# Cores

Roxo - #7367F0

Bg claro #eeedfd

bg escuro #31375a

cinza #515151

Verde escuro #335600

Verde claro #618729

bg apps day – FFf8f7fa

bg apps night - FF161d31

Replace

# grep -rl "Lineage" ./ | xargs sed -i 's/Lineage/PortalRom/g'

Permissoes a pasta

sudo chmod -R 777 /Portal/

**Limpar referencias erradas no repo sync**

rm -rf .repo/project-objects/portalromOS/android\_frameworks\_base.git

rm -rf .repo/projects/frameworks/base.git

rm -rf frameworks/base/.git

rm -rf .repo/projects/hardware/lineage/interfaces.git

**~~Forcar Adb authorized para o device que está bloqueado~~**

~~- Entrar em recovery mode~~

~~- Advanced / Enable adb~~

~~- Na linha de comandos~~

**~~Windows – cd C:\Users\rita\_\.android>~~**

**~~Linux – cd ~/.android~~**

~~adb push adbkey.pub /data/misc/adb/adb\_keys~~

~~adb shell reboot~~

# ROM Fork

1. A base da ROM é a lineage 19.1
2. O que estou a fazer é forks dos repositorios do GIT do lineage e a altera-los de forma a estarem de acordo com a nova rom
   1. Em primeiro lugar escolho o repositorio do lineage que quero
   2. Faço fork para a conta da Retrai e mudo-lhe o nome
   3. Mudo o nome do Brach para PROM-1.0
   4. Vou ao GitKraken e faço o clone deste repositorio para uma pasta de trabalho (não é a pasta da rom)
   5. Mudo os manifestos de acordo com o que é necessário, por exemplo
      1. No repositorio base do android mudei a pasta do vendor/lineage para vendor/portalrom
   6. Nota: para ja vou mudar os que forem necessários para fazer o primeiro build, mas o objectivo é mudar todos.

## Repo

* Para fazer sync do repo novo que criei

cd /portal\_rom/PortalRom

mkdir -p ~/.bin

PATH="${HOME}/.bin:${PATH}"

curl https://storage.googleapis.com/git-repo-downloads/repo > ~/.bin/repo

chmod a+rx ~/.bin/repo

repo version

~~old(repo init -u https://github.com/ReTRAI/PortalRom\_android.git -b PROM-1.0)~~

repo init -u https://github.com/portalromOS/PortalRom\_android.git -b PROM-1.0

repo sync (1ª vez)

repo sync -c --force-sync (depois)

se o repo ficar preso -> repo –-trace sync -j1 -c --force-sync

source build/envsetup.sh

brunch taimen

Apaguei a pasta do vendor/lineage e criei um repositorio vendor/portalrom que vai ter os ficheiros todos.

* 1. Já mudei os configs e os .mk files todos que chamam a pasta vendor/lineage dentro da pasta do vendor
     1. Ainda é preciso mudar todas as referencias do lineage que existam nos ficheiros
     2. Ainda é preciso validar, fora da pasta vendor, se há chamadas aos ficheiros vendor/lineage.

1. Forks que já fiz
   1. android -> PortalRom\_android
   2. android/build -> PortalRom\_android\_build
   3. android/vendor/lineage -> android\_vendor\_portalrom
   4. andoid/device/google/taimen -> android\_device\_google\_taimen
   5. andoid/device/google/wahoo -> android\_device\_google\_wahoo
   6. andoid/kernel/google/wahoo -> android\_kernel\_google\_wahoo
   7. proprietary\_vendor\_google -> proprietary\_vendor\_google
   8. android\_packages\_apps\_Settings -> android\_packages\_apps\_Settings

# Parametrização novo device

Para por o device, tentei

Fazer fork do device/google/taimen

Alterar o lineage\_taimen.mk para vendor/portalrom

Fiz push das alteracoes

Coloquei a referencia deste package no ficheiro /Portal/AOSP/PortalRom/PortalRom\_android/snippets/portal

Fiz --force-sync do repositorio novamente

Vou ver se ficou o device no sitio certo – ficou o device/google/taimen

Coloquei os ficheiros proprietarios da google e qcom na pasta vendor

* [https://developers.google.com/android/drivers#taimenrp1a.201005.004.a1](https://developers.google.com/android/drivers" \l "taimenrp1a.201005.004.a1)
* Download dos zip
* Extract numa pasta qualquer
* Correr os .sh (sh nomeficheiro.sh)
* Colocar as pastas vendor na raiz do Portal/AOSP/android

Depois do sync fiz um breakfast taimen e acho que deu erro.

