**接口说明文档**

**2019.06.28**

# 接口要求



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **表1外部接口** | | | | | |
| 接口 | 发送方 | 接收方 | 传输内容 | 通信方式 | 数据格式 |
| I1 | 数据总线 | 威胁感知设备 | 元数据指定字段信息 | ZMQ | Protobuf |
| I2 | 威胁感知设备 | 数据总线 | 检测结果信息 | ZMQ | Protobuf |
| I3 | 数据总线 | 规整存储软件 | 入库字段信息 | KAFKA | 二进制流 |

# 元数据检测输入输出接口定义

## 接口基本说明

元数据威胁检测输入输出接口在联调测试阶段优先实现zmq+protobuf， 同时也需支持tcp/udp、Kafka等通信方式。

protobuf格式介绍（https://developers.google.cn/protocol-buffers/）

zmq消息介绍（http://zeromq.org/）

zmq中文参考（https://www.cnblogs.com/fengbohello/p/4230135.html）

kafka介绍（https://kafka.apache.org/）

## 接口性能测试

zmq+pb打包解包，单点对单点，单线程，简单测试性能测试数据：

平均1000万条记录，耗时约10秒，数据量220MB。

其中单线程CPU

使用zmq+pb，单进程收发基本能达到80到100wpps的性能要求，测试过程中有少量丢包，在实际开发中优化措施：

1. 对于单点对单点性能不能满足要求时，可以使用多点对多点的方式扩展性能，并发多个单点对单点发送。
2. 接收方收到数据后，把数据交给线程池异步处理，避免解包的阻塞。

占用高，跑满时有约10%发送失败，接收方丢包率小于0.05%。

测试结论：

## Protobuf

### 生成Protobuf源码

需先安装protobuf，然后基于“元数据检测输入输出接口定义”文档后的pb定义，编写pb文件，例如：

// filename : sensor\_log.proto

syntax = "proto2";

package Sensor;

message TESTMSG {

required string line\_info =1;

optional string sip=2;

optional string dip=3;

required int32 sport=4;

required int32 dport=5;

}

然后编译pb文件，生成C源码文件：

protoc-c --c\_out=. sensor\_log.proto

会生成sensor\_log.pb-c.\*文件，加入工程即可使用，链接选项“-lprotobuf-c”。

chengduzz.proto是一份已经写好的pb文件，可以直接使用或者双方协商增删改后使用。

头文件里有pb消息的调用接口，简单示范代码如下：

Sensor\_\_TESTMSG message=SENSOR\_\_TESTMSG\_\_INIT;

message. line\_info = "1234567890";

message.sip= " 127.0.0.1";

message.dip =" 127.0.0.1";

message.sport = 123;

message.dport = 123;

### 打包

char buf[500];

size\_t buf\_len = 500;

sensor\_\_testmsg\_\_pack(&message, (uint32\_t\*)buf);

buf\_len = sensor\_\_testmsg\_\_get\_packed\_size(&message);

### 解包

Sensor\_\_TESTMSGmsg=NULL;

msg = sensor\_\_testmsg\_\_unpack(NULL, buf\_len, (const uint32\_t\*)buf);

（pb打包后的数据使用zmq发送）

sensor\_\_testmsg\_\_free\_unpacked(msg, NULL)

## ZMQ

### zmq环境初始化

使用zmq需要包含头文件，定义一个全局的ctx变量：

#include <zmq.h>

void \*zmq\_ctx;

ctx需要在程序启动时初始化，结束时销毁，指针为NULL表示失败：

zmq\_ctx = zmq\_ctx\_new();

zmq\_ctx\_term(zmq\_ctx);

### zmq发送接口

zmq发送需要使用ZMQ\_PUSH推送模式，函数zmq\_socket创建端口，设置参数，然后连接对端：

int rc; // rc==0 成功，非0失败

int zero = 0;

int hwm = 100000;

char \*who = "tcp://127.0.0.1:5555";

void \*s;

s = zmq\_socket (zmq\_ctx, ZMQ\_PUSH);

if (!s) {...}

rc = zmq\_setsockopt(s, ZMQ\_LINGER, &zero, sizeof (zero));

if (rc != 0) {...}

rc = zmq\_setsockopt (s, ZMQ\_SNDHWM, &hwm, sizeof (hwm));

if (rc != 0) {...}

rc = zmq\_connect (s, who);

if (rc != 0) {...}

// 发送的返回值rc是发送字节数，大于0成功，小于0失败

rc = zmq\_send(s,buf, buf\_len, ZMQ\_DONTWAIT);

### zmq接收接口

zmq接收与发送类似，但使用的是zmq\_bind和zmq\_recv，模式是ZMQ\_PULL，与发送的ZMQ\_PUSH对应。

int rc;

int hwm = 100000;

char \*who = "tcp://\*:5555";

void \*s;

s = zmq\_socket (ctx->zmq\_ctx, ZMQ\_PULL);

if (!s) {...}

rc = zmq\_setsockopt (s, ZMQ\_RCVHWM, &hwm, sizeof (hwm));

if (rc != 0) {...}

rc = zmq\_bind(s, who);

if (rc != 0) {...}

char buf[1024] = {0};

size\_tbuf\_len = 1024;

// 接收的返回值rc是接收字节数，大于0成功，小于0失败

rc = zmq\_recv(s, buf, buf\_len,0);

# 元数据检测输入输出字段定义

## 检测总流程

元数据威胁检测的总体流程是数据总线按我方对字段的需求，重新裁剪元数据并封装成protobuf格式，通过zmq接口发送到威胁感知设备，威胁感知设备对元数据消息解封装取得元数据后进行威胁检测，并输出告警，告警信息封装成protobuf格式，通过zmq接口发送到数据总线处理程序。

