

设备(可视对讲室内机)

网络 SDK 编程指南

V5.1

声 明

非常感谢您购买我公司的产品，如果您有什么疑问或需要请随时联系我们。

- 我们已尽量保证手册内容的完整性与准确性，但也不免出现技术上不准确、与产品功能及操作不相符或印刷错误等情况，如有任何疑问或争议，请以我司最终解释为准。
- 产品和手册将实时进行更新，恕不另行通知。
- 本手册中内容仅为用户提供参考指导作用，请以 SDK 实际内容为准。

目 录

声 明	I
目 录	II
1 SDK 简介	5
2 函数说明	8
2.1 SDK 初始化	8
2.1.1 初始化 SDK NET_DVR_Init	8
2.1.2 释放 SDK 资源 NET_DVR_Cleanup	8
2.2 SDK 本地功能	8
SDK 本地参数配置	8
2.2.1 获取 SDK 本地参数 NET_DVR_GetSDKLocalCfg	8
2.2.2 设置 SDK 本地参数 NET_DVR_SetSDKLocalCfg	9
连接和接收超时时间及重连设置	10
2.2.3 设置网络连接超时时间和连接尝试次数 NET_DVR_SetConnectTime	10
2.2.4 设置重连功能 NET_DVR_SetReconnect	10
2.2.5 设置接收超时时间 NET_DVR_SetRecvTimeOut	10
多网卡绑定	10
2.2.6 获取所有 IP，用于支持多网卡接口 NET_DVR_GetLocalIP	10
2.2.7 设置 IP 绑定 NET_DVR_SetValidIP	11
SDK 版本、状态和能力	11
2.2.8 获取 SDK 的版本号和 build 信息 NET_DVR_GetSDKBuildVersion	11
2.2.9 获取当前 SDK 的状态信息 NET_DVR_GetSDKState	11
2.2.10 获取当前 SDK 的功能信息 NET_DVR_GetSDKAbility	11
SDK 启用写日志	12
2.2.11 启用写日志文件 NET_DVR_SetLogToFile	12
异常消息回调	12
2.2.12 注册接收异常、重连等消息的窗口句柄或回调函数 NET_DVR_SetExceptionCallBack_V30	12
获取错误信息	14
2.2.13 返回最后操作的错误码 NET_DVR_GetLastError	14
2.2.14 返回最后操作的错误码信息 NET_DVR_GetErrorMsg	14
2.3 用户注册	15
2.3.1 激活设备 NET_DVR_ActivateDevice	15
2.3.2 通过解析服务器，获取设备的动态 IP 地址和端口号 NET_DVR_GetDVRIPByResolveSvr_EX	15
2.3.3 用户注册设备 NET_DVR_Login_V40	16
2.3.4 用户注销 NET_DVR_Logout	16
2.4 获取设备能力集	16
2.4.1 获取设备能力集 NET_DVR_GetDeviceAbility	16
2.5 实时预览	17
2.5.1 主码流动态产生一个关键帧 NET_DVR_MakeKeyFrame	17
2.5.2 子码流动态产生一个关键帧 NET_DVR_MakeKeyFrameSub	18
2.5.3 实时预览 NET_DVR_RealPlay_V40	18
2.5.4 停止预览 NET_DVR_StopRealPlay	19
2.5.5 获取预览时用来解码和显示的播放库句柄 NET_DVR_GetRealPlayerIndex	19

2.6	预览时播放声音控制	19
2.6.1	设置声音播放模式 NET_DVR_SetAudioMode	19
2.6.2	独占声卡模式下开启声音 NET_DVR_OpenSound	20
2.6.3	独占声卡模式下关闭声音 NET_DVR_CloseSound	20
2.6.4	共享声卡模式下开启声音 NET_DVR_OpenSoundShare	20
2.6.5	共享声卡模式下关闭声音 NET_DVR_CloseSoundShare	20
2.6.6	调节播放音量 NET_DVR_Volume	20
2.7	实时预览数据捕获	21
2.7.1	注册实时码流数据回调函数 NET_DVR_SetRealDataCallBack	21
2.7.2	注册标准 RTP 码流数据回调函数 NET_DVR_SetStandardDataCallBack	21
2.7.3	捕获数据并保存到指定的文件中 NET_DVR_SaveRealData	22
2.7.4	停止数据捕获 NET_DVR_StopSaveRealData	22
2.8	预览抓图	23
2.8.1	设置抓图模式 NET_DVR_SetCapturePictureMode	23
2.8.2	预览时，单帧数据捕获图片并保存成文件 NET_DVR_CapturePicture	23
2.9	设备 JPEG 抓图	24
2.9.1	单帧数据捕获并保存成 JPEG 图片 NET_DVR_CaptureJPEGPicture	24
2.10	布防、撤防	24
	设置报警等信息上传的回调函数	24
2.10.1	注册报警回调函数 NET_DVR_SetDVRMessageCallBack_V31	24
	布防撤防	25
2.10.2	建立报警上传通道，获取报警等信息 NET_DVR_SetupAlarmChan_V41	25
2.10.3	撤销报警上传通道 NET_DVR_CloseAlarmChan_V30	25
2.11	监听报警	25
2.11.1	启动监听，接收设备主动上传的报警等信息 NET_DVR_StartListen_V30	25
2.11.2	停止监听（支持多线程）NET_DVR_StopListen_V30	26
2.12	远程参数配置	26
	通用参数配置	26
2.12.1	获取设备的配置信息 NET_DVR_GetDVRConfig	26
2.12.2	设置设备的配置信息 NET_DVR_SetDVRConfig	27
	可视对讲参数配置	28
2.12.3	获取设备的配置信息 NET_DVR_GetDVRConfig	28
2.12.4	设置设备的配置信息 NET_DVR_SetDVRConfig	29
2.12.5	批量获取配置信息 NET_DVR_GetDeviceConfig	29
2.12.6	批量设置配置信息 NET_DVR_SetDeviceConfig	30
	长连接参数配置	31
2.12.7	启动长连接远程配置 NET_DVR_StartRemoteConfig	31
2.12.8	发送长连接数据 NET_DVR_SendRemoteConfig	32
2.12.9	关闭长连接配置接口所创建的句柄，释放资源 NET_DVR_StopRemoteConfig	33
	获取设备支持的云台协议	33
2.12.10	获取设备支持的云台协议 NET_DVR_GetPTZProtocol	33
2.13	远程控制	33
2.13.1	远程控制 NET_DVR_RemoteControl	33
2.14	语音对讲	34

2.14.1	启动语音对讲 NET_DVR_StartVoiceCom_V30	34
2.14.2	设置语音对讲客户端的音量 NET_DVR_SetVoiceComClientVolume	35
2.14.3	停止语音对讲 NET_DVR_StopVoiceCom	35
2.15	设备维护管理	35
	在线状态检测	35
2.15.1	设备在线状态检测 NET_DVR_RemoteControl	35
	远程升级	36
2.15.2	设置远程升级时网络环境 NET_DVR_SetNetworkEnvironment	36
2.15.3	远程升级 NET_DVR_Upgrade	36
2.15.4	获取远程升级的进度 NET_DVR_GetUpgradeProgress	36
2.15.5	获取远程升级的状态 NET_DVR_GetUpgradeState	37
2.15.6	关闭远程升级句柄, 释放资源 NET_DVR_CloseUpgradeHandle	37
	日志查找	37
2.15.7	查找设备的日志信息 NET_DVR_FindDVRLog_V30	37
2.15.8	逐条获取查找到的日志信息 NET_DVR_FindNextLog_V30	38
2.15.9	释放查找日志的资源 NET_DVR_FindLogClose_V30	38
	恢复设备默认参数	38
2.15.10	恢复设备默认参数 NET_DVR_RestoreConfig	38
2.15.11	完全恢复出厂默认参数 NET_DVR_RemoteControl	38
	导入/导出配置文件	39
2.15.12	导出配置文件 NET_DVR_GetConfigFile_V30	39
2.15.13	导出配置文件 NET_DVR_GetConfigFile	39
2.15.14	导入配置文件 NET_DVR_SetConfigFile_EX	39
2.15.15	导入配置文件 NET_DVR_SetConfigFile	40
	远程重启	40
2.15.16	重启设备 NET_DVR_RebootDVR	40
3	错误代码及说明	41
3.1	网络通讯库错误码	41
3.2	RTSP 通讯库错误码	46
3.3	软解码库错误码	47
4	附录:结构体	48

1 SDK 简介

设备网络 SDK 是基于设备私有网络通信协议开发的，为嵌入式网络硬盘录像机、NVR、视频服务器、网络摄像机、网络球机、解码器、报警主机、可视对讲主机等网络产品服务的配套模块，用于远程访问和控制设备软件的二次开发。本文档主要介绍可视对讲主机的室内机相关功能。

设备网络 SDK 主要功能：实时码流预览、语音对讲、参数配置等功能。

注意：门口机、管理机等更多接口说明及更新请参见《设备网络 SDK 使用手册.chm》。

设备网络 SDK 包含网络通讯库、软解码库、硬解码库等功能组件，我们提供 Windows 和 Linux 两个版本的 SDK，各自所包含的组件如下：

表 1.1 Windows SDK 组件

网络通讯库	外部接口	HCNetSDK.h	头文件	
		HCNetSDK.lib	LIB 库文件	
		HCNetSDK.dll	DLL 库文件	
	核心组件	HCCore.lib	LIB 库文件	
		HCCore.dll	DLL 库文件	
组件库	设备配置核心组件	HCCoreDevCfg.dll	DLL 库文件	HCNetSDKCom 文件夹
	预览组件	HCPreview.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCPreview.dll	DLL 库文件	HCNetSDKCom 文件夹
	回放组件	HCPlayBack.dll	DLL 库文件	HCNetSDKCom 文件夹
	语音组件	HCVoiceTalk.dll	DLL 库文件	HCNetSDKCom 文件夹
	报警组件	HCAalarm.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCAalarm.dll	DLL 库文件	HCNetSDKCom 文件夹
	显示组件	HCDisplay.dll	DLL 库文件	HCNetSDKCom 文件夹
	行业应用管理配置组件	HCIIndustry.dll	DLL 库文件	HCNetSDKCom 文件夹
	维护管理配置组件	HCGeneralCfgMgr.lib	LIB 库文件	HCNetSDKCom 文件夹
		HCGeneralCfgMgr.dll	DLL 库文件	HCNetSDKCom 文件夹
RTSP 通讯库		StreamTransClient.dll	DLL 库文件	HCNetSDKCom 文件夹
转封装库		SystemTransform.dll	DLL 库文件	HCNetSDKCom 文件夹
语音对讲库		AudioIntercom.dll	DLL 库文件	HCNetSDKCom 文件夹
		OpenAL32.dll	DLL 库文件	HCNetSDKCom 文件夹
字符转码库		libiconv2.dll	DLL 库文件	HCNetSDKCom 文件夹
模拟能力集		LocalXml.zip	XML 文件包	
软解码库		plaympeg4.h	头文件	

		PlayCtrl.lib	LIB 库文件	
		PlayCtrl.dll	DLL 库文件	
		AudioRender.dll	DLL 库文件	
		SuperRender.dll	DLL 库文件	
硬解码库		DsSdk.dll	DLL 库文件	HCNetSDKCom 文件夹

表 1.2 Linux SDK 组件

网络通讯库	外部接口	HCNetSDK.h	头文件	
		libhcnetsdk.so	SO 库文件	
	核心组件	libHCCore.so	SO 库文件	
组件库	设备配置核心组件	libHCCoreDevCfg.so	SO 库文件	HCNetSDKCom 文件夹
	预览组件	libHCPreview.so	SO 库文件	HCNetSDKCom 文件夹
	回放组件	libHCPlayBack.so	SO 库文件	HCNetSDKCom 文件夹
	语音组件	libHCVoiceTalk.so	SO 库文件	HCNetSDKCom 文件夹
	报警组件	libHCAalarm.so	SO 库文件	HCNetSDKCom 文件夹
	显示组件	libHCDisplay.so	SO 库文件	HCNetSDKCom 文件夹
	行业应用管理配置组件	libHCIndustry.so	SO 库文件	HCNetSDKCom 文件夹
	维护管理配置组件	libHCGeneralCfgMgr.so	SO 库文件	HCNetSDKCom 文件夹
hpr 库		libhpr.so	SO 库文件	
RTSP 通讯库		libStreamTransClient.so	SO 库文件	HCNetSDKCom 文件夹
转封装库		libSystemTransform.so	SO 库文件	HCNetSDKCom 文件夹
字符转码库		libiconv2.so	SO 库文件	HCNetSDKCom 文件夹
软解码库		LinuxPlayM4.h	头文件	
		PlayM4.h	头文件	
		libMPCtrl.so	SO 库文件	
		libPlayCtrl.so	SO 库文件	

本版本的设备网络 SDK 开发包中包含以上各个组件, **HCNetSDK.dll、HCCore.dll 必须加载**(对于 Linux SDK, 即 libhcnetsdk.so、libHCCore.so), 其他组件, 用户可以根据需要选择其中的一部分或者全部, 以下将对各个组件在 SDK 中的作用和使用条件分别说明。

- **网络通讯库**是设备网络 SDK 的主体, 主要用于网络客户端与各类产品之间的通讯交互, 负责远程功能调控, 远程参数配置及码流数据的获取和处理等。设备网络 SDK V5.0 针对产品应用业务进行细化, 对之前版本的 SDK 的功能模块进行组件化, 其中外部接口 (HCNetSDK.dll) 仍然保持和设备网络 SDK V4.x 版本保存一致(向下兼容), 其他单独的业务功能 (预览、回放等) 可以加载单独的模块组件, 多个业务功能也可以组合使用。**更新 SDK 时, HCNetSDK.dll、HCCore.dll 以及 HCNetSDKCom 文件夹下的功能组件库文件都需要更新加载, 且 HCNetSDKCom 文件夹名不能修改。**

- **hpr 库**是网络通讯库的依赖库，Linux SDK 使用时和网络通讯库同时加载。
- **RTSP 通讯库**是支持 RTSP 传输协议的网络库。当需要对支持 RTSP 协议的产品进行取流等操作时就必须加载该项组件。
- **转封装库**的功能可以分为两种：一种是将标准码流转换成采用我们公司封装格式的码流。当用户需要对支持 RTSP 协议的产品捕获采用本公司封装格式的码流数据时（即当设置 NET_DVR_RealPlay_V40 接口中的回调函数捕获数据或者调用 NET_DVR_SetRealDataCallBack 接口捕获数据时）必须加载该组件。另一种功能是将标准码流转换成其他格式的封装，如 3GPP、PS 等。例如，当用户需要对支持 RTSP 协议的产品实时捕获指定封装格式的码流数据（对应的 SDK 接口为 NET_DVR_SaveRealData）时必须加载该项组件。
- **语音对讲库**用于语音对讲时通过声卡采集数据并按照指定的编码格式编码码流或者解码播放音频码流数据（不带封装格式的码流数据）。V4.2.2.5 及以前版本 SDK 均采用 windows API 实现相关功能。之后版本默认使用语音对讲库的方式，通过接口 NET_DVR_SetSDKLocalCfg 可以选择之前的 windows API 模式。OpenAL32.dll 为依赖库，语音对讲库模式下必须加载。**Windows64 位或者 Linux 系统下无语音对讲功能。**
- **模拟能力集**：如果需要获取设备能力集（NET_DVR_GetDeviceAbility），建议调用 NET_DVR_SetSDKLocalCfg 启用模拟能力集，此时需要加载 LocalXml.zip（要求和网络通讯库放在同一个目录下）。
- **字符转换库**：电脑字符集和设备字符集不一致时，SDK 内部需要进行字符编码转换，SDK 默认使用 libiconv 库进行类型转换。如果用户不想使用 libiconv 编码库，可以调用 NET_DVR_SetSDKLocalCfg(类型：NET_SDK_LOCAL_CFG_TYPE_BYTE_ENCODE)设置字符转码回调函数，将用户自己的字符编码接口告知 SDK，然后 SDK 将使用用户提供的字符编码接口进行字符串处理。
- **软解码库**主要用于对实时码流数据进行解码显示（实现预览功能）和对录像文件进行回放解码等。用户如果需要在 SDK 内部进行对实时流和录像码流播放显示时（即 NET_DVR_RealPlay_V40 接口的第二个结构体参数的播放句柄设置成有效句柄时）必须加载该组件，而如果用户仅需要用网络通讯库捕获到数据后再外部自行处理就不需要加载该组件，这种情况下用户在外部自行解码将更灵活，可参见软解码库函数说明《播放器 SDK 编程指南》。**Linux 64 位系统不支持软解码功能，预览、回放等窗口句柄传空，仅支持只取流不解码。**

硬解码库需在配备硬解码卡（MD 卡）的前提下使用，通过解码卡的解码与输出功能实现实时流的解码显示及向监视器上矩阵输出的功能。**Windows64 位或者 Linux 系统下无该硬解码库。**对于智能交通摄像机，不需要使用该库文件。

2 函数说明

2.1 SDK 初始化

2.1.1 初始化 SDK `NET_DVR_Init`

函数: `BOOL NET_DVR_Init()`
参数: 无
返回值: `TRUE` 表示成功, `FALSE` 表示失败。
说明: 调用设备网络 SDK 其他函数的前提。

[返回目录](#)

2.1.2 释放 SDK 资源 `NET_DVR_Cleanup`

函数: `BOOL NET_DVR_Cleanup()`
参数: 无
返回值: `TRUE` 表示成功, `FALSE` 表示失败。
说明: 在结束之前最后调用。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

[返回目录](#)

2.2 SDK 本地功能

SDK 本地参数配置

2.2.1 获取 SDK 本地参数 `NET_DVR_GetSDKLocalCfg`

函数: `BOOL NET_DVR_GetSDKLocalCfg(NET_SDK_LOCAL_CFG_TYPE enumType, void *lpOutBuff)`
参数: `[in] enumType` 配置类型, 不同的取值对应不同的 SDK 参数, 详见表 2.1
`[out] lpOutBuff` 输出参数, 不同的配置类型, 输出参数对应不同的结构, 详见表 2.1
返回值: `TRUE` 表示成功, `FALSE` 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。
说明:

表 2.1 本地参数类型

enumType 宏定义	类型值	含义	IpOutBuff 对应结构体
NET_SDK_LOCAL_CFG_TYPE_TCP_PORT_BIND	0	本地 TCP 端口绑定配置	NET_DVR_LOCAL_TCP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_UDP_PORT_BIND	1	本地 UDP 端口绑定配置	NET_DVR_LOCAL_UDP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_MEM_POOL	2	内存池本地配置	NET_DVR_LOCAL_MEM_POOL_CFG
NET_SDK_LOCAL_CFG_TYPE_MODULE_RECV_TIMEOUT	3	按模块配置超时时间	NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG
NET_SDK_LOCAL_CFG_TYPE_ABILITY_PARSE	4	是否使用能力集解析库	NET_DVR_LOCAL_ABILITY_PARSE_CFG
NET_SDK_LOCAL_CFG_TYPE_TALK_MODE	5	对讲模式配置	NET_DVR_LOCAL_TALK_MODE_CFG
NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV	10	心跳交互间隔时间配置	NET_DVR_LOCAL_CHECK_DEV

[返回目录](#)

2.2.2 设置 SDK 本地参数 `NET_DVR_SetSDKLocalCfg`

函数: `BOOL NET_DVR_SetSDKLocalCfg(NET_SDK_LOCAL_CFG_TYPE enumType, void* const lpInBuff)`

参数: [in] enumType 配置类型, 不同的取值对应不同的 SDK 参数, 详见表 2.2

[in] lpInBuff 输入参数, 不同的配置类型, 输出参数对应不同的结构, 详见表

2.2

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

表 2.2 本地参数类型

enumType 宏定义	类型值	含义	lpInBuff 对应结构体
NET_SDK_LOCAL_CFG_TYPE_TCP_PORT_BIND	0	本地 TCP 端口绑定配置	NET_DVR_LOCAL_TCP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_UDP_PORT_BIND	1	本地 UDP 端口绑定配置	NET_DVR_LOCAL_UDP_PORT_BIND_CFG
NET_SDK_LOCAL_CFG_TYPE_MEM_POOL	2	内存池本地配置	NET_DVR_LOCAL_MEM_POOL_CFG
NET_SDK_LOCAL_CFG_TYPE_MODULE_RECV_TIMEOUT	3	按模块配置超时时间	NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG
NET_SDK_LOCAL_CFG_TYPE_ABILITY_PARSE	4	是否使用能力集解析库	NET_DVR_LOCAL_ABILITY_PARSE_CFG
NET_SDK_LOCAL_CFG_TYPE_TALK_MODE	5	对讲模式配置	NET_DVR_LOCAL_TALK_MODE_CFG
NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV	10	心跳交互间隔时间配置	NET_DVR_LOCAL_CHECK_DEV
NET_SDK_LOCAL_CFG_TYPE_CHAR_ENCODE	13	配置字符编码相关处理回调	NET_DVR_LOCAL_BYTE_ENCODE_CONVERT

[返回目录](#)

连接和接收超时时间及重连设置

2.2.3 设置网络连接超时时间和连接尝试次数 **NET_DVR_SetConnectTime**

函数: BOOL NET_DVR_SetConnectTime(DWORD dwWaitTime,DWORD dwTryTime)

参数: [in] dwWaitTime 超时时间, 单位毫秒, 取值范围[300,75000], 实际最大超时时间因系统的 connect 超时时间而不同。

[in] dwTryTimes 连接尝试次数 (保留)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: **SDK 默认建立连接的超时时间为 3 秒。** SDK4.0 及以后版本中当设置的超时时间超过或低于限制的值时接口不返回失败, 将取最接近的上下限限制值作为实际的超时时间。

[返回目录](#)

2.2.4 设置重连功能 **NET_DVR_SetReconnect**

函数: BOOL NET_DVR_SetReconnect (DWORD dwInterval,BOOL bEnableRecon)

参数: [in] dwInterval 重连间隔, 单位:毫秒

[in] bEnableRecon 是否重连, 0-不重连, 1-重连, 参数默认值为 1

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该接口可以同时控制预览、透明通道和布防的重连功能。不调用该接口时, SDK 默认启动预览、透明通道和布防的重连功能, 重连时间间隔为 5 秒。

[返回目录](#)

2.2.5 设置接收超时时间 **NET_DVR_SetRecvTimeOut**

函数: BOOL NET_DVR_SetRecvTimeOut(DWORD nRecvTimeOut)

参数: [in] nRecvTimeOut 接收超时时间, 单位毫秒, 默认为 5000, 最小为 3000 毫秒

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该接口用于设置接收超时时间, 例如预览接收实时流数据、回放下载接收录像数据、报警接收报警信息等接收超时时间。

[返回目录](#)

多网卡绑定

2.2.6 获取所有 IP, 用于支持多网卡接口 **NET_DVR_GetLocalIP**

函数: BOOL NET_DVR_GetLocalIP(char strIP[16][16], DWORD *pValidNum, BOOL *pEnableBind)

参 数: [out] strIP 存放 IP 的缓冲区, 不能为空
 [out] pValidNum 所有有效 IP 的数量
 [out] pEnableBind 是否绑定

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口获取客户端本地多网卡的所有 IP 地址, 可以通过接口 NET_DVR_SetValidIP 选择要使用的 IP 地址。

[返回目录](#)

2.2.7 设置 IP 绑定 **NET_DVR_SetValidIP**

函 数: BOOL NET_DVR_SetValidIP(DWORD dwIPIndex, BOOL bEnableBind)

参 数: [in] dwIPIndex 选择使用的 IP 下标, 由 NET_DVR_GetLocalIP 获取
 [in] bEnableBind 是否绑定

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

SDK 版本、状态和能力

2.2.8 获取 SDK 的版本号和 build 信息 **NET_DVR_GetSDKBuildVersion**

函 数: DWORD NET_DVR_GetSDKBuildVersion()

参 数:

返回值: 获取 SDK 的版本号和 build 信息。

说 明: SDK 的版本号和 build 信息。2 个高字节表示版本号 : 25~32 位表示主版本号, 17~24 位表示次版本号; 2 个低字节表示 build 信息。如 0x03000101: 表示版本号为 3.0, build 号是 0101。

[返回目录](#)

2.2.9 获取当前 SDK 的状态信息 **NET_DVR_GetSDKState**

函 数: BOOL NET_DVR_GetSDKState(LPNET_DVR_SDKSTATE pSDKState);

参 数: [out] pSDKState 状态信息结构, 请参见结构体: [NET_DVR_SDKSTATE](#)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

2.2.10 获取当前 SDK 的功能信息 **NET_DVR_GetSDKAbility**

函 数: BOOL NET_DVR_GetSDKAbility(LPNET_DVR_SDKABL pSDKAbI)

参 数: [out] pSDKAbI 功能信息, 请参见结构体: [NET_DVR_SDKABL](#)
 返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

SDK 启用写日志

2.2.11 启用写日志文件 [NET_DVR_SetLogToFile](#)

函 数: BOOL NET_DVR_SetLogToFile(DWORD bLogEnable, char* strLogDir, BOOL bAutoDel)

参 数: [in] bLogEnable 日志的等级 (默认为 0):
 0-表示关闭日志,
 1-表示只输出 ERROR 错误日志,
 2-输出 ERROR 错误信息和 DEBUG 调试信息,
 3-输出 ERROR 错误信息、DEBUG 调试信息和 INFO 普通信息等所有信息
 [in] strLogDir 日志文件的路径, windows 默认值为"C:\\SdkLog\\"; linux 默认值
 "/home/sdklog/"
 [in] bAutoDel 是否删除超出的文件数, 默认值为 TRUE

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 日志文件路径必须是绝对路径, 且以"\"结尾, 例如"C:\\SdkLog\\", 建议用户先手动创建文件。若未指定文件路径, 则采用默认路径"C:\\SdkLog\\"。可多次调用该接口创建新的日志文件, 同时最多支持创建 10 个文件, 当设置了删除超出的文件时 (即 bAutoDel 为 TRUE), 那么将会自动删除超出的文件。更改目录时到下一次写文件时才会使用新的目录写文件。

[返回目录](#)

异常消息回调

2.2.12 注册接收异常、重连等消息的窗口句柄或回调函数

[NET_DVR_SetExceptionCallback_V30](#)

函 数: [Windows](#) 系统下:

BOOL NET_DVR_SetExceptionCallBack_V30 (UINT nMessage, HWND hWnd, fExceptionCallBack
 cbExceptionCallBack, void* pUser)

[Linux](#) 系统下:

BOOL NET_DVR_SetExceptionCallBack_V30(UINT nMessage, void* hWnd, fExceptionCallBack
 cbExceptionCallBack, void* pUser)

参 数: [in] nMessage 消息, Linux 下该参数保留

[in]hWnd	接收异常消息的窗口句柄，Linux 下该参数保留
[in]cbExceptionCallBack	接收异常消息的回调函数，回调当前异常的相关信息
[in]pUser	用户数据

```
typedef void(CALLBACK* fExceptionCallBack)(DWORD dwType, LONG IUserID, LONG IHandle, void *pUser)
```

[out]dwType	异常或重连等消息的类型，详见表 2.3
[out]IUserID	登录 ID
[out]IHandle	出现异常的相应类型的句柄
[out]pUser	用户数据

表 2.3 异常消息类型

dwType 宏定义	宏定义值	含义
EXCEPTION_EXCHANGE	0x8000	用户交互时异常（注册心跳超时，心跳间隔为 2 分钟）
EXCEPTION_AUDIOEXCHANGE	0x8001	语音对讲异常
EXCEPTION_ALARM	0x8002	报警异常
EXCEPTION_PREVIEW	0x8003	网络预览异常
EXCEPTION_SERIAL	0x8004	透明通道异常
EXCEPTION_RECONNECT	0x8005	预览时重连
EXCEPTION_ALARMRECONNECT	0x8006	报警时重连
EXCEPTION_SERIALRECONNECT	0x8007	透明通道重连
SERIAL_RECONNECTSUCCESS	0x8008	透明通道重连成功
EXCEPTION_PLAYBACK	0x8010	回放异常
EXCEPTION_DISKFMT	0x8011	硬盘格式化
PREVIEW_RECONNECTSUCCESS	0x8015	预览时重连成功
ALARM_RECONNECTSUCCESS	0x8016	报警时重连成功
RESUME_EXCHANGE	0x8017	用户交互恢复
NETWORK_FLOWTEST_EXCEPTION	0x8018	网络流量检测异常

返回值：TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：Windows 下该函数的 hWnd 和 cbExceptionCallBack 不能同时为 NULL，Linux 下 cbExceptionCallBack 不能设置为 NULL，否则将接收不到异常消息。

如果此结构是以回调方式反馈异常消息，那么应用程序中的异常回调函数实现如下，该函数中的参数 dwType 表示异常消息类型（见上表）；IHandle 表示发生异常的相应类型的句柄。

