Connecting to AAF

**Methods to Connect**

* If you are a Servlet in a Container, use CADI Framework with AAF Plugin.  It's very easy, and includes BasicAuth for Services.
* Java Technologies
  + Technologies using Servlet Filters
    - DME2 (and other Servlet Containers) can use Servlet Filters
    - Any WebApp can plug in CADI as a Servlet Filter
    - Jetty can attach a Servlet Filter with Code, or as WebApp
    - Tomcat 7 has a "Valve" plugin, which is similar and supported
  + Use the AAFLur Code directly (shown)
  + All Java Technologies utilize Configuration to set what Security elements are required
    - example: Global Login can be turned on/off, AAF Client needs information to connect to AAF Service
    - There are several specialty cases, which AAF can work with, including embedding all properties in a Web.xml, but the essentials needed are:
      * CADI Jars
      * cadi.properties file (configured the same for all technologies)
      * Encrypt passwords with included CADI technology, so that there are no Clear Text Passwords in Config Files (ASPR)
    - See CADI Deployment on how to perform this with several different technologies.
* AAF Restfully (see RESTFul APIS)

IMPORTANT: If Direct RESTFul API is used, then it is the Client's responsibility to Cache and avoid making an AAF Service Calls too often

Example: A Tool like Cassandra will ask for Authentication hundreds of times a second for the same identity during a transaction.  Calling the AAF Service for each would be slow for the client, and wasteful of Network and AAF Service Capacities.

Rogue Clients ***can and will*** be denied access to AAF.

**J2EE (Servlet Filter) Method**

1. Per J2EE design, the Filter will deny any unauthenticated HTTP/S call; the Servlet will not even be invoked.
   1. Therefore, the Servlet can depend on any transaction making it to their code set is Authenticated.
   2. Identity can be viewed based on the HttpServletRequest Object (request.getUserPrincipal() )
2. Per J2EE design, AAF Filter overloads the HttpServletRequest for a String related to "Role".  (request.isUserInRole("...") )
   1. For AAF, do not put in "Role", but the three parts of requested "Permission", separated by "|", i.e.  "org.onap.aaf.myapp.myperm|myInstance|myAction".
3. NOT REQUIRED: An added benefit, but not required, is a JASPI like interface, where you can add an Annotation to your Servlet.
   1. When used, no transaction will come into your code if the listed Permissions are not Granted to the Incoming Transaction.
   2. This might be helpful for covering separate Management Servlet implementations.

**Servlet Code Snippet**

[?](https://wiki.web.att.com/pages/viewpage.action?spaceKey=aaf&title=Client+Access)

|  |
| --- |
| public void service(ServletRequest req, ServletResponse res) throws ServletException, IOException {      HttpServletRequest request;      try {          request = (HttpServletRequest)req;      } catch (ClassCastException e) {          throw new ServletException("Only serving HTTP today",e);      }        // Note: CADI is OVERLOADING the concept of "isUserInRole".. You need to think "doesUserHavePermssion()"      // Assume that you have CREATED and GRANTED An AAF Permission in YOUR Namespace      // Example Permission:   "org.onap.aaf.myapp.myPerm \* write"        // Think in your head, "Does user have write permission on any instance of org.onap.aaf.myapp.myPerm      if(request.isUserInRole("org.onap.aaf.myapp.myPerm|\*|write")) {          // \*\*\* Do something here that someone with "myPerm write" permissions is allowed to do      } else {          // \*\*\* Do something reasonable if user is denied, like an Error Message      }    } |

Here is a working TestServlet, where you can play with different Permissions that you own on the URL, i.e.:

https://<your machine:port>/caditest/testme?PERM=org.onap.aaf.myapp.myPerm|\*|write

**Sample Servlet (Working example)**

[?](https://wiki.web.att.com/pages/viewpage.action?spaceKey=aaf&title=Client+Access)

