CALLISTO

MANAGERASSISTANT GROUP

REST SERVICES MANUAL

Table Of Contents

REST services manual 3
- Overview 3
- Announcement functions 4
- File management functions 9
- Group management functions 13
- Member management functions 22
- Service functions 31
- VIP management functions 35
- Copyright Information, Disclaimer 39

Overview

Architecture

RESTRepresentational State Transfer interfaces provided by ManagerAssistant Group are in general called using HTTP GET requests to retrieve information. Functions that upload or change data are called using HTTP POST requests.

Authorization

Requests from localhost (127.0.0.1) are handled without authentication.

Requests from external sources require HTTP basic authentication. Use the credentials of a Callisto account that has the privilege REST Authentication enabled. For setting user privileges, see the chapter User administration of the Callisto platform administration manual.

Announcement functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions related to announcements are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/announcements`

addAnnouncement

HTTP request methods: POST; GET

Add an announcement to a group. Announcements are played to callers while waiting for a group member to answer the call.

Parameters

gr_id	Mandatory. The ID of the group.
an_wave	Mandatory. The file name of the wave file to use as announcement.
an_priority	The priority index of the announcement. This index determines the order in which the announcements will be played. A lower index means higher priority. If not set, the announcement will have lowest priority.

Response

The <Response> element contains two child elements:

The <id> element contains the ID of the group to which the addAnouncement action was applied.

The <status> element contains the status of the executed action. The following four statuses are possible:

OK:	The new announcement was added to the group successfully.
PARAMETER_MISSING:	One or both mandatory parameters are missing.
INVALID_GROUP_ID:	No group with the given ID exists.
ERROR:	Unknown error.

Upload the file welcome.wav to the announcements of the group with the ID 2504. Omitting the an_priority parameter will append the new announcement to the bottom of the announcements list.

Example data

gr_id=2504:	ID of the group

an_wave=Welcome.wav:	Name of the audio file

Request

```
curl --user <user>:<password>
 --data "gr_id=2504&an_wave=Wel
come.wav" http://<callisto>/Applications/inbound/<MAGroup>
/rest/announcements/addAnnouncement.asp
```

Response

```
<Callisto>
    <Request>addAnnouncements</Request>
    <Response>
        <id>2504</id>
        <status>OK</status>
    </Response>
</Callisto>
```

deleteAnnouncement

HTTP request methods: POST; GET

Delete an announcement from a group.

Parameter

an_id	Mandatory. The ID of the announcement.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The announcement was deleted successfully.
PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_ANNOUNCEMENT_ID:	No announcement with the given ID exists.
ERROR:	Unknown error.

Delete the announcement with the ID 634.

Example data

- an_id=634: ID of the announcement

Request

```
curl --user <user>:<password>
 --data "an_id=634" http://<callisto>
/Applications/inbound/<MAGroup>/rest/announcements/deleteAnnouncement.asp
```

Response

```
<Callisto>
    <Request>deleteAnnouncements</Request>
    <Response>OK</Response>
</Callisto>
```

editAnnouncement

HTTP request methods: POST; GET

Edit the file name of a present announcement.

Parameters

an_id	Mandatory. The ID of the announcement.
an_wave	Mandatory. New file name to apply.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The announcement was edited successfully.
PARAMETER_MISSING:	One or both mandatory parameters are missing.
INVALID_ANNOUNCEMENT_ID:	No announcement with the given ID exists.
ERROR:	Unknown error.

Rename the announcement with the ID 634 to *welcome.wav*

Example data

an_id=634:	ID of the announcement
an_wave=welcome.wav:	New file name to use

Request

```
curl --user <user>:<password>
 --data "an_id=634&an_wave=welco
me.wav" http://<callisto>/Applications/inbound/<MAGroup>
/rest/announcements/editAnnouncement.asp
```

Response

```
<Callisto>
    <Request>editAnnouncements</Request>
    <Response>OK</Response>
</Callisto>
```

getAnnouncements

HTTP request methods: POST; GET

Retrieve all announcements of a group.

Parameters

gr_id	Mandatory. The ID of the group.

Response

The <Response> element contains an <announcement> element for every available announcement. Each <announcement> element contains the following three child elements:

<id>:	The ID of the announcement.
<wave>:	The file name of the associated wave file.
<priority>:	The priority assigned to the announcement.

If no data can be retreived, the <Response> element contains the status of the executed action. The following three statuses are possible:

PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_GROUP_ID:	No group with the given ID exists.
ERROR:	Unknown error.

Retrieve the metadata of all announcements saved in the group with the ID 2504.

Example data

- gr_id=2504: ID of the group

Request

```
curl --user <user>:<password>
 --data "gr_id=2504" http://<callisto>
/Applications/inbound/<MAGroup>/rest/announcements/getAnnouncements.asp
```

Response

```
<Callisto>
    <Request>getAnnouncements</Request>
    <Response>
        <announcement>
            <id>19</id>
            <wave>welcome.wav</wave>
            <priority>1</priority>
        </announcement>
        <announcement>
            <id>22</id>
            <wave>music_synth.wav</wave>
            <priority>3</priority>
        </announcement>
        <announcement>
            <id>23</id>
            <wave>busy.wav</wave>
            <priority>2</priority>
        </announcement>
        <announcement>
            <id>24</id>
            <wave>music_folk.wav</wave>
            <priority>4</priority>
        </announcement>
```

```
    </Response>
</Callisto>
```

reorderAnnouncements

HTTP request methods: POST; GET

Change the order/priority of an announcement.

Parameters

an_id	Mandatory. The ID of the announcement to reorder.
move	Mandatory. The priority of the announcement is increased or decreased based on this value. Takes the values up or down.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The announcement was reordered successfully.
PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_ANNOUNCEMENT_ID:	No announcement with the given ID exists.
ERROR:	Unknown error.

Move the announcement with the ID 634 up in the announcements list.

