Ubuntu16-SandFs 复现

换源

```
1  deb http://mirrors.aliyun.com/ubuntu/ xenial main restricted universe
   multiverse
2  deb http://mirrors.aliyun.com/ubuntu/ xenial-security main restricted
   universe multiverse
3  deb http://mirrors.aliyun.com/ubuntu/ xenial-updates main restricted
   universe multiverse
4  deb http://mirrors.aliyun.com/ubuntu/ xenial-proposed main restricted
   universe multiverse
5  deb http://mirrors.aliyun.com/ubuntu/ xenial-backports main restricted
   universe multiverse
6  deb-src http://mirrors.aliyun.com/ubuntu/ xenial main restricted universe
   multiverse
7  deb-src http://mirrors.aliyun.com/ubuntu/ xenial-security main restricted
   universe multiverse
8  deb-src http://mirrors.aliyun.com/ubuntu/ xenial-updates main restricted
   universe multiverse
9  deb-src http://mirrors.aliyun.com/ubuntu/ xenial-proposed main restricted
   universe multiverse
10 deb-src http://mirrors.aliyun.com/ubuntu/ xenial-backports main restricted
   universe multiverse
```

编译相关

```
1  sudo apt update -y &&
2  sudo apt install make gcc bc -y &&
3  sudo apt install pkg-config libncurses-dev -y &&
4  sudo apt install flex -y &&
5  sudo apt install bison -y &&
6  sudo apt install -y libncurses5-dev flex bison libssl-dev dpkg-dev libelf-
   dev&&
7  sudo apt install -y wget curl &&
8  sudo apt install fakeroot -y&&
9  sudo apt install rsync kmod cpio -y&&
10 sudo apt install git -y&&
11 sudo apt install clang -y&&
12 sudo apt install llvm -y&&
13 sudo apt-get install libcap-dev -y
14 git clone   --depth=1  https://github.com/sandfs/SandFS-Kernel
15 sudo cp -r SandFS-Kernel/ /usr/src/
16 cd /usr/src/SandFS-Kernel
17 sudo make menuconfig
18 Select 'File systems -> Misellaneous filesystems
19          -->   Sandfs sandboxing file system' and save/exit.
20 sudo make -j4
21 sudo mkdir /lib/modules/4.10.0-rc8+
22 sudo make modules_install
23 sudo make install -j4
24 sudo update-grub
25 # 4.10.0-rc8+
```

```
    --- Miscellaneous filesystems
    <M>   ORANGEFS (Powered by PVFS) support
    <M>   ADFS file system support
    [ ]      ADFS write support (DANGEROUS)
    <M>   Amiga FFS file system support
    <*>   eCrypt filesystem layer support
    [*]      Enable notifications for userspace key wrap/unwrap
    < >   Wrapfs stackable file system (EXPERIMENTAL) (NEW)
    < >   Sandfs sandboxing file system (EXPERIMENTAL) (NEW)
    <M>   Apple Macintosh file system support
    <M>   Apple Extended HFS file system support
    [ ]      HFS+ POSIX Access Control Lists (NEW)
```

```
go@ubuntu:~$ uname -r
4.10.0-rc8+
go@ubuntu:~$ ls
Desktop     Downloads       install.sh   Pictures   Templates
Documents   examples.desktop Music        Public     Videos
go@ubuntu:~$ ls
Desktop     Downloads       install.sh   Pictures   Templates
Documents   examples.desktop Music        Public     Videos
go@ubuntu:~$ cd ..
go@ubuntu:/home$
```

```
 1  git clone https://github.com/sandfs/LibSandFS.git
 2  make clean && sudo LLC=llc CLANG=clang make
 3  go@ubuntu:~/Desktop/LibSandFS$ sudo LLC=llc CLANG=clang make
 4  make -C /lib/modules/`uname -r`/build $PWD/
 5  make[1]: Entering directory '/usr/src/SandFS-Kernel'
 6    CHK     include/config/kernel.release
 7    CHK     include/generated/uapi/linux/version.h
 8    CHK     include/generated/utsrelease.h
 9    CHK     include/generated/timeconst.h
10    CHK     include/generated/bounds.h
11    CHK     include/generated/asm-offsets.h
12    CALL    scripts/checksyscalls.sh
13    HOSTCC  /home/go/Desktop/LibSandFS/src/utils.o
14    HOSTCC  /home/go/Desktop/LibSandFS/src/ebpf.o
15  In file included from ./samples/bpf/libbpf.h:5:0,
16                   from /home/go/Desktop/LibSandFS/src/ebpf.c:15:
17  ./tools/lib/bpf/bpf.h:45:7: warning: 'enum bpf_attach_type' declared inside
    parameter list
18        unsigned int flags);
19        ^
20  ./tools/lib/bpf/bpf.h:45:7: warning: its scope is only this definition or
    declaration, which is probably not what you want
21  ./tools/lib/bpf/bpf.h:46:45: warning: 'enum bpf_attach_type' declared
    inside parameter list
22   int bpf_prog_detach(int attachable_fd, enum bpf_attach_type type);
23                                           ^
24  /home/go/Desktop/LibSandFS/src/ebpf.c: In function 'ebpf_ctrl_update':
25  /home/go/Desktop/LibSandFS/src/ebpf.c:19:29: warning: implicit declaration
    of function 'bpf_update_elem' [-Wimplicit-function-declaration]
26   #define bpf_map_update_elem bpf_update_elem
27                                    ^
```

```
28   /home/go/Desktop/LibSandFS/src/ebpf.c:98:9: note: in expansion of macro
     'bpf_map_update_elem'
29     return bpf_map_update_elem(context->ctrl_fd, (void *) key,
30           ^
31   /home/go/Desktop/LibSandFS/src/ebpf.c: In function 'ebpf_ctrl_delete':
32   /home/go/Desktop/LibSandFS/src/ebpf.c:21:29: warning: implicit declaration
     of function 'bpf_delete_elem' [-Wimplicit-function-declaration]
33    #define bpf_map_delete_elem bpf_delete_elem
34                               ^
35   /home/go/Desktop/LibSandFS/src/ebpf.c:105:9: note: in expansion of macro
     'bpf_map_delete_elem'
36     return bpf_map_delete_elem(context->ctrl_fd, (void *) key);
37           ^
38   /home/go/Desktop/LibSandFS/src/ebpf.c: In function 'ebpf_data_lookup':
39   /home/go/Desktop/LibSandFS/src/ebpf.c:20:29: warning: implicit declaration
