



Cisco Finesse REST API with SSO Guide for Release 12.6(1)

First Published: 2021-04-24

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2010–2021 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1

Introduction 1

Single Sign-On 1

Single Sign-On—Get User Authentication Mode 2

Single Sign-On—Fetch Access Token 3

Single Sign-On—Refresh Existing Access Token 6

Client Integration 7



CHAPTER 1

Introduction

This document describes the procedure of the Finesse REST API authorization with Single Sign-On (SSO).

- [Single Sign-On, on page 1](#)
- [Single Sign-On—Get User Authentication Mode, on page 2](#)
- [Single Sign-On—Fetch Access Token, on page 3](#)
- [Single Sign-On—Refresh Existing Access Token, on page 6](#)
- [Client Integration, on page 7](#)

Single Sign-On

Single Sign-On (SSO) is a mechanism to authenticate users across software systems using a common LDAP identity and this common authentication service provides a token. Multiple applications use this token to authenticate the user across preconfigured applications.

Single Sign-On Components

The following are the SSO components:

Identity Provider (IdP)

- IdP is an application that creates, maintains, and manages identity information for users.
- IdP offers the user authentication as a service. Third-party applications (for example, web applications) outsource the user authentication mechanism to a trusted IdP which is configured within the Organization. For example, Active Directory Windows Server.

Cisco Identity Service (IdS)

- Cisco IdS is the common API endpoint for relaying requests to the IdP by generating the authentication token and validating it.
- Cisco IdS implements an authorization endpoint and token endpoint as part of its OAuth (Open Authorization) server implementation.

Token Types

The following are the token types:

- Access Token—It accesses protected resources. Clients are issued an access token that contains identity information for the user that is encrypted by default.



Note For an SSO enabled user, use the access token in the authorization header of the Finesse REST APIs.

Authorization: Bearer *<access token>*

- Refresh Token—It obtains a new access token before the current access token expires. The IdS generates the refresh token.

The refresh and access token are generated as a pair of tokens. When refreshing the access token, the pair of tokens provide an extra layer of security.

You can configure the expiry time of the refresh token and access token in the IdS administration. When the refresh token expires, you cannot refresh the access token.

Cisco Contact Center Components

The following are the Cisco Contact Center components that support SSO:

- Cisco Finesse
- Cisco Unified Intelligence Center

Single Sign-On—Get User Authentication Mode

The `UserAuthMode` API is available in Cisco Finesse Release 12.5(1) and higher. This API allows a client to get the authentication mode of a user in a Unified CCE deployment that is in hybrid mode (SSO and non-SSO).

UserAuthMode API Details

Administrators can enable or disable this API using the CLI:

```
utils finesse set_property webservice enableUserAuthMode {true|false}
```

By default, the CLI sets the value of this property as *true*.

This API uses either the username or `userId` and it does not require authentication.

| | |
|------------------------------|--|
| URI: | <code>https://<FQDN>/finesse/api/UserAuthMode/<username></code> <code>https://<FQDN>/finesse/api/UserAuthMode/<userId></code> |
| Example URI: | <code>https://finesse1.xyz.com/finesse/api/UserAuthMode/myName</code> <code>https://finesse1.xyz.com/finesse/api/UserAuthMode/1234</code> |
| Security Constraints: | All users can use this API without authentication. |
| HTTP Method: | GET |

| | |
|----------------------------------|--|
| Content Type: | — |
| Input/Output Format: | XML |
| HTTP Request: | — |
| Request Parameters: | — |
| HTTP Response: | 200: Success 403: Forbidden |
| Example Response | <pre><UserAuthMode> <authMode>SSO</authMode> </UserAuthMode></pre> |
| Example Failure Response: | <pre><ApiErrors> <ApiError> <ErrorType>Forbidden</ErrorType> <ErrorMessage>UserAuthModeService is disabled</ErrorMessage> </ApiError> </ApiErrors></pre> |

Single Sign-On—Fetch Access Token

This API gets the access token and refresh token from the Finesse server.



