> 《Zabbix企业级分布式监控系统第2版》随书代码
>
> 代码仓库地址 https://github.com/zabbix-book/zabbix_v2
>
> 书籍购买地址 https://item.jd.com/12653708.html

549页

```
shell# yum -y install  gcc gcc-c++ autoconf  httpd php mariadb-server php-mysql
httpd-manual mod_ssl mod_perl mod_auth_mysql php-gd php-xml php-mbstring php-
ldap php-pear php-xmlrpc php-bcmath mysql-connector-odbc mysql-devel libdbi-
dbd-mysql net-snmp-devel curl-devel    unixODBC-devel   OpenIPMI-devel  java-
devel pcre-devel libxml2-devel openldap-devel libevent-devel openssl-devel
libssh2-devel
```

```
shell# wget
https://sourceforge.net/projects/zabbix/files/ZABBIX%20Latest%20Stable/
4.0.0/zabbix-4.0.0.tar.gz #读者可以下载最新版本，写作本书时最新版本为Zabbix 4.0.0
```

```
shell# groupadd zabbix -g 201                  #添加zabbix用户组
shell# useradd -g zabbix -u 201 -m  zabbix  #增加zabbix用户
shell# tar xvf zabbix-4.0.0.tar.gz          #解压缩包
shell# cd zabbix-4.0.0                          #进入源码目录
shell# ./configure --prefix=/usr --exec-prefix=/usr --bindir=/usr/bin --
sbindir=/usr/sbin --sysconfdir=/etc --datadir=/usr/share --includedir=/usr/
include --libdir=/usr/lib64 --libexecdir=/usr/libexec --localstatedir=/var --
sharedstatedir=/var/lib --mandir=/usr/share/man --infodir=/usr/share/info --
sysconfdir=/etc/zabbix --libdir=/usr/lib64/zabbix  --enable-server --enable-
agent --enable-proxy --enable-ipv6 --enable-java --with-net-snmp --with-ldap --
with-libcurl --with-openipmi --with-unixodbc --with-ssh2 --with-libxml2 --with-
libevent --with-libpcre --with-openssl --with-mysql=/usr/bin/mysql_config
#如果读者只需安装Zabbix-Server端，则开启--enable-server参数即可，其他参数可不用选。
#这里是为了后面的各项功能都可以使用，所以开启了非常多的参数
#如果缺少相应的软件包，在配置过程中会给出提示，使用yum安装所缺少的软件包即可顺利通过安装
shell# make
shell# make install
```

550页

将数据库添加到开机启动项中并启动服务，命令如下：

```
shell# systemctl  enable mariadb   #添加到开机启动项中
shell# systemctl  start  mariadb   #启动服务
```

设置MySQL用户名和密码，命令如下：

```
shell# mysqladmin -uroot password 'mysql_pass'; #设置MySQL的root密码为mysql_pass
shell# mysql -uroot -p #登录数据库，输入刚才设置的密码
mysql> create database zabbix character set utf8; #将zabbix库默认的字符集设置为UTF-8
mysql> grant all privileges on zabbix.* to zabbix@localhost identified by 'zabbix'; #设置zabbix库允许localhost这个IP地址访问，用户名设置为zabbix，密码设置为zabbix
mysql> flush privileges; #刷新权限
```

确保以上操作都正确，测试数据库连接是否正常，命令如下：

```
shell# mysql -uzabbix -pzabbix zabbix
```

导入MySQL库文件，命令如下：

```
shell# cd zabbix-4.0.0 #进入Zabbix源码路径
shell# mysql -uzabbix -pzabbix zabbix < database/mysql/schema.sql
```

551页

如果安装Zabbix-Proxy，则只导入schema.sql即可，无须导入下面的SQL文件；否则，Zabbix-Proxy无法正常工作。

```
shell# mysql -uzabbix -pzabbix zabbix < database/mysql/images.sql
shell# mysql -uzabbix -pzabbix zabbix < database/mysql/data.sql
```

