should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle Master Agreement, Oracle License and Services Agreement, Oracle PartnerNetwork Agreement, Oracle distribution agreement, or other license agreement which has been executed by you and Oracle and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Sample Code

Oracle may provide sample code in SuiteAnswers, the Help Center, User Guides, or elsewhere through help links. All such sample code is provided “as is” and “as available”, for use only with an authorized NetSuite Service account, and is made available as a SuiteCloud Technology subject to the SuiteCloud Terms of Service at www.netsuite.com/tos.

Oracle may modify or remove sample code at any time without notice.

No Excessive Use of the Service

As the Service is a multi-tenant service offering on shared databases, Customer may not use the Service in excess of limits or thresholds that Oracle considers commercially reasonable for the Service. If Oracle reasonably concludes that a Customer's use is excessive and/or will cause immediate or ongoing performance issues for one or more of Oracle's other customers, Oracle may slow down or throttle Customer's excess use until such time that Customer's use stays within reasonable limits. If Customer's particular usage pattern requires a higher limit or threshold, then the Customer should procure a subscription to the Service that accommodates a higher limit and/or threshold that more effectively aligns with the Customer's actual usage pattern.

Beta Features

Oracle may make available to Customer certain features that are labeled “beta” that are not yet generally available. To use such features, Customer acknowledges and agrees that such beta features are subject to the terms and conditions accepted by Customer upon activation of the feature, or in the absence of such terms, subject to the limitations for the feature described in the User Guide and as follows: The beta feature is a prototype or beta version only and is not error or bug free and Customer agrees that it will use the beta feature carefully and will not use it in any way which might result in any loss, corruption or unauthorized access of or to its or any third party's property or information. Customer must promptly report to Oracle any defects, errors or other problems in beta features to support@netsuite.com or other designated contact for the specific beta feature. Oracle cannot guarantee the continued availability of such beta features and may substantially modify or cease providing such beta features without entitling Customer to any refund, credit, or other compensation. Oracle makes no representations or warranties regarding functionality or use of beta features and Oracle shall have no liability for any lost data, incomplete data, re-run time, inaccurate input, work delay, lost profits or adverse effect on the performance of the Service resulting from the use of beta features. Oracle's standard service levels, warranties and related commitments regarding the Service shall not apply to beta features and they may not be fully supported by Oracle's customer support. These limitations and exclusions shall apply until the date that Oracle at its sole option makes a beta feature generally available to its customers and partners as part of the Service without a “beta” label.
Send Us Your Feedback

We'd like to hear your feedback on this document.

Answering the following questions will help us improve our help content:

- Did you find the information you needed? If not, what was missing?
- Did you find any errors?
- Is the information clear?
- Are the examples correct?
- Do you need more examples?
- What did you like most about this document?

Click here to send us your comments. If possible, please provide a page number or section title to identify the content you're describing.

To report software issues, contact NetSuite Customer Support.
Table of Contents

REST Services Overview ........................................................................................................... 1
The REST Roles Service ......................................................................................................... 2
  Authentication for the REST roles Service ......................................................................... 2
  URLs for Accessing the REST roles Service ....................................................................... 3
  Domain Data Returned by the roles Service ........................................................................ 4
  Sample Responses from the roles Service ......................................................................... 4
DataCenterUrls REST Service ................................................................................................. 7
Issue Token and Revoke Token REST Services for Token-based Authentication .................... 9
REST Services Overview

See the following sections for information about a comparison of RESTlets and other integration options, and about the available REST services:

- RESTlets vs. Other NetSuite Integration Options
- The REST Roles Service
- DataCenterUrls REST Service
- Issue Token and Revoke Token REST Services for Token-based Authentication

For information about REST web services, see the help topic SuiteTalk REST Web Services API Guide.
The REST Roles Service

NetSuite provides a REST service called roles, which you can use to retrieve the following information:

- A list of roles that a user belongs to.
- The correct domains for external client access to a NetSuite account. In an integration, domains must be discovered dynamically, because they can change without notice.
- The account type of the returned account (for example, production, release preview, or sandbox).