Note Invoking this API might involve browser redirect to Cisco Identity Service and Identity Provider.

| | |
|------------------------------|---|
| URI: | https://<FQDN>/desktop/sso/token |
| Example URI: | https://finesse1.xyz.com/desktop/sso/token |
| Security Constraints: | Agents and supervisors can use this API. |
| HTTP Method: | GET |
| Content Type: | — |
| Input/Output Format: | JSON |
| HTTP Request: | — |
| Request Parameters: | (Optional) return_user=yes no (Optional) return_refresh_token=true false |
| HTTP Response: | 200: Success 400: Bad Request 401: Unauthorized 500: Internal Server Error |

Example Response:**Response without any parameter:**

```
{
  "token": "eyJhbGciOiJIaXkiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMTI4Q0JDLUhTMjU2In0..LDXjaqAsM89uhdcQt364LA.qXBMK_y58Hkz19k-B8ealJ9LOalB0yNnm9vOvKExf8slCpXAPPlJLnNXGD9_-YTGdjs7lPtEcdI-hSuDmwxxOhdGZc7ekbAadJ6EItZhOGykCYk_CBFmEHKU8-pHV3bdbSUGrCTponA8BMw04-S-N5iuI3vu8fuihcNAeRY_9tjl5jv1hHEnd6zrYLDfH8KcO-V2f9bcFdxHn3BrZk9tMasrsAJNhm8Uo_kg06PXq9omrTbUEKm3f1_lMb3bwqZGXf006WLOngsADRTuHren_CTp5gR8r94LpsbXV7gRaEqsCu9kWo3pfxQsu88LNPRW6RPcjzozupw0A4-jrHBOF_X2XadquanEbBkZIt9VIJhjr6p8bTO5z1H9Z_x7vdMIfeT2pcjqcXKP3NiHlXOaB-tniPX_zN8ckGqIKR7L4wBxYmXUj82cnjBNMkcUsbvP9WmB7ihJw0waz11Tq6WnhtTGeOf0cnorjPm8DOZrcAAjJcSDCpudfj5CgE-OwikeSdWURgYTg_k6Kcct71I3olVLTc6nFRGcYvclvjCfTc1_ooBQ6ZKI_thq0Apnof23516drDxgsDMPiyop69hWcuMoRRK-KKAXr8xK3fiqKjsSse-KMLMGmLZkUusr2Y_Q0YwiEIJk1FJ4n5Qgn-ismhKi-A_Vg3ZicGJ-YyIcYgcs1JGDeqSB10Y0uThqOuMA9eGEHKS1ZGLcZBfX5MGv23dEOoxN9_wLkqazF75m5H_23ycLyN0v9d8uF7_fe7IWB97cI9nDAhaNBdHBR3XYU5GPsBRRS7GknDoWZM_8eTgzc-gFTfyfAJveg_pPr1sSKvWnabqLXUuLDmVcbgA-5UI2Y4HEGkzW85fNOHE9WPpo3cQdxFdrQyHfvFCBdTAOiFcIz_up2nCDB_8oPT7qycm6b58BRJ5EzaTcWapskB73w8nolYJadliQ200YHrDKSs_LJYDeB2iBROSUoVocYlW6GwTv0Ko7NsLv3OtGc_I.Fre8fhy_Y4u11tIfNo6fIA","expires_in":300}
```

