对民用飞行控制系统固件的逆向与漏洞分析

0xbird(zhangpeng@Hillstonenet)
山石网科·神经元攻防实验室

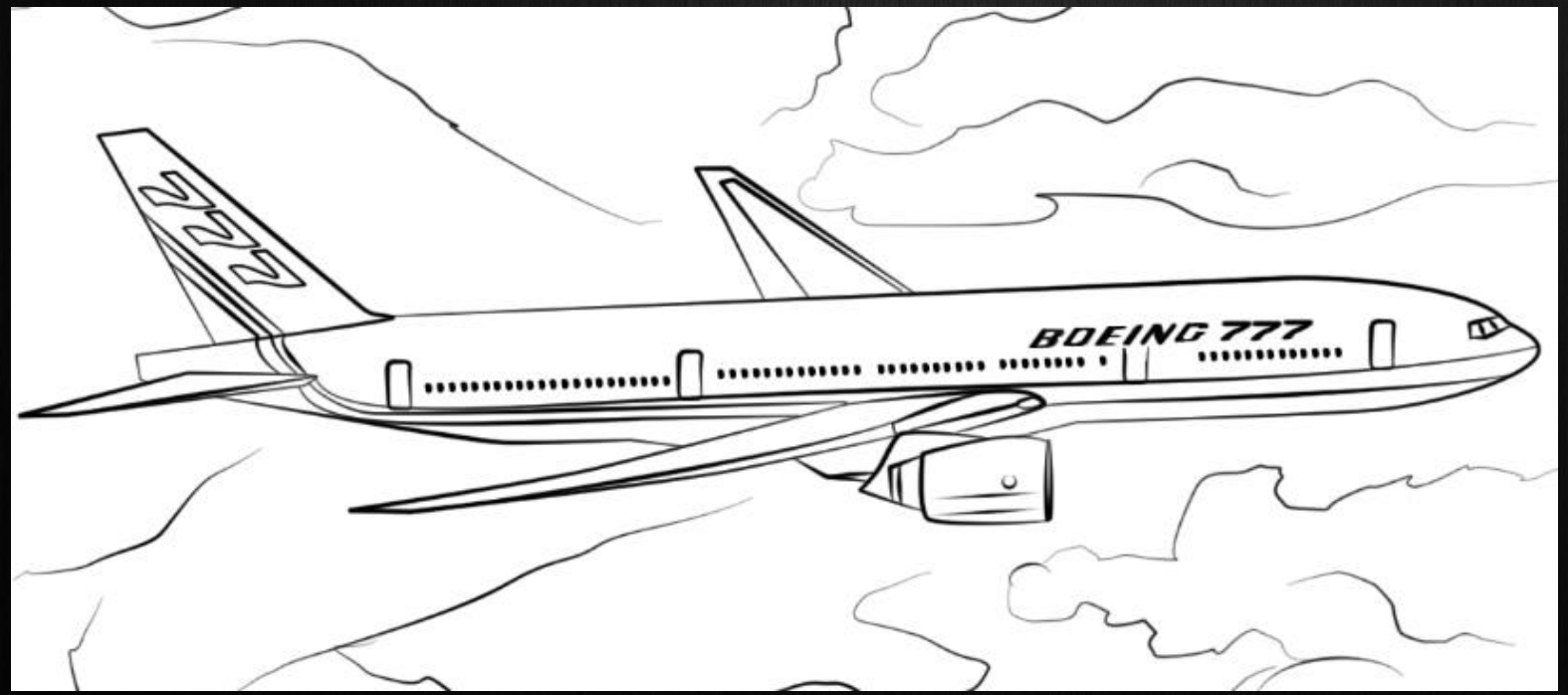
About Me

- 张鹏(0xbird)
- 安全研究员 山石网科@神经元攻防实验室
- 民用航空相关专业
- 研究领域:
  - 智能硬件/无线电/车联网 安全研究
  - PC端应用软件/浏览器 漏洞挖掘
  - CTF比赛主攻pwn&re
  - AI安全&样本对抗&GAN
  - APT组织分析&漏洞病毒样本分析预警

table of Contents

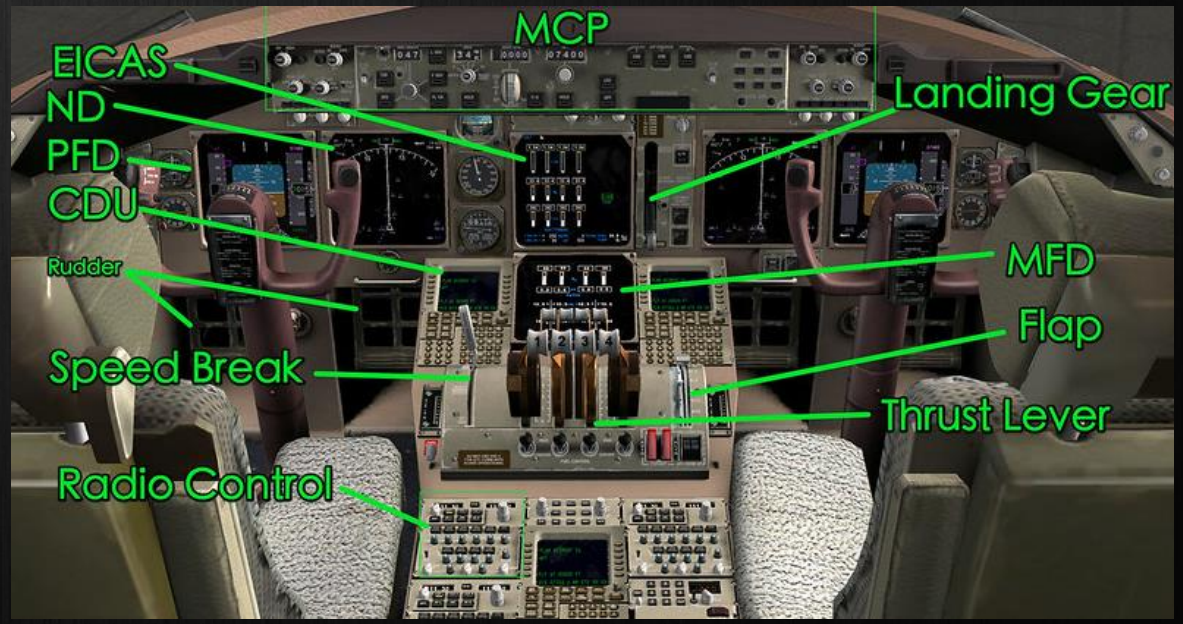
研究背景

911事件以后，各个国家对航空安全非常重视，但是由于飞行机械故障和飞行员错误操作时不时还是会发生坠机事件，在民用航空领域是轻易不出事，但是一出事就是大事，特别是坠机事件是一种国家级的安全事件。



