

NetSDK_C#

编程指导手册（动环主机分册）













V1.0.0

前言

符号约定

在本文档中可能出现下列标识，代表的含义如下。

标识	说明
 危险	表示有高度潜在危险，如果不能避免，会导致人员伤亡或严重伤害。
 警告	表示有中度或低度潜在危险，如果不能避免，可能导致人员轻微或中等伤害。
 注意	表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
 防静电	表示静电敏感的设备。
 当心触电	表示高压危险。
 激光辐射	表示强激光辐射。
 风扇警告	表示危险运动部件，请远离运动风扇叶片。
 当心机械伤人	表示设备部件机械伤人。
 窍门	表示能帮助您解决某个问题或节省您的时间。
 说明	表示是正文的附加信息，是对正文的强调和补充。

修订记录

版本号	修订内容	发布日期
V1.0.0	首次发布。	2021.04

名词解释

以下对本文档中使用的专业名词分别说明，帮助您更好的理解各个业务功能。

名词	说明
设备 ID	外接的用于监测搜集各种数据信息的设备，该 ID 唯一标示该设备。在接口的结构体中通常描述成 DeviceID。
监测点位	外接设备上监测某类型数值的功能抽象，用唯一 ID 标示。在接口的结构体中通常描述成 ID。
遥信	点位的报警信息。
遥测	点位监测数据上报。

目录

前言.....	I
名词解释.....	II
第 1 章 内容简介	1
1.1 概述	1
1.2 适用性.....	1
1.3 动环主机介绍.....	2
第 2 章 主要功能	3
2.1 NetSDK 初始化	3
2.1.1 简介	3
2.1.2 接口总览.....	3
2.1.3 流程说明.....	3
2.1.4 示例代码.....	4
2.2 设备登录	5
2.2.1 简介	5
2.2.2 接口总览.....	5
2.2.3 流程说明.....	5
2.2.4 示例代码.....	6
2.3 获取外设列表.....	7
2.3.1 简介	7
2.3.2 接口总览.....	7
2.3.3 流程说明.....	7
2.3.4 示例代码.....	8
2.4 获取点位信息.....	8
2.4.1 简介	8
2.4.2 接口总览.....	9
2.4.3 流程说明.....	9
2.4.4 示例代码.....	9
2.5 订阅监测点位报警.....	11
2.5.1 简介	11
2.5.2 接口总览.....	11
2.5.3 流程说明.....	11
2.5.4 示例代码.....	12
2.6 订阅监测点位实时信息	13
2.6.1 简介	13
2.6.2 接口总览.....	13
2.6.3 流程说明.....	13
2.6.4 示例代码.....	14
2.7 订阅普通报警.....	15
2.7.1 简介	15
2.7.2 接口总览.....	15
2.7.3 流程说明.....	15
2.7.4 示例代码.....	16
第 3 章 接口函数	18

3.1 NetSDK 初始化	18
3.1.1 NetSDK 初始化 Init	18
3.1.2 NetSDK 清理 Cleanup	18
3.1.3 设置断线重连回调函数 SetAutoReconnect	18
3.1.4 设置网络参数 SetNetworkParam	19
3.2 设备登录	19
3.2.1 用户登录设备 LoginWithHighLevelSecurity	19
3.2.2 用户登出设备 Logout	20
3.3 获取外设列表	20
3.3.1 获取当前主机所接入的外部设备列表 QueryDevState	20
3.4 获取点位信息	20
3.4.1 获取设备点位信息 SCADAGetAttributeInfo	20
3.5 订阅监测点位报警	21
3.5.1 订阅监测点位报警信息 SCADAAlarmAttachInfo	21
3.5.2 取消订阅点位报警信息 SCADAAlarmDetachInfo	21
3.6 订阅监测点位实时信息	22
3.6.1 订阅监测点位实时信息 SCADAAttachInfo	22
3.6.2 取消订阅点位信息 SCADADetachInfo	22
3.7 报警上报	22
3.7.1 设置报警回调函数 SetDVRMessCallBack	22
3.7.2 订阅报警 StartListen	23
3.7.3 停止订阅报警 StopListen	23
第 4 章 回调函数	24
4.1 断线回调函数 fDisConnectCallBack	24
4.2 断线重连回调函数 fHaveReConnectCallBack	24
4.3 监测点位报警回调函数 fSCADAAlarmAttachInfoCallBack	25
4.4 监测点位信息回调函数 fSCADAAttachInfoCallBack	25
4.5 报警回调函数 fMessCallBackEx	25
附录 1 法律声明	27
附录 2 网络安全建议	28

第 1 章 内容简介

1.1 概述

本文档主要介绍关于 C#的 NetSDK 库的封装工程 NetSDKCS 的参考信息，包括主要功能、接口函数和回调函数。

主要功能包括：NetSDK 初始化、设备登录、获取外设列表、获取点位信息、订阅监测点位报警、订阅监测点位实时信息、订阅设备状态报警等。

- C#的 NetSDK 库所包含的文件，请参见表 1-1。

表1-1 NetSDK 库包括的文件

库类型	库文件名称	库文件说明
功能库	dhnetsdk.dll	库文件
	avnetsdk.dll	库文件
配置库	dhconfigsdk.dll	库文件
播放（编码解码）辅助库	dhplay.dll	大华播放库
	fisheye.dll	鱼眼矫正库
dhnetsdk 辅助库	lvsDrawer.dll	图像显示库
	StreamConvertor.dll	转码库

- C#封装工程所包含的文件，请参见表 1-2。

表1-2 NetSDKCS 工程包括的文件

文件名称	文件说明
NetSDK.cs	封装给客户调用的 C#接口
NetSDKStruct.cs	存放所用到的结构体枚举等
OriginalSDK.cs	将 NetSDK 库中的 C 接口引入到 C#工程中

说明

- NetSDK 的功能库和配置库是必备库。
- 功能库是设备网络 NetSDK 的主体，主要用于网络客户端与各类产品之间的通讯交互，负责远程控制、查询、配置及码流数据的获取和处理等。
- NetSDK 库是 NetSDKCS 工程的基础，工程中 OriginalSDK.cs 文件内定义了 NetSDK 库的引用路径，最终使用时请将 NetSDK 库放到相应路径下。用户可自定义引用路径。
- 客户可以在自己的工程内直接引用本封装工程，也可以将封装工程内的文件放到自己工程内使用，还可以参考本封装工程自己封装。
- 本文档主要介绍封装 C 库接口的 C#工程的使用，更多信息请参考 C 版本 NetSDK 库内的开发文档，链接 <https://www.dahuatech.com/service/downloadlists/836.html>。

1.2 适用性

- 推荐内存：不低于 512M。
- NetSDK 支持的系统：Windows 10/Windows 8.1/Windows 7/vista 以及 Windows Server 2008/2003。

1.3 动环主机介绍

动环主机是专为动力、环境监控领域设计的一款优秀的数字监控产品。采用嵌入式 LINUX 操作系统，系统运行稳定；支持防区报警输入及输出、4~20mA 电流型传感器和 RS485 总线制传感器接入，依托公司视频监控基础，结合通用的 H.264 的视频压缩与 G.711 音频压缩技术，实现报警管理、动力环境采集与控制、视频监控、语音对讲与语音投放、网络交换以及光纤环网等为一体的、具有先进的控制技术和强大的网络数据传输能力的全方位监控方案。

产品采用嵌入式设计，安全性高、可靠性好。既可本地独立工作，也可连网组成一个强大的安全监控网，配合使用专业网络视频监控平台（网络）软件，可充分体现其强大的组网和远程监控能力。

