**业务安全动态感知系统**

一、介绍

业务流程风控系统，是一套监测企业业务办理过程中存在的问题的平台。

主要原理，在企业业务办理过程中，存在一些违规的操作，但是不能监测到，给企业带来损失。

二、系统原理

系统利用获取web服务器的流量，还原请求，模拟操作，监测违规造作。

在web服务器之前，对流量进行镜像，通过python工具，还原流量成http请求。对http请求进行分类入库，还原出每个用户的操作，通过对比事先录入的正确操作，达到监测的效果。

三、代码

主要使用httpparse模块

模块的作用是分析流量，还原为http请求，可以直接解析流量的pcap包，也可以直接使用linux系统的libpcap模块进行流量抓取，然后对流量进行还原。

模块主要使用命令行控制，接收参数，把结果输出到命令行。

需要修改模块的开始函数。

需要修改模块，输出到mongodb数据库。

Httpparse模块的目录结构：

|-- httpparse目录

|-- httpcap目录

|-- \_\_main\_\_.py

主要是开始文件，里面的parse\_pcap函数和parse\_live函数是一切的开始，parse\_pcap函数是读取pcap文件，parse\_live函数是利用linux系统的libpcap库来获取流量，输出结果。

|-- \_\_init\_\_.py

空文件，如果需要在外部调用模块需要把\_\_main\_\_.py内容复制到\_\_init\_\_.py里面。直接import httpcap。

|-- cleanups.py

|-- config.py

|-- constant.py

|-- content\_utils.py

|-- httpparser.py

|-- link\_layer.py

|-- live\_cap.py

|-- packet\_parser.py

|-- parse\_pcap.py

|-- pcap.py

|-- pcap\_wrapper.py

|-- pcapng.py

|-- printer.py

主要是输出脚本，通过读取config.py的配置，输出不同的结果，或者输出到文件。后期主要是修改此文件，把数据写入数据库。

|-- reader.py

|-- tcp\_assembly.py

|-- pcappy\_port目录

|-- \_\_init\_\_.py

|-- constants.py

|-- functions.py

|-- types.py

|-- tests 目录

|-- \_\_init\_\_.py

|-- tcp\_assembly\_test.py

|-- utils\_test.py

|-- main.py py文件

自己添加的脚本文件，主要是调用httpcap

四、后期规划

主要是把数据库的内容读取，去除无用数据，通过ip和cookie等，还原出用户的操作（可能需要提取返回的数据参数等，对比正确的数值），对比正确的用户操作。记录违规操作，入库，前台页面的展示。