Response with `return_user=yes` parameter:

```
{
  "token": "eyJhbGciOiJIaXkiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMTI4Q0JDLUhTMjU2In0..LDXjaqAsM89uhdcQt364LA.qXBMK_y58Hkz19k-B8ealJ9LOalB0yNnm9vOvKExf8slCpXAPPlJLnNXGD9_-YTGdjs7lPtEcdI-hSuDmwxxOhdGZc7ekbAadJ6EItZhOGykCYk_CBFmEHKU8-pHV3bdbSUGrCTponA8BMw04-S-N5iuI3vu8fuihcNAeRY_9tjl5jv1hHEnd6zrYLDfH8KcO-V2f9bcFdxHn3BrZk9tMasrsAJNhm8Uo_kg06PXq9omrTbUEKm3f1_lMb3bwqZGXf006WLOngsADRTuHren_CTp5gR8r94LpsbXV7gRaEqsCu9kWo3pfxQsu88LNPRW6RPcjzozupw0A4-jrHBOF_X2XadquanEbBkZIt9VIJhjr6p8bTO5z1H9Z_x7vdMIfeT2pcjqcXKP3NiHlXOaB-tniPX_zN8ckGqIKR7L4wBxYmXUj82cnjBNMkcUsbvP9WmB7ihJw0waz11Tq6WnhtTGeOf0cnorjPm8DOZrcAAjJcSDCpudfj5CgE-OwikeSdWURgYTg_k6Kcct71I3olVLTc6nFRGcYvclvjCfTc1_ooBQ6ZKI_thq0Apnof23516drDxGsDMPiyop69hWcuMoRRK-KKAXr8xK3fiqKjsSse-KMLMGmLZkUusr2Y_Q0YwiEIJk1FJ4n5Qgn-ismhKi-A_Vg3ZicGJ-YyIcYgcs1JGDeqSB10Y0uThqOuMA9eGEHKS1ZGLcZBfX5MGv23dEOoxN9_wLkqazF75m5H_23ycLyN0v9d8uF7_fe7IWB97cI9nDAhaNBdHBR3XYU5GPsBRRS7GknDoWZM_8eTgzc-gFTfyfAJveg_pPr1sSKvWnabqLXUuLDmVcbgA-5UI2Y4HEGkzW85fNOHE9WPpo3cQdxFdrQyHfvFCBdTAOiFcIz_up2nCDB_8oPT7qycm6b58BRJ5EzaTcWapskB73w8nolYJadliQ200YHrDKSs_LJYDeB2iBROSUoVocYlW6GwTv0Ko7NsLv3OtGc_I.Fre8fhy_Y4u11tIfNo6fIA","expires_in":49,
  "user_id": "1001001",
  "realm": "finesse.com",
  "user_principal": "1001001@finesse.com"}
```


| | |
|---|---|
| | <p>Response with return_refresh_token=true parameter:</p> <pre>{ "token": "eyJhbGciOiJkaXIiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMtI4Q0JDLUhTMjU2In0..LDXjaqAsM89uhdcQt364LA.qXBMK_y58Hkz19k-B8ealJ9LoalB0yNnm9vOvKExf8slCpXAPPlJLnNXGD9_-YTgdjs7lPtEcdI-hSuDmwxXOhdGZc7ekbAadJ6EITZhOGykCYk_CBFmEHKU8-pHV3bdbSUGrCTponA8BMw04-S-N5iuI3vu8fuihcNAeRY_9tjl5jv1hHEnd6zrYLDfH8KcO-V2f9bcFdxHn3BrZk9tMasrsAJNhm8Uo_kg06PXq9omrTbUEKm3