

GUIDE

CLIENT API TECHNICAL SPECIFICATION

Ticketing
v2.0

Table of contents

Revision History.....	3
Pilot Study	4
Intended Audience.....	5
Overview.....	5
Currently Supported Functionality	5
Authentication.....	6
New Client Onboarding	6
Authorizing API Calls.....	7
Overview	7
Endpoint Definition.....	9
GET /tickets	9
GET /tickets/{ticket_id}	13
Creating Tickets	18
GET /sites.....	19
GET /circuits.....	21
GET /circuits/{circuit_id}.....	23
POST /tickets	24
Request.....	24
Response.....	27
POST /tickets/{ticket_id}/attachments	28
Response.....	28
POST /tickets/{ticket_id}/notes.....	29
Request.....	29
Response.....	29
Appendix.....	30
Error Handling.....	30
Ticket Status Values	31
Ticket Source Values.....	31

Revision History

Revised By	Version	Page	Description
Chris.Parsons@Charter.com	1.6	-	First Release.
Chris.Parsons@Charter.com	1.7	25	Corrected curl script and parameter name for the Add Attachments endpoint.
Chris.Parsons@Charter.com	1.8	17	Revamped the advice and rules for sending ticket requests.
Chris.Parsons@Charter.com	1.901	24	Added spectrumTicketNumber to create ticket response.
Chris.Parsons@Charter.com	1.902	10	Paging added to GET /tickets.
		14	SiteAddress.id on responses, is NOT the siteGuid, this will be removed in upcoming release.
Chris.Parsons@Charter.com	1.903	13, 17	Added GET /ticket by spectrumTicketNumber capability.
Chris.Parsons@Charter.com	1.904	Page Header & Footer	Corrections to doc version number and copyright year.
Chris.Parsons@Charter.com	1.905	22 & 28	Optional Boolean fields cannot be null, they must be true, false or omitted altogether. Updated the spec to reflect this.
Chris.Parsons@Charter.com	1.91	9 & 14 20 23 25	Updates for the Linden release: GET / Tickets new query parameters and new response fields. GET /sites - count added to response GET /circuit/{circuit_id} POST /tickets - New optional fields when submitting a ticket.
Chris.Parsons@Charter.com	1.92	9 - 10 12	Added updated, resolved, closed date query params. Added readable dates to response. New example response.
Chris.Parsons@Charter.com	2.0		Formatting changes.

Pilot Study

The API service described in this document is currently in the Pilot stage. We reserve the right to make changes to this API specification and to restrict or deny access at any time.

Support

For any client API related support issues, email DL-Ent-Spectrum-API-Support@charter.com.

Intended Audience

This document is intended for Enterprise Client Software Development Teams who wish to implement our REST APIs and need to know how to onboard/authenticate an account to use the APIs. It describes how to get security tokens, understand what each of the endpoints are, and how to use them.

Overview

The Spectrum Enterprise Open API (Application Programming Interface) exposes APIs using the REST protocol. They provide Enterprise clients the ability to integrate their systems directly with some Features of the Spectrum Enterprise portal.

Currently the feature offered through this API is limited to service ticketing, however the scope will be expanded to include other features in future.

This guide explains:

- How to Register/Sign Up to create an account to use these APIs.
- How to obtain security tokens to make the API calls.
- What endpoints are currently available.
- Detailed specifications of each endpoint.

Currently Supported Functionality

- Retrieve all tickets for your organization.
- Retrieve details about a single ticket.
- Retrieve a list of Ticketing Sites (Locations) for your organization.
- Retrieve a list of Circuit Ids for your organization.
- Create a ticket for a new incident or information request.
- Update a ticket by adding an attachment.
- Update a ticket by adding a note (includes submitting close requests).

Authentication

New Client Onboarding

If you wish to use the Spectrum Enterprise Public APIs your organization must first be registered on our PiNxt Gateway.

The process to register your organization is as follows:

- The Spectrum Enterprise (SE) team will request that you provide a public GPG or PGP key (see below).
- SE Team will submit an intake to our API Gateway (PiNxt) team.
- PiNxt will create a set of credentials, (clientId and secret), for your organization, encrypt them (using the PGP or GPG key) and send them to you.
- You decrypt the credentials using the private PGP key.

Note: GPG and PGP encryption are essentially the same thing, PGP is just a proprietary version of GPG. The GPG/PGP encryption is not part of the API authentication. It is the way the Charter API Gateway team ensure secure transmission of the API credentials to clients. It is just an open source way to encrypt data that can then be sent via a less secure method (eg. via email) to the client. Only the client, having created the keys, (and being the sole holder of the private key), can then decrypt the message or file that contains the PiNxt credentials. Once they have the PiNxt credentials, PGP plays no further part in the use of these APIs.

There are several open source tools, eg GPGTools (MAC), Gpg4win (windows) or if on a Mac install gpg with homebrew (brew install gpg2).

GPG References:

- For gpg on the command line and exporting your public key:
<https://help.github.com/en/articles/generating-a-new-gpg-key>
- For using the gpg2 on the command line: https://fedoraproject.org/wiki/Creating_GPG_Keys
- For using gpgtools: <https://gpgtools.tenderapp.com/kb/how-to/first-steps-where-do-i-start-where-do-i-begin-setup-gpgtool>

Authorizing API Calls

Overview

In order to make calls to the Spectrum Enterprise API a 'JSON Web Token', (JWT) is required. The token is a temporary key that provides access to the Client API, it must be included on the header of each request.