**Note:** RESTlets are part of SuiteScript. They are not part of NetSuite's web services feature. Be aware that if a role has the Web Services Only option set to true, a user logged in through that role is permitted to send web services calls only. RESTlet calls receive an INVALID_LOGIN_CREDENTIALS error response.

You call the roles service by sending a request to one of the roles service URLs, as described in URLs for Accessing the REST roles Service. The request must use the get method, and it must also include an NLAuth authorization header. For details, see Authentication for the REST roles Service. If you need to retrieve Customer Center roles, note that you must include a NetSuite account ID in the authorization header.

In response, the service returns data about roles and domains. For examples, see Sample Responses from the roles Service.

**Note:** Using the REST roles service is not an authentication method. Accessing the REST roles service is not considered as a login. This service only returns the list of roles for a user, but does not log the user in.

For specific details on domain data returned by the roles service, see Domain Data Returned by the roles Service.

For information about NLAuth, see the help topic Using User Credentials for RESTlet Authentication.

Account-specific domains are supported for RESTlets. Account-specific domains contain the account ID as part of the domain name. An account-specific domain is not dependent on the data center where the account is hosted. The domain does not change, even if the account is moved to a different data center. Your account-specific domain for RESTlets is shown on the Company Information page on the Company URLs subtab. For more information, see the help topic URLs for Account-Specific Domains. Currently, data center-specific domains are still supported, but you should start using your account-specific domain. See the help topic How to Transition from Data Center-Specific Domains.

**Important:** If you have an integration that makes requests to the production version of the roles service, and these requests are specific to a NetSuite account, your integration must include logic for handling redirection. This logic is necessary because of how the system handles requests specific to a NetSuite account. NetSuite redirects these calls to the instance of the service specific to the data center that hosts the account. For this reason, your integration must include logic for handling the 302 Found response status code, which is the code used when redirection occurs. By contrast, if your authorization header omits a NetSuite account ID, the request is handled without redirection.

Authentication for the REST roles Service

Each call to the REST roles service must include an NLAuth authorization header. This header must identify a user and the user's password. Additionally, if you need to retrieve data about Customer Center
Authentication for the REST roles Service

roles, the account ID must be included in the header. Otherwise, the account ID is optional. Note the following:

- If your request includes a NetSuite account ID, the information returned is specific to that account. All roles are returned for the user, including the Customer Center roles, and any roles created based on the Customer Center role.
- If your request omits a NetSuite account ID, the system returns details for every NetSuite account to which the user identified in the header has access. All roles except Customer Center roles are included.

For examples, see Sample Responses from the roles Service.

For details on domain data returned by the roles service, see Domain Data Returned by the roles Service.

For information about constructing an NLAuth authorization header, see the help topic Using User Credentials for RESTlet Authentication.

URLs for Accessing the REST roles Service

To use the roles service, you send a get request to the appropriate URL. However, the URL that you use can vary in the following ways:

- You use a different URL for each of the following: production, sandbox, and Release Preview accounts. For details, see Sample URLs.
- In some cases, your request may be redirected. If you have an integration that calls the roles service, your integration must include logic to handle this redirection. For details, see Calls that May Require Redirection.

Sample URLs

You can use the following URL to call the roles service, using the Get method. This URL works for production, sandbox, and release preview accounts: https://rest.netsuite.com/rest/roles and similar variants.

Note: NetSuite maintains multiple URLs for the production version of the roles service. Each data center hosts the service on the REST domain specific to that data center. However, you are not expected to know the data center of your account when you send the request. For that reason, you can use any production URL. But be aware that, in some cases, your request may be directed to a different URL, as described in the next section. For a list of all REST service domains, see the help topic Understanding NetSuite URLs.

Note: As of 2017.2, account-specific domains are supported for RESTlets, and you can access your RESTlet domain at the following URL, where 123456 is your account ID: 123456.restlets.api.netsuite.com. The data center-specific domains supported before 2017.2 will continue to be supported. For more information, see the help topic URLs for Account-Specific Domains.

Calls that May Require Redirection

Sometimes calls to the roles service include a NetSuite account ID in the authorization header. If you send this type of request to a production version of the roles service, the call may be redirected to a different URL. Redirection can occur because these requests must be handled by the same data center...
that hosts the NetSuite account. For this reason, if you have an integration that makes this type of call, the integration must include logic to handle redirection.