可应用于能源、天然气、矿业、电信、电力、农业、交通、智能小区、工厂、仓库、资源、水利设施等各项领域、各部门的安全防范。

第 2 章 主要功能

2.1 NetSDK 初始化

2.1.1 简介

初始化是 NetSDK 进行各种业务的第一步。初始化本身不包含监控业务，但会设置一些影响全局业务的参数。

- NetSDK 的初始化将会占用一定的内存。
- 同一个进程内，只有第一次初始化有效。
- 使用完毕后需要调用 NetSDK 清理接口以释放资源。

2.1.2 接口总览

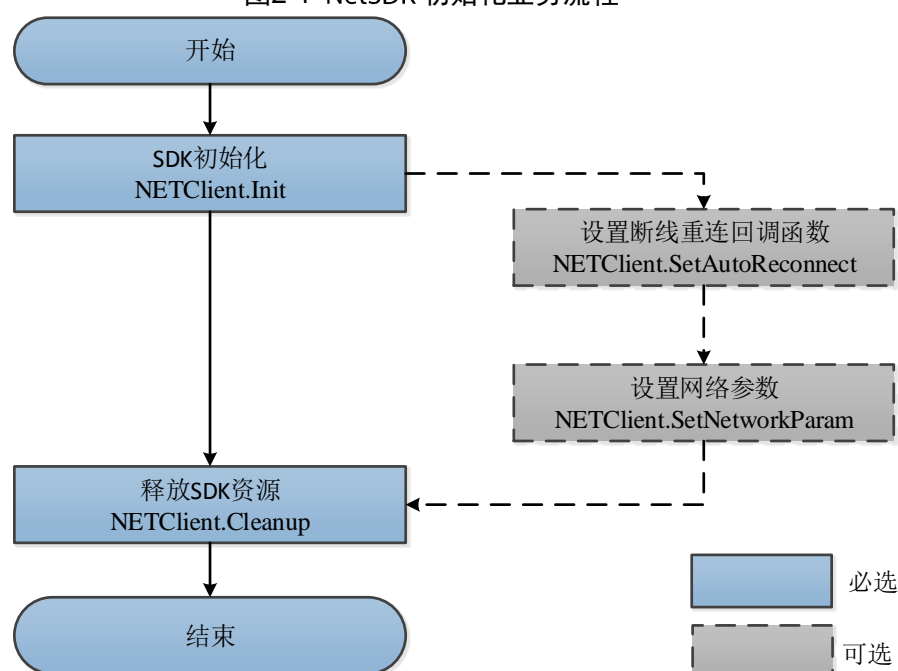
表2-1 NetSDK 初始化接口信息

接口	说明
NETClient.Init	NetSDK 初始化接口
NETClient.Cleanup	NetSDK 清理接口
NETClient.SetAutoReconnect	设置断线重连回调接口
NETClient.SetNetworkParam	设置登录网络环境接口

2.1.3 流程说明

NetSDK 初始化业务流程如图 2-1 所示。

图2-1 NetSDK 初始化业务流程



流程说明

- 步骤1 调用 `NETClient.Init` 完成 NetSDK 初始化流程。
- 步骤2 （可选）调用 `NETClient.SetAutoReconnect` 设置断线重连回调函数，设置后 NetSDK 内部断线自动重连。
- 步骤3 （可选）调用 `NETClient.SetNetworkParam` 设置网络登录参数，参数中包含登录设备超时时间和尝试次数。
- 步骤4 NetSDK 所有功能使用完后，调用 `NETClient.Cleanup` 释放 NetSDK 资源。

注意事项

- NetSDK 的 `NETClient.Init` 和 `NETClient.Cleanup` 接口需成对调用，支持单线程多次成对调用，但建议全局调用一次。
- 初始化： `NETClient.Init` 接口多次调用时，仅在内部用做计数，不会重复申请资源。
- 清理： `NETClient.Cleanup` 接口内会清理所有已开启的业务，如登录、实时监视和报警订阅等。
- 断线重连： NetSDK 可以设置断线重连功能，当遇到一些特殊情况（例如断网、断电等）设备断线时，在 NetSDK 内部会定时持续不断地进行登录操作，直至成功登录设备。断线重连后可以恢复实时监视、录像回放业务、智能事件订阅和报警订阅，其他业务无法恢复。

2.1.4 示例代码

```
// 声明静态回调委托（普通委托可能会出现回调之前已经将其释放了的错误）
private static fDisconnectCallBack m_DisConnectCallBack;    //断线回调
private static fHaveReConnectCallBack m_ReConnectCallBack; //断线重连回调

//实现委托
m_DisConnectCallBack = new fDisconnectCallBack(DisconnectCallBack);
m_ReConnectCallBack = new fHaveReConnectCallBack(ReConnectCallBack);

// 初始化 NetSDK，在初始化时实现断线回调
bool result = NETClient.Init(m_DisConnectCallBack, IntPtr.Zero, null);
if (!result)
{
    MessageBox.Show(NETClient.GetLastError()); //显示错误信息
    return;
}

//设置断线重连回调
NETClient.SetAutoReconnect(m_ReConnectCallBack, IntPtr.Zero);

//设置网络参数
NET_PARAM param = new NET_PARAM()
{
    nWaittime = 10000, // 等待超时时间(毫秒)
    nConnectTime = 5000, // 连接超时时间(毫秒)
};
```

```
NETClient.SetNetworkParam(param);  
  
// 清理初始化资源  
NETClient.Cleanup();
```

2.2 设备登录

2.2.1 简介

设备登录，即用户鉴权，是进行其他业务的前提。

用户登录设备产生唯一的登录 ID，其他功能的 NetSDK 接口需要传入登录 ID 才可执行。登出设备后，登录 ID 失效。

2.2.2 接口总览

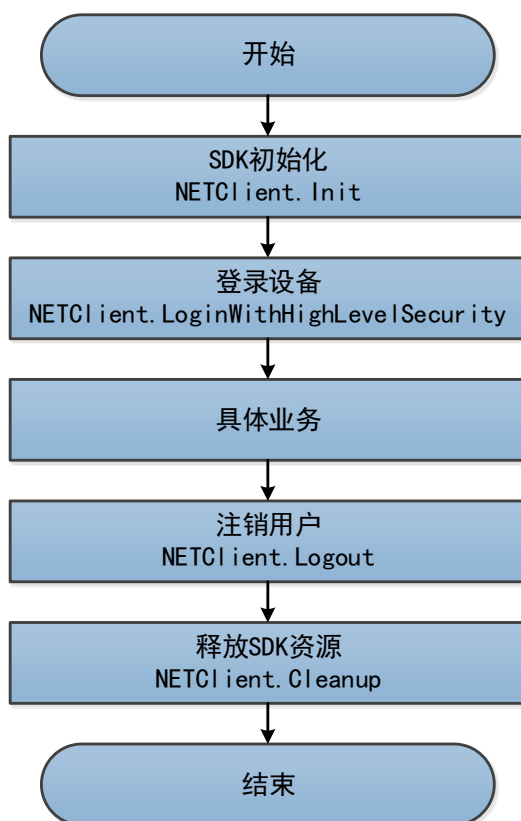
表2-2 设备登录接口信息

接口	说明
NETClient.LoginWithHighLevelSecurity	登录接口
NETClient.Logout	登出接口

2.2.3 流程说明

登录业务流程如图 2-2 所示。

图2-2 登录业务流程



流程说明

- 步骤1 调用 NETClient.Init 完成 NetSDK 初始化流程。
- 步骤2 调用 NETClient.LoginWithHighLevelSecurity 登录设备。
- 步骤3 登录成功后，用户可以实现需要的业务功能。
- 步骤4 业务使用完后，调用 NETClient.Logout 登出设备。
- 步骤5 NetSDK 功能使用完后，调用 NETClient.Cleanup 释放 NetSDK 资源。