示例代码：

```
//注册接收异常消息的回调函数
NET_DVR_SetExceptionCallBack_V30(WM_NULL, NULL, g_ExceptionCallBack, NULL);

//接收异常消息的回调函数的外部实现
void CALLBACK g_ExceptionCallBack(DWORD dwType, LONG IUserID, LONG IHandle, void *pUser)
{
    char tempbuf[256];
```

```

ZeroMemory(tempbuf,256);
switch(dwType)
{
case EXCEPTION_AUDIOEXCHANGE:           //语音对讲时网络异常
    sprintf(tempbuf,"语音对讲时网络异常!!!");
    TRACE("%s",tempbuf);
    //TODO: 关闭语音对讲
    break;
case EXCEPTION_ALARM:                   //报警上传时网络异常
    sprintf(tempbuf,"报警上传时网络异常!!!");
    TRACE("%s",tempbuf);
    //TODO: 关闭报警上传
    break;
case EXCEPTION_PREVIEW:                 //网络预览时异常
    sprintf(tempbuf,"网络预览时网络异常!!!");
    TRACE("%s",tempbuf);
    //TODO: 关闭网络预览
    break;
case EXCEPTION_RECONNECT:               //预览时重连
    break;
default:
    break;
}
}

```

[返回目录](#)

获取错误信息

2.2.13 返回最后操作的错误码 **NET_DVR_GetLastError**

函数: `DWORD NET_DVR_GetLastError()`

参数:

返回值: 返回最后操作的错误码。详见[错误码宏定义](#)

说明: 返回值为错误码。错误码主要分为网络通讯库错误码、RTSP 通讯库错误码和软硬解库错误码。

[返回目录](#)

2.2.14 返回最后操作的错误码信息 **NET_DVR_GetErrorMsg**

函数: `char* NET_DVR_GetErrorMsg(LONG *pErrorNo)`

参数: `[out] pErrorNo` 错误码数值的指针

返回值: 返回值为错误码信息的指针。错误码主要分为网络通讯库错误码、RTSP 通讯库错误码和软硬解库错误码。详见[错误码宏定义](#)

说 明:

[返回目录](#)

2.3 用户注册

2.3.1 激活设备 **NET_DVR_ActivateDevice**

函 数: BOOL NET_DVR_ActivateDevice(char* sDVRIP, WORD wDVRPort, LPNET_DVR_ACTIVATECFG lpActivateCfg)

参 数: [in]sDVRIP 设备 IP 地址
 [in]wDVRPort 设备端口, 默认: 8000
 [in]lpActivateCfg 激活参数, 包括激活使用的初始密码, 详见结构体:
[NET_DVR_ACTIVATECFG](#)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 出厂设备需要先激活, 然后再使用激活使用的初始密码登录设备。

[返回目录](#)

2.3.2 通过解析服务器, 获取设备的动态 IP 地址和端口号

NET_DVR_GetDVRIPByResolveSvr_EX

函 数: BOOL NET_DVR_GetDVRIPByResolveSvr_EX (char* sServerIP, WORD wServerPort, BYTE* sDVRName, WORD wDVRNameLen, BYTE* sDVRSerialNumber, WORD wDVRSerialLen, char* sGetIP, DWORD* dwPort)

参 数: [in]sServerIP 解析服务器的 IP 地址
 [in]wServerPort 解析服务器的端口号, IP Server 解析服务器端口号为 7071, HiDDNS 服务器的端口号为 80
 [in]sDVRName 设备名称
 [in]wDVRNameLen 设备名称的长度
 [in]sDVRSerialNumber 设备的序列号
 [in]wDVRSerialLen 设备序列号的长度
 [out]sGetIP 获取到的设备 IP 地址指针
 [out]dwPort 获取到的设备端口号指针

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口中的设备名称和设备序列号不能同时为空。通过设备域名或者序列号解析出设备当前 IP 地址和端口, 然后调用 [NET_DVR_Login_V40](#) 登录设备。支持的解析服务器有 IPServer 和 hiDDNS。

[返回目录](#)

2.3.3 用户注册设备 **NET_DVR_Login_V40**

- 函 数:** LONG NET_DVR_Login_V40(LPNET_DVR_USER_LOGIN_INFO pLoginInfo, LPNET_DVR_DEVICEINFO_V40 lpDeviceInfo)
- 参 数:** [in]pLoginInfo 登录参数, 包括设备地址、登录用户、密码等, 详见结构体: [NET_DVR_USER_LOGIN_INFO](#)
[out]lpDeviceInfo 设备信息(同步登录即 pLoginInfo 中 bUseAsynLogin 为 0 时有效), 详见结构体: [NET_DVR_DEVICEINFO_V40](#)
- 返回值:** 异步登录的状态、用户 ID 和设备信息通过 [NET_DVR_USER_LOGIN_INFO](#) 结构体中设置的回调函数(fLoginResultCallBack)返回。对于同步登录, 接口返回-1 表示登录失败, 其他值表示返回的用户 ID 值。用户 ID 具有唯一性, 后续对设备的操作都需要通过此 ID 实现。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。
- 说 明:**
- pLoginInfo 中 bUseAsynLogin 为 0 时登录为同步模式, 接口返回成功即表示登录成功; pLoginInfo 中 bUseAsynLogin 为 1 时登录为异步模式, 登录是否成功在输入参数设置的回调函数中返回。
 - 设备同时最多允许 128 个用户注册。
 - SDK 支持 2048 个注册, 返回 UserID 的取值范围为 0~2047。

[返回目录](#)

2.3.4 用户注销 **NET_DVR_Logout**

- 函 数:** BOOL NET_DVR_Logout(LONG IUserID)
- 参 数:** [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
- 返回值:** TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。
- 说 明:** 建议使用此接口实现注销功能。

[返回目录](#)

2.4 获取设备能力集

2.4.1 获取设备能力集 **NET_DVR_GetDeviceAbility**

- 函 数:** BOOL NET_DVR_GetDeviceAbility(LONG IUserID, DWORD dwAbilityType, char* pInBuf, DWORD dwInLength, char* pOutBuf, DWORD dwOutLength)
- 参 数:** [in] IUserID NET_DVR_Login_V40 的返回值
[in] dwAbilityType 能力类型, 具体定义见表 2.4
[in] pInBuf 输入缓冲区指针, 详见表 2.5
[in] dwInLength 输入缓冲区的长度
[out] pOutBuf 输出缓冲区指针, 详见表 2.5
[in] dwOutLength 接收数据的缓冲区的长度

NET_DVR_MakeKeyFrame 或者 [NET_DVR_MakeKeyFrameSub](#) 实现重置 I 帧。

[返回目录](#)

2.5.2 子码流动态产生一个关键帧 NET_DVR_MakeKeyFrameSub

函数: BOOL NET_DVR_MakeKeyFrameSub(LONG IUserID, DWORD IChannel)

参数: [in] IUserID NET_DVR_Login_V40 的返回值
[in] IChannel 通道号

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 此接口用于重置 I 帧, 根据设置的预览参数([NET_DVR_CLIENTINFO](#))为主码流或者子码流分别调用 [NET_DVR_MakeKeyFrame](#) 或者该接口实现重置 I 帧。

[返回目录](#)

2.5.3 实时预览 NET_DVR_RealPlay_V40

函数: LONG NET_DVR_RealPlay_V40(LONG IUserID, LPNET_DVR_PREVIEWINFO lpPreviewInfo, REALDATACALLBACK fRealDataCallBack_V30, void *pUser)

参数: [in] IUserID NET_DVR_Login_V40 的返回值
[in] lpPreviewInfo 预览参数, 包括码流类型、取流协议、通道号、预览窗口句柄等, 详见结构体: [NET_DVR_PREVIEWINFO](#)
[in] fRealDataCallBack_V30 码流数据回调函数
[in] pUser 用户数据

```
typedef void(CALLBACK *REALDATACALLBACK) (LONG IRealHandle, DWORD dwDataType, BYTE *pBuffer, DWORD dwBufSize, void *pUser);
```

[out]IRealHandle 当前的预览句柄
[out]dwDataType 数据类型, 详见表 2.6
[out] pBuffer 存放数据的缓冲区指针
[out]dwBufSize 缓冲区大小
[out]pUser 用户数据

表 2.6 码流数据类型

dwDataType 宏定义	宏定义值	含义
NET_DVR_SYSHEAD	1	系统头数据
NET_DVR_STREAMDATA	2	流数据 (包括复合流或音视频分开的视频流数据)
NET_DVR_AUDIOSTREAMDATA	3	音频数据

返回值: -1 表示失败, 其他值作为 NET_DVR_StopRealPlay 等函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

- 该接口预览参数结构中可以设置当前预览操作是否阻塞 (通过 bBlocked 参数设置), 若设为不阻塞, 表示发起与设备的连接就认为连接成功, 如果发生码流接收失败、播放失败等情况以预览异常的方式通知上层。在循环播放的时候可以减短停顿的时间, 与 NET_DVR_RealPlay 处理一致。若设为阻塞, 表示直到播放操作完成才返回成功与否。
- 该接口中的回调函数可以置为空, 这样该函数将不回调码流数据给用户, 不过用户仍可以通

过接口 [NET_DVR_SetRealDataCallBack](#) 或 [NET_DVR_SetStandardDataCallBack](#) 注册捕获码流数据的回调函数以捕获码流数据。

- fRealDataCallBack_V30 回调函数中不能执行可能会占用时间较长的接口或操作，不建议调用该 SDK（HCNetSDK.dll）本身的接口。
- Linux 64 位系统不支持软解码功能，因此需要将窗口句柄传 NULL，设置回调函数，只取流不解码显示。
- 客户端异常离线时，设备端对取流连接的保持时间为 10 秒。

[返回目录](#)

2.5.4 停止预览 [NET_DVR_StopRealPlay](#)

函 数： LONG NET_DVR_StopRealPlay (LONG IRealHandle)

参 数： [in] IRealHandle 预览句柄，NET_DVR_RealPlay_V40 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

2.5.5 获取预览时用来解码和显示的播放库句柄 [NET_DVR_GetRealPlayerIndex](#)

函 数： int NET_DVR_GetRealPlayerIndex(LONG IRealHandle)

参 数： [in] IRealHandle 预览句柄，NET_DVR_RealPlay_V40 的返回值

返回值： -1 表示失败，其他值表示播放句柄。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 用户可以通过返回的句柄自行实现播放库 SDK 提供的其他功能，详见本公司提供的软解码库函数说明《播放器 SDK 编程指南》。例如使用 PlayM4_GetBMP(LONG nPort,.....)、PlayM4_GetJPEG(LONG nPort,.....)这两个接口时，即可实现将当前预览图像以 BMP 或 JPEG 格式抓图保存到内存中： PlayM4_GetBMP(NET_DVR_GetRealPlayerIndex(),.....) PlayM4_GetJPEG(NET_DVR_GetRealPlayerIndex(),.....)

[返回目录](#)

2.6 预览时播放声音控制

2.6.1 设置声音播放模式 [NET_DVR_SetAudioMode](#)

函 数： BOOL NET_DVR_SetAudioMode(DWORD dwMode)

参 数： [in] dwMode 声音播放模式： 1—独占声卡，单路音频模式；
2—共享声卡，多路音频模式

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明： 不调用该接口设置声音播放模式，默认为独占播放。

[返回目录](#)

2.6.2 独占声卡模式下开启声音 **NET_DVR_OpenSound**

函数: BOOL NET_DVR_OpenSound(LONG IRealHandle)

参数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 如果当前是共享模式播放, 调用该接口将返回失败。以独占方式只能打开一路通道播放, 即依次打开多个通道时仅打开最后一路。

[返回目录](#)

2.6.3 独占声卡模式下开启声音 **NET_DVR_CloseSound**

函数: BOOL NET_DVR_CloseSound()

参数: 无

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.6.4 共享声卡模式下开启声音 **NET_DVR_OpenSoundShare**

函数: BOOL NET_DVR_OpenSoundShare(LONG IRealHandle)

参数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.6.5 共享声卡模式下关闭声音 **NET_DVR_CloseSoundShare**

函数: BOOL NET_DVR_CloseSoundShare (LONG IRealHandle)

参数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.6.6 调节播放音量 **NET_DVR_Volume**

函数: BOOL NET_DVR_Volume(LONG IRealHandle,WORD wVolume)

参数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
[in] wVolume 音量, 取值范围[0,0xffff]

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该接口调节的是 PC 的音量。

[返回目录](#)

2.7 实时预览数据捕获

2.7.1 注册实时码流数据回调函数 **NET_DVR_SetRealDataCallBack**

函数: BOOL NET_DVR_SetRealDataCallBack(LONG IRealHandle, fRealDataCallBack cbRealDataCallBack, DWORD dwUser)

参数: [in] IRealHandle 预览句柄, NET_DVR_RealPlay_V40 的返回值
 [in] cbRealDataCallBack 码流数据回调函数
 [in] dwUser 用户数据

```
typedef void(CALLBACK *fRealDataCallBack)(LONG IRealHandle, DWORD dwDataType, BYTE *pBuffer,
DWORD dwBufSize, DWORD dwUser)
```

[out] IRealHandle 当前的预览句柄
 [out] dwDataType 数据类型, 详见表 2.7
 [out] pBuffer 存放数据的缓冲区指针
 [out] dwBufSize 缓冲区大小
 [out] dwUser 用户数据

表 2.7 码流数据类型

dwDataType 宏定义	宏定义值	含义
NET_DVR_SYSHEAD	1	系统头数据
NET_DVR_STREAMDATA	2	流数据 (包括复合流或音视频分开的视频流数据)
NET_DVR_AUDIOSTREAMDATA	3	音频数据

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

- 此函数包括开始和停止用户处理 SDK 捕获的数据, 当回调函数 cbRealDataCallBack 设为非 NULL 值时, 表示回调和处理数据; 当设为 NULL 时表示停止回调和处理数据。回调的第一个包是 40 个字节的文件头, 供后续解码使用, 之后回调的是压缩的码流。回调数据最大为 256K 字节。
- cbRealDataCallBack 回调函数中不能执行可能会占用时间较长的接口或操作, 不建议调用该 SDK (HCNetSDK.dll) 本身的接口。

[返回目录](#)

2.7.2 注册标准 RTP 码流数据回调函数 **NET_DVR_SetStandardDataCallBack**

函数: BOOL NET_DVR_SetStandardDataCallBack(LONG IRealHandle, fStdDataCallBack cbStdDataCallBack, DWORD dwUser)

参 数: [in]IRealHandle 预览句柄, NET_DVR_RealPlay_V40 的返回值
[in]cbStdDataCallBack 标准码流回调函数
[in]dwUser 用户数据

```
typedef void(CALLBACK *fStdDataCallBack)(LONG IRealHandle, DWORD dwDataType, BYTE *pBuffer,
DWORD dwBufSize, DWORD dwUser)
```

[out]IRealHandle 当前的预览句柄
[out]dwDataType 数据类型, 详见表 2.8
[out]pBuffer 存放数据的缓冲区指针
[out]dwBufSize 缓冲区大小
[out]dwUser 用户数据

表 2.8 标准码流数据类型

dwDataType 宏定义	宏定义值	含义
NET_DVR_SYSHEAD	1	系统头数据
NET_DVR_STD_VIDEODATA	4	标准视频流数据
NET_DVR_STD_AUDIODATA	5	标准音频流数据
NET_DVR_PRIVATE_DATA	2 或者 112	私有数据, 包括智能信息叠加等

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

- 说 明:**
- 此函数包括开始和停止用户处理 SDK 捕获的数据, 当回调函数 cbStdDataCallBack 设为非 NULL 值时, 表示回调和处理数据; 当设为 NULL 时表示停止回调和处理数据。回调的第一个包是 40 个字节的文件头, 供后续解码使用, 之后回调的是标准码流 (含 12 字节的 RTP 头)。
 - cbStdDataCallBack 回调函数中不能执行可能会占用时间较长的接口或操作, 不建议调用该 SDK (HCNetSDK.dll) 本身的接口。
 - **此函数仅支持对于支持 RTSP 协议取流的设备标准码流回调。**

[返回目录](#)

2.7.3 捕获数据并保存到指定的文件中 **NET_DVR_SaveRealData**

函 数: BOOL NET_DVR_SaveRealData(LONG IRealHandle, char *sFileName)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
[in] sFileName 文件路径指针

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

- 说 明:** V5.0.3.2 或以后版本, 通过该接口保存录像, 文件最大限制为 1024MB, 大于 1024M 时, SDK 自动新建文件进行保存, 文件开始将 40 字节头自动写入, 文件名命名规则为“在接口传入的文件名基础上增加数字标识(例如: *_1.mp4、*_2.mp4)”。

[返回目录](#)

2.7.4 停止数据捕获 **NET_DVR_StopSaveRealData**

函 数: BOOL NET_DVR_StopSaveRealData(LONG IRealHandle)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
 返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

2.8 预览抓图

2.8.1 设置抓图模式 [NET_DVR_SetCapturePictureMode](#)

函 数: BOOL NET_DVR_SetCapturePictureMode(DWORD dwCaptureMode)

参 数: [in]dwCaptureMode 抓图模式

```
enum tagPDC_PARAM_KEY{
    BMP_MODE    = 0, //BMP 模式
    JPEG_MODE   = 1  //JPEG 模式
}CAPTURE_MODE
```

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 调用该接口设置抓图模式后, NET_DVR_CapturePicture 可抓取相应的图片。

[返回目录](#)

2.8.2 预览时, 单帧数据捕获图片并保存成文件 [NET_DVR_CapturePicture](#)

函 数: BOOL NET_DVR_CapturePicture(LONG IRealHandle, char *sPicFileName)

参 数: [in] IRealHandle NET_DVR_RealPlay_V40 的返回值
 [in] sPicFileName 保存图象的文件路径, 包含文件名。路径长度和操作系统有关, sdk 不做限制, windows 默认路径长度小于等于 256 字节 (包括文件名在内)。

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

- 在调用该接口之前可以调用 [NET_DVR_SetCapturePictureMode](#) 设置抓图模式, 默认为 BMP 模式。如果抓图模式为 BMP 模式, 抓取的是 BMP 图片, 保存路径后缀应为.bmp; 如果抓图模式为 JPEG 模式, 抓取的是 JPEG 图片, 保存路径后缀应为.jpg。
- 调用该接口进行抓图, 实际是播放库解码抓图, 要求在调用 [NET_DVR_RealPlay_V40](#) 等接口时传入非空的播放句柄 (播放库解码显示), 否则时接口会返回失败 (调用次序错误)。
- 智能交通摄像机车牌抓拍请参考 [3.2 报警 \(布防\) 模块流程](#)。

[返回目录](#)

2.9 设备 JPEG 抓图

2.9.1 单帧数据捕获并保存成 JPEG 图片 `NET_DVR_CaptureJPEGPicture`

函数: `BOOL NET_DVR_CaptureJPEGPicture(LONG IUserID, LONG IChannel, LPNET_DVR_JPEGPARA lpJpegPara, char *sPicFileName)`

参数: [in] IUserID NET_DVR_Login_V40 的返回值
 [in] IChannel 子窗口号(组合)
 [in] lpJpegPara JPEG 图像参数, 请参见结构体 [NET_DVR_JPEGPARA](#)
 [in] sPicFileName 保存 JPEG 图的文件路径

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该接口用于设备的单帧数据捕获, 并保存成 JPEG 图片。IPC 设备支持当前视频分辨率的抓取。

[返回目录](#)

2.10 布防、撤防

设置报警等信息上传的回调函数

2.10.1 注册报警回调函数 `NET_DVR_SetDVRMessageCallback_V31`

函数: `BOOL NET_DVR_SetDVRMessageCallback_V31(MSGCallBack_V31 fMessageCallBack, void* pUser)`

参数: [in]fMessageCallBack 报警信息回调函数
 [in]pUser 用户数据

```
typedef BOOL(CALLBACK * MSGCallBack_V31)(LONG ICommand, NET_DVR_ALARMER *pAlarmer, char *pAlarmInfo, DWORD dwBufLen, void *pUser)
```

[out]ICommand 上传的消息类型, 详见表 2.9
 [out]pAlarmer 报警设备信息, 详见 [NET_DVR_ALARMER](#)
 [out]pAlarmInfo 报警信息, 详见表 2.10
 [out]dwBufLen 报警信息缓存大小
 [out]pUser 用户数据

表 2.9 布防报警信息类型

宏定义	宏定义值	含义
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	0x1132	可视对讲事件记录上传
COMM_ALARM_VIDEO_INTERCOM	0x1133	可视对讲报警上传

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该接口中回调函数的第一个参数 (ICommand) 和第三个参数 (pAlarmInfo) 是密切关联的, 其关系如表 2.10 所示。

表 2.10 报警信息结构

消息类型 (ICommand)	上传内容	pAlarmInfo 对应的结构体
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	可视对讲事件记录信息	NET_DVR_VIDEO_INTERCOM_EVENT
COMM_ALARM_VIDEO_INTERCOM	可视对讲报警信息	NET_DVR_VIDEO_INTERCOM_ALARM

[返回目录](#)

布防撤防

2.10.2 建立报警上传通道，获取报警等信息 **NET_DVR_SetupAlarmChan_V41**

函 数: LONG NET_DVR_SetupAlarmChan_V41(LONG IUserID, LPNET_DVR_SETUPALARM_PARAM lpSetupParam)

参 数: [in]IUserID NET_DVR_Login_V40 的返回值
[in]lpSetupParam 报警布防参数，请参见结构体：[NET_DVR_SETUPALARM_PARAM](#)

返回值: -1 表示失败，其他值作为 NET_DVR_CloseAlarmChan_V30 函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明: 启动布防前，需要调用注册回调函数的接口（如 [NET_DVR_SetDVRMessageCallBack_V31](#)）才能获取到上传的报警等信息。

[返回目录](#)

2.10.3 撤销报警上传通道 **NET_DVR_CloseAlarmChan_V30**

函 数: BOOL NET_DVR_CloseAlarmChan_V30(LONG IAlarmHandle)

参 数: [in]IAlarmHandle NET_DVR_SetupAlarmChan_V41 的返回值

返回值: TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明:

[返回目录](#)

2.11 监听报警

2.11.1 启动监听，接收设备主动上传的报警等信息 **NET_DVR_StartListen_V30**

函 数: LONG NET_DVR_StartListen_V30(char *sLocalIP, WORD wLocalPort, MSGCallBack DataCallback, void* pUserData = NULL)

参 数: [in] sLocalIP PC 机本地 IP 地址，可以置为 NULL
[in] wLocalPort PC 本地监听端口号。由用户设置，必须和设备端设置的一致
[in] DataCallback 报警信息回调函数
[in] pUserData 用户数据

```
typedef void(CALLBACK *MSGCallBack)(LONG ICommand,NET_DVR_ALARMER *pAlarmer,char *pAlarmInfo,DWORD dwBufLen,void *pUser)
```

[out] ICommand	上传的消息类型，详见表 2.11
[out] pAlarmer	报警设备信息，详见 NET_DVR_ALARMER
[out] pAlarmInfo	报警信息，详见表 2.12
[out] dwBufLen	报警信息缓存大小
[out] pUser	用户数据

表 2.11 监听报警信息类型

宏定义	宏定义值	含义
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	0x1132	可视对讲事件记录上传
COMM_ALARM_VIDEO_INTERCOM	0x1133	可视对讲报警上传

返回值: -1 表示失败，其他值作为 NET_DVR_CloseAlarmChan_V30 函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明: 该接口中回调函数的第一个参数 (ICommand) 和第三个参数 (pAlarmInfo) 是密切关联的，其关系如表 2.12 所示。

- SDK 最大能支持 512 路监听。
- 要使 PC 能够收到设备主动发过来的报警等信息，必须将设备的网络配置中的远程报警主机地址(struAlarmHostIpAddr)设置成 PC 机的 IP 地址（与接口中的 sLocalIP 参数一致），远程报警主机端口号(wAlarmHostIpPort)设置成 PC 机的监听端口号（与接口中的 wLocalPort 参数一致）。
- **该接口中的回调函数优先级高于其他回调函数，即设置了该接口中的回调函数，其他回调函数将接收不到报警信息。**

表 2.12 报警信息结构

消息类型 (ICommand)	上传内容	pAlarmInfo 对应的结构体
COMM_UPLOAD_VIDEO_INTERCOM_EVENT	可视对讲事件记录信息	NET_DVR_VIDEO_INTERCOM_EVENT
COMM_ALARM_VIDEO_INTERCOM	可视对讲报警信息	NET_DVR_VIDEO_INTERCOM_ALARM

[返回目录](#)

2.11.2 停止监听（支持多线程）NET_DVR_StopListen_V30

函数: BOOL NET_DVR_StopListen_V30(LONG IListenHandle)

参数: [in] IListenHandle 监听句柄，NET_DVR_StartListen_V30 的返回值

返回值: TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明:

[返回目录](#)

2.12 远程参数配置

通用参数配置

2.12.1 获取设备的配置信息 NET_DVR_GetDVRConfig

函数: BOOL NET_DVR_GetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)

- 参 数:**
- [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 2.13
 - [in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
 - [out] lpOutBuffer 接收数据的缓冲指针, 详见表 2.13
 - [in] dwOutBufferSize 接收数据的缓冲长度(以字节为单位), 不能为 0
 - [out] lpBytesReturned 实际收到的数据长度指针, 不能为 NULL
- 返回值:** TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。
- 说 明:** 不同的获取功能对应不同的结构体和命令号, 如表 2.13 所示。
对于智能交通摄像机, 设备只有一个通道, 通道号即为 1。

表 2.13 通用参数获取

dwCommand 宏定义	dwCommand 含义	通道号	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_DEVICECFG_V40	获取设备参数(扩展)	无效	NET_DVR_DEVICECFG_V40	1100
NET_DVR_GET_USERCFG_V30	获取用户参数	无效	NET_DVR_USER_V30	1006
NET_DVR_GET_NETCFG_V30	获取网络参数	无效	NET_DVR_NETCFG_V30	1000
NET_DVR_GET_PICCFG_V30	获取图象参数	通道号	NET_DVR_PICCFG_V30	1002
NET_DVR_GET_COMPRESSCFG_V30	获取压缩参数	通道号	NET_DVR_COMPRESSIONCFG_V30	1040
NET_DVR_GET_NTPCFG	获取网络应用参数(NTP)	无效	NET_DVR_NTPPARA	224
NET_DVR_GET_DECODERCFG_V30	获取 RS485 串口参数	通道号	NET_DVR_DECODERCFG_V30	1042

[返回目录](#)

2.12.2 设置设备的配置信息 **NET_DVR_SetDVRConfig**

- 函 数:** BOOL NET_DVR_SetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)
- 参 数:**
- [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
 - [in] dwCommand 设备配置命令, 详见表 2.14
 - [in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
 - [in] lpInBuffer 输入数据的缓冲指针, 详见表 2.14
 - [in] dwInBufferSize 输入数据的缓冲长度(以字节为单位)
- 返回值:** TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。
- 说 明:** 不同的获取功能对应不同的结构体和命令号, 如表 2.14 所示。
对于智能交通摄像机, 设备只有一个通道, 通道号即为 1。

表 2.14 通用参数设置

dwCommand 宏定义	dwCommand 含义	通道号	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_DEVICECFG_V40	设置设备参数(扩展)	无效	NET_DVR_DEVICECFG_V40	1101
NET_DVR_SET_USERCFG_V30	设置用户参数	无效	NET_DVR_USER_V30	1007
NET_DVR_SET_NETCFG_V30	设置网络参数	无效	NET_DVR_NETCFG_V30	1001

NET_DVR_SET_NTPCFG	设置网络应用参数(NTP)	无效	NET_DVR_NTPPARA	225
NET_DVR_SET_DECODERCFG_V30	设置 RS485 串口参数	通道号	NET_DVR_DECODERCFG_V30	1043

[返回目录](#)

可视对讲参数配置

2.12.3 获取设备的配置信息 **NET_DVR_GetDVRConfig**

函数: BOOL NET_DVR_GetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpOutBuffer, DWORD dwOutBufferSize, LPDWORD lpBytesReturned)

参数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
[in] dwCommand 设备配置命令, 详见表 2.15
[in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
[out] lpOutBuffer 接收数据的缓冲指针, 详见表 2.15
[in] dwOutBufferSize 接收数据的缓冲长度(以字节为单位), 不能为 0
[out] lpBytesReturned 实际收到的数据长度指针, 不能为 NULL

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 不同的获取功能对应不同的结构体和命令号, 如表 2.15 所示。

表 2.15 ITC 参数获取

dwCommand 宏定义	dwCommand 含义	通道号	lpOutBuffer 对应结构体	宏定义值
NET_DVR_GET_VIDEO_INTERCOM_DEVICEID_CFG	获取可视对讲设备编号	无效	NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG	16001
NET_DVR_GET_OPERATION_TIME_CFG	获取操作时间配置	无效	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG	16004
NET_DVR_GET_VIDEO_INTERCOM_RELATEDDEV_CFG	获取关联网络设备参数	无效	NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG	16006
NET_DVR_GET_VIDEOINTERCOM_STREAM	获取可视对讲流通道参数	通道号, 从 1 开始	NET_DVR_VIDEOINTERCOM_STREAM	16022
NET_DVR_GET_AUDIOIN_VOLUME_CFG	获取音频输入音量调节参数	音频输入口号, 从 1 开始	NET_DVR_VOLUME_CFG	6355
NET_DVR_GET_AUDIOOUT_VOLUME_CFG	获取音频输出音量调节参数	音频输出口号, 从 1 开始	NET_DVR_VOLUME_CFG	6369
NET_DVR_GET_ALARMIN_PARAM	获取防区参数	防区号, 从 0 开始	NET_DVR_ALARMIN_PARAM	1183