     of function 'bpf_lookup_elem' [-Wimplicit-function-declaration]
40    #define bpf_map_lookup_elem bpf_lookup_elem
41                               ^
42   /home/go/Desktop/LibSandFS/src/ebpf.c:118:9: note: in expansion of macro
     'bpf_map_lookup_elem'
43     return bpf_map_lookup_elem(context->data_fd[idx], key, val);
44           ^
45   cp ./samples/bpf/bpf_load.c /home/go/Desktop/LibSandFS/src/.
46   if [ -f ./samples/bpf/libbpf.c ]; then \
47       cp ./samples/bpf/libbpf.c /home/go/Desktop/LibSandFS/src/.; \
48   else \
49       cp ./tools/lib/bpf/bpf.c /home/go/Desktop/LibSandFS/src/libbpf.c; \
50   fi;
51     HOSTCC   /home/go/Desktop/LibSandFS/src/libbpf.o
52   In file included from /home/go/Desktop/LibSandFS/src/libbpf.c:27:0:
53   ./tools/lib/bpf/bpf.h:45:7: warning: 'enum bpf_attach_type' declared inside
     parameter list
54         unsigned int flags);
55         ^
56   ./tools/lib/bpf/bpf.h:45:7: warning: its scope is only this definition or
     declaration, which is probably not what you want
57   ./tools/lib/bpf/bpf.h:46:45: warning: 'enum bpf_attach_type' declared
     inside parameter list
58    int bpf_prog_detach(int attachable_fd, enum bpf_attach_type type);
59                                                 ^
60   /home/go/Desktop/LibSandFS/src/libbpf.c: In function 'bpf_create_map':
61   /home/go/Desktop/LibSandFS/src/libbpf.c:67:6: error: 'union bpf_attr' has
     no member named 'map_flags'
62     attr.map_flags = map_flags;
63         ^
64   /home/go/Desktop/LibSandFS/src/libbpf.c: At top level:
65   /home/go/Desktop/LibSandFS/src/libbpf.c:172:7: warning: 'enum
     bpf_attach_type' declared inside parameter list
66         unsigned int flags)
67         ^
68   /home/go/Desktop/LibSandFS/src/libbpf.c:171:70: error: parameter 3 ('type')
     has incomplete type
69    int bpf_prog_attach(int prog_fd, int target_fd, enum bpf_attach_type type,
70                                                         ^
71   /home/go/Desktop/LibSandFS/src/libbpf.c: In function 'bpf_prog_attach':
72   /home/go/Desktop/LibSandFS/src/libbpf.c:177:6: error: 'union bpf_attr' has
     no member named 'target_fd'
73     attr.target_fd    = target_fd;
```

```
    74        ^
    75  /home/go/Desktop/LibSandFS/src/libbpf.c:178:6: error: 'union bpf_attr' has
        no member named 'attach_bpf_fd'
    76    attr.attach_bpf_fd = prog_fd;
    77        ^
    78  /home/go/Desktop/LibSandFS/src/libbpf.c:179:6: error: 'union bpf_attr' has
        no member named 'attach_type'
    79    attr.attach_type   = type;
    80        ^
    81  /home/go/Desktop/LibSandFS/src/libbpf.c:180:6: error: 'union bpf_attr' has
        no member named 'attach_flags'
    82    attr.attach_flags  = flags;
    83        ^
    84  /home/go/Desktop/LibSandFS/src/libbpf.c:182:17: error: 'BPF_PROG_ATTACH'
        undeclared (first use in this function)
    85    return sys_bpf(BPF_PROG_ATTACH, &attr, sizeof(attr));
    86                   ^
    87  /home/go/Desktop/LibSandFS/src/libbpf.c:182:17: note: each undeclared
        identifier is reported only once for each function it appears in
    88  /home/go/Desktop/LibSandFS/src/libbpf.c: At top level:
    89  /home/go/Desktop/LibSandFS/src/libbpf.c:185:41: warning: 'enum
        bpf_attach_type' declared inside parameter list
    90   int bpf_prog_detach(int target_fd, enum bpf_attach_type type)
    91                                                ^
    92  /home/go/Desktop/LibSandFS/src/libbpf.c:185:57: error: parameter 2 ('type')
        has incomplete type
    93   int bpf_prog_detach(int target_fd, enum bpf_attach_type type)
    94                                                            ^
    95  /home/go/Desktop/LibSandFS/src/libbpf.c: In function 'bpf_prog_detach':
    96  /home/go/Desktop/LibSandFS/src/libbpf.c:190:6: error: 'union bpf_attr' has
        no member named 'target_fd'
    97    attr.target_fd  = target_fd;
    98        ^
    99  /home/go/Desktop/LibSandFS/src/libbpf.c:191:6: error: 'union bpf_attr' has
        no member named 'attach_type'
   100    attr.attach_type = type;
   101        ^
   102  /home/go/Desktop/LibSandFS/src/libbpf.c:193:17: error: 'BPF_PROG_DETACH'
        undeclared (first use in this function)
   103    return sys_bpf(BPF_PROG_DETACH, &attr, sizeof(attr));
   104                   ^
   105  /home/go/Desktop/LibSandFS/src/libbpf.c: In function 'bpf_prog_attach':
   106  /home/go/Desktop/LibSandFS/src/libbpf.c:183:1: warning: control reaches end
        of non-void function [-Wreturn-type]
   107   }
   108   ^
   109  /home/go/Desktop/LibSandFS/src/libbpf.c: In function 'bpf_prog_detach':
   110  /home/go/Desktop/LibSandFS/src/libbpf.c:194:1: warning: control reaches end
        of non-void function [-Wreturn-type]
   111   }
   112   ^
   113  scripts/Makefile.host:124: recipe for target
        '/home/go/Desktop/LibSandFS/src/libbpf.o' failed
   114  make[2]: *** [/home/go/Desktop/LibSandFS/src/libbpf.o] Error 1
   115  Makefile:1658: recipe for target '/home/go/Desktop/LibSandFS/' failed
   116  make[1]: *** [/home/go/Desktop/LibSandFS/] Error 2
   117  make[1]: Leaving directory '/usr/src/SandFS-Kernel'
   118  Makefile:38: recipe for target 'all' failed
```

```
119    make: *** [all] Error 2
```

