**OpenHarmony端云一体化应用开发快速入门练习（中）登录认证**

# 一、登录认证手机

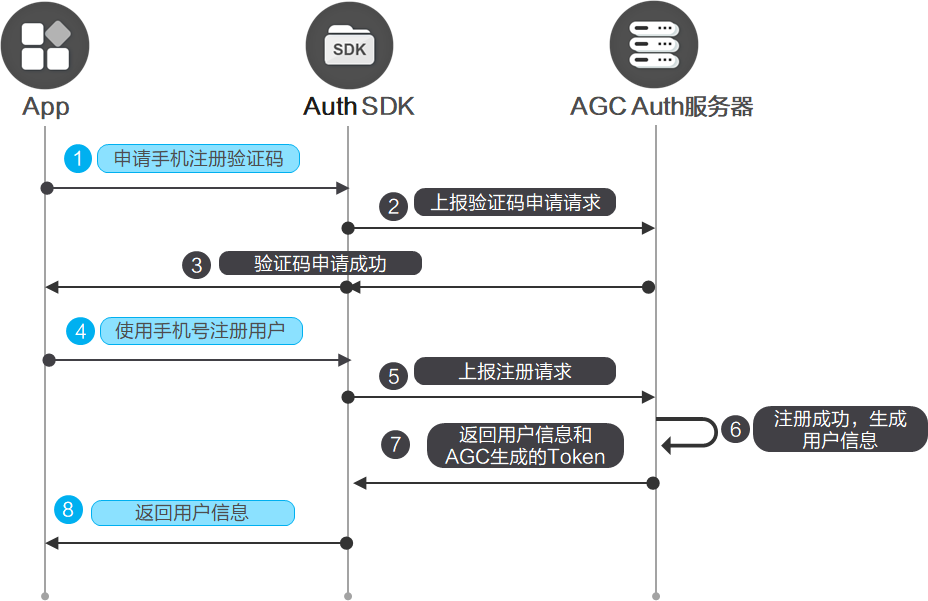
# 可以在应用中集成手机帐号认证方式，您的用户可以使用“手机号码+密码”或者“手机号码+验证码”的方式来登录您的应用。

## （一）前提条件

需要在AGC控制台开通认证服务。

需要先在您的应用中集成SDK。

## （二）注册



1.申请手机号码注册的验证码。

说明：在使用手机号码注册之前，需要先验证您的手机，确保该手机归您所有。

调用[AGConnectAuth.requestPhoneVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section154023210148" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank) 申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestPhoneVerifyCode(countryCode,phoneNumber,verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

1. 使用手机号码注册用户。

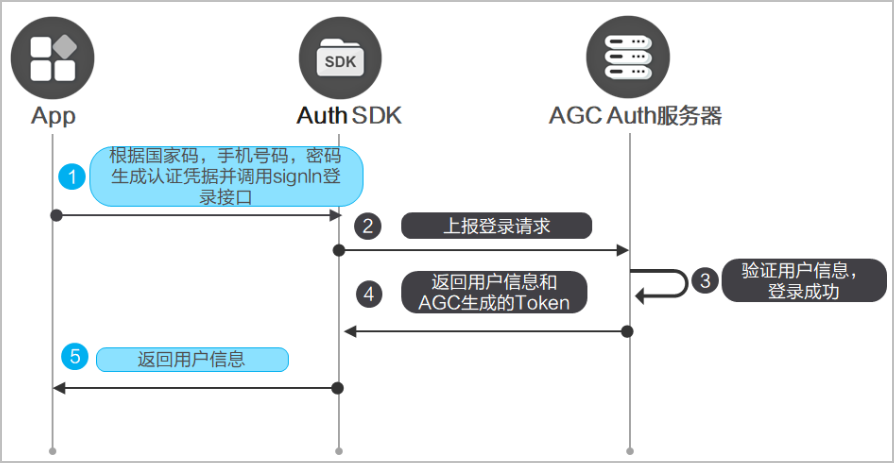
调用[PhoneUserBuilder](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-phoneuser-0000001288100610" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)生成PhoneUser，然后调用[AGConnectAuth.createPhoneUser](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section11815319133" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)注册用户。注册成功后，系统会自动登录，无需再次调用登录接口。

