

# Active System Manager Release 8.1.1 API Reference Guide



# Notes, cautions, and warnings

 **NOTE:** A NOTE indicates important information that helps you make better use of your computer.

 **CAUTION:** A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

 **WARNING:** A WARNING indicates a potential for property damage, personal injury, or death.

**Copyright © 2015 Dell Inc. All rights reserved.** This product is protected by U.S. and international copyright and intellectual property laws. Dell™ and the Dell logo are trademarks of Dell Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

2015 - 09

Rev. A00

# Contents

<b>1 Overview.....</b>	<b>8</b>
Other Documents You May Need.....	8
<b>2 Resource URI Endpoints.....</b>	<b>10</b>
<b>3 Authentication Headers.....</b>	<b>11</b>
<b>4 Generic Query Parameters.....</b>	<b>13</b>
Dates in query strings.....	13
<b>5 HTTP Message Bodies.....</b>	<b>14</b>
<b>6 Usage Examples.....</b>	<b>15</b>
Timezone.....	15
Get the current timezone.....	15
Set the current timezone.....	16
Credential.....	16
Get all defined Credentials.....	16
Define new Credential.....	16
DiscoveryRequest.....	17
Discover Chassis and contained Blade Servers.....	17
ManagedDevice.....	18
Get all Managed Devices.....	18
ServiceTemplate.....	18
Get all ServiceTemplates.....	18
Get a ServiceTemplate by name.....	19
Publish a ServiceTemplate.....	19
Deployment.....	20
Get all deployed Services.....	20
Deploy a new Service.....	20
Tear down a Service.....	21
Firmware.....	21
Check Firmware compliance of all Resources.....	21
<b>7 Exception Handling.....</b>	<b>23</b>
<b>8 Appendix A - API Reference.....</b>	<b>24</b>
/Authenticate/ POST.....	24

/Chassis/ GET.....	24
/Chassis/{refId} PUT.....	26
/Chassis/{refId} GET.....	26
/Chassis/{refId} DELETE.....	27
/Chassis/{refId}/sensorlog DELETE.....	28
/Chassis/{refId}/sensorlog GET.....	28
/Chassis/{serviceTag}/{type} GET.....	29
/Chassis/{serviceTag}/{type} DELETE.....	30
/Chassis/iom/ip/{ip} GET.....	30
/Credential/ POST.....	31
/Credential/ GET.....	32
/Credential/{id} PUT.....	33
/Credential/{id} DELETE.....	34
/Credential/{id} PUT.....	34
/Deployment/ POST.....	35
/Deployment/ GET.....	36
/Deployment/{id} PUT.....	37
/Deployment/{id} GET.....	37
/Deployment/{id} DELETE.....	38
/Deployment/migrate/{serviceId}/{serverId} PUT.....	39
/DiscoveryRequest/ GET.....	39
/DiscoveryRequest/ POST.....	40
/DiscoveryRequest/{id} GET.....	41
/DiscoveryRequest/{id} DELETE.....	41
/DiscoveryRequest/chassislist POST.....	42
/DiscoveryRequest/discoveryresult/{id} GET.....	43
/ManagedDevice/ GET.....	43
/ManagedDevice/ POST.....	45
/ManagedDevice/{refId} DELETE.....	45
/ManagedDevice/{refId} PUT.....	46
/ManagedDevice/{refId} GET.....	47
/ManagedDevice/count GET.....	47
/ManagedDevice/firmware PUT.....	48
/ManagedDevice/puppet/{certName} GET.....	49
/ManagedDevice/puppet/{certName} PUT.....	49
/Network/ GET.....	50
/Network/ POST.....	52
/Network/{networkId} DELETE.....	52
/Network/{networkId} PUT.....	53
/Network/{networkId} GET.....	54
/Network/ipAddress/assign PUT.....	54
/Network/ipAddress/release PUT.....	55

/NTP/ DELETE.....	56
/NTP/ PUT.....	56
/NTP/ GET.....	57
/Proxy/ GET.....	58
/Proxy/ PUT.....	58
/Proxy/test POST.....	59
/ServerPool/ POST.....	59
/ServerPool/ GET.....	60
/ServerPool/{poolId} DELETE.....	62
/ServerPool/{poolId} GET.....	62
/ServerPool/{poolId} PUT.....	63
/ServerPool/{poolId}/generateVirtualIdentity PUT.....	64
/ServiceTemplate/ GET.....	65
/ServiceTemplate/ POST.....	66
/ServiceTemplate/{id} GET.....	67
/ServiceTemplate/{id} DELETE.....	67
/ServiceTemplate/{id} PUT.....	68
/ServiceTemplate/{id}/copy POST.....	69
/ServiceTemplate/{id}/mapToPhysicalResources POST.....	69
/ServiceTemplate/device/{deviceId} GET.....	70
/ServiceTemplate/export POST.....	71
/ServiceTemplate/upload POST.....	72
/ServiceTemplate/upload POST.....	72
/Timezone/ GET.....	73
/Timezone/ PUT.....	74
/Timezone/all GET.....	74

**9 Appendix B - Model Reference..... 76**

asmCredential .....	76
AsmDetailedMessage .....	76
AsmDetailedMessageList .....	77
AuthenticateRequest .....	77
AuthenticateResponse .....	77
availableTimeZones .....	77
Chassis .....	78
ChassisController .....	79
ChassisLogEntry .....	79
ChassisPowerSupply .....	80
ComparatorValue .....	80
CompliantState .....	80
component .....	80
credential .....	81

credentialList .....	82
Deployment .....	82
DeploymentDevice .....	83
DeploymentFilterResponse .....	84
DeploymentStatusType .....	84
DeviceDiscoveryRequest .....	84
DeviceDiscoveryRequests .....	85
DeviceGroup .....	85
DeviceGroupList .....	85
DeviceHealth .....	86
DeviceState .....	86
DeviceType .....	86
dhcpSettings .....	87
DiscoverDeviceType .....	87
DiscoveredDevices .....	88
DiscoveryRequest .....	89
DiscoveryResult .....	89
DiscoveryStatus .....	90
DiskMediaType .....	90
Fabric .....	90
FirmwareComplianceComponents .....	91
FirmwareDeviceInventory .....	91
FirmwareInventory .....	92
FirmwareRepository .....	92
FirmwareUpdateRequest .....	93
GregorianCalendar .....	94
GroupUser .....	94
GroupUserList .....	94
Health .....	94
IKVM .....	95
Interface .....	95
IOM .....	95
IOMLocationType .....	96
IpRange .....	96
Link .....	96
List .....	97
logEntry .....	97
LogSeverity .....	97
ManagedDevice .....	98
MigrationRequest .....	99
Network .....	99
NetworkConfiguration .....	99

ntpSettings .....	100
Partition .....	100
proxySettings .....	101
RAIDConfiguration .....	101
Reason .....	101
references .....	102
RejectedServer .....	102
RepositoryStatus .....	102
SelectedServer .....	102
Server .....	103
ServerSlotType .....	103
ServiceTemplate .....	103
ServiceTemplateCategory .....	104
ServiceTemplateComponentType .....	105
ServiceTemplateOption .....	105
ServiceTemplateSetting .....	105
ServiceTemplateSettingType .....	106
ServiceTemplateUploadRequest .....	106
ServiceTemplateValid .....	107
SoftwareBundle .....	107
SoftwareComponent .....	108
StaticNetworkConfiguration .....	109
testProxyResponse .....	109
timeZone .....	109
UIRaidLevel .....	109
UpdateType .....	110
User .....	110
VirtualDisk .....	111
VirtualDiskConfiguration .....	111
VM .....	111
wizardStatus .....	112

**10 Appendix C -Ruby Code Usage..... 113**

# Overview

This document is intended for a technical audience. It is assumed that the reader is familiar with REST, and programmatic interaction with REST APIs. In reality, any programming languages can be used with these APIs, however, the code examples contained in this guide are written in Ruby. XML is also used extensively for these examples.

Dell Active System Manager (**ASM**) is an acronym for Dell Active System Manager. ASM is intended to automate the processes involved in provisioning, configuring, and managing bare metal hardware: chassis, servers, switches, and network storage devices. ASM supports processes such as automated discovery of hardware, automated installation of operating systems, creation of virtual networks, and firmware management.

Two essential concepts in ASM are, *ServiceTemplate* and *Deployment process*. A *ServiceTemplate* describes a blueprint for provisioning and configuring a collection of devices as a named unit, also known as a *Service*. Deployment is the process of utilizing a *ServiceTemplate* to physically realize what service the *ServiceTemplate* describes. While the names *Deployment* and *Service* are synonymous, ASM-UI uses the term *Service* while the REST API uses the term *Deployment*.

ASM is distributed as a virtual appliance running a collection of interacting web applications that communicate through the REST interfaces. An additional web application, ASM-UI, also residing on the appliance, provides full browser-based GUI access to ASM. This ASM-UI interacts with ASM through the same REST interfaces described in this guide.

It is possible to effectively integrate ASM with any custom UI or access it through programs for better automation experiences.

In this API guide, we have described how to use these APIs with appropriate examples.

A full reference to all of the available REST methods is explained in [Appendix A](#).

 **NOTE:** Most of these REST methods exchange complex data models within the HTTP requests and responses, which can be represented as either XML or JSON depending on the preference. The structure of these models is contained in [Appendix B](#) for further reference.

## Other Documents You May Need

Go to <http://www.dell.com/asmdocs> for additional supporting documents such as:

- *Active System Manager Release 8.1.1 User's Guide*
- *Active System Manager Release 8.1.1 Installation Guide*
- *Active System Manager Release 8.1.1 Compatibility Matrix*
- *Active System Manager Release 8.1.1 Release Notes*



For more information about ASM, including how-to videos, whitepapers, and blogs, see the Active System Manager page on Dell TechCenter:

<http://www.dell.com/asmtechcenter>

## Resource URI Endpoints

An example of resource endpoint is: **https://Api/V1/ServiceTemplate**

From this example, API endpoints are accessible over https (at port 443 ).

All URI paths are prefixed with /Api/V1/ where V1 is the current API version.

Following the prefix above the remainder of the URI path (endpoint) will have the form /<resource>/<action>?<query>.

Where action and query are optional, and resource is one of the following resources:

- Authenticate
- Chassis
- Credential
- Deployment
- DeviceGroup
- DHCP
- DiscoveryRequest
- FirmwareRepository
- ManagedDevice
- Network
- NTP
- Proxy
- Server
- ServiceTemplate
- Timezone
- User
- WizardStatus

Each of these resources is fully detailed in Appendix A, showing the applicable HTTP methods for each, optional actions and query parameters, and relevant payloads for POST and PUT methods.