Once a token is received, the client is expected to cache that token and reuse it for subsequent API calls for the duration of its lifetime. The lifetime of the access token is advertised by the "expires in" property in the response body. This represents the duration of the token's lifetime in seconds (typically tokens last for 1 hour). Clients should NOT generate a new client token with every API invocation. PINXT will be restricting the frequency with which clients can invoke this endpoint to help protect and preserve resources for all clients.

Obtaining an Access Token

To obtain a new access token, a call must be made to our API Gateway using the Client Id and Secret that you received after onboarding (see above for onboarding instructions).

The endpoint to get a new token is; POST <https://apis.spectrum.net/auth/oauth/v2/token>

Request

```
curl -X POST https://apis.spectrum.net/auth/oauth/v2/token
--header "accept: application/json"
--header "Content-Type: application/x-www-form-urlencoded"
--data "grant_type=client_credentials&client_id={<clientId>}&client_secret={<secret>}"
```

Base 64 encoded version of the credentials

```
curl --location --request POST 'https://apis.spectrum.net/auth/oauth/v2/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Basic {base64(clientId:secret)}' \
--data-urlencode 'grant_type=client_credentials'
```

Response

```
{
  "resultCode": "200",
  "resultCodeName": "PROCESSED",
  "resultMessage": "PROCESSED",
  "token_type": "bearer",
  "access_token": "<encrypted_token_XYZ>",
  "expires_in": 3600
}
```

Using An Access Token

A valid Token must be included in the header of subsequent API call.

Example using a token to call GET /tickets endpoint:

```
curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-  
b2b/v1/tickets' \  
--header 'Content-Type: application/json;charset=utf-8' \  
--header 'Accept: application/json;charset=utf-8' \  
--header 'Authorization: Bearer <encrypted_token_XYZ>','
```

When each call is made, it is received by the API Gateway, (PiNxt). The Gateway service decrypts the access token, verifies it and then calls the Client API endpoint, (eg GET /tickets), passing the decrypted token on the header.

Data Scoping

The API service reads the Client Id from the token and uses it to scope the request for every endpoint. This means that each client is limited to only accessing or creating data for their organization, guaranteeing confidentiality.

So, for example, if a call is made by 'ABC Inc' to retrieve the ticket: 1234, but that ticket is linked to 'XYZ Trading', then the call will return an error (403 Forbidden). The request will only succeed if the client id on the ticket matches the client id in the API token. This also applies to circuit and site data.

Endpoint Definition

GET /tickets

Retrieve all tickets for your organization, (created in past 12 months).

GET

/tickets

Retrieve open and/or historical tickets for a portal account.

▼

Query Parameters

Name	Type	Description	Examples
status	String, optional enum (In Progress, Closed)	Status of tickets to be retrieved	Closed
openDateStart	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date opened is on or after this date.	2023-02-25T21:05:14.000Z
openDateEnd	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date opened is on or before this date.	2023-03-01T00:00:00.000Z
updateDateStart	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date updated is on or after this date.	2023-02-25T21:05:14.000Z
updateDateEnd	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date updated is on or before this date.	2023-03-01T00:00:00.000Z
resolvedDateStart	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date resolved is on or after this date.	2023-02-25T21:05:14.000Z
resolvedDateEnd	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date resolved is on or before this date.	2023-03-01T00:00:00.000Z
closedDateStart	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date closed is on or after this date.	2023-02-25T21:05:14.000Z
closedDateEnd	String, optional Date Time (YYYY-MM-DDTHH:MM:SS.000Z)	Only retrieve tickets where date closed is on or before this date.	2023-03-01T00:00:00.000Z
paginate	Boolean, optional	Whether to paginate results. Defaults to false. Page size is 25 tickets.	true
returnPageNumber	Integer, required if paginate is true	If results are paginated then this determines which page to return. Default is 1 (first page).	2

Note: if no parameters specified, paging is ON, so only the first page is returned.
If you need all tickets returned without paging then set paginate = false.
Ticket history goes back 12 months.

Response

Field Name	Data Type	Example	Description
ticketsCount	Integer	21	Total number of tickets returned by this query
resultsRetrieved	String	25 of 93	Number of results on this page vs total results
pageReturned	String	2 of 4	Which page of the total pages of results was returned
previousPageURL	String		URL to get previous page of results
nextPageURL	String		URL to get next page of results
openDateStart	Epoch Date	1693584000	Start opening date/time of tickets returned, in EPOCH (UTC), in seconds
openDateEnd	Epoch Date	1693584000	End opening date/time of tickets returned, in EPOCH (UTC), in seconds
resolvedDateStart	Epoch Date	1693584000	Start resolved date/time of tickets returned, in EPOCH (UTC), in seconds
resolvedDateEnd	Epoch Date	1693584000	End resolved date/time of tickets returned, in EPOCH (UTC), in seconds
closedDateStart	Epoch Date	1693584000	Start closed date/time of tickets returned, in EPOCH (UTC), in seconds
closedDateEnd	Epoch Date	1693584000	End closed date/time of tickets returned, in EPOCH (UTC), in seconds
tickets	Array/objects		Array of ticket objects
Ticket Object	(see definition below)		

Examples

Return first page of all tickets for my account:

```
curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-  
b2b/v1/tickets' \  
--header 'Content-Type: application/json;charset=utf-8' \  
--header 'Accept: application/json;charset=utf-8' \  
--header 'Authorization: Bearer <token>'
```