https://github.com/SamerW/RootAsRole/issues/8

https://www.mail-archive.com/netdev@vger.kernel.org/msg173115.html

```
1    make -C /lib/modules/`uname -r`/build $PWD/
2    make[1]: Entering directory '/usr/src/SandFS-Kernel'
3      CHK     include/config/kernel.release
4      CHK     include/generated/uapi/linux/version.h
5      CHK     include/generated/utsrelease.h
6      CHK     include/generated/timeconst.h
7      CHK     include/generated/bounds.h
8      CHK     include/generated/asm-offsets.h
9      CALL    scripts/checksyscalls.sh
10     HOSTCC  /home/go/Desktop/LibSandFS/src/libbpf.o
11     HOSTCC  /home/go/Desktop/LibSandFS/src/bpf_load.o
12   /home/go/Desktop/LibSandFS/src/bpf_load.c:26:22: fatal error: perf-sys.h: No
     such file or directory
13   compilation terminated.
14   scripts/Makefile.host:124: recipe for target
     '/home/go/Desktop/LibSandFS/src/bpf_load.o' failed
15   make[2]: *** [/home/go/Desktop/LibSandFS/src/bpf_load.o] Error 1
16   Makefile:1658: recipe for target '/home/go/Desktop/LibSandFS/' failed
17   make[1]: *** [/home/go/Desktop/LibSandFS/] Error 2
18   make[1]: Leaving directory '/usr/src/SandFS-Kernel'
19   Makefile:38: recipe for target 'all' failed
20   make: *** [all] Error 2
```


makefile

```
1    EXTRA_CFLAGS += -I$(PWD)/include -I$(objtree)/samples/bpf
2
3    HOSTCFLAGS += -fPIC -I$(objtree)/usr/include -I$(PWD)/include
4    HOSTCFLAGS += -I$(objtree)/samples/bpf -I$(objtree)/tools/lib -
     I$(objtree)/tools/lib/bpf
5    HOSTCFLAGS_bpf_load.o += -I$(objtree)/usr/include -I/usr/src/SandFS-
     Kernel/tools/perf -I/usr/src/SandFS-Kernel/tools/include -Wno-unused-variable
6    HOSTCFLAGS_libbpf.o += -I$(objtree)/usr/include -Wno-unused-variable
7    HOSTLOADLIBES_sandfs += -lelf -lcap -lpthread
```

```
go@ubuntu:~/Desktop/LibSandFS$ ./sandfs
Specify file systen dir to sandbox
Usage: ./sandfs [options] app [arg...]

Executes a shell command under a sandbox.

Options can be:

    -d          File system dir to sandbox
    -s          BPF sandboxing program file path
    -v          Display verbose messages
```

new error

```
1    fd = bpf_load_program(prog_type, prog, insns_cnt, license, kern_version,
2                 bpf_log_buf, BPF_LOG_BUF_SIZE);
3    if (fd < 0) {
4        printf("bpf_load_program() err=%d\n%s", errno, bpf_log_buf);
5        return -1;
6    }
7
8    prog_fd[prog_cnt++] = fd;
```



```
$ sandfs -s sandfs.o -d /home/user /bin/bash
  Non-root   Security Checks   Sandboxed Directory   Untrusted Application
                eBPF code
```



```
go@ubuntu:~/Desktop/LibSandFS$ sudo ./sandfs  -s bpf/sandfs.o -d /home/go/Desktop/a     bin/sh -v
bpf_load_program() err=22
invalid prog_fd[0]=0 and map_fd[0]=6
Failed to load bpf prog!
```

```
1    if (!prog_fd[0] || !map_fd[0]) {
2        ERROR("invalid prog_fd[0]=%d and map_fd[0]=%d\n",
3            prog_fd[0], map_fd[0]);
4        goto err;
5    }
6  go@ubuntu:~/Desktop/LibSandFS$ sudo ./sandfs  -s bpf/sandfs.o -d /home/go
   /bin/sh
7  bpf_load_program() err=22
8  invalid prog_fd[0]=0 and map_fd[0]=6
9  Failed to load bpf prog!
```

函数调用逻辑分析

```
1  static int load_and_attach(const char *event, struct bpf_insn *prog, int
   size)
2  {   ...
3      fd = bpf_load_program(prog_type, prog, insns_cnt, license, kern_version,
4                   bpf_log_buf, BPF_LOG_BUF_SIZE);
5      if (fd < 0) {
6          printf("bpf_load_program() err=%d\n%s", errno, bpf_log_buf);
7          return -1;
8      }
9
10     prog_fd[prog_cnt++] = fd;
11     ...
```

```
12    }
13    /* load sandboxing bpf program */
14    if (prog) {
15        ctx = ebpf_init(prog);
16        if (!ctx) {
17            fprintf(stderr, "Failed to load bpf prog!\n");
18            exit(EXIT_FAILURE);
19    }
20    ebpf_context_t* ebpf_init(char *filename){
21        ...
22
23        if (!prog_fd[0] || !map_fd[0]) {
24            ERROR("invalid prog_fd[0]=%d and map_fd[0]=%d\n",
25              prog_fd[0], map_fd[0]);
26            goto err;
27        }
28        ...
29    }
```

