

公司内部		
第	1.0	版
共	13	页

网卡状态实时采集和报警程序概要设计

文档编号： LINX_<NSQ>_<TS>_<HLD>_1.0

文档名称： 网卡状态实时采集和报警程序概要设计

文档类别： <概要设计>

密　　级： 公司内部

版本信息： 1.0

建立日期： 2009-04-15

创 建 人： 北京凝思科技有限公司研发部

审 核 者： 陈斌

批 准 人： 彭志航

批准日期： 2009-04-16

保 管 人： 武萌

存放位置： svn://……

编辑软件： OpenOffice 2.2

文档修订记录

VER	REV	修订人	修订日期	简要说明	批准人	批准日期
V1.0	<C>	<胡剑>	2009-04-16	<初次创建>	彭志航	2009-04-16
VER（版本编号）：V——版本编号；R——修订编号。						
REV（修订状态）：C——创建；A——增加；M——修改；D——删除。						

文档审批记录

版本号	审批人	角色	审批日期	签字	备注
V1.0	彭志航	部门经理	2009-04-16		

目　录

1.软件概述.....1
2.配置文件选项.....2
3.软件接口.....3
4.相关数据结构.....4
5.程序流程.....7
6.日志.....8

插图

插图　1：程序流程图.....7

1.软件概述

网卡状态实时采集和报警程序运行在D-5000的每个节点，该程序为D-5000平台的底层程序，独立于操作系统，用户名为d5000。网卡状态采集和报警程序定时向内核驱动请求网卡状态和信息，由网卡驱动接口提取网卡寄存器值，然后把相关信息和统计的状态返回到用户空间，将本节点各个网卡（包括虚拟网卡）的状态，写入本地的网卡状态共享内存，应包含时标信息；应提供变化数据写入和周期全数据写入两种方式，周期全数据写入时间可调，暂定为5分钟，如检测到网卡状态变化应实时写入。

网卡状态实时采集和报警程序当发现网卡故障、流量异常和网卡切换时，实时发送UDP报警事件报文，每个事件连续发送3次。

2.配置文件选项

程序运行采用配置文件，配置文件路径为：

/home/d5000/project/conf/sys_netcard_conf.txt。

配置文件内容如下：

```
domain              10                                          //程序运行域
serv                01                                        //网卡报警服务标识
event               2                                       //网卡报警事件 ID 标识
udpport             15000                                     //网卡报警广播地址
sys_netcard_shm_path    /home/d5000/project/data/shm_sys_netcard_info
                                                              //共享内存路径
sys_netcard_sem_path    /home/d5000/project/data/sem_sys_netcard_info
                                                            //共享内存信号量路径
monitor_interval  100                       //监视网卡的时间间隔，单位为毫秒
write_interval    300                 //周期写入共享内存的时间间隔，单位为秒
flow_ interval    60                          //获取流量时间间隔，单位：秒
flow_peak         100   //单块网卡的每秒最大流量，单位：兆字节每秒(Mb/s)
flow_limit        30                                        //设定警报峰值的百分比
udp               eth0          //发送广播包的网卡，通过此网卡获取广播地址
nic               eth1                                      //需要检测的网卡名称
nic               eth2                          //如要监视多个网卡添加规则如此
nic               eth3
```

3.软件接口

该软件通过两种方式向用户提供信息，一种是采用 UDP 报文方式，若网卡出现故障，或者网卡发生切换，或者流量异常时采用此方式向用户发送异常警报信息；另一种是共享内存方式，若出现异常警报，或者完成一次指定的时间周期时，程序会将网卡状态信息写入共享内存。两种方式的具体实施方法如下：

1. UDP 报文方式

异常警报信息以及由故障转为正常的信息均采用 UDP 报文发送，提供相应的接收程序。

2. 共享内存方式

网卡状态等信息实行共享内存写入和获取，服务端提供 API 函数，并提供动态库。

函数为：int init_nic_info(char *shared, char *sem_path);

```
int get_nic_info(char *nic_name, NET_INFO *net_info);

void rls_nic_info(void);
```

