许可证
Astra Automation

NetApp
November 17, 2026

This PDF was generated from https://docs.netapp.com/zh-cn/astra-automation-2310/reference/license_endpoints.html on November 17, 2026. Always check docs.netapp.com for the latest.

目录

许可证 1
Astra API (ACC) 许可证端点 1
创建新许可证 1
HTTP 方法和端点 1
使用说明 1
参数 1
请求正文 1
响应 3
响应 6
响应 7
响应 7
响应 8
错误 8
定义 9
列出所有许可证 10
HTTP 方法和端点 10
使用说明 10
参数 11
响应 11
响应 12
响应 13
响应 13
错误 14
定义 14
检索许可证 19
HTTP 方法和端点 19
使用说明 19
参数 19
响应 19
响应 23
响应 23
响应 24
错误 25
定义 25
修改许可证 27
HTTP 方法和端点 27
使用说明 27
参数 27
请求正文 27

响应 . . . 30
响应 . . . 30
响应 . . . 31
响应 . . . 31
响应 . . . 32
错误 . . . 32
定义 . . . 33
删除许可证 . . . 35
HTTP 方法和端点 . . . 35
使用说明 . . . 35
参数 . . . 35
响应 . . . 35
响应 . . . 35
响应 . . . 36
响应 . . . 36
错误 . . . 37
定义 . . . 37

许可证

Astra API (ACC) 许可证端点

管理加载到 Astra Control 帐户中的许可证。

License API 表示加载到帐户中的许可证。

创建新许可证

创建一个新许可证，指定许可证 API 资源的所需值。使用此端点可有效管理环境中的许可需求。

HTTP 方法和端点

POST `/accounts/{account_id}/core/v1/licenses`

使用说明

指示要创建的 License API 资源的所需值。

参数

名称	Type	在	Required	Description
account_id	string	path	True	包含帐户资源的 ID • 示例：{{.Account}}

请求正文

指示要创建的 License API 资源的所需值。

名称	Type	Required	Description
type	string	True	资源的媒体类型。定义的值为： • "application/astra-license"
version	string	True	资源的版本。定义的值为： • "1.0"

名称	Type	Required	Description
分配	string	False	包含指示许可证分配对象的 ID 的 JSON 字符串。仅在分配许可证时存在。对于 Astra Control Center 许可证，此字段必须填写帐户 ID，并且必须与 POST URI 中指定的帐户 ID 匹配。定义的值为： • 符合 Astra 标识符架构
设备凭证 ID	string	False	包含与凭据 API 关联的 deviceCredential 标识符的 JSON 字符串。仅适用于 Astra Data Store 许可证。定义的值为： • 符合 Astra 标识符架构
licenseText	string	True	包含从 NetApp 许可证系统获取的许可证 JSON 的 base64 编码表示的 JSON 字符串。必须以原始字节级别顺序提供以进行签名验证。定义的值为： • 符合 base64 编码数据架构
元数据	"type_astra_metadata_update"	False	与资源关联的客户端和服务指定的元数据。定义的值为： • 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

▼ 请求示例

```
{
  "type": "application/astra-license",
  "version": "1.0",
  "allocation": "4185d09b-6687-4801-a832-c005a9d94f41",
  "licenseText": "ewogICAic3RhdHVzUmVzcCI6ewogMTYwNzAwIgp9"
}
```

响应

```
Status: 201, Returns the newly created license resource in the JSON
response body.
```

名称	Type	Required	Description
type	string	True	资源的媒体类型。定义的值为： • "application/astra-license"
version	string	True	资源的版本。定义的值为： • "1.0"
id	string	True	资源的全局唯一标识符。定义的值为： • 符合 UUIDv4 架构
分配	string	False	包含指示许可证分配对象的 ID 的 JSON 字符串。仅在分配许可证时存在。对于 Astra Control Center 许可证，此字段必须填写帐户 ID，并且必须与 POST URI 中指定的帐户 ID 匹配。定义的值为： • 符合 Astra 标识符架构
hostID	string	False	包含许可证绑定到的主机标识符的 JSON 字符串。仅在许可证处于主机锁定状态时存在。定义的值为： • 最小长度：1 • 最大长度：63
设备凭证 ID	string	False	包含与凭据 API 关联的 deviceCredential 标识符的 JSON 字符串。仅适用于 Astra Data Store 许可证。定义的值为： • 符合 Astra 标识符架构