Criou um roomservice.xml no .repo/localmanifest e agora dá erro de device duplicado

Vou a pasta PortalRom\_android e vou tirar a referencia do device/vendor/taimen /Portal/AOSP/PortalRom/PortalRom\_android/snippets/portal e vou colocar o roomservice.xml correcto

Alem do taimen é preciso o wahoo (unified kernel for pixel2 e pixel2xl)

<?xml version="1.0" encoding="UTF-8"?>

<manifest>

<project path="device/google/taimen" remote="github" name="ReTRAI/android\_device\_google\_taimen" revision="PROM-1.0"/>

<project path="device/google/wahoo" remote="github" name="ReTRAI/android\_device\_google\_wahoo" revision="PROM-1.0"/>

<project path="kernel/google/wahoo" remote="github" name="ReTRAI/android\_kernel\_google\_wahoo" revision="PROM-1.0"/>

</manifest>

Fiz fork do android\_device\_google\_wahoo e android\_kernel\_google\_wahoo tb e mudei o .mk que tinha vendor/lineage para vendor/portalrom

Depois fiz o repo init e o repo force sync

Ainda nao aparece como opcao para fazer build

Tentei fazer source build/envsetup.sh + brunch taimen, mas está a aparecer uns erros

1. vendor/qcom/taimen/device-partial.mk:34: error: \_nic.PRODUCTS.[[device/google/taimen/lineage\_taimen.mk]]: "vendor/google/taimen/taimen-vendor.mk" does not exist.
   1. Fiz fork ao <https://github.com/TheMuppets/proprietary_vendor_google> para ter a tree dos dispositivos da google
   2. Coloquei este repositorio no /Portal/AOSP/PortalRom/PortalRom\_android/snippets/portal
   3. Fiz push da alteracao e --force-sync

Depois disto fiz outra vez build/envsetup.sh + brunch taimen, deu erro

1. build/make/core/base\_rules.mk:335: error: vendor/qcom/taimen/proprietary: MODULE.TARGET.APPS.ims already defined by vendor/google/taimen.
   1. Tirei a pasta vendor/qcom/taimen que tinha adicionado antes

Depois disto fiz outra vez build/envsetup.sh + brunch taimen, deu erro

1. /Portal/AOSP/PortalRom/PortalRom/kernel/google/wahoo/scripts/basic/fixdep.c:106:10: fatal error: 'sys/types.h' file

not found

#include <sys/types.h>

^~~~~~~~~~~~~

1 error generated.

make[2]: \*\*\* [scripts/Makefile.host:91: scripts/basic/fixdep] Error 1

make[1]: \*\*\* [/Portal/AOSP/PortalRom/PortalRom/kernel/google/wahoo/Makefile:442: scripts\_basic] Error 2

make[1]: Leaving directory '/Portal/AOSP/PortalRom/PortalRom/out/target/product/taimen/obj/KERNEL\_OBJ'

make: \*\*\* [Makefile:152: sub-make] Error 2

make: Leaving directory '/Portal/AOSP/PortalRom/PortalRom/kernel/google/wahoo'

15:52:19 ninja failed with: exit status 1

* 1. Vi um post a dizer para instalar as bibliotecas libc6-dev e gcc-multilib para fazer build do kernel
     1. sudo apt-get install -y libc6-dev
     2. sudo apt-get install -y gcc-multilib

Depois disto fiz outra vez build/envsetup.sh + brunch taimen, deu erro

1. kernel/google/wahoo/scripts/extract-cert.c:21:10: fatal error: 'openssl/bio.h' file not found
   1. Vi um post a dizer para instalar as bibliotecas libssl-devpara fazer build do kernel
      1. sudo apt-get install -y libssl-dev

Depois disto fiz outra vez build/envsetup.sh + brunch taimen – Build terminou com sucesso

# Para instalar:

Na pasta /out/target/product/taimen estão os ficheiros do lineage:

* unlock bootloader
  + Settings – USB Debugging and OEM unlocking on
  + Entrar em fastboot mode - adb reboot bootloader
  + E correr
    - fastboot flashing unlock, ou
    - fastboot flashing unlock\_critical (para o pixel2xl), ou
    - fastboot oem unlock
* boot.img - which is the LineageOS boot image, and contains the recovery-ramdisk.
  + Instalar o boot.img
    - adb reboot bootloader
    - cd /portal\_rom/PortalRom/out/target/product/taimen/
    - fastboot flash boot boot.img
    - usar o menu do fastboot para entrar em recovery mode
* lineage-19.1-20220914-UNOFFICIAL-taimen.zip (vai criar uma nova a cada dia, nao faz replace) - which is the LineageOS installer package
  + instalar o .zip
    - No menu do recovery
    - Escolher factory reset -> Format data / factory reset
    - Depois entrar no Apply Update -> Apply from ADB
    - cd /portal\_rom/PortalRom/out/target/product/taimen/
    - adb sideload portalrom-19.1-20220914-UNOFFICIAL-taimen.zip
  + quando acabar voltar atras e fazer reboot system now

Obter o jar do AOSP

* No projeto Android da aplicação criar uma pasta dentro da folder “app” com o nome “sdk”
* Dentro dessa pasta colocar todos os .jar do android necessário
  + Ficam todos dentro da pasta out
  + Por exemplo o .jar do framework fica em /portal\_rom/PortalRom/out/target/common/obj/JAVA\_LIBRARIES/framework\_intermediates
  + O ficheiro necessário normalmente é o classes-header.jar
* Depois de ter a pasta com os paths criados, editar o ficheiro build gradle e colocar

gradle.projectsEvaluated {

tasks.withType(JavaCompile) {

Set<File> fileSet = options.bootstrapClasspath.getFiles()