|  |
| --- |
| package org.onap.aaf.cadi.debug;  import java.io.FileInputStream;  import java.io.IOException;  import java.net.InetAddress;  import java.net.UnknownHostException;  import java.util.HashMap;  import java.util.Map;  import java.util.Map.Entry;  import java.util.Properties;  import javax.servlet.Servlet;  import javax.servlet.ServletConfig;  import javax.servlet.ServletException;  import javax.servlet.ServletRequest;  import javax.servlet.ServletResponse;  import javax.servlet.http.HttpServletRequest;  import org.eclipse.jetty.server.Server;  import org.eclipse.jetty.server.ServerConnector;  import org.eclipse.jetty.server.handler.ContextHandler;  import org.eclipse.jetty.servlet.FilterHolder;  import org.eclipse.jetty.servlet.FilterMapping;  import org.eclipse.jetty.servlet.ServletContextHandler;  import org.eclipse.jetty.servlet.ServletHandler;  import org.onap.aaf.cadi.filter.CadiFilter;  import org.onap.aaf.cadi.filter.RolesAllowed;  import org.onap.aaf.cadi.jetty.MiniJASPIWrap;    public class CSPServletTest {      public static void main(String[] args) {          // Go ahead and print Test reports in cadi-core first          Test.main(args);          String hostname=null;          try {              hostname = InetAddress.getLocalHost().getHostName();          } catch (UnknownHostException e) {              e.printStackTrace();              System.exit(1);          }          Properties props = new Properties();          Map<String,String> map = new HashMap<String,String>();          try {              FileInputStream fis = new FileInputStream("run/cadi.properties");              try {                  props.load(fis);                  String key,value;                  for( Entry<Object, Object> es  : props.entrySet()) {                      key = es.getKey().toString();                      value = es.getValue().toString();                      map.put(key,value);                      if(key.startsWith("AFT\_") || key.startsWith("DME2")) {                          System.setProperty(key,value);                      }                  }              } finally {                  fis.close();              }          } catch(IOException e) {              System.err.println("Cannot load run/cadi.properties");              System.exit(1);          }          String portStr = System.getProperty("port");          int port = portStr==null?8080:Integer.parseInt(portStr);          try {              // Add ServletHolder(s) and Filter(s) to a ServletHandler              ServletHandler shand = new ServletHandler();                FilterHolder cfh = new FilterHolder(CadiFilter.class);              cfh.setInitParameters(map);                shand.addFilterWithMapping(cfh, "/\*", FilterMapping.ALL);              shand.addServletWithMapping(new MiniJASPIWrap(MyServlet.class),"/\*");              // call initialize after start                ContextHandler ch = new ServletContextHandler();              ch.setContextPath("/caditest");              ch.setHandler(shand);              for( Entry<Object,Object> es : props.entrySet()) {                  ch.getInitParams().put(es.getKey().toString(), es.getValue().toString());              }              //ch.setErrorHandler(new MyErrorHandler());                // Create Server and Add Context Handler              final Server server = new Server();              ServerConnector http = new ServerConnector(server);              http.setPort(port);              server.addConnector(http);              server.setHandler(ch);                // Start              server.start();              shand.initialize();                System.out.println("To test, put http://"+ hostname + ':' + port + "/caditest/testme in a browser or 'curl'");              // if we were really a server, we'd block the main thread with this join...              // server.join();              // But... since we're a test service, we'll block on StdIn              System.out.println("Press <Return> to end service...");              System.in.read();              server.stop();              System.out.println("All done, have a good day!");          } catch (Exception e) {              e.printStackTrace();              System.exit(1);          }      }      @RolesAllowed({"org.onap.aaf.myapp.myPerm|myInstance|myAction"})      public static class MyServlet implements Servlet {          private ServletConfig servletConfig;            public void init(ServletConfig config) throws ServletException {              servletConfig = config;          }            public ServletConfig getServletConfig() {              return servletConfig;          }            public void service(ServletRequest req, ServletResponse res) throws ServletException, IOException {              HttpServletRequest request;              try {                  request = (HttpServletRequest)req;              } catch (ClassCastException e) {                  throw new ServletException("Only serving HTTP today",e);              }                res.getOutputStream().print("<html><header><title>CSP Servlet Test</title></header><body><h1>You're good to go!</h1><pre>" +                      request.getUserPrincipal());                String perm = request.getParameter("PERM");              if(perm!=null)                  if(request.isUserInRole(perm)) {                      if(perm.indexOf('|')<0)                          res.getOutputStream().print("\nCongrats!, You are in Role " + perm);                        else                          res.getOutputStream().print("\nCongrats!, You have Permission " + perm);                  } else {                      if(perm.indexOf('|')<0)                          res.getOutputStream().print("\nSorry, you are NOT in Role " + perm);                        else                          res.getOutputStream().print("\nSorry, you do NOT have Permission " + perm);                  }                res.getOutputStream().print("</pre></body></html>");            }            public String getServletInfo() {              return "MyServlet";          }            public void destroy() {          }      }  } |