名称	Type	Required	Description
isEvaluation	string	True	表示许可证是否为评估许可证的 JSON 字符串。定义的值为： • "true" • "false"
licenseProtocol	string	True	包含从许可证中提取的许可证协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseText	string	True	包含从 NetApp 许可证系统获取的许可证 JSON 的 base64 编码表示的 JSON 字符串。必须以原始字节级别顺序提供以进行签名验证。定义的值为： • 符合 base64 编码数据架构
validFromTimestamp	string	True	包含指示许可证何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
validUntilTimestamp	string	True	包含指示许可证何时不再有效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
产品	string	True	包含从许可证中提取的产品字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
productVersion	string	True	包含从许可证中提取的版本字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

名称	Type	Required	Description
产品 SN	string	True	包含从许可证中提取的序列号字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
features	string	True	包含从许可证中提取的包值的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
容量	string	True	包含从许可证中提取的容量的 JSON 字符串。
capacity2	string	True	包含从许可证中提取的辅助容量的 JSON 字符串。
插件	数组["插件"]	False	包含许可证中附加组件列表的 JSON 对象的 JSON 数组。附加组件是指在未来可能发生的许可证变更。
元数据	"type_astra_metadata_update"	True	与资源关联的客户端和服务指定的元数据。定义的值为： • 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

▼ 响应示例


```
{
  "type": "application/astra-license",
  "version": "1.0",
  "id": "6db53b32-8417-5d3c-8ef0-552fc602c78f",
  "allocation": "4185d09b-6687-4801-a832-c005a9d94f41",
  "hostID": "99132549-e0c2-4203-9d1e-598628b4ff9b",
  "isEvaluation": "false",
  "licenseProtocol": "ASTRASDS-ENT-SUBS",
  "licenseText": "ewogICAic3RhdHVzUmVzcCI6ewogMTYwNzAwIgp9",
  "validFromTimestamp": "2020-08-06T00:00:00.000000Z",
```

```
  "validUntilTimestamp": "2021-08-06T00:00:00.000000Z",
  "product": "Astra Control Center",
  "productVersion": "1.0",
  "productSN": "320000046",
  "features": "ASTRA-ENT-STD",
  "capacity": "4000",
  "capacity2": "0",
  "addons": [
    {
      "startDate": "2020-08-06T00:00:00.000000Z",
      "endDate": "2021-08-06T00:00:00.000000Z",
      "capacity": "4000",
      "licenseProtocol": "ASTRA-ENT-STD",
      "features": "p-nfs,dm-storage-policy-based-pvs-for-k8s,dm-spbm-
based-vvols-for-vms,dm-storage-efficiency,dm-data-at-rest-encryption"
    }
  ],
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}
```


响应

```
Status: 401, Unauthorized
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
```

```
}
```

响应

```
Status: 400, Bad request
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例

```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```

响应

```
Status: 409, Conflict
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	
invalidFields	数组["invalidFields"]	False	无效请求正文字段列表

▼ 响应示例

```
{
  "type": "https://astra.netapp.io/problems/10",
  "title": "JSON resource conflict",
```

```
  "detail": "The request body JSON contains a field that conflicts with an idempotent value.",
  "status": "409"
}
```


响应

```
Status: 403, Forbidden
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```


错误

```
Status: 404, Not found
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 错误响应示例


