

数字视频录像机接口协议

杭州雄迈信息有限公司

二〇〇八年十一月

Revision record 修订记录

Date 日期	Revision version 修订版本	CR ID / Defect ID CR号	Section Number 修改章节	Description 描述	Author 作者
2008-11-19	1.0.0		All	初稿	
2013-01-11				增加主动注册	罗友荣

Catalog 目录

1 概述	6
1.1 说明.....	6
1.2 适用范围	6
1.3 参考资料	6
1.4 术语表.....	6
2 系统体系结构.....	6
3 消息	7
3.1 控制流消息定义	7
3.2 媒体流消息头部	8
4 工作流程	8
4.1 登录登出保活流程.....	9
4.1.1 普通登录	9
4.1.2 主动注册登录	10
4.1.3 登出.....	10
4.1.4 保活.....	10
4.2 获取设备信息流程.....	10
4.3 设置获取配置信息流程	10
4.4 设置获取网络配置流程	10
4.5 设置获取告警配置流程	10
4.6 获取能力级流程	10
4.7 云台控制流程.....	10
4.8 监视控制流程.....	10
4.8.1 普通监视控制	11
4.8.2 主动注册监视控制.....	11
4.9 回放控制流程.....	13
4.9.1 普通回放控制	13
4.9.2 主动注册回放控制.....	15
4.10 录像下载流程.....	17
4.11 语言对讲流程.....	17
4.11.1 普通语音对讲.....	17
4.11.2 主动注册语音对讲	19
4.12 升级流程	21
4.13 录像管理流程.....	23
4.14 图片管理流程.....	23
4.15 日志管理流程.....	23
4.16 硬盘管理流程.....	23
4.17 用户管理流程.....	23
4.18 服务管理流程.....	23
4.19 设备自动搜索流程.....	23
5 语法与约定	23
5.1 登录登出保活协议 (C1)	23
5.1.1 普通登录	24
5.1.2 主动注册登录	25
5.1.3 登出.....	27
5.1.4 保活.....	29
5.2 获取设备信息 (C2)	30

5.2.1	获取系统信息请求.....	30
5.2.2	返回系统信息.....	31
	"DeviceRunTime": "0x0000019A".....	32
5.3	设置/获取配置相关请求 (C3)	34
5.3.1	设置配置请求.....	34
5.3.2	获取配置请求.....	36
5.3.3	获取默认配置请求.....	37
5.4	获取能力级请求 (C12)	38
5.6	云台控制 (C13)	40
5.4.1	云台控制请求.....	40
5.5	监视控制 (C14)	41
5.5.1	监视控制请求.....	41
5.6	回放控制 (C15)	42
5.6.1	回放控制请求.....	42
5.7	语音对讲(C16).....	43
5.7.1	语音对讲请求.....	43
5.8	文件查询 (C17)	45
5.9	日志查询 (C17)	46
5.10	系统时间查询 (C18)	48
5.11	系统管理 (C18)	49
5.11.1	系统管理请求.....	49
5.12	硬盘管理 (C19)	51
5.12.1	硬盘管理请求.....	51
5.13	用户管理 (C20)	52
5.13.1	获取完整权限列表.....	52
5.13.2	获取用户信息.....	53
5.13.3	获取用户组信息.....	55
5.13.4	添加用户组.....	56
5.13.5	修改用户组.....	57
5.13.6	删除用户组.....	59
5.13.7	添加用户.....	60
5.13.8	修改用户.....	61
5.13.9	删除用户.....	62
5.13.10	修改用户密码.....	63
5.14	告警上报 (C21)	65
5.14.1	布警.....	65
5.14.2	撤警.....	65
5.14.3	告警上报.....	66
5.14.4	网络告警输入.....	67
4.34.5	告警上报报警中心.....	68
5.15	系统升级 (C22)	69
5.15.1	升级控制.....	69
5.15.2	升级数据发送.....	71
5.15.3	升级进度通知.....	71
5.16	设备自动搜索 (C22)	72
5.16.1	IP自动搜索.....	72
5.17	系统信息导入导出 (C24)	73
5.17.1	配置导出.....	73
5.17.2	配置导入.....	74
5.17.3	日志导出.....	74
5.18	网络键盘 (C25)	74

6	参考资料清单.....	75
6.1	返回码定义.....	75
6.2	命令码定义.....	76
6.2.1	登录登出保活协议C1消息号.....	76
6.2.2	获取设备信息C2消息号.....	77
6.2.3	设置/获取配置相关信息请求C3~C11消息号.....	77
6.2.4	系统调试消息号.....	77
6.2.5	获取能力级请求C12消息号.....	77
6.2.6	云台控制C13消息号.....	78
6.2.7	监视控制C14消息号.....	78
6.2.8	回放控制C15消息号.....	78
6.2.9	语音对讲C16消息号.....	78
6.2.10	文件查询C17消息号.....	78
6.2.11	系统管理C18消息号.....	79
6.2.12	硬盘管理C19消息号.....	79
6.2.13	用户管理C20消息号.....	79
6.2.14	告警上报C21消息号.....	79
6.2.15	系统升级C22消息号.....	80
6.2.16	设备自动搜索C23消息号.....	80
6.2.17	系统信息导入导出C24消息号.....	80
6.2.18	网络控制键盘C25消息号.....	80
6.2.19	网络抓拍C26消息号.....	81
6.2.20	透明串口C27消息号.....	81
6.2.21	网络登录时间同步.....	81
6.2.22	抓图上传消息号.....	81

Figure List 图目录

图1	DVRIP体系结构图.....	6
----	-----------------	---

1 概述

1.1 说明

本协议是DVR和DVR、DVR和客户端之间、DVR和中心服务器、以及DVR和手机等各种终端之间的接口协议，简称DVRIP（DVR Interface Protocol）。

1.2 适用范围

本协议适用于DVR同支持其的各类客户端、控制设备，客户端一般通过网络连接DVR，控制设备一般与DVR串行方式通讯。

1.3 参考资料

1.4 术语表

术语	含义
DVR	Digital Video Recorder
DVRIP	DVR Interface Protocol
PU	前端，指编码器，包括DVR，DVS，IPC等
CU	客户端，指WEB，SNS等

2 系统体系结构

较新款DVR一般是具有网络功能的独立设备。全网可以有一台或多台DVR设备，每台DVR可以与多个客户端建立连接。特别地，DVR还可以同DVR、报警服务器等主动建立连接。

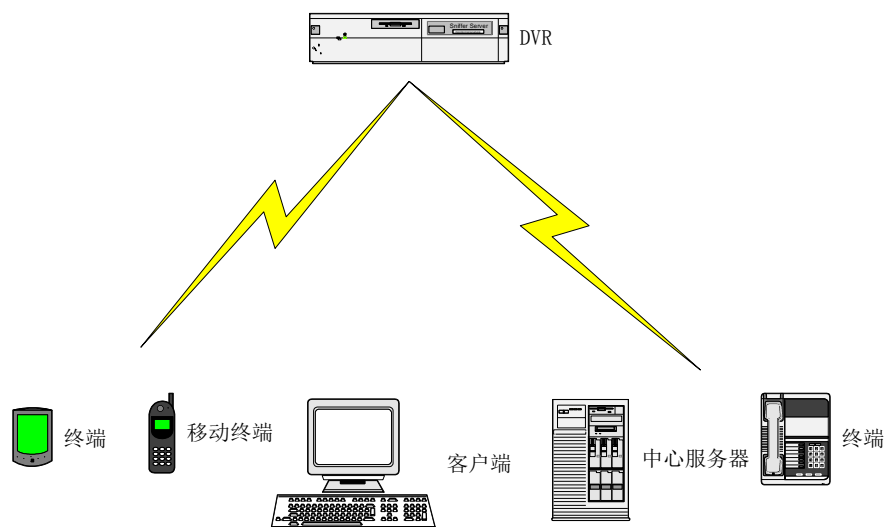


图1 DVRIP体系结构图

3 消息

3.1 控制流消息定义

传输控制流信息时需要加入此头部信息用于区别具体的控制命令码，头部采用二进制定长格式。协议包括协议头和数据区。其中，协议头为32字节固定长度，数据区为变长可选数据，数据区的长度在协议头中指示。

下图表为控制流消息格式：

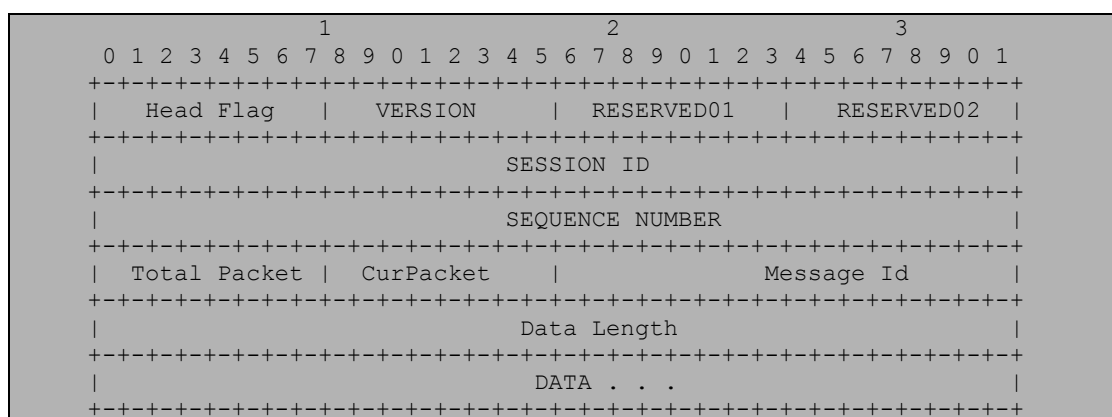


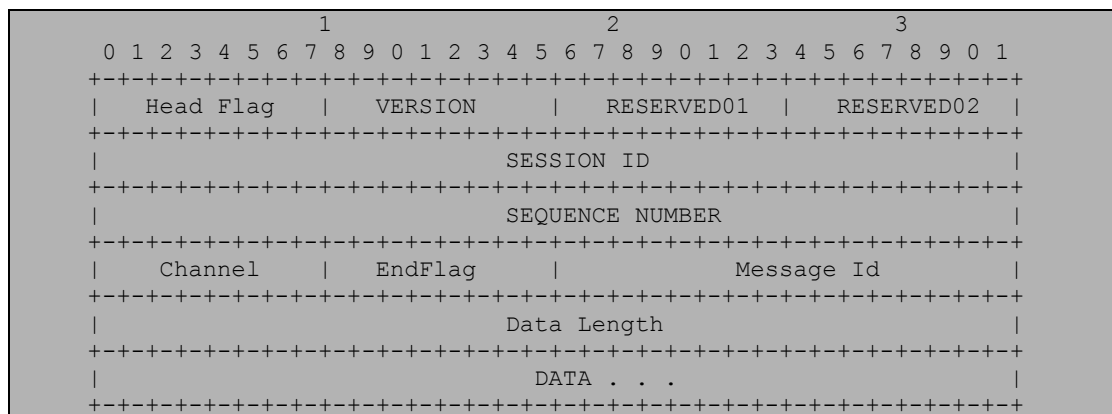
表1 控制流消息格式

- **Head Flag:** 协议头标识, 1字节, 固定为0xFF
- **VERSION:** 版本号, 当前为1
- **RESERVED01:** 保留位。
- **RESERVED02:** 保留位。
- **SESSION ID:** 会话ID, 每个CU与PU建立连接后, PU会给CU分配一个唯一的会话ID, PU请求登录时, 该字段添0, 登录成功后PU在响应消息中填充该字段, 此后双方在消息头部, 都要填充会话ID
- **SEQUENCE NUMBER:** 包序号, 启动后从0开始递增, 到达最大后, 在从0开始
- **Total Packet:** 总包数, 为0和1时, 表示只有一个数据包, 大于1表示有多个数据包, 每个数据包大小不超过16K (不包括头部长度)
- **CurPacket:** 当前包序号, 只有在Total Packet大于1时才有意义, 表示当前是第几个数据包, 下标从0开始
- **Message Id:** 消息码, 详见消息码定义表
- **Data Length:** 数据区长度, 字节为单位, 最大不超过16K
- **DATA:** 数据区, 最大不超过16K, 目前都是通过Json来封装, 以后可支持XML, 2进制流扩展

