

基金项目:重庆市气象部门业务技术攻关资助项目(YWJSGG-202111)

表 2 返回数据结构表

序号	字节数	数值	说明
0	4	“ZBXD”	首部字符串
4	1	0x01	
5	8	n	监控值字符串长度,低位在前
13	n	xxx	监控值字符串

Zabbix 监控过程有两种模式,分别是被动模式(默认)和主动模式,它们都是相对 Zabbix Agent 来说的。主动模式主要是为了减轻 Zabbix Server 的负荷。

2.2 被动模式数据交互过程

这个过程对于 Zabbix Agent 来说是被动地接收 Zabbix Server 的数据并作出响应。Zabbix Server 按照各监控数据项的时间原则轮流循环地发送包含 ITEM 集合的数据,如图 2 所示。

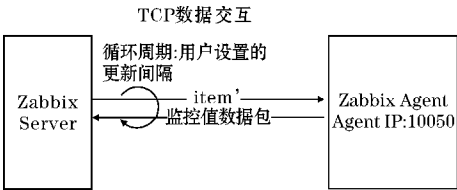


图 2 被动模式数据交互图

需要注意的是,这里的轮流发送并不是在一个 TCP 连接里面,而是每一个监控项都是由一个独立的 TCP 连接完成,如此反复。

2.3 主动模式数据交互过程

在 Zabbix Agent 配置文件中设置 ServerActive = ip (Zabbix Server IP) 以后,Zabbix 就启动主动模式。Zabbix Agent 主动向 Zabbix Server 发送 TCP 连接并分别发送请求报文: { " request ":" active checks ", " host ":" hostname ", " port ": 10051 }。Zabbix Server 将返回报文: ZBX\_NOTSUPPORTED. Invalid. item key format、{ " response ":" failed ", " info ":" host [ hostname ] not found " } 或 { " response ":" success ", " data ": [ { " key ":" item 键值 ", " delay ": 120, " lastlogsize ": 0, " mtime ": 0 }, ... ] },前者表示,没有设置主动模式监控项、没有匹配到主机;后者即为本主机在 Zabbix Server 中设置的主动模式监控项列表。无论返回报文如何,Zabbix Agent 都会每 2 min 刷新一次请求,只要有主动模式监控项,它就会依照各 delay 值依次循环将各包含监控值的数据包主动发送给 Zabbix Server,如图 3 所示。

其中,请求数据包、监控项列表数据包和监控值数据包的封装都与表 1 和表 2 结构相同。

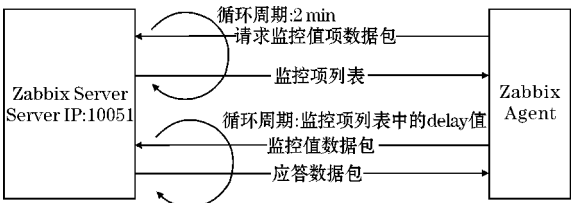


图 3 主动模式数据交互图

2.4 非 Zabbix Agent 设备的监控

Zabbix 除使用 Zabbix Agent 实现监控外,还可以通过一些协议来实现对非 Zabbix Agent 环境设备的监控,最常用的就是 SNMP 协议<sup>[6-7]</sup>。它是一种网络管理标准协议,其目标是管理互联网 Internet 上众多厂家生产的软硬件平台,其数据交互过程如图 4 所示。

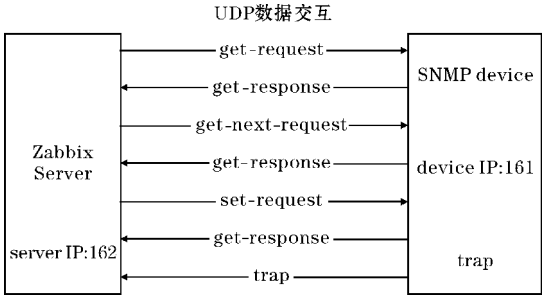


图 4 SNMP 协议数据交互过程图

其中,数据包的封装、具体过程等可参考文献[8]。

3 气象业务集中监控的开发

当前气象信息中心所关注的监控对象很多,Zabbix 现有数据采集功能也几乎可以涵盖所有<sup>[9]</sup>,但都是针对计算机主机和服务器的。非 Zabbix Agent 环境的数据采集<sup>[10]</sup>固然可以用 SNMP 协议解决,但是目前气象业务中还有很多监控对象没有 SNMP 协议接口,或者以其他异构形式提供数据,加之 Zabbix 脚本处理能力和灵活性限制,这就需要开发设计能够适合气象业务自己的 Agent 和 SNMP 协议代理程序,以满足全面集约化监控的要求。

