iControl® REST API 用户指南

版本 14,1

目录

REST

About Representational State Transfer

关于表述性状态传递

表述性状态传递
（rest）描述了web服务的一种体系结构样式，其中客户机和服务器交换资源的表示。rest模型将资源定义为信息源，并将表示定义为描述资源状态的数据。rest
web服务使用http协议在客户机和服务器之间进行通信，特别是通过post、get、put和delete方法来创建、读取、更新和删除元素或集合。一般来说，rest查询云科®系统的配置对象的资源，并创建、删除或修改这些配置对象的表示。

icontrol®rest实现遵循rest模型：

•使用rest作为基于资源的接口，并基于名词创建api方法。

•采用无状态协议和mime数据类型，并利用http协议中内置的身份验证机制和缓存。

•支持json格式进行文档编码。

•用统一资源标识符（uri）结构表示资源和集合的层次结构。

•返回HTTP响应代码以指示操作的成功或失败。

•在资源引用中包含链接以适应发现。

概述：URI 格式和结构

rest体系结构的原理描述了通过统一资源标识符（uniform resource
identifier，uri）来标识资源。uri标识web资源的名称；在本例中，uri还表示tmsh中模块和组件的树结构。您可以使用web服务请求指定uri来创建、读取、更新或删除云科
®系统配置的某些组件或模块。在rest体系结构的上下文中，系统配置是资源表示的同义词，web服务使用icontrol®rest api请求读写该表示。

注:

对 iControl REST 的请求使用 ADMIN(默认的管理帐户)。一旦熟悉了 API，就可以为具有各种权限的 iControlREST 用户创建用户帐户。

对于这里显示的URI片段，uri 的管理 ip 是完全限定域名或 ip 地址。

```
https://<management-ip>/mgmt/tm/...
```

在icontrol
rest中，所有请求的uri结构都包含字符串/mgmt/tm/以标识流量管理的命名空间。附加到该字符串的任何标识符都指定集合。

```
https://<management-ip>/mgmt/tm/...
```

代码段中的省略号指示指定组织集合的位置，该集合是指向IControl
rest中其他资源的链接的集合。组织集合是tmsh中模块的功能等价物。换句话说，icontrol
rest中的组织集合apm是apm模块。在icontrol
rest中，可以使用以下uri访问apm集合中的所有资源：

```
https://192.168.25.42/mgmt/tm/apm
```

下面的示例中的uri扩展了这种方法，它指定了报表集合中的所有资源。可以将集合视为tmsh子模块的等价物。icontrol rest集合包含集合或资源。

```
https://192.168.25.42/mgmt/tm/apm/report
```

以下示例中的uri指定一个资源，它是一组实体。在icontrol rest中，实体是一个可以配置的属性，例如“destaddrmax”：2048。资源也可以包含子集合。用tmsh的话说，资源相当于一个组件。

```
https://192.168.25.42/mgmt/tm/apm/report/default-report
```

Important: *iControl REST only supports secure access through HTTPS, so you must include credentials with each REST call. Use the same credentials you use for the 云科 device manager interface.*

重点 icontrol rest 只支持通过 https 进行安全访问，因此每次 rest 调用都必须包含凭据。使用与云科设备管理器接口相同的凭据。

关于保留的 ascII 字符

要容纳使用字符（不属于未保留的 ascii 字符集）的云科配置对象，请使用百分号（%）和两个十六进制数字在 uri 中表示它们。未保留的字符集包括：[a-z][a-z][0-9]短划线（-）、下划线（u）、句点（.）和颚化符（~）

必须对不属于未保留字符集的任何字符进行编码，以便包含在 uri 方案中。例如，非默认路由域中的 IP 地址包含一个百分号以指示特定路由域中的地址，例如 192.168.25.90%3，应进行编码以将%字符替换为%25。

关于 REST 资源标识符

A URI is the representation of a resource that consists of a protocol, an address, and a path structure to identify a resource and optional query parameters. Because the representation of folder and partition names in `tmsh` often includes a forward slash (/), URI encoding of folder and partition names must use a different character to represent a forward slash in iControl® REST. To accommodate the forward slash in a resource name, iControl REST maps the forward slash to a tilde (~) character. When a resource name includes a forward slash (/) in its name, substitute a tilde (~) for the forward slash in the path. For example, a resource name, such as `/Common/plist1`, should be modified to the format shown here:
uri是由协议、地址和路径结构组成的资源的表示，用于标识资源和可选查询参数。由于tmsh中文件夹和分区名称的表示通常包含正斜杠（/），因此文件夹和分区名称的uri编码必须使用不同的字符来表示icontrol®rest中的正斜杠。要在资源名称中容纳正斜杠，icontrol rest将正斜杠映射为波浪号（~）字符。当资源名称中包含正斜杠（/）时，请用颚化符（~）替换路径中的正斜杠。例如，资源名（如/common/plist1）应修改为如下所示的格式：

```
https://management-ip/mgmt/tm/security/firewall/port-list/~Common~plist1
```

关于 http 协议

超文本传输协议（http1.1）描述了建立在标识集合或资源的统一资源标识符（uri）上的方法和头。构成绝对路径的uri部分包括指定资源或集合路径的端点，如/mgmt。除了标识事务的x-yk-rest-coordination-id头之外，icontrol®rest不定义任何其他http头。集合是一组相同类型的资源，集合可以是资源集合，也可以是指向资源的链接的组织集合。在http方法的上下文中，uri将资源或集合标识为请求的目标。

除了资源的路径之外，查询参数还允许对get请求的结果集进行优化。查询字符串以问号（？）开头。字符，由优化响应数据的表达式组成。icontrol rest查询参数是odata查询参数以及几个自定义查询参数的实现。要区分自定义查询参数和OData查询参数，IControl rest custom query parameters将省略美元符号（$）作为参数的第一个字符。

根据uri的不同，icontrol rest方法的语义表现不同。对于post请求，uri指示请求在其下创建从属资源的资源。HTTP认为从属资源是一个新实体而不是对现有实体的修改。如果从属资源已经存在，则该协议考虑创建与错误相同的资源的请求。对于PUT请求，URI引用现有资源，并且请求修改现有资源。对于补丁请求，URI引用现有资源，并且请求将更改合并到资源中。

为了满足不同的需求，icontrol rest同时实现了patch和put方法。在icontrol rest中，patch方法只修改您在请求中指定的属性。put方法修改在请求中指定的属性，并将其余属性设置为默认值或空值。

icontrol rest方法的语义对于集合和资源的行为不同，如下表所述。

Method	描述
GET	对于集合和资源，iControlREST 支持 get。还支持查询字符串。
POST	对于集合和资源，iControlREST 都支持 POST。

DELETE	对于集合，iControlREST 不支持 delete。对于资源，iControlREST 支持 delete。
PUT	对于集合，icontrol rest 不支持 put。对于资源，icontrol rest 支持 put。对于 11.6 及更早版本，icontrol rest 仅部分支持资源的 put。

Method	描述
PATCH	对于集合，iControlREST 不支持 patch。对于资源，iControlREST 支持 patch。

关于 JSON 请求和响应语义

当icontrol
rest处理get请求时，它会生成一个响应代码和一个tjsonbody。同样，错误响应包含json格式的附加描述性文本。为了指示响应中文本体的格式，icontrol
rest将http内容类型头设置为application/json。icontrol
rest的响应包含描述配置对象或资源统计信息的属性。在icontrol
rest中，术语property指json对象中的名称/值或键/值对。

json术语由两种结构组成：对象和数组。对象是一个或多个名称/值对的集合，如图所示：

```
{ "partition":"Common" }
```

对于get请求，属性由json对象或数组或两者组成。请注意，名称和值以双引号（“”）显示，名称和值之间有冒号（：）分隔符。对于包含多个名称对的对象，其他名称/值对由逗号（，）分隔。来自icontrol rest的一个典型的（尽管有些过时）响应示例演示了json主体格式。

```
{
   "kind":"tm:ltm:ltmcollectionstate",
   "selfLink":"https://localhost/mgmt/tm/ltm?ver=11.5.0",
   "items":[
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/auth?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/classification?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/data-group?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/dns?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/global-settings?ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/html-rule?ver=11.5.0"
         }
      },
      {
         "reference":{
```

```
            "link":"https://../mgmt/tm/ltm/message-routing?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/monitor?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/persistence?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/profile?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/default-node-monitor?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/ifile?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/lsn-pool?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/nat?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/node?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/policy?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/policy-strategy?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/pool?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/rule?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/snat?
            ver=11.5.0"
         }
      },
      {
```

```
            "reference":{
               "link":"https://../mgmt/tm/ltm/snat-translation?ver=11.5.0"
            }
         },
         {
            "reference":{ "link":"https://../mgmt/tm/ltm/snatpool?
               ver=11.5.0"
            }
         },
         {
            "reference":{
               "link":"https://../mgmt/tm/ltm/traffic-class?ver=11.5.0"
            }
         },
         {
            "reference":{ "link":"https://../mgmt/tm/ltm/virtual?
               ver=11.5.0"
            }
         },
         {
            "reference":{
               "link":"https://../mgmt/tm/ltm/virtual-address?ver=11.5.0"
            }
         }
      ]
}
```

关于更多的 iControl REST 属性

实现包括一些traffic management
shell（tmsh）输出中不存在的文档属性。这些差异在表中记录，并显示在对集合或资源的get请求的响应中，如示例所示。

PropertyName	描述
kind	一个唯一的识别类型
Generation	资源的生成号。修改资源或相关资源会更改值。该值不一定单调递增。例如，如果修改子集合中的资源，则修改可能会导致父对象发生更改。
selfLink	一个指向此资源的链接

```
{
   "kind":"tm:sys:software:image:imagecollectionstate",
   "selfLink":"https://localhost/mgmt/tm/sys/software/image?ver=11.5.0",
   "items":[
      {
         "kind":"tm:sys:software:image:imagestate",
         "name":"BIGIP-11.5.0.0.0.191.iso",
         "fullPath":"BIGIP-11.5.0.0.0.191.iso",
         "generation":38,

"selfLink":"https://../mgmt/tm/sys/software/image/BIGIP-11.5.0.0.0.191.iso?ver=11.5.0",

         "build":"0.0.191",
         "buildDate":"Wed Nov 27 14 03 09 PST 2013",
         "checksum":"fab5b673486ccc1ec20fbe6cea51dyk0",
         "fileSize":"1751 MB",
         "lastModified":"Tue Dec  3 01:30:32 2013",
         "product":"云科",
```

```
         "verified":"yes",
         "version":"11.5.0"
      },
      {
         "kind":"tm:sys:software:image:imagestate",
         "name":"BIGIP-tmos-bugs-staging-11.5.0.0.0.237.iso",
         "fullPath":"BIGIP-tmos-bugs-staging-11.5.0.0.0.237.iso",
         "generation":37,

"selfLink":"https://../software/image/BIGIP-tmos-bugs-staging-11.5.0.0.0.237.iso?ver=11.5.0",

         "build":"0.0.237",
         "buildDate":"Wed Dec 4 14 14 44 PST 2013",
         "checksum":"bb4ae4838a5743fa209f67a1b56dedef",
         "fileSize":"1843 MB",
         "lastModified":"Wed Dec  4 15:32:28 2013",
         "product":"云科",
         "verified":"yes",
         "version":"11.5.0"
      }
   ]
}
```

```
root@(云科 1)(...)(tmos)# list sys software image
sys software image BIGIP-11.4.0.321.0.iso {
    build 321.0
    build-date "Mon Feb 11 07 23 24 PST 2013"
    checksum f9411fde01d6a3521d4ae393e9bb077c
    file-size "1522 MB"
    last-modified "Mon Feb 11 09:35:50 2013"
    product 云科
    verified yes
    version 11.4.0
}
root@(云科 1)(...)(tmos)#
```

关于空值和属性

.标志通常由软件组合成一个位，用于指示状态，例如0或1，并分别指示开或关。icontrol®rest显示用标志名和空值设置的标志。如果标志的值为none，则icontrol rest将从输出中忽略该属性。

注意 若要post 或 put 只有一个值的标志，请在 json 正文中输入值为空的属性名

```
{
   "kind":"tm:sys:software:volume:volumecollectionstate",
   "selfLink":"https://localhost/mgmt/tm/sys/software/volume?ver=11.5.0",
   "items":[
      {
         "kind":"tm:sys:software:volume:volumestate",
         "name":"MD1.1",
         "fullPath":"MD1.1",
         "generation":34,

"selfLink":"https://localhost/mgmt/tm/sys/software/volume/MD1.1?ver=11.5.0",
         "basebuild":"0.0.191",
         "build":"0.0.191",
         "product":"云科",
         "status":"complete",
```

```
         "version":"11.5.0",
         "media":[
            {
               "name":"MD1.1",
               "media":"array",
               "size":"default"
            }
         ]
      },
      {
         "kind":"tm:sys:software:volume:volumestate",
         "name":"MD1.2",
         "fullPath":"MD1.2",
         "generation":35,

"selfLink":"https://localhost/mgmt/tm/sys/software/volume/MD1.2?ver=11.5.0",
         "active":null,
         "apiRawValues":{

         },
         "basebuild":"0.0.237",
         "build":"0.0.237",
         "product":"云科",
         "status":"complete",
         "version":"11.5.0",
         "media":[
            {
               "name":"MD1.2",
               "defaultBootLocation":null,
               "media":"array",
               "size":"default"
            }
         ]
      },
      {
         "kind":"tm:sys:software:volume:volumestate",
         "name":"MD1.3",
         "fullPath":"MD1.3",
         "generation":36,

"selfLink":"https://localhost/mgmt/tm/sys/software/volume/MD1.3?ver=11.5.0",
         "status":"complete",
         "media":[
            {
               "name":"MD1.3",
               "media":"array",
               "size":"default"
            }
         ]
      }
   ]
}
```


About reserved property names 关于保留的特性名称

icontrol®rest保留了几个属性名，最明显的是单词name和generation。一些tmsh组件包含具有保留属性名的属性。当icontrol
rest在json主体中遇到保留名称时，它将用相应的替换名tmname或tmgeneration替换保留名称。

关于属性名称格式的差异

icontrol®rest 中的属性和选项名称使用的命名约定与流量管理（tmsh）shell 不同。在 tmsh 中，属性名由小写字符组成。对于包含多个单词的属性名称，连字符将单词分隔开。icontrol rest 对属性名使用大小写骆驼（camel case）约定，其中属性的第一个单词是小写的，名称中的所有其他单词都是大写的。

例如，属性 build date（如 tmsh 中所示）在 icontrol rest 中显示为 builddate。

关于 JSON 格式和编码

icontrol®rest支持以下字符串编码规范：

•W3CXML数字模式

•ISO 3166适用于国家和地区

•经纬度ISO 6709

•货币ISO

•日期和时间的RFC 3339

•奥尔森时区数据库

•持续时间可以表示为自Unix纪元（1970年1月1日00:00:00 UTC）以来的秒数，最多为小数时间的一微秒。

对于特定于配置中某个属性的日期和时间，将时间单位合并到名称中的属性名（如checkIntervalDays）提供有关时间单位的提示。

关于 API 版本

随着时间的推移，对icontrol®rest api的修改可能需要为版本分配新的版本号。为了将请求限制为特定版本的api，icontrol rest接受api版本参数作为uri的选项。要使用特定的api版本，请指定ver参数、api版本号（如11.5.0），并将字符串附加到uri的末尾，就像使用任何查询参数一样。

```
GET https://192.168.25.42/mgmt/tm/ltm?ver=11.5.0
```

响应的json主体包括selflink属性中的api版本号以及任何链接。对于icontrol rest，响应中资源的版本号与请求中发送的版本号匹配。如果不指定api的版本，则该版本默认为当前版本。为了保持与api未来版本的向后兼容性，响应将包含与请求中指定的版本号匹配的资源。如果icontrol rest无法生成与请求兼容的响应，则返回错误代码。

注意：尽管有些 rest 实现使用 http 头来管理版本信息，但 icontrolrest 不使用任何 http 头来标识 api。

---

iControl REST 密码更改

密码过期时，yk®icontrol®rest会阻止当前登录用户访问云科®系统上的资源。要继续使用云科系统，用户必须在访问系统上的资源之前创建新密码。密码更改的编程方法使用对/mgmt/tm/auth/user端点的修补程序请求，您可以访问该端点来更改密码。

如果选择使用icontrol
rest更改密码，则必须在json主体中提供密码。密码必须符合为您的组织建立的密码策略。您可以使用tmsh查找对您的组织有效的密码策略。如果您提供的密码满足密码要求，icontrol
rest将根据您的请求生成200 ok消息。如果密码不满足要求，则icontrol
rest将拒绝带有http错误响应的密码更改请求。

使用 iControlREST 请求更改密码

When your password expires, a 云科® system blocks access to resources. To re-enable access to resources, you can access the `/mgmt/tm/auth/user` endpoint to change your password.密码过期时，云科®系统会阻止对资源的访问。要重新启用对资源的访问，可以通过/mgmt/tm/auth/user来更改密码。

若去改密，请在 JSON 正文中提供新密码。

```
{
  "password": "<password>"
}
```

如果您提供的密码无法满足密码的要求（如密码长度或唯一性），则请求将失败。否则，请求将返回一条成功消息。可以使用tmsh命令列表验证密码策略来查找密码策略。

1. 向/mgmt/tm/auth/user 发出 patch 请求，并包含 json 主体。

```
PATCH https://192.168.25.42/mgmt/tm/auth/user
```

在此任务中，您通过发出 IControl®rest 请求更改了密码。

关于用户帐户的 iControl 和身份验证

IControl®rest不再要求您为单个用户帐户授予IControl
rest资源的权限。从版本12.0开始，用户自动拥有对rest资源的访问权，但用户必须获取用于身份验证的令牌，并在所有rest请求中包含该令牌。云科
®系统的管理员仍然可以使用基本身份验证发出rest请求。基本身份验证需要一个由用户ID、冒号（：）和密码组成的base64编码字符串。

请求用于 iControl REST 身份验证的令牌

作为云科
系统的管理员，您可以使用基本身份验证进行icontrol®rest调用。对于缺乏管理员权限的用户，用户必须请求一个令牌，该令牌可用于对发出rest api请求的用户进行身份验证。

1. 要创建身份验证令牌，请向云科
®系统发出POST请求。必须将名称和密码值都用双引号（“”）括起来，与任何json字符串一样。

```
POST https://172.68.25.42/mgmt/shared/authn/login

{
  "username": <user name>,
  "password": <user password>,
  "loginProviderName": "tmos"
}
```

文档指定 login reference，它引用登录提供程序。在本例中，login providername 属性允许您指定名称而不是引用。对于大多数情况，请使用 loginprovidername 和 tmos。

要在 rest 请求中使用令牌，请复制令牌属性的字符串并保存它。令牌由一系列随机字母和数字组成。在本例中，字符串是 492d3316e5456378b4ac9b5e2fa923595f0da65a。令牌的生命周期为 8 小时。

2. 要发出rest请求，请将令牌添加到请求头。必须像使用任何json字符串一样，将标记括在双引号（“”）内。

```
GET https://172.68.25.42/mgmt/tm/ltm

{ "X-YK-Auth-Token": "492D3316E5456378B4AC9B5E2FA923595F0DA65A" }
```

在本例中，您获取了要包含在 IControl rest 请求中的令牌。

概述：跨源资源共享的基本原理

浏览器中的同源策略控制两个不同源之间的交互，例如使用 xmlhttprequest（xhr）对象的请求。此外，同源策略还指出，从特定网站下载数据的浏览器不能与另一个并非来自同一网站的资源交互，在该网站上，协议、端口号和主机名标识该网站。虽然有实现安全跨站点数据传输的机制，但跨源资源共享（cors）通过添加新的 http 头来描述或枚举一组源，以及在传输客户端数据之前确定请求的可行性，从而实现安全跨站点数据传输。cors 头允许客户机和服务器之间的通信，以建立此类请求的限制。

CORS 支持两种类型的请求：简单请求和飞行前请求。简单的请求由 get、head 或 post 请求组成。对于 post 请求，发送到服务器的数据的内容类型必须是 application/x-www-form-urlencoded、multipart/form data 或 text/plain。简单请求的一个 fi 条件是请求不设置自定义头。

对于修改 web 资源的 http 方法，cors 标准定义了一个 prefl 功能，使客户端能够确定服务器是否允许请求。如果请求包含 get、head 或 post 以外的方法，或指定 application/x-www-form-urlencoded、multipart/form 数据或带有 post 请求的 text/plain 以外的内容类型头。在客户端发送带有数据的请求之前，客户端使用 options 方法发出请求以查询服务器。

最后，客户端通过在请求中包含源 http 报头来发起跨源请求。客户端还包括跨源请求中的访问控制请求方法和访问控制请求头。允许跨源请求的服务器使用 http 访问控制 allow origin 头和请求源的值来响应访问控制请求方法头和支持的方法，以及访问控制请求头和支持的值。

跨源资源共享请求头部

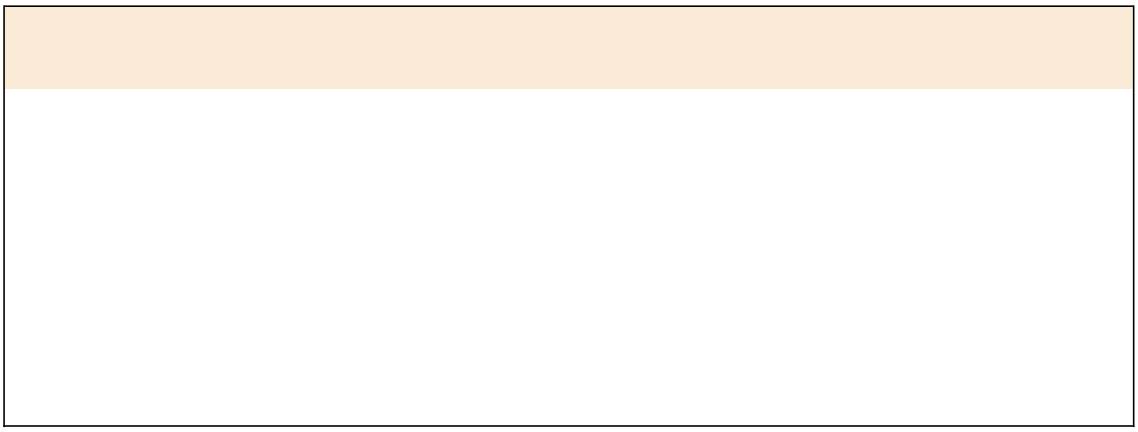
根据跨源资源共享（cors）规范，此表列出了客户端发送的请求头。

HTTP header	描述
Origin	指定指示交叉源或 preflight 源的 uri。
Access-Control-Request-Method 访问控制请求方法	具体说明客户端将在请求中发送的 http 方法。
Access-Control-Request-Headers 访问控制请求头部	具体说明在一个请求中客户端将包含的 http 头部

Cross-Origin Resource Sharing response headers 跨源资源共享响应头部

specification.根据跨源资源共享（CORS）规范，此表列出了服务器响应preflight请求而发送的响应头。

HTTP header	描述
Access-Control-Allow-Origin 访问控制允许源	指定允许访问资源的uri。 对于icontrol®rest用户，此标题列出允许请求的来源。IControl rest实现不允许通配符（*）。
Access-Control-Expose-Headers 访问控制公开头部	指定可安全公开的 http 头列表。对于 IControl rest 用户，此头是客户端可以访问的 YK®特定头的列表。
Access-Control-Max-Age 访问控制最长时间	指定缓存prefl请求结果的时间长度。在此时间段过期后，客户端应丢弃结果。该值要么是会话超时值的较小值，或者是一天。
Access-Control-Allow-Credentials	指示在凭据设置为 true 时是否公开响应。对于 icontrol rest 用户，此头表示允许 cors 请求中使用身份验证 cookie。将该值指定为 true。如果您不需要 cookies 进行身份验证，请不要指定此头。还必须将 xmlhttprequest 对象的 withcredentials 属性设置为 true，才能使 cors 请求成功。



关于带有 iControlREST 的外部身份验证提供程序

iControl REST支持对其他提供程序的外部身份验证，如active directory（ad）或radius。与云科系统上本地身份验证提供程序以外的提供程序进行身份验证需要一个令牌，您可以使用该令牌访问icontrol
rest中的资源。令牌由32个随机字符组成，主要是数字和大写的ascii字符，在一段时间内有效。在令牌过期之前，服务器将根据您提交的身份验证令牌验证您的身份。当令牌过期时，您只需从提供程序获取一个新令牌。

注意 在使用基于令牌的身份验证发出 rest 请求之前，必须从外部身份验证提供程序获取令牌。

您可以通过调用rest
api中的用户身份验证方法来创建令牌。在发出令牌创建请求之前，必须从系统管理员处获取标识外部身份验证提供程序的登录引用。要创建身份验证令牌，请发出post请求，并在请求的json主体中指定用户名、密码和登录引用。此请求将身份验证令牌与用户名关联。如果令牌创建请求成功，则响应包含与此类似的json主体。

REST


```
{
   "username":"auser",
   "loginReference":{

"link":"https://localhost/mgmt/cm/system/authn/providers/ldap/298d4aa5-d255-438f-997d-7f984109dd5d/login"

   },
   "token":{
      "uuid":"69c4b1c8-efdc-429a-b50c-723e92703a2b",
      "name":"492D3316E5456378B4AC9B5E2FA923595F0DA65A",
      "token":"492D3316E5456378B4AC9B5E2FA923595F0DA65A",
      "userName":"USERNAME",
      "user":{

"link":"https://localhost/mgmt/cm/system/authn/providers/ldap/298d4aa5-d255-438f997d7f984109dd5d/users/a25e2147-92e0-4349-ac99-7c844c3d30c2"

      },
      "groupReferences":[

      ],
      "timeout":1200,
      "startTime":"2014-07-08T17:14:34.305-0700",
      "address":"192.168.2.2",
      "partition":"[All]",
      "generation":1,
      "lastUpdateMicros":1404864874295548,
      "expirationMicros":1404866074305000,
      "kind":"shared:authz:tokens:authtokenitemstate",

"selfLink":"https://localhost/mgmt/shared/authz/tokens/69c4b1c8-efdc-429a-b50c-723e92703a2b"
```

```
    },
    "generation":0,
    "lastUpdateMicros":0
}
```

token 属性值标识要包含在请求中。在 json 主体中，标记是标记对象内部的字符串 492d3316e5456378b4ac9b5e2fa923595f0da65a。要通过资源验证，必须在 rest 请求中包含 x-yk-auth-token 头并在头中指定令牌值。如果您希望在本地进行身份验证，可以将授权头留空。有关获取和使用身份验证令牌的更多信息，请参阅 BIG-IQ®Systems:Rest API 参考。

Get 请求

Discovering modules and components 发现模块和组件

.icontrol®rest支持通过get请求进行发现。当您研究组织集合时，资源的结构变得更加明显。发现组织集合的另一个好处是icontrol rest和tmsh之间的关系。