```
{
```

```
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't found.",
  "status": "404"
}
```

定义

▼ 查看定义

类型 Astra 标签

名称	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

与资源关联的客户端和服务指定的元数据。定义的值为：

- 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

名称	Type	Required	Description
labels	数组["类型 Astra 标签"]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
已创建	string	False	
modifiedBy	string	False	

插件

名称	Type	Required	Description
startDate	string	False	包含指示此附加组件何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
endDate	string	False	包含时间戳的 JSON 字符串，指示此附加组件何时不再有效。定义的值为： • 符合 ISO-8601 日期时间方案

名称	Type	Required	Description
features	string	False	包含与此加载项关联的功能列表的 JSON 字符串。定义的值为： • 架构未验证字符串内容。
容量	string	False	包含从加载项提取的容量的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseProtocol	string	False	包含与此附加组件关联的许可协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

invalidParams

名称	Type	Required	Description
name	string	True	无效查询参数的名称
原因	string	True	查询参数无效的原因

invalidFields

名称	Type	Required	Description
name	string	True	无效请求正文字段的名称
原因	string	True	请求正文字段无效的原因

列出所有许可证

检索集合中的所有许可证，返回匹配资源的 JSON 数组。您可以在请求中指定单个字段，以自定义为每个许可证返回的信息。

HTTP 方法和端点

GET `/accounts/{account_id}/core/v1/licenses`

使用说明

返回集合中所有匹配资源的 JSON 数组。如果在请求中指定了单个字段，则项目 JSON 数组将包含按指定顺序为每个匹配资源请求的字段。

参数

名称	Type	在	Required	Description
account_id	string	path	True	包含帐户资源的 ID • 示例：{{.Account}}
包括	string	查询	False	指示列出集合时应返回哪些字段。 URL 示例：“include=id”、“include=id,name” OpenAPI 示例：“id” • 示例：id,product
limit	string	查询	False	指示列出集合时要返回的最大资源数。 URL 示例：“limit=2” OpenAPI 示例：“2”
筛选器	string	查询	False	指示在列出集合时应匹配哪些字段以返回资源。 支持的运算符有： • eq - 相等 • lt/gt - 小于/大于 • lte/gte - 小于或等于/大于或等于 URL 示例：“filter=field%20eq%20%27value%27” OpenAPI 示例：“field eq 'value'”

响应

```
Status: 200, Returns a JSON array of all matching resources in the
collection. If individual fields are specified in the request, the items
JSON array will contain the fields requested for each matching resource in
the order specified.
```

名称	Type	Required	Description
个项	数组[任意一个："license_1.0_get_response_body"，数组[]]	True	
元数据	"type_astra_metadata_list"	True	

▼ 响应示例


```
{
  "type": "application/astra-licenses",
  "version": "1.0",
  "items": [
    [
      "c8aa757a-5318-489f-8134-a2a1a6956a3e",
      "Astra SDS"
    ],
    [
      "0f7cc03d-9afc-4192-af7f-b29ab24f8f2a",
      "Astra Enterprise"
    ]
  ],
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}
```


响应

```
Status: 401, Unauthorized
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例

```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```


响应

```
Status: 400, Bad request
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	
invalidParams	数组["invalidParams"]	False	无效查询参数列表

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```


响应

```
Status: 403, Forbidden
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	

名称	Type	Required	Description
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```


错误

```
Status: 404, Not found
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 错误响应示例


```
{
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't found.",
  "status": "404"
}
```


定义

▼ 查看定义

插件

名称	Type	Required	Description
startDate	string	False	包含指示此附加组件何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
endDate	string	False	包含时间戳的 JSON 字符串，指示此附加组件何时不再有效。定义的值为： • 符合 ISO-8601 日期时间方案
features	string	False	包含与此加载项关联的功能列表的 JSON 字符串。定义的值为： • 架构未验证字符串内容。
容量	string	False	包含从加载项提取的容量的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseProtocol	string	False	包含与此附加组件关联的许可协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

类型 Astra 标签

名称	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

与资源关联的客户端和服务指定的元数据。定义的值为：

- 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

名称	Type	Required	Description
labels	数组["类型 Astra 标签"]	False	

名称	Type	Required	Description
creationTimestamp	string	False	
modificationTimestamp	string	False	
已创建	string	False	
modifiedBy	string	False	

license_1.0_get_response_body

名称	Type	Required	Description
type	string	True	资源的媒体类型。定义的值为： • "application/astra-license"
version	string	True	资源的版本。定义的值为： • "1.0"
id	string	True	资源的全局唯一标识符。定义的值为： • 符合 UUIDv4 架构
分配	string	False	包含指示许可证分配对象的 ID 的 JSON 字符串。仅在分配许可证时存在。对于 Astra Control Center 许可证，此字段必须填写帐户 ID，并且必须与 POST URI 中指定的帐户 ID 匹配。定义的值为： • 符合 Astra 标识符架构
hostID	string	False	包含许可证绑定到的主机标识符的 JSON 字符串。仅在许可证处于主机锁定状态时存在。定义的值为： • 最小长度：1 • 最大长度：63