Return all tickets for my account as a single collection:

```
curl --location 'https://apis.spectrum.net/entservices/ticketing-  
b2b/v1/tickets?paginate=false' \  
--header 'Content-Type: application/json;charset=utf-8' \  
--header 'Accept: application/json;charset=utf-8' \  
--header 'X_TXN_ID: e8a763c9-4c5b-48c0-97c2-c18d70084de2' \  
--header 'Authorization: Bearer <token>'
```

Return all tickets for my account opened between 4pm Nov 1st 2023 and 10am Dec 31st 2023 as a single collection:

```
curl --location 'https://apis.spectrum.net/entservices/ticketing-  
b2b/v1/tickets?openDateStart=2023-11-01T16:00:00.000Z&openDateEnd=2023-12-  
31T10:00:00.000Z&paginate=false' \  
--header 'Content-Type: application/json;charset=utf-8' \  
--header 'Accept: application/json;charset=utf-8' \  
--header 'X_TXN_ID: e8a763c9-4c5b-48c0-97c2-c18d70084de2' \  
--header 'Authorization: Bearer <token>'
```

Example Response

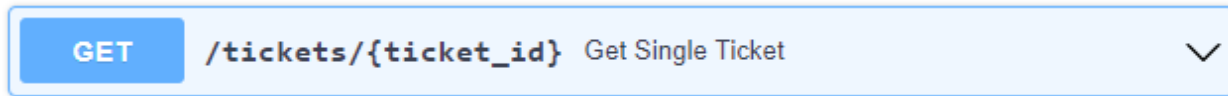
```

1  ✓ {
2    "ticketsCount": 72,
3    "resultsRetrieved": "25 of 72",
4    "pageReturned": "1 of 3",
5    "previousPageURL": "https://apis.spectrum.net/entservices/ticketing-b2b/v1/tickets?paginate=true&returnPageNumber=1",
6    "nextPageURL": "https://apis.spectrum.net/entservices/ticketing-b2b/v1/tickets?paginate=true&returnPageNumber=2",
7    "closedDateStart": 1693584000,
8    "closedDateEnd": 1757289600,
9  ✓  "tickets": [
10   ✓  {
11      "id": "RSGADG2LITIC5ASBZCZMSAYTNXPKZI",
12      "spectrumTicketNumber": "ECT-82521505",
13      "customerTicketNumber": "",
14      "billingAccountNumber": "",
15      "status": "Closed",
16      "ticketReportSource": "B2B-API",
17      "needsClientAttention": false,
18      "businessImpact": "Information",
19  ✓  "impactedServices": [
20      [
21      ],
22      "impactedCircuits": [],
23      "siteGuid": "CSGADGH1JT34HARQG9QLRPGM361HU8",
24  ✓  "siteAddress": {
25      "id": "7706F9DF9264182A6E728CA27C26F616",
26      "addressLine1": "3918 S KINGS AVE",
27      "city": "BRANDON",
28      "state": "FL",
29      "zip5": "33511",
30      "latitude": "27.9056",
31      "longitude": "-82.2881"
32    },
33  ✓  "siteContacts": [
34      "SPECTRUM SITE EBOND TEST USER"
35    ],
36  ✓  "contacts": [
37  ✓  {
38      "name": "SPECTRUM SITE EBOND TEST USER",
39      "type": "SITE",
40      "phoneNumber": "1231231234",
41      "email": "TESTYTHESITESTER@EMAIL.COM"
42    }
43  ],
44      "ticketType": "ECT",
45      "ticketSubType": "ticket",
46      "assignedAction": "",
47      "dateCreated": 1696369378,
48      "dateCreatedYMD": "2023-10-03T21:42:58.000+0000",
49      "dateUpdated": 1696824000,
50      "dateUpdatedYMD": "2023-10-09T04:00:00.000+0000",
51      "dateResolved": 1696824000,
52      "dateResolvedYMD": "2023-10-09T04:00:00.000+0000",
53      "dateClosed": 1696824000,
54      "dateClosedYMD": "2023-10-09T04:00:00.000+0000",
55      "resolutionCode": "Automation-Closed-Ticketing System-Ticket",
56      "resolutionMessage": "Automation Closed Ticket at Ticketing System"
57    },
58  ✓  {
59      "id": "RSGADG2LITIC5ASCC166SBCCESRUM1",

```

GET /tickets/{ticket_id}

Retrieve a single ticket by ticket id. (The ticket must be for your organization).



Query Parameters

Name	Type	Description	Examples
ticket_id	string, required	The ticket.id OR the spectrumTicketNumber of the ticket to be retrieved	632012b-d3w2-4354-23d3- 45343432341b3d1 OR ECT-12345678

Response

Field Name	Data Type	Example	Description
Ticket Object	(see definition below)		
customerActionMinuteDetails	array/objects		
action	string	Customer-Awaiting Additional Information,	The type of action required
startTime	number	1601391028	The date/time of when the phase started, in EPOCH (UTC), in seconds
endTime	number	1601410615	The date/time of when the phase ended, in EPOCH (UTC), in seconds
notes	array/objects		
dateCreated	number	1601391028	The date/time of when ticket was created, in EPOCH (UTC), in seconds
noteType	string	General	The type of note (eg. General, Escalation, Handoff ETS, etc)
note	string	We are reaching out to confirm if your issue has been resolved	The contents of the note.