创建日志保存目录，并修改权限，目录如下：

```
shell# mkdir /var/log/zabbix
shell# chown zabbix.zabbix /var/log/zabbix[这是什么命令，缺少文字说明]
```

```
shell# cp misc/init.d/fedora/core/zabbix_* /etc/init.d/
shell# chmod 755 /etc/init.d/zabbix_*
shell# sed -i "s#BASEDIR=/usr/local#BASEDIR=/usr/#g" /etc/init.d/zabbix_server
shell# sed -i "s#BASEDIR=/usr/local#BASEDIR=/usr/#g" /etc/init.d/zabbix_agentd
```

```
shell# cp -r ./zabbix-X.X.X/frontends/php/ /var/www/html/zabbix
shell# chown -R apache.apache /var/www/html/zabbix
```

```
shell# chkconfig zabbix_server on
shell# service zabbix_server start
shell# systemctl enable httpd
shell# systemctl start httpd #如果启动失败，请检查配置文件是否正确
```

```
shell# vim  /etc/php.ini
max_execution_time=300
memory_limit=128M
post_max_size=16M
upload_max_filesize=2M
max_input_time=300
date.timezone=Asia/Shanghai
```

552页

```
hell# wget  https://sourceforge.net/projects/zabbix/files/ZABBIX%20Latest%
20Stable/4.0.0/zabbix-4.0.0.tar.gz
shell# groupadd  zabbix  -g  201
shell# useradd -g  zabbix  -u  201  -m  zabbix
shell# tar xvf   zabbix-4.0.0.tar.gz
shell# cd  zabbix-4.0.0
shell# ./configure --prefix=/usr  --sysconfdir=/etc//zabbix  --enable-agent
shell# make
shell# make   install
shell# mkdir   /var/log/zabbix
shell# chown  zabbix.zabbix   /var/log/zabbix
shell# cp   misc/init.d/fedora/core/zabbix_agentd    /etc/init.d/
shell# chmod  755 /etc/init.d/zabbix_agentd
shell# sed -i "s#BASEDIR=/usr/local#BASEDIR=/usr/#g" /etc/init.d/zabbix_agentd
shell# sed  -i  "s#tmp/zabbix_agentd.log#var/log/zabbix/zabbix_agentd.log#g"
/etc/zabbix/zabbix_agentd.conf
shell# sed  -i  "#UnsafeUserParameters=0#aUnsafeUserParameters=1\n"
/etc/zabbix/zabbix_agentd.conf
#/etc/zabbix/zabbix_agentd.conf配置可参看3.2.3节，这里的X.X.X.X为
#Zabbix-Server的IP地址
启动Zabbix-Agent服务的命令如下：

shell# chkconfig zabbix_agentd  on
shell# service zabbix_agentd  start
```

553页

https://github.com/zabbix-book/zabbix-rpmbuild

```
shell# yum  install rpm-build
```

```
shell# yum install -y gcc make mysql-devel openldap-devel libssh2-devel net-snmp-devel curl-devel unixODBC-devel OpenIPMI-devel java-devel  postgresql-devel net-snmp-devel openldap-devel gnutls-devel sqlite-devel curl-devel OpenIPMI-devel libssh2-devel java-devel libxml2-devel libevent-devel openssl-devel
shell# rpm -ivh http://repo.zabbix.com/non-supported/rhel/7/x86_64/fping-3.10-1.el7.x86_64.rpm
```

