

Tencent Cloud Automation Tools

Product Introduction

Product Documentation



Copyright Notice

©2013-2019 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

Product Introduction

Overview

Strengths

Use Cases

Cloud Access Management

Use Limits

Product Introduction

Overview

Last updated : 2022-05-23 15:27:20

What is TencentCloud Automation Tools?

TencentCloud Automation Tools (TAT) is a native Ops deployment tool for Tencent Cloud CVM and Lighthouse instances. TAT provides an automated approach for batch execution of commands (Shell, PowerShell, Python, etc.) for common daily tasks, such as running automated Ops scripts, polling processes, installing/uninstalling software, updating applications, and installing patches, without the need of instance login and password.

Batch execution

TAT enables you to remotely manage large-scale instances in a secure and reliable manner. TAT can execute Shell, PowerShell and Python commands in batches without using jump servers. You can use it for the automatic operation of common management tasks.

Interactive session management

TAT provides a browser-based interactive Shell, which enables you to manage instances without the need to open inbound ports or manage SSH keys.

Public command library

Tencent Cloud provides commands for common instance management and Ops tasks. You can also configure custom parameters as needed. For more requirements, please [contact us](#).

Command Execution Status

When you execute a command on an instance, the instance-level command [execution status](#) include the following:

API return message	Status in console	Description
PENDING	Waiting to be delivered	The command is waiting to be delivered by the system.
DELIVERING	Delivering	The command is being delivered to the target instance

DELIVER_DELAYED	Delayed delivery	The command delivery is delayed.
DELIVER_FAILED	Delivery failed	Failed to deliver the command
RUNNING	Executing	The command is being executed.
TIMEOUT	Command timeout	The command failed to complete in the specified timed out period.
SUCCESS	Command completed	The command execution is completed. Note that it does not mean that the task is successful. You need to check the `Output` and `ExitCode` of the result to see whether it's successful.
TASK_TIMEOUT	Task execution timeout	The task is ended due to the command execution timeout.
FAILED	Command failed	Unable to execute the command, or the command execution failed.

Strengths

Last updated : 2022-05-23 15:27:20

Batch execution

TAT enables you to remotely manage large-scale instances in a secure and reliable manner without logging in to instances, or using jump servers or SSH.

Integration tools

TAT can be integrated to your applications by using APIs and SDK.

Querying history

The historical records of executing the operating system script commands by TAT can be queried.

Implementing automation

Create common operation tasks and save them as templates to improve the efficiency of daily tasks, such as editing the registry, managing users, and installing software and patches.

Use Cases

Last updated : 2022-05-23 15:27:20

TAT is applicable in the following scenarios:

- Upload and run automated Ops scripts
- Perform common operations
- Run existing scripts
- Batch install software or applications
- Require polling processes
- Install patches or security updates
- Modify the host name or user login password

Cloud Access Management

Last updated : 2022-05-23 15:27:20

Overview

[Cloud Access Management \(CAM\)](#) is a web-based Tencent Cloud service that helps you with the security management of access permissions for resources under your Tencent Cloud account. With CAM, you can create, manage, and terminate users or user groups, and can use identity and policy management to control the permissions other users have to use Tencent Cloud resources. Policies can be used to authorize or block the use of specified resources by users to complete specified tasks. When you use CAM, you can associate policies with a user or user group to perform permissions control.

TAT is connected with CAM for permission controlling.

Access Control Levels

TAT supports the access control by resources and tags.

- **Resource-level control:** Specify a policy to assign a sub-account with permissions to a single resource. For details, see [Creating Custom Policy](#).
- **Control by tags:** Add tags to resources for access control

Preset Policies

Preset policy	Permissions granted
QcloudTATReadOnlyAccess	TAT read-only permission
QcloudTATFullAccess	TAT read/write permission

Types of Manageable Resources

TAT supports resource-level authorization. You can grant a specified sub-account the API permission of a specified resource.

