Splunk® Enterprise Securing Splunk Enterprise 8.0.1

Create authentication tokens

Generated: 12/29/2019 3:01 am
Create authentication tokens

Authentication tokens let users of Splunk platform environments access Representational State Transfer (REST) endpoint resources or use the Splunk CLI in those environments. You can create tokens in Splunk Web or use an API call to a REST endpoint on the instance where the tokens are to reside.

At this time, you can not use the CLI to create authentication tokens.

Prerequisites for token creation

- You must enable token authentication. See Enable or disable token authentication.
- Your Splunk platform account must satisfy at least one of the following criteria before you can create authentication tokens:
  - If you want to create tokens for yourself, your account must hold a role that has the `edit_tokens_own` capability.
  - If you want to create tokens for any user on the instance, your account must hold a role that has the `edit_tokens_all` capability. See About defining roles with capabilities for additional information on Splunk platform capabilities.
- You must be prepared to save or share the token immediately after you create it. You only have one opportunity to do so, and the full token cannot be recalled after you close the "New token" dialog box.

Supported user types for token creation

You can only create tokens for users that exist on the Splunk platform instance where you create the token. The users that exist on the instance depend on the authentication scheme that the instances uses:

- Native Splunk authentication: Where user accounts exist only on that specific instance
- Authentication through a Lightweight Directory Access Protocol (LDAP) server
- Authentication through a single-sign-on (SSO) scheme that uses Security Assertion Markup Language (SAML)

The Splunk platform confirms that the user you entered exists and raises an error message if the user does not exist.
Considerations for creating tokens on instances that use the LDAP authentication scheme

When you create a token on an instance that uses the LDAP authentication scheme, the LDAP server treats that creation as a login for LDAP caching purposes. The Splunk platform connects to the LDAP server to validate the user and any associated LDAP groups.

Considerations for creating tokens on instances that use the SAML authentication scheme

You might need to perform additional tasks before you can create tokens for users who use the SAML authentication scheme to access Splunk platform resources. This need depends on the identity provider (IdP) that the Splunk platform instance connects to with SAML.

- If the SAML IdP supports Attribute Query Requests (AQRs), then no additional work is required, and tokens that you create should be immediately usable by users who access a Splunk platform instance that uses the SAML scheme.
- If the SAML IdP does not support AQRs, then you must configure authentication extensions that interface with the IdP. See Configure SAML authentication extensions for additional information and instructions.

Save or share the token immediately after you create it

When you create an authentication token, for security purposes, you only have one opportunity to see the entire token. After you specify a user and audience for the token and click "Create", the token appears as a string of text in the "New Token" dialog box. You must copy this token and paste it into another document such as a text file before closing the "New Token" dialog box.

Token users must have the full token to authenticate without credentials. If you close the Create Token dialog box before saving the full token somewhere, then you must create a new token, as you cannot recover the one that you previously created.

Configure token expiry and "Not Before" settings

When you create a new token, you can set whether or not the token expires, and whether or not it is valid before a certain time. Both of these choices are optional, which means that you can configure a token to last forever and be available for use immediately.
If you want an authentication token to expire, you must set an expiry date and time for it. You can set an absolute time, such as "Friday, February 1, 2019 at 10:30", or you can set a relative time, which is a certain period of time from the current time, for example, three days from now, a week from now, or two months from now. Expiration times can never be in the past.

If you do not set an expiration time, then the Splunk platform uses the default global expiration time, which is "never". You can change this default. See Set a default relative token expiration time using configuration files.

If you do not want the token to be valid immediately, you can set a "Not Before" time for it. This means that even though you create the token now, it cannot be used until the Not Before time has passed. For example, if today is Friday, January 25, 2019, and you do not want the token to be used until the following Friday, you can set an absolute time of Friday, February 1, 2019 at 00:00, or a relative time of +7d (7 days from now). The "Not before" time can never be in the past, nor can it be after the expiration time.

Relative time versus absolute time

When you specify expiration and "Not before" times, you can specify an absolute time or a relative time. You can specify either type of time format for either type of time.