**Java Direct (AAFLur) Method**

The AAFLur is the exact component used within all the Plugins mentioned above.  It is written so that it can be called standalone as well, see the Example as follows

package org.onap.aaf.example;

import java.util.ArrayList;

import java.util.List;

import java.util.Properties;

import org.onap.aaf.cadi.Access;

import org.onap.aaf.cadi.Permission;

import org.onap.aaf.cadi.aaf.v2\_0.AAFAuthn;

import org.onap.aaf.cadi.aaf.v2\_0.AAFCon;

import org.onap.aaf.cadi.aaf.v2\_0.AAFLurPerm;

import org.onap.aaf.cadi.config.Config;

import org.onap.aaf.cadi.lur.aaf.AAFPermission;

import org.onap.aaf.cadi.lur.aaf.test.TestAccess;

public class ExamplePerm2\_0 {

public static void main(String args[]) {

// Normally, these should be set in environment. Setting here for clarity

Properties props = System.getProperties();

props.setProperty("AFT\_LATITUDE", "32.780140");

props.setProperty("AFT\_LONGITUDE", "-96.800451");

props.setProperty("AFT\_ENVIRONMENT", "AFTUAT");

props.setProperty(Config.AAF\_URL,

// "https://DME2RESOLVE/service=org.onap.aaf.authz.AuthorizationService/version=2.0/envContext=DEV/routeOffer=BAU\_SE"

"https://DME2RESOLVE/service=org.onap.aaf.authz.AuthorizationService/version=2.0/envContext=TEST/routeOffer=BAU\_SE"

);

props.setProperty(Config.AAF\_USER\_EXPIRES,Integer.toString(5\*60000)); // 5 minutes for found items to live in cache

props.setProperty(Config.AAF\_HIGH\_COUNT,Integer.toString(400)); // Maximum number of items in Cache);

props.setProperty(Config.CADI\_KEYFILE,"keyfile"); //Note: Be sure to generate with java -jar <cadi\_path>/lib/cadi-core\*.jar keygen keyfile

// props.setProperty("DME2\_EP\_REGISTRY\_CLASS","DME2FS");

// props.setProperty("AFT\_DME2\_EP\_REGISTRY\_FS\_DIR","../../authz/dme2reg");

// Link or reuse to your Logging mechanism

Access myAccess = new TestAccess(); //

//

try {

AAFCon<?> con = new AAFConDME2(myAccess);

// AAFLur has pool of DME clients as needed, and Caches Client lookups

AAFLurPerm aafLur = con.newLur();

// Note: If you need both Authn and Authz construct the following:

AAFAuthn<?> aafAuthn = con.newAuthn(aafLur);

// Do not set Mech ID until after you construct AAFAuthn,

// because we initiate "401" info to determine the Realm of

// of the service we're after.

con.basicAuth("xxxx@aaf.abc.com", "XXXXXX");

try {

// Normally, you obtain Principal from Authentication System.

// For J2EE, you can ask the HttpServletRequest for getUserPrincipal()

// If you use CADI as Authenticator, it will get you these Principals from

// CSP or BasicAuth mechanisms.

String id = "xxxx@aaf.abc.com"; //"cluster\_admin@gridcore.abc.com";

// If Validate succeeds, you will get a Null, otherwise, you will a String for the reason.

String ok = aafAuthn.validate(id, "XXXXXX");

if(ok!=null)System.out.println(ok);

ok = aafAuthn.validate(id, "wrongPass");

if(ok!=null)System.out.println(ok);

// AAF Style permissions are in the form

// Type, Instance, Action

AAFPermission perm = new AAFPermission("org.onap.aaf.grid.core.coh",":dev\_cluster", "WRITE");

// Now you can ask the LUR (Local Representative of the User Repository about Authorization

// With CADI, in J2EE, you can call isUserInRole("org.onap.aaf.mygroup|mytype|write") on the Request Object

// instead of creating your own LUR

System.out.println("Does " + id + " have " + perm);

if(aafLur.fish(id, perm)) {

System.out.println("Yes, you have permission");

} else {

System.out.println("No, you don't have permission");

}

System.out.println("Does Bogus have " + perm);

if(aafLur.fish("Bogus", perm)) {

System.out.println("Yes, you have permission");

} else {

System.out.println("No, you don't have permission");