Protobuf格式介绍（<https://developers.google.cn/protocol-buffers/>）

Zmq消息介绍（<http://zeromq.org/>）

## 输入字段类型约定

元数据消息输入及检测结果输出的数据类型约定如下：

1. 时间类型全部采用uint64类型，不使用字符串类型；
2. IP地址采用uint32类型表示，为了兼容IPv4和IPv6地址，将IP字段设置为repeated字段。
3. 16进制内容约定采用bytes类型

## 威胁检测设备与数据总线交互protobuf格式

### 数据总线发送的元数据消息格式

message SENSOR\_LOG

{

required int32 mesage\_type =1;

optional META\_HTTP http =2;

optional META\_DNS dns =3;

optional META\_MAIL mail =4;

optional META\_SSL ssl =5;

optional META\_SQL sql =6;

optional META\_FTP ftp =7;

}

### 威胁检测设备返回的检测结果消息格式

message RESULT\_LOG

{

required uint32 message\_type = 1;//消息类型，详见7.10

optional RULE\_DOLOG rule\_dolog = 2;//规则或IOC告警日志

optional DNS\_DOLOG dns\_dolog = 3;//DNS告警日志

optional SANDBOX\_DOLOG sandbox\_dolog = 4;//沙箱告警日志

}

## 数据总线输入元数据消息定义

元数据消息需要的字段信息如下表，每个消息是一个完整的检测单元，应包含完整的请求信息和响应信息字段，针对具体环境限制，如果是单边流量导致只有请求或者只有响应的数据，则只填充对应的字段，其余的留空。

以下标棕色的为通用字段，标红色的为检测字段，其他为展示字段。

### HTTP

### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| uri | string | 是 | 请求的资源 | HTTP包头的URI字段 |
| host | string | 是 | 请求的域名 | HTTP包头的host字段 |
| origin | string | 否 | 请求的原始来源 | HTTP包头的origin字段 |
| cookie | string | 否 | Cookie | HTTP包头的cookie字段 |
| agent | string | 否 | 请求者信息 | HTTP包头的user-Agent字段 |
| referer | string | 否 | 链接来源 | HTTP包头的referer字段 |
| req\_data | bytes | 否 | 请求方向body的数据 | 截取请求方向部分数据，最多1000字节 |
| rsp\_data | bytes | 否 | 响应方向body的数据 | 截取响应方向部分数据，最多1000字节 |
| method | string | 是 | HTTP请求方法 | HTTP包头的method字段 |
| status | int32 | 否 | http状态码 | http响应状态 |
| setcookie | string | 否 |  | http包头中的Set\_Cookie字段 |
| content\_type | string | 否 |  | http包头中的Content\_Type字段 |
| accept\_language | string | 否 |  | HTTP包头的accept\_language字段 |

#### META\_HTTP

syntax = "proto2";

package Sensor;

message META\_HTTP

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

repeated uint32 sip = 3 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

required string uri = 9 ; //请求的资源

required string host = 11; //请求的域名

optional string origin = 13; //请求的原始来源

optional string cookie = 14; //Cookie

optional string agent = 15; //请求者信息

optional string referer = 16; //链接来源

optional bytes req\_data = 18; //请求方向body的数据

optional bytes rsp\_data = 19; //响应方向body的数据

required string method = 20; //HTTP请求方法

optional int32 status = 21; //http状态码

optional string setcookie = 22;

optional string content\_type = 23;

optional string accept\_language = 24;

}

### DNS

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| dns\_type | int32 | 是 | DNS访问类型 | 请求为0，响应为1 |
| reply\_code | int32 | 否 | DNS响应结果状态 | 响应报文中Flags字段中的reply code信息 |
| host | string | 是 | host | 请求的域名信息 |
| addr | string | 重复 | 地址资源 | DNS的A记录，表示该host对应的IP地址信息 |
| cname | string | 重复 | 域名的规范名称 | DNS的CNAME记录，表示该host域名的其他别名记录 |
| mx | string | 重复 | 邮件交换记录 | DNS的MX记录 |
| txt | string | 重复 |  | 域名附加信息 |

#### META\_DNS

message META\_DNS

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

repeated uint32 sip = 3 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

required int32 dns\_type = 9 ; //DNS访问类型

optional int32 reply\_code = 10; //DNS响应结果状态

required string host = 12; //host

repeated string addr = 14; //地址资源

repeated string cname = 15; //域名的规范名称

repeated string mx = 16; //邮件交换记录

repeated string txt = 17;

}

### POP3/SMTP/IMAP

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| proto | string | 是 | 应用协议 | 邮件使用的应用协议（pop3，imap，smtp） |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| mid | string | 否 | 邮件id号 | 数据包中的message-id信息 |
| time | string | 是 | 邮件发送/接收时间 | 邮件包头中的邮件发送/接收时间 |
| mail\_from | string | 是 | 邮件发送人 | 来自于邮件头相应字段 |
| to | string | 是 | 邮件接收人 | 来自于邮件头相应字段 |
| cc | string | 是 | 邮件抄送人 | 来自于邮件头相应字段 |
| subject | string | 是 | 主题 | 来自于邮件头相应字段 |
| plain | string | 是 | 邮件正文 | 来自于邮件头相应字段 |
| attach\_md5 | string | 是 | 附件md5 | 来自于邮件头相应字段(属于mail\_attachment，可能有多个附件) |
| mime\_type | string | 是 | 附件mime\_type（字典库） | 来自于邮件头相应字段(属于mail\_attachment，可能有多个附件) |
| name | string | 是 | 附件名字 | 来自于邮件头相应字段(属于mail\_attachment，可能有多个附件) |
| bcc | string | 否 | 邮件密送人 | 来自于邮件头相应字段 |

