



# Red Hat Subscription Management 1

## Using APIs in Red Hat Subscription Management

authorizing, managing, and troubleshooting APIs in Red Hat Subscription  
Management



# Red Hat Subscription Management 1 Using APIs in Red Hat Subscription Management

---

authorizing, managing, and troubleshooting APIs in Red Hat Subscription Management

## Legal Notice

Copyright © 2022 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## Abstract

Red Hat Subscription Management offers developed and documented APIs to help you better automate, manage, and track your subscriptions to Red Hat products.

---

## Table of Contents

<b>1. USING APIS IN RED HAT SUBSCRIPTION MANAGEMENT</b> .....	<b>2</b>
<b>2. USING TOKENS FOR AUTHENTICATION</b> .....	<b>2</b>
2.1. Generating a new offline token	2
2.2. Generating a new refresh token	3
<b>3. ACCESSING AVAILABLE RED HAT SUBSCRIPTION MANAGEMENT APIS</b> .....	<b>4</b>
<b>4. TROUBLESHOOTING API ERRORS</b> .....	<b>4</b>
4.1. Troubleshooting error 403	5
4.2. Troubleshooting error 429	5
<b>CHAPTER 1. APPENDIX A. REVISION HISTORY</b> .....	<b>6</b>

## 1. USING APIS IN RED HAT SUBSCRIPTION MANAGEMENT

Using APIs in Red Hat Subscription Management can help you more effectively keep track of and automate how you manage your Red Hat subscriptions and entitlement usage. By using APIs in Red Hat Subscription Management, you can:

- Control which tooling you use for which products
- Better manage your system inventory
- Update and secure your systems more efficiently
- Continue receiving official support for your Red Hat products

Red Hat Subscription Management APIs use OAuth 2.0 for authorization. To obtain a token and access the APIs, you will need the following pieces of information:

- Offline token generated on the Red Hat Subscription Management API Tokens page
- Client ID = rhsm-api
- Token URL = <https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token>

## 2. USING TOKENS FOR AUTHENTICATION

Offline and refresh tokens are used by Red Hat Subscription Management to authenticate your system after you set up your account using your secret to authenticate your Customer Portal account.



### WARNING

Please use password management that is consistent with networking best practices. It is never safe to store any passwords or credentials in plaintext. Treat your offline token with the same security measures that you would a password to protect it against unauthorized use.

### 2.1. Generating a new offline token

An offline token never expires as long as it is used at least once every 30 days and is used to create access tokens for the Red Hat Subscription Management APIs. It works as a password and allows you to continue being able to authenticate your account without having to create new refresh tokens.

#### Procedure

1. Visit the [Red Hat Subscription Management API Tokens page](#) .



```

3JIZGhhdc1leHRIcm5hbClsmF1ZCI6InJoc20tYXBpliwic3ViljoiZjo1MjhkNzZmZi1mNzA4LTQzZWQtO
GNkNS1mZTE2ZjRmZTBjZTY6cmhuLXN1cHBvcnQta3RvcmlldXliLCJ0eXAiOiJPZmZsaW5lIiwiaXpwIj
oicmhzbS1hcGkiLCJhdXRoX3RpbWUiOjAsInNlc3Npb25fc3RhdGUiOiJmMGRiYjhhkNC00ZTRILTQ2NT
QtODQ0Yy02ZjM3MDRjODQ0MjliLCJyZWZfV9hY2Nlc3MiOncicm9sZXMiOlsicG9ydGFsX21hbmFnZ
V9zdWJzY3JpcHRpb25zliwib2ZmbGluZV9hY2Nlc3MiLCJjYW5kbGVwaW5fc3lzdGVtX2FjY2Vzc192aW
V3X2VkaXRfYWxslwiYWRTaW46b3JnOmFsbClInBvcnRhbf9tYW5hZ2VfY2FzZXMiLCJwb3J0YWxfxc3
zdGVtX21hbmFnZW1lbnQiLCJwb3J0YWxfZG93bmxxvYWQlXX0sInJlc291cmNIX2FjY2Vzcyl6e319.S_p
mAWzQUc04f0uGHN9rRYd4sH1t4IPnEwCcOH1aBL9Qo4_EbXPWCrtnf84f1pfuKJTQwUS-
DldY6eloyVEsGgnqkygBKh270bu_bNXCNAuLJigEMsYx_2VzdnwWLptWS2_FUaNwe7Tai8qXwd8F0g
e0Zjoi3P15S_8z4Tp79uD-
qKcvwz6NIPKCOZwEbwZqOkJDZ8JKTIK8O0jfqdtHMfaWwlXMXdvx3B70tTOtHjQGAsxZA2dPPvqVGu
yMOMmC3bMaISReUbtDwsCV-
eAZplDfDZthr4k4JbmG9lwq1aATaF3aCwfpebcmolZGHE4_RLZrXCZKApXVVvRxcOrJytxlZrbDHq6oz
X7j-
j1SE3kuexcSLvlodmfTlxwPX9g7aqJu2ZLno54NxQSgYO8lQqSvScFgLtbX5f_FUS0lw6yRWWJy2o2fnvf
Gk83rt5UYTtlb8Xd1GXcpHf8Y110nVy21BetSQY__VpahF_eZghBNxS689GJnwUqAwlu01pOlB26mmH
aydHc3hqUsudZydRbaFf17nR6gQP8lCtp6b0z5hgVHLG4ZJ7i4MmEL6C5G4xHUaUs6RZgJUSsc2DzL
W0b7rSQj41JuvTmSgD8bMrnVokmkAbfvxjKGc7E8n2GylmO7JiKb3RA7_o0xOTRYDla_Ns-
InigJkUIQZUzt7JI","token_type":"bearer","not-before-policy":0,"session_state":"f0dbb8d4-4e4e-4654-
844c-6f3704c84422","scope":"offline_access"}%

