Resource: access_tokens

Access token handling

http://{device}/api/mgmt.aaa/2.0/token

JSON

```
{
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>access_tokens</td>
<td>&lt;object&gt;</td>
<td>Access token handling</td>
<td></td>
</tr>
</tbody>
</table>

Links

access_tokens: request

Request a new access token

POST http://{device}/api/mgmt.aaa/2.0/token

Request Body

Provide a request body with the following structure:

JSON

```
{
    "generate_refresh_token": boolean,
    "refresh_token": refresh_token_assertion,
    "state": string,
    "user_credentials": user_credentials_assertion
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>access_tokens.links.request.generate_refresh_token</td>
<td>&lt;boolean&gt;</td>
<td>If True, a refresh token will also be returned</td>
<td>Optional;</td>
</tr>
<tr>
<td>access_tokens.links.request.request_refresh_token</td>
<td>&lt;refresh_token_assertion&gt;</td>
<td>A refresh token assertion</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.state</td>
<td>&lt;string&gt;</td>
<td>Optional opaque value to pass back with the token</td>
<td>Optional;</td>
</tr>
<tr>
<td>access_tokens.links.request.request.user_credentials</td>
<td>&lt;user_credentials_assertion&gt;</td>
<td>A username/password assertion</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.oneOf[0]</td>
<td>&lt;object&gt;</td>
<td>Required properties: [user_credentials];</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.oneOf[0].&lt;prop&gt;</td>
<td>&lt;any&gt;</td>
<td>Optional;</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.oneOf[1]</td>
<td>&lt;object&gt;</td>
<td>Required properties: [refresh_token];</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.oneOf[1].generate_refresh_token</td>
<td>&lt;boolean&gt;</td>
<td>Optional; Values: False;</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.request.oneOf[1].&lt;prop&gt;</td>
<td>&lt;any&gt;</td>
<td>Optional;</td>
<td></td>
</tr>
</tbody>
</table>

Response Body

On success, the server returns a response body with the following structure:

JSON

```
{
    "access_token": string,
    "expires_at": integer,
    "refresh_token": string,
    "state": string,
    "token_type": string
}
```
<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>access_tokens.links.request.response</td>
<td>&lt;object&gt;</td>
<td>Granted access token details</td>
<td>Required properties: [access_token, expires_at, token_type];</td>
</tr>
<tr>
<td>access_tokens.links.request.response. access_token</td>
<td>&lt;string&gt;</td>
<td>The new access token</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.response. expires_at</td>
<td>&lt;integer&gt;</td>
<td>The Unix epoch time when the access token will expire</td>
<td></td>
</tr>
<tr>
<td>access_tokens.links.request.response. refresh_token</td>
<td>&lt;string&gt;</td>
<td>The new refresh token, if requested. This token must be stored securely.</td>
<td>Optional;</td>
</tr>
<tr>
<td>access_tokens.links.request.response. state</td>
<td>&lt;string&gt;</td>
<td>The value of the state field in the request, if present</td>
<td>Optional;</td>
</tr>
<tr>
<td>account_policy</td>
<td>&lt;object&gt;</td>
<td>Global account settings</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>account_policy.login_policy</td>
<td>&lt;object&gt;</td>
<td>Settings related to login attempts</td>
<td>Required properties: [count, wait_time];</td>
</tr>
<tr>
<td>account_policy.login_policy.count</td>
<td>&lt;integer&gt;</td>
<td>Maximum failed login attempts before temporary account lock. 0 disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.login_policy.wait_time</td>
<td>&lt;integer&gt;</td>
<td>Length in minutes for temporary account lock. N/A when count is 0.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy</td>
<td>&lt;object&gt;</td>
<td>Password complexity and expiration settings</td>
<td>Required properties: [permit_empty_passwords, minimum_length, lower_case, upper_case, digits, symbols, repeat, difference, dictionary_check, change_frequency, reuse_interval, expiration];</td>
</tr>
<tr>
<td>account_policy.password_policy. change_frequency</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of days a user must wait between password changes. 0 disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy. dictionary_check</td>
<td>&lt;boolean&gt;</td>
<td>Disallow passwords based on common words</td>
<td></td>
</tr>
<tr>
<td>account_policy.password_policy. difference</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of character differences required between two passwords. 0 disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------</td>
</tr>
<tr>
<td>account_policy.password_policy.digits</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of digits. 0 Disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration</td>
<td>&lt;object&gt;</td>
<td>Password expiration settings</td>
<td>Required properties: [time, inactive, warn];</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.inactive</td>
<td>&lt;object&gt;</td>
<td>Account inactivity settings</td>
<td></td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.inactive.enabled</td>
<td>&lt;boolean&gt;</td>
<td>Whether to mark accounts inactive if their password remains expired for a period of time</td>
<td>Optional;</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.inactive.value</td>
<td>&lt;integer&gt;</td>
<td>Number of days before an account with an expired password is marked inactive</td>
<td>Optional; Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.time</td>
<td>&lt;object&gt;</td>
<td>Password expiration settings</td>
<td></td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.time.enabled</td>
<td>&lt;boolean&gt;</td>
<td>Whether to expire passwords after a period of time</td>
<td>Optional;</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.time.value</td>
<td>&lt;integer&gt;</td>
<td>Number of days before a password expires</td>
<td>Optional; Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.expiration.warn</td>
<td>&lt;integer&gt;</td>
<td>Number of days before password expiration to start warning a user</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.lower_case</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of lowercase characters. 0 Disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.minimum_length</td>
<td>&lt;integer&gt;</td>
<td>Minimum password length.</td>
<td>Range: 1 to 64;</td>
</tr>
<tr>
<td>account_policy.password_policy.permit_empty_passwords</td>
<td>&lt;boolean&gt;</td>
<td>Allow users to have empty passwords.</td>
<td></td>
</tr>
<tr>
<td>account_policy.password_policy.repeat</td>
<td>&lt;integer&gt;</td>
<td>Maximum times the same character can repeat consecutively. 0 Disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.reuse_interval</td>
<td>&lt;integer&gt;</td>
<td>Number of previous passwords to save. When setting a new password, the user cannot use a password that exists in their password history. 0 Disables this check.</td>
<td>Range: 0 to 10;</td>
</tr>
<tr>
<td>account_policy.password_policy.symbols</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of symbols. 0 Disables this check.</td>
<td>Minimum 0;</td>
</tr>
<tr>
<td>account_policy.password_policy.upper_case</td>
<td>&lt;integer&gt;</td>
<td>Minimum number of uppercase characters. 0 Disables this check.</td>
<td>Minimum 0;</td>
</tr>
</tbody>
</table>