#### META\_MAIL

message META\_MAIL

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

required string proto = 3 ; //应用协议

repeated uint32 sip = 4 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

optional string mid = 10; //邮件id号

required string time = 11; //邮件发送/接收时间

required string mail\_from = 12; //邮件发送人

required string to = 13; //邮件接收人

required string cc = 14; //邮件抄送人

required string subject = 15; //主题

required string plain = 16; //邮件正文

required string attach\_md5 = 17; //附件md5

required string mime\_type = 18; //附件mime\_type（字典库）

required string name = 19; //附件名字

optional string bcc = 21; //邮件密送人

}

### SSL

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| version | uint32 | 是 | 版本号 | SSL的版本号(0x0300)、TLS1.0(0x0301) 、TLS1.1(0x0302) |
| session\_id | string | 是 | 会话id | 本次会话的session\_id, server hello 报文中的session\_id |
| server\_name | string | 是 | 服务器名字 | SSL client\_hello报文中的扩展字段 |
| issuer\_name | string | 是 | 每封证书中的颁发者的名字 | 每封证书中的颁发者的名字(属于SSL\_CERT，可能有多个cert) |
| notbefore | string | 是 | 该证书的有效期的起始时间 | 该证书的有效期的起始时间(属于SSL\_CERT，可能有多个cert) |
| notafter | string | 是 | 该证书的有效期的终止时间 | 该证书的有效期的起始时间(属于SSL\_CERT，可能有多个cert) |
| public\_key | string | 是 | 该证书的公钥 | 该证书的公钥(属于SSL\_CERT，可能有多个cert) |
| user\_name | string | 是 | Sever端的证书的持有者 | 最上层证书的使用者，即本次SSL连接Sever端的证书的持有者 |

#### META\_SSL

message META\_SSL

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

repeated uint32 sip = 3 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

required uint32 version = 9 ; //版本号

required string session\_id = 10; //会话id

required string server\_name = 11; //服务器名字

required string issuer\_name = 12; //每封证书中的颁发者的名字

required string notbefore = 13; //该证书的有效期的起始时间

required string notafter = 14; //该证书的有效期的终止时间

required string public\_key = 15; //该证书的公钥

required string user\_name = 16; //Sever端的证书的持有者

}

### ORACLE/MSSQL/MYSQL/POSTGRESQL

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| proto | string | 是 | 数据库应用协议 | 根据报文中得到的协议字段转为可读的字符串  tds:填充mssql  tns:填充oracle  mysql:填充mysql  postgresql:填充postgresql |
| version | string | 是 | 协议版本 | 协议的版本号 |
| db\_type | string | 是 | 数据库类型 | mssql，oracle，mysql，postgresql四个值 |
| user | string | 是 | 用户 | 报文提取 |
| db\_name | string | 是 | 数据库 | 报文提取 |
| ret\_code | string | 是 | 数据库操作返回的状态信息 | sql语句执行的结果，Success或具体的失败信息 |
| sql\_info | string | 是 | 操作信息 | 报文提取的sql语句 |
| normal\_ret | string | 否 | 返回结果(原始信息) | 原始返回值，可能是ok，access等，后期要做归一化处理，例如都处理成ok |

#### META\_SQL

message META\_SQL

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

repeated uint32 sip = 3 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

required string proto = 9 ; //数据库应用协议

required string version = 10; //协议版本

required string db\_type = 11; //数据库类型

required string user = 12; //用户

required string db\_name = 13; //数据库

required string ret\_code = 14; //数据库操作返回的状态信息

required string sql\_info = 15; //操作信息

optional string normal\_ret = 16; //返回结果(原始信息)

}

### FTP

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 重复 | 线路信息 | 标识数据来源 |
| access\_time | uint64 | 是 | 日志生成时间 | 元数据生成时间 |
| sip | uint32 | 重复 | 源ip | 客户端IP |
| sport | int32 | 是 | 源端口 | 客户端应用端口 |
| dip | uint32 | 重复 | 目的ip | 服务端IP |
| dport | int32 | 是 | 目的端口 | 服务端应用端口 |
| user | string | 是 | 用户名 | FTP用户名 |
| op | uint32 | 是 | 操作命令 | 上传与下载 |
| ret | string | 是 | 操作结果 | 对应OP的返回结果，提取的原始数据 |