* 1. let user = new PhoneUserBuilder()
  2. .setCountryCode("countryCode")
  3. .setPhoneNumber("phoneNumber")
  4. .setPassword("password") //可以给用户设置初始密码。填写后后续可以用密码来登录
  5. .setVerifyCode('verifyCode')
  6. .build();
  7. agconnect.auth().createPhoneUser(user)
  8. .then(result => {
  9. // 创建用户成功
  10. })
  11. .catch(error => {
  12. // 创建用户失败
  13. })

3.登录成功后可以调用[AGConnectAuth.getCurrentUser](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section87068861218" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)获取用户帐号数据。

* 1. agconnect.auth().getCurrentUser();

## （二）密码登录



1.在应用的登录界面，初始化[AGConnectAuth](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实例，获取AGC的用户信息，检查是否有已经登录的用户。如果有，则可以直接进入用户界面，否则显示登录界面。

* 1. agconnect.auth().getCurrentUser().then(user=>{
  2. if(user){
  3. //业务逻辑
  4. }
  5. });

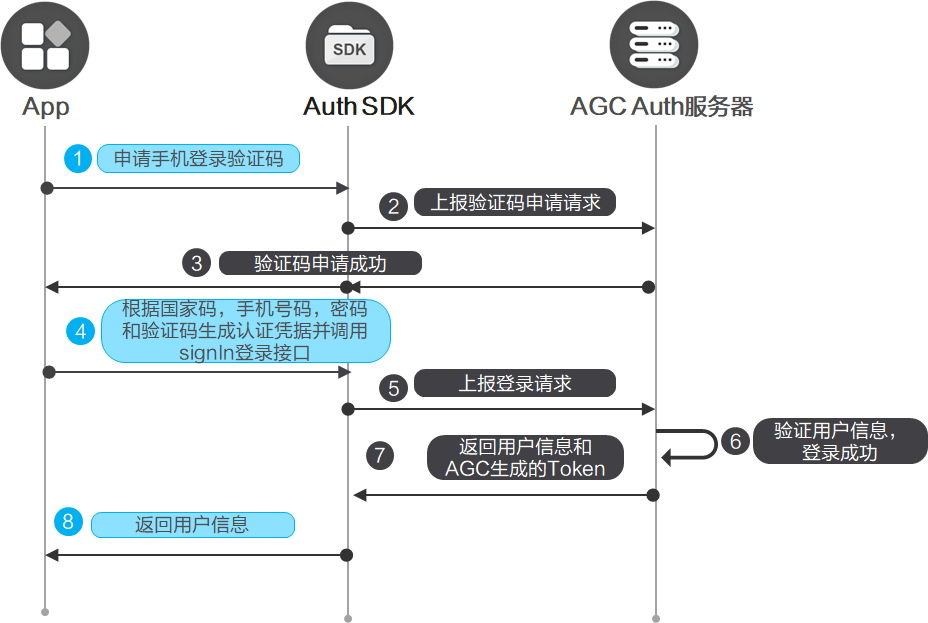
2.调用[PhoneAuthProvider.credentialWithPassword](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-phoneauthprovider-0000001340420421" \l "section571196104114" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)采用密码方式创建手机帐号凭证。

* 1. credential = agconnect.auth().PhoneAuthProvider.credentialWithPassword(countryCode,phoneNumber,password);

3.获取credential成功后，调用[AGConnectAuth.signIn](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section136957141012" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实现登录。

* 1. agconnect.auth().signIn(credential)
  2. .then(user => {
  3. //登录成功
  4. }).catch(error => {
  5. //登录失败
  6. });

## （三）验证码登录



1.在应用的登录界面，初始化[AGConnectAuth](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实例，获取AGC的用户信息，检查是否有已经登录的用户。如果有，则可以直接进入用户界面，否则显示登录界面。