**Links**

**account_policy: get**

GET http://{device}/api/mgmt.aaa/2.0/account_policy

Response Body
- Returns an account_policy data object.

**account_policy: set**

PUT http://{device}/api/mgmt.aaa/2.0/account_policy

Request Body
- Provide an account_policy data object.

Response Body
- Returns an account_policy data object.

**Resource: passwords**

Local user password management

http://{device}/api/mgmt.aaa/2.0/passwords

JSON

```json
{
}
```
**Links**

**passwords: change_password**

Change a local user's password. Old password is required when changing your own password.

**Request Body**

Provide a request body with the following structure:

```json
{
  "new_password": string,
  "old_password": string,
  "user": string
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>passwords.links.change_password.request</td>
<td>&lt;object&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>passwords.links.change_password.request.new_password</td>
<td>&lt;string&gt;</td>
<td>New password, in plaintext</td>
<td></td>
</tr>
<tr>
<td>passwords.links.change_password.request.old_password</td>
<td>&lt;string&gt;</td>
<td>Old password, in plaintext</td>
<td>Optional;</td>
</tr>
<tr>
<td>passwords.links.change_password.request.user</td>
<td>&lt;string&gt;</td>
<td>User account to change password</td>
<td></td>
</tr>
</tbody>
</table>

**Response Body**

On success, the server returns a response body with the following structure:

```json
{
  "changed": boolean,
  "user": string
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>passwords.links.change_password.response</td>
<td>&lt;object&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>passwords.links.change_password.response.changed</td>
<td>&lt;boolean&gt;</td>
<td>Whether the password was successfully changed</td>
<td></td>
</tr>
<tr>
<td>passwords.links.change_password.response.user</td>
<td>&lt;string&gt;</td>
<td>User account</td>
<td></td>
</tr>
</tbody>
</table>

**Resource: permission_group**

Service resource group used to assign permissions

http://{device}/api/mgmt.aaa/2.0/permission_groups/{name}

```json
{
  "description": string,
  "name": string,
  "pretty_name": string,
  "resources": [ service_resource ]
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>permission_group</td>
<td>&lt;object&gt;</td>
<td>Service resource group used to assign permissions</td>
<td>Required properties: [name, pretty_name, description, resources];</td>
</tr>
<tr>
<td>Property Name</td>
<td>Type</td>
<td>Description</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>permission_group.name</td>
<td>&lt;string&gt;</td>
<td>Unique ID for group</td>
<td>Read-only; Pattern: '[a-zA-Z0-9_]+$';</td>
</tr>
<tr>
<td>permission_group.pretty_name</td>
<td>&lt;string&gt;</td>
<td>Display name for group</td>
<td></td>
</tr>
<tr>
<td>permission_group.resources</td>
<td>&lt;array of &lt;service_resource&gt;&gt;</td>
<td>List of resources that exist in this group</td>
<td></td>
</tr>
<tr>
<td>permission_group.resources[items]</td>
<td>&lt;service_resource&gt;</td>
<td>Resources available from a service. If only the service_name property is present, all resources are included.</td>
<td></td>
</tr>
</tbody>
</table>

**Links**

**permission_group: get**

