**FirstNet** SMS for Microsoft Entra

# Setup guide

How your administrator turns on FirstNet SMS sign-in codes for Microsoft Entra ID

November 2026 · Enable-FirstNetSms.ps1 at commit aab3dcf

## What this does

When your users sign in with multifactor authentication and choose a text message, Microsoft Entra ID creates the one-time code. With FirstNet as your phone provider, **FirstNet delivers that SMS** instead of Microsoft. Entra still checks the code the user types.

This uses Entra's external phone provider feature. A small Microsoft Function runs in **your** Azure subscription and passes each code to FirstNet over an authenticated connection. FirstNet never gets access to your tenant.

Your administrator turns it on with FirstNet's enablement script, in four short runs:

1. **Deploy:** the script checks your tenant, deploys the Function and prints a **setup code**.
2. **Approve:** you email the setup code to FirstNet, and FirstNet approves it.
3. **Activate:** the script turns on FirstNet SMS for a small pilot group.
4. **Widen:** once the pilot works, the script widens it to more users.

## Before you start

- **Licences:** Microsoft Entra ID P1 or P2 for the users who will get SMS codes.
- **Azure:** an enabled Azure subscription in the same tenant, where you are **Owner**. The Owner role must have no conditions, because the setup assigns roles. If your Owner assignment has a condition, ask for one with "Allow user to assign all roles".
- **Directory roles:** Global Administrator, or both Application Administrator and Privileged Role Administrator. For the Activate step you also need Authentication Policy Administrator. If you use Privileged Identity Management (PIM), activate the roles first.
- **Where to run it:** Azure Cloud Shell in **PowerShell** (`https://shell.azure.com`). It already has the Azure CLI and Microsoft Graph PowerShell.
- **Cost:** the Function keeps one instance always ready, so the first sign-in after a quiet period is not slow. That is a small fixed monthly cost in your subscription.

## Step 1. Run the script

Open `https://shell.azure.com`, sign in as your administrator and choose **PowerShell**. Then download and run the script:

```
iwr https://raw.githubusercontent.com/FirstNet-Technology-Services-PTY-Ltd/
ExternalPhoneProvider-AzureFunction-Sample/aab3dcf/setup/Enable-FirstNetSms.ps1 -
OutFile Enable-FirstNetSms.ps1; ./Enable-FirstNetSms.ps1
```

This is one command on a single line. The easiest way is to copy it from the portal (**mfa.firstnet.co.za**), which shows it ready to paste.

You can also download the script from the FirstNet portal, upload it in Cloud Shell (**Manage files > Upload**) and run `./Enable-FirstNetSms.ps1`. It is the same file.

The script:

- Asks which subscription to use, if you have more than one.
- Checks the prerequisites and marks each one `[ OK ]` or `[FAIL]`, with how to fix it. If anything fails, nothing is created: fix it and run the script again.
- Runs Microsoft's setup with FirstNet's settings. **Review its plan and answer Y.** If it asks which application to use, press **Enter** to create a new one.
- Keeps one Function instance always ready.
- Prints a **setup code** that starts with `FNSMS1.`

The Function is deployed in **South Africa North** by default. To use another Azure region, add the option below. If you do, tell FirstNet, because FirstNet must allow your Function's address.

```
./Enable-FirstNetSms.ps1 -Location <region>
```

> If the setup stops part-way, fix the cause and run the script again. It reuses what the first run created. To see the setup code again later, run it with the **SetupCode** option, as below.

```
./Enable-FirstNetSms.ps1 -SetupCode
```

## Step 2. Send FirstNet the setup code

Email the setup code to **sales@firstnet.co.za**. It contains only identifiers: your tenant ID, the Function's application ID and address, your organisation's name and your contact address. **It contains no passwords or secrets.**

FirstNet checks the code, approves it and lets you know. Wait for that confirmation before Step 4: until then, FirstNet refuses your Function's requests.

