index, WORD length, BYTE * pText, BYTE* pResult);	
功能	计算 HMAC-SHA1 的哈希值	
输入参数	WORD Index	ViKey 操作句柄
	WORD length	数据长度
	BYTE * pText	数据内容
输出参数	BYTE* pResult	返回结果(HMAC-SHA1 特征码)
权限类别	用户权限	
返回值	返回 0 表示成功	

5.37 VikeyGetTime 获取加密锁当前时间

函数原型	DWORD VikeyGetTime(WORD Index, PVIKEYTIME pTime);
功能	获取时间型加密锁(ViKeyTime)的设备时间 该函数仅对有时间功能的加密狗(ViKeyTime)有效
输入参数	WORD Index ViKey 操作句柄
输出参数	PVIKEYTIME pTime 返回加密锁的设备时间
权限类别	匿名权限
返回值	返回 0 表示成功

5.38 VikeyGetValidTime 获取加密锁到期时间

函数原型	DWORD VikeyGetValidTime(WORD Index, PVIKEYTIME pTime);
功能	获取时间型加密锁(ViKeyTime)的到期时间 该函数仅对有时间功能的加密狗(ViKeyTime)有效
输入参数	WORD Index ViKey 操作句柄
输出参数	PVIKEYTIME pTime 返回加密锁的到期时间
权限类别	用户权限
返回值	返回 0 表示成功

5.39 VikeySetValidTime 设置加密锁到期时间

函数原型	DWORD VikeySetValidTime(WORD Index, PVIKEYTIME pTime);
功能	设置时间型加密锁(ViKeyTime)的到期时间 该函数仅对有时间功能的加密狗(ViKeyTime)有效
输入参数	WORD Index ViKey 操作句柄
输入参数	PVIKEYTIME pTime 到期时间
权限类别	管理员权限

返回值	返回 0 表示成功

5.40 VikeyCheckValidTime 检测加密锁的时钟是否到期

函数原型	DWORD VikeyCheckValidTime(WORD Index, BYTE * pIsValid);
功能	检测时间型加密锁(ViKeyTime)的时间限制功能是否到期 该函数仅对有时间功能的加密狗(ViKeyTime)有效
输入参数	WORD Index ViKey 操作句柄
输出参数	BYTE * pIsValid 返回是否到期结果 0 表示已经到期 1 表示没有到期
权限类别	用户权限
返回值	返回 0 表示成功

5.41 VikeySM3 国密 SM3 哈希算法

函数原型	DWORD VikeySM3(WORD Index, WORD length, BYTE * pText, BYTE* pResult);
功能	对输入数据进行哈希计算
输入参数	WORD Index ViKey 操作句柄 WORD length 数据长度 BYTE * pText 数据内容
输出参数	BYTE* pResult 返回 SM3 哈希结果
权限类别	匿名权限
返回值	返回 0 表示成功

5.42 VikeySM4SetKey 设置 SM4 加密算法的秘钥

函数原型	DWORD VikeySM4SetKey(WORD Index, BYTE * pKey);

功能	设置 SM4 加密算法的秘钥
输入参数	WORD Index ViKey 操作句柄 BYTE * pKey 秘钥内容
权限类别	管理员权限
返回值	返回 0 表示成功

5.43 VikeySM4Encrypt 执行 SM4 加密

函数原型	DWORD VikeySM4Encrypt(WORD Index, WORD InLength, BYTE * pText, BYTE* pResult, WORD *OutLength);
功能	执行 SM4 加密
输入参数	WORD Index ViKey 操作句柄 WORD InLength 加密数据长度 BYTE * pText 加密数据内容
输出参数	BYTE* pResult 返回加密结果数据 WORD *OutLength 返回加密结果数据的长度
权限类别	匿名权限
返回值	返回 0 表示成功

5.44 VikeySM4Decrypt 执行 SM4 解密

函数原型	DWORD VikeySM4Decrypt(WORD Index, WORD InLength, BYTE * pText, BYTE* pResult, WORD *OutLength);
功能	执行 SM4 解密
输入参数	WORD Index ViKey 操作句柄 WORD InLength 解密数据长度 BYTE * pText 解密数据内容
输出参数	BYTE* pResult 返回解密结果数据 WORD *OutLength 返回解密结果数据的长度