Ticket Object

Field Name	Data Type	Example	Description
id	string	632012b-d3w2-4354-23d3-B45343432341b3d1	Unique id of the ticket (Guid) within Charter ticketing system
spectrumTicketNumber	string	ECT-12345678	User friendly ticket id/name within Charter ticketing system
customerTicketNumber	string	TEST9999	Id of the corresponding ticket in your system, if there is one.
billingAccountNumber	string	8260259230587430	Charter Service location billing account number
status	string	In Progress	Ticket status, eg In Progress or Closed
ticketReportSource	string	Inbound Call	How issue was reported to Charter (Email, Inbound Call, SE.Net, B2B-API)
needsClientAttention	boolean	true	Indicates if ticket is waiting on a response from the client.
businessImpact	string	Outage	Outage, Impairment, Failover or Information. Failover indicates we switched over to backup systems temporarily - but with no significant impact on network performance.
impactedServices	array/strings	["Data-FIA"]	Type of service impacted (derived from the Circuit id)
impactedCircuits	array/strings	["90.L2YY.000232..TWCC"]	List of circuit ids impacted by the issue
siteGuid	string	EDFR1DIER02ERKI877SDW	Unique id of the site. This is the id to use if submitting issues for the site without a circuit id.
siteAddress	Object		
id	string		This is NOT the same as the siteGuid and will be removed in upcoming release.
addressLine1	string	123 Main Street	Street address of the Site
city	string	DALLAS	Site city
state	string	TX	Site state
zip5	string	12345	Site zipcode
latitude	string	32.876065	Site latitude coordinates
longitude	string	-96.669000	Site Longitude coordinates
siteContacts	array/strings	"John Doe", Fred Smith"	List of the contact's names
contacts	array/objects		
name	string	John Doe	Name of contact person
type	string	Local Contact	Type of contact eg "Site", "Technical", etc
phoneNumber	string	123-123-1234	Contact phone number
email	string	john.doe@emailprovider.com	Contact email
ticketType	string	ECT	Always 'ECT' this field has been added for a possible future expansion to include other types of ticket.

Field Name	Data Type	Example	Description
ticketSubType	string	No Connectivity	Examples: No Connectivity Degraded Services, Other
assignedAction	string	Maintenance - Waiting for Approvals	Identifies the action needed to resolve the ticket, when it is in the 'Assigned' or 'In Progress' state.
dateCreated	integer	1632500084	Ticket creation date/time, in EPOCH (UTC), in seconds
dateCreatedYMD	integer	2023-09-15T05:55:29.123+0000	Ticket creation date/time, in readable format (UTC)
dateUpdated	integer	1632500084	Ticket updated date/time, in EPOCH (UTC), in seconds
dateUpdatedYMD	integer	2023-09-15T05:55:29.123+0000	Ticket updated date/time, in readable format (UTC)
dateResolved	integer	1632500084	Ticket resolved date/time, in EPOCH (UTC), in seconds
dateResolvedYMD	integer	2023-09-15T05:55:29.123+0000	Ticket resolved date/time, in readable format (UTC)
dateClosed	integer	1632500084	Ticket closed date/time, in EPOCH (UTC), in seconds
dateClosedYMD	integer	2023-09-15T05:55:29.123+0000	Ticket closed date/time, in readable format (UTC)
resolutionCode	string	Spectrum-Verified-Service-Maintenance Notification	Code identifying resolution type
resolutionMessage	string	Spectrum-VerifiedMaintenance Notification at Service	Description of resolution type

Notes

All Date Times are UTC.

Epoch dates are an integer, the number of seconds since midnight January 1, 1970.
Example, 1601391028, converts to Tuesday, 29 September 2020 14:50:28.

Readable dates in response are YYYY-MM-DDTHH:MM:SS.mmm+0000 (+0000 is zero offset from UTC).

Example Response

```

{
  "id": "RSGADGJJIOCU6AS0P48V5J0R1XHZCX",
  "spectrumTicketNumber": "ECT-82979269",
  "customerTicketNumber": "",
  "billingAccountNumber": "",
  "status": "Closed",
  "ticketReportSource": "B2B-API",
  "needsClientAttention": false,
  "businessImpact": "Outage",
  "impactedServices": [
    "Data-Broadband Internet"
  ],
  "impactedCircuits": [
    "81.L1XX.011988..CHTR"
  ],
  "siteGuid": "CSGADGH1JT34HARQG9QLRPGM361HU8",
  "siteAddress": {
    "id": "7706F9DF9264182A6E728CA27C26F616",
    "addressLine1": "3918 S KINGS AVE",
    "city": "BRANDON",
    "state": "FL",
    "zip5": "33511",
    "latitude": "27.9056",
    "longitude": "-82.2881"
  },
  "siteContacts": [
    "JOHN DOE"
  ],
  "contacts": [
    {
      "name": "JOHN DOE",
      "type": "SITE",
      "phoneNumber": "1231231234",
      "email": "SEVSICH@ING-EMAIL.COM"
    }
  ],
  "ticketType": "ECT",
  "ticketSubType": "ticket",
  "assignedAction": "",
  "dateCreated": 1711034311,
  "dateCreatedYMD": "2024-03-21T15:18:31.000Z",
  "dateUpdated": 1711512000,
  "dateUpdatedYMD": "2024-03-27T04:00:00.000Z",
  "dateResolved": 1711512000,
  "dateResolvedYMD": "2024-03-27T04:00:00.000Z",
  "dateClosed": 1711512000,
  "dateClosedYMD": "2024-03-27T04:00:00.000Z",
  "resolutionCode": "Automation-Closed-Ticketing System-Ticket",
  "resolutionMessage": "Automation Closed Ticket at Ticketing System",
  "customerActionMinuteDetails": [],
  "notes": [
    {
      "dateCreated": 1711034311,
      "dateCreatedYMD": "2024-03-21T15:18:31.000Z",
      "noteType": "Customer-Note",
      "note": "Customer Provided CID: 81.L1XX.011988..CHTR<br><br><strong>Issue Description: </strong>TEST TICKET - Client API Test</br><strong>Impacted Services: </strong>Data-Fiber Internet</br><strong>Business Impact: </strong>Outage</br><strong>Does building have power? </strong>Yes</br><strong>Does equipment have power? </strong>Yes</br><strong>Trouble Shooting Steps: </strong>Turned off and on again. Test ticket.</br><strong>Access Hours:</strong> 09:00 to 17:00</br><strong>Additional Information: </strong>This is a test ticket for ING - Spectrum integration API call"
    }
  ],
}

