# 运维平台设计

## 术语

SNMP：简单网络管理协议。

SYSLOG：系统日志协议

## 设计约束

1. SNMP运维平台需支持Mib库扩展
2. 服务器基本信息监控项目受限于第三方软件net-snmp，但可以使用子代理进行扩展

## 结构设计

1. **整体结构**



浅蓝色为各组件，深蓝色为第三方软件，白色为已开发类，绿色是需要开发的。

## 功能设计

1. **SNMP功能**
2. Trap功能启/停用
3. Trap参数配置，包括使用协议、服务器地址、服务器端口、字符编码、SNMP协议、消息类型、社区名、认证等级、用户名、摘要算法、密码、加密算法、加密密码、上下文名称
4. 服务器基本信息监控启用/停用
5. 服务器基本信息监控参数配置，包括允许访问源、允许访问协议、社区名、认证等级、用户名、摘要算法、密码、加密算法、加密密码、上下文名称
6. **SYSLOG功能**
7. 功能启/停用
8. 参数配置，包括使用协议、服务器地址、服务器端口、字符编码

## 业务流程

1. **功能启/禁用和配置更新**



1. **日志处理**



## 数据库设计

使用已有表tbl\_sysconfig

1. **SNMP预置数据**

#### Trap功能预置数据

|  |  |  |
| --- | --- | --- |
| 参数名称 | s\_key | s\_value |
| 是否启用 | snmp\_enable | false |
| 服务器地址 | snmp\_host | 127.0.0.1 |
| 服务器端口 | snmp\_port | 162 |
| 字符编码 | snmp\_charset | utf8 |
| 使用协议 | snmp\_protocol | 2 |
| Trap类型 | snmp\_type | 1 |
| 社区名称 | snmp\_community | public |
| 认证等级 | snmp\_sec\_level | 1 |
| 用户名 | snmp\_username |  |
| 摘要算法 | snmp\_digest |  |
| 密码 | snmp\_password |  |
| 加密算法 | snmp\_encrypt |  |
| 加密密码 | snmp\_priv |  |
| 上下文名称 | snmp\_context |  |

#### 服务器基本信息监控功能预置数据

|  |  |  |
| --- | --- | --- |
| 参数名称 | s\_key | s\_value |
| 是否启用 | snmp\_monitor\_enable | false |
| 允许访问源 | snmp\_source | 127.0.0.1 |
| 允许v1协议 | snmp\_v1 | true |
| v1社区名称 | snmp\_v1\_community | public |
| 允许v2协议 | snmp\_v2c | true |
| v2社区名称 | snmp\_v2c\_community | public |
| 允许v3协议 | snmp\_v3 | true |
| 认证等级 | snmp\_v3\_sec\_level | 3 |
| 用户名 | snmp\_v3\_username | V3 |
| 摘要算法 | snmp\_v3\_digest | 1 |
| 密码 | snmp\_v3\_password | password |
| 加密算法 | snmp\_v3\_encrypt | 1 |
| 加密密码 | snmp\_v3\_priv | priv |
| 上下文名称 | snmp\_v3\_context |  |

1. **SYSLOG预置数据**

|  |  |  |
| --- | --- | --- |
| 参数名称 | s\_key | s\_value |
| 是否启用 | syslog\_enable | false |
| 使用协议 | syslog\_protocol | udp |
| 服务器地址 | syslog \_host | 127.0.0.1 |
| 服务器端口 | syslog \_port | 514 |
| 字符编码 | syslog \_charset | utf8 |

## 消息接口

1. **SNMP配置消息**

#### Trap功能配置消息

|  |
| --- |
| <!-- 修改Trap配置，消息名称：logserver.snmpconfig.update -->  <!-- 请求消息 -->  <updateconfig>  <!-- 启用策略，0：禁用、1：启用 -->  <enable>0</enable>  <!-- SNMP管理中心地址 -->  <host>127.0.0.1</host>  <!-- SNMP管理中心监听端口 -->  <port>162</port>  <!-- 字符集 -->  <charset>utf8</charset>  <!-- 使用协议，1:snmpv1、2:snmpv2c、3:snmpv3 -->  <protocol value=”2”>  <!-- Trap类型，类型值在不同协议中代表不同含义 -->  <!-- snmpv1，1:Trap -->  <!-- snmpv2c，1:Trap、2:Inform -->  <!-- snmpv3，1:Notification、2:Inform -->  <type>1</trap>  <!-- 社区名 -->  <community>public</community>  <!-- 认证等级 -->  <!-- 1:无认证无加密、2:认证无加密、3:认证加密 -->  <level>1</level>  <!-- 用户名 -->  <username>username</username>  <!-- 摘要算法 1:MD5、2:SHA -->  <digest>1</digest>  <!-- 密码 -->  <password>password</password>  <!-- 加密算法 -->  <encrypt>1</encrypt>  <!-- 加密密码 1:DES、2:AES、3:AES128、4:AES192、5:AES256 -->  <priv>priv</priv>  <!-- 上下文名称 -->  <context>contextName</context>  </protocol>  </updateconfig>  <!-- 响应消息 -->  <updateconfig result="success/failure">  <!-- 结果描述，result=”failure”时有效 -->  <errdesc></errdesc>  </updateconfig> |