3.2 媒体流消息头部

媒体流包括监视，回放，录像下载的音视频数据，语言对讲数据，升级时的升级文件数据
图片文件数据。

下图为媒体流消息格式：



- Head Flag: 协议头标识, 1字节, 固定为0xFF
- VERSION: 版本号, 当前为1
- RESERVED01: 保留位。
- RESERVED02: 保留位。
- SESSION ID: 为登录成功后, 返回的会话ID
- SEQUENCE NUMBER: 包序号, 启动后从0开始递增, 到达最大后, 在从0开始
- Channel: 通道号, 在数据为回放和监视时需要标识对应通道
- EndFlag: 包结束标识, 在录像回放, 录像下载, 图片下载, 升级时使用, 0x01表示数据结束, 其他情况为0x0
- Message Id: 消息码, 详见消息码定义表
- Data Length: 数据区长度, 字节为单位
- DATA: 数据区

4 工作流程

整个系统得工作流程主要可以划分为以下几个部分, PU与CU进行交互成功后, 除了有需要返回的数据外, 还带有一个返回码, 具体见返回码定义表, 无特别说明, 控制流都是通过主socket进行通信

- 登录下线保活流程
- 获取设备信息流程
- 设置获取编码信息流程
- 设置获取录像相关配置流程

- 设置获取告警相关配置流程
- 设置获取网络相关流程
- 设置获取串口配置流程
- 设置获取云台配置流程
- 设置获取用户信息
- 设置获取预览配置流程
- 设置获取普通配置流程
- 设置获取自动维护配置流程
- 获取能力级流程
- 云台控制流程
- 监视控制流程
- 回放控制流程
- 语言对讲流程
- 文件查询流程
- 日志管理流程
- 硬盘管理流程
- 用户管理流程

4.1 登录登出保活流程

4.1.1 普通登录

CU请求进行登录，主要的流程如下：



- CU与PU建立主socket用于控制流传输。
- CU通过主socket发送登录请求，请求中包含用户名，密码，加密方式，登录类型。

- PU验证用户成功，返回SID，通道个数，设备类型。

4.1.2 主动注册登录

PU请求进行登录，主要的流程如下：



- PU与CU建立主socket用于控制流传输
- PU通过主socket发送登录请求，请求中包含用户名，密码，加密方式，设备类型，通道个数。
- CU验证用户成功，返回SID。

4.1.3 登出

4.1.4 保活

4.2 获取设备信息流程

4.3 设置获取配置信息流程

4.4 设置获取网络配置流程

4.5 设置获取告警配置流程

4.6 获取能力级流程

4.7 云台控制流程

4.8 监视控制流程

4.8.1 普通监视控制

CU请求进行视频监视，主要的流程如下：



- CU与PU建立socket用于媒体数据传输
- CU通过新建的socket发送认领请求，请求中包含通道号，码流类型，消息头的SID字段，为CU的SID号
- CU请求视频监视，请求中包含通道，码流类型，PU通过这个识别对应的媒体数据发送socket
- PU鉴权成功后，返回OK，并开启捕获模块，开始视频数据发送，视频数据的包序号从1开始递增
- CU请求关闭视频监视，请求中包含通道，码流类型，停止捕获模块，停止视频数据发送，PU返回应答信息，关闭对应媒体数据发送socket

4.8.2 主动注册监视控制

主动注册请求进行视频监视，主要的流程如下：



- CU通过主socket发送主动注册监视请求，请求中包含通道号，码流类型
- PU鉴权返回成功。
- PU与CU建立socket用于媒体数据传输
- PU通过新建的socket发送认领请求，请求中包含通道号，码流类型，消息头的SID字段，为CU的SID号
- CU请求视频监视，请求中包含通道，码流类型，PU通过这个识别对应的媒

体数据发送socket

- PU鉴权成功后，返回OK，并开启捕获模块，开始视频数据发送，视频数据的包序号从1开始递增
- CU请求关闭视频监视，请求中包含通道，码流类型，停止捕获模块，停止视频数据发送，PU返回应答信息，关闭对应媒体数据发送socket

4.9 回放控制流程

4.9.1 普通回放控制

CU请求进行视频监视，主要的流程如下：

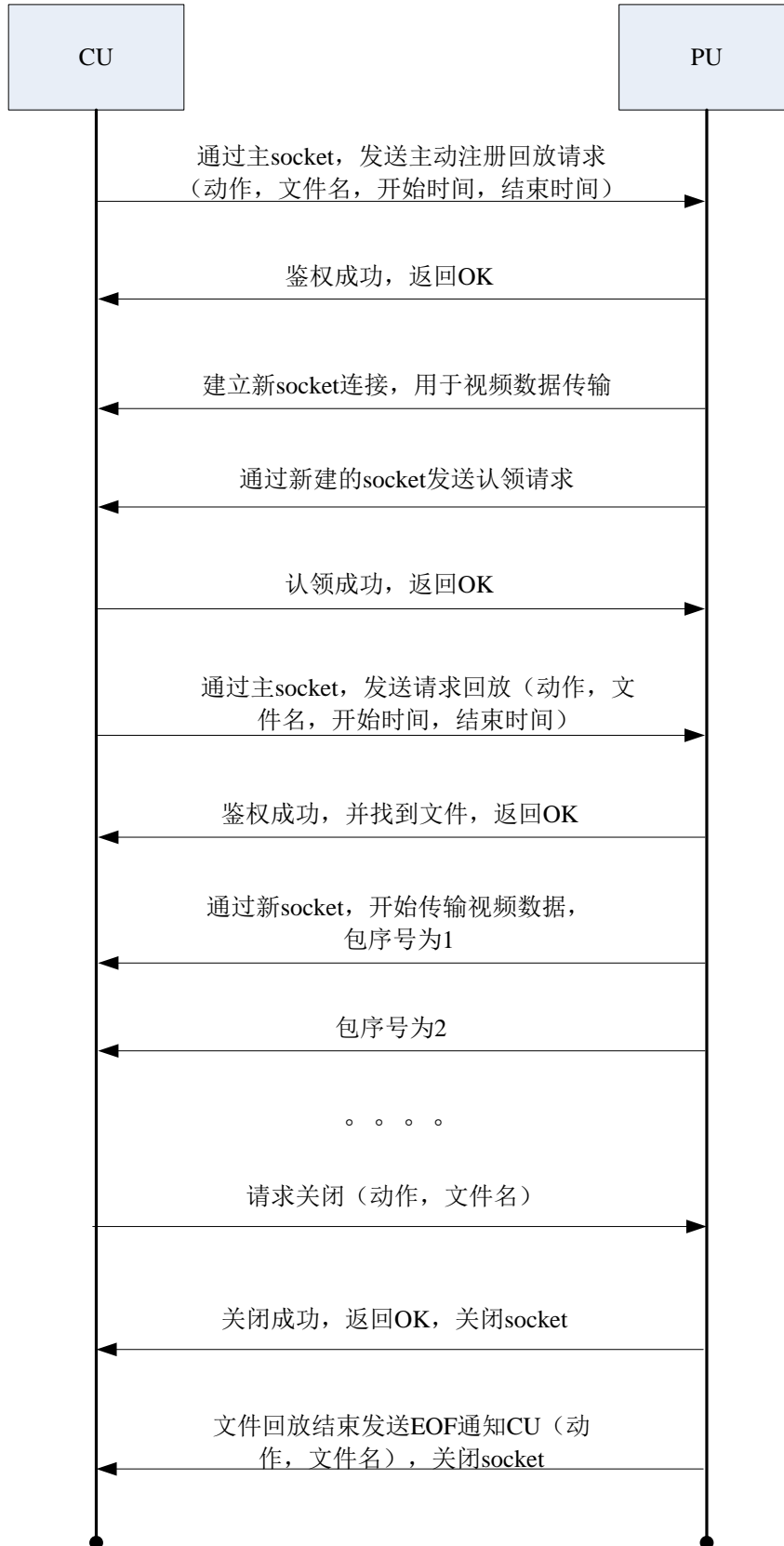


- CU与PU建立socket用于，回放数据传输
- CU通过新建socket发送认领请求，请求中包含通道号，消息头的SID字段，为CU的SID号

- CU请求网络回放，请求中包含文件名（来源于录像文件查询时PU传给CU的返回结果的文件名），开始时间和结束时间，录像回放会在这个时间区间内回放
- PU鉴权成功并且找到对应文件后，返回OK，并且PU开启捕获模块，开始视频数据发送，视频数据的包序号从1开始递增，通过新建的socket发送给CU
- 在回放期间内，CU可以进行，暂停回放，继续回放，定位回放，停止回放，关闭回放，关闭回放时会断开socket连接，而停止回放不会关闭socket连接。停止回放后在开始回放不需要重新建立连接，所有控制流的命令都是通过主socket发送
- CU关闭网络回放时，请求中包含通道，码流类型，PU停止捕获模块，停止视频数据发送，PU返回应答信息，关闭新建socket
- 当文件回放结束时，PU会在媒体数据包的包头endflag字段设置为1, 停止数据发送
- 对于每个CU最多支持4路回放（4路共用一个socket连接进行数据传输，因此当建立第一路回放通道时需要新建socket，此后的其他3个通道不需要新建socket了，直接请求网络回放即可），并且每个通道只支持1路回放，但不同的CU可以同时请求同一路回放。

4.9.2 主动注册回放控制

主动注册请求进行视频回放，主要的流程如下：



- CU通过主socket发起主动注册回放请求，请求文件名（来源于录像文件查询时PU传给CU的返回结果的文件名），开始时间和结束时间。
- PU鉴权返回成功。

- PU与CU建立socket用于媒体数据传输
- PU通过新建socket发送认领请求，请求中包含通道号，消息头的SID字段，为CU的SID号
- CU请求网络回放，请求中包含文件名，开始时间和结束时间，录像回放会在这个时间区间内回放
- PU鉴权成功并且找到对应文件后，返回OK，并且PU开启捕获模块，开始视频数据发送，视频数据的包序号从1开始递增，通过新建的socket发送给CU
- 在回放期间内，CU可以进行，暂停回放，继续回放，定位回放，停止回放，关闭回放，关闭回放时会断开socket连接，而停止回放不会关闭socket连接。停止回放后在开始回放不需要重新建立连接，所有控制流的命令都是通过主socket发送
- CU关闭网络回放时，请求中包含通道，码流类型，PU停止捕获模块，停止视频数据发送，PU返回应答信息，关闭新建socket
- 当文件回放结束时，PU会在媒体数据包的包头endflag字段设置为1，停止数据发送
- 对于每个CU最多支持4路回放（4路共用一个socket连接进行数据传输，因此当建立第一路回放通道时需要新建socket，此后的其他3个通道不需要新建socket了，直接请求网络回放即可），并且每个通道只支持1路回放，但不同的CU可以同时请求同一路回放。