## Authentication Headers

All API requests require 3 custom HTTP headers for authentication.

- X-dell-auth-key
- X-dell-auth-signature
- X-dell-auth-timestamp

In order to compute the authentication headers, first a REST call must be made to the /Authenticate endpoint. The Full URI is **/Api/V1/Authenticate**.

This call must be a POST request supplying either an XML or JSON payload. For illustration, the XML payload will follow the structure below providing the user name and password:

```
<AuthenticateRequest>
  <userName>admin</userName>
  <domain>ASMLLOCAL</domain>
  <password>abc123</password>
</AuthenticateRequest>
```

This call must be a POST request supplying either an XML or JSON payload. For illustration, the XML payload will follow the structure below providing the user name and password:

Depending on the Accept header, the response will contain either an XML or JSON payload. For illustration we show the XML version:

```
<AuthenticateResponse>
  <userName>admin</userName>
  <domain>ASMLLOCAL</domain>
  <role>Administrator</role>
  <apiKey>34b3577f7c3c03174a9a506b</apiKey>
  <apiSecret>9a6d9692ba64142e6a1934f9be994f3b0ae63959a6132c8b</apiSecret>
</AuthenticateResponse>
```


The returned apiKey and apiSecret from the XML response above will then be used to generate the security headers. The method is to concatenate 5 values and then compute a digest using the apiSecret from above. For illustration, a Ruby implementation of the generated headers is shown. This relies on the base64 and openssl gems to compute the signature and is extracted from the supplied Ruby ASM module discussed further in [Appendix C](#).

```
apiKey = '34b3577f7c3c03174a9a506b'
apiSecret = '9a6d9692ba64142e6a1934f9be994f3b0ae63959a6132c8b'
timestamp = Time.now.to_i.to_s

# Concatenate the following values
requestString = "%s:%s:%s:%s:%s" %
[apiKey, httpMethod, uriPath, userAgent, timestamp]

# Compute a digest on concatenated string using apiSecret
hash_str = OpenSSL::HMAC.digest('sha256', apiSecret, requestString)
signature = Base64.strict_encode64(hash_str)
```

```
headers['x-dell-auth-key'] = apiKey
headers['x-dell-auth-signature'] = signature
headers['x-dell-auth-timestamp'] = timestamp
```

 **NOTE:** Due to the timestamp component, authentication of future requests may fail if the 2 systems communicating with each other have system clocks that are not well synchronized. So, this is essential before anything else, to synchronize the system clocks of 2 different systems.

## Generic Query Parameters

Many GET methods which return a list of objects support generic query parameters for filtering, sorting, and pagination. These methods are indicated in the API reference section but the detailed syntax is described here.

- **filter**=<operator>,<valid filter column>,<value>,<value2>...
- operator is one of ( eq, co )
- where valid filter columns are listed for each supporting method in the API reference
- Examples:
  - "eq,columnName,columnValue"  
To include resources where columnName is columnValue.
  - "eq,columnName,columnValue,columnValue2[,columnName3...]"  
To include resources where columnName is columnValue OR columnValue2.
  - "eq,columnName,"  
To include resources where columnName is empty.
  - "eq,columnName,,columnValue[,columnName2...]"  
To include resources where columnName is empty OR columnValue.
  - "co,columnName,columnValue"

To include resources where columnName *contains* columnValue:

- **sort**=[-]<valid sort column>, where valid sort columns are listed for each supporting method in the API reference and a minus sign preceding the column name indicates descending order
- **offset**= <integer> default is 0
- **limit**= <integer> default is 50

## Dates in query strings

Some of the columns that are sortable or filterable are dates such as *createdDate*. When using the filter parameter with date strings, only the *eq* operator is valid. The format for these date strings is:

*dd-MM-yyyy*

example: *30-07-2015*

## HTTP Message Bodies

All HTTP Requests and Responses support both XML and JSON content types. The examples used throughout this document show HTTP body content in XML form. The represented data models accepted and returned by the API calls can be quite complex. The structure of these data models is detailed in Appendix B in a language independent representation. While it is possible to craft say an XML request body from this reference documentation, a much more convenient method is to GET an existing object from ASM in the desired representation ( XML or JSON ) and save it to a file for use as a skeleton for future requests involving that model type.

Since a fresh system will not have any existing Deployments. A Deployment model is shown below for illustration. It is actually only a skeleton because it is lacking the essential ServiceTemplate content that would make it useful. For more information see the Usage Example called *Deploy a Service*.

```
<?xml version="1.0" encoding="UTF-8"?>
  <Deployment>
    <deploymentName>Ottomatic</deploymentName>
    <numberOfDeployments>1</numberOfDeployments>
    <deploymentDescription>test REST deployment</deploymentDescription>
    <allUsersAllowed>>false</allUsersAllowed>
    <canCancel>>false</canCancel>
    <canDelete>>false</canDelete>
    <canDeleteResources>>false</canDeleteResources>
    <canEdit>>false</canEdit>
    <canMigrate>>false</canMigrate>
    <canRetry>>false</canRetry>
    <canScaleupApplication>>false</canScaleupApplication>
    <canScaleupCluster>>false</canScaleupCluster>
    <canScaleupServer>>false</canScaleupServer>
    <canScaleupStorage>>false</canScaleupStorage>
    <canScaleupVM>>false</canScaleupVM>
    <updateServerFirmware>>false</updateServerFirmware>
    <individualTeardown>>false</individualTeardown>
    <teardown>>false</teardown>
    <retry>>false</retry>
    <serviceTemplate>
    </serviceTemplate>
  </Deployment>
```

## Usage Examples

The following examples use a simple Ruby module called ASM which utilizes the publicly available RestClient gem to perform the HTTP requests. This module contains utilities to generate the security headers so we can focus on the API calls themselves. See Appendix C for detailed information on using and configuring the ASM module.

In addition to the API requests, there is sometimes significant XML processing required to prepare the request bodies or parse the responses. A utility module ASM::Payload is provided for simple get/set operations on XML document elements and simple load/save operations on XML files for use as starting templates and intermediate data storage between API calls. This module is not required but is useful for simplifying the example code that follows.

### Timezone

#### Get the current timezone

```
require 'ASMConfig'

# Get timezone settings
url = ASM::API::URI("/Timezone")

response = ASM::API::sign {
  RestClient.get url
}
# Save it for future use as a payload
payload = ASM::Payload.from_xml(response)
payload.save_xml('timezone.xml')
```

Here we utilize a block construct to sign our request. All RestClient invocations inside the sign block will automatically be signed. The url is specified simply as '/Timezone' because ASM has already been configured with the base path prefix /Asm/V1. The response object as XML is:

```
<timeZone>
  <timeZone>(UTC-06:00) Central Time (US & Canada)</timeZone>
  <timeZoneId>11</timeZoneId>
</timeZone>
```

which we saved in the file 'timezone.xml' as a payload used in the next example to set the timezone.

## Set the current timezone

Here we utilize the simple XML utility class called `ASM::Payload` to do simple manipulations of our XML.

```
require 'ASMConfig'

payload = ASM::Payload.load_xml('timezone.xml')

payload.set('timeZoneId','15')

# Set Timezone
response = ASM::API::sign {
  url = ASM::API::URI("/Timezone")
  RestClient.put url, payload.to_xml, :content_type => :xml
}
```

## Credential

Credential objects can be created and referenced in ASM. These named credentials are used to gain access to the various Resources during the Discovery process. Credentials are also typed according to their target resource such as, Server, Storage, VCenter, IOM, Chassis, and SCVMM.

### Get all defined Credentials

Default credentials are defined for several hardware types. These can be listed for the purpose of obtaining their reference id's needed for the Discovery process and to utilize as XML skeleton.

```
require 'ASMConfig'

url = ASM::API::URI('/Credential')
begin
  response = ASM::API::sign {
    RestClient.get url, :content_type => :xml
  }
  payload = ASM::Payload.from_xml(response)
  payload.save_xml('credentials.xml')
rescue RestClient::Exception => e
  print "Got exception with status: %d\n" % e.response.code
  print "%s\n" % e.response
end
```

### Define new Credential

Define a new Server Credential using a skeleton file previously saved to a file.

```
require 'ASMConfig'

url = ASM::API::URI('/Credential')

cred = ASM::Payload::load_xml('credential_skeleton.xml')

# Set Credential parameters
cred.set('label', 'ServerX')
cred.set('username', 'admin')
cred.set('password', 'abc123')
cred.set('protocol', '')
cred.set('snmpCommunityString', '')
```



```

response = ASM::API::sign {
  RestClient.post url, cred.to_xml, :content_type => :xml
}

payload = ASM::Payload.from_xml(response)
credId = payload.get('id')

# Save it for future use as a payload
payload.save_xml("credential_%s.xml" % credId)

```

## DiscoveryRequest

A DiscoveryRequest is a set of credential references and a set of IP ranges. A sweep of the given IP ranges will attempt to determine if Resources of the given types, specified by the credential types referenced in the request, exist at those IP addresses.

### Discover Chassis and contained Blade Servers

Using the predefined chassis and server credentials, initiate a discovery request against an IP address that is known to belong to a Chassis.

```

require 'ASMConfig'

url = ASM::API::URI('/DiscoveryRequest')

# Get credential ids
creds = ASM::Payload::load_xml('credentials.xml')
server_cred = creds.doc.xpath('//serverCredential').select{|c|
c.at_xpath('label').text == 'Dell PowerEdge Default'}.first.xpath('id').text
chassis_cred = creds.doc.xpath('//chassisCredential').select{|c|
c.at_xpath('label').text == 'Dell chassis default'}.first.xpath('id').text

disc = ASM::Payload::load_xml('discovery_skeleton.xml')
disc.set('deviceStartIp', '172.17.4.17')
disc.set('deviceEndIp', '172.17.4.17')
disc.set('unmanaged', 'false')

disc.appendChild('DiscoverIPRangeDeviceRequest',
  'deviceChassisCredRef' => chassis_cred,
  'deviceServerCredRef' => server_cred,
  'bDiscoverDevicesInChassis' => 'true')

begin
  response = ASM::API::sign {
    RestClient.post url, disc.to_xml, :content_type => :xml
  }

  payload = ASM::Payload.from_xml(response)

  # Get the job status
  jobStatus = payload.get('DiscoveryRequest/status')
  print "<status>%s</status>\n" % jobStatus

rescue RestClient::Exception => e
  print "Got exception with status: %d\n" % e.response.code
  print "%s\n" % e.response
end

```

## ManagedDevice

ManagedDevice is a generic interface to any Resources on the network which is being managed by ASM. A successful discovery request results in the creation of instances of ManagedDevice.