Example data

an_id=634:	ID of the announcement
move=up:	Command to increase the announcement's priority

Request

```
curl --user <user>:<password>
 --data "an_id=634&move=up" http://<callisto>
/Applications/inbound/<MAGroup>/rest/announcements/reorderAnnouncements.asp
```

Response

```
<Callisto>
    <Request>reorderAnnouncements</Request>
    <Response>OK</Response>
</Callisto>
```

File management functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions used for file management are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/files`

getMP3s

HTTP request method: GET

Returns a list of all MP3 files that are uploaded to this application instance. These files can be used for call notifications.

Response

The <Response> element contains an <MP3File> element for every MP3 file uploaded to the application instance. The content of each element is the file name.

If an error occurs, the <Response> element contains the status of the executed action. The following four statuses are possible:

FOLDER_MISSING:	No folder named *mp3* exists for the application instance.
ERROR:	Unknown error.

Get a list of all MP3 files that are uploaded to this ManagerAssistant Group instance.

Request

```
curl --user <user>:<password> http://<callisto>
/Applications/inbound/<MAGroup>/rest/files/getMP3s.asp
```

Response

```
<Callisto>
    <Request>addAnnouncements</Request>
    <Response>
        <MP3File>Analog1.mp3</MP3File>
        <MP3File>Analog2.mp3</MP3File>
        <MP3File>AreYouThere.mp3</MP3File>
        <MP3File>Chime.mp3</MP3File>
```

```
        <MP3File>CiscoStandard.mp3</MP3File>
        <MP3File>CiscoSymphonic.mp3</MP3File>
        <MP3File>CiscoSynth.mp3</MP3File>
        <MP3File>MusicBox.mp3</MP3File>
        <MP3File>newCall.mp3</MP3File>
        <MP3File>Piano1.mp3</MP3File>
        <MP3File>Piano2.mp3</MP3File>
    </Response>
</Callisto>
```

getWAVs

HTTP request method: GET

Returns a list of all WAV files that are uploaded to this application instance. These files can be used for announcements and greetings.

Response

The <Response> element contains an <WAVFile> element for every WAVfile uploaded to the application instance. The content of each element is the file name.

If an error occurs, the <Response> element contains the status of the executed action. The following four statuses are possible:

FOLDER_MISSING:	No folder named *PCM* exists for the application instance.
ERROR:	Unknown error.

Get a list of all WAV files that are uploaded to this ManagerAssistant Group instance.

Request

```
curl --user <user>:<password> http://<callisto>
/Applications/inbound/<MAGroup>/rest/files/getWAVs.asp
```

Response

```
<Callisto>
    <Request>addAnnouncements</Request>
    <Response>
        <WAVFile>Music_01.wav</WAVFile>
        <WAVFile>Music_02.wav</WAVFile>
        <WAVFile>Music_03.wav</WAVFile>
        <WAVFile>Music_04.wav</WAVFile>
        <WAVFile>Welcome.wav</WAVFile>
        <WAVFile>Willkommen.wav</WAVFile>
    </Response>
</Callisto>
```

uploadMP3

HTTP request method: POST

Upload an MP3 file to the application instance's *mp3* folder.

Parameter

upload	Mandatory. File or array of files to upload.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The file was uploaded successfully.
FOLDER_MISSING:	No folder named *mp3* exists for the application instance.
ERROR:	Unknown error.

Upload the file *Alert.mp3* to the application instance.

Example data

- upload=@home/myfiles/Alert.mp3: Path of the file to upload

Request

```
curl --user <user>:<password> -F "
upload=@home/myfiles/Alert.mp3" http://<callisto>
/Applications/inbound/<MAGroup>/rest/files/uploadMP3.asp
```

Response

```
<Callisto>
    <Request>uploadMP3</Request>
    <Response>OK</Response>
</Callisto>
```

uploadWAV

HTTP request method: POST

Upload a WAV file to the application instance's *PCM* folder.

Parameter

upload	Mandatory. File or array of files to upload.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The file was uploaded successfully.
FOLDER_MISSING:	No folder named *PCM* exists for the application

	instance.
ERROR:	Unknown error.

Upload the file *music_piano.wav* to the application instance.

Example data

- upload=@home/myfiles/Music_Piano.wav: Path of the file to upload

Request

```
curl --user <user>:<password> -F "
upload=@home/myfiles/Music_Piano.wav" http://<callisto>
/Applications/inbound/<MAGroup>/rest/files/uploadWAV.asp
```

Response

```
<Callisto>
    <Request>uploadWAV</Request>
    <Response>OK</Response>
</Callisto>
```

Group management functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions used for group management are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/groups`

addGroup

HTTP request method: POST

Add a new group to the ManagerAssistant Group instance.

Parameters

gr_name
Mandatory. The name of the group.

gr_welcome
Name of the WAV file used as announcement when the manager is called.

Default value: empty string (i.e., no announcement)

gr_ringParallel
Boolean. Set to 1 to enable the Ring assistant parallel option or to 0 to disable it.

Default value: 0

gr_number
The group's Manager number. Dialing this number directs a call to the group manager.

Default value: empty string (i.e., no manager number)

gr_managerTimeout
Integer. Sets the Manager Timeout duration in seconds. Set to 0 to disable manager timeout.

Default value: 0

gr_assistantTimeout
Integer. Sets the Assistant Timeout duration in seconds. Set to 0 to disable assistant timeout.

