broadsoft

BroadSoft is
now part of Cisco.

broadsoft
broadworks

Network Server Location API

Interface Specification and Developer Guide

Release 23.0

Document Version 1

BroadSoft | 9737 Washingtonian Blvd | Suite 350 | Gaithersburg, MD 20878 | broadsoft.com

BroadWorks® Guide

Copyright Notice

Trademarks

Any product names mentioned in this document may be trademarks or registered trademarks of BroadSoft or their respective companies and are hereby acknowledged.

This document is printed in the United States of America.

Document Revision History

Release	Version	Reason for Change	Date	Author
14.0	1	Updated document for re-branding.	January 31, 2006	Roberta Boyle
14.0	1	Added general updates to document.	June 19, 2006	Michel Tougas
14.0	1	Edited document.	August 3, 2006	Patricia Renaud
15.0	1	Updated document for Releases 14.sp1 through 14.sp6 and Release 15.0.	July 11, 2008	Patricia Renaud
16.0	1	Updated document for Release 16.0.	March 30, 2009	Yves Racine
16.0	1	Edited changes and published document.	May 27, 2009	Andrea Fitzwilliam
17.0	1	Added changes introduced in Release 17.0.	March 12, 2010	Juliette D'Almeida
17.0	1	Edited changes and published document.	March 31, 2010	Andrea Fitzwilliam
18.0	1	Added changes introduced in Release 18.0.	October 5, 2011	Pierre Drapeau
18.0	1	Edited changes and published document.	October 19, 2011	Andrea Fitzwilliam
19.0	1	Adding changes for Release 19.0.	October 10, 2012	Pierre Drapeau
19.0	1	Edited changes and published document.	November 21, 2012	Patricia Renaud
20.0	1	Updated document for Release 20.0.	September 5, 2013	Pierre Drapeau
20.0	1	Edited changes and published document.	October 18, 2013	Joan Renaud
21,0	1	Updated document for Release 21.0.	September 24, 2014	Yves Racine
21.0	1	Edited changes and updated copyright notice.	September 29, 2014	Joan Renaud
21.0	1	Rebranded and published document.	December 9, 2014	Joan Renaud
22,0	1	Updated document for Release 22.0.	October 7, 2016	Eric Ross
22.0	1	Edited changes and published document.	December 12, 2016	Joan Renaud
22.0	2	Fixed parameters list in section *6.1 Location Requests* for PR-54044.	March 29, 2018	Gabriel Petrella
22.0	2	Edited changes and published document.	April 12, 2018	Jessica Boyle
23.0	1	Updated document for Release 23.0.	September 10, 2018	Yves Racine
23.0	1	Rebranded document for Cisco. Edited changes and published document.	September 30, 2018	Joan Renaud

Table of Contents

1 Summary of Changes

This section describes the changes to this document for each release and document version.

1.1 Changes for Release 23.0, Document Version 1

The document was updated from Release 22.0, version 2.

1.2 Changes for Release 22.0, Document Version 2

This version of the document includes the following change:

- Fixed parameters list in section *6.1 Location Requests* for PR-54044.

1.3 Changes for Release 22.0, Document Version 1

This version of the document includes the following changes:

- Reflected the change in configuring the Network Server using a Uniform Resource Locator (URL).
- Reflected the options for Amplify.

1.4 Changes for Release 21.0, Document Version 1

The document was updated from Release 20.0, version 1.

1.5 Changes for Release 20.0, Document Version 1

The document was updated from Release 19.0, version 1.

1.6 Changes for Release 19.0, Document Version 1

This document is updated to reflect the introduction of the Network Element (NE) maintenance partitions and its impact on the location Application Programming Interface (API). In Release 19.0, the network can be split into one or more NE maintenance partitions. A NE maintenance partition is defined as a set of zero or more Execution Servers, zero or more Profile Servers, and zero or more Xtended Services Platforms. Therefore, a NE maintenance partition is a subset of the entire network, and the location API response should only return servers listed under a specific NE maintenance partition.

1.7 Changes for Release 18.0, Document Version 1

A new parameter is added to the user location API request, *LocateUser*. It is used to return the list of servers of interest, either the provisioning servers or the call processing servers.

1.8 Changes for Release 17.0, Document Version 1

A new parameter is added to the Network Server user location API request, *LocateUser*. It is used to return (in the response) the list of valid Xtended Services Platforms corresponding to the Application Server where the user is located.

