**客户1275404问题总结\_gms集成zero touch**

|  |  |
| --- | --- |
| Release Date |  |
| Document No. |  |
| Version |  |
| Document Type |  |
| Platform |  |
| OS Version |  |

**声明 Statement**

本文件所含数据和信息都属于紫光展锐所有的机密信息，紫光展锐保留所有相关权利。本文件仅为信息参考之目的提供，不包含任何明示或默示的知识产权许可，也不表示有任何明示或默示的保证，包括但不限于满足任何特殊目的、不侵权或性能。当您接受这份文件时，即表示您同意本文件中内容和信息属于紫光展锐机密信息，且同意在未获得紫光展锐书面同意前，不使用或复制本文件的整体或部分，也不向任何其他方披露本文件内容。紫光展锐有权在未经事先通知的情况下，在任何时候对本文件做任何修改。紫光展锐对本文件所含数据和信息不做任何保证，在任何情况下，紫光展锐均不负责任何与本文件相关的直接或间接的、任何伤害或损失。

All data and information contained in or disclosed by this document is confidential and proprietary information of UNISOC and all rights therein are expressly reserved. This document is provided for reference purpose, no license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document, and no express and implied warranties, including butwithout limitation, the implied warranties of fitness for any particular purpose, and non-infringement, as well as any performance. By accepting this material, the recipient agrees that the material and the information contained therein is to be held in confidence and in trust and will not be used, copied, reproduced in whole or in part, nor its contents revealed in any manner to others without the express written permission of UNISOC. UNISOC may make any changes at any time without prior notice. Although every reasonable effort is made to present current and accurate information, UNISOC makes no guarantees of any kind with respect to the matters addressed in this document. In no event shall UNISOC be responsible or liable, directly or indirectly, for any damage or loss caused or alleged to be caused by or in connection with the use of or reliance on any such content.

**关键字 Keywords**

Zero Touch、DPC

**版本历史 Revision history**

|  |  |  |  |
| --- | --- | --- | --- |
| **版本Version** | **日期Date** | **作者 Author** | **描述 Description** |
|  |  | **Hursion.zhang** |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**前 言 Foreword**

一 范围Scope

Android 8.0 及更高版本提供在运行 Google 移动服务 (GMS) 的设备上支持Zero touch的功能。使用zero touch功能，Enterprise 客户可自动配置其拥有的设备，无需用户执行任何操作。

在支持zero touch功能的设备中，系统会在设置向导中自动执行 Enterprise 设置（设备所有者）。首次启动时，设备会向zero touch服务器执行签入，以验证其是否已被分配给企业。如果已被分配，设备将启动设备所有者配置，并下载 Device Policy Client 应用，然后通过该应用完成托管设备的设置。展锐平台默认是不支持Zero touch的，需要参考本文进行配置生效，并且此功能需要和google申请accout id和对应的白名单，否则无法正常工作。

二 内容定义 Details Definitions

N/A。

三 参考文献 References

官方配置链接：

<https://docs.partner.android.com/gms/building/enterprise/zero-touch>

IMEI白名单：<https://docs.google.com/forms/d/e/1FAIpQLSd8SOkfmRVjY1Dr0kIt5LUY75umqmVaLpJRwvfMt91j-_C9xw/viewform>

ZTP服务端配置指导：

<https://docs.partner.android.com/aer/oem/test/tests-pre?hl=en&ref_topic=7651436&visit_id=637289946586281802-2891179253&rd=1#ztp>

ZTP服务端门户：

<https://partnerdash.google.com/zerotouch#a=417675012&p=configurations-page>

**目 录Contents**

[声明 Statement 2](#_Toc45891917)

[关键字 Keywords 3](#_Toc45891918)

[版本历史 Revision history 4](#_Toc45891919)

[前 言 Foreword 5](#_Toc45891920)

[1 概览 Overview 8](#_Toc45891921)

[2 手机端实现zero touch 9](#_Toc45891922)

[2.1 要求 9](#_Toc45891923)

[2.2 实现（在设备上） 10](#_Toc45891924)