}

// Or you can all for all the Permissions available

List<Permission> perms = new ArrayList<Permission>();

aafLur.fishAll(id,perms);

for(Permission prm : perms) {

System.out.println(prm.getKey());

}

// It might be helpful in some cases to clear the User's identity from the Cache

aafLur.remove(id);

} finally {

aafLur.destroy();

}

} catch (Exception e) {

e.printStackTrace();

}

}

}

There are two current AAF Lurs which you can utilize:

* **Org.onap.aaf.cadi.aaf.v2\_0.AAFLurPerm** is the **default**, and will fish based on the Three-fold "Permission" standard in AAF

To run this code, you will need from a SWM deployment (org.onap.aaf.cadi:cadi, then soft link to jars needed):

* cadi-core-<version>.jar
* cadi-aaf-<version>-full.jar

   or by Maven

<dependency>

<groupId>org.onap.aaf.cadi</groupId>  
<artifactId>cadi-aaf</artifactId>  
<version>THE\_LATEST\_VERSION</version>  
<classifier>full</classifier>

</dependency>

   If you need the Java Client definitions only,

   Also needed are the DME2 Client libraries:

* dme2-<version>.jar
* discovery-clt-<version>.jar

**RESTFul API**

* RESTful APIs are the core of AAF Service.  If you have Authentication and Authorization, your identity can also utilize the RESTful API as is, without java or other specific clients.
* As stated above, use of the RESTful API means that the Client is responsible to cover caching for repeated calls.
  + If you choose the Direct API route, and your service does not behave responsibly, **it can and will be turned off** at runtime.
* To use the API, please read and follow the Current API documentation with the RESTful mechanism of your choice.
  + Some use RESTful api with Browser tools, like "RESTClient" on Firefox
  + JavaScript users, until there are special Java Script Clients created, will need to access with JavaScript APIs.
    - Note: AAF supports and recommends JSON for interactions
* Notes: Do not neglect to set the "Content-Type" or "Accepts" Properties in your JavaScript calls according to Documentation.