注意事项

- 登录句柄：登录成功时接口返回值非 0（即句柄可能小于 0，也属于登录成功）；同一设备登录多次，每次的登录句柄不一样。如果无特殊业务，建议只登录一次，登录的句柄可以重复用于其他各种业务。
- 句柄重复：登录句柄有可能与存在过的句柄相同，属于正常现象。例如登录设备 A 获得 loginIDA，将 loginIDA 注销，再次进行登录操作，可能又获取到 LoginIDA。但是在句柄的整个生命周期内，不会出现重复的句柄。
- 登出：接口内部会释放登录会话中已打开的业务，但建议用户不要依赖登出接口的清理功能。例如打开监视后，在不需要使用监视时，用户应该调用结束监视的接口。
- 登录与登出配对使用，登录会消耗一定的内存和 socket 信息，在登出后释放资源。
- 登录失败：可以通过 NETClient.GetLastError 接口获取失败信息。
- 设备断线以后，设备的登录 ID 会失效，待设备重连后该登录 ID 会重新生效。

2.2.4 示例代码

```
//登录设备
NET_DEVICEINFO_Ex m_DeviceInfo = new NET_DEVICEINFO_Ex();
IntPtr m_LoginID = NETClient.LoginWithHighLevelSecurity(ip, port, name, password,
EM_LOGIN_SPAC_CAP_TYPE.TCP, IntPtr.Zero, ref m_DeviceInfo);
if (IntPtr.Zero == m_LoginID)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}

// 退出设备
if (IntPtr.Zero != m_LoginID)
{
    bool result = NETClient.Logout(m_LoginID);
    if (!result)
    {
        MessageBox.Show(this, NETClient.GetLastError());
        return;
    }
    m_LoginID = IntPtr.Zero;
}
```

2.3 获取外设列表

2.3.1 简介

获取主机上连接的设备的 ID。

2.3.2 接口总览

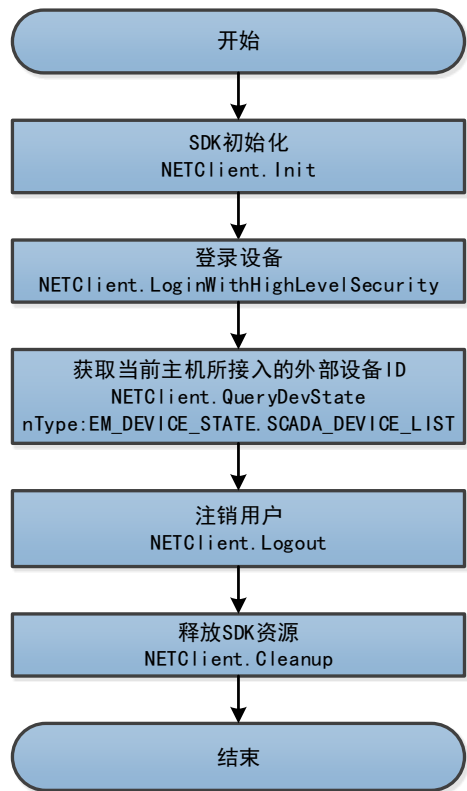
表2-3 获取外设列表接口信息

接口	说明
NETClient.QueryDevState	获取当前主机所接入的外部设备 ID

2.3.3 流程说明

NetSDK 获取外设列表流程如图 2-3 所示。

图2-3 获取外设列表业务流程



流程说明

- 步骤1 调用 NETClient.Init 完成 NetSDK 初始化流程。
- 步骤2 调用 NETClient.LoginWithHighLevelSecurity 登录设备。
- 步骤3 调用 NETClient.QueryDevState 获取外设列表，参数 nType 值为 EM_DEVICE_STATE.SCADA_DEVICE_LIST，对应结构体 NET_SCADA_DEVICE_LIST。
- 步骤4 业务使用完后，调用 NETClient.Logout 登出设备。

步骤5 NetSDK 功能使用完后，调用 NETClient.Cleanup 释放 NetSDK 资源。

2.3.4 示例代码

```
NET_SCADA_DEVICE_LIST device_list = new NET_SCADA_DEVICE_LIST();
device_list.dwSize = (uint)Marshal.SizeOf(typeof(NET_SCADA_DEVICE_LIST));
device_list.nMax = 64;
device_list.pstuDeviceIDInfo = IntPtr.Zero;
device_list.pstuDeviceIDInfo =
Marshal.AllocHGlobal(Marshal.SizeOf(typeof(NET_RECORDSET_ACCESS_CTL_CARD)));

NET_SCADA_DEVICE_ID_INFO[] list_info = new NET_SCADA_DEVICE_ID_INFO[64];
for (int i = 0; i < 64; i++)
{
    list_info[i] = new NET_SCADA_DEVICE_ID_INFO();
    Marshal.StructureToPtr(list_info[i], IntPtr.Add(device_list.pstuDeviceIDInfo, i *
Marshal.SizeOf(typeof(NET_SCADA_DEVICE_ID_INFO))), true);
}

object objInfo = device_list;
bool ret = NETClient.QueryDevState(m_LoginID, EM_DEVICE_STATE.SCADA_DEVICE_LIST, ref objInfo,
typeof(NET_SCADA_DEVICE_LIST), 10000);
if (!ret)
{
    MessageBox.Show(NETClient.GetLastError());
    return;
}
else
{
    device_list = (NET_SCADA_DEVICE_LIST)objInfo;
    for (int i = 0; i < device_list.nRet; i++)
    {
        list_info[i] =
(NET_SCADA_DEVICE_ID_INFO)Marshal.PtrToStructure(IntPtr.Add(device_list.pstuDeviceIDInfo, i *
Marshal.SizeOf(typeof(NET_SCADA_DEVICE_ID_INFO))), typeof(NET_SCADA_DEVICE_ID_INFO));
    }
}
```

2.4 获取点位信息

2.4.1 简介

针对动环主机上的设备，按照各个设备 ID 获取对应的监测点位信息。

2.4.2 接口总览

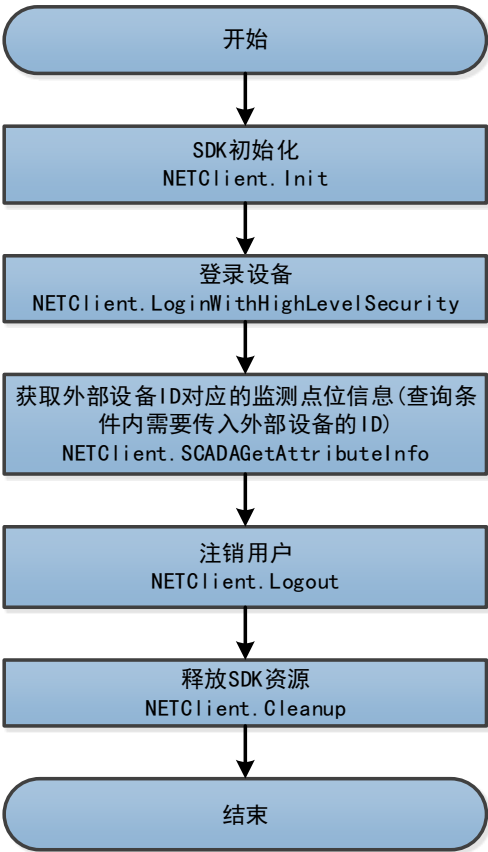
表2-4 获取点位信息的接口信息

接口	说明
NETClient.SCADAGetAttributeInfo	获取设备点位信息

2.4.3 流程说明

接口调用流程如图 2-4 所示。

图2-4 获取点位信息业务流程



流程说明

- 步骤1 调用 NETClient.Init 完成 NetSDK 初始化流程。
- 步骤2 初始化成功后，调用 NETClient.LoginWithHighLevelSecurity 登录设备。
- 步骤3 调用 NETClient.SCADAGetAttributeInfo，获取外部设备 ID 对应下的监测点位信息。其中查询条件 stuCondition 内的 szDeviceID 传入 NETClient.QueryDevState 获取到的设备 ID。
- 步骤4 业务使用完后，调用 NETClient.Logout 登出设备。
- 步骤5 NetSDK 功能使用完后，调用 NETClient.Cleanup 释放 NetSDK 资源。