A new parameter is added to the response of the Network Server user location request *LocateUser*. The *sessionReplicationEnabled* parameter is used to indicate whether Session Data Replication is enabled on the hosting network element.

A new location API request (UpdateServerVersion) is also introduced for the Xtended Services Platform to send its version to the Network Server.

1.9 Changes for Release 16.0, Document Version 1

There were no changes to this document for Release 16.0.

1.10 Changes for Release 15.0, Document Version 1

There were no changes to this document for Release 15.0.

1.11 Changes for Releases 14.sp1 through 14.sp6, Document Version 1

There were no changes to this document for Releases 14.sp1 through 14.sp6.

2 Purpose

This document is designed to assist partners and customers who either want to replace the Network Server as the user location authority or need to perform a “lookup” to the Network Server location database. This document covers the protocol details and the format of the messages that are exchanged (Extensible Markup Language [XML] over Hypertext Transfer Protocol [HTTP]).

2.1 Background Information

The BroadWorks Xtended Services Platform farm can be used to connect to any hosting network element (NE) server at the back end.

A hosting NE server can be an Application Server in Application Server data mode or it can be an Execution Server (XS) or Profile Server (PS) in Home Subscriber Server (HSS) mode. To identify which hosting NE servers host a given end user, the Xtended Services Platform uses the Network Server location application programming interface (API).

In Amplify, the Network Location Function (NLF) is the entry point providing means to locate BroadWorks subscribers in the system. The SIP entry point into BroadWorks can be the Network Server (NS) and/or the NLF. For more information on configuring the NLF, see the *BroadWorks Execution Server Amplify Command Line Interface Administration Guide* [3].

3 General Instructions

3.1 Configure Another Server as User Location Authority

The user location is usually hosted in a web application deployed on the Network Servers. Server(s) that are meant to replace the implementation of the user location must be configured under the Xtended Services Platform command line interface (CLI), at the *XSP_CLI/System/CommunicationUtility/DefaultSettings* level. In this case, the configured Network Server Uniform Resource Locator (URL) is meant to be resolved and this represents the list of servers that is implementing the user location. The use of a URL allows support for HTTP and HTTPS as well as A/AAAA or SRV lookup.

For more information on configuring the Xtended Services Platform, see the *BroadWorks Xtended Services Platform Command Line Interface Administration Guide* [1].

4 Document Conventions

The following are the conventions used in this document to identify different types of information.

4.1 Message Format

Messages that are exchanged as part of the Location API are in XML format carried using the HTTP protocol.

Fixed data appears as it should be entered. Variable data appears in angle brackets. For example:

```
<username>
```

Optional data, which you enter to apply an option for a command, is also enclosed in square brackets. For example:

```
[<password>]
```

5 Software Upgrade Considerations

In addition to providing the Xtended Services Platform with the list of hosting NE servers where the user is located, the location API is used to coordinate software upgrades that span across multiple server types, particularly the Xtended Services Platform, the Application Server, and the Profile Server. This is necessary because during software upgrade (or patch application) the Xtended Services Platform must be able to serve web pages for multiple versions (n and n+1) of the software. Therefore, the server that implements the location API must also be able to return the version, patch level, minimum web patch level, and web impacting patches of the Application Servers/Profile Servers where the user is hosted.

6 Message Format

6.1 Location Requests

Messages are carried using the HTTP protocol. Therefore, requests are sent to the server using an encoded URL containing the URL of the user to locate.

```
http://<host>/servlet/LocateUser?
url=<user@domain>&linePort=<linePort@domain>&dn=<xxxxxxxxxx>&groupId=<myG
roupId>&returnCompatibleXSP=<true|false>&callPRequest=<true|false>&limitT
oCluster=<true|false>&requestingAddress=<xxx.xxx.xxx.xxx>&isCluster=<true
|false>&primaryFirst=<true|false>
```

Note that the textual LocateUser servlet command is used not only to get the location for a specific user, but also for a linePort, a dn, or a groupId. The user / linePort / dn / groupId is referred to as the *Identity*.

The <host> represents one of the resolved hosts of the Network Server cluster that has been provisioned using the Xtended Services Platform command line interface (CLI) under *XSP_CLI/System/CommunicationUtility/DefaultSettings*. If more than one server is specified, they are tried in order, starting with the first location server authority. An alarm is generated whenever a server on the resolved list cannot be reached.

