Triton Management Service Control

Table of contents

> ⚠ **Attention**
>
> NVIDIA Triton Management Service (TMS) will reach the end of life on July 31, 2024. The version 1.4.0 is the last release.

Triton Management Service Control (`tmsctl`) is a command line utility for interacting with Triton Management Service (TMS). It provides commands for interacting with TMS, creating and managing leases, and managing service configuration.

Download tmsctl from NGC

This document includes a reference of all commands, as well as an explanation of some configuration options that can be used to control the behavior of `tmsctl`.

Command Reference

- lease name
  - lease name create
  - lease name delete
  - lease name list
- pool
  - pool create
  - pool delete
  - pool list
  - pool status
- target
  - target add
  - target list
  - target remove
  - target set

Common Options

Many commands share a few common options. These are documented here.

`(--target|-t):<target>`

Specify the instances of TMS on which to perform the operation.

Valid targets are URLs beginning with `http://` or `https://`, or the name of a named instance (see the target command).

Examples:

- `--target https://www.example.com:30345`: will connect to TMS at the specified URL.

- `--target my_tms`: will connect to a TMS instance named `my_tms` previously specified via tmsctl target add.

Unless a default target is specified via tsmctl target commands, all commands require the `--target` option.

`(--porcelain|-z)`

Formats output in an easy-to-parse format for scripts; avoids fancy formatting for human readers.

Porcelain output does not attempt to colorize output or insert unnecessary whitespace to improve readability.

This output is not guaranteed to be stable between releases.

Lease

The `tmsctl lease` commands allow you to perform operations on leases, such as creating, renewing, and releasing them.

Lease Create

`tmsctl lease create [(--target|-t):<target>] [(--porcelain|-z)] (--model|-m):<model> [{Duration Options}] [{Automatic Renewal Options}] [{Autoscaling Options}] [{Triton Options}]`

`tmsctl lease create [(--target|-t):<target>] [(--porcelain|-z)] (--model|-m):<model> (--triton-pool|-p):<name> --quota:<quota> [{Duration Options}] [{Automatic Renewal Options}]`

Connects to `<target>` and creates a lease for Triton Inference Server to serve one or more `<model>`.

Provides a `<lease-id>` for the newly created lease when successful.

An error code will be returned when no default `<target>` exists and `(--target|-t)` has not been specified.

To learn about how to package models, please see the model repository documentation.

Options

`--model|-m:<model>`

`<model>` is a comma-separated list of `<name>=<value>` pairs describing a model.

This option can be included multiple times, once for each unique model required. All models in a lease will be loaded and provided by a single Triton Inference Server. If the set of models is too large or requires too many resources, Triton may fail to load them. In the event of a failure an error will be returned and the lease made invalid.

The allowed `<name>=<value>` pairs are:

- `name` (required): the name of the model. Must match the name expected by Triton.
- `uri` (required): the URI from which to get the model.
- `count` (optional, default=0): the number of instances of the model to load, or 0 to use the model's default count.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Automatic Renewal Options

`--auto-renew`

Makes new new lease eligible for automatic renewal. Auto-renewal rules are determined by the server.

`--auto-renew-activity-window`

The time window during which a lease must be used before expiration for it to be automatically renewed. Durations must be specified in `(<hours>h)(<minutes>m)(<seconds>s)` format (e.g. `1h30m15s`, `1h30m`, `1h`, `1m30s`).

Autoscaling Options

Options related to automatically scaling the number of instances of a lease.

Note: Autoscaling and using pooled Triton instances are mutually exclusive. If any of these options are used at the same time pool options are used, an error is reported.

`--enable-autoscaling`

Enable autoscaling for this lease. This is automatically turned on if any of the other options related to autoscaling are set.

`--autoscaling-max-replicas`

Set the maximum number of replicas when autoscaling. Valid values are any positive integer. Implies `--enable-autoscaling` when provided.

`--autoscaling-min-replicas`

Set the minimum number of replicas when autoscaling. Valid values are any non-negative integer. Implies `--enable-autoscaling` when provided.

`--autoscaling-metric-cpu-utilization`

Set the state of autoscaling based on CPU utilization. Valid values are `enable`, `disable`, and `server-default` (default).

`--autoscaling-metric-cpu-utilization-threshold`

Set the threshold for autoscaling based on CPU utilization. The value must be a number between 0 (exclusive) and 100 (inclusive).

`--autoscaling-metric-gpu-utilization`

Set the state of autoscaling based on CPU utilization. Valid values are `enable`, `disable`, and `server-default` (default).

`--autoscaling-metric-gpu-utilization-threshold`

Set the threshold for autoscaling based on GPU utilization. The value must be a number between 0 (exclusive) and 100 (inclusive).

`--autoscaling-metric-queue-time`

Set the state of autoscaling based on queue time. Valid values are `enable`, `disable`, and `server-default` (default).

`--autoscaling-metric-queue-time-threshold`

Set the threshold for autoscaling based on queue time. Durations must be specified in `(<hours>h)(<minutes>m)(<seconds>s)` format (e.g. `1h30m15s`, `1h30m`, `1h`, `1m30s`).

`--autoscaling-metric-queue-time-percentage`

Set the state of autoscaling based on the percentage of time inference requests spend in the queue. Valid values are `enable`, `disable`, and `server-default` (default).

`--autoscaling-metric-queue-time-percentage-threshold`

Set the threshold for autoscaling based on the percentage of time inference requests spend in the queue. The value must be a number between 0 (exclusive) and 100 (inclusive).

Duration Options

`--duration`

The initial duration of the lease. Durations must be specified in `(<hours>h)(<minutes>m)(<seconds>s)` format (e.g. `1h30m15s`, `1h30m`, `1h`, `1m30s`).