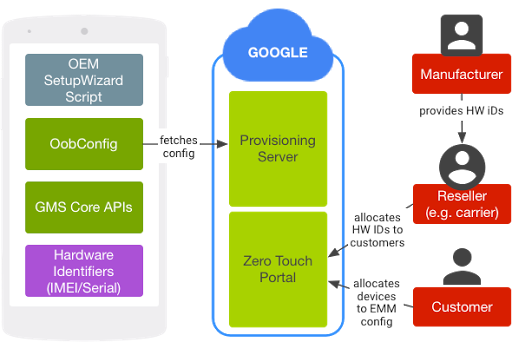
[2.3 配置（服务器端、白名单） 15](#_Toc45891925)

[2.4 测试（端到端注册） 15](#_Toc45891926)

[3 Bug1275404遇到的问题复盘： 17](#_Toc45891927)

1. **概览 Overview**

如需使用zero touch设备，需要由销售渠道（OEM/转销商）为企业分配一组设备。该流程从 OEM 开始，由 OEM 创建一组设备标识符（IMEI 或序列号）并将其传递给转销商，然后转销商将设备标识符映射上传给 Enterprise 客户，最后由客户为已分配的设备配置正确的配置参数。



**图 1.** Zero touch工作流程

1. 制造商实现对zero touch功能的支持。
2. 制造商将一批设备分配给转销商（例如，运营商）。制造商必须提供硬件标识符（IMEI 或序列号），以便转销商能够跟踪其商品目录中每个设备的标识符。
3. 转销商在[Zero touch门户](https://partnerdash.google.com/zerotouch)上为 Enterprise 客户创建客户帐号。
4. 转销商将批量设备出售给 Enterprise 客户。然后，转销商会通过Zero touch门户或转销商 API 将这些设备（使用硬件标识符）分配给客户。
5. 客户使用Zero touch门户为每台设备分配 EMM 配置。此操作会在 Google 配置服务器中创建设备配置记录 (DPR)。
6. 设备首次打开时，将检查 Google 配置服务器以验证设备是否具有 DPR：

* 如果存在 DPR，Google 配置服务器会将其发送给设备，设备将使用该记录启动使用指定配置值的托管配置。
* 如果不存在 DPR，设置将继续正常运行且没有延迟（在一般使用情况下，设置期间不会产生额外费用）。

如果在正常设置向导流程中，设备无法向 Google 服务器执行签入，则设备将保留一个标志，指示应在连接到网络时签入。在后续签入过程中，如果设备配置为zero touch，设备会通知用户并在两小时后恢复出厂设置。在后续签入前，无法使用 OEM 解锁设备的功能（**设置** > **开发者**），但可以使用设备的其他功能。

1. **手机端实现zero touch**

计划在设备上实现zero touch支持的设备制造商必须完成以下四个部分中所述的所有步骤。

* 1. **要求**

确保设备符合下述 ID、API 和 APK 要求。如需启用Zero touch，必须使用以下组件。

#### 有效的硬件标识符

每个实现zero touch的设备都必须采用唯一标识符表示给定的制造商和型号对（依据 [Android CDD](https://source.android.com/compatibility/9/android-9-cdd#3_2_2_build_parameters)）。如果设备不支持电话功能，必须提供唯一的序列号；如果设备支持电话功能，必须提供全局唯一的 IMEI/MEID。

转销商可使用这些标识符对设备进行唯一标识。如果转销商为特定标识符创建 DPR，则具有此标识符并实现Zero touch的任何设备都将使用此 DPR 在Zero touch模式下进行配置。

制造商必须确保正确设置其出厂流程，以便将正确的唯一标识符填充到设备上。Google 配置会在签入过程中使用并验证设备报告的 [MANUFACTURER](https://developer.android.com/reference/android/os/Build#MANUFACTURER) 值。制造商必须在进行任何集成测试之前请求将 MANUFACTURER 值列入白名单。

