

DIVING INTO ANDROID ENTERPRISE APIS WITH CALYXOS

Aayush Gupta

About Me

- Independent Contractor
- Android Dev @ The Calyx Institute
- Senior Staff & DevRel @ XDA Developers
- Organizer @ GDG Bhilai
- FOSS Contributor

Agenda

- About Us (Institute & OS)
- Android Enterprise APIs
- CalyxOS & Work Profiles
- Bellis (Work Profile Manager)
- More Features

ABOUT US

The Calyx Institute and CalyxOS

The Calyx Institute

- A Non-Profit Education and Research Organization
- Devoted to Studying, Testing, Developing and Implementing Privacy Tech & Tools
- Develops & Maintains CalyxOS



CalyxOS

- Alternative Android Firmware based upon AOSP
- User friendly
- Focuses on Privacy & Security
- Completely FOSS



ANDROID ENTERPRISE APIS

A set of APIs and other tools provided by Google for developers to build enterprise management solutions for Android devices.

Android Enterprise APIs

Android provides organizations with a secure and flexible mobility platform—combining devices, apps, and management.

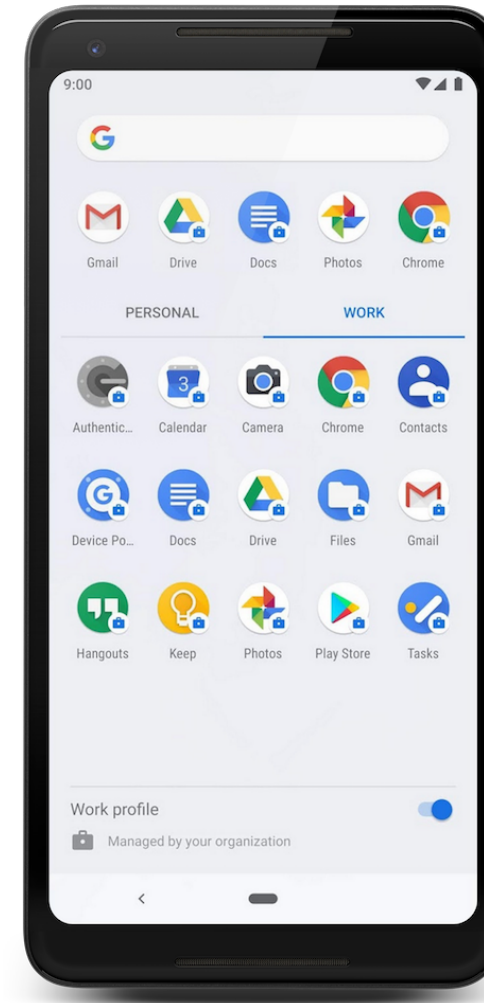
- Work Profiles
- Managed Configurations
- Dedicated Devices
- Single Sign-On (SSO)



Work Profiles

A self-contained profile on an Android device that isolates work apps and data from personal apps and data.