### Get all Managed Devices

Assuming the Discovery process succeeded, ASM will now know about some or all of the devices available on the network.

```
require 'ASMConfig'

url = ASM::API::URI('/ManagedDevice')

response = ASM::API::sign {
  RestClient.get url
}

doc = Nokogiri::XML(response)

# Print selected info for each device
print "%30s%15s%20s%20s%15s\n" % %w(ServiceTag State IP Model Type)
print
"-----\n"
-----\n"

doc.xpath('//ManagedDevice').sort_by{|d| d.at_xpath('deviceType').content}.each
do |device|

  svc_tag = device.at_xpath('serviceTag').content
  state = device.at_xpath('state').content
  ip_addr = device.at_xpath('ipAddress').content
  model = device.at_xpath('model').content
  dev_type = device.at_xpath('deviceType').content

  print "%30s%15s%20s%20s%15s\n" % [svc_tag, state, ip_addr, model, dev_type]

end
```

## ServiceTemplate

A ServiceTemplate is a reusable structure that defines the network topology and components of a future service deployment. A trivial example template could be a single server component setup to run Linux. Deploying this template would result in a single server being chosen from the available discovered resources and installing Linux on it.

### Get all ServiceTemplates

Here we retrieve all ServiceTemplates and output the names of those not in draft mode ( published ). The draft property determines if a template is in the published state. Only published templates can be deployed.

```
require 'ASMConfig'

url = ASM::API::URI('/ServiceTemplate')
```

```

result = ASM::API::sign {
  RestClient.get url
}
# Do some XML processing to find all published templates
doc = Nokogiri::XML(result)
templates = doc.xpath('//ServiceTemplate')
published = templates.select{|t| t.at_xpath('draft').content == 'false'}
template_names = published.collect{|t| t.at_xpath('templateName').content}

print "Published Templates:\n"
template_names.each{|name| print "%s\n" % name}

```

## Get a ServiceTemplate by name

Here we retrieve an existing ServiceTemplate from ASM named 'TwoServers' by passing the 'filter' query parameter. See the detail for ServiceTemplate in Appendix B for the full list of names that can be sorted and filtered on. Note that even if the template name is unique this method will always return a container around the result(s). For example, in the case of XML the root element is <ServiceTemplates>.

```

require 'ASMConfig'

url = ASM::API::URI('/ServiceTemplate')

template_name = 'TwoServers'
template_filter = "eq,name,#{template_name}"

response = ASM::API::sign {
  RestClient.get url, {:params=>{:filter=>template_filter}}
}

```

The response object is not shown for this example because it is too verbose. However, within the ServiceTemplate XML are parameters designated with the element <requiredAtDeploy>. If this value is true, then this parameter will be required when this ServiceTemplate is deployed. Otherwise the parameter is optional. For example, the server host name is required in this extracted snippet of the ServiceTemplate. Save the result above to a file inspect the full XML for a ServiceTemplate.

```

<parameters>
  <id>os_host_name</id>
  <value/>
  <type>STRING</type>
  <displayName>Host Name</displayName>
  <required>true</required>
  <requiredAtDeployment>true</requiredAtDeployment>
  <hideFromTemplate>true</hideFromTemplate>
  <min>0</min>
  <max>0</max>
  <dependencyTarget>generate_host_name</dependencyTarget>
  <dependencyValue>>false</dependencyValue>
</parameters>

```

## Publish a ServiceTemplate

In order to use a ServiceTemplate to deploy a Service, the ServiceTemplate must be first *published*. This is accomplished by setting the ServiceTemplate element <draft> to false. i.e. it is no longer in the draft state.

```

require 'ASMConfig'

templateId = "ff8080814be03f17014bea250f100b7d"
url = ASM::API::URI("/ServiceTemplate/%s"%templateId)

```

```

response = ASM::API::sign {
  RestClient.get url
}
# Set draft to false
payload = ASM::Payload.from_xml(response)
payload.set('/draft', 'false')

# Send it back as a PUT
response = ASM::API::sign {
  RestClient.put url, payload.to_xml, :content_type => :xml
}

```

## Deployment

While the ASM UI refers to a Service, the REST API resource is not Service, but rather Deployment. The two terms are synonymous. Deployments require an instance of a ServiceTemplate with its required parameters supplied.

### Get all deployed Services

```

require 'ASMConfig'

response = ASM::API::sign {
  url = ASM::API::URI("/Deployment")
  RestClient.get url
}

doc = Nokogiri::parse(response)

doc.xpath('//Deployment').each do |deploy|
  id = deploy.at_xpath('id').content
  name = deploy.at_xpath('deploymentName').content
  status = deploy.at_xpath('jobStatus').content
  print "%s : %s : %s\n" % [id,name,status]
end

```

After getting all Services we print the id, name, and status of each. Services that are still in the act of being deployed will have a status of IN\_PROGRESS. So calling this method with the Deployment Id as a path parameter can also be used to periodically poll a Deployment to find out when it has finished.

### Deploy a new Service

Here, after having previously gotten an existing ServiceTemplate by name, and saved it to a file, we then set the required parameters in the ServiceTemplate file, and finally embed the ServiceTemplate into a prefabricated Deployment structure that is read from a file. As mentioned previously we can monitor the status of this Deployment by performing a GET with the deployment id as a path parameter.

```

require 'ASMConfig'

url = ASM::API::URI('/Deployment')

template_name = 'TwoServers'
template = ASM::Payload::load_xml("%s.xml"%template_name)

deployment = ASM::Payload::load_xml('deployment_skeleton.xml')

# Set Deployment parameters
deployment.set("deploymentName", 'ManualAPIDeploymentDeux')
deployment.set("deploymentDescription", 'Manual API Deployment')

```

```

deployment.set("numberOfDeployments", '1')

# Inject the template into the deployment
deployment.replaceInner('serviceTemplate',
template.doc.at_xpath('ServiceTemplate'))
deployment.save_xml('deploy_payload.xml')

response = ASM::API::sign {
  RestClient.post url, deployment.to_xml, :content_type => :xml
}
doc = Nokogiri::parse(response)
deploymentId = doc.xpath('//Deployment/id').text

print "<id>%s</id>\n" % deploymentId

```

## Tear down a Service

In order to completely tear down a Service we set the teardown element to true for all components in the Deployment plus the Deployment itself, and then we PUT this Deployment back to ASM.

```

require 'ASMConfig'

deploymentId = "ff8080814aba74ce014ac455e9b80951"
url = ASM::API::URI("/Deployment/%s"%deploymentId)

response = ASM::API::sign {
  RestClient.get url
}
payload = ASM::Payload.from_xml(response)
payload.set('Deployment/teardown', 'true')
payload.set_all('components/teardown', 'true')
payload.save_xml('teardown.xml')

response = ASM::API::sign {
  RestClient.put url, payload.to_xml, :content_type => :xml
}
doc = Nokogiri::parse(response)
deploymentId = doc.xpath('//Deployment/id').text
jobStatus = doc.xpath('//Deployment/jobStatus').text

print "<jobStatus>%s</jobStatus>\n" % jobStatus

```

## Firmware

Firmware versions reported by individual resource components are compared against a firmware repository to determine if component firmware is up-to-date with the latest required versions.

### Check Firmware compliance of all Resources

Here we check the firmware compliance state of all resources. Note this is done by the POST method and supplying a ManagedDevice as the request body. The response is the same ManagedDevice with the addition of a compliance field.

```

require 'ASMConfig'

# Get all managed devices
url = ASM::API::URI('/ManagedDevice')
response = ASM::API::sign {
  RestClient.get url
}

```

```

}
doc = Nokogiri::XML(response)
doc.xpath('//ManagedDevice').each do |device|

  url = ASM::API::URI('/FirmwareRepository/compliance')

  begin
    response = ASM::API::sign {
      RestClient.post url, device.to_xml, :content_type => :xml
    }

    doc2 = Nokogiri::XML(response)

    serviceTag = doc2.at_xpath('//serviceTag').content
    compliance = doc2.at_xpath('//compliance').content

    print "Device %s firmware is %s\n" % [serviceTag,compliance]

  rescue RestClient::Exception => e
    print "Got exception with status: %d\n" % e.response.code
    print "%s\n" % e.response
  end
end
end

```

## Exception Handling

When an HTTP error code is returned the response payload will contain information about the error that occurred. For the purpose of illustration the following exception response was intentionally generated by trying to get a ServiceTemplate with an id that doesn't exist. In this case the HTTP response was 404 and the XML response is below. See the API reference section for the relevant HTTP responses that may be returned by each REST endpoint.

The example code used to catch this exception and report the error is below.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<AsmDetailedMessageList>
  <Messages>
    <messages>
      <messageBundle>AsmManagerMessages</messageBundle>
      <messageCode>ASM0042</messageCode>
      <severity>ERROR</severity>
      <category>USER_FACING</category>
      <displayMessage>Unable to find the Template Id: ABC.</displayMessage>
      <responseAction>Make sure that a correct Template Id is entered and retry
the operation. Template Ids are case sensitive. Verify the list of devices
under the Summary tab to make sure whether or not the Template already exists.</
responseAction>
      <detailedMessage>The Template Id cannot be found.</detailedMessage>
      <agentId>ASM Manager</agentId>
      <timeStamp>2015-03-10T14:34:24.866Z</timeStamp>
      <sequenceNumber>0</sequenceNumber>
    </messages>
  </Messages>
</AsmDetailedMessageList>

require 'ASMConfig'

templateId = "ABC" // bogus id
url = ASM::API::URI("/ServiceTemplate/%s"%templateId)

