操作类型

标志性 log: `avc: denied { 操作权限 } for pid=7201 comm="进程名" scontext=u:r:源类型:s0 tcontext=u:r:目标类型:s0 tclass=访问类型 permissive=0`

源类型：授予访问的类型，通常是进程的域类型，

目标类型：客体的类型，它被授权可以访问的类型

操作权限：表示主体对客体访问时允许的操作类型（也叫做访问向量）。

关键文件

政策文件

以 `*.te` 结尾的文件是 SELinux 政策源代码文件，用于定义 域及其标签。

其中 file.te , property.te 文件比较特殊，为自定义 **目标类型**

上下文的描述文件

在上下文的描述文件中为对象指定标签。

`file_contexts` 用于为文件分配标签，并且可供多种用户空间组件使用。 一般为 sys 文件系统分配标签

`genfs_contexts` 用于为不支持扩展属性的文件系统（例如，`proc` 或 `vfat`）分配标签也可为sysfs分配。

配置文件MK

```
在 /android/device/mediatek/system/common/BoardConfig.mk中
    ifeq ($(strip $(MTK_BSP_PACKAGE)), yes)
BOARD_SEPOLICY_DIRS := \
      device/mediatek/sepolicy/basic/non_plat \
      device/mediatek/sepolicy/bsp/non_plat
BOARD_PLAT_PUBLIC_SEPOLICY_DIR := \
      device/mediatek/sepolicy/basic/plat_public \
      device/mediatek/sepolicy/bsp/plat_public
BOARD_PLAT_PRIVATE_SEPOLICY_DIR := \
      device/mediatek/sepolicy/basic/plat_private \
      device/mediatek/sepolicy/bsp/plat_private

修改或添加政策文件和上下文的描述文件后，更新 BoardConfig.mk makefile 以引用 sepolicy 子目录和每个新的政策文件。注意的是里面有basic, bsp目录.
其中basic 目录所有的版本都会用到到; bsp 目录则是bsp 版本 + Turnkey 版本会用到到;
```

策略添加

一般问题

报错格式

`avc: denied { 操作权限 } for pid=7201 comm="进程名" scontext=u:r:源类型:s0 tcontext=u:r:目标类型:s0 tclass=访问类型 permissive=0`

解决措施

在源类型.te文件中添加：

```
 allow  源类型 目标类型:访问类型 {操作权限};
```

示例

```
avc: denied { read } for name="sar_state" dev="sysfs" ino=42139
scontext=u:r:system_server:s0 tcontext=u:object_r:sysfs_prudoct_info:s0
tclass=file permissive=0
```

```
//在 system_server.te 中
allow  system_server  sysfs_prudoct_info  file  {open} ;
```

neverallow问题

google不允许某些源类型访问目标类型

报错信息

violated by allow 源类型 目标类型 :访问类型{操作权限}; neverallow failures occurred.

解决措施

更改目标类型

```
//1.在 file.te 定义新的目标标签，即上文提到的政策文件
type mytest_file , fs_type, sysfs_type, mlstrustedobject;
//2.1 在 file_contexts 为对象指定标签，即上文提到的描述文件   sysfs
路径  u:object_r : mytest_file:s0
//2.2 在 genfs_contexts 为对象指定标签，即上文提到的描述文件   procfs
genfscon proc   /subdir  u:object_r:proc_freqhopping:s0 //subdir 为去掉proc的节点路径
genfscon sysyfs /subdir  u:object_r:proc_freqhopping:s0 //subdir 为去掉sysfs的节点路径
//在源类型.te文件中添加权限
allow  源类型 目标类型:访问类型 {操作权限};
//更新init.project.rc
chmod 0666 路径
chown root system 路径
```

示例

```
//添加的SELinux权限违反了google的neverallow规则，google不允许名为radio源类型访问标签名为vendor_data_file的目标类型
violated by allow radio vendor data file:file {create setattr lock unlink rename open}
violated by allow radio vendor data file:dir  {ioctl read write lock add_name open}
2 neverallow failures occurred
Error while expanding policy

// 1.file.te中添加
type tct_time_data_file, file_type, data_file_type, mlstrustedobject;
//2.file_contexts中
/data/vendor/time_code(/.*)?       u:object_r:tct_time_data_file:s0 //操作目录中所有文件
//3.radio.te中
allow radio tct_time_data_file:dir rw_dir_perms;
allow radio tct_time_data_file:file { open write read create rw_file_perms setattr};
allow radio tct_time_data_file:file rename;
allow radio tct_time_data_file:file unlink;

ls -Z 可以看到/data/vendor/time_code 下所有文件被更改为tct_time_data_file 类型
```