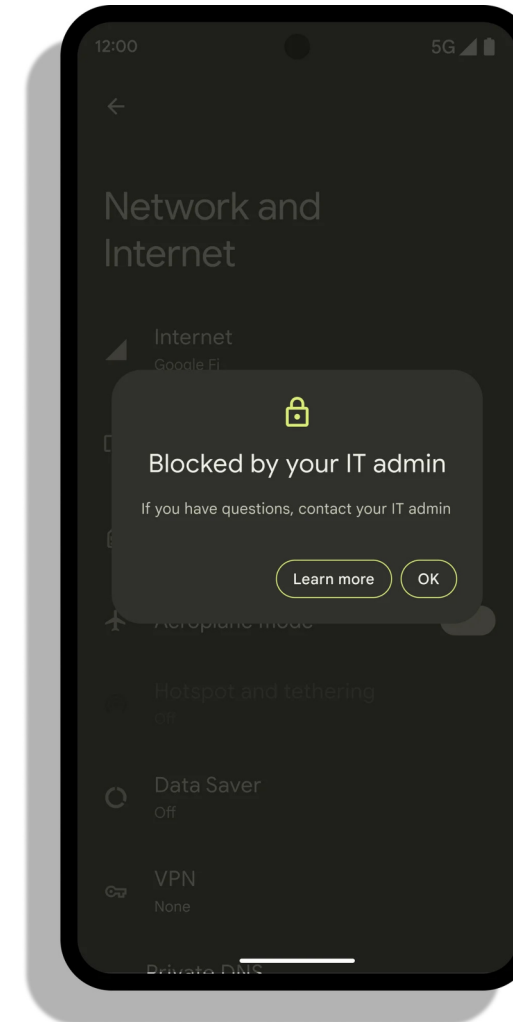
- Controlled by an IT Admin
- Separate Apps & Data From Personal Profile (Primary)
- Also referred as Managed Profiles
- Most Intent Do Not Cross Profiles
- System Apps Can Be Limited
- Separate Storage Areas



Managed Configurations

Managed configurations, allow the organization's IT admin to remotely specify settings for apps

- Enables IT Admin to Change App Settings
- Previously Known as Application Restrictions

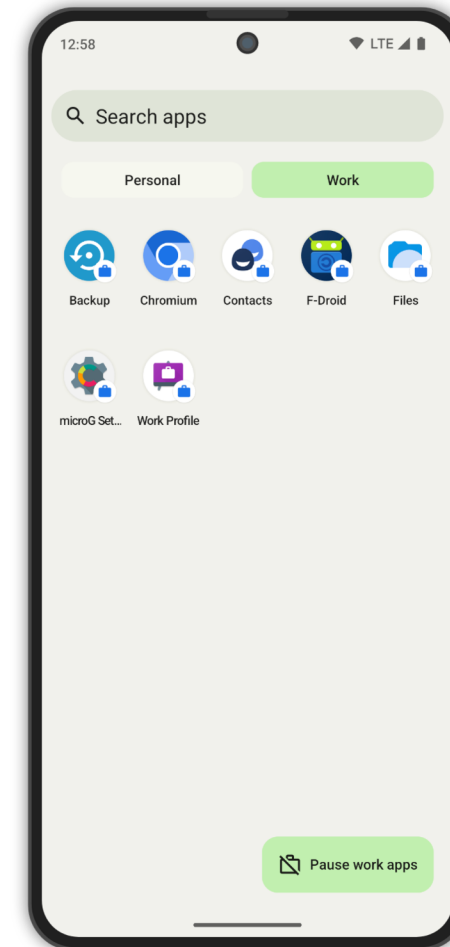


CALYXOS & WORK PROFILES

Features and Development

CalyxOS & Work Profiles

- Created & Managed by User
- Cross-Profile Sharing (In-App Share)
- Permission Auto-Revoke for Work Profiles
- File Operations Across Profiles (Files App, Work Instance)
- Connected Work & Personal Apps
- Apps & Content Access Management (Bellis)

