ZFS源码编译和调试

作者	时间	QQ群
perrynzhou@gmail.com	2021/09/28	672152841

存储内核技术交流

微信扫描二维码，关注我的公众号

开源存储问题解答社区:https://github.com/perrynzhou/deep-dive-storage-in-china

OS环境

```
$ uname -r
3.10.0-1160.42.2.el7.x86_64
$ cat /etc/redhat-release
CentOS Linux release 7.9.2009 (Core)
```

ZFS源码依赖安装

- centos7

```
// centos7版本的依赖
$ sudo yum install epel-release gcc make autoconf automake libtool rpm-
build libtirpc-devel libblkid-devel libuuid-devel libudev-devel openssl-
devel zlib-devel libaio-devel libattr-devel elfutils-libelf-devel kernel-
devel-$(uname -r) python python2-devel python-setuptools python-cffi
```

```
libffi-devel ncompress
$ sudo yum install --enablerepo=epel dkms python-packaging

$ yum install kernel-debug-devel-$(uname -r) kernel-debug-$(uname -r)
kernel-tools-$(uname -r) kernel-tools-libs-$(uname -r) perf  gdb -y
// 备注: zfs在0.8之前的版本需要独立安装spl,自从0.8版本以及以后版本都整合到zfs的主代码分支了
```

- centos8

```
sudo dnf install --skip-broken epel-release gcc make autoconf automake
libtool rpm-build kernel-rpm-macros libtirpc-devel libblkid-devel libuuid-
devel libudev-devel openssl-devel zlib-devel libaio-devel libattr-devel
elfutils-libelf-devel kernel-devel-$(uname -r) python3 python3-devel
python3-setuptools python3-cffi libffi-devel ncompress
sudo dnf install --skip-broken --enablerepo=epel --enablerepo=powertools
python3-packaging dkms
```

ZFS 源码编译

- zfs编译

```
// 源码下载
$ git clone https://github.com/openzfs/zfs && cd zfs
$ git checkout master

$ cd zfs
$ sh autogen.sh
$ CFLAGS="-ggdb3 -O0" ./configure   --enable-debug  --enable-debuginfo   --
enable-debug-kmem  --enable-debug-kmem-tracking --with-spec=redhat  --with-
linux=/usr/src/kernels/$(uname -r)
$ make -s -j$(nproc)
$ make rpm
$ yum localinstall *.$(uname -p).rpm
```

生成写入过程函数调用栈

```
$ modprobe  zfs
$ zpool create cow /dev/sdc
$ zfs create cow/fs1

// 防止在perf过程出现unknown符号
$ echo 0 > /proc/sys/kernel/kptr_restrict
```

```
// perf记录函数调用关系

// 方式1
$ perf record -F 99 -a -g -- dd if=/dev/urandom of=/cow/fs1/testfile.out

// 方式2
$ perf record  --call-graph dwarf -F 99 -a -s dd if=/dev/urandom
of=/cow/fs1/testfile.out

$ git clone  https://github.com/brendangregg/FlameGraph.git && cd FlameGraph


// 文本形式函数调用生成
$ perf report --stdio -i ~/perf.data

// 生成火焰图
$ perf script -i ~/perf.data > out.perf
$ ./stackcollapse-perf.pl out.perf > out.folded
$ ./flamegraph.pl out.folded > dd-file-on-zfs.svg
```