* 1. agconnect.auth().getCurrentUser().then(user=>{
  2. if(user){
  3. //业务逻辑
  4. }
  5. });

2.调用[AGConnectAuth.requestPhoneVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section154023210148" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请手机登录验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestPhoneVerifyCode(countryCode, phoneNumber,verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

3.调用[PhoneAuthProvider.credentialWithVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-phoneauthprovider-0000001340420421" \l "section2631133144120" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)采用验证码方式创建手机帐号凭证。

* 1. credential = agconnect.auth().PhoneAuthProvider.credentialWithVerifyCode(countryCode, phoneNumber, verifyCode);

4.获取credential成功后，调用[AGConnectAuth.signIn](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section136957141012" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实现登录。

* 1. agconnect.auth().signIn(credential)
  2. .then(user => {
  3. //登录成功
  4. }).catch(error => {
  5. //登录失败
  6. });

## （四）修改手机号码

注意，修改手机号码需要用户处于登录状态。

1.调用[AGConnectAuth.requestPhoneVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section154023210148" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestPhoneVerifyCode(countryCode,phoneNumber,verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectUser.updatePhone](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section11672161715314" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)修改手机号码。

* 1. user.updatePhone(countryCode, newPhone, newVerifyCode, lang);

说明：对于修改手机号码操作，要求用户必须在5分钟内登录过应用才能执行。若登录已超时，请参见[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)先完成重认证。

## （五）修改密码

注意：修改密码时需要用户处于登录状态。

1.调用[AGConnectAuth.requestPhoneVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section154023210148" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.RESET\_PASSWORD)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestPhoneVerifyCode(countryCode,phoneNumber,verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectUser.updatePassword](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section152730591310" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)修改密码。

* 1. // 手机provider为11
  2. user.updatePassword(newPassword, verifyCode, AGConnectAuthCredentialProvider.Phone\_Provider);

说明：对于修改手机密码操作，要求用户必须在5分钟内登录过应用才能执行。若登录已超时，请参见[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)先完成重认证。

## （六）重置密码

注意：重置密码时用户可以不登录。

1.调用[AGConnectAuth.requestPhoneVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section154023210148" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.RESET\_PASSWORD)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestPhoneVerifyCode(countryCode,phoneNumber,verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectAuth.resetPasswordByPhone](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section9778844181618" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)重置密码。

* 1. agconnect.auth().resetPasswordByPhone(countryCode, phoneNumber, newPassword, verifyCode);

## （七）更多信息

如果想让用户可以使用多个帐号登录您的应用，可以[将多个帐号进行关联](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-associated-0000001310862120)。

当用户不需要使用应用，或者需要切换其他帐号登录认证，可以先执行[登出](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-logout-0000001339851721)。

当用户需要注销当前用户，可以进行[销户](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-deregistration-0000001339452249)。

对于销户、修改密码、关联帐号以及重置手机帐号和邮箱帐号等敏感操作，为了提高安全性，需要用户必须在5分钟内登录过才能执行。如果用户执行敏感操作时登录超过5分钟，需要[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)后再执行敏感操作。

可以参考[异常处理](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-troubleshooting-0000001339732413)实现自己的异常处理机制，从而减少异常情况的发生。

可以使用云函数触发器来接收用户注册、登录、销户等关键事件，从而[扩展认证服务的能力](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-extension-0000001053372708)。

