

tgtd优化为多进程模式建议

问题说明

tgtd是用户态的iscsi的程序，通过用户态的程序对外提供iscsi的接口的服务，通常来说，用户态和内核态各有好处，内核态的程序出现问题，需要重启服务器进行服务的恢复，用户态通过重启进程即可，对整个操作系统的影响相对较小，目前我们的系统采用的是tgtd的用户态的程序

tgtd默认是通过单进程，多线程的方式对外进行服务的，也就是默认提供一个tgtd的进程，在进程内进行服务的配置，这个正常来说一般是没有问题的，出现问题的地方是在如果多个target或者lun中的某一个出现问题的时候，需要进行重启的时候，就会对其它的服务产生影响，正常情况重载是可以处理配置更新的，这里说的是必须重启的场景

同样的一般来说，服务器上的程序，如果对外服务是对等关系，能够多进程的时候，是会通过多进程的处理，性能也会更好一点

那么目前使用的是单进程的方式，是可以优化成多进程的方式的，并且能够同时兼容之前的单进程模式的，本篇会讲述如何处理

TGT启动流程分析

启动的文件在/usr/lib/systemd/system/tgtd.service

启动服务流程

- 启动一个TGTD进程
- 通过tgtadm让服务离线
- 通过tgt-admin执行当前的配置文件
- 通过tgtadm让服务进行上线

重载服务的流程

- 通过tgt-admin更新配置文件内的配置

停止服务流程

- 通过tgtadm让服务离线
- 通过tgt-admin更新配置文件为空
- 通过tgtadm让进程离线

通过上面可以看到服务是启动进程后，对进程进行加减配置来改变服务的，属于单进程处理，并且配置是通过一个include加载所有配置的方式，这个后面我们改造的需要改成独立控制的

tgtadm是通过/var/run/tgtd/内部的socket进行通信的，而启动进程的时候是可以通过portal去指定ip和端口的，因此我们只需要改造一下启动文件就能够通过不修改内部逻辑的情况下实现支持多进程

多进程实现方式

systemd比init启动的好处是增加了一些标准化的控制，比如通过@来读取参数来区分实例，这个在ceph里面启动osd的时候可以看到

```
systemctl start ceph-osd@0
```

这里是可以把@后面的0作为一个参数传递到启动里面去的，那么我们的改造也通过这种方式来处理

读取指定的配置文件

通过@后面把配置文件的前缀传递到启动脚本里面去

假设配置文件为

```
/etc/tgt/conf.d/my.conf
```

通过systemd环境变量设置

```
Environment="TGTD_CONFIG=/etc/mytgt/conf.d/%i.conf"
```

读取端口和socket设置

我们上面读取到了指定的配置文件，这里就通过指定的配置文件内部设置固定参数来实现

设置环境变量的配置文件为我们指定配置文件

```
EnvironmentFile=/etc/mytgt/conf.d/%i.conf
```

设置指定的参数

```
SOCKET_OPTS="3263"  
PORTAL_OPTS="0.0.0.0:3263"
```

这里是固定的写法，指定端口，上下一致

这两个参数传递到启动脚本里面了，内部主要三个命令，tgt, tgtd, tgtadm, tgt-admin, 我们需要传参进去即可 语法如下

```
tgt      传递 --iscsi portal=${PORTAL_OPTS} -C ${SOCKET_OPTS}  
tgtd     传递 --control-port ${SOCKET_OPTS}  
tgt-admin 传递 -C ${SOCKET_OPTS}
```