要发现结构，请使用 get 方法请求 icontrol rest 并指定一个组织集合，如本例所示。

```
GET https://192.168.25.42/mgmt/tm/ltm
```

REST


```
{
   "items":[
      {
         "reference":{ "link":"https://localhost/mgmt/tm/ltm/auth?
            ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/classification?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/data-group?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/ltm/dns?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/global-settings?ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/html-rule?ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/message-routing?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/monitor?
            ver=11.5.0"
         }
      },
      {
```

```
      "reference":{ "link":"https://../mgmt/tm/ltm/persistence?
         ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/profile?
         ver=11.5.0"
      }
   },
   {
      "reference":{
        "link":"https://../mgmt/tm/ltm/default-node-monitor?ver=11.5.0"

      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/ifile?
         ver=11.5.0"
      }
   },
   {
      "reference":{
         "link":"https://../mgmt/tm/ltm/lsn-pool?ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/nat?
         ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/node?
         ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/policy?
         ver=11.5.0"
      }
   },
   {
      "reference":{
         "link":"https://../mgmt/tm/ltm/policy-strategy?ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/pool?
         ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/rule?
         ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/snat?
         ver=11.5.0"
      }
   },
   {
      "reference":{
         "link":"https://../mgmt/tm/ltm/snat-translation?ver=11.5.0"
      }
   },
   {
      "reference":{ "link":"https://../mgmt/tm/ltm/snatpool?
         ver=11.5.0"
      }
```

```
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/traffic-class?ver=11.5.0"
         }
      },
      {
         "reference":{ "link":"https://../mgmt/tm/ltm/virtual?
            ver=11.5.0"
         }
      },
      {
         "reference":{
            "link":"https://../mgmt/tm/ltm/virtual-address?ver=11.5.0"
         }
      }
   ],
   "kind":"tm:ltm:ltmcollectionstate",
   "selfLink":"https://localhost/mgmt/tm/ltm?ver=11.5.0"
}
```


如果您熟悉命令行工具，请使用 curl 或类似的实用程序来请求 icontrol rest。在 uri 中，指定一个组织集合。例如，命令 curl-k-u admin:admin-x get https://192.168.25.42/mgmt/tm/ltm 请求 ltm 组织集合。

注意 IControl rest 资源的内容可能不具有子集合级别下其 tmsh 对应资源的所有属性和选项。

注意 未在云科系统上设置的模块不会出现在输出中

Note:

关于分页属性

Icontrol rest 支持大型集合的分页选项。分页的实现利用开放数据协议（OData）查询参数提供可用于导航大型结果集的信息。请求大型集合时，icontrol rest 响应包括用于标识集合的 uri、结果集的下一页、结果集的上一页的属性，以及结果中的项目总数、总页数、当前页、每页的项目数，以及当前页中项目数的计数。icontrol®rest 根据筛选的结果集计算这些值。

属性	描述
selfLink	The URI of the collection, including any query parameters. 集合的 uri，包括任何查询参数。
nextLink	The next set of data in the result set. Includes the $skip query parameter in the link.结果中的下一组数据。在链接中包含$skip 查询参数。
previousLink	The previous set of data in the result set. Not present in the first set of data. 结果中的上一组数据。不在第一组数据中
currentItemCount	A count of the number of items in the result set, either as the value of the $top query parameter, or the remaining number of items if less than the number requested. 结果中项目数的计数，可以作为$top 查询参数的值，也可以是小于请求数量的剩余项目数。
itemsPerPage	The number of items to display per page.每页显示的项目数。
pageIndex	The current page in the result set.结果中的当前页。
totalPages	The total number of pages in the result set, equal to the result of (totalItems / itemsPerPage), rounded up to the next integer value.结果中的总页数，等于（totalitems/itemsperpage）的结果，四舍五入为下一个整数值。
startIndex	The index of the first item in the result set.结果中第一项的索引
totalItems	The number of items in the result set, as calculated by the $inlinecount=allpages query parameter. 由$inlinecount=allpages 查询参数计算的结果中的项目数。

About query parameters

IControl®rest为查询语言和系统查询选项实现了开放数据协议（OData）建议的子集。odata协议定义了系统查询选项，这些选项是查询字符串参数，用于管理由url标识的结果集中数据的表示。例如，可以包含或排除结果集中的行，将查询约束到管理分区中包含的资源，或指定icontrol rest的特定版本。除了asm模块之外，查询参数仅限于获取请求。

要使用查询参数，请将查询参数表达式附加到请求uri的末尾。所有查询参数表达式都以问号开头（？），后跟查询参数名、比较运算符或逻辑运算符以及值。值遵循icontrol rest的camel case命名约定。OData查询参数以美元符号（$）开头，而自定义查询参数不以美元符号开头。例如，可以指定响应仅包含以下请求中的name属性：

```
GET https://localhost/mgmt/tm/ltm/pool/?$select=name
```

若要指定其他查询参数，请在每个其他查询参数前面加上与号（&），然后指定查询参数表达式

。下表列出了OData查询参数的IControl
rest实现的参数。所有OData查询参数都以美元符号（$）开头。请注意，$filter参数（如果使用）将结果集限制为特定的管理分区。

Parameter 参数	Description 描述
`$filter`	Specifies an administrative partition to query for a result set. This parameter filters the result set by partition name and does not fully implement the corresponding OData query parameter. The `asm` module fully implements the OData query parameter.指定要查询结果集的管理分区。此参数按分区名称筛选结果集，但未完全实现相应的 OData 查询参数。asm 模块完全实现了 odata 查询参数。
`$select`	Specifies a subset of the properties that will appear in the result set. 指定将显示在结果集中的属性的子集。
`$skip`	Specifies the number of rows to skip in the result set. The result set is chosen from the remaining rows. 指定要在结果集中跳过的行数。结果集是从剩余的行中选择的
`$top`	Specifies the first N rows of the result set.指定结果的前 n 行。

icontrol rest支持odata建议中描述的比较运算符和逻辑运算符

Operator 操作	Description 描述
eq	Equal to 等于
ne	Not equal to 不等于
lt	Less than 小于
le	Less than or equal to 小于等于
gt	Greater than 大于
ge	Greater than or equal to 大于等于
and	True if both operands are true 与
or	True if either operand is true 或
not	Negation of operand 非

注意 icontrol rest 只支持带有$filter 参数的 eq 运算符。

icontrol rest包含几个自定义查询参数。自定义查询参数在参数名中不包含美元符号（$）字符。

Parameter	Description
`expandSubcollections`	Specifies that iControl REST expand any references to sub collections when set to `true`. By default, the response to a GET request only contains links for sub collection reference properties.指定 IControl rest 在设置为 true 时展开对子集合的任何引用。默认情况下，对 get 请求的响应仅包含子集合引用属性的链接。
`options`	Specifies the options to a query request. This parameter takes values that are compatible with the `tmsh` command-line options. 指定查询请求的选项。此参数采用与 tmsh 命令行选项兼容的值
`ver`	Specifies the version number of the iControl® REST API to use when making a request. Defaults to the current version if you do not specify a value. 指定发出请求时要使用的 IControl®rest API 的版本号。如果未指定值，则默认为当前版本

Paging through large collections 在大集合中分页

如果在单个get请求中处理包含大量项的集合，则会消耗大量的网络带宽和处理能力。查询参数允许您管理多页响应。icontrol®rest支持odata系统查询参数$top和$skip以返回页面项集。

使用$TopQueq参数指定 云 科
设备返回的最大项目数。如果使用curl并从unix命令行运行此命令，请在美元符号字符（$）前面加上反斜杠字符（\），以防止对字符进行shell解释。

```
curl -k -u admin:admin -X GET https://192.168.25.42/mgmt/tm/sys?\$top=4
```

要查询前n个数据项，请指定uri，并将$top查询参数附加到uri。此查询显示sys集合输出中的前四项。响应指示分别用作下一页和上一页导航标记的nextLink和previousLink属性。

要请求下一个 n 个数据项，请使用与上一个示例相同的 uri 并将$skip 查询参数附加到 uri。此示例显示 sys 集合输出中的下四个项。响应还指示作为数据导航标记的 nextLink 和 previousLink 属性。

关于子集扩展

icontrolrest支持expandsubcollections查询参数。在tmsh中，confi组件包含属性、子组件以及相关的非子组件。例如，可以独立于包含该组件的组件创建关联组件，例如包含LTM®池的虚拟服务器（tmsh中的LTM虚拟组件），即使将LTM池创建为单独的任务。

如果设置为true，expandSubCollections查询参数将显示所有子组件，但从响应中忽略任何关联的非子组件。

尽管该命令创建了一个冗长的输出块，但查询参数除了显示组件的属性外，还显示子集合的属性。与其他查询参数一样，expandSubCollections参数不支持get请求以外的请求。

```
https://192.168.25.42/mgmt/tm/ltm/virtual/my-VS/?expandSubcollections=true
```


```
{
   "kind":"tm:ltm:virtual:virtualstate",
   "name":"my-VS",
   "fullPath":"my-VS",
   "generation":1,

"selfLink":"https://../tm/ltm/virtual/my-VS?expandSubcollections=true&ver=11.5.0",

   "autoLasthop":"default",
   "cmpEnabled":"yes",
   "connectionLimit":0,
   "destination":"/Common/10.2.1.189:0",
   "enabled":null,
   "gtmScore":0,
   "ipProtocol":"tcp",
   "mask":"255.255.255.255",
   "mirror":"disabled",
   "mobileAppTunnel":"disabled",
   "nat64":"disabled",
   "pool":"/Common/my-Pool",
   "rateLimit":"disabled",
   "rateLimitDstMask":0,
   "rateLimitMode":"object",
   "rateLimitSrcMask":0,
   "source":"0.0.0.0/0",
   "sourceAddressTranslation":{
      "type":"automap"
   },
   "sourcePort":"preserve",
   "synCookieStatus":"not-activated",
   "translateAddress":"enabled",
   "translatePort":"disabled",
   "vlansDisabled":null,
   "vsIndex":2,
   "policiesReference":{
      "link":"https://../tm/ltm/virtual/~Common~my-VS/policies?ver=11.5.0",
      "isSubcollection":true,
      "items":[
         {
            "kind":"tm:ltm:virtual:policies:policiesstate",
            "name":"asm_auto_l7_policy  my-VS",
            "partition":"Common",
            "fullPath":"/Common/asm_auto_l7_policy  my-VS",
            "generation":1,
```

```
"selfLink":"https://../~Common~my-VS/policies/~Common~asm_auto_l7_policy  my-VS?ver=11.5.0"

         }
      ]
   },
   "securityLogProfiles":[ "\"/Common/Lo
      g illegal requests\""
   ],
   "fwRulesReference":{
      "link":"https://../tm/ltm/virtual/~Common~my-VS/fw-rules?ver=11.5.0",
      "isSubcollection":true
   },
   "profilesReference":{
      "link":"https://../tm/ltm/virtual/~Common~my-VS/profiles?ver=11.5.0",
      "isSubcollection":true,
      "items":[
         {
            "kind":"tm:ltm:virtual:profiles:profilesstate",
            "name":"http",
            "partition":"Common",
            "fullPath":"/Common/http",
            "generation":1,

"selfLink":"https://../tm/ltm/virtual/~Common~my-VS/profiles/~Common~http?ver=11.5.0",

            "context":"all"
         },
         {
            "kind":"tm:ltm:virtual:profiles:profilesstate",
            "name":"tcp",
            "partition":"Common",
            "fullPath":"/Common/tcp",
            "generation":1,

"selfLink":"https://../tm/ltm/virtual/~Common~my-VS/profiles/~Common~tcp?ver=11.5.0",

            "context":"all"
         },
         {
            "kind":"tm:ltm:virtual:profiles:profilesstate",
            "name":"websecurity",
            "partition":"Common",
            "fullPath":"/Common/websecurity",
            "generation":1,

"selfLink":"https://../tm/ltm/virtual/~Common~my-VS/profiles/~Common~websecurity?ver=11.5.0",

            "context":"all"
         }
      ]
   }
}
```


Expanding a sub-collection reference 扩展子集合引用

来自icontrol®rest的响应可以包括对子集合的引用。扩展子集合
查询参数扩展对子集合的引用。
查看特定资源的详细信息，包括其子集合的详细信息，追加字符串
expandSubCollections=对uri为true。不要在该查询参数前面加上美元符号（$）。

为了了解这些差异，此示例显示了对具有子集合扩展的资源的get请求。响应包含issubcollection属性，设置为true，以指示子集合。输出仅包含对子集合的引用。

```
https://192.168.42.25/mgmt/tm/ltm/pool/~Common~my-Pool
```

```
{ "allowNat" : "yes",
  "allowSnat" : "yes",
  "description" : "sdfds",
  "fullPath" : "/Common/my-Pool",
  "generation" : 1,
  "ignorePersistedWeight" : "disabled",
  "ipTosToClient" : "pass-through",
  "ipTosToServer" : "pass-through",
  "kind" : "tm:ltm:pool:poolstate",
  "linkQosToClient" : "pass-through",
  "linkQosToServer" : "pass-through",
  "loadBalancingMode" : "round-robin",
  "membersReference" : { "isSubcollection" : true,
     "link" : "https://../mgmt/tm/ltm/pool/~Common~my-Pool/members?ver=11.5.0"

    },
  "minActiveMembers" : 0,
  "minUpMembers" : 0,
  "minUpMembersAction" : "failover",
  "minUpMembersChecking" : "disabled",
  "name" : "my-Pool",
  "partition" : "Common",
  "queueDepthLimit" : 0,
  "queueOnConnectionLimit" : "disabled",
  "queueTimeLimit" : 0,
  "reselectTries" : 0,
  "selfLink" : "https://../mgmt/tm/ltm/pool/~Common~my-Pool?ver=11.5.0",
  "slowRampTime" : 10
}
```

要查看展开的子集合，此示例使用 expandSubCollections 查询参数。icontrol®rest 支持自定义 expandsubcollections 查询参数，该参数在名称中省略美元符号（$）。

```
https://192.168.25.42/mgmt/tm/ltm/pool/~Common~my-Pool/?expandSubcollections=true
```

```
{ "allowNat" : "yes",
  "allowSnat" : "yes",
  "description" : "sdfds",
  "fullPath" : "/Common/my-Pool",
  "generation" : 1,
  "ignorePersistedWeight" : "disabled",
  "ipTosToClient" : "pass-through",
  "ipTosToServer" : "pass-through",
  "kind" : "tm:ltm:pool:poolstate",
  "linkQosToClient" : "pass-through",
  "linkQosToServer" : "pass-through",
  "loadBalancingMode" : "round-robin",
  "membersReference" : { "isSubcollection" : true,
      "items" : [ { "address" : "1.1.1.1",
            "connectionLimit" : 0,
            "dynamicRatio" : 1,
            "fullPath" : "/Common/block:0",
            "generation" : 1,
```

```
            "inheritProfile" : "enabled",
            "kind" : "tm:ltm:pool:members:membersstate",
            "logging" : "disabled",
            "monitor" : "default",
            "name" : "block:0",
            "partition" : "Common",
            "priorityGroup" : 0,
            "rateLimit" : "disabled",
            "ratio" : 1,
            "selfLink" :
"https://../tm/ltm/pool/~Common~my-Pool/members/~Common~block:0?ver=11.5.0",
            "session" : "user-enabled",
            "state" : "unchecked"
         } ],
      "link" : "https://../tm/ltm/pool/~Common~my-Pool/members?ver=11.5.0"
    },
  "minActiveMembers" : 0,
  "minUpMembers" : 0,
  "minUpMembersAction" : "failover",
  "minUpMembersChecking" : "disabled",
  "name" : "my-Pool",
  "partition" : "Common",
  "queueDepthLimit" : 0,
  "queueOnConnectionLimit" : "disabled",
  "queueTimeLimit" : 0,
  "reselectTries" : 0,
  "selfLink" :
"https://../tm/ltm/pool/~Common~my-Pool?expandSubcollections=true&ver=11.5.0",

  "slowRampTime" : 10
}
```

从管理分区返回资源

要访问管理分区，请在 get 请求中使用$filter query 参数指定分区中的资源。

1.使用 uri 末尾的$filter query 选项访问 common 以外的分区。

2.通过创建以下字符串对 uri 进行编码：？ $filter=分区%20eq%20fw_objs

要使用筛选器参数，此示例显示一个 get 请求，该请求使用筛选器设置将查询限制到特定分区。请求的响应出现在第二个 block 中。

```
GET https://192.168.25.42/mgmt/tm/ltm/pool/?$filter=partition eq fw_objs
```

```
{
   "kind":"tm:ltm:pool:poolcollectionstate",

"selfLink":"https://../mgmt/tm/ltm/pool?$filter=partition%20eq%20fw_objs&ver=11.5.0",

   "items":[
      {
         "kind":"tm:ltm:pool:poolstate",
         "name":"tcb-pool2",
         "partition":"fw_objs",
         "fullPath":"/fw_objs/tcb-pool2",
         "generation":9587,
```

使用 iControlREST 获得统计输出

control®rest支持使用get请求生成统计输出。输出由只读统计数据组成，以json格式显示。使用/stats端点产生与tmsh show命令等效的统计输出。

要获取资源的统计结果，请将endpoint/stats附加到uri。

```
GET https://192.168.25.42/mgmt/tm/ltm/pool/stats
```

```
{
 "kind": "tm:ltm:pool:poolstats",
 "generation": 9,
 "selfLink": "https://localhost/mgmt/tm/ltm/pool/members/stats?ver=13.0.0",

 "entries":
  { "https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/stats":
  {
   "nestedStats": {
    "kind": "tm:ltm:pool:poolstats",
    "selfLink":
"https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/stats?ver=13.0.0",

    "entries":
     { "activeMemberCnt":
     {
      "value": 0
     },
     "availableMemberCnt":
      { "value": 1
     },
```

```
     "curSessions":
      { "value": 0
     },
     "memberCnt": {
      "value": 1
     },
     "minActiveMembers":
      { "value": 0
     },
     "monitorRule":
      { "description":
      "none"
     },
     "tmName": {
      "description": "/Common/pool1"
     },
     "serverside.bitsIn":
      { "value": 0
     },
     "serverside.bitsOut":
      { "value": 0
     },
     "serverside.curConns":
      { "value": 0
     },
     "serverside.maxConns":
      { "value": 0
     },
     "serverside.pktsIn":
      { "value": 0
     },
     "serverside.pktsOut":
      { "value": 0
     },
     "serverside.totConns":
      { "value": 0
     },
     "status.availabilityState":
      { "description": "unknown"
     },
     "status.enabledState":
      { "description":
      "enabled"
     },
     "status.statusReason": {
      "description": "The children pool member(s) either don't have service
 checking enabled, or service check results are not available yet"
     },
     "totRequests":
      { "value": 0
     },
    "https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/members/stats":
 {
      "nestedStats": {
       "kind": "tm:ltm:pool:members:membersstats",
       "selfLink":
"https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/members/stats?ver=13.0.0",

       "entries": {

"https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/members/~Common~1.1.1.1:80/stats":
 {
         "nestedStats": {
          "kind": "tm:ltm:pool:members:membersstats",
          "selfLink":
"https://localhost/mgmt/tm/ltm/pool/members/~Common~pool1/members/~Common~1.1.1.1:80/stats?ver=13.0.0",

          "entries": {
           "addr": {
```

```
            },
            "connq.ageEdm":
             { "value": 0
            },
            "connq.ageEma":
             { "value": 0
            },
            "connq.ageHead":
             { "value": 0
            },
            "connq.ageMax":
             { "value": 0
            },
            "connq.depth":
             { "value": 0
            },
            "connq.serviced":
             { "value": 0
            },
            "curSessions":
             { "value": 0
            },
            "monitorRule":
             { "description":
             "none"
            },
            "monitorStatus":
             { "description":
             "unchecked"
            },
            "nodeName": {
             "description": "/Common/1.1.1.1"
            },
            "poolName": {
             "description": "/Common/pool1"
            },
            "port":
             { "value":
             80
            },
              ... (truncated for readability) ...

            "sessionStatus":
             { "description":
             "enabled"
            },
            "status.availabilityState":
             { "description": "unknown"
            },
            "status.enabledState":
              { "description":
              "enabled"
            },
            "status.statusReason": {
             "description": "Pool member does not have service checking
enabled"
            },
            "totRequests":
             { "value": 0
            }
           }
          }
         }
        }
       }
      }
     }
```

统计信息作为嵌套对象组织在响应中。在每个嵌套级别，响应都包含一个nestedstats对象，该对象包含对象的条目。每个对象的元数据（kind，selflink）构成每个nestedstats对象块的一部分。

*注意*在版本 11.5 之前，响应对象不包含 NestedStats 对象。

POST 和 PUT 请求

关于 POST 和 PUT 的 JSON 格式

与 GET 请求不同，POST 或 PUT 请求包含 JSON 正文。 创建或修改资源时，可以使用与 GET 请求中所示相同的 JavaScript Object Notation（JSON）格式来定义对象的配置。 使用 POST 从 JSON 主体创建新的配置对象，并使用 PUT 或 PATCH 编辑具有 JSON 主体的现有配置对象。

JSON 主体的格式由遵循对象模型的对象组成，如下所示：

```
{ "partition":"Common" }
```

名称和值都用双引号引起来，并且冒号将名称和值对分开。 对于包含多个名称对的对象，逗号（，）分隔其他名称/值对。 JSON 值必须是对象，数组，数字，字符串或以下三个文字名称之一：false，null 或 true。 另一个结构是 JSON 数组或集合，它是值的有序列表，如下所示：

```
[ { "components":8, "isSubcomponent":"true" } ]
```

在 JSON 格式中，方括号将对象括在数组中。 数组中的对象遵循 JSON 标准的名称/值对。 名称/值对是云科系统配置的属性。 对于 iControl REST，可以将名称/值对视为属性名称和属性值。

在 REST 调用中，声明要发布的对象的格式。 对于 iControl REST，指定格式 application / json。 例如，在 curl 命令中，指定 HTTP 标头-H“ Content-Type：application / json”以声明 JSON 格式：

```
curl -k -u username:password -H "Content-Type: application/json"
           -X http-method uri
```

在 JSON 中，需要定义配置对象的名称。 然后，使用对与 GET 请求相似的名称和属性，包含该对象的属性名称和值。 您忽略的任何属性都将恢复为 PUT 请求的现有值或默认值。 如果使用 curl 等工具，则可以在命令行中指定 JSON 正文。 本指南中的几个示例演示了如何在命令行中包含 JSON 正文。

使用 iControl 创建新资源

使用 iControl®REST API，您可以通过在 iControl REST 集合上使用 POST 方法，并指定要创建为 JSON 的资源，将新资源添加到云科系统。 创建资源时，iControl REST 会将所有未指定的属性设置为其默认值。

要添加新的配置对象，请将资源名称指定为 JSON name/value，并在 URI 中指定集合的路径。

```
POST https://192.168.25.42/mgmt/tm/ltm/pool

{ "name":"tcb-pool-0" }
```

POST 操作的响应显示了一个新的配置对象。

```
{ "allowNat" : "yes",
  "allowSnat" : "yes",
  "fullPath" : "tcb-pool-0",
  "generation" : 5,
  "ignorePersistedWeight" :
  "disabled", "ipTosToClient" : "pass-
  through", "ipTosToServer" : "pass-
  through", "kind" :
  "tm:ltm:pool:poolstate",
  "linkQosToClient" : "pass-through",
  "linkQosToServer" : "pass-through",
  "loadBalancingMode" : "round-robin",
  "membersReference" : { "isSubcollection" :
      true, "link" :
"https://localhost/mgmt/tm/ltm/pool/~Common~tcb-pool-0/members?ver=11.6.0"
    },
  "minActiveMembers" : 0,
  "minUpMembers" : 0,
  "minUpMembersAction" : "failover",
  "minUpMembersChecking" : "disabled",
  "name" : "tcb-pool-0",
  "queueDepthLimit" : 0,
  "queueOnConnectionLimit" :
  "disabled", "queueTimeLimit" : 0,
  "reselectTries" : 0,
  "selfLink" : "https://localhost/mgmt/tm/ltm/pool/tcb-pool-
  0?ver=11.6.0", "serviceDownAction" : "none",
  "slowRampTime" : 10
}
```

通过发出 POST 请求创建新的 pool 对象后，即可使用该对象。

用 PATCH 修改资源

使用 PATCH 方法，可以在不影响任何其他属性的情况下修改资源的属性。

要在云科系统中修改对象，请在 URI 中指定资源。 不要在 URI 中指定集合。

```
PATCH https://192.168.25.42/mgmt/tm/pool/~Common~tcb-pool2

{"member": [{:name":"192.168.25.32:80", "description":"Tertiary web server"}]
 }
```

PATCH 请求后会显示对资源的更改。

```
{
   "kind":"tm:ltm:pool:poolstate",
```

```
    "name":"tcb-pool2",
    "partition":"Common",
    "fullPath":"/Common/tcb-
    pool2", "generation":59,
    "selfLink":"https://../mgmt/tm/ltm/pool/~Common~tcb-
    pool2?ver=11.5.0", "allowNat":"yes",
    "allowSnat":"yes",
    "ignorePersistedWeight":"disabled",
    "ipTosToClient":"pass-through",
    "ipTosToServer":"pass-through",
    "linkQosToClient":"pass-through",
    "linkQosToServer":"pass-through",
    "loadBalancingMode":"round-robin",
    "minActiveMembers":0,
    "minUpMembers":0,
    "minUpMembersAction":"failover",
    "minUpMembersChecking":"disabled",
    "queueDepthLimit":0,
    "queueOnConnectionLimit":"disabled"
    , "queueTimeLimit":0,
    "reselectTries":0,
    "slowRampTime":10,
    "membersReference":
    {

 "link":"https://../mgmt/tm/ltm/pool/~Common~tcb-
       pool2/members?ver=11.5.0", "isSubcollection":true
    }
}
```

完成 PATCH 请求后，您可以查看对单个资源的更改。

关于只读属性

如果使用 PUT 或 POST 方法指定只读属性，则 iControl®REST 会接受该请求并生成错误响应。 如果您指定了除只读属性之外的其他属性，则尽管包含了只读属性，有效的 PUT 或 POST 请求也不会产生错误。

例如，以下 curl 命令在现有 cm 设备对象 timeZone 中指定一个只读属性。 来自 iControl®REST 的响应指示缺少属性名称。 在这种情况下，iControl®REST 将忽略只读属性，并生成第二个块中显示的错误消息。

```
curl -k -u admin:admin -H "Content-Type:
    \ application/json" -X PUT -d \
    '{"time-zone":"EDT"}' \
    https://192.168.25.42/mgmt/tm/cm/device/bigip
    1
```

```
{
   "code":400,
   "message":"one or more properties must be
   specified", "errorStack":[

   ]
}
```

在特定分区中添加或修改

要在管理分区中添加或修改资源，请将 partition 属性添加到 JSON 主体以修改配置对象。 同时在命令行上使用查询选项，或在 JSON 正文中包含 partition 属性。 请注意，$ filter 查询参数仅适用于 GET 请求。