|  |  |  |  |
| --- | --- | --- | --- |
| AAF API Reference | | | |
| **Entity** | **Method** | **Path Info** | **Description** |
| PERMISSION | POST | /authz/perm | Create a Permission  Permission consists of:   * type - a Namespace qualified identifier specifying what kind of resource is being protected * instance - a key, possibly multi-dimensional, that identifies a specific instance of the type * action - what kind of action is allowed   Note: instance and action can be an \*  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/PermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/PermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | PUT | /authz/perm | Set Description for Permission  Add Description Data to Perm  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/PermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/PermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | DELETE | /authz/perm | Delete a Permission  Delete the Permission referenced by PermKey.  You cannot normally delete a permission which is still granted to roles,  however the "force" property allows you to do just that. To do this: Add  'force=true' as a query parameter.  WARNING: Using force will ungrant this permission from all roles. Use with care.  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/PermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/PermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | DELETE | /authz/perm/:name/:type/:action | Delete a Permission  Delete the Permission referenced by :type :instance :action  You cannot normally delete a permission which is still granted to roles,  however the "force" property allows you to do just that. To do this: Add  'force=true' as a query parameter  WARNING: Using force will ungrant this permission from all roles. Use with care.  Parameters  type : string (Required)  instance : string (Required)  action : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/PermKey+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybUtleStqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/PermKey+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybUtleSt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | PUT | /authz/perm/:type/:instance/:action | Update a Permission  Rename the Permission referenced by :type :instance :action, and rename (copy/delete) to the Permission described in PermRequest  Parameters  type : string (Required)  instance : string (Required)  action : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406, 409  ContentType:  [application/PermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/PermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/perms/:type | Get Permissions by Type  List All Permissions that match the :type element of the key  Parameters  type : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/perms/:type/:instance/:action | Get Permissions by Key  List Permissions that match key; :type, :instance and :action  Parameters  type : string (Required)  instance : string (Required)  action : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/perms/ns/:ns | Get PermsByNS  List All Permissions that are in Namespace :ns  Parameters  ns : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MC4y)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/perms/role/:role | Get Permissions by Role  List All Permissions that are granted to :role  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | POST | /authz/perms/user/:user | Get Permissions by User, Query AAF Perms  List All Permissions that match user :user  'user' must be expressed as full identity (ex: id@full.domain.com)  Present Queries as one or more Permissions (see ContentType Links below for format).  If the Caller is Granted this specific Permission, and the Permission is valid  for the User, it will be included in response Permissions, along with  all the normal permissions on the 'GET' version of this call. If it is not  valid, or Caller does not have permission to see, it will be removed from the list  \*Note: This design allows you to make one call for all expected permissions  The permission to be included MUST be:  .access|:[:key]|  examples:  org.onap.aaf.myns.access|:ns|write  org.onap.aaf.myns.access|:role:myrole|create  org.onap.aaf.myns.access|:perm:mytype:myinstance:myaction|read  Parameters  user : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/perms/user/:user | Get Permissions by User  List All Permissions that match user :user  'user' must be expressed as full identity (ex: id@full.domain.com)  Parameters  user : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Perms+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MC4y)  [application/Perms+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUGVybXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
| ROLE | POST | /authz/role | Create Role  Roles are part of Namespaces  Examples:   * org.onap.aaf - The team that created and maintains AAF * org.onap.aaf.csp - The team that created Global Login   Roles do not include implied permissions for an App. Instead, they contain explicit Granted Permissions by any Namespace in AAF (See Permissions)  Restrictions on Role Names:   * Must start with valid Namespace name, terminated by . (dot/period) * Allowed Characters are a-zA-Z0-9.\_- * role names are Case Sensitive   The right questions to ask for defining and populating a Role in AAF, therefore, are:   * 'What Job Function does this represent?' * 'Does this person perform this Job Function?'   Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/RoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/RoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | PUT | /authz/role | Set Description for role  Add Description Data to a Role  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/RoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/RoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | DELETE | /authz/role | Delete Role  Delete the Role referenced by RoleKey  You cannot normally delete a role which still has permissions granted or users assigned to it,  however the "force" property allows you to do just that. To do this: Add 'force=true'  as a query parameter.  WARNING: Using force will remove all users and permission from this role. Use with care.  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/RoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/RoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | DELETE | /authz/role/:role | Delete Role  Delete the Role named :role  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/Role+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZStqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Role+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZSt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | DELETE | /authz/role/:role/perm | Delete Permission from Role  Ungrant a permission from Role :role  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  ContentType:  [application/RolePermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/RolePermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | POST | /authz/role/perm | Add Permission to Role  Grant a Permission to a Role  Permission consists of:   * type - a Namespace qualified identifier specifying what kind of resource is being protected * instance - a key, possibly multi-dimensional, that identifies a specific instance of the type * action - what kind of action is allowed   Note: instance and action can be an \*  Note: Using the "force" property will create the Permission, if it doesn't exist AND the requesting ID is allowed to create. It will then grant  the permission to the role in one step. To do this: add 'force=true' as a query parameter.  