#### Android 框架 API

设备出厂映像必须为 Android 8.0 或更高版本。

#### 预加载的 Google APK

设备系统映像必须包含 OobConfig.apk（在 [8.0 及更高版本的 GMS 软件包](https://docs.partner.android.com/gms/building/integrating/gms-download)中提供）。此外，Zero touch依赖于 GMS Core API，并且只能在运行 GMS 的设备上使用。

* 1. **实现（在设备上）**

如需启用Zero touch，必须执行以下步骤：

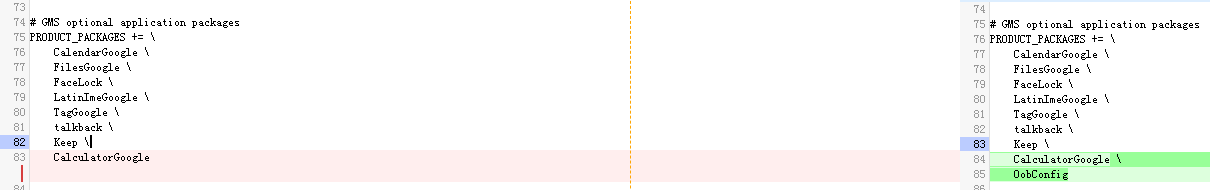
1. 配置框架。

在 frameworks/base/core/res/res/values/config.xml 或者overlay文件中，确保已设置 OOB 配置软件包。示例：



Gms包中默认没有编译oobConfig，可以在对应的gms.mk中添加oobConfig

gerrit示例（<http://review.source.unisoc.com/gerrit/#/c/664878/1/products/gms.mk>）



1. 配置oem unlock。

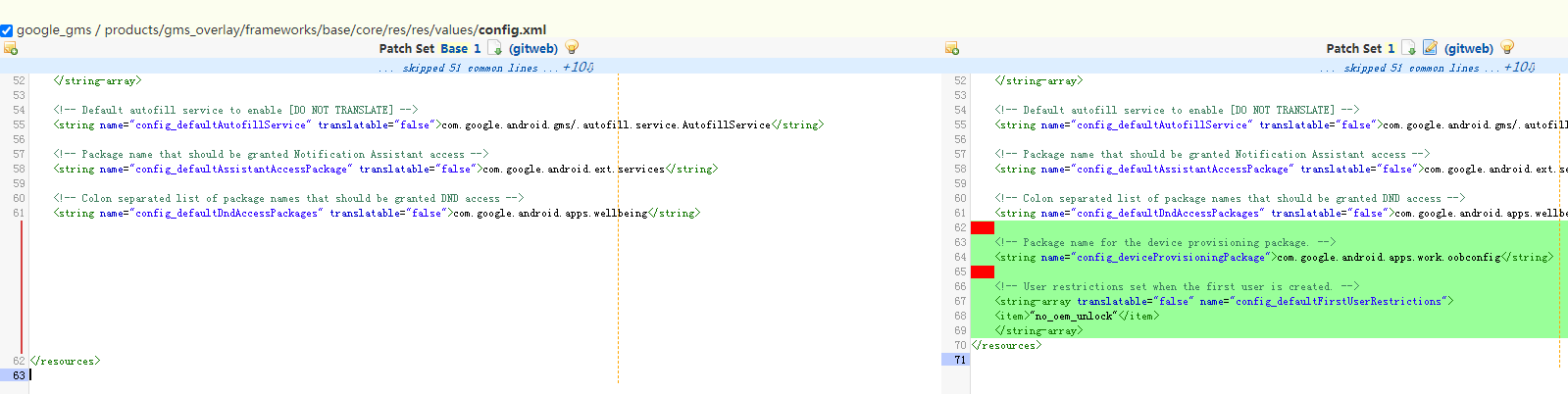
为确保已分配 DPR 的设备可以检索该记录，制造商应配置引导加载程序锁定，以确保在设备成功签入之前无法解锁该引导加载程序。如需配置锁定，请执行以下操作：

确保在 OemLockManager#isOemUnlockAllowedByUser 和 OemLockManager#isOemUnlockAllowedByCarrier() 返回 true 之后才能解锁引导加载程序。（这些函数在文件 frameworks/base/core/java/android/service/oemlock/OemLockManager.java 中定义。）