要使用 PUT 方法修改配置对象，请在分区中标识对象的分区属性。

This example uses the POST method to create a resource in a partition other than the `Common` partition. Specify the name of the resource, and the partition in which to create it, in the JSON body. The response to the request is shown in the third block.

```
POST https://192.168.25.42/mgmt/tm/ltm/pool
```

```
{ "name":"tcb-pool2", "partition":"~fw_objs" }
```

```
{
   "kind":"tm:ltm:pool:poolstate",
   "name":"tcb-pool2",
   "partition":"fw_objs",
   "fullPath":"/fw_objs/tcb-
   pool2", "generation":7810,

"selfLink":"https://localhost/mgmt/tm/ltm/pool/~fw_objs~tcb-pool2?ver=11.5.0",

   "allowNat":"yes",
   "allowSnat":"yes",
   "ignorePersistedWeight":"disabled",
   "ipTosToClient":"pass-through",
   "ipTosToServer":"pass-through",
   "linkQosToClient":"pass-through",
   "linkQosToServer":"pass-through",
   "loadBalancingMode":"round-robin",
   "minActiveMembers":0,
   "minUpMembers":0,
   "minUpMembersAction":"failover",
   "minUpMembersChecking":"disabled",
   "queueDepthLimit":0,
   "queueOnConnectionLimit":"disabled"
   , "queueTimeLimit":0,
   "reselectTries":0,
   "slowRampTime":10,
   "membersReference":
   {

"link":"https://../mgmt/tm/ltm/pool/~fw_objs~tcb-
      pool2/members?ver=11.5.0", "isSubcollection":true
   }
}
```

Following the creation of a new configuration object, this example modifies the member collection by using a PUT request. The URI includes the full path to the resource to modify. Specify the partition property, as

well as any properties you wish to modify. The partition property in the JSON body matches the folder name. The response to the request is shown in the third block.

```
PUT https://192.168.25.42/mgmt/tm/ltm/pool/~fw_objs~tcb-pool2
```

```
{ "name":"tcb-pool2", "partition":"/fw_objs",
"members": [ {"name":"192.168.25.32", "description":"Marketing server"} ] }
```

```
{
   "kind":"tm:ltm:pool:poolstate",
   "name":"tcb-pool2",
   "partition":"fw_objs",
   "fullPath":"/fw_objs/tcb-
   pool2", "generation":7914,

"selfLink":"https://localhost/mgmt/tm/ltm/pool/~fw_objs~tcb-pool2?ver=11.5.0",

   "allowNat":"yes",
   "allowSnat":"yes",
   "description":"This pool exists in the fw_objs
   partition.", "ignorePersistedWeight":"disabled",
   "ipTosToClient":"pass-through",
   "ipTosToServer":"pass-through",
   "linkQosToClient":"pass-through",
   "linkQosToServer":"pass-through",
   "loadBalancingMode":"round-robin",
   "minActiveMembers":0,
   "minUpMembers":0,
   "minUpMembersAction":"failover",
   "minUpMembersChecking":"disabled",
   "queueDepthLimit":0,
   "queueOnConnectionLimit":"disabled"
   , "queueTimeLimit":0,
   "reselectTries":0,
   "slowRampTime":10,
   "membersReference":
   {

"link":"https://../mgmt/tm/ltm/pool/~fw_objs~tcb-
      pool2/members?ver=11.5.0", "isSubcollection":true
   }
}
```

关于相对分区和文件夹名称

如果在分区正文中使用相对文件夹路径，则 iControl®REST 会参照于父分区来解释文件夹名称。根据请求的类型，通过在 URI 中指定$ filter = partition eq 文件夹名称查询参数或在 JSON 主体中指定 partition 属性来设置父分区。 $ filter 查询参数适用于 GET 请求，而 JSON 正文中的 partition 属性适用于 PATCH，POST 或 PUT 请求。 例如，如果$ filter = partition 查询选项设置为/ eu，并且 JSON 正文有对 france 文件夹的引用，则 iControl®REST 会将文件夹路径解释为/ eu / france。 为避免分区和文件夹名称含糊不清，请对 JSON 正文中的所有文件夹使用绝对路径，例如/ eu / france。

$ filter 查询参数与 OData 查询参数的不同之处在于，它仅支持 iControl REST 中按分区名称进行过滤。

删除访问策略管理器资源

使用 iControl®REST，您可以删除 Access Policy Manager™（APM™）资源。

要删除访问策略管理器（APM）资源（例如样本日志设置资源），请对/ mgmt / tm / apm / log-setting 路径中的资源发出 DELETE 请求。

```
DELETE https://192.168.25.42/mgmt/tm/apm/log-setting/sample-log-setting
```

iControl REST 不会为 DELETE 请求生成响应输出，但是您可以验证资源的删除。

分区

关于管理分区

许多类型的云科系统对象（例如配置文件和池）存储在管理分区中。分区是具有管理边界的容器，您可以使用访问权限对其进行控制。通过对管理分区的受限访问，安全模型对 config 对象施加了更大的控制权，从而减少了意外更改系统配置的可能性。

公用分区包含所有默认配置文件，预配置的监视器，默认身份验证 iRules，root 用户和 admin 用户帐户以及路由域 0，这是默认路由域。公共分区是由云科安装过程创建的。如果系统上没有其他管理分区，则将在“公共”分区中创建所有对象。所有管理员都可以访问公共分区。具有与其用户帐户关联的管理员或资源管理员角色的管理员可以创建分区。

创建其他分区时，可以将用户帐户与该分区关联，并授予管理该分区的权限。在大多数情况下，您要么授予用户对单个分区的访问权限，要么向用户授予对所有分区的通用访问权限。有权访问单个分区的用户只能在该分区中创建对象。如果授予用户对所有分区的通用访问权限，则用户必须通过在请求 URI 中指定 sys / folder 名称空间和文件夹名称来选择要在其中创建对象的分区。

每个分区在 sys / folder 命名空间中都有一个对应的文件夹，包括 Common 分区，该文件夹具有关联的/ Common 文件夹。创建或删除分区时，可以在 iControl®REST URI 中指定名称空间。

重要：无论您具有何种管理访问权限，都无法删除通用分区。

创建文件夹

您可以使用 iControl®REST 方法和属性来创建用于管理目的的文件夹。 有三种创建文件夹的方法。

重要提示：您必须向 ***iControl REST*** 发出单独的请求，才能在分区上分配用户权限。

---

1. 您可以通过指定路径和文件夹名称作为资源名称来创建根级别的文件夹。 要创建名为 fw_objs 的根目录文件夹，请发出 POST 请求，如下所示：

```
POST https://192.168.25.42/mgmt/tm/sys/folder

{
   "name": "fw_objs",
   "partition": "/"
}
```

结果对象将具有以下属性：

```
{
    "deviceGroup": "none",
    "fullPath":
    "/fw_objs",
    "generation": 393,
    "hidden": "false",
    "inheritedDevicegroup": "true",
    "inheritedTrafficGroup": "true",
    "kind":
    "tm:sys:folder:folderstate",
    "name": "fw_objs",
    "noRefCheck": "false",
    "selfLink":
    "https://localhost/mgmt/tm/sys/folder/~fw_objs?ver=...", "subPath":
    "/",
    "trafficGroup": "/Common/traffic-group-
    1", "trafficGroupReference": {
        "link":
"https://localhost/mgmt/tm/cm/traffic-group/~Common~traffic-group-1?ver=..."
```

2. 如果要在/ Common 文件夹中创建名为 fw_objs 的文件夹，可以通过在 name 属性中仅指定文件夹名称来进行。要在/ Common 文件夹中创建一个名为 fw_objs 的文件夹，请发出 POST 请求，如下所示：

```
POST https://192.168.25.42/mgmt/tm/sys/folder

{
   "name": "fw_objs"
}
```

在此步骤中，未指定路径，并且在 Common 分区（iControl REST 的默认分区）中创建了文件夹。 如果将上一步的结果对象与此步骤的结果对象进行比较，您会注意到当前对象不包括 subPath 属性或 partition 属性。 在“公用”分区中创建文件夹时，iControl REST 不会在响应中包括封闭的分区属性。

生成的对象具有以下属性：

```
{
    "deviceGroup":
    "none", "fullPath":
    "fw_objs",
    "generation": 403,
    "hidden": "false",
    "inheritedDevicegroup": "true",
    "inheritedTrafficGroup": "true",
    "kind":
    "tm:sys:folder:folderstate",
    "name": "fw_objs",
    "noRefCheck": "false",
    "selfLink":
    "https://localhost/mgmt/tm/sys/folder/fw_objs?ver=...",
    "trafficGroup": "/Common/traffic-group-1",
    "trafficGroupReference":
        { "link":
"https://localhost/mgmt/tm/cm/traffic-group/~Common~traffic-group-1?ver=..."
```

3. 通过指定文件夹对象的其他属性来创建文件夹层次结构。

要创建文件夹/ fw_objs / fw_objs，请使用 POST 请求并指定分区，子路径和名称。

```
POST https://192.168.25.42/mgmt/tm/sys/folder

{
   "partition": "/",
   "subPath":
   "fw_objs", "name":
   "fw_objs"
}
```

您可以在此示例中将 partition 属性指定为/ fw_objs，而不是分别指定 partition 和 sub 路径。 通常，顶级斜杠（'/'）和分区名称之间的所有内容都构成一个子路径。 否则，如第一个示例所示，当单个名称前面带有顶级斜杠（'/'）时，它将构成分区名称。

生成的对象具有以下属性：

```
{
    "deviceGroup": "none",
    "fullPath":
    "/fw_objs/fw_objs",
    "generation": 410,
    "hidden": "false",
    "inheritedDevicegroup": "true",
    "inheritedTrafficGroup": "true",
    "kind":
    "tm:sys:folder:folderstate",
    "name": "fw_objs",
    "noRefCheck": "false",
    "partition":
    "fw_objs", "selfLink":
"https://localhost/mgmt/tm/sys/folder/~fw_objs~fw_objs?ver=..."
    , "trafficGroup": "/Common/traffic-group-1",
    "trafficGroupReference": {
        "link":
"https://localhost/mgmt/tm/cm/traffic-group/~Common~traffic-group-1?ver=..."

    }
```

删除管理分区

可以使用 DELETE 请求删除除 Common 之外的管理分区。 在 URI 中，指定要删除的分区的文件夹名称，然后提交不带 JSON 正文的请求。 由于文件夹名称包含正斜杠，因此必须使用代字号字符指定文件夹名称。

着重提示：如果分区为空，才能删除该分区。 尝试删除分区之前，请删除分区中的所有对象。

要删除分区，请指定 DELETE 方法和文件夹路径 / mgmt / tm / sys / folder /
在 URI 中。 用波浪号（～）替换文件夹名称中的每个正斜杠（/）。

47

48

在此示例中，iControl®REST 请求从系统配置中删除/ fw_objs 分区。 响应包括指示成功或失败的响应代码，但是除非请求中有错误，否则响应不会生成 JSON 正文。

```
curl -k -u admin:admin -H "Content-Type: \
    application/json" -X DELETE \
    https://192.168.25.42/mgmt/tm/sys/folder/~fw_objs
    \
    |python -m json.tool
```

Transactions

关于 iControl REST 事务模型

云科®系统中的某些管理操作需要多个命令，在某些情况下，这些命令取决于其他命令的成功结果。为了适应此类复杂流程，iControl®REST 提供了事务，其中事务是作为单个工作单元执行的一系列单独命令的序列。事务的工作方式与关系数据库系统相似。在处理数据库事务时，如果所有 SQL 命令都成功运行，则关系数据库系统将提交更改。如果任何 SQL 命令失败，则关系数据库系统将回滚所有变化。 iControl REST 支持类似的功能，其中每个模块都对应了 Web 服务的一个功能。

用于创建，删除，修改或查询资源的 iControl REST 方法组成了事务的各个命令。但是，事务不是在到达时处理每个命令，而是将多个命令聚合到单个原子操作中。以这种方式，原子事务保证了事务的全部或全部语义。如果事务中的所有单个命令都成功完成，则事务成功完成。相反，如果事务中的任何命令失败，则整个事务都会失败。如果事务失败，则 iControl REST 将回滚在失败的操作之前完成的所有命令。

关于 iControl REST 事务阶段

事务的生命周期经历三个阶段：

Creation 创建

使用 POST 请求创建事务时，将发生此阶段。

Modification 修改

当将命令添加到事务中或对事务中的命令顺序进行更改时，将发生此阶段。

Commit 提交

当 iControl REST 运行事务时，将发生此阶段。

iControl REST 为事务保留一个名称空间。在事务框架内用于创建，删除，修改或查询资源的所有命令均使用 iControl REST 事务资源命名空间

/ mgmt / tm / transaction。此命名空间可防止命令在收到请求时由 iControl REST 自动运行。 iControl REST 会创建一个事务来响应包含空 JSON 正文的 POST 请求。作为响应，iControl REST 生成事务的标识符。创建事务时，事务资源将三个属性与该事务关联：

一个只读的 transId 属性，用于在交易有效期内标识交易。

指示事务状态的可写状态属性。此属性的值为：STARTED，UPDATING，

VALIDATING，COMPLETED 或 FAILED。除了提交事务外，您永远不会更改可写属性状态的值。

一个只读的 timeoutSeconds 属性，指定将命令添加到事务的时间段。 iControl REST 将该值设置为 120 秒。

在修改阶段，如果请求包含有效的事务标识符，则 iControl REST 会向事务添加命令。与创建事务的请求一样，添加命令的请求也是 POST 方法，用于指定事务名称空间。除了向事务中添加命令之外，还可以从事务中删除命令或更改事务中命令的顺序。命令按照接收顺序添加到事务中。 iControl REST 将命令标识符分配给添加到事务中的每个命令。对现有事务的任何更改（例如命令顺序的更改）都必须包含事务标识符和命令标识符。删除命令还需要事务标识符和命令标识符。

事务的最后阶段是提交阶段。准备好运行事务时，可以发出 PATCH 请求并指定事务状态，以向 iControl REST 指示它应运行该事务。您必须在请求中指定交易标识符。

注意：在 iControl REST 版本 11.6.0 中，您可以为每个用户创建多个事务。

关于提交验证

iControl®REST API 提供了一个属性来验证事务，而无需实际进行任何操作

确认对云科®系统进行了更改。 通过使用此属性，iControl REST 可以在尝试提交事务之前确定成功事务的可能性。 要使用此功能，请像往常一样创建一个事务，并在使用 PATCH 请求提交事务时在 JSON 主体中指定 validateOnly 属性。 如果事务请求未生成任何错误，则 iControl REST 将返回 HTTP 200 OK。

要验证交易请求，请在 PATCH 请求的 JSON 正文中指定“ validateOnly”：true。 该属性的值默认为 false。 如果在提交阶段以外的任何其他阶段指定属性，则 iControl REST 会忽略该属性。

关于提交的其他属性

与事务相关的两个值得注意的属性：executionTimeout 和 asyncExecution 属性。
executeTimeout 属性是一个只读属性，它指定事务在事务超时之前可以运行的时间。 为防止事务无限期运行，该属性将事务限制为 300 秒。 asyncExecution 属性是一个布尔属性，它允许事务在后台运行。 如果在提交请求中将“ asyncExecution”设置为 true，则该请求将返回 202 接受状态，以指示该事务已被接受以进行处理。 您必须轮询异步事务的状态。 如果不确定如何检查

请求的状态，请参阅有关创建 iControl®REST 事务的文档。

创建 iControl REST 事务

事务允许您将一系列命令作为单个工作单元运行。 在填充事务之前，必须通过指定事务端点来创建事务。

1. 要创建事务，请对/ tm / transaction 命名空间使用 POST 方法。 您必须在请求中包含一个空的 JSON 正文。

```
POST https://192.168.25.42/mgmt/tm/transaction
{ }
```

如果 POST 请求成功，则响应中包含事务标识符。 您必须在请求中包括事务标识符，以指示操作是事务的一部分。 注意响应中的三个事务属性：transId，state 和 timeoutSeconds。

```
{
   "transId":1389812351
   , "state":"STARTED",
   "timeoutSeconds":120,
   "kind":"tm:transactionstate"
   ,
   "selfLink":"https://localhost/mgmt/tm/transaction/1389812351?ver=11.5.0"
}
```

2. 要查看现有事务，请在查询请求中指定事务端点之一。 要检索集合中的所有事务，请指定 URI https：// <server name> / mgmt / tm / transaction。 要检索特定事务，请指定 URI https：// <server name> / mgmt / tm / transaction / <transId>，其中 transId 是交易的标识符。 如果未在一百二十（120）秒内将命令添加到事务中，则该事务将过期。

```
GET https://192.168.25.42/mgmt/tm/transaction
```

```
GET https://192.168.25.42/mgmt/tm/transaction/<transId>
```

修改创建后的事务

创建事务后，可以通过添加命令来填充事务。 各个命令包括事务执行的操作。 命令按接收顺序添加，但是您可以删除命令或更改事务中命令的顺序。

1. 要将命令添加到事务中，请使用以下指定的 post 方法。（示例中事务 ID 值的 X-YK-REST-Coordination-Id HTTP 标头：1389812351）。 在示例中，请求创建一个新池，并将单个成员添加到池中。

```
POST

https://192.168.25.42/mgmt/tm/ltm/pool X-

YK-REST-Coordination-Id:1389812351

{
   "name":"tcb-xact-pool",
   "members": [ {"name":"192.168.25.32:80","description":"First pool
for transactions"} ]
```

该响应表明 iControl®REST 已将操作添加到事务中。

```
{
   "transId":1389812351
   , "state":"STARTED",
   "timeoutSeconds":120,
   "kind":"tm:transactionstate"
   ,
   "selfLink":"https://localhost/mgmt/tm/transaction/1389813931?ver=11.5.0"
}
```

2. （可选）要查询单个事务，请指定 URI https：// <server name> / mgmt / tm / transaction / transId，其中 transId 是事务的标识符。

```
GET https://192.168.25.42/mgmt/tm/transaction/138912351
```

3. （可选）要获取事务中的命令列表，请指定 URI https：// <server name> / mgmt / tm / transaction / transId / commands，其中 transId 是事务的标识符。

```
GET https://192.168.25.42/mgmt/tm/transaction/138912351/commands
```

4. （可选）要获取单个操作的详细信息，请指定 URI https：// <server name> / mgmt / tm / transaction / transId / commands / commandId，其中，transId 是事务的标识符，commandId 是标识符 的操作。

```
GET https://192.168.25.42/mgmt/tm/transaction/138912351/commands/1
```

5. （可选）要从事务中删除命令，请指定 URI https：// <server name> / mgmt / tm / transaction / transId / commands / commandId，其中，transId 是事务的标识符，commandId 是的标识符 命令。 iControl REST 对事务中的其余命令重新编号。

```
DELETE https://192.168.25.42/mgmt/tm/transaction/138912351/commands/1
```

6. (可选）要更改评估顺序，请指定 URI https：// <server name> / mgmt / tm / transaction / transId / commands / commandId，其中，transId 是事务的标识符，commandId 是命令的标识符。 在 JSON 消息主体中，指定键/值对“ evalOrder”：y，其中 y 表示新的 evalOrder 值。 此操作将执行命令。

提交 iControl REST 事务

完成向事务中添加命令后，您对命令的评估顺序感到满意，然后可以通过提交事务来运行命令序列。 事务中的每个操作必须成功完成。 如果操作失败，则事务将回滚所有更改并返回错误。 如果您选择此时不运行该事务，则可以删除该事务。

1. 要提交事务，请使用 PATCH 方法。 在 JSON 主体中，将事务的状态指定为 VALIDATING。

```
PATCH https://localhost/mgmt/tm/transaction/1389812351
{ "state":"VALIDATING" }
```

2. （可选）要删除事务，请指定 URI https：// localhost / mgmt / tm / transaction / transId，其中 transId 是事务标识。 iControl®REST 删除与此事务关联的所有操作。

```
DELETE https://localhost/mgmt/tm/transaction/1389812351
```

关于 iControl REST 异步任务

iControl®REST 请求以同步方式运行，并在很短的时间内完成，通常只需几秒钟。单个 iControl REST 请求可能会运行更长的时间，并且这样做不会提供请求最终成功或失败的任何迹象。在某些情况下，请求可能会在请求完成之前超时。

iControl REST 通过允许某些端点执行异步任务来解决与长时间运行的请求相关的问题。长时间运行的请求通常需要 60 秒钟以上才能完成。如果端点表中存在要定位的端点，则应考虑将请求设为异步任务。对异步任务 URI 的 POST 请求通知 iControl REST 创建任务，然后响应任务状态的其他请求。作为对 POST 请求的初始响应的一部分，iControl REST 返回一个 JSON 主体，该主体包括用于轮询任务的自我链接。要监视异步任务，请创建任务，然后按标识符轮询任务以确定任务的状态。所有异步任务都处于以下状态之一：UPDATING，VALIDATING，COMPLETED 或 FAILED。 iControl REST 将任务的初始状态设置为 UPDATING，然后返回 HTTP 200 状态代码以指示任务的创建。

异步任务完成后，iControl REST 会将任务状态更改为 COMPLETED。对完成任务的轮询请求的响应包括 JSON 主体，该主体具有指向任务结果的自我链接。查看结果后，应按顺序删除结果，然后删除任务。

异步任务端点

下表列出了按功能组织的常见 iControl®REST API 以及相应的异步任务路径。

…	.	.
Save/Load config	POST tm/sys/config	POST tm/task/sys/config
Save/Load UCS	POST tm/sys/ucs	POST tm/task/sys/ucs
Load IP geolocation data	POST tm/sys/geoip	POST tm/task/sys/geoip
Load classification signatures	POST tm/sys/classification-signature	POST tm/task/sys/classification-signature
Failover	POST tm/sys/failover	POST tm/task/sys/failover
Load ®	POST tm/ltm/dns-express-db	POST tm/task/ltm/dns-express-db
Load URL DB feed list	POST tm/ltm/classification/urldb-feed-list	POST tm/task/ltm/classification/urldb-feed-list

Load classification	POST tm/ltm/classification/signatures	POST tm/task/ltm/classification/signatures
Update signatures	POST	POST
Install	POST tm/apm/epsec/epsec-package	POST tm/task/apm/epsec/epsec-package
Create	POST tm/vcmp/guest	POST tm/task/vcmp/guest
Verify	POST tm/wom/verify-config	POST tm/task/wom/verify-config
Diagnose	POST tm/wom/diagnose-conn	POST tm/task/wom/diagnose-conn
Load/Save/Publish	POST tm/wam/policy	POST tm/task/wam/policy
Load	POST tm/security/firewall/fqdn-entity	POST tm/task/security/firewall/fqdn-entity
Load IP intelligence	POST	POST
Load/ update	POST tm/security/anti-fraud/signatures-update	POST tm/task/security/anti-fraud/signatures-update
Load/update anti-fraud engine	POST tm/security/anti-fraud/engine-update	POST tm/task/security/anti-fraud/engine-update
Loa	POST tm/pem/subscribers	POST tm/task/pem/subscribers
Start/Stop/Restar	POST tm/ilx/plugin	POST tm/task/ilx/plugin
Add device to trust	POST tm/cm/add-to-trust	POST tm/task/cm/add-to-trust
Remove device	POST tm/cm/remove-from-trust	POST tm/task/cm/remove-from-trust

使用异步任务

异步任务为长期运行的同步任务提供了替代方法。

1. 要创建异步任务，请在异步任务终结点表中找到任务的终结点。 对于此示例，为/ tm / sys / ucs（/ tm / task / sys / ucs）标识相应的路径，并提供 JSON 正文。

```
POST https://192.168.25.42/mgmt/tm/task/sys/ucs

{
   "command": "save",
   "name": "myUcs"
}
```

在请求的响应中，找到参考路径（selfLink）以查询任务状态。 您将在后续步骤中使用这个结束方式。

```
{
   "command":"save",
   "name":"myUcs",
   "selfLink":"https://localhost/mgmt/tm/task/sys/ucs/1234&ver=12.0.0"
   , "_taskID":"1234",
   "_taskState":"UPDATING"
   ,
   "_taskTimeInStateMs":0,

"_taskResultLink":"https://localhost/mgmt/tm/task/sys/ucs/1234/result&ver=12.0.0",

   "_taskWaitTime":30000
}
```

要启动任务，请在 PUT 请求中修改任务的状态，并将 selfLink 指定为路径。 将 VALIDATING 指定为_taskState 属性的值。 您可以安全地省略 URI 中的 version 参数。

请注意，如果您不修改任务的状态，则该任务将不会运行，最终将被删除。

```
PUT https://192.168.25.42/mgmt/tm/task/sys/ucs/1234

{
   "_taskState": "VALIDATING"
}
```

如果请求成功，您应该看到类似于以下内容的响应：

```
{
   "_code": 202,
   "errorStack":
   [],
   "message": "Task will execute asynchronously."
}
```

验证异步任务的状态。

2. 要监视任务的进度，您可以定期向参考路径发出 GET 请求以检查任务的状态。

```
GET https://192.168.25.42/mgmt/tm/task/sys/ucs/1234
```

在某个时候的响应应表明任务已完成。

```
{
   "_taskId":1234,

"_taskResultLink":"https://localhost/mgmt/tm/task/sys/ucs/1234/result&ver=12.0.0",

   "_taskState":"COMPLETED"
   ,
   "_taskTimeInStateMs":0,
   "selfLink":"https://localhost/mgmt/tm/task/sys/ucs/1234&ver=12.0.0"
}
```

4. 任务完成后，向结果发出 GET 请求。

```
GET https://192.168.25.42/mgmt/tm/task/sys/ucs/1234/result
```

在此示例中，您提交并启动了异步任务，并查看了任务的结果。 查看任务的结果后，删除结果，然后按 URI 删除初始任务。

命令

关于其他 **tmsh** 全局命令

并非所有的《 Traffic Management Shell（tmsh）参考》命令都有对应的 HTTP 方法。 对于资源的列表或显示请求，GET 请求可以很好地映射到所请求的操作，但是引用包括不直接对应于 HTTP 方法的全局命令。 iControl®REST 实现了以下 tmsh 命令集：

cp
generate
install
load
mv
publish
reboot
restart
reset_stats
run
save
send-mail
start
stop

iControl REST 通过将命令以及选项映射到 JSON 格式来支持这些 tmsh 命令。 tmsh 命令的 iControl REST 格式遵循以下一般方法：

使用 POST 方法。

在 URI 中为 tmsh 命令指定一个名称空间。

将命令和选项指定为 JSON 正文中属性的值。

要运行该命令，请使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/application/template 以及该命令的 JSON 正文。 在每个示例中，在请求正文中使用相对 URI。

使用 **cp** 命令

实用程序命令没有直接映射到 HTTP 方法的功能，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/application/template 以及 JSON 指定实用程序命令名称的主体。