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/RolePermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/RolePermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | PUT | /authz/role/perm | Set a Permission's Roles  Set a permission's roles to roles given  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/RolePermRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/RolePermRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZVBlcm1SZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/roles/:role | GetRolesByFullName  List Roles that match :role  Note: You must have permission to see any given role  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Roles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Roles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/roles/name/:name | GetRolesByNameOnly  List all Roles for only the Name of Role (without Namespace)  Note: You must have permission to see any given role  Parameters  name : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Roles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Roles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/roles/ns/:ns | GetRolesByNS  List all Roles for the Namespace :ns  Note: You must have permission to see any given role  Parameters  ns : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Roles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MC4y)  [application/Roles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/roles/perm/:type/:instance/:action | GetRolesByPerm  Find all Roles containing the given Permission.Permission consists of:   * type - a Namespace qualified identifier specifying what kind of resource is being protected * instance - a key, possibly multi-dimensional, that identifies a specific instance of the type * action - what kind of action is allowed   Notes: instance and action can be an \*  You must have permission to see any given role  Parameters  type : string (Required)  instance : string (Required)  action : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Roles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Roles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/roles/user/:name | GetRolesByUser  List all Roles that match user :name  'user' must be expressed as full identity (ex: id@full.domain.com)  Note: You must have permission to see any given role  Parameters  name : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Roles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Roles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vUm9sZXMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | POST | /authz/userRole | Request User Role Access  Create a UserRole relationship (add User to Role)  A UserRole is an object Representation of membership of a Role for limited time.  If a shorter amount of time for Role ownership is required, use the 'End' field.  \*\* Note: Owners of Namespaces will be required to revalidate users in these roles  before Expirations expire. Namespace owners will be notified by email.  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/UserRoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/UserRoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/userRole/:user/:role | Get if User is In Role  Returns the User (with Expiration date from listed User/Role) if it exists  Parameters  user : string (Required)  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  Accept:  [application/Users+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Users+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | DELETE | /authz/userRole/:user/:role | Delete User Role  Remove Role :role from User :user.  Parameters  user : string (Required)  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | PUT | /authz/userRole/extend/:user/:role | Extend Expiration  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | PUT | /authz/userRole/role | Update Users for a role  Set a Role's users to the users specified in the UserRoleRequest object.  WARNING: Users supplied will be the ONLY users attached to this role  If no users are supplied, role's users are reset.  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  ContentType:  [application/UserRoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/UserRoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | PUT | /authz/userRole/user | Update Roles for a user  Set a User's roles to the roles specified in the UserRoleRequest object.  WARNING: Roles supplied will be the ONLY roles attached to this user  If no roles are supplied, user's roles are reset.  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  ContentType:  [application/UserRoleRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/UserRoleRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVSZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/userRoles/role/:role | Get UserRoles by Role  List all Users that are attached to Role specified in :role  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/UserRoles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTAuMg==)  [application/UserRoles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/userRoles/user/:user | Get UserRoles by User  List all UserRoles for :user  Parameters  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/UserRoles+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTAuMg==)  [application/UserRoles+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlclJvbGVzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
| NAMESPACE | POST | /authz/ns | Create a Namespace  Namespace consists of:   * name - What you want to call this Namespace * responsible(s) - Person(s) who receive Notifications and approves Requests   regarding this Namespace. Companies have Policies as to who may take on  this Responsibility. Separate multiple identities with commas   * admin(s) - Person(s) who are allowed to make changes on the namespace,   including creating Roles, Permissions and Credentials. Separate multiple  identities with commas  Note: Namespaces are dot-delimited (i.e. com.myCompany.myApp) and must be  created with parent credentials (i.e. To create com.myCompany.myApp, you must  be an admin of com.myCompany or com  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/NsRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNSZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/NsRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNSZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | PUT | /authz/ns | Set a Description for a Namespace  Replace the Current Description of a Namespace with a new one  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  ContentType:  [application/NsRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNSZXF1ZXN0K2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/NsRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNSZXF1ZXN0K3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | DELETE | /authz/ns/:ns | Delete a Namespace  Delete the Namespace :ns. Namespaces cannot normally be deleted when there  are still credentials associated with them, but they can be deleted by setting  the "force" property. To do this: Add 'force=true' as a query parameter  WARNING: Using force will delete all credentials attached to this namespace. Use with care.  if the "force" property is set to 'force=move', then Permissions and Roles are not deleted,but are retained, and assigned to the Parent Namespace. 