研究背景

波音777飞机比起之前的机型，它实现了一些新的功能，这些功能组件会和常规电子航空系统集成在一起，比如机组信息系统，数据管理系统。



研究背景

这种程度的高度集中意味着航空电子设备和飞行娱乐系统可能是在同一个网络环境中运行的，完全有可能通过飞行娱乐系统中的漏洞进入飞行控制系统。

研究背景

波音服务器在公网上泄露的一些存储库文件：
- 机组信息系统的文件系统/维护系统（CIS/MS）的固件
- 网络系统（ONS）的固件

主要是波音737，波音747，波音777型号的文件

波音777飞机架构和系统介绍

1.通用核心系统（CCS）

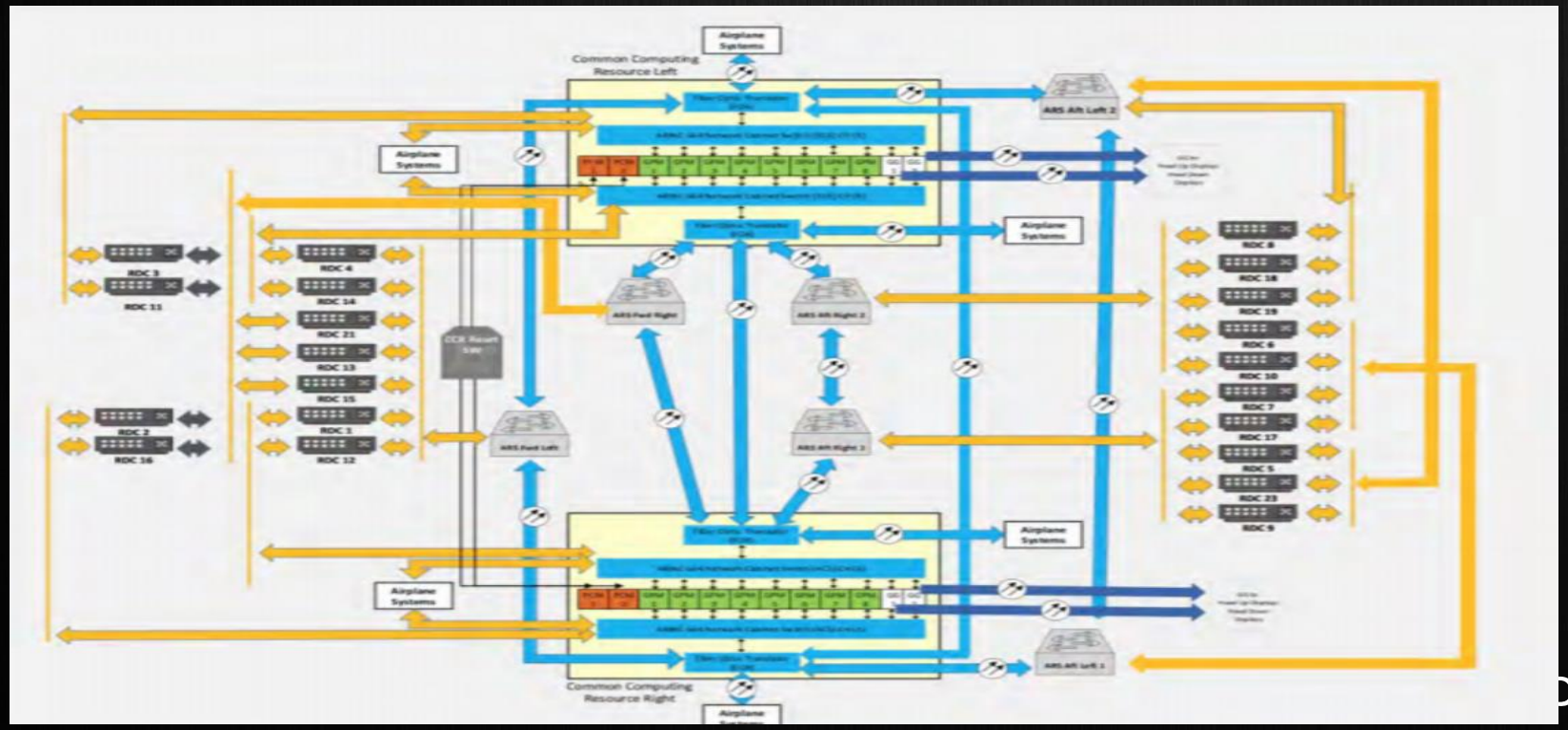
CCS中主要有以下原件：

- 通用处理模块（GPM）：支持一些功能性的处理需求
- 远程数据控制模块（RDC）：支持系统的模电信号和串行数字接口
- 航空电子全双工网络：用于支持各平台的通信

波音777飞机架构和系统介绍

这些原件会被封装成便于管理的单元，CCS就会被分成以下组件：

- 两个通用计算资源池（CCR）
- 通用数据网络（CND）
- 21个远程数据控制器（RDC）



波音777飞机架构和系统介绍

2.通用计算资源池（CCR）

每个通用计算资源池包括以下组件：

- 两个功率调节模块（PCM）
- 八个通用处理模块（GPM）
- 两个ARINC 664-P7交换机（ACS）
- 两个光纤转换器模块（FOX）

波音777飞机架构和系统介绍

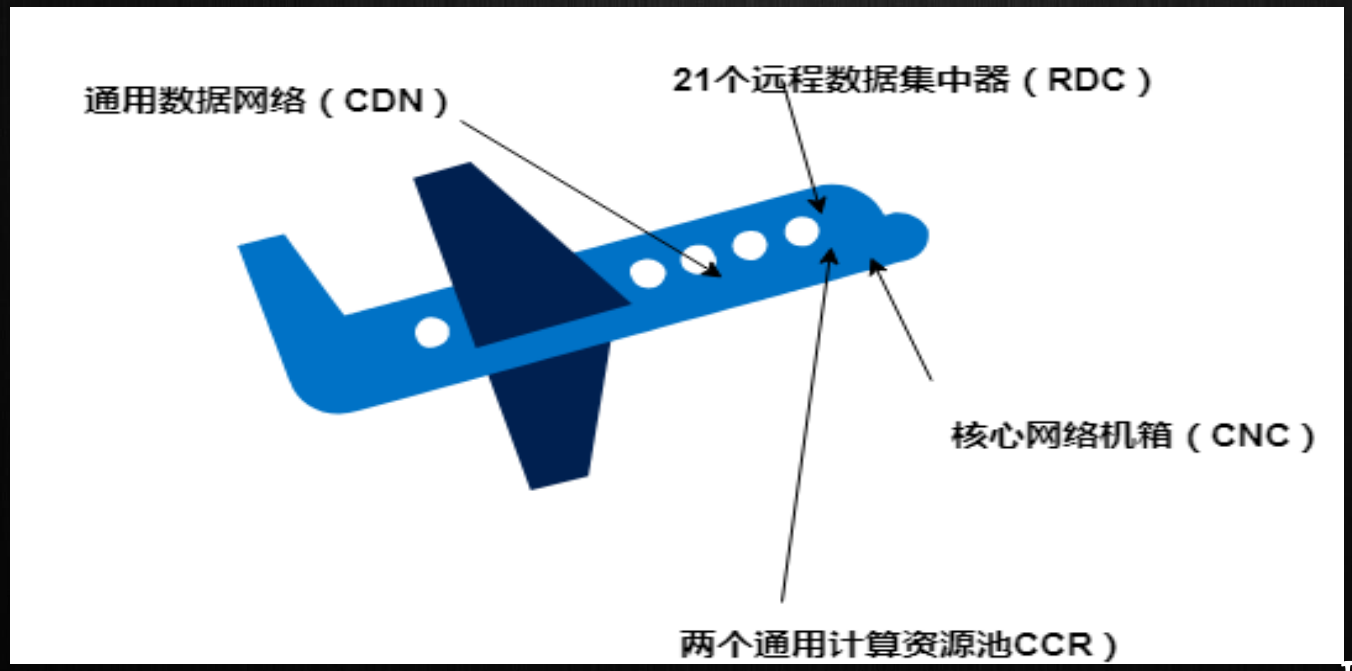
这些计算机资源池（CCR）中的通用处理模块（GPM）可以托管下面的功能系统：

机舱温度控制系统
远程配电系统（RPDS）
电子冷却系统
发动机防火系统
发动机防冰指示系统
起落架指示和控制系统
推力管理功能
飞机调节监控功能
设备冷却系统
发电机/总线功率控制单元
通信管理功能
液压系统控制
机舱警报系统
照明系统
飞行管理功能
舱门控制系统