3.1 开发目标

应实现以下目标:(1)可以实现对所有非 Zabbix Agent 数据的封装与传输。(2)支持自定义监控项的键值和 OID。(3)Zabbix Web 不需要任何新增功能,常规操作即可以和新增数据的监控目标对接。

3.2 核心功能的设计和实现

由于非 Zabbix Agent 数据采集和控制都是通过硬件接口(如 RS232, 485, GPRS, PCI 卡等)或 IP 方式实现的<sup>[11]</sup>,种类繁多。默认采集数据(监控值)已存放到指定的缓冲区。这里重点介绍数据通信核心部分,即

Agent 和 SNMP 协议的缓冲区中监控值的封装与传输。

### 3.2.1 收发缓冲区的定义

从对 Zabbix 交互数据的分析得知,无论是主动模式还是被动模式、无论是请求还是返回,其数据包的前 13 字节都是相对固定的,由此定义 如下发送缓冲区:

```
byte[] smes = new byte[1024]; //发送缓冲区
smes[0] = 0x5a; // "Z" //首部字符串赋值
smes[1] = 0x42; // "B"
smes[2] = 0x58; // "X"
smes[3] = 0x44; // "D"
smes[4] = 0x01; // "01" //间隔符
```

接收缓冲区则定义为

```
byte[] rmes = new byte[1024]; //接收缓冲区
```

### 3.2.2 被动模式的实现

被动模式的主要代码如下:

```
while(true) {
//在本地 IP:10050 端口创建 TCP 监听
ServerSocket server = new ServerSocket(10050);
Socket s = server.accept();
InputStream in = s.getInputStream();
int len = in.read( rmes ); //读 Zabbix Server 请求数据
```

String keystr=new String(rmes, 13, len-13); //解析出键值

解析出的键值和自定义键值列表比较,如果相等就把对应的采集数据缓冲区的监控值赋给 str

```
int length=str.length(); //计算监控值字符串长度
smes[5] = (byte)(length & 0xFF); //放到发送缓冲区,低位在前处理
```

```
smes[6] = (byte)((length >> 8) & 0xFF);
smes[7] = (byte)((length >> 16) & 0xFF);
smes[8] = (byte)((length >> 24) & 0xFF);
//监控值字符串追加到 smes[13],至此返回数据包封装完毕。
```

```
OutputStream out = s.getOutputStream();
out.write(smes, 0, length+13); //将数据包发送给 Zabbix Server
in.close();
out.close();
s.close();
server.close();
}
```

### 3.2.3 主动模式的实现

主动模式的实现主要由两部分组成,代码如下。

(i) 获得主动模式监控项列表部分,循环周期为 2 m  
 { //向 Zabbix Server IP:10051 发起 TCP 连接  
 Socket c=new Socket(IP,10050);

```
InetAddress addr=InetAddress.getLocalHost();
hostnamestr=addr.getHostName() //获得主机名
计算请求报文:{"request":"active checks",
"host":"hostname",
"port":10051
}
```

长度("hostname"需换成 hostnamestr),放到发送缓冲区 smes[5],低位在前处理,并将报文追加到 smes[13]

```
OutputStream out = c.getOutputStream();
out.write(smes, 0, length + 13); //发送请求数据包
out.flush();
InputStream in = c.getInputStream();
len = in.read(rmes);
String keylist = new String(rmes, 13, len - 13);
解析出键值列表的各键值和对应的 delay 等
out.close();
in.close();
c.close();
}
```

(ii) 提交监控值部分,循环周期为 delay

这部分程序功能是把键值列表的各键值对应的采集数据缓冲区的监控值,按照自己对应的 delay 循环提交给 Zabbix Server。包含监控值的报文如下。

```
{"request": "agent data",
"session": "4d10c7d271f34705281f90b8301905be",
"data": [
{"host": "MS-NVNFPPJMPXZP",
"key": "kkkkk", //键值
"value": "xxxx", //采集数据缓冲区的监控值
"id": 36,
"clock": 1612672200,
"ns": 706231500}
],
"clock": 1612672201,
"ns": 432382200}
```