begin
  response = ASM::API::sign {
    RestClient.get url
  }
rescue RestClient::Exception => e
  print "Got exception with status: %d\n" % e.response.code
  print "%s\n" % e.response
end
```

# Appendix A - API Reference

## /Authenticate/ POST

### Description

Confirm User access credentials

### Method

POST

### Response Class

AuthenticateResponse

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Description of user to be authenticated	N/A	body	AuthenticateRequest

### Response Status Codes

HTTP Status Code	Reason
201	Authentication successful
401	User not found, domain not found or password does not match

## /Chassis/ GET

### Description

Retrieve a list a Chassis Devices



## Method

GET

## Response Class

[ Chassis ]

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Valid sort columns: health,managementIP,serviceTag,credentialRefId	N/A	query	string
filter	N/A	Valid filter columns: health,managementIP,serviceTag,credentialRefId	N/A	query	array
offset	N/A	Specify pagination offset	0	query	integer
limit	N/A	Specify page limit	50	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	Retrieved
400	Bad Request, verify input parameters are correct
401	No login information specified in the request
403	User doesn't have privileges to access this operation
500	Server Internal Error, contact your system administrator

## /Chassis/{refId} PUT

### Description

Update The Chasis configuration

### Method

PUT

### Response Class

Chassis

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Chassis to update	N/A	body	Chassis
refId	(required)	Id of Chassis to update	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
204	Update completed successfully
400	Problem with the update data, check error response for details
404	Chassis not found

## /Chassis/{refId} GET

### Description

Retrieve an individual Chassis Device

### Method

GET

### Response Class

Chassis

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Device Id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	retrieved
400	Bad Request, verify Device id is correct
404	Bad Request, verify Device id is correct

# /Chassis/{refId} DELETE

## Description

Deletes an existing Chassis Device

## Method

DELETE

## Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Device Id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	deleted
400	Bad Request, verify Device id is correct.

## /Chassis/{refId}/sensorlog DELETE

### Description

Clear an individual Chassis Device Sensor Event Logs

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Device Id	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
204	updated
400	Bad Request, verify Device object is correct
404	Bad Request, verify Device Id is correct

## /Chassis/{refId}/sensorlog GET

### Description

Retrieve an individual Chassis Device Sensor Event Logs

### Method

GET

### Response Class

[ ChassisLogEntry ]

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Device Id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	retrieved
400	Bad Request, verify Device id is correct
404	Bad Request, verify Device id is correct

## /Chassis/{serviceTag}/{type} GET

### Description

Retrieve a Chassis Device associated to a specific Server or IOM

### Method

GET

### Response Class

Chassis

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
serviceTag	(required)	Service Tag	N/A	path	string
type	(required)	Type	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	retrieved
400	Bad Request, verify Service tag and Type are correct
404	Bad Request, verify Service tag and Type are correct

## /Chassis/{serviceTag}/{type} DELETE

### Description

Deletes an existing server or IOA in chassis

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
serviceTag	(required)	Service Tag	N/A	path	string
type	(required)	Type	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
204	deleted
400	Bad Request, verify Device id is correct.

## /Chassis/iom/ip/{ip} GET

### Description

Retrieve an individual IOM Device

### Method

GET

### Response Class

IOM

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
ip	(required)	Device IP	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	retrieved
400	Bad Request, verify Device IP is correct
404	Bad Request, verify Device IP is correct

## /Credential/ POST

### Description

Add a new credential

### Method

POST

### Response Class

credential

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Credential to create	N/A	body	credential

## Response Status Codes

HTTP Status Code	Reason
201	Credential created
400	Bad Request, verify credential data object is correct

## /Credential/ GET

### Description

Retrieve a list of credentials

### Method

GET

### Response Class

credentials

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
type	N/A	filter on credential type	N/A	query	com.dell.asm.encryptedmgr.client.CredentialType
sort	N/A	Specify sort columns in a comma separated list of column names to sort. Default order is ascending. Column name can be prefixed with a minus sign to indicate descending for that column.	N/A	query	string
filter	N/A	Specify filter criteria, Example co,label,default	N/A	query	array
offset	N/A	Specify pagination offset	0	query	integer
limit	N/A	Specify page limit, can not	50	query	integer



Parameter	Value	Description	Default Value	Parameter Type	Data Type
		exceed system maximum limit.			

## Response Status Codes

HTTP Status Code	Reason
200	OK

## /Credential/{id} PUT

### Description

Updates an existing credential

### Method

PUT

### Response Class

credential

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Credential to update	N/A	body	credential
id	(required)	Id of credential to update	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Update completed successfully
400	Bad Request, verify credential data object is correct
404	Credential to be updated was not found.

## /Credential/{id} DELETE

### Description

Deletes an existing credential

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Id of credential to delete	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
204	Delete completed successfully
400	Bad Request, verify credential id is correct

## /Credential/{id} PUT

### Description

Updates an existing credential

### Method

PUT

### Response Class

credential

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Credential to update	N/A	body	credential
id	(required)	Id of credential to update	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Update completed successfully
400	Bad Request, verify credential data object is correct
404	Credential to be updated was not found.

## /Deployment/ POST

### Description

Create new Deployment

### Method

POST

### Response Class

Deployment

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Deployment to be created	N/A	body	Deployment

## Response Status Codes

HTTP Status Code	Reason
201	Deployment created
400	Invalid Parameters to create Deployment or problems with the Resource Adapters

HTTP Status Code	Reason
409	Deployment already exists

## /Deployment/ GET

### Description

Retrieve all Deployments with filter, sort, paginate which returns Array of Deployment class

### Method

GET

### Response Class

[ Deployment ]

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Valid sort columns: name,createdBy,createdDate,updatedBy,updatedDate,expirationDate,deploymentDesc,marsHalledTemplateData,health	N/A	query	string
filter	N/A	Filter Criteria	N/A	query	array
offset	N/A	Pagination Offset	0	query	integer
limit	N/A	Page Limit	50	query	integer

### Response Status Codes

HTTP Status Code	Reason
200	All Deployment Retrieved on filter, sort, paginate

## /Deployment/{id} PUT

### Description

Update Deployment

### Method

PUT

### Response Class

Deployment

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Deployment to be updated to	N/A	body	Deployment
id	(required)	Deployment Id (String)	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
204	Deployment updated
400	Invalid Parameters to Update Deployment or problem with the Resource Adapters
404	Deployment not found

## /Deployment/{id} GET

### Description

Retrieve Deployment based on deployment ID

### Method

GET

### Response Class

Deployment

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Deployment Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Deployment retrived
404	Deployment not found

# /Deployment/{id} DELETE

## Description

Delete Deployment -- this operation is idempotent

## Method

DELETE

## Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Deployment Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Deployment deleted
500	Unable to delete Deployment.

## /Deployment/migrate/{serviceld}/{serverId} PUT

### Description

Migrate Deployment based on deployment ID

### Method

PUT

### Response Class

Deployment

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Deployment Id (String)	N/A	body	string
serverId	(required)		N/A	path	string
serviceld	(required)		N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
200	Deployment migrated
404	Deployment not found

## /DiscoveryRequest/ GET

### Description

Retrieve a Array of DiscoveryRequest

### Method

GET

### Response Class

[ DiscoveryRequest ]

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Valid sort columns: id,status,status Message	N/A	query	string
filter	N/A	filter criteria	N/A	query	array
offset	N/A	pagination offset	N/A	query	integer
limit	N/A	page limit	N/A	query	integer

## Response Status Codes

HTTP Status Code	Reason
400	Problem with a query parameter, check response for details

## /DiscoveryRequest/ POST

### Description

discover devices of Ip range

### Method

POST

### Response Class

DiscoveryRequest

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A		N/A	body	DiscoveryRequest



## Response Status Codes

HTTP Status Code	Reason
202	Create the discovery resource
400	Bad Request, verify Discovery request object is correct

## /DiscoveryRequest/{id} GET

### Description

Retrieve Device from Inventory based on Id

### Method

GET

### Response Class

DiscoveryRequest

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Discovery ID	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Discovery result retrieved for the job
400	Bad Request, verify sort, filter, and pagination are valid
404	Job not found in the discovery

## /DiscoveryRequest/{id} DELETE

### Description

Delete Device from Discover result -- this operation is idempotent

## Method

DELETE

## Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Discovery ID	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Job deleted from inventory

# /DiscoveryRequest/chassislist POST

## Description

discover devices of Ip range

## Method

POST

## Response Class

DiscoveryRequest

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A		N/A	body	DiscoveryRequest

## Response Status Codes

HTTP Status Code	Reason
202	Create the discovery resource
400	Bad Request, verify Discovery request object is correct

## /DiscoveryRequest/discoveryresult/{id} GET

### Description

Retrieve a Array of DiscoveryResults

### Method

GET

### Response Class

[ DiscoveryResult ]

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	ref_id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
400	Problem with a query parameter, check response for details

## /ManagedDevice/ GET

### Description

Retrieve all Devices from Inventory with filter.

### Method

GET

## Response Class

[ ManagedDevice ]

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Valid sort columns: displayName,serviceTag,refId,health,refType,deviceType,ipAddress,state,model <a href="#">model</a> ,statusMessage,createdDate,createdBy,updatedDate,updatedBy,healthMessage,complaint,infraTemplateDate,infraTemplateId,serverTemplateDate,serverTemplateId,inventoryDate,complianceCheckDate,discoveredDate,identityRef,vendor	N/A	query	string
filter	N/A	Filter Criteria	N/A	query	array
offset	N/A	Pagination Offset	0	query	integer
limit	N/A	Page Limit	50	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	Device(s) retrieved from inventory on filter, sort, paginate

## /ManagedDevice/ POST

### Description

Create Device in Inventory, return array of Managed Devices created

### Method

POST

### Response Class

[ ManagedDevice ]

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A		N/A	body	[Lcom.dell.asm.asmcore.asmmanager.client.deviceinventory.ManagedDevice;

### Response Status Codes

HTTP Status Code	Reason
201	Devices created in inventory
400	Invalid Device to create in inventory, first error encountered error causes an error response to return. Call will not return a list of errors.
409	Device already exists in inventory, first error encountered error causes an error response to return. Call will not return a list of errors.

## /ManagedDevice/{refId} DELETE

### Description

Delete Device from Inventory -- this operation is idempotent

### Method

DELETE

## Response Class

ManagedDevice

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Reference Id	N/A	path	string
forceDelete	N/A	Force Delete	false	query	boolean

## Response Status Codes

HTTP Status Code	Reason
204	Device deleted from inventory
400	Unable to delete from RA's inventory

## /ManagedDevice/{refId} PUT

### Description

Update Device in Inventory

### Method

PUT

## Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Device to be updated to	N/A	body	ManagedDevice
refId	(required)	Reference Id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
201	Device updated in inventory
400	Invalid Device
404	Device not found in inventory

## /ManagedDevice/{refId} GET

### Description

Retrieve Device from Inventory based on refId

### Method

GET

### Response Class

ManagedDevice

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
refId	(required)	Reference Id	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Device retrieved from inventory
404	Device not found in inventory

## /ManagedDevice/count GET

### Description

Retrieve Device total count

### Method

GET

## Response Class

ManagedDevice

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
filter	N/A	Filter Criteria	N/A	query	array