[返回目录](#)

2.12.4 设置设备的配置信息 **NET_DVR_SetDVRConfig**

函 数: BOOL NET_DVR_SetDVRConfig(LONG IUserID, DWORD dwCommand, LONG IChannel, LPVOID lpInBuffer, DWORD dwInBufferSize)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值
 [in] dwCommand 设备配置命令, 详见表 2.16
 [in] IChannel 通道号, 如果命令不需要通道号, 该参数无效, 置为 0xFFFFFFFF 即可
 [in] lpInBuffer 输入数据的缓冲指针, 详见表 2.16
 [in] dwInBufferSize 输入数据的缓冲长度(以字节为单位)

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 不同的获取功能对应不同的结构体和命令号, 如表 2.16 所示。

表 2.16 ITC 参数设置

dwCommand 宏定义	dwCommand 含义	通道号	lpInBuffer 对应结构体	宏定义值
NET_DVR_SET_VIDEO_INTERCOM_DEVICEID_CFG	设置可视对讲设备编号	无效	NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG	16002
NET_DVR_SET_OPERATION_TIME_CFG	设置操作时间配置	无效	NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG	16005
NET_DVR_SET_PRIVILEGE_PASSWORD	设置权限密码配置信息	无效	NET_DVR_PRIVILEGE_PASSWORD_CFG	16003
NET_DVR_SET_VIDEO_INTERCOM_RELATEDDEV_CFG	设置关联网络设备参数	无效	NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG	16007
NET_DVR_SET_VIDEOINTERCOM_STREAM	设置可视对讲流通道参数	通道号, 从 1 开始	NET_DVR_VIDEOINTERCOM_STREAM	16023
NET_DVR_SET_AUDIOIN_VOLUME_CFG	设置音频输入音量调节参数	音频输入入口号, 从 1 开始	NET_DVR_VOLUME_CFG	6356
NET_DVR_SET_AUDIOOUT_VOLUME_CFG	设置音频输出音量调节参数	音频输出出口号, 从 1 开始	NET_DVR_VOLUME_CFG	6370
NET_DVR_SET_ALARMIN_PARAM	设置防区参数	防区号, 从 0 开始	NET_DVR_ALARMIN_PARAM	1181

[返回目录](#)

2.12.5 批量获取配置信息 **NET_DVR_GetDeviceConfig**

函 数: BOOL NET_DVR_GetDeviceConfig(LONG IUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpOutBuffer, DWORD dwOutBufferSize)

参 数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值

[in] dwCommand	设备配置命令，详见表 2.17
[in] dwCount	一次要获取配置参数的个数，0 和 1 都表示 1 个信息，2 表示 2 个信息，最大 64 个
[in] lpInBuffer	配置条件缓冲区，详见表 2.18
[in] dwInBufferSize	缓冲区长度
[out] lpStatusList	错误信息列表，和要查询的配置一一对应，例如 lpStatusList[2]就对应 lpInBuffer[2]，由用户分配内存，每个错误信息为 4 个字节，参数值：0- 成功，大于 0-失败
[out] lpOutBuffer	设备返回的参数内容（详见表 2.18），和要查询的监控点一一对应。如果某个监控点对应的 lpStatusList 信息为大于 0 值，对应 lpOutBuffer 的内容就是无效的
[in] dwOutBufferSize	输出缓冲区大小

表 2.17 ITC 参数批量获取命令

dwCommand 宏定义	含义	宏定义值
NET_DVR_GET_STREAM_SRC_INFO	获取流的来源信息	6017
NET_DVR_GET_ALLSTREAM_SRC_INFO	获取所有流信息（全部获取）	9105

返回值：TRUE 表示成功，但不代表每一个配置都成功，哪一个成功，对应查看 lpStatusList[n]值；FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：该接口是带有发送数据的批量获取配置信息的通用接口。lpInBuffer 指定需要获取哪 dwCount 个信息，lpOutBuffer 保存获取得到的 dwCount 个相应配置信息。不同的获取功能对应不同的结构体和命令号，如表 2.18 所示。全部获取时 dwCount 置为 0xffffffff，lpInBuffer 置为 NULL，dwInBufferSize 置为 0，lpStatusList 置为 NULL，lpOutBuffer 前 4 个字节为返回的个数。如果设置的 lpOutBuffer 缓冲区不足，仅返回部分信息，可以根据返回的个数（前 4 字节的值）重新获取。

表 2.18 批量获取 ITC 参数

dwCommand 宏定义	lpInBuffer 对应结构体	lpOutBuffer 对应结构体
NET_DVR_GET_STREAM_SRC_INFO	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_STREAM_SRC_INFO
NET_DVR_GET_ALLSTREAM_SRC_INFO	NULL	4 字节个数(N) + N 个 NET_DVR_STREAM_SRC_CFG

[返回目录](#)

2.12.6 批量设置配置信息 [NET_DVR_SetDeviceConfig](#)

函数：BOOL NET_DVR_SetDeviceConfig(LONG lUserID, DWORD dwCommand, DWORD dwCount, LPVOID lpInBuffer, DWORD dwInBufferSize, LPVOID lpStatusList, LPVOID lpInParamBuffer, DWORD dwInParamBufferSize)

参数：

[in] lUserID	用户 ID 号，NET_DVR_Login_V40 的返回值
[in] dwCommand	设备配置命令，详见表 2.19
[in] dwCount	一次要设置的配置参数个数，0 和 1 都表示 1 个，2 表示 2 个，最大 64 个
[in] lpInBuffer	配置条件缓冲区，详见表 2.20
[in] dwInBufferSize	缓冲区长度
[out] lpStatusList	错误信息列表，和要设置的配置一一对应，例如 lpStatusList[2]就

	对应 lpInBuffer[2]，由用户分配内存，每个错误信息为 4 个字节，参数值：0- 成功，大于 0-失败
[in] lpInParameter	需要设置给设备的参数内容（详见表 2.20），和 lpInBuffer 一一对应。如果某个配置对应的 lpStatusList 信息为大于 0 值，表示对应的 lpInBuffer 设置失败，为 0 则设置成功
[in] dwInParameterSize	设置内容缓冲区大小

表 2.19 ITC 参数批量设置命令

dwCommand 宏定义	含义	宏定义值
NET_DVR_SET_STREAM_SRC_INFO	设置流的来源信息	6018

返回值：TRUE 表示成功，但不代表每一个配置都成功，哪一个成功，对应查看 lpStatusList[n]值；FALSE 表示全部失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：该接口是带有发送数据的批量设置子设备配置信息的通用接口，lpInBuffer 指定需要设置哪 dwCount 个，lpInParameter 是设置 dwCount 个配置的参数信息。不同的获取功能对应不同的结构体和命令号，如表 2.20 所示。

表 2.20 批量设置 ITC 参数

dwCommand 宏定义	lpInBuffer 对应结构体	lpInParameter 对应结构体
NET_DVR_SET_STREAM_SRC_INFO	dwCount 个 NET_DVR_STREAM_INFO	dwCount 个 NET_DVR_STREAM_SRC_INFO

[返回目录](#)

长连接参数配置

2.12.7 启动长连接远程配置 **NET_DVR_StartRemoteConfig**

函数：LONG NET_DVR_StartRemoteConfig(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferLen, fRemoteConfigCallback cbStateCallback, LPVOID pUserData)

参数：

- [in] IUserID 用户 ID 号，NET_DVR_Login_V30 返回值
- [in] dwCommand 配置命令，详见表 2.21
- [in] lpInBuffer 输入参数，具体内容跟配置命令相关，详见表 2.21
- [in] dwInBufferLen 输入缓冲的大小
- [in] cbStateCallback 状态回调函数
- [in] pUserData 用户数据

```
typedef void(CALLBACK *fRemoteConfigCallback)(DWORD dwType, void *lpBuffer, DWORD dwBufLen, void *pUserData)
```

- [out] dwType 配置状态，具体取值参见表 2.22
- [out] lpBuffer 存放数据的缓冲区指针，具体取值参见表 2.22
- [out] dwBufLen 缓冲区大小
- [out] pUserData 用户数据

返回值：TRUE 表示成功，FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说明：不同的功能对应不同的命令号(dwCommand)，同时 lpInBuffer 对应不同的结构体，如表 2.21 所示。调用该接口启动长连接远程配置后，还需要调用其他接口获取或者设置相关参数，如表 2.23 所示。

表 2.21 长连接配置

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer	回调函数
NET_DVR_VIDEO_CALL_SIGNAL_PROCESS	16032	可视对讲信令处理	NET_DVR_VIDEO_CALL_COND	信息数据

表 2.22 回调状态信息

dwType	含义	lpBuffer 对应内容
NET_SDK_CALLBACK_TYPE_STATUS	状态值	typedef enum{ NET_SDK_CALLBACK_STATUS_SUCCESS = 1000, //成功 NET_SDK_CALLBACK_STATUS_PROCESSING, //处理中 NET_SDK_CALLBACK_STATUS_FAILED //失败 }NET_SDK_CALLBACK_STATUS_NORMAL;
NET_SDK_CALLBACK_TYPE_PROGRESS	进度值	lpBuffer 的值表示进度(DWORD)
NET_SDK_CALLBACK_TYPE_DATA	信息数据	lpBuffer 的值表示信息数据, dwCommand 为 NET_DVR_VIDEO_CALL_SIGNAL_PROCESS 时, 对应结构体: NET_DVR_VIDEO_CALL_PARAM

表 2.23 后续接口调用

dwCommand 宏定义		后续接口调用
NET_DVR_VIDEO_CALL_SIGNAL_PROCESS	可视对讲信令处理	NET_DVR_SendRemoteConfig

[返回目录](#)

2.12.8 发送长连接数据 [NET_DVR_SendRemoteConfig](#)

函数: BOOL NET_DVR_SendRemoteConfig(LONG IHandle, DWORD dwDataType, char *pSendBuf, DWORD dwBufSize)

参数: [in] IHandle 长连接句柄, NET_DVR_StartRemoteConfig 的返回值
[in] dwDataType 数据类型, 详见表 2.24
[in] pSendBuf 保存发送数据的缓冲区, 与 dwDataType 有关, 详见表 2.24
[in] dwBufSize 发送数据的长度

返回值: TRUE 表示成功, FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。

说明: 在调用该接口之前, 必须先调用 [NET_DVR_StartRemoteConfig](#) 获取长连接句柄。不同的数据类型(dwDataType), pSendBuf 对应不同的结构体, 如表 2.24 所示。

表 2.24 长连接发送数据

dwCommand 宏定义	宏定义值	含义	dwDataType 值	pSendBuf 对应结构体
NET_DVR_VIDEO_CALL_SIGNAL_PROCESS	16032	可视对讲信令处理	0	NET_DVR_VIDEO_CALL_PARAM

[返回目录](#)

2.12.9 关闭长连接配置接口所创建的句柄，释放资源

NET_DVR_StopRemoteConfig

函数： BOOL NET_DVR_StopRemoteConfig(LONG IHandle)
参数： [in] IHandle 句柄，NET_DVR_StartRemoteConfig 的返回值
返回值： TRUE 表示成功，FALSE 表示失败。获取错误码调用 [NET_DVR_GetLastError](#)。
说明：

[返回目录](#)

获取设备支持的云台协议

2.12.10 获取设备支持的云台协议 NET_DVR_GetPTZProtocol

函数： BOOL NET_DVR_GetPTZProtocol(LONG IUserID, NET_DVR_PTZCFG *pPtzcfg)
参数： [in] IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
 [out] pPtzcfg 设备的云台协议，请参见 [NET_DVR_PTZCFG](#)
返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。
说明： 在配置前端云台协议时必须调用该接口获取当前设备支持的云台协议。

[返回目录](#)

2.13 远程控制

2.13.1 远程控制 NET_DVR_RemoteControl

函数： BOOL NET_DVR_RemoteControl(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)
参数： [in] IUserID 用户 ID 号，NET_DVR_Login_V30 返回值
 [in] dwCommand 控制命令，详见表 2.25
 [in] lpInBuffer 输入参数，跟控制命令相关，详见列表
 [in] dwInBufferSize 输入参数长度
返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。
说明： 不同的控制功能对应不同的命令号，同时 lpInBuffer 对应不同的结构体，如表 2.25 所示。

表 2.25 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_REMOTECONTROL_OPERATION_AUTH	16010	操作权限验证	NET_DVR_OPERATION_AUTH
NET_DVR_REMOTECONTROL_NOTICE_DATA	16008	公告数据下发	NET_DVR_NOTICE_DATA

[返回目录](#)

2.14 语音对讲

2.14.1 启动语音对讲 **NET_DVR_StartVoiceCom_V30**

函数: LONG NET_DVR_StartVoiceCom_V30(LONG IUserID, DWORD dwVoiceChan, BOOL bNeedCBNoEncData, fVoiceDataCallBack cbVoiceDataCallBack, void* pUser)

参数: [in]IUserID NET_DVR_Login_V40 的返回值
[in]dwVoiceChan 语音通道号, 从 1 开始
[in]bNeedCBNoEncData 需要回调的语音数据类型: 0- 编码后的语音数据, 1- 编码前的 PCM 原始数据
[in]cbVoiceDataCallBack 音频数据回调函数
[in]pUser 用户数据指针

```
typedef void(CALLBACK *fVoiceDataCallBack)(LONG IVoiceComHandle, char *pRecvDataBuffer,
DWORD dwBufSize, BYTE byAudioFlag, void *pUser)
```

[out]IVoiceComHandle NET_DVR_StartVoiceCom_V30 的返回值
[out]pRecvDataBuffer 存放音频数据的缓冲区指针
[out]dwBufSize 音频数据大小
[out]byAudioFlag 音频数据类型: 0- 本地采集的数据, 1- 设备发送过来的语音数据
[out]pUser 用户数据指针

返回值: -1 表示失败, 其他值作为 NET_DVR_StopVoiceCom 等函数的句柄参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: Windows 7 操作系统下, 如果不外接音频设备, 该接口将返回失败。在调用开始语音对讲之前可先配置设备的语音对讲音频编码类型, 即可先调用参数配置中的 NET_DVR_COMPRESSION_AUDIO 结构配置。

当前音频为 G722 编码时, 音频数据的采样频率为 16000, 16 位采样且是单通道的。因此, 音频播放格式应如下定义:

```
const int SAMPLES_PER_SECOND = 16000;
const int CHANNEL = 1;
const int BITS_PER_SAMPLE = 16;
WAVEFORMATEX m_wavFormatEx;
m_wavFormatEx.cbSize = sizeof(m_wavFormatEx);
m_wavFormatEx.nBlockAlign = CHANNEL * BITS_PER_SAMPLE / 8;
m_wavFormatEx.nChannels = CHANNEL;
m_wavFormatEx.nSamplesPerSec = SAMPLES_PER_SECOND;
m_wavFormatEx.wBitsPerSample = BITS_PER_SAMPLE;
m_wavFormatEx.nAvgBytesPerSec = SAMPLES_PER_SECOND*m_wavFormatEx.nBlockAlign
```

当前音频为 G711 或者 G726 编码时, 音频数据的采样频率为 8000, 16 位采样且是单通道的。因此, 音频播放格式应如下定义:

```
const int SAMPLES_PER_SECOND_G711_MU = 8000;
const int CHANNEL = 1;
const int BITS_PER_SAMPLE = 16;
```

```

WAVEFORMATEX m_wavFormatEx;
m_wavFormatEx.cbSize = sizeof(m_wavFormatEx);
m_wavFormatEx.nBlockAlign = CHANNEL * BITS_PER_SAMPLE / 8;
m_wavFormatEx.nChannels = CHANNEL;
m_wavFormatEx.nSamplesPerSec = SAMPLES_PER_SECOND_G711_MU;
m_wavFormatEx.wBitsPerSample = BITS_PER_SAMPLE;
m_wavFormatEx.nAvgBytesPerSec = SAMPLES_PER_SECOND_G711_MU*
m_wavFormatEx.nBlockAlign;

```

[返回目录](#)

2.14.2 设置语音对讲客户端的音量 **NET_DVR_SetVoiceComClientVolume**

函数: `BOOL NET_DVR_SetVoiceComClientVolume(LONG lVoiceComHandle, WORD wVolume)`
参数: `[in]lVoiceComHandle` `NET_DVR_StartVoiceCom_V30` 的返回值
`[in]wVolume` 设置音量, 取值范围[0,0xffff]
返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.14.3 停止语音对讲 **NET_DVR_StopVoiceCom**

函数: `BOOL NET_DVR_StopVoiceCom(LONG lVoiceComHandle)`
参数: `[in]lVoiceComHandle` `NET_DVR_StartVoiceCom_V30` 返回值
返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.15 设备维护管理

在线状态检测

2.15.1 设备在线状态检测 **NET_DVR_RemoteControl**

函数: `BOOL NET_DVR_RemoteControl(LONG lUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)`
参数: `[in]lUserID` 用户 ID 号, `NET_DVR_Login_V40` 的返回值
`[in]dwCommand` 控制命令, 详见表 2.26
`[in]lpInBuffer` 输入参数, 具体内容跟控制命令相关, 详见表 2.26
`[in]dwInBufferSize` 输入参数长度

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 该控制命令用于手动检测设备是否在线, 接口返回 TRUE 表示在线, FALSE 表示与设备通信失败或者返回错误状态。设备在线状态自动巡检功能通过 [NET_DVR_SetSDKLocalCfg](#) (配置类型: NET_SDK_LOCAL_CFG_TYPE_CHECK_DEV) 进行配置。

表 2.26 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_CHECK_USER_STATUS	20005	检测设备是否在线	NULL

[返回目录](#)

远程升级

2.15.2 设置远程升级时网络环境 [NET_DVR_SetNetworkEnvironment](#)

函数: BOOL NET_DVR_SetNetworkEnvironment(DWORD dwEnvironmentLevel)

参数: [in] dwEnvironmentLevel 网络环境级别

```
enum{
    LOCAL_AREA_NETWORK = 0, //局域网环境
    WIDE_AREA_NETWORK   //广域网环境
}
```

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 接口中的网络环境级别参数分为两类,

[LOCAL_AREA_NETWORK](#) 表示局域网环境(网络环境好, 通讯流畅);

[WIDE_AREA_NETWORK](#) 表示广域网环境(网络环境差, 易阻塞)。

在调用远程升级接口之前, 可以通过此接口适应不同的升级环境。

[返回目录](#)

2.15.3 远程升级 [NET_DVR_Upgrade](#)

函数: LONG NET_DVR_Upgrade(LONG IUserID, char *sFileName)

参数: [in] IUserID NET_DVR_Login_V40 的返回值
[in] sFileName 升级的文件路径 (包括文件名)。路径长度和操作系统有关, sdk 不做限制, windows 默认路径长度小于等于 256 字节 (包括文件名在内)。

返回值: -1 表示失败, 其他值作为 NET_DVR_GetUpgradeState 等函数的参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.15.4 获取远程升级的进度 [NET_DVR_GetUpgradeProgress](#)

函数: int NET_DVR_GetUpgradeProgress(LONG IUpgradeHandle)

参 数: [in] IUpgradeHandle NET_DVR_Upgrade 的返回值
 返回值: -1 表示失败, 0~100 表示升级进度。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

2.15.5 获取远程升级的状态 NET_DVR_GetUpgradeState

函 数: int NET_DVR_GetUpgradeState(LONG IUpgradeHandle)
 参 数: [in] IUpgradeHandle NET_DVR_Upgrade 的返回值
 返回值: -1 表示失败, 其他值定义: 1- 升级成功; 2- 正在升级; 3- 升级失败; 4- 网络断开, 状态未知; 5- 升级文件语言版本不匹配。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

2.15.6 关闭远程升级句柄, 释放资源 NET_DVR_CloseUpgradeHandle

函 数: BOOL NET_DVR_CloseUpgradeHandle(LONG IUpgradeHandle)
 参 数: [in] IUpgradeHandle NET_DVR_Upgrade 的返回值
 返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明:

[返回目录](#)

日志查找

2.15.7 查找设备的日志信息 NET_DVR_FindDVRLog_V30

函 数: LONG NET_DVR_FindDVRLog_V30(LONG IUserID, LONG ISelectMode, DWORD dwMajorType, DWORD dwMinorType, LPNET_DVR_TIME lpStartTime, LPNET_DVR_TIME lpStopTime, BOOL bOnlySmart)
 参 数: [in]IUserID NET_DVR_Login_V40 的返回值
 [in]ISelectMode 查询方式: 0- 全部; 1- 按类型; 2- 按时间; 3- 按时间和类型
 [in]dwMajorType 日志主类型 (S.M.A.R.T 搜索时无效), 0 表示全部类型, 其他类型定义请参见[设备日志主类型](#)
 [in]dwMinorType 日志次类型 (S.M.A.R.T 搜索时无效), 0 表示全部类型, 根据不同的主类型的次类型定义请参见[设备日志次类型](#)
 [in]lpStartTime 文件的开始时间, 请参见结构体 [NET_DVR_TIME](#)
 [in]lpStopTime 文件结束时间, 请参见结构体 [NET_DVR_TIME](#)
 [in]bOnlySmart 是否只搜索带 S.M.A.R.T 信息的日志

返回值: -1 表示失败, 其他值作为 NET_DVR_FindNextLog_V30 等函数的参数。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说 明: 该接口如果用于搜索普通日志信息, 一般设备支持 2000 条, 而搜索带 S.M.A.R.T 信息(硬盘运行

日志记录)的日志最大只支持 500 条。通常不需要搜索详细的 S.M.A.R.T 信息时，置 bOnlySmart 为 FALSE 即可完成所有日志信息的搜索。

[返回目录](#)

2.15.8 逐条获取查找到的日志信息 **NET_DVR_FindNextLog_V30**

函数: LONG NET_DVR_FindNextLog_V30(LONG ILogHandle, LPNET_DVR_LOG_V30 lpLogData)

参数: [in] ILogHandle 日志查找句柄, NET_DVR_FindDVRLog_V30 的返回值

[out] lpLogData 保存日志信息的指针, 请参见结构体 [NET_DVR_LOG_V30](#)

返回值: -1 表示失败, 其他值表示当前的获取状态等信息。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明: 在调用该接口获取查找日志之前, 必须先调用 NET_DVR_FindDVRLog_V30 得到当前的查找句柄。

[返回目录](#)

2.15.9 释放查找日志的资源 **NET_DVR_FindLogClose_V30**

函数: BOOL NET_DVR_FindLogClose_V30(LONG ILogHandle)

参数: [in] ILogHandle 日志查找句柄, NET_DVR_FindDVRLog_V30 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

恢复设备默认参数

2.15.10 恢复设备默认参数 **NET_DVR_RestoreConfig**

函数: BOOL NET_DVR_RestoreConfig(LONG IUserID)

参数: [in] IUserID 用户 ID 号, NET_DVR_Login_V40 的返回值

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通过错误码判断出错原因。

说明:

[返回目录](#)

2.15.11 完全恢复出厂默认参数 **NET_DVR_RemoteControl**

函数: BOOL NET_DVR_RemoteControl(LONG IUserID, DWORD dwCommand, LPVOID lpInBuffer, DWORD dwInBufferSize)

参数: [in] IUserID 用户 ID 号, NET_DVR_Login_V30 返回值

[in] dwCommand 控制命令, 详见表 2.27

[in] lpInBuffer 输入参数, 跟控制命令相关, 详见列表

[in] dwInBufferSize 输入参数长度

返回值: TRUE 表示成功, FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码, 通

过错误码判断出错原因。

说明： 不同的控制功能对应不同的命令号，同时 `lpInBuffer` 对应不同的结构体，如表 2.27 所示。

表 2.27 远程控制命令

dwCommand 宏定义	宏定义值	控制功能	lpInBuffer 对应结构体
NET_DVR_COMPLETE_RESTORE_CTRL	3420	设置完全恢复出厂值	NET_DVR_COMPLETE_RESTORE_INFO

[返回目录](#)

导入/导出配置文件

2.15.12 导出配置文件 `NET_DVR_GetConfigFile_V30`

函数： `BOOL NET_DVR_GetConfigFile_V30(LONG lUserID, char *sOutBuffer, DWORD dwOutSize, DWORD *pReturnSize)`

参数：
 [in] lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
 [out] sOutBuffer 存放配置参数的缓冲区
 [in] dwOutSize 缓冲区大小
 [out] pReturnSize 实际获得的缓冲区大小

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明： 当 `sOutBuffer = NULL`、`dwOutSize = 0` 且 `pReturnSize != NULL` 时用于获取参数配置文件的所需的缓冲区长度；

当 `sOutBuffer != NULL` 且 `dwOutSize != 0` 时用于获取参数配置文件的所需的缓冲区内容。

[返回目录](#)

2.15.13 导出配置文件 `NET_DVR_GetConfigFile`

函数： `BOOL NET_DVR_GetConfigFile(LONG lUserID, char *sFileName)`

参数：
 [in] lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
 [in] sFileName 存放保存配置文件的文件路径（二进制文件）

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说明：

[返回目录](#)

2.15.14 导入配置文件 `NET_DVR_SetConfigFile_EX`

函数： `BOOL NET_DVR_SetConfigFile_EX(LONG lUserID, char *sInBuffer, DWORD dwInSize)`

参数：
 [in] lUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
 [in] sInBuffer 存放配置参数的缓冲区
 [in] dwInSize 缓冲区大小

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通

过错误码判断出错原因。

说 明：

[返回目录](#)

2.15.15 导入配置文件 **NET_DVR_SetConfigFile**

函 数： BOOL NET_DVR_SetConfigFile(LONG IUserID, char *sFileName)

参 数： [in] IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值
 [in] sFileName 存放保存配置文件的文件路径（二进制文件）

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

远程重启

2.15.16 重启设备 **NET_DVR_RebootDVR**

函 数： BOOL NET_DVR_RebootDVR(LONG IUserID)

参 数： [in] IUserID 用户 ID 号，NET_DVR_Login_V40 的返回值

返回值： TRUE 表示成功，FALSE 表示失败。接口返回失败请调用 [NET_DVR_GetLastError](#) 获取错误码，通过错误码判断出错原因。

说 明：

[返回目录](#)

3 错误代码及说明

3.1 网络通讯库错误码

错误名称	错误值	说明
NET_DVR_NOERROR	0	没有错误。
NET_DVR_PASSWORD_ERROR	1	用户名密码错误。注册时输入的用户名或者密码错误。
NET_DVR_NOENOUGHPRI	2	权限不足。该注册用户没有权限执行当前对设备的操作，可以与远程用户参数配置做对比。
NET_DVR_NOINIT	3	SDK 未初始化。
NET_DVR_CHANNEL_ERROR	4	通道号错误。设备没有对应的通道号。
NET_DVR_OVER_MAXLINK	5	连接到设备的用户个数超过最大。
NET_DVR_VERSIONNOMATCH	6	版本不匹配。SDK 和设备的版本不匹配。
NET_DVR_NETWORK_FAIL_CONNECT	7	连接设备失败。设备不在线或网络原因引起的连接超时等。
NET_DVR_NETWORK_SEND_ERROR	8	向设备发送失败。
NET_DVR_NETWORK_RECV_ERROR	9	从设备接收数据失败。
NET_DVR_NETWORK_RECV_TIMEOUT	10	从设备接收数据超时。
NET_DVR_NETWORK_ERRORDATA	11	传送的数据有误。发送给设备或者从设备接收到的数据错误，如远程参数配置时输入设备不支持的值。
NET_DVR_ORDER_ERROR	12	调用次序错误。
NET_DVR_OPERNOPERMIT	13	无此权限。
NET_DVR_COMMANDTIMEOUT	14	设备命令执行超时。
NET_DVR_ERRORSERIALPORT	15	串口号错误。指定的设备串口号不存在。
NET_DVR_ERRORALARMPORT	16	报警端口错误。指定的设备报警输出端口不存在。
NET_DVR_PARAMETER_ERROR	17	参数错误。SDK 接口中给入的输入或输出参数为空。
NET_DVR_CHAN_EXCEPTION	18	设备通道处于错误状态
NET_DVR_NODISK	19	设备无硬盘。当设备无硬盘时，对设备的录像文件、硬盘配置等操作失败。
NET_DVR_ERRORDISKNUM	20	硬盘号错误。当对设备进行硬盘管理操作时，指定的硬盘号不存在时返回该错误。
NET_DVR_DISK_FULL	21	设备硬盘满。
NET_DVR_DISK_ERROR	22	设备硬盘出错
NET_DVR_NOSUPPORT	23	设备不支持。
NET_DVR_BUSY	24	设备忙。
NET_DVR_MODIFY_FAIL	25	设备修改不成功。
NET_DVR_PASSWORD_FORMAT_ERROR	26	密码输入格式不正确
NET_DVR_DISK_FORMATING	27	硬盘正在格式化，不能启动操作。
NET_DVR_DVRNORESOURCE	28	设备资源不足。
NET_DVR_DVRPRATEFAILED	29	设备操作失败。
NET_DVR_OPENHOSTSOUND_FAIL	30	语音对讲、语音广播操作中采集本地音频或打开音频输出失败。
NET_DVR_DVRVOICEOPENED	31	设备语音对讲被占用。