2.4.4 示例代码

```
NET_IN_SCADA_GET_ATTRIBUTE_INFO stuAttributeInfoIn = new  
NET_IN_SCADA_GET_ATTRIBUTE_INFO();
```

```

stuAttributeInfoIn.dwSize = (uint)Marshal.SizeOf(typeof(NET_IN_SCADA_GET_ATTRIBUTE_INFO));
stuAttributeInfoIn.stuCondition = new NET_GET_CONDITION_INFO();
stuAttributeInfoIn.stuCondition.szDeviceID = "A15DFG56";

NET_OUT_SCADA_GET_ATTRIBUTE_INFO stuAttributeInfoOut = new
NET_OUT_SCADA_GET_ATTRIBUTE_INFO();
stuAttributeInfoOut.dwSize =
(uint)Marshal.SizeOf(typeof(NET_OUT_SCADA_GET_ATTRIBUTE_INFO));
stuAttributeInfoOut.nMaxAttributeInfoNum = 20;
stuAttributeInfoOut.pstuAttributeInfo = IntPtr.Zero;
stuAttributeInfoOut.pstuAttributeInfo =
Marshal.AllocHGlobal((int)(Marshal.SizeOf(typeof(NET_ATTRIBUTE_INFO)) *
stuAttributeInfoOut.nMaxAttributeInfoNum));

NET_ATTRIBUTE_INFO[] attributeInfoList = new
NET_ATTRIBUTE_INFO[stuAttributeInfoOut.nMaxAttributeInfoNum];
for (int index = 0; index < stuAttributeInfoOut.nMaxAttributeInfoNum; index++)
{
    IntPtr pIndexBuf = IntPtr.Add(stuAttributeInfoOut.pstuAttributeInfo, index *
Marshal.SizeOf(typeof(NET_ATTRIBUTE_INFO)));
    Marshal.StructureToPtr(attributeInfoList[index], pIndexBuf, true);
}

IntPtr pstInParam = IntPtr.Zero;
pstInParam =
Marshal.AllocHGlobal(Marshal.SizeOf(typeof(NET_IN_ACCESS_FACE_SERVICE_INSERT)));
Marshal.StructureToPtr(stuAttributeInfoIn, pstInParam, true);

IntPtr pstOutParam = IntPtr.Zero;
pstOutParam =
Marshal.AllocHGlobal(Marshal.SizeOf(typeof(NET_OUT_SCADA_GET_ATTRIBUTE_INFO)));
Marshal.StructureToPtr(stuAttributeInfoOut, pstOutParam, true);

bool ret = NETClient.SCADAGetAttributeInfo(m_LoginID, pstInParam, pstOutParam, 5000);
if (!ret)
{
    MessageBox.Show(NETClient.GetLastError());
    return;
}
else
{
    var attribute_info =
(NET_OUT_SCADA_GET_ATTRIBUTE_INFO)Marshal.PtrToStructure(pstOutParam,
typeof(NET_OUT_SCADA_GET_ATTRIBUTE_INFO));
    if (attribute_info.nRetAttributeInfoNum > 0)
    {
        for (int i = 0; i < attribute_info.nRetAttributeInfoNum; i++)

```

```
        {  
            attributelnfoList[i] =  
(NET_ATTRIBUTE_INFO)Marshal.PtrToStructure(IntPtr.Add(attribute_info.pstuAttributelnfo,  
Marshal.SizeOf(typeof(NET_ATTRIBUTE_INFO)) * i), typeof(NET_ATTRIBUTE_INFO));  
        }  
    }  
}
```

2.5 订阅监测点位报警

2.5.1 简介

监测各个监测点位的报警信息。

2.5.2 接口总览

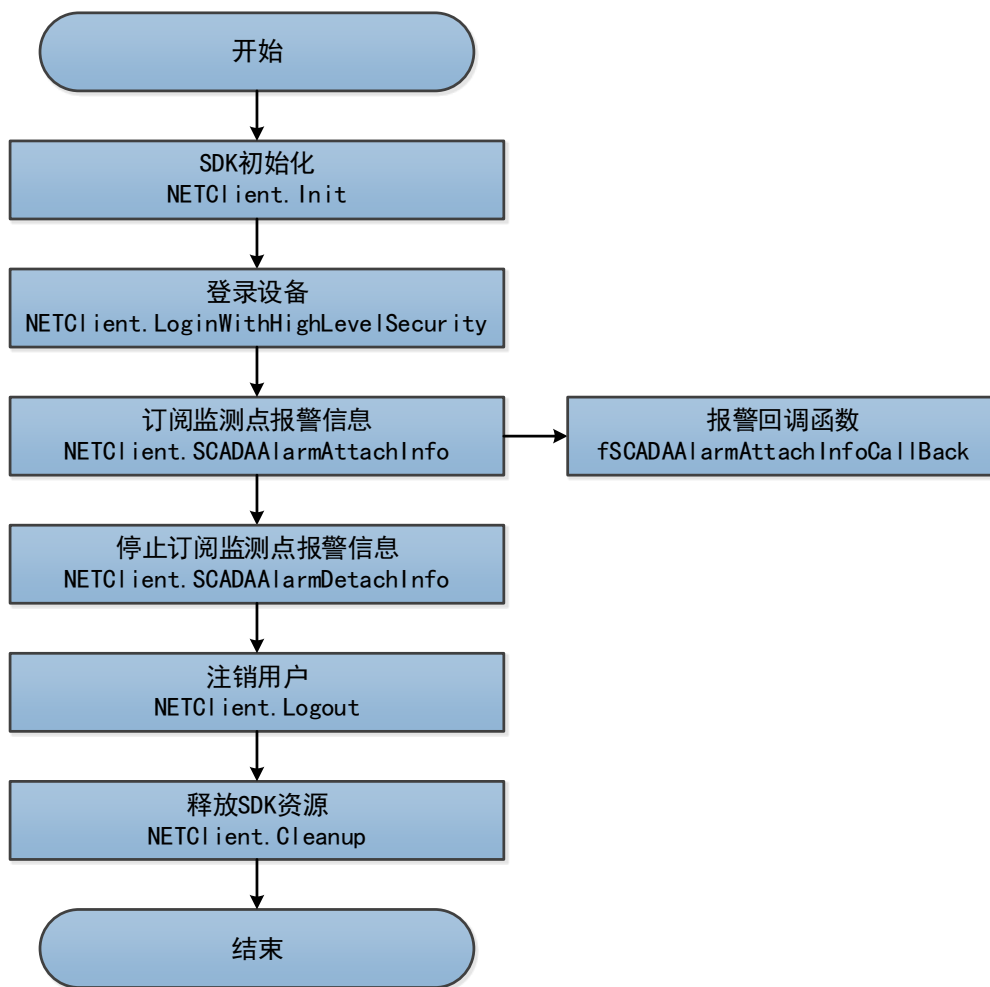
表2-5 监测点位报警接口信息

接口	说明
NETClient.SCADAAlarmAttachInfo	订阅监测点位报警信息
NETClient.SCADAAlarmDetachInfo	取消订阅监测点位报警信息