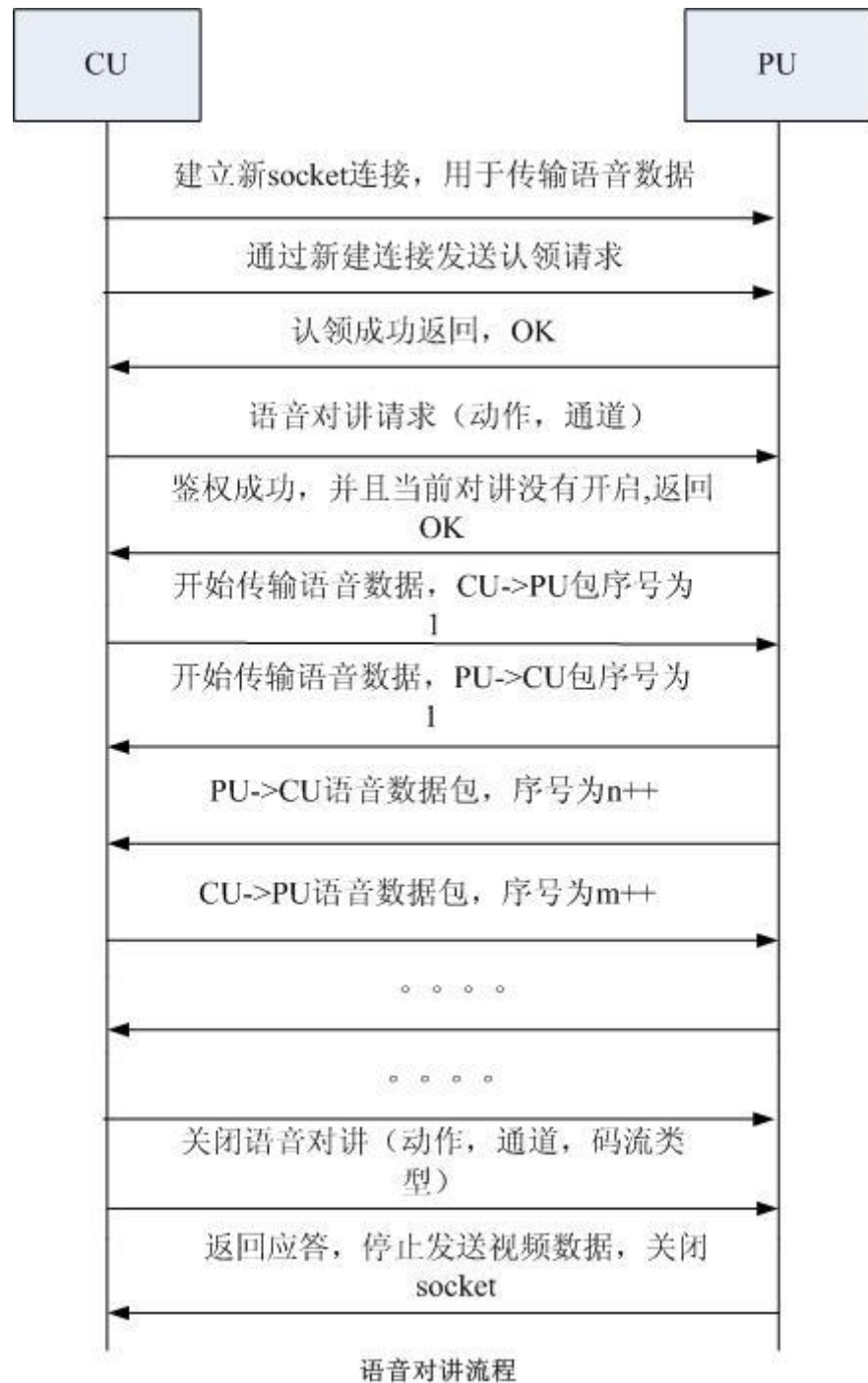
4.10 录像下载流程

录像下载和录像回放共用一套流程，处理时也是作为一路回放处理，也受每个CU最多支持4路回放限制，见回放控制流程。主动注册相同。

4.11 语言对讲流程

4.11.1 普通语音对讲

CU请求进行视频监视，主要的流程如下：



- CU与PU建立新socket用于，对讲数据传输
- CU通过新建socket发送认领请求，请求中包含通道号（目前只支持一路对讲，通道号为0），消息头的SID字段，为CU的SID号
- CU通过主socket请求语音对讲，请求中包含通道号（目前只支持一路对讲，通道号为0）
- 同时只支持一个CU与PU进行语音对讲，如果已经与其他CU开启了语音对讲，返回已经开始对讲错误码
- PU鉴权成功后，PU开启捕获模块，开始语音数据发送，语音数据的包序号从1开始递增，同时CU也开始语音数据发送，语音数据的包序号从1开始递增

- CU请求关闭语音对讲，请求中包含通道号，PU停止捕获模块，停止语音数据发送，PU返回应答信息，关闭新建socket，CU收到应答信息后，停止向PU发送语音数据，关闭新建socket

4.11.2 主动注册语音对讲

主动注册请求进行语音对讲，主要的流程如下：

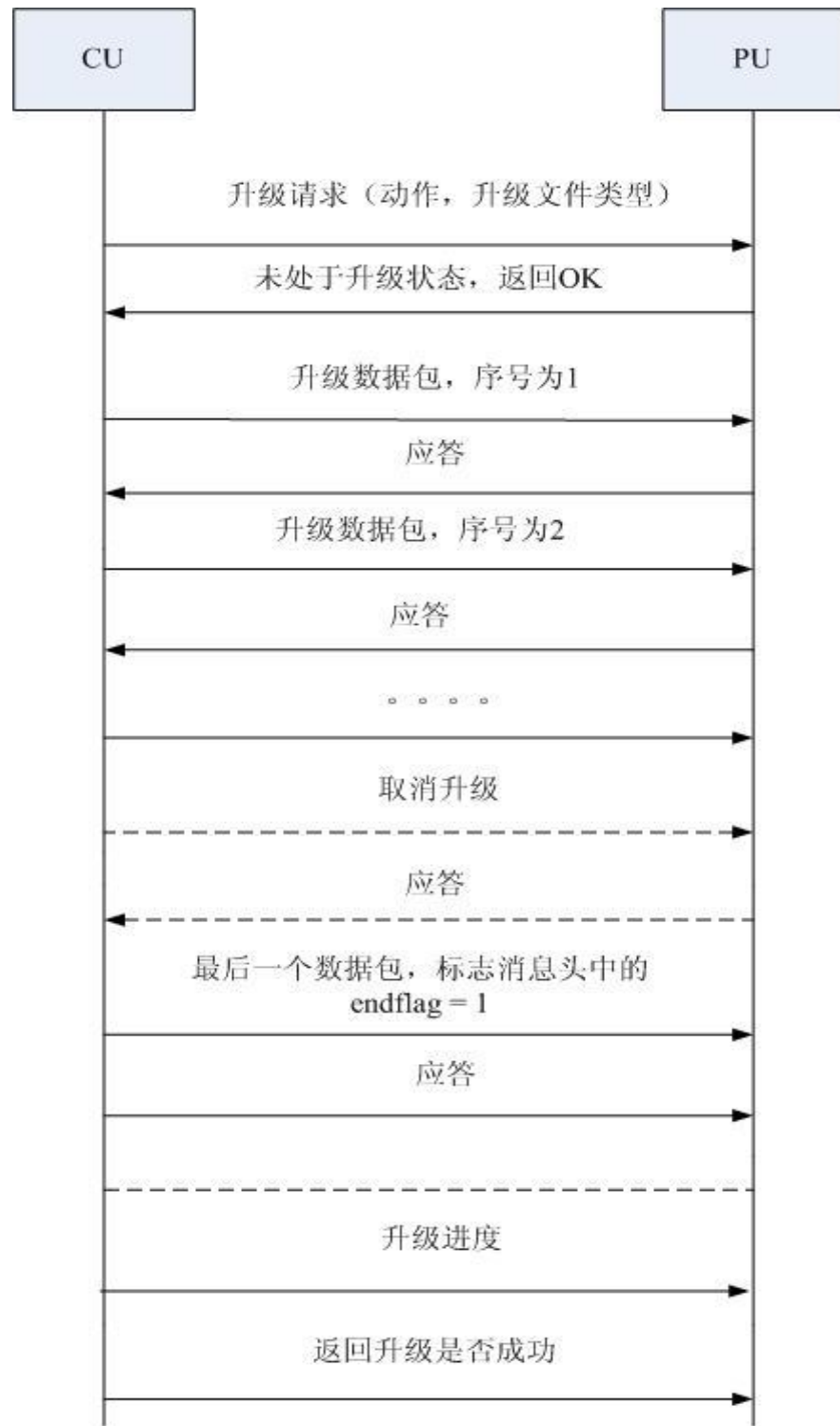


- CU通过主socket发起主动注册语音对讲请求，请求中包含通道号（目前只支持一路对讲，通道号为0）。

- 同时只支持一个CU与PU进行语音对讲，如果已经与其他CU开启了语音对讲，返回已经开始对讲错误码
- PU与CU建立socket用于语音数据传输
- PU通过新建socket发送认领请求，请求中包含通道号（目前只支持一路对讲，通道号为0），消息头的SID字段，为CU的SID号
- CU通过主socket请求语音对讲，请求中包含通道号（目前只支持一路对讲，通道号为0）
- PU鉴权成功后，PU开启捕获模块，开始语音数据发送，语音数据的包序号从1开始递增，同时CU也开始语音数据发送，语音数据的包序号从1开始递增
- CU请求关闭语音对讲，请求中包含通道号，PU停止捕获模块，停止语音数据发送，PU返回应答信息，关闭新建socket，CU收到应答信息后，停止向PU发送语音数据，关闭新建socket

4.12 升级流程

目前升级采用的是被动升级，即需要CU主动发起升级请求，主要流程如下：



升级流程

- CU通过主socket请求进行系统升级，消息，如果PU中处于升级状态，返回已经处于升级错误码，如果CU无升级权限，返回无权限错误码，CU停止升级
- CU开始发送升级数据时，包序号从1开始递增，另外需要设置消息头中的EndFlag字段，如果是最后一个数据包EndFlag = 1，其他EndFlag = 0
- PU每收到一个数据包都会回一个应答信息，如果数据出错，会在应答信息的Ret字段中标

识，并且停止升级

- 收到最后一个数据包后，PU开始升级，升级结束后会返回升级是否成功信息，升级成功后PU重启
- 在发送数据过程中，CU可以取消升级，一旦PU开始升级，CU就不能取消升级了
- 升级开始后，PU会通过UPGRADE_PROGRESS命令告诉CU升级的进度，进度在ret中标志（0~100），该命令CU不需要返回应答，升级是否成功，通过UPGRADE_PROGRESS命令告知，ret = 515: 成功， ret = 514: 失败

4.13 录像管理流程

4.14 图片管理流程

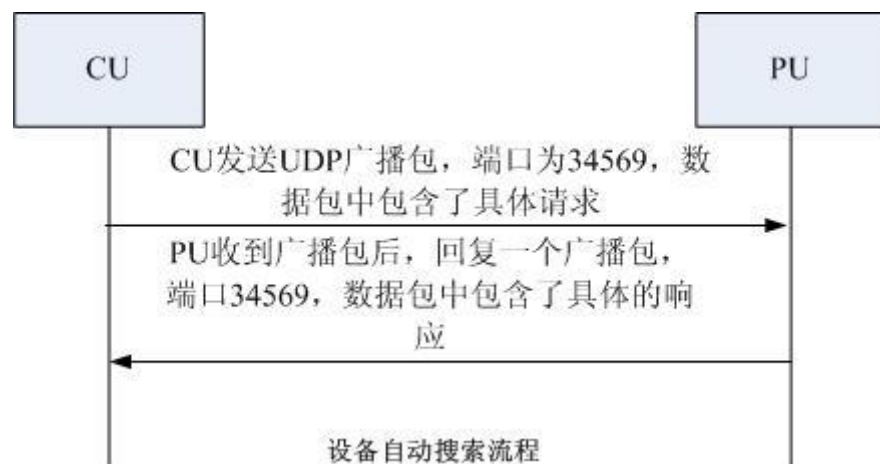
4.15 日志管理流程

4.16 硬盘管理流程

4.17 用户管理流程

4.18 服务管理流程

4.19 设备自动搜索流程



- 采用的是UDP广播包，发送，双方的UDP端口固定使用34569

5 语法与约定

5.1 登录登出保活协议（C1）

5.1.1 普通登录

1 登录请求

消息名称	CU_PU_登录请求
方 向	CU→ PU
功能描述	CU请求登陆
消息内容	具体见下表
说 明	无

登录请求消息定义

```
<! ELEMENT LOGIN_REQ(UserName, PassWord, EncodeType, LonginType) >
<!-- UserName: 用户名, 即在PU端建立的帐号 -->
<! ELEMENT UserName (#STRING)>

<!-- PassWord: 密码 -->
<! ELEMENT PassWd (#STRING)>

<!-- EncodeType: 加密方式: 不加密, 3DES -->
<! ELEMENT EncodeType (None | 3DES)>

<!-- LonginType: 登陆方式: GUI,Console,DVRIP-Web,DVRIP-SNS ,DVRIP-Mobile,DVRIP-
Server , DVRIP-Upgrade , DVRIP-AutoSearch , DVRIP-NetKeyboard -->
<! ELEMENT LonginType(GUI | Console | DVRIP-Web | DVRIP-SNS| DVRIP-Mobile| DVRIP-
Server|DVRIP-Upgrade| DVRIP-AutoSearch| DVRIP-NetKeyboard)>
```

消息体示例如下:

```
{
  "EncryptType" : "MD5",
  "LoginType" : "DVRIP-Web",
  "PassWord" : "6QNMIQGe",
  "UserName" : "admin"
}
```