动态库为：libnic_shm.so。

使用方法请参阅使用说明文档。

4.相关数据结构

(1). 写入共享内存的包含网卡状态信息的数据类型

```
typedef   struct  _net_info
{
      KEYID ID;                                   /*ID  国调保留 long */
      char  charname[NIC_NAME_LEN];                    /*设备名称 */
      char  descr[128];                              /*设备中文描述 */
      struct sockaddr addr;                               /* IP 地址 */
      struct sockaddr broadaddr;                        /*广播地址 */
      struct sockaddr netmask;                          /*子网掩码 */
      struct sockaddr hwaddr;                        /* mac 地址 */
      int         time_stamp;       /* 时标信息 精确到秒 读系统时钟*/
      short flags;                                  /* 状态标记 */
      int         mtu;                              /* 最大传输单元*/
      int         tx_queue_len;                     /* 传输队列长度 */
      int   average_flow                          /*网卡平均流量, 单位字节每秒*/
      unsigned long long rx_packets;                /* 接收包的数量 */
      unsigned long long tx_packets;                /* 发送包的数量 */
      unsigned long long rx_bytes;                  /* 收到的总比特数 */
      unsigned long long tx_bytes;                  /* 发送的总比特数 */
      unsigned long rx_errors;                       /* 收到出错包数 */
      unsigned long tx_errors;                      /* 发送包出错数 */
      unsigned long rx_dropped;                   /* 因接收缓存不足丢包数 */
      unsigned long tx_dropped;                   /* 因发送缓存不足丢包数 */
      unsigned long rx_multicast;                        /* 组播包数 */
      unsigned long collisions;                          /* 发生碰撞次数*/
      unsigned long rx_fifo_errors;                /* 管道空间溢出次数 */
      unsigned long tx_carrier_errors;
      unsigned long tx_fifo_errors;
}NETCARD_INFO, *LPNETCARD_INFO;
```

(2). 发送UDP报文的包含警报信息的数据类型

```
typedef struct  _d5000_nic_alarm
{
      MSG_FRAME                    tMsgFrame ;
      SYS_NETCARD_ALARM            tSysNetcardAlarm ;
}D5000_NIC_ALARM, *LPD5000_NIC_ALARM;
```

(3). 网卡状态信息报文头

```
typedef struct {                                  /* message frame       */
      short   len;                                /* message length      */
      short   seqno;                              /* send sequence       */
      short   serv;                               /* services ID         */
      short   event;                              /* event ID            */
      unsigned char   domain;                     /* domain ID           */
      unsigned char   ctxt;                       /* Context ID          */
      short   stid;                               /* SOURCE task id      */
      short   dtid;                               /* DESTINATION task ID */
      unsigned char   ver_coding;                 /* 版本号 + 编码       */
      unsigned char   mes_type;                   /* 帧类型              */
   } MSG_FRAME, *LPMSG_FRAME;
```

(4). 报警信息数据结构

```
/* 状态标记 1:网卡故障 2:网卡恢复 3:网卡切换 4:流量异常。*/
#define NETCARD_ALARM_FAULT          1
#define NETCARD_ALARM_RESUME         2
#define NETCARD_ALARM_SWITCH         3
#define NETCARD_ALARM_ABNORMAL       4
typedef struct  _sys_netcard_alarm
{
      char fault_devname[NIC_NAME_LEN];       /*故障设备名称, 主机名+网
```

```
卡名 */
        char switch_devname[NIC_NAME_LEN];     /*切换设备名称 */
        short  flags;                          /* 状态标记 1:网卡故障 2:网卡恢复
                                                  3:网卡切换 4:流量异常。
*/
        short retrytimes;                                  /* 0,1,2 重发次数 */
} SYS_NETCARD_ALARM, *LPSYS_NETCARD_ALARM;
```

