

3 Proxy

概述

Zabbix proxy 是一个可以从一个或多个受监控设备采集监控数据并将信息发送到 Zabbix server 的进程，主要是代表 Zabbix server 工作。所有收集的数据都在本地缓存，然后传输到 proxy 所属的 Zabbix server。

部署 Zabbix proxy 是可选的，但可能非常有利于分担单个 Zabbix server 的负载。如果只有代理采集数据，则 Zabbix server 上会减少 CPU 和磁盘 I/O 的开销。

Zabbix proxy 是无需本地管理员即可集中监控远程位置、分支机构和网络的理想解决方案。

Zabbix proxy 需要使用独立的数据库。

值得注意的是 Zabbix proxy 支持 SQLite、MySQL 和 PostgreSQL 作为数据库。使用 Oracle 或 DB2 需要您承担一定的风险，例如，在自动发现规则中的遇到问题 [返回值](#)。

详见：[在分布式环境中使用 Zabbix proxy](#)。

Proxy 进程

通过二进制包安装的组件

Zabbix proxy 进程以守护进程（Daemon）运行。Zabbix proxy 的启动可以通过执行以下命令来完成：

```
shell> service zabbix-proxy start
```

上述命令在大多数的 GNU/Linux 系统下都可以正常完成。如果是其他系统，您可能要尝试以下命令来运行：

```
shell> /etc/init.d/zabbix-proxy start
```

类似的 Zabbix proxy 的停止、重启、查看状态，则需要执行以下命令：

```
shell> service zabbix-proxy stop
shell> service zabbix-proxy restart
shell> service zabbix-proxy status
```

手动启动

如果以上操作均无效，您可能需要手动启动，找到 Zabbix proxy 二进制文件的路径并且执行：

```
shell> zabbix_proxy
```

您可以将以下命令行参数用于 Zabbix proxy。

```
-c --config <file>      配置文件路径
-R --runtime-control <option>  执行管理功能
-h --help                帮助
-V --version             显示版本号
```

运行时机制的控制不支持 OpenBSD 和 NetBSD 系统。

使用命令行参数运行 Zabbix proxy 的示例::

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf
shell> zabbix_proxy --help
shell> zabbix_proxy -V
```

运行时控制

运行时控制包含的选项:

选项	描述	目标
config_cache_reload	重新加载配置缓存。如果当前正在加载缓存，则忽略。 主动模式下的 Zabbix proxy 将连接到 Zabbix server 并请求配置数据。	
housekeeper_execute	启动管家程序。忽略当前正在进行中的管家程序。	
log_level_increase[=<target>]	增加日志级别，如果未指定目标，将影响所有进程。	pid - 进程标识符 (1 to 65535)
log_level_decrease[=<target>]	降低日志级别，如果未指定目标，则会影响所有进程。	process type - 指定进程的所有类型 (例如 poller) process type,N - 进程类型和编号 (例如 poller,3)

单一 Zabbix 进程的日志级别改变后，进程的 PIDs 的值也会改变，允许的范围为1~65535。在具有大 PIDs <process type,N> 目标选项可更改单个进程的日志级别。

例如，使用 config_cache_reload 选项重新加载 proxy 的配置缓存:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R
config_cache_reload
```

例如，使用 housekeeper_execute 选项来触发管家服务执行:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R
housekeeper_execute
```

例如，使用 log_level_increase 选项来改变日志级别:

增加所有进程的日志级别:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R
log_level_increase
```

增加第二个 Poller 进程的日志级别:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R  
log_level_increase=poller,2
```

增加 PID 为 1234 进程的日志级别:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R  
log_level_increase=1234
```

降低 http poller 进程的日志级别:

```
shell> zabbix_proxy -c /usr/local/etc/zabbix_proxy.conf -R  
log_level_decrease="http poller"
```

进程用户

Zabbix proxy 允许使用非 root 用户运行。它将以任何非 root 用户的身份运行。因此，使用非 root 用户运行 proxy 是没有任何问题的。

如果你试图以“root”身份运行它，它将会切换到一个已经“写死”的“zabbix”用户，该用户必须存在于您的系统上。如果您只想以“root”用户运行 proxy，您必须在 proxy 配置文件里修改‘AllowRoot’参数。

配置文件

有关配置 Zabbix proxy 的详细信息，请查阅 [配置文件](#) 章节。

支持的平台

Zabbix proxy 在与 Zabbix server 相同的 [受支持的平台](#) 列表上运行。

语言环境

值得注意的是，Zabbix proxy 需要 UTF-8 语言环境，以便可以正确解释某些文本项。大多数现代类 Unix 系统都默认使用 UTF-8 语言环境，但是，有些系统可能需要专门设置。

From:
<https://www.zabbix.com/documentation/4.0/> - **Zabbix Documentation 4.0**

Permanent link:
<https://www.zabbix.com/documentation/4.0/zh/manual/concepts/proxy>

Last update: **2018/08/21 07:51**