名称	Type	Required	Description
设备凭证 ID	string	False	包含与凭据 API 关联的 deviceCredential 标识符的 JSON 字符串。仅适用于 Astra Data Store 许可证。定义的值为： • 符合 Astra 标识符架构
isEvaluation	string	True	表示许可证是否为评估许可证的 JSON 字符串。定义的值为： • "true" • "false"
licenseProtocol	string	True	包含从许可证中提取的许可证协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseText	string	True	包含从 NetApp 许可证系统获取的许可证 JSON 的 base64 编码表示的 JSON 字符串。必须以原始字节级别顺序提供以进行签名验证。定义的值为： • 符合 base64 编码数据架构
validFromTimestamp	string	True	包含指示许可证何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
validUntilTimestamp	string	True	包含指示许可证何时不再有效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
产品	string	True	包含从许可证中提取的产品字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

名称	Type	Required	Description
productVersion	string	True	包含从许可证中提取的版本字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
产品 SN	string	True	包含从许可证中提取的序列号字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
features	string	True	包含从许可证中提取的包值的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
容量	string	True	包含从许可证中提取的容量的 JSON 字符串。
capacity2	string	True	包含从许可证中提取的辅助容量的 JSON 字符串。
插件	数组["插件"]	False	包含许可证中附加组件列表的 JSON 对象的 JSON 数组。附加组件是指在未来可能发生的许可证变更。
元数据	"type_astra_metadata_update"	True	与资源关联的客户端和服务指定的元数据。定义的值为： • 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

type_astra_metadata_list

名称	Type	Required	Description
继续	string	False	
count	数字	False	

名称	Type	Required	Description
labels	数组["类型 Astra 标签"]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
已创建	string	False	
modifiedBy	string	False	

invalidParams

名称	Type	Required	Description
name	string	True	无效查询参数的名称
原因	string	True	查询参数无效的原因

检索许可证

检索许可证，提供存储的许可证 API 资源的详细信息。使用此选项可检查当前许可证配置并确保符合许可要求。

HTTP 方法和端点

GET `/accounts/{account_id}/core/v1/licenses/{license_id}`

使用说明

返回 JSON 响应正文中存储的 License API 资源。

参数

名称	Type	在	Required	Description
account_id	string	path	True	包含帐户资源的 ID • 示例：{{.Account}}
license_id	string	path	True	要列出的许可证集合的 ID

响应

```
Status: 200, Returns the stored License API resource in the JSON response body.
```

名称	Type	Required	Description
type	string	True	资源的媒体类型。定义的值为： • "application/astra-license"
version	string	True	资源的版本。定义的值为： • "1.0"
id	string	True	资源的全局唯一标识符。定义的值为： • 符合 UUIDv4 架构
分配	string	False	包含指示许可证分配对象的 ID 的 JSON 字符串。仅在分配许可证时存在。对于 Astra Control Center 许可证，此字段必须填写帐户 ID，并且必须与 POST URI 中指定的帐户 ID 匹配。定义的值为： • 符合 Astra 标识符架构
hostID	string	False	包含许可证绑定到的主机标识符的 JSON 字符串。仅在许可证处于主机锁定状态时存在。定义的值为： • 最小长度：1 • 最大长度：63
设备凭证 ID	string	False	包含与凭据 API 关联的 deviceCredential 标识符的 JSON 字符串。仅适用于 Astra Data Store 许可证。定义的值为： • 符合 Astra 标识符架构
isEvaluation	string	True	表示许可证是否为评估许可证的 JSON 字符串。定义的值为： • "true" • "false"