For example, suppose you are calling the service to retrieve roles information for a user in your NetSuite account. You may not know which data center your account is hosted in, especially if you have never called the roles service before. This knowledge gap is expected. The service is designed so that you can send your request to any of the available production URLs for the service. However, if your account is hosted in the EU and you send the request to a URL associated with a North American data center, your request will be redirected to https://rest.eu2.netsuite.com. For this reason, your integration must include logic for handling the 302 Found response status code, which is the code used when redirection occurs.

By contract, if your authorization header omits a NetSuite account ID, your request is handled without redirection.

## Domain Data Returned by the roles Service

The REST roles service lets you dynamically discover the following types of domains for any NetSuite account.

<table>
<thead>
<tr>
<th>Type of Domain</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESTlet</td>
<td>https://&lt;accountID&gt;.restlets.netsuite.com</td>
</tr>
<tr>
<td>System</td>
<td>https://&lt;accountID&gt;.app.netsuite.com</td>
</tr>
<tr>
<td>Web services</td>
<td>https://&lt;accountID&gt;.suitetalk.api.netsuite.com</td>
</tr>
</tbody>
</table>

It is important to dynamically discover these domains because, for any NetSuite account, these domains can change. They can change because NetSuite hosts data in multiple data centers, and the location of your account data can change. Therefore, any integration that includes full URLs must include logic for dynamically discovering these domains. A hard-coded URL could fail.

**Note:** As of 2017.2, account-specific domains are supported for RESTlets, and you can access your RESTlet domain at the following URL, where 123456 is your account ID: 123456.restlets.api.netsuite.com. The data center-specific domains supported before 2017.2 will continue to be supported. For more information, see the help topic URLs for Account-Specific Domains.

## Sample Responses from the roles Service

The following sections show sample responses from the REST roles service:

- Calls that Include a NetSuite Account ID
- Calls that Omit a NetSuite Account ID

### Calls that Include a NetSuite Account ID

A typical call to the roles service includes a NetSuite account ID in its authorization header. For example, the header might look like the following:

```
NLAuth nlauth_account=023456, nlauth_email=jsmith@example.com, nlauth_signature=Welcome123
```

---

REST Services
In response, the system returns data specific to that NetSuite account and user, as shown in the following example.

```json
{
  "account": {
    "internalId": "1234567",
    "name": "Test Acct",
    "type": "PRODUCTION"
  },
  "role": {
    "internalId": 14,
    "name": "Customer Center"
  },
  "dataCenterURLs": {
    "webservicesDomain": "https://<accountID>.suitetalk.api.netsuite.com",
    "restDomain": "https://<accountID>.restlets.api.netsuite.com",
    "systemDomain": "https://<accountID>.app.netsuite.com"
  }
}
```

With this approach, the system returns all roles for the user in the specified account. Results include the Customer Center role and any custom roles created based on the Customer Center role, if the user belongs to those roles. If the account ID is omitted, the service does not return Customer Center roles.

**Note:** For information about NLAuth, see the help topic *Using User Credentials for RESTlet Authentication.*

### Calls that Omit a NetSuite Account ID

A call to the roles service can omit a NetSuite account ID. With these calls, the authorization header includes only user and password data, as follows:

```
NLAuth nlauth_email=jsmith@example.com, nlauth_signature=Welcome123
```

With this type of call, the system returns data on all accounts to which the user has access. However, the system does not return the Customer Center role, nor any custom role that was created based on the Customer Center role. To have the system return Customer Center roles, use the method described in **Calls that Include a NetSuite Account ID**.