NET_DVR_TIMEINPUTERROR	32	时间输入不正确。
NET_DVR_NOSPECFILE	33	回放时设备没有指定的文件。
NET_DVR_CREATEFILE_ERROR	34	创建文件出错。本地录像、保存图片、获取配置文件和远程下载录像时创建文件失败。
NET_DVR_FILEOPENFAIL	35	打开文件出错。设置配置文件、设备升级、上传审讯文件时打开文件失败。
NET_DVR_OPERNOTFINISH	36	上次的操作还没有完成
NET_DVR_GETPLAYTIMEFAIL	37	获取当前播放的时间出错。
NET_DVR_PLAYFAIL	38	播放出错。
NET_DVR_FILEFORMAT_ERROR	39	文件格式不正确。
NET_DVR_DIR_ERROR	40	路径错误
NET_DVR_ALLOC_RESOURCE_ERROR	41	SDK 资源分配错误。
NET_DVR_AUDIO_MODE_ERROR	42	声卡模式错误。当前打开声音播放模式与实际设置的模式不符出错。
NET_DVR_NOENOUGH_BUF	43	缓冲区太小。接收设备数据的缓冲区或存放图片缓冲区不足。
NET_DVR_CREATESOCKET_ERROR	44	创建 SOCKET 出错。
NET_DVR_SETSOCKET_ERROR	45	设置 SOCKET 出错。
NET_DVR_MAX_NUM	46	个数达到最大。分配的注册连接数、预览连接数超过 SDK 支持的最大数。
NET_DVR_USERNOTEXIST	47	用户不存在。注册的用户 ID 已注销或不可用。
NET_DVR_WRITEFLASHERROR	48	写 FLASH 出错。设备升级时写 FLASH 失败。
NET_DVR_UPGRADEFAIL	49	设备升级失败。网络或升级文件语言不匹配等原因升级失败。
NET_DVR_CARDHAVEINIT	50	解码卡已经初始化过。
NET_DVR_PLAYERFAILED	51	调用播放库中某个函数失败。
NET_DVR_MAX_USERNUM	52	登录设备的用户数达到最大。
NET_DVR_GETLOCALIPANDMACFAIL	53	获得本地 PC 的 IP 地址或物理地址失败。
NET_DVR_NOENCODEING	54	设备该通道没有启动编码。
NET_DVR_IPMISMATCH	55	IP 地址不匹配。
NET_DVR_MACMISMATCH	56	MAC 地址不匹配。
NET_DVR_UPGRADELANGMISMATCH	57	升级文件语言不匹配。
NET_DVR_MAX_PLAYERPORT	58	播放器路数达到最大。
NET_DVR_NOSPACEBACKUP	59	备份设备中没有足够空间进行备份。
NET_DVR_NODEVICEBACKUP	60	没有找到指定的备份设备。
NET_DVR_PICTURE_BITS_ERROR	61	图像素位数不符，限 24 色。
NET_DVR_PICTURE_DIMENSION_ERROR	62	图片高*宽超限，限 128*256。
NET_DVR_PICTURE_SIZ_ERROR	63	图片大小超限，限 100K。
NET_DVR_LOADPLAYERSDKFAILED	64	载入当前目录下 Player Sdk 出错。
NET_DVR_LOADPLAYERSDKPROC_ERROR	65	找不到 Player Sdk 中某个函数入口。
NET_DVR_LOADDSSDKFAILED	66	载入当前目录下 DsSdk 出错。
NET_DVR_LOADDSSDKPROC_ERROR	67	找不到 DsSdk 中某个函数入口。
NET_DVR_DSSDK_ERROR	68	调用硬解码库 DsSdk 中某个函数失败。

NET_DVR_VOICEMONOPOLIZE	69	声卡被独占。
NET_DVR_JOINMULTICASTFAILED	70	加入多播组失败。
NET_DVR_CREATEDIR_ERROR	71	建立日志文件目录失败。
NET_DVR_BINDSOCKET_ERROR	72	绑定套接字失败。
NET_DVR_SOCKETCLOSE_ERROR	73	socket 连接中断，此错误通常是由于连接中断或目的地不可达。
NET_DVR_USERID_ISUSING	74	注销时用户 ID 正在进行某操作。
NET_DVR_SOCKETLISTEN_ERROR	75	监听失败。
NET_DVR_PROGRAM_EXCEPTION	76	程序异常。
NET_DVR_WRITEFILE_FAILED	77	写文件失败。本地录像、远程下载录像、下载图片等操作时写文件失败。
NET_DVR_FORMAT_READONLY	78	禁止格式化只读硬盘。
NET_DVR_WITHSAMEUSERNAME	79	远程用户配置结构中存在相同的用户名。
NET_DVR_DEVICETYPE_ERROR	80	导入参数时设备型号不匹配。
NET_DVR_LANGUAGE_ERROR	81	导入参数时语言不匹配。
NET_DVR_PARAVERSION_ERROR	82	导入参数时软件版本不匹配。
NET_DVR_IPCHAN_NOTALIVE	83	预览时外接 IP 通道不在线。
NET_DVR_RTSP_SDK_ERROR	84	加载标准协议通讯库 StreamTransClient 失败。
NET_DVR_CONVERT_SDK_ERROR	85	加载转封装库失败。
NET_DVR_IPC_COUNT_OVERFLOW	86	超出最大的 IP 接入通道数。
NET_DVR_MAX_ADD_NUM	87	添加录像标签或者其他操作超出最多支持的个数。
NET_DVR_PARAMMODE_ERROR	88	图像增强仪，参数模式错误（用于硬件设置时，客户端进行软件设置时错误值）。
NET_DVR_CODESPITTER_OFFLINE	89	码分器不在线。
NET_DVR_BACKUP_COPYING	90	设备正在备份。
NET_DVR_CHAN_NOTSUPPORT	91	通道不支持该操作。
NET_DVR_CALLINEINVALID	92	高度线位置太集中或长度线不够倾斜。
NET_DVR_CALCANCELCONFLICT	93	取消标定冲突，如果设置了规则及全局的实际大小尺寸过滤。
NET_DVR_CALPOINTOUTRANGE	94	标定点超出范围。
NET_DVR_FILTERRECTINVALID	95	尺寸过滤器不符合要求。
NET_DVR_DDNS_DEVOFFLINE	96	设备没有注册到 ddns 上。
NET_DVR_DDNS_INTER_ERROR	97	DDNS 服务器内部错误。
NET_DVR_ALIAS_DUPLICATE	150	别名重复（EasyDDNS 的配置）
NET_DVR_DEV_NET_OVERFLOW	800	网络流量超过设备能力上限
NET_DVR_STATUS_RECORDFILE_WRITING_NOT_LOCK	801	录像文件在录像，无法被锁定
NET_DVR_STATUS_CANT_FORMAT_LITTLE_DISK	802	由于硬盘太小无法格式化
抓拍机错误码		
NET_DVR_ERR_LANENUM_EXCEED	1400	车道数超出能力。
NET_DVR_ERR_PRAREA_EXCEED	1401	牌识区域过大。
NET_DVR_ERR_LIGHT_PARAM	1402	信号灯接入参数错误。
NET_DVR_ERR_LANE_LINE_INVALID	1403	车道线配置错误。
NET_DVR_ERR_STOP_LINE_INVALID	1404	停止线配置错误。
NET_DVR_ERR_LEFTORRIGHT_LINE_INVALID	1405	左/右转分界线配置错误。

NET_DVR_ERR_LANE_NO_REPEAT	1406	叠加车道号重复。
NET_DVR_ERR_PRAREA_INVALID	1407	牌识多边形不符合要求。
NET_DVR_ERR_LIGHT_NUM_EXCEED	1408	视频检测交通灯信号灯数目超出最大值。
NET_DVR_ERR_SUBLIGHT_NUM_INVALID	1409	视频检测交通灯信号灯子灯数目不合法
NET_DVR_ERR_LIGHT_AREASIZE_INVALID	1410	视频检测交通灯输入信号灯框大小不合法。
NET_DVR_ERR_LIGHT_COLOR_INVALID	1411	视频检测交通灯输入信号灯颜色不合法。
NET_DVR_ERR_LIGHT_DIRECTION_INVALID	1412	视频检测交通灯输入灯方向属性不合法。
NET_DVR_ERR_LACK_IOABLITY	1413	IO 口实际支持的能力不足
NET_DVR_ERR_FTP_PORT	1414	FTP 端口号非法（端口号重复或者异常）
NET_DVR_ERR_FTP_CATALOGUE	1415	FTP 目录名非法（启用多级目录，多级目录传值为空）
NET_DVR_ERR_FTP_UPLOAD_TYPE	1416	FTP 上传类型非法（单 ftp 只支持全部/双 ftp 只支持卡口和违章）
NET_DVR_ERR_FLASH_PARAM_WRITE	1417	配置参数时写 FLASH 失败
NET_DVR_ERR_FLASH_PARAM_READ	1418	配置参数时读 FLASH 失败
NET_DVR_ERR_PICNAME_DELIMITER	1419	FTP 图片命名分隔符非法
NET_DVR_ERR_PICNAME_ITEM	1420	FTP 图片命名项非法（例如 分隔符）
NET_DVR_ERR_PLATE_RECOGNIZE_TYPE	1421	牌识区域类型非法（矩形和多边形有效性校验）
NET_DVR_ERR_CAPTURE_TIMES	1422	抓拍次数非法（有效值是 0~5）
NET_DVR_ERR_LOOP_DISTANCE	1423	线圈距离非法（有效值是 0~2000ms）
NET_DVR_ERR_LOOP_INPUT_STATUS	1424	线圈输入状态非法（有效值）
NET_DVR_ERR_RELATE_IO_CONFLICT	1425	测速组 IO 关联冲突
NET_DVR_ERR_INTERVAL_TIME	1426	连拍间隔时间非法（0~6000ms）
NET_DVR_ERR_SIGN_SPEED	1427	标志限速值非法（大车标志限速不能大于小车标志限速）
NET_DVR_ERR_PIC_FLIP	1428	图像配置翻转（配置交互影响）
NET_DVR_ERR_RELATE_LANE_NUMBER	1429	关联车道数错误(重复 有效值校验 1~99)
NET_DVR_ERR_TRIGGER_MODE	1430	配置抓拍机触发模式非法
NET_DVR_ERR_DELAY_TIME	1431	触发延时时间错误(2000ms)
NET_DVR_ERR_EXCEED_RS485_COUNT	1432	超过最大 485 个数限制
NET_DVR_ERR_RADAR_TYPE	1433	雷达类型错误
NET_DVR_ERR_RADAR_ANGLE	1434	雷达角度错误
NET_DVR_ERR_RADAR_SPEED_VALID_TIME	1435	雷达有效时间错误
NET_DVR_ERR_RADAR_LINE_CORRECT	1436	雷达线性矫正参数错误
NET_DVR_ERR_RADAR_CONST_CORRECT	1437	雷达常量矫正参数错误
NET_DVR_ERR_RECORD_PARAM	1438	录像参数无效（预录时间不超过 10s）
NET_DVR_ERR_LIGHT_WITHOUT_COLOR_AND_DIRECTION	1439	视频检测信号灯配置信号灯个数，但是没有勾选信号灯方向和颜色的
NET_DVR_ERR_LIGHT_WITHOUT_DETECTION_REGION	1440	视频检测信号灯配置信号灯个数，但是没有画检测区域
NET_DVR_ERR_RECOGNIZE_PROVINCE_PARAM	1441	牌识参数省份参数的合法性
NET_DVR_ERR_SPEED_TIMEOUT	1442	IO 测速超时时间非法（有效值大于 0）
NET_DVR_ERR_NTP_TIMEZONE	1443	ntp 时区参数错误
NET_DVR_ERR_NTP_INTERVAL_TIME	1444	ntp 校时间隔错误
NET_DVR_ERR_NETWORK_CARD_NUM	1445	可配置网卡数目错误
NET_DVR_ERR_DEFAULT_ROUTE	1446	默认路由错误

NET_DVR_ERR_BONDING_WORK_MODE	1447	bonding 网卡工作模式错误
NET_DVR_ERR_SLAVE_CARD	1448	slave 网卡错误
NET_DVR_ERR_PRIMARY_CARD	1449	Primary 网卡错误
NET_DVR_ERR_DHCP_PPOE_WORK	1450	dhcp 和 pppoE 不能同时启动
NET_DVR_ERR_NET_INTERFACE	1451	网络接口错误
NET_DVR_ERR_MTU	1452	MTU 错误
NET_DVR_ERR_NETMASK	1453	子网掩码错误
NET_DVR_ERR_IP_INVALID	1454	IP 地址不合法
NET_DVR_ERR_MULTICAST_IP_INVALID	1455	多播地址不合法
NET_DVR_ERR_GATEWAY_INVALID	1456	网关不合法
NET_DVR_ERR_DNS_INVALID	1457	DNS 不合法
NET_DVR_ERR_ALARMHOST_IP_INVALID	1458	告警主机地址不合法
NET_DVR_ERR_IP_CONFLICT	1459	IP 冲突
NET_DVR_ERR_NETWORK_SEGMENT	1460	IP 不支持同网段
NET_DVR_ERR_NETPORT	1461	端口错误
NET_DVR_ERR_PPPOE_NOSUPPORT	1462	PPPOE 不支持
NET_DVR_ERR_DOMAINNAME_NOSUPPORT	1463	域名不支持
NET_DVR_ERR_NO_SPEED	1464	未启用测速功能
NET_DVR_ERR_IOSTATUS_INVALID	1465	IO 状态错误
NET_DVR_ERR_BURST_INTERVAL_INVALID	1466	连拍间隔非法
NET_DVR_ERR_RESERVE_MODE	1467	备用模式错误
NET_DVR_ERR_LANE_NO	1468	叠加车道号错误
NET_DVR_ERR_COIL_AREA_TYPE	1469	线圈区域类型错误
NET_DVR_ERR_TRIGGER_AREA_PARAM	1470	触发区域参数错误
NET_DVR_ERR_SPEED_LIMIT_PARAM	1471	违章限速参数错误
NET_DVR_ERR_LANE_PROTOCOL_TYPE	1472	车道关联协议类型错误
NET_DVR_ERR_INTERVAL_TYPE	1473	连拍间隔类型非法
NET_DVR_ERR_INTERVAL_DISTANCE	1474	连拍间隔距离非法
NET_DVR_ERR_RS485_ASSOCIATE_DEVTYPE	1475	RS485 关联类型非法
NET_DVR_ERR_RS485_ASSOCIATE_LANENO	1476	RS485 关联车道号非法
NET_DVR_ERR_LANENO_ASSOCIATE_MULTIRS485	1477	车道号关联多个 RS485 口
NET_DVR_ERR_LIGHT_DETECTION_REGION	1478	视频检测信号灯配置信号灯个数, 但是检测区域宽或高为 0
NET_DVR_ERR_DN2D_NOSUPPORT	1479	不支持抓拍帧 2D 降噪
NET_DVR_ERR_IRISMODE_NOSUPPORT	1480	不支持的镜头类型
NET_DVR_ERR_WB_NOSUPPORT	1481	不支持的白平衡模式
NET_DVR_ERR_IO_EFFECTIVENESS	1482	IO 口的有效性
NET_DVR_ERR_LIGHTNO_MAX	1483	信号灯检测器接入红/黄灯超限(16)
NET_DVR_ERR_LIGHTNO_CONFLICT	1484	信号灯检测器接入红/黄灯冲突
NET_DVR_ERR_CANCEL_LINE	1485	直行触发线
NET_DVR_ERR_STOP_LINE	1486	待行区停止线
NET_DVR_ERR_RUSH_REDLIGHT_LINE	1487	闯红灯触发线
NET_DVR_ERR_IOOUTNO_MAX	1488	IO 输出口编号越界
NET_DVR_ERR_IOOUTNO_AHEADTIME_MAX	1489	IO 输出口提前时间超限

NET_DVR_ERR_IOOUTNO_IOWORKTIME	1490	IO 输出口有效持续时间超限
NET_DVR_ERR_IOOUTNO_FREQMULTI	1491	IO 输出口脉冲模式下倍频出错
NET_DVR_ERR_IOOUTNO_DUTYRATE	1492	IO 输出口脉冲模式下占空比出错
NET_DVR_ERR_VIDEO_WITH_EXPOSURE	1493	以曝闪起效, 工作方式不支持视频
NET_DVR_ERR_PLATE_BRIGHTNESS_WITHOUT_FLASHDET	1494	车牌亮度自动使能闪光灯仅在车牌亮度补偿模式下起效
NET_DVR_ERR_RECOGNIZE_TYPE_PARAM	1495	识别类型非法 车牌识别参数(如大车、小车、背向、正向、车标识别等)
NET_DVR_ERR_PALTE_RECOGNIZE_AREA_PARAM	1496	牌识参数非法 牌识区域配置时判断出错
NET_DVR_ERR_PORT_CONFLICT	1497	端口有冲突
NET_DVR_ERR_LOOP_IP	1498	IP 不能设置为回环地址
NET_DVR_ERR_DRIVELINE_SENSITIVE	1499	压线灵敏度出错(视频电警模式下)
NET_DVR_ERR_EXCEED_MAX_CAPTURE_TIMES	1600	抓拍模式为频闪时最大抓拍张数为 2 张(IVT 模式下)
NET_DVR_ERR_REDAR_TYPE_CONFLICT	1601	相同 485 口关联雷达类型冲突

3.2 RTSP 通讯库错误码

错误名称	错误值	说明
NET_DVR_RTSP_GETPORTFAILED	407	获取 RTSP 端口错误
NET_DVR_RTSP_DESCRIBESENDTIMEOUT	411	RTSP DESCRIBE 发送超时
NET_DVR_RTSP_DESCRIBESENDERROR	412	RTSP DESCRIBE 发送失败
NET_DVR_RTSP_DESCRIBERECEVTIMEOUT	413	RTSP DESCRIBE 接收超时
NET_DVR_RTSP_DESCRIBERECEVDATALOST	414	RTSP DESCRIBE 接收数据错误
NET_DVR_RTSP_DESCRIBERECEVERROR	415	RTSP DESCRIBE 接收失败
NET_DVR_RTSP_DESCRIBESERVERERR	416	RTSP DESCRIBE 服务器返回 401,501 等错误
NET_DVR_RTSP_SETUPSENDTIMEOUT	421	RTSP SETUP 发送超时
NET_DVR_RTSP_SETUPSENDERROR	422	RTSP SETUP 发送错误
NET_DVR_RTSP_SETUPRECVTIMEOUT	423	RTSP SETUP 接收超时
NET_DVR_RTSP_SETUPRECVDATALOST	424	RTSP SETUP 接收数据错误
NET_DVR_RTSP_SETUPRECVERROR	425	RTSP SETUP 接收失败
NET_DVR_RTSP_OVER_MAX_CHAN	426	设备超过最大连接数
NET_DVR_RTSP_PLAYSENDTIMEOUT	431	RTSP PLAY 发送超时
NET_DVR_RTSP_PLAYSENDERROR	432	RTSP PLAY 发送错误
NET_DVR_RTSP_PLAYRECVTIMEOUT	433	RTSP PLAT 接收超时
NET_DVR_RTSP_PLAYRECVDATALOST	434	RTSP PLAY 接收数据错误
NET_DVR_RTSP_PLAYRECVERROR	435	RTSP PLAY 接收失败
NET_DVR_RTSP_PLAYSERVERERR	436	RTSP PLAY 设备返回错误状态
NET_DVR_RTSP_TEARDOWNSENDTIMEOUT	441	RTSP TEARDOWN 发送超时
NET_DVR_RTSP_TEARDOWNSENDERROR	442	RTSP TEARDOWN 发送错误
NET_DVR_RTSP_TEARDOWNRECVTIMEOUT	443	RTSP TEARDOWN 接收超时

NET_DVR_RTSP_TEARDOWNRECVDATALOST	444	RTSP TEARDOWN 接收数据错误
NET_DVR_RTSP_TEARDOWNRECVERRORE	445	RTSP TEARDOWN 接收失败
NET_DVR_RTSP_TEARDOWNSERVERERR	446	RTSP TEARDOWN 设备返回错误状态

3.3 软解码库错误码

错误名称	错误值	说明
NET_PLAYM4_NOERROR	500	没有错误
NET_PLAYM4_PARA_OVER	501	输入参数非法
NET_PLAYM4_ORDER_ERROR	502	调用顺序不对
NET_PLAYM4_TIMER_ERROR	503	多媒体时钟设置失败
NET_PLAYM4_DEC_VIDEO_ERROR	504	视频解码失败
NET_PLAYM4_DEC_AUDIO_ERROR	505	音频解码失败
NET_PLAYM4_ALLOC_MEMORY_ERROR	506	分配内存失败
NET_PLAYM4_OPEN_FILE_ERROR	507	文件操作失败
NET_PLAYM4_CREATE_OBJ_ERROR	508	创建线程事件等失败
NET_PLAYM4_CREATE_DDRAW_ERROR	509	创建 directDraw 失败
NET_PLAYM4_CREATE_OFFSCREEN_ERROR	510	创建后端缓存失败
NET_PLAYM4_BUF_OVER	511	缓冲区满, 输入流失败
NET_PLAYM4_CREATE_SOUND_ERROR	512	创建音频设备失败
NET_PLAYM4_SET_VOLUME_ERROR	513	设置音量失败
NET_PLAYM4_SUPPORT_FILE_ONLY	514	只能在播放文件时才能使用此接口
NET_PLAYM4_SUPPORT_STREAM_ONLY	515	只能在播放流时才能使用此接口
NET_PLAYM4_SYS_NOT_SUPPORT	516	系统不支持, 解码器只能工作在 Pentium 3 以上
NET_PLAYM4_FILEHEADER_UNKNOWN	517	没有文件头
NET_PLAYM4_VERSION_INCORRECT	518	解码器和编码器版本不对应
NET_PLAYM4_INIT_DECODER_ERROR	519	初始化解码器失败
NET_PLAYM4_CHECK_FILE_ERROR	520	文件太短或码流无法识别
NET_PLAYM4_INIT_TIMER_ERROR	521	初始化多媒体时钟失败
NET_PLAYM4_BLT_ERROR	522	位拷贝失败
NET_PLAYM4_UPDATE_ERROR	523	显示 overlay 失败
NET_PLAYM4_OPEN_FILE_ERROR_MULTI	524	打开混合流文件失败
NET_PLAYM4_OPEN_FILE_ERROR_VIDEO	525	打开视频流文件失败
NET_PLAYM4_JPEG_COMPRESS_ERROR	526	JPEG 压缩错误
NET_PLAYM4_EXTRACT_NOT_SUPPORT	527	不支持该文件版本.
NET_PLAYM4_EXTRACT_DATA_ERROR	528	提取文件数据失败

4 附录:结构体

4.1 NET_DVR_ACTIVATECFG:设备激活参数

```
struct{
    DWORD    dwSize;
    BYTE     sPassword[PASSWD_LEN];
    BYTE     byRes[108];
}NET_DVR_ACTIVATECFG,*LPNET_DVR_ACTIVATECFG;
```

Members

dwSize

结构体大小

sPassword

初始密码，密码等级弱或者以上

byRes

保留，置为 0

Remarks

- 出厂设备需要先激活，然后再使用激活使用的初始密码登录设备。
- 将密码输入分为数字(0~9)、小写字母(a~z)、大写字母(A~Z)、特殊符号 (:\除外) 4 类，等级分为 4 个等级，如下所示：
 - 1) 等级 0 (风险密码): 密码长度小于 8 位，或者只包含 4 类字符中的任意一类，或者密码与用户名一样，或者密码是用户名的倒写。例如：12345、abcdef。
 - 2) 等级 1 (弱密码): 包含两类字符，且组合为 (数字+小写字母) 或 (数字+大写字母)，且长度大于等于 8 位。例如：abc12345、123ABCDEF
 - 3) 等级 2 (中密码): 包含两类字符，且组合不能为 (数字+小写字母) 和 (数字+大写字母)，且长度大于等于 8 位。例如：12345***++、ABCDabcd。
 - 4) 等级 3 (强密码): 包含三类字符及以上，且长度大于等于 8 位。例如：Abc12345、abc12345++。

4.2 NET_DVR_AGAIN_RELATEDDEV:二次确认机关联网设备参数

```
struct{
    NET_DVR_IPADDR  struSIPServer;
    NET_DVR_IPADDR  struCenterAddr;
    WORD            wCenterPort;
    BYTE            byRes1[2];
    NET_DVR_IPADDR  struIndoorUnit;
    BYTE            byRes[588];
}NET_DVR_AGAIN_RELATEDDEV,*LPNET_DVR_AGAIN_RELATEDDEV;
```

Members

struSIPServer

SIP 服务器 IP 地址

struCenterAddr
中心平台 IP 地址

wCenterPort
中心平台端口

byRes
保留，置为 0

struIndoorUnit
室内主机 IP 地址

byRes
保留，置为 0

4.3 NET_DVR_ALARMER:报警设备信息

```
struct{
    BYTE    byUserIDValid;
    BYTE    bySerialValid;
    BYTE    byVersionValid;
    BYTE    byDeviceNameValid;
    BYTE    byMacAddrValid;
    BYTE    byLinkPortValid;
    BYTE    byDeviceIPValid;
    BYTE    bySocketIPValid;
    LONG    lUserID;
    BYTE    sSerialNumber[SERIALNO_LEN];
    DWORD   dwDeviceVersion;
    char    sDeviceName[NAME_LEN];
    BYTE    byMacAddr[MACADDR_LEN];
    WORD    wLinkPort;
    char    sDeviceIP[128];
    char    sSocketIP[128];
    BYTE    byIpProtocol;
    BYTE    byRes2[11];
}NET_DVR_ALARMER,*LPNET_DVR_ALARMER;
```

Members

byUserIDValid
userid 是否有效：0—无效；1—有效

bySerialValid
序列号是否有效：0—无效；1—有效

byVersionValid
版本号是否有效：0—无效；1—有效

byDeviceNameValid
设备名字是否有效：0—无效；1—有效

byMacAddrValid
MAC 地址是否有效：0—无效；1—有效

byLinkPortValid

Login 端口是否有效：0—无效；1—有效

byDeviceIPValid

设备 IP 是否有效：0—无效；1—有效

bySocketIPValid

Socket IP 是否有效：0-无效；1-有效

lUserID

NET_DVR_Login 或 NET_DVR_Login_V30 返回值，布防时有效

sSerialNumber

序列号

dwDeviceVersion

版本信息：V3.0 以上版本支持的设备最高 8 位为主版本号，次高 8 位为次版本号，低 16 位为修复版本号；V3.0 以下版本支持的设备高 16 位表示主版本，低 16 位表示次版本

sDeviceName

设备名称

byMacAddr

MAC 地址

wLinkPort

设备通讯端口

sDeviceIP

设备 IP 地址

sSocketIP

报警主动上传时的 Socket IP 地址

byIpProtocol

IP 协议：0- IPV4，1- IPV6

byRes2

保留，置为 0

4.4 NET_DVR_ALARMIN_PARAM:防区参数

```

struct{
    DWORD                dwSize;
    BYTE                 byName[NAME_LEN];
    WORD                 wDetectorType;
    BYTE                 byType;
    BYTE                 byUploadAlarmRecoveryReport;
    DWORD                dwParam;
    NET_DVR_SCHEDTIME   struAlarmTime[MAX_DAYS][MAX_TIMESEGMENT];
    BYTE                 byAssociateAlarmOut[MAX_ALARMHOST_ALARMOUT_NUM];
    BYTE                 byAssociateSirenOut[8];
    BYTE                 bySensitivityParam;
    BYTE                 byArrayBypass;
    BYTE                 byJointSubSystem;
    BYTE                 byModuleStatus;
}

```

```

WORD            wModuleAddress;
BYTE            byModuleChan;
BYTE            byModuleType;
WORD            wZoneIndex;
WORD            wInDelay;
WORD            wOutDelay;
BYTE            byAlarmType;
BYTE            byZoneResistor;
float           fZoneResistorManual;
BYTE            byRes2[32];

```

```
}NET_DVR_ALARMIN_PARAM, *LPNET_DVR_ALARMIN_PARAM;
```

Members

dwSize

结构体大小

byName

防区名称

wDetectorType

防区探测器类型，具体定义如下：

```

enum _DETECTOR_TYPE_{
    PANIC_BUTTON = 0,
    MAGNETIC_CONTACT,
    SMOKE_DETECTOR,
    ACTIVE_INFRARED_DETECTOR,
    PASSIVE_INFRARED_DETECTOR,
    GLASS_BREAK_DETECTOR,
    VIBRATION_DETECTOR,
    DUAL_TECHNOLOGY_PIR_DETECTOR,
    TRIPLE_TECHNOLOGY_PIR_DETECTOR,
    HUMIDITY_DETECTOR,
    TEMPERATURE_DETECTOR,
    COMBUSTIBLE_GAS_DETECTOR,
    DYNAMIC_SWITCH,
    CONTROL_SWITCH,
    OTHER_DETECTOR = 0xffff
}DETECTOR_TYPE