Default value: 0

gr_overflow
Sets the Overflow action. The following values are available:

-1: No overflow. The call is redirected again to the first available assistant or manager in the current

	group. -3 or busy: The caller recieves a busy signal and the call is terminated. voicemail:<*box number*>: The caller can leave a voicemail message to the number set as <*voicemail box number*>. <*phone number*>: The call is redirected to the number set as <*phone number*>. **Default value:** -1
gr_showVIPcall	Sets the Show VIP calls option. The following values are available: 0 or manager: The call will only be displayed on the manager's phone and be hidden from all other group members. 1 or all: Like a regular call, the call will be shown on all group members' phone screens. 2 or marked: The call will be shown on all group members' phone screens, and the word <*VIP*> will be appended to the caller's name or number. **Default value:** 1
gr_VIPCallsRouting	Sets the VIP call routing option. The following values are available: 0: Unanswered VIP calls are redirected to all assistants consecutively, regardless of their presence state. 1: Unanswered VIP calls are directed only to the manager; no redirection to assistants occurs, and parallel calling is disabled. 2: Unanswered VIP calls are directed to the manager and to assistants whose presence state is set to *available*. **Default value:**0
gr_assistantToMng	Boolean. Set to 1 to enable the Connect assistants directly with the manager option or to 0 to disable it. **Default value:** 0
gr_showCalls	Boolean. Set to 1 to enable the Show call information option or to 0 to disable it. **Default value:** 1
gr_VIPThreshold	Set the VIP threshold. A number between 1 and 5 will set the corresponding VIP level as threshold; setting

	the value to 0 will mark all callers as VIP members, setting it to a number greater than 5 will disable the threshold (i.e., only contacts set in the *VIP List* tab will be considered VIP callers). **Default value:** 5
gr_callerName	Set the Caller name identification. Default value: empty string (i.e., caller name will be displayed as received by the UCM)
gr_callerNumber	Set the Caller number identification. Default value: empty string (i.e., caller number will be displayed as received by the UCM)
gr_mngCallAlarm	Sets the Manager call notification tone on System Phones.
gr_assistCallAlarm	Sets the Assistant call notification tone on System Phones. In groups without manager, this option sets the Call notification tone.
gr_mngCallAlarmJ	Sets the Manager call notification tone on Jabber, COC, and the standalone gadget.
gr_assistCallAlarmJ	Sets the Assistant call notification tone on Jabber, COC, and the standalone gadget. In groups without manager, this option sets the Call notification tone Jabber.

Response

The <Response> element contains two child elements:

The <id> element contains the ID of the newly created group.

The <status> element contains the status of the executed action. The following four statuses are possible:

OK:	The new group was created successfully.
PARAMETER_MISSING:	One or multiple mandatory parameters are missing.
ALREADY_EXISTS:	A group with this name already exists.
ERROR:	Unknown error.

Create a new group.

Example data

gr_name=groupName	The group will be named “Accounting”.
gr_welcome=Welcome.wav	The file *Welcome.wav* will be used as greeting announcement.
gr_ringParallel=1	The Ring assistant parallel option will be enabled.
gr_number=5555	The group’s manager number will be 5555.
gr_managerTimeout=10	The Manager Timeout will be set to 10 seconds.
gr_assistantTimeout=10	The Assistant Timeout will be set to 10 seconds.
gr_overflow=4321	In case of Overflow, the call will be directed to the number 4321.
gr_showVIPCall=2	VIP calls will be shown on all group members’ phone screens.
gr_assistantToMng=1	The Connect assistant directly with the manager option will be enabled.

gr_showCalls=0	The Show call information option will be disabled.
gr_mngCallAlarm=Harp.wav	The file *harp.wav* will be played to all group members' phones when the manager receives a call.
gr_assistCallAlarm=analog.wav	The file *analog.wav* will be played to all group members' phones when an assistant receives a call.
gr_VIPThreshold=3	Calls from contacts with two or more stars will be treated as VIP calls.
gr_mngCallAlarmJ=Analog1.mp3	The file *Analog1.mp3* will be played to all group members' Jabber/COC/gadget when the manager receives a call.
gr_assistCallAlarmJ=Analog2.mp3	The file *Analog2.mp3* will be played to all group members' Jabber/COC/gadget when an assistant receives a call.

The following two values contain special characters, so they are submitted using percent-encoding:

gr_callerName=%managerShortName%**%callerName%	The mask for the caller name identification.
gr_callerNumber=%managerNumber%**%callerNumber%	The mask for the caller number identification.

Request

```
curl --user <user>:<password>
 --data "gr_name=groupName&gr_welcome=Welcome.wav&gr_ringParallel=1&gr_number=5555&gr_managerTimeout=10&gr_assistantTimeout=10&gr_overflow=4321&gr_showVIPCall=2&gr_assistantToMng=1&gr_showCalls=0&gr_mngCallAlarm=Harp.wav&gr_assistCallAlarm=Analog.wav&gr_VIPThreshold=3&gr_mngCallAlarmJ=Analog1.mp3&gr_assistCallAlarmJ=Analog2.mp3" --data-urlencode "gr_callerName=%managerShortName%**%callerName%" -
-data-
urlencode "g
r_callerNumber=%managerN
umber%**%callerNumber%" http://<callisto>
/Applications/inbound/<MAGroup>/rest/groups/addGroup.asp
```

Response

```
<Callisto>
    <Request>addGroup</Request>
    <Response>
        <id>45</id>
        <status>OK</status>
    </Response>
</Callisto>
```

deleteGroup

HTTP request methods: POST; GET

Delete a group from the ManagerAssistant Group instance.

Parameters

gr_id	Mandatory. The ID of the group to delete.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The group was deleted successfully.
PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_GROUP_ID:	A group with this ID doesn't exist.
ERROR:	Unknown error.

Delete the group with the ID 45.

Example data

- gr_id=45: The ID of the group to delete.

Request

```
curl --user <user>:<password>
 --data "gr_id=45" http://<callisto>
/Applications/inbound/<MAGroup>/rest/groups/deleteGroup.asp
```

Response

```
<Callisto>
    <Request>deleteGroup</Request>
    <Response>
        <id>45</id>
        <status>OK</status>
    </Response>
</Callisto>
```

editGroup

HTTP request method: POST

Edit any parameter of a group.