2.5.3 流程说明

监测点位报警流程如图 2-5 所示。

图2-5 订阅监测点报警业务流程



流程说明

- 步骤1 调用 NETClient.Init 完成 NetSDK 初始化流程。
- 步骤2 初始化成功后，调用 NETClient.LoginWithHighLevelSecurity 登录设备。
- 步骤3 调用 NETClient.SCADAAlarmAttachInfo 函数，向设备订阅报警。对应输入输出结构体分别为：NET_IN_SCADA_ALARM_ATTACH_INFO 和 NET_OUT_SCADA_ALARM_ATTACH_INFO。订阅成功后，设备上报的报警事件通过 NET_IN_SCADA_ALARM_ATTACH_INFO 结构体内设置的回调函数 cbCallBack 通知用户。
- 步骤4 报警上报功能使用完毕后，调用 NETClient.SCADAAlarmDetachInfo 函数停止向设备订阅报警。
- 步骤5 业务使用完后，调用 NETClient.Logout 登出设备。
- 步骤6 NetSDK 功能使用完后，调用 NETClient.Cleanup 释放 NetSDK 资源。

2.5.4 示例代码

```

NET_IN_SCADA_ALARM_ATTACH_INFO inInfo = new NET_IN_SCADA_ALARM_ATTACH_INFO();
inInfo.dwSize = (uint)Marshal.SizeOf(typeof(NET_IN_SCADA_ALARM_ATTACH_INFO));
inInfo.cbCallBack = m_SCADAAlarmAttachInfoCallBack;

NET_OUT_SCADA_ALARM_ATTACH_INFO outInfo = new NET_OUT_SCADA_ALARM_ATTACH_INFO();
    
```

```
outInfo.dwSize = (uint)Marshal.SizeOf(typeof(NET_OUT_SCADA_ALARM_ATTACH_INFO));

m_AttachAlarm = NETClient.SCADAAlarmAttachInfo(m_LoginID, inInfo, outInfo, 3000);
if (IntPtr.Zero == m_AttachAlarm)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}

bool ret = NETClient.SCADAAlarmDetachInfo(m_AttachAlarm);
if (ret)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}
m_AttachAlarm = IntPtr.Zero;

private void SCADAAlarmAttachInfoCallBack(IntPtr lAttachHandle, IntPtr pInfo, int nBufLen, IntPtr dwUser)
{
    this.BeginInvoke((Action)UpdateReConnectUI);
}
```

2.6 订阅监测点位实时信息

2.6.1 简介

监测各个监测点位普通信息上报。

2.6.2 接口总览

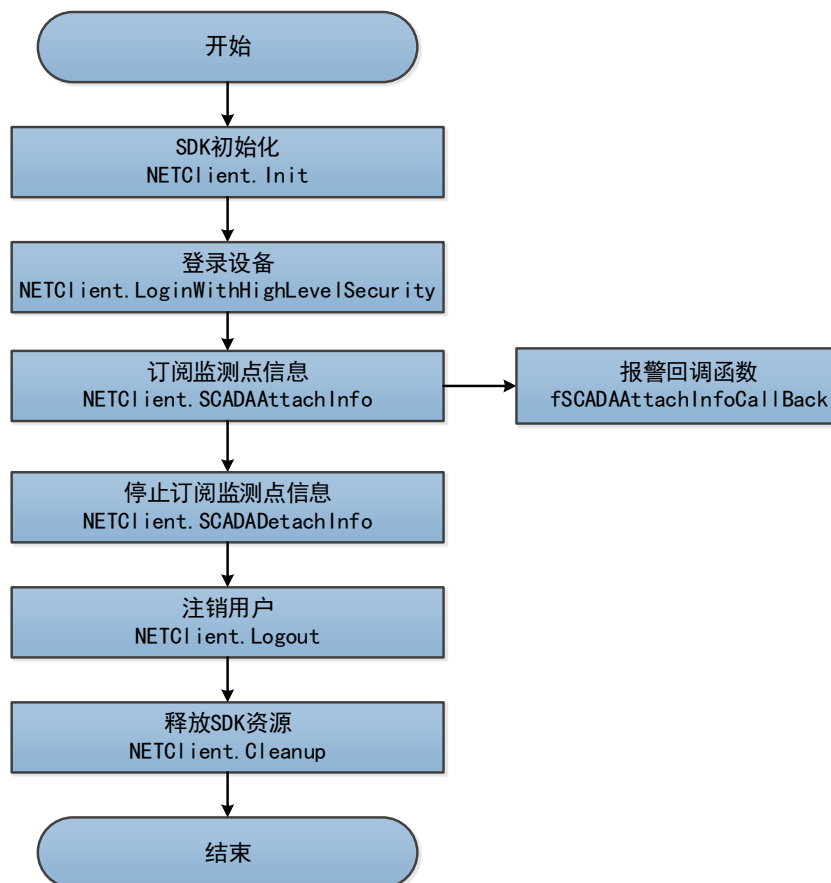
表2-6 监测点位信息

接口	说明
NETClient.SCADAAttachInfo	订阅监测点位实时信息
NETClient.SCADADetachInfo	取消监测点位信息订阅

2.6.3 流程说明

监测点位信息流程如图 2-6 所示。

图2-6 订阅监测点位实时信息业务流程



流程说明

- 步骤1 调用 NETClient.Init 完成 NetSDK 初始化流程。
- 步骤2 初始化成功后，调用 NETClient.LoginWithHighLevelSecurity 登录设备。
- 步骤3 调用 NETClient.SCADAAttachInfo 函数，向设备订阅报警。对应输入输出结构体分别为：NET_IN_SCADA_ATTACH_INFO 和 NET_OUT_SCADA_ATTACH_INFO。订阅成功后，设备上报的报警事件通过 NET_IN_SCADA_ATTACH_INFO 结构体内设置的回调函数 cbCallBack 通知用户。
- 步骤4 报警上报功能使用完毕后，调用 NETClient.SCADADetachInfo 函数停止向设备订阅报警。
- 步骤5 业务使用完后，调用 NETClient.Logout 登出设备。
- 步骤6 NetSDK 功能使用完后，调用 NETClient.Cleanup 释放 NetSDK 资源。

2.6.4 示例代码

```

NET_IN_SCADA_ATTACH_INFO inInfo = new NET_IN_SCADA_ATTACH_INFO();
inInfo.dwSize = (uint)Marshal.SizeOf(typeof(NET_IN_SCADA_ATTACH_INFO));
inInfo.cbCallBack = SCADAAttachInfoCallBack;
inInfo.emPointType = EM_NET_SCADA_POINT_TYPE.ALL;

NET_OUT_SCADA_ATTACH_INFO outInfo = new NET_OUT_SCADA_ATTACH_INFO();
outInfo.dwSize = (uint)Marshal.SizeOf(typeof(NET_OUT_SCADA_ATTACH_INFO));

m_AttachInfo = NETClient.SCADAAttachInfo(m_LoginID, inInfo, outInfo, 3000);
    
```

```
if (IntPtr.Zero == m_AttachInfo)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}

bool ret = NETClient.SCADADetachInfo(m_AttachInfo);
if (ret)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}
m_AttachInfo = IntPtr.Zero;

private void SCADAAttachInfoCallBack(IntPtr lLoginID, IntPtr lAttachHandle, IntPtr pInfo, int nBufLen,
IntPtr dwUser)
{
    this.BeginInvoke((Action)UpdateReConnectUI);
}
```