#### META\_FTP

message META\_FTP

{

repeated string line\_info = 1 ; //线路信息

required uint64 access\_time = 2 ; //日志生成时间

repeated uint32 sip = 4 ; //源ip

required int32 sport = 5 ; //源端口

repeated uint32 dip = 6 ; //目的ip

required int32 dport = 8 ; //目的端口

required string user = 10; //用户名

required uint32 op = 12; //操作命令

required string ret = 13; //操作结果

}

## 输出告警信息格式定义

输出告警日志保持和当前产品中的采集器到分析平台的传输字段和传输方式一致，使用zmq传输方式，消息封装成protobuf格式发送。告警携带的信息如下定义。

### 元数据威胁检测告警

#### 字段信息

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| line\_info | string | 是 | 线路号 |  |
| write\_date | uint64 | 是 | 告警入库时间 |  |
| sip | uint32 | 重复 | 源ip |  |
| sport | uint32 | 是 | 源端口 |  |
| dip | uint32 | 重复 | 目的ip |  |
| dport | uint32 | 是 | 目的端口 |  |
| tproto | uint32 | 是 | 传输层协议 | 详见7.13 |
| appid | uint32 | 否 | 应用层协议 | 详见7.14 |
| attack\_ip | uint32 | 是 | 攻击者IP |  |
| victim\_ip | uint32 | 否 | 受害者IP |  |
| attack\_type | uint32 | 否 | 威胁类型 | 详见7.1 |
| kill\_chain | string | 是 | 攻击链标签 | 详见7.12 |
| confidence | int32 | 是 | 置信度 | 详见7.9 |
| ioc\_desc | IOC\_DESC | 是 | 威胁情报告警附加描述字段 |  |
| rule\_desc | RULE\_DESC | 重复 | 特征检测告警附加描述字段 |  |
| public\_date | uint64 | 重复 | 规则或IOC发布时间 |  |
| vendor\_id | string | 否 | 厂商ID |  |
| device\_ip | uint32 | 是 | 设备IP |  |
| white\_list | uint32 | 否 | 白名单类型 | 详见7.11 |
| severity | uint32 | 否 | 威胁等级 | 详见7.2 |
| meta\_info | bytes | 否 | 原始元数据 |  |
| http\_payload | PCAP\_HTTP | 否 | PCAP HTTP攻击负载 |  |
| other\_payload | PCAP\_OTHER | 否 | PCAP Other攻击负载 |  |
| pcap\_path | string | 否 | pcap包路径 |  |

#### Protobuf消息结构

message RULE\_DOLOG

{

repeated string line\_info = 1;//线路号

optional uint64 write\_date = 2;//告警入库时间

repeated uint32 sip = 3;//源ip

repeated uint32 dip = 4;//目的ip

optional uint32 sport = 5;//源端口

optional uint32 dport = 6;//目的端口

optional uint32 tproto = 7;//传输层协议id

optional uint32 appid = 8;//应用类型id

repeated uint32 attack\_ip = 9;//攻击者ip

repeated uint32 victim\_ip = 10;//受害者ip

required uint32 attack\_type = 11;// 威胁类型

required string kill\_chain = 12;//攻击链

optional int32 confidence = 13;//置信度

optional IOC\_DESC ioc\_desc = 14;//IOC附加描述

optional RULE\_DESC rule\_desc = 15;//规则附加描述

optional uint64 public\_date = 16;//规则或IOC发布时间

optional string vendor\_id = 17;//厂商id

repeated uint32 device\_ip = 18;//设备ip

optional uint32 white\_list = 19;//白名单

optional uint32 severity = 20;// 威胁等级

required bytes meta\_info = 21; //原始元数据

optional PCAP\_HTTP http\_payload = 22; //PCAP HTTP攻击负载

optional PCAP\_OTHER other\_payload = 23; //PCAP Other攻击负载

optional string pcap\_path = 24; //pcap包路径

}

### PCAP检测HTTP协议攻击负载字段

#### 字段信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| method | string | 否 | http 请求方法 |  |
| host | string | 否 | http 请求host |  |
| uri | string | 否 | http uri |  |
| referrer | string | 否 | http referrer |  |
| agent | string | 否 | http useragent |  |
| cookie | string | 否 | http cookie |  |
| parameter | string | 否 | http请求参数 |  |
| req\_header | string | 否 | http请求头 |  |
| req\_body | string | 否 | http请求体 |  |
| rsp\_status | uint32 | 否 | http状态码 |  |
| rsp\_content\_length | uint32 | 否 | http响应体字节数 |  |
| rsp\_content\_type | string | 否 | http响应体类型 |  |
| rsp\_header | string | 否 | http响应头 |  |
| rsp\_body | string | 否 | http 响应体 |  |
| rsp\_body\_len | uint32 | 否 | http响应体字节数 |  |

#### Protobuf消息结构

**message PCAP\_HTTP**

**{**

optional string method =1; //http请求方法

optional string host =2; //http请求host

optional string uri =3; //http请求uri

optional string referrer =4; //http referrer头

optional string agent =5; //http aget头

optional string cookie =6; //http cookie头

optional string parameter =7; //http请求参数

optional string req\_header =8; //http请求头

optional string req\_body =9; //http请求体

optional uint32 rsp\_status =10; //http状态码

optional uint32 rsp\_content\_length =11; //http响应体字节数

optional string rsp\_content\_type =12; //http响应体类型

optional string rsp\_header =13; //http响应头

optional string rsp\_body =14; //http 响应体

optional uint32 rsp\_body\_len =15; //http响应体字节数

**}**

### PCAP检测其他协议（非HTTP）攻击负载字段

#### 字段信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| packet\_size | uint32 | 否 | 攻击负载字节数 |  |
| packet\_data | Bytes | 否 | 攻击负载数据 |  |

#### Protobuf消息结构

**message PCAP\_OTHER**

{

optional uint32 packet\_size =1; //攻击负载字节数

optional bytes packet\_data =2; //攻击负载数据

}

### 特征检测告警描述附加字段

#### 字段信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| rule\_id | uint32 | 否 | 规则id |  |
| rule\_name | string | 否 | 规则名称 |  |
| affected\_system | string | 否 | 受影响系统 |  |
| cve\_id | string | 否 | cve id |  |
| cnnvd\_id | string | 否 | cnnvd编号 |  |
| detail\_info | string | 否 | 详细信息 |  |
| solution | string | 否 | 解决方案 |  |
| webrules\_tag | string | 否 | web攻击类规则标签 | 0为非web攻击，1为web攻击 |
| attack\_result | uint32 | 否 | 攻击结果 | 详见7.4 |