名称	Type	Required	Description
licenseProtocol	string	True	包含从许可证中提取的许可证协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseText	string	True	包含从 NetApp 许可证系统获取的许可证 JSON 的 base64 编码表示的 JSON 字符串。必须以原始字节级别顺序提供以进行签名验证。定义的值为： • 符合 base64 编码数据架构
validFromTimestamp	string	True	包含指示许可证何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
validUntilTimestamp	string	True	包含指示许可证何时不再有效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
产品	string	True	包含从许可证中提取的产品字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
productVersion	string	True	包含从许可证中提取的版本字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
产品 SN	string	True	包含从许可证中提取的序列号字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

名称	Type	Required	Description
features	string	True	包含从许可证中提取的包值的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
容量	string	True	包含从许可证中提取的容量的 JSON 字符串。
capacity2	string	True	包含从许可证中提取的辅助容量的 JSON 字符串。
插件	数组["插件"]	False	包含许可证中附加组件列表的 JSON 对象的 JSON 数组。附加组件是指在未来可能发生的许可证变更。
元数据	"type_astra_metadata_update"	True	与资源关联的客户端和服务指定的元数据。定义的值为： • 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

▼ 响应示例


```
{
  "type": "application/astra-license",
  "version": "1.0",
  "id": "6db53b32-8417-5d3c-8ef0-552fc602c78f",
  "allocation": "4185d09b-6687-4801-a832-c005a9d94f41",
  "hostID": "99132549-e0c2-4203-9d1e-598628b4ff9b",
  "isEvaluation": "false",
  "licenseProtocol": "ASTRASDS-ENT-SUBS",
  "licenseText": "ewogICAic3RhdHVzUmVzcCI6ewogMTYwNzAwIgp9",
  "validFromTimestamp": "2020-08-06T00:00:00.000000Z",
  "validUntilTimestamp": "2021-08-06T00:00:00.000000Z",
  "product": "Astra Control Center",
  "productVersion": "1.0",
  "productSN": "320000046",
  "features": "ASTRA-ENT-STD",
  "capacity": "4000",
```

```
  "capacity2": "0",
  "addons": [
    {
      "startDate": "2020-08-06T00:00:00.000000Z",
      "endDate": "2021-08-06T00:00:00.000000Z",
      "capacity": "4000",
      "licenseProtocol": "ASTRA-ENT-STD",
      "features": "p-nfs,dm-storage-policy-based-pvs-for-k8s,dm-spbm-
based-vvols-for-vms,dm-storage-efficiency,dm-data-at-rest-encryption"
    }
  ],
  "metadata": {
    "labels": [],
    "creationTimestamp": "2022-10-06T20:58:16.305662Z",
    "modificationTimestamp": "2022-10-06T20:58:16.305662Z",
    "createdBy": "8f84cf09-8036-51e4-b579-bd30cb07b269"
  }
}
```


响应

```
Status: 401, Unauthorized
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```


响应

```
Status: 400, Bad request
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	
invalidParams	数组["invalidParams"]	False	无效查询参数列表

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```


响应

```
Status: 403, Forbidden
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```

错误

```
Status: 404, Not found
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 错误响应示例


```
{
  "type": "https://astra.netapp.io/problems/2",
  "title": "Collection not found",
  "detail": "The collection specified in the request URI wasn't found.",
  "status": "404"
}
```


定义

▼ 查看定义

插件

名称	Type	Required	Description
startDate	string	False	包含指示此附加组件何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
endDate	string	False	包含时间戳的 JSON 字符串，指示此附加组件何时不再有效。定义的值为： • 符合 ISO-8601 日期时间方案

名称	Type	Required	Description
features	string	False	包含与此加载项关联的功能列表的 JSON 字符串。定义的值为： • 架构未验证字符串内容。
容量	string	False	包含从加载项提取的容量的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseProtocol	string	False	包含与此附加组件关联的许可协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