Parameters

One of the following parameters must be set for the command to be valid. If more than one parameter is set, the one selected follows the order of the following list:

- url: The url <user@domain> represents the complete user ID (username + domain) of the user trying to log in.
- linePort: A line/port in the location server is a string formatted as "linePortAddress@linePortDomain".
- dn: The 10 digit directory number (DN).
- groupId: The group ID.

By default, a list of valid Xtended Services Platforms is returned corresponding to the hosting NE server where the *Identity* is located. The *returnCompatibleXSP* is an optional Boolean parameter and can be omitted. Unless specifically set to "false", the response message contains a block of data, *XSPServerArray*, which consists of a list of Xtended Services Platform IP addresses.

The requester must specify the list of servers of interest, which are either the provisioning servers or the call processing servers. The reason for the distinction is that the treatment of provisioning and call processing can be done on separate physical servers. A *callPRequest* optional parameter indicates whether the list of call processing servers or the list of provisioning servers is returned in the location response. The list of call processing servers can be a list of Application Server nodes or a list of Execution Server nodes. The list of provisioning servers can be a list of Application Server nodes or a list of Profile Server nodes. All servers returned are by definition sharing the same Network Element (NE) maintenance partition, since servers must be defined against a single unique NE maintenance partition.

- If *callPRequest* is "true", the list of call processing servers associated with the *Identity* specified in the request is returned in the location response.
- If *callPRequest* is "false", the list of provisioning servers associated with the *Identity* specified in the request is returned in the location response.

- If *callPRequest* is omitted from the location request, the request is treated as a provisioning request, that is, as if *callPRequest* is specified and equal to "false".

Following is information on how the list of returned servers is built for the two types of location requests for a user.

Location provisioning request

- The Network Server examines the subscriber maintenance partition of the group's *Identity* (if none exists at the group level, it takes the one at the service provider/enterprise level, and then if none exist at that level, it takes the system one). The Profile Server hosting NE nodes are all returned except those not in the NE maintenance partition of the subscriber maintenance partition which are excluded. The hosting NE associated with the group is set manually or automatically when the Network Server synchronization request is received from the Application Server or Profile Server for the group to which the *Identity* belongs. The IP address of the Application Server or Profile Server that originated the synchronization is matched against the addresses of the hosting NE node provisioned on the Network Server.
  - If a match is found, the hosting NE of the group is set to the hosting NE associated with the matching node.
  - If no match is found, the synchronization request is refused.
- If the hosting NE is in primary-secondary mode, the active server is returned first on the list of the responses and the remaining nodes of the hosting NE are next on the list. This case only occurs in the hosting model, and not in the HSS hosting model.

  If the hosting NE is in load-balancing mode, the order of the servers on the list varies between requests in a random fashion. This is an attempt to distribute the requests equally between the servers.

Location call processing request

- The response includes a list of hosting NE nodes linked to the hosting NE currently chosen by the Network Server to process calls for the enterprise/service provider/group to which the *Identity* belongs.
- The hosting NE chosen belongs to the set associated with the enterprise/service provider/group to which the *Identity* belongs.
- The first server returned on the list is the current hosting NE node chosen by the Network Server to process calls for the *Identity*.
- The other servers on the list are the remaining hosting NE nodes that belong to the chosen hosting NE. They are all returned except the nodes not in the NE maintenance partition of the chosen hosting NE node.

Note that for the hosting model composed only of the Application Server in primary-secondary mode, the location response is the same whether the request is for the list of call processing servers or provisioning servers. The *callPRequest* parameter can be omitted and set to "true" or "false". The response returned is the same in all cases.

The *requestingAddress* parameter is an optional parameter. In a HSS hosting model, the returned list of Profile Servers will be ordered and prioritized with servers sharing the same data center of the *requestingAddress's* data center, if found. Note that this parameter is normally not provided in the request as the requesting address, representing the Xtended Services Platform server making the request, is automatically computed in the Network Server from the HTTP request form.

(**Deprecated**) The *primaryFirst* is an optional Boolean parameter that can be set to "true" or "false" by the Xtended Services Platform. This parameter indicates whether the Xtended Services Platform expects the list of hosting NE servers returned by the user location authority to be ordered with the primary server first.

- When this parameter is set to "true", the primary server should be returned first in the list.
- When this parameter is set to "false", the serving hosting NE server should be returned first (that is, the primary server under normal circumstances and the secondary Application Server when the user is "migrated").