要使用 cp 命令进行复制，请使用 POST 方法发出 iControl®REST 请求，并在 JSON 正文中指定属性。

要使用 cp 命令复制文件，请发出 POST 请求。 在 JSON 主体中，指定命令，文件名和目标文件名。

```
POST /mgmt/tm/sys/application/template
```

使用 **generate** 命令

全局命令（例如 generate）没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/ltm/rule 以及 JSON。 指定命令名称的主体。
要使用 generate 命令生成签名脚本，请使用 POST 方法发出 iControl®REST 请求，并在 JSON 正文中指定属性。

要使用 generate 命令生成签名脚本，请发出 POST 请求。 在 JSON 主体中，指定命令，脚本名称，选项和签名密钥。 签名密钥属性名称使用带连字符的名称，而不是 iControl®REST 的大小写命名约定。

使用 **install** 命令

诸如 install 之类的全局命令没有直接映射到 HTTP 方法。 因此，您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/software/image，以及一个指定命令名称的 JSON 正文。 本主题显示两个 install 命令示例。

1. 要使用 install 命令安装和更新组件，请使用 POST 方法和 JSON 正文发出 iControl®REST 请求。

```
POST /mgmt/tm/sys/software/image
```

```
{
    "command":"install",
    "name":"BIGIP-11.5.0.930.400.iso",
    "volume":"HD1.3"
}
```

2. 要执行相同的任务并利用 install 命令的选项，请按照前面的步骤操作，并在 JSON 正文中指定 create-volume 和 reboot 选项。 创建卷属性名称使用带连字符的名称，而不是 camel-casing convention 形式的 iControl REST。

```
POST /mgmt/tm/sys/software/image
```

```
{
    "command":"install"
    , "options":[
       {
          "create-volume":true
       },
       {
          "reboot":true
       }
    ],
    "name":"BIGIP-11.4.0.737.400.42.iso",
    "volume":"HD1.1"
}
```

使用 iControl REST 创建密钥

您可以使用 iControl®REST 请求向密钥端点发出请求，而不是使用 tmsh key 命令创建私钥。

要创建密钥，请使用 POST 方法发出 iControl REST 请求，并在 JSON 正文中指定密钥的名称。

```
POST https://192.168.25.42/mgmt/tm/sys/crypto/key
```

```
{
  "name": "key-no-part.key"
}
```

注意：您必须在要创建的密钥名称中指定扩展名（密钥）。 如果省略扩展名，尽管成功创建了密钥，***iControl REST*** 仍会生成错误响应。

```
POST /mgmt/tm/sys/config
```

```
{
  "kind": "tm:sys:crypto:key:keycollectionstate",
  "selfLink":
  "https://localhost/mgmt/tm/sys/crypto/key?ver\u003d13.1.0", "items": [
.
.
.
    {
      "kind": "tm:sys:crypto:key:keystate",
      "name": "/Common/key-no-part.key",
      "fullPath": "/Common/key-no-
      part.key", "generation": 44690,
      "selfLink":
"https://localhost/mgmt/tm/sys/crypto/key/~Common~key-no-part.key?ver\u003d13.1.0",

      "keySize": "2048",
      "keyType": "rsa-
      private",
      "securityType": "normal"
    },
.
.
.
  ]
}
```

使用 **load** 命令

诸如 load 之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/config 以及 JSON 指定命令名称的主体。 通过使用 POST 方法和 JSON 主体发出 iControl®REST 请求，使用 load 命令来加载云科®系统配置。

要使用 load 命令替换正在运行的配置，请发出 POST 请求。 在 JSON 正文中，指定命令。

```
{
    "command":"load",

    "name": "default"
}
```

使用 **mv** 命令

```
POST /mgmt/tm/asm/policy
```

像 mv 这样的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/cm/device 以及 JSON 指定命令名称的主体。

要使用 mv 命令进行复制，请使用 POST 方法发出 iControl®REST 请求，并在 JSON 正文中指定属性。

要使用 mv 命令移动或重命名对象，请发出 POST 请求。 在 JSON 正文中，指定命令，名称和目标：

```
POST /mgmt/tm/cm/device
```

使用 **publish** 命令

全局命令（例如发布）没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/asm/policy 带有指定命令名称的 JSON 正文。
通过使用 POST 方法发出 iControl®REST 请求并在 JSON 正文中指定属性来发布策略中的更改。

在 JSON 主体中，指定命令，策略名称和应用程序服务。 应用程序服务属性名称使用带连字符的名称，而不是 iControl REST 的驼峰式命名约定。

使用 reboot 命令

诸如重新启动之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys 以及一个 JSON 正文， 或指定命令的名称。

POST / mgmt / tm / sys 通过使用 POST 方法发出 iControl®REST 请求并在 JSON 主体中指定属性，来重新引导系统或将系统引导到其他卷中。

```
POST /mgmt/tm/sys/service
```

要使用 reboot 命令重新引导系统，请发出 POST 请求。 在 JSON 正文中，指定命令。

使用 **restart** 命令

诸如重新启动之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/service 以及 JSON 指定命令名称的主体。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 主体中指定属性来重新启动服务。

要使用重新启动命令重新启动服务，请发出 POST 请求。 在 JSON 主体中，指定命令和要重新启动的服务的名称。

使用 **reset-stats** 命令

诸如 reset-stats 之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/ltm/virtual，以及 指定命令名称的 JSON 主体。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 正文中指定属性来重置组件的统计信息。

要使用 reset-stats 命令重置组件的统计信息，请发出 POST 请求。 在 JSON 主体中，指定命令和组件名称。

```
POST /mgmt/tm/util/ping
```

使用 **run** 命令

像 run 这样的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/util/ping 以及 JSON。 指定命令名称的主体。
通过使用 POST 方法发出 iControl®REST 请求并在 JSON 主体中指定属性来运行程序。 。

要使用 run 命令运行命令，请发出 POST 请求。 在 JSON 主体中，指定命令和命令选项。

使用 **save** 命令

诸如 save 之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/config 以及 JSON。 指定命令名称的主体。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 正文中指定属性，来保存云科®系统的运行配置。

要使用 save 命令保存正在运行的配置，请发出 POST 请求。 在 JSON 正文中，指定命令。

要使用 save 命令可用的选项，请在 JSON 正文中指定命令和选项。

使用 **send-mail** 命令

诸如 send-mail 之类的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/analytics/application-security/ 报告，以及指定命令名称的 JSON 正文。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 正文中指定属性，向收件人发送电子邮件。

To send e-mail using the send-mail command, make a POST request. In the JSON body, specify the command. Specify the options, as well as the recipients, in the JSON body. Several of the property names use a hyphenated name instead of the camel case naming convention of iControl® REST.

```
POST /mgmt/tm/analytics/application-security/report
```

```
{
    "command":"send-mail",
    "view-by":"ip",
    "format":"pdf",
    "email-
    addresses":[
       "wchen@yk.com"
     ],
     "measures":[
        "illegal-transactions"
     ],
     "limit":20,
     "order-by":[
        {
           "measure":"illegal-
           transactions", "sort-
```

```
        }
      ],
      "smtp-config-override":"smtpserver"
}
```

使用 **start** 命令

start 这样的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/icall/handler/perpetual， 以及指定命令名称的 JSON 正文。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 主体中指定属性来启动服务。

要使用 start 命令启动服务，请发出 POST 请求。 在 JSON 主体中，指定命令和服务名称。

使用 **stop** 命令

像 stop 这样的全局命令没有直接映射到 HTTP 方法，因此您必须使用 POST 方法并指定一个绝对 URI，例如 https://192.168.25.42/mgmt/tm/sys/icall/handler/perpetual， 以及指定命令名称的 JSON 正文。

通过使用 POST 方法发出 iControl®REST 请求并在 JSON 正文中指定属性来停止服务。

To stop a service using the stop command, make a POST request. In the JSON body, specify the command and the name of the service.

```
{
    "command":"stop",
    "name":"perphd1"
}
```

应用安全管理

应用安全管理 和 iControl REST 比较

如果使用 Application Security Manager™（ASM™），则应了解 ASM 与 iControl REST 的区别。

Application Security Manager™（ASM™）与 iControl®REST 有很多共同点。 与 iControl REST 中的任何组织集合一样，ASM 支持 API 的发现，通用方法以及一组查询参数。 但是，ASM 提供了一些使其与 iControl REST 区别开来的功能，如下表所示。

- ASM 资源 URI 包含标识资源的 MD5 哈希。
- ASM 实现了更多的开放数据协议（OData）查询参数，函数和运算符。
- ASM 不实现自定义查询参数，例如 expandSubcollections。
- ASM 不支持/ stats 端点。
- ASM 支持任务，而不支持事务。
- 下表列出了 ASM 支持的 HTTP 方法。

GET	对于集合和其他资源，ASM支持GET方法进行检索或搜索。 ASM中的$ filter查询参数支持包括比iControl REST更多的选项。
POST	对于集合和其他资源，ASM支持POST方法来创建实体。 POST请求必须包含JSON正文，尽管JSON正文可能为空。
DELETE	对于大多数集合，ASM支持DELETE方法。 ASM支持删除与$ filter查询匹配的集合子集。
PUT	
PATCH	对于集合，ASM支持PATCH方法。 在ASM中，如果您在URI中指定查询选项，则PATCH可以更新多个实体。

ASM 实现了 OData version4，并为 OData 版本 3 字符串函数提供了一些支持。 下表列出了 ASM 支持的查询选项和功能的限制。

$filter	指定用于检索，更新或删除操作的过滤器。 在ASM中，$ filter支持contains，endswith，startswith和substringof。

$skip	指定要在结果集中跳过的行数。 从剩余的行中选择结果集。
$orderby	“指定显示项目的顺序。$ orderby 参数不能应用于扩展字段内部的子字段。 例如$orderby=requestPolicy/name on

与 iControl REST 一样，ASM 还支持 OData 协议描述的比较和逻辑运算符。 下表列出了 ASM 运算符。

and	如果两个操作数均为真运算符，则为真。
or	如果任一操作数为 true，则为 true。 在 ASM 中，$ filter 支持适

ASM 支持聚合的 OData 函数 SUM，AVG，MAX 和 MIN。 下表列出了

ASM 名称空间。

/tm/asm/policies	不支持更新或删除许多请求的集合。 此 namespace 中的集合： • /tm/asm/policies/<MD5Hash>/methods • /tm/asm/policies/<MD5Hash>/filetypes

名称空间	描述
	• /tm/asm/policies/<MD5Hash>/blocking-settings/violations • /tm/asm/policies/<MD5Hash>/blocking-settings/evasions • /tm/asm/policies/<MD5Hash>/blocking-settings/http-protocols • /tm/asm/policies/<MD5Hash>/blocking-settings/web-services-securities • /tm/asm/policies/<MD5Hash>/urls • /tm/asm/policies/<MD5Hash>/parameters • /tm/asm/policies/<MD5Hash>/urls/<MD5Hash>/parameters • /tm/asm/policies/<MD5Hash>/whitelist-ips • /tm/asm/policies/<MD5Hash>/gwt-profiles • /tm/asm/policies/<MD5Hash>/json-profiles • /tm/asm/policies/<MD5Hash>/xml-profiles • /tm/asm/policies/<MD5Hash>/signatures

检索 Application Security Manager 资源

与 iControl®REST 行为一致，Application Security Manager™（ASM™）支持查询名称空间/mgmt / tm / asm 中的路径。 与 iControl®REST 中的任何其他组织集合一样，您可以发出 GET 请求以发现 ASM 的资源。

1. 向路径/ mgmt / tm / asm 发出请求以查询 ASM 资源。
2. 要发现 ASM 的资源，请向根的路径（/ mgmt / tm / asm）发出 GET 请求，如本示例所示

```
GET https://192.168.25.42/mgmt/tm/asm

{
   "selfLink":"https://localhost/mgmt/tm/asm"
   , "kind":"tm:asm:asmcollectionstate",
   "items":[
      {
         "reference":{ "link":"https://localhost/mgmt/t
            m/asm/tasks"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/asm/signa
            ture-update"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/a
            sm/policies"
         }
      },
      {
         "reference":{
            "link":"https://localhost/mgmt/tm/asm/policy-templates"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/asm
            /signatures"
         }
      },
      {
```

```
         "reference":{ "link":"https://localhost/mgmt/tm/asm/signatu
            re-statuses"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/asm/sig
            nature-sets"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/asm/signat
            ure-systems"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/asm/a
            ttack-types"
         }
      }
   ]
}
```

3. 要展开响应中的链接之一，请发出另一个针对资源的 GET 请求。本示例扩展了上一个请求的响应中的链接之一。 请注意，每个 URI 都包含一个哈希字符串作为资源标识符。

```
GET https://192.168.25.42/mgmt/tm/asm/policies

{
   "selfLink":"https://localhost/mgmt/tm/asm/policies"
   , "kind":"tm:asm:policies:policycollectionstate",
   "items":[
      {
         "policyBuilderReference":{ "link":"https://../mgmt/tm/asm/policies/

MwavowFbOsSD-Fgt4trP6A/policy-builder"

         },
         "blockingSettingReference":{ "link":"https://../mgmt/tm/asm/policies/

MwavowFbOsSD-Fgt4trP6A/blocking-settings",

            "isSubCollection":true
         },
         "cookieReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-
            Fgt4trP6A/cookies", "isSubCollection":true
         },
         "hostNameReference":{ "link":"https://../mgmt/tm/asm/policies/Mwa

vowFbOsSD-Fgt4trP6A/host-names",

            "isSubCollection":true
         },
        "selfLink":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-Fgt4trP6A",

         "stagingSettings":{ "signatureSt
            aging":true,
            "enforcementReadinessPeriod":
            7
         },
         "versionDeviceName":"10000-1-
         E12U39.sh", "signatureReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-Fgt4trP6A/signatures",
```

```
            "isSubCollection":true
         },
         "createdDatetime":"2013-12-
         06T19:29:54Z", "filetypeReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-
            Fgt4trP6A/filetypes", "isSubCollection":true
         },
         "id":"MwavowFbOsSD-Fgt4trP6A",
         "modifierName":"admin",
         "versionDatetime":"2013-12-
         26T23:12:57Z", "subPath":"/Common",
         "versionLastChange":"Policy Attributes  [update]: Policy
Builder determined that security policy \"/Common/my-VS\" is unstable.",
         "active":true,
         "caseInsensitive":false
         , "name":"my-VS",
         "description":"",
         "fullPath":"/Common/my-VS",
         "policyBuilderEnabled":true
         , "trustXff":false,
         "partition":"Common",
         "attributes":{
            "pathParameterHandling":"as-
            parameters",
            "triggerAsmIruleEvent":"disabled",
            "maskCreditCardNumbersInRequest":true,
            "inspectHttpUploads":false,
            "maximumHttpHeaderLength":2048,
            "maximumCookieHeaderLength":2048,
            "useDynamicSessionIdInUrl":false
         },
         "xmlProfileReference":{ "link":"https://../mgmt/tm/asm/policies/Mwa

vowFbOsSD-Fgt4trP6A/xml-profiles",

            "isSubCollection":true
         },
         "methodReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-
            Fgt4trP6A/methods", "isSubCollection":true
         },
         "customXffHeaders":[

         ],
         "creatorName":"admin",
         "kind":"tm:asm:policies:policystate"
         , "urlReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-
            Fgt4trP6A/urls", "isSubCollection":true
         },
         "virtualServers":
            [ "/Common/my-
            VS"
         ],
         "headerReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-
            Fgt4trP6A/headers", "isSubCollection":true
         },
         "protocolIndependent":false,
         "lastUpdateMicros":1.386358822e+15
         , "signatureSetReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-Fgt4trP6A/signature-

            sets", "isSubCollection":true
         },
```

```
            400,
            401,
            404,
            407,
            417,
            503
         ],
         "parameterReference":{ "link":"https://../mgmt/tm/asm/policies/Mw

avowFbOsSD-Fgt4trP6A/parameters",

            "isSubCollection":true
         },
         "jsonProfileReference":{ "link":"https://../mgmt/tm/asm/policies/Mw

avowFbOsSD-Fgt4trP6A/json-profiles",

            "isSubCollection":true
         },
         "applicationLanguage":"utf-8",
         "enforcementMode":"transparent"
         , "isModified":false,
         "gwtProfileReference":{

"link":"https://../mgmt/tm/asm/policies/MwavowFbOsSD-Fgt4trP6A/gwt-profiles",

            "isSubCollection":true
         },
         "whitelistIpReference":{ "link":"https://../mgmt/tm/asm/policies/Mw

avowFbOsSD-Fgt4trP6A/whitelist-ips",

            "isSubCollection":true
         },
         "versionPolicyName":"/Common/Dummy-VS"
      }
   ]
}
```

4. 要搜索资源的属性，请发出 GET 请求，并将查询字符串附加到 URI，如本示例所示。

```
GET https://192.168.25.42/mgmt/tm/asm/policies?$filter=name eq my-VS
```

创建应用程序安全管理器资源

与 iControl®REST 行为一致，Application Security Manager™（ASM™）支持在路径 / mgmt / tm / asm 中创建资源。 与 iControl®REST 中的任何其他组织集合一样，您可以发出 POST 请求以在 ASM 中创建资源。

要创建新资源，请使用 namespace / mgmt / tm / asm 发出 POST 请求。

```
POST https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/urls

{
   "name": "/login.php",
   "protocol": "http",
```

```
   "description": "A Login Page"
}
```

```
{
   "id": "<MD5HASH>",
   "name": "/login.php",
   "kind":
   "tm:asm:policies:urls:urlState",
   "selfLink":
"https://localhost/mgmt/tm/asm/policies/<MD5HASH>/urls/XPiqHHfl7UsVKku63zrd-g",

   "protocol": "http",
   "type":
   "explicit",
   "staging": true,
   "description": "A Login Page",
   "modifiedDatetime": "1990-12-
   31T23:59:60Z", "allowed": true,
   "checkFlow": false,
   "navigationParameters": false,
   "checkMetachars": true,
   "clickjackingProtection":
   false, "contentProfiles: [
      {
         "headerName": "*",
         "headerValue": "*",
         "headerOrder":
         "default", "type":
         "http",
         "inClassification":
         false
      }
   ]
   "parameterReference":
      { "link":
"https://localhost/mgmt/tm/asm/policies/<MD5HASH>/urls/XPiqHHfl7UsVKku63zrd-g/parameters"
```

更新 Application Security Manager 资源

与 iControl®REST 行为一致，Application Security Manager™（ASM™）支持更新路径 / mgmt / tm / asm 中的资源。 与 iControl®REST 中的任何其他资源一样，您可以使用 PATCH 请求更新 ASM 集合或其他资源。

1. 要更新资源，请对路径 / mgmt / tm / asm 中的资源发出 PATCH 请求，并包含 JSON 正文。

```
PATCH https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/urls/

{
   "clickjackingProtection":
   true,
   "clickjackingtype":"Never"
}
```

2. 要使用单个请求更新多个 ASM 实体，请发出 PATCH 请求并在 URI 中指定查询参数。

```
PATCH https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/urls?$filter=type
 eq explicit
```

```
{ "staging": false }
```

在 Application Security Manager 中删除资源

与 iControl®REST 行为一致，Application Security Manager™（ASM™）的路径包括名称空间/ mgmt / tm / asm / tasks / import-policy /中的端点。 与 iControl REST 中的任何其他资源一样，您可以发出 DELETE 请求以删除 ASM 中的资源。

1. 要删除资源，请发出 DELETE 请求并在路径中指定/ mgmt / tm / asm / tasks / import-policy

```
DELETE
https://192.168.25.42/mgmt/tm/asm/tasks/import-policy/ZuJ5QPuFj9r_LwbrDgoPsg
```

```
{
   "isBase64":false,
   "status":"FAILURE"
   ,
   "name":"TCB policy",
   "lastUpdateMicros":1.389135008e+15
   ,
   "kind":"tm:asm:tasks:import-policy:import-policy-taskstate",

"selfLink":"https://../mgmt/tm/asm/tasks/import-

policy/ZuJ5QPuFj9r_LwbrDgoPsg",

   "filename":"tcbpolicy.xml",
   "id":"ZuJ5QPuFj9r_LwbrDgoPsg",
   "startTime":"2014-01-
   07T22:50:08Z"
```

2. 要删除多个实体，请发出 DELETE 请求并在 URI 中指定查询参数。

```
DELETE
https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/urls/?$filter=staging
eq true
```

应用程序安全管理器策略

如果使用 Application Security Manager™（ASM™）导入，导出或激活策略，则应了解 ASM 与 iControl REST 的不同之处。

iControl®REST 支持导入，导出和激活策略的 Application Security Manager™（ASM™）功能。 各个任务主题说明了请求的所有必需属性。

file	指定要导入的XML格式的内联内容。 对于导入请求，将输入内联内容。
isBase64	指示内联内容是否为Base64编码。 适用于输入和输出内容。

在 Application Security Manager 中导入策略

iControl®REST 支持 Application Security Manager™（ASM™）任务，以从另一个 ASM 系统导入策略。 您可以将导入的策略用作另一个系统上的基本策略。.

1. （可选）要上传从中导入策略的文件，请使用 POST 方法并指定/ tm / asm / file-transfer / upload。 您必须在请求中指定文件名。

```
POST https://192.168.25.42/mgmt/tm/asm/file-transfer/uploads/<filename>
```

2. 要导入策略，请向/ mgmt / tm / asm / tasks / import-policy 路径发出 POST 请求。
3. 在 JSON 主体中，指定一个标识导入数据源的属性。 您必须提供列表中一个属性：

   - file
   - filename
   - policyReferenceTemplate

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/import-policy

{
   "filename":
   "mypolicy.xml", "name":
   "NewPolicy"
}
```

```
{
    "id":
    "oqNah2PxtwwE4YyAHGekNQ",
    "name": "NewPolicy",
    "filename": "mypolicy.xml"
    "kind": "tm:asm:tasks:import-
    policy:importpolicytaskstate", "lastUpdateMicros":
    1370459676272126,
    "status":
    "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ",
```

4. 发出 GET 请求并在 URI 中指定 id 属性，以确定策略导入操作是否成功。响应显示结果和状态属性，这些属性指示请求成功。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ

{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:import-
    policy:importpolicytaskstate", "name": "NewPolicy",
    "filename": "mypolicy.xml"
    "lastUpdateMicros":
    1370459676272126, "status":
    "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-
    04:00", "endTime": "2013-06-05T15:14:56-
    04:00",
    "result": {
        "policyReference":
            { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"
        }
    }
```

Exporting a policy in Application Security Manager

iControl®REST 支持将策略导出到另一台服务器的 Application Security Manager™（ASM™）任务。 您可以将导出的策略用作另一个系统上的基本策略。

1. 要导出策略，请向/ mgmt / tm / asm / tasks / export-policy 发出 POST 请求。 您必须在请求中指定 filename 属性或 inline 属性。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/export-policy

{
   "filename":
   "exported_file.xml", "minimal":
   true, "policyReference": {
            "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w
"
    }
}
```

对请求的响应包含以下数据：

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "filename":
    "exported_file.xml",
    "policyReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w
"
    },
    "minimal": true,
    "kind": "tm:asm:tasks:export-
    policy:exportpolicytaskstate", "lastUpdateMicros":
```

```
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/export-policy/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. （可选）要确定策略导出操作的状态，请使用 GET 方法并指定请求的 ID。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/export-policy/oqNah2PxtwwE4YyAHGekNQ
```

对请求的响应包含以下数据：

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "filename":
    "exported_file.xml",
    "policyReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w
"
    },
    "minimal": true,
    "kind": "tm:asm:tasks:export-
    policy:exportpolicytaskstate", "lastUpdateMicros":
    1370459676272126,
    "status": "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/export-policy/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-
    04:00", "endTime": "2013-06-05T15:14:56-
    04:00",
    "result": {
        "filename":
```

3. （可选）要下载文件，请使用 GET 方法并指定/ tm / asm / file-transfer / downloads，以及导出文件的名称。 您必须在请求中指定文件名。

```
GET https://192.168.25.42/mgmt/tm/asm/file-transfer/downloads/<filename>
```

在 **Application Security Manager** 中应用策略

iControl®REST 支持 Application Security Manager™（ASM™）任务，以手动应用保护网站的策略。

1. 要应用策略，请使用/ tm / asm / tasks / apply-policy 路径发出 POST 请求。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/apply-policy

{
    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"
```

```
    }
}
```

对请求的响应包含以下数据：

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:apply-
    policy:applypolicytaskstate", "policyReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w
"
    },
    "lastUpdateMicros":
    1370459678272126, "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/apply-
    policy/oqNah2PxtwwE4YyAHGekNQ", "startTime": "2013-06-05T15:14:36-
    04:00"
```

2. 要确定应用策略操作的状态，请向同一路径发出 GET 请求。

```
GET https://192.168.25.42/mgmt/tm/asm/tasks/apply-policy
```

对请求的响应包含以下数据：

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:apply-
    policy:applypolicytaskstate", "policyReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w
"
    },
    "lastUpdateMicros":
    1370459678272126, "status":
    "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/apply-
    policy/oqNah2PxtwwE4YyAHGekNQ", "startTime": "2013-06-05T15:14:36-
    04:00"
```

在 Application Security Manager 中查找策略差异

您可以确定两个策略之间的差异，以解决、合并差异。 差异功能的这一方面仅在 iControl®REST API 中可用。

1. 要找到策略之间的差异，请选择两个策略进行比较，并在 JSON 正文中表示这些策略，如图所示。 您可以比较具有以下共同特征的任何两个策略：编码，区分大小写和协议独立性。

```
{
 "firstPolicyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/example_1"}
,
 "secondPolicyReference": {"link":
"https://localhost/mgmt/tm/asm/policies/example_2"
}
```

1. 向/ mgmt / tm / asm / tasks / policy-diff 端点发出 POST 请求，并包含您创建的 JSON 正文。

2.