最终问题还是内核编译问题

https://www.mail-archive.com/netdev@vger.kernel.org/msg165979.html

```
1    cat System.map | grep "sandfs"
2
3    sandfs@ubuntu:/usr/src/SandFS-Kernel$ cat System.map | grep "sandfs"
4    ffffffff81355f50 t sandfs_d_release
5    ffffffff81355fe0 t sandfs_d_revalidate
6    ffffffff81356150 t sandfs_compat_ioctl
7    ffffffff81356190 t sandfs_fasync
8    ffffffff813561d0 T sandfs_read_iter
9    ffffffff81356260 T sandfs_write_iter
10   ffffffff81356320 t sandfs_file_llseek
11   ffffffff81356370 t sandfs_file_release
12   ffffffff813563b0 t sandfs_flush
13   ffffffff81356410 t sandfs_unlocked_ioctl
14   ffffffff81356480 t sandfs_mmap
15   ffffffff81356570 t sandfs_write
16   ffffffff81356730 t sandfs_read
17   ffffffff813568c0 t sandfs_readdir
18   ffffffff81356920 t sandfs_fsync
19   ffffffff813569e0 t sandfs_open
20   ffffffff81356b30 T sandfs_read_dirent
21   ffffffff81356b50 t sandfs_permission
22   ffffffff81356b70 t sandfs_setattr
23   ffffffff81356d30 t sandfs_xattr_get
24   ffffffff81356e10 t sandfs_getattr
25   ffffffff81356ef0 t sandfs_listxattr
26   ffffffff81356fd0 t sandfs_readlink
27   ffffffff813570b0 t sandfs_get_link
28   ffffffff81357170 t sandfs_symlink
29   ffffffff813572b0 t sandfs_create
30   ffffffff81357410 t sandfs_mknod
31   ffffffff81357570 t sandfs_rmdir
32   ffffffff813576c0 t sandfs_mkdir
33   ffffffff81357810 t sandfs_unlink
34   ffffffff813579a0 t sandfs_link
```

```
35  ffffffff81357b70 t sandfs_rename
36  ffffffff81357d10 t sandfs_xattr_set
37  ffffffff81357e80 T sandfs_mount
38  ffffffff81357ed0 T sandfs_parse_options
39  ffffffff81357fe0 T sandfs_read_super
40  ffffffff813582e0 T sandfs_umount_begin
41  ffffffff81358330 T sandfs_remount_fs
42  ffffffff81358360 T sandfs_put_super
43  ffffffff813583b0 t sandfs_evict_inode
44  ffffffff813583f0 t sandfs_destroy_inode
45  ffffffff81358410 t sandfs_alloc_inode
46  ffffffff81358450 t sandfs_fh_to_parent
47  ffffffff81358470 t sandfs_nfs_get_inode
48  ffffffff813584a0 t sandfs_fh_to_dentry
49  ffffffff813584c0 t sandfs_statfs
50  ffffffff81358560 T sandfs_init_inode_cache
51  ffffffff813585a0 T sandfs_destroy_inode_cache
52  ffffffff813585c0 t sandfs_inode_test
53  ffffffff813585e0 t sandfs_inode_set
54  ffffffff813585f0 T sandfs_init_dentry_cache
55  ffffffff81358630 T sandfs_destroy_dentry_cache
56  ffffffff81358650 T sandfs_free_dentry_private_data
57  ffffffff81358690 T sandfs_new_dentry_private_data
58  ffffffff813586d0 T sandfs_iget
59  ffffffff81358870 t __sandfs_interpose.isra.7
60  ffffffff813588c0 t __sandfs_lookup.isra.8
61  ffffffff81358bd0 T sandfs_interpose
62  ffffffff81358bf0 T sandfs_lookup
63  ffffffff81358d40 t sandfs_direct_IO
64  ffffffff81358d60 t sandfs_page_mkwrite.part.1
65  ffffffff81358d70 t sandfs_page_mkwrite
66  ffffffff81358e00 t sandfs_fault.part.2
67  ffffffff81358e10 t sandfs_fault
68  ffffffff81358f50 T bpf_sandfs_read_args
69  ffffffff81359300 T bpf_sandfs_write_args
70  ffffffff813593c0 t sandfs_prog_func_proto
71  ffffffff81359460 t sandfs_prog_is_valid_access
72  ffffffff813594a0 T sandfs_request_bpf_op
73  ffffffff813594e0 T sandfs_clear_bpf_ops
74  ffffffff81359540 T sandfs_set_bpf_ops
75  ffffffff81a51900 R sandfs_dops
76  ffffffff81a519a0 R sandfs_dir_fops
77  ffffffff81a51aa0 R sandfs_main_fops
78  ffffffff81a51bc0 R sandfs_xattr_handler
79  ffffffff81a51c00 R sandfs_main_iops
80  ffffffff81a51cc0 R sandfs_dir_iops
81  ffffffff81a51d80 R sandfs_symlink_iops
82  ffffffff81a51e60 R sandfs_export_ops
83  ffffffff81a51ec0 R sandfs_sops
84  ffffffff81a51fa0 R sandfs_vm_ops
85  ffffffff81a52020 R sandfs_aops
86  ffffffff81a52260 r sandfs_prog_ops
87  ffffffff81a52280 r bpf_sandfs_write_args_proto
88  ffffffff81a522c0 r bpf_sandfs_read_args_proto
89  ffffffff81d59d20 r sandfs_tl
90  ffffffff81ec7760 D sandfs_xattr_handlers
91  ffffffff81ec7780 d sandfs_fs_type
92  ffffffff81ffd24e t init_sandfs_fs
```
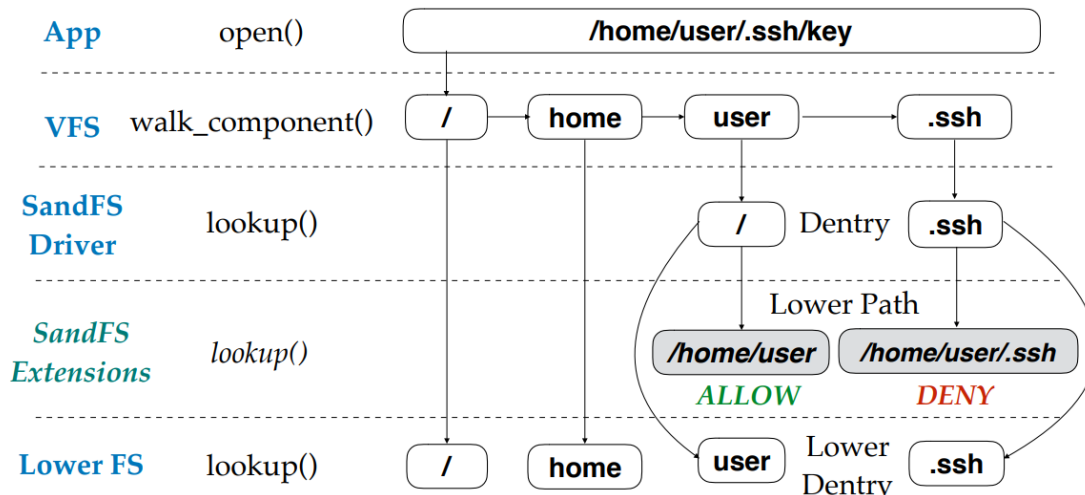
```
93  ffffffff81ffd29c T sandfs_register_bpf_prog_ops
94  ffffffff821a2a80 t __initcall_init_sandfs_fs6
95  ffffffff821c980f t exit_sandfs_fs
96  ffffffff823e0938 b sandfs_inode_cachep
97  ffffffff823e0940 b sandfs_dentry_cachep
```