# 二、登录认证邮箱

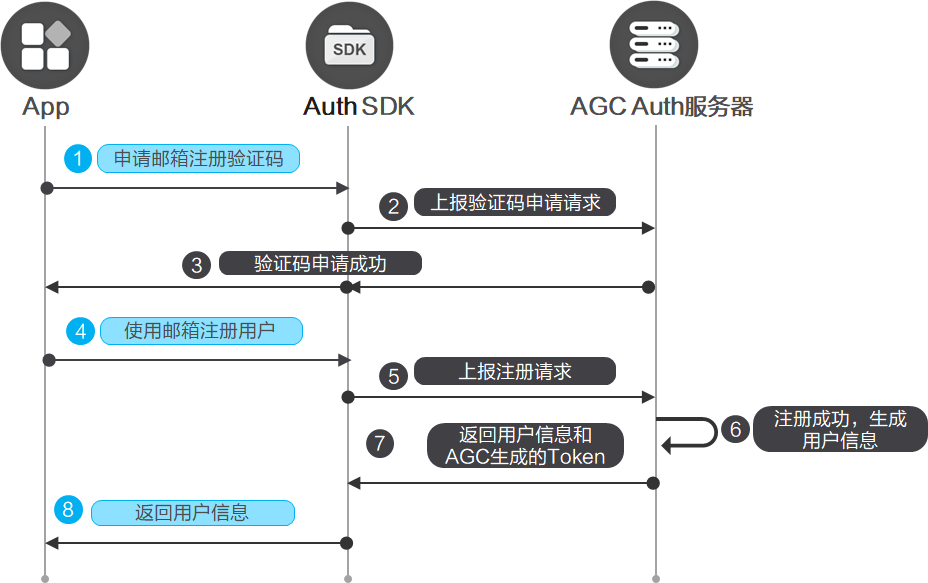
可以在应用中集成邮箱帐号认证方式，您的用户可以使用“邮箱地址+密码”或者“邮箱地址+验证码”的方式来登录您的应用。

## （一）前提条件

需要在AGC控制台[开通认证服务](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-enable-service-0000001274125746)。

需要先在您的应用中[集成SDK](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-integration-0000001339732409)。

## （二）注册



1.在使用邮箱注册之前，需要先验证邮箱，确保该邮箱帐户归您所有。

调用[AGConnectAuth.requestEmailVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section18901230101318" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestEmailVerifyCode('email',verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.使用邮箱帐号注册用户。

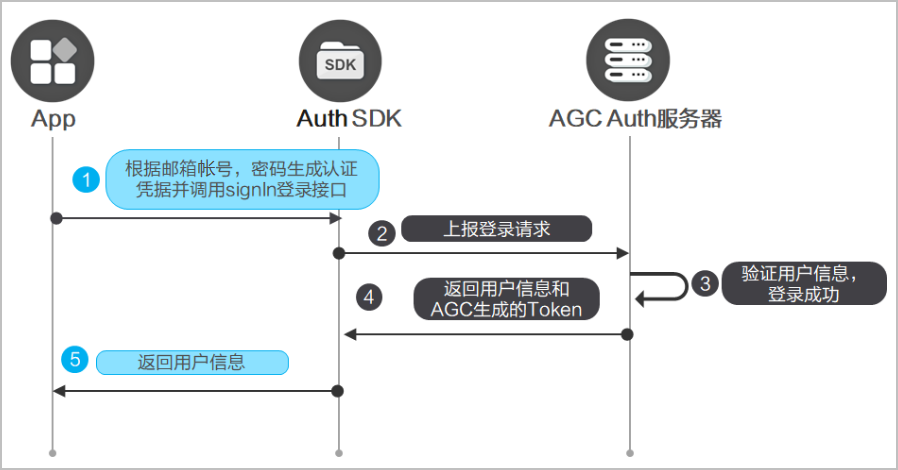
调用[EmailUserBuilder](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-emailuser-0000001287940690" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)生成EmailUser，然后调用[AGConnectAuth.createEmailUser](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section20867736181218" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)注册用户。注册成功后，系统会自动登录，无需再次调用登录接口。

* 1. let emailUser = new EmailUserBuilder()
  2. .setEmail("email")
  3. .setPassword("password") //可以给用户设置初始密码。填写后后续可以用密码来登录
  4. .setVerifyCode("verifyCode")
  5. .build();
  6. agconnect.auth().createEmailUser(emailUser)
  7. .then(result => {
  8. //创建帐号成功后，默认已登录
  9. })
  10. .catch(error => {
  11. //创建用户失败
  12. })

3.登录成功后可以调用[AGConnectAuth.getCurrentUser](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section87068861218" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)获取用户帐号数据。