f1_lMb3bwqZGXf006WLOngsADRTuHren_CTp5gR8r94LpsbXV7gRaEqsCu9kWo3pfxQsu88LNPRW6Rcjozupw0A4-jrHBOF_X2XaDquanEbBkZit9VlJhjr6p8bTO5z1H9Z_x7vdMIfeT2pcjqcXKP3NiHlXOaB-tniPX_zN8ckGqIKR7L4wBxYmXUj82cnjBNMkcUsbvP9WMb7ihJw0waz11Tq6WnhtTGeOf0cnorjPm8DOZrcAAjJcSDCpudfj5CqE-OwikesDWURgYTg_k6Kcct71I3o1VLTc6nFRGcYvclvjCfTc1_ooBQ6ZKI_thq0Apnof23516drDxGsDMPiyop69hWCuMORRK-KKAXr8xK3fiqKjSse-KMLMGrMLZkUsr2Y_Q0YwiEIJk1FJ4n5Qgn-ismhKi-A_Vg3ZicGJ-YyIcYgcs1JGDeqSB10Y0uThqOuMA9eGEHKS1ZGLcZBfX5MGv23dEOoxN9_wLkqazF75m5H_23ycLyN0v9d8uF7_fe7IWB97cI9nDAhANdBHR3XYU5GPsBRRS7GknDoWZM_8eTgzc-gFTfYfAJveg_pPr1sSKvWnabqLXUuLdMvcVbqA-5UI2Y4HEGKzW85fNOHE9WPpo3cQdxFdRQyHfvFCBdTAoiFcIz_uP2nCDB_8oPT7qycm6b58BRJ5EzaTcWapsk73w8no1YJadliQ200YHrDKSs_LJYDeB2iBROSuVocYlW6GwTv0Ko7NsLv3OtGc_I.Fre8fhy_Y4ulltIfNo6fIA","refresh_token": "eyJhbGciOiJkaXIiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMtI4Q0JDLUhTMjU2In0..gdULcCw-3nh_R-FnIHPXDQ.0kYaOssJH76ro2iD5JCYBuZqVMFrdpA0FDeK_bm9ClcuJaaU4vNwnw7r6G7qT4AFdQZqnIMq0kBSzpbfKVIchOs95usysQA3IYi2d8dlEblIkdRiPWyWhNbhj_KjcNM-Rjt9SbinW3df18NL_OBOMTs0HYa8DwdcXnP_62QonodfHAIgVH9i8VgoHKBjCLUTNcUbVX43pdxYOVlWuOOZh4Uzp-8L1dn16PE0wU7wBvyl2J6xTrAEZ1Beya2pCP-zn8eCYTqAEiuzbm-DVP9b4G7w4qusQ_Z347b00a782o7_Ui4_mpb2kaOP5wYByR0ftCkGLsAnPyJ5BpB89Sd2TnXmRWMhc72vqu1AM67UimxUpjWxVDZem0QowbYKldEWZSMUP744hLgrGFEDanWQKFKoOMzqhprfHCz3VZYs6kZvhvWRsmoU8hI9j0qklfNuhJkCs-UeF6GnlN7FNsxEFHAvCxfG_ofGXQcEjoeK0cYPzUWhIqV94bapeYm6sRH7d38QGBRm2D-mkdsQyMcQvH66NJ3uYpmlInwNoNQuaS0Feo40Tpms5CI49jblNr17k5Kpk6Q7DMKcCuyKK4QyxPY9uN-7y213XnmtG3jhqOTjQ9t_1YdBh-RFHbv6M2KWWIHZ57CpRpSNCo-wN1VRaOkbQaLwLwEU_Hk-Cp4.RRHsyddqhFEtH0xvRc5EjA","expires_in":49}</pre> |
| <p>Example Failure Response:</p> | <pre>{"error": "invalid_redirectUri", "error_description": "Invalid Redirect URI."}</pre> |