#### 服务器基本信息监控配置消息

|  |
| --- |
| <!-- 修改服务器基本信息监控配置，消息名称：logserver.sysconfig.update -->  <!-- 请求消息 -->  <updateconfig>  <!-- 启用策略，0：禁用、1：启用 -->  <enable>0</enable>  <!-- 允许访问源，可以填IP，也可填网段如10.10.0.0/16 -->  <source>127.0.0.1</source>  <!-- 允许使用v1协议 -->  <snmpV1 value=”true”>  <!-- 社区名 -->  <community>public</community>  </snmpV1>  <!-- 允许使用v2c协议 -->  <snmpV2c value=”true”>  <!-- 社区名 -->  <community>public</community>  </snmpV2c>  <!-- 允许使用v3协议 -->  <snmpV3 value=”true”>  <!-- 认证等级 -->  <!-- 1:无认证无加密、2:认证无加密、3:认证加密 -->  <level>1</level>  <!-- 用户名 -->  <username>username</username>  <!-- 摘要算法 1:MD5、2:SHA -->  <digest>1</digest>  <!-- 密码 -->  <password>password</password>  <!-- 加密算法 -->  <encrypt>1</encrypt>  <!-- 加密密码 1:DES、2:AES、3:AES128、4:AES192、5:AES256 -->  <priv>priv</priv>  <!-- 上下文名称 -->  <context>contextName</context>  </snmpV3>  </updateconfig>  <!-- 响应消息 -->  <updateconfig result="success/failure">  <!-- 结果描述，result=”failure”时有效 -->  <errdesc></errdesc>  </updateconfig> |

1. <digest>1</digest>**SYSLOG配置消息**

|  |
| --- |
| <!-- 修改SYSLOG配置，消息名称：logserver.syslogconfig.update -->  <!-- 请求消息 -->  <updateconfig>  <!-- 启用策略，0：禁用、1：启用 -->  <enable>0</enable>  <!-- 使用协议，支持udp、tcp、unix\_syslog、unix\_socket协议 -->  <protocol>udp</protocol>  <!-- SYSLOG运维平台地址(unix\_syslog、unix\_socket协议不填写)-->  <host>127.0.0.1</host>  <!-- SYSLOG运维平台监听端口(unix\_syslog、unix\_socket协议不填写)-->  <port>514</port>  <!-- 字符集 -->  <charset>utf8</charset>  </updateconfig>  <!-- 响应消息 -->  <updateconfig result="success/failure">  <!-- 结果描述，result=”failure”时有效 -->  <errdesc></errdesc>  </updateconfig> |

## API接口

1. **SNMP-API接口**

|  |
| --- |
| **public** **interface** SNMPTool  {  /\*\*  \* 使用默认参数初始化SNMP工具  \*/  **public** **void** init();  /\*\*  \* 初始化SNMP工具  \* **@param** params snmp参数  \*/  **public** **void** init(SNMPParams params);  /\*\*  \* 销毁SNMP工具  \*/  **public** **void** destroy();  /\*\*  \* 发送Trap消息  \* **@param** pdu 被发送的Trap  \* **@return** 如果pdu为inform类型，则true:表示发送成功、false表示发送失败  \*/  **public** **boolean** send(SNMPPDU pdu);  /\*\*  \* 获取SNMP参数  \* **@return** SNMP参数  \*/  **public** SNMPParams getSNMPParams();  } |

|  |
| --- |
| **public** **class** SNMPPDU  {  **public** **static** **final** **int** ***TYPE\_V1\_TRAP*** = PDU.***V1TRAP***;  **public** **static** **final** **int** ***TYPE\_V2\_TRAP*** = PDU.***TRAP***;  **public** **static** **final** **int** ***TYPE\_V2\_INFORM*** = PDU.***INFORM***;  **public** **static** **final** **int** ***TYPE\_V3\_NOTIFICATION*** = PDU.***NOTIFICATION***;  **public** **static** **final** **int** ***TYPE\_V3\_INFORM*** = PDU.***INFORM***;  **private** PDU pdu; // SNMP标准PDU  **public** SNMPPDU(**final** **int** protocol, **final** **int** type);  /\*\*  \* 添加属性  \* **@param** oid 属性ID  \* **@param** value 属性值  \*/  **public** **void** add(**final** String oid, **final** Object value);  /\*\*  \* 获取SNMP标准PDU  \* **@return**  \*/  **public** PDU getPDU();  } |

