Red Hat Customer Portal 1

Customer Portal Integration Guide

Guidance for integrating your application with the Red Hat Customer Portal

Last Updated: 2023-07-05

Red Hat Customer Portal 1 Customer Portal Integration Guide

Guidance for integrating your application with the Red Hat Customer Portal

Legal Notice

Copyright © 2013 Red Hat, Inc..

This document is licensed by Red Hat under the Creative Commons Attribution-ShareAlike 3.0 Unported License. If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

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 ® is the registered trademark of Linus Torvalds in the United States and other countries.

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

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

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

Node.js ® 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 ® 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

This document provides information about using APIs exposed by the Red Hat Customer Portal in order to query and update customer cases.

Table of Contents

1. INTRODUCTION 2

2. CLIENTS 2

3. COMMON USAGE 2

4. AUTHENTICATION 2

5. EXAMPLES 3

A. REVISION HISTORY 5

1. INTRODUCTION

It is now possible to integrate with the Customer Portal at http://access.redhat.com using APIs [1]. This document outlines common use cases including examples in common clients and frameworks. The official API documentation can be found on the Customer Portal at https://developers.redhat.com/api-catalog/api/case-management and includes both API endpoints and payload information.

2. CLIENTS

The Customer Portal API is, by nature, client-agnostic. It is expressed as a set of resource URLs which send and receive XML data. Some common clients and platforms include, but are not limited to, the following:

Client	Platform/Environment	Comments
cURL [a]	Command Line (Linux, UNIX, Mac OS X, Microsoft Windows)	Easy and transparent way to test commands and do simple integration
Apache HTTP Client [b]	Java	Most common Java library with which to talk HTTP; offers little semantic value beyond simple HTTP; no intrinsic binding of XML
RESTeasy Client [c]	Java	Full Java model integration; no need to think about HTTP or XML; uses Apache HTTP Client underneath

[a] http://curl.haxx.se/

[b] http://hc.apache.org/httpclient-3.x/

[c] http://docs.jboss.org/resteasy/docs/2.0.0.GA/userguide/html/RESTEasy_Client_Framework.html

3. COMMON USAGE

For clarity, this document includes cURL examples for all use cases and includes examples using other frameworks as a courtesy from Red Hat. cURL most clearly illustrates the nature of interacting with RESTful resources. For more information, see the man page for cURL by using the command **man curl**.

4. AUTHENTICATION

You must authenticate each request for the Customer Portal. To authenticate a request, generate an authentication token based on your Customer Portal username and password, then declare that token in the **Authorization** header of each subsequent request.

For more information about requesting an authentication token, see Getting started with Red Hat APIs.

5. EXAMPLES

The following examples demonstrate how to pass the authentication token through a bearer header using cURL.

Example 1. List Cases

Return 10 cases from the logged-in user's account

```
$ curl -X POST -H ‘Content-Type: application/json’ -H ‘Authorization: Bearer $token’ --data
‘{
 "offset" : 1,
 "maxResults": 10
}’
https://api.access.redhat.com/support/v1/cases/filter
```

Example 2. Filter by last update date

```
$ curl -X POST -H ‘Content-Type: application/json’ -H ‘Authorization: Bearer $token’ --data
‘{
“startDate” : “2021-01-01”,
“endDate”: “2021-12-31”
}’
https://api.access.redhat.com/support/v1/cases/filter
```

Example 3. List Case Comments

This example lists case comments for case 0000000.

Replace **0000000** with the number of a case to which you have access.

```
$ curl -H ‘Authorization: Bearer $token’
https://api.access.redhat.com/support/v1/cases/0000000/comments
```

Example 4. Create a Case

The POST operation is used in this example, which creates a case under the **RHEL 6** product using the default group, severity level, and type.

```
$ curl -X POST -H 'Content-Type: application/json -H ‘Authorization: Bearer $token’ --data
'{
"product":"Red Hat Enterprise Linux",
"version":"7.0",
"summary": "Example Case",
"description": "Example Case created with cURL”
}'
https://api.access.redhat.com/support/v1/cases
```

Example 5. Update a Case

The PUT method is used here to update data on an existing case. Most fields can be updated in this way.

Case 000000's product is changed to Red Hat Enterprise Linux.

```
$ curl -X PUT -H 'Content-Type: application/json -H ‘Authorization: Bearer $token’ --data
'{
"product": “Red Hat Enterprise Linux”,
"version":"7.0"
}'
https://api.access.redhat.com/support/v1/cases/0000000
```

Example 6. Escalate a Case for Management Attention

This will escalate a case for management attention:

```
$ curl -X PUT -H 'Content-Type: application/json -H ‘Authorization: Bearer $token’ --data
'{
"requestManagementEscalation": true
}'
https://api.access.redhat.com/support/v1/cases/0000000
```

Example 7. Add a New Case Comment

This example uses the POST method to add a new comment to case 0000000.

```
$ curl -X POST -H 'Content-Type: application/json -H ‘Authorization: Bearer $token’ --data
'{
“commentBody”: “Test comment!  This can contain lots of information, etc.”
}'
https://api.access.redhat.com/support/v1/cases/0000000/comments
```

Example 8. Add a File Attachment to a Case

File attachments are unique because they carry no XML payload. This example uses a form-encoded POST to transmit a file named **test.txt** to case 0000000. API supports file upload till 1GB.

```
$ curl -X POST -F 'file=@test.txt' -H ‘Authorization: Bearer $token’
https://api.access.redhat.com/support/v1/cases/0000000/attachments
```

A. REVISION HISTORY

Revision 2.0-2	**2022-03-23**	**Navinya Shende**
Updates examples with Hydra API		
Revision 2.0-1.400	**2013-12-18**	**Rüdiger Landmann**
Rebuild with publican 4.0.0		
Revision 2.0-1	**Fri Nov 1 2013**	**Zac Dover**
Publish for new site		
Revision 0.0-0	**Tue Sep 24 2013**	**Misty Stanley-Jones**
Converted existing document to Docbook		

[1] http://en.wikipedia.org/wiki/Representational_State_Transfer