波音777飞机架构和系统介绍

3.通用数据网络（CND）

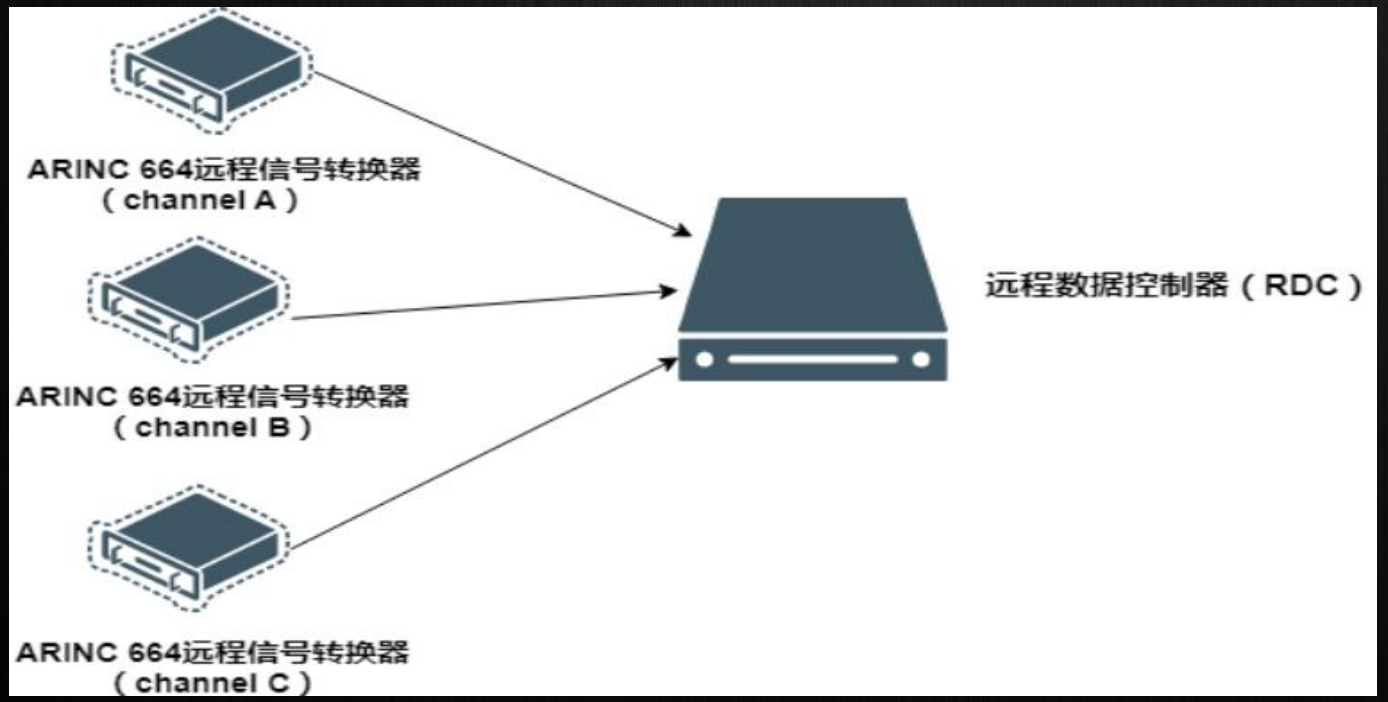
CND是一种数字数据网络系统，主要使用光缆和铜缆连接飞机上的各个网络设备。



波音777飞机架构和系统介绍

4.远程数据控制器（RDC）

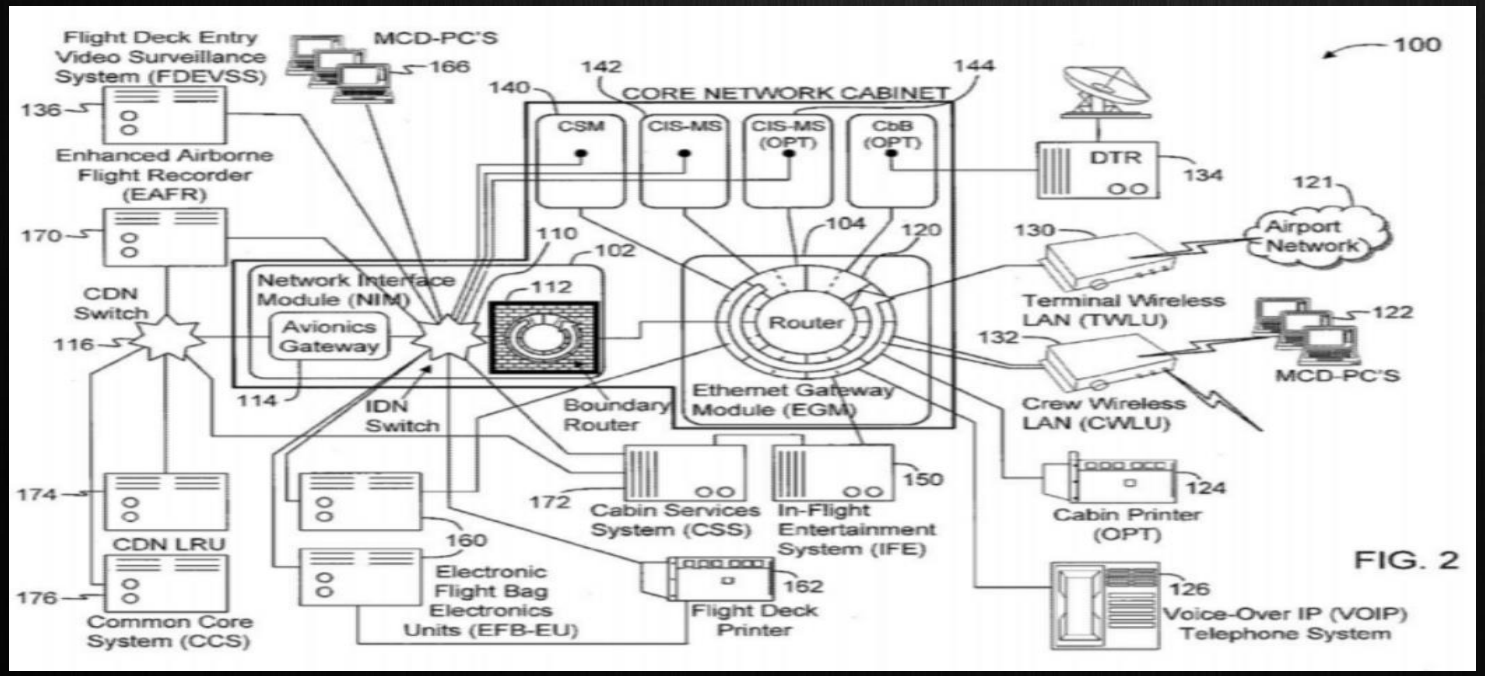
通用核心系统（CCS）有21个远程数据控制器（RDC）



波音777飞机架构和系统介绍

5.机组信息系统/维护系统（CIS/MS）

机组信息系统/维护系统（CIS/MS）提供了给飞行机组人员，维修工程师，航空公司工程师访问飞行操作系统和查看数据的功能接口。



波音777飞机架构和系统介绍

机组信息系统/维护系统（CIS/MS）的主要组成部分是核心网络机箱（CNC），包括以下模块：

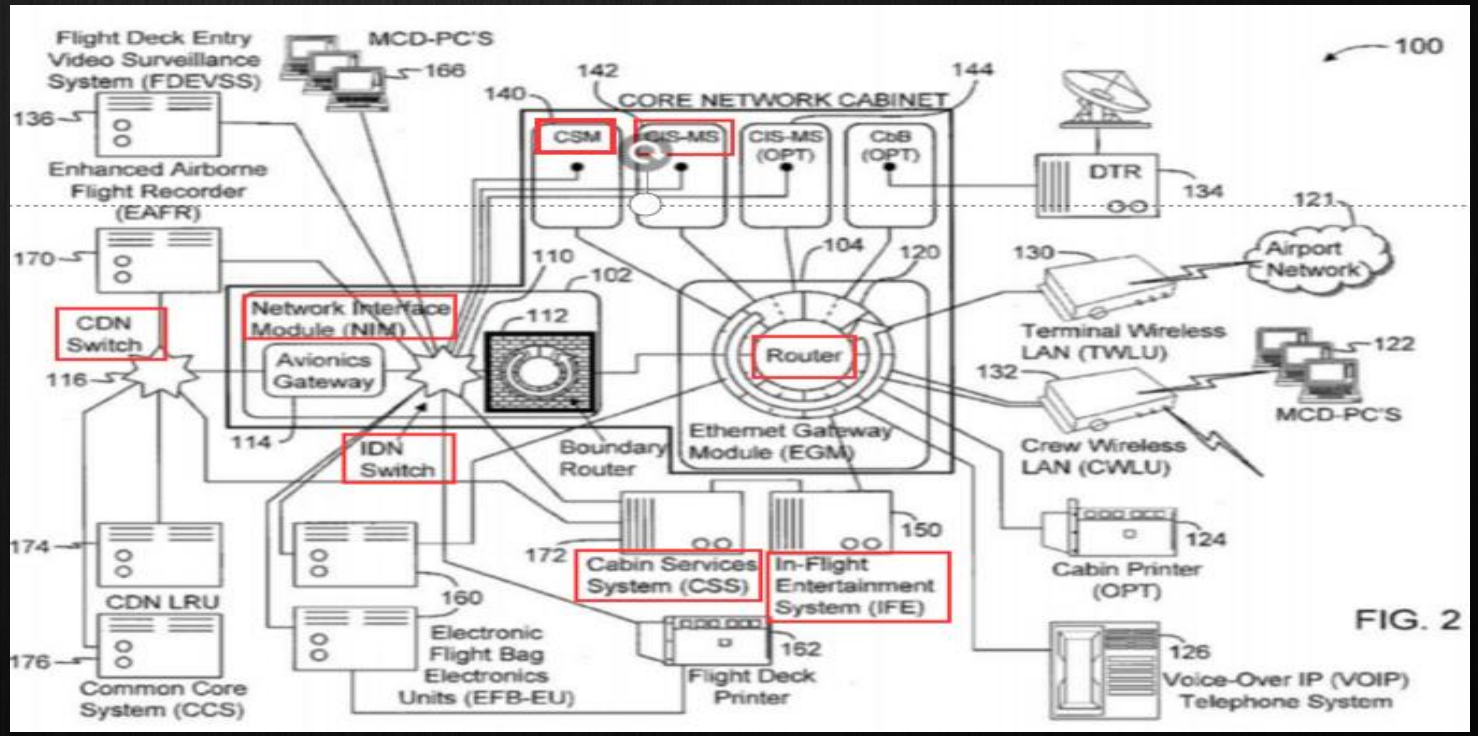
- 网络接口模块（NIM）
- 网关模块（EGM）
- 控制服务器模块（CSM）
- 机组信息系统/维护系统文件服务器模块（CIS/MS）