'force=move' is not permitted at or below Application Scope  Parameters  ns : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 424  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | POST | /authz/ns/:ns/admin/:id | Add an Admin to a Namespace  Add an Identity :id to the list of Admins for the Namespace :ns  Note: :id must be fully qualified.  Parameters  ns : string (Required)  id : string (Required)  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | DELETE | /authz/ns/:ns/admin/:id | Remove an Admin from a Namespace  Remove an Identity :id from the list of Admins for the Namespace :ns  Note: :id must be fully qualified .  Parameters  ns : string (Required)  id : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | DELETE | /authz/ns/:ns/attrib/:key | delete an Attribute from a Namespace  Delete an attribute in the Namespace  You must be given direct permission for key by AAF  Parameters  ns : string (Required)  key : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | POST | /authz/ns/:ns/attrib/:key/:value | Add an Attribute from a Namespace  Create an attribute in the Namespace  You must be given direct permission for key by AAF  Parameters  ns : string (Required)  key : string (Required)  value : string (Required)  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | PUT | /authz/ns/:ns/attrib/:key/:value | update an Attribute from a Namespace  Update Value on an existing attribute in the Namespace  You must be given direct permission for key by AAF  Parameters  ns : string (Required)  key : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | POST | /authz/ns/:ns/responsible/:id | Add a Responsible Identity to a Namespace  Add an Identity :id to the list of Responsibles for the Namespace :ns  Note: :id must be fully qualified .  Parameters  ns : string (Required)  id : string (Required)  Expected HTTP Code  201  Explicit HTTP Error Codes  403, 404, 406, 409  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | DELETE | /authz/ns/:ns/responsible/:id | Remove a Responsible Identity from Namespace  Remove an Identity :id to the list of Responsibles for the Namespace :ns  Note: A namespace must have at least 1 responsible party  Parameters  ns : string (Required)  id : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  ContentType:  [application/Void+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCtqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Void+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVm9pZCt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | GET | /authz/ns/attrib/:key | get Ns Key List From Attribute  Read Attributes for Namespace  Parameters  key : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  Accept:  [application/Keys+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vS2V5cytqc29uO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsYXBwbGljYXRpb24vanNvbjtxPTEuMDt2ZXJzaW9uPTIuMCwqLyo7cT0xLjA=)  [application/Keys+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vS2V5cyt4bWw7cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCx0ZXh0L3htbDtxPTEuMDt2ZXJzaW9uPTIuMA==) |
|  | GET | /authz/nss/:id | Return Information about Namespaces  Lists the Admin(s), Responsible Party(s), Role(s), Permission(s)  Credential(s) and Expiration of Credential(s) in Namespace :id  Parameters  id : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Nss+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/Nss+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/nss/admin/:user | Return Namespaces where User is an Admin  Accept:  [application/Nss+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/Nss+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/nss/children/:id | Return Child Namespaces  Lists all Child Namespaces of Namespace :id  Note: This is not a cached read  Parameters  id : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404  Accept:  [application/Nss+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/Nss+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/nss/either/:user | Return Namespaces where User Admin or Owner  Accept:  [application/Nss+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=0.2](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTAuMg==)  [application/Nss+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
|  | GET | /authz/nss/responsible/:user | Return Namespaces where User is Responsible  Accept:  [application/Nss+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK2pzb247cT0xLjA7Y2hhcnNldD11dGYtODt2ZXJzaW9uPTIuMCxhcHBsaWNhdGlvbi9qc29uO3E9MS4wO3ZlcnNpb249Mi4wLCovKjtxPTEuMA==)  [application/Nss+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vTnNzK3htbDtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLHRleHQveG1sO3E9MS4wO3ZlcnNpb249Mi4w) |
| USER | GET | /authn/basicAuth | Is given BasicAuth valid?  !!!! DEPRECATED without X509 Authentication STOP USING THIS API BY DECEMBER 2017, or use Certificates !!!! Use /authn/validate instead Note: Validate a Password using BasicAuth Base64 encoded Header. This HTTP/S call is intended as a fast User/Password lookup for Security Frameworks, and responds 200 if it passes BasicAuth security, and 403 if it does not.  Expected HTTP Code  200  Explicit HTTP Error Codes  403  Accept:  text/plain |
|  | POST | /authn/validate | Is given Credential valid?  Validate a Credential given a Credential Structure. This is a more comprehensive validation, can do more than BasicAuth as Credential types exp  Expected HTTP Code  200  Explicit HTTP Error Codes  403  ContentType:  [application/CredRequest+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vQ3JlZFJlcXVlc3QranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/CredRequest+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vQ3JlZFJlcXVlc3QreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/users/:user/:role | Get if User is In Role  Accept:  [application/Users+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Users+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/users/perm/:type/:instance/:action | Get Users By Permission  List all Users that have Permission specified by :type :instance :action  Parameters  type : string (Required)  instance : string (Required)  action : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  404, 406  Accept:  [application/Users+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Users+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |
|  | GET | /authz/users/role/:role | Get Users By Role  Returns the User (with Expiration date from listed User/Role) if it exists  Parameters  user : string (Required)  role : string (Required)  Expected HTTP Code  200  Explicit HTTP Error Codes  403, 404, 406  Accept:  [application/Users+json;q=1.0;charset=utf-8;version=2.0,application/json;q=1.0;version=2.0,\*/\*;q=1.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMranNvbjtxPTEuMDtjaGFyc2V0PXV0Zi04O3ZlcnNpb249Mi4wLGFwcGxpY2F0aW9uL2pzb247cT0xLjA7dmVyc2lvbj0yLjAsKi8qO3E9MS4w)  [application/Users+xml;q=1.0;charset=utf-8;version=2.0,text/xml;q=1.0;version=2.0](https://zlp14185.vci.att.com:8200/gui/example/YXBwbGljYXRpb24vVXNlcnMreG1sO3E9MS4wO2NoYXJzZXQ9dXRmLTg7dmVyc2lvbj0yLjAsdGV4dC94bWw7cT0xLjA7dmVyc2lvbj0yLjA=) |