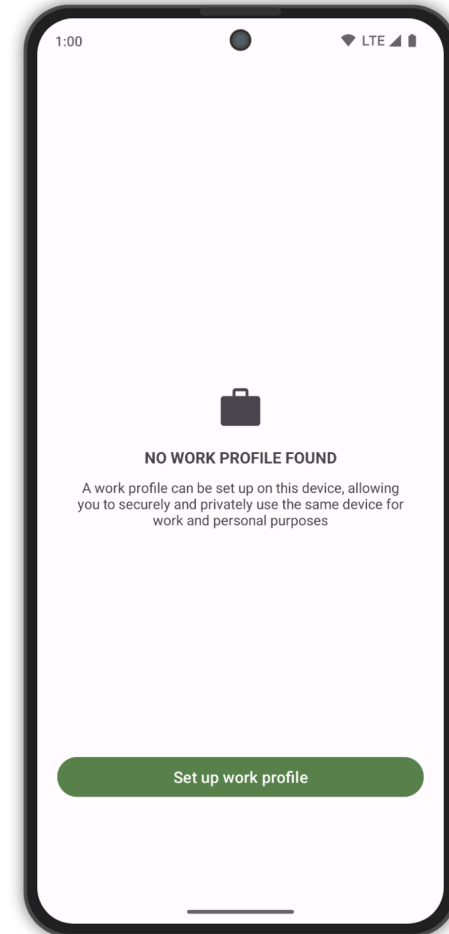


BELLIS

Bellis is a Work Profile manager app, based upon the Android BasicManagedProfile Sample.

