

团体标准

T/TAF 353.4—2026

移动互联网应用程序（APP）最小必要申请 使用权限要求 第4部分：联系人

Mobile Internet application permission application and usage
minimization and necessity requirements—
Part 4: Contacts

2026-06-17 发布

2026-06-17 实施

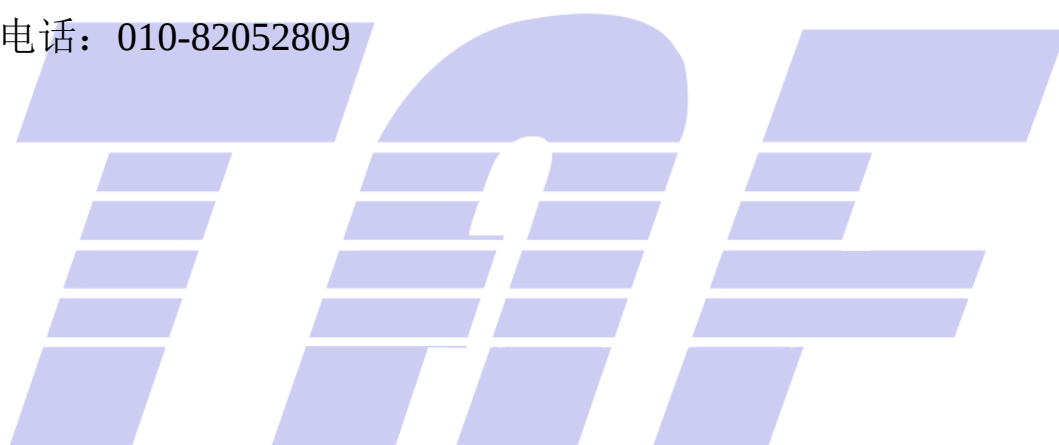
电信终端产业协会 发布

版权声明

本文件的版权属于电信终端产业协会，任何单位和个人未经许可，不得进行技术文件的纸质和电子等任何形式的复制、印刷、出版、翻译、传播、发行、合订和宣贯等，也不得未经允许采用其具体内容编制本团体以外各类标准和技术文件。如有以上需要请与本团体联系。

邮箱：tafrb@taf.org.cn

电话：010-82052809



目 次

前言 II

引言 III

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 缩略语 1

5 基本原则 2

6 典型场景 2

7 技术要求 3

8 评估要求 4

附录 A （资料性） Android 系统联系人选择器接口 5

附录 B （资料性） Android 系统 APP 申请使用联系人权限代码示例 8

参考文献 9

前 言

本文件按照 GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是 T/TAF 353《移动互联网应用程序（APP）最小必要申请使用权限要求》的第4部分。T/TAF 353 已经发布了以下部分：

- 第1部分：总则；
- 第2部分：相册；
- 第3部分：文件；
- 第4部分：联系人；
- 第5部分：通话记录；
- 第6部分：日历；
- 第7部分：短信；

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由电信终端产业协会（TAF）提出并归口。

本文件起草单位：中国信息通信研究院、OPPO 广东移动通信有限公司、荣耀终端股份有限公司、维沃移动通信有限公司、小米通讯技术有限公司、华为终端有限公司、北京三星通信技术研究有限公司、阿里巴巴（北京）软件服务有限公司、友盟同欣（北京）科技有限公司、玩咖欢聚科技（北京）有限公司、卓望数码技术（深圳）有限公司、厦门美柚股份有限公司。

本文件主要起草人：李腾、杨明慧、陈鑫爱、王艳红、王炎、周飞、常浩伦、武林娜、王淞鹤、杜云、李京典、李可心、姚一楠、贾科、闫金保、赵晓娜、赵小平、吴越、张丹静雅、衣强、张圣迪、杨艺、王彬、贾紫薇、姚栋、刘艾婧、刘伟峰、黄鹏华、钱雷、薛源。