类型 Astra 标签

名称	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

与资源关联的客户端和服务指定的元数据。定义的值为：

- 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

名称	Type	Required	Description
labels	数组["类型 Astra 标签"]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
已创建	string	False	
modifiedBy	string	False	

invalidParams

名称	Type	Required	Description
name	string	True	无效查询参数的名称
原因	string	True	查询参数无效的原因

修改许可证

通过使用提供的信息替换存储的数据，同时保留某些关键值，来修改许可证资源。您可以使用此端点高效地更新许可证授权。

HTTP 方法和端点

PUT `/accounts/{account_id}/core/v1/licenses/{license_id}`

使用说明

将整个存储资源替换为提供的 JSON 请求正文，同时保留不可由用户修改的键值。

副作用

- 重新计算授权以添加或合并通过 Entitlement API 可见的许可证授权。

参数

名称	Type	在	Required	Description
account_id	string	path	True	包含帐户资源的 ID • 示例：{{.Account}}
license_id	string	path	True	要列出的许可证集合的 ID

请求正文

将整个存储资源替换为提供的 JSON 请求正文，同时保留不可由用户修改的键值。

副作用

- 重新计算授权以添加或合并通过 Entitlement API 可见的许可证授权。

名称	Type	Required	Description
type	string	True	资源的媒体类型。定义的值为： • "application/astra-license"
version	string	True	资源的版本。定义的值为： • "1.0"

名称	Type	Required	Description
id	string	False	资源的全局唯一标识符。定义的值为： • 符合 UUIDv4 架构
分配	string	False	包含指示许可证分配对象的 ID 的 JSON 字符串。仅在分配许可证时存在。对于 Astra Control Center 许可证，此字段必须填写帐户 ID，并且必须与 POST URI 中指定的帐户 ID 匹配。定义的值为： • 符合 Astra 标识符架构
hostID	string	False	包含许可证绑定到的主机标识符的 JSON 字符串。仅在许可证处于主机锁定状态时存在。定义的值为： • 最小长度：1 • 最大长度：63
设备凭证 ID	string	False	包含与凭据 API 关联的 deviceCredential 标识符的 JSON 字符串。仅适用于 Astra Data Store 许可证。定义的值为： • 符合 Astra 标识符架构
isEvaluation	string	False	表示许可证是否为评估许可证的 JSON 字符串。定义的值为： • "true" • "false"
licenseProtocol	string	False	包含从许可证中提取的许可证协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

名称	Type	Required	Description
licenseText	string	False	包含从 NetApp 许可证系统获取的许可证 JSON 的 base64 编码表示的 JSON 字符串。必须以原始字节级别顺序提供以进行签名验证。定义的值为： • 符合 base64 编码数据架构
validFromTimestamp	string	False	包含指示许可证何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
validUntilTimestamp	string	False	包含指示许可证何时不再有效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
产品	string	False	包含从许可证中提取的产品字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
productVersion	string	False	包含从许可证中提取的版本字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
产品 SN	string	False	包含从许可证中提取的序列号字符串的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
features	string	False	包含从许可证中提取的包值的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
容量	string	False	包含从许可证中提取的容量的 JSON 字符串。

名称	Type	Required	Description
capacity2	string	False	包含从许可证中提取的辅助容量的 JSON 字符串。
插件	数组["插件"]	False	包含许可证中附加组件列表的 JSON 对象的 JSON 数组。附加组件是指在未来可能发生的许可证变更。
元数据	"type_astra_metadata_update"	False	与资源关联的客户端和服务指定的元数据。定义的值为： • 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

▼ 请求示例

```
{
  "type": "application/astra-license",
  "version": "1.0",
  "allocation": "4185d09b-6687-4801-a832-c005a9d94f41",
  "licenseText": "ewogICAic3RhdHVzUmVzcCI6ewogMTYwNzAwIgp9"
}
```

响应

```
Status: 204, Indicates if the license resource was updated.
```

响应

```
Status: 401, Unauthorized
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	

名称	Type	Required	Description
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```


响应

```
Status: 400, Bad request
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```


响应

```
Status: 409, Conflict
```

名称	Type	Required	Description
type	string	True	

名称	Type	Required	Description
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	
invalidFields	数组["invalidFields"]	False	无效请求正文字段列表

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/10",
  "title": "JSON resource conflict",
  "detail": "The request body JSON contains a field that conflicts with
an idempotent value.",
  "status": "409"
}
```


响应