## Response Status Codes

HTTP Status Code	Reason
200	Device Count retrieved from inventory

## /ManagedDevice/firmware PUT

### Description

Update Device Firmware

### Method

PUT

## Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A		N/A	body	FirmwareUpdateRequest

## Response Status Codes

HTTP Status Code	Reason
201	Device updated Firmware
400	Invalid Device
404	Device not found in inventory



## /ManagedDevice/puppet/{certName} GET

### Description

Retrieve Device from Inventory based on certName

### Method

GET

### Response Class

ManagedDevice

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
certName	(required)	Certificate Name	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
200	Device retrieved from inventory
404	Device not found in inventory

## /ManagedDevice/puppet/{certName} PUT

### Description

Update Device in Inventory

### Method

PUT

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Device to be updated to	N/A	body	ManagedDevice
certName	(required)	Certificate Name	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
201	Device updated in inventory
400	Invalid Device
404	Device not found in inventory

## /Network/ GET

### Description

Retrieve networks

### Method

GET

### Response Class

[ Network ]

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Specify sort columns(Supported columns are:name,description,type,vlanId,createdBy,createdAtDate,updatedAtDate,updatedBy,updatedDate) in a comma	N/A	query	string

Parameter	Value	Description	Default Value	Parameter Type	Data Type
		separated list of column names to sort. Default order is ascending. Column name can be prefixed with a minus sign to indicate descending for that column.			
filter	N/A	Specify filter criteria(Supported columns are:name,description,type,vlanId,createdBy,updatedBy,createDate,updateDate(For dates supports only eq filter also the format of input should be dd-MM-yyyy)), Example co,name,CorporateNetwork	N/A	query	array
offset	N/A	Specify pagination offset	0	query	integer
limit	N/A	Specify page limit, cannot exceed system limit of 50	50	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	Networks is retrieved
400	Problem with a query parameter, check response for details
404	Networks not found

## /Network/ POST

### Description

Add a new network

### Method

POST

### Response Class

Network

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Network object that needs to be added to ASM	N/A	body	Network

### Response Status Codes

HTTP Status Code	Reason
201	Network created
400	Invalid input or network supplied or verify other input data is correct
409	Network already exists

## /Network/{networkId} DELETE

### Description

Delete an existing network

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
networkId	(required)	Id of network to delete	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Network deleted successfully
400	Invalid network id supplied

## /Network/{networkId} PUT

### Description

Update an existing network

### Method

PUT

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Network object that contains update fields	N/A	body	Network
networkId	(required)	Id of network to update	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Network updated successfully
400	Invalid input for id or network supplied or verify other input data is correct

HTTP Status Code	Reason
404	Network not found
409	Network with similar name already exists

## /Network/{networkId} GET

### Description

Find a network by id

### Method

GET

### Response Class

Network

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
networkId	(required)	Id of network to retrieve	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
200	Network retrieved successfully
404	Network not found

## /Network/ipAddress/assign PUT

### Description

Assigns a list of ip addresses to a usage Guid

### Method

PUT

### Response Class

string

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
networkId	(required)	Id of network to update	N/A	query	string
numberToReserve	(required)	Number of ip addresses to reserve	N/A	query	integer
usageGUID	(required)	Usage GUID to associate with ipAddress	N/A	query	string

## Response Status Codes

HTTP Status Code	Reason
204	IP Addresses assigned successfully
400	Invalid network id supplied

## /Network/ipAddress/release PUT

### Description

Releases ip addresses assigned to a usage guid

### Method

PUT

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
usageGUID	(required)	Usage GUID associated with ipAddress(es)	N/A	query	string

## Response Status Codes

HTTP Status Code	Reason
204	Released IP addresses successfully
400	Invalid input for usageGUID
404	Reservation associated with usageGUID not found

## /NTP/ DELETE

### Description

Turn off NTP settings on ASM appliance

### Method

DELETE

### Response Class

void

Response Content-Type: application/json, application/xml

## Response Status Codes

HTTP Status Code	Reason
204	Successfully set the NTP settings to off.
500	Unable to turn off NTP settings on the ASM appliance.

## /NTP/ PUT

### Description

Apply NTP settings on ASM appliance

### Method

PUT

### Response Class

void

Response Content-Type: application/json, application/xml



## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	NTPSetting object that needs to be set on ASM appliance	N/A	body	ntpSettings

## Response Status Codes

HTTP Status Code	Reason
204	Successfully set the NTP settings to on.
500	Unable to turn on NTP settings on the ASM appliance.

## /NTP/ GET

### Description

Retrieve NTP settings from ASM appliance

### Method

GET

### Response Class

ntpSettings

Response Content-Type: application/json, application/xml

## Response Status Codes

HTTP Status Code	Reason
200	Retrieved NTP settings from ASM appliance successfully.
500	Unable to get NTP settings from the ASM appliance.

## /Proxy/ GET

### Description

Retrieve proxy settings from ASM appliance

### Method

GET

### Response Class

proxySettings

Response Content-Type: application/json, application/xml

### Response Status Codes

HTTP Status Code	Reason
200	ASM appliance proxy settings retrieved successfully
500	Unable to retrieve proxy settings from ASM appliance

## /Proxy/ PUT

### Description

Set the proxy settings on ASM appliance

### Method

PUT

### Response Class

proxySettings

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	ProxySetting object that needs to be set on ASM appliance	N/A	body	proxySettings

## Response Status Codes

HTTP Status Code	Reason
200	Proxy settings set successfully in ASM appliance
500	Unable to set the proxy settings in ASM appliance

## /Proxy/test POST

### Description

Retrieve proxy settings from ASM appliance

### Method

POST

### Response Class

testProxyResponse

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	ProxySetting object that needs to be tested on ASM appliance	N/A	body	proxySettings

## Response Status Codes

HTTP Status Code	Reason
200	Passed test for proxy settings defined in ASM appliance
500	Unable to test proxy settings defined in ASM appliance

## /ServerPool/ POST

### Description

Add a new pool

## Method

POST

## Response Class

Pool

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Pool object that needs to be added to ASM	N/A	body	Pool

## Response Status Codes

HTTP Status Code	Reason
201	Pool created
400	Invalid input or pool supplied or verify other input data is correct
409	Pool already exists

## /ServerPool/ GET

### Description

Retrieve pools

### Method

GET

### Response Class

[ Pool ]

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Specify sort columns(Supported columns are:name,description,createdBy,createdDate,updatedBy,update dDate) in a comma separated list of column names to sort. Default order is ascending. Column name can be prefixed with a minus sign to indicate descending for that column.	N/A	query	string
filter	N/A	Specify filter criteria(Supported columns are:name,description,createdBy,updatedBy,updatedDate,create dDate(For dates supports only eq filter also the format of input should be dd-MM-yyyy)), Example co,name,Global	N/A	query	array
offset	N/A	Specify pagination offset	0	query	integer
limit	N/A	Specify page limit, cannot exceed system limit of 50	50	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	Pools is retrieved
400	Problem with a query parameter, check response for details
404	Pools not found

## /ServerPool/{poolId} DELETE

### Description

Delete an existing pool

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
poolId	(required)	Id of pool to delete	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Pool deleted successfully
400	Invalid pool id supplied

## /ServerPool/{poolId} GET

### Description

Find a pool by id

## Method

GET

## Response Class

Pool

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
poolId	(required)	Id of pool to retrieve	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	Pool retrieved successfully
404	Pool not found

## /ServerPool/{poolId} PUT

### Description

Update an existing pool

### Method

PUT

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	Pool object that contains update fields	N/A	body	Pool
poolId	(required)	Id of pool to update	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	Pool updated successfully
400	Invalid input for id or pool supplied or verify other input data is correct
404	Pool not found
409	Pool with similar name already exists

## /ServerPool/{poolId}/generateVirtualIdentity PUT

### Description

Generate virtual identities in the pool for the given type and count

### Method

PUT

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
poolId	(required)	Id of pool to generate virtual identities	N/A	path	string
type	(required)	Type of virtual identity pool to	N/A	query	string



Parameter	Value	Description	Default Value	Parameter Type	Data Type
count	(required)	generate virtual identities Number of virtual identities to generate	N/A	query	integer

## Response Status Codes

HTTP Status Code	Reason
204	Virtual Identities generated successfully
400	Invalid input for pool id or type or number of virtual identities to generate or verify other input data is correct
404	Pool not found

## /ServiceTemplate/ GET

### Description

Retrieve all ServiceTemplates with filter, sort, paginate which returns Array of ServiceTemplate.class

### Method

GET

### Response Class

[ ServiceTemplate ]

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
sort	N/A	Supported sort columns are: name,createdBy,createdDate,updatedBy,updatedDate	N/A	query	string
filter	N/A	Filter Criteria	N/A	query	array
offset	N/A	Pagination Offset	0	query	integer

Parameter	Value	Description	Default Value	Parameter Type	Data Type
limit	N/A	Page Limit	50	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	All ServiceTemplates Retrived on filter, sort, paginate

## /ServiceTemplate/ POST

### Description

Create new ServiceTemplate

### Method

POST

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Infrastructure Template to be created	N/A	body	ServiceTemplate

## Response Status Codes

HTTP Status Code	Reason
201	ServiceTemplate created
400	Invalid Parameters to create ServiceTemplate or problems with the Resource Adapters
409	Template already exists

## /ServiceTemplate/{id} GET

### Description

Retrieve ServiceTemplate based on ServiceTemplate id

### Method

GET

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Infrastructure Template Id (String)	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
200	ServiceTemplate retrieved
404	ServiceTemplate not found

## /ServiceTemplate/{id} DELETE

### Description

Delete ServiceTemplate -- this operation is idempotent

### Method

DELETE

### Response Class

void

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Infrastructure Template Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	ServiceTemplate deleted
500	Unable to delete ServiceTemplate.

## /ServiceTemplate/{id} PUT

### Description

Update Template

### Method

PUT

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Infrastructure Template to be updated to	N/A	body	ServiceTemplate
id	(required)	Infrastructure Template Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
204	InfrastructureTemplate updated
400	Invalid Parameters to Update ServiceTemplate

HTTP Status Code	Reason
404	Template not found

## /ServiceTemplate/{id}/copy POST

### Description

Copy a ServiceTemplate

### Method

POST

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Infrastructure Template settings	N/A	body	ServiceTemplate
id	(required)	Infrastructure Template Id (String)	N/A	path	string

### Response Status Codes

HTTP Status Code	Reason
201	ServiceTemplate copied successfully
400	Invalid Parameters to copy ServiceTemplate or problems with the Resource Adapters
409	Template already exists

## /ServiceTemplate/{id}/mapToPhysicalResources POST

### Description

Do Physical Resource allocation based on ServiceTemplate

## Method

POST

## Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
id	(required)	Infrastructure Template Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
201	Map Service Template to physical resources
400	Invalid Parameters to copy ServiceTemplate or problems with the Resource Adapters
409	Template already exists

## /ServiceTemplate/device/{deviceId} GET

### Description

Retrieve Default Template customized for specified device

### Method

GET

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
deviceId	(required)	Device Id (String)	N/A	path	string

## Response Status Codes

HTTP Status Code	Reason
200	ServiceTemplate retrieved
404	ServiceTemplate not found

## /ServiceTemplate/export POST

### Description

Export a ServiceTemplate

### Method

POST

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Service Template	N/A	body	ServiceTemplate
encPassword	N/A	Encryption password	N/A	query	string
useEncPwdFromBackup	N/A	use password from backup	N/A	query	boolean

## Response Status Codes

HTTP Status Code	Reason
201	ServiceTemplate exported successfully
400	Invalid Parameters to copy ServiceTemplate or problems with the Resource Adapters
404	Template not found

## /ServiceTemplate/upload POST

### Description

Upload a ServiceTemplate

