



Cloudezza
RADIUS-as-a-ServiceSM

Cloudezza® API Documentation Guide

Cloudezza, Inc.
2225 East Bayshore Road, Suite 200
Palo Alto, CA, 94303

July, 2013



Cloudessa RADIUS API

Cloudessa offers a powerful Application Program Interface (API) that can be used to integrate Cloudessa RADIUS authentication services with user and guest management systems.

The API framework allows different platforms to interact with the network access control infrastructure.

The Cloudessa API is a secure JavaScript Object Notation (JSON) web services API that allows you to:

- Create, delete and modify users
- Create, delete and modify all user groups
- Create, delete and modify guest users
- Create, delete, and modify virtual RADIUS servers
- Get a list of all active sessions

Using the Cloudessa API, user management systems can dynamically update the RADIUS server to instantly create or delete users in the Cloudessa internal database, and set or modify access rights.

When a user attempts to access the network, Cloudessa uses the updated credentials as the basis for allowing the user to access the network.

- For example, a custom Captive Portal self-registration application can be presented. When a user logs in, the application can make an API call to Cloudessa to add the user.
- When a user attempts to access the WiFi network, the Access Point (AP) sends a RADIUS request to Cloudessa RADIUS. Cloudessa authenticates the user based on the updated credentials.

To receive technical assistance with your Cloudessa API project, or to discuss any advanced requirements you may have, please contact us at support@cloudessa.com

Table of Contents

LOGIN/LOGOUT SERVICE	4
AUTH LOGIN	4
AUTH LOGOUT	4
USER SERVICE	5
GET USER BY ID	5
GET USER BY LOGIN	5
GET ALL USERS	6
DELETE USER	6
UPDATE USER	7
CREATE USER	8
USER GROUP SERVICE	9
GET ALL USER GROUPS	9
GET USER GROUP BY ID	9
GET USER GROUP BY NAME	10
CREATE USER GROUP	11
UPDATE USER GROUP	12
DELETE USER GROUP	13
ADD USER TO USER GROUP	14
REMOVE USER FROM USER GROUP	15
GUEST USERS SERVICE	16
GET ALL GUEST USERS	16
GET GUEST USER BY ID	16
GET USER GROUP BY LOGIN	17
CREATE GUEST USER	18
DELETE GUEST USER	19
VIRTUAL RADIUS SERVER SERVICE	20
GET ALL VIRTUAL RADIUS SERVERS	20
GET VIRTUAL RADIUS SERVER BY ID	20
GET VIRTUAL RADIUS SERVER BY NAME	21
CREATE VIRTUAL RADIUS SERVER	22
UPDATE VIRTUAL RADIUS SERVER	23
DELETE VIRTUAL RADIUS SERVER	23
ADD USER GROUP TO AUTHORIZED ON VIRTUAL RADIUS SERVER	24
REMOVE USER GROUP FROM AUTHORIZED ON VIRTUAL RADIUS SERVER	25
ACTIVE SESSIONS SERVICE	26
GET ALL ACTIVE SESSIONS	26

Login/Logout Service

Auth login

Creates a persistent API session by accepting the user's email address and password. The application must not store the user password, but may cache and store the token.

usage Method(GET)

<https://app.cloudessa.com/account/rest/login/username={email}&password={password}>

Response

equivalent to auth token

Parameters

Name	Type	Description
email	string	The user's login email address
password	string	The user's password

Auth Logout

Invalidates the current token being used. After successfully calling this function, no further API calls requiring a token will succeed using this token. If the invalidation is successful, this will return Success.

usage Method(GET)

<https://app.cloudessa.com/account/rest/logout/token={token}>

Response

Equivalent to call result

Parameters

Name	Type	Description
token	string	Auth token value

User Service

Get User by id

returns user identity for a given user id

usage Method(GET)

<https://app.cloudessa.com/account/rest/user/get/id={userid}&token={token}>

Response

if successful returns HttpStatus.OK and response body is user object

Error codes

400	User with given id was not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
userid	long	The user userid that you want to retrieve.