It is important to note that the *primaryFirst* parameter serves a special purpose.

- For provisioning requests, the Xtended Services Platform always tries to connect to the user's primary server. For call processing requests, the Xtended Services Platform always tries to connect to the serving hosting NE server to properly receive call notifications and other important messages.

The following parameters are also deprecated:

- *isCluster*
- *limitToCluster*

In general, a typical locationAPI request should carry no more than the following parameters:

- any one of the *url (user)*, *linePort*, *dn*, or *groupId* parameters (mandatory)
- *callPRequest* either true or false (the default is false when omitted)

6.2 Content-type Header

When returning responses to the location API client (that is, the Xtended Services Platform), the user location server must set the HTTP content-type to the following:

```
text/html; charset=ISO-8859-1
```

6.3 Location Responses

The Xtended Services Platform expects a response in the following format when no error occurs and the parameter, *returnCompatibleXSP,* set to "false", or the Xtended Services Platform is not part of a farm. In this case, the xspServerArray is not returned, as shown in the following example.

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsportal.HostingNEInfoRequest
callPRequest="false" elementStartIndex="-1" hostingNe="myhostingNe"
isCluster="" linePort="" numberOfElements="-1"
sessionReplicationEnabled=""
url="<user@domain>" or linePort="<linePortAddress@linePortDomain>" or dn
="<xxxxxxxxxx>" or groupId="<myGroupId>"

>
<applicationServerArray>
<com.broadsoft.protocols.nsportal.ApplicationServerData address="<host
address>"
 minimumPatchLevel="<minimumPatchLevel>"
 patchLevel="<patchLevel>"
 type="primary"
 version="<version>"
```

```
 webImpactingPatches="<webImpactingPatches>"

>
</com.broadsoft.protocols.nsportal.ApplicationServerData>
<com.broadsoft.protocols.nsportal.ApplicationServerData address="<host
address 2>"
 minimumPatchLevel="<minimumPatchLevel>"
 patchLevel="<patchLevel>"
 type="secondary"
 version="<version>"
 webImpactingPatches="<webImpactingPatches>"

>
</com.broadsoft.protocols.nsportal.ApplicationServerData>
</applicationServerArray>
</com.broadsoft.protocols.nsportal.HostingNEInfoRequest>
```

The Xtended Services Platform expects a response in the following format, when no error occurs and the parameter, *returnCompatibleXSP,* is set to "true" or not present in the request.

```
<com.broadsoft.protocols.nsportal.HostingNEInfoRequest
callPRequest="false" elementStartIndex="-1" hostingNe="myhostingNe"
isCluster="" linePort="" numberOfElements="-1"
sessionReplicationEnabled=""
 url="<user@domain>" or linePort="<linePortAddress@linePortDomain>" or dn
="<xxxxxxxxxx>" or groupId="<myGroupId>"
 >
<applicationServerArray>
<com.broadsoft.protocols.nsportal.ApplicationServerData address="<host
address>"
 minimumPatchLevel="<minimumPatchLevel>"
 patchLevel="<patchLevel>"
 type="primary"
 version="<version>"
 webImpactingPatches="<webImpactingPatches>"
>
</com.broadsoft.protocols.nsportal.ApplicationServerData>
<com.broadsoft.protocols.nsportal.ApplicationServerData address="<host
address 2>"
 minimumPatchLevel="<minimumPatchLevel>"
 patchLevel="<patchLevel>"
 type="secondary"
 version="<version>"
 webImpactingPatches="<webImpactingPatches>"

>
</com.broadsoft.protocols.nsportal.ApplicationServerData>
</applicationServerArray>

<xspServerArray>
<com.broadsoft.protocols.nsportal.XSPServerData address="<host address
1>"
</broadsoft.protocols.nsportal.XSPServerData>

<com.broadsoft.protocols.nsportal.XSPServerData address="<host address
1>"
</broadsoft.protocols.nsportal.XSPServerData>
</xspServerArray>