2.7 订阅普通报警

2.7.1 简介

报警上报实现方式为：通过 NetSDK 登录设备并向设备订阅报警功能，设备监测到报警事件立即发送给 NetSDK，通过报警回调函数即可获取相应报警信息。

2.7.2 接口总览

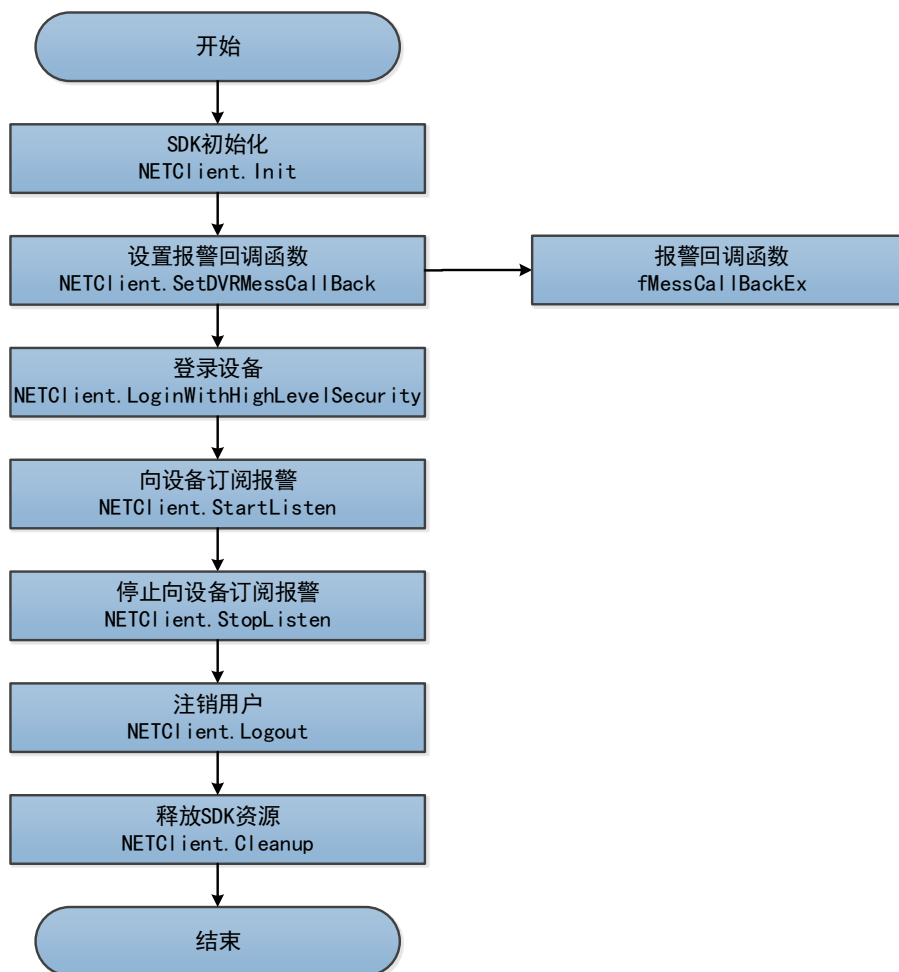
表2-7 报警上报的接口信息

接口	说明
NETClient.SetDVRMessCallBack	设置报警回调函数接口
NETClient.StartListen	订阅报警扩展接口
NETClient.StopListen	停止订阅报警

2.7.3 流程说明

报警上报流程如图 2-7 所示。

图2-7 订阅普通报警业务流程



流程说明

- 步骤1 调用 NETClient.Init 函数，完成 NetSDK 初始化流程。
- 步骤2 调用 NETClient.SetDVRMessCallBack 函数，设置报警回调函数，该接口需在报警订阅之前调用。
- 步骤3 调用 NETClient.LoginWithHighLevelSecurity 函数登录设备。
- 步骤4 调用 NETClient.StartListen 函数，向设备订阅报警。订阅成功后，设备上报的报警事件通过 NETClient.SetDVRMessCallBack 函数设置的回调函数通知用户。
- 步骤5 报警上报功能使用完毕后，调用 NETClient.StopListen 函数停止向设备订阅报警。
- 步骤6 调用 NETClient.Logout 函数登出设备。
- 步骤7 NetSDK 功能使用完后，调用 NETClient.Cleanup 函数释放 NetSDK 资源。

2.7.4 示例代码

```

// 声明静态回调委托
private static fMessCallBackEx m_AlarmCallBack;
m_AlarmCallBack = new fMessCallBackEx(AlarmCallBackEx);
// 设置报警回调
NETClient.SetDVRMessCallBack(m_AlarmCallBack, IntPtr.Zero);
  
```

```

// 报警回调处理
private bool AlarmCallBackEx(int ICommand, IntPtr ILoginID, IntPtr pBuf, uint dwBufLen, IntPtr
pchDVRIP, int nDVRPort, bool bAlarmAckFlag, int nEventID, IntPtr dwUser)
{
    EM_ALARM_TYPE type = (EM_ALARM_TYPE)ICommand;
    switch (type)
    {
        case EM_ALARM_TYPE.ALARM_SCADA_DEV_ALARM:
            NET_ALARM_SCADA_DEV_INFO info =
(NET_ALARM_SCADA_DEV_INFO)Marshal.PtrToStructure(pBuf,
typeof(NET_ALARM_SCADA_DEV_INFO));
            this.BeginInvoke(new Action(() =>
            {
                var item = new ListViewItem();
                alarm_listView.EndUpdate();
            }));
            break;
    }

    return true;
}

// 订阅报警
bool ret = NETClient.StartListen(m_LoginID);
if (!ret)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}

// 停止报警订阅
bool ret = NETClient.StopListen(m_LoginID);
if (!ret)
{
    MessageBox.Show(this, NETClient.GetLastError());
    return;
}

```

第 3 章 接口函数

3.1 NetSDK 初始化

3.1.1 NetSDK 初始化 Init

表3-1 NetSDK 初始化 Init

选项	说明	
描述	对整个 NetSDK 进行初始化	
函数	bool Init(fDisconnectCallBack cbDisconnect, IntPtr dwUser, NETSDK_INIT_PARAM? stulnitParam);	
参数	[in]cbDisconnect	断线回调函数
	[in]dwUser	断线回调函数的用户参数
	[in]stulnitParam	NetSDK 初始化参数
返回值	失败返回 false, 成功返回 true	
说明	<ul style="list-style-type: none">调用网络 NetSDK 其他函数的前提回调函数设置成 null 时, 设备断线后不会回调给用户Init 传入的 dwUser 参数, 会在回调函数 cbDisconnect 内的同字段 dwUser 返回。这样便于用户定位。其他函数同理。	

3.1.2 NetSDK 清理 Cleanup

表3-2 NetSDK 清理 Cleanup

选项	说明
描述	清理 NetSDK
函数	void Cleanup()
参数	无
返回值	无
说明	NetSDK 清理接口, 在结束前最后调用

3.1.3 设置断线重连回调函数 SetAutoReconnect

表3-3 设置断线重连回调函数 SetAutoReconnect

选项	说明
描述	设置自动重连回调函数
函数	void SetAutoReconnect(fHaveReConnectCallBack cbAutoConnect, IntPtr dwUser);

选项	说明	
参数	[in]cbAutoConnect	断线重连回调函数
	[in]dwUser	断线重连回调函数的用户参数
返回值	无	
说明	设置断线重连回调接口。如果回调函数设置为 null ，则不自动重连	

3.1.4 设置网络参数 SetNetworkParam

表3-4 设置网络参数 SetNetworkParam

选项	说明	
描述	设置网络环境相关参数	
函数	void SetNetworkParam(NET_PARAM? netParam);	
参数	[in]netParam	网络延迟、重连次数、缓存大小等参数
返回值	无	
说明	可根据实际网络环境，调整参数	

3.2 设备登录

3.2.1 用户登录设备 LoginWithHighLevelSecurity

表3-5 用户登录设备 LoginWithHighLevelSecurity

选项	说明	
描述	用户登录设备	
函数	IntPtr LoginWithHighLevelSecurity(string pchDVRIP, ushort wDVRPort, string pchUserName, string pchPassword, EM_LOGIN_SPAC_CAP_TYPE emSpecCap, IntPtr pCapParam, ref NET_DEVICEINFO_Ex deviceInfo);	
参数	[in]pchDVRIP	设备 IP
	[in]wDVRPort	设备端口
	[in]pchUserName	用户名
	[in]pchPassword	密码
	[in]emSpecCap	登录类别
	[in]pCapParam	登录类别参数
	[out]deviceInfo	设备信息
返回值	失败返回 0，成功返回非 0 的值	
说明	无	