```plaintext
GET http://{device}/api/mgmt.aaa/2.0/permission_groups/{name}
```

Response Body
Returns a permission_group data object.

**Relations**

**permission_group: instances**

Related resource
permission_groups

**Resource: permission_groups**

Collection of service resource groups

```plaintext
http://{device}/api/mgmt.aaa/2.0/permission_groups
```

**JSON**

```json
{
    "items": [permission_group]
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>permission_groups</td>
<td>&lt;object&gt;</td>
<td>Collection of service resource groups</td>
<td>Required properties: [items];</td>
</tr>
<tr>
<td>permission_groups.items</td>
<td>&lt;array of&lt;permission_group&gt;&gt;</td>
<td>List of service resource groups</td>
<td></td>
</tr>
<tr>
<td>permission_groups.items[items]</td>
<td>&lt;permissions_group&gt;</td>
<td>Service resource group used to assign permissions</td>
<td></td>
</tr>
</tbody>
</table>

**Links**

**permission_groups: get**

```plaintext
GET http://{device}/api/mgmt.aaa/2.0/permission_groups
```

Response Body
Returns a permission_groups data object.

**Resource: radius_server**

A RADIUS authentication server

```plaintext
http://{device}/api/mgmt.aaa/2.0/radius_servers/items/{id}
```

**JSON**
```json
{
    "enabled": boolean,
    "host": string,
    "id": integer,
    "new_key": string,
    "port": integer,
    "timeout": integer
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius_server</td>
<td>&lt;object&gt;</td>
<td>A RADIUS authentication server</td>
<td>Required properties: [host, port, timeout];</td>
</tr>
<tr>
<td>radius_server.enabled</td>
<td>&lt;boolean&gt;</td>
<td>Whether this RADIUS server is enabled or not. A server is enabled when it is in the server_order list in the radius_servers resource.</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>radius_server.host</td>
<td>&lt;string&gt;</td>
<td>The hostname or IP address of the RADIUS server</td>
<td></td>
</tr>
<tr>
<td>radius_server.id</td>
<td>&lt;integer&gt;</td>
<td>Server ID</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>radius_server.new_key</td>
<td>&lt;string&gt;</td>
<td>The secret key used to encrypt communications. An empty string indicates no encryption is used.</td>
<td>Optional;</td>
</tr>
<tr>
<td>radius_server.port</td>
<td>&lt;integer&gt;</td>
<td>The port of the RADIUS server</td>
<td>Range: 1 to 65535; Default is 1812;</td>
</tr>
<tr>
<td>radius_server.timeout</td>
<td>&lt;integer&gt;</td>
<td>Max time, in seconds, to wait for the server to respond to an auth request.</td>
<td>Range: 1 to 30;</td>
</tr>
</tbody>
</table>

### Links

**radius_server: delete**

DELETE http://{device}/api/mgmt.aaa/2.0/radius_servers/items/{id}

Response Body

On success, the server does not provide any body in the responses.

**radius_server: get**

GET http://{device}/api/mgmt.aaa/2.0/radius_servers/items/{id}

Response Body

Returns a `radius_server` data object.

**radius_server: set**

PUT http://{device}/api/mgmt.aaa/2.0/radius_servers/items/{id}

Request Body

Provide a `radius_server` data object.

Response Body

Returns a `radius_server` data object.

### Resource: radius_servers

The configured RADIUS servers

http://{device}/api/mgmt.aaa/2.0/radius_servers

JSON