```

PANIC_BUTTON

紧急开关

MAGNETIC_CONTACT

门磁开关

SMOKE_DETECTOR

烟感探测器

ACTIVE_INFRARED_DETECTOR

主动红外探测器

PASSIVE_INFRARED_DETECTOR

被动红外探测器
GLASS_BREAK_DETECTOR
 玻璃破碎探测器
VIBRATION_DETECTOR
 震动探测器
DUAL_TECHNOLOGY_PIR_DETECTOR
 双鉴移动探测器
TRIPLE_TECHNOLOGY_PIR_DETECTOR
 三技术探测器
HUMIDITY_DETECTOR
 湿度探测器
TEMPERATURE_DETECTOR
 温感探测器
COMBUSTIBLE_GAS_DETECTOR
 可燃气体探测器
DYNAMIC_SWITCH
 随动开关
CONTROL_SWITCH
 控制开关
OTHER_DETECTOR
 其他探测器

byType

防区报警类型，0- 即时防区，1- 24 小时有声防区，2- 延时防区，3- 内部防区，4- 钥匙布撤防防区，5- 火警防区，6- 周界防区，7- 24 小时无声防区，8- 24 小时辅助防区，9- 24 小时震动防区，0xff- 无

byUploadAlarmRecoveryReport

是否上传防区报警恢复报告：0-不上传，1-上传

dwParam

防区参数，延时防区延时多长时间

struAlarmTime

布防时间时间段

byAssociateAlarmOut

防区关联报警输出

byAssociateSirenOut

关联警号输出，数组 0 表示警号 1，数组 1 表示警号 2，以此类推。值为 1 表示输出，0 表示不输出

bySensitivityParam

防区灵敏度参数：0- 10ms，1- 250ms，2- 500ms，3- 750ms

byArrayBypass

是否加入旁路组：0- 不支持组旁路，1- 支持组旁路

byJointSubSystem

防区所属的子系统号，该参数只能获取

byModuleStatus

外接防区模块状态：1- 在线，2- 离线，该参数只能获取

wModuleAddress

模块地址，扩展模块从 1~253，0xFFFF 表示无效，该参数只能获取

byModuleChan

模块通道号，从 1 开始，最大值根据模块类型来决定，0xFF 表示无效，该参数只能获取

byModuleType

模块类型：1- 本地防区，2- 单防区，3- 双防区，4- 8 防区，5- 8 路模拟量防区，6- 单防区触发器

wZoneIndex

防区号，该参数只能获取

wInDelay

进入延时，取值范围：0~255 秒

wOutDelay

退出延时，取值范围：0~255 秒

byAlarmType

报警器类型：0- 无效，1- 常开，2- 常闭

byZoneResistor

防区电阻，单位：千欧，取值：0- 无效，1- 2.2，2- 4.7，3- 5.6，4- 8.2，0xff- 自定义

fZoneResistorManual

防区手动电阻，取值范围：1.0~10.0，精确到小数点后一位，单位：千欧，*byZoneResistor* 为 0xff 时有效

byRes2

保留

4.5 NET_DVR_CLIENTINFO:预览参数

```
struct{
    LONG    IChannel;
    LONG    ILinkMode;
    HWND    hPlayWnd;
    char    *sMultiCastIP;
}NET_DVR_CLIENTINFO, *LPNET_DVR_CLIENTINFO;
```

Members

IChannel

通道号，对于摄像机，只有一个通道，通道号为 1

ILinkMode

最高位(31)为 0 表示主码流，为 1 表示子码流；0~30 位表示连接方式：0- TCP 方式，1- UDP 方式，2- 多播方式，3- RTP 方式，4- RTP over RTSP，5- RTSP over HTTP

例如子码流 TCP 连接，则 *ILinkMode*=0x80000000

hPlayWnd

播放窗口的句柄，为 NULL 表示不显示图像

sMultiCastIP

多播组地址

Remarks:

Windows 下该结构中的 *hPlayWnd* 参数若设置为 NULL，则 SDK 仍取流，但不进行解码显示，所以仍可以录像。Linux 下该结构中的 *hPlayWnd* = {0}则 SDK 仍取流，不进行解码显示，所以仍可以录像，但是不能设置 *hPlayWnd* = 0(即 NULL)，否则非法结构地址会导致调用 *hPlayWnd.x* 等去判断的时候崩溃。

当客户选择普通协议传输时，若设备支持 RTSP 协议，SDK 将直接转成 RTSP 方式连接设备。

Linux 下

HWND 表示播放窗口的句柄，定义为：

```
typedef unsigned int HWND;
```

如果使用 Qt 进行界面开发，示例如下：

```
NET_DVR_CLIENTINFO tmpclientinfo;
```

```
tmpclientinfo.hPlayWnd = (HWND)m_framePlayWnd->GetPlayWndId();
```

4.6 NET_DVR_COLOR: 图像参数

```
struct{
    BYTE    byBrightness;
    BYTE    byContrast;
    BYTE    bySaturation;
    BYTE    byHue;
}NET_DVR_COLOR, *LPNET_DVR_COLOR;
```

Members

byBrightness

亮度，取值范围[0,255]

byContrast

对比度，取值范围[0,255]

bySaturation

饱和度，取值范围[0,255]

byHue

色调，取值范围[0,255]

4.7 NET_DVR_COMPLETE_RESTORE_INFO: 完全恢复出厂值控制参数

```
struct{
    DWORD    dwSize;
    DWORD    dwChannel;
    BYTE     byRes[64];
}NET_DVR_COMPLETE_RESTORE_INFO,*LPNET_DVR_COMPLETE_RESTORE_INFO;
```

Members

dwSize

结构体大小

dwChannel

设备通道号，目前该参数无效

byRes

保留，置为 0

Remarks

远程控制设备完全恢复出厂默认值，包括设备 IP 地址会恢复成 192.0.0.64。只有 admin 管理员账号才支持该功能。

完全恢复之后设备会恢复为未激活状态，调用 [NET_DVR_ActivateDevice](#) 可以激活设备。

4.8 NET_DVR_COMPRESSIONCFG_V30:通道压缩参数

```
struct{
    DWORD                dwSize;
    NET_DVR_COMPRESSION_INFO_V30 struNormHighRecordPara;
    NET_DVR_COMPRESSION_INFO_V30 struRes;
    NET_DVR_COMPRESSION_INFO_V30 struEventRecordPara;
    NET_DVR_COMPRESSION_INFO_V30 struNetPara;
}NET_DVR_COMPRESSIONCFG_V30, *LPNET_DVR_COMPRESSIONCFG_V30;
```

Members

dwSize

结构体大小

struNormHighRecordPara

录像的码流压缩参数（即主码流的压缩参数）

struRes

保留，置为 0

struEventRecordPara

事件触发压缩参数，智能交通摄像机不支持，保留

struNetPara

网传的码流压缩参数（即子码流的压缩参数）

4.9 NET_DVR_COMPRESSION_INFO_V30:码流压缩参数

```
struct{
    BYTE    byStreamType;
    BYTE    byResolution;
    BYTE    byBitrateType;
    BYTE    byPicQuality;
    DWORD   dwVideoBitrate;
    DWORD   dwVideoFrameRate;
    WORD    wIntervalFrameI;
    BYTE    byIntervalBPFrame;
    BYTE    byres1;
    BYTE    byVideoEncType;
    BYTE    byAudioEncType;
    BYTE    byVideoEncComplexity;
    BYTE    byres[9];
}NET_DVR_COMPRESSION_INFO_V30, *LPNET_DVR_COMPRESSION_INFO_V30;
```

Members

byStreamType

码流类型：0-视频流，1-复合流

byResolution

分辨率: 0-DCIF(528*384/528*320), 1-CIF(352*288/352*240), 2-QCIF(176*144/176*120), 3-4CIF(704*576/704*480)或 D1(720*576/720*486), 4-2CIF(704*288/704*240), 6-QVGA(320*240), 7-QQVGA(160*120), 12-384*288, 13-576*576, 16-VGA(640*480), 17-UXGA(1600*1200), 18-SVGA(800*600), 19-HD720P(1280*720), 20-XVGA(1280*960), 21-HD900P(1600*900), 23-1536*1536, 24-1920*1920, 27-1920*1080p, 28-2560*1920, 29-1600*304, 30-2048*1536, 31-2448*2048, 32-2448*1200, 33-2448*800, 34-XGA(1024*768), 35-SXGA(1280*1024), 36-WD1(960*576/960*480), 37-1080i(1920*1080), 38-WXGA(1440*900), 39-HD_F(1920*1080/1280*720), 40-HD_H(1920*540/1280*360), 41-HD_Q(960*540/630*360), 42-2336*1744, 43-1920*1456, 44-2592*2048, 45-3296*2472, 46-1376*768, 47-1366*768, 48-1360*768, 49-WSXGA+, 50-720*720, 51-1280*1280, 52-2048*768, 53-2048*2048, 54-2560*2048, 55-3072*2048, 56-2304*1296, 57-WXGA(1280*800), 58-1600*600, 59-1600*900, 60-2752*2208, 0xff-Auto(使用当前码流分辨率)

byBitrateType

码率类型: 0-变码率, 1-定码率

byPicQuality

图象质量: 0-最好, 1-次好, 2-较好, 3-一般, 4-较差, 5-差

dwVideoBitrate

码率: 0-保留, 1-16K(保留), 2-32K, 3-48k, 4-64K, 5-80K, 6-96K, 7-128K, 8-160k, 9-192K, 10-224K, 11-256K, 12-320K, 13-384K, 14-448K, 15-512K, 16-640K, 17-768K, 18-896K, 19-1024K, 20-1280K, 21-1536K, 22-1792K, 23-2048K, 24-3072K, 25-4096K, 26-8192K, 27-16384K。

最高位(31位)置成1表示是自定义码流, 0~30位表示码流值, 最小值16k

dwVideoFrameRate

帧率: 0-全部, 1-1/16, 2-1/8, 3-1/4, 4-1/2, 5-1, 6-2, 7-4, 8-6, 9-8, 10-10, 11-12, 12-16, 13-20, 14-15, 15-18, 16-22

wIntervalFrameI

I 帧间隔, 0xffff-无效

byIntervalBPFrame

帧格式: 0-BBP 帧, 1-BP 帧, 2-单 P 帧, 0xff-无效

byres1

保留, 置为 0

byVideoEncType

视频编码类型: 0-私有 264, 1-标准 h264, 2-标准 mpeg4, 7-M-JPEG, 0xff-无效

byAudioEncType

音频编码类型: 0-OggVorbis, 1-G711_U, 2-G711_A, 6-G726, 0xff-无效

byVideoEncComplexity

视频编码复杂度: 0-低, 1-中, 2-高

byres

保留, 置为 0

Remarks

当修改设备视频编码类型时, 需要重启设备生效。

4.10 NET_DVR_DDNSPARA_V30:网络应用参数 (DDNS)

```
struct{
```

```

BYTE    byEnableDDNS;
BYTE    byHostIndex;
BYTE    byRes1[2];
struct{
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassword[PASSWD_LEN];
    BYTE    sDomainName[MAX_DOMAIN_NAME];
    BYTE    sServerName[MAX_DOMAIN_NAME];
    WORD    wDDNSPort;
    BYTE    byRes[10];
}struDDNS[MAX_DDNS_NUMS];
BYTE    byRes2[16];
}NET_DVR_DDNSPARA_V30,*LPNET_DVR_DDNSPARA_V30;

```

Members

byEnableDDNS

是否使能：0- 否，1- 是

byHostIndex

0- Private DNS, 1- Dydns, 2- PeanutHull(花生壳), 3- NO-IP, 4- hiDDNS

byRes1

保留，置为 0

sUsername

DDNS 账号用户名

sPassword

DDNS 账号密码

sDomainName

域名

sServerName

DDNS 对应的服务器地址，可以是 IP 地址或域名

wDDNSPort

DDNS 端口

byRes

保留，置为 0

byRes2

保留，置为 0

4.11 NET_DVR_DECODERCFG_V30:云台解码器(RS485)参数

```

struct{
    DWORD    dwSize;
    DWORD    dwBaudRate;
    BYTE     byDataBit;
    BYTE     byStopBit;
    BYTE     byParity;
    BYTE     byFlowcontrol;
}

```

```

WORD        wDecoderType;
WORD        wDecoderAddress;
BYTE        bySetPreset[MAX_PRESET_V30];
BYTE        bySetCruise[MAX_CRUISE_V30];
BYTE        bySetTrack[MAX_TRACK_V30];
}NET_DVR_DECODERCFG_V30, *LPNET_DVR_DECODERCFG_V30;

```

Members

dwSize

结构体大小

dwBaudRate

波特率(bps), 0-50, 1-75, 2-110, 3-150, 4-300, 5-600, 6-1200, 7-2400, 8-4800, 9-9600, 10-19200, 11-38400, 12-57600, 13-76800, 14-115.2k

byDataBit

数据有几位: 0-5 位, 1-6 位, 2-7 位, 3-8 位

byStopBit

停止位: 0-1 位, 1-2 位

byParity

是否校验: 0-无校验, 1-奇校验, 2-偶校验

byFlowcontrol

是否流控: 0-无, 1-软流控,2-硬流控

wDecoderType

解码器类型, 通过 [NET_DVR_GetPTZProtocol](#) 获取, 该值对应于结构 [NET_DVR_PTZ_PROTOCOL](#) 中的 *dwType*

wDecoderAddress

解码器地址: [0,255]

bySetPreset

预置点是否设置: 0-没有设置, 1-设置

bySetCruise

巡航是否设置: 0-没有设置, 1-设置

bySetTrack

花样扫描是否设置: 0-没有设置, 1-设置

Remarks

在早前的设备中规定了一系列云台协议, 但在后期的设备仅保留一部分常用的协议, 所以在配置解码器类型时必须调用 [NET_DVR_GetPTZProtocol](#) 获取当前设备支持的云台协议。

4.12 NET_DVR_DEVICECFG_V40:设备参数

```

struct{
    DWORD    dwSize;
    BYTE     sDVRName[NAME_LEN];
    DWORD    dwDVRID;
    DWORD    dwRecycleRecord;
    BYTE     sSerialNumber[SERIALNO_LEN];
    DWORD    dwSoftwareVersion;
}

```

```

DWORD    dwSoftwareBuildDate;
DWORD    dwDSPSoftwareVersion;
DWORD    dwDSPSoftwareBuildDate;
DWORD    dwPanelVersion;
DWORD    dwHardwareVersion;
BYTE     byAlarmInPortNum;
BYTE     byAlarmOutPortNum;
BYTE     byRS232Num;
BYTE     byRS485Num;
BYTE     byNetworkPortNum;
BYTE     byDiskCtrlNum;
BYTE     byDiskNum;
BYTE     byDVRType;
BYTE     byChanNum;
BYTE     byStartChan;
BYTE     byDecordChans;
BYTE     byVGANum;
BYTE     byUSBNum;
BYTE     byAuxoutNum;
BYTE     byAudioNum;
BYTE     byIPChanNum;
BYTE     byZeroChanNum;
BYTE     bySupport;
BYTE     byEsataUseage;
BYTE     byIPCPlug;
BYTE     byStorageMode;
BYTE     bySupport1;
WORD     wDevType;
BYTE     byDevTypeName[24];
BYTE     byRes2[16];
}NET_DVR_DEVICECFG_V40,*LPNET_DVR_DEVICECFG_V40;

```

Members

dwSize

结构体大小

sDVRName

设备名称

dwDVRID

设备 ID 号，用于遥控器，v1.4 的设备号范围为(0-99)，v1.5 及以上版本的设备号为(0-255)

dwRecycleRecord

是否循环录像：0—不是；1—是

以下参数不可更改

sSerialNumber

设备序列号

dwSoftwareVersion

软件版本号，V3.0 以上版本支持的设备最高 8 位为主版本号，次高 8 位为次版本号，低 16 位为修复版本号；V3.0 以下版本支持的设备高 16 位表示主版本，低 16 位表示次版本

dwSoftwareBuildDate

软件生成日期，0xYYYYMMDD

dwDSPSoftwareVersion

DSP 软件版本，高 16 位是主版本，低 16 位是次版本

dwDSPSoftwareBuildDate

DSP 软件生成日期，0xYYYYMMDD

dwPanelVersion

前面板版本，高 16 位是主版本，低 16 位是次版本

dwHardwareVersion

硬件版本，高 16 位是主版本，低 16 位是次版本

byAlarmInPortNum

设备报警输入个数

byAlarmOutPortNum

设备报警输出个数

byRS232Num

设备 232 串口个数

byRS485Num

设备 485 串口个数

byNetworkPortNum

网络口个数

byDiskCtrlNum

硬盘控制器个数

byDiskNum

硬盘个数

byDVRType

设备类型

byChanNum

设备模拟通道个数

byStartChan

起始通道号

byDecordChans

设备解码路数

byVGANum

VGA 口的个数

byUSBNum

USB 口的个数

byAuxoutNum

辅口的个数

byAudioNum

语音口的个数

byIPChanNum

最大数字通道

byZeroChanNum

零通道编码个数

bySupport

能力，位与结果为 0 表示不支持，1 表示支持

bySupport & 0x1，表示是否支持智能搜索

bySupport & 0x2，表示是否支持备份

bySupport & 0x4，表示是否支持压缩参数能力获取

bySupport & 0x8，表示是否支持双网卡

bySupport & 0x10，表示支持远程 SADP

bySupport & 0x20，表示支持 Raid 卡功能

bySupport & 0x40，表示支持 IPSAN 搜索

bySupport & 0x80，表示支持 rtp over rtsp

byEsataUseage

Esata 的默认用途，0-默认备份，1-默认录像

byIPCPlug

0-不支持即插即用，1-支持即插即用

byStorageMode

0-盘组模式，1-磁盘配额

bySupport1

能力集扩充，位与结果为 0 表示不支持，1 表示支持

bySupport1 & 0x1，表示是否支持 snmp v30

bySupport1 & 0x2，支持区分回放和下载

wDevType

设备型号

byDevTypeName

设备型号名称

byRes2

保留，置为 0

Remarks

如果 byDVRType 是 0，则接口中解析 wDevType 作为设备型号，设备端同时将设备型号的名称传过来。

如果 byDVRType 不是 0，则接口中将不解析 wDevType 及 byDevTypeName，使用已有的设备型号及名称对 byDVRType、wDevType、byDevTypeName 进行填充，其中 byDVRType=wDevType。

建议开发时使用 wDevType、byDevTypeName，而不要使用 byDVRType，sdk 内部兼容。

4.13 NET_DVR_DEVICEINFO_V30:设备参数

```
struct{
    BYTE    sSerialNumber[SERIALNO_LEN];
    BYTE    byAlarmInPortNum;
    BYTE    byAlarmOutPortNum;
    BYTE    byDiskNum;
    BYTE    byDVRType;
    BYTE    byChanNum;
    BYTE    byStartChan;
```

```

BYTE    byAudioChanNum;
BYTE    byIPChanNum;
BYTE    byZeroChanNum;
BYTE    byMainProto;
BYTE    bySubProto;
BYTE    bySupport;
BYTE    bySupport1;
BYTE    bySupport2;
WORD    wDevType;
BYTE    bySupport3;
BYTE    byMultiStreamProto;
BYTE    byStartDChan;
BYTE    byStartDTalkChan;
BYTE    byHighDChanNum;
BYTE    byRes2[11];
}NET_DVR_DEVICEINFO_V30,*LPNET_DVR_DEVICEINFO_V30;

```

Members

sSerialNumber

序列号

byAlarmInPortNum

报警输入个数

byAlarmOutPortNum

报警输出个数

byDiskNum

硬盘个数

byDVRType

设备类型，详见下文列表

byChanNum

设备模拟通道个数，数字（IP）通道最大个数通过 NET_DVR_GetDVRConfig（配置命令 NET_DVR_GET_IPPARACFG_V40）获取（dwDChanNum）。

byStartChan

模拟通道的起始通道号，目前设备模拟通道号从 1 开始，数字通道的起始通道号通过 NET_DVR_GetDVRConfig（配置命令 NET_DVR_GET_IPPARACFG_V40）获取（dwStartDChan）。

byAudioChanNum

设备语音通道数

byIPChanNum

设备最大数字通道个数，低 8 位，高 8 位见 byHighDChanNum。可以根据 IP 通道个数来判断是否调用 NET_DVR_GetDVRConfig（配置命令 NET_DVR_GET_IPPARACFG_V40）获取模拟和数字通道相关参数（NET_DVR_IPPARACFG_V40）。

byZeroChanNum

零通道编码个数

byMainProto

主码流传输协议类型：0-private，1-rtsp

bySubProto

子码流传输协议类型：0-private，1-rtsp

bySupport

能力，位与结果为 0 表示不支持，1 表示支持

bySupport & 0x1, 表示是否支持智能搜索

bySupport & 0x2, 表示是否支持备份

bySupport & 0x4, 表示是否支持压缩参数能力获取

bySupport & 0x8, 表示是否支持双网卡

bySupport & 0x10, 表示支持远程 SADP

bySupport & 0x20, 表示支持 Raid 卡功能

bySupport & 0x40, 表示支持 IPSAN 目录查找

bySupport & 0x80, 表示支持 rtp over rtsp

bySupport1

能力集扩充，位与结果为 0 表示不支持，1 表示支持

bySupport1 & 0x1, 表示是否支持 snmp v30

bySupport1 & 0x2, 表示是否支持区分回放和下载

bySupport1 & 0x4, 表示是否支持布防优先级

bySupport1 & 0x8, 表示智能设备是否支持布防时间段扩展

bySupport1 & 0x10, 表示是否支持多磁盘数（超过 33 个）

bySupport1 & 0x20, 表示是否支持 rtsp over http

bySupport1 & 0x40, 表示是否支持延时预览

bySupport1 & 0x80, 表示是否支持车牌新报警信息

bySupport2

能力集扩充，位与结果为 0 表示不支持，1 表示支持

bySupport2 & 0x1, 表示解码器是否支持通过 URL 取流解码

wDevType

设备型号，详见下文列表

bySupport3

能力集扩展，位与结果为 0 表示不支持，1 表示支持

bySupport3 & 0x1, 表示是否支持多码流

byMultiStreamProto

多码流是否支持 rtsp 协议取流，按位表示，0-不支持，1-支持：bit0- 码流 3，bit1- 码流 4，依次类推

byStartDChan

起始数字通道号，0 表示无效

byStartDTalkChan

起始数字对讲通道号，区别于模拟对讲通道号，0 表示无效

byHighDChanNum

数字通道个数，高 8 位

byRes2

保留，置为 0

Remarks

如果 *byDVRType* 是 0, 则接口中解析 *wDevType* 作为设备型号; 如果 *byDVRType* 非 0, 则接口中 *byDVRType* 和 *wDevType* 值相等, 都是 *byDVRType*。推荐使用 *wDevType* 作为设备类型。

4.14 NET_DVR_DEVICEINFO_V40:设备参数结构体

```
struct{
    NET_DVR_DEVICEINFO_V30    struDeviceV30;
    BYTE                      bySupportLock;
    BYTE                      byRetryLoginTime;
    BYTE                      byPasswordLevel;
    BYTE                      byRes1;
    DWORD                    dwSurplusLockTime;
    BYTE                      byCharEncodeType;
    BYTE                      byRes2[255];
}NET_DVR_DEVICEINFO_V40,*LPNET_DVR_DEVICEINFO_V40;
```

Members

struDeviceV30

设备参数

bySupportLock

设备是否支持锁定功能，*bySupportLock* 为 1 时，*dwSurplusLockTime* 和 *byRetryLoginTime* 有效

byRetryLoginTime

剩余可尝试登陆的次数，用户名、密码错误时，此参数有效

byPasswordLevel

密码安全等级：0- 无效，1- 默认密码，2- 有效密码，3- 风险较高的密码，当管理员用户的密码为出厂默认密码（12345）或者风险较高的密码时，建议上层客户端提示用户更改密码

byRes1

保留，置为 0

dwSurplusLockTime

剩余时间，单位：秒，用户锁定时此参数有效。在锁定期间，用户尝试登陆，不管用户名密码输入对错，设备锁定剩余时间重新恢复到 30 分钟

byCharEncodeType

字符编码类型（SDK 所有接口返回的字符串编码类型，透传接口除外）：0- 无字符编码信息(老设备)，1- GB2312(简体中文)，2- GBK，3- BIG5(繁体中文)，4- Shift_JIS(日文)，5- EUC-KR(韩文)，6- UTF-8，7- ISO8859-1，8- ISO8859-2，9- ISO8859-3，...，依次类推，21- ISO8859-15(西欧)

byRes2

保留，置为 0

Remarks

- 将密码输入分为数字(0~9)、小写字母(a~z)、大写字母(A~Z)、特殊符号 (:\除外) 4 类，等级分为 4 个等级，如下所示：

等级 0 (风险密码)：密码长度小于 8 位，或者只包含 4 类字符中的任意一类，或者密码与用户名一样，或者密码是用户名的倒写。例如：12345、abcdef。

等级 1 (弱密码)：包含两类字符，且组合为（数字+小写字母）或（数字+大写字母），且长度大于等于 8 位。例如：abc12345、123ABCDEF。

等级 2 (中密码)：包含两类字符，且组合不能为（数字+小写字母）和（数字+大写字母），且长度大于等于 8 位。例如：12345***++、ABCDabcd。

等级 3 (强密码)：包含三类字符及以上，且长度大于等于 8 位。例如：Abc12345、abc12345++。

4.15 NET_DVR_DIRECT_CONNECT_CHAN_INFO:IP 通道信息

```

struct{
    BYTE                byEnable;
    BYTE                byProType;
    BYTE                byZeroChan;
    BYTE                byPriority;
    BYTE                sUserName[NAME_LEN];
    BYTE                sPassword[PASSWD_LEN];
    BYTE                byDomain[MAX_DOMAIN_NAME];
    NET\_DVR\_IPADDR     struIP;
    WORD                wDVRPort;
    BYTE                byStreamType;
    BYTE                byOnline;
    DWORD              dwChannel;
    BYTE                byTransProtocol;
    BYTE                byLocalBackUp;
    WORD                wDirectLastTime;
    BYTE                byChanNo[CHAN_NO_LEN];
}NET_DVR_DIRECT_CONNECT_CHAN_INFO, *LPNET_DVR_DIRECT_CONNECT_CHAN_INFO;

```

Members

byEnable

是否启用

byProType

协议类型: 0- 私有协议(default), 1- 松下协议, 2- 索尼, 更多协议通过 [NET_DVR_GetIPCProtoList_V41](#) 获取

byZeroChan

是否是零通道: 0- 不是, 1- 是

byPriority

优先级, 取值范围: 1~5

sUserName

用户名

sPassword

密码

byDomain

设备域名

struIP

IP 地址

wDVRPort

端口号

byStreamType

码流类型: 0- 主码流, 1- 子码流

byOnline

在线状态（只读）：0- 不在线，1- 在线

dwChannel

通道号

byTransProtocol

协议类型：0- TCP，1- UDP，2- 多播

byLocalBackUp

本地备份：0- 不启用 CVR 本地备份，1- 启用 CVR 本地备份（即录像时的流一份保存在录像卷，一份保存在存档卷）

wDirectLastTime

导播持续时间，单位：s，取值范围通过能力集获取

byChanNo

通道编号，用于 VAG 取流

4.16 NET_DVR_ETHERNET_V30:以太网配置

```
struct{
    NET_DVR_IPADDR    struDVRIP;
    NET_DVR_IPADDR    struDVRIPMask;
    DWORD             dwNetInterface;
    WORD              wDVRPort;
    WORD              wMTU;
    BYTE              byMACAddr[MACADDR_LEN];
    BYTE              byRes[2];
}NET_DVR_ETHERNET_V30, *LPNET_DVR_ETHERNET_V30;
```

Members

struDVRIP

设备 IP 地址

struDVRIPMask

设备 IP 地址掩码

dwNetInterface

网络接口：1-10MBase-T；2-10MBase-T 全双工；3-100MBase-TX；4-100M 全双工；5-10M/100M/100M 自适应；6-1000M 全双工

wDVRPort

设备端口号

wMTU

MTU 设置，默认 1500

byMACAddr

设备物理地址

byRes

保留

Remarks

MTU 的设置范围为 500-9676，若 MTU 设置过小客户端将无法注册到设备，并且客户端预览、回放、配置参数也会失败。

4.17 NET_DVR_INDOOR_UNIT_DEVICEID:室内机编号参数

```
struct{
    WORD    wFloorNumber;
    WORD    wRoomNumber;
    WORD    wDevIndex;
    BYTE    byRes[122];
}NET_DVR_INDOOR_UNIT_DEVICEID,*LPNET_DVR_INDOOR_UNIT_DEVICEID;
```

