# 业务功能参数异常检测算法文档

## 一、设计原理

用户在使用每一个业务系统中的功能的时候，都会通过get/post方式提交一定量的参数到业务系统，业务系统根据接收到的参数，进行相应内容信息的返回。若用户通过后台修改自己提交的参数，那么可以产生数据窃取、非法访问业务系统等恶意事件。

对于，针对业务系统的每一个业务功能，建立业务功能基线。用以检测业务功能的某些特定请求是否合规、合法，检测对象有参数类型（字母、数据、符号、中文）、长度。

## 二、数据源

### （1）训练数据

#### 训练样本生成/采集方式

获取hdfs上的业务系统最近一个星期的https记录，以此作为训练样本。

#### 数据源地址

Hdfs：10.130.10.62

路径：hdfs://10.130.10.62:9000/usr/hadoop/nta/http/

#### 数据源格式

|  |  |
| --- | --- |
| 字段 | 备注 |
| flow\_id | id |
| srcip | 源ip |
| dstip | 目的ip |
| host | 域名 |
| protocol | 协议 |
| request\_date | 请求时间 |
| response\_date | 响应时间 |
| request\_urL | 请求url |
| request\_uri | 请求uri |

### （2）检测数据

#### 检测数据生成/采集方式

获取hdfs上的业务系统最近一小时的https记录，以此作为检测数据。

#### 数据源地址

Hdfs：10.130.10.62

路径：hdfs://10.130.10.62:9000/usr/hadoop/nta/http/

#### 数据源格式

|  |  |
| --- | --- |
| 字段 | 备注 |
| flow\_id | id |
| srcip | 源ip |
| dstip | 目的ip |
| host | 域名 |
| protocol | 协议 |
| request\_date | 请求时间 |
| response\_date | 响应时间 |
| request\_urL | 请求url |
| request\_uri | 请求uri |

## 三、特征处理

1）业务功能定义

业务功能指业务系统中带有提交参数的子功能或URL。对于常见的URL，一般语法格式为(带方括号[]的为可选项)：

protocol :// hostname[:port] / path / [;parameters][?query]#fragment

我们将其写成以下格式：

protocol://host:path?key1=value1&key2=value2...

**定义方法一：**

由于http/https有get和post两种请求方式。对于get型请求，我们直接获取URL来处理；对于post型请求，请求参数分为两种：url上的参数和表单上的参数。我们将url上的参数和表单的参数拼接一起，作为总体参数，即构造出了带有完整参数的URL。

这里我们定义的业务功能有以下三点的要求：

①相同host

②相同path

③在三个不同IP都访问了这个host+path的前提下，存在共同使用的key-value（可选）

对于满足以上三点的url，构造“host+path+共同使用的key-value”以此作为业务功能。具体例子如下：

IP1-URL1:https://113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?action=add&format=normal&display=1&username=232445467

IP2-URL2:https://113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?action=add&format=normal&username=231145467

IP3-URL3:https://113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?action=add&username=23221467

IP4-URL4:https://113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?action=del&username=23221207

那么最终定下来的业务功能是：

113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?action=add

**定义方法二：**

客户配置。

2）请求参数的定义

①GET型

对于GET型请求，url上的参数就是其请求参数。

请求网址例子：

https://113.108.195.66:28080/WorkFlowArea/WorkFlowRun/Index?flowid=7885774a-b16e-4073-9532-b3295ce8169f&stepid=624480e4-da39-460a-86cf-a134f33f68ab&instanceid=8b78638d-2f8f-431e-a936-b5015ff9fffd&taskid=6ef99934-925c-40fc-9105-37404412d8b8&groupid=51955654-e37b-4eef-b612-f09e647c059b&display=1

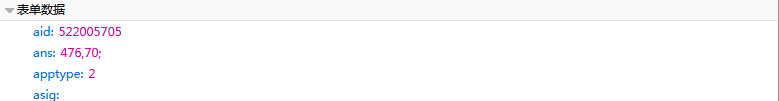
②POST型

对于POST型请求，需要结合url中的参数和表单数据，以此作为完整请求参数。

请求网址例子：

<https://ssl.captcha.qq.com/cap_union_new_verify?random=1521622298522>

表单数据：



## 四、算法模型

### （1）算法名称

无

### （2）算法详情

无

## 五、训练过程详情步骤

### （1）训练周期

一周训练一次

### （2）训练步骤

1. GET型请求的参数检测

①筛选包含业务功能的url

过滤条件需要满足如下条件：

a、目的IP是业务系统所在IP；

b、Content-type属于text/html、application/json、application/x-www-form-urlencoded 其中一种；

c、Uri中包含“.action”关键字；

过滤条件为为：a && (b || c)