### Method

POST

### Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Service Template upload request	N/A	body	ServiceTemplateUploadRequest

## Response Status Codes

HTTP Status Code	Reason
201	ServiceTemplate uploaded successfully
400	Invalid Parameters to upload template

## /ServiceTemplate/upload POST

### Description

Upload a ServiceTemplate

### Method

POST



## Response Class

ServiceTemplate

Response Content-Type: application/xml, application/json

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	N/A	Service Template upload request	N/A	body	ServiceTemplateUploadRequest

## Response Status Codes

HTTP Status Code	Reason
201	ServiceTemplate uploaded successfully
400	Invalid Parameters to upload template

## /Timezone/ GET

### Description

Retrieve ASM appliance timezone

### Method

GET

## Response Class

timeZone

Response Content-Type: application/json, application/xml

## Response Status Codes

HTTP Status Code	Reason
200	ASM appliance timezone retrieved successfully.
500	Unable to retrieve ASM appliance Time Zone

## /Timezone/ PUT

### Description

Sets the appliance timezone

### Method

PUT

### Response Class

Response

Response Content-Type: application/json, application/xml

### Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
body	(required)	TimeZoneInfo that needs to be set on ASM appliance	N/A	body	timeZone

### Response Status Codes

HTTP Status Code	Reason
200	ASM appliance timezone set successfully.
500	Unable to set the ASM appliance Time Zone

## /Timezone/all GET

### Description

Retrieve list of available timezones

### Method

GET

### Response Class

availableTimeZones

Response Content-Type: application/json, application/xml

## Parameters

Parameter	Value	Description	Default Value	Parameter Type	Data Type
offset	N/A	Specify pagination offset	N/A	query	integer
limit	N/A	Specify page limit, can not exceed system maximum limit.	N/A	query	integer

## Response Status Codes

HTTP Status Code	Reason
200	Available timezones list retrieved successfully.
500	Unable to retrieve list of available timezones

## Appendix B - Model Reference

### asmCredential

Name: asmCredential

Property Name	Property Type	Required
references	references	false
credential	credential	true
link	Link	false

### AsmDetailedMessage

Name: AsmDetailedMessage

Property Name	Property Type	Required
severity	string	false
messageBundle	string	false
sequenceNumber	integer	false
timeStamp	string	false
category	string	false
correlationId	string	false
displayMessage	string	false
messageCode	string	false
agentId	string	false
responseAction	string	false
detailedMessage	string	false

## AsmDetailedMessageList

Name: AsmDetailedMessageList

Property Name	Property Type	Required
messages	<a href="#">[AsmDetailedMessage]</a>	false

## AuthenticateRequest

Name: AuthenticateRequest

Property Name	Property Type	Required
userName	string	false
domain	string	false
password	string	false

## AuthenticateResponse

Name: AuthenticateResponse

Property Name	Property Type	Required
userName	string	true
apiSecret	string	true
domain	string	true
role	string	true
apiKey	string	true

## availableTimeZones

Name: availableTimeZones

Property Name	Property Type	Required
availableTimeZones	<a href="#">[timeZone]</a>	false

# Chassis

Name: Chassis

Property Name	Property Type	Required
defaultPowerCapUpperBoundWatts	integer	false
defaultPowerCapUpperBoundBTU	integer	false
powerSupplies	<a href="#">[ChassisPowerSupply]</a>	false
kVMs	<a href="#">[KVM]</a>	false
model	string	false
deviceId	string	false
managementIPStatic	boolean	false
credentialRefId	string	false
assetTag	string	false
health	Health	false
defaultPowerCapLowerBoundWatts	integer	false
defaultPowerCapLowerBoundBTU	integer	false
midPlaneVersion	string	false
lastFirmwareUpdateTime	GregorianCalendar	false
iomCount	integer	false
managementIP	string	false
iOMs	<a href="#">[IOM]</a>	false
controllers	<a href="#">[ChassisController]</a>	false
serviceTag	string	false
serverCount	integer	false
aisle	string	false
rack	string	false
datacenter	string	false
rackslot	string	false
dnsName	string	false

Property Name	Property Type	Required
powerCapPercent	integer	false
name	string	false
servers	<a href="#">[Server]</a>	false
lastUpdateTime	GregorianCalendar	false
deviceType	string	false
refId	string	false
displayName	string	false
url	Link	false
refType	string	false

## ChassisController

Name: ChassisController

Property Name	Property Type	Required
controllerPrimary	boolean	false
controllerFWVersion	string	false
controllerName	string	false
controllerPresent	boolean	false
id	string	false

## ChassisLogEntry

Name: ChassisLogEntry

Property Name	Property Type	Required
severity	LogSeverity	false
logTime	GregorianCalendar	false
message	string	false

## ChassisPowerSupply

Name: ChassisPowerSupply

Property Name	Property Type	Required
present	boolean	false
capacity	string	false
powerStatus	string	false
id	string	false
slot	string	false

## ComparatorValue

Enum: ComparatorValue

Property Name	Property Type	Required
values	["minimum", "maximum", "exact"]	

## CompliantState

Enum: CompliantState

Property Name	Property Type	Required
values	["NA", "compliant", "noncompliant", "unknown", "updaterequired"]	

## component

Name: component

Property Name	Property Type	Required
componentValid	ServiceTemplateValid	false
componentID	string	false



Property Name	Property Type	Required
refId	string	false
cloned	boolean	false
clonedFromId	string	false
helpText	string	false
manageFirmware	boolean	false
asmGUID	string	false
puppetCertName	string	false
relatedComponents	Map,string>	false
teardown	boolean	false
name	string	false
resources	<a href="#">[ServiceTemplateCategory]</a>	false
id	string	false
type	ServiceTemplateComponentType	false
serialNumber	string	false
iP	string	false

## credential

Name: credential

Property Name	Property Type	Required
domain	string	false
label	string	true
updatedAt	string	false
createdDate	string	false
updatedBy	string	false
id	string	false
link	Link	false
username	string	true
password	string	false
createdBy	string	false

## credentialList

Name: credentialList

Property Name	Property Type	Required
totalRecords	integer	false
credentialList	<a href="#">[asmCredential]</a>	false

## Deployment

Name: Deployment

Property Name	Property Type	Required
updatedAt	GregorianCalendar	false
createdAt	GregorianCalendar	false
updatedBy	string	false
jobDetails	<a href="#">[logEntry]</a>	false
canMigrate	boolean	false
canScaleupStorage	boolean	false
canScaleupNetwork	boolean	false
canScaleupServer	boolean	false
canScaleupVM	boolean	false
canDeleteResources	boolean	false
canRetry	boolean	false
canEdit	boolean	false
canDelete	boolean	false
canCancel	boolean	false
canScaleupApplication	boolean	false
canScaleupCluster	boolean	false
firmwareRepository	FirmwareRepository	false
serviceTemplate	ServiceTemplate	false
allUsersAllowed	boolean	false
templateValid	boolean	false

Property Name	Property Type	Required
assignedUsers	[ <a href="#">User</a> ]	false
vms	[ <a href="#">VM</a> ]	false
deploymentDevice	[ <a href="#">DeploymentDevice</a> ]	false
deploymentName	string	false
scheduleDate	string	false
numberOfDeployments	integer	false
deploymentDescription	string	false
firmwareRepositoryId	string	false
updateServerFirmware	boolean	false
teardown	boolean	false
individualTeardown	boolean	false
retry	boolean	false
id	string	false
owner	string	false
status	DeploymentStatusType	false
createdBy	string	false

## DeploymentDevice

Name: DeploymentDevice

Property Name	Property Type	Required
statusStartTime	string	false
statusEndTime	string	false
logDump	string	false
statusMessage	string	false
componentId	string	false
deviceType	DeviceType	false
refId	string	false
ipAddress	string	false
serviceTag	string	false
status	DeploymentStatusType	false
refType	string	false

## DeploymentFilterResponse

Name: DeploymentFilterResponse

Property Name	Property Type	Required
numberRequestedServers	integer	false
numberSelectedServers	integer	false
failedPoolName	string	false
selectedServers	<a href="#">[SelectedServer]</a>	false
rejectedServers	<a href="#">[RejectedServer]</a>	false
failedPoolId	string	false

## DeploymentStatusType

Enum: DeploymentStatusType

Property Name	Property Type	Required
values	["Pending", "In progress", "Complete", "Error", "Cancelled"]	

## DeviceDiscoveryRequest

Name: DeviceDiscoveryRequest

Property Name	Property Type	Required
config	string	false
deviceChassisCredRef	string	false
deviceServerCredRef	string	false
deviceSwitchCredRef	string	false
deviceVCenterCredRef	string	false
deviceStorageCredRef	string	false
deviceSCVMMCredRef	string	false
deviceStartIp	string	false

Property Name	Property Type	Required
deviceEndIp	string	false
deviceBMCServerCredRef	string	false
deviceType	DeviceType	false
unmanaged	boolean	false

## DeviceDiscoveryRequests

Name: DeviceDiscoveryRequests

Property Name	Property Type	Required
discoverIpRangeDeviceRequests	[ <a href="#">DeviceDiscoveryRequest</a> ]	false

## DeviceGroup

Name: DeviceGroup

Property Name	Property Type	Required
updatedAt	GregorianCalendar	false
createdAt	GregorianCalendar	false
updatedBy	string	false
groupDescription	string	false
groupUserList	GroupUserList	false
managedDeviceList	ManagedDeviceList	false
groupSeqId	integer	false
groupName	string	true
link	Link	false
createdBy	string	false

## DeviceGroupList

Name: DeviceGroupList

Property Name	Property Type	Required
totalCount	integer	false
deviceGroup	[ <a href="#">DeviceGroup</a> ]	true

## DeviceHealth

Enum: DeviceHealth

Property Name	Property Type	Required
values	["green", "yellow", "red", "unknown"]	

## DeviceState

Enum: DeviceState

Property Name	Property Type	Required
values	["Discovered", "Updating", "Configuration Error", "Pending Configuration Template", "Pending Delete", "Pending", "Ready", "Discovery Failed", "Delete Failed", "Deleted", "Offline", "Deployed", "Deploying", "unmanaged"]	

## DeviceType

Enum: DeviceType

Property Name	Property Type	Required
values	["Chassis", "Server", "TOR", "Chassis M1000e", "Chassis VRTX", "Rack Server", "Blade Server", "Aggregator IOM", "MXL IOM", "Storage", "Compellent", "Equallogic", "Generic Switch", "Dell Switch", "vCenter", "Virtual Machine", "netapp", "scvmm",	

Property Name	Property Type	Required
	"Unknown", "BMC", "Chassis FX2", "FX2 Sled"]	

## dhcpSettings

Name: dhcpSettings

Property Name	Property Type	Required
enabled	boolean	false
domain	string	false
startingIpAddress	string	false
endingIpAddress	string	false
defaultLeaseTime	integer	false
maxLeaseTime	integer	false
dns	string	false
gateway	string	false
subnet	string	false
netmask	string	false

## DiscoverDeviceType

Enum: DiscoverDeviceType

Property Name	Property Type	Required
values	["IDRAC6", "CMC", "IDRAC7", "IDRAC8", "UNKNOWN", "SERVER", "IOM", "VRTX", "VCENTER", "EQUALLOGIC", "COMPELLENT", "FORCE10", "BROCADE", "POWERCONNECT", "NETAPP", "CISCONEXUS", "FORCE10IOM", "EQUALLOGIC_NODISCOVER", "DELL_IOM_84", "SCVMM", "BMC", "CMC_FX2", "FX2_IOM"]	

# DiscoveredDevices

Name: DiscoveredDevices

Property Name	Property Type	Required
config	string	false
parentJobId	string	true
deviceRefId	string	true
firmwareDeviceInventories	[ <a href="#">FirmwareDeviceInventory</a> ]	false
statusMessage	string	false
model	string	false
serverType	string	false
deviceType	DeviceType	false
unmanaged	boolean	false
refId	string	true
healthState	string	false
healthStatusMessage	string	false
facts	string	false
iomCount	integer	false
ipAddress	string	true
discoverDeviceType	DiscoverDeviceType	false
serviceTag	string	true
serverCount	integer	false
credId	string	true
systemId	string	false
status	DiscoveryStatus	false
configuration	string	false
refType	string	true
jobId	string	false
vendor	string	false



# DiscoveryRequest

Name: DiscoveryRequest

Property Name	Property Type	Required
statusMessage	string	false
discoveryRequestList	DeviceDiscoveryRequests	false
devices	[ <a href="#">DiscoveredDevices</a> ]	false
totalCount	integer	false
id	string	true
link	Link	false
status	DiscoveryStatus	false

# DiscoveryResult

Name: DiscoveryResult

Property Name	Property Type	Required
config	string	false
parentJobId	string	false
deviceRefId	string	false
firmwareDeviceInventories	[ <a href="#">FirmwareDeviceInventory</a> ]	false
statusMessage	string	false
model	string	false
serverType	string	false
deviceType	DeviceType	false
refId	string	false
healthState	string	false
healthStatusMsg	string	false
iomCount	integer	false
ipAddress	string	false
discoverDeviceType	DiscoverDeviceType	false
serviceTag	string	false

Property Name	Property Type	Required
serverCount	integer	false
systemId	string	false
status	DiscoveryStatus	false
refType	string	false
jobId	string	false
vendor	string	false

## DiscoveryStatus

Enum: DiscoveryStatus

Property Name	Property Type	Required
values	["Pending", "Connected", "Unsupported", "InProgress", "Success", "Failed to discover", "Error discovering"]	

## DiskMediaType

Enum: DiskMediaType

Property Name	Property Type	Required
values	["requireSSD", "requireHDD", "any"]	

## Fabric

Name: Fabric

Property Name	Property Type	Required
enabled	boolean	false
usedForfc	boolean	false
redundancy	boolean	false
nictype	string	false
name	string	false

Property Name	Property Type	Required
interfaces	<a href="#">[Interface]</a>	false
id	string	false

## FirmwareComplianceComponents

Name: FirmwareComplianceComponents

Property Name	Property Type	Required
embeddedRepoComponents	<a href="#">[SoftwareComponent]</a>	false
defaultRepoComponents	<a href="#">[SoftwareComponent]</a>	false
compliantState	CompliantState	false

## FirmwareDeviceInventory

Name: FirmwareDeviceInventory

Property Name	Property Type	Required
subvendorID	string	false
subdeviceID	string	false
componentID	string	false
vendorID	string	false
deviceID	string	false
parent_job_id	string	false
ipaddress	string	false
firmwareComplianceComponent s	FirmwareComplianceComponent s	false
discoveryResult	DiscoveryResult	false
servicetag	string	false
deviceInventory	ManagedDevice	false
fqdd	string	false
componentType	string	false
name	string	false
id	string	false

Property Name	Property Type	Required
systemId	string	false
version	string	false