确保在 frameworks/base/core/res/res/values/config.xml 文件中为设备上的第一个用户设置 OEM\_UNLOCK。示例：

<!-- User restrictions set when the first user is created. -->  
    <string-array translatable="false" name="config\_defaultFirstUserRestrictions">  
    <item>"no\_oem\_unlock"</item>  
    </string-array>

gerrit示例（<http://review.source.unisoc.com/gerrit/#/c/664878/1/products/gms_overlay/frameworks/base/core/res/res/values/config.xml>）



1. 将zero touch脚本添加到设置向导脚本中。

确保 wizard\_script\_zero\_touch\_flow.xml 文件是设置向导脚本的一部分。此脚本按正确顺序调用 Google 组件，位于 Android 8.0 及更高版本的 GMS 软件包中partner\_gms/apps/GmsSampleIntegration/res/raw/wizard\_script\_zero\_touch\_flow.xml 下。

注意：不应自定义设置向导脚本

1. 在设置向导中配置 zero\_touch 操作。

使用以下任一选项：

默认 zero\_touch 操作（推荐）。想要使用默认 WizardAction 调用zero touch的 OEM 必须使用 partner\_gms/apps/GmsSampleIntegration/res/raw/wizard\_script.xml（GMS 软件包中包含的脚本）中提供的 zero\_touch 操作。启动 zero\_touch 操作的 WizardAction 如下所示，仅供参考：

<!-- Zero touch provisioning (for enterprise) [RECOMMENDED] -->  
     <WizardAction id="zero\_touch" wizard:script="android.resource://com.google.android.gmsintegratio n/raw/wizard\_script\_zero\_touch\_flow" >  
           <result wizard:name="dpm\_user\_complete"  
               wizard:resultCode="111"  
               wizard:action="post\_dpm\_user\_flow" />  
     </WizardAction>

注意：更改 zero\_touch 操作顺序可能会对zero touch功能产生不利影响。

自定义的 zero\_touch 操作。如果 OEM 希望自定义操作以使用自定义设置流程，可以使用以下方法：

<WizardAction id="ACTION\_THAT\_ESTABLISHES\_NETWORK\_CONNECTIVITY" wizard:uri="ACTIVITY">  
       <result wizard:action="gms\_checkin" />  
    </WizardAction>  
  
    <WizardAction id="gms\_checkin" wizard:uri="intent:#Intent;action=com.google.android.setupwizard.GMS\_CHECKIN;end">  
       <result wizard:action="zero\_touch" />  
    </WizardAction>

在您的主 WizardScript 中，在向导自定义软件包中添加对zero touch流脚本的引用：

<WizardAction id="zero\_touch" wizard:script="LOCATION\_OF\_ZERO\_TOUCH\_FLOW\_SCRIPT">  
       <result wizard:name="dpm\_user\_complete" wizard:resultCode="111" />  
    </WizardAction>

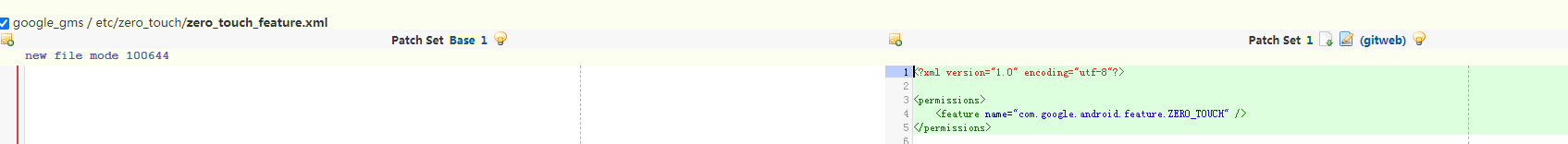
在使用自定义流程时，请确保设备已连接到网络并在调用 zero\_touch 操作前执行 Google 签入；如果完全跳过网络连接，则无需运行zero touch。

