

# Username considerations for external authentication

## In this article

About usernames with external authentication

About usernames for managed user accounts

About username normalization

Resolving username problems

---

When you use Enterprise Managed Users, GitHub Enterprise Cloud follows certain rules to determine the username for each user account in your enterprise.

**Note:** This article only applies to Enterprise Managed Users. If you use GitHub Enterprise Cloud without Enterprise Managed Users, usernames are created by users, not GitHub.

## About usernames with external authentication [↗](#)

If you use an enterprise with Enterprise Managed Users, members of your enterprise authenticate to access GitHub through your SAML identity provider (IdP). For more information, see "[About Enterprise Managed Users](#)" and "[About authentication for your enterprise](#)."

GitHub automatically creates a username for each person when their user account is provisioned via SCIM, by normalizing an identifier provided by your IdP, then adding an underscore and short code. If multiple identifiers are normalized into the same username, a username conflict occurs, and only the first user account is created. You can resolve username problems by making a change in your IdP so that the normalized usernames will be unique and within the 39-character limit.

**Note:** Because GitHub adds an underscore and short code to the normalized identifier provided by your IdP when creating each username, conflicts can only occur within each enterprise with managed users. Managed user accounts can share IdP identifiers or email addresses with other user accounts on GitHub.com that are outside the enterprise.

## About usernames for managed user accounts [↗](#)

When your enterprise with managed users is created, you will choose a short code that will be used as the suffix for your enterprise members' usernames. The short code must be unique to your enterprise, a three-to-eight character alphanumeric string, and contain no special characters. The setup user who configures SAML SSO has a username in the format of **@SHORT-CODE\_admin**.

When you provision a new user from your identity provider, the new managed user account will have a GitHub username in the format of **@IDP-USERNAME\_SHORT-CODE**. The IDP-USERNAME component is formed by normalizing the SCIM `userName`

attribute value sent from the IdP.

Identity provider	GitHub username
Azure Active Directory (Azure AD)	IDP-USERNAME is formed by normalizing the characters preceding the @ character in the UPN (User Principal Name), which does not include the #EXT# for guest accounts.
Okta	IDP-USERNAME is the normalized username attribute provided by the IdP.

These rules may result in your IdP providing the same IDP-USERNAME for multiple users. For example, for Azure AD, the following UPNs will result in the same username:

- bob@contoso.com
- bob@fabrikam.com
- bob#EXT#fabrikamcom@contoso.com
- bob\_example#EXT#fabrikamcom@contoso.com
- bob\_example.com#EXT#fabrikamcom@contoso.com

This will cause a username conflict, and only the first user will be provisioned. For more information, see "[Resolving username problems](#)."

Username, including underscore and short code, must not exceed 39 characters.

## About username normalization

Username, for user accounts on GitHub.com can only contain alphanumeric characters and dashes ( - ).

When you configure SAML authentication, GitHub Enterprise Cloud uses the SCIM `userName` attribute value sent from the IdP to determine the username for the corresponding user account on GitHub.com. If this value includes unsupported characters, GitHub Enterprise Cloud will normalize the username per the following rules.

- 1 GitHub Enterprise Cloud will normalize any non-alphanumeric character in your account's username into a dash. For example, a username of `mona.the.octocat` will be normalized to `mona-the-octocat`. Note that normalized usernames also can't start or end with a dash. They also can't contain two consecutive dashes.
- 2 Usernames created from email addresses are created from the normalized characters that precede the @ character.
- 3 Usernames created from domain accounts are created from the normalized characters after the \ separator.
- 4 If multiple accounts are normalized into the same GitHub Enterprise Cloud username, only the first user account is created. Subsequent users with the same username won't be able to sign in. For more information, see "[Resolving username problems](#)."

## Examples of username normalization

Identifier on provider	Normalized username on GitHub	Result
The.Octocat	the-octocat_SHORT-CODE	This username is created successfully.

!The.Octocat	-the-octocat_SHORT-CODE	This username is not created, because it starts with a dash.
The.Octocat!	the-octocat-_SHORT-CODE	This username is not created, because it ends with a dash.
The!!Octocat	the--octocat_SHORT-CODE	This username is not created, because it contains two consecutive dashes.
The!Octocat	the-octocat_SHORT-CODE	This username is not created. Although the normalized username is valid, it already exists.
The.Octocat@example.com	the-octocat_SHORT-CODE	This username is not created. Although the normalized username is valid, it already exists.
internal\\The.Octocat	the-octocat_SHORT-CODE	This username is not created. Although the normalized username is valid, it already exists.
mona.lisa.the.octocat.from.github.united.states@example.com	mona-lisa-the-octocat-from-github-united-states_SHORT-CODE	This username is not created, because it exceeds the 39-character limit.

## Resolving username problems [🔗](#)

When a new user is being provisioned, if the username is longer than 39 characters (including underscore and short code), or conflicts with an existing user in the enterprise, the provisioning attempt will fail with a `409` error.

To resolve this problem, you must make one of the following changes in your IdP so that all normalized usernames will be within the character limit and unique.

- Change the `userName` attribute value for individual users that are causing problems
- Change the `userName` attribute mapping for all users
- Configure a custom `userName` attribute for all users

When you change the attribute mapping, usernames of existing managed user accounts will be updated, but nothing else about the accounts will change, including activity history.

**Note:** GitHub Support cannot provide assistance with customizing attribute mappings or configuring custom expressions. You can contact your IdP with any questions.

## Resolving username problems with Azure AD [🔗](#)

To resolve username problems in Azure AD, either modify the User Principal Name value for the conflicting user or modify the attribute mapping for the `userName` attribute. If you modify the attribute mapping, you can choose an existing attribute or use an expression to ensure that all provisioned users have a unique normalized alias.

- 1 In Azure AD, open the GitHub Enterprise Managed User application.
- 2 In the left sidebar, click **Provisioning**.

- 3 Click **Edit Provisioning**.
- 4 Expand **Mappings**, then click **Provision Azure Active Directory Users**.
- 5 Click the GitHub `userName` attribute mapping.
- 6 Change the attribute mapping.
  - To map an existing attribute in Azure AD to the `userName` attribute in GitHub, click your desired attribute field. Then, save and wait for a provisioning cycle to occur within about 40 minutes.
  - To use an expression instead of an existing attribute, change the Mapping type to "Expression", then add a custom expression that will make this value unique for all users. For example, you could use `[FIRST NAME] - [LAST NAME] - [EMPLOYEE ID]`. For more information, see [Reference for writing expressions for attribute mappings in Azure Active Directory](#) in Microsoft Docs.

## Resolving username problems with Okta

To resolve username problems in Okta, update the attribute mapping settings for the GitHub Enterprise Managed User application.

- 1 In Okta, open the GitHub Enterprise Managed User application.
- 2 Click **Sign On**.
- 3 In the "Settings" section, click **Edit**.
- 4 Update the "Application username format."

## Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)