波音777飞机架构和系统介绍

下图描述了机组信息系统/维护系统（CIS/MS）的体系结构

主要分为三个网络：

1.开放数据网络（ODN）
2.隔离数据网络（IDN）
3.通用数据网络（CDN）



波音777飞机架构和系统介绍

6.飞行娱乐系统（IFE）

飞行娱乐系统（IFE）的架构如下：

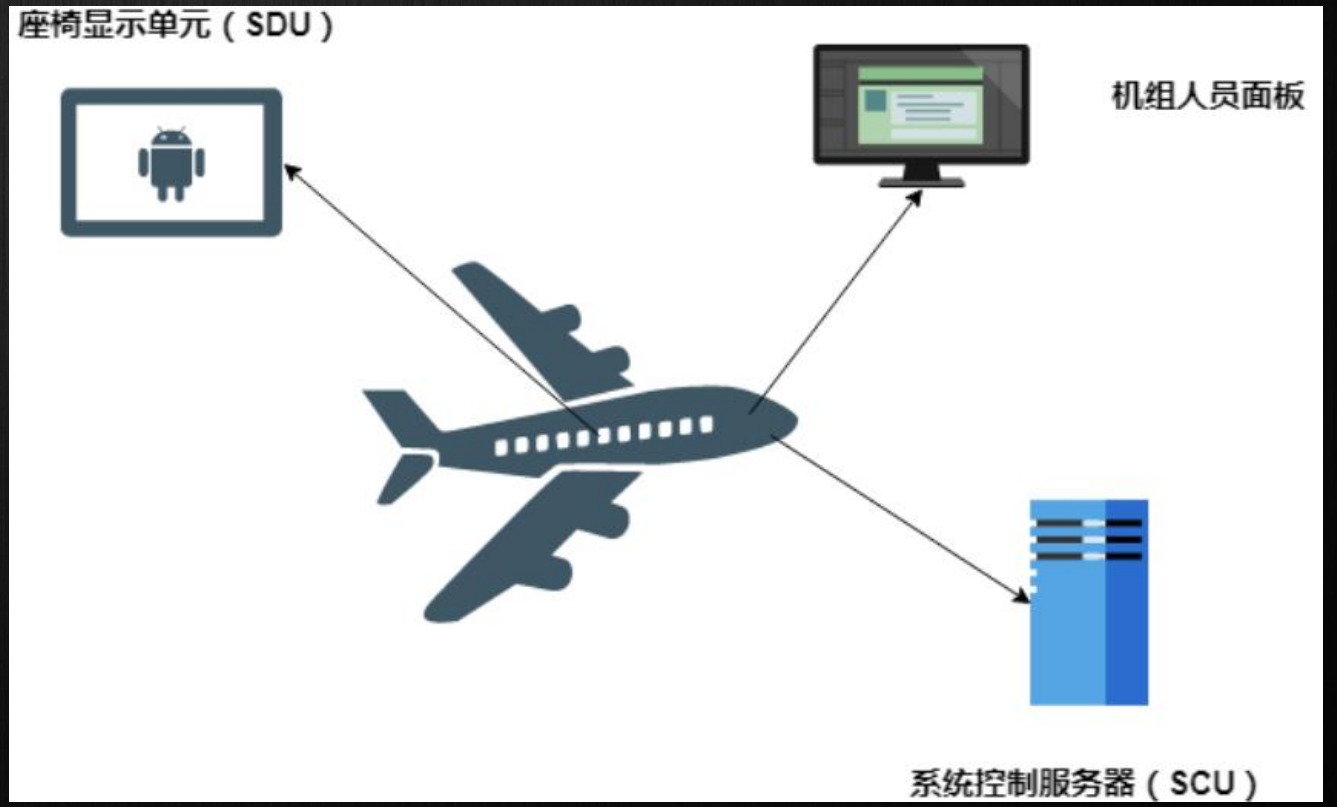
系统控制单元（SCU）：是一台机载服务器，可以查询飞机的实时信息，比如风速，海拔，机舱室温，主要是通过航空电子总线接收数据，可以将数据发送到每个座椅前面的座椅显示单元（SDU）上显示数据。