2 登录请求返回

消息名称	PU_CU_登录请求响应消息
方 向	PU→ CU
功能描述	CU请求登陆响应消息

消息内容 具体见下表
说明 无

登录请求响应消息定义

```
<! ELEMENT LOGIN_RSP(Ret, SessionID, DeviceType, ChnNum, ExtraChannel, AliveInterval) >
<!-- Ret: 返回值, 其他参数只有在返回成功时才有意义 -->
<! ELEMENT Ret (#NUMBER)>

<!-- SessionID: 会话ID, 8位的字符串组成, 每个CU登录成功后, PU都会返回一个唯一的SessionID给CU, 此后的通信就需要依赖于此SessionID -->
<! ELEMENT SessionID (#STRING)>

<!-- DeviceType: 设备类型 -->
<! ELEMENT DeviceType (DVR | DVS | IPC ) >

<!-- ChnNum: 设备通道数 -->
<! ELEMENT ChnNum (#NUMBER) >

<!--ExtraChannel: 扩展通道数, 即组合编码通道-->
<! ELEMENT ExtraChannel (#NUMBER) >

<!-- AliveInterval: PU 与CU进行保活的时间周期, 秒为单位 -->
<! ELEMENT AliveInterval (#NUMBER) >
```

消息体示例如下:

```
{
  "AliveInterval" : 20,
  "ChannelNum" : 8,
  "ExtraChannel" : 1,
  "DeviceType" : "DVR",
  "Ret" : 100,
  "SessionID" : "0x0000000B"}
```

5.1.2 主动注册登录

1 登录请求

消息名称 PU_CU_登录请求

方 向	PU—> CU
功能描述	PU请求登陆
消息内容	具体见下表
说 明	无

登录请求消息定义

```
<! ELEMENT LOGIN_REQ(UserName, PassWord, EncodeType, LonginType) >
<!-- UserName: 用户名，即在CU端建立的帐号 -->
<! ELEMENT UserName (#STRING)>

<!-- PassWord: 密码 -->
<! ELEMENT PassWd (#STRING)>

<!-- EncodeType: 加密方式：不加密，3DES -->
<! ELEMENT EncodeType (None | 3DES)>

<!-- DeviceType: 设备类型 -->
<! ELEMENT DeviceType (DVR | DVS | IPC ) >

<!-- ChnNum: 设备ID -->
<! ELEMENT DeviceID (#NUMBER) >

<!-- ChnNum: 设备通道数 -->
<! ELEMENT ChnNum (#NUMBER) >
```

消息体示例如下：

```
{
  "UserName" : "admin",
  "PassWord" : "6QNMIQGe",
  "EncryptType": "NONE",
  "DeviceType": "DVR",
  "DeviceID": "123456",
  "ChannelNum": "16"
}
```

2 登录请求返回

消息名称	CU_PU_登录请求响应消息
方 向	CU—> PU

功能描述	PU请求登陆响应消息
消息内容	具体见下表
说 明	无

登录请求响应消息定义

```

<! ELEMENT LOGIN_RSP(Ret, SessionID) >

<!-- SessionID: 会话ID, 8位的字符串组成, 每个PU登录成功后, CU都会返回一个唯一的
SessionID给PU, 此后的通信就需要依赖于此SessionID -->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值, 其他参数只有在返回成功时才有意义 -->
<! ELEMENT Ret (#NUMBER)>

<!-- Ret: 配置名字 -->
<! ELEMENT Name (#STRING)>

消息体示例如下:

{
  "Name": "Login",
  "Ret": 100,
  "SessionID": "0x00000001"
}

```

5.1.3 登出

1 主动登出请求

消息名称	CU_PU_登出请求
方 向	CU→PU
功能描述	CU请求登出
消息内容	具体见下表
说 明	无

登出请求消息定义

```

<! ELEMENT LOGOUT_REQ(Name,SessionID) >

<!-- Name: 用户名, 即在PU端建立的帐号 -->
<! ELEMENT Name (#STRING)>

```

```
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
```

消息体示例如下:

```
{
  "Name" : ""
  " SessionID " : "00000001",
}
```

2 主动登出请求返回

消息名称	PU_CU_登出请求响应消息
方 向	PU—> CU
功能描述	CU请求登陆响应消息
消息内容	具体见下表
说 明	无

登录请求响应消息定义

```
<! ELEMENT LOGOUT_RSP(Name,SessionID) >
```

```
<!-- Name: 用户名, 即在PU端建立的帐号 -->
```

```
<! ELEMENT Name (#STRING)>
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

```
<!-- Ret: 返回值 -->
```

```
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "Name" : ""
  " SessionID " : "00000001",
}
```

3 强制登出请求

消息名称	PU_CU_强制登出请求
------	--------------

方 向	PU→ CU
功能描述	PU强制将登录的用户登出
消息内容	具体见下表
说 明	无

强制登出请求消息定义

```
<! ELEMENT FORCELOGOUT_REQ(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
```

消息体示例如下:

```
{
    " SessionID " : "00000001",
}
```

强制登录请求响应消息定义

```
<! ELEMENT FORCELOGOUT_RSP(SessionID , Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

5.1.4 保活

6.2.3.1 保活请求

消息名称	CU_PU_保活请求
方 向	CU→ PU
功能描述	CU根据登录返回的保活周期，请求与PU进行保活
消息内容	具体见下表
说 明	无

登出请求消息定义

```
<! ELEMENT   KEEPALIVE_REQ(SessionID, Name) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>
```

消息体示例如下：

```
{
  "Name" : "KeepAlive",
  "SessionID" : "0x0000000B"
}
```

6.2.3.2 保活返回

消息名称	PU_CU_保活请求响应消息
方 向	PU→ CU
功能描述	CU请求保活响应消息
消息内容	具体见下表
说 明	无

登录请求响应消息定义

```
<! ELEMENT   KEEPALIVE_RSP(SessionID , Name, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>
```

```
<!-- Ret: 返回值 -->
```

```
<! ELEMENT   Ret (#NUMBER)>
```

消息体示例如下：

```
{
  "Name" : "KeepAlive",
  "Ret" : 100,
  "SessionID" : "0x0000000B"
}
```

5.2 获取设备信息 (C2)

5.2.1 获取系统信息请求

消息名称	CU_PU_获取系统信息请求
方 向	CU—> PU
功能描述	CU获取设备的基本系统信息
消息内容	具体见下表
说 明	无

获取系统信息请求消息定义

```

<! ELEMENT  SYSINFO_REQ(SessionID, Name) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>
<!--Name: 系统信息名称, 包括设备信息, 设备状态, 磁盘信息-->
<! ELEMENT Name (#STRING) -->

```

消息体示例如下:

```

//设备信息
{
    "SessionID" : "00000001",
    "Name" : "SystemInfo"
}
//磁盘信息
{
    "Name" : "StorageInfo",
    "SessionID" : "0x00000002"
}
//设备状态
{
    "Name" : " WorkState",
    "SessionID" : "0x00000002"
}

```

5.2.2 返回系统信息

消息名称	PU_CU_系统信息请求响应消息
方 向	PU—> CU
功能描述	CU请求系统信息响应消息
消息内容	具体见下表
说 明	无

系统信息请求响应消息定义

```
<! ELEMENT  SYSINFO_RSP(SessionID, Ret, Name, SystemInfo) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT  Ret (#NUMBER)>

<!-- 系统信息，包括设备信息（SystemInfo），设备状态（WorkState），磁盘信息
（StorageInfo），见《JsonConfigFormat》-->
<! ELEMENT  SystemInfo (#STRUCT)>
```

消息体示例如下：

```
//设备信息
{
  "Name" : "SystemInfo",
  "Ret" : 100,
  "SessionID" : "0xb",
  "SystemInfo" : {
    "AlarmInChannel" : 8,
    "AlarmOutChannel" : 2,
    "BuildTime" : "2009-02-13 12:03:12",
    "EncryptVersion" : "Unknown",
    "HardWareVersion" : "Unknown",
    "SerialNo" : "00000000",
    "SoftWareVersion" : "JF1.00.R01",
    "TalkInChannel" : 1,
    "TalkOutChannel" : 1,
    "VideoInChannel" : 8,
    "VideoOutChannel" : 1,
    "ExtraChannel" : 1,
    "AudioInChannel" : 4
    "DeviceRunTime": "0x0000019A"
  }
}

//磁盘信息
{
```



```
"Name" : "StorageInfo",
"Ret" : 100,
"SessionID" : "0x2",
"StorageInfo" : [
  {
    "PartNumber" : 1,
    "Partition" : [
      {
        "DirverType" : 0,
        "IsCurrent" : true,
        "LogicSerialNo" : 0,
        "NewEndTime" : "2009-02-16 11:52:05",
        "NewStartTime" : "2000-11-30 10:58:42",
        "OldEndTime" : "2063-11-30 00:00:23",
        "OldStartTime" : "2000-00-00 00:00:00",
        "RemainSpace" : "0x00073130",
        "Status" : 0,
        "TotalSpace" : "0x000746FC"
      },
      {
        "DirverType" : 0,
        "IsCurrent" : false,
        "LogicSerialNo" : 0,
        "NewEndTime" : "0000-00-00 00:00:00",
        "NewStartTime" : "0000-00-00 00:00:00",
        "OldEndTime" : "0000-00-00 00:00:00",
        "OldStartTime" : "0000-00-00 00:00:00",
        "RemainSpace" : "0x00000000",
        "Status" : 0,
        "TotalSpace" : "0x00000000"
      }
    ],
    "PhysicalNo" : 0
  }
]
```

//工作状态

```

{
  "Name" : "WorkState",
  "Ret" : 100,
  "SessionID" : "0x3",
  "WorkState" :
  {
    "AlarmState" :
    {
      "AlarmIn" : 8,
      "AlarmOut" : 1,
      "VideoBlind" : 0,
      "VideoLoss" : 1,
      "VideoMotion" : 0
    },
    "ChannelState" :
    [
      { "Bitrate" : 13, "Record" : false },
      { "Bitrate" : 9, "Record" : false },
      { "Bitrate" : 14, "Record" : false },
      { "Bitrate" : 13, "Record" : false },
      { "Bitrate" : 14, "Record" : false },
      { "Bitrate" : 14, "Record" : false },
      { "Bitrate" : 13, "Record" : false },
      { "Bitrate" : 9, "Record" : false },
    ]
  }
}

```

5.3 设置/获取配置相关请求（C3）

5.3.1 设置配置请求

消息名称	CU_PU_设置配置请求
方 向	CU→ PU
功能描述	CU向PU设置配置
消息内容	具体见下表
说 明	无

设置配置请求消息定义

```

<! ELEMENT CONFIG_SET(SessionID, ConfigName) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- ConfigName: 配置名称: "AVEnc.Encode", "AVEnc.VideoWidget", "AVEnc.VideoColor",
"Record", "Detect.MotionDetect", "Detect.BlindDetect", "Detect.LossDetect", "Alarm.LocalAlarm",
"Alarm.NetAlarm", "Alarm.NetIPConflict", "Alarm.NetAbort", "Storage.StorageNotExist",
"Storage.StorageLowSpace", "Storage.StorageFailure", "Alarm.PTZAlarmProtocol",
"NetWork.NetCommon", "NetWork.NetIPFilter", "NetWork.NetDHCP", "NetWork.NetDDNS",
"NetWork.NetEmail", "NetWork.NetNTP", "NetWork.NetPPPoE",
"NetWork.NetDNS", "NetWork.NetARSP", "NetWork.NetMobile", "NetWork.Upnp", "NetWork.NetFTP",
"NetWork.AlarmServer", "Uart.Comm", "Uart.PTZ", "Uart.PTZPreset", "Uart.PTZTour",
"Video.Tour", "Video.GUISet", "Video.TVAdjust", "Video.AudioInFormat", "Video.Play",
"General.General", "General.Location", "General.AutoMaintain", "ChannelTitle",
"Storage.Snapshot", "AVEnc.CombineEncode", "NetWork.NetFTP" 详细见《JsonConfigFormat》
-->
<! ELEMENT ConfigName(#STRING)>

<!-- ConfigObject: 具体配置信息的字符串描述, 见《JsonConfigFormat》 -->
<! ELEMENT Encode (#STRUCT) -->

设置编码配置请求示例如下:

{
    "SessionID": "0x1234",
    "Name": "General.General"
}

```

返回消息:

设置编码配置请求响应消息定义

```

<! ELEMENT CONFIG_SET_RSP(SessionID, ConfigName, Ret) >
设置编码配置返回示例如下:
{
    "General.General": {},
    "Name": "General.General"
    "SessionID": "0x1234",
    "Ret": 100,
}