* 1. agconnect.auth().getCurrentUser();

## （三）密码登录



1.在应用的登录界面，初始化[AGConnectAuth](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实例，获取AGC的用户信息，检查是否有已经登录的用户。如果有，则可以直接进入用户界面，否则显示登录界面。

* 1. agconnect.auth().getCurrentUser().then(user=>{
  2. if(user){
  3. //业务逻辑
  4. }
  5. });

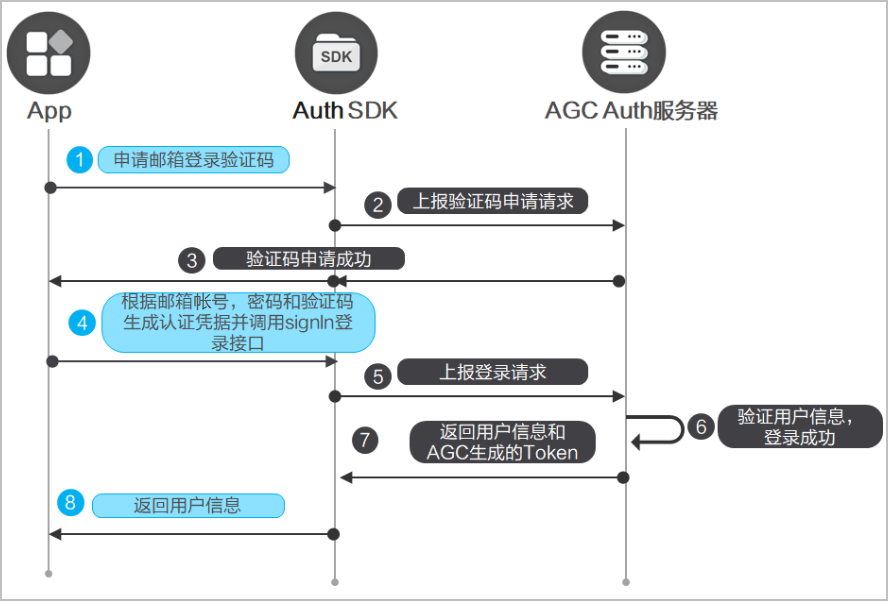
2.调用[EmailAuthProvider.credentialWithPassword](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-emailauthprovider-0000001340579873" \l "section973510438382" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)，采用密码方式创建邮箱帐号凭证（credential）。

* 1. credential = agconnect.auth().EmailAuthProvider.credentialWithPassword(emailStr, password);

3.获取credential成功后，调用[AGConnectAuth.signIn](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section136957141012" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实现登录。

* 1. agconnect.auth().signIn(credential)
  2. .then(user => {
  3. //登录成功
  4. }).catch(error => {
  5. //登录失败
  6. });

## （四）验证码登录



1.在应用的登录界面，初始化[AGConnectAuth](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实例，获取AGC的用户信息，检查是否有已经登录的用户。如果有，则可以直接进入用户界面，否则显示登录界面。

* 1. agconnect.auth().getCurrentUser().then(user=>{
  2. if(user){
  3. //业务逻辑
  4. }
  5. });

2.调用[AGConnectAuth.requestEmailVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section18901230101318" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请登录验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestEmailVerifyCode('email',verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

3.调用[EmailAuthProvider.credentialWithVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-emailauthprovider-0000001340579873" \l "section86232180395" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)采用验证码方式创建邮箱帐号凭证（credential）。

* 1. credential = agconnect.auth().EmailAuthProvider.credentialWithVerifyCode(emailStr, password, verifyCode);

password参数可以不传，如果同时输入了密码和验证码，则会对密码和验证码都做验证。

4.获取credential成功后，调用[AGConnectAuth.signIn](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section136957141012" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)实现登录。

* 1. agconnect.auth().signIn(credential)
  2. .then(user => {
  3. //登录成功
  4. }).catch(error => {
  5. //登录失败
  6. });