复现结果

more ideas

```
1  sudo ./sandfs -s bpf/sandfs.o -d /home/sandfs /bin/sh
```

-v 输出

```
1  sandfs@ubuntu:~/Desktop/LibSandFS$  sudo ./sandfs -v  -s bpf/sandfs.o -d
   /home/sandfs  /bin/sh
2
3  === Parent ===
4  UID: 0 EUID: 0
```

```
  5   PID: 10996 TID: 10996
  6   Mount Points:
  7       sysfs /sys
  8       proc /proc
  9       udev /dev
 10       devpts /dev/pts
 11       tmpfs /run
 12       /dev/sda1 /
 13       securityfs /sys/kernel/security
 14       tmpfs /dev/shm
 15       tmpfs /run/lock
 16       tmpfs /sys/fs/cgroup
 17       cgroup /sys/fs/cgroup/systemd
 18       pstore /sys/fs/pstore
 19       cgroup /sys/fs/cgroup/cpu,cpuacct
 20       cgroup /sys/fs/cgroup/net_cls,net_prio
 21       cgroup /sys/fs/cgroup/cpuset
 22       cgroup /sys/fs/cgroup/memory
 23       cgroup /sys/fs/cgroup/blkio
 24       cgroup /sys/fs/cgroup/hugetlb
 25       cgroup /sys/fs/cgroup/perf_event
 26       cgroup /sys/fs/cgroup/pids
 27       cgroup /sys/fs/cgroup/freezer
 28       cgroup /sys/fs/cgroup/devices
 29       systemd-1 /proc/sys/fs/binfmt_misc
 30       debugfs /sys/kernel/debug
 31       mqueue /dev/mqueue
 32       hugetlbfs /dev/hugepages
 33       fusectl /sys/fs/fuse/connections
 34       binfmt_misc /proc/sys/fs/binfmt_misc
 35       vmware-vmblock /run/vmblock-fuse
 36       tmpfs /run/user/1000
 37       gvfsd-fuse /run/user/1000/gvfs
 38   Credentials and Capabilities =
      cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_ki
      ll,cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,cap_net_bind_servi
      ce,cap_net_broadcast,cap_net_admin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,c
      ap_sys_module,cap_sys_rawio,cap_sys_chroot,cap_sys_ptrace,cap_sys_pacct,cap
      _sys_admin,cap_sys_boot,cap_sys_nice,cap_sys_resource,cap_sys_time,cap_sys_
      tty_config,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfca
      p,cap_mac_override,cap_mac_admin,cap_syslog,cap_wake_alarm,cap_block_suspen
      d,37+ep
 39
 40   Loaded sandbox progam bpf/sandfs.o with fd=7
 41
 42   === AFTER CLONE_NEWUSER | CLONE_NEWNS ===
 43   UID: 65534 EUID: 65534
 44   PID: 10996 TID: 10996
 45   Mount Points:
 46       /dev/sda1 /
 47       udev /dev
 48       devpts /dev/pts
 49       tmpfs /dev/shm
 50       mqueue /dev/mqueue
 51       hugetlbfs /dev/hugepages
 52       tmpfs /run
 53       tmpfs /run/lock
 54       vmware-vmblock /run/vmblock-fuse
```

```
55      tmpfs /run/user/1000
56      gvfsd-fuse /run/user/1000/gvfs
57      sysfs /sys
58      securityfs /sys/kernel/security
59      tmpfs /sys/fs/cgroup
60      cgroup /sys/fs/cgroup/systemd
61      cgroup /sys/fs/cgroup/cpu,cpuacct
62      cgroup /sys/fs/cgroup/net_cls,net_prio
63      cgroup /sys/fs/cgroup/cpuset
64      cgroup /sys/fs/cgroup/memory
65      cgroup /sys/fs/cgroup/blkio
66      cgroup /sys/fs/cgroup/hugetlb
67      cgroup /sys/fs/cgroup/perf_event
68      cgroup /sys/fs/cgroup/pids
69      cgroup /sys/fs/cgroup/freezer
70      cgroup /sys/fs/cgroup/devices
71      pstore /sys/fs/pstore
72      debugfs /sys/kernel/debug
73      fusectl /sys/fs/fuse/connections
74      proc /proc
75      systemd-1 /proc/sys/fs/binfmt_misc
76      binfmt_misc /proc/sys/fs/binfmt_misc
77  Credentials and Capabilities =
    cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_ki
    ll,cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,cap_net_bind_servi
    ce,cap_net_broadcast,cap_net_admin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,c
    ap_sys_module,cap_sys_rawio,cap_sys_chroot,cap_sys_ptrace,cap_sys_pacct,cap
    _sys_admin,cap_sys_boot,cap_sys_nice,cap_sys_resource,cap_sys_time,cap_sys_
    tty_config,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfca
    p,cap_mac_override,cap_mac_admin,cap_syslog,cap_wake_alarm,cap_block_suspen
    d,37+ep
78
79
80  === Mounted Sandfs ===
81  UID: 65534 EUID: 65534
82  PID: 10996 TID: 10996
83  Mount Points:
84      /dev/sda1 /
85      udev /dev
86      devpts /dev/pts
87      tmpfs /dev/shm
88      mqueue /dev/mqueue
89      hugetlbfs /dev/hugepages
90      tmpfs /run
91      tmpfs /run/lock
92      vmware-vmblock /run/vmblock-fuse
93      tmpfs /run/user/1000
94      gvfsd-fuse /run/user/1000/gvfs
95      sysfs /sys
96      securityfs /sys/kernel/security
97      tmpfs /sys/fs/cgroup
98      cgroup /sys/fs/cgroup/systemd
99      cgroup /sys/fs/cgroup/cpu,cpuacct
100     cgroup /sys/fs/cgroup/net_cls,net_prio
101     cgroup /sys/fs/cgroup/cpuset
102     cgroup /sys/fs/cgroup/memory