报文中设置了身份认证机制,限于篇幅不再详述。数据封装、传输与(ii)相同。

### 3.2.4 SNMP 协议代理的实现

为简化开发流程,使用 SNMP4J 框架<sup>[12]</sup>实现,主要代码如下。

```
mSS=new org.snmp4j.transport.DefaultUdpTransportMapping(
new org.snmp4j.smi.UdpAddress(java.net.InetAddress.getByName(本地IP), 161));
mSNMP = new org.snmp4j.Snmp(mSS);
mSNMP.addCommandResponder(this);
mSS.listen(); //161 端口 UDP 监听
```



```
try {  
    //SNMP 协议提交监控值  
    vPDU.setType(PDU.RESPONSE);  
    vPDU.set(0, new VariableBinding(vPDU.get(0).  
getOid(), new Integer32( xxxx))); //int  
    //vPDU.set(0, new VariableBinding(vPDU.get  
(0).getOid(), new OctetString("ssss"))); //String  
    aEvent.getMessageDispatcher().returnResponsePdu()  
    aEvent.getMessageProcessingModel(),  
    aEvent.getSecurityModel(),  
    aEvent.getSecurityName(),  
    aEvent.getSecurityLevel(), vPDU,
```

```
aEvent.getMaxSizeResponsePDU(), ref,  
statusInformation);  
}  
catch(org.snmp4j.MessageExceptionvException) {  
}
```

可以自定义 OID 来实现多种数据的代理传输, Zabbix Server 根据 OID 选取数据。

3.3 采集数据结果对比和实际应用

用 Zabbix Agent 采集一个主机的进程数为例,与使用本文设计的 Agent 和 SNMP 协议代理传输(用 Zabbix API 采集)同样的数据做比较,如图 5 所示。