权限类别	匿名权限
返回值	返回 0 表示成功

5.45 VikeySM2CreateKey 创建 SM2 秘钥对

函数原型	DWORD VikeySM2CreateKey(WORD Index, BYTE * pPrivateKey, BYTE* pPublicKey);
功能	创建 SM2 秘钥对
输出参数	BYTE* pPrivateKey 返回私钥 32 字节 BYTE* pPublicKey 返回公钥 64 字节
权限类别	管理员权限
返回值	返回 0 表示成功

5.46 VikeySM2CalcPubKey 根据 SM2 私钥计算公钥

函数原型	DWORD VikeySM2CalcPubKey(WORD Index, BYTE * pPrivateKey, BYTE* pPublicKey);
功能	根据 SM2 私钥计算公钥
输入参数	BYTE * pPrivateKey SM2 密钥对私钥
输出参数	BYTE* pPublicKey SM2 密钥对公钥
权限类别	管理员权限
返回值	返回 0 表示成功

5.47 VikeySM2Sign SM2 私钥签名

函数原型	DWORD VikeySM2Sign(WORD Index, BYTE * pPrivateKey, BYTE* pUserID, WORD wUserIDLength, BYTE* pData, WORD wDataLength, BYTE* pSignR, BYTE* pSignS);
功能	SM2 私钥签名
输入参数	BYTE * pPrivateKey SM2 密钥对私钥

	BYTE* pUserID 用户 ID WORD wUserIDLength 用户 ID 数据长度 BYTE* pData 要签名的数据 WORD wDataLength 要签名的数据长度
输出参数	BYTE* pSignR 返回签名数据 R BYTE* pSignS 返回签名数据 S
权限类别	用户权限
返回值	返回 0 表示成功

5.48 VikeySM2Verify SM2 公钥验证签名

函数原型	DWORD VikeySM2Verify(WORD Index, BYTE * pPublicKey, BYTE* pUserID, WORD wUserIDLength, BYTE* pData, WORD wDataLength, BYTE* pSignR, BYTE* pSignS);
功能	SM2 公钥验证签名
输入参数	BYTE * pPublicKey SM2 密钥对公钥 BYTE* pUserID 用户 ID WORD wUserIDLength 用户 ID 数据长度 BYTE* pData 要验签的数据 WORD wDataLength 要验签的数据长度 BYTE* pSignR 签名数据 R BYTE* pSignS 签名数据 S
权限类别	用户权限
返回值	返回 0 表示验证签名通过

5.49 VikeySM2Encrypt SM2 加密

函数原型	DWORD VikeySM2Encrypt(WORD Index, BYTE * pPublicKey, BYTE* pData, WORD wDataLength, BYTE* pResult, WORD *wResultLength);
功能	SM2 加密

输入参数	BYTE * pPublicKey SM2 密钥对公钥 BYTE* pData 要加密的数据 WORD wDataLength 要加密的数据长度
输出参数	BYTE* pResult 返回加密结果数据 WORD *wResultLength 返回加密结果数据长度
权限类别	用户权限
返回值	返回 0 表示成功

5.50 VikeySM2Decrypt SM2 解密

函数原型	DWORD VikeySM2Decrypt(WORD Index, BYTE * pPrivateKey, BYTE* pData, WORD wDataLength, BYTE* pResult, WORD *wResultLength);
功能	SM2 解密
输入参数	BYTE * pPrivateKey SM2 密钥对私钥 BYTE* pData 要解密的数据 WORD wDataLength 要解密的数据长度
输出参数	BYTE* pResult 返回解密结果数据 WORD *wResultLength 返回解密结果数据长度
权限类别	用户权限
返回值	返回 0 表示成功

6、数据区权限说明

每个加密锁的数据区前半部分为用户数据区，后半部分为管理员数据区。例如 ViKeySTD 共有 2KB 的数据区，则前 1KB 为用户数据区，后 1K 为管理员数据区。下面列表对数据的权限进行解释。

	匿名权限	用户权限	管理员权限
读用户数据区	×	√	√
写用户数据区	×	√	√
读管理员数据区	×	√	√
写管理员数据区	×	×	√

7、联系我们

用户反馈

我们非常欢迎用户对我们的产品的任何反馈，请与我们直接联系，我们会给您满意的答复。

官方淘宝店:https://ivikey.taobao.com/

电话：18917081416

QQ:3243647362

客服微信