Note When you use the **return_refresh_token=true** query parameter in this API, access token and refresh token cookies are not added to the response. All information is provided as part of the response body, which can be directly used by the third-party clients.

Use this query parameter when third-party clients use Cisco Finesse SSO APIs alongside Finesse desktop in the same browser. Using this query parameter prevents agent logout from Finesse desktop due to desktop cookie overriding by third-party clients.

Single Sign-On—Refresh Existing Access Token

This API allows a user to refresh an existing access token that is about to expire.


Note

- Third-party applications have to refresh the access token after 75% of the token expiry time is elapsed.
- Invoking this API might involve browser redirect to Cisco Identity Service and Identity Provider.

| | |
|------------------------------|---|
| URI: | https://<FQDN>/desktop/sso/token |
| Example URI: | https://finesse1.xyz.com/desktop/sso/token |
| Security Constraints: | Agents and supervisors can use this API. |
| HTTP Method: | POST |
| Content Type: | application/x-www-form-urlencoded |
| Input/Output Format: | JSON |
| HTTP Request: | token=<token value> |
| Request Parameters: | (Optional) return_user=yes |
| HTTP Response: | 200: Success 400: Bad Request 401: Unauthorized 500: Internal Server Error |

| | |
|----------------------------------|---|
| Example Response | <pre>{ "token": "eyJhbGciOiJIaXkiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMjI4Q0JDLUhmTmJlU2In0..521UM8q8d7wM5naKgWzPhA.NkHEH7SatpXPOVqQobJstaz51HBCMTcIej5qdIJ0ZwjCnV7u8iKGcv7t5cLYruV6WZfJn8z7iSckXdduDqmRserhBDnbpk-gd5jqNj9r2ZStfeBZIx6Phng6EMWUjtK9cbrO79MenQ7u7Y3Hhe7P7qvQiaTweUw7No09NFGat-ICzhHbTF8D4WKFhFefw1J-q55ktcdD-CmMs-KXYrma8DLltjF9ii9dCYHFFc2nKBETzdYWR2ple4B6_Lv0npg8OSU53LyTT3ObHm6TvWZ09KYrWUWMKNFas73Gx7rYro4C7Tc4pYb9ZfJmkt6coRIocMteYCrqCy7ufRqO-BPObNIah_Jo2VQ_ww0-5wE-cMUUDpGa5X2nMtP2YUH4sb7b_SHX9Xq_w6cwLRcBiDXjyG17Smk1RzFlaxj2A9R06a71VjzmUsjq4UtrT7_Iyfs9RrFX9jhnXX1VB8Dqgh-Pnb16rsskRg7TPP4EV9fwdSbha-omrMKqFz5BFWMhNaFCHtJQWtXxNRK802ybyzXwR3KGeINS D3dOGj2vWRpnhuTB9veHr9InSrc2s67rspguN7YX2bkIEEQNBCY3X5rf_UMyGSlPvlArh6b-_yZXk62kXmYJWJ7gluTRwTaou87Cj83fqdaIOYMNIOeZhZqDmKDOZqMmVW_Aj-9-Tn01TXkKmsPvqtoJYCN1T_3fZrvhzJLImy0whXgEtxc88MYNOCsupSkIuCRNpoOGgWXATdF1GHPUnQPstW2GsZEfbdY5R1X9x3SZXtngH4XFMgYtMjP129X8pvAT_AY35JtRzpdryRPdAYrEc72tkY_xWLBahpSAKrcX7x8gtMRZmV5HlKs7_sWlamje0gaMKFlqh8i56XWbwnsUSdKLC-LZDtvWZ5wYuHPY1CSwC0oT91HytWBXo3GSXSV1iqy75ud6KrvrJg3WG2k_2biqxpC0S9MsATT2WGtGBt5ko2wEcn6A.l_JfM6gAelSswEeGFAOKwg", "expires_in": 300 }</pre> |
| Example Failure Response: | <pre>{ "errorType": "AUTH_ERROR", "errorData": "refresh-token", "errorMessage": "Invalid Token" }</pre> |



Note If the token was initially fetched with the `return_refresh_token=true` query parameter, then the refresh token in request payload is mandatory.

Client Integration

Clients can use the Finesse REST APIs in SSO mode. For thick client integrations, the following are browser like behaviors that thick clients must ensure to exhibit:

- Follow server issued redirects.
- Store and forward cookies.
- Honor the various cookie attributes.
- Execute JavaScript in HTML responses.

Procedure

Step 1 Use the API to get the system's authentication mode. The authentication mode can be found in the response as the value of the `systemAuthMode`.

Note If the system's authentication mode is SSO, then you can skip step 2.

Step 2 Use the **Single Sign-On—Get User Authentication Mode** to get a specific user's authentication mode.

Note You must use browser components that allow redirections and IdP form submissions. You cannot use Postman or AJAX for the Single Sign-On APIs.

Step 3 Use the **Single Sign-On—Get User Authentication Mode** with the `return_user` query parameter set to **yes** to get the user's access token.

The username must be provided in a cookie or a URL query parameter with a key of `cc_username`. The value is a URL encoded username, which can be the `loginName` or `peripheralId` for whom the token is requested.

Example:

```
https://finesse1.xyz.com/desktop/sso/token?return_refresh_token=true
```

The result of the API request will redirect the request to the IdS page which then redirects to the IdP page.

Step 4 On the IdP page, enter the **username** and the **password**.

- On successful authentication, the response body contains the access token and `user_id` value. The access token returns the response in a JSON Web Encryption format (performs intermediate redirects at this point).