座椅显示单元（SDU）：最新版本都是基于android的，可以用于接收飞机的实时信息，也可以看电影，看电子书，甚至可以接入到互联网中。

机组人员面板（cabin crew panel）：机组人员可以通过这些单元控制飞机上的功能，比如机舱的灯，发布公告，来满足乘客需求。

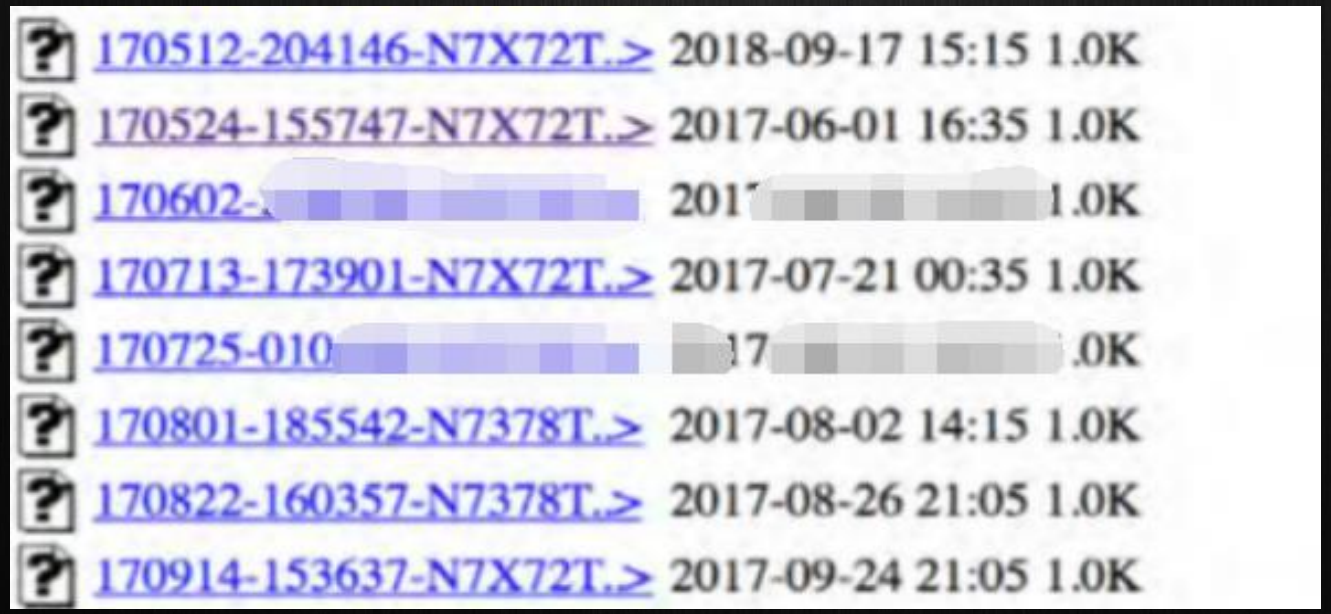
波音777飞机架构和系统介绍

6.飞行娱乐系统（IFE）



逆向波音777飞机的机组信息系统固件

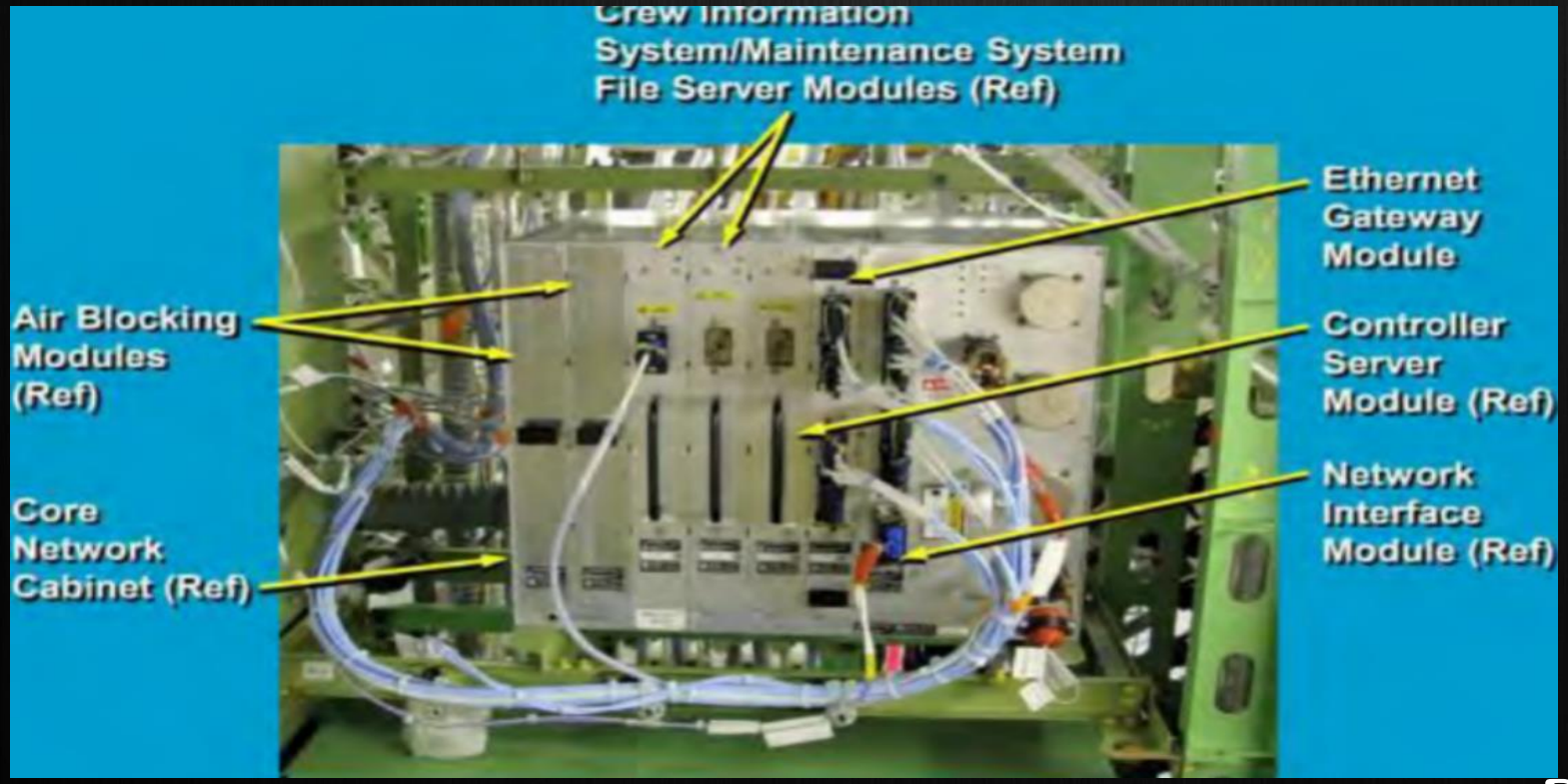
下面的逆向和漏洞分析主要是对霍尼韦尔开发的机组信息系统/维护系统（CIS/MS）的研究，因为其中的漏洞可能可以从开放数据网络（ODN）进入到到通用数据网络（CDN）中。



逆向波音777飞机的机组信息系统固件

1.核心网络机箱（CNC）的攻击面

核心网络机箱（CNC）的原理图如下



逆向波音777飞机的机组信息系统固件

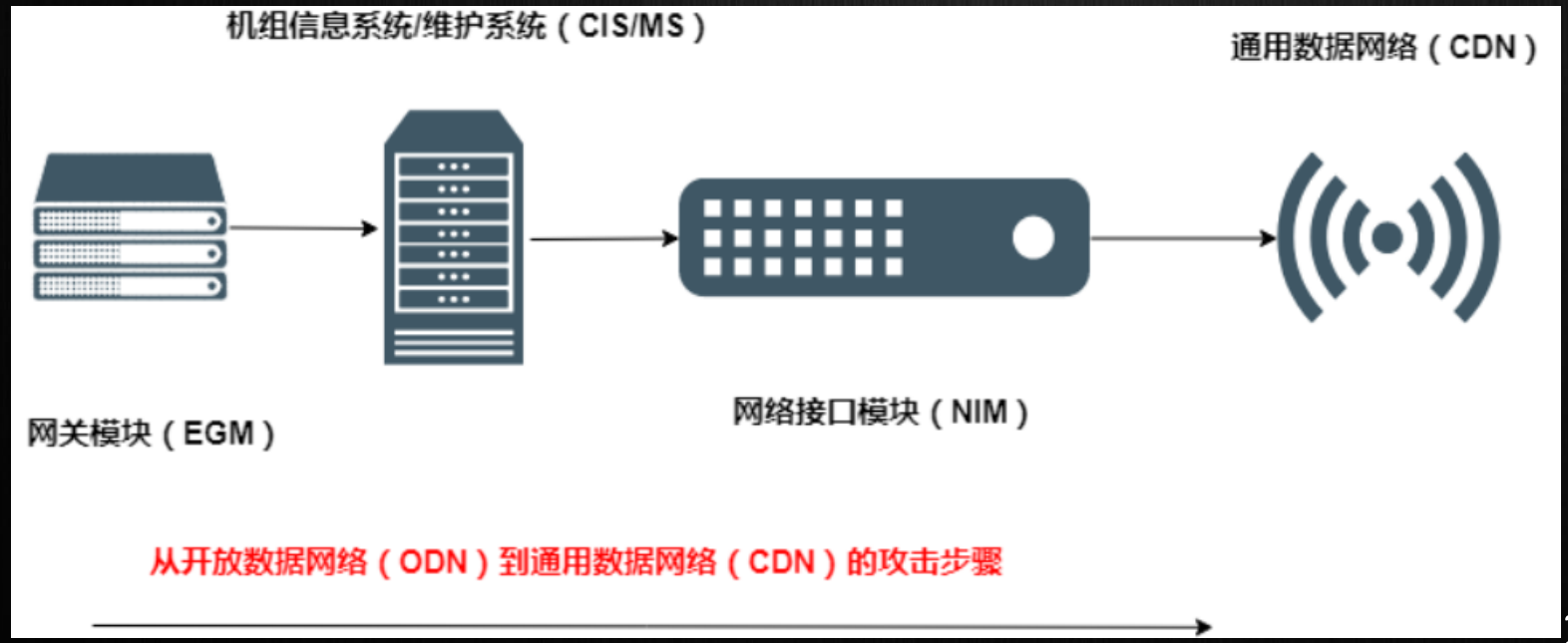
1.核心网络机箱（CNC）的攻击面

网络接口模块（NIM）和网关模块（EGM）是实现数据过滤和网络隔离的主要模块，只要拿下网络接口模块（NIM），网关模块（EGM）和机组信息系统/维护系统文件服务器模块（CIS/MS），就会在网络中出现一条到达核心网络的链路

逆向波音777飞机的机组信息系统固件

下面总结了从开放数据网络（ODN）到通用数据网络（CDN）需要实施的步骤：

1.机载娱乐系统（IFE）和航空无线网络都在开放数据网络（ODN）中，只需要找到一个网关模块（EGM）中的漏洞就可以进入机组信息系统/维护系统（CIS/MS）中；
2.这一步可以利用vXworks 6.0中的漏洞进入机组信息系统/维护系统（CIS/MS）中；
3.现在就可以通过网络接口模块（NIM）到达通用数据网络（CDN）中；



逆向波音777飞机的机组信息系统固件

2.固件信息

主要关心：网络接口模块（NIM），网关模块（EGM）和机组信息系统/维护系统模块（CIS/MS）

网关模块（EGM）是基于linux的，首先考虑的就是分析防火墙规则了解网关模块（EGM）会允许哪些流量可以通过；
机组信息系统/维护系统模块（CIS/MS）是在vXworks 6.0上运行的，用于支持网络和数据互联，没有发现NX和编译器保护；
网络接口模块（NIM）的终端系统使用的是GE公司的航空A664-P7终端系统，需要了解终端的连接方式。

固件的编译时间2013年，基于vXworks 6.0使用GCC 3.1.2编译的

```
GCC: (GNU) 3.1.2 20020712 (Wind River vxworks-6.0) (built20130227)
```