List<File> newFileList = new ArrayList<>();

newFileList.add(new File(“./app/sdk/framework\_intermediates/classes-header.jar”)) - adicionar tantas linhas destas quanto necessário

newFileList.addAll(fileSet)

options.bootstrapClasspath = files(newFileList.toArray())

}

}

* Ainda no gradle adicionar a linha compileOnly files(‘sdk/framework\_intermediates/classes-header.jar’) nas dependecies do build.gradle, quantas vezes for necessário
* fazer sync now para atualizar o gradle

Correr system app no emulador

* Criar um ficheiro na raiz da pasta da aplicação
  + - cat > installSystem.sh
    - sudo chmod +x installSystem.sh
    - sudo nano installSystem.sh
* Colocar o seguinte código

#!/bin/bash

app\_package="com.portalrom.setupwizard"

dir\_app\_name="SetupWizard"

MAIN\_ACTIVITY="MainActivity"

ADB="adb"

ADB\_SH="$ADB shell"

path\_sysapp="/system/app"

apk\_host="./app/build/outputs/apk/debug/app-debug.apk"

apk\_name=$dir\_app\_name".apk"

apk\_target\_dir="$path\_sysapp/$dir\_app\_name"

apk\_target\_sys="$apk\_target\_dir/$apk\_name"

a=$($ADB shell "pm list packages | grep -i 'portalrom' | wc -l") #count packages with this name

##Delete previous APK

echo "DELETE PREVIOUS APK"

rm -f $apk\_host

##Compile the APK:

echo "COMPILE APK"

./gradlew assembleDebug || exit -1 #exit on failure

#ADB ACTION – RUN BY HAND THE FIRST TIME

#$ADB root 2> /dev/null

#$ADB shell "avbctl disable-verification"

#$ADB reboot 2> /dev/null

##COPY APK

echo "COPY APK"

tmp\_dir=/sdcard/tmp

$ADB root 2> /dev/null

$ADB remount 2> /dev/null

$ADB\_SH "su 0 mount -o rw,remount /system"

$ADB\_SH "chmod 777 /system"

$ADB\_SH "mkdir -p $tmp\_dir" 2> /dev/null

$ADB\_SH "mkdir -p $apk\_target\_dir" 2> /dev/null

$ADB push $apk\_host $tmp\_dir/$apk\_name 2> /dev/null

$ADB\_SH "mv $tmp\_dir/$apk\_name $apk\_target\_sys"

$ADB\_SH "rmdir $tmp\_dir" 2> /dev/null

##Give Permissions

echo "PERMISSIONS"

$ADB\_SH "chmod 755 $apk\_target\_dir"

$ADB\_SH "chmod 644 $apk\_target\_sys"

##Unmount

echo "UNMOUNT"

$ADB\_SH "mount -o remount,ro/"

##Stop the app

echo "STOP APP"

$ADB shell "am force-stop $app\_package"

##NEED REBOOT TO INSTALL AS SYSTEM AFTER FIRST TIME

if [ "$a" -eq "0" ]

then

echo "REBOOT"

$ADB reboot 2> /dev/null

sleep 30

fi

##After reboot open app in emulator

Nota: Iniciar primeiro o emulador antes de carregar play no Android Studio. Depois do emulador estar ligado clicar play. Ás vezes não assume à primeira (clicar play novamente)

* Parametrizar android studio
  + Run > Edit Configurations
  + General tab
  + Instalation Options > Deploy > Nothing
  + Launch Options > Launch > Nothing
  + Before launch > + > Run External Tool
    - Nome
    - Tool Settings > Program > Seleccionar o ficheiro .sh
* Forçar o emulador do Android Studio a correr como -writable-system
  + Na pasta /home/retrai/Android/Sdk/emulador
  + renomear o ficheiro emulador
    - mv emulator emulator-original
  + criar um ficheiro novo com o nome emulator
    - cat > emulator
  + Dentro do ficheiro colocar

#!/bin/bash

/home/retrai/Android/Sdk/emulator/emulator-original $@ -writable-system

* Na primeira vez que corre o emulador
  + Start emulador
  + Deixar terminar de arrancar
  + No terminal
    - adb root
    - adb shell avbctl disable-verification
    - adb reboot
  + A partir daqui o script já corre sempre

Pode ser preciso dar permissões à app → No emulador não está na priv-apps

#Add Permission

adb shell pm grant com.portalrom.setupwizard android.permission.CHANGE\_CONFIGURATION

* Para verificar se ficou instalado no Android Studio
  + View > Tool Windows > Device File Explorer
  + Deve estar instalada na pasta colocada no ficheiro (ex. system/priv-apps)
  + Abrir a app no emulador, por causa do reboot não abre pelo script

> Se der erro com a key, apagar a que existe e criar outra

keytool -genkey -v -keystore ~/.android/debug.keystore -storepass android -alias androiddebugkey -keypass android -dname "CN=Android Debug,O=Android,C=US"