## （五）修改邮箱地址

注意：修改邮箱时需要用户处于登录状态。

1.调用[AGConnectAuth.requestEmailVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section18901230101318" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.REGISTER\_LOGIN)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestEmailVerifyCode('email',verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectUser.updateEmail](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section7264195173019" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)修改邮箱地址。

* 1. user.updateEmail(newEmail, newVerifyCode, lang);

说明：对于修改邮箱地址操作，要求用户必须在5分钟内登录过应用才能执行。若登录已超时，请参见[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)先完成重认证。

## （六）修改密码

注意：修改密码时需要用户处于登录状态。

1.调用[AGConnectAuth.requestEmailVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section18901230101318" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.RESET\_PASSWORD)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestEmailVerifyCode('email',verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectUser.updatePassword](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section152730591310" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)修改密码。

* 1. // 邮箱provider为12
  2. user.updatePassword(newPassword, verifyCode, provider);

说明：对于修改邮箱密码操作，要求用户必须在5分钟内登录过应用才能执行。若登录已超时，请参见[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)先完成重认证。

## （七）重置密码

注意：重置密码时用户可以不登录。

1.调用[AGConnectAuth.requestEmailVerifyCode](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section18901230101318" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)申请验证码。

* 1. let verifyCodeSettings = new VerifyCodeSettingBuilder()
  2. .setAction(VerifyCodeAction.RESET\_PASSWORD)
  3. .setLang('zh\_CN')
  4. .setSendInterval(60)
  5. .build();
  6. agconnect.auth().requestEmailVerifyCode('email',verifyCodeSettings)
  7. .then(verifyCodeResult => {
  8. //验证码申请成功
  9. }).catch(error => {
  10. //验证码申请失败
  11. });

2.调用[AGConnectAuth.resetPasswordByEmail](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section1053817349175" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)重置密码。

* 1. agconnect.auth().resetPasswordByEmail(email, newPassword, verifyCode);

## （八）更多信息

如果想让用户可以使用多个帐号登录您的应用，可以[将多个帐号进行关联](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-associated-0000001310862120)。

当用户不需要使用应用，或者需要切换其他帐号登录认证，可以先执行[登出](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-logout-0000001339851721)。

当用户需要注销当前用户，可以进行[销户](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-deregistration-0000001339452249)。

对于销户、修改密码、关联帐号以及重置手机帐号和邮箱帐号等敏感操作，为了提高安全性，需要用户必须在5分钟内登录过才能执行。如果用户执行敏感操作时登录超过5分钟，需要[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)后再执行敏感操作。

您可以参考[异常处理](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-troubleshooting-0000001339732413)实现自己的异常处理机制，从而减少异常情况的发生。

您可以使用云函数触发器来接收用户注册、登录、销户等关键事件，从而[扩展认证服务的能力](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-extension-0000001053372708)。

# 登录认证关联帐号

# （一）前提条件

您需要在AGC控制台[开通认证服务](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-enable-service-0000001274125746)。

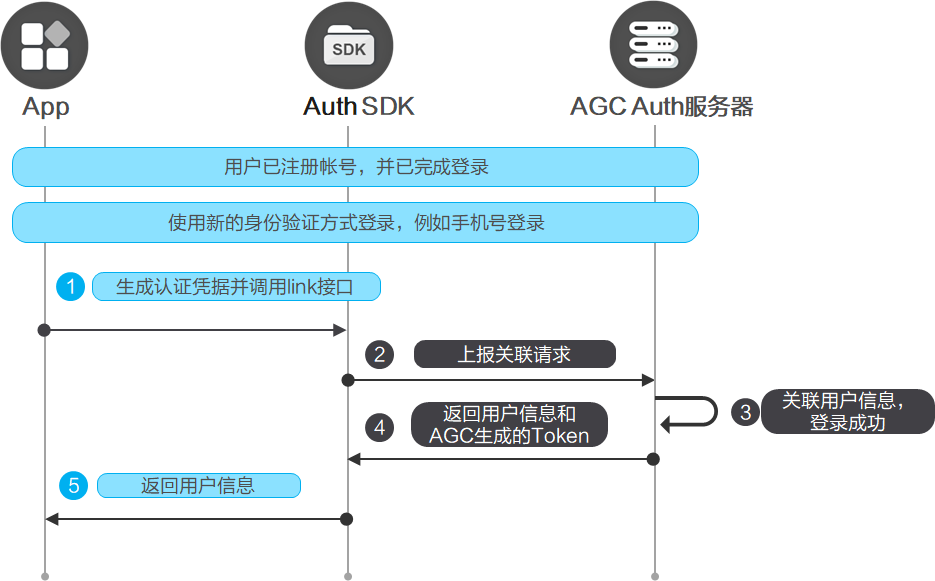
您需要先在您的应用中[集成SDK](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-integration-0000001339732409)。