引 言

随着移动互联网的快速发展，APP 已成为用户获取信息和服务的主要载体，但其过度申请和滥用权限的问题日益突出，严重威胁用户信息安全和合法权益。

T/TAF 353《移动互联网应用程序（APP）最小必要申请使用权限要求》系列标准旨在对 APP 最小必要的申请使用权限提出要求，由 7 部分构成。

- 第 1 部分：总则。目的在于规定 APP 最小必要申请使用权限的基本原则、基本要求，以及 APP 开发运营者、移动应用分发平台、移动智能终端、第三方评估机构在最小必要权限管理方面的管理要求。
- 第 2 部分：相册。目的在于规定移动互联网应用程序（APP）最小必要的申请访问相册的基本原则、典型场景、技术要求和评估要求。
- 第 3 部分：文件。目的在于规定移动互联网应用程序（APP）最小必要的申请使用文件的基本原则、典型场景、技术要求和评估要求。
- 第 4 部分：联系人。目的在于规定移动互联网应用程序（APP）最小必要的申请使用终端系统联系人权限的基本原则、典型场景、技术要求和评估要求。
- 第 5 部分：通话记录。目的在于规定移动互联网应用程序（APP）最小必要的申请使用通话记录权限的基本原则、典型场景、技术要求和评估要求。
- 第 6 部分：日历。目的在于规定移动互联网应用程序（APP）最小必要的申请使用日历访问权限的基本原则、典型场景、技术要求和评估要求。
- 第 7 部分：短信。目的在于规定移动互联网应用程序（APP）最小必要的申请使用短信的基本原则、典型场景、技术要求和评估要求。

移动互联网应用程序（APP）最小必要申请使用权限要求 第4部分： 联系人

1 范围

本文件规定了移动互联网应用程序（APP）最小必要的申请使用终端系统联系人权限的基本原则、典型场景和技术要求。

本文件适用于移动互联网应用程序（APP）开发运营者规范申请使用终端系统联系人权限，也适用于主管部门、第三方评估机构、移动应用分发平台、移动智能终端等对 APP 申请使用终端系统联系人权限行为进行监督、管理和评估。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

T/TAF 353.1—2026 移动互联网应用程序（APP）最小必要申请使用权限要求 第1部分：总则

3 术语和定义

T/TAF 353.1—2026 界定的以及下列术语和定义适用于本文件。

3.1

终端联系人信息 user contact information

用户在移动终端系统上添加生成的联系人信息，包括联系人姓名、手机号码、邮箱地址、公司及职位、住宅地址、昵称、头像、生日、群组信息等个人相关信息。

3.2

联系人权限 contacts permissions

移动智能终端操作系统向 APP 开放的，使 APP 能够读取、写入终端联系人信息的权限。

注：联系人权限分为读联系人权限和写联系人权限，以 Android 系统为例，读联系人权限为 android.permission.READ_CONTACTS，写联系人权限为 android.permission.WRITE_CONTACTS。

3.3

联系人选择器 contacts picker

移动智能终端操作系统向 APP 开放的选择终端联系人信息的交互组件，为用户提供终端联系人信息的选择界面和交互流程，使 APP 无需申请联系人权限，即可读取用户选择的终端联系人信息。

4 缩略语

下列缩略语适用于本文件。

APP：移动互联网应用程序（Mobile Internet Application）

5 基本原则

APP 申请读取联系人的基本原则如下：

- a) 应满足 T/TAF 353.1—2026 总则中的原则；
- b) 选择器优先原则：APP 读取终端联系人信息应优先使用联系人选择器实现，由用户主动选择单个或多个终端联系人信息提供给 APP，而非通过申请读全量联系人权限访问全部终端联系人信息。

注：如联系人权限本身可以实现为用户提供终端联系人信息的选择和交互，可视为具备和联系人选择器同等功能。

6 典型场景

6.1 APP 可申请使用联系人权限场景

APP（包括嵌入的 SDK）应按照业务功能场景的需要来申请使用联系人权限，仅在联系人权限为 APP 实现业务功能的不可替代的必要条件时才能申请，不应过度申请权限。APP 可申请使用联系人读取权限的典型场景见表 1。未被表 1 所覆盖的其他场景，应遵循第 5 章所述基本原则。

表 1 可申请全量联系人权限的典型场景

场景	描述	细分场景	可申请权限
			读联系人
穿戴设备的配套应用	通过此类应用，在穿戴设备来电或通话时可自动匹配联系人数据，用于实时显示姓名、备注等信息。	来电场景：医疗健康、运动健身 APP 在穿戴设备上显示来电提醒	√
数据备份、恢复及同步	通过读取联系人数据，用于自动备份及同步联系人信息到用户指定的存储（设备、云盘等）	备份及同步：数据同步软件在多端设备上同步数据	√
	读取联系人数据用于在不同设备间进行传输同步	同步：手机搬家在新旧设备之间同步数据	√
短信/电话清理时识别联系人信息	当启用广告短信或通话记录清理功能时，识别联系人号码用于不做清理	垃圾清理软件清理垃圾短信及通话记录	√
输入法形成本地词库	读取联系人信息用于形成词库，便于快捷输入联系人姓名或号码	输入法形成联系人本地词库	√

6.2 APP 不应申请使用联系人访问权限场景

APP 出现以下情形不应申请使用联系人权限。

- a) 联系人权限的使用与 APP 的服务场景并没有直接关联。
- b) 仅以服务体验、产品研发、算法推荐、风险控制为由申请权限。
- c) 获取数据是为了将其出售或分享给第三方用于分析或广告用途。
- d) 无需联系人权限就能完成 APP 的目标任务，实现核心功能。例如使用联系人选择权完成读联系人功能。联系人选择器相关代码及接口示例见附录 A。
- e) 不应申请使用联系人权限的典型场景见表 2。