103     cgroup /sys/fs/cgroup/blkio
104     cgroup /sys/fs/cgroup/hugetlb
```

```
105        cgroup /sys/fs/cgroup/perf_event
106        cgroup /sys/fs/cgroup/pids
107        cgroup /sys/fs/cgroup/freezer
108        cgroup /sys/fs/cgroup/devices
109        pstore /sys/fs/pstore
110        debugfs /sys/kernel/debug
111        fusectl /sys/fs/fuse/connections
112        proc /proc
113        systemd-1 /proc/sys/fs/binfmt_misc
114        binfmt_misc /proc/sys/fs/binfmt_misc
115        /home/sandfs /home/sandfs
116    Credentials and Capabilities =
       cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_ki
       ll,cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,cap_net_bind_servi
       ce,cap_net_broadcast,cap_net_admin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,c
       ap_sys_module,cap_sys_rawio,cap_sys_chroot,cap_sys_ptrace,cap_sys_pacct,cap
       _sys_admin,cap_sys_boot,cap_sys_nice,cap_sys_resource,cap_sys_time,cap_sys_
       tty_config,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfca
       p,cap_mac_override,cap_mac_admin,cap_syslog,cap_wake_alarm,cap_block_suspen
       d,37+ep
117
118    PARENT ./sandfs: PID of child created by clone() is 10997
119
120    === Child Process : Before reset_caps ===
121    UID: 65534 EUID: 65534
122    PID: 10997 TID: 10997
123    Mount Points:
124        /dev/sda1 /
125        udev /dev
126        devpts /dev/pts
127        tmpfs /dev/shm
128        mqueue /dev/mqueue
129        hugetlbfs /dev/hugepages
130        tmpfs /run
131        tmpfs /run/lock
132        vmware-vmblock /run/vmblock-fuse
133        tmpfs /run/user/1000
134        gvfsd-fuse /run/user/1000/gvfs
135        sysfs /sys
136        securityfs /sys/kernel/security
137        tmpfs /sys/fs/cgroup
138        cgroup /sys/fs/cgroup/systemd
139        cgroup /sys/fs/cgroup/cpu,cpuacct
140        cgroup /sys/fs/cgroup/net_cls,net_prio
141        cgroup /sys/fs/cgroup/cpuset
142        cgroup /sys/fs/cgroup/memory
143        cgroup /sys/fs/cgroup/blkio
144        cgroup /sys/fs/cgroup/hugetlb
145        cgroup /sys/fs/cgroup/perf_event
146        cgroup /sys/fs/cgroup/pids
147        cgroup /sys/fs/cgroup/freezer
148        cgroup /sys/fs/cgroup/devices
149        pstore /sys/fs/pstore
150        debugfs /sys/kernel/debug
151        fusectl /sys/fs/fuse/connections
152        proc /proc
153        systemd-1 /proc/sys/fs/binfmt_misc
154        binfmt_misc /proc/sys/fs/binfmt_misc
```

```
155        /home/sandfs /home/sandfs
156    Credentials and Capabilities =
       cap_chown,cap_dac_override,cap_dac_read_search,cap_fowner,cap_fsetid,cap_ki
       ll,cap_setgid,cap_setuid,cap_setpcap,cap_linux_immutable,cap_net_bind_servi
       ce,cap_net_broadcast,cap_net_admin,cap_net_raw,cap_ipc_lock,cap_ipc_owner,c
       ap_sys_module,cap_sys_rawio,cap_sys_chroot,cap_sys_ptrace,cap_sys_pacct,cap
       _sys_admin,cap_sys_boot,cap_sys_nice,cap_sys_resource,cap_sys_time,cap_sys_
       tty_config,cap_mknod,cap_lease,cap_audit_write,cap_audit_control,cap_setfca
       p,cap_mac_override,cap_mac_admin,cap_syslog,cap_wake_alarm,cap_block_suspen
       d,37+ep
157
158
159    === Child Process : After reset_caps ===
160    UID: 65534 EUID: 65534
161    PID: 10997 TID: 10997
162    Mount Points:
163        /dev/sda1 /
164        udev /dev
165        devpts /dev/pts
166        tmpfs /dev/shm
167        mqueue /dev/mqueue
168        hugetlbfs /dev/hugepages
169        tmpfs /run
170        tmpfs /run/lock
171        vmware-vmblock /run/vmblock-fuse
172        tmpfs /run/user/1000
173        gvfsd-fuse /run/user/1000/gvfs
174        sysfs /sys
175        securityfs /sys/kernel/security
176        tmpfs /sys/fs/cgroup
177        cgroup /sys/fs/cgroup/systemd
178        cgroup /sys/fs/cgroup/cpu,cpuacct
179        cgroup /sys/fs/cgroup/net_cls,net_prio
180        cgroup /sys/fs/cgroup/cpuset
181        cgroup /sys/fs/cgroup/memory
182        cgroup /sys/fs/cgroup/blkio
183        cgroup /sys/fs/cgroup/hugetlb
184        cgroup /sys/fs/cgroup/perf_event
185        cgroup /sys/fs/cgroup/pids
186        cgroup /sys/fs/cgroup/freezer
187        cgroup /sys/fs/cgroup/devices
188        pstore /sys/fs/pstore
189        debugfs /sys/kernel/debug
190        fusectl /sys/fs/fuse/connections
191        proc /proc
192        systemd-1 /proc/sys/fs/binfmt_misc
193        binfmt_misc /proc/sys/fs/binfmt_misc
194        /home/sandfs /home/sandfs
195    Credentials and Capabilities =
196
197    $
```