#### Protobuf消息结构

message RULE\_DESC

{

required uint32 rule\_id =1; //规则id

required string rule\_name =2; //规则名称

optional string affected\_system =3; //受影响系统

optional string cve\_id =4; //cve id

optional string cnnvd\_id =5; //cnnvd编号

optional string detail\_info =6; //详细信息

optional string solution =7; //解决方案

optional string webrules\_tag =8; //web攻击类规则标签

optional uint32 attack\_result =9; //攻击结果

}

### 威胁情报检测告警描述附加字段

#### 字段信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| ioc\_type | string | 否 | IOC类型 |  |
| ioc\_value | string | 否 | IOC内容 |  |
| nid | string | 否 | IOC唯一标识 |  |
| targeted | bool | 否 | 定向攻击（APT事件） | True/False，True表示是定向攻击 |
| malicious\_family | string | 否 | 恶意家族名称 |  |
| tag | string | 否 | 恶意家族的tag |  |
| campaign | string | 否 | 恶意事件/团伙 |  |
| platform | string | 否 | 影响平台 |  |
| current\_status | string | 否 | 当前状态 | 活跃或非活跃 |
| desc | String | 否 | 告警描述 |  |

#### Protobuf消息结构

message IOC\_DESC

{

required string ioc\_type = 1;// IOC类型

required string ioc\_value = 2;// IOC内容

required string nid = 3;// IOC唯一标识

optional bool targeted = 4;// 定向攻击

required string malicious\_family = 5;// 恶意家族

repeated string tag = 6;// 恶意家族的tag

optional string campaign = 7;// 攻击事件/团伙

repeated string platform = 8;// 影响平台

optional string current\_status = 9;// 当前状态

optional string desc = 10;// 描述信息

}

# DNS流量对接

## 输入流量protobuf协议字段

### CHECK\_FIELDS字段定义

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| access\_time | uint64 | 否 | 发起（响应）请求的时间戳 |  |
| domain | string | 是 | 查询的域名 |  |
| type | uint32 | 是 | Dns包类型，见包类型对照表 | 详见7.8 |
| qcode | uint32 | 是 | dns请求消息中请求类型，见请求类型对照表 | 详见7.6 |
| acode | uint32 | 否 | dns响应消息中响应码，见响应码对照表 | 详见7.7 |
| tid | uint32 | 否 | 请求的transaction id号 |  |
| pktSize | uint32 | 否 | 包总长度 |  |
| answer | ResourceRecord | 重复 | answer资源信息 |  |
| authority | ResourceRecord | 重复 | authority资源信息 |  |
| additional | ResourceRecord | 重复 | additional资源信息 |  |

### DNS字段定义

| **字段** | **数据类型** | **是否必选** | **注释** | **备注** |
| --- | --- | --- | --- | --- |
| sip | uint32 | 重复 | 原ip地址 |  |
| dip | uint32 | 重复 | 目标ip地址 |  |
| sport | uint32 | 是 | 原端口 |  |
| dport | uint32 | 是 | 目标端口 |  |
| sender\_ip | uint32 | 是 | 发数据方的IP地址 |  |
| CHECK\_FIELDS | CheckFileds | 否 | 要检测的dns信息 |  |
| dns\_pkg | bytes | 否 | 原始dns信息 |  |

### DNS资源记录ResourceRecord各字段定义

| **字段** | **数据类型** | **是否必选** | **注释** | | **备注** |
| --- | --- | --- | --- | --- | --- |
| rtype | uint32 | 是 | 资源类型，含义同请求类型 | |  |
| rdata | string | 是 | 可用数据部分 | | Rtype为其他时此值可为空字符串 |
| Rtype(记录类型) | rdata |
| 1(A) | IP地址的字符串形式 |
| 2(NS) | Name server |
| 5(CNAME) | Cname字符串 |
| 12(PTR) | Ptr字符串 |
| 15(MX) | Mail exchange字符串 |
| 16(TXT) | TXT字符串 |
| 28(AAAA) | Ipv6地址 |
| rname | string | 否 | 资源的name，一般为请求的域名 | |  |
| ttl | uint32 | 否 | 资源生存时间ttl | |  |
| Ns | string | 否 | 字符串，内容为Pri Name Server | | Rtype为6时这些选项需要包含内容 |
| Mbox | string | 否 | 字符串，内容为Responsible Mail Box | |
| Serial | uint32 | 否 | 整数，Serial号 | |
| Refresh | uint32 | 否 | 整数，Refresh间隔 | |
| Retry | uint32 | 否 | 整数，Retry间隔 | |
| Expire | uint32 | 否 | 整数，过期ttl | |  |
| Minttl | uint32 | 否 | 整数，最小ttl | |  |

## 输入流量protobuf协议字段

syntax = "proto2";