Parameters

gr_name	Mandatory. The name of the group.
gr_welcome	Name of the WAV file used as announcement when the manager is called. **Default value:** empty string (i.e., no announcement)
gr_ringParallel	Boolean. Set to 1 to enable the Ring assistant parallel option or to 0 to disable it. **Default value:** 0
gr_number	The group's Manager number. Dialing this number directs a call to the group manager. **Default value:** empty string (i.e., no manager number)

gr_managerTimeout	Integer. Sets the Manager Timeout duration in seconds. Set to 0 to disable manager timeout. **Default value:** 0
gr_assistantTimeout	Integer. Sets the Assistant Timeout duration in seconds. Set to 0 to disable assistant timeout. **Default value:** 0
gr_overflow	Sets the Overflow action. The following values are available: -1: No overflow. The call is redirected again to the first available assistant or manager in the current group. -3 or busy: The caller recieves a busy signal and the call is terminated. voicemail:<*box number*>: The caller can leave a voicemail message to the number set as <*voicemail box number*>. <*phone number*>: The call is redirected to the number set as <*phone number*>. **Default value:** -1
gr_showVIPcall	Sets the Show VIP calls option. The following values are available: 0 or manager: The call will only be displayed on the manager's phone and be hidden from all other group members. 1 or all: Like a regular call, the call will be shown on all group members' phone screens. 2 or marked: The call will be shown on all group members' phone screens, and the word <*VIP*> will be appended to the caller's name or number. **Default value:** 1
gr_VIPCallsRouting	Sets the VIP call routing option. The following values are available: 0: Unanswered VIP calls are redirected to all assistants consecutively, regardless of their presence state. 1: Unanswered VIP calls are directed only to the manager; no redirection to assistants occurs, and parallel calling is disabled. 2: Unanswered VIP calls are

	directed to the manager and to assistants whose presence state is set to *available*. **Default value:** 0
gr_assistantToMng	Boolean. Set to 1 to enable the Connect assistants directly with the manager option or to 0 to disable it. **Default value:** 0
gr_showCalls	Boolean. Set to 1 to enable the Show call information option or to 0 to disable it. **Default value:** 1
gr_VIPThreshold	Set the VIP threshold. A number between 1 and 5 will set the corresponding VIP level as threshold; setting the value to 0 will mark all callers as VIP members, setting it to a number greater than 5 will disable the threshold (i.e., only contacts set in the *VIP List* tab will be considered VIP callers). **Default value:** 5
gr_callerName	Set the Caller name identification. Default value: empty string (i.e., caller name will be displayed as received by the UCM)
gr_callerNumber	Set the Caller number identification. Default value: empty string (i.e., caller number will be displayed as received by the UCM)
gr_mngCallAlarm	Sets the Manager call notification tone on System Phones.
gr_assistCallAlarm	Sets the Assistant call notification tone on System Phones. In groups without manager, this option sets the Call notification tone.
gr_mngCallAlarmJ	Sets the Manager call notification tone on Jabber, COC, and the standalone gadget.
gr_assistCallAlarmJ	Sets the Assistant call notification tone on Jabber, COC, and the standalone gadget. In groups without manager, this option sets the Call notification tone Jabber.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The group was edited successfully.
PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_GROUP_ID:	A group with this ID doesn't exist.
ERROR:	Unknown error.

Enable the Show call information option in the group with the ID 45.

Example data

gr_id=45:	The ID of the group to edit.
gr_showCalls=1:	The new parameter to set. In this case, enable Show call information.

Request

```
curl --user <user>:<password>
 --data "gr_id=45&gr_showCalls=1" http
://<callisto>/Applications/inbound/<MAGroup>/rest/groups/editGroup.asp
```

Response

```
<Callisto>
    <Request>editGroup</Request>
    <Response>
        <id>45</id>
        <status>OK</status>
    </Response>
</Callisto>
```

getGroups

HTTP request methods: POST; GET

Return a list of all groups, filtered by name.

Parameters

gr_name	Only groups whose names contain this string are returned. **Default value:** empty string (i.e., all groups will be returned)
gr_id	Return the group with this exact ID.

Response

The <Response> element contains a <group> element for every group that matches the criteria. Each group element contains an <id> element as well as an element for every available parameter that can be set using addGroup and editGroup.

If an error occurs, the <Response> element will contain the text ERROR.

Retrieve information on all groups containing the word “support”.

Example data

- gr_name=support: Only groups with a name containing “support” are displayed.

Request

```
curl --user <user>:<password>
 --data "gr_name=support" http://<callisto>
/Applications/inbound/<MAGroup>/rest/groups/getGroups.asp
```

Response

```
<Callisto>
    <Request>getGroups</Request>
    <Response>
        <group>
            <id>26</id>
            <name>1st level support</name>
            <welcome>Welcome.wav</welcome>
            <ringParallel>0</ringParallel>
            <number>5544</number>
            <managerTimeout>10</managerTimeout>
            <assistantTimeout>10</assistantTimeout>
            <overflow>5555</overflow>
            <showVIPCall>1</showVIPCall>
            <VIPCallsRouting>0</VIPCallsRouting>
            <assistantToMng>0</assistantToMng>
            <showCalls>1</showCalls>
            <callerName>%managerShortName%**%callerName%</callerName>
            <callerNumber>%managerNumber%**%callerNumber%</callerNumber>
            <mngCallAlarm>Analog1.raw</mngCallAlarm>
            <assistCallAlarm>Analog2.raw</assistCallAlarm>
            <VIPThreshold>5</VIPThreshold>
            <mngCallAlarmJ>Analog1.mp3</mngCallAlarmJ>
            <assistCallAlarmJ>Analog2.mp3</assistCallAlarmJ>
        </group>
        <group>
            <id>29</id>
            <name>2nd level support</name>
            …
        </group>
        <group>
            <id>71</id>
            <name>Internal support</name>
            …
        </group>
    </Response>
</Callisto>
```

Member management functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions used for member management are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/members`

addMember

HTTP request methods: POST; GET

Add a member to a group.