```

```
}
```

5.3.2 获取配置请求

消息名称	CU_PU_获取配置请求
方 向	CU→ PU
功能描述	CU向PU获取配置
消息内容	具体见下表
说 明	无

获取配置请求消息定义

```
<! ELEMENT CONFIG_GET(SessionID, ConfigName) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

```
<!-- ConfigName: 配置名称: 见设置配置请求 -->
```

```
<! ELEMENT ConfigName(#STRING)>
```

获取串口配置请求的消息体如下:

```
{  
  "SessionID": "0x1234",  
  "Name": "Uart .Comm"  
}
```

返回消息:

获取配置请求响应消息定义

```
<! ELEMENT CONFIG_GET_RSP(SessionID, Ret, Name, ConfigObject) >
```

获取串口配置成功时定义的返回消息格式如下，其它配置类似，失败时由**Ret**字段描述相应的失败信息。

```
{  
  "SessionID": "0x1234",  
  "Ret": 100,  
  "Name": "Uart .Comm"  
  "Comm": {}  
}
```

5.3.3 获取默认配置请求

消息名称	CU_PU_获取默认配置请求
方 向	CU→ PU
功能描述	CU向PU设置叠加配置
消息内容	具体见下表
说 明	无

设置配置请求消息定义

```
<! ELEMENT DEFAULT_CONFIG_GET(SessionID, ConfigName) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

```
<!-- ConfigName: 配置名称: 见设置配置请求 -->
```

```
<! ELEMENT ConfigName(#STRING)>
```

获取串口配置请求的消息体如下:

```
{  
  "SessionID" : "0x1234",  
  "Name" : "Uart .Comm"  
}
```

返回消息:

设置叠加配置请求响应消息定义

```
<! ELEMENT DEFAULT_CONFIG_GET_RSP(SessionID, Ret, Name, ConfigObject) >
```

获取串口配置成功时定义的返回消息格式如下，其它配置类似，失败时由Ret字段描述相应的失败信息。

```
{  
  "SessionID" : "0x1234",  
  "Ret" : 100,  
  "Name" : "Uart .Comm"  
  "Comm" : {}  
}
```

5.4 获取能力级请求 (C12)

消息名称	CU_PU_获取能力集请求
方 向	CU→ PU
功能描述	CU向PU请求获取系统能力集
消息内容	具体见下表
说 明	无

获取系统功能列表请求消息定义

```
<! ELEMENT ABILITY_GET(SessionID, Name) >
<!-- SessionID: 会话ID-->
<!--Name 能力集合名称, 目前有" EncodeCapability"(编码能力集), "BlindCapability"(遮挡能力集), "MotionArea"(动检能力集), "DDNSService"(DDNS服务类型能力集), "ComProtocol"(串口协议能力集), "PTZProtocol (云台能力集)", "TalkAudioFormat"(对讲能力集), "MultiLanguage"(语言支持能力集)-->
<! ELEMENT SessionID (#STRING)>
<! ELEMENT Name(#SRING)>
获取编码能力集示例如下:
{
    "SessionID": "0x1234",
    "Name": "SystemFunction"
}
```

返回消息:

获取系统功能列表请求响应消息定义

```
<! ELEMENT ABILITY_GET_RSP(SessionID, Ret, Name, AbilityObject) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
<!--Name 能力集名称-->
<! ELEMENT Name(#STRING)>
<!--AbilityObject 具体的能力集对象-->
消息体示例如下:
{
```

```
    "SessionID" : "0x1234",  
    "Ret" : 100,  
    "Name" : "SystemFunction",  
    "SystemFunction" : {}  
}
```

5.6 云台控制（C13）

5.4.1 云台控制请求

消息名称	CU_PU_进行云台操作
方 向	CU—> PU
功能描述	CU向PU请求云台操作
消息内容	具体见下表
说 明	无

云台操作请求消息定义

```
<! ELEMENT PTZ_REQ(SessionID, Name, Operation) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Operation: 见《JsonConfigFormat》 -->
<! ELEMENT Operation (#STRUCT)>
```

消息体示例如下：

```
{
  "SessionID" : "0x1234",
  "Name" : "PTZControl",
  "OPPTZControl" : {
    "Command" : "DirectionUp",
    "Parameter" : {
      "AUX" : { "Number" : 0, "Status" : "On" },
      "Channel" : 0,
      "MenuOpts" : "Enter",
      "Pattern" : "Start",
      "Preset" : 0,
      "Step" : 5,
      "Tour" : 0 }
    },
  "SessionID" : "0x14"
}
```

返回消息：

云台操作请求响应消息定义

```
<! ELEMENT PTZ_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "SessionID": "0x1234",
  "Ret": 100,
  "Name": ""
}
```

5.5 监视控制 (C14)

5.5.1 监视控制请求

消息名称	CU_PU_进行监视控制
方 向	CU→ PU
功能描述	CU向PU请求监视控制
消息内容	具体见下表
说 明	无

监视控制请求消息定义

```
<! ELEMENT MONITOR_REQ(SessionID, Operation) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Operation: 见《JsonConfigFormat》 -->
<! ELEMENT Operation (#STRUCT)>
```

消息体示例如下:

```
{
  "Name": "OPMonitor",
  "OPMonitor": {
    "Action": "Claim",
    "Parameter": {
```

```

        "Channel" : 0,
        "StreamType" : "Main",
        "TransMode" : "TCP"
    }
},
"SessionID" : "0xb"
}

```

返回消息:

监视控制请求响应消息定义

```

<! ELEMENT  MONITOR_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT  Ret (#NUMBER)>

```

消息体示例如下:

```

{
  "Name" : " OPMonitor ",
  "Ret" : 100,
  "SessionID" : "0x0000000C"
}

```

5.6 回放控制 (C15)

5.6.1 回放控制请求

消息名称	CU_PU_进行回放控制
方 向	CU→ PU
功能描述	CU向PU请求回放控制
消息内容	具体见下表
说 明	无

回放控制请求消息定义

```

<! ELEMENT  PLAY_REQ(SessionID, Name, Operation) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

```

```
<!-- Operation: 见《JsonConfigFormat》 -->
```

```
<! ELEMENT Operation (#STRUCT)>
```

消息体示例如下:

```
{  
  "Name" : "OPPlayBack",  
  "OPPlayBack" : {  
    "Action" : "Claim",  
    "EndTime" : "2009-02-16 10:23:30",  
    "Parameter" : {  
      "FileName" : "/idea0/2009-02-16/001/10.08.41-  
10.23.30[R][@7cb][0].h264",  
      "TransMode" : "TCP"  
    },  
    "StartTime" : "2009-02-16 10:08:41"  
  },  
  "SessionID" : "0x2"  
}
```

返回消息:

回放控制请求响应消息定义

```
<! ELEMENT PLAY_RSP(SessionID, Ret) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

```
<!-- Ret: 返回值 -->
```

```
<! ELEMENT Ret (#NUMBER)>
```

消息体如下:

```
{  
  "Name" : "OPPlayBack",  
  "Ret" : 100,  
  "SessionID" : "0x00000002"  
}
```

5.7 语音对讲(C16)

5.7.1 语音对讲请求

消息名称	CU_PU_进行语音对讲
方 向	CU→ PU
功能描述	CU向PU请求语音对讲
消息内容	具体见下表
说 明	无

语音对讲请求消息定义

```
<! ELEMENT TALK_REQ(SessionID, Operation) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Operation: 见《JsonConfigFormat》 -->
<! ELEMENT Operation (#STRUCT)>
```

消息体示例如下:

```
{
  "Name" : "OPTalk",
  "OPTalk" : {
    "Action" : "Claim",
    "AudioFormat" : {
      "BitRate" : 0,
      "EncodeType" : "G711_ALAW",
      "SampleBit" : 8,
      "SampleRate" : 8 }
    },
  "SessionID" : "0x3"
}
```

返回消息:

语音对讲请求响应消息定义

```
<! ELEMENT TALK_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "Name" : "OPTalk",
  "Ret" : 100,
  "SessionID" : "0x0000000C"
}
```

5.8 文件查询 (C17)

消息名称	CU_PU_进行文件查询
方 向	CU→ PU
功能描述	CU向PU请求查询文件，这里文件包括：录像文件，图片文件，日志文件
消息内容	具体见下表
说 明	无

文件查询请求消息定义

```
<! ELEMENT FILESEARCH_REQ(SessionID, Name, OPFileQuery) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
<!-- Name: 操作名称 -->
<!-- OPFileQuery: 见《JsonConfigFormat》 -->
<! ELEMENT OPFileQuery(#STRUCT)>
```

消息体示例如下：

```
{
  "Name" : "OPFileQuery",
  "OPFileQuery" : {
    "BeginTime" : "2009-02-08 00:00:00",
    "Channel" : 0,
    "DriverTypeMask" : "0x0000FFFF",
    "EndTime" : "2009-02-08 09:53:03",
    "Event" : "AMRH",
    "Type" : "h264"
  },
  "SessionID" : "0xb"
}
```

返回消息：

日志管理请求响应消息定义

```
<! ELEMENT FILESEARCH_RSP(SessionID, Name, Ret, FileList) >
```

```

<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT   Ret (#NUMBER)>

<!-- FileList: 文件信息列表, 每条查询结果为一个FileInfo, 见《JsonConfigFormat》 -->
<! ELEMENT   FileList(#STRUCT)>

```

消息体示例如下:

```

{
  "Name" : " FileList ",
  " FileList " : [
    {
      "BeginTime" : "2009-02-16 10:08:41",
      "DiskNo" : 0,
      "EndTime" : "2009-02-16 10:23:30",
      "FileLength" : "0x0000B000",
      "FileName" : "/idea0/2009-02-16/001/10.08.41-10.23.30[R][@7cb][0].h264",
      "SerialNo" : 0
    },
    {
      "BeginTime" : "2009-02-16 10:24:08",
      "DiskNo" : 0,
      "EndTime" : "2009-02-16 10:34:42",
      "FileLength" : "0x00008000",
      "FileName" : "/idea0/2009-02-16/001/10.24.08-10.34.42[R][@85d][0].h264",
      "SerialNo" : 0
    },
  ],
  "Ret" : 100,
  "SessionID" : "0x2"
}

```

5.9 日志查询 (C17)

消息名称 CU_PU_进行文件查询

方 向 CU→ PU

功能描述	CU向PU请求查询日志信息
消息内容	具体见下表
说明	无

日志管理请求消息定义

```
<! ELEMENT LOGSEARCH_REQ(SessionID, Name, OPLogQuery) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- OPLogQuery: 见《JsonConfigFormat》 -->
<! ELEMENT OPLogQuery (#STRUCT)>
```

消息体示例如下:

```
{
  "Name" : "OPLogQuery",
  "OPLogQuery" : {
    "BeginTime" : "2009-02-16 00:00:00",
    "EndTime" : "2009-02-17 00:00:00",
    "LogPosition" : 0,
    "Type" : "LogAll"
  },
  "SessionID" : "0x2"
}
```

返回消息:

日志管理请求响应消息定义

```
<! ELEMENT LOGSEARCH_RSP(SessionID, Ret, LogList) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>

<!-- LogList: 文件信息列表, 每条查询结果为一个LogList, 见《JsonConfigFormat》 -->
<! ELEMENT LogList(#STRUCT)>
```

消息体示例如下:

```
{
```

```

"LogList" : [
  {
    "Data" : "default",
    "Position" : 2038,
    "Time" : "2009-02-16 11:09:28",
    "Type" : "LogIn",
    "User" : "System"
  },
  {
    "Data" : "Abnormal",
    "Position" : 2039,
    "Time" : "2009-02-16 11:09:30",
    "Type" : "ShutDown",
    "User" : "System"
  },
],
"Name" : "LogList",
"Ret" : 100,
"SessionID" : "0x3"
}

```

5.10 系统时间查询 (C18)

消息名称	CU_PU_查询PU系统时间
方 向	CU—> PU
功能描述	CU向PU请求系统管理
消息内容	具体见下表
说 明	无

系统时间查询请求消息定义

```

<! ELEMENT  TIMEQUERY_REQ (SessionID, Name) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>
<!-- Name: 操作名称 -->
<!-- OPTimeQuery: 见《JsonConfigFormat》-->

```

消息体示例如下：

设备操作

```
{  
  "Name" : "OPTimeQuery",  
  "SessionID" : "0x2"  
}
```

返回消息:

系统时间查询请求响应消息定义