554页

```
shell# useradd admin
```

```
shell# su - admin
shell# mkdir -pv rpmbuild/{BUILD,RPMS,SOURCES,SPECS,SRPMS}
shell# echo "% _topdir  /home/admin/rpmbuild" >~/.rpmmacros
```

```
shell# cd /home/admin
shell# wget https://raw.githubusercontent.com/zabbix-book/zabbix-rpmbuild/master/src/zabbix-4.0.0-2.el7.src.rpm
shell# rpm -ivh /home/admin/zabbix-4.0.0-2.el7.src.rpm
```

```
shell# cd  /home/admin/rpmbuild/SPECS
shell# rpmbuild -ba zabbix.spec #此处会提示需要依赖包，依次安装
Executing(%prep): /bin/sh -e /var/tmp/rpm-tmp.FK5o3t
+ umask 022
+ cd /home/admin/rpmbuild/BUILD
+ cd /home/admin/rpmbuild/BUILD
+ rm -rf zabbix-4.0.0
+ /usr/bin/gzip -dc /home/admin/rpmbuild/SOURCES/zabbix-4.0.0.tar.gz
+ /usr/bin/tar -xf -
+ STATUS=0
+ '[' 0 -ne 0 ']'
+ cd zabbix-4.0.0
......省略输出内容......
#打包生成如下RPM文件
Wrote: /home/admin/rpmbuild/SRPMS/zabbix-4.0.0-2.el7.centos.src.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-agent-4.0.0-2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-get-4.0.0-2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-sender-4.0.0-2.el7.centos.x86_64.rpm
```

```
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-proxy-mysql-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-proxy-pgsql-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-proxy-sqlite3-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-java-gateway-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-server-mysql-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/x86_64/zabbix-server-pgsql-4.0.0-
2.el7.centos.x86_64.rpm
Wrote: /home/admin/rpmbuild/RPMS/noarch/zabbix-web-4.0.0-
2.el7.centos.noarch.rpm
Wrote: /home/admin/rpmbuild/RPMS/noarch/zabbix-web-mysql-4.0.0-
2.el7.centos.noarch.rpm
Wrote: /home/admin/rpmbuild/RPMS/noarch/zabbix-web-pgsql-4.0.0-
2.el7.centos.noarch.rpm
Wrote: /home/admin/rpmbuild/RPMS/noarch/zabbix-web-japanese-4.0.0-
2.el7.centos.noarch.rpm
Executing(%clean): /bin/sh -e /var/tmp/rpm-tmp.pDqbD3
+ umask 022
+ cd /home/admin/rpmbuild/BUILD
+ cd zabbix-4.0.0
+ rm -rf /home/admin/rpmbuild/BUILDROOT/zabbix-4.0.0-2.el7.centos.x86_64
+ exit 0
```

555页

```
shell# rpm2cpio zabbix-4.0.0-2.el7.src.rpm |cpio -div
warning: rpm2cpio: NOKEY, key ID a14fe591
config.patch
fonts-config.patch
fping3-sourceip-option.patch
zabbix-4.0.0.tar.gz
zabbix-agent.init
zabbix-agent.service
zabbix-java-gateway.init
zabbix-java-gateway.service
zabbix-logrotate.in
zabbix-proxy.init
zabbix-proxy.service
zabbix-server.init
zabbix-server.service
zabbix-tmpfiles.conf
zabbix-web22.conf
zabbix-web24.conf
zabbix.spec
```

```
zabbix_java_gateway-sysd
```

556页

```
sed -i \
    -e 's|# PidFile=.*|PidFile=%{_localstatedir}/run/%{name}/
zabbix_agentd.pid|g' \
    -e 's|^LogFile=.*|LogFile=%{_localstatedir}/log/%{name}/
zabbix_agentd.log|g' \
    -e '/# UnsafeUserParameters=0/aUnsafeUserParameters=1\n' \
    -e '/# Include.*zabbix_agentd.conf.d\//aInclude=\/etc\/zabbix\/
zabbix_agentd.conf.d\/\n' \
    -e '/StartAgents=3/aStartAgents=5\n' \
    -e 's|# LogFileSize=.*|LogFileSize=0|g' \
    -e 's|Server=127.0.0.1$|Server=127.0.0.1,10.10.10.1|g' \
    -e 's|ServerActive=127.0.0.1$|ServerActive=127.0.0.1:10051,10.10.10.1:
10051|g' \
    -e 's|# EnableRemoteCommands=0|EnableRemoteCommands=1|g' \
    -e 's|# LogRemoteCommands=0|LogRemoteCommands=1|g' \
    -e 's|LogFileSize=0|LogFileSize=10|g' \
    -e 's|/usr/local|/usr|g' \
     $RPM_BUILD_ROOT%{_sysconfdir}/%{name}/zabbix_agentd.conf

sed -i \
    -e 's|/usr/local|/usr|g' \
    -e '/# UnsafeUserParameters=0/aUnsafeUserParameters=1\n' \
    -e 's@# Include=/usr/etc/zabbix_agentd.conf.d@Include=/etc/zabbix/
zabbix_agentd.conf.d@g' \
     $RPM_BUILD_ROOT%{_sysconfdir}/%{name}/zabbix_agent.conf
```

```
%post agent
if [ $1 -eq 1 ]; then
sed -i "s@Hostname=Zabbix server@Hostname=$HOSTNAME@g" /etc/zabbix/
zabbix_agentd.conf
getent group zabbix >/dev/null || groupadd -r  zabbix
getent passwd zabbix >/dev/null || useradd -r -g zabbix -d %{_sharedstatedir}/
zabbix -s   /sbin/nologin  -c "zabbix user" zabbix
/sbin/chkconfig zabbix-agent on
/sbin/service zabbix-agent start
chown root:zabbix /bin/netstat
chmod 4755 /bin/netstat
fi
```