https://blog.csdn.net/dog250/article/details/104028841
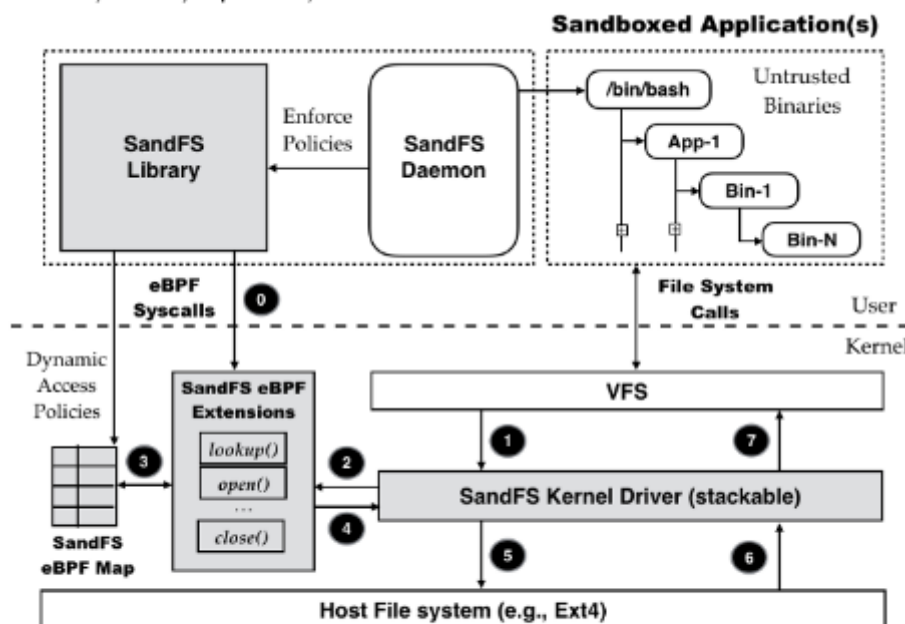
sandfs与sandbox application 进程通信

Figure 2: Overview of SandFS framework. The components introduced have been highlighted in grey.

```
1  sandfs@ubuntu:~/Desktop/LibSandFS$ ps all
2  F    UID    PID    PPID PRI  NI    VSZ     RSS WCHAN    STAT TTY        TIME
   COMMAND
3  4     0   1110       1  20   0   23012   1824 -        Ss+  tty1       0:00
   /sbin/age
4  4     0   1254    1206  20   0  482480  87176 -        Ssl+ tty7       0:31
   /usr/lib/
5  0  1000   2434    2427  20   0   29720   5404 wait     Ss   pts/2      0:00 bash
6  0  1000   4654    2427  20   0   29708   5288 wait_w   Ss+  pts/11     0:00 bash
7  4     0  10105    2434  20   0   61868   3932 -        S    pts/2      0:00 sudo
   ./sa
8  4     0  10106   10105  20   0   18696    904 -        S    pts/2      0:00
   ./sandfs
9  0     0  10107   10106  20   0   29692   5224 -        S+   pts/2      0:00
   /bin/bash
10 0  1000  10123    2427  20   0   29708   5156 wait     Ss   pts/19     0:00 bash
11 4     0  10135   10123  20   0   62100   3900 -        S    pts/19     0:00 sudo
   ./sa
12 4     0  10136   10135  20   0   18696    928 -        S    pts/19     0:00
   ./sandfs
13 0     0  10137   10136  20   0    4512    848 -        S+   pts/19     0:00
   /bin/sh
14 0  1000  10167    2427  20   0   29708   5160 wait_w   Ss+  pts/21     0:00 bash
15 4     0  35580   11071  20   0   71412  42292 -        Ss+  pts/22     0:05
   /usr/bin/
16 0  1000  36581    2427  20   0   29700   5300 wait     Ss   pts/20     0:00 bash
17 0     0  36697   35580  20   0    4504    736 -        S+   pts/22     0:00
   /bin/sh /
18 0  1000  36698   36581  20   0   35984   1528 -        R+   pts/20     0:00 ps
   all
```

在上面的sanfds与/bin/bash进程中可以看出，sandfs作为父进程，/bin/bash作为子进程。

使用管道来对于父进程与子进程进行同步，

```
1    /* We use a pipe to synchronize the parent and child, in order to
2       ensure that the parent sets the UID and GID maps before the child
3       calls execve(). This ensures that the child maintains its
4       capabilities during the execve() in the common case where we
5       want to map the child's effective user ID to 0 in the new user
6       namespace. Without this synchronization, the child would lose
7       its capabilities if it performed an execve() with nonzero
8       user IDs (see the capabilities(7) man page for details of the
9       transformation of a process's capabilities during execve()). */
```