```



```

    {
      "dateCreated": 1711132534,
      "dateCreatedYMD": "2024-03-22T18:35:34.000Z",
      "noteType": "Customer-Note",
      "note": "Add comments 1- test"
    }
  ]
}

```

Example Calls:

Using the ticket.id

```

curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-
b2b/v1/tickets/ABCDEF03ABCABCAR123LUROQWE41GDD' \
--header 'Content-Type: application/json;charset=utf-8' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Bearer <Token>'

```

Using the spectrumTicketNumber

```

curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-
b2b/v1/tickets/ECT-12345678' \
--header 'Content-Type: application/json;charset=utf-8' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Bearer <Token>'

```

Creating Tickets

Guidelines

We ask that you make every effort to include all relevant data when submitting a ticket. For example, we require at least one contact, but if your source ticket has recorded three contacts, we ask that you include all three on the request. Also if you record any special instructions for the site, for example site access rules, include them either in the description or additional info fields. This will greatly improve the usefulness and efficiency of the API and reduce the need for follow up phone calls and emails from our support teams.

Site and Circuit Rules

- Either a site id (siteGuid) or a circuit id, MUST be provided. These identify the service location for us.
- Up to five circuit ids can be submitted on a single trouble ticket, but they should be circuits at the same service location.
- If there are issues for circuits at different locations, then submit one request per circuit/location.
- In the absence of a circuit id, the site id is used to identify the service location.
- The GET /tickets/sites endpoint returns a list of the site addresses, (service locations), associated with your account, along with the site id (siteGuid) for each.
- The GET /tickets/circuits endpoint returns a list of the circuits associated with your account along with their terminating address.
- The circuit ids and/or the site id are validated. If they are not linked to your organization you will receive an error. If you believe they should be valid, contact support to report the data inaccuracy.

Alarm Tickets

If you are submitting a ticket that is the result of an automated network alarm, indicate this by setting Alarm = true, on the request.

Avoid automatically sending every alarm ticket to us. We expect to receive only those issues that truly require our attention. Failure to follow this rule may result in a restriction, or denial of access to the API.

If you intend to automate the feed of your tickets to us, think carefully about how you will limit them to only the ones that are Spectrum issues.

Power Questions

There are 2 fields about power;

buildingPower - this is verification that the building in which the Charter equipment is installed, has power.

equipmentPower - this is verification that the Charter Equipment specifically, has power. So if there is for example a rack of various components from various vendors, this indicates that the Charter components have power.

The fields are optional, omitting them is interpreted as power could not be verified.

MAKE EVERY ATTEMPT TO VERIFY THE SPECTRUM EQUIPMENT HAS POWER BEFORE SENDING US A TICKET.

GET /sites

GET /sites Get sites for a cxid or company name



This is a Site lookup to assist you in providing our site id, (SiteGuid), when submitting ticket requests. It retrieves all currently active ticketing sites, (locations), for your organization.

Response

Field Name	Data Type	Example	Description
sitesCount	Integer	25	Number of sites returned in the response
sites	array/objects		
cxId	String	f13df5ae-e258-468c-9785-a22deaew2fb23	Unique identifier for your organization in Spectrum Enterprise datasets
addressLine1	String	123 MAIN ST	Street Address
addressLine2	String	STORE 123	Unit or Suite Address
city	String	DALLAS	City
State	String	TX	State
zipCode	String	12345	Zip
siteGuid	String	er67sq5ae-g343-565h-7688-s22gbdsdw2gb56	Unique id for the Site (service location). *This is the location id needed when creating new tickets.*

```

1  {
2    "sitesCount": 2,
3    "sites": [
4      {
5        "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
6        "addressLine1": "3918 S KINGS AVE",
7        "addressLine2": "",
8        "city": "BRANDON",
9        "state": "FL",
10       "zipCode": "33511",
11       "siteGuid": "CSGADGH1JT34HARQG9QLRPGM361HU8"
12     },
13     {
14       "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
15       "addressLine1": "6399 S FIDDLERS GREEN CIR",
16       "addressLine2": "",
17       "city": "ENGLEWOOD",
18       "state": "CO",
19       "zipCode": "80111",
20       "siteGuid": "CSGADGH1JT34HARQG9S9RPGM541HYP"
21     }
22   ]
23 }

```

Example Call

```
curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-  
b2b/v1/sites' \  
--header 'Content-Type: application/json;charset=utf-8' \  
--header 'Accept: application/json;charset=utf-8' \  
--header 'Authorization: Bearer <Token>'
```

GET /circuits

GET

/circuits Get circuits for a cxid

This is a circuit lookup to assist you in providing the circuit id, when submitting ticket requests. It retrieves all currently active circuits for your organization. It retrieves a list of all network circuit ids for your organization.