Parameters

gr_id	Mandatory. The ID of the group to which the member is added.
usr_Name	Mandatory. The Callisto user name to associate with the new member.
mb_number	Mandatory. The member's phone number.
gm_shortName	The member's short name. If omitted, a short name will be generated automatically.
gm_type	Boolean. Set the type of the member. • 1: manager • 0: regular member / assistant **Default value:** 0
displayedGroup	Set which group to show on the member's phone status display. <*group id*>: The ID of the group to display. 0: Hide the status display. -1: Leave the current display settings unaltered. **Default value:** 0

Response

The <Response> element contains two child elements:

The <id> element contains the ID of the newly added member.

The <status> element contains the status of the executed action. The following four statuses are possible:

OK:	The new member was added to the group successfully.
PARAMETER_MISSING:	One or more mandatory parameters are missing.
INVALID_GROUP_ID:	No group with the given ID exists.
INVALID_USERNAME:	No Callisto user with the given name exists.
MANAGER_ALREADY_EXISTS:	If a member in the group is already assigned as manager, and the new member gm_type is set to 1, this error will be thrown as a group cannot have more than one manager.
ERROR:	Unknown error.

Add the user Maria.Rossi to the group with the ID 45.

Example data

gr_id=45:	ID of the group.
usr_Name=Maria.Rossi:	Name of the member (as set in Callisto).
mb_number=1071:	Phone number of the member.
gm_shortName=Mar:	Short name for the member.
gm_type=1:	The member will be assigned the role of the group manager.

Request

```
curl --user <user>:<password>
 --data "gr_id=45&usr_Name=Maria.Rossi&mb_number=1071&gm_shortName=Mar&gm_type=
1" http://<callisto>/Applications/inbound/<MAGroup>/rest/members/addMember.asp
```

Response

```
<Callisto>
    <Request>addMember</Request>
    <Response>
        <id>221</id>
        <status>OK</status>
    </Response>
</Callisto>
```

deleteMember

HTTP request methods: POST; GET

Add a member to a group.

Parameters

mb_id	Mandatory if usr_Name is omitted. The ID of the member to delete.
usr_Name	Mandatory if mb_id is omitted. The Callisto user name associated with the member to delete. A user can be associated with multiple members (i.e. when a user has multiple phones).

	If both mb_id and usr_Name are provided, mb_id has priority.
gr_id	The ID of the group from which the member is to be removed. **Default value:** empty string (i.e., the member is deleted from all groups)

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The member was deleted successfully.
PARAMETER_MISSING:	One mandatory parameter is missing.
INVALID_MEMBER_ID:	No member with the given ID exists.
ERROR:	Unknown error.

Add the user Maria.Rossi to the group with the ID 45.

Example data

mb_id=221:	ID of the member.
gr_id=45:	ID of the group.

Request

```
curl --user <user>:<password>
 --data "mb_id=221&gr_id=45" http://<callisto>
/Applications/inbound/<MAGroup>/rest/members/deleteMember.asp
```

Response

```
<Callisto>
    <Request>deleteMember</Request>
    <Response>OK</Response>
</Callisto>
```

editMember

HTTP request methods: POST; GET

Edit the parameters of a group member.

Parameters

mb_id	Mandatory if usr_Name is omitted. The ID of the member to delete.
usr_Name	Mandatory if mb_id is omitted. The Callisto user name associated with the member to delete. A user can be associated with multiple members (i.e. when a user has multiple phones). If both mb_id and usr_Name are provided,

	only mb_id is used.
gr_id	ID of the group in which the member' parameters are changed. **Default value:** empty string (i.e., parameters will be changed for every instance where the member is present)
mb_number	The member's phone number. If usr_Name is used to determine the user, only change the mb_number if the user is associated with one single mb_id. If the user is associated with multiple members, changing the mb_number can cause unpredictable results.
gm_shortName	The member's short name.
displayedGroup	Set which group to show on the member's phone status display. <*group id*>: The ID of the group to display. 0: Hide the status display. -1: Leave the current display settings unaltered. **Default value:** 0

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The member's data was edited successfully.
PARAMETER_MISSING:	One mandatory parameter is missing.
INVALID_MEMBER_ID:	No member with the given ID exists.
INVALID_USERNAME:	No Callisto user with the given name exists.
ERROR:	Unknown error.

In the group with the ID 45, change the short name of the member with the ID 221 to Ros.

Example data

mb_id=221:	ID of the member.
gr_id=45:	ID of the group.
gm_shortName=Ros:	Short name for the member.

Request

```
curl --user <user>:<password>
 --data "mb_id=221&gr_id=4
5&gm_shortName=Ros" http://<callisto>
/Applications/inbound/<MAGroup>/rest/members/editMember.asp
```

Response

```
<Callisto>
    <Request>editMember</Request>
    <Response>OK</Response>
</Callisto>
```

getMembers

HTTP request methods: POST; GET

Return all members in the application instance. Members can be filtered by first name, last name, phone number, short name or group in which they are. Filters by logical conjunction (i.e., only results that contain a match for *all* filters will be shown).

Parameters

mb_id	Find a member by ID.
usr_Name	Filter members by the associated Callisto user name.
gm_type	Boolean. Filter by type of the member. • 1: manager • 0: regular member/assistant
firstName	Filter by first name.
lastName	Filter by last name.
mb_number	Filter by phone number.
gm_shortName	Filter by short name.
gr_id	Show only members of the group with this ID.

Response

The <Response> element contains a <member> element for every member that matches the criteria. Each <member> element contains the parameters that can be set using addMember and editMember. Additional elements are:

<id>:	The member's ID.
<index>:	The position of the member inside the group member list.
<userState>:	The current presence status of the member.
<forwardDestination>:	The current forwarding destination of the member.
<statusMessage>:	The current status message of the member.
<lineState>:	The current phone status of the member.

If an error occurs, the <Response> element will contain the text ERROR.

Get all assistants from the group with the ID 45.

Example data

gm_type=0:	Filter by members whose type is *assistant*.
gr_id=45:	Limit results to members of the group with the ID 45.