|  |
| --- |
| **public** **class** SNMPParams  {  **public** **static** **final** **int** ***PROTOCOL\_V1*** = SnmpConstants.***version1***;  **public** **static** **final** **int** ***PROTOCOL\_V2c*** = SnmpConstants.***version2c***;  **public** **static** **final** **int** ***PROTOCOL\_V3*** = SnmpConstants.***version3***;  **public** **static** **final** **int** ***DEFAULT\_PROTOCOL*** = ***PROTOCOL\_V2c***;  **public** **static** **final** String ***DEFAULT\_HOST*** = "127.0.0.1";  **public** **static** **final** **int** ***DEFAULT\_PORT*** = 162;  **public** **static** **final** String ***DEFAULT\_COMMUNITY*** = "public";  **public** **static** **final** String ***DEFAULT\_CHARSET*** = "utf8";  **public** **static** **final** **int** ***DEFAULT\_TYPE*** = 1;  **private** String host = ***DEFAULT\_HOST***; // 主机地址  **private** **int** port = ***DEFAULT\_PORT***; // 服务端口  **private** String charset = ***DEFAULT\_CHARSET***; // 默认字符集  **private** **int** protocol = ***DEFAULT\_PROTOCOL***; // 发送协议  **private** String community = ***DEFAULT\_COMMUNITY***; // 社区名称  **private** **int** securetyLevel; // 安全等级  **private** String username; // 用户名  **private** **int** digestAlgorithm; // 摘要算法  **private** String password; // 密码  **private** **int** encryptAlgorithm; // 加密算法  **private** String priv; // 加密密码  **private** String contextName; // 上下文名称  **private** **int** timeout; // 超时时间  **private** **int** retries; // 重试次数  } |

1. **SYSLOG-API接口**

|  |
| --- |
| **public** **interface** SysLogTool  {  **public** **static** **final** String ***PROTOCOL\_UDP*** = SyslogConfigIF.***UDP***;  **public** **static** **final** String ***PROTOCOL\_TCP*** = SyslogConfigIF.***TCP***;  **public** **static** **final** String ***PROTOCOL\_UNIX\_SYSLOG*** = SyslogConfigIF.***UNIX\_SYSLOG***;  **public** **static** **final** String ***DEFAULT\_PROTOCOL*** = ***PROTOCOL\_UDP***;  **public** **static** **final** String ***DEFAULT\_CHARSET*** = "utf8";  **public** **static** **final** **int** ***DEFAULT\_PORT*** = 514;  **public** **static** **final** String ***DEFAULT\_HOST*** = "127.0.0.1";  **public** **static** **final** **int** ***DEFAULT\_FACILITY*** = SyslogConfigIF.***FACILITY\_USER***;  /\*\*  \* 使用默认参数初始化SysLog工具  \* (协议：udp，主机地址：127.0.0.1，端口号：514，字符集：utf-8，设施:用户)  \*/  **public** **void** init();  /\*\*  \* 初始化SysLog工具,设施类型使用默认值-用户  \* **@param** protocol 使用协议  \* **@param** host 主机地址  \* **@param** port 端口号  \* **@param** charset 字符集  \*/  **public** **void** init(String protocol, String host, **int** port, String charset);  /\*\*  \* 初始化SysLog工具  \* **@param** protocol 使用协议  \* **@param** host 主机地址  \* **@param** port 端口号  \* **@param** charset 字符集  \* **@param** facility 设施类型  \*/  **public** **void** init(String protocol, String host, **int** port, String charset,  **int** facility);  /\*\*  \* 销毁 SysLog工具  \*/  **public** **void** destroy();  /\*\*  \* 发送日志消息  \* **@param** level 消息等级  \* **@param** content 消息内容  \*/  **public** **void** log(**int** level, **final** String content);  /\*\*  \* 获取使用协议  \* **@return** 使用协议  \*/  **public** String getProtocol();  /\*\*  \* 获取主机地址  \* **@return** 主机地址  \*/  **public** String getHost();  /\*\*  \* 获取端口  \* **@return** 端口  \*/  **public** **int** getPort();  /\*\*  \* 获取使用字符集  \* **@return** 字符集  \*/  **public** String getCharset();  /\*\*  \* 获取发送设施类型  \* **@return** 设施类型  \*/  **public** **int** getFacility();  } |

## 效果展示

具体日志格式可以后期进行调整

1. **SNMP接收Trap效果**



snmp效果与具体平台有关，显示效果为SNMPc平台

1. **Syslog接收日志效果**