In CAM, the types of TAT resources that can be authorized are as follows:

Resource Type	Resource Description Method in Authorization Policy
Remote command-related	<code>qcs::tat:\$region:\$account:command/\$commandId</code>

APIs supporting action-level authorization include:

API name	Description	Resource
CreateCommand	Create a command	*

APIs supporting resource-level authorization include:

API name API description	Resource type	Resource (in six-segment format)
DeleteCommand Delete a command	Command	<code>qcs::tat:\$region:\$account:command/\$command</code>
DescribeAutomationAgents Query the agent running status	CVM instances, Lighthouse instances	<code>qcs::cvm:\$region:\$account:instance/\$instanceId</code> <code>qcs::lighthouse:\$region:\$account:instance/\$instanceId</code>
DescribeCommands Query a command	Command	<code>qcs::tat:\$region:\$account:command/\$command</code>
DescribeInvocations Query the execution result	Command	<code>qcs::tat:\$region:\$account:command/\$command</code>
DescribeInvocationTasks Query the execution tasks	Command, CVM instances, Lighthouse instances	<code>qcs::tat:\$region:\$account:command/\$command</code> <code>qcs::cvm:\$region:\$account:instance/\$instanceId</code> <code>qcs::lighthouse:\$region:\$account:instance/\$instanceId</code>
InvokeCommand Invoke a command	Command, CVM instances, Lighthouse instances	<code>qcs::tat:\$region:\$account:command/\$command</code> <code>qcs::cvm:\$region:\$account:instance/\$instanceId</code> <code>qcs::lighthouse:\$region:\$account:instance/\$instanceId</code>
ModifyCommand Modify a command	Command	<code>qcs::tat:\$region:\$account:command/\$command</code>

API name API description	Resource type	Resource (in six-segment format)
PreviewReplacedCommandContent Query the command after rendering	Command	<code>qcs::tat:\$region:\$account:command/\$command</code>
RunCommand Run a command	Command, CVM instances, Lighthouse instances	<code>qcs::tat:\$region:\$account:command/\$command</code> <code>qcs::cvm:\$region:\$account:instance/\$instance</code> <code>qcs::lighthouse:\$region:\$account:instance/\$instance</code>

Examples

Check the examples below to learn about how to control permissions by using CAM.

- [Allow a user to modify and delete commands](#)
- [Allow a user to check command details](#)
- [Allow a user to check command execution results](#)
- [Disallow a user from executing a specified command](#)
- [Disallow a user from executing any commands](#)
- [Disallow a user from executing any commands on a specified CVM](#)
- [Disallow a user from executing commands on any CVMs](#)
- [Disallow a user from executing any commands on a specified Lighthouse instance](#)
- [Disallow a user from executing commands on any Lighthouse instances](#)
- [Allow a user to execute the specified commands on the specified CVM](#)
- [Allow a user to execute the specified commands on the specified Lighthouse instance](#)
- [Disallow a user to check the command execution result on a CVM](#)
- [Disallow a user to check the command execution result on a Lighthouse instance](#)
- [Disallow a user to check the Agent status on a CVM](#)
- [Disallow a user to check the Agent status on a Lighthouse instance](#)

Note :

Guangzhou region is used for all the examples below. Replace `$account` with the Tencent Cloud root account of the user.

- Allow a user to modify and delete the command `cmd-xxxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "resource": [
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxx"
      ],
      "action": [
        "tat:ModifyCommand",
        "tat>DeleteCommand"
      ]
    }
  ]
}
```

- Allow a user to check the details of the command `cmd-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "resource": [
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxx"
      ],
      "action": [
        "tat:DescribeCommands"
      ]
    }
  ]
}
```

- Allow a user to check the result of the command `cmd-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "resource": [
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxx"
      ],

```

```
"action": [  
  "tat:DescribeInvocations",  
  "tat:DescribeInvocationTasks"  
]  
}  
]
```

- Disallow a user from executing the command `cmd-xxxxxxxx`

```
{  
  "version": "2.0",  
  "statement": [  
    {  
      "effect": "deny",  
      "resource": [  
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxxx"  
      ],  
      "action": [  
        "tat:InvokeCommands"  
      ]  
    }  
  ]  
}
```

- Disallow a user from executing any commands