## （二）将身份验证提供方凭据与用户帐号关联

可以将身份验证提供方凭据关联至现有用户帐号，允许用户使用多个身份验证提供方服务登录您的应用。无论用户使用哪个帐号登录，均可通过同一AGC用户ID识别用户。例如，使用手机帐号登录的用户可以关联邮箱帐号，以后便可使用这两种方法中的任意一种登录。

说明：关联帐号前，需要为应用增加对两个或多个身份验证提供方（可能包括匿名身份验证）的支持。

关联的认证方式只能有一个帐号，例如手机帐号关联邮箱帐号，只能关联一个邮箱帐号，不能关联多个。另外被关联的帐号需要没有登录过应用，例如已经通过认证服务登录过的邮箱帐号也无法进行关联。



1.使用任意身份验证提供方让用户登录。

2.按照新身份验证提供方的登录流程逐步进行，直到调用[AGConnectAuth.signIn](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agcauth-0000001340660285" \l "section136957141012" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)方法前停止。

3.为新的身份验证提供方获取[AGConnectAuthCredential](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectauthcredential-0000001288100606" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)，示例如下。

* 1. let credential;
  2. // 用于关联手机号的credential
  3. credential = agconnect.auth().PhoneAuthProvider.credentialWithVerifyCode(countryCode, phonenumber, verifyCode);
  4. // 用于关联邮箱的credential
  5. credential = agconnect.auth().EmailAuthProvider.credentialWithVerifyCode(email, verifyCode);

4.将credential传递到已登录用户的[AGConnectUser.link](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section155963910159" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)接口中。关联成功后，即可以使用任意一个提供方的凭证来登录相同的AGC帐号。

* 1. agconnect.auth().getCurrentUser().then(async user=>{
  2. await user.link(credential);
  3. });

## （三）取消身份验证提供方凭据与用户帐号的关联

也可以取消身份验证提供方凭据与用户帐号的关联，以便用户不再使用该身份验证提供方进行登录。

取消关联时，需提供要取消的身份验证提供方ID，然后调用[AGConnectUser.unlink](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-References/openharmony-auth-agconnectuser-0000001288260562" \l "section6182458151510" \t "https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/_blank)接口进行取消。

说明：当仅有一个身份验证提供方时不能进行取消关联操作。

1. agconnect.auth().getCurrentUser().then(async user=>{
2. await user.unlink(credentialProvider);
3. });

## （四）更多信息

当用户不需要使用应用，或者需要切换其他帐号登录认证，可以先执行[登出](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-logout-0000001339851721)。

当用户需要注销当前用户，可以进行[销户](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-deregistration-0000001339452249)。

对于销户、修改密码、关联帐号以及重置手机帐号和邮箱帐号等敏感操作，为了提高安全性，需要用户必须在5分钟内登录过才能执行。如果用户执行敏感操作时登录超过5分钟，需要[帐号重认证](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-reauthenticate-0000001286972452)后再执行敏感操作。

可以参考[异常处理](https://developer.huawei.com/consumer/cn/doc/development/AppGallery-connect-Guides/agc-auth-openharmony-troubleshooting-0000001339732413)实现自己的异常处理机制，从而减少异常情况的发生。