The following snippet shows a sample response to a request that omitted a NetSuite account ID.

```json
[
  {
    "account": {
      "internalId": "023456"
    },
    "role": {
      "internalId": 3,
      "name": "Administrator"
    },
    "dataCenterURLs": {
      "webservicesDomain": "https://<accountID>.suitetalk.api.netsuite.com",
    }
  }
]
Sample Responses from the roles Service

```
"restDomain": "https://<accountID>.restlets.api.netsuite.com",
"systemDomain": "https://<accountID>.app.netsuite.com"
},

{ "account": {  
  "internalId": "123456"  
  "name": "Account 2"  
  }  
  "role": {  
    "internalId": 1  
    "name": "Accountant"  
  }  
  "dataCenterURLs": {  
    "webservicesDomain": "https://<accountID>.suitetalk.api.netsuite.com",
    "restDomain": "https://<accountID>.restlets.api.netsuite.com",
    "systemDomain": "https://<accountID>.app.netsuite.com"
    }  
},

{ "account": {  
  "internalId": "123456"  
  "name": "Account 2"  
  }  
  "role": {  
    "internalId": 3  
    "name": "Administrator"  
  }  
  "dataCenterURLs": {  
    "webservicesDomain": "https://<accountID>.suitetalk.api.netsuite.com",
    "restDomain": "https://<accountID>.restlets.api.netsuite.com",
    "systemDomain": "https://<accountID>.app.netsuite.com"
    }  
}
```

⚠️ **Important:** Strings must be escaped using RFC 3986. If you do not escape characters in the header, you may receive an INVALID_LOGIN_ATTEMPT error. For more information about percent encoding, go to [https://tools.ietf.org/html/rfc5849#section-3.6](https://tools.ietf.org/html/rfc5849#section-3.6).
DataCenterUrls REST Service

The DataCenterUrls REST service lets you obtain the correct URL for external client application access to NetSuite. When you build an integration between NetSuite and an external client application, you must incorporate logic that can determine the correct URL for the appropriate NetSuite account.

For details about dynamic discovery of URLs, see the help topics Effects of Accounts' Host Data Center on URLs and Dynamic Discovery of URLs for SOAP Web Services and RESTlet Clients.

Usage Notes

The DataCenterUrls REST service does not require authentication. This makes it ideal for integrations that use token-based authentication.

When you submit your request, you must provide your account ID as an input parameter so that the information returned will be specific to your account.

You can use the DataCenterUrls REST service to discover any of the following domains:

- Account-specific web services domains: https://<accountID>.suitetalk.api.netsuite.com
- Account-specific RESTlet domains: https://<accountID>.restlets.api.netsuite.com

The information returned is different for system domains and for web services and REST domains because of the introduction of account-specific domains for SOAP web services and RESTlets in version 2017.1. In version 2017.1, the web services and REST domains become data center agnostic. These domains are unique to your account, do not contain information about the data center location but contain your account ID as part of the domain name. For more information, see the help topic Understanding NetSuite URLs.

You can send a request to one of the following URL for production, sandbox, and release preview accounts: https://rest.netsuite.com/rest/datacenterurls

Sample REST DataCenterUrls Request and Response

Send your HTTP GET request in the following format: https://rest.netsuite.com/rest/datacenterurls?account=123456

**Important:** If you receive the following error, try sending the request in the following format: https://rest.netsuite.com/rest/datacenterurls?account=123456&c=123456. The c parameter ensures that your request is routed correctly according to your version and that you receive the correct response.

This service does not require an authorization or a content-type header.

When sending a correctly formed request, the response is returned in one of the following formats. The returned information should not be parsed to retrieve any additional information.

Examples of Returned Information

Environments where account-specific domains are used:
The full request and response can look as follows.

Request:

GET /rest/datacenterurls?account=123456 HTTP/1.1
Host: <host>

Response:

HTTP/1.1 200 OK
Date: Thu, 22 Jun 2017 05:48:56 GMT
Content-Length: 179
Content-Type: application/json


REST DataCenterUrls Errors

If you send a GET request that contains a nonexistent account ID, you receive the default NetSuite domains in the response.


Additionally, you may receive the following errors.

<table>
<thead>
<tr>
<th>Error</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>USER_ERROR: You need to provide a proper value for the required field: account.</td>
<td>This error is returned if you send a GET request that does not include the account ID input parameter.</td>
</tr>
<tr>
<td>USER_ERROR: Company ID is invalid</td>
<td>This error is returned if the account ID in the request contains an invalid character, such as a space or a percent sign.</td>
</tr>
<tr>
<td>HTTP Error 405 - Method Not Allowed</td>
<td>The DataCenterUrls service accepts only the HTTP GET method. This error is returned if the request contains an unsupported HTTP method.</td>
</tr>
</tbody>
</table>

REST Services
Issue Token and Revoke Token REST Services for Token-based Authentication

You can call a token endpoint to issue a token, to revoke a token, and to obtain information about a token. See the following sections:

- Calling a token endpoint to issue a token
- Calling a token endpoint to revoke a token
- Calling a token endpoint to obtain user information based on a token

**Important:** You can use TBA with those integrations that require the Administrator role. Administrators can only create tokens for their own use by clicking the Manage Access Tokens link in the Settings portlet, or by using the token endpoint.

In addition to creating a token manually through the NetSuite UI, developers and users can issue or revoke their own tokens programmatically using a token endpoint. You can also use a token endpoint to obtain information about a token.

Use the appropriate domain to call the token endpoint:

- RESTlet domain: `https://<accountID>.restlets.api.netsuite.com/`
- System domain: `https://<accountID>.app.netsuite.com`

Users cannot programmatically issue or revoke tokens for other users using a token endpoint. For information about creating tokens for other users through the NetSuite UI, see the help topic Viewing, Editing, Creating, and Revoking TBA Tokens.

Account-specific domains are supported for RESTlets. For example, if your account ID is 123456, your account-specific REST domain would be `123456.restlets.api.netsuite.com`. For more information, see the help topic URLs for Account-Specific Domains. See also the Integration Domains section in the topic How to Transition from Data Center-Specific Domains.

**Important:** Whether using The Three-Step TBA Authorization Flow, or calling The IssueToken Endpoint, an Integration record is created and automatically installed in your account. The Require Approval during Auto-Installation of Integration preference affects whether this new record is automatically enabled. You can manage the preference at Setup > Integration > SOAP Web Services Preferences. If the box for the Require Approval during Auto-Installation of Integration preference is not checked (set to false) the State field on the new application is automatically set to Enabled, and all requests are permitted. However, if the box is checked (set to true) the State field on the new integration record is set to Waiting for Approval. In the latter case, you must manually edit the record and set the State to Enabled. Until you set the state to Enabled, all requests sent by that application are blocked.

Calling a token endpoint to issue a token

- Use the NetSuite NLAuth Authorization header. The token is created under the role specified in the NLAuth Authorization header. For more information, see the help topic Using User Credentials for RESTlet Authentication.
- Parameters must be URL encoded. This is particularly important for parameters which include special characters like spaces, for example, token name.
A token endpoint consumes two GET parameters. For an **issuetoken** request, the Consumer Key parameter is mandatory, and the Name (the name of the token) is optional.

For example:

```
https://<accountID>.restlets.api.netsuite.com/rest/issuetoken?consumerKey=<CONSUMER_KEY>&name=<TOKEN_NAME>
```

The issue token endpoint has been extended to accommodate the requirement for mandatory 2FA for highly privileged roles. There is an optional parameter, nlauth_otp, that you can include in the NLAuth Authorization header. For more information, see the help topic Mandatory 2FA, the IssueToken Endpoint, and nlauth_otp.

**Calling a token endpoint to revoke a token**

- In a call to a token endpoint to revoke a token, use either:
  - The NetSuite **NLAuth** Authorization header. See the help topic Using User Credentials for RESTlet Authentication in the topic Authentication for RESTlets.
  - The **OAuth** Authorization header. See Required Data for Using TBA with RESTlets in the topic Authentication for RESTlets.

- A token endpoint consumes two GET parameters. For a **revoketoken** request, both the Consumer Key parameter and the Token parameter are mandatory.

Here is an example of a request:

```
https://<accountID>.restlets.api.netsuite.com/rest/revoketoken?consumerKey=<CONSUMER_KEY>&token=<TOKEN>
```

**Calling a token endpoint to obtain user information based on a token**

The tokeninfo endpoint returns information about a user based on the access token. The endpoint is `https://<accountID>.restlets.api.netsuite.com/rest/tokeninfo`, where `<accountID>` is a variable for the company’s account ID. A response to a GET request contains data in JSON format, including information such as:

- Company Name
- Company ID (account ID)
- Role Name
- Role ID
- Entity ID