557页

```
"cURL library support >= 7.28.0 is required for Elasticsearch history backend"
```

https://www.elastic.co/downloads/elasticsearch

```
shell# rpm -ivh https://artifacts.elastic.co/downloads/elasticsearch/
elasticsearch-6.1.2.rpm
shell# yum install java-1.8.0 -y
```

```
shell# systemctl daemon-reload           #重新加载systemd进程
shell# systemctl enable elasticsearch  #开机自启动
shell# systemctl start elasticsearch   #启动服务
```

```
shell# tail -f /var/log/elasticsearch/elasticsearch.log
[2018-10-20T17:35:55,407][INFO ][o.e.x.m.j.p.l.CppLogMessageHandler]
[controller/3091] [Main.cc@109] controller (64 bit): Version 6.4.2 (Build
660eefe6f2ea55) Copyright (c) 2018 Elasticsearch BV
[2018-10-20T17:35:56,116][DEBUG][o.e.a.ActionModule ] Using REST wrapper from
plugin org.elasticsearch.xpack.security.Security
[2018-10-20T17:35:56,556][INFO ][o.e.d.DiscoveryModule ] [PYjVnDh] using
discovery type [zen]
[2018-10-20T17:35:57,924][INFO ][o.e.n.Node ] [PYjVnDh] initialized
[2018-10-20T17:35:57,925][INFO ][o.e.n.Node ] [PYjVnDh] starting ...
[2018-10-20T17:35:58,180][INFO ][o.e.t.TransportService ] [PYjVnDh]
publish_address {127.0.0.1:9300}, bound_addresses {127.0.0.1:9300}
```

558页

```
shell# netstat -nlput|grep 9200
tcp6       0      0 127.0.0.1:9200    :::*     LISTEN      3042/java
```

```
shell# vim /etc/elasticsearch/jvm.options
-Xms1g
-Xmx1g
```

第1~5部分

https://github.com/zabbix-book/es-mapping-create/blob/master/step-1-create_elastic_mapping.sh

第7部分

https://github.com/zabbix-book/es-mapping-create/blob/master/step-2-create_elastic_template.sh

第8部分

https://github.com/zabbix-book/es-mapping-create/blob/master/step-3-create_elastic_pipeline.sh

```
shell# curl -X PUT \
http://127.0.0.1:9200/uint \
 -H 'content-type:application/json' \
 -d '{
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "type" : "long"
            }
         }
      }
   }
}'
#创建成功，提示如下
{"acknowledged":true,"shards_acknowledged":true,"index":"uint"}
```

559页

```
shell# curl -X PUT \
http://127.0.0.1:9200/dbl \
 -H 'content-type:application/json' \
 -d '{
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
```

```
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "type" : "double"
            }
         }
      }
   }
}'
#创建成功，提示如下
{"acknowledged":true,"shards_acknowledged":true,"index":"dbl"}
```

560页

```
shell# curl -X PUT \
http://127.0.0.1:9200/str \
 -H 'content-type:application/json' \
 -d '{
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "fields" : {
                  "analyzed" : {
                     "index" : true,
                     "type" : "text",
                     "analyzer" : "standard"
                  }
               },
               "index" : false,
               "type" : "text"
            }
```

```
          }
       }
    }
}'
#创建成功，提示如下
{"acknowledged":true,"shards_acknowledged":true,"index":"str"}
```

561页

```
shell# curl -X PUT \
http://127.0.0.1:9200/text \
 -H 'content-type:application/json' \
 -d '{
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "fields" : {
                  "analyzed" : {
                     "index" : true,
                     "type" : "text",
                     "analyzer" : "standard"
                  }
               },
               "index" : false,
               "type" : "text"
            }
         }
      }
   }
}'
#创建成功，提示如下
{"acknowledged":true,"shards_acknowledged":true,"index":"text"}
```

562页

```
shell# curl -X PUT \
http://127.0.0.1:9200/log \
 -H 'content-type:application/json' \
 -d '{
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "fields" : {
                  "analyzed" : {
                     "index" : true,
                     "type" : "text",
                     "analyzer" : "standard"
                  }
               },
               "index" : false,
               "type" : "text"
            }
         }
      }
   }
}'
#创建成功，提示如下
{"acknowledged":true,"shards_acknowledged":true,"index":"log"}
```