3.2.2 用户登出设备 Logout

表3-6 用户登出设备 Logout

选项	说明	
描述	用户登出设备	
函数	bool Logout(IntPtr ILoginID);	
参数	[in]ILoginID	NETClient.LoginWithHighLevelSecurity 的返回值
返回值	失败返回 false，成功返回 true	
说明	无	

3.3 获取外设列表

3.3.1 获取当前主机所接入的外部设备列表 QueryDevState

表3-7 获取外设列表 QueryDevState

选项	说明	
描述	获取当前主机所接入的外部设备列表	
函数	bool QueryDevState(IntPtr ILoginID, int nType, ref object obj, Type typeName, int waittime);	
参数	[in]ILoginID	NETClient.LoginWithHighLevelSecurity 返回值
	[in]nType	查询信息类型，当获取外设列表时 nType 为 EM_DEVICE_STATE.SCADA_DEVICE_LIST
	[out]obj	用于接收查询返回的数据的缓存，对应 NET_SCADA_DEVICE_LIST 结构体
	[in]typeName	查询结构体的类型
	[in]waittime	查询状态等待时间
返回值	失败返回 false，成功返回 true	
说明	无	

3.4 获取点位信息

3.4.1 获取设备点位信息 SCADAGetAttributeInfo

表3-8 获取设备点位信息 SCADAGetAttributeInfo

选项	说明
描述	获取设备点位信息

选项	说明	
函数	bool SCADAGetAttributeInfo(IntPtr ILoginID, IntPtr pstInParam, IntPtr pstOutParam, int nWaitTime);	
参数	[in]ILoginID	NETClient.LoginWithHighLevelSecurity 返回值
	[in]pstInParam	输入参数
	[out]pstOutParam	输出参数
	[in]nWaitTime	查询状态等待时间
返回值	失败返回 0，成功返回非 0 的值	
说明	无	

3.5 订阅监测点位报警

3.5.1 订阅监测点位报警信息 SCADAAlarmAttachInfo

表3-9 订阅监测点位报警信息 SCADAAlarmAttachInfo

选项	说明	
描述	订阅监测点位报警信息	
函数	IntPtr SCADAAlarmAttachInfo(IntPtr ILoginID, NET_IN_SCADA_ALARM_ATTACH_INFO pInParam, NET_OUT_SCADA_ALARM_ATTACH_INFO pOutParam, int nWaitTime = 3000);	
参数	[in]ILoginID	NETClient.LoginWithHighLevelSecurity 返回值
	[in]pInParam	订阅输入参数
	[out]pOutParam	订阅输出参数
	[in]waittime	等待时间
返回值	失败返回 0，成功返回非 0 的值	
说明	无	

3.5.2 取消订阅点位报警信息 SCADAAlarmDetachInfo

表3-10 取消订阅点位报警信息 SCADAAlarmDetachInfo

选项	说明	
描述	取消订阅点位报警信息	
函数	bool SCADAAlarmDetachInfo(IntPtr IAttachHandle)	
参数	[in]IAttachHandle	NETClient.SCADAAlarmAttachInfo 返回值
返回值	失败返回 false，成功返回 true	

选项	说明
说明	无

3.6 订阅监测点位实时信息

3.6.1 订阅监测点位实时信息 SCADAAttachInfo

表3-11 订阅监测点位实时信息 SCADAAttachInfo

选项	说明	
描述	订阅监测点位实时信息	
函数	<pre>IntPtr SCADAAttachInfo(IntPtr ILoginID, NET_IN_SCADA_ATTACH_INFO pInParam, NET_OUT_SCADA_ATTACH_INFO pOutParam, int nWaitTime);</pre>	
参数	[in]ILoginID	NETClient.LoginWithHighLevelSecurity 返回值
	[in]pInParam	订阅输入参数
	[out]pOutParam	订阅输出参数
	[in]waittime	等待时间
返回值	失败返回 0，成功返回非 0 的值	
说明	无	

3.6.2 取消订阅点位信息 SCADADetachInfo

表3-12 取消订阅点位信息 SCADADetachInfo

选项	说明	
描述	取消订阅点位信息	
函数	<pre>bool SCADADetachInfo(IntPtr IAttachHandle);</pre>	
参数	[in]IAttachHandle	NETClient.SCADAAttachInfo 返回值
返回值	失败返回 false，成功返回 true	
说明	无	

3.7 报警上报

3.7.1 设置报警回调函数 SetDVRMessCallBack

表3-13 设置报警回调函数 SetDVRMessCallBack

选项	说明
描述	设置报警回调函数

选项	说明	
函数	<pre>void SetDVRMessCallBack(fMessCallBackEx cbMessage, IntPtr dwUser);</pre>	
参数	[in]cbMessage	<ul style="list-style-type: none"> 消息回调函数，可以回调设备的状态，如报警状态 当设置为 0 时表示禁止回调
	[in]dwUser	用户自定义数据
返回值	无	
说明	<ul style="list-style-type: none"> 设置设备消息回调函数，用来得到设备当前状态信息，与调用顺序无关，NetSDK 默认不回调 此回调函数 fMessCallBack 必须先调用报警消息订阅接口 StartListen 才生效 	

3.7.2 订阅报警 StartListen

表3-14 订阅报警 StartListen

选项	说明	
描述	订阅报警	
函数	<pre>bool StartListen(IntPtr lLoginID);</pre>	
参数	[in]lLoginID	NETClient.LoginWithHighLevelSecurity 返回值
返回值	失败返回 false，成功返回 true	
说明	订阅设备消息，得到的消息从 SetDVRMessCallBack 的设置值回调出来	

3.7.3 停止订阅报警 StopListen

表3-15 停止订阅报警 StopListen

选项	说明	
描述	停止订阅报警	
函数	<pre>bool StopListen(IntPtr lLoginID);</pre>	
参数	[in]lLoginID	NETClient.LoginWithHighLevelSecurity 返回值
返回值	失败返回 false，成功返回 true	
说明	无	

第 4 章 回调函数

4.1 断线回调函数 fDisconnectCallBack

表4-1 断线回调函数 fDisconnectCallBack

选项	说明	
描述	断线回调函数	
函数	public delegate void fDisconnectCallBack(IntPtr lLoginID, IntPtr pchDVRIP, int nDVRPort, IntPtr dwUser);	
参数	[out]lLoginID	NETClient.LoginWithHighLevelSecurity 的返回值
	[out]pchDVRIP	断线的设备 IP
	[out]nDVRPort	断线的设备端口
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.2 断线重连回调函数 fHaveReConnectCallBack

表4-2 断线重连回调函数 fHaveReConnectCallBack

选项	说明	
描述	断线重连回调函数	
函数	public delegate void fHaveReConnectCallBack(IntPtr lLoginID, IntPtr pchDVRIP, int nDVRPort, IntPtr dwUser);	
参数	[out]lLoginID	NETClient.LoginWithHighLevelSecurity 的返回值
	[out]pchDVRIP	断线后重连成功的设备 IP
	[out]nDVRPort	断线后重连成功的设备端口
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.3 监测点位报警回调函数 fSCADAAlarmAttachInfoCallBack

表4-3 监测点位报警信息回调函数 fSCADAAlarmAttachInfoCallBack

选项	说明	
描述	监测点位报警信息回调函数	
函数	<pre>public delegate void fSCADAAlarmAttachInfoCallBack(IntPtr lAttachHandle, IntPtr pInfo, int nBufLen, IntPtr dwUser);</pre>	
参数	[out]lAttachHandle	NETClient.SCADAAlarmAttachInfo 的返回值
	[out]pInfo	报警信息数据块地址
	[out]nBufLen	报警信息数据块的长度，单位：字节
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.4 监测点位信息回调函数 fSCADAAttachInfoCallBack