```
{  
  "version": "2.0",  
  "statement": [  
    {  
      "effect": "deny",  
      "resource": [  
        "qcs::tat:ap-guangzhou:$account:command/*"  
      ],  
      "action": [  
        "tat:InvokeCommand",  
        "tat:RunCommand"  
      ]  
    }  
  ]  
}
```

- Disallow a user from executing any commands on the CVM `ins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::cvm:ap-guangzhou:$account:instance/ins-xxxxxxx"
      ],
      "action": [
        "tat:InvokeCommand",
        "tat:RunCommand"
      ]
    }
  ]
}
```

- Disallow a user from executing commands on any CVMs

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::cvm:ap-guangzhou:$account:instance/*"
      ],
      "action": [
        "tat:InvokeCommand",
        "tat:RunCommand"
      ]
    }
  ]
}
```

- Disallow a user from executing any commands on the Lighthouse instance `lhins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
```

```
"resource": [
  "qcs::lighthouse:ap-guangzhou:$account:instance/lhins-xxxxxxx"
],
"action": [
  "tat:InvokeCommand",
  "tat:RunCommand"
]
}
]
}
```

- Disallow a user from executing commands on any Lighthouse instances

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::lighthouse:ap-guangzhou:$account:instance/*"
      ],
      "action": [
        "tat:InvokeCommand",
        "tat:RunCommand"
      ]
    }
  ]
}
```

- Allow a user to execute the command `cmd-xxxxxxx` or `cmd-yyyyyyy` on the CVM `ins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "allow",
      "resource": [
        "qcs::cvm:ap-guangzhou:$account:instance/ins-xxxxxxx",
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxx",
        "qcs::tat:ap-guangzhou:$account:command/cmd-yyyyyyy"
      ],
      "action": [
        "tat:InvokeCommand"
      ]
    }
  ]
}
```

```
}  
]  
}
```

- Allow a user to execute the command `cmd-xxxxxxx` or `cmd-yyyyyyy` on the Lighthouse instance

```
lhins-xxxxxxx
```

```
{  
  "version": "2.0",  
  "statement": [  
    {  
      "effect": "allow",  
      "resource": [  
        "qcs::lighthouse:ap-guangzhou:$account:instance/lhins-xxxxxxx",  
        "qcs::tat:ap-guangzhou:$account:command/cmd-xxxxxxx",  
        "qcs::tat:ap-guangzhou:$account:command/cmd-yyyyyyy"  
      ],  
      "action": [  
        "tat:InvokeCommand"  
      ]  
    }  
  ]  
}
```

- Disallow a user from checking the command execution result on the CVM `ins-xxxxxxx`

```
{  
  "version": "2.0",  
  "statement": [  
    {  
      "effect": "deny",  
      "resource": [  
        "qcs::cvm:ap-guangzhou:$account:instance/ins-xxxxxxx"  
      ],  
      "action": [  
        "tat:DescribeInvocationTasks"  
      ]  
    }  
  ]  
}
```

- Disallow a user from checking the command execution result on the Lighthouse instance `lhins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::lighthouse:ap-guangzhou:$account:instance/lhins-xxxxxxx"
      ],
      "action": [
        "tat:DescribeInvocationTasks"
      ]
    }
  ]
}
```

- Disallow a user from checking the Agent status on the CVM `ins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::cvm:ap-guangzhou:$account:instance/ins-xxxxxxx"
      ],
      "action": [
        "tat:DescribeAutomationAgentStatus"
      ]
    }
  ]
}
```

- Disallow a user from checking the Agent status on the Lighthouse instance `lhins-xxxxxxx`

```
{
  "version": "2.0",
  "statement": [
    {
      "effect": "deny",
      "resource": [
        "qcs::lighthouse:ap-guangzhou:$account:instance/lhins-xxxxxxx"
      ],
      "action": [
```



```
"tat:DescribeAutomationAgentStatus"  
]  
}  
]  
}
```

Use Limits

Last updated : 2022-05-23 15:27:20

Use Limits

- TAT agent is installed as a built-in tool for Linux-based Lighthouse instances created after Dec 15, 2020. For instances created before this time, you need to install it manually, as instructed in [Installing TAT Agent](#).
- TAT agent is only available for VPC-based instances.