```
Status: 403, Forbidden
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```


错误

```
Status: 404, Not found
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 错误响应示例


```
{
  "type": "https://astra.netapp.io/problems/1",
  "title": "Resource not found",
  "detail": "The resource specified in the request URI wasn't found.",
  "status": "404"
}
```


定义

▼ 查看定义

插件

名称	Type	Required	Description
startDate	string	False	包含指示此附加组件何时生效的时间戳的 JSON 字符串。定义的值为： • 符合 ISO-8601 日期时间方案
endDate	string	False	包含时间戳的 JSON 字符串，指示此附加组件何时不再有效。定义的值为： • 符合 ISO-8601 日期时间方案
features	string	False	包含与此加载项关联的功能列表的 JSON 字符串。定义的值为： • 架构未验证字符串内容。

名称	Type	Required	Description
容量	string	False	包含从加载项提取的容量的 JSON 字符串。定义的值为： • 架构未验证的字符串内容
licenseProtocol	string	False	包含与此附加组件关联的许可协议的 JSON 字符串。定义的值为： • 架构未验证的字符串内容

类型 Astra 标签

名称	Type	Required	Description
name	string	True	
value	string	True	

type_astra_metadata_update

与资源关联的客户端和服务指定的元数据。定义的值为：

- 符合 Astra 元数据架构如果在创建时未指定，则将创建没有标签的元数据对象。如果在更新时未指定，则元数据对象的标签、creationTimestamp 和 createdBy 将保留而不进行修改。

名称	Type	Required	Description
labels	数组["类型 Astra 标签"]	False	
creationTimestamp	string	False	
modificationTimestamp	string	False	
已创建	string	False	
modifiedBy	string	False	

invalidParams

名称	Type	Required	Description
name	string	True	无效查询参数的名称
原因	string	True	查询参数无效的原因

invalidFields

名称	Type	Required	Description
name	string	True	无效请求正文字段的名称

名称	Type	Required	Description
原因	string	True	请求正文字段无效的原因

删除许可证

从系统中删除指定的许可证。使用此端点可有效管理许可证授权。

HTTP 方法和端点

DELETE `/accounts/{account_id}/core/v1/licenses/{license_id}`

使用说明

删除指定的许可证 API 资源。无法删除自动安装的评估许可证。

副作用

- 重新计算授权以删除通过 Entitlement API 可见的许可证授权。如果自动安装的评估许可证在删除之前存在于此系统上，则会自动将其重新应用于系统。

参数

名称	Type	在	Required	Description
account_id	string	path	True	包含帐户资源的 ID • 示例：{{.Account}}
license_id	string	path	True	要列出的许可证集合的 ID

响应

```
Status: 204, Indicates if the license resource was deleted.
```

响应

```
Status: 401, Unauthorized
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	

名称	Type	Required	Description
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/3",
  "title": "Missing bearer token",
  "detail": "The request is missing the required bearer token.",
  "status": "401"
}
```


响应

```
Status: 400, Bad request
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	
invalidParams	数组["invalidParams"]	False	无效查询参数列表

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/5",
  "title": "Invalid query parameters",
  "detail": "The supplied query parameters are invalid.",
  "status": "400"
}
```


响应

```
Status: 403, Forbidden
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 响应示例


```
{
  "type": "https://astra.netapp.io/problems/11",
  "title": "Operation not permitted",
  "detail": "The requested operation isn't permitted.",
  "status": "403"
}
```


错误

```
Status: 404, Not found
```

名称	Type	Required	Description
type	string	True	
标题	string	True	
详细信息	string	True	
状态	string	True	
correlationID	string	False	

▼ 错误响应示例


```
{
  "type": "https://astra.netapp.io/problems/1",
  "title": "Resource not found",
  "detail": "The resource specified in the request URI wasn't found.",
  "status": "404"
}
```


定义

▼ 查看定义

invalidParams

名称	Type	Required	Description
name	string	True	无效查询参数的名称
原因	string	True	查询参数无效的原因

版权信息

版权所有 © 2026 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。