message ResourceRecord{

optional string rname = 1; // name，一般与请求的name相同

required uint32 rtype = 2; //资源类型，其范围同请求类型

required string rdata = 3; //数据部分，比如，如果为A响应，则为IP地址；如果类型为MX，则此处为 Mail exchange

optional uint32 ttl = 4; // TTL

optional string Ns = 5; // Primer Name Server

optional string Mbox = 6; // Responsible Mail Box

optional uint32 Serial = 7; // Serial number

optional uint32 Refresh =8; // Refresh interval

optional uint32 Retry = 9; // retry interval

optional uint32 Expire = 10; // expire limit

optional uint32 Minttl = 11; //minimum ttl

}

message CHECK\_FIELDS

{

optional uint64 access\_time = 1; //发起（响应）请求时间的时间戳

required string qname = 2; //请求的dns域名

required uint32 type = 3; //包类型,见对照表

required uint32 qcode = 4; //请求码（请求类型）,见对照表

optional uint32 acode = 5; //响应码,见对照表

optional uint32 tid = 6; //dns 事务ID

optional uint32 pktSize = 7; //dns 请求或响应包的长度

repeated ResourceRecord answer = 8; //answer信息，可以有多条

repeated ResourceRecord authority = 9; //authority 信息，可以有多条

repeated ResourceRecord additional =10; //additional 信息，可以有多条

}

message DNS

{

repeated uint32 sip = 1; //源IP地址

repeated uint32 dip = 2; //目标IP地址

required uint32 sport = 3; //源端口

required uint32 dport = 4; //目标端口

repeated uint32 sender\_ip = 5; //发送数据方ip地址

optional CHECK\_FIELDS check\_fields = 6; //要检测的dns信息

optional bytes dns\_pkg = 7; //dns流量负载，也即UDP负载部分,当check\_fields为空时填写此值

}

## DNS告警输出字段

### 字段定义

| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| --- | --- | --- | --- | --- |
| log\_time | uint64 | 是 | 时间 |  |
| ip | uint32 | 重复 | 受害IP |  |
| domain | string | 是 | 请求的域名 |  |
| threat\_name | string | 否 | 威胁的名字 |  |
| severity | uint32 | 是 | 告警级别 | 详见7.2 |
| alert\_type | uint32 | 是 | 告警类型 | 详见7.5 |
| attack\_type | uint32 | 是 | 威胁类型 | 详见7.1 |
| qcode | uint32 | 是 | 请求码 | 详见7.6 |
| confidence | int32 | 是 | 置信度 | 详见7.9 |
| white\_list\_type | uint32 | 是 | 白名单类型 | 详见9.11 |
| abnormal\_dns\_result | uint32 | 是 | 异常dns检测结果 |  |
| vendor\_id | string | 是 | 厂商id |  |
| device\_ip | uint32 | 是 | 设备IP |  |
| meta\_info | bytes | 是 | 原始元数据 |  |

### Protobuf消息结构

message DNS\_DOLOG

{

required uint64 log\_time = 1; //返回长整形，可精确到毫秒

repeated uint32 ip = 2; //受害IP，兼容IPV4和IPV6

required string domain = 3; //请求的域名

optional string threat\_name = 4; //威胁的名字

required int32 threat\_level = 5; //告警级别

required uint32 alert\_type = 6; //告警类型

required uint32 attack\_type = 7; //威胁类型

required uint32 qcode = 8; //请求类型

required int32 confidence = 9; //置信度

required int32 white\_list\_type =10; //白名单类型

required int32 abnormal\_dns\_result =11; //异常dns检测结果

optional string vendor\_id = 12;//厂商id

repeated uint32 device\_ip = 13;//设备ip

required bytes meta\_info =14; //原始元数据

}

# 沙箱文件对接

## 输入文件字段定义

输入接口使用ICE格式，沙箱这边提供服务端，接口如下：

module Sandbox {

interface Server {

void Receive(string s);

void Send(string s);

}

}

## 沙箱告警输出字段定义

### 字段定义

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段 | 数据类型 | 是否必选 | 注释 | 备注 |
| task\_info | string | 是 | 任务号 |  |
| line\_info | string | 是 | 线路号 |  |
| write\_date | uint64 | 是 | 告警入库时间 |  |
| file\_name | string | 是 | 文件名 |  |
| file\_size | uint32 | 是 | 文件大小 |  |
| file\_md5 | string | 是 | 文件md5 |  |
| file\_type | string | 是 | 文件类型 |  |
| attack\_type | string | 否 | 威胁类型 | 详见7.1 |
| file\_path | string | 是 | 告警文件路径 |  |
| severity | uint32 | 是 | 告警文件威胁级别 | 详见7.2 |
| vendor\_id | string | 是 | 厂商id |  |
| device\_ip | uint32 | 是 | 设备IP |  |
| meta\_info | bytes | 是 | 原始元数据 | 沙箱报告json格式内容 |

### Protobuf消息结构

message SANDBOX\_DOLOG {

required string task\_info = 1; // 任务号

repeated string line\_info = 2;//线路号

optional uint64 write\_date = 3;//告警入库时间

required string file\_name = 4;//文件名

required int32 file\_size = 5;// 文件大小

required string file\_md5 = 6;// 文件md5

required string file\_type = 7; // 文件类型

required string file\_path = 8; //告警文件路径

required uint32 severity = 9;// 告警文件威胁级别

required uint32 attack\_type = 10; //威胁类型

optional string vendor\_id = 11;//厂商id

repeated uint32 device\_ip = 12;//设备ip

required bytes meta\_info =13; //沙箱报告json格式内容（转bytes）

}

# 上报状态到设备管控接口

暂定为写状态到本地文件，格式自定义。每隔30秒把状态追加到文件中。

状态文件一天一个，名字自定义，文件保留为30天。

## 日志文件名格式定义

|  |  |  |
| --- | --- | --- |
| 日志文件名称 | 举例 | 说明 |
| xxxx\_时间.log | Device\_YW\_20190806.log | Device：设备类型，  YW：运维，  时间：年月日 |

## 日志格式定义

|  |  |  |
| --- | --- | --- |
| 日志格式 | 举例 | 说明 |
| 时间 StartTime=时间，Runtime=数值 | 2019-08-05 23:48:47 StartTime=2019-08-05 23:48:47, RunTime=24… | StartTime:启动时间，  RunTime: 运行时间 |