Members

wFloorNumber
层号

wRoomNumber
房间号

wDevIndex
室内机编号，取值范围：0~10

byRes
保留

4.18 NET_DVR_INDOOR_UNIT_OPERATION_TIME_CFG:室内机操作时间配置

```
struct{
    DWORD   dwMaxMonitoringTime;
    DWORD   dwMaxRingTime;
    DWORD   dwCallForwardingTime;
    BYTE    byRes[116];
}NET_DVR_INDOOR_UNIT_OPERATION_TIME_CFG,*LPNET_DVR_INDOOR_UNIT_OPERATION_TIME_CFG;
```

Members

dwMaxMonitoringTime
最大监视时间，取值范围：[10,60]，单位：秒

dwMaxRingTime
最大振铃时间，取值范围：[15,60]，单位：秒

dwCallForwardingTime
呼叫转移超时时间，取值范围：[0,20]，单位：秒

byRes
保留，置为 0

4.19 NET_DVR_INDOOR_UNIT_RELATEDDEV:室内机关联的网络设备

```
struct{
    NET\_DVR\_IPADDR    struOutdoorUnit;
```

```

NET_DVR_IPADDR    struManageUnit;
NET_DVR_IPADDR    struSIPServer;
NET_DVR_IPADDR    struAgainUnit;
BYTE              byOutdoorType;
BYTE              byOutInConnectMode;
BYTE              byIndoorConnectMode;
BYTE              byRes1;
NET_DVR_IPADDR    struIndoorUnit;
BYTE              byRes[300];
}NET_DVR_INDOOR_UNIT_RELATEDDEV,*LPNET_DVR_INDOOR_UNIT_RELATEDDEV;

```

Members

struOutdoorUnit

主门口机 IP

struManageUnit

管理机 IP

struSIPServer

SIP 服务器 IP

struAgainUnit

二次确认机 IP

byOutdoorType

主门口机类型，0 保留，1-主单元门口机，2-主别墅门口机

byOutInConnectMode

门口机与室内分机组网模式：1-门口机与室内分机同一局域网,2-门口机与室内分机不同局域网

byIndoorConnectMode

室内主机与室内分机组网模式：1-通过无线网卡、2-通过有线网卡

byRes1

保留，置为 0

struIndoorUnit

室内主机 IP

byRes

保留，置为 0

4.20 NET_DVR_OUTDOOR_UNIT_RELATEDDEV:门口机/围墙机/别墅门口 机关联网络设备参数

```

struct{
NET_DVR_IPADDR    struMainOutdoorUnit;
NET_DVR_IPADDR    struManageUnit;
NET_DVR_IPADDR    struSIPServer;
BYTE              byRes[592];
}NET_DVR_OUTDOOR_UNIT_RELATEDDEV,*LPNET_DVR_OUTDOOR_UNIT_RELATEDDEV;

```

Members

struMainOutdoorUnit

主门口机 IP 地址，副门口机时有效

struManageUnit

管理机 IP 地址

struSIPServer

SIP 服务器 IP 地址，副门口机时无效

byRes

保留，置为 0

4.21 NET_DVR_IPADDR:IP 地址

```
struct{
    char    slpV4[16];
    BYTE    slpV6[128];
}NET_DVR_IPADDR, *LPNET_DVR_IPADDR;
```

Members

slpV4

设备 IPv4 地址

slpV6

设备 IPv6 地址

4.22 NET_DVR_JPEGPARA:JPEG 图像信息

```
struct{
    WORD    wPicSize;
    WORD    wPicQuality;
}NET_DVR_JPEGPARA,*LPNET_DVR_JPEGPARA;
```

Members

wPicSize

图片尺寸: 0-CIF, 1-QCIF, 2-D1, 3-UXGA(1600x1200), 4-SVGA(800x600), 5-HD720p(1280x720), 6-VGA, 7-XVGA, 8-HD900p, 9-HD1080, 10-2560*1920, 11-1600*304, 12-2048*1536, 13-2448*2048, 14-2448*1200, 15-2448*800, 16-XGA(1024*768), 17-SXGA(1280*1024), 18-WD1(960*576/960*480),19-1080i

wPicQuality

图片质量系数: 0-最好, 1-较好, 2-一般

4.23 NET_DVR_LOCAL_ABILITY_PARSE_CFG:能力集解析库配置

```
struct{
    BYTE    byEnableAbilityParse;
    BYTE    byRes[127];
}NET_DVR_LOCAL_ABILITY_PARSE_CFG,*LPNET_DVR_LOCAL_ABILITY_PARSE_CFG;
```

Members

byEnableAbilityParse

使用能力集解析库：0-不使用，1-使用，默认不使用

byRes

保留，置为 0

Remarks

模拟能力集默认禁用，调用该接口可以启用模拟能力集，支持获取设备各种能力。如果需要获取能力集（NET_DVR_GetDeviceAbility），可以调用此接口来启用模拟能力集，并且需要加载 LocalXml.zip（要求和 SDK 库文件放在同一个目录下）。

4.24 NET_DVR_LOCAL_BYTE_ENCODE_CONVERT: 字符编码转换参数

```
struct{
    CHAR_ENCODE_CONVERT    fnCharConvertCallBack;
    BYTE                    byRes[256];
}NET_DVR_LOCAL_BYTE_ENCODE_CONVERT,*LPNET_DVR_LOCAL_BYTE_ENCODE_CONVERT;
```

Members**fnCharConvertCallBack**

字符编码转换回调函数

byRes

保留，置为 0

Callback Function

```
typedef int(CALLBACK *CHAR_ENCODE_CONVERT)(
    char        *pInput,
    DWORD       dwInputLen,
    DWORD       dwInEncodeType,
    char        *pOutput,
    DWORD       dwOutputLen,
    DWORD       dwOutEncodeType
);
```

Callback Function Parameters**pInput**

[in] 输入字符串，内存由 SDK 申请，字符串数据也由 SDK 提供

dwInputLen

[in] 输入字符串缓冲区大小

dwInEncodeType

[in] 输入的字符编码格式：0- 无字符编码信息(老设备)，1- GB2312(简体中文)，2- GBK，3- BIG5(繁体中文)，4- Shift_JIS(日文)，5- EUC-KR(韩文)，6- UTF-8，7- ISO8859-1，8- ISO8859-2，9- ISO8859-3，...，依次类推，21- ISO8859-15(西欧)

pOutput

[out] 输出字符串，内存由 SDK 申请，存放使用用户字符编码接口转换之后的字符串

dwOutputLen

[out] 输出字符串缓冲区大小

dwOutEncodeType

[out] 输出字符编码格式：0- 无字符编码信息(老设备)，1- GB2312(简体中文)，2- GBK，3- BIG5(繁体

中文), 4- Shift_JIS(日文), 5- EUC-KR(韩文), 6- UTF-8, 7- ISO8859-1, 8- ISO8859-2, 9- ISO8859-3, ..., 依次类推, 21- ISO8859-15(西欧)

Remarks

- 回调函数的返回值: -1 表示失败, 0 表示成功 (内存足够存放转换以后的字符串)。
- 设备的字符编码类型在登录接口返回, 对应 NET_DVR_DEVICEINFO_V40 结构体中的参数 byCharEncodeType。SDK 内部需要字符编码转换时, SDK 默认使用 libiconv 库进行类型转换。如果不想使用 libiconv 编码库, 可以调用 NET_DVR_SetSDKLocalCfg(类型: NET_SDK_LOCAL_CFG_TYPE_BYTE_ENCODE) 设置字符转码回调函数, 告知 SDK 用户自己的字符编码接口, 然后 SDK 将使用用户提供的字符编码接口进行字符串处理。

4.25 NET_DVR_LOCAL_CHECK_DEV:设备在线巡检参数

```
struct{
    DWORD    dwCheckOnlineTimeout;
    DWORD    dwCheckOnlineNetFailMax;
    BYTE     byRes[256];
}NET_DVR_LOCAL_CHECK_DEV, *LPNET_DVR_LOCAL_CHECK_DEV;
```

Members

dwCheckOnlineTimeout

巡检时间间隔, 单位: ms, 取值范围: 30s~120s, 0 表示用默认值(120s), 推荐设置 30s

dwCheckOnlineNetFailMax

由于网络原因失败的最大累加次数, 达到该次数, SDK 才回调用户异常消息, 0 表示使用默认值 1, 推荐设置 3 次

byRes

保留, 置为 0

Remarks

- SDK 按照该结构体中的时间间隔对设备进行自动巡检, 巡检过程中如果连失败或者重连成功在 [NET_DVR_SetExceptionCallBack_V30](#) 设置的异常消息回调函数中返回, 对应异常消息类型为: EXCEPTION_EXCHANGE、RESUME_EXCHANGE。
- 推荐设置 30s 时间间隔、3 次, 即心跳间隔为 1.5 分钟。
- 通过 [NET_DVR_RemoteControl](#) (命令: NET_DVR_CHECK_USER_STATUS) 可以手动检测设备在线状态。

4.26 NET_DVR_LOCAL_MEM_POOL_CFG:内存池本地配置

```
struct{
    DWORD    dwAlarmMaxBlockNum;
    DWORD    dwAlarmReleaseInterval;
    BYTE     byRes[60];
}NET_DVR_LOCAL_MEM_POOL_CFG, *LPNET_DVR_LOCAL_MEM_POOL_CFG;
```

Members

dwAlarmMaxBlockNum

报警模块内存池最多向系统申请的内存块 (block) 个数, 每个 block 为 64MB, 超过这个上限则不向系统申请, 0 表示无上限

dwAlarmReleaseInterval

报警模块空闲内存释放的间隔，单位：秒，为 0 表示不释放空闲的内存

byRes

保留，置为 0

4.27 NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG:按模块配置超时时间

```
struct{
    DWORD    dwPreviewTime;
    DWORD    dwAlarmTime;
    DWORD    dwVodTime;
    DWORD    dwElse;
    BYTE     byRes[512];
}NET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG,*LPNET_DVR_LOCAL_MODULE_RECV_TIMEOUT_CFG;
```

Members*dwPreviewTime*

预览模块超时时间，单位：毫秒，取值范围：0~3000,000

dwAlarmTime

报警模块超时时间，单位：毫秒，取值范围：0~3000,000

dwVodTime

回放模块超时时间，单位：毫秒，取值范围：0~3000,000

dwElse

其他模块超时时间，单位：毫秒，取值范围：0~3000,000

byRes

保留，置为 0

4.28 NET_DVR_LOCAL_TALK_MODE_CFG:对讲模式配置

```
struct{
    BYTE     byEnableAbilityParse;
    BYTE     byRes[127];
}NET_DVR_LOCAL_TALK_MODE_CFG,*LPNET_DVR_LOCAL_TALK_MODE_CFG;
```

Members*byTalkMode*

对讲模式：0- 使用对讲库（默认），1- 使用 windows api 模式

byRes

保留，置为 0

Remarks

V4.2.2.5 及以前版本 SDK 均采用 windows API 实现相关功能。之后版本默认使用语音对讲库的方式，语音对讲库模式下必须加载 AudioIntercom.dll 和 OpenAL32.dll。

4.29 NET_DVR_LOCAL_TCP_PORT_BIND_CFG:本地 TCP 端口绑定配置

```
struct{
    WORD    wLocalBindTcpMinPort;
    WORD    wLocalBindTcpMaxPort;
    BYTE    byRes[60];
}NET_DVR_LOCAL_TCP_PORT_BIND_CFG,*LPNET_DVR_LOCAL_TCP_PORT_BIND_CFG;
```

Members

wLocalBindTcpMinPort

本地绑定 TCP 最小端口

wLocalBindTcpMaxPort

本地绑定 TCP 最大端口

byRes

保留，置为 0

Remarks

端口绑定的策略是：给一个端口段，可以保证使用的端口都是在这个段里（多播除外），但不能保证每一个段内的端口都用到，因为是循环利用的；端口池中取出的端口会去尝试绑定，如果被占用了，将取下一个，如果段内每一个都绑定不了，则连接操作返回失败。建议最好不要设置系统预留的端口（1-1024），比如 80 等。

设置的最大端口应该大于等于最小端口，[0,0]表示清除绑定，[0, 非 0]将设置失败，因为 0 不能进行绑定。

4.30 NET_DVR_LOCAL_UDP_PORT_BIND_CFG:本地 UDP 端口绑定配置

```
struct{
    WORD    wLocalBindUdpMinPort;
    WORD    wLocalBindUdpMaxPort;
    BYTE    byRes[60];
}NET_DVR_LOCAL_UDP_PORT_BIND_CFG,*LPNET_DVR_LOCAL_UDP_PORT_BIND_CFG;
```

Members

wLocalBindUdpMinPort

本地绑定 UDP 最小端口

wLocalBindUdpMaxPort

本地绑定 UDP 最大端口

byRes

保留，置为 0

Remarks

端口绑定的策略是：给一个端口段，可以保证使用的端口都是在这个段里（多播除外），但不能保证每一个段内的端口都用到，因为是循环利用的；端口池中取出的端口会去尝试绑定，如果被占用了，将取下一个，如果段内每一个都绑定不了，则连接操作返回失败。建议最好不要设置系统预留的端口（1-1024），比如 80 等。

设置的最大端口应该大于等于最小端口，[0,0]表示清除绑定，[0, 非 0]将设置失败，因为 0 不能进行绑定。

4.31 NET_DVR_LOG_V30:日志信息

```

struct{
    NET\_DVR\_TIME    strLogTime;
    DWORD           dwMajorType;
    DWORD           dwMinorType;
    BYTE            sPanelUser[MAX_NAMELEN];
    BYTE            sNetUser[MAX_NAMELEN];
    NET\_DVR\_IPADDR  struRemoteHostAddr;
    DWORD           dwParaType;
    DWORD           dwChannel;
    DWORD           dwDiskNumber;
    DWORD           dwAlarmInPort;
    DWORD           dwAlarmOutPort;
    DWORD           dwInfoLen;
    char            sInfo[LOG_INFO_LEN];
}NET_DVR_LOG_V30,*LPNET_DVR_LOG_V30;

```

Members

strLogTime

日志时间

dwMajorType

报警主类型，定义请参见[设备日志主类型](#)

dwMinorType

报警次类型，根据不同的主类型的次类型定义请参见[设备日志次类型](#)

sPanelUser

操作面板的用户名

sNetUser

网络操作的用户名

struRemoteHostAddr

远程主机地址

dwParaType

对于 DS-90xx 设备，当日志次类型为 MINOR_START_VT 或者 MINOR_STOP_VT 时，表示语音对讲端口号。

当日志的主类型为 MAJOR_OPERATION=03（操作），且次类型为 MINOR_LOCAL_CFG_PARM=0x52（本地配置参数）或 MINOR_REMOTE_GET_PARM=0x76（远程获得参数）或

MINOR_REMOTE_CFG_PARM=0x77（远程配置参数）时，该参数类型有效，其含义如下：

宏定义	宏定义值	含义
PARA_VIDEOOUT	0x1	视频输出结构配置
PARA_IMAGE	0x2	图像参数结构配置
PARA_ENCODE	0x4	压缩参数结构配置
PARA_NETWORK	0x8	网络参数结构配置
PARA_ALARM	0x10	报警参数结构配置

PARA_EXCEPTION	0x20	异常参数结构配置
PARA_DECODER	0x40	解码器参数结构配置
PARA_RS232	0x80	RS232 参数结构配置
PARA_PREVIEW	0x100	本地预览参数结构配置
PARA_SECURITY	0x200	用户权限参数结构配置
PARA_DATETIME	0x400	本地系统配置
PARA_FRAMEYPE	0x800	帧信息参数结构配置

dwChannel

通道号

dwDiskNumber

硬盘号

dwAlarmInPort

报警输入端口

dwAlarmOutPort

报警输出端口

dwInfoLen

日志附加信息长度

sInfo

日志附加信息

4.32 NET_DVR_MANAGE_UNIT_DEVICEID: 管理机编号参数

```
struct{
    WORD    wPeriod;
    WORD    wDevIndex;
    BYTE    byRes[124];
}NET_DVR_MANAGE_UNIT_DEVICEID,*LPNET_DVR_MANAGE_UNIT_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wDevIndex

管理机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

4.33 NET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG: 管理机操作时间配置

```
struct{
    DWORD    dwMaxMonitoringTime;
```

```

DWORD    dwMaxRingTime;
DWORD    dwMaxTalkTime;
BYTE     byRes[116];
}NET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG,*LPNET_DVR_MANAGE_UNIT_OPERATION_TIME_CFG;

```

Members

dwMaxMonitoringTime

最大监视时间，范围：[10,60]，单位：秒

dwMaxRingTime

最大振铃时间，范围：[15,60]，单位：秒

dwMaxTalkTime

最大通话时间，单位：秒

byRes

保留，置为 0

4.34 NET_DVR_MANAGE_UNIT_RELATEDDEV:管理机关联网络设备参数

```

struct{
    NET_DVR_IPADDR    struSIPServer;
    BYTE              byRes[880];
}NET_DVR_MANAGE_UNIT_RELATEDDEV,*LPNET_DVR_MANAGE_UNIT_RELATEDDEV;

```

Members

struSIPServer

SIP 服务器 IP 地址

byRes

保留，置为 0

4.35 NET_DVR_NETCFG_V30:网络配置

```

struct{
    DWORD                dwSize;
    NET_DVR_ETHERNET_V30 struEtherNet[MAX_ETHERNET];
    NET_DVR_IPADDR      struRes1[2];
    NET_DVR_IPADDR      struAlarmHostIpAddr;
    WORD                 wRes2[2];
    WORD                 wAlarmHostIpPort;
    BYTE                 byUseDhcp;
    BYTE                 byRes3;
    NET_DVR_IPADDR      struDnsServer1IpAddr;
    NET_DVR_IPADDR      struDnsServer2IpAddr;
    BYTE                 byIpResolver[MAX_DOMAIN_NAME];
    WORD                 wIpResolverPort;
    WORD                 wHttpPortNo;
    NET_DVR_IPADDR      struMulticastIpAddr;
}

```

```

NET_DVR_IPADDR          struGatewayIpAddr;
NET_DVR_PPPOECFG        struPPPoE;
BYTE                    byRes[64];
}NET_DVR_NETCFG_V30,*LPNET_DVR_NETCFG_V30;

```

Members

dwSize

大小

struEtherNet

以太网口

struRes1

保留，置为 0

struAlarmHostIpAddr

报警主机 IP 地址

wRes2

保留，置为 0

wAlarmHostIpPort

报警主机端口号

byUseDhcp

是否启用 DHCP: 0xff-无效; 0-不启用; 1-启用

byRes3

保留，置为 0

struDnsServer1IpAddr

域名服务器 1 的 IP 地址

struDnsServer2IpAddr

域名服务器 2 的 IP 地址

byIpResolver

IP 解析服务器域名或 IP 地址 (8000 设备不支持域名)

wIpResolverPort

IP 解析服务器端口号

wHttpPortNo

HTTP 端口号

struMulticastIpAddr

多播组地址

struGatewayIpAddr

网关地址

struPPPoE

PPPoE 参数

byRes

保留，置为 0

Remarks

3.0 协议以下的设备，参数 *byUseDhcp* 为 0xff-无效，将其 IP 地址填成空，设备会自动去获取 DHCP。

4.36 NET_DVR_NOTICE_DATA:公告数据下发参数

```

struct{
    DWORD                dwSize;
    NET\_DVR\_TIME\_EX      struTime;
    BYTE                 byNoticeNumber[MAX_NOTICE_NUMBER_LEN];
    BYTE                 byNoticeTheme[MAX_NOTICE_THEME_LEN];
    BYTE                 byNoticeDetail[MAX_NOTICE_DETAIL_LEN];
    BYTE                 byLevel;
    BYTE                 byPicNum;
    BYTE                 byRes1[2];
    NET\_DVR\_NOTICE\_PIC  struNoticePic[MAX_NOTICE_PIC_NUM];
    BYTE                 byRes2[128];
}NET_DVR_NOTICE_DATA, *LPNET_DVR_NOTICE_DATA;

```

Members

dwSize

结构体大小

struTime

公告时间

byNoticeNumber

公告编号

byNoticeTheme

公告主题

byNoticeDetail

公告详情

byLevel

公告等级：1- 广告类信息，2- 物业信息，3- 报警类信息，4- 通知类信息

byPicNum

公告图片数量

byRes1

保留，置为 0

struNoticePic

公告图片信息

byRes2

保留，置为 0

Remarks

- 公告数据下发参数能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：IP_VIEW_DEV_ABILITY）中节点<NoticeData>。

4.37 NET_DVR_NOTICE_PIC:公告图片信息

```

struct{
    BYTE                *pPicData;

```

```

DWORD   dwPicDataLen;
BYTE    byRes[32];
}NET_DVR_NOTICE_PIC,*LPNET_DVR_NOTICE_PIC;

```

Members

pPicData

缓冲区指针，保存图片数据（单张图片长度最大 512KB，默认 jpeg 格式）

dwPicDataLen

图片数据长度

byRes

保留，置为 0

4.38 NET_DVR_NOTICEDATA_RECEIPT_INFO:公告信息阅读回执信息

```

struct{
    BYTE    byNoticeNumber[MAX_NOTICE_NUMBER_LEN];
    BYTE    byRes[224];
}NET_DVR_NOTICEDATA_RECEIPT_INFO,*LPNET_DVR_NOTICEDATA_RECEIPT_INFO;

```

Members

byNoticeNumber

公告编号

byRes

保留，置为 0

4.39 NET_DVR_NTTPARA:网络应用参数(NTP)

```

struct{
    BYTE    sNTPServer[64];
    WORD    wInterval;
    BYTE    byEnableNTP;
    signed char    cTimeDifferenceH;
    signed char    cTimeDifferenceM;
    BYTE    res1;
    WORD    wNtpPort;
    BYTE    res2[8];
}NET_DVR_NTTPARA,*LPNET_DVR_NTTPARA;

```

Members

sNTPServer

NTP 服务器域名或者 IP 地址

wInterval

校时间隔时间，以分钟或小时为单位（通过能力集 [NET_DVR_GetDeviceAbility](#) 获取，对应网络应用参数能力集：DEVICE_NETAPP_ABILITY）

byEnableNTP

NTP 校时是否启用：0—否，1—是

cTimeDifferenceH

与国际标准时间的时差（小时），-12 ... +13

cTimeDifferenceM

与国际标准时间的时差（分钟），0, 30, 45

res1

保留，置为 0

wNtpPort

NTP 服务器端口，设备默认为 123

res2

保留，置为 0

4.40 NET_DVR_OPERATION_AUTH:操作权限验证参数

```
struct{
    DWORD    dwSize;
    BYTE     byPassword[PASSWD_LEN];
    BYTE     byRes[128];
}NET_DVR_OPERATION_AUTH,*LPNET_DVR_OPERATION_AUTH;
```

Members

dwSize

结构体大小

byPassword

操作密码

byRes

保留，置为 0

4.41 NET_DVR_OUTDOOR_FENCE_DEVICEID:围墙机编号参数

```
struct{
    WORD     wPeriod;
    WORD     wDevIndex;
    BYTE     byRes[124];
}NET_DVR_OUTDOOR_FENCE_DEVICEID,*LPNET_DVR_OUTDOOR_FENCE_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wDevIndex

围墙机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

4.42 NET_DVR_OUTDOOR_UNIT_DEVICEID: 门口机编号参数

```
struct{
    WORD    wPeriod;
    WORD    wBuildingNumber;
    WORD    wUnitNumber;
    WORD    wFloorNumber;
    WORD    wDevIndex;
    BYTE    byRes[118];
}NET_DVR_OUTDOOR_UNIT_DEVICEID,*LPNET_DVR_OUTDOOR_UNIT_DEVICEID;
```

Members

wPeriod

期号，取值范围：[0,9]

wBuildingNumber

楼号

wUnitNumber

单元号

wFloorNumber

层号

wDevIndex

门口机序号，每一层中门口机的序号唯一，从 0 开始

byRes

保留，置为 0

4.43 NET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG: 门口机/围墙机/别墅门口机/二次确认机操作时间配置

```
struct{
    DWORD    dwMaxMessageTime;
    DWORD    dwMaxTalkTime;
    BYTE    byRes[120];
}NET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG,*LPNET_DVR_OUTDOOR_UNIT_OPERATION_TIME_CFG;
```

Members

dwMaxMessageTime

最大留言时间，范围：[30,60]，单位：秒

dwMaxTalkTime

最大通话时间，范围：[90,120]，单位：秒

byRes

保留，置为 0

4.44 NET_DVR_PICCFG_V30:通道图像

```

struct{
    DWORD                dwSize;
    BYTE                 sChanName[NAME_LEN];
    DWORD                dwVideoFormat;
    char                 reservedData[64];
    DWORD                dwShowChanName;
    WORD                 wShowNameTopLeftX;
    WORD                 wShowNameTopLeftY;
    NET_DVR_VILOST_V30  struVILost;
    NET_DVR_VILOST_V30  struRes;
    NET_DVR_MOTION_V30  struMotion;
    NET_DVR_HIDEALARM_V30 struHideAlarm;
    DWORD                dwEnableHide;
    NET_DVR_SHELTER     struShelter[MAX_SHELTERNUM];
    DWORD                dwShowOsd;
    WORD                 wOSDTopLeftX;
    WORD                 wOSDTopLeftY;
    BYTE                 byOSDType;
    BYTE                 byDispWeek;
    BYTE                 byOSDAattrib;
    BYTE                 byHourOsdType;
    BYTE                 byFontSize;
    BYTE                 byOSDColorType;
    BYTE                 byAlignment;
    BYTE                 byOSDMilliSecondEnable;
    NET_DVR_RGB_COLOR   struOsdColor;
    BYTE                 byRes[56];
}NET_DVR_PICCFG_V30,*LPNET_DVR_PICCFG_V30;

```

Members

dwSize

结构体大小

sChanName

通道名称

dwVideoFormat

视频制式：1-NTSC，2-PAL

reservedData

保留，置为 0

dwShowChanName

预览的图象上是否显示通道名称：0-不显示，1-显示（区域大小 704*576）

wShowNameTopLeftX

通道名称显示位置的 x 坐标

wShowNameTopLeftY

通道名称显示位置的 y 坐标

struVILost

视频信号丢失报警参数

struRes

保留，置为 0

struMotion

移动侦测报警参数

struHideAlarm

遮挡报警参数

dwEnableHide

是否启动遮挡：0-否，1-是

struShelter

遮挡区域参数

dwShowOsd

预览的图象上是否显示 OSD：0-不显示，1-显示（区域大小 704*576）

wOSDTopLeftX

OSD 的 x 坐标

wOSDTopLeftY

OSD 的 y 坐标

byOSDType

OSD 类型(年月日格式):

0—XXXX-XX-XX 年月日

1—XX-XX-XXXX 月日年

2—XXXX 年 XX 月 XX 日

3—XX 月 XX 日 XXXX 年

4—XX-XX-XXXX 日月年

5—XX 日 XX 月 XXXX 年

byDispWeek

是否显示星期：0-不显示，1-显示

byOSDAttrib

OSD 属性（透明/闪烁）:

1—透明，闪烁

2—透明，不闪烁

3—闪烁，不透明

4—不透明，不闪烁

byHourOsdType

小时制：0 表示 24 小时制，1 表示 12 小时制或 am/pm

byFontSize

字体大小：0- 小，1- 中，2- 大

byOSDColorType

OSD 颜色模式：0- 默认（黑白），1-自定义(颜色见 struOsdColor)

byAlignment

对齐方式：0- 自适应，1- 右对齐

byOSDMilliSecondEnable

视频叠加时间支持毫秒（智能交通摄像机 V4.1 及以后版本支持）：0- 不叠加，1- 叠加

struOsdColor

OSD 自定义颜色

byRes

保留，置为 0

Remarks

设备通道名称、OSD、移动侦测、视频遮挡等图像参数配置能力，对应设备图像参数能力集（VideoPicAbility），相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：DEVICE_VIDEOPIC_ABILITY。

4.45 NET_DVR_PPPOECFG:PPPoE 配置

```
struct{
    DWORD          dwPPPOE;
    BYTE           sPPPoEUser[NAME_LEN];
    char           sPPPoEPassword[PASSWD_LEN];
    NET\_DVR\_IPADDR struPPPoEIP;
}NET_DVR_PPPOECFG, *LPNET_DVR_PPPOECFG;
```

Members**dwPPPOE**

是否启用 PPPoE：0-不启用，1-启用

sPPPoEUser

PPPoE 用户名

sPPPoEPassword

PPPoE 密码

struPPPoEIP

PPPoE IP 地址

4.46 NET_DVR_PREVIEWINFO:预览参数

```
struct{
    LONG          lChannel;
    DWORD         dwStreamType;
    DWORD         dwLinkMode;
    HWND          hPlayWnd;
    BOOL          bBlocked;
    BOOL          bPassbackRecord;
    BYTE          byPreviewMode;
    BYTE          byStreamID[STREAM_ID_LEN];
    BYTE          byProtoType;
    BYTE          byRes[222];
}NET_DVR_PREVIEWINFO, *LPNET_DVR_PREVIEWINFO;
```