lastUpdateTime	string	false
jobId	string	false

## FirmwareInventory

Name: FirmwareInventory

Property Name	Property Type	Required
subvendorID	string	false
bPresent	boolean	false
updateable	boolean	false
firmwareId	string	false
subdeviceID	string	false
componentID	string	false
vendorID	string	false
deviceID	string	false
deviceRef	string	false
fqdd	string	false
componentType	string	false
name	string	false
version	string	false
lastUpdateTime	string	false

## FirmwareRepository

Name: FirmwareRepository

Property Name	Property Type	Required
componentCount	integer	false
deployments	<a href="#">[Deployment]</a>	false
updatedAt	string	false

Property Name	Property Type	Required
createdDate	string	false
softwareBundles	<a href="#">[SoftwareBundle]</a>	false
downloadStatus	RepositoryStatus	false
sourceType	string	false
updatedBy	string	false
md5Hash	string	false
sourceLocation	string	false
softwareComponents	<a href="#">[ SoftwareComponent ]</a>	false
diskLocation	string	false
bundleCount	integer	false
defaultCatalog	boolean	false
name	string	false
id	string	false
username	string	false
filename	string	false
password	string	false
embedded	boolean	false
createdBy	string	false

## FirmwareUpdateRequest

Name: FirmwareUpdateRequest

Property Name	Property Type	Required
scheduleDate	string	false
scheduleType	string	false
idList	string	false
updateType	UpdateType	false
exitMaintenanceMode	boolean	false

## GregorianCalendar

Name: GregorianCalendar

Property Name	Property Type	Required
value	string	true

## GroupUser

Name: GroupUser

Property Name	Property Type	Required
enabled	boolean	false
userName	string	false
userSeqId	integer	true
firstName	string	false
lastName	string	false
role	string	true

## GroupUserList

Name: GroupUserList

Property Name	Property Type	Required
totalRecords	integer	false
groupUsers	<a href="#">[GroupUser]</a>	false

## Health

Enum: Health

Property Name	Property Type	Required
values	["GREEN", "YELLOW", "RED", "UNKNOWN"]	

## IKVM

Name: IKVM

Property Name	Property Type	Required
present	boolean	false
partNumber	string	false
manufacturer	string	false
firmwareVersion	string	false
name	string	false
id	string	false

## Interface

Name: Interface

Property Name	Property Type	Required
enabled	boolean	false
partitioned	boolean	false
usedforfc	boolean	false
redundancy	boolean	false
partitions	<a href="#">[Partition]</a>	false
nictype	string	false
name	string	false
interfaces	<a href="#">[Interface]</a>	false
id	string	false

## IOM

Name: IOM

Property Name	Property Type	Required
model	string	false
managementIPStatic	boolean	false
health	Health	false
managementIP	string	false
serviceTag	string	false
location	<a href="#">IOMLocationType</a>	false
id	string	false
slot	integer	false
supported	boolean	false

## IOMLocationType

Enum: IOMLocationType

Property Name	Property Type	Required
values	["A1", "A2", "B1", "B2", "C1", "C2", "UNKNOWN"]	

## IpRange

Name: IpRange

Property Name	Property Type	Required
endingIp	string	true
startingIp	string	true
id	string	false

## Link

Name: Link



Property Name	Property Type	Required
href	string	true
title	string	false
rel	string	true
type	string	false

## List

Name: List

Property Name	Property Type	Required
empty	boolean	false

## logEntry

Name: logEntry

Property Name	Property Type	Required
componentId	string	false
executionId	string	false
message	string	false
timestamp	string	false
level	string	false

## LogSeverity

Enum: LogSeverity

Property Name	Property Type	Required
values	["Normal", "Warning", "Critical", "Unknown"]	

# ManagedDevice

Name: ManagedDevice

Property Name	Property Type	Required
config	string	false
failuresCount	integer	false
firmwareUpdateTime	string	false
firmwareDeviceInventories	<a href="#">[FirmwareDeviceInventory]</a>	false
statusMessage	DeviceState	false
manufacturer	string	false
cpuType	string	false
healthMessage	string	false
infraTemplateDate	GregorianCalendar	false
infraTemplateId	string	false
serverTemplateDate	GregorianCalendar	false
serverTemplateId	string	false
inventoryDate	GregorianCalendar	false
complianceCheckDate	GregorianCalendar	false
discoveredDate	GregorianCalendar	false
model	string	false
health	DeviceHealth	false
deviceType	DeviceType	true
refId	string	true
detailLink	Link	false
operatingSystem	string	false
numberOfCPUs	integer	false
memoryInGB	integer	false
nics	integer	false
facts	string	false
ipAddress	string	true
discoverDeviceType	DiscoverDeviceType	true
serviceTag	string	true

Property Name	Property Type	Required
compliance	CompliantState	false
deviceGroupList	DeviceGroupList	false
credId	string	false
state	DeviceState	false
systemId	string	false
refType	string	false

## MigrationRequest

Name: MigrationRequest

Property Name	Property Type	Required
serverList	string	false
poolList	string	false
autoMigrate	boolean	false

## Network

Name: Network

Property Name	Property Type	Required
description	string	false
vlanId	integer	false
staticNetworkConfiguration	StaticNetworkConfiguration	false
name	string	false
id	string	false
type	string	false
static	boolean	false

## NetworkConfiguration

Name: NetworkConfiguration

Property Name	Property Type	Required
allUsedInterfaces	<a href="#">[Interface]</a>	false
usedInterfaces	<a href="#">[Interface]</a>	false
servertype	string	false
fabrics	<a href="#">[Fabric]</a>	false
interfaces	<a href="#">[Interface]</a>	false
id	string	false

## ntpSettings

Name: ntpSettings

Property Name	Property Type	Required
preferredNTPServer	string	false
secondaryNTPServer	string	false

## Partition

Name: Partition

Property Name	Property Type	Required
maximum	integer	false
minimum	integer	false
lanMacAddress	string	false
iscsiQN	string	false
networks	string	false
networkObjects	<a href="#">[Network]</a>	false
iscsiMacAddress	string	false
wwnn	string	false
wwpn	string	false
name	string	false
id	string	false

## proxySettings

Name: proxySettings

Property Name	Property Type	Required
enabled	boolean	false
userName	string	false
userCredentialEnabled	boolean	false
proxyServer	string	false
port	string	false
password	string	false

## RAIDConfiguration

Name: RAIDConfiguration

Property Name	Property Type	Required
hddHotSpares	string	false
ssdHotSpares	string	false
virtualDisks	<a href="#">VirtualDisk</a>	false

## Reason

Enum: Reason

Property Name	Property Type	Required
values	["RAID", "BAD_STATE", "NIC_CAPS", "SD_NOT_PRESENT", "BAD_RACK_SERVER_MODEL", "BAD_BLADE_SERVER_MODEL", "ALREADY_DEPLOYED", "CLONED_MODEL_MISMATCH", "FIRMWARE_NOT_COMPLIANT", "OTHER"]	

## references

Name: references

Property Name	Property Type	Required
policies	integer	false
devices	integer	false
totalReferences	integer	false

## RejectedServer

Name: RejectedServer

Property Name	Property Type	Required
reason	Reason	false
refId	string	false

## RepositoryStatus

Enum: RepositoryStatus

Property Name	Property Type	Required
values	["pending", "copying", "error", "available"]	

## SelectedServer

Name: SelectedServer

Property Name	Property Type	Required
componentId	string	false
refId	string	false
serviceTag	string	false
raidConfiguration	RAIDConfiguration	false

# Server

Name: Server

Property Name	Property Type	Required
slotType	ServerSlotType	false
model	string	false
managementIPStatic	boolean	false
assetTag	string	false
health	Health	false
managementIP	string	false
serviceTag	string	false
slotName	string	false
id	string	false
slot	string	false
supported	boolean	false

# ServerSlotType

Enum: ServerSlotType

Property Name	Property Type	Required
values	["HALF", "FULL", "QUARTER", "UNKNOWN"]	

# ServiceTemplate

Name: ServiceTemplate

Property Name	Property Type	Required
components	<a href="#">[component]</a>	false
updatedAt	GregorianCalendar	false
createdAt	GregorianCalendar	false

Property Name	Property Type	Required
updatedBy	string	false
firmwareRepository	FirmwareRepository	false
templateDescription	string	false
wizardPageNumber	integer	false
draft	boolean	false
manageFirmware	boolean	false
allUsersAllowed	boolean	false
assignedUsers	<a href="#">[User]</a>	false
templateName	string	false
templateValid	ServiceTemplateValid	false
templateVersion	string	false
templateLocked	boolean	false
enableApps	boolean	false
enableCluster	boolean	false
enableServer	boolean	false
enableStorage	boolean	false
enableVMs	boolean	false
lastDeployedDate	GregorianCalendar	false
id	string	false
category	string	false
attachments	[string]	false
createdBy	string	false

## ServiceTemplateCategory

Name: ServiceTemplateCategory

Property Name	Property Type	Required
id	string	false
displayName	string	false
parameters	<a href="#">[ServiceTemplateSetting]</a>	false



## ServiceTemplateComponentType

Enum: ServiceTemplateComponentType

Property Name	Property Type	Required
values	["CONFIGURATION", "TOR", "STORAGE", "SERVER", "CLUSTER", "VIRTUALMACHINE", "SERVICE", "TEST"]	

## ServiceTemplateOption

Name: ServiceTemplateOption

Property Name	Property Type	Required
dependencyTarget	string	false
dependencyValue	string	false
name	string	false
value	string	false

## ServiceTemplateSetting

Name: ServiceTemplateSetting

Property Name	Property Type	Required
max	integer	false
min	integer	false
infolcon	boolean	false
networkConfiguration	NetworkConfiguration	false
toolTip	string	false
networks	[ <a href="#">Network</a> ]	false
possibleValuesDisplayName	[string]	false
possibleValues	[string]	false
dependencyTarget	string	false

Property Name	Property Type	Required
dependencyValue	string	false
raidConfiguration	RAIDConfiguration	false
hideFromTemplate	boolean	false
requiredAtDeployment	boolean	false
value	string	false
id	string	false
type	ServiceTemplateSettingType	false
readOnly	boolean	false
displayName	string	false
group	string	false
options	<a href="#">[ServiceTemplateOption]</a>	false
generated	boolean	false
required	boolean	false

## ServiceTemplateSettingType

Enum: ServiceTemplateSettingType

Property Name	Property Type	Required
values	["BOOLEAN", "STRING", "PASSWORD", "INTEGER", "LIST", "TEXT", "NETWORKCONFIGURATION", "ENUMERATED", "RAIDCONFIGURATION", "BIOSCONFIGURATION"]	

## ServiceTemplateUploadRequest

Name: ServiceTemplateUploadRequest

Property Name	Property Type	Required
description	string	false
createCategory	boolean	false
encryptionPassword	string	false

Property Name	Property Type	Required
manageFirmware	boolean	false
assignedUsers	[string]	false
templateName	string	false
useEncPwdFromBackup	boolean	false
firmwarePackageld	string	false
managePermissions	boolean	false
allStandardUsers	boolean	false
content	string	false
category	string	false

## ServiceTemplateValid

Name: ServiceTemplateValid

Property Name	Property Type	Required
valid	boolean	false
messages	<a href="#">[AsmDetailedMessage]</a>	false

## SoftwareBundle

Name: SoftwareBundle

Property Name	Property Type	Required
description	string	false
updatedAt	string	false
createdAt	string	false
userBundle	boolean	false
bundleDate	string	false
updatedAt	string	false
deviceType	string	false
userBundlePath	string	false
fwRepositoryId	string	false
softwareComponents	<a href="#">[SoftwareComponent]</a>	false

Property Name	Property Type	Required
deviceModel	string	false
criticality	string	false
name	string	false
id	string	false
version	string	false
link	Link	false
createdBy	string	false

## SoftwareComponent

Name: SoftwareComponent

Property Name	Property Type	Required
updatedAt	string	false
createdAt	string	false
updatedBy	string	false
dellVersion	string	false
componentId	string	false
deviceId	string	false
vendorId	string	false
vendorVersion	string	false
hashMd5	string	false
packageId	string	false
subDeviceId	string	false
subVendorId	string	false
systemIDs	[string]	false
componentType	string	false
name	string	false
id	string	false
path	string	false
category	string	false
createdBy	string	false

## StaticNetworkConfiguration

Name: StaticNetworkConfiguration

Property Name	Property Type	Required
gateway	string	false
subnet	string	false
ipRange	<a href="#">IpRange</a>	false
dnsSuffix	string	false
primaryDns	string	false
secondaryDns	string	false
ipAddress	string	false

## testProxyResponse

Name: testProxyResponse

Property Name	Property Type	Required
testProxyResult	boolean	false
testProxyDescription	string	false

## timeZone

Name: timeZone

Property Name	Property Type	Required
timeZoneld	string	false
timeZone	string	false

## UIRaidLevel

Enum: UIRaidLevel

Property Name	Property Type	Required
values	["raid0", "raid1", "raid5", "raid6", "raid10", "raid50", "raid60"]	

## UpdateType

Enum: UpdateType

Property Name	Property Type	Required
values	["SERVICE", "DEVICE"]	

## User

Name: User

Property Name	Property Type	Required
enabled	boolean	false
userName	string	true
updatedAt	GregorianCalendar	false
createdDate	GregorianCalendar	false
updatedBy	string	false
userSeqId	integer	false
updatePassword	boolean	false
domainName	string	true
firstName	string	false
lastName	string	false
email	string	false
phoneNumber	string	false
systemUser	boolean	false
link	Link	false
role	string	true
password	string	true
createdBy	string	false

# VirtualDisk

Name: VirtualDisk

Property Name	Property Type	Required
mediaType	MediaType	false
controller	string	false
raidLevel	UIRaidLevel	false
physicalDisks	[string]	false
configuration	VirtualDiskConfiguration	false

# VirtualDiskConfiguration

Name: VirtualDiskConfiguration

Property Name	Property Type	Required
numberofdskis	integer	false
disktype	DiskMediaType	false
raidlevel	UIRaidLevel	false
id	string	false
comparator	ComparatorValue	false

# VM

Name: VM

Property Name	Property Type	Required
vmIpaddress	string	false
vmManufacturer	string	false
vmModel	string	false
vmServiceTag	string	false
certificateName	string	false

## wizardStatus

Name: wizardStatus

<b>Property Name</b>	<b>Property Type</b>	<b>Required</b>
isSetupCompleted	boolean	false
showGettingStarted	boolean	false
isDeploymentCompleted	boolean	false
isNetworkCompleted	boolean	false
isTemplateCompleted	boolean	false
isConfigureCompleted	boolean	false
seqId	integer	false
link	Link	false



## Appendix C - Ruby Code Usage

Several configuration options are available for the ASM Ruby module. In order to run the code examples, minimally one will need to set the username, password, and baseURI in the file AsmConfig.rb