```json
{
  "available_encryption": [
    "string"
  ],
  "encryption_protocol": "string",
  "server_priority": [
    "integer"
  ],
  "servers": [ "radius_server" ]
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>radius_servers</td>
<td>&lt;object&gt;</td>
<td>The configured RADIUS servers</td>
<td>Required properties: [server_priority, encryption_protocol];</td>
</tr>
<tr>
<td>radius_servers.available_encryption</td>
<td>&lt;array of &lt;string&gt;&gt;</td>
<td>Available encryption protocols</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>radius_servers.encryption_protocol</td>
<td>&lt;string&gt;</td>
<td>The encryption protocol to use. Available protocols are listed under available_encryption.</td>
<td>Read-only;</td>
</tr>
<tr>
<td>radius_servers.server_priority</td>
<td>&lt;array of &lt;integer&gt;&gt;</td>
<td>The order in which authentication requests are made to the configured servers. Servers not in this list will be disabled.</td>
<td></td>
</tr>
<tr>
<td>radius_servers.servers</td>
<td>&lt;array of &lt;radius_server&gt;&gt;</td>
<td>The configured RADIUS servers</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>radius_servers.servers[items]</td>
<td>&lt;radius_server&gt;</td>
<td>A RADIUS authentication server</td>
<td></td>
</tr>
</tbody>
</table>

**Links**

radius_servers: create

```
POST http://{device}/api/mgmt.aaa/2.0/radius_servers
```

**Request Body**

Provide a `radius_server` data object.

**Response Body**

Returns a `radius_server` data object.

radius_servers: get

```
GET http://{device}/api/mgmt.aaa/2.0/radius_servers
```

**Response Body**

Returns a `radius_servers` data object.

radius_servers: set

```
PUT http://{device}/api/mgmt.aaa/2.0/radius_servers
```

**Request Body**

Provide a `radius_servers` data object.

**Response Body**

Returns a `radius_servers` data object.

**Resource: refresh_tokens**

Active refresh tokens. Only the first few characters of the token are revealed, enough for the caller to differentiate the tokens they own.

```
http://{device}/api/mgmt.aaa/2.0/refresh_tokens
```
### Property Name | Type | Description | Notes
--- | --- | --- | ---
refresh_tokens | <object> | Active refresh tokens. Only the first few characters of the token are revealed, enough for the caller to differentiate the tokens they own. | Required properties: [items];
refresh_tokens.items | <array of <object>> | | |
refresh_tokens.items.items | <object> | A single refresh token | Required properties: [user, partial_token, issued_at, last_redeemed, times_redeemed];
refresh_tokens.items.items.issued_at | <any> | The Unix epoch time that the refresh token was issued | |
refresh_tokens.items.items.last_redeemed | <integer> | The Unix epoch time that the token was last redeemed. 0 if it has never been used. | Minimum 0;
refresh_tokens.items.items.partial_token | <string> | The first few characters of the token | |
refresh_tokens.items.items.times_redeemed | <integer> | The number of times the token has been redeemed | Minimum 0;
refresh_tokens.items.items.user | <string> | The user name which owns this token | |

### Links

**refresh_tokens: get**

[GET](http://{device}/api/mgmt.aaa/2.0/refresh_tokens)

Response Body

Returns a refresh_tokens data object.

**refresh_tokens: revoke**

Revoke a refresh token

[POST](http://{device}/api/mgmt.aaa/2.0/refresh_tokens/revoke)

Request Body

Provide a request body with the following structure:

```json
{
    "refresh_token": string
}
```

Response Body

On success, the server does not provide any body in the responses.