dmesg

列出加载到内核中的驱动

```
1   dmesg  | grep  "sandfs"
2   [   795.901313] ****************************************************************
3   [   795.901314] **    NOTICE NOTICE NOTICE NOTICE NOTICE NOTICE NOTICE    **
4   [   795.901314] **                                                       **
5   [   795.901314] ** trace_printk() being used. Allocating extra memory.   **
6   [   795.901315] **                                                       **
7   [   795.901315] ** This means that this is a DEBUG kernel and it is      **
8   [   795.901315] ** unsafe for production use.                            **
9   [   795.901315] **                                                       **
10  [   795.901316] ** If you see this message and you are not debugging     **
11  [   795.901316] ** the kernel, report this immediately to your vendor!   **
12  [   795.901316] **                                                       **
13  [   795.901316] **    NOTICE NOTICE NOTICE NOTICE NOTICE NOTICE NOTICE    **
14  [   795.901317] ****************************************************************
15  [   795.912339] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
16  [   795.912365] Sandfs: option fd:7
17  [   795.912366] Sandfs prog_fd=7
18  [   795.912584] Setting SANDFS bpf ops
19  [   795.912585] SANDFS bpf program updated
20  [   795.912587] sandfs: mounted on top of /home/sandfs type ext4
21  [ 1138.512716] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
22  [ 1138.512720] Sandfs: option fd:7
23  [ 1138.512721] Sandfs prog_fd=7
24  [ 1138.512927] Setting SANDFS bpf ops
25  [ 1138.512928] SANDFS bpf program updated
26  [ 1138.512929] sandfs: mounted on top of /home/sandfs type ext4
27  [ 1241.042575] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
28  [ 1241.042578] Sandfs: option fd:7
29  [ 1241.042579] Sandfs prog_fd=7
30  [ 1241.042582] Setting SANDFS bpf ops
31  [ 1241.042583] SANDFS bpf program updated
32  [ 1241.042584] sandfs: mounted on top of /home/sandfs type ext4
33  sandfs@ubuntu:~$ dmesg | grep "andfs"
34  [ 2654.952870] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
35  [ 2654.952881] Sandfs: option fd:7
36  [ 2654.952888] Sandfs prog_fd=7
37  [ 2654.952911] sandfs: mounted on top of /home/sandfs type ext4
38  [ 2668.829781] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
```

```
39  [ 2668.829792] Sandfs: option fd:7
40  [ 2668.829800] Sandfs prog_fd=7
41  [ 2668.829859] sandfs: mounted on top of /home/sandfs type ext4
42  [ 2674.352095] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
43  [ 2674.352105] Sandfs: option fd:7
44  [ 2674.352113] Sandfs prog_fd=7
45  [ 2674.352137] sandfs: mounted on top of /home/sandfs type ext4
46  [ 2679.289700] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
47  [ 2679.289731] Sandfs: option fd:7
48  [ 2679.289739] Sandfs prog_fd=7
49  [ 2679.289762] sandfs: mounted on top of /home/sandfs type ext4
50  [ 2685.065565] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
51  [ 2685.065576] Sandfs: option fd:7
52  [ 2685.065583] Sandfs prog_fd=7
53  [ 2685.065606] sandfs: mounted on top of /home/sandfs type ext4
54  [ 2691.251265] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
55  [ 2691.251275] Sandfs: option fd:7
56  [ 2691.251282] Sandfs prog_fd=7
57  [ 2691.251433] sandfs: mounted on top of /home/sandfs type ext4
58  [ 2733.228056] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
59  [ 2733.228067] Sandfs: option fd:7
60  [ 2733.228074] Sandfs prog_fd=7
61  [ 2733.228098] sandfs: mounted on top of /home/sandfs type ext4
62  [ 2874.241281] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
63  [ 2874.241292] Sandfs: option fd:7
64  [ 2874.241299] Sandfs prog_fd=7
65  [ 2874.241322] sandfs: mounted on top of /home/sandfs type ext4
66  [ 2920.103408] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
67  [ 2920.103419] Sandfs: option fd:7
68  [ 2920.103426] Sandfs prog_fd=7
69  [ 2920.103449] sandfs: mounted on top of /home/sandfs type ext4
70  [ 3108.623858] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
71  [ 3108.623876] Sandfs: option fd:7
72  [ 3108.623888] Sandfs prog_fd=7
73  [ 3108.623924] sandfs: mounted on top of /home/sandfs type ext4
74  [ 3349.289914] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
75  [ 3349.289933] Sandfs: option fd:7
76  [ 3349.289945] Sandfs prog_fd=7
77  [ 3349.289985] sandfs: mounted on top of /home/sandfs type ext4
78  [ 3813.707688] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
79  [ 3813.707706] Sandfs: option fd:7
80  [ 3813.707718] Sandfs prog_fd=7
81  [ 3813.707759] sandfs: mounted on top of /home/sandfs type ext4
82  [ 4164.049007] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
83  [ 4164.049024] Sandfs: option fd:7
84  [ 4164.049035] Sandfs prog_fd=7
85  [ 4164.049099] sandfs: mounted on top of /home/sandfs type ext4
86  [ 4458.575881] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
87  [ 4458.575892] Sandfs: option fd:7
88  [ 4458.575899] Sandfs prog_fd=7
89  [ 4458.575974] sandfs: mounted on top of /home/sandfs type ext4
90  [ 4797.491475] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
91  [ 4797.491492] Sandfs: option fd:7
92  [ 4797.491504] Sandfs prog_fd=7
93  [ 4797.491575] sandfs: mounted on top of /home/sandfs type ext4
94  [ 5730.343941] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
95  [ 5730.343957] Sandfs: option fd:7
96  [ 5730.343969] Sandfs prog_fd=7
```

```
 97  [ 5730.344006] sandfs: mounted on top of /home/sandfs type ext4
 98  [ 6448.568649] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
 99  [ 6448.568659] Sandfs: option fd:7
100  [ 6448.568666] Sandfs prog_fd=7
101  [ 6448.568689] sandfs: mounted on top of /home/sandfs type ext4
102  [ 6811.377367] Sandfs: request to mount on /home/sandfs with prog_fd=fd=7
103  [ 6811.377393] Sandfs: option fd:7
104  [ 6811.377402] Sandfs prog_fd=7
105  [ 6811.377497] sandfs: mounted on top of /home/sandfs type ext4
```