Example Response:

```
{ "token": "eyJhbGciOiJIaXkiLCJpdHkiOiJKV1QiLCJlbmMiOiJBMTI4Q0JDLUhtMTU2In0..0
KelINtPSTQJGHthEo6GGg.iJl7anAh8Edt07pOQmHZ7BLgK-ozMiy5Fy42Pkj8FQ3xUMQvq5coGwSnHCEr1
deuNft5i6685L5aHzzGe3VChWRPOnHveKaOebgjdNahFXnIF033H4hZ-sKf zGSdIiXDIqv71lOGhzwDWw
jYA7icEgIqaTttu5VVMpSI5eCwwd8uf1SXA7_rNU9bM1Kkra-v PRyqJX4h2gR-1vbk1q3L1GETaAnWqDIE
4MAZELtuD-J5O4Kid26lwkzdq5PL_PH51-lyw_Mz ds0JiE4ZWcG_JmAa4BtZfKGS1z4S6Laj7scxi7WW7u
5-lfBsn6ixhouC3Jdz4N3FJQZ6IizQ 0tUwOWb0HoD6tsU6mH4r5tYeRjtG1tJff71BIakZl6N1fCB0ZN2
P51PMPWqfIqgw8r0H5Ar 1xDNpmfs_b0e01HwfCczWKqn96wFIUeP0IBpk2uv7-8C5NvM6U72Vbs9SjUH7T
3b8Zzt59mg Hnu2fdluW5OfrdLxF6BgiI20_p6jjI0D7HPrqpX-I7RGSWB79fEfM0IOFAEY04kwwrJBjX8hI
Mu-2Ahc38NW7i-mZVzbE28K2pPwyeFMR6UWKI10ztDsAyEQa89DI1bOukFa57CDQzF4mR5szmczaLeoXTAC
hy8qPmVAEjMtAvSIQSwb9W-D6HoxBQM6gF-SthlEd2n82gkbb1-Gg4GF3gz0Sa-_kf1sYj62mF4PUsJVN4_
cFXRgzdyhjkDielwabBJPbt0oAXR2qj3qH7TBxLj4hYdKbPq2bDd9eFNeMhTMX49hYrONGWaSz3CdBb3fby
177ALj-AHlGU2mbZ10fa3hUj5h1H3p7QwGwGs1Ka56DHK3cTcPszAvMdtVF0MzSk90Dr0gJVFFKpvoY2alb
d3xG6rbbQmvWGoSYWgT7l2dzUYokEOEjN6halaZmObcnjxS-NeqCRrve-22zwqFwD-fEdjRf9ATtq32UcB_
RmlnubdJondJQ.5kix_gQZGr6xGye3cUE18g", "refresh_token": "eyJhbGciOiJIaXkiLCJpdHkiOiJK
V1QiLCJlbmMiOiJBMTI4Q0JDLUhtMTU2In0..gdULcCw-3nh_R-FnIHPXDQ.0kYaOssJH76ro2iD5JCYBuZ
qVMFRDpA0FDeK_bm9ClcuJaaU4vNwnw7r6G7qT4AFdQZqnIMq0kBSzpbFKVichOs95usysQA3IYi2d8d1E
blIkdrIPWYWhNbhj_KjcnM-Rjt9SbinW3df18NL_OBOMTs0HYa8DwdcXnP_62QonodfHAiGVH9i8VgoHKBj
CLUTNcUvVX43pdxYovLwuOOZh4Uzp-8Lldn16PE0wU7wBvyl2J6xTrAEZ1Beya2pCP-zn8eCYTqAEiuzbm-
DVP9b4G7w4qusQ_Z347b00a782o7_Ui4_mpbT2kaOP5wYByR0ftCkGLsAnPyJ5BpB89Sd2TnXmRWMhc72vq
ulAM67UimxUpjWxVDZem0QowbYKldEWZSMUP744hLgrGFekdanWQKFKoOmZqhpFHCz3VZYs6kZvhvWRsmo
U8hI9j0qklfNuhJkCs-UeF6GnlN7FNsxEfHAvCXgF_oxGFQcEjoeK0cYPzUWhIqV94bapeYm6sRH7d38QGB
Rm2D-mkdsQyMcQvH66NJ3uYPmlInwNoNQuaS0feo40Tpms5CIt49jblNrl7k5Kpk6Q7DMKcCuyKK40yxPY9
uN-7y2l3XnmtG3jhgOTjQ9t_1YdBh-RFHbv6M2KWWIHZ57CpRpsNCo-wN1VRaOkbQaLwLwEU_Hk-Cp4.RRH
syddqFEtH0xvRc5EjA", "expires_in": 3564, "user_id": "sjefferson", "realm": "finesse.com",
"user_principal": "sjefferson@finesse.com" }
```

- The response also contains the token expiry time in seconds.

Note In a Unified CCX deployment, the username, `loginName`, and `loginId` can be used interchangeably for the Finesse REST API calls.

Step 5 This step is for Unified CCE deployments only. In a Unified CCE deployment, the `user_id` obtained from the IdS token can be either the `loginName` or the `peripheralId` (`loginId`). The Finesse REST APIs can only accept the `loginId`. Use the **User—Get User Id from loginName** API to get the `loginId` from the `user_id` of the IdS token.