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/policy-diff
```

该任务将端点返回给策略之间的差异的集合，例如出现在一个策略中但不在另一个策略中的实体。

```
{
 "firstPolicyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/example_1"},
 "secondPolicyReference": {"link":
"https://localhost/mgmt/tm/asm/policies/example_2"},
 "differenceReference": { "link":
"https://localhost/mgmt/tm/asm/policy-diffs/8AcZwsnx7gvk34CV22hYrw/differences/example?ver=13.1.0"},

 "lastUpdateMicros": 0,
 "id": ""
}
```

您已创建了两个策略之间的差异的集合，可用于合并和解决差异。

合并 Application Security Manager 中的策略差异

您可以使用策略差异任务的输出来合并两个策略之间的差异。 收集差异后，可以指定如何将差异合并到策略中。

要合并策略文件之间的差异，请向 POST 请求
/ mgmt / tm / asm / tasks / policy-merge 端点。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/policy-merge
```

```
{
  "policyDiffReference": {"link": "/mgmt/tm/asm/policy-diffs/example"},
  "addMissingEntitiesToFirst": true,
  "addMissingEntitiesToSecond": true,
  "handleCommonEntities": "ignore",
  "handleMissingEntitiesEnum": ["ignore", "accept-from-first",
"accept-from-second"],
  "itemFilter": ""
}
```

itemFilter 属性使您可以选择要合并的差异子集。 您可以将其视为应用于差异集合的$ filter 查询参数的等效项。

Application Security Manager signatures

如果使用 Application Security Manager™（ASM™）signatures，则应了解 ASM 与 iControl REST 的区别。

iControl®REST 支持 Application Security Manager™（ASM™）功能来检查，导出或更新签名。

属性	描述
文件	以 XML 格式指定内联导入或导出的内容。
inline	指示是否在响应中内联包含导出的签名。
isBase64	指示内联内容是 Base64 编码的（输入还是输出）。 如果将 inline 设置为 TRUE，则导出的签名是 Base64 编码的。
文件名	指定本地签名文件的名称。
isUserDefined	指示签名是否被认为是用户定义的签名。

检查在 Application Security Manager 的 signatures

iControl®REST 支持 Application Security Manager™（ASM™）任务，以检查签名以更新签名文件。

1. 要检查新签名，请对/ tm / asm / tasks / check-signatures 发出 POST 请求命名空间，并包含一个空的 JSON 正文（{}）。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/check-signatures
```

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:check-signatures:check-signaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/check-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. 要确定检查新签名操作的状态，请发出 GET 请求。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/check-signatures/oqNah2PxtwwE4YyAHGekNQ
```

对请求的响应包含以下数据：

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:check-signatures:check-signaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/check-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
        "updatesAvailable": false
    }
}
```

在 Application Security Manager 更新 signatures

iControl®REST 支持 Application Security Manager™（ASM™）任务来更新签名。

1. （可选）要上传文件以更新签名，请使用 POST 方法并指定 / tm / asm / file-transfer / upload 端点。 您必须在请求中指定文件名。

```
POST https://192.168.25.42/mgmt/tm/asm/file-transfer/uploads/<filename>
```

2. 要更新签名，请对/ tm / asm / tasks / update-signatures 命名空间发出 POST 请求，并包括一个空的 JSON 正文（{}）。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/update-signatures

{}
```

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:update-signatures:update-signaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/update-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

3. 要确定更新签名操作的状态，请发出 GET 请求。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/update-signatures/oqNah2PxtwwE4YyAHGekNQ
```

响应包含任务的结果。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:update-signatures:update-signaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/update-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
        "signatureStatusReference":
            { "link":
"https://localhost/mgmt/tm/asm/signature_statuses/vagoQLF6uOoBKvS8h3C19w"
        }
    }
}
```

在 pplication Security Manager 导出 signature

iControl®REST 支持 Application Security Manager™（ASM™）任务，以导出签名以在另一个 ASM 系统上使用。

1. 要导出签名，请对/ tm / asm / tasks / export-signatures 命名空间发出 POST 请求，并在 JSON 正文中指定输出文件的名称。

```
POST https://192.168.25.42//mgmt/tm/asm/tasks/export-signatures

{
   "filename": "exported_file.xml",
}
```

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "filename": "exported_file.xml",
    "kind": "tm:asm:tasks:export-signatures:exportsignaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/export-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. （可选）要确定导出签名操作的状态，请发出 GET 请求。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/export-signatures/oqNah2PxtwwE4YyAHGekNQ
```

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "filename": "exported_file.xml",
    "kind": "tm:asm:tasks:export-signatures:exportsignaturestaskstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/export-signatures/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
        "filename": "exported_file.xml",
    }
}
```

3. (可选）要下载文件，请使用 GET 方法并指定 / tm / asm / file-transfer / downloads 端点，以及导出文件的名称。 您必须在请求中指定文件名。

```
GET https://192.168.25.42/mgmt/tm/asm/file-tranfer/downloads/exported_file.xml
```

在 Application Security Manager 检查 signature 信息状态

iControl®REST 支持 Application Security Manager™（ASM™）功能来检索签名的签名状态信息。签名状态包括有关签名文件添加和删除的信息。

要检索签名状态信息，请对/ tm / asm / signature-statuses 发出 GET 请求命名空间。

```
GET https://192.168.25.42/mgmt/tm/asm/signature-statuses/<MD5HASH>
```

items 属性显示 signature 状态。

```
{
    "selfLink": "https://localhost/mgmt/tm/asm/signature-statuses",
    "kind": "tm:asm:signature-statuses:signature-statuscollectionstate",
    "items": [
        {
            "sigsAdded": 0,
            "isUserDefined": false,
            "readme": "Attack Signature Database packaged with version
11.5.0\n\n\ .... ",
            "sigsUpdatedMinor": 0,
            "sigsDeleted": 0,
            "modifiedSignatures": [],
            "loadTime": "2013-10-10T06:43:30Z",
            "sigsTotal": 0,
            "sigsUpdated": 0,
            "selfLink":
"https://localhost/mgmt/tm/asm/signature-statuses/cHzbviRdfEv6l_RRieAdqw",
            "kind": "tm:asm:signature-statuses:signature-statusstate",
            "timestamp": "2013-10-08T09:06:15Z",
            "sigsUpdatedMajor": 0,
            "id": "cHzbviRdfEv6l_RRieAdqw"
        }
    ]
}
```

在 Application Security Manager 检索 signature 系统

iControl®REST 支持 Application Security Manager™（ASM™）功能来检索签名系统。 您必须提供签名系统的 MD5 hash 才能检索。

要检索签名系统信息，请使用/ tm / asm / signature-systems 发出 GET 请求命名空间。

```
GET https://192.168.25.42/mgmt/tm/asm/signature-systems/MD5HASH
```

该响应显示 signature 系统信息，作为到资源的链接。

```
{
    "selfLink":
"https://localhost/mgmt/tm/asm/signature-systems/EStDgGiP9nSPgKBhSlDyvQ",
    "kind": "tm:asm:signature-systems:signature-systemstate",
    "name": "General Database",
```

```
    "id": "EStDgGiP9nSPgKBhSlDyvQ"
}
```

Application Security Manager 架构加载

如果使用 Application Security Manager™（ASM™）管理架构，则应了解 iControl®REST 如何支持架构加载任务。

iControl®REST 为 XML 模式文件上传提供了一个端点。 Application Security Manager™（ASM™）通过使用上载然后关联到策略的架构文件来验证传入的数据。

属性	描述
文件名	指定 XML 模式文件的名称。
内容	将文件内容指定为 XML。

在 Application Security Manager 加载架构文件

将 XML 模式文件与配置文件相关联需要能够上载 XML 模式文件。 上载架构文件之后，可以运行单独的任务以将验证文件与配置文件相关联。
要上传 XML 模式文件，请使用 POST 方法并在
/ tm / asm / policies 命名空间。

```
POST
https://192.168.25.42/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA/xml-validation-files

{
            "fileName": "softwareupdate.wsdl",
            "contents": "<validation></validation>"
}
```

```
{
            "selfLink":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA/xml-validation-files/d7loGosItLc_ODXuPz83Uw",

            "kind":
"tm:asm:policies:xml-validation-files:xml-validation-filestate",
            "fileName": "softwareupdate.wsdl",
            "contents": "<begin></begin>",
            "lastUpdateMicros": 1393332020000000,
            "id": "d7loGosItLc_ODXuPz83Uw",
            "isReferenced": false
}
```

Application Security Manager 策略重构

如果使用 Application Security Manager™（ASM™）还原策略，则应了解 iControl®REST 如何实施 ASM。

iControl®REST 支持 Application Security Manager™（ASM™）功能，以基于策略历史记录还原策略。 还原策略修订时，必须在请求正文中包含 policyHistoryRevision 属性，并指定要从中还原的策略修订。 如果在请求的正文中提供了 policyReference 属性或 name 属性，则该任务将覆盖该策略。 否则，任务将创建新策略。

属性	描述
policyHistoryRevision	指定要还原的历史修订的链接。

在 Application Security Manager 还原策略修改

Application Security Manager™（ASM™）中的 policyHistoryReference 属性使任务可以还原策略修订。 如果 JSON 正文包含 policyReference 或 name 属性，则该任务将覆盖该策略。 否则，任务将创建新策略。

1. 要恢复策略修订，请对/ tm / asm / task / import-policy 名称空间使用 POST 方法。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/import-policy

{
   "policyHistoryReference":
      { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/history-revisions/hGKdiXU7US4S4qtgexijUQ"

   },
   "policyReference": {
     "link": "https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"

   }
}
```

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:import-policy:importpolicytaskstate",
    "policyHistoryReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/history-revisions/hGKdiXU7US4S4qtgexijUQ"

    },
    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"
    },
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. 要检查请求的状态，请使用/ tm / asm / task / import-policy 发出 GET 请求命名空间，并附加上一个响应的 id 属性。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ
```

响应显示请求的状态属性。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "kind": "tm:asm:tasks:import-policy:importpolicytaskstate",
    "lastUpdateMicros": 1370459676272126,
    "policyHistoryReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/history-revisions/hGKdiXU7US4S4qtgexijUQ"

    },
    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"
    },
    "status": "COMPLETED",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-policy/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
        "policyReference":
            { "link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w"
        }
    }
}
```

Application Security Manager 导入漏洞

如果使用 Application Security Manager™（ASM™）导入漏洞数据，则应了解 iControl®REST 如何实现 ASM。

iControl®REST 支持 Application Security Manager™（ASM™）功能，以从 fi 导入漏洞或从扫描仪下载漏洞。 您必须在 JSON 正文中包含 policyReference 属性。

属性	描述
policyReference	通过链接描述当前策略的路径。
文件	以 XML 格式指定文件内容。
文件名	指定要读取的文件的名称。
isBase64	指示文件是否包含 Base64 编码的数据。
scanId	指定扫描 ID。 如果您未指定文件属性，则对于 Cenzic Hailstorm 是必需的。
subscriptionId	指定订阅 ID。 如果您未指定文件属性，则对于 Cenzic Hailstorm 是必需的。
onlyGetDomainNames	指示任务是否解析输入文件，然后生成所有漏洞的计数而不导入漏洞。
importAllDomainNames	指示任务是否解析输入文件并导入所有漏洞。
domainNames	指定任务为其解析输入文件并导入所有漏洞的域名。

在 Application Security Manager 导入漏洞信息

iControl®REST 支持 Application Security Manager™（ASM™）功能，可从文件或扫描仪等来源导入漏洞数据。

1. 要导入漏洞，请对/ tm / asm / tasks / import-vulnerabilities 使用 POST 方法命名空间。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/import-vulnerabilities

{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
    "importAllDomainNames": false,
    "domainNames": [
        ""
    ],
    "subscriptionId": "4132",
    "scanId": "3883"
}
```

```
{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
    "isBase64": false,
    "importAllDomainNames": false,
    "status": "NEW",
    "lastUpdateMicros": 1395567859000000,
    "domainNames": [
        ""
    ],
    "subscriptionId": "4132",
    "scanId": "3883",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-vulnerabilities/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:import-vulnerabilities:import-vulnerabilities-taskstate",
    "id": "8PacFCQc0Umx45mheqdyew",
    "startTime": "2014-03-23T09:44:15Z",
    "result": {}
}
```

2. 要检索导入漏洞任务的状态，请使用 GET 方法。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/import-vulnerabilities/8PacFCQc0Umx45mheqdyew
```

对请求的响应包含以下数据：

```
{
    "isBase64": false,
    "importAllDomainNames": false,
    "status": "COMPLETED",
    "lastUpdateMicros": 1395567859000000,
    "domainNames": [
        ""
    ],
    "onlyGetDomainNames": false,
    "subscriptionId": "4132",
```

```
    "scanId": "3883",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/import-vulnerabilities/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:import-vulnerabilities:import-vulnerabilities-taskstate",
    "policyReference": {
        "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA"
    },
    "id": "8PacFCQc0Umx45mheqdyew",
    "startTime": "2014-03-23T09:44:15Z",
    "result": {
        "vulnerableHosts": [
            {
                "vulnerabilityCount": "4",
                "domainName": ""
            },
            {
                "vulnerabilityCount": "41",
                "domainName": "crackme.cenzic.com"
            }
        ]
    }
}
```

在 Application Security Manager 查询漏洞评估订阅

Application Security Manager™（ASM™）支持对第三方扫描仪的订阅。 您可以向 ASM 查询活动的漏洞评估订阅。

注意: *ASM 仅支持对 Cenzic Hailstorm 的订阅。*

---

1. 要确定活动的漏洞评估订阅，请将 POST 方法与 / tm / asm / tasks / get-vulnerability-assesssment-subscriptions 命名空间，并指定 JSON 正文中的 policyReference 属性。

```
POST
https://192.168.25.42/mgmt/tm/asm/tasks/get-vulnerability-assessment-subscriptions

{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" }
}
```

该响应显示指示新请求的请求状态属性和标识用于其他操作的请求的 id 属性。

```
{
    "kind":
"tm:asm:tasks:get-vulnerability-assessment-subscriptions:get-vulnerability-assessment-subscriptions-taskstate",

    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/get-vulnerability-assessment-subscriptions/pCCSkFyRGWeAf6Kwcpj38w",

    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA"
    },
```

```
    "status": "New",
    "id": "pCOSkFyRGWeAf6Kwcpj38w",
    "startTime": "2014-03-24T09:35:57Z",
    "lastUpdateMicros": 1395653765000000,
    "result": { }
}
```

2. 要获取此请求的输出，请使用 GET 方法和 / tm / asm / tasks / get-vulnerability-assessmentment-subscriptions 命名空间，并附加 URI 的 id 属性。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/get-vulnerability-assessment-subscriptions/pCOSkFyRGWeAf6Kwcpj38w
```

```
{
    "kind":
"tm:asm:tasks:get-vulnerability-assessment-subscriptions:get-vulnerability-assessment-subscriptions-taskstate",

    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/get-vulnerability-assessment-subscriptions/pCOSkFyRGWeAf6Kwcpj38w",

    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA"
    },
    "status": "COMPLETED",
    "id": "pCOSkFyRGWeAf6Kwcpj38w",
    "startTime": "2014-03-24T09:35:57Z",
    "lastUpdateMicros": 1395653765000000,
    "result": {
        "subscriptions": [
            {
                "scans": [
                    {
                        "scanId": "3870",
                        "completeDateTime": "2013-04-03T08:33:27Z",
                        "status": "Complete"
                    },
                    {
                        "scanId": "3883",
                        "completeDateTime": "2013-04-09T08:55:50Z",
                        "status": "Complete"
                    }
                ],
                "url":
"http://crackme.cenzic.com/Kelev/register/register.php",
                "productId": "YK Trial Scan",
                "subscriptionId": "4132"
            }
        ]
    }

}
```

在 Application Security Manager 启动漏洞评估

通过漏洞评估，可以访问第三方扫描程序，例如 Cenzic Hailstorm。 汇编/任务名称空间包括用于启动扫描的端点。

1. 要启动漏洞评估，请通过以下方式发出 POST 请求：
/ tm / asm / tasks / initiate-vulnerability-assessment 命名空间。 包括
JSON 主体中的 policyReference 和 subscriptionId 属性。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/initiate-vulnerability-assessment

{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
    "subscriptionId": "4132"
}
```

响应显示请求的状态和 id 属性。

```
{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
    "status": "NEW",
    "lastUpdateMicros": 1395567859000000,
    "subscriptionId": "4132",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/initiate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:initiate-vulnerability-assessment:initiate-vulnerability-assessment-taskstate",

    "id": "8PacFCQc0Umx45mheqdyew",
    "startTime": "2014-03-23T09:44:15Z",
    "result": {}
}
```

2. 要检索发起漏洞评估操作的状态，请使用 GET 方法和
/ tm / asm / tasks / initiate-vulnerability-assesssment 命名空间，并将 id 属性附加到 URI。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/initiate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew
```

该响应显示请求状态和 scanId 属性。

```
{
    "status": "COMPLETED",
    "lastUpdateMicros": 1395567859000000,
    "subscriptionId": "4132",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/initiate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:initiate-vulnerability-assessment:initiate-vulnerability-assessment-taskstate",

    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA"
    },
    "id": "8PacFCQc0Umx45mheqdyew",
    "startTime": "2014-03-23T09:44:15Z",
    "result": {
         "scanId": 4920
    }
}
```

在 Application Security Manager 终止漏洞评估

通过漏洞评估，可以访问第三方扫描程序，例如 Cenzic Hailstorm。 汇编/任务名称空间包括一个终止扫描的端点。

1. 要终止漏洞评估，请通过以下方式发出 POST 请求：
/ tm / asm / tasks / terminate-vulnerability-assessment 命名空间。 在 policyReference 属性中包含 JSON 正文。

```
POST
https://192.168.25.42/mgmt/tm/asm/tasks/terminate-vulnerability-assessment

{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
}
```

对请求的响应包括标识查询请求的 ID。

```
{
    "policyReference": { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA" },
    "status": "NEW",
    "lastUpdateMicros": 1395567859000000,
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/terminate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:terminate-vulnerability-assessment:terminate-vulnerability-assessment-taskstate",

    "id": "8PacFCQc0Umx45mheqdyew",
    "startTime": "2014-03-23T09:44:15Z",
    "result": {}
}
```

2. 要获取终止漏洞评估操作的状态，请使用 GET 方法和
/ tm / asm / tasks / terminate-vulnerability-assessment 命名空间，并将 id 属性附加到 URI。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/terminate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew
```

响应显示请求状态。

```
{
    "status": "COMPLETED",
    "lastUpdateMicros": 1395567859000000,
    "subscriptionId": "4132",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/terminate-vulnerability-assessment/8PacFCQc0Umx45mheqdyew",

    "kind":
"tm:asm:tasks:terminate-vulnerability-assessment:terminate-vulnerability-assessment-taskstate",

    "policyReference":
        { "link":
"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA"
    },
    "id": "8PacFCQc0Umx45mheqdyew",
```

```
    "startTime": "2014-03-23T09:44:15Z",
    "result": {
    }
}
```

Application Security Manager 漏洞解决

如果使用 Application Security Manager™（ASM™）来解决漏洞，则应了解 iControl®REST 如何实现 ASM。

Application Security Manager™（ASM™）支持用于解决漏洞的选项，例如暂存针对漏洞的建议更改。

属性	描述
getPreResolveMessages	表示该任务仅显示每个漏洞的建议更改，但不实施更改。
stageVulnerabilities	指示应上载对策略所做的更改。
vulnerabilities	将对漏洞的引用指定为引用的集合。

解决 Application Security Manager 的漏洞

解决漏洞时，Application Security Manager™（ASM™）会配置安全策略以保护 Web 应用程序免受漏洞侵害。 如果选择，则可以发布漏洞，以便有更多时间测试安全策略。 否则，ASM 将立即将更改应用于安全策略。

1. .要解决漏洞，请在 POST 方法中使用/ tm / asm / tasks / resolve-vulnerabilities 命名空间，并指定漏洞属性。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/resolve-vulnerabilities

{
"vulnerabilities": [
  {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

  {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}

 ]
}
```

响应包括请求状态和 id 属性。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "vulnerabilities": [
        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}
```

```
     ],
       "kind":
"tm:asm:tasks:resolve-vulnerabilities:resolvevulnerabilitiesstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. 要确定此操作的状态，请将 GET 方法与
/ tm / asm / tasks / resolve-vulnerabilities 命名空间，并将 id 属性附加到 URI。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ
```

响应显示结果属性。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "vulnerabilities": [
        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}

     ],
    "kind":
"tm:asm:tasks:resolve-vulnerabilities:resolvevulnerabilitiesstate",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ",

    "lastUpdateMicros": 1370459676272126,
    "status": "COMPLETED",
    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
                  "message": "The system does not automatically mitigate the
detection of an SQL injection vulnerability created as a result of a scanner
payload that includes distractive meta characters.\nIn order to mitigate
this vulnerability, manually add the disallowed meta characters to the
vulnerable parameter.\nNote: Characters such as '\"< when injected may change
 the SQL query."
              }
}
```

识别 Application Security Manager 的漏洞

iControl®REST 支持 Application Security Manager™（ASM™）任务，无需更改安全策略即可解决漏洞并获取标识漏洞的消息。

1. 要检索预解析消息，请使用 POST 方法和 / tm / asm / tasks / resolve-vulnerabilities 命名空间，并指定漏洞和 getPreResolveMessages 属性。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/resolve-vulnerabilities

{
"vulnerabilities": [
  {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

  {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}

 ],
 "getPreResolveMessages": true
}
```

响应显示请求状态和 id 属性。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "vulnerabilities": [
        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}

     ],
    "getPreResolveMessages": true
       "kind":
"tm:asm:tasks:resolve-vulnerabilities:resolvevulnerabilitiesstate",
    "lastUpdateMicros": 1370459676272126,
    "status": "NEW",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ",

    "startTime": "2013-06-05T15:14:36-04:00"
}
```

2. 要确定此操作的状态，请将 GET 方法与 / tm / asm / tasks / resolve-vulnerabilities 命名空间，并将 id 属性附加到 URI。

```
GET
https://192.168.25.42/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ
```

响应包括结果属性和文本消息数据。

```
{
    "id": "oqNah2PxtwwE4YyAHGekNQ",
    "vulnerabilities": [
        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/abcdef1234567890"},

        {"link":
"https://localhost/mgmt/tm/asm/policies/vagoQLF6uOoBKvS8h3C19w/vulnerabilities/qwertytrewqa1234"}

     ],
    "getPreResolveMessages": true
    "kind":
```

```
"tm:asm:tasks:resolve-vulnerabilities:resolvevulnerabilitiesstate",
    "selfLink":
"https://localhost/mgmt/tm/asm/tasks/resolve-vulnerabilities/oqNah2PxtwwE4YyAHGekNQ",

    "lastUpdateMicros": 1370459676272126,
    "status": "COMPLETED",
    "startTime": "2013-06-05T15:14:36-04:00",
    "endTime": "2013-06-05T15:14:56-04:00",
    "result": {
                "message": "The following attack signature sets will be
assigned to the security policy: Cross Site Scripting Signatures, SQL
Injection Signatures\nStaging will be disabled for all signatures of Signature
 Set: Cross Site Scripting Signatures, SQL Injection Signatures"
              }
}
```

在 Application Security Manager 导出数据保护

您可以对不属于为 ASM®启用的集中式管理基础结构（CMI）设备组的云科®系统使用相同的 cookie 加密种子。

要导出数据保护，请向 POST 请求 / mgmt / tm / asm / tasks / export-data-protection 端点。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/export-data-protection
```

```
{
  "filename": "example-export-keys"
}
```

```
{
  "endTime": "1970-01-01T00:00:00Z",
  "filename": "",
   "id": "",
  "lastUpdateMicros": 0,
  "result": {
    "message": "",
    "file": ""
  },
  "startTime": "1970-01-01T00:00:00Z",
  "status": "NEW",
  "statusEnums": [
    "NEW",
    "FAILURE",
    "COMPLETED",
    "STARTED"
  ]
}
```

在 Application Security Manager 导入数据保护

完成数据保护导出任务后，您可以在另一个数据中心的另一个云科®系统上导入数据保护密钥以共享流量。 通过共享密钥，您可以在数据中心之间共享相同应用程序的流量。

要导入数据保护，请向 POST 请求 / mgmt / tm / asm / tasks / import-data-protection 端点。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/import-data-protection
```

```
{
  "filename": "example-export-keys"
}
```

```
{
  "endTime": "1970-01-01T00:00:00Z",
  "file": "",
  "filename": "",
  "graceAcceptingIntervalInMinutes": 2880,
  "graceSigningIntervalInMinutes": 30,
  "id": "",
  "lastUpdateMicros": 0,
  "result": {
    "DataProtectionReference": {
      "link": "https://localhost/mgmt/tm/asm/data-protection?ver=13.0.0"
    },
    "message": ""
  },
  "startTime": "1970-01-01T00:00:00Z",
  "status": "NEW",
  "statusEnums": [
    "NEW",
    "FAILURE",
    "COMPLETED",
    "STARTED"
  ]
}
```

在 Application Security Manager 导入证书

Application Security Manager™（ASM®）支持导入 SSL 证书的任务。 导入证书后，该证书可用于 XML 配置文件的 Web 服务安全（WSS）保护。

要导入证书，请对/ mgmt / tm / asm / tasks / import-certificate 发出 POST 请求端点。

```
POST https://192.168.25.42/mgmt/tm/asm/tasks/import-certificate
```

```
{
  "certificateName": "",
  "certificateType": "client",
```

```
  "certificateTypeEnums": ["client", "server"],
  "file": "",
  "filename": "",
  "saveExpiredOrUntrustedCertificate": false,
  "certificateReference": {"link":
"https://localhost/mgmt/tm/asm/certificates/example?ver=13.1.0"},
  ... [Other standard task fields]
}
```

属性 saveExpiredOrUntrustedCertificate 是一个布尔值，允许您导入不受信任或已过期的证书。

Web 爬虫配置设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取配置设置，则可以使用 iControl®REST API 检索或修改这些设置。

iControl REST 将属性公开给 CONFI 会话事务异常设置。 如果您在“流量管理” UI（TMUI）中配置 Web 抓取设置，则此处描述的值符合您熟悉的设置。 字符串 webScrapingConfiguration 标识此资源对象的顶级成员。

属性	描述
sessionTransactionAnomalyBlock	指示系统是否以布尔值阻止会话事务异常。
sessionOpeningAnomalyAlarm	指示系统是否以布尔值发送有关会话打开异常的警报。
suspiciousClientsAlarm	指示系统是否以布尔值发送有关会话打开异常的警报。
sessionTransactionAnomalyAlarm	指示系统是否以布尔值发送有关会话事务异常的警报。
suspiciousClientsBlock	指示是否将可疑客户端上以布尔值的块阻止。
sessionOpeningAnomalyBlock	指示系统是否以布尔值形式阻止会话打开异常。
usePersistentStorage	指示系统是否使用持久存储的客户端标识数据（布尔值）。
botDetectionBlock	指示系统是否在僵尸程序检测时阻止，以布尔值表示。
useFingerprint	表示系统是否使用来收集浏览器属性（布尔值）。
botDetectionAlarm	指示系统是否以 Boolean 形式发送关于机器人检测的警报。

会话事务异常设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取配置设置，则可以使用 iControl®REST API 检索和修改这些设置。

iControl REST 将属性公开给 CONFI 会话事务异常设置。 如果您在“流量管理 UI”（TMUI）中配置 Web 抓取设置，则本主题中描述的值符合您熟悉的设置。 字符串 sessionTransactionsAnomaly 标识此资源对象的顶级成员。