There are no query parameters, this endpoint will only return circuits for your organization, based on the authorization credentials.

Response

Field Name	Data Type	Example	Description
circuitsCount	Integer	3	Number of circuit objects returned.
circuits	Array/objects		
circuitId	String	99.A9AA.A123456.AA.AAAA	Unique id of a circuit. *This is the circuit id needed when creating new tickets.*
cxId	String	f13df5ae-3333-468c-9785-a22deaew2fb23	Unique identifier for your organization in Spectrum Enterprise datasets
location	Object		
streetAddress1	String	123 MAIN ST	Street Address
streetAddress2	String	STORE 123	Unit or Suite Address
city	String	DALLAS	City
State	String	TX	State
zip	String	12345	Zip

Example Response

```

1  {
2    "circuitsCount": 5,
3    "circuits": [
4      {
5        "circuitId": "51.L1XX.010501..TWCC",
6        "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
7        "location": {
8          "streetAddress1": "11921 N MOPAC EXPY",
9          "streetAddress2": "",
10         "city": "AUSTIN",
11         "state": "TX",
12         "zip": "78759"
13       }
14     },
15     {
16       "circuitId": "81.L1XX.011988..CHTR",
17       "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
18       "location": {
19         "streetAddress1": "3918 S KINGS AVE",
20         "streetAddress2": "",
21         "city": "BRANDON",
22         "state": "FL",
23         "zip": "33511"
24       }
25     },
26     {
27       "circuitId": "94.L1XX.001462..CHTR",
28       "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
29       "location": {
30         "streetAddress1": "6399 S FIDDLERS GREEN CIR",
31         "streetAddress2": "",
32         "city": "ENGLEWOOD",
33         "state": "CO",
34         "zip": "80111"
35       }
36     },
37     {
38       "circuitId": "94.L1XX.001463..CHTR",
39       "cxId": "2A8C9B32-D557-4170-ABC1-CDA9CE30D621",
40       "location": {

```

Example Call

```

curl --location --request GET 'https://apis.spectrum.net/entservices/ticketing-
b2b/v1/circuits' \
--header 'Content-Type: application/json;charset=utf-8' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Bearer <Token>'

```

GET /circuits/{circuit_id}

GET

`/circuits/{circuit_id}`

Get a single circuit



This is a single circuit lookup to allow you to validate a circuit id.

There are no query parameters, this endpoint will only return the circuit if it is linked to your organization, based on the authorization credentials.

If no circuit is returned, check the circuit id value and formatting.
If you are sure it is a valid circuit id in the correct format,
then contact: DL-Ent-Spectrum-API-Support@charter.com.

Response

Field Name	Data Type	Example	Description
circuitId	String	94.L1XX.001464..CHTR	Unique id of a circuit. *This is the circuit id needed when creating new tickets.*
cxId	String	2ABC9B32-e258-468c-9785-a22deaew2fb23	Unique identifier for your organization in Spectrum Enterprise datasets
location	Object		
streetAddress1	String	123 MAIN ST	Street Address
streetAddress2	String	STORE 123	Unit or Suite Address
city	String	DALLAS	City
state	String	TX	State
zip	String	12345	Zip

```

1  {
2      "circuitId": "94.L1XX.001464..CHTR",
3      "cxId": "2ABC9B32-D557-4170-ABC1-CDA9CE30D621",
4      "location": {
5          "streetAddress1": "6399 S FIDDLERS GREEN CIR",
6          "streetAddress2": "",
7          "city": "ENGLEWOOD",
8          "state": "CO",
9          "zip": "80111"
10     }
11 }

```

POST /tickets

POST**/tickets** Create a New Ticket.

Request

A single ticket object is required to be in the request body.

Field Name	Data Type	Rules	Example	Description
siteGuid	string	required if no circuitId	e34f564qw-e456-123c-9785-n322ade1d1	This is the id of the 'Site' where ticket issue is located. To get a list of valid site ids for your organization, use the GET /sites endpoint.
circuits	array	required if no siteGuid		Circuit affected by the degraded service or outage.
circuitId	string	must be valid if no siteGuid	"51.L4XX.000589..TWCC"	If multiple circuits are included, (up to 5), they MUST be for services at the same site.
submitterId	string	required	Joe.Smith@email.com	User id of ticket creator or default value for alarm eg alarm@xyz.com
alarm	boolean	optional	true, false Or omit the field (same as false) NOTE: including the field with no value is NOT valid (if null then remove the field from the request or insert the word 'null' i.e "alarm": null,)	Was the ticket generated by an automatic network alarm?
issueDescription	string	required	free format string	Describe the issue
issueStartDate	string	Optional	Free format string	Date that issue was first noticed. This is provided to our support for informational purposes. It does not impact SLAs
issueStartTime	string	Optional	Free format string	Time of day the issue was first noticed. This is provided to our support for informational purposes. It does not impact SLAs
customerTicketId	string	required	free format string	Id of the corresponding ticket in your system, if there is one.
businessImpact	enum	required	Outage, Impairment, Information	Level of impact - Outage, degraded service or information needed.
buildingPower	boolean	optional	'true' - power supply has been verified. 'false' - power is out 'null' or field is omitted - power could not be verified	Does the building housing Charter equipment have power?

