一、TR069 LAN节点故障

EC 613000733273 svn://10.67.9.15/CSP_201_L00/branches/CSP_OV_WII_30P2 r3256

【问题描述】

在TR069 网管上disable LAN端口后，PC还是可以连接Modem以及正常登陆，页面显示该LAN口disable。

【问题分析】

经检查，发现enable/disable lan接口都是使用swethcore的字符设备ioctl函数实现的。但是我们内核态并不支持这两个操作。用户态代码调用这个命令的时候，我们也没有给任何的提示信息。改动见下面的附件。

【问题总结】

1. 防御式编程要告警

在这个函数中，我们检查Func_id，如果未实现，那么返回错误值。可以看作防御式编程的一种。即：检查收到的参数，处理未定义或不可预期的错误。

```
if ( IoctlArg.Func_id > 1)
    return -EINVAL;
```

但是在当前上下文中，这段代码有一个问题：防御了，却没有有效处理防御结果。发现了错误，就要及时处理，而不是返回一个很模糊的-1。返回给用户态后，用户态也没有处理这个-1。这导致问题暴露后，很难定位。

常见的出错处理有三种：

Assert，直接退出进程，或重启系统

Warning，在串口打印警告

Status value，通过返回一个有效的错误码，告知错误。

这里应该使用warning的方式。

2. Lock和unlock的粒度

为了避免写冲突，在进入ioctl的时候加了lock，在退出ioctl时unlock。

但是注意这段代码，if(dev == NULL)的时候，却没有加unlock，没有配对。曾经在831II项目中，就出现过一个这样的故障，写了lock，但是在if语句中没有unlock提前返回。

```
    rtnl_lock();
    struct net_device *dev = dev_get_by_name( IoctlArg.Dev_name );
    if (dev == NULL )
{
        。。。
        return -EINVAL;
}
```

有一些方法可以避免这种问题的出现：

1）配对编程，总是先写lock和unlock，然后再添加代码，避免遗忘。

2）减小lock的粒度。譬如这里，是不需要对所有ioctl都加lock的，只需要对一些临界代码加lock即可。也可以避免写错代码。

3. 使用宏，而不是直接使用0、1、2、3

在错误代码中，全部使用0、1、2作为case的判断条件。

为什么不能这样？代码的可读性差，维护代码的时候不知道0、1代表什么意思。另外，由于这里的0、1在用户态有对应的宏，导致不能使用source insight快速定位代码。

二、web页面不能访问

EC 613000314590 svn://10.67.9.15/CSP_201_L00/branches/CSP_OV_WII_30P2 r1624

【问题描述】：

端口限速5M，过载发包8M及以上时，web页面无法访问

【问题分析】：

带流量时网页无法访问，在本地复现故障时发现，串口有大量打印信息。对照代码，发现在提版本前一天入库的代码中，驱动发包函数中有一个打印语句没有去除，导致带流量情况会有大量打印信息，从而严重影响Modem性能。

【问题总结】：

应该是一个很低级的问题，但是导致系统测试与性能相关的测试全部失败，影响很坏。

【后续改进】

svn提交时，首先使用svn diff确认修改是否完整，无关代码都要移除。

同时要充分自测，考虑到修改可能的影响。

三、 url过滤导致modem死机

EC 613000314557 svn://10.67.9.15/CSP_201_L00/branches/CSP_OV_WII_30P2 r1632

EC 613000402765 svn://10.67.9.15/CSP_201_L00/branches/CSP_OV_WII_30P2 r1824

【问题描述】：

1. 在设定未允许的状态下，添加中兴公司主页，无法登陆，http://www.zte.com.cn/cn/，但如果去掉“/cn/”则可以登陆

2. 配置好URL过滤后,然后将网卡禁用再启用，modem自动重启，基本上必现。若网卡一直为启用状态，也有机率发生modem自动重启。

【问题分析】：

这里有两个故障，第一个是功能性的，由于有的HTTP头部的GET字段不包含host地址，导致匹配失败。因此修改了代码，首先检查HTTP头部是否有HOST字段，如果有HOST字段，则将之放在一个全局的数组中。然后再去检查GET字段，并将GET字段和HOST字段拼装为一个url进行匹配。

这样修改后，第一个故障解决了。且自测通过。测试部测试也通过。但是在两个月之后的一次测试中，测试部发现，将我们的modem挂在PC下，会导致重启。经定位，是配置了url过滤触发的故障。

经检查代码发现，全局数组的长度为256字节，当时认为256字节时足够长的。自测及常规测试时，由于url的GET字段和HOST字段都比较短，因此是不会出问题的。但是PC会自动发送一些HTTP包，这些包的GET字段可能非常长，甚至有上千字节。由于urlfilter中没有考虑数组越界，且使用了memcpy拷贝内存，从而导致了死机故障。

【问题总结】

在自测及之前的系统测试中都没有暴露故障，因为一些偶然因素才触发。事后到现网下发现，有很多的网页的GET字段都是超级长的。不过在自测时只是使用故障中的网址，不够充分，出现故障。

【后续改进】

1. 写代码，及设计测试用例要考虑充分，考虑到所有的情况。

2. 数组越界问题。可以对照代码走查表，检查代码，及早发现问题。

在培训的过程中，王飞杰提到他碰到的两个问题。

四、 当多重if嵌套和内存分配释放相遇时

在进入函数时申请UB，之后进入多重的if判断，if中有可能提前return，此时忘记释放UB了。

这种情况，建议不要在多重if嵌套中return，而是使用goto，在函数的末尾处理异常，并释放UB。

如果goto不允许使用，那么也不要在if嵌套中return，而是使用一些诸如status来保存if的判断结果，然后在if判断完成后，再根据status决定是否需要释放UB。以保证成对编程。总之，代码逻辑是可以优化的。

五、 多线程同步的问题

线卡会定时30秒的扫描设备状态，然后上报。但是有的时候扫描的时间过长，有四五分钟的时间。而定时的间隔只有30秒。出现问题。

分析认为，这可能是一个多线程同步（也有可能是线程和定时器同步）的问题。理想的情况是，在定时器和线程之间使用信号量，当扫描完成后，才触发信号量开始计时。