563页

```
### Option: HistoryStorageDateIndex
#       Enable preprocessing of history values in history storage to store
values in different indices based on date.
#       0 - disable
#       1 - enable
#
# Mandatory: no
# Default:
HistoryStorageDateIndex=1 #开启按天索引的功能
```

```
#src/libs/zbxhistory/history_elastic.c
static int  elastic_add_values(zbx_history_iface_t *hist, const
zbx_vector_ptr_t *history)
{
    ......省略部分代码......
    zbx_json_init(&json_idx, ZBX_IDX_JSON_ALLOCATE);

    zbx_json_addobject(&json_idx, "index");
    zbx_json_addstring(&json_idx, "_index", value_type_str[hist->value_type],
ZBX_JSON_TYPE_STRING);
    zbx_json_addstring(&json_idx, "_type", "values", ZBX_JSON_TYPE_STRING);
    if (1 == CONFIG_HISTORY_STORAGE_PIPELINES)//pipeline的开关
    {
        zbx_snprintf(pipeline, sizeof(pipeline), "%s-pipeline", value_type_str
[hist->value_type]);
        zbx_json_addstring(&json_idx, "pipeline", pipeline,
ZBX_JSON_TYPE_STRING);
    }
    ......省略部分代码......
}
```

564页

```
shell# curl -X PUT \
 http://127.0.0.1:9200/_template/text_template \
 -H 'content-type:application/json' \
 -d '{
   "template": "text*",           #匹配的数据类型为text*，其他数据类型类似
   "index_patterns": ["text*"],   #匹配的数据类型为text*，其他数据类型类似
   "settings" : {
      "index" : {
         "number_of_replicas" : 1,
         "number_of_shards" : 5
      }
   },
   "mappings" : {
```

```
      "values" : {
         "properties" : {
            "itemid" : {
               "type" : "long"
            },
            "clock" : {
               "format" : "epoch_second",
               "type" : "date"
            },
            "value" : {
               "fields" : {
                  "analyzed" : {
                     "index" : true,
                     "type" : "text",
                     "analyzer" : "standard"
                  }
               },
               "index" : false,
               "type" : "text"
            }
         }
      }
   }
}'
#创建成功，提示如下
{"acknowledged":true}
```

565页

```
shell# curl -X PUT \
http://127.0.0.1:9200/_ingest/pipeline/uint-pipeline \
 -H 'content-type:application/json' \
 -d '{
  "description": "daily uint index naming",
  "processors": [
    {
      "date_index_name": {           #按日期对字段索引
        "field": "clock",              #匹配哪个字段
        "date_formats": ["UNIX"], #日期格式，可用字段有ISO8601、UNIX、UNIX_MS和
TAI64N
        "index_name_prefix": "uint-", #索引名称前缀
        "date_rounding": "d" #可用字段有y(年)、M(月)、w(星期)、d(日)、h(小时)、m(分
钟)和s(秒)
      }
    }
  ]
}'
#创建成功，提示如下
```

```
{"acknowledged":true}
```

https://www.elastic.co/guide/en/elasticsearch/reference/6.3/date-index-name-processor.html

```
配置Zabbix-Server，命令如下：
shell# vim  /etc/zabbix/zabbix_server.conf
### Option: HistoryStorageURL
# History storage HTTP[S] URL.
#
# Mandatory: no
# Default:
HistoryStorageURL=http://192.168.0.15:9200

### Option: HistoryStorageTypes
#   Comma separated list of value types to be sent to the history storage.
#
# Mandatory: no
# Default:
HistoryStorageTypes=uint,dbl,str,log,text
HistoryStorageDateIndex=1
重启Zabbix-Server服务，命令如下：

shell# systemctl restart zabbix-server
```

566页

```
shell# vim /etc/zabbix/web/zabbix.conf.php
global $DB, $HISTORY; //一定要加上$HISTORY，否则无法生效
//$HISTORY['url']   = [
// 'uint' => 'http://localhost:9200',
// 'text' => 'http://localhost:9200'
//];
// Value types stored in Elasticsearch.
//$HISTORY['types'] = ['uint', 'text'];
$HISTORY['url']   = 'http://192.168.0.15:9200';
$HISTORY['types'] = ['uint','dbl','str','log','text'];
```