</com.broadsoft.protocols.nsportal.HostingNEInfoRequest>
```

… where:

Return Value	Description
<linePort@domain>	The linePort as passed in the request.
<xxxxxxxxxx >	The dn as passed in the request.
<myGroupId>	The groupId as passed in the request.
<user@domain>	The user URL as passed in the request.
<host address>	The IP address of the first hosting NE server host in the cluster (for example, 192.168.1.1).
<host address 2>	The IP address of the second hosting NE server host in the cluster (for example, 192.168.1.2).
<host address 1>	The IP address of an Xtended Services Platform (part of the XSPServerData).
<patchLevel>	The level at which the target hosting NE server is patched (for example, mp2).
<minimumPatchLevel>	The minimum patch level that the Xtended Services Platform must have to serve pages for the target hosting NE server (for example, [empty] or “unspecified”).
<version>	The version of the software load running on the hosting NE server (for example, AS_Rel_14.0_1.487).
<webImpactingPatches>	The web impacting patches that the Xtended Services Platform must have to serve pages for the target hosting NE server (for example, [empty] or “unspecified”).

6.4 Returning Error Messages

Whenever an error occurs during processing, the location server should return XML content in the following format.

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsportal.Error detail="<Error detail>"
 id="<id>"
 summary="<Error summary>"
>
</com.broadsoft.protocols.nsportal.Error>
```

The values that the Network Server returns are as follows.

Error ID	Error Detail	Error Summary
1	[empty]	Unspecified URL/LinePort/dn/groupId
2	[exception message]	Cannot complete request

Error ID 1 is unlikely to occur whenever an Xtended Services Platform generates requests because the user’s URL (user@domain), lineport (linePortAddress@linePortDomain), dn, or groupId is always specified. Error ID 2 occurs for exceptions that occur on the Network Server while processing the request.

Note that the location client (that is, the Xtended Services Platform) does not inspect these values although it does record them in an Xtended Services Platform log. The Xtended Services Platform treats them equally, that is, it tries to use the next Network Server on the list until the request can be fulfilled or until the list of Network Servers is exhausted (in which case the login attempt is abandoned).

When implementing an alternate location API server, it is recommended to use Error ID 2 (id=2). This sets the error summary to "Cannot complete request" and it sets the error detail to a meaningful message that appears in a Xtended Services Platform log that can then be used for problem resolution or debugging purposes.

6.5 Xtended Services Platform Software Version Notification Messages

In addition to sending requests to the location server to determine the hosting NE server hosts to contact, the Xtended Services Platform uses the location API to periodically send its version and patch-level information to the location server.

This is required because whenever an Xtended Services Platform detects that it is not able to serve web pages that are compatible with the target hosting NE server for a given user, it queries the location server to find an Xtended Services Platform host in the farm that is compatible and transparently redirects the user to that host. This ensures a smooth upgrade experience for web users.

The URL that the Xtended Services Platform posts to synchronize its software version information is in the following format.

```
http://<host>/servlet/UpdateServerVersion?address=<address>
&version=<version>&patchLevel=<patchlevel>&webPatchLevel=<webpatchlevel>&
webImpactingPatches=<webImpactingPatches>
```

… where:

Parameter	Description
<host>	The IP address of the location server.
<address>	The address of the Xtended Services Platform that is synchronizing its version information.
<patchlevel>	The level at which the Xtended Services Platform is patched (for example, mp1).
<webpatchlevel>	The web patch level of the Xtended Services Platform (for example, mp0).
<version>	The software version of the Xtended Services Platform.
<webImpactingPatches>	A list of semi-comma separated patch numbers that affect the communication between an Application Server and Xtended Services Platform.

In response to the software version synchronization notifications, the Xtended Services Platform expects a 200 HTTP_OK response with content set to the following:

```
Successfully updated <address> with version=<version>,
patchLevel=<patchlevel>, webPatchLevel=<webpatchlevel>,
webImpactingPatches=<webImpactingPatches>
```

Failure to synchronize software version information is not something that prevents the Xtended Services Platform from running or operating correctly. The software version is synchronized once every 20 minutes.

6.6 Web Server Software Version Notification Messages[1]

The API request (UpdateWebServerVersion) is used by the Web Server to send its version to the Network Server. The URL that the Web Server posts to synchronize its software version information is in the following format.

```
http://<host>/servlet/UpdateWebServerVersion?address=<hostname>&version=<version>&webImpactingPatches=<webImpactingPatches>
```

… where:

Parameter	Description
<host>	The IP address of the location server.
<hostname>	The host name of the Xtended Services Platform that is synchronizing its version information.
<version>	The software version of the Xtended Services Platform (for example, XSP_Rel_19.0_1.281).
<webImpactingPatches>	A list of semi-comma separated patch numbers that affect the communication between an Application Server and Xtended Services Platform.