---

Resource: remote_authentication
Remote authentication settings

http://{device}/api/mgmt.aaa/2.0/remote_authentication

### JSON

```json
{
    "auth_methods_available": [
        string
    ],
    "auth_sequence": [
        string
    ],
    "default_roles": [
        integer
    ],
    "next_method_on_reject": boolean
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>remote_authentication</td>
<td>&lt;object&gt;</td>
<td>Remote authentication settings</td>
<td>Required properties: [auth_sequence, next_method_on_reject, default_roles];</td>
</tr>
<tr>
<td>remote_authentication.auth_methods_available</td>
<td>&lt;array of &lt;string&gt;&gt;</td>
<td>Available authentication methods</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>remote_authentication.auth_sequence</td>
<td>&lt;array of &lt;string&gt;&gt;</td>
<td>Authentication methods to use, in priority order of first to last. Possible values are listed in auth_methods_available.</td>
<td>Read-only;</td>
</tr>
<tr>
<td>remote_authentication.auth_sequence[items]</td>
<td>&lt;string&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remote_authentication.default_roles</td>
<td>&lt;array of &lt;integer&gt;&gt;</td>
<td>The roles to assign to a remotely-authenticated user when the authentication server does not specify any.</td>
<td></td>
</tr>
<tr>
<td>remote_authentication.default_roles[items]</td>
<td>&lt;integer&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>remote_authentication.next_method_on_reject</td>
<td>&lt;boolean&gt;</td>
<td>If True, when a login is rejected, the system will still attempt to authenticate via the next method in auth_sequence. If False, the login attempt is denied immediately when any auth method rejects the user credentials.</td>
<td></td>
</tr>
</tbody>
</table>

### Links

**remote_authentication: get**

GET http://{device}/api/mgmt.aaa/2.0/remote_authentication

Response Body

Returns a `remote_authentication` data object.

**remote_authentication: set**

PUT http://{device}/api/mgmt.aaa/2.0/remote_authentication

Request Body

Provide a `remote_authentication` data object.

Response Body

Returns a `remote_authentication` data object.

### Resource: role

A set of permissions that may be assigned to a user

http://{device}/api/mgmt.aaa/2.0/roles/{id}

JSON
{ "description": string,
"id": integer,
"member_of": [ integer ]
},
"permissions": [
{ "operation": string,
  "permission_group": string
}
],
"pretty_name": string,
"system_default": boolean
}

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>role</td>
<td>&lt;object&gt;</td>
<td>A set of permissions that may be assigned to a user</td>
<td>Required properties: [pretty_name];</td>
</tr>
<tr>
<td>role.description</td>
<td>&lt;string&gt;</td>
<td>Role description</td>
<td>Optional;</td>
</tr>
<tr>
<td>role.id</td>
<td>&lt;integer&gt;</td>
<td>Unique role identifier</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>role.member_of</td>
<td>&lt;array of &lt;integer&gt;&gt;</td>
<td>Roles that this role is a member of</td>
<td>Optional;</td>
</tr>
<tr>
<td>role.permissions[items]</td>
<td>&lt;object&gt;</td>
<td>A set of permissions granted to this role</td>
<td>Optional;</td>
</tr>
<tr>
<td>role.permissions[items].operation</td>
<td>&lt;string&gt;</td>
<td>The access rights granted to the permission group</td>
<td>Optional; Values: read_only, read_write;</td>
</tr>
<tr>
<td>role.permissions[items].permission_group</td>
<td>&lt;string&gt;</td>
<td>The permission group</td>
<td>Optional;</td>
</tr>
<tr>
<td>role.pretty_name</td>
<td>&lt;string&gt;</td>
<td>Unique role name</td>
<td></td>
</tr>
<tr>
<td>role.system_default</td>
<td>&lt;boolean&gt;</td>
<td>If true, role is a system default entry that cannot be modified</td>
<td>Read-only; Optional;</td>
</tr>
</tbody>
</table>

**Links**

**role: delete**

DELETE http://{device}/api/mgmt.aaa/2.0/roles/{id}

Response Body
On success, the server does not provide any body in the responses.

**role: get**

GET http://{device}/api/mgmt.aaa/2.0/roles/{id}

Response Body
Returns a role data object.

**role: set**

PUT http://{device}/api/mgmt.aaa/2.0/roles/{id}

Request Body
Provide a role data object.

Response Body
Returns a role data object.

**Resource: role_names**

The list of roles and their names

http://{device}/api/mgmt.aaa/2.0/role_names
### JSON

```json
{
  "items": [
    {
      "description": string,
      "id": integer,
      "pretty_name": string
    }
  ]
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>role_names</td>
<td>&lt;object&gt;</td>
<td>The list of roles and their names</td>
<td>Required properties: [items];</td>
</tr>
<tr>
<td>role_names.items</td>
<td>&lt;array of</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;object&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role_names.items.id</td>
<td>&lt;integer&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role_names.items.pretty_name</td>
<td>&lt;string&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>role_names.items.description</td>
<td>&lt;string&gt;</td>
<td>Role description</td>
<td></td>
</tr>
<tr>
<td>role_names.items[items]</td>
<td>&lt;object&gt;</td>
<td>Name and description for a single role</td>
<td>Required properties: [id, pretty_name, description];</td>
</tr>
</tbody>
</table>

### Links

**role_names: get**

GET `{device}/api/mgmt.aaa/2.0/role_names`

**Response Body**

Returns a `role_names` data object.

### Resource: roles

All configured roles

http://{device}/api/mgmt.aaa/2.0/roles

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>roles</td>
<td>&lt;object&gt;</td>
<td>All configured roles</td>
<td></td>
</tr>
<tr>
<td>roles.items</td>
<td>&lt;array of</td>
<td>A role identifier</td>
<td>Optional;</td>
</tr>
<tr>
<td></td>
<td>&lt;role&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>roles.items[items]</td>
<td>&lt;role&gt;</td>
<td>A set of permissions that may be assigned to a user</td>
<td></td>
</tr>
</tbody>
</table>

### Links

**roles: create**

POST `{device}/api/mgmt.aaa/2.0/roles`

**Request Body**

Provide a `role` data object.