Request

```
curl --user <user>:<password>
 --data "gm_type=0&gr_id=45" http://<callisto>
/Applications/inbound/<MAGroup>/rest/members/getMembers.asp
```

Response

```
<Callisto>
    <Request>getMembers</Request>
```

```
    <Response>
        <member>
            <id>198</id>
            <firstName>Rafael</firstName>
            <lastName>Hidalgo</lastName>
            <number>1175</number>
            <type>0</type>
            <grId>45</grId>
            <userName>Rafael.Hidalgo</userName>
            <index>1</index>
            <shortName>Raf</shortName>
            <displayedGroup>45</displayedGroup>
            <userState>0</userState>
            <forwardDestination></forwardDestination>
            <statusMessage></statusMessage>
            <lineState>0</lineState>
        </member>
        <member>
            <id>102</id>
            <firstName>Aradhana</firstName>
            <lastName>Jaggi</lastName>
            …
        </member>
        <member>
            <id>103</id>
            <firstName>Léonard</firstName>
            <lastName>Saligny</lastName>
            …
        </member>
        …
    </Response>
</Callisto>
```

orderByUserList

HTTP request methods: POST; GET

Reorder a group's member list by providing an ordered list.

Parameters

gr_id	Mandatory. Group whose member list is to be reordered.
usr_list	Mandatory. A comma-separated list of all users in the desired order. Use the username as set in Callisto. All members' usernames must be present.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The member list was reordered successfully.
PARAMETER_MISSING:	One ore more mandatory parameters are missing.
INVALID_PARAMETERS:	Either no group with the given ID exists, or the

	provided list doesn't contain all group members.
ERROR:	Unknown error.

Reorder the member list of the group with the ID 64, which has four members.

Example data

gr_id=64: usr_list=maria.santos, chiyo.fujimoto,romeo.girard, frederique.voclain:	Apply the member list reordering to this group. Every member in the group, set in the desired order.

Request

```
curl --user <user>:<password>
 --data "gr_id=64&usr_list=maria.santos,chiyo.fujimoto,romeo.girard,frederique.
voclain" http://<callisto>/Applications/inbound/<MAGroup>
/rest/members/orderByUserList.asp
```

Response

```
<Callisto>
    <Request>orderByUserList</Request>
    <Response>OK</Response>
</Callisto>
```

reorderMembers

HTTP request methods: POST; GET

Move a single member up or down in a group's member list.

Parameters

mb_id	Mandatory if usr_Name is omitted. The ID of the member to reorder.
usr_Name	Mandatory if mb_id is omitted. The Callisto user name associated with the member to reorder. A user can be associated with multiple members (i.e. when a user has multiple phones). In this case, the action will only be applied to the first matching member. If both mb_id and usr_Name are provided, mb_id has priority.
gr_id	Mandatory. Group whose member list is to be reordered.
move	Mandatory. Possible values are up or down. Set whether the user is to be moved up or down in the member list.

Response

The <Response> element contains the status of the executed action. The following four statuses are

possible:

OK:	The member's order in the member list was reordered successfully.
PARAMETER_MISSING:	One ore more mandatory parameters are missing.
INVALID_PARAMETERS:	Either no group with the given ID exists, or the provided list doesn't contain all group members.
ERROR:	Unknown error.

In the group with the ID 64, move the member chiyo.fujimoto up one position in the member list.

Before executing reorderMembers

1. maria.santos
2. chiyo.fujimoto
3. romeo.girard
4. frederiquie.voclain

After executing reorderMembers

1. chiyo.fujimoto
2. maria.santos
3. romeo.girard
4. frederiquie.voclain

Example data

mb_id=279:	The ID of chiyo.fujimoto.
gr_id=64:	The ID of the group to apply the action.
move=up:	Move chiyo.fujimoto up in the member list.

Request

```
curl --user <user>:<password>
 --data "mb_id=279&gr_id=64&move=up
" http://<callisto>/Applications/inbound/<MAGroup>
/rest/members/reorderMembers.asp
```

Response

```
<Callisto>
    <Request>reorderMembers</Request>
    <Response>OK</Response>
</Callisto>
```

setPresence

HTTP request methods: POST; GET

Set a single member's presence status and status message.

Parameters

mb_id	Mandatory if usr_Name is omitted. The ID of the member whose presence status to set.
usr_Name	Mandatory if mb_id is omitted. The Callisto user name associated with the member whose presence status to set. A user can be associated with multiple members (i.e. when a user has multiple phones). If both mb_id and usr_Name are provided, mb_id has

	priority.
presenceState	Mandatory. The member's new presence state. The following values are accepted: • 0: available • 1: away • 3: busy • 4: DND • 6: offline
presenceMessage	Set a custom status message. Use percent-encoding if the message contains special characters.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The member's presence status was updated successfully.
PARAMETER_MISSING:	One ore more mandatory parameters are missing.
INVALID_PARAMETERS:	Either no group with the given ID exists, or the provided list doesn't contain all group members.
INVALID_STATE:	The provided presence status value isn't valid.
INVALID_MEMBER_ID:	No member with the given ID exists.
ERROR:	Unknown error.

Change the presence status of the member with the ID 279 to *away* and set the member's status message to "Out for lunch."

Example data

mb_id=279	The ID of the member whose status to change.
presenceState=1	The presence status will be set to *away*.
presenceMessage=Out%20for%20lunch	The new status message (percent-encoded).

Request

```
curl --user <user>:<password> --data "mb_id=279&presenceState=1
&presenceMessage=Out%20for%20lunch"
http://<callisto>/Applications/inbound/<MAGroup>/rest/members/setPresence.asp
```

Response

```
<Callisto>
    <Request>setPresence<</Request>
    <Response>OK</Response>
</Callisto>
```

Service functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions related to phone services and service integration are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/services`

getIntegrationURLs

HTTP request methods: POST; GET

Retrieves the integration URLs. With these URLs, the ManagerAssistant Group instance can be integrated in Jabber, COC, or as a standalone gadget in any modern web browser.