表 2 不应申请联系人权限的场景

场景	描述
话费充值/亲情号码设置	用于在使用话费充值、流量直充、为好友充值、设置亲情账号等功能时快速选择联系人号码。
设置来电视频铃声	用于选择特定号码设置来电视频铃声。
寄件信息快速填充	用于在寄件时快捷选择和输入寄件人、收件人的姓名、电话号码等信息。
手机转账/转账通知	用于在手机号转账、短信通知收/付款人、维护常用收款人、产品推荐等服务中，读取联系人快速选择手机号。
骚扰拦截	获取联系人信息用于添加骚扰拦截黑名单或通话放行白名单。
分享行程/文件/IOT 设备/内容	在分享行程/文件/IOT 设备/内容，读取联系人列表，用于用户选择接收者。
特殊信息提醒	读取用户联系人数据，用于设置生日/日程提醒等。
紧急联系功能	添加紧急联系人用于自动分享行程、紧急求助、一键报警等。
快捷拨号	选取联系人配置成快速拨号快捷键，用于点击快速拨打电话。
主体信息填充	在挂号就诊、火车票购买、贷款申请等生活服务类场景，用于快速填充挂号人、购票人、贷款人等信息
好友推荐与关系拓展	读取用户的联系人信息，获取好友信息，为用户推荐可能认识的人及其内容或添加通讯录好友
商业及营销活动	读取用户的联系人信息用于推广活动、问卷调查等商业活动。
金融反欺诈及风控	获取联系人数据用于判断是否存在团体欺诈风险、授信审核及风控决策，以保障支付安全
注：如系统联系人选择器无法支持全选，骚扰拦截、好友推荐与关系拓展场景需要读取全量联系人的可申请联系人权限。	

7 技术要求

7.1 APP 申请联系人权限

APP 申请联系人权限应满足如下要求。

- APP（包括嵌入的 SDK）申请所需权限应在声明文件（例如 AndroidManifest.xml）中严格按照格式规范逐个声明，代码示例见附录 B。
- APP 应在需要读联系人的功能页面启动运行时动态申请权限，不应提前申请，以提高用户对权限申请必要性的理解。
- APP 申请权限时应同步告知权限申请目的，目的应明确且易于理解，不包含广告及任何欺诈、诱骗、误导用户授权的描述。
- 如 APP 仅需使用权限组中部分权限，不应在权限声明文件中声明同一权限组其他权限，例如当 APP 仅需使用读联系人权限时，不应在 AndroidManifest.xml 中声明写联系人权限。
- 如用户拒绝或撤回授予某服务类型非必要系统权限，APP 不应强制退出或关闭，且不影响与此权限无关的业务功能使用。
- APP 不得采用默认、捆绑或使用其他手段变相欺骗、误导、强迫用户授予权限。
- 如用户明确拒绝 APP 业务功能所需权限，APP 不应频繁申请权限干扰用户正常使用，除非由用户主动触发功能，且没有该权限参与此业务功能无法实现。

7.2 APP 使用联系人权限

APP 使用联系人权限应满足如下要求。

- a) APP 在使用联系人权限前，应检查自身是否已获得权限。联系人权限检查及处理相关代码示例见附录 B。
- b) APP 被授予权限后应遵循最小必要原则合理使用权限，不应超频次、超范围、超精度收集终端联系人信息。
- c) 若 APP 功能发生变化而不再需要使用联系人权限时，应及时停止权限申请和使用。对于更新之后的 APP 应重新评估权限申请使用的必要性。

8 评估要求

APP 申请使用权限的评估要求如下。

- a) APP 提供者应提供联系人权限相关的功能描述、截图/视频演示、权限请求触发路径描述等辅助材料供相关方判断 APP 申请使用联系人权限的必要性。
- b) 相关方可以根据第 6 章列出的典型场景、APP 提供者提交的辅助材料等判断申请使用联系人权限的业务场景是否合理。
- c) 相关方可根据第 7 章规定的技术要求判断 APP 是否按规定申请和使用联系人访问权限。