Example:

```
https://finesse1.xyz.com/finesse/api/User/sjefferson
```

Example Response:

```
<User>
  <dialogs>/finesse/api/User/1001002/Dialogs</dialogs>
  <extension></extension>
  <firstName>AGENT</firstName>
  <lastName>98411</lastName>
  <loginId>98411</loginId>
  <loginName>sjefferson</loginName>
  <mediaType>1</mediaType>
  <pendingState></pendingState>
  <reasonCodeId>-1</reasonCodeId>
  <roles>
    <role>Agent</role>
  </roles>
  <settings>
    <wrapUpOnIncoming>OPTIONAL</wrapUpOnIncoming>
  </settings>
  <state>LOGOUT</state>
  <stateChangeTime></stateChangeTime>
  <teamId>5000</teamId>
  <teamName>FunctionalAgents</teamName>
  <uri>/finesse/api/User/sjefferson</uri>
  <wrapUpTimer>30</wrapUpTimer>
</User>
```

Step 6 Get the loginId from the **loginId** field.

All subsequent Finesse REST API requests must use the loginId from the <User> response, instead of the username/user_id/loginName.

Example:

To login the agent *sjefferson* using the Finesse REST API, you must use the loginId of 98411.

```
https://finesse1.xyz.com/finesse/api/User/98411
```

```
<User>
  <state>LOGIN</state>
  <extension>98411</extension>
</User>
```

Step 7 To avoid the authentication and authorization flow again, the access token must be refreshed before the expiry time. Use the [Single Sign-On—Refresh Existing Access Token, on page 6](#) with the `return_user` query parameter set to **yes** to refresh the user's access token.

The username must be provided in a cookie or a URL query parameter with a key of `cc_username`. The value is a URL encoded username, which can be the loginName or peripheralId for whom the token is requested.

Example Response:

```
{"token": "eyJhbGciOiJIaXkiLCJpdHkiOiJpYXkiLCJlbnMiOiJBMtI4Q0JDLUhmTmJlU2In0..0
KelINTpSTQJGHthEo6GGg.iJl7anAh8Edt07pOQmHZ7BLgK-ozMiy5Fy42Pkj8FQ3xUMQVvq5coGwSnHCEr1
deuNft5i6685L5aHZzGe3VChWRPOnHveKaOEbgjgdNAhFXnIF033H4hZ-sKf zGSdIiXDIqv711OGhzdWw
jYA7icEgIqaTttu5VVMpsI5eCwwd8uf1SXA7_rNU9bM1Kkra-v PRyqJX4h2gR-1vbk1q3LlGEtaAnWqDIE
4MAZELtuD-J5O4Kid26lwKzdq5PL_PH51-lyw_Mz ds0JiE4ZWcG_JmAa4BtZfKGs1z4S6Laj7scxi7WW7u
5-1fBsn6ixhouC3Jdz4N3FJQZ6IizQ 0tUwOwb0HoD6tsU6mH4r5tYeRjtG1iTJff71BIAkZl6N1fCB0ZN2
P5lPMWpQfIqqw8r0H5Ar 1xDNpmfs_b0e01HwfCczWKqn96wFIUeP0IBpk2uv7-8C5NvM6U72Vbs9SjUH7T 3b8zZzt9mg
Hnu2fdluW50frdLx6BGiI20_p6jjI0D7HPrqpX-I7RGSWB79fEFm0IOFAEy04kwvRJBjX8hI
```

```
Mu-2AHc38NW7i-mZVzbe28K2pPwyeFMR6UWKI10ztDsAyEQA89DI1bOukFa57CDQzF4mR5s zmczaLeoXTAC
hY8qPMvAEjMtAvSIQSwb9W-D6HoxBQM6gF-Sth1eD2n82gkbb1-Gg4GF3gzo Sa-_kf1sYj62mF4PUsJVN4_
cFXRgzdyhjkDielwabBJPbt0oAXR2qj3qH7TBxLj4hYdKbPq2b Dd9eFNeMhTMX49hYrONGWasz3CdBb3fby1
77ALj-AHlGU2mbZ1Ofa3hUj5h1H3p7QwGWGs1Ka 56DHK3cTcPszAvMdtVF0MZsK9ODr0gJVFFKPvoY2albd3
xG6rbbQmvWGoSYWgT7l2dzUYokEOE jN6halaZmOBcnjxS-NeqCRve-22zwqFwD-fEdjRf9Attq32UcB_Rml
nubDJONDJQ.5kix_gQZ Gr6xGye3cUE18g", "expires_in":3564, "user_id": "sjefferson",
"realm": "finesse.com", "user_principal": "sjefferson@finesse.com"}
```