Members

IChannel

通道号，设备模拟通道号从 1 开始。

dwStreamType

码流类型：0-主码流，1-子码流，2-码流 3，3-虚拟码流，以此类推

dwLinkMode

连接方式：0- TCP 方式，1- UDP 方式，2- 多播方式，3- RTP 方式，4-RTP/RTSP，5-RSTP/HTTP

hPlayWnd

播放窗口的句柄，为 NULL 表示不解码显示。

bBlocked

0- 非阻塞取流，1- 阻塞取流。如果阻塞取流，SDK 内部 connect 失败将会有 5s 的超时才能够返回，不适合于轮询取流操作。

bPassbackRecord

0-不启用录像回传，1-启用录像回传。ANR 断网补录功能，客户端和设备之间网络异常恢复之后自动将前端数据同步过来，需要设备支持。

byPreviewMode

预览模式：0- 正常预览，1- 延迟预览

byStreamID

流 ID，为字母、数字和"_"的组合，IChannel 为 0xffffffff 时启用此参数

byProtoType

应用层取流协议：0- 私有协议，1- RTSP 协议。主子码流支持的取流协议通过登录返回结构参数 NET_DVR_DEVICEINFO_V30 的 byMainProto、bySubProto 值得知。设备同时支持私协议和 RTSP 协议时，该参数才有效，默认使用私有协议，可选 RTSP 协议。

byRes

保留

Remarks

- 该结构体中可以设置当前预览操作是否阻塞（通过 bBlocked 参数设置）。若设为不阻塞，表示发起与设备的连接就认为连接成功，如果发生码流接收失败、播放失败等情况以预览异常的方式通知上层。在循环播放的时候可以减短停顿的时间，与 NET_DVR_RealPlay 处理一致。若设为阻塞，表示直到播放操作完成才返回成功与否。
- dwStreamType、bPassbackRecord、byPreviewMode、byStreamID 这些参数的取值需要设备支持。
- NET_DVR_RealPlay_V40 支持多播方式预览（dwLinkMode 设为 2），不需要传多播组地址，底层自动从设备获取已配置的多播组地址（NET_DVR_NETCFG_V30->struMulticastIpAddr）并以该多播组地址实现多播。
- Linux 64 位系统不支持软解码功能，因此需要将窗口句柄传 NULL，设置回调函数，只取流不解码显示。

4.47 NET_DVR_PRIVILEGE_PASSWORD_CFG:权限密码配置

```
struct{
    DWORD    dwSize;
    BYTE     byPwdType;
    BYTE     byRes1[3];
    BYTE     byOldPassword[PASSWD_LEN];
    BYTE     byNewPassword[PASSWD_LEN];
    BYTE     byRes2[128];
}NET_DVR_PRIVILEGE_PASSWORD_CFG, *LPNET_DVR_PRIVILEGE_PASSWORD_CFG;
```

Members**dwSize**

结构体大小

byPwdType

密码类型，具体定义如下：

```
enum _PRIVILEGE_PASSWORD_TYPE_ENUM_{
    ENUM_PRIVILEGE_PASSWORD_ENGINEERING        = 1,
    ENUM_PRIVILEGE_PASSWORD_SETUPALARM        = 2,
    ENUM_PRIVILEGE_PASSWORD_HOUSEHOLDER_UNLOCK = 3,
    ENUM_PRIVILEGE_PASSWORD_ANTI_HIJACKING    = 4
}PRIVILEGE_PASSWORD_TYPE_ENUM
```

ENUM_PRIVILEGE_PASSWORD_ENGINEERING

工程密码

ENUM_PRIVILEGE_PASSWORD_SETUPALARM

布/撤防密码

ENUM_PRIVILEGE_PASSWORD_HOUSEHOLDER_UNLOCK

户主开锁密码

ENUM_PRIVILEGE_PASSWORD_ANTI_HIJACKING

防劫持密码

byRes1

保留，置为 0

byOldPassword

旧密码

byNewPassword

新密码

byRes2

保留，置为 0

Remarks

- 权限密码配置能力，对应 IP 可视对讲主机能力集（接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：IP_VIEW_DEV_ABILITY）中节点<PrivilegePwd>。

4.48 NET_DVR_PTZCFG:云台协议

```
struct{
    DWORD          dwSize;
    NET\_DVR\_PTZ\_PROTOCOL struPtz[PTZ_PROTOCOL_NUM];
    DWORD          dwPtzNum;
    BYTE           byRes[8];
}NET_DVR_PTZCFG, *LPNET_DVR_PTZCFG;
```

Members**dwSize**

结构体大小

struPtz

协议信息，最多 200 种

dwPtzNum

有效的 PTZ 协议数目，从 0 开始（即总数为该值加 1）

byRes

保留，置为 0

4.49 NET_DVR_PTZ_PROTOCOL:云台协议信息

```
struct{
    DWORD    dwType;
    BYTE     byDescribe[DESC_LEN];
}NET_DVR_PTZ_PROTOCOL, *LPNET_DVR_PTZ_PROTOCOL;
```

Members

dwType

协议类型值

byDescribe

协议描述符

4.50 NET_DVR_PU_STREAM_URL:URL 取流配置

```
struct{
    BYTE     byEnable;
    BYTE     strURL[240];
    BYTE     byTransPortocol;
    WORD     wIPID;
    BYTE     byChannel;
    BYTE     byRes[7];
}NET_DVR_PU_STREAM_URL,*LPNET_DVR_PU_STREAM_URL;
```

Members

byEnable

是否启用：0- 禁用，1- 启用

strURL

取流 URL 路径

byTransPortocol

传输协议类型：0-TCP，1-UDP

wIPID

设备 ID 号，wIPID = iDevInfoIndex + iGroupNO*64 +1

byChannel

设备通道号

byRes

保留，置为 0

4.51 NET_DVR_PU_STREAM_URL_CFG:URL 取流路径构成参数

```

struct{
    BYTE    byEnable;
    BYTE    byRes[3];
    BYTE    byStreamMediaIP[64];
    WORD    wStreamMediaPort;
    BYTE    byTransmitType;
    BYTE    byRes1[33];
    BYTE    byDevIP[64];
    WORD    wDevPort;
    BYTE    byChannel;
    BYTE    byTransMode;
    BYTE    byProType;
    BYTE    byTransProtocol;
    BYTE    byRes3[2];
    BYTE    sUserName[NAME_LEN];
    BYTE    sPassWord[PASSWD_LEN];
    BYTE    byRes2[28];
}NET_DVR_PU_STREAM_URL_CFG,*LPNET_DVR_PU_STREAM_URL_CFG;

```

Members

byEnable

是否启用：0- 禁用，1- 启用

byRes

保留，置为 0

byStreamMediaIP

流媒体 IP 地址

wStreamMediaPort

流媒体端口

byTransmitType

流媒体传输协议：0- TCP，1- UDP，2-RTSP

byRes1

保留，置为 0

byDevIP

设备 IP 地址

wDevPort

设备端口号

byChannel

设备通道号

byTransMode

传输模式：0- 主码流，1- 子码流

byProType

厂家协议类型：0- 私有协议，1- 大华，2- 汉邦，3- 郎驰，4- 蓝色星际

byTransProtocol

传输协议类型：0- TCP，1- UDP，2- 多播方式，3- RTP

byRes3

保留，置为 0

sUserName

设备登陆用户名

sPassWord

设备登陆密码

byRes

保留，置为 0

4.52 NET_DVR_SCHEDTIME:起止时间段参数

```
struct{
    BYTE    byStartHour;
    BYTE    byStartMin;
    BYTE    byStopHour;
    BYTE    byStopMin;
}NET_DVR_SCHEDTIME, *LPNET_DVR_SCHEDTIME;
```

Members**byStartHour**

开始时间：时

byStartMin

开始时间：分

byStopHour

结束时间：时

byStopMin

结束时间：分

4.53 NET_DVR_SDKABL:SDK 功能信息

```
struct{
    DWORD    dwMaxLoginNum;
    DWORD    dwMaxRealPlayNum;
    DWORD    dwMaxPlayBackNum;
    DWORD    dwMaxAlarmChanNum;
    DWORD    dwMaxFormatNum;
    DWORD    dwMaxFileSearchNum;
    DWORD    dwMaxLogSearchNum;
    DWORD    dwMaxSerialNum;
    DWORD    dwMaxUpgradeNum;
    DWORD    dwMaxVoiceComNum;
    DWORD    dwMaxBroadCastNum;
```

```
DWORD    dwRes[10];
}NET_DVR_SDKABL,*LPNET_DVR_SDKABL;
```

Members

dwMaxLoginNum

最大注册用户数

dwMaxRealPlayNum

最大实时预览的路数

dwMaxPlayBackNum

最大回放或下载的路数

dwMaxAlarmChanNum

最大建立报警通道的路数

dwMaxFormatNum

最大硬盘格式化的路数

dwMaxFileSearchNum

最大文件搜索的路数

dwMaxLogSearchNum

最大日志搜索的路数

dwMaxSerialNum

最大建立透明通道的路数

dwMaxUpgradeNum

最大升级的路数

dwMaxVoiceComNum

最大语音转发的路数

dwMaxBroadCastNum

最大语音广播的路数

dwRes

保留，置为 0

4.54 NET_DVR_SDKLOCAL_CFG:模拟能力集参数

```
struct{
    BYTE    byEnableAbilityParse;
    BYTE    byVoiceComMode;
    BYTE    byRes[510];
}NET_DVR_SDKLOCAL_CFG,*LPNET_DVR_SDKLOCAL_CFG;
```

Members

byEnableAbilityParse

使用能力集解析库：0- 不使用，1- 使用，默认不使用

byVoiceComMode

对讲模式：0- 使用对讲库（默认），1- 使用 windows api 模式

byRes

保留

4.55 NET_DVR_SDKSTATE:SDK 状态信息

```
struct{
    DWORD    dwTotalLoginNum;
    DWORD    dwTotalRealPlayNum;
    DWORD    dwTotalPlayBackNum;
    DWORD    dwTotalAlarmChanNum;
    DWORD    dwTotalFormatNum;
    DWORD    dwTotalFileSearchNum;
    DWORD    dwTotalLogSearchNum;
    DWORD    dwTotalSerialNum;
    DWORD    dwTotalUpgradeNum;
    DWORD    dwTotalVoiceComNum;
    DWORD    dwTotalBroadCastNum;
    DWORD    dwRes[10];
}NET_DVR_SDKSTATE,*LPNET_DVR_SDKSTATE;
```

Members

dwTotalLoginNum

当前注册的用户数

dwTotalRealPlayNum

当前实时预览的路数

dwTotalPlayBackNum

当前回放或下载的路数

dwTotalAlarmChanNum

当前建立报警通道的路数

dwTotalFormatNum

当前硬盘格式化的路数

dwTotalFileSearchNum

当前文件搜索的路数

dwTotalLogSearchNum

当前日志搜索的路数

dwTotalSerialNum

当前建立透明通道的路数

dwTotalUpgradeNum

当前升级的路数

dwTotalVoiceComNum

当前语音转发的路数

dwTotalBroadCastNum

当前语音广播的路数

dwRes

保留，置为 0

4.56 NET_DVR_SETUPALARM_PARAM:报警布防参数

```

struct{
    DWORD    dwSize;
    BYTE     byLevel;
    BYTE     byAlarmInfoType;
    BYTE     byRetAlarmTypeV40;
    BYTE     byRetDevInfoVersion;
    BYTE     byRetVQDAlarmType;
    BYTE     byFaceAlarmDetection;
    BYTE     bySupport;
    BYTE     byRes[9];
}NET_DVR_SETUPALARM_PARAM, *LPNET_DVR_SETUPALARM_PARAM;

```

Members

dwSize

结构体大小

byLevel

设备不支持，置为 0

byAlarmInfoType

设备不支持，置为 0

byRetAlarmTypeV40

设备不支持，置为 0

byRetDevInfoVersion

设备不支持，置为 0

byRetVQDAlarmType

设备不支持，置为 0

byFaceAlarmDetection

设备不支持，置为 0

bySupport

设备不支持，置为 0

byRes

保留，置为 0

4.57 NET_DVR_STREAM_MODE_TYPE:取流方式

```

struct{
    BYTE     byGetStreamType;
    BYTE     byRes[3];
    NET\_DVR\_STREAM\_TYPE\_UNION    uGetStream;
}NET_DVR_STREAM_MODE_TYPE,*LPNET_DVR_STREAM_MODE_TYPE;

```

Members

byGetStreamType

取流方式：0- 直接从设备取流；1- 从流媒体取流；2- 通过 IP Server 获得 ip 地址后取流；3- 通过

IPServer 找到设备，再通过流媒体去设备的流； 4- 通过流媒体由 URL 去取流； 6- 通过结构体参数组装 URL 去取流（包含用户名和密码）

byRes

保留，置为 0

uGetStream

不同取流方式

4.58 NET_DVR_STREAM_SRC_CFG:流的来源配置

```
struct{
    DWORD                dwSize;
    NET\_DVR\_STREAM\_INFO   struStreamID;
    NET\_DVR\_STREAM\_SRC\_INFO struStreamSrcInfo;
}NET_DVR_STREAM_SRC_CFG,*LPNET_DVR_STREAM_SRC_CFG;
```

Members

dwSize

结构体大小

struStreamInfo

流信息

struStreamSrcInfo

流的来源信息

4.59 NET_DVR_STREAM_SRC_INFO:流的来源信息

```
struct{
    DWORD                dwSize;
    NET\_DVR\_STREAM\_MODE\_TYPE struStreamSrcInfo;
}NET_DVR_STREAM_SRC_INFO,*LPNET_DVR_STREAM_SRC_INFO;
```

Members

dwSize

结构体大小

struStreamSrcInfo

流的来源信息

4.60 NET_DVR_STREAM_TYPE_UNION:取流方式联合体

```
union{
    NET\_DVR\_DIRECT\_CONNECT\_CHAN\_INFO   struChanInfo;
    NET\_DVR\_PU\_STREAM\_URL               struStreamUrl;
    NET\_DVR\_PU\_STREAM\_URL\_CFG          struStreamUrlCfg;
}NET_DVR_STREAM_TYPE_UNION,*LPNET_DVR_STREAM_TYPE_UNION;
```

Members

struChanInfo

IP 通道信息

struStreamUrl

URL 取流配置信息

struStreamUrlCfg

通过结构体参数构成 URL 去取流

4.61 NET_DVR_STREAM_INFO:流信息

```
struct{
    DWORD    dwSize;
    BYTE     byID[STREAM_ID_LEN];
    DWORD    dwChannel;
    BYTE     byRes[32];
}NET_DVR_STREAM_INFO,*LPNET_DVR_STREAM_INFO;
```

Members

dwSize

结构体大小

byID

流 ID，为字母、数字和"_"的组合。智能交通摄像机不支持流 ID 标识功能，设为 0

dwChannel

设备通道号

byRes

保留，置为 0

4.62 NET_DVR_TIME:时间参数

```
struct{
    DWORD    dwYear;
    DWORD    dwMonth;
    DWORD    dwDay;
    DWORD    dwHour;
    DWORD    dwMinute;
    DWORD    dwSecond;
}NET_DVR_TIME, *LPNET_DVR_TIME;
```

Members

dwYear

年

dwMonth

月

dwDay

日

dwHour

时
dwMinute
 分
dwSecond
 秒

4.63 NET_DVR_TIME_EX:时间参数

```
struct{
    WORD    wYear;
    BYTE    byMonth;
    BYTE    byDay;
    BYTE    byHour;
    BYTE    byMinute;
    BYTE    bySecond;
    BYTE    byRes;
}NET_DVR_TIME_EX, *LPNET_DVR_TIME_EX;
```

Members

wYear
年

byMonth
月

byDay
日

byHour
时

byMinute
分

bySecond
秒

byRes
保留

4.64 NET_DVR_UNLOCK_RECORD_INFO:可视对讲开锁记录信息

```
struct{
    BYTE    byUnlockType;
    BYTE    byRes1[3];
    BYTE    byControlSrc[NAME_LEN];
    DWORD   dwPicDataLen;
    BYTE    *pImage;
    BYTE    byRes2[212];
}NET_DVR_UNLOCK_RECORD_INFO, *LPNET_DVR_UNLOCK_RECORD_INFO;
```

Members*byUnlockType*

开锁方式，具体定义如下：

```
enum _UNLOCK_TYPE_ENUM_{
    ENUM_UNLOCK_PASSWORD          = 1,
    ENUM_UNLOCK_HIJACKING         = 2,
    ENUM_UNLOCK_CARD              = 3,
    ENUM_UNLOCK_HOUSEHOLDER     = 4,
    ENUM_UNLOCK_CENTER_PLATFORM = 5
}UNLOCK_TYPE_ENUM
```

ENUM_UNLOCK_PASSWORD

密码开锁

ENUM_UNLOCK_HIJACKING

劫持开锁

ENUM_UNLOCK_CARD

刷卡开锁

ENUM_UNLOCK_HOUSEHOLDER

户主开锁

ENUM_UNLOCK_CENTER_PLATFORM

中心平台开锁

byRes1

保留，置为 0

byControlSrc

操作发起源信息，刷卡开锁时为卡号，其余情况下为设备编号

dwPicDataLen

图片数据长度

plmage

缓冲区指针，保存图片数据，默认 jpeg 格式

byRes2

保留，置为 0

4.65 NET_DVR_USER_INFO_V30:单用户参数

struct{

```
    BYTE          sUserName[NAME_LEN];
    BYTE          sPassword[PASSWD_LEN];
    BYTE          byLocalRight[MAX_RIGHT];
    BYTE          byRemoteRight[MAX_RIGHT];
    BYTE          byNetPreviewRight[MAX_CHANNUM_V30];
    BYTE          byLocalPlaybackRight[MAX_CHANNUM_V30];
    BYTE          byNetPlaybackRight[MAX_CHANNUM_V30];
    BYTE          byLocalRecordRight[MAX_CHANNUM_V30];
    BYTE          byNetRecordRight[MAX_CHANNUM_V30];
    BYTE          byLocalPTZRight[MAX_CHANNUM_V30];
```

```

BYTE                byNetPTZRight[MAX_CHANNUM_V30];
BYTE                byLocalBackupRight[MAX_CHANNUM_V30];
NET_DVR_IPADDR    struUserIP;
BYTE                byMACAddr[MACADDR_LEN];
BYTE                byPriority;
BYTE                byAlarmOnRight;
BYTE                byAlarmOffRight;
BYTE                byBypassRight;
BYTE                byRes[14];
}NET_DVR_USER_INFO_V30,*LPNET_DVR_USER_INFO_V30;

```

Members

sUserName

用户名

sPassword

密码

byLocalRight

本地操作权限，参数取值为 1 表示使能：

数组 0—本地控制云台

数组 1—本地手动录象

数组 2—本地回放

数组 3—本地设置参数

数组 4—本地查看状态、日志

数组 5—本地高级操作（升级，硬盘管理（格式化、设置硬盘属性、设置盘组、阵列扩容、RAID 固件升级））

数组 6—本地查看参数

数组 7—本地管理模拟和 IP camera

数组 8—本地备份

数组 9—本地关机/重启

byRemoteRight

远程操作权限，参数取值为 1 表示使能：

数组 0—远程控制云台

数组 1—远程手动录象

数组 2—远程回放

数组 3—远程设置参数（恢复默认参数，写日志）

数组 4—远程查看状态、日志

数组 5—远程高级操作（升级，硬盘管理（格式化、设置硬盘属性、设置盘组、阵列扩容、RAID 固件升级），JPEG 抓图，前面板锁定与解锁）

数组 6—远程发起语音对讲

数组 7—远程预览

数组 8—远程请求报警上传、报警输出

数组 9—远程控制，本地输出

数组 10—远程控制串口

数组 11—远程查看参数

数组 12—远程管理模拟和 IP camera

数组 13—远程关机/重启

byNetPreviewRight

远程可以预览的通道：1-有权限，0-无权限

byLocalPlaybackRight

本地可以回放的通道：1-有权限，0-无权限

byNetPlaybackRight

远程可以回放的通道：1-有权限，0-无权限

byLocalRecordRight

本地可以录像的通道：1-有权限，0-无权限

byNetRecordRight

远程可以录像的通道：1-有权限，0-无权限

byLocalPTZRight

本地可以控制 PTZ 的通道：1-有权限，0-无权限

byNetPTZRight

远程可以控制 PTZ 的通道：1-有权限，0-无权限

byLocalBackupRight

本地备份权限通道：1-有权限，0-无权限

struUserIP

用户 IP 地址(为 0 时表示允许任何地址)

byMACAddr

物理地址

byPriority

优先级：0xff-无，0-低，1-中，2-高

无（表示不支持优先级的设置）

低（默认权限：包括本地和远程回放，本地和远程查看日志和状态，本地和远程关机/重启）

中（包括本地和远程控制云台，本地和远程手动录像，本地和远程回放，语音对讲和远程预览，本地备份，本地/远程关机/重启）

高（管理员）

byAlarmOnRight

报警输入口布防权限

byAlarmOffRight

报警输入口撤防权限

byBypassRight

报警输入口旁路权限

byRes

保留，置为 0

4.66 NET_DVR_USER_LOGIN_INFO:用户登录参数

```
struct{
    char                sDeviceAddress[NET_DVR_DEV_ADDRESS_MAX_LEN];
    BYTE                byRes1;
    WORD                wPort;
    char                sUserName[NET_DVR_LOGIN_USERNAME_MAX_LEN];
```

```

char                sPassword[NET_DVR_LOGIN_PASSWD_MAX_LEN];
fLoginResultCallBack cbLoginResult;
void                *pUser;
BOOL                bUseAsynLogin;
BYTE                byRes2[128];
}NET_DVR_USER_LOGIN_INFO,*LPNET_DVR_USER_LOGIN_INFO;

```

Members

dwSize

结构体大小

sDeviceAddress

设备地址，IP 或者普通域名

byRes1

保留，设为 0

wPort

设备端口号，例如：8000

sUserName

登录用户名，例如：admin

sPassword

登录密码，例如：12345

cbLoginResult

登录状态回调函数，bUseAsynLogin 为 0 时有效

pUser

用户数据

bUseAsynLogin

是否异步登录：0- 否，1- 是

byRes2

保留，置为 0

Callback Function

```

typedef void(CALLBACK *fLoginResultCallBack)(
LONG                UserID,
DWORD               dwResult,
LPNET_DVR_DEVICEINFO_V30 lpDeviceInfo,
void                *pUser);

```

Callback Function Parameters

UserID

[out] 用户 ID，[NET_DVR_Login_V40](#) 的返回值

dwResult

[out] 登录状态：0- 异步登录失败，1- 异步登录成功

lpDeviceInfo

[out] 设备信息，设备序列号、通道、能力等参数

pUser

[out] 用户数据

4.67 NET_DVR_USER_V30:用户参数

```
struct{
    DWORD                dwSize;
    NET_DVR_USER_INFO_V30 struUser[MAX_USERNUM_V30];
}NET_DVR_USER_V30,*LPNET_DVR_USER_V30;
```

Members

dwSize

结构体大小

struUser

用户信息参数

4.68 NET_DVR_VIDEO_INTERCOM_ALARM:可视对讲报警信息

```
struct{
    DWORD                dwSize;
    NET_DVR_TIME_EX     struTime;
    BYTE                 byDevNumber[MAX_DEV_NUMBER_LEN];
    BYTE                 byAlarmType;
    BYTE                 byRes1[3];
    NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION uAlarmInfo;
    BYTE                 byRes2[256];
}NET_DVR_VIDEO_INTERCOM_ALARM,*LPNET_DVR_VIDEO_INTERCOM_ALARM;
```

Members

dwSize

结构体大小

struTime

报警时间

byDevNumber

设备编号

byAlarmType

报警类型：1- 防区报警，2- 防拆报警，3- 劫持报警，4- 多次密码开锁失败报警，5- 门没开，6- 门没关，7- SOS(呼救报警)

byRes1

保留，置为 0

uAlarmInfo

报警信息，byAlarmType 为 1 时有效

byRes2

保留，置为 0

4.69 NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION:可视对讲报警信息联合体

```
union{
    BYTE                               byLen[256];
    NET\_DVR\_ZONE\_ALARM\_INFO           struZoneAlarm;
}NET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION, *LPNET_DVR_VIDEO_INTERCOM_ALARM_INFO_UNION;
```

Members

byLen

联合体大小，256 字节

struZoneAlarm

防区报警信息，报警类型为“1-防区报警”时有效

4.70 NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG:可视对讲设备编号配置

```
struct{
    DWORD                               dwSize;
    BYTE                                 byUnitType;
    BYTE                                 byIsAutoReg;
    BYTE                                 byRes1[2];
    NET\_DVR\_VIDEO\_INTERCOM\_UNIT\_DEVICEID\_UNION uVideoIntercomUnit;
    BYTE                                 byRes2[128];
}NET_DVR_VIDEO_INTERCOM_DEVICEID_CFG,*LPNET_DVR_VIDEO_INTERCOM_DEVICEID_CFG;
```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，3-室内机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byIsAutoReg

是否自动注册：0- 否，1- 是

byRes1

保留，置为 0

uVideoIntercomUnit

设备编号参数，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

- 设备编号参数能力，对应 IP 可视对讲主机能力集（接口：NET_DVR_GetDeviceAbility，能力集类型：IP_VIEW_DEV_ABILITY）中节点<DeviceID>。
- 二次确认机不配置编号。

4.71 NET_DVR_VIDEO_INTERCOM_EVENT:可视对讲事件记录信息

```

struct{
    DWORD                dwSize;
    NET\_DVR\_TIME\_EX      struTime;
    BYTE                 byDevNumber[MAX_DEV_NUMBER_LEN];
    BYTE                 byEventType;
    BYTE                 byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_EVENT\_INFO\_UINON uEventInfo;
    BYTE                 byRes2[256];
}NET_DVR_VIDEO_INTERCOM_EVENT, *LPNET_DVR_VIDEO_INTERCOM_EVENT;

```

Members

dwSize

结构体大小

struTime

事件记录时间

byDevNumber

设备编号

byEventType

事件信息类型：1- 开锁记录，2- 公告信息阅读回执

byRes1

保留，置为 0

uEventInfo

事件信息，不同的类型对应不同的结构

byRes2

保留，置为 0

4.72 NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON:可视对讲事件记录信息联合体

```

union{
    BYTE                 byLen[256];
    NET\_DVR\_UNLOCK\_RECORD\_INFO      struUnlockRecord;
    NET\_DVR\_NOTICEDATA\_RECEIPT\_INFO struNoticedataReceipt;
}NET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON, *LPNET_DVR_VIDEO_INTERCOM_EVENT_INFO_UINON;

```

Members

byLen

联合体大小，256 字节

struUnlockRecord

开锁记录

struNoticedataReceipt

公告信息阅读回执

4.73 NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG:操作时间配置

```

struct{
    DWORD                dwSize;
    BYTE                 byUnitType;
    BYTE                 byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_OPERATION\_TIME\_UNION uVideoIntercomUnit;
    BYTE                 byRes2[128];
}NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG, *LPNET_DVR_VIDEO_INTERCOM_OPERATION_TIME_CFG;

```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，3- 室内机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byRes1

保留，置为 0

uVideoIntercomUnit

操作时间配置联合体，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

- 操作时间配置能力，对应 IP 可视对讲主机能力集（接口：NET_DVR_GetDeviceAbility，能力集类型：IP_VIEW_DEV_ABILITY）中节点<OperationTime>。

4.74 NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION:可视对讲操作时间配置联合体

```

union{
    BYTE                 byLen[128];
    NET\_DVR\_INDOOR\_UNIT\_OPERATION\_TIME\_CFG struIndoorUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_OPERATION\_TIME\_CFG struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_OPERATION\_TIME\_CFG struManageUnit;
}NET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION,
*LPNET_DVR_VIDEO_INTERCOM_OPERATION_TIME_UNION;

```

Members

byLen

联合体大小，128 字节

struIndoorUnit

室内机操作时间配置

struOutdoorUnit

门口机/围墙机/别墅门口机/二次确认机操作时间配置

struManageUnit

管理机操作时间配置

4.75 NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG:可视对讲关联网络 设备配置

```
struct{
    DWORD                dwSize;
    BYTE                 byUnitType;
    BYTE                 byRes1[3];
    NET\_DVR\_VIDEO\_INTERCOM\_UNIT\_RELATEDDEV\_UNION uVideoIntercomUnit;
    BYTE                 byRes2[128];
}NET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG, *LPNET_DVR_VIDEO_INTERCOM_RELATEDDEV_CFG;
```

Members

dwSize

结构体大小

byUnitType

设备类型：1- 门口机，2- 管理机，3-室内机，4- 围墙机，5- 别墅门口机，6- 二次确认机

byRes1

保留，置为 0

uVideoIntercomUnit

关联网络设备配置联合体，不同的设备类型对应不同的结构

byRes2

保留，置为 0

Remarks

- 关联网络设备配置能力，对应 IP 可视对讲主机能力集（接口：NET_DVR_GetDeviceAbility，能力集类型：IP_VIEW_DEV_ABILITY）中节点<RelateDevice>。