属性	描述
maximumSessionTransactionsPerSecond	指定如果每个会话的事务数等于或大于此数目，则系统将流量视为攻击。 默认值为 400。
minimumSessionsTransactionsPerSecond	指定如果每个会话的事务数等于或大于此数目，并且达到按值增加的会话事务或达到值的会话事务，则系统将流量视为攻击。 如果每个会话的事务数小于该值，即使达到增加值的会话事务之一或达到会话价值的会话事务，系统也不会将流量视为攻击。 默认值为 200。
preventionDuration	指定在系统检测到并停止攻击后系统阻止会话异常攻击的时间长度（以秒为单位），除非系统较早地检测到攻击结束。 该系统通过阻止请求来防止攻击。 默认值为 1800。
sessionTransactionsPerSecondIncreaseRate	指定如果会话中的事务数比正常多此百分比，则系统将流量视为攻击。 正常是指过去一个小时内整个网站每个会话的平均交易次数。 默认值为 500。

Bot 检测设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取配置设置，则可以使用 iControl®REST API 检索和修改这些设置。

iControl®REST 公开了用于配置 Bot Detection 设置的属性。 如果您在“流量管理” UI（TMUI）中配置 Web 抓取设置，则此处描述的值符合您熟悉的设置。 字符串 botDetection 标识此资源对象的顶级成员。

属性	描述
rapidSurfingMaximumDistinctPages	指定在系统将客户端源视为漫游器之前，可以在指定的时间内刷新一页的最大次数。 默认值为 120。
rapidSurfingMaximumChangedPages	指定在系统将客户端源视为漫游器之前的指定时间内可以加载的不同页面的数量.
checkEventsSequenceEnforcement	指示系统是否执行事件序列强制实施。 配置此设置可以通过跟踪浏览器触发以检测不规则序列的事件的顺序来保护 Web 应用程序免受僵尸程序的攻击。 当检测到不规则序列时，为了防止误报，客户端不会立即标记为

Property	Description
rapidSurfingMaximumTimeDuration	指定刷新一个网页或刷新一次最少数量的页面所需的最长时间，以使系统怀疑机器人已请求该页面。 默认值为 30。
blockingPeriod	指定系统认为不安全的请求数，如果安全策略处于阻止模式，则阻止请求。 系统在宽限间隔内未检测到有效的客户端，并自动生成“检测到 Web 爬网”冲突。 此外，系统不再检查这些请求以进行 Web 抓取。 客户端发送此设置中指定的请求数后，系统将重新激活宽限间隔。 默认值为 500。
graceThreshold	指定系统在尝试检测客户端是否为人类时检查的最大请求数。 系统做出该确定后，将立即停止检查请求。 缺省值是 100。一旦客户端确定该客户端有效，系统将根据安全间隔设置指定允许且不检查接下来的几个请求。 如果系统在宽限间隔内未检测到有效的客户端，则系统将发出并继续发出“检测到 Web Scraping”违规，直到达到阻止时间段设置中指定的请求数。
safeIntervalThreshold	指定系统认为安全的请求数。 系统确定这些请求是由人工支持的客户端发送的，因此不再检查这些请求以进行 Web 抓取。 客户端发送的请求数达到设置中指定的值后，系统将重新激活宽限间隔。 默认值为 2000。

=会话打开异常设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取设置，则可以使用 iControl®REST API 检索和修改这些设置。

iControl®REST 将属性公开给 CONFI 会话打开异常设置。 如果您在“流量管理” UI（TMUI）中配置会话打开异常设置，则此处描述的属性符合您熟悉的设置。 字符串 sessionOpeningAnomaly 标识此资源对象的顶级成员。

属性	描述
minimumSessionsOpenedPerSecond	指定如果每秒打开的会话数等于或大于此数量，并且至少每秒打开的会话数增加一个或达到每秒打开的会话数，则系统将流量视为攻击 。 如果每秒打开的会话数小于此数量，则系统不会将此流量视为即使达到每秒打开的会话数增加一次或达到每秒打

Property	Description
checkSessionOpeningAnomaly	指示系统是否通过 IP 地址将会话打开异常检测为布尔值。
clientSideIntegrityDefense	指示系统是通过从检测到的 IP 地址向每个新的会话请求发送 JavaScript 质询，还是等待响应来确定客户端是合法浏览器还是非法脚本。 默认值为 false。
rateLimiting	指示系统是通过从检测到的 IP 地址向每个新的会话请求发送 JavaScript 质询，还是等待响应来确定客户端是合法浏览器还是非法脚本。 默认值为
maximumSessionsOpenedPerSecond	指定如果每秒打开的会话数等于或大于此数目，则系统将流量视为攻击。 默认值为 50。
dropIpAddressesWithBadReputation	指示系统是否根据系统的 IP 地址信誉数据库丢弃信誉差的 IP 地址发出的请求。 像往常一样，没有不良信誉的攻击 IP 地址会受到速率限制。 默认值为禁用。
sessionsOpenedPerSecondIncreaseRate	指定如果每秒打开的会话数增加此数量，则系统将流量视为攻击。 默认值为 500。
preventionDuration	指定在系统检测到并停止攻击后系统阻止会话打开异常攻击的时间长度（以秒为单位），除非系统较早地检测到攻击结束。 默认值为 1800。

会话打开阈值设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取设置，则可以使用 iControl®REST API 检索和修改这些设置。

iControl REST 公开了用于配置会话打开阈值设置的属性。 如果您在 Traffi Management UI（TMUI）中配置会话打开阈值设置，则此处描述的值符合您熟悉的设置。 字符串 sessionOpeningThresholds 标识资源对象的顶级成员。

属性	描述
checkFingerprintResets	指示系统是否使用指纹来检测 Cookie 删除事件。 指纹假设每个浏览器都有唯一的指纹，因此系统会收集浏览器属性以识别浏览器和漫游器。 默认值为 false。
openingPersistentStorageResetsDuration	指定系统在确定请求为网页抓取攻击并阻止可疑的非法请求之前，系统必须检测指定数量的 cookie 删除事件的时间长度（以秒为单位）.

属性	描述
openingPersistentStorageInconsistencyEventsMaximum	指定系统必须检测以确定网络抓取攻击的完整性故障事件的数量。 预设值为 3。
preventionDuration	指定在系统检测到并停止攻击后系统阻止会话打开阈值攻击的时间长度（以秒为单位），除非系统较早地检测到攻击结束。 该系统通过拒绝来自攻击客户端的请求来防止攻击。 系统基于存储在攻击浏览器的持久性存储中的唯一标识号来识别攻击客户端。 默认值为 1800。
checkStorageInconsistency	指示系统是否阻止将其标识为完整性故障事件的请求。 默认值为 false。
checkStorageResets	指示系统是否使用持久性设备标识来检测 cookie 删除事件，默认值为 false。
openingPersistentStorageResetsMaximum	指定在系统确定攻击为 Web 抓取攻击并阻止可疑的非法请求之前，系统必须在指定的时间段内检测到的 Cookie 删除事件的数量。
fingerprintResetsTimeWindow	指定系统在确定请求为 Web 抓取攻击并阻止可疑的非法请求之前，系统必须检测指定数量的 cookie 删除事件的时间长度（以秒为单位）。 默认值为 600。
openingPersistentStorageInconsistencyEventsDuration	指定系统在确定攻击为 Web 抓取攻击之前必须检测完整性故障事件的时间长度（以秒为单位）。 默认值为 600。
fingerprintResetsThreshold	指定在系统确定请求为 Web 抓取攻击并阻止可疑的非法请求之前，系统必须在指定的时间段内检测到的 Cookie 删除事件的数量。

可疑客户端设置

如果使用 Application Security Manager™（ASM™）管理 Web 抓取设置，则可以使用 iControl®REST API 检索和修改这些设置。

iControl REST 将属性公开给可疑的客户端设置。 如果您在“流量管理” UI（TMUI）中配置可疑客户端设置，则此处描述的值符合您熟悉的设置。 字符串 suspiciousClients 标识资源对象的顶级成员。

属性	描述
detectBrowsersWithScrapingExtensions	指示系统是否调查浏览器中的 Web 抓取插件，以确定是否应将客户端视为可疑客户端。 默认值为 false。

属性	描述
scrapingExtensions	指定一组视为非法的 Web 抓取扩展。 如果系统确定客户端可疑，它将记录并阻止来自该客户端的请求。
preventionDuration	指定系统确定客户端可疑后，系统阻止客户端发出请求的时间长度（以秒为单位）。 默认值为 300。

iControl REST Web 爬虫设置

iControl®REST 支持 Application Security Manager™（ASM™）Web 抓取设置的可编程性。 iControl REST 接口提供了一个支持查询和修改请求的端点。 作为单一资源，Web 抓取资源支持 GET 请求以检索当前的 Web 抓取设置，以及 PATCH 请求以修改资源属性。 PATCH 方法允许将资源部分表示为请求实体，这意味着您只需要指定要更改的属性，而不是整个资源。

检索网页爬虫设置

iControl®REST 通过允许检索云科®系统的 Web 抓取设置来支持 Application Security Manager™（ASM™）功能。 您可以使用 iControl REST API 自动从多个云科系统中检索设置。

要检索网络抓取设置，请向 GET 请求/ tm / asm / policies / <MDHASH> /网络抓取端点。

```
GET https://192.168.25.42/mgmt/tm/asm/policies/<MDHASH>/web-scraping
```

iControl REST 检索所有流量模式的 Web 抓取设置。

```
{
   "suspiciousClients":{ "detectBrowsersWithScrapin
      gExtensions":false, "preventionDuration":
      300, "scrapingExtensions":[]
   },
   "sessionOpeningThresholds":{ "checkFingerprintRes
      ets":true, "checkStorageInconsistency":true,
      "checkStorageResets":true,
      "openingPersistentStorageResetsDuration": 707,
      "openingPersistentStorageResetsMaximum": 77,
      "fingerprintResetsTimeWindow": 607,
      "openingPersistentStorageInconsistencyEventsMaximum": 7,
      "persistentStorageMaxPreventionDuration": 1807,
      "openingPersistentStorageInconsistencyEventsDuration": 677,
      "fingerprintResetsThreshold": 17
   },
   "sessionOpeningAnomaly":{ "minimumSessionsOpe
      nedPerSeconds": 22,
      "checkSessionOpeningAnomaly":true,
      "PreventionDuration": 1802,
      "clientSideIntegrityDefense":true,
      "rateLimiting":true,
      "maximumSessionsOpenedPerSeconds": 52,
      "dropIpAddressesWithBadIpReputation":true,
```

```
      "sessionsOpenedPerSecondsIncressRate": 502
   },
   "botDetection":{ "rapidSurfingMaximumDistin
      ctPages": 301,
      "rapidSurfingMaximumChangedPages": 1201,
      "checkEventSequenceEnforcement":true,
      "rapidSurfingMaximumTimeDuration": 311,
      "unsafeIntervalTreshold": 10011,
      "graceTreshold": 1001,
      "safeIntervalTreshold": 20001
   },
   "sessionTransactionsAnomaly":{ "maximumSession
      TransactionsPerSecond": 403,
      "minimumSessionTransactionsPerSecond": 203,
      "maximumTransactionPreventionDuration": 1803,
      "sessionTransactionsPerSecondIncreaseRate": 503
   },
   "webScrapingConfiguration":{ "alarmOnBotDete
      ction":true,
      "blockOnSessionTransactionAnomaly":false,
      "alarmOnSessionOpeningAnomaly":true,
      "alarmOnSuspiciousClients":true,
      "alarmOnSessionTransactionAnomaly":true,
      "blockOnBotDetection":false,
      "blockOnSessionOpeningAnomaly":false,
      "usePersistentStorage":true,
      "useFingerprint":true,
      "blockOnSuspiciousClients":true,
      "persistentDataValidityPeriod": 126
   },

"selfLink":"https://localhost/mgmt/tm/asm/policies/xpqbO1mYOtgfv13j1khKeA/web-scraping?ver=12.0.0",

   "kind":"tm:asm:policies:web-scraping-settings:web-scraping-settingsstate"
}
```

修改 web 爬虫设置

iControl®REST 通过启用对云科®系统的 Web 抓取设置的修改来支持 Application Security Manager™（ASM™）功能。 您可以使用 iControl REST API 自动从多个云科系统中修改设置。

要修改会话事务异常的预防持续时间属性，请向/ tm / asm / policies / <MDHASH> / web-scraping 端点发出 PATCH 请求。 在 JSON 正文中，指定要修改的流量模式的顶级成员，并指定所需的属性更改。 要更改会话事务异常的多个设置，请在资源对象中指定多个属性，以逗号分隔。

```
PATCH https://192.168.25.42/mgmt/tm/asm/policies/<MDHASH>/web-scraping
{
   "sessionTransactionsAnomaly": { "preventionDuration": 2400 }
}
```

JSON 正文必须至少包含一个用于识别流量模式的顶级成员，例如 sessionTransactionsAnomaly。

学习建议模型

如果您在 Application Security Manager™（ASM™）中使用 Policy Builder 功能，则此表中的属性将像在 JSON 正文中一样显示，以响应 GET 请求。

属性	描述
id	指定引用的唯一标识符。
creationDatetime	将建议的创建时间指定为日期时间值。
lastOccurrenceDatetime	指定上次匹配建议的请求发生的时间。
status	指定建议的学习状态。 可能的值为：待处理，被忽略，接受或 接受并分阶段。 接受和登台状态为目标实体启用登台标志（如果适用），并实现在 entityChanges 字段中指定的更改。
alwaysManual	表示将以布尔值形式手动学习建议。 如果为 true，则永远不会自动学习建议。
comment	指定关于建议的用户注释。
isRead	指示用户已阅读建议（布尔值）。
score	指定基于 R2 或 R3 测量值的索引，该索引反映建议的强度。
triggerType	指定建议的原因。 可能的值为：违反缓解或策略优化。
violationReference	如果触发类型是违规，请指定触发建议的违规类型。 此属性不是必需的。
entityChanges	如果您接受建议，则指定要应用于实体或 entityReference 的更改。
entityKind	指定建议元素的类型。 此属性不是必需的。
entityName	指定项目实例的名称。 此属性不是必需的。
action	指定项目的建议操作。 可能的值为：delete，add-or-update，update-append 或 update-remove。 此值不是必需的。
entity, entityReference	如果策略中存在实体，则指定对策略的引用； 否则，指定要创建的实体的详细信息。
parentEntityReference	指定对与通配符值匹配的父策略实体的引用。
occurrenceCount	指定触发建议的请求数。

Property	Description
trustedIpCount	在触发建议的受信任客户端列表上，指定不同的客户端 IP 地址的数量。
untrustedIpCount	指定触发建议的唯一客户端 IP 地址的数量（不在受信任客户端列表中）。
trustedSessionCount	指定触发建议的来自受信任 IP 地址的不同客户端会话的数量。
untrustedSessionCount	指定触发建议的不同客户端会话（不是来自受信任 IP 地址）的数量。
sampleRequests	指定来自各种 IP 地址和会话的代表请求的代表请求的集合。
description	指定要实施的更改的描述。
refinement, refinementReference	指定类型为何时的建议原因细化政策而不是违规缓解。
signatureReference	指定对攻击签名的引用，可以作为对另一个对象的替代，也可以对签名本身进行更改，例如禁用签名。
metachar	指定对元字符的引用，可以作为对另一个对象的替代，也可以修改元字符本身，例如允许字符本身。
averageViolationRating	指定建议的平均违规等级（如果适用）。
violationRatingCounts	指定每个请求的违规等级数。

关于在 iControl REST 中使用 Policy Builder

Application Security Manager™（ASM™）安全策略通过称为统一学习和策略构建的框架进行修改。 Unifi 学习和策略构建支持对安全策略的手动和自动更新。作为管理员，您可以使用 iControl®REST API 检索策略构建器建议并修改策略建议。您可以执行的操作包括按分数或类型对建议进行排序，查看有关建议的更多详细信息或查看有关建议的详细信息。 iControl REST 在/suggestions 端点上支持三种方法：GET，DELETE 和 PATCH。除了获取查看建议集合的 GET 请求之外，您可能还有其他原因

修改单个建议以更改建议的状态，添加评论或将建议标记为已读。您可以使用 HTTP PATCH 方法来修改状态，注释或 isRead 属性。顺便说一句，如果您修改除上述属性外的其他属性，iControl REST 将忽略请求中的那些属性。有关策略构建器对象的描述，请参考学习建议对象主题。

有关策略生成器的更多信息，请参阅云科®Application Security Manager（ASM）12.0 文档。

检索策略生成器建议

您可以通过发出 GET 请求来检索有关 Application Security Manager™（ASM™）策略的建议。 默认情况下，ASM 检索前 500 个实体。

要检索有关 ASM 策略的建议，请向/ suggestions 端点发出 GET 请求以获取特定的 ASM 策略。

```
GET https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/suggestions
```

示例中的字符串 abcd1234 表示策略的假设 MD5HASH 值。 实际的 MD5 哈希值类似于以下字符串：d57fb462a2364e494ed824d523acbfcd。

响应中包含最多 1000 个实体的政策建议。

```
{
   "selfLink":"https://localhost/mgmt/tm/asm/policies/abcd1234/suggestions",

   "kind":"tm:asm:policies:suggestions:suggestioncollectionstate",
   "items":[
      {
         "id":"123456",

"selfLink":"https://localhost/mgmt/tm/asm/policies/abcd1234/suggestions/123456"

         "kind":"tm:asm:policies:suggestions:suggestionstate",
         "creationDatetime":"2013-11-21T22:01:21Z",
         "lastOccurrenceDatetime":"2013-12-10T21:01:21Z",
         "status":"active",
         "alwaysManual":false,
         "comment":"",
         "isRead":false,
         "score":76,
         "occurrenceCount":378,
         "trustedClientIpCount":0,
         "trustedSessionCount":0,
         "untrustedClientIpCount":4,
         "untrustedSessionCount":3,
         "triggerType":"violation",
         "violationReference":{

"link":"https://localhost/mgmt/tm/asm/violations/ufg0smEkZrpmkoDHfSPGdQ"
         },
         "parentEntityReference":{ "link":"htt
            ps://localhost/......."
         }
                  "entityReference":{ "link":
            "https://localhost/......."
         },

            "entity":{ "kind":"tm:asm:policies:urls:param
            eterstate", "name":"foo",
            "level":"url",
            "url":{
               "name":"/foo.php",
               "protocol":"http",

            }
         },
         "entityChanges":{ "signat
            ureOverrides":[
               {
                  "signatureReference":{

"link":"https://localhost/mgmt/tm/asm/signatures/N64gk_aRPRtaPA4Mt50_LQ"
                  },
                  "enabled":false,

               },

            },
```

```
            "requestReferences":[
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123000"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123001"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123002"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123003"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123004"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123005"
               },
               {
"link":"https://localhost/mgmt/tm/asm/events/requests/123006"
               },

            ],

         }
      ]
   }
```

修改策略构建器建议

您可以通过发出 PATCH 请求来修改对 Application Security Manager™（ASM™）策略的建议。 ASM 限制了可以更改的策略构建器属性。

要修改有关 ASM 策略的建议，请向/ suggestions / <id>发出 PATCH 请求 特定 ASM 策略的终结点。 本示例将状态更改为“忽略”。

```
PATCH https://192.168.25.42/mgmt/tm/asm/policies/<MD5HASH>/suggestions/465768
```

```
{
   "status": "ignored"
}
```

MD5 哈希是一种单向加密哈希函数。 实际的 MD5 哈希值类似于以下字符串： d57fb462a2364e494ed824d523acbfcd。

关于 Device ID

设备标识符（设备 ID）由标识客户端应用程序的不透明字符串组成。 字符串的声明目的仅是为了识别虚拟服务器的客户端应用程序。 利用设备 ID 的 Application Security Manager™（ASM™）功能包括暴力登录，会话感知和会话劫持预防。

使用指纹识别设备

如果您使用 Application Security Manager™（ASM™）管理设备 ID 设置，则可以使用 iControl®REST API 检索和修改这些设置。

Application Security Manager（ASM）支持使用指纹识别设备并公开表中列出的配置属性。 使用/ mgmt / tm / security / dos / profile /应用程序端点。

属性	描述
deviceIdClientSideDefense	指示是否根据带有 CS 质询的设备 ID 进行缓解。
deviceIdCaptchaChallenge	指示是否根据带有 CAPTCHA 质询的设备 ID 进行缓解。
deviceIdRateLimiting	指示是否基于具有阻止请求的设备 ID 进行缓解。
deviceIdRequestBlockingMode	在 deviceId RateLimiting 时指定缓解措施启用为速率限制或全部阻止。
deviceIdMaximumTps	指定每个设备 ID 的最大 TPS，以引起怀疑。
deviceIdMinimumTps	指定每个设备 ID 的最小 TPS。
deviceIdTpsIncreaseRate	指定每个设备 ID 引起怀疑的增加百分比。

Application Security Manager（ASM）支持使用指纹识别设备并公开表中列出的启用属性。 使用 / mgmt / tm / asm / policies / <MD5HASH> / brute-force-attack-preventions 端点。

属性	描述
alarm	指示是否发送警报，是对还是错。 默认为 true。
block	指示是否阻止请求，为 true 或 false。 默认为 true。
bruteForceProtectionForAllLoginPages	指示对所有登录页面采取措施，为是或否。 该属性仅可用于默认的蛮力保护项目。 默认为 false。
useDeviceId	指示是否将基于设备 ID 的尝试计数为真或假。 默认为 false。
loginAttemptsFromTheSameClient	指定阻止前的登录尝试次数。 默认为 5。

属性	描述
preventionDuration	从一个枚举中指定一个值，该值可设置防止暴力攻击防护的持续时间（以秒为单位），或设置为无限制的字符串。
measurementPeriod	指定衡量登录尝试的时间段，以秒为单位。 默认为 1。
id	将标识符指定为字符串。
reEnableLoginAfter	指定重新启用登录之前要等待的时间间隔（以秒为单位）。 默认值为 600。
urlReference	指定 URL 引用。
detectionCriteria	将检测条件指定为 JSON 对象，由 failLoginAttemptsIncreasePercent，failedLoginAttemptsRateReached 和 minimumFailedLoginAttempts 组成。 所有值都是整数，分别默认为 500、100、20。
preventionPolicy	将预防策略指定为 JSON 对象，由 sourceIpBasedClientSideIntegrityDefense，sourceIpBasedRateLimiting，urlBasedClientSideIntegrityDefense 和 urlBasedRateLimiting 组成。 所有值都是布尔值，可以是 true 或 false。 分别默认为 false，true，false 和 true。
suspiciousCriteria	将可疑条件指定为 JSON 对象，由 failedLoginAtttemptsIncreasePercent 和 failedLoginAttemptsRateReached 组成。 所有值都是整数。 默认分别为 500 和 20。

在 URL 强制方法

如果您使用 Application Security Manager™（ASM™）管理每个 URL 允许或不允许的方法列表，则可以使用 iControl®REST API 修改那些设置。

Application Security Manager（ASM）支持一种机制，该机制定义每个 URL 的允许或不允许的方法列表，并公开表中列出的配置属性。 使用 / mgmt / tm / asm / policies / <MD5HASH> / URL 端点。

属性	描述
methodsOverrideOnUrlCheck	指示是否启用覆盖，是 true 还是假。 默认为 false。
methodOverrides	以 JSON 格式指定键值对数组。 对于 allowed 属性，默认为 true。 您必须为 method 属性指定一个值。

使用指纹的会话感知机制

如果使用 Application Security Manager™（ASM™）管理会话感知，则可以使用 iControl®REST API 检索和修改这些设置。

Application Security Manager（ASM）支持使用指纹的会话感知机制，并公开表中列出的配置属性。 使用
/ mgmt / tm / asm / policies / <MD5HASH> /会话跟踪端点。

属性	描述
checkDeviceIdThreshold	指示是否管理设备 ID 范围，为 true 或错误。
deviceIdThreshold	指定启用设备 ID 检查时每个设备 ID 范围的违例数。

会话劫持预防

如果您使用 Application Security Manager™（ASM™）来管理策略设置，则可以使用 iControl®REST API 来检索和修改这些设置。

Application Security Manager™（ASM™）通过为每个客户端设备分配唯一的标识符来减轻会话劫持。 通过维护会话的设备 ID 信息，ASM 可以确定会话是否已被劫持。 使用/ mgmt / tm / asm / policies / <MD5HASH> /会话跟踪端点。

属性	描述
sessionTrackingConfi racking SessionHijackingByDeviceId	指示是否通过策略在 ASM 中启用了会话劫持预防。

关于 WebSockets

WebSocket 协议在 HTTP 连接的上下文中定义了客户端和服务器之间的双向全双工通信通道。 WebSocket 连接通过发送带有值 websocket 的升级头从现有的 HTTP 连接启动。 作为握手的一部分
在客户端和服务器之间，服务器发送 101 交换协议作为响应。 Application Security Manager™（ASM™）支持 WebSocket 安全策略，作为具有 confi 属性的独特协议。 WebSocket 的规范可以在 RFC 6455-WebSocket 协议中找到。

WebSocket 协议

Application Security Manager™（ASM™）支持 WebSocket 协议的安全策略设置。

Application Security Manager™（ASM™）支持 WebSocket 协议，并公开表中列出的属性。 使用/ mgmt / tm / asm / policies / <MD5HASH> / websocket-urls 端点。

属性	描述
id	指定策略的标识符。
name	指定策略的名称。
nameBase64Encoded	指示名称是否以 Base64 格式编码。 默认为 false。
type	从 WebSocket 类型的枚举中指定一个值，或者显式或通配符。
description	为 WebSocket URL 指定可选描述.
lastUpdateMicros	指定上次更新时间（以微秒为单位）。
learnNewEntities	从枚举中指定一个值，总是或永不。

Attribute	Description
protocol	指定来自 WebSocket 协议枚举的值 ws 或 wss。
isAllowed	表示允许的 URL 为 true； 对于不允许的 URL，为 false。
metaCharsOnWebsocketUrlCheck	指示是否检查 URL 中的元字符，是对还是错。 默认为 false。 仅适用于通配符 URL 类型。
metacharOverrides	指定优先于全局 URL 元字符设置的 isAllowed 值数组和相应的十六进制值。 默认为 true 和 0x0。
performStaging	指示是否启用登台，为 true 或 false.
wildcardOrder	将通配符的匹配顺序指定为整数。 默认为 0（零）。
wildcardIncludesSlash	指示是否为通配符值匹配 URL 的多个段，如 true 或 false。
html5CrossOriginRequestsEnforcement	将 CORS 设置指定为 JSON 对象。 该对象包含 crossDomainAllowedOrigin，一个 JSON 对象和 ImplementationMode 属性。 ImplementationMode 枚举值包括 删除所有标头，禁用并强制执行。
extension	指定要进行握手的动作。 枚举值包括忽略，阻止或删除。 默认删除.
checkPayload	指示是否检查消息有效负载，为 true 或错误。
allowTextMessage	指示在消息有效负载中是否允许使用自由格式的文本。 仅在 checkPayload 为 true 时设置。 默认为 true。
allowJsonMessage	指示消息有效负载中是否允许使用 JSON。 仅在 checkPayload 为 true 时设置。 默认为 false。
allowBinaryMessage	指示在消息有效负载中是否允许二进制内容。 仅在 checkPayload 为 true 时设置。 默认为 false。
plainTextProfile	指定指向 WebSocket 消息的纯文本配置文件的链接。 仅在 allowTextMessage 为 true 时设置。
jsonProfile	指定指向 WebSocket 消息的 JSON 配置文件的链接。 仅在 allowJsonMessage 为 true 时设置。
binaryMessageMaxSize	指定最大二进制消息大小，为整数。 默认值为 10000。仅在 allow Binary Message 为 true 时设置。
messageFrameMaxSize	指定 WebSocket 框架的最大大小，以字节为单位。
messageFrameMaxCount	指定每帧的最大消息片段数，为整数。 默认为 100。