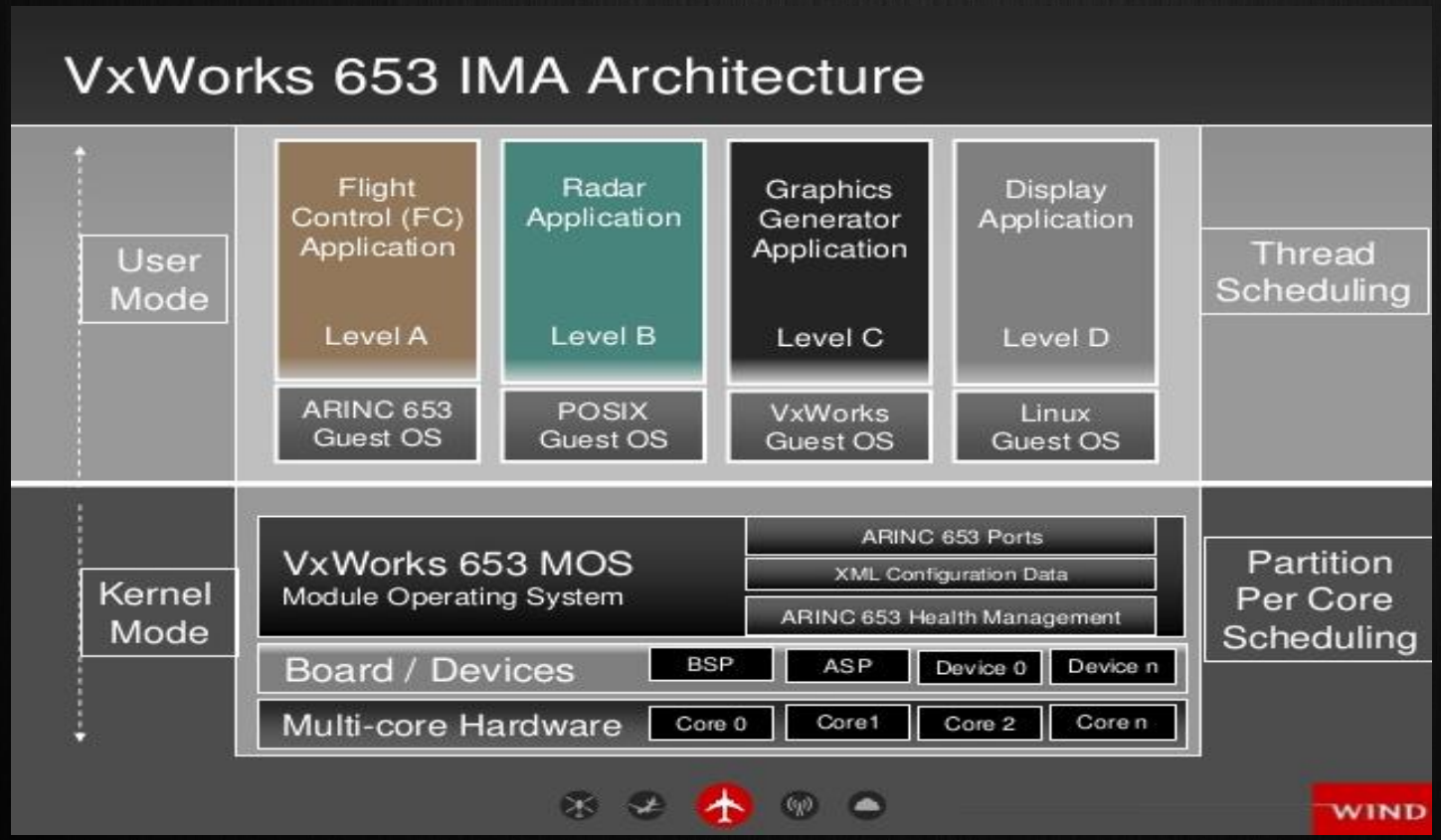
逆向波音777飞机的机组信息系统固件

vXworks在用户态下的Processs和Libraries：
Processs：
Manager.vxe，ODLF.vxe，tftpd.vxe，FBM.vxe，mtf_main.vxe，mtf_rtp.vxe，omls.vxe，wlanmf_rtp.vxe

Libraries：
DiskUtilities.so，DiskplayUtilities.so，AMI.so，OrderedList.so，SNMP.so，cisUtl.so，JSON.so，Messaging.so

逆向波音777飞机的机组信息系统固件

vXworks的体系结构：



在机组信息系统固件中发现的漏洞

在固件中发现了很多不安全的函数调用，比如strcpy，sprintf，strcat等函数，在可执行文件和共享库代码中都发现了这样的函数，这些不安全的函数就是机组信息系统/维护系统（CIS/MS）固件最重要的漏洞挖掘点。

下面会讲几个固件中的漏洞，可以让从开放数据网络（ODN）进入通用数据网络（CDN）变成现实。

在机组信息系统固件中发现的漏洞

1.manager.vxe的TFTP栈溢出漏洞
manager.vxe主要用于文件传输服务，远程未经身份验证的攻击者可以利用其中的漏洞进行任意代码执行。

```
value = recvfrom(
serversocket,
&requestbuffer,
0x200u,0,
&clientaddr
&clientaddrlength
);

if (value = -1)
  {
    sprintf(&log_buffer,"TFTP --> could not read on TFTP port &d",fs_listen_port[server_instance]);
    rtpLog(3,0,&log_buffer);
    goto LABEL_107;
  }

  opcode = ((unisigned_int8)requestbuffer << 8) | ((requestbuffer & 0xFF00) >> 8);
  strcpy(&fileforoptneg,(const char*)&requestbuffer + 2,0x80u);
  v20 = 0;
  sprintf(
    &log_buffer,
    "%s --> %s request Received for file %s from",
    &fs_tftp_task_name[20 * server_instance],
    &opcode_string[5 * opcode],
    &fileforoptneg);
```

在机组信息系统固件中发现的漏洞

漏洞利用：
在这种操作中，攻击者可以控制TFTP请求中的两个参数，分别是目标文件和操作码，为了实现任意代码执行，需要将log_buffer溢出0x700字节，还要使操作码字符串索引引用到足够大的内存空间存放shellcode。

```
.data:080DF040                          public opcode_string
.data:080DF040 opcode_string            db 'INV',0
.data:080DF044                          db   0
.data:080DF045 aRrq                     db 'RRQ',0
.data:080DF049                          db   0
.data:080DF04A aWrq                     db 'WRQ',0
.data:080DF04E                          db   0
.data:080DF04F aData                    db 'DATA',0
.data:080DF054 aAck                     db 'ACK',0
.data:080DF058                          db   0
.data:080DF059 aErr_0                   db 'ERR',0
.data:080DF05D                          db   0
.data:080DF05E aOack                    db 'OACK',0
.data:080DF063                          db   0
```

在机组信息系统固件中发现的漏洞

2.TFTP RRQ文件名缓冲区溢出漏洞

这一部分的代码主要用于处理TFTP的RRQ文件名，inbuffer_ptr指针会指向这个请求，在地址0x8068664上inbuffer_ptr会增加两个字节，strcpy会把源文件分配到大小为0x100byte的tmpstr栈缓冲区上，这里明显存在一个栈溢出漏洞。

```
.text:0806865E    sub     esp, 8
.text:08068661    mov     eax, [ebp+inBuffer_ptr]
.text:08068664    add     eax, 2
.text:08068667    push    eax                 ; src
.text:08068668    lea     eax, [ebp+tmpstr]
.text:0806866E    push    eax                 ; dest
.text:0806866F    call    strcpy
.text:08068674    add     esp, 10h
.text:08068677    sub     esp, 0Ch
.text:0806867A    lea     eax, [ebp+tmpstr]
.text:08068680    push    eax                 ; s
.text:08068681    call    strlen
.text:08068686    add     esp, 10h
.text:08068689    mov     [ebp+nFilenameLen], eax
.text:0806868C    sub     esp, 8
.text:0806868F    push    2Eh                 ; c
.text:08068691    lea     eax, [ebp+tmpstr]
.text:08068697    push    eax                 ; s
.text:08068698    call    strchr
.text:0806869D    add     esp, 10h
.text:080686A0    mov     [ebp+ptrTmp], eax
.text:080686A3    cmp     [ebp+ptrTmp], 0
.text:080686A7    jz      loc_8068883
```

在机组信息系统固件中发现的漏洞

2.TFTP RRQ文件名缓冲区溢出漏洞

漏洞利用：
可以从TFTP socket上读取0x200 Byte，构建ROP链控制EIP和相关寄存器达到代码执行。

"A"*160 +p32(strcpy_plt) +p32(vul) +p32(1)+p32(buf2) + p32(20)