			NOTE: including the field with no value is NOT valid (if null then remove the field from the request or insert the word 'null' i.e "buildingPower": null,)	
equipmentPower	boolean	optional	'true' - power supply has been verified. 'false' - power is out 'null' or field is omitted - power could not be verified NOTE: including the field with no value is NOT valid (if null then remove the field from the request or insert the word 'null' i.e "equipmentPower": null,)	Is the CHARTER EQUIPMENT receiving power and turned on?
troubleShootingStepsTaken	string	optional	free format string	What steps have been taken already
contacts	array	Required, must have one where type="Site"		Who to contact regarding the issue.
type	enum	required	Technical, Site	Type of contact person; Site or Technical (a Site contact is required)
firstName	string	required	free format string	
lastName	string	required	free format string	
phoneNumber	string	required	free format string	
Email	string	required	free format string	
availableStartTime	string	optional	free format string	From what time of day is this contact available
availableEndTime	string	optional	free format string	To what time of day is this contact available
unavailableDays	string	optional	free format string	Days when this contact will not be available
siteAccessStartTime	time (24hr)	optional, format HH:MM	09:00	From what time (of day) can site be accessed
siteAccessEndTlme	time (24hr)	optional, format HH:MM	17:00	At what time (of day) does site access end
additionalInfo	string	optional	free format string	Any other notes.

Example Request Body

```

{
  "siteGuid": "e34f564qw-e456-123c-9785-n322ade1d1",
  "circuits": [
    "71.L1MK.177355.MOCK",
    "51.L4XX.000589..TWCC",
    "51.L1MK.019427..TWCC",
    "71.MKWI.177355.MOCK"
  ],
  "submitterId": "Joe.Smith@xyz.com",
  "alarm": false,
  "issueDescription": "Describe your problems here",
  "customerTicketId": "ABC1234",
  "businessImpact": "Outage",
  "buildingPower": "Yes",
  "equipmentPower": "No",
  "troubleShootingStepsTaken": "Turned it off and on again",
  "contacts": [
    {
      "type": "Site",
      "firstName": "Johnny",
      "lastName": "Jacobsen",
      "phoneNumber": "303-123-1111",
      "email": "jj@anybiz.com"
    },
    {
      "type": "Technical",
      "firstName": "Freddy",
      "lastName": "Fourfingers",
      "phoneNumber": "303-223-1111",
      "email": "ff@anybiz.com"
    },
    {
      "type": "Site",
      "firstName": "Steve",
      "lastName": "Smith",
      "phoneNumber": "303-123-1111",
      "email": "ss@anybiz.com"
    }
  ],
  "siteAccessStartTime": "09:00",
  "siteAccessEndTime": "17:00",
  "additionalInfo": "Please use access code 1234 for front gate"
}

```

Response

A successful response returns the ticket id and the ticket number of the newly created ticket.

```

200 Success
{
  "ticketId": "RSGJR030ERPZ4ARQI5JVRPI9LJ5SIE",
  "spectrumTicketNumber": "ECT-12345678"
}

```

The ticketId is needed when using the GET /ticket/<ticketId> endpoint, (this will soon be enhanced to accept either the id or number). The ticket number is given out by our support team as a reference for the enquiry or issue.


Example

```

curl --location --request POST 'https://apis.spectrum.net/entservices/ticketing-b2b/v1/tickets' \
--header 'Content-Type: application/json;charset=utf-8' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Bearer <token>' \
--data-raw '
{
  "siteGuid": "GHTER5VOGS4TQAPWXW7V8C000TEST",
  "circuits": ["22.L1ZZ.803238..TEST","51.LEXT.803376..TEST"],
  "submitterId": "John.Smith@emailprovider.com",
  "alarm": false,
  "issueDescription": "TEST TICKET - Client API Test",
  "customerTicketId": "VZW123",
  "businessImpact": "Outage",
  "buildingPower": true,
  "equipmentPower": true,
  "troubleShootingStepsTaken": "Turned off and on again",
  "contacts": [
    {
      "type": "Site",
      "firstName": "John",
      "lastName": "Doe",
      "phoneNumber": "123-123-1234",
      "email": "john.doe@emailprovider.com"
    }
  ],
  "siteAccessStartTime": "09:00",
  "siteAccessEndTime": "17:00",
  "additionalInfo": "This is only a test not a real ticket"
}'

```

POST /tickets/{ticket_id}/attachments

POST /tickets/{ticket_id}/attachments Add multiple file attachments to a ticket by ticketId. 

Request

This endpoint accepts 1 path parameter and two formData parameters, all are required.

Parameter	Data Type	Description
ticket_id	string, required	Unique id of the ticket to which file will be attached.
file	formData:file of type; jpg, png, gif, xls, xls, csv, xml, pdf, docx, doc, txt, msg or zip	A single file of not more than 1 Mb
submitterId	formData:string	User id of the request submitter

Response

A successful response returns the ticket id to which the file was attached.

```
200 Success
{ "ticketId": "123e4567-e89b-12d3-a456-426655440000" }
```

Example

```
curl --location 'https://apis.spectrum.net/entservices/ticketing-b2b/v1/tickets/AAAAAG2LITIC5ASC17U0SB1I8LWQAE/attachments' \
--header 'Accept: application/json' \
--header 'Authorization: Bearer <token>' \
--form 'file=@"/path/to/filename.txt"' \
--form 'submitterId="Tom.Cruise@hollywood.com"'
```

POST /tickets/{ticket_id}/notes

POST**/tickets/{ticket_id}/notes** Add notes to an existing ticket

Request

This endpoint accepts 1 path parameter and data in the body.

