1. 需要编译userdebug版本
2. 关闭selinux
   device/rockchip/common

```
diff --git a/BoardConfig.mk b/BoardConfig.mk
index 3706d7b7..8099919f 100755
--- a/BoardConfig.mk
+++ b/BoardConfig.mk
@@ -59,7 +59,7 @@ BOARD_BOOT_HEADER_VERSION ?= 2
 BOARD_MKBOOTIMG_ARGS :=
 BOARD_PREBUILT_DTBOIMAGE ?= $(TARGET_DEVICE_DIR)/dtbo.img
 BOARD_ROCKCHIP_VIRTUAL_AB_ENABLE ?= false
-BOARD_SELINUX_ENFORCING ?= true
+BOARD_SELINUX_ENFORCING ?= false
```

3. 修改su.cpp，注释用户组权限检测
   system/extras/su/su.cpp

```
diff --git a/su/su.cpp b/su/su.cpp
index 1a1ab6bf..af3d2a68 100644
--- a/su/su.cpp
+++ b/su/su.cpp
@@ -80,8 +80,8 @@ void extract_uidgids(const char* uidgids, uid_t* uid, gid_t*
gid, gid_t* gids, i
 }

 int main(int argc, char** argv) {
-    uid_t current_uid = getuid();
-    if (current_uid != AID_ROOT && current_uid != AID_SHELL) error(1, 0, "not
allowed");
+    //uid_t current_uid = getuid();
+    //if (current_uid != AID_ROOT && current_uid != AID_SHELL) error(1, 0, "not
allowed");

     // Handle -h and --help.
     ++argv;
```

4. 给 su 文件默认授予 root 权限
   system/core/libcutils/fs_config.cpp

```
diff --git a/libcutils/fs_config.cpp b/libcutils/fs_config.cpp
index 5805a4d19..92e93e76f 100644
--- a/libcutils/fs_config.cpp
+++ b/libcutils/fs_config.cpp
@@ -188,7 +188,7 @@ static const struct fs_path_config android_files[] = {
     // the following two files are INTENTIONALLY set-uid, but they
     // are NOT included on user builds.
     { 06755, AID_ROOT,      AID_ROOT,      0, "system/xbin/procmem" },
-    { 04750, AID_ROOT,      AID_SHELL,     0, "system/xbin/su" },
+    { 06755, AID_ROOT,      AID_SHELL,     0, "system/xbin/su" },

     // the following files have enhanced capabilities and ARE included
     // in user builds.
```

frameworks/base/core/jni/com_android_internal_os_Zygote.cpp

```
diff --git a/core/jni/com_android_internal_os_Zygote.cpp
b/core/jni/com_android_internal_os_Zygote.cpp
index 9eede83e21e5..694eec2a40ac 100644
--- a/core/jni/com_android_internal_os_Zygote.cpp
+++ b/core/jni/com_android_internal_os_Zygote.cpp
@@ -656,6 +656,7 @@ static void EnableKeepCapabilities(fail_fn_t fail_fn) {
 }

 static void DropCapabilitiesBoundingSet(fail_fn_t fail_fn) {
+/*
   for (int i = 0; prctl(PR_CAPBSET_READ, i, 0, 0, 0) >= 0; i++) {;
     if (prctl(PR_CAPBSET_DROP, i, 0, 0, 0) == -1) {
       if (errno == EINVAL) {
@@ -666,6 +667,7 @@ static void DropCapabilitiesBoundingSet(fail_fn_t fail_fn) {
     }
   }
+  */
 }
```

kernel/security/commoncap.c

```
diff --git a/security/commoncap.c b/security/commoncap.c
index f86557a8e43f6..19124dd6239a1 100644
--- a/security/commoncap.c
+++ b/security/commoncap.c
@@ -1147,12 +1147,12 @@ int cap_task_setnice(struct task_struct *p, int nice)
 static int cap_prctl_drop(unsigned long cap)
 {
        struct cred *new;
-
+/*
        if (!ns_capable(current_user_ns(), CAP_SETPCAP))
                return -EPERM;
        if (!cap_valid(cap))
                return -EINVAL;
-
+*/
        new = prepare_creds();
```

```
        if (!new)
                return -ENOMEM;
```