在机组信息系统固件中发现的漏洞

3.ParseLUSFile内存破坏漏洞
ParseLUSFile函数是在diskUtils.so文件中实现的，解析时会从ODLF.vxe中读取一个16位的value值，这个值是从LUS缓冲区中读取的，问题在于这个LUS缓冲区的内容是可控的。

```
for(idxf = bytesReadc;idaf < bytesReadc + 2;++idaf)
  fileData->numberHeaders += buffer[idxf] << (8 - 8*(idxf - bytesReadc));
bytesReadd = bytesReadc + 2;

for (hdrIndex = 0;hdrIndex < fileData->numberHeaders;++hdrIndex)
{
  fileData->file[hdrIndex].headerNameLength = 0;
  for (idxg = bytesReadd;idxg < bytesReadd + 1; ++idxg )
    fileData->file[hdrIndex].headerNameLength += buffer[idxg] << 8* (idxg - bytesReadd);
  v4 = byteReadd + 1;
  strncpy(fileData->file[hdrIndex].headerNameLength,(const char*)&buffer[v4],fileData->file[hdrIndex].headerNameLength);
  fileData->file[hdrIndex].headerName[fileData->file[hdrIndex].headerNameLength] = 0;
  bytesReade = fileData->file[hdrIndex].headerNameLength + v4;
  fileData->file[hdrIndex].partNumberLength = 0;
```

在机组信息系统固件中发现的漏洞

3.ParseLUSFile内存破坏漏洞

漏洞利用：
攻击者使用可控值破坏LUS缓冲区，构造ROP链控制EIP和所需寄存器就可以达到任意代码执行。

在机组信息系统固件中发现的漏洞

4.vXworks中syscall提权漏洞

机组信息系统/维护系统（CIS/MS）的vXworks 6.0中有一组自定义的系统调用。

几个问题：
1.并没有验证从用户层传进来的指针，这样可能会实现内核层的任意内存读写；
2.这里面也使用了不安全的函数，可能会触发其他漏洞。

```
.data:0081C020                  public syscallGroupTbl
.data:0081C020 syscallGroupTbl  dd 0                    ; DATA XREF: rtpSysctlSyscall+4D↑r
.data:0081C020                                          ; rtpSysctlSyscall+AE↑r ...
.data:0081C024 dword_81C024     dd 0                    ; DATA XREF: rtpSysctlSyscall+11E↑r
.data:0081C024                                          ; syscallGroupRegister+12E↑w ...
.data:0081C028                  align 10h
.data:0081C030                  dd offset FSMSYSTEMRtnTbl
.data:0081C034                  db  26h ; &
```

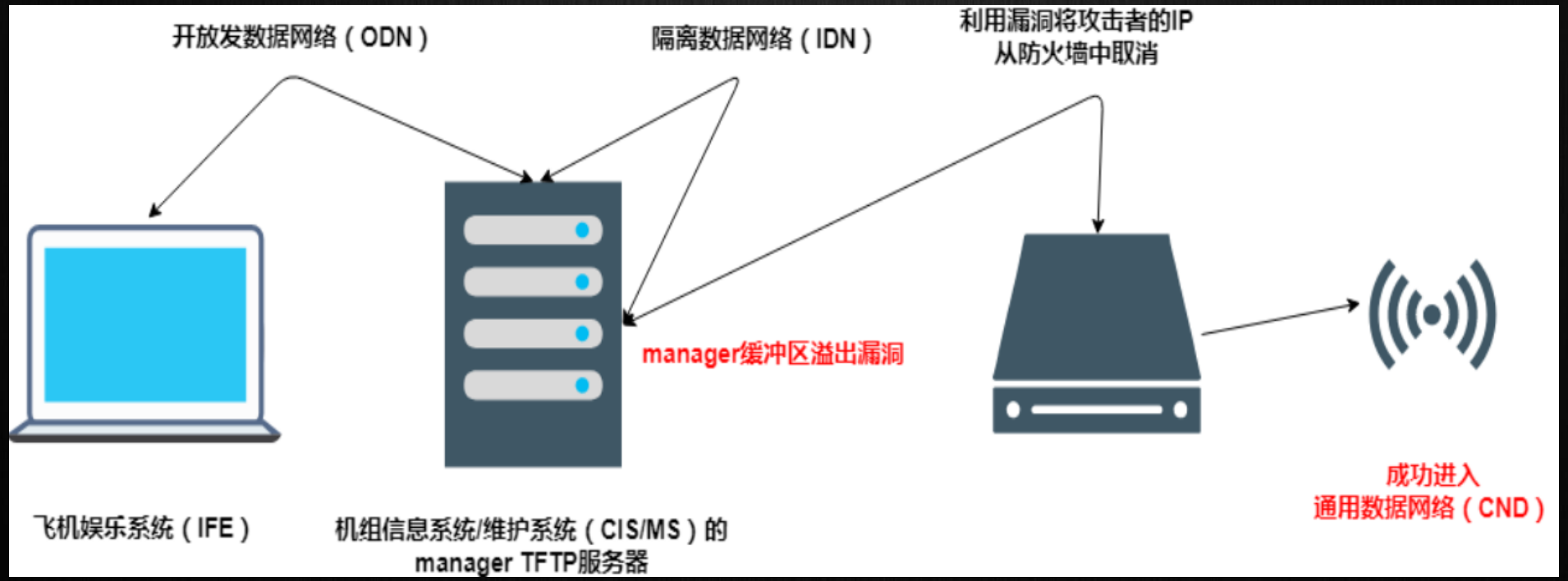
在机组信息系统固件中发现的漏洞

以下面的cissFwSetByDynFirewallRuleSc为例，它的系统调用号是0x224，在这里并没有做参数验证，并且它也允许添加任意防火墙规则到机组信息系统/维护系统（CIS/MS）的数据包过滤模块中。

```
.text:003246E3                  public cissFwSetDynFirewallRuleSc
.text:003246E3 cissFwSetDynFirewallRuleSc proc near     ; DATA XREF: .data:0081BFF0↓o
.text:003246E3
.text:003246E3 arg_0            = dword ptr  8
.text:003246E3
.text:003246E3                  push    ebp
.text:003246E4                  mov     ebp, esp
.text:003246E6                  sub     esp, 18h
.text:003246E9                  mov     eax, [ebp+arg_0]
.text:003246EC                  mov     eax, [eax+0Ch]
.text:003246EF                  mov     [esp+0Ch], eax
.text:003246F3                  mov     eax, [ebp+arg_0]
.text:003246F6                  mov     eax, [eax+8]
.text:003246F9                  mov     [esp+8], eax
.text:003246FD                  mov     eax, [ebp+arg_0]
.text:00324700                  mov     eax, [eax+4]
.text:00324703                  mov     [esp+4], eax
.text:00324707                  mov     eax, [ebp+arg_0]
.text:0032470A                  mov     eax, [eax]
.text:0032470C                  mov     [esp], eax
.text:0032470F                  call    cissFwSetDynFirewallRule
.text:00324714                  leave
.text:00324715                  retn
.text:00324715 cissFwSetDynFirewallRuleSc endp
```

现实环境下的攻击场景

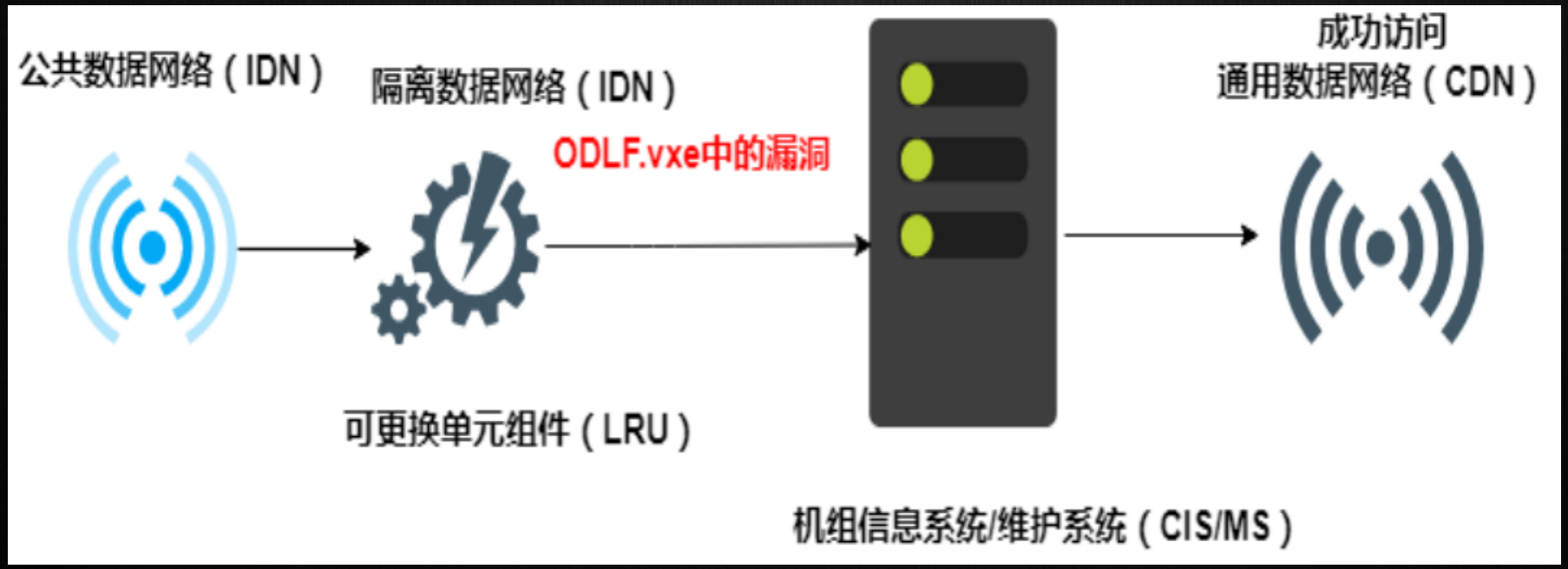
1.从机载娱乐系统（IFE）进入通用数据网络（CDN）
已经可以控制机载娱乐系统（IFE）的攻击者可以通过下面的步骤到达通用数据网络（CDN）：



