

Get allowlisted

Once you have built your Snap, tested it, and published it to npm, you can make it available to MetaMask users.

As part of the MetaMask Snaps Open Beta launch, individual Snaps must be put on an allowlist before users can install them. This means that at this time, only selected, reviewed Snaps can be installed. In the future, this system will be opened up.

Important By including a Snap on the allowlist, Consensys is not endorsing, recommending, or guaranteeing the safety of a Snap for your use or use for any reason. Always do your own research before installing a Snap.

Prerequisites

To be added to the allowlist, a Snap must:

- Have publicly available source code.
- You don't need to publish your code with an open source license, but users should be able to read the source code of the Snap package.
- Be [published](#) to npm.
- Not impair our compliance with laws or regulations.

Furthermore, if your Snap uses any of the following API methods related to key management, you must provide evidence of a third-party audit from an approved auditor:

- [snap_getBip32Entropy](#)
- [snap_getBip32PublicKey](#)
- [snap_getBip44Entropy](#)
- [snap_getEntropy](#)
- [snap_manageAccounts](#)

The audit must cover the Snap source code that is to run within the Snaps system, and any modules used for key management. You must provide the commit that was audited and the commit that has any fixes documented in the audit report.

A list of approved third-party auditors and details about the audit process are available on the [MetaMask Snaps Builder Engagement Program](#).

Steps

1. Submit your Snap

Fill out the [MetaMask Snaps Directory Information form](#). The form requests information about your Snap, including the following:

- Snap name
 - The name of your Snap.
- This must match the `proposedName` field in the Snap [manifest file](#).
- .
- You cannot use these words in the name: "MetaMask," "Snap," "Meta," or "Mask."
- Snap builder name and URL
 - The company, project, or personal site for your brand, where users can learn more about you.
- Snap website URL
 - A website where users can interact with your Snap.
- If your Snap does not require a website to be used, you can leave this blank.
- If your Snap works with multiple websites, you can include the URLs of additional websites in the long description, but this URL should be an official website designed to interact with your Snap.
- Snap short description
 - A one or two sentence description of your Snap.
- Try not to say "is a MetaMask Snap" – users already know this!
- Snap long description

- - A description of your Snap's features and how to use them.
- You can use line breaks, lists, and URLs.
- You cannot use HTML.
- If applicable, describe quick steps to onboard and use the Snap.
- For example: After installing the Snap, visit the companion dapp at <https://voyager-snap.linea.build>
- to connect an account and track your Linea Voyage progress.
- GitHub repository and npm package URLs
- - The public GitHub repo that hosts your Snap's
- source code, and the npm package of your [published Snap](#)
- .
- If your Snap's source code is hosted on a different site, such as GitLab, you can link to that instead.
- Snap version number to be allowlisted
- - The Snap version number as specified in package.json
- and snap.manifest.json
- .
- Make sure that the version numbers match and that the Snap has been built with the correct shasum
- .
- Snap auditor and audit report
- - A PDF or URL of the [required audit report](#)
- ,
- if your Snap uses one or more of the key management API methods.
- If your Snap doesn't require an audit, leave this field blank.
- note
- The audit report will be made public.
- You can ask your auditor to publish the report on their website and provide the link to us.
- Customer support details
- - [Customer support information](#)
- to ensure a smooth user experience for your Snap.
- This allows MetaMask to escalate any issues that a user might encounter with your Snap.
- The escalation contact will be kept confidential within MetaMask, and the rest of the information
- will be public.
- You must provide an escalation contact and at least one other customer support item.
- Images
- - Screenshots or promotional images to help users get an idea of what your Snap can do.
- Here's your opportunity to provide compelling previews of your Snap!
- Demo video
- - A video walking through how to use your Snap.
- This will help MetaMask review your Snap, and might also be used by the MetaMask marketing team.

2. Allowlist review

Your Snap will be reviewed by the MetaMask Snaps team to ensure it is functional and well-designed. If the Snap requires an audit, the team will review the audit report to ensure that all vulnerabilities with medium or higher risk have been addressed. All Snaps require at least two approvals to be allowlisted.

3. Directory listing

Once your Snap is on the allowlist, it will appear in the [MetaMask Snaps Directory](#) . You can direct users to the directory to find and install your Snap.

4. Distribute your Snap

You can deploy a companion dapp where users can learn about your Snap and install it, or you can integrate your Snap with your existing dapp.

If your Snap is designed to communicate with dapps, you can encourage other dapp developers to [connect to your Snap](#) .

note While testing your Snap, you might have designed your dapp to require MetaMask Flask. Once your Snap is allowlisted, you should update your dapp to support any flavor of MetaMask, and show the orange MetaMask logo instead of the purple Flask logo.

5. Update your Snap

The allowlist uses strict versioning for all Snaps. After publishing a new version of your Snap to npm, you must re-submit it for allowlisting by filling out the [MetaMask Snaps Directory Information Update form](#) . Users will not be able to install a new version until it is allowlisted.

You can also update any information about your Snap using the form. For fields that you don't need to update, you can leave them blank or enter "N/A." When providing the new version to be allowlisted, you should also note whether previous versions of your Snap should be removed from the allowlist (effectively, replaced with the new version).

[Edit this page](#)