You indicate the absolute time by specifying a date and time, including year, month, day, hour, minute, and second. In Splunk Web, you can use the date picker to choose the appropriate date and time, or you can type it in. If you type it in, you must specify it in the following format:

`YYYY-MM-DDTHH:MM:SS[+HH:MM]`

The T between DD and HH is the actual letter T. The +HH:MM represents the time zone that you want to use, and is optional.

You indicate the relative time by specifying a string that represents an amount of time beyond the current time. In general, specify the following format:

`+<number>[s|m|h|d]@[s|m|h|d] [+<number>[s|m|h|d]]`

The letters s, m, h, and d are identifiers that represent seconds, minutes, hours, and days, respectively. You can also use the following words as identifiers:

- seconds, secs, sec
- minutes, mins, min
If you want to round down to a certain time, you can include the `@d` modifier in the time string. For example, if you want a token to expire 10 days from now at the beginning of the day, you can specify `+10d@d`. This string reads as "10 days from now, on the day."

You can also concatenate different identifiers. For example, to have a token expire 15 days and 5 hours, on the hour, from now, you can specify `+15d+5h@h`.

**Use Splunk Web to create authentication tokens**

1. In the system bar, click **Settings > Tokens**.
2. Click **New Token**. The "New Token" dialog box appears.

If you see a message that says you have not enabled token authentication, see Enable token authorization for instructions on how to enable token authorization.

3. In the "New Token" dialog, enter the Splunk platform user that you want to create the token for in the **User** field.
4. Enter a short description of the token purpose in the **Audience** field.
5. (Optional) In the **Expiration** drop down list, select one of **Absolute Time** or **Relative Time**. This selection determines what to enter in the text field below the drop down list.
   1. If you selected **Absolute Time**, then two text fields appear under the drop down.
      1. Enter a valid date into the first field. You can also click the field to select a date from a pop-up calendar.
      2. Enter a valid 24-hour time in the second field.
   2. Otherwise, one text field appears under the drop down list.
      1. Enter a string that represents how long after the current time you want the token to remain valid. For example, if you want the token to expire 10 days from now, enter `+10d` into this field.

6. (Optional) In the **Not Before** drop-down, select one of **Absolute Time** or **Relative Time**.
7. Repeat the step you used for the "Expiration" control. The "Not before" time can neither be in the past, nor can it be later than the "Expiration" time.
8. Click **Create**. The New Token window updates the **Token** field to show you the token that has been generated.
9. Select all of the token text in the field. Depending on your operating system and browser, you can click on the "Token" field, then either triple click or press Ctrl-A or Command-A on your keyboard.

Confirm that you have selected all of the token text. There are no further opportunities to see the whole token after you close the window.

10. Copy the text from the Token field.
11. Paste the token into a text file, e-mail, or other form of communication to the person you have authorized to use the token.

Confirm that you share the token only with those who you have authorized to use it. Anybody who has the full token can use it to authenticate.

12. Click Close.
13. Share the token with its authorized user.

Use REST to create authentication tokens

You can also create authentication tokens by making an API call to the services/authorization/tokens REST endpoint on a Splunk platform instance with the cURL command. The cURL command is not available on Windows PowerShell; instead you can use the Invoke-RestMethod command on PowerShell version 3.0 and higher.

If you have an existing valid token, you can use it rather than user credentials to authenticate when creating a new token. The user account that is associated with the token must satisfy the previously-described criteria for token creation.

See "Example API calls" later in this topic for examples on token creation.

1. Open a shell prompt.
2. Generate the token.

```
curl [-k] [-u <username>:<password>|-H "Authorization: Bearer <existing_token>" -X POST
https://<servername>:<management_port>/services/authorization/tokens?output_mode=json
--data name=<token_user> --data audience=<audience>
[--data-urlencode expires_on=[<relative_time>|<absolute_time>]]
[--data-urlencode not_before=[<relative_time>|<absolute_time>]]
```

See the syntax variable table that follows this procedure for a description of each variable in the syntax.
3. In the output that appears, look for the "token:" string. The text immediately after this string, up to the next " character, is the token.
4. Share the whole token with the person who is to use it.
5. Close the shell prompt.