```
require 'ASM'
require 'XMLPayload'

ASM::API::userName = 'admin'
ASM::API::password = 'abc123'

ASM::API::debug = false

# Globally enable automatic signing of all RestClient requests
#ASM::API::autoEnableAuth

ASM::API::baseURI = 'https://localhost/Api/V1'

# XMLPayload is a simple module to help process XML payloads
ASM::Payload.basePath = './xml_payloads'

# Set this to false if want to ignore problems with self-signed certs.
ASM::API::sslVerify = false

# hack to deal with self-signed certificates
module RestClient
  class Request
    def self.execute(args, & block)
      unless ASM::API::sslVerify
        args[:verify_ssl] = false
      end
      new(args).execute(& block)
    end
  end
end
```

The following Ruby code illustrates several methods for configuring ASM and signing requests.

```
require 'ASM'

ASM::API::userName = 'admin'
ASM::API::password = 'abc123'

ASM::API::baseURI = 'http://localhost/Api/V1'

# Fetch a service template by name
url = ASM::API::URI('/ServiceTemplate')
template_name = 'uploaded'
template_filter = "eq,name,#{template_name}"

# Enable auto signing for all RestClient calls
ASM::API::enableAutoAuth
result = RestClient.get url, {:params=>{:filter=>template_filter}}
```

```
# Disable auto signing for all RestClient calls
ASM::API::disableAutoAuth
result = RestClient.get url, {:params=>{:filter=>template_filter}}

# Sign a single request
ASM::API::sign do
  result = RestClient.get url, {:params=>{:filter=>template_filter}}
  result = RestClient.get url, {:params=>{:filter=>template_filter}}
end

# Logout to demo automatic login below
ASM::API::logout

# Sign a single request with supplied creds
ASM::API::sign('admin','admin') {
  RestClient.get url, {:params=>{:filter=>template_filter}}
}
```