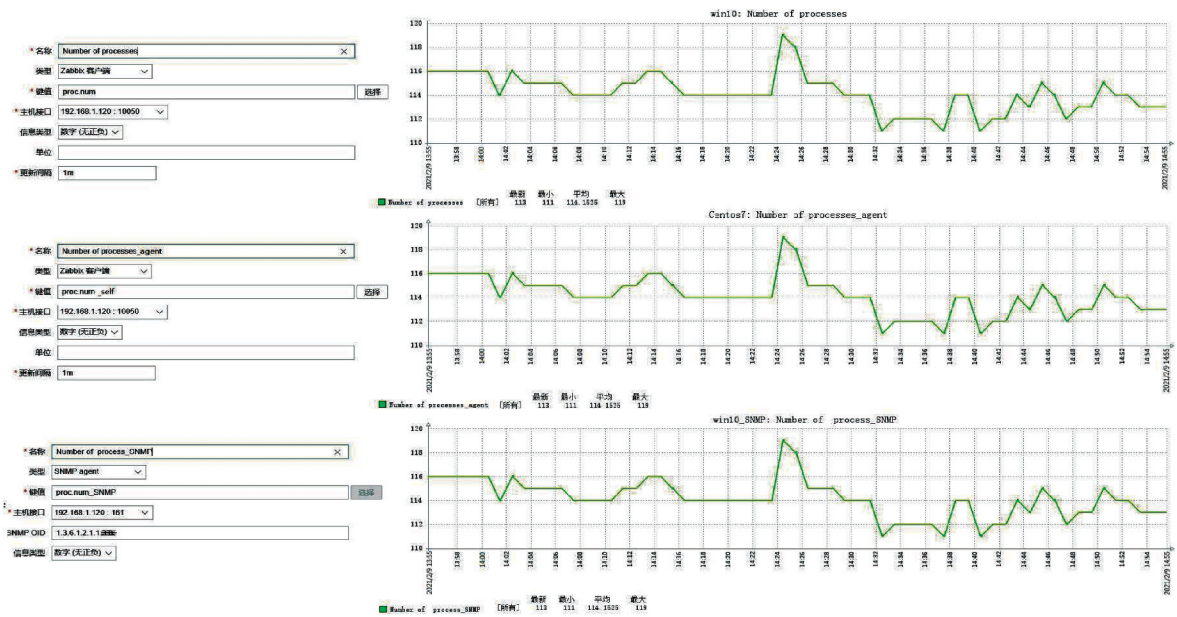


图5 传输数据结果对比图

可以看出没有延迟、丢包等错误,结果是完全相同的,其他类型数据亦是如此。据此,本设计用于气象数据的采集传输是完全可以满足要求。

图 6 为采用本文设计的 Agent 和 SNMP 协议代理程序采集传输的实际应用实例—温湿度数据曲线。

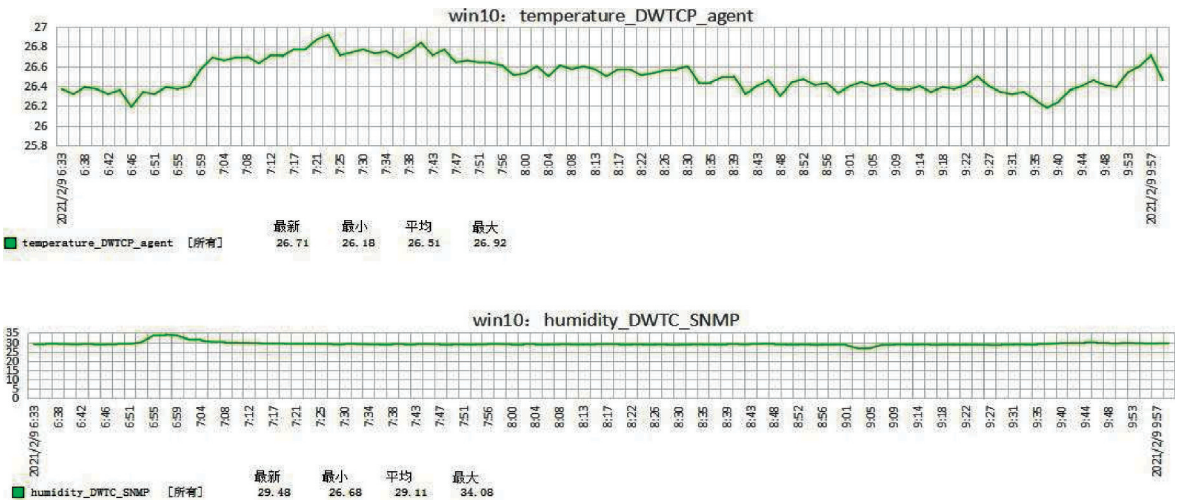


图6 温湿度曲线图

## 4 结束语

通过对 Zabbix Server 和 Zabbix Agent、SNMP 之间数据交互的分析研究,解决了气象综合业务运维中的非 Zabbix Agent 数据采集传输问题,实现了监控数据种类的全覆盖。下一步的工作会将故障定位作为重点,努力实现气象综合业务运维的全自动化。

## 参考文献:

- [1] 杨立苑,张玮. 省级 CIMISS 常用气象数据灾备服务系统研究[J]. 气象环境与科学,2020,43(4):134-139.
- [2] Vladishev A. Zabbix: an enterprise-class open source distributed monitoring solution[EB/OL]. <http://www.zabbix.com/>[Accessed: 21-Jul-2010],2010-07-21.
- [3] Zhao Zhe,Tan Haibo,ZhaoHe,et al. Network monitoring system based on Zabbix[J]. Computer Technology and Development,2018,28(1):144-149.
- [4] 杨杰,曾凌波,彭运勇,等. 面向大规模集群的自动化监控系统[J]. 计算机工程与科学,2020,42(10):1801-1806.
- [5] 梁业佳. 基于 Zabbix 的铁路运输动态监控系统

- 设计[J]. 科技通报,2020,36(8):35-39.
- [6] Case J,Fedor M,Schoffstall M,et al. Simple network management protocol(SNMP):RFC1157-A[S/OL]. <http://ip-doc.com/rfc/rfc1157>. 2020-02-19.
- [7] 张春强. 深入理解 Net-SNMP[M]. 北京:清华大学出版社,2015.
- [8] 徐展琦,郭彦涛,丁喆,等. SNMP 协议用于卫星网管的改进与增强[J]. 无线通信技术,2015,41(6):1-5.
- [9] Sun Jiancheng. Design and implementation of elementary school informatization system based on Dingtalk and Wechat platform[D]. Chengdu:University of Electronic Science and technology of China,2019.
- [10] Zhi Lianyi. Research and implementation of cluster monitoring data collection and analysis system in cloud environment[D]. Nanjing:Southeast University,2016.
- [11] 杨柳庆,张勇,孙丽慧,等. 基于 TDD 协议的单载波频带靶机擦控系统[J]. 宇航计测技术,2020,40(6):37-42.
- [12] 周长红,符腾飞,靖丹. 一种可验证菊花链环网的软件设计方法[C]. 第十六届中国航空测控技术年会论文集,2019.

## Research and Application of Zabbix in Centralized Monitoring of Meteorological Services

LI Wenzhao, ZHAO Siliang, ZHAO Fang

(Chongqing Meteorological Information and Technology Support Center,Chongqing 401147,China)

**Abstract:** In response to the demand of intensive daily operation monitoring and maintenance management of the information center, according to the analysis and research of Zabbix interactive data, the Zabbix monitoring process is introduced in detail. Combining with the characteristics of information and data in the operation and maintenance of integrated meteorological services, a non-Zabbix Agent data collection and transmission method is proposed, and the main implementation program is designed. Application practice indicates that the program can satisfy the daily requirements of actual monitoring and achieve full coverage of monitoring data types. It is of practical significance to improve the level of centralized monitoring of meteorological services.

**Keywords:** monitoring; Zabbix Server; Zabbix Agent; Non-Zabbix Agent; SNMP