现实环境下的攻击场景

2.从任意可更换单元组件（LRU）进入通用数据网络（CDN）：

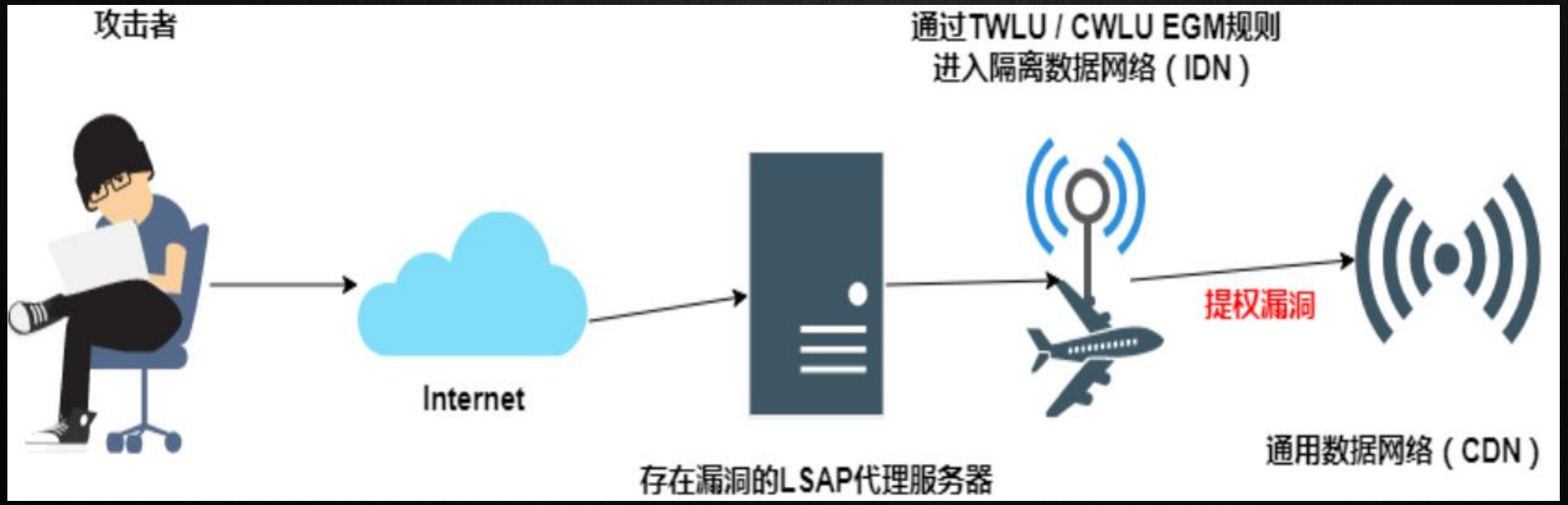
ODLF是用于实现飞机上的数据加载的，LRU是一些可更换的单元组件，ODLF可以通过ARINC615协议更新LRU中的固件。



现实环境下的攻击场景

3.从外部网络进入通用数据网络（CDN）

- 攻击在公网上存在漏洞的LSAP代理服务器
- 这样攻击者就可以控制LSAP存储库的上行链路和下行链路请求
- 攻击者就可以通过TWLU / CWLU EGM规则进入隔离数据网络（IDN）
- 利用之前的提权漏洞获得对通用数据网络（CDN）的访问权限



安全措施和总结

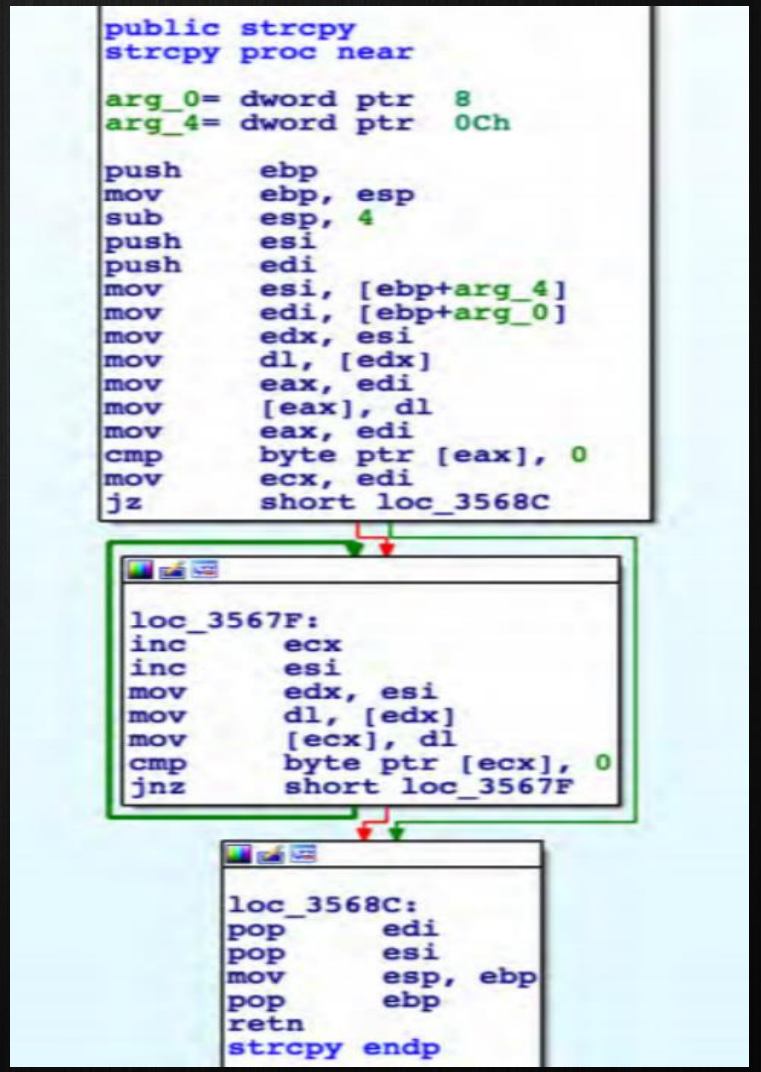
出现这些漏洞很大程度上的一个原因就是没有编译器级别的缓解措施，也没有NX保护，导致不安全的函数裸奔可以直接被利用，因此需要使用安全函数加入漏洞缓解机制重新编译固件。

安全措施和总结

下面是libc.so中的strcpy函数

很明显可以看到里面没有Canary保护

```
text:0807846C                   add     esp, 10h
text:0807846F                   mov     eax, 0
text:08078474                   mov     edi, [ebp+var_4]
text:08078477                   leave
text:08078478                   retn
text:08078478 server_task       endp
```



安全措施和总结

其实在这个分享中也没有什么比较新颖的技术，这里面的漏洞和漏洞利用技术在十年前就已经在使用了，但是考虑到这些漏洞是出现在航空领域，对人身安全的影响还是挺大的，因此这些漏洞和技术细节还是有一定价值和意义的。

之前有报道说美国某个空军基地授权一家安全公司对F-15战斗机做安全测试，他们也在F-15的航空电子系统中发现了一些可利用漏洞，希望不久的将来国内的民航领域和军方系统也会有这样的安全业务，良性的安全测试是对系统安全性最好的保证。

Thank You

山石网科

技术的信仰者