**Response Body**

Returns a `role` data object.

**roles: get**
GET http://{device}/api/mgmt.aaa/2.0/roles

Response Body
Returns a `roles` data object.

**Resource: tacacs_server**

A TACACS+ authentication server

http://{device}/api/mgmt.aaa/2.0/tacacs_servers/items/{id}

**JSON**

```json
{
  "enabled": boolean,
  "host": string,
  "id": integer,
  "new_key": string,
  "port": integer
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>tacacs_server</td>
<td>&lt;object&gt;</td>
<td>A TACACS+ authentication server</td>
<td>Required properties: [host, port];</td>
</tr>
<tr>
<td>tacacs_server.enabled</td>
<td>&lt;boolean&gt;</td>
<td>Whether this TACACS+ server is enabled or not. A server is enabled when it is in the server_order list in the tacacs_servers resource.</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>tacacs_server.host</td>
<td>&lt;string&gt;</td>
<td>The hostname or IP address of the TACACS+ server</td>
<td></td>
</tr>
<tr>
<td>tacacs_server.id</td>
<td>&lt;integer&gt;</td>
<td>Server ID</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>tacacs_server.new_key</td>
<td>&lt;string&gt;</td>
<td>The secret key used to encrypt communications. An empty string indicates no encryption is used.</td>
<td>Optional;</td>
</tr>
<tr>
<td>tacacs_server.port</td>
<td>&lt;integer&gt;</td>
<td>The post of the TACACS+ server</td>
<td>Range: 1 to 65535; Default is 49;</td>
</tr>
</tbody>
</table>

**Links**

**tacacs_server: delete**

DELETE http://{device}/api/mgmt.aaa/2.0/tacacs_servers/items/{id}

Response Body
On success, the server does not provide any body in the responses.

**tacacs_server: get**

GET http://{device}/api/mgmt.aaa/2.0/tacacs_servers/items/{id}

Response Body
Returns a `tacacs_server` data object.

**tacacs_server: set**

PUT http://{device}/api/mgmt.aaa/2.0/tacacs_servers/items/{id}

Request Body
Provide a `tacacs_server` data object.

Response Body
Returns a `tacacs_server` data object.

**Resource: tacacs_servers**

The configured TACACS+ servers
HTTP://{device}/api/mgmt.aaa/2.0/tacacs_servers

JSON

```
{
    "server_priority": [integer],
    "servers": [tacacs_server],
    "timeout": integer
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>tacacs_servers</td>
<td>&lt;object&gt;</td>
<td>The configured TACACS+ servers</td>
<td>Required properties: [server_priority, timeout];</td>
</tr>
<tr>
<td>tacacs_servers.server_priority</td>
<td>&lt;array of &lt;integer&gt;&gt;</td>
<td>The order in which authentication requests are made to the configured servers. Servers not in this list will be disabled.</td>
<td></td>
</tr>
<tr>
<td>tacacs_servers.servers</td>
<td>&lt;array of &lt;tacacs_server&gt;&gt;</td>
<td>The configured TACACS+ servers</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>tacacs_servers.servers</td>
<td>&lt;tacacs_server&gt;</td>
<td>A TACACS+ authentication server</td>
<td></td>
</tr>
<tr>
<td>tacacs_servers.timeout</td>
<td>&lt;integer&gt;</td>
<td>Max time, in seconds, to wait for a server to respond to an auth request.</td>
<td>Range: 1 to 30;</td>
</tr>
</tbody>
</table>

Links

**tacacs_servers: create**

POST http://{device}/api/mgmt.aaa/2.0/tacacs_servers

Request Body

Provide a **tacacs_server** data object.

Response Body

Returns a **tacacs_server** data object.

**tacacs_servers: get**

GET http://{device}/api/mgmt.aaa/2.0/tacacs_servers

Response Body

Returns a **tacacs_servers** data object.

**tacacs_servers: set**

PUT http://{device}/api/mgmt.aaa/2.0/tacacs_servers

Request Body

Provide a **tacacs_servers** data object.

Response Body

Returns a **tacacs_servers** data object.