In response to the software version synchronization notifications, the Web Server expects a 200 HTTP_OK response with content set to the following:

```
Successfully updated <hostname> with version=<version>
```

Failure to synchronize software version information is not something that prevents the Web Server from running or operating correctly. The software version is synchronized once every 20 minutes.

6.7 Web Server Software Version Queries[2]

Whenever a Web Server detects that it is not able to serve web pages that are compatible with the target Application Server for a given user, it queries the location server to find another Web Server host in the farm that is compatible. This is done with the following posted URL.

```
http://<host>/servlet/GetWebServer?remoteAddress=<remoteAddress>&remoteHost=<remoteHost>&version=<version>&patchLevel=<patchlevel>&webImpactingPatches=<webImpactingPatches>
```

… where:

Parameter	Description
<host>	The IP address of the location server.
<patchlevel>	The level at which the target Web Server must be patched (for example, mp1). In this case, every Web Server that is patched with at least mp1 must be returned.
<version>	The software version of the desired Web Server (for example, WS_Rel_19.0_1.281).
webImpactingPatches	A list of semi-comma separated patch numbers that affect the communication between an Application Server and Web Server.

[1] This is no longer used by the Xtended Services Platform.
[2] This is no longer used by the Xtended Services Platform.

6.8 Web Server Software Version Queries Responses[3]

The expected response is in the following format.

```
<?xml version="1.0" encoding="UTF-8"?>

<com.broadsoft.protocols.nsportal.WebServerRequest
 minimumWebPatchLevel="<requestpatchlevel>"
 version="<requestversion>"
>
<webServerArray>
<com.broadsoft.protocols.nsportal.WebServerData address="192.168.12.18"
 isBrandingMaster="false"
 patchLevel="<patchlevel>"
 type=""
 version="<version>"
 webPatchLevel="<webPatchLevel>"
 webImpactingPatches=<webImpactingPatches>
>
</com.broadsoft.protocols.nsportal.WebServerData>
<com.broadsoft.protocols.nsportal.WebServerData address="192.168.12.16"
 isBrandingMaster="true"
 patchLevel="<patchlevel>"
 type=""
 version="<version>"
 webPatchLevel="<webPatchLevel>"
 webImpactingPatches=<webImpactingPatches>
>
</com.broadsoft.protocols.nsportal.WebServerData>
<com.broadsoft.protocols.nsportal.WebServerData address="192.168.8.215"
 isBrandingMaster="false"
 patchLevel="<patchlevel>"
 type=""
 version="<version>"
 webPatchLevel="<webPatchLevel>"
 webImpactingPatches==<webImpactingPatches>
>
</com.broadsoft.protocols.nsportal.WebServerData>
</webServerArray>
</com.broadsoft.protocols.nsportal.WebServerRequest>
```

…where:

Parameter	Description
<requestpatchlevel>	The patch level appearing originally in the request (for example, mp0).
<requestversion>	The version appearing originally in the request (for example, WS_Rel_19.0_1.487).

For each entry in the webServerArray, the following information must be specified.

<patchlevel>	The level at which the Web Server is patched (for example, MP.ws.12.0.411.mp1, mp3, or [empty]).
<version>	The software version of the Web Server (for example, WS_Rel_19.0_1.487).
<webPatchLevel>	The web patch level of the Web Server (for example, mp1).
<webImpactingPatches>	A list of semi-comma separated patch numbers that affect the communication between an Application Server and Web Server.

[3] This is no longer used by the Xtended Services Platform.

The *isBrandingMaster* attribute specifies whether the Web Server is allowed to present the *BroadWorks Web Portal branding* pages to the administrators. This is because only some Web Servers in a farm are able to control and customize the web branding for all Web Servers in a farm.

The *type* attribute is not used.

When an error code must be returned, the same format is expected as specified in section *6.4 Returning Error Messages*.

References

[1] BroadSoft Inc. 2018. *BroadWorks Xtended Services Platform Command Line Interface Administration Guide, Release 23.0.* Available from BroadSoft at xchange.broadsoft.com.

[2] BroadSoft Inc. 2018. *BroadWorks Xtended Services Platform Configuration Guide, Release 23.0.* Available from BroadSoft at xchange.broadsoft.com.

[3] BroadSoft Inc. 2018. *BroadWorks Execution Server Amplify Command Line Interface Administration Guide, Release 23.0.* Available from BroadSoft at xchange.broadsoft.com.