# 字典定义

## 威胁类型（attack\_type）字典定义

|  |  |  |
| --- | --- | --- |
| 类型编号 | 一级类型 | 二级类型 |
| 0x0100 | APT事件 |  |
| 0x0101 |  | APT事件 |
| 0x0102 |  | 用户自定义 |
| 0x0103 |  | 其他APT事件 |
| 0x0200 | 攻击利用 |  |
| 0x0201 |  | SQL注入 |
| 0x0202 |  | URL跳转 |
| 0x0203 |  | 代码执行 |
| 0x0204 |  | 非授权访问/权限绕过 |
| 0x0205 |  | 跨站脚本攻击（XSS） |
| 0x0206 |  | 跨站请求伪造（CSRF） |
| 0x0207 |  | 逻辑/设计错误 |
| 0x0208 |  | 命令执行 |
| 0x0209 |  | 目录遍历 |
| 0x020a |  | 配置不当/错误 |
| 0x020b |  | 弱口令 |
| 0x020c |  | 文件包含 |
| 0x020d |  | 文件读取 |
| 0x020e |  | 文件上传 |
| 0x020f |  | 文件下载 |
| 0x0210 |  | 文件写入 |
| 0x0211 |  | 溢出攻击 |
| 0x0212 |  | 信息泄露 |
| 0x0213 |  | 浏览器劫持 |
| 0x0214 |  | 暴力猜解 |
| 0x0215 |  | 网络钓鱼 |
| 0x0216 |  | webshell上传 |
| 0x0217 |  | 用户自定义 |
| 0x0218 |  | 其他攻击利用 |
| 0x0300 | 恶意软件 |  |
| 0x0301 |  | 后门程序 |
| 0x0302 |  | 僵尸网络 |
| 0x0303 |  | 特洛伊木马 |
| 0x0304 |  | 电脑病毒 |
| 0x0305 |  | 间谍软件 |
| 0x0306 |  | 恶意广告 |
| 0x0307 |  | 远控木马 |
| 0x0308 |  | 键盘记录 |
| 0x0309 |  | 窃密木马 |
| 0x030a |  | 网络蠕虫 |
| 0x030b |  | 勒索软件 |
| 0x030c |  | 黑市工具 |
| 0x030d |  | 流氓推广 |
| 0x030e |  | 用户自定义 |
| 0x030f |  | 其他恶意软件 |
| 0x0400 | 拒绝服务 |  |
| 0x0401 |  | 用户自定义 |
| 0x0402 |  | 其他拒绝服务 |
| 0x0500 | 侦察 |  |
| 0x0501 |  | 端口扫描 |
| 0x0502 |  | 网络扫描 |
| 0x0503 |  | 用户自定义 |
| 0x0504 |  | 其他侦察 |

## 危害级别(severity)字典定义

|  |  |  |
| --- | --- | --- |
| 危害级别 | 取值 | 注释 |
| 危急 | 4 |  |
| 高危 | 3 |  |
| 中危 | 2 |  |
| 低危 | 1 |  |

## 白名单类型（white\_list）字典定义

|  |  |  |
| --- | --- | --- |
| 白名单类型 | 取值 | 注释 |
| 网络爬虫 | 2 |  |
| 域名服务器 | 3 |  |
| CDN | 1 |  |
| 不是白名单 | 0 |  |
| 其他 | 4 |  |

## 攻击结果（attack\_result）字典定义

|  |  |  |
| --- | --- | --- |
| 攻击结果 | 取值 | 备注 |
| 0 | 企图 |  |
| 1 | 成功 |  |
| 2 | 失陷 |  |
| 3 | 失败 |  |

## 告警类型（alter\_type）字典定义：

|  |  |  |
| --- | --- | --- |
| 值 | 类型 | 备注 |
| 1 | 威胁情报 |  |
| 2 | DGA规则 |  |
| 3 | FastFlux |  |
| 4 | DNS重绑定 |  |
| 5 | DNS反射放大攻击 |  |
| 6 | IDN可疑域名 |  |
| 7 | Sinkhole |  |
| 8 | 环路地址 |  |
| 9 | 可疑动态域名 |  |
| 10 | 心跳域名 |  |
| 11 | DNS隐蔽通道 |  |
| 12 | 黑名单 |  |

## Dns 请求码（qcode）字典定义:

|  |  |  |
| --- | --- | --- |
| 请求码 | 类型 | 备注 |
| 1 | A |  |
| 2 | NS |  |
| 5 | CNAME |  |
| 6 | SOA |  |
| 12 | PTR |  |
| 15 | MX |  |
| 16 | TXT |  |
| 28 | AAAA |  |
| 255 | ANY |  |

## Dns响应码（acode）字典定义：

|  |  |  |
| --- | --- | --- |
| 响应码 | 含义 | 备注 |
| 0 | NOERROR |  |
| 1 | FORMERR |  |
| 2 | SERVFAIL |  |
| 3 | NXDOMAIN |  |
| 4 | NOTIMP |  |
| 5 | REFUSED |  |
| 6 | YXDOMAIN |  |
| 7 | YXRRSET |  |
| 8 | NXRRSET |  |
| 9 | NOTAUTH |  |
| 10 | NOTZONE |  |

## Dns包类型（type）字典定义：

|  |  |  |
| --- | --- | --- |
| 值 | 类型 | 备注 |
| 0 | 请求包 |  |
| 1 | 响应包 |  |

## 置信度（confidence）字典定义：

|  |  |  |
| --- | --- | --- |
| 值 | 类型 | 备注 |
| 1 | 低 |  |
| 2 | 中 |  |
| 3 | 高 |  |

## 消息类型（message\_type）字典定义

|  |  |  |
| --- | --- | --- |
| 值 | 类型 | 备注 |
| 1 | 规则或ioc告警 |  |
| 2 | dns告警 |  |
| 3 | 沙箱告警 |  |

