# DDOS攻击检测

## 一、DDoS\_Syn\_flood

### 1、设计原理

黑客利用工具或者肉鸡，不停的向目标服务器发送syn报文。当目标服务器接收到syn报文后，并返回一个syn ack命令，即服务器开始保持半开连接状态，而攻击肉鸡在接收到syn ack命令后不再返回状态出来，让其一直保持半开连接状态，直至服务器资源被大量消耗，使正常的连接请求无法被响应。

### 2、数据源

|  |  |
| --- | --- |
| **字段** | **备注** |
| srcip | 源IP地址 |
| dstip | 目的IP地址 |
| srcport | 源端口 |
| dstport | 目的端口 |
| protocol | 协议类型 |
| syn | SYN位置值不为空的报文个数 |
| ack | ACK位置值不为空的报文个数 |
| rst | RST位置值不为空的报文个数 |
| fin | FIN位置值不为空的报文个数 |
| urg | URG位置值不为空的报文个数 |
| uppkts | 上行包数量 |
| downpkts | 下行包数量 |
| upbytesize | 上行字节数 |
| downbytesize | 下行字节数 |
| starttime | 开始时间 |
| endtime | 结束时间 |
| recordtime | 记录时间 |

### 3、特征处理

1. 无

### 4、模型算法

对于一个黑客来说，可能使用一台资产多线程攻击服务器，也可能使用多台机器对一台服务器进行攻击。且syn的攻击源都是仅仅只是发送大量syn包，且返回ack包后没有后文，即满足如下条件：

1. 外部IP处于源IP，可能是一个源IP多个源端口，也可能是多个源地址多个源端口
2. 外部源IP给服务器发送syn包，且标志位仅仅只有syn为1，其他位置都为0
3. 服务器返回大量仅仅只有syn和ack标志的数据

当前检测的方式为，假设某个服务器在10s内接收标志位仅仅只有syn为1的数据超过3000条，则认为收到syn攻击。

## 二、DDoS\_ack\_flood

### 1、设计原理

黑客利用工具或者肉鸡，不停的向目标服务器发送ack报文。当目标服务器接收到ack报文后，并返回一个ack命令，即服务器开始保持连接状态，而攻击肉鸡在接收到ack命令后不再返回状态出来，让其一直保持连接状态，直至服务器资源被大量消耗，使正常的连接请求无法被响应。

### 2、数据源

|  |  |
| --- | --- |
| **字段** | **备注** |
| srcip | 源IP地址 |
| dstip | 目的IP地址 |
| srcport | 源端口 |
| dstport | 目的端口 |
| protocol | 协议类型 |
| syn | SYN位置值不为空的报文个数 |
| ack | ACK位置值不为空的报文个数 |
| rst | RST位置值不为空的报文个数 |
| fin | FIN位置值不为空的报文个数 |
| urg | URG位置值不为空的报文个数 |
| uppkts | 上行包数量 |
| downpkts | 下行包数量 |
| upbytesize | 上行字节数 |
| downbytesize | 下行字节数 |
| starttime | 开始时间 |
| endtime | 结束时间 |
| recordtime | 记录时间 |

### 3、特征处理

1. 无

### 4、模型算法

对于一个黑客来说，可能使用一台资产多线程攻击服务器，也可能使用多台机器对一台服务器进行攻击。且ack的攻击源都是仅仅只是发送大量ack包，且返回ack包后没有后文，即满足如下条件：

1. 外部IP处于源IP，可能是一个源IP多个源端口，也可能是多个源地址多个源端口
2. 外部源IP给服务器发送ack包，且标志位仅仅只有ack为1，其他位置都为0
3. 服务器返回大量仅仅只有ack标志的数据

当前检测的方式为，假设某个服务器在10s内接收标志位仅仅只有ack为1的数据超过3000条，则认为收到syn攻击。

## 三、DDoS\_udp\_flood/ DDoS\_ICMP\_flood

### 1、设计原理

对于一个黑客来说，可能使用一台资产多线程攻击服务器，也可能使用多台机器对一台服务器进行攻击。黑客利用udp协议或者icmp协议大量向服务器发起小包请求，且没有连接状态，由此造成服务器崩溃。

### 2、数据源

|  |  |
| --- | --- |
| **字段** | **备注** |
| srcip | 源IP地址 |
| dstip | 目的IP地址 |
| srcport | 源端口 |
| dstport | 目的端口 |
| protocol | 协议类型 |
| syn | SYN位置值不为空的报文个数 |
| ack | ACK位置值不为空的报文个数 |
| rst | RST位置值不为空的报文个数 |
| fin | FIN位置值不为空的报文个数 |
| urg | URG位置值不为空的报文个数 |
| uppkts | 上行包数量 |
| downpkts | 下行包数量 |
| upbytesize | 上行字节数 |
| downbytesize | 下行字节数 |
| starttime | 开始时间 |
| endtime | 结束时间 |
| recordtime | 记录时间 |

### 3、特征处理

1. 无

### 4、模型算法

对于一个黑客来说，可能使用一台资产多线程攻击服务器，也可能使用多台机器对一台服务器进行攻击。且udp/icmp的攻击源都是仅仅只是发送大量udp/icmp协议请求，即满足如下条件：

1. 外部IP处于源IP，可能是一个源IP多个源端口，也可能是多个源地址多个源端口
2. 外部源IP给服务器发送大量udp/icmp请求

当前检测的方式为，假设某个服务器在10s内接收udp/icmp数据超过3000条，则认为收到udp/icmp flood攻击。

## 四、CC攻击

### 1、设计原理

根据监控的页面资源是否超过所设定的阈值，以及页面的访问量、访问时间等基线，结合威胁情报判断业务功能url是否遭受了cc攻击。

### 2、数据源

http协议

|  |  |
| --- | --- |
| 字段解释 | 字段名 |
| 源IP | srcip |
| 目的IP | dstip |
| 协议 | proto |
| 访问链接 | uri |
| 访问HOST | host |
| 请求时间 | request\_date |
| 响应时间 | response\_date |
| 请求内容类型 | Content\_type |
| 响应内容类型 | Content\_type |
| 表单数据 | Content\_pathc |

### 3、特征处理

1. 无

### 4、模型算法

当某个业务系统功能url满足如下三个条件，则表示业务系统功能url受到了cc攻击：

1. 业务系统功能的URL平均访问次数和平均访问时间
2. 匹配威胁情报发现CC攻击
3. CPU、内存、磁盘使用超过特定阈值