然后就可以针对独立进程配置了 完成的配置文件如下

```
[Unit]  
Description=tgtd iSCSI target daemon  
After=network.target  
  
[Service]  
#EnvironmentFile=/etc/sysconfig/tgtd  
EnvironmentFile=/etc/mytgt/conf.d/%i.conf  
Environment="TGTD_CONFIG=/etc/mytgt/conf.d/%i.conf"
```

```

ExecStart=/usr/sbin/tgtd -f $TGTD_OPTS --iscsi portal=${PORTAL_OPTS} -C
${SOCKET_OPTS}
# see bz 848942. workaround for a race for now.
ExecStartPost=/bin/sleep 5
# Put tgtd into "offline" state until all the targets are configured.
# We don't want initiators to (re)connect and fail the connection
# if it's not ready.
ExecStartPost=/usr/sbin/tgtadm --control-port ${SOCKET_OPTS} --op update --mode
sys --name State -v offline
# Configure the targets.
ExecStartPost=/usr/sbin/tgt-admin -C ${SOCKET_OPTS} -e -c $TGTD_CONFIG
# Put tgtd into "ready" state.
ExecStartPost=/usr/sbin/tgtadm --control-port ${SOCKET_OPTS} --op update --
mode sys --name State -v ready

# Update configuration for targets. Only targets which
# are not in use will be updated.
ExecReload=/usr/sbin/tgt-admin -C ${SOCKET_OPTS} --update ALL -c $TGTD_CONFIG

# NOTE: Shutdown of the iscsi target may cause data corruption
# for initiators that are connected.
ExecStop=/usr/sbin/tgtadm --control-port ${SOCKET_OPTS} --op update --mode sys
--name State -v offline
# Remove all targets. It only removes targets which are not in use.
ExecStop=/usr/sbin/tgt-admin -C ${SOCKET_OPTS} --update ALL -c /dev/null
# tgtd will exit if all targets were removed
ExecStop=/usr/sbin/tgtadm --control-port ${SOCKET_OPTS} --op delete --mode
system

[Install]
WantedBy=multi-user.target

```

兼容性问题

未对原始程序做任何修改，所以不会改变程序自身的稳定性，原始程序配置文件直接写在/etc/tgt/下面，默认3260端口，需要独立配置端口的程序写在/etc/mytgt/下面

默认的通过

```
systemctl start tgtd
```

独立设置的通过

```
systemctl start tgtd@mytgt
```

查看进程，正常情况下就是如下的状态

```
[root@lab101 system]# ps -ef|grep tgt
root      2322      1  0 10:10 ?          00:00:00 /usr/sbin/tgtd -f
root      2387      1  0 10:10 ?          00:00:00 /usr/sbin/tgtd -f --iscsi
portal=0.0.0.0:3263 -C 3263
root      2792    1539  0 11:18 pts/0    00:00:00 grep --color=auto tgt
[root@lab101 system]# netstat -tunlp|grep tgt
tcp        0      0 0.0.0.0:3260          0.0.0.0:*          LISTEN
2322/tgtd
tcp        0      0 0.0.0.0:3263          0.0.0.0:*          LISTEN
2387/tgtd
tcp6       0      0 :::3260              :::*                LISTEN
2322/tgtd
```

查询默认的target信息

```
tgt-admin -s
```

查询指定端口3263的进程的信息

```
tgt-admin -C 3263 -s
```

通过上面的修改后就可以实现多进程管理了，这个改造是有理论支撑的，有公司进行了专利申请，这种属于程序内部控制，我们自己按自己的方式实现即可

<https://patents.google.com/patent/CN107277131A/zh>

本发明公开了一种分布式存储系统中TGT稳定性优化方法，所述优化方法应用于分布式存储系统或单机存储使用，其特征在于，该优化方法是使用TGT对外提供iSCSI Target服务时，通过多进程与模块拆分方式，解决单机内TGT形成单点故障的问题，增加故障隔离性，极大地减少了故障影响范围，将系统稳定性大幅度提升；用户态进程在操作系统实现了资源隔离与抽象，用户态进程之间可以做到完全隔离，互不影响，利用进程隔离性，将TGT服务更加细粒度化，更加可控，更小的影响范围。

管理平台来说，如果用的默认3260端口的就放到默认路径下面去，如果改变了端口的，同一端口的就放到同一个配置文件里面即可