属性	描述
checkMessageFrameMaxSize	指示是否检查最大指定值，是 true 还是 false。 如果为 false，则允许任何消息大小。
checkMessageFrameMaxCount	指示是否检查最大指定值，是 true 还是 false。 如果为 false，则允许任何消息大小。
checkBinaryMessageMaxSize	指示是否检查最大指定值，是 true 还是 false。 如果为 false，则允许任何消息大小。

下表显示了 crossDomainAllowedOrigin 对象的属性。

属性	描述
includeSubDomains	指示是否包含子域，如 true 或 true 假。 默认为 false。
originName	将原点指定为字符串。
originPort	指定源端口号，为整数。 默认为 all 以指定所有端口。
originProtocol	指定枚举，http，http / https 或 https。 默认为 http / https。

下表中显示了 jsonProfile 对象的属性。

属性	描述
description	以字符串形式指定概要文件的描述。
metacharElementCheck	指示是否检查元字符，为 true 或错误。
attackSignatureCheck	指示是否检查攻击特征，如是或真假。
isReferenced	指示是否引用配置文件，是 true 还是假。
defenseAttributes	将防御属性指定为 JSON 对象。
sensitiveData	指定敏感数据，形式为 parameterName 字符串。
lastUpdateMicros	指定上次更新时间（以微秒为单位）。
metacharOverrides	指定 metachar 替代（作为 isAllowed 的数组，为 true 或 false），以及 metachar（作为十六进制值）。
name	指定配置文件的名称，以字符串形式。
signatureOverrides	指定签名替代，作为启用数组，如是或否，以及 signatureReference，一个对象。
id	指定配置文件的标识符，以字符串形式。

下表中显示了 jsonProfile 中的 defenceAttributes 对象的属性。

属性	描述
maximumTotalLengthOfJSONData	以整数形式指定 JSON 数据的长度。

属性	描述
maximumValueLength	以整数形式指定值的长度。
maximimStructureDepth	以整数形式指定结构的深度。
maximumArrayLength	以整数形式指定数组的长度。
tolerateJSONParsingWarnings	指示是否忽略 JSON 解析器警告，例如对或错。

plainTextProfile 对象的属性显示在下表中。

属性	描述
description	以字符串形式指定概要文件的描述。
metacharElementCheck	指示是否检查元字符，为 true 或错误。
attackSignatureCheck	指示是否检查攻击特征，如是或真假。
isReferenced	指示是否引用配置文件，是 true 还是假。
defenseAttributes	将防御属性指定为 JSON 对象。
lastUpdateMicros	指定上次更新时间（以微秒为单位）。
metacharOverrides	指定 metachar 替代（作为 isAllowed 的数组，为 true 或 false），以及 metachar（作为十六进制值）。
name	指定配置文件的名称，以字符串形式
signatureOverrides	指定签名替代，作为启用数组，如是或否，以及 signatureReference，一个对象。
id	指定配置文件的标识符，以字符串形式。

下表显示了 plainTextProfile 中的 defenceAttributes 对象的属性。

属性	描述
maximumTotalLength	以整数形式指定数据长度。
maximumLineLength	指定行的长度，为整数。
performPercentDecoding	指示是否进行百分比解码，真或假。

关于 AX/JSON 登录

除了 HTTP 身份验证和 HTML 表单身份验证之外，现代 Web 应用程序框架还使用 AJAX 身份验证。 典型的 AJAX 身份验证请求由登录表单的 POST 请求和 JSON 响应组成。 Application Security Manager™（ASM™）支持 AJAX 登录页面。

AJAX/JSON 认证

如果您使用 Application Security Manager™（ASM™）管理 AJAX / JSON 身份验证设置，则可以使用 iControl®REST API 检索和修改这些设置。

Application Security Manager™（ASM™）公开表中列出的属性。 使用 / mgmt / tm / asm / policies / <MD5HASH> / login-pages URI 作为特定登录页面资源的路径。

属性	描述
urlReference	指定 URL 路径。
authenticationType	将请求的身份验证类型指定为枚举。 允许的值包括：none，http-basic，ntlm，form，ajax-or-json-request 或 http-digest。 对于 AJAX / JSON，将 ajax-or-json-request 指定为认证类型。
usernameParameterName	指定与用户登录名相对应的 JSON 元素的名称。
usernameParameterNameBase64Encoded	指示是否使用 usernameParameterName 属性以 Base64 编码指定。
passwordParameterName	指定与用户密码相对应的 JSON 元素的名称。
passwordParameterNameBase64Encoded	指示是否使用 passwordParameterName 属性以 Base64 编码指定。
isReferenced	指示登录页面是否在其他地方被引用为 true 或 false。
id	指定登录页面的标识符。
accessValidation	指定将 AJAX / JSON 身份验证用作 JSON 对象的条件。 条件是名称/值对，其中名称为：cookieContains，headerContains，parameterContains，responseContains，responseHttpStatus 或 responseOmits。

下表列出了 accessValidation 对象的属性。

属性	描述
cookieContains	指定 cookie 中包含的字符串。
headerContains	指定包含为标题的字符串。
parameterContains	指定包含为查询参数的字符串。
responseContains	指定响应中包含的字符串。
responseOmits	指定响应中不包含的字符串。
responseHttpStatus	以字符串形式指定响应状态。
parameterContainsBase64Encoded	指示查询参数是否为 Base64 编码。
responseContainsBase64Encoded	指示响应是否包含 Base64 编码值。

属性	描述
responseOmitsBase64Encoded	指示响应是否不包含 Base64 编码值。

Application Security Manager（ASM）还公开了下表中列出的注销页面的属性。 使用/ mgmt / tm / asm / policies / <MD5HASH> / login-enforcement URI 作为指定注销页面资源的路径。

属性	描述
expirationTimePeriod	指示启用还是禁用到期设置。 默认情况下，如果没有收到请求，会话将在 600 秒后过期。
authenticatedUrls	将经过身份验证的 URL 指定为数组。
logoutUrls	指定 requestContains 作为字符串，urlReference 作为 JSON 对象，requestOmits 作为字符串。

访问策略管理器

关于访问策略管理器

访问策略管理器®（APM®）为云科®系统提供安全的标识和访问管理。iControl® REST 公开了 APM 的终端，以实现有规划的对 apm 资源进行访问和更优的自动化。

APM 遵循本指南前面描述的其他原则：

- URI 结构支持对集合和资源的一致访问
- 资源中的链接，包括自链接和支持发现
- JSON 编码简化了资源的表示
- HTTP 传输提供与资源交互的方法，以及安全性、身份验证、缓存和内容协商。

概述：URI 的格式与结构

rest 体系结构的原理描述了通过统一资源标识符 Uniform Resource Identifier(URI）来标识资源。一个 URI 显示一个 web 资源的名称；在本例中，URI 还表示 tmsh 模块和组件的树结构。您可以使用 web 服务请求指定 URI 来创建、读取、更新或删除云科®系统配置中的某些组件或模块。在 REST 体系结构的上下文中，系统配置是资源表示的同义词，web 服务请求使用 iControl® REST API 来表示。.

提示：对于 iControl REST 的请求默认使用管理帐户 admin。一旦熟悉了 API，就可以为具有各种权限的 iControl REST 用户创建用户账号。

对于此处显示的 URI 字段，URI 的管理 ip 组件是完全限定域名（FQDN）或云科设备的一个 ip 地址。

```
https://<management-ip>/mgmt/tm/...
```

在 iControl REST 中，URI 结构的所有请求中将包含/mgmt/tm/的字符串添加到以标识流量管理的命名空间。附加到该字符串上的任何标识符都指定集合。

```
https://<management-ip>/mgmt/tm/...
```

代码段中的省略号表示是组织集合的位置，该集合是指向 IControl REST 中其他资源链接的集合。组织集合是 tmsh 模块中的功能等价物。换句话说，此组织收藏的 apm 是 icontrol rest 中的一个 apm 模块。在 icontrol rest 中，可以使用以下 URI 访问 apm 集合中的所有资源：

```
https://192.168.25.42/mgmt/tm/apm
```

下面的示例中的 URI 扩展了这种方法，它指定了报表集合中的所有资源。可以将集合视为 tmsh 子模块的等价物。iControl REST 集合包含一些集合或资源。

```
https://192.168.25.42/mgmt/tm/apm/report
```

以下示例中的 uri 指定一个资源，它是一组实体。在 iControl REST 中，实体是一个可以配置的属性，例如“destAddrMax”：2048。资源也可以包含子集合。用 tmsh 的话说，资源相当于一个组件。

```
https://192.168.25.42/mgmt/tm/apm/report/default-report
```

重要提示：iControl REST 只支持通过 HTTPS 进行安全访问，因此每次 REST 调用都必须包含凭据。使用与云科设备管理器接口相同的凭据。

关于资源格式

JavaScript Object Notation（JSON）定义了 iControl® REST 中数据交换的格式。JSON 标准定义了一种人类可读的格式，部分基于 javascript 编程语言。与 SOAP web 服务通用的可扩展标记语言（XML）类似，JSON 描述了 REST web 服务请求中客户端和服务器之间交换的数据结构。iControl REST 处理格式化为 JavaScript Object Notation （JSON）格式的请求体，并在响应中生成 JSON 体。对 DELETE 请求的响应通常不包括 JSON 主体。

JSON 由两个结构组成：名称/值对（键/值对）作为对象组织和作为数组的有序价值列表。对象包含在大括号{}中，数组包含在方括号‘[]'中。JSON 对象可以包含对象、字符串、数字、数组、布尔值（true 或 false）或 null。有关 JSON 的更多信息，请参阅 RFC 7159 JavaScript Object Notation (JSON) 数据交换格式。

关于创建资源

使用 HTTP POST 方法创建新资源。iControl® REST 支持在访问策略管理器（APM）中创建资源的 POST 操作。必须在 POST 请求中包含 JSON 主体，即使 JSON 主体为。.

关于检索资源

使用 HTTP GET 方法检索资源。 iControl® REST 支持在访问策略管理器(APM®)中检索资源或资源集合的 GET 操作。另外，iControl REST 支持 Open Data Protocol (OData) $filter 查询参数来优化结果集。

关于更新资源

使用 HTTP 补丁或 PUT 方法更新资源。 iControl® REST 支持 HTTP 修补程序操作，以更新访问策略管理器（APM®）中的资源。使用修补程序更新特定的

属性并保持其他属性不变。iControl REST 还支持 HTTP PUT 操作来更新资源，并警告所有未指定的属性都被分配了默认值。

关于删除资源

使用 HTTP DELETE 方法删除资源。 iControl® REST 支持访问策略管理器（APM®）中的删除操作。iControl REST 返回删除请求的 HTTP 响应代码，但不包括 JSON 主体。

HTTP 响应代码

这些表列出了 iControl® REST 为每个请求生成的常见 HTTP 响应代码。

响应代码	返回	说明
200 OK	所有 HTTP 方法	指示请求已成功完成。
201 创建	POST	指示请求创建了资源，例如在创建 iControl REST 事务时。

响应代码	返回	说明
400 错误的请求	所有 HTTP 方法	指示格式错误的请求，例如资源的名称不正确。
401 未经授权	所有 HTTP 方法	指示省略的 http 授权头，或者您缺少完成请求所需的足够权限。
403 被禁止的	所有 HTTP 方法	指示为管理员提供的凭据缺少请求的足够权限，或者尝试执行不受支持的操作，例如删除属性。
404 找不到	所有 HTTP 方法	表示试图访问不存在的资源。
409 冲突	POST, PUT	指示创建已存在的资源的尝试。如果尝试使用 POST 方法创建资源，并且资源已经存在，ICONTROL REST 会生成此响应。
415 不支持的媒体类型	POST, PUT	指示请求中包含格式不正确的 JSON 正文，或者可能指定了不正确的 内容类型标题值。

响应代码	返回	说明
500 内部服务器错误	所有 HTTP 方法	指示 IControl rest 进程不可用，例如进程尚未启动时。
501 未实现	POST	指示端点不存在，或相应的 不支持 tmsh 请求。

检索访问策略管理器资源

使用 iControl® REST，您可以查询访问策略管理器（APM®）资源。

1. 要发现访问策略管理器（APM）资源，请向终结点发出 GET 请求 /mgmt/tm/apm.

```
GET https://192.168.25.42/mgmt/tm/apm
```

响应显示 APM 集合的结构

```
{
   "kind":"tm:apm:apmcollectionstate",
   "selfLink":"https://localhost/mgmt/tm/apm?ver=12.1.0",
   "items":[
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/aaa?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/configuration
            ?ver=12.1.0"

         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/epsec?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/ntlm?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/policy?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/profile?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/report?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/resource?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/saml?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/sso?
            ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/acl?
            ver=12.1.0"
         }
```

```
      },
      {
         "reference":{
           "link":"https://localhost/mgmt/tm/apm/apm-avr-config?ver=12.1.0"

         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/application?
            ver=12.1.0"
         }
      },
      {
         "reference":{

"link":"https://localhost/mgmt/tm/apm/application-family?ver=12.1.0"
         }
      },
      {
         "reference":{

"link":"https://localhost/mgmt/tm/apm/application-filter?ver=12.1.0"
         }
      },
      {
         "reference":{
            "link":"https://localhost/mgmt/tm/apm/log-setting?ver=12.1.0"
         }
      },
      {
         "reference":{
            "link":"https://localhost/mgmt/tm/apm/risk-class?ver=12.1.0"
         }
      },
      {
         "reference":{ "link":"https://localhost/mgmt/tm/apm/session?
            ver=12.1.0"
         }
      },
      {
         "reference":{
            "link":"https://localhost/mgmt/tm/apm/swg-scheme?ver=12.1.0"
         }
      },
      {
         "reference":{
            "link":"https://localhost/mgmt/tm/apm/url-filter?ver=12.1.0"
         }
      }
   ]
}
```

2. 要发现 APM 中的某个资源，如/ntlm，请向端点发出 GET 请求 `/mgmt/tm/apm/ntlm`.

```
GET https://192.168.25.42/mgmt/tm/apm/ntlm
```

响应显示/mgmt/tm/apm/ntlm 命名空间中的资源。

```
{
   "kind":"tm:apm:ntlm:ntlmcollectionstate",
   "selfLink":"https://localhost/mgmt/tm/apm/ntlm?ver=12.1.0",
   "items":[
      {
```

```
          "reference":{

"link":"https://localhost/mgmt/tm/apm/ntlm/machine-account?ver=12.1.0"
          }
       },
       {
          "reference":{
            "link":"https://localhost/mgmt/tm/apm/ntlm/ntlm-auth?ver=12.1.0"

          }
       }
    ]
}
```

访问策略管理器终结点

iControl® REST 支持此处列出的访问策略管理器（APM®）终结点。所有端点都与流量管理命名空间/mgmt/tm 相关。

终点	说明
/apm/aaa	配置授权、身份验证和记帐（AAA）设置。您可以将 apm 配置为使用各种服务器来提供用户身份验证、访问资源的授权和用户活动的记帐。APM 支持 RADIUS、RSA Native SecurID 和 Windows 活动目录等。
/apm/acl	使用访问控制列表（ACL）限制对主机和端口组合的访问。
/apm/apm-avr-config	配置应用程序可见性和报告（AVR）的设置。
apm/application	指定可以通过修改默认的“允许”或“阻止”操作来控制的基于 Web 的应用程序。
/apm/application-family	指定应用程序的类别，例如即时消息或电子邮件。
/apm/application-filter	指定可用于允许或阻止访问应用程序的应用程序筛选器。
/apm/configuration	指定安全 Web 网关（SWG）初始化的设置。
/apm/epsec	配置 APM 以启用客户端和服务器端端点安全检查。
/apm/log-setting	配置 APM 以记录访问策略事件或审核事件。
/apm/ntlm	将 APM 配置为使用 NTLM。您可以为 APM 创建一个计算机帐户以加入 Windows 域。使用计算机帐户的身份验证请求创建与域控制器通信的安全通道。
/apm/policy	为方案分配配置策略。
/apm/profile	配置流量处理的配置文件。
/apm/report	配置报表设置。
/apm/resource	指定网络访问和 Web 访问资源。
/apm/risk-class	指定风险等级。
/apm/saml	为安全断言标记语言（SAML）框架配置 APM，以创建、请求和交换身份验证和授权数据。

终点	说明
	您可以将 APM 配置为本机 SAML 2.0 身份提供程序（IDP），或者另一个 SAML IDP 的代理，如活动目录联合服务（ADFS）。
/apm/session	检索和管理用户会话。
/apm/sso	配置 APM 以使用单点登录（SSO）功能。您可以为 SSO 定义用户名、密码和身份验证方法的属性，以及许多基于 HTTP 表单的 SSO 对象属性。
/apm/swg-scheme	配置安全 Web 网关（SWG）方案以筛选和分类 URL。方案允许您为特定日期或一天中的特定时间对 URL 筛选器进行分组和计划。
/apm/url-filter	将 APM 配置为使用 URL 筛选器指定要允许或阻止的一个或多个 URL 类别。使用此终结点，可以创建多个 URL 筛选器。除了默认的 URL 过滤器外，还可以删除 URL 过滤器。

在 APM 中配置 LDAP 设置

身份验证、授权和审核设置允许您在访问策略管理器（APM）中配置 LDAP 设置。LDAP 是许多供应商支持的 X.500 目录访问协议（DAP）的轻量级实现。 iControl® REST API 允许您配置 LDAP 服务器配置，但不能配置 LDAP 服务器的功能。

1. 在尝试添加 LDAP 账户对其进行配置之前，请向`/mgmt/mt/apm/aaa/ldap/example` 获取引用对象的终结点。

```
GET https://192.168.25.42/mgmt/tm/apm/aaa/ldap/example
```

可以使用/example 端点获取 APM 资源或 IControl rest 中任何资源的表示。

```
{
   "kind":"tm:apm:aaa:ldap:ldapcollectionstate",
   "selfLink":"https://localhost/mgmt/tm/apm/aaa/ldap/example?ver=12.1.0",
   "items":[
      {
         "propertyDescriptions":{
             "address":"",
            "adminDn":"",
            "adminEncryptedPassword":"",
            "appService":"",
            "baseDn":"",
            "cleanupCache":"",
            "description":"",
            "groupCacheTtl":"",
            "isLdaps":"",
            "locationSpecific":"",
            "pool":"",
            "port":"",
            "schemaAttr":{
               "groupMember":"",
               "groupMemberValue":"",
               "groupMemberof":"",
               "groupObjectClass":"",
```

```
                "userMemberof":"",
                "userObjectClass":""
             },
             "serversslProfile":"",
             "timeout":"",
             "usePool":""
          },
          "address":"any6",
          "adminDn":"",
          "adminEncryptedPassword":"",
          "appService":"",
          "baseDn":"",
          "cleanupCache":"none",
          "description":"",
          "groupCacheTtl":30,
          "isLdaps":"false",
          "locationSpecific":"true",
          "pool":"",
          "port":389,
          "schemaAttr":{
             "groupMember":"member",
             "groupMemberValue":"dn",
             "groupMemberof":"memberOf",
             "groupObjectClass":"group",
             "userMemberof":"memberOf",
             "userObjectClass":"user"
          },
          "serversslProfile":"",
          "timeout":15,
          "usePool":"enabled",
          "naturalKeyPropertyNames":[
             "name",
             "partition",
             "subPath"
          ]
       }
    ]
}
```

2. 要将 LDAP 服务器设置配置为与 APM 一起使用时，请向/mgmt/tm/apm/aaa/ldap 终端。确保将 application/json 指定内容类型。

```
POST https://192.168.25.42/mgmt/tm/apm/aaa/ldap

{
  "name": "test_aaa_ldap",
  "address": "10.1.1.1",
  "adminDn": "\"CN=administrator, CN=users, DC=mydomain, DC=com\"",
  "adminEncryptedPassword": "p4s8w07d",
  "usePool": "disabled"
}
```

此示例使用对象中找到的属性的一小部分。如 JSON 所示，必须对 JSON 字符串中的引号（'\'）进行转义以保留引号。如果要将 LDAP 服务器用作身份验证或查询服务器，则必须使用可视化策略编辑器并手动进行更改。

响应包含一个状态代码（200ok），指示请求是否成功，但 iControl REST 也在响应中包含新创建的资源。

```
{
   "kind":"tm:apm:aaa:ldap:ldapstate",
   "name":"test_aaa_ldap",
   "fullPath":"test_aaa_ldap",
```

```
   "generation":30,

"selfLink":"https://localhost/mgmt/tm/apm/aaa/ldap/test_aaa_ldap?ver=12.1.0",

   "address":"10.1.1.1",
   "adminDn":"CN=Administrator, CN=Users, DC=mydomain, DC=com",
   "adminEncryptedPassword":"$M$Uq$lXbiDrLRf0Ogq4zAX0pvYQ==",
   "cleanupCache":"none",
   "groupCacheTtl":30,
   "isLdaps":"false",
   "locationSpecific":"true",
   "port":389,
   "schemaAttr":{ "groupMember":"
      member",
      "groupMemberValue":"dn",
      "groupMemberof":"memberOf",
      "groupObjectClass":"group",
      "userMemberof":"memberOf",
      "userObjectClass":"user"
   },
   "timeout":15,
   "usePool":"disabled"
}
```

3. 要删除 LDAP 设置，请发出删除请求，并从上一步指定 LDAP 服务器名称（test_a a a-ldap）

```
DELETE https://192.168.25.42/mgmt/tm/apm/aaa/ldap/test_aaa_ldap
```

iControl REST 删除资源并用 HTTP 响应进行响应。响应不包括 JSON 主体。

在本例中，您配置了 LDAP 服务器设置。使用 reference 对象作为起点，通过指定一小组属性创建一个新的 LDAP 服务器。查看新的 LDAP 服务器后，通过指定资源名称删除该服务器。

在 APM 中创建自定义类别

在云科®系统上，您可以选择在 URL 数据库中使用默认类别集，或定义 URL 类别和筛选器。如果您有安全 web 网关（SWG）订阅，则可以创建自定义 URL 类别来扩展 URL 数据库。如果您没有 SWG 订阅，您仍然可以创建自定义 URL 类别。使用 iControl® REST API，您可以按照两步过程创建自定义 URL 类别，然后将自定义类别附加到 URL 筛选器。

1. 要创建自定义类别，请向/sys/url db/url 类别终结点发出 GET 请求。使用响应来确定是否存在类别，以及是否允许或阻止类别。

```
GET https://192.168.25.42/mgmt/tm/sys/url-db/url-category
```

```
{
   "kind":"tm:sys:url-db:url-category:url-categorycollectionstate",
  "selfLink":"https://localhost/mgmt/tm/sys/url-db/url-category?ver=12.1.0",

   "items":[
```

```
      {
         "kind":"tm:sys:url-db:url-category:url-categorystate",
         "name":"Entertainment",
         "partition":"Common",
         "fullPath":"/Common/Entertainment",
         "generation":1,

"selfLink":"https://localhost/mgmt/tm/sys/url-db/url-category/~Common~Entertainment?ver=12.1.0",

         "catNumber":10,
         "defaultAction":"allow",
         "description":"Sites with information about entertainment.",
         "displayName":"Entertainment",
         "isCustom":"false",
         "isRecategory":"false",
         "parentCatNumber":0,
         "severityLevel":0
      },
... (Truncated for readability)
      {
         "kind":"tm:sys:url-db:url-category:url-categorystate",
         "name":"Business",
         "partition":"Common",
         "fullPath":"/Common/Business",
         "generation":1,

"selfLink":"https://localhost/mgmt/tm/sys/url-db/url-category/~Common~Business?ver=12.1.0",

         "catNumber":1902,
         "defaultAction":"block",
         "displayName":"Business",
         "isCustom":"true",
         "isRecategory":"false",
         "parentCatNumber":0,
         "severityLevel":0,
         "urls":[
            {
               "name":"http://www.example.com/*",
               "type":"glob-match"
            },
            {
               "name":"http://www.example.com/?/",
               "type":"exact-match"
            }
         ]
      }
   ]
}
```

2. 在确定自定义类别不存在之后，创建自定义类别。与其他APM 示例一样，如果希望看到对象的示例表示，请将/example 端点附加到上一步中的 URL。

```
POST https://192.168.25.42/mgmt/tm/sys/url-db/url-category

{
  "displayName": "my-custom-category",
  "defaultAction": "block",
  "urls": [ ]
}
```

3. 要将自定义类别附加到URL 筛选器，请向/tm/apm/url 筛选器发出 POST 请求。

```
POST https://192.168.25.42/mgmt/tm/apm/url-filter

{
  "name": "my-url-filter",
  "allowedCategories": "my-custom-category"
}
```

与任何 iControl REST 请求一样，响应显示请求的结果。

在本例中，您创建了一个自定义 URL 类别，并将该自定义类别附加到 URL 筛选器。

在 APM 中管理用户会话

访问策略管理器（APM）使用会话标识符（会话 ID）跟踪用户会话。这个 APM 中的访问信息端点允许您对所有用户会话的列表发出 iControl®rest 请求。响应包含每个会话的会话 ID、用户登录名和 IP 地址。作为会话管理过程的一部分，可以向会话终结点发出 iControl REST 请求以删除特定会话。

1. 要查看 APM 中的当前用户会话，请向/mgmt./tm/apm/access info 发出 GET 请求。

```
GET https://192.168.25.42/mgmt/tm/apm/access-info
```

对该请求的响应包括以下数据：

```
{
  "apiRawValues":
    { "apiAnonymous":
    {
     "apm::access-info" "914c727f (login user=user1) client (IP=10.20.36.2)"
...(Truncated for readability)
  "kind": "tm:apm:access-info:access-infostats",
  "selfLink": "https://localhost/mgmt/tm/apm/access-info?ver=12.1.0"
}
```

2. 要查看特定用户名的会话，请向 mgmt/tm/apm/access-info 发出 GET 请求，端点并使用选项查询参数指定用户名

```
GET https://192.168.25.42/mgmt/tm/apm/access-info?ver=12.0.0&options=apm-user
```

以类似的方式，您还可以指定一个 IP 地址，以获取指定 IP 地址的所有会话的列表。使用与示例中相同的查询参数（选项）。

3. 要删除会话，请发出删除请求，并将标识资源的会话标识附加到/mgmt/tm/apm/session 端点。

```
DELETE https://192.168.25.42/mgmt/tm/apm/session/914c727f
```