②url转码

将url进行编码转换，如将“%21”转换成“!”。其中常用字符url编码格式如下：

|  |  |  |  |
| --- | --- | --- | --- |
| **ASCII字符** | **URL编码** | **ASCII字符** | **URL编码** |
| 空格 | %20 | : | %3A |
| ! | %21 | ; | %3B |
| \ | %22 | < | %3C |
| # | %23 | = | %3D |
| $ | %24 | > | %3E |
| & | %26 | ? | %3F |
| ' | %27 | @ | 40% |
| ( | %28 | \ | %5C |
| ) | %29 | | | %7C |
| \* | %2A | } | %7D |
| + | %2B | { | %7B |
| , | %2C |  |  |

具体见<http://www.w3school.com.cn/tags/html_ref_urlencode.html>

③分割功能和参数

将uri按照“?”分割出业务功能（资源定位）和请求参数。索引0为业务功能，即资源定位的字符；索引1为请求参数的字符。

对于POST型的请求，其参数字符也包括表单的所有数据，将表单的数据与url上的参数连接起来处理。

对于参数字符，使用“&”进行切分，得到一个key-value的列表，再使用“=”切分，得到每个key-value对。

④建立业务功能请求参数基线

按业务功能进行分组，计算每个参数key对应所有value的字符长度、和字符类型（大小写字母、数字、符号、中文），得到基线：

业务功能:{key:{(参数长度范围,字符类型)}}

⑤将基线结果存储到postgresql

## 六、检测过程详情步骤

### （1）检测周期

一小时检测一次

### （2）检测步骤

注意：需要训练好基线后，才能运行检测部分的代码。

①筛选上一小时的https数据，获得业务系统的https数据

②获取postgresql上的基线结果

③将业务功能请求uri中每个key对应的value，与该功能的基线作对比；若字符长度比基线长，或字符类型与基线不一样，那么告警。

④存储告警数据

## 七、输出结果

### （1）模型保存位置

基线结果保存到Postgrel，表结构为

|  |  |
| --- | --- |
| id | Row id |
| app\_utility | 业务功能 |
| app\_uri | 业务功能中的uri |
| app\_param | 业务功能中的参数 |
| param\_name | 除业务功能外的参数名称 |
| param\_length | 参数长度范围 |
| param\_type | 参数类型 |
| response\_baseline | 响应基线值（3σ）（没用到此字段） |
| response\_type | 响应基线类型（size表示响应报文大小基线，time表示响应时间基线）（没用到此字段） |
| time | 保存时间 |
| result\_type | 结果类型（0表示参数检测结果，1表示响应结果） |

### （2）检测结果

#### 保存方式

保存到POSTGRESQL或HDFS

#### 保存位置

POSTGRESQL：10.130.10.22

#### 结果格式

检测结果保存到ES，表结构为

|  |  |
| --- | --- |
| 字段 | 备注 |
| flow\_id | id |
| srcip | 源ip |
| dstip | 目的ip |
| host | 域名 |
| protocol | 协议 |
| request\_date | 请求时间 |
| response\_date | 响应时间 |
| request\_urL | 请求url |
| app\_utlity | 业务功能 |
| app\_uri | 请求uri |
| app\_para | 参数字符 |
| extra\_app\_para\_baseline | 参数的基线  （key1=value1&len\_min-len\_max&char#key2=value2&len\_min-len\_max&char） |
| parameter\_detect\_result | 参数检测结果（异常长度、异常类型） |
| baseline\_time | 响应时间基线（没用到此字段） |
| baseline\_size | 响应报文大小基线（没用到此字段） |
| detect\_time\_result | 响应时间检测结果（没用到此字段） |
| detect\_size\_result | 响应报文大小检测结果（没用到此字段） |
| result\_type | 结果类型（0为参数检测结果，1为响应检测结果） |

#### 结果示例



## 模型评估

无