## 白名单类型（white\_list）字典定义

|  |  |  |
| --- | --- | --- |
| 值 | 类型 | 备注 |
| 0 | 不是白名单 |  |
| 1 | cdn |  |
| 2 | 网络爬虫 |  |
| 3 | 域名服务器 |  |
| 4 | 其它 |  |

枚举类型对应值，.0为不是白名单.1为CDN，2为网络爬虫，3为域名服务器，4为其它

## 攻击链（kill\_chain,攻击阶段）字典定义

|  |  |  |  |
| --- | --- | --- | --- |
| 攻击链标签 | | | |
| 攻击链编号 | 一级标签 | 二级标签 | 三级标签 |
| 0x01000000 | 侦察 |  |  |
| 0x01010000 |  | 端口扫描 |  |
| 0x01020000 |  | 信息泄露 |  |
| 0x01030000 |  | ip扫描 |  |
| 0x01040000 |  | 子域名收集 |  |
| 0x01050000 |  | 网络扫描 |  |
| 0x02000000 | 入侵 |  |  |
| 0x02010000 |  | 漏洞探测 |  |
| 0x02020000 |  | 漏洞利用 |  |
| 0x02030000 |  | 拒绝服务 |  |
| 0x02040000 |  | 暴力破解 |  |
| 0x02050000 |  | 高危操作 |  |
| 0x02050100 |  |  | 数据库操作 |
| 0x02050200 |  |  | 弱口令成功登录 |
| 0x02050300 |  |  | 运行可疑文件 |
| 0x02050400 |  |  | 主动下载恶意程序 |
| 0x02060000 |  | 网络钓鱼 |  |
| 0x02070000 |  | 水坑攻击 |  |
| 0x02080000 |  | 物理攻击 |  |
| 0x02090000 |  | 远程桌面被控 |  |
| 0x02090000 |  | 样本投递 |  |
| 0x03000000 | 命令控制 |  |  |
| 0x03010000 |  | 主机受控 |  |
| 0x03020000 |  | 黑客工具上传 |  |
| 0x03030000 |  | 服务器中转行为 |  |
| 0x03040000 |  | 提权 |  |
| 0x03050000 |  | 关闭杀毒软件 |  |
| 0x03060000 |  | 主机信息获取 |  |
| 0x03070000 |  | 恶意组件下载 |  |
| 0x03080000 |  | 配置信息上报 |  |
| 0x03090000 |  | 混合功能控制 |  |
| 0x030a0000 |  | 命令控制服务器连接 |  |
| 0x030b0000 |  | 自身保护 |  |
| 0x030c0000 |  | 关闭防火墙 |  |
| 0x030d0000 |  | 恶意文件写入 |  |
| 0x030e0000 |  | 运行风险程序 |  |
| 0x04000000 | 横向渗透 |  |  |
| 0x04010000 |  | 内网侦察 |  |
| 0x04020000 |  | 嗅探攻击 |  |
| 0x04030000 |  | 内网漏洞探测 |  |
| 0x04040000 |  | 内网漏洞利用 |  |
| 0x04050000 |  | 网络蠕虫 |  |
| 0x05000000 | 目的执行 |  |  |
| 0x05010000 |  | 文件下载 |  |
| 0x05020000 |  | 拖库行为 |  |
| 0x05030000 |  | 数据服务器连接 |  |
| 0x05040000 |  | 资源消耗 |  |
| 0x05050000 |  | 加密勒索 |  |
| 0x05060000 |  | 挖矿行为 |  |
| 0x05070000 |  | 数据外泄 |  |
| 0x05080000 |  | 信息收集 |  |
| 0x05090000 |  | 键盘记录 |  |
| 0x06000000 | 痕迹清理 |  |  |
| 0x06010000 |  | 文件删除 |  |
| 0x06010100 |  |  | 后门删除 |
| 0x06020000 |  | 关闭攻击服务 |  |
| 0x06030000 |  | 清除日志 |  |
| 0x06040000 |  | 凭据清理 |  |

## 传输层协议（tproto）字典定义

|  |  |  |
| --- | --- | --- |
| 协议名称 | 协议编号 | 备注 |
| TCP | 6 |  |
| UDP | 17 |  |

## 应用层协议（appid）字典定义

|  |  |  |
| --- | --- | --- |
| 应用层协议编号 | 应用层协议名称 | 备注 |
| 3 | netbios-ss |  |
| 4 | smtp |  |
| 7 | h245 |  |
| 12 | nfs |  |
| 14 | pptp |  |
| 19 | ssl |  |
| 21 | db2 |  |
| 35 | imap |  |
| 36 | t3 |  |
| 37 | wins |  |
| 41 | t\_120 |  |
| 43 | cvs |  |
| 45 | ftp-data |  |
| 46 | ftp |  |
| 49 | radius |  |
| 50 | dhcp |  |
| 64 | telnet |  |
| 70 | pop3 |  |
| 72 | ldap |  |
| 76 | teamviewer |  |
| 77 | http |  |
| 83 | oracle |  |
| 88 | msrpc |  |
| 97 | mssql-db |  |
| 98 | postgres |  |
| 104 | nntp |  |
| 106 | irc |  |
| 108 | kerberos |  |
| 110 | ntp |  |
| 111 | rpc |  |
| 120 | sybase |  |
| 121 | mongodb |  |
| 122 | vnc |  |
| 123 | netbios-ns |  |
| 130 | dns |  |
| 140 | smb |  |
| 143 | snmp |  |
| 144 | mysql |  |
| 145 | cotp |  |
| 146 | ssh |  |
| 147 | tftp |  |
| 148 | icq |  |
| 149 | rtsp |  |