Parameter

login

The login method to use for accessing the services. The following values are available:

manual:	The user will be prompted to log in manually when accessing the URL. Recommended for terminals used by multiple users.
auto:	Provides URLs containing two xxx placeholders that need to be replaced with a user's username and password. Accessing this URL will perform automatic login using the provided credentials. Recommended for Jabber integration.
coc:	To be used with the Callisto Operator Console. Will perform automatig login based in the credentials used in the COC application.

If this parameter is omitted, all available URLs are returned.

Response

If the parameter login was set, the <Response> element contains the requested URL in an element labeled <IntegrationURL>.

If the parameter login was not set, the <Response> element contains an element labeled <IntegrationURLs> which contains three elements labeled <manual>, <auto>, and <coc>, each containing the respective integration URL.

Retrieve the integration URL using manual login.

Example data

- login=manual: Retrieve the URL using manual login.

Request

```
curl --user <user>:<password>
 --data "login=manual" http://<callisto>
/Applications/inbound/<MAGroup>/rest/services/getIntegrationURLs.asp
```

Response

```
<Callisto>
    <Request>getIntegrationURLs</Request>
    <Response>
        <IntegrationURL>http://<callisto>
/Applications/inbound/<MAGroup>/src/Jabber.asp</IntegrationURL>
    </Response>
</Callisto>
```

Retrieve all integration URLs.

Request

```
curl --user <user>:<password> http://<callisto>
/Applications/inbound/<MAGroup>/rest/services/getIntegrationURLs.asp
```

Response

```
<Callisto>
    <Request>getIntegrationURLs</Request>
    <Response>
        <IntegrationURLs>
            <manual>http://<callisto>
/Applications/inbound/<MAGroup>/src/Jabber.asp</manual>
            <auto>http://<callisto>
/LogonPage.asp?page=/Applications/inb
ound/<MAGroup>/src/Jabber.asp&amp;user=xxx&amp;pwd=xxx</auto>
            <coc>http://<callisto>
/LogonPage.asp?page=/Applications/inb
ound/<MAGroup>/src/Jabber.asp&amp;user=%user%&amp;pwd=%pwd%</coc>
```

```
        </IntegrationURLs>
    </Response>
</Callisto>
```

getPhoneServiceURLs

HTTP request methods: POST; GET

Retrieves URLs used for CUCM phone services.

Parameter

service	The name of the service whose URL to retrieve. The following values are available: MainMenu: URL to access the phone main menu. 1ButtonService: URL to access the 1 Button Service. TransferCall: URL to access the transfer call service. PickupCalls: URL to access the pickup calls service. DirectCall: URL to access the direct call service. ChangeGroup: URL to access the change group service. ChangeStatus: URL to access the change status service. TeamStatus: URL to access the team status service. SetForwarding: URL to access the set forwarding service. If this parameter is omitted, all available URLs are returned.

Response

The <Response> element contains the requested URLs in respectively labeled elements.

Retrieve the integration URL for the 1 Button Service.

Example data

- service=1ButtonService: Retrieve the URL for the 1 Button Service.

Request

```
curl --user <user>:<password>
 --data "service=1ButtonService" http:/
/<callisto>/Applications/inbound/<MAGroup>
/rest/services/getPhoneServiceURLs.asp
```

Response

```
<Callisto>
    <Request>services</Request>
    <Response>
        <1ButtonService>http://<callisto>
/Applications/inbound/<MAGroup>
/src/OneButton.asp?device=#DEVICENAME#</1ButtonService>
    </Response>
</Callisto>
```

Retrieve the integration URLs for all phone services.

Request

```
curl --user <user>:<password> http://<callisto>
/Applications/inbound/<MAGroup>/rest/services/getPhoneServiceURLs.asp
```

Response

```
<Callisto>
    <Request>getIntegrationURLs</Request>
    <Response>
        <MainMenu>http://<callisto>/Applications/inbound/<MAGroup>
/src/PhoneMenu.asp?device=#DEVICENAME#</MainMenu>
        <1ButtonService>http://<callisto>
/Applications/inbound/<MAGroup>
/src/OneButton.asp?device=#DEVICENAME#</1ButtonService>
        <TransferCall>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=transferCall&device=#DEVICENAME#</TransferC
all>
        <PickupCall>http://<callisto>
/Applications/inbound/<MAGroup>
/src/src/IndependentServices.asp?action=pickupCalls&device=#DEVICENAME#</Pickup
Call>
        <DirectCall>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=directCall&device=#DEVICENAME#</DirectCall>
        <ChangeGroup>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=changeGroup&device=#DEVICENAME#</ChangeGrou
p>
        <ChangeStatus>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=changeStatus&device=#DEVICENAME#</ChangeSta
tus>
        <TeamStatus>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=teamStatus&device=#DEVICENAME#</TeamStatus>
        <SetForwarding>http://<callisto>
/Applications/inbound/<MAGroup>
/src/IndependentServices.asp?action=setForwarding&device=#DEVICENAME#</SetForwa
rding>
    </Response>
</Callisto>
```

VIP management functions

All examples shown here use the following placeholders. Replace these placeholders with the appropriate data:

- <*user*> and <*password*>: Credentials of a Callisto user with REST Authentication privilege.
- <*callisto*>: The base domain name or IP address of the Callisto server.
- <*MAGroup*>: The name of the ManagerAssistant Group application instance to call.

All functions related to VIP contact management and service integration are called by URLs containing the following path:

`http://<callisto>/Applications/inbound/<MAGroup>/rest/vip`

addVIPNumber

HTTP request methods: POST; GET

Add a VIP contact to a manager's VIP list.

Parameters

vn_number:	Mandatory. The number of the VIP contact to add.
vn_description:	A description for the VIP contact.
mb_id:	Mandatory. The ID of the manager to whose list the VIP contact is added.

Response

The <Response> element contains two child elements:

The <id> element contains the ID of the newly added VIP contact.

The <status> element contains the status of the executed action. The following four statuses are possible:

OK:	The VIP contact was added successfully.
PARAMETER_MISSING:	One or more mandatory parameters are missing.
ERROR:	Unknown error.

