访问控制

基于角色的访问控制（RBAC）

跟许多其他服务一样，密钥管理服务通过实施策略文件中定义的策略规则来支持其APIs的保护。密钥管理服务在其配置文件 */etc/barbican/barbinca.conf* 中存储对策略JSON文件的引用，通常此文件被命名为 *policy.json* ，存储在 */etc/barbican/* 目录下。

每个密钥管理API调用在策略文件中都有一行，用于标识适用的访问级别：

```
API_NAME: RULE_STATEMENT or MATCH_STATEMENT
```

RULE_STATEMENT 可以是其他 *RULE_STATEMENT* 或 *MATCH_STATEMENT* ：

```
RULE_STATEMENT: RULE_STATEMENT or MATCH_STATEMENT
```

MATCH_STATEMENT 是一组必须在API的调用者提供的令牌与所讨论的API的参数或目标实体之间匹配的标识符。例如：

```
"secrets:post": "role:admin or role:creator"
```

通过POST请求创建一个新的机密，那么在你的令牌中，你必须拥有admin或creator角色。

默认策略

OpenStack中的策略引擎非常灵活，允许你针对特定的云执行自定义策略。密钥管理服务附带了一个示例 *policy.json* 文件，可帮助你快速开始。示例策略中定义了5种不同的角色：

密钥管理：服务管理
负责密钥管理服务的云管理员。可以访问所有的管理APIs比如项目配额。

管理
项目管理员。可以访问管理角色作用域下的项目的所有资源。

创建者
拥有此角色的用户可以创建新的资源，并且只能删除最初由其创建（拥有）的资源。拥有此角色的用户不能删除同一项目下其他用户创建的资源。还可以在项目范围内完全访问项目下已有的机密。

观察者
此角色用户允许访问现有资源，但不允许上传新的机密或删除已有机密。

审计

此角色用户只允许访问资源元数据，所以此用户无法解密机密。

Access Control List（ACL）API

在项目级别确定机密的所有权带来了一些限制。例如，用户没有简便的方法上传只有自己有权访问的机密，也没有简便的方法来授予用户仅访问单个机密的权限。

为解决上述问题，密钥管理服务提供了*访问控制列表API* 。完整信息，请参阅*ACL API用户指南*。