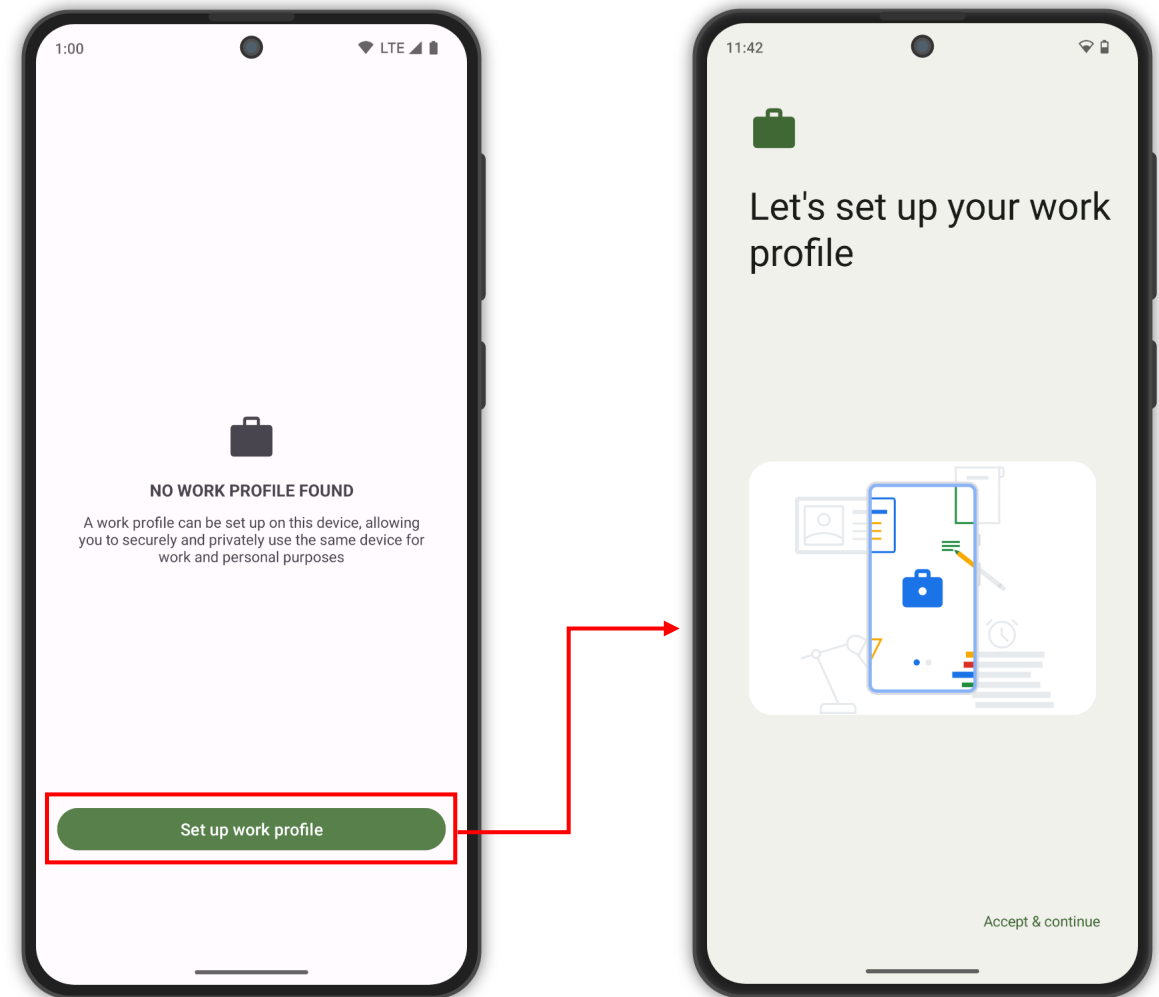
Bellis

- Based upon the Android BasicManagedProfile Sample
- Simple and Easy UI/UX
- Kotlin and Material3
- Compatible With Both AOSP and Gradle Build Systems



Bellis (Cont.)

- Primary Button Triggers Provision
- Once Provision is Complete, Bellis receives an Activity Intent, Namely ACTION_PROVISIONING_SUCCESSFUL
- Bellis Sets Profile Name & Owner, Enables Default Apps and Removes Some Restrictions
- Redirects User to Setup Wizard



```

private fun provisionManagedProfile(context: Context) {
    val intent =
Intent(DevicePolicyManager.ACTION_PROVISION_MANAGED_PROFILE).apply {
        val component = BasicDeviceAdminReceiver.getComponentName(context)

putExtra(DevicePolicyManager.EXTRA_PROVISIONING_DEVICE_ADMIN_COMPONENT_NAME,
component)
        putExtra(
            DevicePolicyManager.EXTRA_PROVISIONING_MODE,
DevicePolicyManager.PROVISIONING_MODE_MANAGED_PROFILE_ON_PERSONAL_DEVICE
        )
    }
    try {
        startForResult.launch(intent)
        activity?.finish()
    } catch (exception: ActivityNotFoundException) {
        Toast.makeText(
            context,
            context.getString(R.string.managed_provisioning_not_enabled),
            Toast.LENGTH_SHORT
        ).show()
    }
}

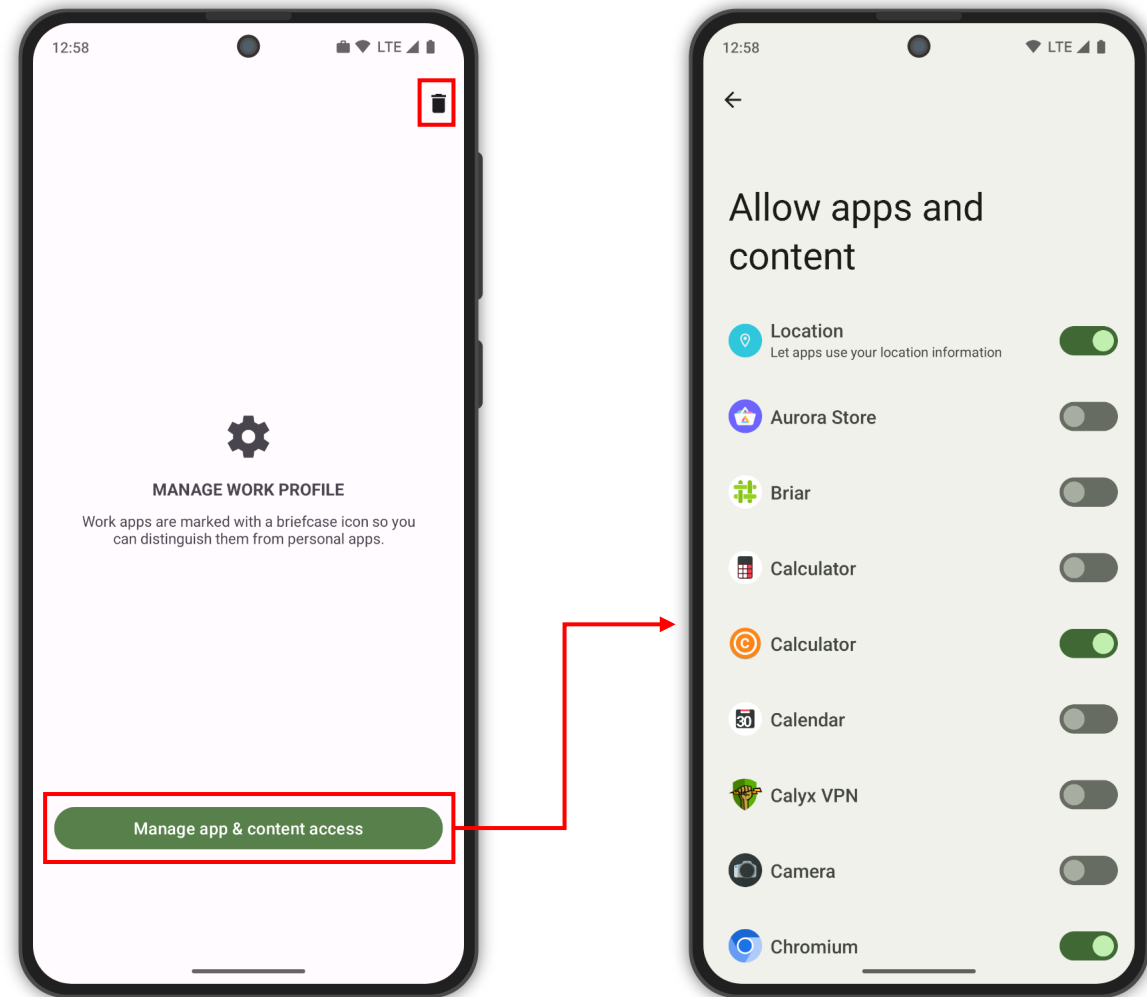
```