Get User by Login

returns user for a given user login

usage Method(GET)

<https://app.cloudessa.com/account/rest/user/get/login={login}&token={token}>

Response

if successfull returns HttpStatus.OK and response body is user object

Error codes

400	User with given login was not found in the system
-----	---

Parameters

Name	Type	Description
token	string	Auth token value
login	string	Login of user that you want to retrieve

Get All Users

returns all users for an owned organization

usage Method(GET)

<https://app.cloudessa.com/account/rest/user/getAll/token={token}>

Response

returns array of users

Parameters

Name	Type	Description
token	string	Auth token value

Delete User

delete user for a given userid

usage Method(DELETE)

<https://app.cloudessa.com/account/rest/user/delete/id={userid}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	User with given id has not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
userid	long	The user userid that you want to delete

Update User

update user for a given user object

usage Method(PUT)

<https://app.cloudessa.com/account/rest/user/update/token={token}>

Body

user object

Response

if successful returns HttpStatus.OK

Error codes

400	User with given id not found in the system
400	Email Address is already taken
400	User login already taken
400	Password length must be from 6 to 64 characters

Parameters

Name	Type	Description
token	string	Auth token value

Create User

create user for a given user object

usage Method(POST)

<https://app.cloudessa.com/account/rest/user/create/token={token}>

Body

user object

Response

if successful returns HttpStatus.CREATED, body is userid long

Error codes

400	User with given id not found in the system
400	Email Address is already taken
400	User login already taken
400	Password length must be from 6 to 64 characters

Parameters

Name	Type	Description
token	string	Auth token value

User group service

Get All User Groups

returns all user groups for owned organization

usage Method(GET)

<https://app.cloudessa.com/account/rest/group/get/token={token}>

Response

returns array of user groups

Parameters

Name	Type	Description
token	string	Auth token value

Get User Group by id

returns user for a given user id

usage Method(GET)

<https://app.cloudessa.com/account/rest/group/get/id={id}&token={token}>

Response

if successful returns HttpStatus.OK body is user group object

Error codes

400	User group with given id was not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
id	long	The user group id that you want to retrieve.

Get User Group by Name

returns user group for a given user group name

usage Method(GET)

<https://app.cloudessa.com/account/rest/group/get/name={name}&token={token}>

Response

if successful returns HttpStatus.OK body is user group object

Error codes

400	User group with given name was not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
name	string	Name of user group that you want to retrieve.

Create User Group

create user group for a given user group object

usage Method(POST)

<https://app.cloudessa.com/account/rest/group/create/token={token}>

Body

user object

Response

if successful returns HttpStatus.CREATED, body is created user group id long

Error codes

400	User group with specified name already exists
-----	---

Parameters

Name	Type	Description
token	string	Auth token value

Update User Group

update user for a given user object

usage Method(PUT)

<https://app.cloudessa.com/account/rest/group/update/token={token}>

Body

user object

Response

if successful returns HttpStatus.OK

Error codes

400	User group with given id has not found in the system
400	Cannot change user group name - it's taken by another user group.

Parameters

Name	Type	Description
token	string	Auth token value

Delete User Group

delete user group with a given id.

usage Method(DELETE)

<https://app.cloudessa.com/account/rest/group/delete/id={id}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	User group with given id has not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
id	long	The user group id that you want to delete

Add User to User Group

add specified user to a specified user group.

usage Method(PUT)

<https://app.cloudessa.com/account/rest/usertogroup/add/userid={userid}&groupid={groupid}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	User with given id was not found in the system
400	User group with given id has not found in the system

Parameters

Name	Type	Description
token	string	Auth token value
userid	long	id of user which you want to add to the user group
groupid	long	id of user group to which you want to add user

Remove User from User Group

remove specified user from a specified user group.

usage Method(PUT)

<https://app.cloudessa.com/account/rest/usertogroup/remove/userid={userid}&groupid={groupid}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	User with given id was not found in the system
400	User group with given id has not found in the system

Parameters

Name	Type	Description
token	string	Auth token value
userid	long	id of user which you want to remove from the user group
groupid	long	id of user group from which you want to remove user

Guest Users Service

Get All Guest Users

returns a list of all guest users

usage Method(GET)

<https://app.cloudessa.com/account/rest/guestuser/get/token={token}>

Response

returns array of guest users

Parameters

Name	Type	Description
token	string	Auth token value

Get Guest User by id

returns guest user for a given user id.

usage Method(GET)

<https://app.cloudessa.com/account/rest/guestuser/get/id={id}&token={token}>

Response

if successful returns HttpStatus.OK, body is guest user object

Error codes

400	Guest user with specified id was not found
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
id	long	Id of guest user that you want to retrieve

Get User Group by Login

returns guest user for a given login

usage Method(GET)

<https://app.cloudessa.com/account/rest/guestuser/get/login={login}&token={token}>

Response

if successful returns HttpStatus.OK, body is guest user object

Error codes

400	Guest user with specified login was not found
-----	---

Parameters

Name	Type	Description
token	string	Auth token value
login	string	Login of guest user that you want to retrieve

Create Guest User

create guest user for a given guest user object

usage Method(POST)

<https://app.cloudessa.com/account/rest/guestuser/create/serverid={serverid}&token={token}>

Body

guest user object

Response

if successful returns HttpStatus.CREATED, body is created guest user id long

Error codes

400	Virtual RADIUS server with specified id was not found in the system.
400	Guest user with specified login already exists.