受影响的用户将无法再访问资源。用户必须重新登录。如果请求成功，则响应为200 OK。

在本例中，您向 APM 发出了一个 iControl REST 请求，以获取所有用户会话的列表，并发出了一个附加请求以删除特定会话。

列出 OAuth 令牌

对于所有 OAuth 令牌示例，您必须在云科®系统上配置访问策略管理器（APM），以充当授权服务器（AS）或外部提供商（如 Facebook 或 Google）的客户端。请遵循云科®访问策略管理器®中概述的步骤：身份验证和单点登录，版本 13.0 指南。

如果需要所有 OAuth 令牌的详细信息，可以查询默认数据库。对于在大型 IP 系统上配置的授权服务器（AS），向 AS 发出 GET 请求。

要查询令牌数据库，请发出 GET 请求。

响应包含特定于数据库中每个令牌的属性。您可以使用请求的输出来获取吊销令牌所需的 oauthid 和客户机 ID 属性。

```
GET https://192.168.25.42/mgmt/tm/apm/oauth/token-details
```

在本例中，您请求列出数据库中的所有令牌。

获取 OAuth 令牌的计数

对于所有 OAuth 令牌示例，您必须在云科®系统上配置访问策略管理器（APM），以充当授权服务器（AS）或外部提供商（如 Facebook 或 Google）的客户端。请遵循云科®访问策略管理器®中概述的步骤：身份验证和单点登录，版本 13.0 指南。

作为管理对资源的访问的任务的一部分，您可能需要查询 OAuth 数据库以获取用户令牌数的计数。根据您选择的配置，此类请求使用默认数据库实例。根据所需的输出，您可以按应用程序名称查询所有令牌的计数或令牌的计数。

1. 要查询令牌计数，发出GET 请求。

```
GET https://192.168.25.42/mgmt/tm/apm/oauth/token-details/stats?ver=13.0
```

响应将包含 json 内容，类似于以下输出：

```
{
   "apiRawValues" : { "apiAnonymous" : "Total tokens : 7\n" },
   "kind" : "tm:apm:oauth:token-details:token-detailscollectionstats",
   "selfLink" :
"https://localhost/mgmt/tm/apm/oauth/token-details/stats?ver=12.1.0"
}
```

2. 要查询与应用程序相关联的令牌计数，请发出GET 请求。

```
GET
https://192.168.25.42/mgmt/tm/apm/oauth/token-details/stats?ver=13.0&options="app-name",
 "application name"
```

在请求中以双引号（“”）提供应用程序的名称，而不是在示例中显示的字符串。

按应用程序计数响应将包含 JSON 内容，类似于以下输出：

```
{
   "apiRawValues" : { "apiAnonymous" : "Total tokens : 5\n" },
   "kind" : "tm:apm:oauth:token-details:token-detailscollectionstats",
   "selfLink" :
"https://localhost/mgmt/tm/apm/oauth/token-details/stats?ver=12.1.0"
}
```

在本例中，您请求了数据库中所有令牌的计数，然后请求了特定应用程序的所有令牌的计数。

撤销 OAuth 令牌

对于所有 OAuth 令牌示例，您必须在云科®系统上配置访问策略管理器（APM），以充当授权服务器（AS）或外部提供商（如 Facebook 或 Google）的客户端。请遵循云科®访问策略管理器®中概述的步骤：身份验证和单点登录，版本 13.0 指南。

如果需要撤销 OAuth 令牌，请调用 REST 来处理撤销。对于在云科系统上配置的授权服务器（AS），向 AS 发出 POST 请求。

要撤销令牌，请发出 POST 请求。OAuth ID 和客户机 ID 属性可以在令牌详细信息列表的输出中找到。这两个属性都是字符串值，必须在 JSON 正文中用引号括起来

```
POST https://192.168.25.42/mgmt/tm/apm/oauth/token-details
```

```
{
   "command" : "revoke",
   "name" : "<oauthid>",
   "client-id" : "<clientid>",
   "db-instance" : "<database name>"
}
```

在本例中，您撤销了一个令牌。您使用上一个示例的输出来查找标识标记的关联属性。

API 生命周期

REST API 生命周期策略

REST API 生命周期策略描述了一种管理 REST 集合和 tmsh 资源的目的或寿命的方法。生命周期管理提供了几个影响资源和方法的用例，而 REST API 生命周期策略旨在提供关于资源和属性的有用信息，这些资源和属性可能是新的、实验性的，或者正在逐步淘汰。虽然资源或资源属性的弃用是最有可能使用的用例，但是对于早期访问特征，以及内部使用或测试，存在其他用例。您应该将资源或属性的折旧解释为不鼓励使用资源或属性，而不是在短期内删除资源或资源属性。我们的目标是让您在更改发生之前了解更改。

API 生命周期策略引入状态值，默认值为无状态。NO_STATUS 值表示尚未确定资源或属性，并且 REST 不会记录这些资源的使用情况。REST 只记录与您配置的状态值匹配的资源和属性的使用情况。对于弃用和早期访问资源，自定义 REST 头（X-YK-Api-Status）指示以下值之一：

- 废弃的_资源
- 弃用资源
- 早期访问资源
- 提前进入资源

使用 REST API 生命周期更改

REST API 生命周期策略的实现提供了 API 状态值和作为请求日志项的附加信息。下面的示例演示了一个带有各种 HTTP 动词和相应的头（如果有的话）以及日志项的请求。

重要提示：此功能仅适用于资源集合。

1. 要生成资源查询的生命周期输出，请向/mgmt/tm/ltm/profile/ocsp 装订参数端点

```
GET https://192.168.25.42/mgmt/tm/ltm/profile/ocsp-stapling-params
```

```
{
  "kind" :
"tm:ltm:profile:ocsp-stapling-params:ocsp-stapling-paramscollectionstate",
  "selfLink":
"https://localhost/mgmt/tm/ltm/profile/ocsp-stapling-params?ver=13.0.0"
}
```

2. 要查找资源的详细信息，请在/var/log/icrd 日志。

```
Dec 30 23:59:36 localhost notice icrd_child: 18826,18853,iControl REST Child

     Daemon,WARNING,[api-status-warning]: ltm/profile/ocsp-stapling-params:
 deprecated
```

日志消息包括标识，例如[api status warning]以指示 REST API 生命周期管理的日志条目

3. 要生成资源属性查询的输出，请向/mgmt./tm/ltm/profile/fast14/fast14 端点。
.

```
GET https://192.168.25.42/mgmt/tm/ltm/profile/fastl4/fastL4
```

```
{
....
  "serverTimestamp": "disabled",
  "softwareSynCookie": "disabled",
  "synCookieEnable": "enabled",
....
}
```

4. 要查找资源属性的详细信息，请在/var/log/icrd 日志。

```
Dec 31 00:05:02 localhost notice icrd_child: 18826,18853,    iControl REST
Child
      Daemon,WARNING,[api-status-warning]: ltm/profile/fastl4: no status;
properties: deprecated:
      ltm/profile/fastl4/hardware-syn-cookie,
ltm/profile/fastl4/software-syn-cookie
```

5. 要生成用于创建新资源的生命周期输出，请向/mgmt/tm/ltm/profile/ocsp 装订参数端点。指定 JSON 主体，如图所示。

```
POST https://192.168.25.42/mgmt/tm/ltm/profile/ocsp-stapling-params
```

```

```

注意：响应包括头和值 *X-YK-Api-Status: DEPRECATED_RESOURCE*

6. 要查看请求的日志信息，请在/var/log/icrd 日志中找到条目

```
Jan  5 01:14:08 localhost notice icrd_child[2562]:  2562, 2567,    iControl
 REST Child
     Daemon,WARNING,[api-status-warning]: ltm/profile/ocsp-stapling-params:
 deprecated
```

7. 如要生成用于创建新资源的生命周期输出，并且已弃用的API Allowed 设置为 false，请向 /mgmt./tm/ltm/profile/ocsp 装订参数端点发出 POST 请求。指定 JSON 主体，如图所示：

```
POST https://192.168.25.42/mgmt/tm/ltm/profile/ocsp-stapling-params
```

```
{
  "name": "myocsp",
  "dnsResolver": "dns-resolver-1"
}
```

注意：响应包括状态消息 *HTTP/1.1 404 Not Found.*

8. 要查看请求的日志信息，请在/var/log/rest javad.0.1.log 日志中找到改条目。请注意，该条目出现在与前面十里不同的日志文件中。

```
[WARNING][157][05 Jan 2017 01:23:42 UTC][8100/mgmt/shared/resolver/groups
      ForwarderPassThroughWorker] [api-status-warning] The deprecate API
     /mgmt/tm/ltm/profile/ocsp-stapling-params/ is not available as per the

      /shared/settings/api-status/availability
```

9. 要生成用于创建资源属性的输出，请向/mgmt/tm/ltm/profile/fastl4/fastl4 端点。指定 JSON 主体，如图所示。

```
POST https://192.168.25.42/mgmt/tm/ltm/profile/fastl4
```

```
{
  "name": "myfastl4",
  "softwareSynCookie": "enabled"
}
```

注意：响应包括头和值 *X-YK-Api-Status: DEPRECATED_PROPERTY。*

10. 要查看请求的日志信息，请在/var/log/icrd 日志中找到该条目。

```
Jan  5 17:26:53 localhost notice icrd_child[2562]:  2562, 2568,    iControl
 REST Child
      Daemon,WARNING,[api-status-warning]: ltm/profile/fastl4: no status;
properties: deprecated:
      ltm/profile/fastl4/hardware-syn-cookie,
ltm/profile/fastl4/software-syn-cookie
```

11. 若要生成用于创建新资源属性的生命周期输出，并且已弃用的“允许 API”设置为 false，请向 /mgmt./tm/ltm/profile/fast14 终结点发出 POST 请求。指定 JSON 主体，如图所示。

```
POST https://192.168.25.42/mgmt/tm/ltm/profile/fastl4
```

```
{
  "name": "myfastl4",
  "softwareSynCookie": "enabled"
}
```

注意：响应包括状态消息 HTTP/1.1 404 Not Found，以及头和值 x-yk-api-status:DEPRECATED_PROPERTY。

12. 要查看请求的日志信息，请在/var/log/icrd 日志中找到条目。

在 tmsh 中使用 REST API 生命周期更改

下面的示例显示了与前面的 REST 示例等效的 tmsh。

1. 要生成资源的 API 生命周期输出，请运行 tmsh 命令列出资源。

（tmos）#list ltm configuration file ocsp binding parameters

```
api-status-warning] ltm/profile/ocsp-stapling-params is deprecated
ltm profile ocsp-stapling-params ocsp-cur {
    dns-resolver dns-resolver-cur
}
```

2. 要查找资源的详细信息，请在 /var/log/ltm 日志。

```
Dec 30 16:00:43 localhost warning tmsh[1409]: 01420013:4: [api-status-warning]

      ltm/profile/ocsp-stapling-params is deprecated
```

3. 要为资源属性生成 API 生命周期输出，请运行 tmsh 命令，如图所示。
(tmos)# list ltm profile fastl4 fastL4 software-syn-cookie

```
[api-status-warning] ltm/profile/fastl4, properties : deprecated :
        software-syn-cookieltm profile fastl4 fastL4 {
...
   reassemble-fragments disabled
```

```
    reset-on-timeout enabled
    software-syn-cookie enabled
}
```

4. 要查找资源属性的详细信息，请在 /var/log/ltm 日志。

```
Dec 30 16:02:49 localhost warning tmsh[1731]: 01420013:4: [api-status-warning]
      ltm/profile/fastl4, properties : deprecated : software-syn-cookie
```

5. 要生成用于创建新资源的生命周期输出，请运行 tmsh 命令，如图所示。输出出现在命令之后。

```
(tmos)# create ltm profile ocsp-stapling-params myocsp dns-resolver
dns-resolver-1
```

```
[api-status-warning] ltm/profile/ocsp-stapling-params is deprecated
```

6. 要查找资源的详细信息，请在 /var/log/ltm 日志

```
Jan  5 09:49:54 localhost warning tmsh[4542]: 01420013:4: [api-status-warning]
      ltm/profile/ocsp-stapling-params is deprecated
```

7. 要生成用于创建新资源的生命周期输出，并将不推荐使用的 API Allowed 设置为 false，请运行 tmsh 命令，如图所示。输出出现命令之后。

```
(tmos)# create ltm profile ocsp-stapling-params myocsp dns-resolver
dns-resolver-1
```

```
[api-status-warning] ltm/profile/ocsp-stapling-params is deprecated. This
command is not
      available or has properties which are not available.
```

8. 要查找资源的详细信息，请在 /var/log/ltm 日志。

```
Jan  5 09:49:01 localhost warning tmsh[4493]: 01420013:4: [api-status-warning]
      ltm/profile/ocsp-stapling-params is deprecated
```

9. 要生成用于创建资源属性的输出，请运行tmsh 命令，如图所示。输出出现在命令之后。

```
(tmos)# create ltm profile fastl4 myfastl4 software-syn-cookie enabled
```

```
[api-status-warning] ltm/profile/fastl4, properties : deprecated :
software-syn-cookie
```

10. 要查找资源属性的详细信息，请在 `/var/log/ltm` 日志

```
Jan  5 09:44:28 localhost warning tmsh[4426]: 01420013:4: [api-status-warning]
      ltm/profile/fastl4, properties : deprecated : software-syn-cookie
```

11. 要生成用于创建新资源的生命周期输出，并将不推荐使用的 API Allowed 设置为 false，请运行 tmsh 命令，如图所示。输出出现命令之后。

```
(tmos)# create ltm profile fastl4 myfastl4 software-syn-cookie enabled
```

```
[api-status-warning] ltm/profile/fastl4, properties : deprecated :
software-syn-cookie
```

12. 要查找资源属性的详细信息，请在 `/var/log/ltm` 日志

```
Jan  5 09:44:28 localhost warning tmsh[4426]: 01420013:4: [api-status-warning]
      ltm/profile/fastl4, properties : deprecated : software-syn-cookie
```

配置 REST API 生命周期设置

REST API 生命周期策略支持可配置的 API 状态和日志设置。通过启用特定设置，可以启用 REST 和 TMSH 记录信息。

1. 要查看REST 的设置，发出 GET 请求。

```
GET https://192.168.25.42/mgmt/shared/settings/api-status/availability/
```

查询结果将类似于以下代码片段：

```
{
   "deprecatedApiAllowed": "true",
   "earlyAccessApiAllowed": "true",
   "testOnlyApiAllowed": "false"
}
```

重要提示：这些设置会影响资源的可见性。如果将任何状态指定为禁用，则将无法在 REST 请求或 tmsh 中查看该资源。REST 请求将生成 404（未找到）响应代码。在 tmsh 中，tab completion 不会公开这些资源。

2. 若要更改API 的状态设置，请发出修补程序或 POST 请求。在 JSON 主体中，指定要更改的可见性设置。例如

```
PATCH https://192.168.25.42/mgmt/shared/settings/api-status/availability/
```

```
{
   "earlyAccessApiAllowed": "false"
}
```

通过发出查询请求确认更改成功。

3. 要制定生成日志项的资源设置，请发出GET 请求。

```
GET https://192.168.25.42/mgmt/shared/settings/api-status/log/resource
```

查询结果将再次类似于以下内容：

```
{
   "deprecatedApiAllowed": "true",
   "earlyAccessApiAllowed": "true",
   "testOnlyApiAllowed": "false"
}
```

4. 要制定生成日志项的属性资源设置，请发出GET 请求（请忽略结果）。

```
GET
https://192.168.25.42/mgmt/shared/settings/api-status/log/resource-property
```

对于任一端点，按图所示制作修补程序或 POST 请求以修改任何设置。在本主题中，

您查询并配置了 API 生命周期的可见性和日志记录设置。

使用 tmsh 配置 REST API 生命周期设置

REST API 生命周期策略支持可配置的 API 状态和日志设置。可以使用 tmsh 命令配置 API 状态和日志设置。

1. 要查看设置，请键入以下 tmsh 命令。
   ```
   (tmos)# list mgmt shared settings api-status availability
   ```
2. 要修改不推荐的设置，请键入以下 tmsh 命令。
   ```
   (tmos)# modify mgmt shared settings api-status availability
   { deprecatedApiAllowed value false }
   ```
   该命令将状态更改为禁用。
3. 要查看日志资源的设置，请键入以下 tmsh 命令
   ```
   (tmos)# list mgmt shared settings api-status log resource
   ```

通过在命令中指定 resource property 而不是 resource，可以查看资源属性的日志设置

```
mgmt shared settings api-status log resource
{
  deprecatedApiAllowed true
  testOnlyApiAllowed true
  earlyAccessApiAllowed true
}
```

在本主题中，您使用 tmsh 命令查看和修改 api 状态和日志设置。

附加功能

关于示例后缀

URI 末尾包含/example 后缀会提示 iControl® REST 生成资源的示例表示。可以在 GET 请求中使用/example suff 来生成包含所有属性（包括空属性）的表示。示例表示还包括描述每个属性的帮助文本字符串和资源的自然键列表。自然密钥由一个或多个用户友好的属性组成，这些属性唯一标识资源，如区号/电话号码。

在 iControl REST 中，一个自然键在 JSON 中表示为一个 NaturalKeysPropertyNames 属性，它是一个名称、分区和子路径的数组。名称、分区和子路径构成对象的完整路径。对于公共分区中的资源，iControl REST 将分区名称作为自然键省略。如果对象是单例对象，则 NaturalKeysPropertyNames 数组为空。

仅支持 tmsh 命令 SHOW、LIST、DELETE、LOAD、SAVE、INSTALL 或 RUN 的云科®系统组件没有默认字段值。对于这些组件，对`/example`终结点不生成具有默认值的响应。

Application Security Manager™ (ASM™)资源的示例表示仅包括默认值和可能的枚举值。ASM 资源的示例表示不包括属性的说明作为帮助文本。

如果存在缺省值，则示例表示指定属性的默认值。如果特性没有默认值，则表示形式包括：

- 字符串属性的空字符串（“”）
- 数值特性为零（0）
- 布尔属性为 false
- 空的 JSON 数组或对象

如果属性从枚举中获取值，则表示形式将可接受的值显示为数组。iControl REST 还将后缀 Enums 附加到此数组的名称以标识枚举。

提示：复制示例表示，对副本进行更改，然后将更改粘贴到 POST 请求的 JSON 主体中。

关于访问策略管理器

访问策略管理器（APM）为云科®系统提供安全的标识和访问管理。iControl® REST 公开了 APM 端点，以实现对 APM 资源的编程访问和自动化的好处。

APM 遵循本指南前面描述的其他原则：

- URI 结构支持对集合和资源的一致访问
- 资源中的链接，包括自链接、支持发现
- JSON 编码简化了资源的表示

- HTTP 传输提供了与资源交互的方法，以及安全性、身份验证、缓存和内容协商

关于 HTTP 响应代码

对所有 IControl®rest 请求的响应都包含一个响应代码，如下所示。

成功回应

响应代码	说明
200 OK	Indicates success for all methods.

Error responses

响应代码	HTTP 方法	说明
400 错误的请求	全部的	可能的原因包括： • 格式错误的 HTTP 请求 • 请求中资源的名称不正确
401 未经授权	全部的	可能的原因包括： • 缺少 HTTP 授权头 • 为管理员提供的凭据权限不足
403 被禁止的	全部的	可能的原因包括： • 为管理员提供的凭据权限不足 • 尝试执行不受支持的操作，例如删除属性
404 找不到	全部的	可能的原因包括： • 试图访问数据库中不再存在的资源
409 冲突	POST, PUT	可能的原因包括： • 试图创建已存在的资源 指示请求的状态更改与资源的当前状态之间的冲突。例如，如果您发布已经存在的资源，则这是错误响应。
415 不支持的媒体类型	POST, PUT	可能的原因包括： • 指定不正确的内容类型标题值 • 使用 POST 或 PUT 请求指定格式错误的 JSON 主体

响应代码	HTTP 方法	说明
500 内部服务器错误	全部的	可能的原因包括： • 进程未运行时尝试访问 IControl REST
501 未实现	POST	可能的原因包括： • 试图访问不存在的端点 • 试图通过 iControl REST 调用不受支持的 tmsh 命令

关于日志文件

从控制台或到云科®设备的 SSH 连接中，可以找到 iControl®REST 的以下日志文件：

- `/var/log/restjavad-audit.0.log` 显示对 `iControl REST` 服务的所有身份验证。这是每一个 `REST` 调用的有序列表。
- `/var/log/restjavad.0.log` 包含有关到 `iControl REST` 服务的连接的信息，例如返回的错误。
- `/var/log/icrd` 显示 `icrd` 进程的操作，该进程管理用于处理其余调用的线程
- `/var/log/ltm` 包含来自 `mcpd` 的消息，`icrd` 调用该进程来管理系统配置。

使用标准的 Unix 命令处理这些文件，如 tail、grep 和 less。在本例中，会话通过 ssh 登录到一个云科系统，并使用 tail-f 监视
/var/log/restjavad-audit.0.log 日志文件：

```
juser@bench2:~/$ ssh root@192.168.25.42
Password: default
Last login: Fri Mar 29 09:03:25 2013 from 192.168.98.174
[root@localhost:Active:Standalone] config # tail -f
/var/log/restjavad-audit.0.log
[I][339][29 Mar 2013 16:04:06 UTC][ForwarderPassThroughWorker] \
  [run]{"user":"admin","method":"PUT",\
  "uri":"http://localhost:8100/mgmt/tm/ltm/pool/dns-pool2",\
  "status":"succeeded","from":"192.168.96.37"}
[I][340][29 Mar 2013 16:04:06 UTC][ForwarderPassThroughWorker] \
  [run]{"user":"admin","method":"GET",\
  "uri":"http://localhost:8100/mgmt/tm/ltm/pool","\
  status":"succeeded","from":"192.168.96.37"}
[I][341][29 Mar 2013 16:04:06 UTC][ForwarderPassThroughWorker] \
  [run]{"user":"admin","method":"DELETE",\
  "uri":"http://localhost:8100/mgmt/tm/ltm/pool/test-pool2",\
  "status":"succeeded","from":"192.168.96.37"}
[I][342][29 Mar 2013 16:04:07 UTC][ForwarderPassThroughWorker] \
  [run]{"user":"admin","method":"POST",\
  "uri":"http://localhost:8100/mgmt/tm/sys/folder",\
  "status":"succeeded","from":"192.168.96.37"}
[I][343][29 Mar 2013 16:04:07 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"admin","method":"DELETE",\
  "uri":"http://localhost:8100/mgmt/tm/sys/folder/~fw_objs",\
  "status":"succeeded","from":"192.168.96.37"}
```

```
[I][344][29 Mar 2013 16:04:07 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"admin","method":"DELETE",\
  "uri":"http://localhost:8100/mgmt/tm/sys/folder/~eu~east~romania",\
  "status":"succeeded","from":"192.168.96.37"}
[I][345][29 Mar 2013 16:04:07 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"admin","method":"POST",\
  "uri":"http://localhost:8100/mgmt/shared/authz",\
  "status":"succeeded","from":"192.168.96.37"}
[I][346][29 Mar 2013 16:04:07 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"admin","method":"GET",\
  "uri":"http://localhost:8100/mgmt/shared/authz",\
  "status":"succeeded","from":"192.168.96.37"}
[I][347][29 Mar 2013 16:04:10 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"dns_admin","method":"GET",\
  "uri":"http://localhost:8100/mgmt/tm/sys",\
  "status":"succeeded","from":"192.168.96.37"}
[I][350][29 Mar 2013 16:04:10 UTC][ForwarderPassThroughWorker]\
  [run] {"user":"admin","method":"GET",\
  "uri":"http://localhost:8100/mgmt/tm/ltm/pool/http-pool?$stats=true",\
  "status":"succeeded","from":"192.168.96.37"}
...
```

如果需要调整 ICRD 的日志记录级别，请联系云科®网络技术支持 (`http://www.yk.com/support/`).

关于公共 URIs

如果可以通过 iControl® REST 请求访问 URI，则该 URI 被认为是公共的。一般来说，以下所有内容都是公开的：

- 交通管理外壳（tmsh）模块
- 交通管理外壳（tmsh）组件
- 可通过 tmsh show 命令访问的任何组件属性。

要查看组件属性，请对父组件发出 GET 请求。默认情况下，不能使用 GET 请求直接通过公共 URI 获取它们。

公共 URI 用于提供对这些组件属性中的一些属性的直接访问。iControl REST 进程允许在整个包含对象（组件或集合）的 PUT 请求不易处理的情况下使用这些方法。

在许多情况下，路径的第二到最后一部分是组件的名称，您需要在路径的最后一部分之前为该组件提供特定的对象名称。例如，要访问公共 uri/mgmt/tm/gtm/pool/members，必须指定要为其指定成员的 dns 池，例如 pool5 成员的/mgmt/tm/gtm/pool/pool5/members。

法律声明

法律声明

出版日期

本文件于 2019 年 5 月 20 日发布。

出版物编号

MAN-0526-04

版权

YK 网络公司（YK）相信它提供的信息是准确可靠的。但是，YK 不承担使用本信息的责任，也不承担使用本信息可能导致的对专利或第三方其他权利的任何侵犯。YK 的任何专利、版权或其他知识产权下均不得通过暗示或其他方式授予许可，除非适用的用户许可明确说明。YK 保留随时更改规格的权利，恕不另行通知。

商标

有关 YK 商标和服务标记的当前列表，请参见 *http://www.yk.com/about/guidelines-policies/trademarks/*.

此处的所有其他产品和公司名称可能是其各自所有者的商标。

专利

本产品可能受到下列一项或多项专利的保护： *https://yk.com/about-us/policies/patents*

链路控制器可用性

此产品目前在美国不可用。

出口管制通知

本产品可能包括加密软件。根据《出口管理法》，美国政府可将从美国出口该产品视为刑事犯罪。

射频干扰警告

这是 A 级产品。在家庭环境中，本产品可能会引起无线电干扰，在这种情况下，用户可能需要采取适当的措施。

通讯委员会规章

根据 FCC 规则第 15 部分的规定，该设备经过测试，符合 A 级数字设备的限制。这些限制旨在为

设备在商业环境中运行时提供合理的保护，防止有害干扰。该单元生成、使用和

会辐射射频能量，如果不按照说明书安装和使用，可能会对无线电通信造成有害干扰。在居民区内操作本设备可能会造成有害干扰，在这种情况下，用户将需要自费采取任何可能需要的措施来纠正干扰。除非制造商明确批准，否则对本设备的任何修改都可能使用户根据 FCC 规则第 15 部分操作本设备的权限失效。

除非制造商明确批准，否则对本设备的任何修改都可能使用户根据 FCC 规则第 15 部分操作本设备的权限失效。

加拿大法规遵从性

该 A 级数字设备符合加拿大 ICES-003.

标准符合性

本产品符合制造时适用于信息技术产品的 IEC、欧盟、ANSI/UL 和加拿大 CSA 标准。

索引

A

B

C

D

E

F

G

H

I

J

K

L

M

O

P

Q

R

S

T

U

V

W