Add the VIP number 3132 to the VIP list of a manager.

Example data

vn_number=3132:	The number to add.
mb_id=27:	Add the number to this manager's VIP list.
vn_description=Project leader:	A description for the new VIP contact.

Request

```
curl --user <user>:<password> --data "vn_number=3132&mb_id=27" --data-
urlencode"vn_description=Projec
```

```
t leader" http://<callisto>/Applications/inbound/<MAGroup>
/rest/vip/addVIPNumber.asp
```

Response

```
<Callisto>
    <Request>getIntegrationURLs</Request>
    <Response>
        <id>716</id>
        <status>OK</status>
    </Response>
</Callisto>
```

deleteVIPNumber

HTTP request methods: POST; GET

Delete a VIP contact from the application instance.

Parameter

vn_id	Mandatory. The ID of the VIP contact to delete.

Response

The <Response> element contains the status of the executed action. The following four statuses are possible:

OK:	The VIP contact was added successfully.
PARAMETER_MISSING:	The mandatory parameter is missing.
INVALID_VIP_NUMBER_ID:	No VIP contact with the given ID exists.
ERROR:	Unknown error.

Delete the VIP contact with the ID 716.

Example data

- vn_id=716: Delete the VIP contact with this ID.

Request

```
curl --user <user>:<password>
 --data "vn_id=716" http://<callisto>
/Applications/inbound/<MAGroup>/rest/vip/deleteVIPNumber.asp
```

Response

```
<Callisto>
    <Request>deleteVIPNumber</Request>
    <Response>OK</Response>
</Callisto>
```

getVIPNumbers

HTTP request methods: POST; GET

Retrieve VIP contact information from this application instance. The returned contacts can be limited to a number and/or a specific manager's VIP list.

Parameters

mb_id	Limit results to the VIP list of the manager with this ID.
vn_number	Limit results to VIP contacts with this VIP number.

Response

The <Response> element contains a <VIPNumber> element for every VIP contact that matches the criteria. Each <VIPNumber> element contains the folowing data:

<id>:	The VIP contact's ID.
<number>:	The number of the VIP contact.
<description>:	The description of the contact.
<mangId>:	The ID of the manager in whose list this VIP contact is saved.
<FirstName>:	The VIP contact's first name.
<LastName>:	The VIP contact's last name.

If an error occurs, the <Response> element will contain the text ERROR.

Show all VIP contacts in the list of the manager with the ID 72.

Example data

- mb_id=72: Limit results to this manager's VIP list.

Request

```
curl --user <user>:<password>
 --data "mb_id=72" http://<callisto>
/Applications/inbound/<MAGroup>/rest/vip/getVIPNumbers.asp
```

Response

```
<Callisto>
    <Request>getVIPNumbers</Request>
    <Response>
        <VIPNumber>
            <id>253</id>
            <number>1077</number>
            <description>Head of Accounting</description>
            <mngId>72</mngId>
            <FirstName>Gabriela</FirstName>
            <LastName>Kronberger</LastName>
        </member>
        <member>
            <id>260</id>
```

```
            …
        </member>
        <member>
            <id>288</id>
            …
        </member>
        …
    </Response>
</Callisto>
```

Copyright

Copyright © 2026 CTModule AG; All Rights Reserved

This document contains proprietary information of CTModule AG. No part of the work described herein may be reproduced. Reverse engineering of the hardware or software is prohibited and is protected by patent law.

This material or any portion of it may not be copied in any form or by any means, stored in a retrieval system, adopted or transmitted in any form or by any means (electronic, mechanical, photographic, graphic, optic or otherwise), or translated to any language or computer language without the prior written permission of CTModule AG.

The information in this document is subject to change without notice. CTModule AG makes no representation or warranties with respect to the contents herein and shall not be responsible for any loss or damage caused to users either by direct or indirect use of this information. This document may contain information about third party products or processes. This third party information is out of the sphere of influence of CTModule AG. Therefore CTModule AG shall not be responsible for the correctness or legitimacy of this information.

While due care has been taken to deliver accurate documentation, CTModule AG does not warrant that this document is error-free. If you find any errors, inconsistencies, omissions or other problems related to this document, please report this in writing by email to box@ctmodule.com at CTModule AG.

CTMaker, VAS, Callisto, Callisto Express, Callisto ISDN, Callisto One, Callisto UCM, Callisto Cruise, Callisto Hospitality, Callisto X Mobile, COC Express , COC UCM and CTModule AG are trademarks and the CTModule logo is a service mark of CTModule AG.

All other products or company names mentioned herein are used for identification purposes only, and may be trademarks or registered trademarks of their respective owners.

The following description of software, hardware or process of CTModule AG or other third party provider may be included with your product and will be subject to the software, hardware or other license agreement.

Disclaimer

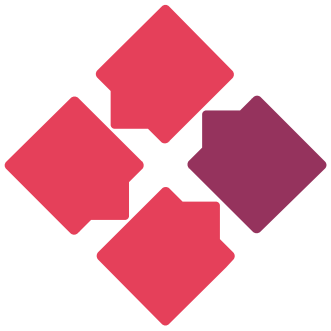
Changes are possible without notice.

The General Terms and Conditions (GTC) of CTModule AG, Switzerland apply.

All rights reserved.

For more information about CTModule AG, visit the CTModule web site at

www.ctmodule.com



CTMODULE

COMMUNICATION TECHNOLOGY MODULES

CTMODULE AG

Lehnweg 1

CH-3123 Belp/Berne

Switzerland

T: +41 (0)31 531 11 11

F: +41 (0)31 531 11 12

sales@ctmodule.com

OFFICE GERMANY

Frankfurter Straße 92

D-65760 Eschborn/Frankfurt

Germany

T: *+49 6196 2049173-0*

F: *+49 6196 2049173-9*

sales-d@ctmodule.com

OFFICE SERBIA

Gospodara Vučića 145

RS-11000 Belgrade

Serbia

T: +381 18 308076

sales@ctmodule.com