```

The `access_token` is what needs to be set/used as an authorization token to perform the API call.

```

# token=`curl https://sso.redhat.com/auth/realms/redhat-external/protocol/openid-connect/token -d
grant_type=refresh_token -d client_id=rhsm-api -d refresh_token=$offline_token | jsonValue
access_token`

```

### 3. ACCESSING AVAILABLE RED HAT SUBSCRIPTION MANAGEMENT APIS

Red Hat provides a [Swagger](#) file to describe the specifications of the Red Hat Subscription Management APIs. The Swagger specification includes information about the API endpoints available, input parameters, expected output, and possible error responses. The swagger file can be imported into REST clients like Postman or RESTlet to automatically build a library of API calls.

### 4. TROUBLESHOOTING API ERRORS

Table 1. API errors

Code	Explanation	Resolution
400	BadRequest error	Validate that you entered the API call correctly and try again.
401	Unauthorized	Generate a new authorization token.
403	Forbidden	Generate a new authorization token.
404	Not found	Resource not found or does not exist.



Code	Explanation	Resolution
429	Too many requests	Reduce the frequency of requests
500	Internal server error	The problem is on Red Hat's end. Wait a minute and try your request again.

#### 4.1. Troubleshooting error 403

Error 403 is a "not authorized" error, meaning that the authentication you are using for Red Hat Subscription Management APIs has failed. There are two possible solutions you can try.

##### Procedure

1. To authenticate through the Red Hat Subscription Management gateway, ensure the authorization header includes the text "bearer" before entering your API call:

```
curl -H "Authorization: Bearer <token>" <api_url>
```

2. If the header is correct, create a new token. Refresh tokens last for five minutes.

#### 4.2. Troubleshooting error 429

Error 429 is a "rate limiting" error, meaning that your account has exceeded the number of allowed requests per second. This limit applies to all users of a single Red Hat account.

##### Procedure

Extract the header of the response, which includes: \* **X-RateLimit-Limit**: the total requests/sec allowed \* **X-RateLimit-Remaining**: the number of requests/sec remaining (this will be a negative integer) \* **X-RateLimit-Delay**: the number of seconds the requester should wait before trying again

Adjust the rate of requests to the X-RateLimit-Limit value and start again once the X-RateLimit-Delay time has passed.

## CHAPTER 1. APPENDIX A. REVISION HISTORY

Table 1.1. Revision History

Revision	Date	Changes Made	Author
Revision 1.1-0	Tues Sept 19 2019	Procedures for offline, refresh tokens have changed	Anni Bond
Revision 1.0-1	Wed May 8 2019	Added detail to prerequisites about where to get a secret	Anni Bond
Version 1.0-0	Fri May 3 2019	Initial creation	Anni Bond