## Step 3. Make sure users are asked for MFA

SMS codes are sent only when a user is asked for MFA and chooses a text message. Check two things for your pilot users:

- **A mobile number is registered** for SMS sign-in on each pilot user.
- **Users are actually prompted.** With security defaults on, users mostly use Microsoft Authenticator and are rarely asked for SMS. For the pilot, use a Conditional Access policy that requires MFA for the pilot group instead. Before you turn security defaults off, make sure Conditional Access policies require MFA for all your users and administrators.

## Step 4. Turn it on for a pilot group

When FirstNet confirms, run this in Cloud Shell, with one or two of your own test users instead of the examples:

```
./Enable-FirstNetSms.ps1 -Activate -PilotUsers user1@contoso.com,user2@contoso.com
```

- The script asks you to confirm that FirstNet has approved your setup code.
- It creates the security group **FirstNet-SMS-Pilot** (choose another name with `-PilotGroup`) and adds the users.
- It **backs up your SMS sign-in policy** and prints the backup's file name. Download a copy from Cloud Shell (**Manage files > Download**) and keep it.
- It shows the change and asks before applying it.

If SMS sign-in is already in use in your tenant, the script won't narrow it to the pilot group, because that would switch SMS off for everyone else. Instead it keeps your existing SMS users and sends a share of their SMS sign-ins through FirstNet. Add `-Percent 5`, for example. The share is counted in sign-ins, not users, and it's a starting point: see Step 5.

Changes can take a few minutes to reach sign-ins (Microsoft says up to an hour). Then a pilot user signs in and chooses a text message. Tell FirstNet the time, so FirstNet can check the SMS on its side.

## Step 5. Widen the rollout

When the pilot works, send more SMS sign-ins through FirstNet, for example a quarter first and then all of them:

```
./Enable-FirstNetSms.ps1 -Expand -Percent 25
./Enable-FirstNetSms.ps1 -Expand -Percent 100 -AllUsers
```

`-AllUsers` makes SMS sign-in available to all users, not only the pilot group.

**Reach 100% well before 1 February 2027.** Microsoft retires its own SMS and voice codes for most users on that date (Global Administrators and external users on 1 July 2027). A sign-in that isn't routed through FirstNet doesn't fall back to FirstNet when Microsoft's SMS is blocked or stops, so it can fail. Until you reach 100%, some users can still miss their code.

## Voice calls (optional)

FirstNet can also deliver codes by voice call, through the same Function and app. Add `-Channel Both` to the `-Activate` and `-Expand` commands to route SMS and voice through FirstNet, or run them with `-Channel Voice` later for voice alone:

```
./Enable-FirstNetSms.ps1 -Activate -Channel Voice -PilotUsers user1@contoso.com
./Enable-FirstNetSms.ps1 -Expand -Channel Voice -Percent 100 -AllUsers
```

The script backs up the voice policy to its own file (`voice-policy-backup-*.json`); `-Rollback` with that file undoes it. Microsoft retires its own voice codes on the same date as SMS, so take voice to 100% before 1 February 2027 too if you keep it.

## Undo

To put your SMS sign-in policy back exactly as it was, use the backup file from Step 4:

```
./Enable-FirstNetSms.ps1 -Rollback '<path to the backup file>'
```

Microsoft then sends SMS codes again, until it retires its own SMS (1 February 2027 for most users). To remove everything, also delete the resource group the script created in your subscription (its name starts with `fnetsms-epp-rg-`), and tell FirstNet so it can remove your access.

## If something goes wrong

- **[FAIL] Owner of the subscription:** you need the Owner role without a condition (see "Before you start").
- **[FAIL] Directory roles:** you are missing a role. If you use PIM, activate the roles and run the script again.
- **[FAIL] Flex Consumption available:** pick another region with `-Location`, and tell FirstNet.
- **"SMS is already routed to another provider":** another provider is set up. Roll that back first; the script changed nothing.
- **A user didn't get a code:** in the Entra admin center, open the user's sign-in log entry and copy its **correlation ID** and time. Send them to FirstNet.

## Your usage and spend

Once you're set up, your administrators can sign in to the FirstNet customer portal at **mfa.firstnet.co.za** to see the codes sent, your prices, this month's spend and your budget. For help at any step, email **sales@firstnet.co.za**.