Parameters

Name	Type	Description
token	string	Auth token value
serverid	long	id of Virtual RADIUS server to which you want to assign this guest user

Delete Guest User

delete guest user with a given id

usage Method(DELETE)

<https://app.cloudessa.com/account/rest/guestuser/delete/id={id}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	Guest user with given id has not found in the system
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
id	long	id of user which you want to delete

Virtual RADIUS Server Service

Get All Virtual RADIUS Servers

returns list of all virtual RADIUS server

usage Method(GET)

<https://app.cloudessa.com/account/rest/server/get/token={token}>

Response

returns array of virtual RADIUS servers

Parameters

Name	Type	Description
token	string	Auth token value

Get Virtual RADIUS Server by id

returns guest user for a given user id

usage Method(GET)

<https://app.cloudessa.com/account/rest/server/get/id={id}&token={token}>

Response

if successful returns HttpStatus.OK, body is virtual RADIUS server object

Error codes

400	Server with specified id was not found
-----	--

Parameters

Name	Type	Description
token	string	Auth token value
id	long	Id of server that you want to retrieve

Get Virtual RADIUS Server by Name

returns virtual RADIUS server for a given server name

usage Method(GET)

<https://app.cloudessa.com/account/rest/server/get/name={name}&token={token}>

Response

if successful returns HttpStatus.OK, body is virtual RADIUS server object

Error codes

400	Virtual RADIUS server with specified name was not found
-----	---

Parameters

Name	Type	Description
token	string	Auth token value
name	string	Name of server that you want to retrieve

Create Virtual RADIUS Server

create virtual RADIUS server

usage Method(POST)

<https://app.cloudessa.com/account/rest/server/create/token={token}>

Body

virtual RADIUS server object

Response

if successful returns HttpStatus.CREATED, body is created server id

Error codes

400	Virtual RADIUS server with specified name already exists
-----	--

Parameters

Name	Type	Description
token	string	Auth token value

Update Virtual RADIUS Server

update virtual RADIUS server for a given server object

usage Method(PUT)

<https://app.cloudessa.com/account/rest/server/update/token={token}>

Body

server object

Response

if successful returns HttpStatus.OK

Error codes

400	Virtual RADIUS server with given id has not found
-----	---

Parameters

Name	Type	Description
token	string	Auth token value

Delete Virtual RADIUS Server

delete virtual RADIUS server with a given id

usage Method(DELETE)

<https://app.cloudessa.com/account/rest/server/delete/id={id}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	Virtual RADIUS server with given id was not found
-----	---

Parameters

Name	Type	Description
token	string	Auth token value
id	long	id of server which you want to delete

Add User Group to Authorized on Virtual RADIUS Server

add specified user group to authorized on specified server

usage Method(PUT)

<https://app.cloudessa.com/account/rest/grouptovserver/add/groupid={groupid}&serverid={serverid}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	Group with given id was not found
400	Virtual RADIUS server with given id was not found

Parameters

Name	Type	Description
token	string	Auth token value
groupid	long	id of user group which you want to add to server
serverid	long	id of server to which you want to add user group

Remove User Group from Authorized on Virtual RADIUS Server

remove specified user group from authorized on specified server

usage Method(PUT)

<https://app.cloudessa.com/account/rest/grouptovserver/remove/groupid={groupid}&serverid={serverid}&token={token}>

Response

if successful returns HttpStatus.OK

Error codes

400	Group with given id was not found
400	Virtual RADIUS server with given id was not found

Parameters

Name	Type	Description
token	string	Auth token value
groupid	long	id of user group which you want to remove from server
serverid	long	id of server to which you want to add user group

Active sessions service

Get All Active Sessions

returns list of all currently active sessions

usage Method(GET)

<https://app.cloudessa.com/account/rest/activesession/getAll/token={token}>

Response

return an array of active sessions

Parameters

Name	Type	Description
token	string	Auth token value