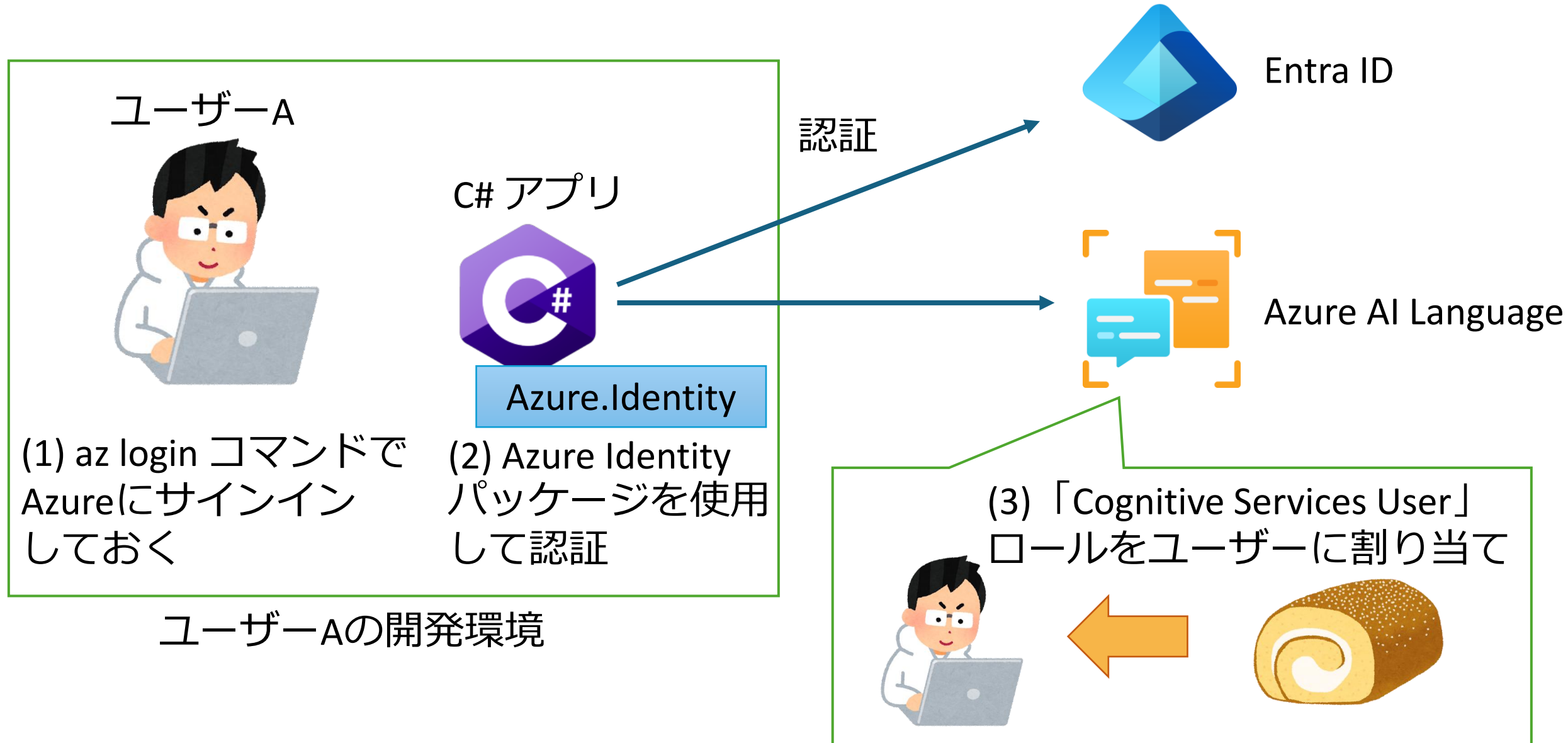


キーを使わずに サービスにアクセスするには？

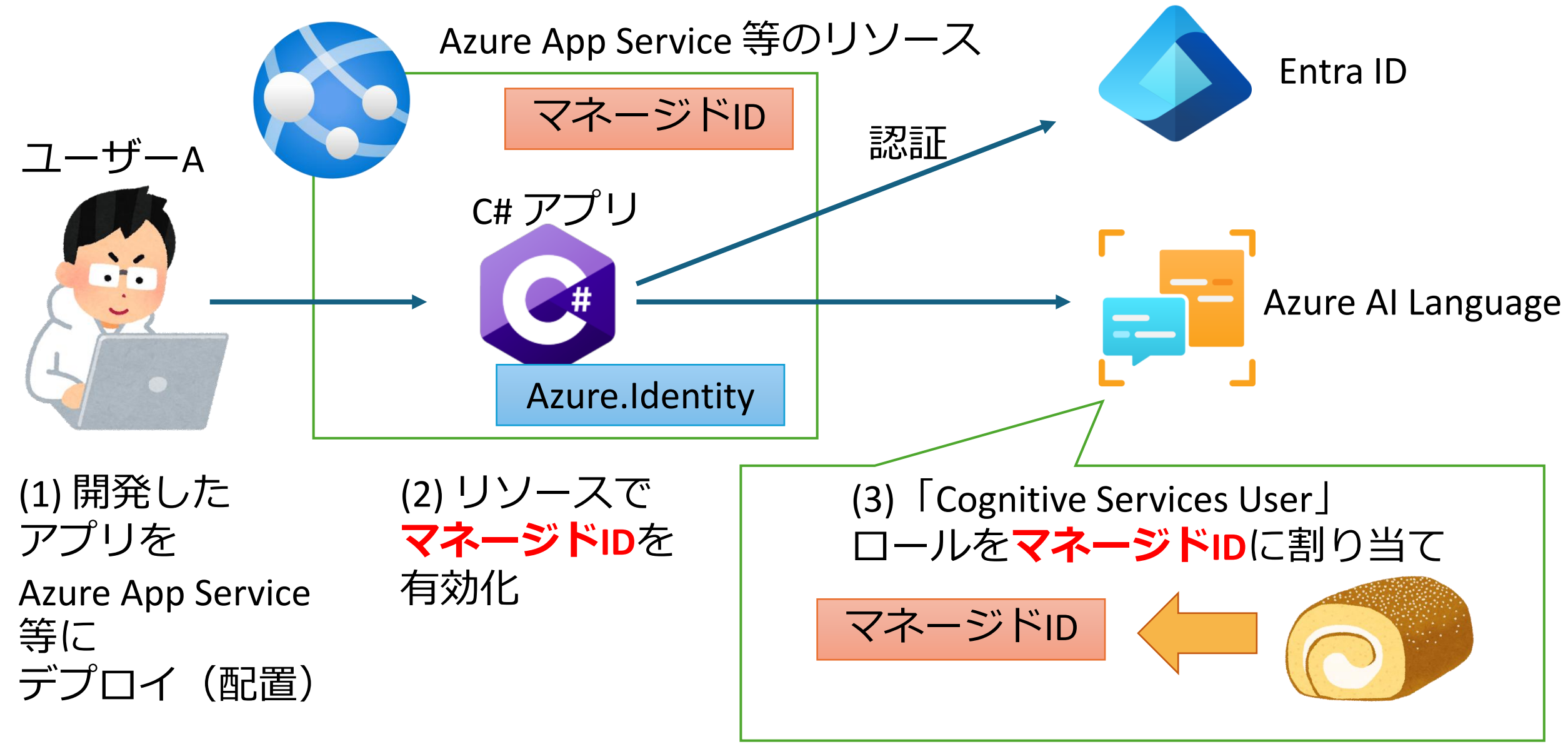
1. キー認証ではなく **Entra ID認証** を使用します。
 - プログラムで Azure.Identity パッケージを使用して認証を行います
 2. アクセス元となる Entra ID の ID（開発者ユーザーのユーザー ID や、アプリを実行する Azure リソースのマネージド ID）に対して、適切な **ロールを割り当て** します
 - Azure AI services にアクセスする際は「Cognitive Services User」ロールや「Cognitive Services Contributor (共同作成者)」ロールなどを使用
- これで、開発環境でも、Azure 環境（本番環境）でも、キーなしでアプリから Azure サービスにアクセスできます。

開発者のローカル環境での開発時:

開発者のユーザーIDで認証、**開発者に割り当てたロール**で承認



Azure環境でのアプリ運用時:
リソースに割り当てたマネージドIDで認証、マネージドIDに割り当てたロールで承認



Azure Identityパッケージ（と、アクセス先のAzureサービスのSDK）をプロジェクトに追加

```
dotnet add package Azure.Identity  
dotnet add package Azure.AI.Language.Text --version 1.0.0-beta.2
```

Azure Identityを使用して認証

```
using Azure.Identity;  
using Azure.AI.Language.Text;  
  
var endpoint = new Uri("エンドポイント");  
var credential = new DefaultAzureCredential();  
var client = new TextAnalyticsClient(endpoint, credential);  
  
...
```

アクセス先の
サービスの
キーが不要

Azure Identityパッケージ（と、アクセス

```
dotnet add package Azure.Identity
dotnet add package Azure.AI.Language.Text
```

Azure Identityを使用して認証

```
using Azure.Identity;
using Azure.AI.Language.Text;

var endpoint = new Uri("エンドポイント");
var credential = new DefaultAzureCredential();
var client = new TextAnalyticsClient(endpoint, credential);

...
```

開発環境でこのコードが実行される場合、Azure CLI（az login）のサインイン情報（等）、つまり開発者のユーザーIDを利用して、Entra ID 認証が行われる。

Azure環境でこのコードが実行される場合は、このコードが実行されているリソース（Azure App Service 等）のマネージドIDを使用して、Entra ID認証が行われる。