[Bellis Code]


```
class MainActivity : AppCompatActivity() {  
  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.main_activity)  
  
        when (intent.action) {  
            DevicePolicyManager.ACTION_PROVISIONING_SUCCESSFUL -> {  
                PostProvisioningHelper.completeProvisioning(this)  
                launchSUW()  
            }  
        }  
    }  
  
    private fun launchSUW() {  
        val setupWizard = "org.lineageos.setupwizard"  
        val setupWizardActivity = ".SetupWizardActivity"  
  
        val intent = Intent(Intent.ACTION_MAIN).apply {  
            setClassName(setupWizard, setupWizard + setupWizardActivity)  
            addFlags(Intent.FLAG_ACTIVITY_NEW_TASK)  
        }  
        startActivity(intent)  
    }  
}
```

Bellis (Cont.)

- On Completing Setup, User Finds Themselves On Bellis (Work Profile Instance)
- Fragment Contains Two Buttons Allowing Users Either To Manage or Delete Work Profile
- Manage Button Launches Intent in Settings App
- Delete Button Shows DialogFragment for Deleting Work Profile



```

override fun onCreateView(view: View, savedInstanceState: Bundle?) {
    super.onCreateView(view, savedInstanceState)

    view.findViewById<Toolbar>(R.id.toolbar)?.setOnMenuItemClickListener {
        when (it.itemId) {
            R.id.removeProfile -> RemoveProfileDialogFragment.show(this)
            else -> Log.d(TAG, "Unexpected itemId: ${it.itemId}")
        }
        true
    }
    view.findViewById<View>(R.id.app_and_content_access)?.setOnClickListener {
        val intent = Intent("android.settings.USER_SETTINGS")
        intent.apply {
            addFlags(Intent.FLAG_ACTIVITY_NEW_TASK)
            addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP)
            addFlags(Intent.FLAG_ACTIVITY_NO_HISTORY)
            putExtra(Intent.EXTRA_USER, Process.myUserHandle())
        }
        it.context.startActivity(intent)
    }
}
}