5.程序流程

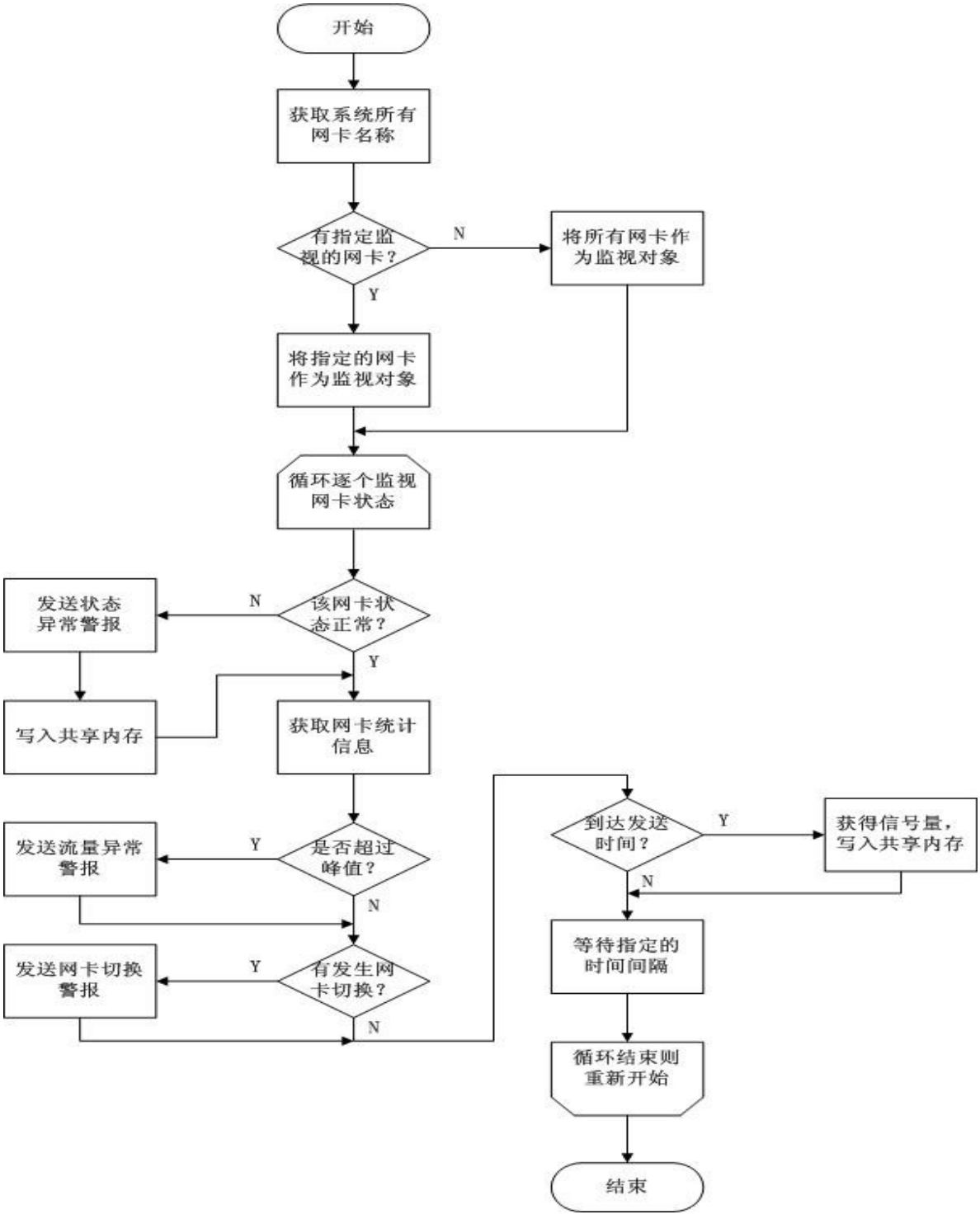


插图 1：程序流程图

流程文字说明：

1.获取系统所有网卡名称，且从配置文件中读取指定要监视的网卡名称及相关参数设置。若指定网卡名称及配置选项不符合要求程序将提示错误信息，并退出。

2.开始循环监视各个网卡状态信息。若网卡状态异常，则发送异常警报，并将异常信息写入共享内存。

3.读取并统计网卡状态信息，并计算规定时间间隔内的平均流量所占带宽的百分比，若超过峰值则发送异常警报，并将异常信息写入共享内存。

4.实时检测网卡是否发生切换，若发生切换则发送异常警报。

5.检测等待时间周期是否已完成，若等待时间已到则读取所有网卡状态信息，并写入共享内存。

6.本次循环结束，并进入下次循环。

6.日志

1 内容及格式

应客户要求，日志报文采用标准 SysLog 日志格式，主要结构包括固定的头部字段和内容字段。头部字段包括三个域：分别是时间标签、生成消息的模块名称（可能包含态名）。时间标签主要是消息生成的日期和时间，如 Sep 12 10:23:44.001，精确到毫秒。内容字段包括日志优先级和具体的日志内容。

下面是系统的一条日志范例，每条日志分为 4 部分内容，分别表示为时间、应用或进程名、日志内容三部分。

例如：

```
2009-4-19 10:37:13.001  fe.telcom  ALERT :process restart
2009-4-19 10:37:13.001  realtime scada.analog_proc  ALERT :process restart
```

2 优先级

日志的优先级参考标准 Linux 日志优先级定义，根据实际情况，目前定义为 5 种优先

级别，0 号优先级最高，4 号优先级最低。

0——LOG_EMERG 用于紧急事件消息，比如应用模块崩溃之前的提示消息或者是应用发生切换。

1——LOG_ALERT 用于报告出现的警告问题。

2——LOG_NOTICE 用于报告系统日常信息，比如运行模块的启停、用户会话信息等。

3——LOG_INFO 用于系统资源的提示信息，比如系统 CPU 利用率、内存利用率等。

4——LOG_DEBUG 用于装置应用模块的调试信息，通常在调试一个模块时使用

3 日志记录目录和名称

日志记录统一存放在 D-5000 用户的~/var/log/netcard 目录下，日志文件通常为生成日期+_应用/进程名.log ，如 20090419_应用/进程名.log ，权限为 0644。