```
<! ELEMENT TIMEQUERY_RSP(SessionID, Ret, OPTimeQuery) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT SessionID (#STRING)>  
  
<!-- Ret: 返回值 -->  
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{  
  "OPTimeQuery" : "2009-09-24 09:20:24",  
  "Ret" : 100,  
  "SessionID" : "0x2"  
}
```

5.11 系统管理 (C18)

5.11.1 系统管理请求

消息名称	CU_PU_进行系统管理
方 向	CU—> PU
功能描述	CU向PU请求系统管理
消息内容	具体见下表
说 明	无

系统管理请求消息定义

```
<! ELEMENT SYSMANAGER_REQ(SessionID, Name, Operation) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT SessionID (#STRING)>  
<!--Name: 请求管理的名称，目前有 "OPMachine","OPLogManager", "OPDefaultConfig",  
"OPTimeSetting",-->
```

<!-- Operation: 见《JsonConfigFormat》 -->

<! ELEMENT Operation (#STRUCT)>

消息体示例如下:

设备操作

```
{  
  "Name" : "OPMachine",  
  "OPMachine" : {  
    "Action" : "Reboot"  
  },  
  "SessionID" : "0x2"  
}
```

日志管理

```
{  
  "Name" : "OPLogManager",  
  "OPLogManager" : {  
    "Action" : "RemoveAll"  
  },  
  "SessionID" : "0x3"  
}
```

恢复默认

```
{  
  "Name" : "OPDefaultConfig",  
  "OPDefaultConfig" : {  
    "Account" : false,  
    "Alarm" : false,  
    "CommPtz" : false,  
    "Encode" : false,  
    "General" : true,  
    "NetCommon" : false,  
    "NetServer" : false,  
    "Preview" : false,  
    "Record" : false  
  },  
  "SessionID" : "0x3"  
}
```

时间设置

```
{  
  "Name" : "OPTimeSetting",  
  "OPTimeSetting" : "2009-09-24 09:08:50",  
  "SessionID" : "0x2"  
}
```

返回消息:

日志管理请求响应消息定义

```
<! ELEMENT  SYSMANAGER_RSP(Name,SessionID, Ret) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT   SessionID (#STRING)>  
  
<!-- Ret: 返回值 -->  
<! ELEMENT   Ret (#NUMBER)>  
消息体示例如下:  
{  
  Name:"",  
  "Ret" : 100,  
  "SessionID" : "0x00000002"  
}
```

5.12 硬盘管理 (C19)

5.12.1 硬盘管理请求

消息名称	CU_PU_进行硬盘管理
方 向	CU—> PU
功能描述	CU向PU请求硬盘管理
消息内容	具体见下表
说 明	无

硬盘管理请求消息定义

```
<! ELEMENT  DSIKMANAGER_REQ(Name,SessionID, Operation) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT   SessionID (#STRING)>  
<!--Name: 请求管理的名称, 目前有 "OPStorageManager",-->  
<!-- Operation: 见《JsonConfigFormat》 -->
```

```
<! ELEMENT Operation (#STRUCT)>
```

消息体示例如下:

```
{  
"Name" : "OPStorageManager",  
"OPStorageManager" : { "Action" : "SetType", "PartNo" : 1, "SerialNo" : 0, "Type" : "ReadWrite" },  
"SessionID" : "0x3"}
```

返回消息:

硬盘管理请求响应消息定义

```
<! ELEMENT DSIKMANAGER_RSP(Name,SessionID, Ret) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

```
<!-- Ret: 返回值 -->
```

```
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{  
"Name" : "", "  
Ret" : 100,  
"SessionID" : "0x00000003"  
}
```

5.13 用户管理 (C20)

5.13.1 获取完整权限列表

消息名称	CU_PU_获取完整权限列表请求
方 向	CU→ PU
功能描述	CU向PU请求获取完整权限列表
消息内容	具体见下表
说 明	无

获取完整权限列表请求消息定义

```
<! ELEMENT FULLAUTHORITYLIST_GET(SessionID) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT SessionID (#STRING)>
```

返回消息:

获取完整权限列表请求响应消息定义

```
<! ELEMENT FULLAUTHORITYLIST_GET_RSP(SessionID, Ret, Users) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>

<!-- Users: 用户权限列表, 见《JsonConfigFormat》 -->
<! ELEMENT Users(#STRUCT) -->
消息体示例如下:
{
  "AuthorityList" : [ "ShutDown", "RecordConfig", "Backup", "StorageManager", "PTZControl",
"Account", "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain", "GeneralConfig",
"EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig", "PtzConfig",
"DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04", "Monitor_05",
"Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03", "Replay_04",
"Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
  "Ret" : 100,
  "SessionID" : "0x00000002"
}
```

5.13.2 获取用户信息

消息名称	CU_PU_获取用户权限配置请求
方 向	CU→ PU
功能描述	CU向PU请求用户权限配置
消息内容	具体见下表
说 明	无

获取用户权限配置请求消息定义

```
<! ELEMENT USERS_GET(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
```

返回消息:

获取用户权限配置请求响应消息定义

```
<! ELEMENT  USERS_GET_RSP(SessionID, Ret, Users) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT  Ret (#NUMBER)>

<!-- Users: 用户权限配置, 见《JsonConfigFormat》 -->
<! ELEMENT  Users(#STRUCT) -->
消息体示例如下:
{
"Ret" : 100,
"SessionID" : "0x00000002",
"Users" : [
    {
        "AuthorityList" : [ "ShutDown", "RecordConfig", "Backup", "StorageManager",
"PTZControl", "Account", "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain",
"GeneralConfig", "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig",
"PtzConfig", "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",
"Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",
"Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
        "Group" : "admin",
        "Memo" : "user's account",
        "Name" : "user",
        "Password" : "9kwf1kJHJ",
        "Reserved" : true,
        "Sharable" : true
    },
    {
        "AuthorityList" : [ "ShutDown", "RecordConfig", "Backup", "StorageManager",
"PTZControl", "Account", "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain",
"GeneralConfig", "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig",
"PtzConfig", "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",
"Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",
"Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
        "Group" : "admin",
        "Memo" : "admin 's account",
```

```

        "Name" : "admin",
        "Password" : "6QNMIQGe",
        "Reserved" : true,
        "Sharable" : true
    },
]
}

```

5.13.3 获取用户组信息

消息名称	CU_PU_获取用户组权限配置请求
方 向	CU→ PU
功能描述	CU向PU请求用户组权限配置
消息内容	具体见下表
说 明	无

获取用户组权限配置请求消息定义

```

<! ELEMENT GROUPS_GET(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

```

返回消息:

获取用户组权限配置请求响应消息定义

```

<! ELEMENT GROUPS_GET_RSP(SessionID, Ret, Groups) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>

<!-- Groups: 用户组权限配置, 见《JsonConfigFormat》 -->
<! ELEMENT Groups(#STRUCT) -->

```

消息体示例如下:

```

{
  "Groups" : [
    {
      "AuthorityList" : [

```

```

        "ShutDown", "RecordConfig", "Backup", "StorageManager", "PTZControl", "Account",
        "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain", "GeneralConfig",
        "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig", "PtzConfig",
        "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04", "Monitor_05",
        "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03", "Replay_04",
        "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
        "Memo" : "administrator group",
        "Name" : "admin"
    },
    {
        "AuthorityList" : [ "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",
        "Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",
        "Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
        "Memo" : "user group",
        "Name" : "user" }
    ],
    "Ret" : 100,
    "SessionID" : "0x00000002"
}

```

5.13.4 添加用户组

消息名称	CU_PU_添加用户组
方 向	CU→ PU
功能描述	CU向PU请求添加用户组
消息内容	具体见下表
说 明	无

添加用户组请求消息定义

```

<! ELEMENT  ADDGROUP_REQ(SessionID, Name, Groups) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Groups: 用户组的组名不能与已经存在的组冲突, 见《JsonConfigFormat》 -->
<! ELEMENT  Groups(#STRUCT)>

```

消息体示例如下:


```

{
  "Group" : {
    "AuthorityList" : [ "ShutDown", "RecordConfig", "Backup", "StorageManager",
"PTZControl", "Account", "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain",
"GeneralConfig", "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig",
"PtzConfig", "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",
"Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",
"Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
    "Memo" : "group0001",
    "Name" : "group0001"
  },
  "Name" : "Group",
  "SessionID" : "0x2"
}

```

返回消息:

添加用户组请求响应消息定义

```

<! ELEMENT  ADDGROUP_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT  Ret (#NUMBER)>
消息体示例如下:
{
  "Ret" : 100,
  "SessionID" : "0x00000002"
}

```

5.13.5 修改用户组

消息名称	CU_PU_修改用户组
方 向	CU→ PU
功能描述	CU向PU请求修改用户组
消息内容	具体见下表
说 明	无

修改用户组请求消息定义

```
<! ELEMENT  MODIFYGROUP_REQ(SessionID, GroupName, Groups) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- GroupName: 用户组名, 当修改的用户组信息里包括用户组名时, 这里的用户组名为原来的
组名, 否则会修改失败-->
<! ELEMENT   GroupName(#STRING)>

<!-- Groups: 见《JsonConfigFormat》 -->
<! ELEMENT   Groups(#STRUCT)>
```

消息体示例如下:

```
{
  "Group" : {
    "AuthorityList" : [ "ShutDown", "RecordConfig", "Backup", "StorageManager",
"PTZControl", "Account", "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain",
"GeneralConfig", "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig",
"PtzConfig", "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",
"Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",
"Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],
    "Memo" : "group0002",
    "Name" : "group0002"
  },
  "GroupName" : "group0001",
  "SessionID" : "0x00000002"
}
```

返回消息:

修改用户组请求响应消息定义

```
<! ELEMENT  MODIFYGROUP_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT   Ret (#NUMBER)>
```

消息体示例如下:

```
{
```

```
"Ret" : 100,  
"SessionID" : "0x00000002"  
}
```

5.13.6 删除用户组

消息名称	CU_PU_删除用户组
方 向	CU→ PU
功能描述	CU向PU请求删除用户组
消息内容	具体见下表
说 明	无

删除用户组请求消息定义

```
<! ELEMENT  DELETEDGROUP_REQ(SessionID, Name) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT  SessionID (#STRING)>  
  
<!-- Name: 用户组名 -->  
<! ELEMENT  Name(#STRING)>  
  
消息体示例如下:  
{  
  "Name" : "user001",  
  "SessionID" : "0x00000002"  
}
```

返回消息:

删除用户组请求响应消息定义

```
<! ELEMENT  DELETEDGROUP_RSP(SessionID, Ret) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT  SessionID (#STRING)>  
  
<!-- Ret: 返回值 -->  
<! ELEMENT  Ret (#NUMBER)>  
消息体示例如下:  
{  
  "Ret" : 100,  
  "SessionID" : "0x00000002"  
}
```

```
}
```

5.13.7 添加用户

消息名称	CU_PU_添加用户
方 向	CU→ PU
功能描述	CU向PU请求添加用户
消息内容	具体见下表
说 明	无

添加用户请求消息定义

```
<! ELEMENT  ADDUSER_REQ(SessionID, Name, Users) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT   SessionID (#STRING)>  
  
<!-- Users: 见《JsonConfigFormat》 -->  
<! ELEMENT   Users(#STRUCT)>
```

消息体示例如下:

```
{  
  "Name" : "User",  
  "SessionID" : "0x2",  
  "User" : {  
    "AuthorityList" : [  
      "ShutDown", "RecordConfig", "Backup", "StorageManager", "PTZControl", "Account",  
      "SysInfo", "QueryLog", "DelLog", "SysUpgrade", "AutoMaintain", "GeneralConfig",  
      "EncodeConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig", "PtzConfig",  
      "DefaultConfig", "Talk_01", "Monitor_01", "Monitor_02", "Monitor_03", "Monitor_04",  
      "Monitor_05", "Monitor_06", "Monitor_07", "Monitor_08", "Replay_01", "Replay_02", "Replay_03",  
      "Replay_04", "Replay_05", "Replay_06", "Replay_07", "Replay_08" ],  
    "Group" : "admin",  
    "Memo" : "user00001",  
    "Name" : "user001",  
    "Password" : "Jszew7cB",  
    "Reserved" : false,  
    "Sharable" : true  
  }  
}
```

返回消息:

添加用户请求响应消息定义

```
<! ELEMENT  ADDUSER_RSP(SessionID, Ret) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT  SessionID (#STRING)>
```

```
<!-- Ret: 返回值 -->
```

```
<! ELEMENT  Ret (#NUMBER)>
```

消息体示例如下:

```
{  
"Ret" : 100,  
"SessionID" : "0x00000002"  
}
```

5.13.8 修改用户

消息名称 CU_PU_修改用户

方 向 CU→ PU

功能描述 CU向PU请求修改用户信息, 包括了用户名, 密码, 权限

消息内容 具体见下表

说 明 无

修改用户请求消息定义

```
<! ELEMENT  MODIFYUSER_REQ(SessionID, UserName, Users) >
```

```
<!-- SessionID: 会话ID-->
```

```
<! ELEMENT  SessionID (#STRING)>
```

```
<!-- UserName: 用户名, 当修改的用户信息里包括用户名时, 这里的用户名为原来的名, 否则会  
修改失败-->
```

```
<! ELEMENT  UserName(#STRING)>
```

```
<!-- Users: 见《JsonConfigFormat》 -->
```

```
<! ELEMENT  Users(#STRUCT)>
```

消息体示例如下:

```
{  
"User" : {  
  "AuthorityList" : [  
    ]  
  }  
}
```

```

    "ShutDown", "RecordConfig", "Backup", "PTZControl", "Account", "SysInfo",
    "GeneralConfig", "CommConfig", "NetConfig", "AlarmConfig", "VideoConfig", "PtzConfig",
    "DefaultConfig", "Talk_01", "Monitor_02", "Monitor_04", "Monitor_05", "Monitor_07",
    "Monitor_08", "Replay_01", "Replay_02", "Replay_03", "Replay_04", "Replay_05", "Replay_06",
    "Replay_07", "Replay_08" ],
    "Group" : "admin",
    "Memo" : "user00001",
    "Name" : "user001",
    "Password" : "",
    "Reserved" : false,
    "Sharable" : true
  },
  "UserName" : "user001" ,
  "SessionID" : "0x00000002"
}

```

返回消息:

修改用户请求响应消息定义

```

<! ELEMENT  MODIFYUSER_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT  Ret (#NUMBER)>
消息体示例如下:
{
  "Ret" : 100,
  "SessionID" : "0x00000002"
}

```

5.13.9 删除用户

消息名称	CU_PU_删除用户
方 向	CU→ PU
功能描述	CU向PU请求删除用户
消息内容	具体见下表
说 明	无

删除用户请求消息定义

```

<! ELEMENT  DELETEUSER_REQ(SessionID, Name) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- Name: 用户名 -->
<! ELEMENT   Name (#STRING)>

```

消息体示例如下:

```

{
"Name" : "user001",
"SessionID" : "0x00000002"
}

```

返回消息:

删除用户请求响应消息定义

```

<! ELEMENT  DELETEGROUP_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT   Ret (#NUMBER)>

```

消息体示例如下:

```

{
"Ret" : 100,
"SessionID" : "0x00000002"
}

```

5.13.10 修改用户密码

消息名称	CU_PU_修改用户密码
方 向	CU→ PU
功能描述	CU向PU请求修改用户密码
消息内容	具体见下表
说 明	无

修改用户密码请求消息定义

```

<! ELEMENT      MODIFYPASSWORD_REQ(SessionID,   UserName,   OldPassword,
NewPassword, EncryptType) >

```

```
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- UserName: 用户名-->
<! ELEMENT   UserName(#STRING)>

<!-- OldPassword: 旧密码-->
<! ELEMENT   OldPassword(#STRING)>

<!-- NewPassword: 新密码-->
<! ELEMENT   NewPassword(#STRING)>

<!-- EncryptType: 密码加密方式-->
<! ELEMENT   EncryptType(#STRING)>
```

消息体示例如下:

```
{
  "EncryptType" : "MD5",
  "NewPassWord" : "tlJwpbo6",
  "PassWord" : "9kwf1kHJ",
  "SessionID" : "0x0E4EC747",
  "UserName" : "user"
}
```

返回消息:

修改用户密码请求响应消息定义

```
<! ELEMENT   MODIFYPASSWORD_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT   SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT   Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "Ret" : 100,
  "SessionID" : "0x00000002"
}
```


5.14 告警上报 (C21)

5.14.1 布警

消息名称	CU_PU_布警
方 向	CU→ PU
功能描述	CU向PU请求布警
消息内容	具体见下表
说 明	无

布警请求消息定义

```
<! ELEMENT  GUARD_REQ(SessionID) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT  SessionID (#STRING)>
```

消息体示例如下:

```
{  
"SessionID" : "0x00000003"  
}
```

返回消息:

布警请求响应消息定义

```
<! ELEMENT  GUARD_RSP(SessionID, Ret) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT  SessionID (#STRING)>
```

<!-- Ret: 返回值 -->

```
<! ELEMENT  Ret (#NUMBER)>
```

消息体示例如下:

```
{  
"Ret" : 100,  
"SessionID" : "0x00000002"  
}
```

5.14.2 撤警

消息名称	CU_PU_撤警
------	----------

方 向	CU→ PU
功能描述	CU向PU请求撤警
消息内容	具体见下表
说 明	无

撤警请求消息定义

```
<! ELEMENT  UNGUARD_REQ(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>
```

消息体示例如下:

```
{
  "SessionID": "1",
}
```

返回消息:

撤警请求响应消息定义

```
<! ELEMENT  UNGUARD_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT  SessionID (#STRING)>
```

<!-- Ret: 返回值 -->

```
<! ELEMENT  Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "SessionID": "1",
  "Ret": 100
}
```

5.14.3 告警上报

消息名称	CU_PU_告警上报
方 向	PU→ CU
功能描述	PU向CU上报告警信息，当PU端有告警产生时主动上报告警信息给CU
消息内容	具体见下表
说 明	无

告警上报消息定义

```
<! ELEMENT ALARM_REQ(SessionID, Name, AlarmInfo) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
<!-- Name: SessionID 报警名称-->
<!-- AlarmInfo: 告警信息, 见《JsonConfigFormat》 -->
<! ELEMENT AlarmInfo(#STRING)>
```

消息体示例如下:

```
{
  "AlarmInfo" : {
    "Channel" : 0,
    "Event" : "VideoMotion",
    "StartTime" : "2009-02-16 13:36:11",
    "Status" : "Start"
  },
  "Name" : "AlarmInfo",
  "SessionID" : "0x2"
}
```

返回消息:

告警上报消息响应定义

```
<! ELEMENT ALARM_RSP(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "SessionID": "1",
  "Ret": 100
}
```

5.14.4 网络告警输入

消息名称	CU_PU_网络告警输入
方 向	CU—> PU
功能描述	CU向PU发送网络告警信息

消息内容 具体见下表
说明 无

网络告警输入消息定义

```
<! ELEMENT NET_ALARM_REQ(SessionID, Name, NetAlarmInfo) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT SessionID (#STRING)>  
<!-- Name: 报警操作-->  
<!-- NetAlarmInfo: 告警信息, 见《JsonConfigFormat》-->  
<! ELEMENT NetAlarmInfo (#STRING)>
```

消息体示例如下:

```
{  
  "Name" : "OPNetAlarm",  
  "OPNetAlarm" : {  
    "Event" : 0,  
    "State" : 1  
  },  
  "SessionID" : "0x2"  
}
```

返回消息:

网络告警输入响应消息定义

```
<! ELEMENT NET_ALARM_RSP(SessionID, Name, Ret) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT SessionID (#STRING)>  
<!-- Name: 报警操作-->  
<!-- Ret: 返回值 -->  
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{  
  "Name" : "OPNetAlarm",  
  "Ret" : 100,  
  "SessionID" : "0x00000002"  
}
```

4.34.5 告警上报报警中心

消息名称	告警上报报警中心
方 向	CU—> 报警中心
功能描述	CU向报警中心发送告警或者日志信息信息
消息内容	具体见下表
说 明	无

网络告警输入消息定义

```
<! ELEMENT ALARMCENTER_MSG_REQ (NetAlarmCenterMsg) >
```

```
<!-- NetAlarmCenterMsg: 告警信息, 见《JsonConfigFormat》 -->
```

```
<! ELEMENT NetAlarmCenterMsg(#STRING)>
```

消息体示例如下:

```
{
  "Address" : "0x0B0A060A",
  "Channel" : 0,
  "Descrip" : "",
  "Event" : "MotionDetect",
  "SerialID" : "003344236523",
  "StartTime" : "2010-06-24 17:04:22",
  "Status" : "Stop",
  "Type" : "Alarm"
}
```

```
{ "Alarm.AlarmOut" : [ { "AlarmOutStatus" : "OPEN",
  "AlarmOutType" : "MANUAL" }, { "AlarmOutStatus" : "OPEN",
  "AlarmOutType" : "AUTO" }, ...], "Name" : "Alarm.AlarmOut",
  "SessionID" : "0x3" }
```

```
.{ "Name" : "", "Ret" : 100, "SessionID" : "0x00000003" }
```

5.15 系统升级 (C22)

5.15.1 升级控制

消息名称	CU_PU_进行升级控制
方 向	CU—> PU

功能描述	CU向PU请求系统升级操作
消息内容	具体见下表
说明	无

系统升级请求消息定义

```
<! ELEMENT UPGRADE_REQ(SessionID, Name, Operation) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Operation: 见《JsonConfigFormat》 -->
<! ELEMENT Operation(#STRUCT)>
```

消息体示例如下:

```
{
  "Name" : "OPSystemUpgrade",
  "OPSystemUpgrade" : {
    "Action" : "Start",
    "Type" : "System"
  },
  "SessionID" : "0x5"
}
```

返回消息:

系统升级请求响应消息定义

```
<! ELEMENT UPGRADE_RSP(SessionID, Name, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "Name" : " OPSystemUpgrade ",
  "Ret" : 100,
  "SessionID" : "0x00000005"
}
```

5.15.2 升级数据发送

消息名称	CU_PU_发送升级数据
方 向	CU→ PU
功能描述	CU向PU发送升级数据
消息内容	具体见下表
说 明	无

发送升级数据请求消息定义

```
<! ELEMENT UPGRADE_DATA(DATA) >  
<!-- DATA: 数据，直接为2进制的裸数据-->
```

消息体示例如下：

返回消息：

系统升级请求响应消息定义

```
<! ELEMENT UPGRADE_DATA_RSP(SessionID, Ret) >  
<!-- SessionID: 会话ID-->  
<! ELEMENT SessionID (#STRING)>  
  
<!-- Ret: 返回值 -->  
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下：

5.15.3 升级进度通知

消息名称	PU_CU_发送升级进度通知
方 向	PU→ CU
功能描述	CU向PU发送升级进度通知
消息内容	具体见下表
说 明	该消息无返回消息

发送升级进度通知请求消息定义

```
<! ELEMENT UPGRADE_PROGRESS(SessionID, Ret) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>
```

消息体示例如下:

```
{
  "Name" : "",
  "Ret" : 1,
  "SessionID" : "0x00000003"
}
```

5.16 设备自动搜索 (C22)

5.16.1 IP自动搜索

消息名称	CU_PU_进行IP自动搜索
方 向	CU→ PU
功能描述	CU广播发送IP自动搜索请求
消息内容	具体见下表
说 明	无

IP自动搜索请求消息定义

```
<! ELEMENT IPSEARCH_REQ(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
```

消息体示例如下:

返回消息:

5.16.2 IP自动搜索响应消息定义


```

<! ELEMENT IPSEARCH_RSP(SessionID, Ret, NetCommon) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- Ret: 返回值 -->
<! ELEMENT Ret (#NUMBER)>

<!-- NetCommon: 见《JsonConfigFormat》 -->
<! ELEMENT NetCommon (#STRUCT)>

```

消息体示例如下:

```

{
  "NetWork.NetCommon" : {
    "GateWay" : "0x0100060A",
    "HostIP" : "0x7200060A",
    "HostName" : "LocalHost",
    "HttpPort" : 80,
    "MaxBps" : 0,
    "MonMode" : "TCP",
    "SSLPort" : 8443,
    "Submask" : "0x0000FFFF",
    "TCPMaxConn" : 10,
    "TCPPort" : 34567,
    "TransferPlan" : "AutoAdapt",
    "UDPPort" : 34568,
    "UseHSDownload" : false
  },
  "Ret" : 100,
  "SessionID" : "0x00000000"
}

```

5.17 系统信息导入导出 (C24)

5.17.1 配置导出

消息名称	CU_PU_配置导出
方向	CU→PU

功能描述	CU获取PU的所有配置信息文件
消息内容	具体见下表
说 明	无

配置导出请求消息定义