**Resource: user**

A user configured for local authentication

HTTP://{device}/api/mgmt.aaa/2.0/users/{name}

JSON
```json
{
  "account_never_inactive": boolean,
  "description": string,
  "enable": boolean,
  "logged_in": boolean,
  "login_failure": {
    "count": integer,
    "date": timestamp,
    "source": string
  },
  "name": string,
  "new_password": {
    "cleartext": string,
    "hashed": string
  },
  "password": {
    "change_allowed_in": integer,
    "expires_on": timestamp,
    "locks_on": timestamp
  },
  "password_never_expires": boolean,
  "roles": [integer]
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>user</td>
<td>&lt;object&gt;</td>
<td>A user configured for local authentication</td>
<td>Required properties: [name];</td>
</tr>
<tr>
<td>user:account_never_inactive</td>
<td>&lt;boolean&gt;</td>
<td>User account will never become inactive, preventing login, due to an expired password</td>
<td>Optional; Default is False;</td>
</tr>
<tr>
<td>user:description</td>
<td>&lt;string&gt;</td>
<td>Description for the user account</td>
<td>Optional; Default is ;</td>
</tr>
<tr>
<td>user:enable</td>
<td>&lt;boolean&gt;</td>
<td>Whether the user is allowed to log in</td>
<td>Optional; Default is False;</td>
</tr>
<tr>
<td>user:logged_in</td>
<td>&lt;boolean&gt;</td>
<td>User is currently logged in</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>user:login_failure</td>
<td>&lt;object&gt;</td>
<td>Login failure information</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>user:login_failure.count</td>
<td>&lt;integer&gt;</td>
<td>Number of failed login attempts</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>user:login_failure.date</td>
<td>&lt;timestamp&gt;</td>
<td>Date, in Unix epoch time, of the most recent login failure</td>
<td>Read-only; Optional; Seconds since January 1, 1970;</td>
</tr>
<tr>
<td>user:login_failure.source</td>
<td>&lt;string&gt;</td>
<td>Address of the most recent login failure</td>
<td>Read-only; Optional;</td>
</tr>
<tr>
<td>user:name</td>
<td>&lt;string&gt;</td>
<td>Account name</td>
<td></td>
</tr>
<tr>
<td>user:new_password</td>
<td>&lt;object&gt;</td>
<td>Set this user's password to a new value. Used for creation and bulk importing of users. Normal password changes should go through the password resource. Changing passwords using this API may result in passwords which violate the password policy.</td>
<td>Optional;</td>
</tr>
<tr>
<td>user:new_password.cleartext</td>
<td>&lt;string&gt;</td>
<td>Set the user's password in plain text</td>
<td>Optional;</td>
</tr>
<tr>
<td>user:new_passwordhashed</td>
<td>&lt;string&gt;</td>
<td>Set the user's password as a hash</td>
<td>Optional;</td>
</tr>
<tr>
<td>user:new_password.oneOf[0]</td>
<td>&lt;object&gt;</td>
<td>Required properties: [cleartext];</td>
<td></td>
</tr>
<tr>
<td>user:new_password.oneOf[0].&lt;prop&gt;</td>
<td>&lt;any&gt;</td>
<td>Optional;</td>
<td></td>
</tr>
<tr>
<td>user:new_password.oneOf[1]</td>
<td>&lt;object&gt;</td>
<td>Required properties: [hashed];</td>
<td></td>
</tr>
<tr>
<td>user:new_password.oneOf[1].&lt;prop&gt;</td>
<td>&lt;any&gt;</td>
<td>Optional;</td>
<td></td>
</tr>
<tr>
<td>user:password</td>
<td>&lt;object&gt;</td>
<td>Password settings</td>
<td>Required properties: [expires_on, locks_on, change_allowed_in]; Optional;</td>
</tr>
<tr>
<td>user:password.change_allowed_in</td>
<td>&lt;integer&gt;</td>
<td>Days remaining until the user can change their password. A value of 0 indicates the password may be changed immediately.</td>
<td>Read-only;</td>
</tr>
<tr>
<td>user:password.expires_on</td>
<td>&lt;timestamp&gt;</td>
<td>Date, in Unix epoch time, after which the user’s password will expire. If 0, the password will never expire.</td>
<td>Read-only; Seconds since January 1, 1970;</td>
</tr>
<tr>
<td>user:password.locks_on</td>
<td>&lt;timestamp&gt;</td>
<td>Date, in Unix epoch time, after which the user account will be locked due to an expired password. If 0, the account will never be locked.</td>
<td>Read-only; Seconds since January 1, 1970;</td>
</tr>
<tr>
<td>user:password_never_expires</td>
<td>&lt;boolean&gt;</td>
<td>User account password will never expire</td>
<td>Optional; Default is False;</td>
</tr>
<tr>
<td>user:roles</td>
<td>&lt;array of &lt;integer&gt;&gt;</td>
<td>List of roles granted to this user</td>
<td>Optional;</td>
</tr>
<tr>
<td>user:roles[items]</td>
<td>&lt;integer&gt;</td>
<td>The role identifier</td>
<td></td>
</tr>
<tr>
<td>user:status</td>
<td>&lt;string&gt;</td>
<td>Status of the account</td>
<td>Read-only; Optional; Values: active, inactive, disabled, login_failure_lockout;</td>
</tr>
</tbody>
</table>
**Links**

**user: delete**

DELETE http://{device}/api/mgmt.aaa/2.0/users/{name}

Response Body
On success, the server does not provide any body in the responses.

**user: get**

GET http://{device}/api/mgmt.aaa/2.0/users/{name}

Response Body
Returns an **user** data object.

**user: set**

PUT http://{device}/api/mgmt.aaa/2.0/users/{name}

Request Body
Provide an **user** data object.

Response Body
Returns an **user** data object.

**Resource: users**

The users configured for local authentication