表4-4 监测点位信息回调函数 fSCADAAttachInfoCallBack

选项	说明	
描述	监测点位信息回调函数	
函数	<pre>public delegate void fSCADAAttachInfoCallBack(IntPtr lLoginID, IntPtr lAttachHandle, IntPtr pInfo, int nBufLen, IntPtr dwUser);</pre>	
参数	[out]lLoginID	NETClient.LoginWithHighLevelSecurity 的返回值
	[out]lAttachHandle	NETClient.SCADAAttachInfo 的返回值
	[out]pInfo	上报信息数据块地址
	[out]nBufLen	上报信息数据块的长度，单位：字节
	[out]dwUser	回调函数的用户参数
返回值	无	
说明	无	

4.5 报警回调函数 fMessCallBackEx

表4-5 报警回调函数 fMessCallBackEx

选项	说明
描述	报警回调函数

选项	说明	
函数	<pre>public delegate bool fMessCallBackEx(int ICommand, IntPtr ILoginID, IntPtr pBuf, uint dwBufLen, IntPtr pchDVRIP, int nDVRPort, bool bAlarmAckFlag, int nEventID, IntPtr dwUser);</pre>	
参数	[out]ICommand	报警类型，具体请参见表 4-6
	[out]ILoginID	登录接口返回值
	[out]pBuf	接收报警数据的缓存，根据调用的侦听接口和 ICommand 值不同，填充的数据不同
	[out]dwBufLen	pBuf 的长度，单位：字节
	[out]pchDVRIP	设备 ip
	[out]nDVRPort	端口
	[out]bAlarmAckFlag	<ul style="list-style-type: none"> TRUE: 该事件为可以进行确认的事件 FALSE: 该事件无法进行确认
	[out]nEventID	返回事件的 ID
	[out]dwUser	用户自定义数据
返回值	TRUE 回调函数执行正确，FALSE 执行错误	
说明	一般在应用程序初始化时调用设置回调，在回调函数中根据不同的设备 ID 和命令值做出不同的处理	

表4-6 报警类型说明

报警类型	报警类型名称	pBuf
ALARM_SCADA_DEV_ALARM	检测采集设备报警事件	NET_ALARM_SCADA_DEV_INFO 结构体

附录1 法律声明

商标声明

- VGA 是 IBM 公司的商标。
- Windows 标识和 Windows 是微软公司的商标或注册商标。
- 在本文档中可能提及的其他商标或公司的名称，由其各自所有者拥有。

责任声明

- 在适用法律允许的范围内，在任何情况下，本公司都不对因本文档中相关内容及描述的产品而产生任何特殊的、附随的、间接的、继发性的损害进行赔偿，也不对任何利润、数据、商誉、文档丢失或预期节约的损失进行赔偿。
- 本文档中描述的产品均“按照现状”提供，除非适用法律要求，本公司对文档中的所有内容不提供任何明示或暗示的保证，包括但不限于适销性、质量满意度、适合特定目的、不侵犯第三方权利等保证。

隐私保护提醒

您安装了我们的产品，您可能会采集人脸、指纹、车牌、邮箱、电话、GPS 等个人信息。在使用产品过程中，您需要遵守所在地区或国家的隐私保护法律法规要求，保障他人的合法权益。如，提供清晰、可见的标牌，告知相关权利人视频监控区域的存在，并提供相应的联系方式。

关于本文档

- 本文档供多个型号产品使用，产品外观和功能请以实物为准。
- 如果不按照本文档中的指导进行操作而造成的任何损失由使用方自己承担。
- 本文档会实时根据相关地区的法律法规更新内容，具体请参见产品的纸质、电子光盘、二维码或官网，如果纸质与电子档内容不一致，请以电子档为准。
- 本公司保留随时修改本文档中任何信息的权利，修改的内容将会在本文档的新版本中加入，恕不另行通知。
- 本文档可能包含技术上不准确的地方、或与产品功能及操作不相符的地方、或印刷错误，以公司最终解释为准。
- 如果获取到的 PDF 文档无法打开，请使用最新版本或最主流的阅读工具。

附录2 网络安全建议

保障设备基本网络安全的必须措施：

1. 使用复杂密码

请参考如下建议进行密码设置：

- 长度不小于 8 个字符。
- 至少包含两种字符类型，字符类型包括大小写字母、数字和符号。
- 不包含账户名称或账户名称的倒序。
- 不要使用连续字符，如 123、abc 等。
- 不要使用重叠字符，如 111、aaa 等。

2. 及时更新固件和客户端软件

- 按科技行业的标准作业规范，设备的固件需要及时更新至最新版本，以保证设备具有最新的功能和安全性。设备接入公网情况下，建议开启在线升级自动检测功能，便于及时获知厂商发布的固件更新信息。
- 建议您下载和使用最新版本客户端软件。

增强设备网络安全的建议措施：

1. 物理防护

建议您对设备（尤其是存储类设备）进行物理防护，比如将设备放置在专用机房、机柜，并做好门禁权限和钥匙管理，防止未经授权的人员进行破坏硬件、外接设备（例如 U 盘、串口）等物理接触行为。

2. 定期修改密码

建议您定期修改密码，以降低被猜测或破解的风险。

3. 及时设置、更新密码重置信息

设备支持密码重置功能，为了降低该功能被攻击者利用的风险，请您及时设置密码重置相关信息，包含预留手机号/邮箱、密保问题，如有信息变更，请及时修改。设置密保问题时，建议不要使用容易猜测的答案。

4. 开启账户锁定

出厂默认开启账户锁定功能，建议您保持开启状态，以保护账户安全。在攻击者多次密码尝试失败后，其对应账户及源 IP 将会被锁定。

5. 更改 HTTP 及其他服务默认端口

建议您将 HTTP 及其他服务默认端口更改为 1024~65535 间的任意端口，以减小被攻击者猜测服务端口的风险。

6. 使能 HTTPS

建议您开启 HTTPS，通过安全的通道访问 Web 服务。

7. MAC 地址绑定

建议您在设备端将其网关设备的 IP 与 MAC 地址进行绑定，以降低 ARP 欺骗风险。

8. 合理分配账户及权限

根据业务和管理需要，合理新增用户，并合理为其分配最小权限集合。

9. 关闭非必需服务，使用安全的模式

如果没有需要，建议您关闭 SNMP、SMTP、UPnP 等功能，以降低设备面临的风险。

如果有需要，强烈建议您使用安全的模式，包括但不限于：

- SNMP：选择 SNMP v3，并设置复杂的加密密码和鉴权密码。
- SMTP：选择 TLS 方式接入邮箱服务器。
- FTP：选择 SFTP，并设置复杂密码。
- AP 热点：选择 WPA2-PSK 加密模式，并设置复杂密码。

10. 音视频加密传输

如果您的音视频数据包含重要或敏感内容，建议启用加密传输功能，以降低音视频数据传输过程中被窃取的风险。

11. 安全审计

- 查看在线用户：建议您不定期查看在线用户，识别是否有非法用户登录。
- 查看设备日志：通过查看日志，可以获知尝试登录设备的 IP 信息，以及已登录用户的关键操作信息。

12. 网络日志

由于设备存储容量限制，日志存储能力有限，如果您需要长期保存日志，建议您启用网络日志功能，确保关键日志同步至网络日志服务器，便于问题回溯。

13. 安全网络环境的搭建

为了更好地保障设备的安全性，降低网络安全风险，建议您：

- 关闭路由器端口映射功能，避免外部网络直接访问路由器内网设备的服务。
- 根据实际网络需要，对网络进行划区隔离：若两个子网间没有通信需求，建议使用 VLAN、网闸等方式对其进行网络分割，达到网络隔离效果。
- 建立 802.1x 接入认证体系，以降低非法终端接入专网的风险。
- 开启设备 IP/MAC 地址过滤功能，限制允许访问设备的主机范围。