Field Name	Data Type	Description
ticket_id	Path parameter string, required	Unique id of the ticket to which note will be attached.
note	string, max length: 10,000 char	The contents of the note to be added
submitterId	String, max length 254	The email address associated to the user creating the ticket.
Action	Enum Customer-Note Customer-Closure	The type of action associated with the adding of this note.

Response

A successful response returns the ticket id to which the note was added.

200 Success

```
{ "ticketId": "123e4567-e89b-12d3-a456-426655440000" }
```

Example

```
curl --location --request POST 'https://apis.spectrum.net/entservices/ticketing-b2b/v1/tickets/TESTR030AHPHVARP1BLURO1I94TEST/notes' \
--header 'Content-Type: application/json;charset=utf-8' \
--header 'Accept: application/json;charset=utf-8' \
--header 'Authorization: Bearer <Token>' \
--data-raw '
{
  "note": "We turned it off and on again - seems to be working now.",
  "submitterId": "johnSmith@emailprovide.com",
  "action": "Customer-Note"
}
'
```

We do not allow updates to be made directly to existing tickets via the API. But this endpoint can be used to add notes or respond to action items on an issue. Adding a note will bring the ticket back to the attention of support.

Close a Ticket: there is no way to directly close a ticket using the API, but you can use this endpoint to request a ticket be closed. Set Action = "Customer-Closure" and describe why you are requesting the ticket be closed, in the note field.

Escalate a Ticket: The API can be used to prompt support to look at a ticket, adding a note re-prioritizes it for them, as mentioned above, but to formally escalate a ticket to a higher level, follow our standard procedure found here:

<https://enterprise.spectrum.com/support/faq/account/how-to-escalate-technical-issue.html>

Appendix

Error Handling

401 Unauthorized – issues with the token
Check the token is valid and has not expired

Example Response:

```
{
  "resultMessage": "Access token expired",
  "resultCode": "2076",
  "resultCodeName": "INVALID_TOKEN"
}
```

403 Forbidden – problem authenticating the credentials, url is wrong, or ticket id does not match credentials
Check the correct credentials were used to obtain a token, check the url does not have typos, check the request parameters. This error could occur, for example, if an attempt is made to retrieve a ticket that does not match the authenticated client id.

If the problem persists, contact support, it may be a configuration issue on our side.

400 Bad Request – issues with the submitted data, or ticket may not be updateable.
Check the request, ensure all required fields have data, no invalid values, etc.

Example Response:

```
{
  "message": "One or more invalid circuits : 99.A9AA.A123456.AA.AAAA"}
}
```

If an attempt is made to update a ticket that is not in an updateable state...

Example Response

```
{
  "message": "Ticket is not in editable state"
}
```

JSON Parse errors

These indicate an issue with the formatting of the JSON ticket object in the request body.
ie a missing comma, quote mark or value.

For example if an optional Boolean field is passes as "fieldname": , it will trigger this error.

If the field is null/empty then the JSON format is "fieldname": null, or omit the field altogether.

Example Response:

```
"error": "Bad Request",
"message": "JSON parse error: Unexpected character (' ' (code 44)): expected a value; nested exception is
com.fasterxml.jackson.core.JsonParseException: Unexpected character (' ' (code 44)): expected a value\n at [Source:
(org.springframework.util.StreamUtils$NonClosingInputStream); line: 7, column: 21]"
```

404 Not Found – resource not found
Server was found but could not locate the requested page – check the url is accurate.

500 Server Error – the network resource was currently unavailable. Please try again later.
If the problem persists, contact support.

204 No Content – This is not an error but indicates that the request returned no data.

Ticket Status Values

Indicates what stage of processing the ticket is in.

Value	Description
In Progress	Ticket is new or in progress.
Closed	Ticket has been resolved or closed

Note: this is not the actual status of the ticket in our Remedy ticketing system. We have simplified the values for external consumption.

Ticket Source Values

How was the issue communicated to Charter, (eg Enterprise Portal, phone call, email, etc).

Value	Description
Inbound Call	The issue was called in to support via a telephone call.
Email	The issue was sent in an email to support.
Alarm	Ticket was generated by a Charter network alarm.
SE.NET	Ticket was submitted via the Spectrum Enterprise Portal.
B2B-API	Ticket was submitted via POST /tickets client API.
B2B-Alarm	Alarm ticket was submitted via the POST /tickets client API.

About Spectrum Enterprise

Spectrum Enterprise, a part of Charter Communications, Inc., is a national provider of scalable, fiber technology solutions serving many of America's largest businesses and communications service providers. The broad Spectrum Enterprise portfolio includes [networking and managed services solutions: Internet access, Ethernet access and networks, Voice and TV solutions](#). The Spectrum Enterprise team of experts works closely with clients to achieve greater business success by providing solutions designed to meet their evolving needs. For more information, visit <https://enterprise.spectrum.com>.

©2025 Charter Communications. All rights reserved. Spectrum Enterprise is a registered trademark of Charter Communications. All other logos, marks, designs, and otherwise are the trademarks and intellectual property of their respective third-party owners. Not all products, pricing and services are available in all areas. Pricing and actual speeds may vary. Restrictions may apply. Subject to change without notice.