附 录 A

（资料性）

Android 系统联系人选择器接口

A.1 单选联系人

联系人选择器支持单选联系人功能的代码示例如下。

```
final int REQUEST_CODE_SINGLE_PICK_CONTACTS = 1001;
public void pickSingleContact() {
    Intent intent = new Intent(Intent.ACTION_PICK);
    intent.setData(ContactsContract.CommonDataKinds.Phone.CONTENT_URI);
    //intent.setType(ContactsContract.CommonDataKinds.Phone.CONTENT_TYPE);
    startActivityForResult(intent, REQUEST_CODE_SINGLE_PICK_CONTACTS);
}
@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == REQUEST_CODE_SINGLE_PICK_CONTACTS) {
        Log.d(TAG, "resultCode:" + resultCode);
        if (resultCode == Activity.RESULT_OK) {
            if (data == null) {
                return;
            }
            Uri contactUri = data.getData();
            if (contactUri == null) {
                return;
            }
            try (Cursor cursor = getContentResolver().query(contactUri, null, null, null, null)) {
                if (cursor != null && cursor.moveToFirst()) {
                    int nameIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME);
                    String name = cursor.getString(nameIndex);
                    int phoneIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER);
                    String phone = cursor.getString(phoneIndex);
                    Log.d(TAG, "name:" + name + ",phone:" + phone);
                }
            }
        } else if (resultCode == Activity.RESULT_CANCELED) {
```

```

        Log.d(TAG, "user canceled");
    }
}
}

```

A.2 多选联系人

联系人选择器支持多选联系人功能的代码示例如下。

```

final int REQUEST_CODE_MULTIPLE_PICK_CONTACTS = 10002;
public void pickMultipleContact() {
    Intent intent = new Intent(Intent.ACTION_PICK);
    intent.putExtra(Intent.EXTRA_ALLOW_MULTIPLE, true);
    intent.setData(ContactsContract.CommonDataKinds.Phone.CONTENT_URI);
    //intent.setType(ContactsContract.CommonDataKinds.Phone.CONTENT_TYPE);
    startActivityForResult(intent, REQUEST_CODE_MULTIPLE_PICK_CONTACTS);
}

@Override
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == REQUEST_CODE_MULTIPLE_PICK_CONTACTS) {
        Log.d(TAG, "resultCode:" + resultCode);
        if (resultCode == Activity.RESULT_OK) {
            if (data == null) {
                return;
            }
            ClipData clipData = data.getClipData();
            if (clipData == null) {
                return;
            }
            for (int i = 0; i < clipData.getItemCount(); i++) {
                ClipData.Item item = clipData.getItemAt(i);
                if (item == null) {
                    continue;
                }
                Uri contactUri = item.getUri();
                if (contactUri == null) {
                    continue;
                }
                try (Cursor cursor = getContentResolver().query(contactUri, null, null, null, null))
            {
                if (cursor != null && cursor.moveToFirst()) {
                    int nameIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME);

```

```
        String name = cursor.getString(nameIndex);
        int phoneIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER);
        String phone = cursor.getString(phoneIndex);
        Log.d(TAG, "name:" + name + ",phone:" + phone);
    }
}
}
} else if (resultCode == Activity.RESULT_CANCELED) {
    Log.d(TAG, "user canceled");
}
}
}
```



附 录 B

(资料性)

Android 系统 APP 申请使用联系人权限代码示例

B.1 权限声明

```
<uses-permission android:name="android.permission.READ_CONTACTS" />
```

B.2 权限检查与申请

```
// 检查是否已获得 READ_CONTACTS 权限
if (ContextCompat.checkSelfPermission(this,
    Manifest.permission.READ_CONTACTS) == PackageManager.PERMISSION_GRANTED) {
    // 已获得权限，可以执行读取通讯录的操作
    readContacts();
} else {
    // 未获得权限，需要申请
    ActivityCompat.requestPermissions(this,
        new String[] {Manifest.permission.READ_CONTACTS},
        REQUEST_CODE_READ_CONTACTS);
}

// 处理权限申请结果回调
@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions, int[]
grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == REQUEST_CODE_READ_CONTACTS) {
        if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
            // 用户同意授权
            readContacts();
        } else {
            // 用户拒绝授权
            Toast.makeText(this, "需要通讯录权限才能使用此功能",
Toast.LENGTH_SHORT).show();
        }
    }
}
```

参 考 文 献

- [1] YD/T 2407—2021 移动智能终端安全能力技术要求
- [2] YD/T 6192—2024 移动应用分发平台 服务和管理要求
- [3] T/TAF 125—2023 应用分发平台 APP 审核规范
- [4] T/TAF 078.8—2025 APP 用户权益保护测评规范 第8部分：移动应用分发平台管理
- [5] T/TAF 188—2023 软件开发工具包（SDK）收集个人信息技术要求



电信终端产业协会团体标准

移动互联网应用程序（APP）最小必要申请使用权限要求
第4部分：联系人

T/TAF 353.4—2026

*

版权所有 侵权必究

电信终端产业协会发布

地址：北京市西城区新街口外大街28号

电话：010-82052809

电子版下载网址：www.taf.org.cn