1. 声明zero touch功能标记。

为了让客户和转销商能够轻松识别支持Zero touch的设备（以及方便进行测试），制造商应在支持Zero touch的所有 build 中声明 com.google.android.feature.ZERO\_TOUCH 功能。

Gerrit示例（<http://review.source.unisoc.com/gerrit/#/c/664878/1/etc/zero_touch/zero_touch_feature.xml>）





1. 测试设备端zero touch实现。

如需验证设备上的实现是否正确以及zero touch功能是否正常运行，请在启动后立即确认设备 logcat 是否包含以下行：

08-31 22:22:20.856 10038 7546 8240 I OobConfig: zeroTouchConfig set: false, simLockConfig set: false, disallowOemUnlock restriction set: false

如果出现任何符合此格式的行（无论 zeroTouchConfig、simLockConfig 或 disallowOemUnlock 字段的值如何），表示正在调用zero touch代码。只有当设备连接到互联网以检索配置且配置为zero touch时，此行才会显示在日志中。

* 1. **配置（服务器端、白名单）**
* 需要制造商和google申请白名单：

<https://docs.google.com/forms/d/e/1FAIpQLSd8SOkfmRVjY1Dr0kIt5LUY75umqmVaLpJRwvfMt91j-_C9xw/viewform>

如遇到问题可直接联系google TAM

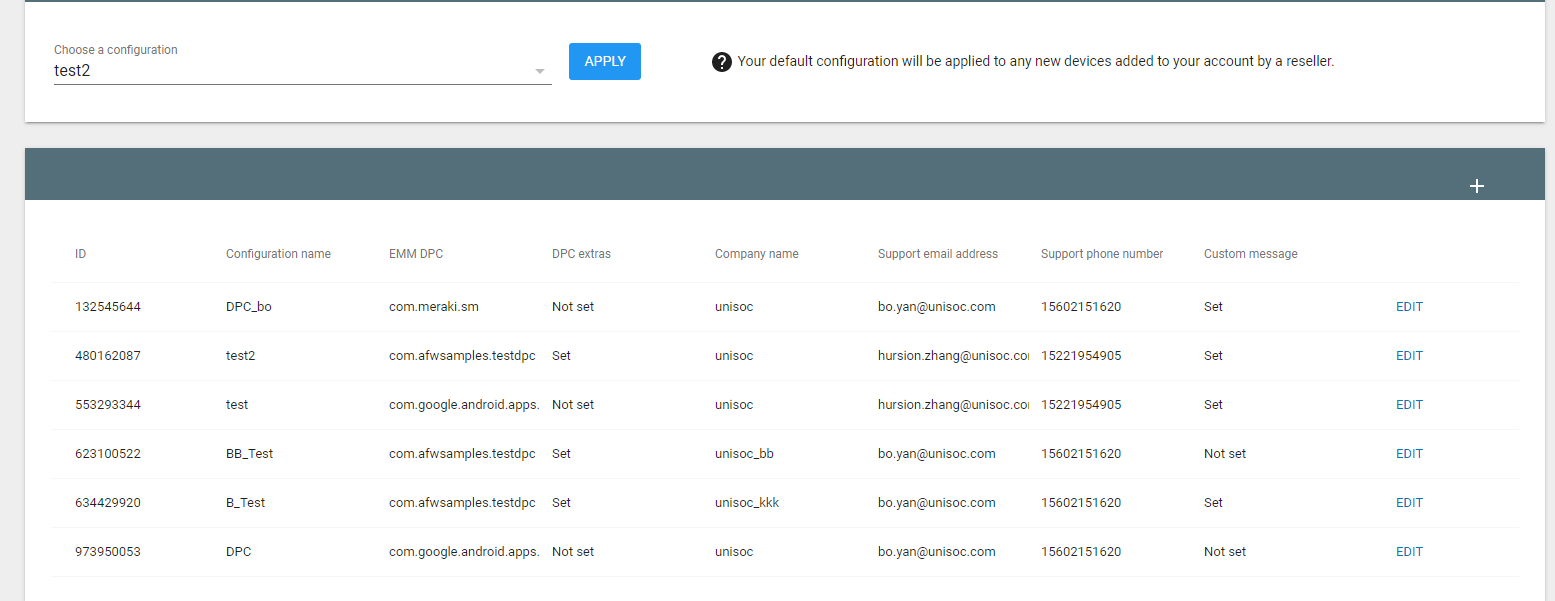
* 1. **测试（端到端注册）**

OEM 和 EMM 合作伙伴必须对发布 build 进行端到端测试，并且可以对 userdebug build 或工程 build 进行测试。

1.联系google TAM申请account id。

2提供要注册到客户帐号进行测试的设备标识符列表。

**测试配置**。创建客户帐号后，您必须登录客户门户（<https://partnerdash.google.com/zerotouch#a=417675012&p=configurations-page>）并创建可配置为使用 [Test DPC](https://play.google.com/store/apps/details?id=com.afwsamples.testdpc)（推荐）或 EMM DPC 的测试配置。如需使用 Test DPC，请在zero touch门户中创建新配置(<https://docs.partner.android.com/aer/oem/test/tests-pre?hl=en&ref_topic=7651436&visit_id=637289946586281802-2891179253&rd=1#ztp>)，然后在 EMM DPC 菜单中选择 **Test DPC** 作为 DPC 应用，如下面的示例所示。



**图 2.** Test DPC 测试配置

1. **Bug1275404遇到的问题复盘：**

此问题主要由于平台默认未集成zero touch，并且未做过相关测试，对zero touch功能和测试手法上都比较陌生，复现问题需要经过google、unisoc、sprocomm、blu、blu客户等一系列流转，比较耗费时间精力，对主要问题点做下复盘。

1. BLU运营商使用的EMM DPC添加的token只能在user版本上验证，debug版本无法触发zero touch，导致log信息缺失，难以分析；
2. Sprocomm一开始未添加“com.google.android.feature.ZERO\_TOUCH ” feature导致zero touch未正常工作，未下载dpc enroll机器，通过对比视频和排查代码提交确认；
3. Dpc正常下载后再创建account id时出问题，通过afw#setup账号抓取异常机器bugreport（自己下载开源代码编译工具实现），解析后发现CloudDeviceAdminService bind失败的问题，怀疑是被power这边限制了自启动服务；