**Sample output of token creation**

Following is sample output from a REST call to generate a token.

```json
{"links":{"create":"/services/authorization/tokens/_new"},"origin":"https://mysplunk.com:8089/services/authorization/token ... 8xqSiSunYGylupG114KujO3-E88TtM-Tf51qtVpJydR3JSJpPfaf17w"},
```

The token is embedded in the output:

```plaintext
eyJraWQiOiJzcGx1bmsuc2VjcmV0IiwibXVsdGluZyI6InRlbmdueyJ9.
```

**Syntax variable table**

Determine the meaning of the variables used in the previous syntax using the following table:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Meaning</th>
<th>Required?</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>The user that is creating the new token</td>
<td>If using a token: No</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Otherwise: Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If using a token: No</td>
</tr>
<tr>
<td>password</td>
<td>The password for the user that is creating the new token</td>
<td>Otherwise: Yes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If using a token: Yes</td>
</tr>
<tr>
<td>existing_token</td>
<td>A valid, existing token that has already been issued to the user that is creating the new token</td>
<td>Otherwise: No</td>
</tr>
<tr>
<td>servername</td>
<td>The Splunk platform instance where the token is to reside</td>
<td>Yes</td>
</tr>
<tr>
<td>management_port</td>
<td>The management port of the Splunk instance. Is usually 8089 but can be different depending on how the instance is configured.</td>
<td>Yes</td>
</tr>
<tr>
<td>token_user</td>
<td></td>
<td>Yes</td>
</tr>
</tbody>
</table>
Variable | Meaning | Required?
--- | --- | ---
**audience** | The purpose for which the token is being created. Splunk Enterprise uses this to let you group tokens that have been assigned to different users. | **Yes**

**expires_on** | An argument that specifies when a token expires. Expired tokens are invalid and authorization requests that contain them are rejected. | **No**

**not_before** | An argument that specifies a future time when the token is to become valid. "Not before" tokens are invalid until the "Not before" time has passed, and authorization requests that occur before that time with these kinds of tokens are rejected. | **No**

**absolute_time** | A string that represents a specific date and time. Must be in the format YYYY-MM-DDTHH:MM:SS[+HH:MM]. Include the time zone offset when able. For example: 2019-02-09T07:35:00+07:00 | **No, uses default expiration time if not included**

**relative_time** | A string that represents a period of time past the current time. Must be in the format +<number>[s][m][h][d]. For example, +90m means 90 minutes from now, +15d means 15 days from now. | **No**

**Example API calls for creating tokens**

The following example is of an administrator using their credentials to create a token for user "jdoe" with an expiration of 30 days from the current time:

curl -k -u admin:Ch#ng3d! -X POST https://splunk1.server.com:8089/services/authorization/tokens?output_mode=json --data name=jdoe --data audience=Users --data-urlencode expires_on=+30d

The following example is of an admin using a valid token to create another token for user "sallyjane" for the purpose "Managers" that uses the default token expiration time.
curl -X POST -H "Authorization: Bearer ejy23898hjkl2QJkJl..."
https://mysplunk.com:8089/services/authorization/tokens?output_mode=json
--data name=sallyjane --data audience=Managers

The following example is of an admin creating a token for user "bobd" on February 5, 2019. The admin wants the token to become valid on March 1, 2019 at midnight and expire around 60 days after that. The instance uses a nonstandard management port 44514:

curl -k -u admin:Ch#ng3d! -X POST
https://mysplunk.com:44514/services/authorization/tokens?output_mode=json
--data name=bobd --data audience=Accountants --data-urlencode
not_before=2019-03-01T00:00:00 --data-urlencode expires_on=+85d

The following example is of an admin creating a token for user "steveg" for the purpose "Operations". The token is not to become valid before 10 days, on the day, from the current time, and should expire 45 days, on the day, plus 2 hours, on the hour, after it becomes valid. The instance uses a nonstandard management port 38182:

curl -k -u admin:Ch#ng3d! -X POST
https://mysplunk.com:38182/services/authorization/tokens?output_mode=json
--data name=steveg --data audience=Operations --data-urlencode
not_before=+10d@d --data-urlencode expires_on=+55d@d+2h@h

**Use, manage, and delete tokens**

After you created tokens, you can do the following:

- Manage or delete tokens. See Manage or delete authentication tokens.
- Use tokens to authenticate. See Use authentication tokens.