```
http://{device}/api/mgmt.aaa/2.0/users
```

<table>
<thead>
<tr>
<th>JSON</th>
</tr>
</thead>
<tbody>
<tr>
<td>{</td>
</tr>
<tr>
<td>&quot;items&quot;: [ <strong>user</strong> ]</td>
</tr>
<tr>
<td>}</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>users</td>
<td>&lt;object&gt;</td>
<td>The users configured for local authentication</td>
<td></td>
</tr>
<tr>
<td>users.items</td>
<td>&lt;array of &lt;user&gt;&gt;</td>
<td></td>
<td>Optional;</td>
</tr>
<tr>
<td>users.items[items]</td>
<td>&lt;user&gt;</td>
<td>A user configured for local authentication</td>
<td></td>
</tr>
</tbody>
</table>

**Links**

**users: create**

POST http://{device}/api/mgmt.aaa/2.0/users

Request Body
Provide an **user** data object.

Response Body
Returns an **user** data object.

**users: get**

GET http://{device}/api/mgmt.aaa/2.0/users
Response Body
Returns an `users` data object.

**Type: authentication**
The type of authentication used

**JSON**

```json
{
  "authentication": <string>
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>authentication</td>
<td>&lt;string&gt;</td>
<td>The type of authentication used</td>
<td>Values: local, RADIUS, TACACS+, other-remote;</td>
</tr>
</tbody>
</table>

**Type: refresh_token_assertion**
A refresh token assertion

**JSON**

```json
{
  "refresh_token_assertion": <string>
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>refresh_token_assertion</td>
<td>&lt;string&gt;</td>
<td>A refresh token assertion</td>
</tr>
</tbody>
</table>

**Type: service_resource**
Resources available from a service. If only the `service_name` property is present, all resources are included.

**JSON**

```json
{
  "*all_except": [ string ],
  "*only_include": [ string ],
  "*service_name": string
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_resource</td>
<td>&lt;object&gt;</td>
<td>Resources available from a service. If only the service_name property is present, all resources are included.</td>
<td>Required properties: [service_name];</td>
</tr>
<tr>
<td>service_resource.all_except</td>
<td>&lt;array of &lt;string&gt;&gt;</td>
<td>List of resources excluded from this group. All other resources from this service are included.</td>
<td>Optional;</td>
</tr>
<tr>
<td>service_resource.all_except</td>
<td>&lt;array of string&gt;</td>
<td>List of resources included in this group. All other resources from this service are excluded.</td>
<td>Optional;</td>
</tr>
<tr>
<td>service_resource.not</td>
<td>&lt;object&gt;</td>
<td>Name of the service</td>
<td>Read-only;</td>
</tr>
<tr>
<td>service_resource.not</td>
<td>&lt;any&gt;</td>
<td>Required properties: [all_except, only_include];</td>
<td>Optional;</td>
</tr>
</tbody>
</table>

**Type: user_credentials_assertion**
A username/password assertion

**JSON**

```json
{
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_resource</td>
<td>&lt;object&gt;</td>
<td>Resources available from a service. If only the service_name property is present, all resources are included.</td>
<td>Required properties: [service_name];</td>
</tr>
</tbody>
</table>
```json
{
  "password": string,
  "username": string
}
```

<table>
<thead>
<tr>
<th>Property Name</th>
<th>Type</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>user_credentials_assertion</td>
<td>&lt;object&gt;</td>
<td>A username/password assertion</td>
<td>Required properties: [username, password];</td>
</tr>
<tr>
<td>user_credentials_assertion.password</td>
<td>&lt;string&gt;</td>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>user_credentials_assertion.username</td>
<td>&lt;string&gt;</td>
<td>Username</td>
<td></td>
</tr>
</tbody>
</table>