4.76 NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION:可视对讲设 备编号参数联合体

```
union{
    BYTE                 byLen[128];
    NET\_DVR\_INDOOR\_UNIT\_DEVICEID        struIndoorUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_DEVICEID       struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_DEVICEID        struManageUnit;
    NET\_DVR\_OUTDOOR\_FENCE\_DEVICEID      struFenceUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_DEVICEID       struVillaOutdoorUnit;
}NET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION,
*LPNET_DVR_VIDEO_INTERCOM_UNIT_DEVICEID_UNION;
```

Members*byLen*

联合体大小, 128 字节

struIndoorUnit

室内机设备编号参数

struOutdoorUnit

门口机设备编号参数

struManageUnit

管理机设备编号参数

struFenceUnit

围墙机设备编号参数

struVillaOutdoorUnit

别墅门口机设备编号参数

Remarks

- 设备根据配置信息, 生成设备编号字符串, 规则如下:
 - 1) 管理机的长号为*00000001XX; *代表期号, 001 代表管理机, XX 代表管理机编号(从 01 开始, 01、02、..., 加 1 递增)。
 - 2) 门口机或者别墅门口机的编号为*XXXXX000\$\$; *代表期号, XXX 代表楼号, XX 单元号, 000 代表门口机, \$\$代表门口机序号, 序号为 00 则代表主门口机, 其他取值则代表副门口机。
 - 3) 围墙机的长号为*00000002XX; *代表期号, 002 代表围墙机, XX 代表围墙机编号(从 01 开始, 01、02、..., 加 1 递增)。
 - 4) 室内机的编号为*XXXXX\$\$\$\$; *代表期号, XXX 代表楼号, XX 代表单元号, \$\$\$代表层号, \$\$代表房间号。
- 实际应用时同一管理机可能属于多期楼房, 对于此种情况, 管理机的期号赋值可以为任何一个存在期号, 只要保证期号、管理机序号及其他组合信息生成的设备编号是唯一的即可。

4.77 NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION:可视对讲 关联网络设备配置联合体

```
union{
    DWORD                                     dwRes[256];
    NET\_DVR\_INDOOR\_UNIT\_RELATEDDEV          struIndoorUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_RELATEDDEV         struOutdoorUnit;
    NET\_DVR\_MANAGE\_UNIT\_RELATEDDEV         struManageUnit;
    NET\_DVR\_OUTDOOR\_UNIT\_RELATEDDEV         struVillaUnit;
    NET\_DVR\_AGAIN\_RELATEDDEV              struAgainUnit;
}NET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION,
*LPNET_DVR_VIDEO_INTERCOM_UNIT_RELATEDDEV_UNION;
```

Members*dwRes*

联合体大小, 256*4 共 1024 字节

struIndoorUnit

室内机关联网络设备参数配置

struOutdoorUnit

门口机/围墙机关联网络设备参数配置

struManageUnit

管理机关联网络设备参数配置

struVillaUnit

别墅门口机关联网络设备参数配置

struAgainUnit

二次确认机关联网络设备参数配置

4.78 NET_DVR_VIDEOINTERCOM_STREAM:可视对讲流通道参数

```
struct{
    DWORD    dwSize;
    BYTE     byVisDevID[MAX_NAMELEN];
    BYTE     byDeviceName[NAME_LEN];
    BYTE     bySourceType;
    BYTE     byRes[255];
}NET_DVR_VIDEOINTERCOM_STREAM, *LPNET_DVR_VIDEOINTERCOM_STREAM;
```

Members

dwSize

结构体大小

byVisDevID

可视对讲设备编号，即长号，设备类型为 IPC 时全部为 0

byDeviceName

设备名称

bySourceType

视频源类型：0- 无意义，1- IPC，2- DVR/DVS/NVR，3- 门口机，4- 围墙机，5- 二次确认机

byRes

保留，置为 0

Remarks

- 设备是否支持可视对讲流通道参数配置或者支持的参数能力，可以通过设备能力集进行判断，对应 IP 可视对讲主机能力集 (IpViewDevAbility)，相关接口：[NET_DVR_GetDeviceAbility](#)，能力集类型：IP_VIEW_DEV_ABILITY，节点：<VideoIntercomStream>。

4.79 NET_DVR_VOLUME_CFG:音量调节参数配置

```
struct{
    DWORD    dwSize;
    WORD     wVolume[MAX_AUDIOOUT_PRO_TYPE];
    BYTE     byPhantomPowerSupply;
    BYTE     byEnableAEC;
    BYTE     byRes1[2];
}
```

```

BYTE        byEnableFBC[MAX_AUDIOOUT_PRO_TYPE];
BYTE        byRes[20];
}NET_DVR_VOLUME_CFG,*LPNET_DVR_VOLUME_CFG;

```

Members

dwSize

结构体大小

wVolume

音量大小，数组 0 表示音频输出，数组 1 表示音频编码，具体索引代表含义以能力集为准

byPhantomPowerSupply

是否使用幻象电源供电(音频输入通道为 MIC 时有效): 0- 无意义, 1- 不供电, 2- 供电

byEnableAEC

是否启用全局的回声消除: 0- 不启用, 1- 启用

byRes1

保留, 置为 0

byEnableFBC

是否启用 FBC(啸叫抑制): 0- 不启用, 1- 启用

byRes

保留, 置为 0

4.80 NET_DVR_ZONE_ALARM_INFO:防区报警信息

```

struct{
    BYTE        byZoneName[NAME_LEN];
    DWORD       dwZonendex;
    BYTE        byZoneType;
    BYTE        byRes[219];
}NET_DVR_ZONE_ALARM_INFO,*LPNET_DVR_ZONE_ALARM_INFO;

```

Members

byZoneName

防区名称

dwZonendex

防区号

byZoneType

防区类型, 具体定义如下:

```

enum _ALARM_ZONE_TYPE_ENUM_{
    ENUM_ALARM_ZONE_TYPE_MANUAL           = 0,
    ENUM_ALARM_ZONE_TYPE_MAGNETIC         = 1,
    ENUM_ALARM_ZONE_TYPE_SMOKE            = 2,
    ENUM_ALARM_ZONE_TYPE_ACTIVE_INFRARED  = 3,
    ENUM_ALARM_ZONE_TYPE_PASSIVE_INFRARED = 4,
    ENUM_ALARM_ZONE_TYPE_GAS              = 11
}ALARM_ZONE_TYPE_ENUM

```

ENUM_ALARM_ZONE_TYPE_MANUAL

紧急开关报警
 ENUM_ALARM_ZONE_TYPE_MAGNETIC
 门磁报警
 ENUM_ALARM_ZONE_TYPE_SMOKE
 烟感报警
 ENUM_ALARM_ZONE_TYPE_ACTIVE_INFRARED
 主动红外报警
 ENUM_ALARM_ZONE_TYPE_PASSIVE_INFRARED
 被动红外报警
 ENUM_ALARM_ZONE_TYPE_GAS
 煤气报警

byRes

保留，置为 0

4.81 NET_DVR_VIDEO_CALL_COND:可视对讲信令处理条件参数

```
struct{
    DWORD    dwSize;
    BYTE     byRes[128];
}NET_DVR_VIDEO_CALL_COND, *LPNET_DVR_VIDEO_CALL_COND;
```

Members

dwSize

结构体大小

byRes

保留，置为 0

4.82 NET_DVR_VIDEO_CALL_PARAM:可视对讲信令处理参数

```
struct{
    DWORD    dwSize;
    DWORD    dwCmdType;
    BYTE     byRes[128];
}NET_DVR_VIDEO_CALL_PARAM, *LPNET_DVR_VIDEO_CALL_PARAM;
```

Members

dwSize

结构体大小

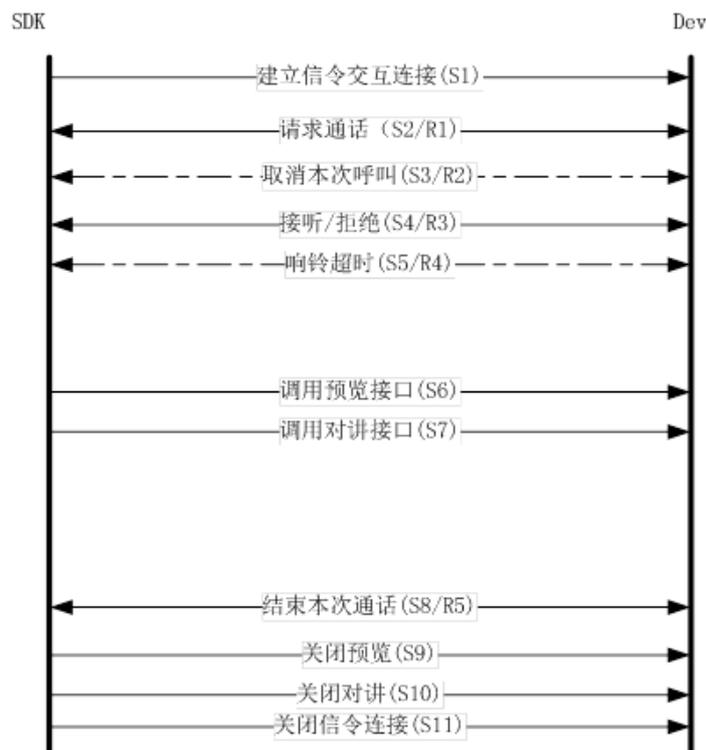
dwCmdType

信令类型：0- 请求呼叫，1- 取消本次呼叫，2- 接听本次呼叫，3- 拒绝本地来电呼叫，4- 被叫响铃超时，5- 结束本次通话，6- 设备正在通话中，7- 设备正在通话中

byRes

保留，置为 0

Remarks



- 可视通话能力，对应 IP 可视对讲主机能力集（接口：NET_DVR_GetDeviceAbility，能力集类型：IP_VIEW_DEV_ABILITY）中节点<VideoCall>。

- 该长连接配置接口结合报警、预览及对讲接口，可以完成可视通话的功能，具体流程机制如下所示：
客户端主动发起对讲：

1)当客户端主动发起连接请求后，客户端如果需要建立可视通话，则需要发送通话请求信令（S2），设备侧接收到该请求后，如果无人接听的情况下，30s 后设备端会返回响铃超时（R4），此时应用层直接关闭信令链接（S11）即可。

2)在客户端发送 S2 请求后，设备未返回 R3 及 R4 之前，客户端可以直接发送 S3，取消本次呼叫，然后直接调用 S7 即可。在客户端发送 S2 请求且设备做了应答后，则无法取消本次呼叫。

3)在客户端发送 S2 请求后，如果设备侧如果拒绝接听 R3，则客户端直接关闭连接即可。

4)在客户端发送 S2 请求后，如果设备接听，则客户端后续打开预览（不播放声音）、对讲即可进行视频通话（S6、S7）；如果中间任何一方想关闭连接，需要先要发送结束本次通话（S8/R5），后续关闭预览、对讲及信令连接即可。

设备侧主动发起对讲：

1)当客户端主动发起连接请求后，设备段如果需要建立可视通话，则需要发送通话请求信令（R1），客户端接收到该请求后，如果无人接听的情况下，30s 后会返回响铃超时（S5），此时应用层直接关闭信令链接（S11）即可。

2)在设备端发送 R1 请求后，客户端未返回 S4 及 S5 之前，设备端可以直接发送 R2，取消本次呼叫，然后直接调用 S7 即可。在设备端发送 R1 请求且客户端做了应答后，则无法取消本次呼叫。

3)在设备端发送 R1 请求后，如果客户如果拒绝接听 R3，则客户端直接关闭连接即可。

4)在设备侧发送 R1 请求后，如果客户端接听，则客户端后续打开预览（不播放声音）、对讲即可进行视频通话（S6、S7）；如果中间任何一方想关闭连接，需要先要发送结束本次通话（S8/R5），后续关闭预览、对讲及信令连接即可。

4.83 设备日志信息

dwMajorType 主类型

宏定义	宏定义值	含义
MAJOR_ALARM	0x1	报警
MAJOR_EXCEPTION	0x2	异常
MAJOR_OPERATION	0x3	操作
MAJOR_INFORMATION	0x4	日志附加信息
MAJOR_EVENT	0x5	事件

dwMinorType 次类型

主类型的宏定义	宏定义值	含义
MAJOR_ALARM	0x1	报警
次类型的宏定义	宏定义值	含义
MINOR_ALARM_IN	0x1	报警输入
MINOR_ALARM_OUT	0x2	报警输出
MINOR_MOTDET_START	0x3	移动侦测报警开始
MINOR_MOTDET_STOP	0x4	移动侦测报警结束
MINOR_HIDE_ALARM_START	0x5	遮挡报警开始
MINOR_HIDE_ALARM_STOP	0x6	遮挡报警结束
MINOR_VCA_ALARM_START	0x7	智能报警开始
MINOR_VCA_ALARM_STOP	0x8	智能报警结束
MINOR_ITS_ALARM_START	0x09	交通事件报警开始
MINOR_ITS_ALARM_STOP	0x0a	交通事件报警结束
MINOR_NETALARM_START	0x0b	网络报警开始
MINOR_NETALARM_STOP	0x0c	网络报警结束
MINOR_NETALARM_RESUME	0x0d	网络报警恢复

主类型的宏定义	宏定义值	含义
MAJOR_EXCEPTION	0x2	异常
次类型的宏定义	宏定义值	含义
MINOR_RAID_ERROR	0x20	阵列异常
MINOR_VI_LOST	0x21	视频信号丢失
MINOR_ILLEGAL_ACCESS	0x22	非法访问
MINOR_HD_FULL	0x23	硬盘满
MINOR_HD_ERROR	0x24	硬盘错误

MINOR_DCD_LOST	0x25	MODEM 掉线(保留)
MINOR_IP_CONFLICT	0x26	IP 地址冲突
MINOR_NET_BROKEN	0x27	网络断开
MINOR_REC_ERROR	0x28	录像出错
MINOR_IPC_NO_LINK	0x29	IPC 连接异常
MINOR_VI_EXCEPTION	0x2a	视频输入异常(只针对模拟通道)
MINOR_IPC_IP_CONFLICT	0x2b	IPC 的 IP 地址冲突
MINOR_SENCE_EXCEPTION	0x2c	场景异常
MINOR_PIC_REC_ERROR	0x2d	抓图出错,获取图片文件失败
MINOR_VI_MISMATCH	0x2e	视频制式不匹配

主类型的宏定义	宏定义值	含义
MAJOR_OPERATION	0x3	操作
次类型的宏定义	宏定义值	含义
MINOR_START_DVR	0x41	开机
MINOR_STOP_DVR	0x42	关机
MINOR_STOP_ABNORMAL	0x43	异常关机
MINOR_REBOOT_DVR	0x44	本地重启设备
MINOR_LOCAL_LOGIN	0x50	本地登陆
MINOR_LOCAL_LOGOUT	0x51	本地注销登陆
MINOR_LOCAL_CFG_PARM	0x52	本地配置参数
MINOR_LOCAL_PLAYBYFILE	0x53	本地按文件回放或下载
MINOR_LOCAL_PLAYBYTIME	0x54	本地按时间回放或下载
MINOR_LOCAL_START_REC	0x55	本地开始录像
MINOR_LOCAL_STOP_REC	0x56	本地停止录像
MINOR_LOCAL_PTZCTRL	0x57	本地云台控制
MINOR_LOCAL_PREVIEW	0x58	本地预览(保留不使用)
MINOR_LOCAL_MODIFY_TIME	0x59	本地修改时间(保留不使用)
MINOR_LOCAL_UPGRADE	0x5a	本地升级
MINOR_LOCAL_RECFILE_OUTPUT	0x5b	本地备份录像文件
MINOR_LOCAL_FORMAT_HDD	0x5c	本地初始化硬盘
MINOR_LOCAL_CFGFILE_OUTPUT	0x5d	导出本地配置文件
MINOR_LOCAL_CFGFILE_INPUT	0x5e	导入本地配置文件
MINOR_LOCAL_COPYFILE	0x5f	本地备份文件
MINOR_LOCAL_LOCKFILE	0x60	本地锁定录像文件
MINOR_LOCAL_UNLOCKFILE	0x61	本地解锁录像文件
MINOR_LOCAL_DVR_ALARM	0x62	本地手动清除和触发报警

MINOR_IPC_ADD	0x63	本地添加 IPC
MINOR_IPC_DEL	0x64	本地删除 IPC
MINOR_IPC_SET	0x65	本地设置 IPC
MINOR_LOCAL_START_BACKUP	0x66	本地开始备份
MINOR_LOCAL_STOP_BACKUP	0x67	本地停止备份
MINOR_LOCAL_COPYFILE_START_TIME	0x68	本地备份开始时间
MINOR_LOCAL_COPYFILE_END_TIME	0x69	本地备份结束时间
MINOR_LOCAL_ADD_NAS	0x6a	本地添加网络硬盘
MINOR_LOCAL_DEL_NAS	0x6b	本地删除 NAS 盘
MINOR_LOCAL_SET_NAS	0x6c	本地设置 NAS 盘
MINOR_REMOTE_LOGIN	0x70	远程登录
MINOR_REMOTE_LOGOUT	0x71	远程注销登陆
MINOR_REMOTE_START_REC	0x72	远程开始录像
MINOR_REMOTE_STOP_REC	0x73	远程停止录像
MINOR_START_TRANS_CHAN	0x74	开始透明传输
MINOR_STOP_TRANS_CHAN	0x75	停止透明传输
MINOR_REMOTE_GET_PARM	0x76	远程获取参数
MINOR_REMOTE_CFG_PARM	0x77	远程配置参数
MINOR_REMOTE_GET_STATUS	0x78	远程获取状态
MINOR_REMOTE_ARM	0x79	远程布防
MINOR_REMOTE_DISARM	0x7a	远程撤防
MINOR_REMOTE_REBOOT	0x7b	远程重启
MINOR_START_VT	0x7c	开始语音对讲
MINOR_STOP_VT	0x7d	停止语音对讲
MINOR_REMOTE_UPGRADE	0x7e	远程升级
MINOR_REMOTE_PLAYBYFILE	0x7f	远程按文件回放
MINOR_REMOTE_PLAYBYTIME	0x80	远程按时间回放
MINOR_REMOTE_PTZCTRL	0x81	远程云台控制
MINOR_REMOTE_FORMAT_HDD	0x82	远程格式化硬盘
MINOR_REMOTE_STOP	0x83	远程关机
MINOR_REMOTE_LOCKFILE	0x84	远程锁定文件
MINOR_REMOTE_UNLOCKFILE	0x85	远程解锁文件
MINOR_REMOTE_CFGFILE_OUTPUT	0x86	远程导出配置文件
MINOR_REMOTE_CFGFILE_INTPUT	0x87	远程导入配置文件
MINOR_REMOTE_RECFILE_OUTPUT	0x88	远程导出录象文件
MINOR_REMOTE_DVR_ALARM	0x89	远程手动清除和触发报警
MINOR_REMOTE_IPC_ADD	0x8a	远程添加 IPC

MINOR_REMOTE_IPC_DEL	0x8b	远程删除 IPC
MINOR_REMOTE_IPC_SET	0x8c	远程设置 IPC
MINOR_REBOOT_VCA_LIB	0x8d	重启智能库
MINOR_REMOTE_ADD_NAS	0x8e	远程添加 NAS 盘
MINOR_REMOTE_DEL_NAS	0x8f	远程删除 NAS 盘
MINOR_REMOTE_SET_NAS	0x90	远程设置 NAS 盘
MINOR_LOCAL_START_REC_CDRW	0x91	本地开始刻录
MINOR_LOCAL_STOP_REC_CDRW	0x92	本地停止刻录
MINOR_REMOTE_START_REC_CDRW	0x93	远程开始刻录
MINOR_REMOTE_STOP_REC_CDRW	0x94	远程停止刻录
MINOR_LOCAL_PIC_OUTPUT	0x95	本地备份图片文件
MINOR_REMOTE_SET_NAS	0x96	远程备份图片文件
MINOR_LOCAL_INQUEST_RESUME	0x97	本地恢复审讯事件
MINOR_REMOTE_INQUEST_RESUME	0x98	远程恢复审讯事件
MINOR_REMOTE_BYPASS	0xd0	远程旁路
MINOR_REMOTE_UNBYPASS	0xd1	远程旁路恢复
MINOR_REMOTE_SET_ALARMIN_CFG	0xd2	远程设置报警输入参数
MINOR_REMOTE_GET_ALARMIN_CFG	0xd3	远程获取报警输入参数
MINOR_REMOTE_SET_ALARMOUT_CFG	0xd4	远程设置报警输出参数
MINOR_REMOTE_GET_ALARMOUT_CFG	0xd5	远程获取报警输出参数
MINOR_REMOTE_ALARMOUT_OPEN_MAN	0xd6	远程手动开启报警输出
MINOR_REMOTE_ALARMOUT_CLOSE_MAN	0xd7	远程手动关闭报警输出
MINOR_REMOTE_ALARM_ENABLE_CFG	0xd8	远程设置报警主机的 RS485 串口使能状态
MINOR_DBDATA_OUTPUT	0xd9	导出数据库记录
MINOR_DBDATA_INPUT	0xda	导入数据库记录
MINOR_MU_SWITCH	0xdb	级联切换
MINOR_MU_PTZ	0xdc	级联 PTZ 控制
MINOR_LOCAL_CONF_REB_RAID	0x101	本地配置自动重建
MINOR_LOCAL_CONF_SPARE	0x102	本地配置热备
MINOR_LOCAL_ADD_RAID	0x103	本地创建阵列
MINOR_LOCAL_DEL_RAID	0x104	本地删除阵列
MINOR_LOCAL_MIG_RAID	0x105	本地迁移阵列
MINOR_LOCAL_REB_RAID	0x106	本地手动重建阵列
MINOR_LOCAL_QUICK_CONF_RAID	0x107	本地一键配置
MINOR_LOCAL_ADD_VD	0x108	本地创建虚拟磁盘
MINOR_LOCAL_DEL_VD	0x109	本地删除虚拟磁盘
MINOR_LOCAL_RP_VD	0x10a	本地修复虚拟磁盘

MINOR_LOCAL_FORMAT_EXPANDVD	0x10b	本地扩展虚拟磁盘扩容
MINOR_LOCAL_RAID_UPGRADE	0x10c	本地 raid 卡升级
MINOR_LOCAL_STOP_RAID	0x10d	本地暂停 RAID 操作(即安全拔盘)
MINOR_REMOTE_CONF_REB_RAID	0x111	远程配置自动重建
MINOR_REMOTE_CONF_SPARE	0x112	远程配置热备
MINOR_REMOTE_ADD_RAID	0x113	远程创建阵列
MINOR_REMOTE_DEL_RAID	0x114	远程删除阵列
MINOR_REMOTE_MIG_RAID	0x115	远程迁移阵列
MINOR_REMOTE_REB_RAID	0x116	远程手动重建阵列
MINOR_REMOTE_QUICK_CONF_RAID	0x117	远程一键配置
MINOR_REMOTE_ADD_VD	0x118	远程创建虚拟磁盘
MINOR_REMOTE_DEL_VD	0x119	远程删除虚拟磁盘
MINOR_REMOTE_RP_VD	0x11a	远程修复虚拟磁盘
MINOR_REMOTE_FORMAT_EXPANDVD	0x11b	远程虚拟磁盘扩容
MINOR_REMOTE_RAID_UPGRADE	0x11c	远程 raid 卡升级
MINOR_REMOTE_STOP_RAID	0x11d	远程暂停 RAID 操作(即安全拔盘)
MINOR_LOCAL_START_PIC_REC	0x121	本地开始抓图
MINOR_LOCAL_STOP_PIC_REC	0x122	本地停止抓图
MINOR_LOCAL_SET_SNMP	0x125	本地配置 SNMP
MINOR_LOCAL_TAG_OPT	0x126	本地标签操作
MINOR_REMOTE_START_PIC_REC	0x131	远程开始抓图
MINOR_REMOTE_STOP_PIC_REC	0x132	远程停止抓图
MINOR_REMOTE_SET_SNMP	0x135	远程配置 SNMP
MINOR_REMOTE_TAG_OPT	0x136	远程标签操作
MINOR_REMOTE_TAG_OPT	0x136	远程标签操作
MINOR_LOCAL_VOUT_SWITCH	0x140	本地输出口切换操作
MINOR_SET_TRIGGERMODE_CFG	0x1001	设置触发模式参数
MINOR_GET_TRIGGERMODE_CFG	0x1002	获取触发模式参数
MINOR_SET_IOOUT_CFG	0x1003	设置 IO 输出参数
MINOR_GET_IOOUT_CFG	0x1004	获取 IO 输出参数
MINOR_GET_TRIGGERMODE_DEFAULT	0x1005	获取触发模式推荐参数
MINOR_GET_ITCSTATUS	0x1006	获取状态检测参数
MINOR_SET_STATUS_DETECT_CFG	0x1007	设置状态检测参数
MINOR_GET_STATUS_DETECT_CFG	0x1008	获取状态检测参数
MINOR_GET_VIDEO_TRIGGERMODE_CFG	0x1009	获取视频电警模式参数
MINOR_SET_VIDEO_TRIGGERMODE_CFG	0x100a	设置视频电警模式参数

主类型的宏定义	宏定义值	含义
MAJOR_INFORMATION	0x4	附加信息
次类型的宏定义	宏定义值	含义
MINOR_HDD_INFO	0xa1	硬盘信息
MINOR_SMART_INFO	0xa2	S.M.A.R.T 信息
MINOR_REC_START	0xa3	开始录像
MINOR_REC_STOP	0xa4	停止录像
MINOR_REC_OVERDUE	0xa5	过期录像删除
MINOR_LINK_START	0xa6	连接前端设备
MINOR_LINK_STOP	0xa7	断开前端设备
MINOR_NET_DISK_INFO	0xa8	网络硬盘信息
MINOR_RAID_INFO	0xa9	raid 相关信息
MINOR_PIC_REC_START	0xb3	开始抓图
MINOR_PIC_REC_STOP	0xb4	停止抓图
MINOR_PIC_REC_OVERDUE	0xb5	过期图片文件删除

主类型的宏定义	宏定义值	含义
MAJOR_EVENT	0x5	事件
次类型的宏定义	宏定义值	含义
MINOR_LEGAL_CARD_PASS	0x01	合法卡认证通过
MINOR_CARD_AND_PSW_PASS	0x02	刷卡加密码认证通过
MINOR_CARD_AND_PSW_FAIL	0x03	刷卡加密码认证失败
MINOR_CARD_AND_PSW_TIMEOUT	0x04	刷卡加密码认证超时
MINOR_CARD_AND_PSW_OVER_TIME	0x05	刷卡加密码超次
MINOR_CARD_NO_RIGHT	0x06	未分配权限
MINOR_CARD_INVALID_PERIOD	0x07	无效时段
MINOR_CARD_OUT_OF_DATE	0x08	卡号过期
MINOR_INVALID_CARD	0x09	无此卡号
MINOR_ANTI_SNEAK_FAIL	0x0a	反潜回认证失败
MINOR_INTERLOCK_DOOR_NOT_CLOSE	0x0b	互锁门未关闭
MINOR_NOT_BELONG_MULTI_GROUP	0x0c	卡不属于多重认证群组
MINOR_INVALID_MULTI_VERIFY_PERIOD	0x0d	卡不在多重认证时间段内
MINOR_MULTI_VERIFY_SUPER_RIGHT_FAIL	0x0e	多重认证模式超级权限认证失败
MINOR_MULTI_VERIFY_REMOTE_RIGHT_FAIL	0x0f	多重认证模式远程认证失败
MINOR_MULTI_VERIFY_SUCCESS	0x10	多重认证成功
MINOR_LEADER_CARD_OPEN_BEGIN	0x11	首卡开门开始
MINOR_LEADER_CARD_OPEN_END	0x12	首卡开门结束

MINOR_ALWAYS_OPEN_BEGIN	0x13	常开状态开始
MINOR_ALWAYS_OPEN_END	0x14	常开状态结束
MINOR_LOCK_OPEN	0x15	门锁打开
MINOR_LOCK_CLOSE	0x16	门锁关闭
MINOR_DOOR_BUTTON_PRESS	0x17	开门按钮打开
MINOR_DOOR_BUTTON_RELEASE	0x18	开门按钮放开
MINOR_DOOR_OPEN_NORMAL	0x19	正常开门（门磁）
MINOR_DOOR_CLOSE_NORMAL	0x1a	正常关门（门磁）
MINOR_DOOR_OPEN_ABNORMAL	0x1b	门异常打开（门磁）
MINOR_DOOR_OPEN_TIMEOUT	0x1c	门打开超时（门磁）
MINOR_ALARMOUT_ON	0x1d	报警输出打开
MINOR_ALARMOUT_OFF	0x1e	报警输出关闭
MINOR_ALWAYS_CLOSE_BEGIN	0x1f	常关状态开始
MINOR_ALWAYS_CLOSE_END	0x20	常关状态结束
MINOR_MULTI_VERIFY_NEED_REMOTE_OPEN	0x21	多重多重认证需要远程开门
MINOR_MULTI_VERIFY_SUPERPASSWD_VERIFY_SUCCESS	0x22	多重认证超级密码认证成功事件
MINOR_MULTI_VERIFY_REPEAT_VERIFY	0x23	多重认证重复认证事件
MINOR_MULTI_VERIFY_TIMEOUT	0x24	多重认证重复认证事件