```

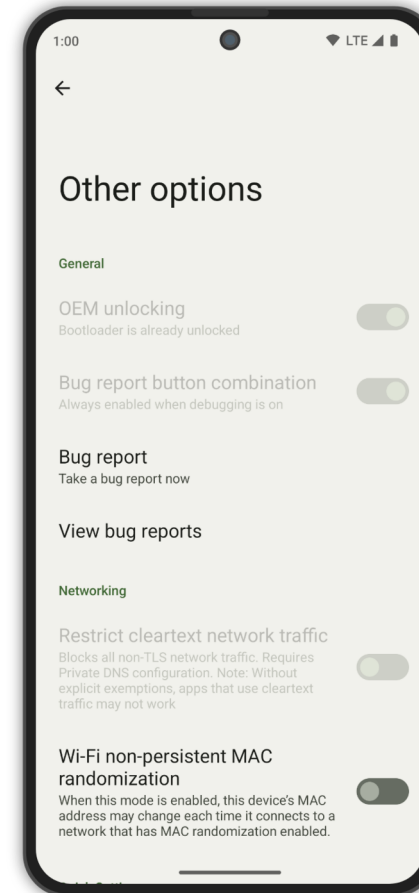
```
UserHandle userHandle = getIntent().getParcelableExtra(Intent.EXTRA_USER,
UserHandle.class);
if (userHandle != null) {
    int userId = userHandle.getIdentifier();
    if (mUserManager.isManagedProfile(userId)) {
        openUserDetails(mUserManager.getUserInfo(userId), true);
    }
}
```

MORE FEATURES

Other Options, Security Levels, Multiple Profiles and more

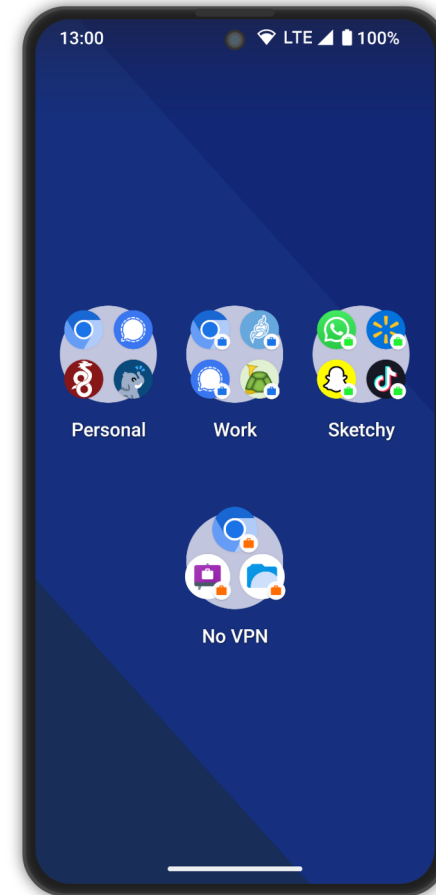
Other Options

- Contains Options Used by Users Frequently (Contained in Developer Options) for Security & Privacy Gains
- Maintains Security Requirement for Certain Switches
- Available When **DISALLOW_DEBUGGING_FEATURES** Restriction is Set



Multiple Profiles

- Allows Provisioning Multiple Managed Profiles
- Isolate apps not just from your workplace but from each other
- Turn off entire sets of apps at once
- Protect and encrypt some apps with a separate passphrase
- Use different VPNs for different profiles
- Currently in Development



Security Levels

- Configurable Device Security Level
- Offers Three Different Levels, Inspired From Tor Browser
- Standard Offers Default Features
- Safer Builds Upon Standard, Sets Default Timeout for Wi-Fi, Bluetooth and Device Reboot, Sets Orbot as Always-On-VPN
- Safest Builds Upon Safer, Sets `DISALLOW_INSTALL_UNKNOWN_SOURCES_GLOBALLY` and `DISALLOW_DEBUGGING_FEATURES` Restrictions, and More
- Currently in Development

THANK YOU

CONNECT WITH US