S01DE0A 04-13 13:24:51.873 845 1908 D PowerController.BgClean: in autolaunch black list: com.google.android.apps.work.clouddpc, callingPackage = android, reason = bind-service denyed!

S01DE0B 04-13 13:24:51.874 845 1908 E DevicePolicyManager: Binding: ComponentInfo{com.google.android.apps.work.clouddpc/com.google.android.apps.work.clouddpc.vanilla.services.admin.CloudDeviceAdminService} u0 failed.

S01DE0C 04-13 13:24:51.875 845 1908 I DevicePolicyManager: Device owner set: ComponentInfo{com.google.android.apps.work.clouddpc/com.google.android.apps.work.clouddpc.receivers.CloudDeviceAdminReceiver} on user 0

最后通过和power同事确认添加白名取消这个服务自启动的限制后正常获取account id。

修改如下：

<?xml version='1.0' encoding='utf-8' standalone='yes' ?>

<app\_powersave\_config>

<package name="com.pp.assistant" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="1" secondarylaunch="1" lockscreencleanup="0" consumertype="0" />

<package name="com.spreadtrum.itestapp" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="com.comcat.activity" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="com.sprd.sleepwakeuptest" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="sprdtest.message" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="com.greenpoint.android.mc10086.activity" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="com.jio.emiddleware" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="2" consumertype="0" />

<package name="com.sprd.bmte.coulomb" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="0" secondarylaunch="0" lockscreencleanup="2" consumertype="0" />

<package name="com.google.android.apps.work.clouddpc" optimize="1" alarm="0" wakelock="0" network="0" autolaunch="2" secondarylaunch="2" lockscreencleanup="0" consumertype="0" />

</app\_powersave\_config>