```
<! ELEMENT CONFIG_EXPORT_REQ(SessionID) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>
```

消息体示例如下：

返回消息：

配置导出响应消息定义

```
<! ELEMENT CONFIG_EXPORT_RSP() >
数据区为zip格式的配置文件数据流（大小在64K之内）
```

5.17.2 配置导入

5.17.3 日志导出

5.18 网络键盘（C25）

消息名称	CU_PU_网络键盘控制命令
方 向	CU→ PU
功能描述	CU向PU发送网络键盘控制命令
消息内容	具体见下表
说 明	无

网络键盘控制命令定义

```
<! ELEMENT NET_KEYBOARD_REQ (SessionID, Name, OPNetKeyboard) >
<!-- SessionID: 会话ID-->
<! ELEMENT SessionID (#STRING)>

<!-- OPNetKeyboard: 按键信息, 见《JsonConfigFormat》 -->
<! ELEMENT OPNetKeyboard (#STRING)>
```

消息体示例如下：

```
{
  "Name" : "OPNetKeyboard",
  "OPNetKeyboard" : {
    "Status" : "KeyUp",
    "Value" : "1"
  },
  "SessionID" : "0x3"
}
```

返回消息：

网络键盘控制命令响应

<! ELEMENT NET_KEYBOARD_RSP (SessionID, Name, Ret) >

消息体示例如下：

```
{
  "Name" : "OPNetKeyboard",
  "Ret" : 100,
  "SessionID" : "0x00000003"
}
```

6 参考资料清单

6.1 返回码定义

返回码	表述
100	OK
101	未知错误
102	版本不支持
103	非法请求
104	该用户已经登录
105	该用户未登录
106	用户名密码错误
107	无权限
108	超时
109	查找失败，没有找到对应文件
110	查找成功，返回全部文件
111	查找成功，返回部分文件
112	该用户已经存在
113	该用户不存在
114	该用户组已经存在
115	该用户组不存在
116	

117	消息格式错误
118	未设置云台协议
119	没有查询到文件
120	配置为启用
121	MEDIA_CHN_NOTCONNECT 数字通道未连接
150	成功, 设备需要重启
202	用户未登录
203	密码不正确
204	用户非法
205	用户被锁定
206	用户在黑名单中
207	用户名已经登录
208	输入不合法
209	索引重复如要增加的用户已经存在等
210	不存在对象, 用于查询时
211	对象不存在
212	账号正在使用中
213	子集超范围(如组的权限超过权限表, 用户权限超出组的权限范围等等)
214	密码不合法
215	密码不匹配
216	保留帐号
502	命令不合法
503	对讲已经开启
504	对讲未开启
511	已经开始升级
512	未开始升级
513	升级数据错误
514	升级失败
515	升级成功
521	还原默认失败
522	需要重启设备
523	默认配置非法
602	需要重启应用程序
603	需要重启系统
604	写文件出错
605	特性不支持
606	验证失败
607	配置不存在
608	配置解析出错

6.2 命令码定义

6.2.1 登录登出保活协议C1消息号

命令名称	消息号	说明
LOGIN_REQ	999/1000	登录请求

LOGIN_RSP	1000	登录响应
LOGOUT_REQ	1001	登出请求
LOGOUT_RSP	1002	等出响应
FORCELOGOUT_REQ	1003	强制登出请求
FORCELOGOUT_RSP	1004	强制登出响应
KEEPALIVE_REQ	1005	保活请求
KEEPALIVE_RSP	1006	保活响应

6.2.2 获取设备信息C2消息号

命令名称	消息号	说明
SYSINFO_REQ	1020	获取系统信息请求
SYSINFO_RSP	1021	获取系统信息请求响应

6.2.3 设置/获取配置相关信息请求C3~C11消息号

命令名称	消息号	说明
CONFIG_SET	1040	设置配置请求
CONFIG_SET_RSP	1041	设置配置响应
CONFIG_GET	1042	获取配置请求
CONFIG_GET_RSP	1043	获取配置响应
DEFAULT_CONFIG_GET	1044	获取默认配置请求
DEFAULT_CONFIG_GET_RSP	1045	获取默认配置相应
CONFIG_CHANNELTILE_SET	1046	设置通道名称
CONFIG_CHANNELTILE_SET_RSP	1047	设置通道名称响应
CONFIG_CHANNELTILE_GET	1048	获取通道名称请求
CONFIG_CHANNELTILE_GET_RSP	1049	获取通道名称响应
CONFIG_CHANNELTILE_DOT_SET	1050	设置通道名称点阵请求
CONFIG_CHANNELTILE_DOT_SET_RSP	1051	设置通道名称点阵响应

6.2.4 系统调试消息号

命令名称	消息号	说明
SYSTEM_DEBUG_REQ	1052	系统调试请求
SYSTEM_DEBUG_RSP	1053	系统调试响应

6.2.5 获取能力级请求C12消息号

命令名称	消息号	说明
ABILITY_GET	1360	能力级获取请求

ABILITY_GET_RSP	1361	能力级获取响应

6.2.6 云台控制C13消息号

命令名称	消息号	说明
PTZ_REQ	1400	云台控制请求
PTZ_RSP	1401	云台控制响应

6.2.7 监视控制C14消息号

命令名称	消息号	说明
MONITOR_REQ	1410	实时监视请求
MONITOR_RSP	1411	实时监视请求响应
MONITOR_DATA	1412	实时监视数据
MONITOR_CLAIM	1413	监视认领请求
MONITOR_CLAIM_RSP	1414	监视认领请求响应

6.2.8 回放控制C15消息号

命令名称	消息号	说明
PLAY_REQ	1420	回放请求
PLAY_RSP	1421	回放请求响应
PLAY_DATA	1422	回放数据
PLAY_EOF	1423	回放文件结束
PLAY_CLAIM	1424	回放认领请求
PLAY_CLAIM_RSP	1425	回放认领请求响应
DOWNLOAD_DATA	1426	下载数据

6.2.9 语音对讲C16消息号

命令名称	消息号	说明
TALK_REQ	1430	对讲请求
TALK_RSP	1431	对讲请求响应
TALK_CU_PU_DATA	1432	CU到PU的对讲数据
TALK_PU_CU_DATA	1433	PU到CU的对讲数据
TALK_CLAIM	1434	对讲认领请求
TALK_CLAIM_RSP	1435	对讲认领请求响应

6.2.10 文件查询C17消息号

命令名称	消息号	说明
FILESEARCH_REQ	1440	录像文件查询
FILESEARCH_RSP	1441	录像文件查询响应
LOGSEARCH_REQ	1442	日志查询

LOGSEARCH_RSP	1443	日志查询响应
FILESEARCH_BYTIME_REQ	1444	通过时间搜索文件请求
FILESEARCH_BYTIME_RSP	1445	通过时间搜索文件响应

6.2.11 系统管理C18消息号

命令名称	消息号	说明
SYSMANAGER_REQ	1450	系统管理请求
SYSMANAGER_RSP	1451	系统管理请求响应
TIMEQUERY_REQ	1452	获取PU系统时间
TIMEQUERY_RSP	1453	获取PU系统时间响应

6.2.12 硬盘管理C19消息号

命令名称	消息号	说明
DSIKMANAGER_REQ	1460	磁盘管理请求
DSIKMANAGER_RSP	1461	磁盘管理请求响应

6.2.13 用户管理C20消息号

命令名称	消息号	说明
FULLAUTHORITYLIST_GET	1470	获取所有权限的列表
FULLAUTHORITYLIST_GET_RSP	1471	获取所有权限的列表响应
USERS_GET	1472	获取所有用户的权限列表
USERS_GET_RSP	1473	获取所有用户的权限列表响应
GROUPS_GET	1474	获取所有组的权限列表
GROUPS_GET_RSP	1475	获取所有组的权限列表响应
ADDGROUP_REQ	1476	添加组请求
ADDGROUP_RSP	1477	添加组请求响应
MODIFYGROUP_REQ	1478	修改组请求
MODIFYGROUP_RSP	1479	修改组请求响应
DELETEGROUP_REQ	1480	删除组请求
DELETEGROUP_RSP	1481	删除组请求响应
ADDUSER_REQ	1482	添加用户请求
ADDUSER_RSP	1483	添加用户请求响应
MODIFYUSER_REQ	1484	修改用户请求
MODIFYUSER_RSP	1485	修改用户请求响应
DELETEUSER_REQ	1486	删除用户请求
DELETEUSER_RSP	1487	删除用户请求响应
MODIFYPASSWORD_REQ	1488	修改用户密码
MODIFYPASSWORD_RSP	1489	修改用户密码响应

6.2.14 告警上报C21消息号

命令名称	消息号	说明
------	-----	----

GUARD_REQ	1500	布警请求
GUARD_RSP	1501	布警请求响应
UNGUARD_REQ	1502	撤警请求
UNGUARD_RSP	1503	撤警请求响应
ALARM_REQ	1504	告警请求（唯一一个PU主动上报给CU的消息）
ALARM_RSP	1505	告警请求响应
NET_ALARM_REQ	1506	网络告警
NET_ALARM_RSP	1507	网络告警响应
ALARMCENTER_MSG_REQ	1508	告警上报告警中心

6.2.15 系统升级C22消息号

命令名称	消息号	说明
UPGRADE_REQ	1520	升级请求
UPGRADE_RSP	1521	升级请求响应
UPGRADE_DATA	1522	升级数据包
UPGRADE_DATA_RSP	1523	升级数据包响应
UPGRADE_PROGRESS	1524	升级进度
UPGRADE_INFO_REQ	1525	升级信息请求
UPGRADE_INFO_RSP	1526	升级信息响应

6.2.16 设备自动搜索C23消息号

命令名称	消息号	说明
IPSEARCH_REQ	1530	IP自动搜索请求
IPSEARCH_RSP	1531	IP自动搜索请求响应
IP_SET_REQ	1532	IP设置请求
IP_SET_RSP	1533	IP设置响应

6.2.17 系统信息导入导出C24消息号

命令名称	消息号	说明
CONFIG_IMPORT_REQ	1540	配置导入请求
CONFIG_IMPORT_RSP	1541	配置导入请求响应
CONFIG_EXPORT_REQ	1542	配置导出请求
CONFIG_EXPORT_RSP	1543	配置导出请求响应
LOG_EXPORT_REQ	1544	日志导出请求
LOG_EXPORT_RSP	1545	日志导出请求响应

6.2.18 网络控制键盘C25消息号

命令名称	消息号	说明
NET_KEYBOARD_REQ	1550	网络键盘控制命令
NET_KEYBOARD_RSP	1551	网络键盘控制命令响应

6.2.19 网络抓拍C26消息号

命令名称	消息号	说明
NET_SNAP_REQ	1560	网络抓拍请求
NET_SNAP_RSP	1561	网路抓拍响应
SET_IFRAME_REQ	1562	设置框架请求
SET_IFRAME_RSP	1563	设置框架响应

6.2.20 透明串口C27消息号

命令名称	消息号	说明
RS232_READ_REQ	1570	RS232读请求
RS232_READ_RSP	1571	RS232读响应
RS232_WRITE_REQ	1572	RS232写请求
RS232_WRITE_RSP	1573	RS232写响应
RS485_READ_REQ	1574	RS485读请求
RS485_READ_RSP	1575	RS485读响应
RS485_WRITE_REQ	1576	RS485写请求
RS485_WRITE_RSP	1577	RS485写响应
TRANSPARENT_COMM_REQ	1578	透明串口请求
TRANSPARENT_COMM_RSP	1579	透明串口响应
RS485_TRANSPARENT_DATA_REQ	1580	RS485透明数据请求
RS485_TRANSPARENT_DATA_RSP	1581	RS485透明数据响应
RS232_TRANSPARENT_DATA_REQ	1582	RS232透明数据请求
RS232_TRANSPARENT_DATA_RSP	1583	RS232透明数据响应

6.2.21 网络登录时间同步

命令名称	消息号	说明
SYNC_TIME_REQ	1590	网络登录时间同步请求
SYNC_TIME_RSP	1591	网络登录时间同步响应

6.2.22 抓图上传消息号

命令名称	消息号	说明
PHOTO_GET_REQ	1600	抓图上传请求
PHOTO_GET_RSP	1601	抓图上传响应