`--renewal-duration`

The duration for which the lease renews when renewed. Durations must be specified in `(<hours>h)(<minutes>m)(<seconds>s)` format (e.g. `1h30m15s`, `1h30m`, `1h`, `1m30s`).

Triton Options

Options related to how the Triton instance is created.

Note: Specifying Triton options and using pooled Triton instances are mutually exclusive. If any of these options are used at the same time pool options are used, an error is reported.

`--triton-image|-i`

Specifies the Triton container image to be used to deployment the lease. `<triton-image>` must be in the allowed list of Triton container images, managed by the TMS administrator.

`--triton-resources`

Specifies the hardware resources to allocate to the Triton server for this lease.

Expected format:
`cpu=<count>,gpu=<count>,repository-size=<memory>,system-memory=<memory>,shared-memory=<memory>`
, where `<count>` is expected to be a positive integer, and `<memory>` is expected to be a positive number followed by Ki, Mi, or Gi to indicate the amount of memory.

When not provided and a pool is not specified, server-configured defaults are used.

Triton Pool Options

Options related to the creation of Triton pools.

Note: Specifying pool options is mutually exclusive with autoscaling options and Triton options. If any of these options are used at the same time as those, an error is reported.

`--triton-pool|-p`

Specifies the Triton Pool, by name, that the lease should be deployed into. Must be specified along with the `--quota` option.

`--quota`

Specifies the amount of available quota the lease will consume from a single instance of Triton in the target pool. Must be specified along with `(--triton-pool|-p):<name>`. Must be greater than zero.

Lease List

`tmsctl lease list [(--target|-t):<target>] [(--porcelain|-z)]`

`tmsctl leases [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and list all active and pending leases.

By default, a summary of each lease will be listed. Adding the `--verbose` flag will increase the amount of output.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Lease Release

`tmsctl lease release <lease-id> [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and release lease `<lease-id>`.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

`<lease-id>`

Unique identifier of a lease.

Lease Renew

`tmsctl lease renew <lease-id> [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and renew lease `<lease-id>`.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

`<lease-id>`

Unique identifier of a lease.

Lease Status

tmsctl lease status <lease-id> [(--target|-t):<target>] [(--porcelain|-z)]

Connects to <target> to get the current status of a lease.

When no default <target> exists and (--target|-t) has not been specified, an error will occur.

Options

--porcelain|-z

Formats output in an easy-to-parse format for scripts. See Common Options.

--target|-t

Determines the Triton Management Service to connect to. See Common Options.

<lease-id>

Unique identifier of a lease.

Lease Name

Provides functionality for managing names associated with leases.

Lease Name Create

tmsctl lease name create (--name:)<lease-name> (--lease:)<lease-id> [(--target|-t):<target>] [(--porcelain|-z)]

Creates a new <lease-name> for an existing lease <lease-id> .

When no default <target> exists and (--target|-t) has not been specified, an error will occur.

Options

`--lease`

The unique identifier of a lease to which the name should refer. May be be specified without the `--lease` flag if the name is specified first.

`--name`

The name of a lease to create. May be be specified without the `--name` flag if it is the first positional argument.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

`<lease-id>`

The unique identifier of a lease to which the name should refer.

`<lease-name>`

The name of a lease to create.

Lease Name Delete

`tmsctl lease name delete (--name:)<lease-name> ((--lease:)<lease-id>|--force) [(--target|-t):<target>] [(--porcelain|-z)]`

Deletes an existing `<lease-name>` for a specified lease `<lease-id>`.

The lease `<lease-id>` associated with the `<lease-name>` does not have to be specified if the `--force` flag is provided.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--force`

Delete the name regardless of what lease it currently refers to.

`--lease`

The unique identifier of a lease to which the name should refer. If this does not match what the name actually refers to, an error occurs. May be be specified without the `--lease` flag if the name is specified first.

`--name`

The name of a lease to delete. May be be specified without the `--name` flag if it is the first positional argument.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

`<lease-id>`

The unique identifier of a lease to which the name should refer. If this does not match what the name actually refers to, an error occurs. May be be specified without the `--lease` flag if the name is specified first.

`<lease-name>`

The name of a lease to delete.

Lease Name List

`tmsctl lease name list [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and lists all lease names associated with existing leases.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

`tmsctl lease name move (--name:)<lease-name> ((--source-lease:)<source-lease-id>|--force) (--target-lease:)<target-lease-id> [(--target|-t):<target>] [(--porcelain|-z)]`

Moves a `<lease-name>` from one lease `<source-lease-id>` to another `<target-lease-id>`.

The lease `<source-lease-id>` associated with the `<lease-name>` does not have to be specified if the `--force` flag is provided.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--force`

Move the name regardless of what lease it currently refers to.

`--name`

The name of a lease to move. May be be specified without the `--name` flag if it is the first positional argument.

`--source-lease` The unique identifier of the lease to which the name should currently refer. If this does not match what the name actually refers to, an error occurs. May be be specified without the `--source-lease` flag if the name is specified first.

`--target-lease` The unique identifier of the new lease to which the name should refer. May be be specified without the `--target-lease` flag if the name and source lease are specified first.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Pool

Commands for managing Triton pools.

Pool Create

tmsctl pool create <name> (--instance-quota|-q):<quota> [--disable-backend-uniqueness] [(--instances|-c):<count>] [--triton-resources:<resources>] [--triton-container:<image-name>] [(--target|-t):<target>][(--porcelain|-z)]

An error code will be returned when no default exists and (–target|-t) has not been specified.

Options

`--disable-backend-uniqueness`

Disables Triton backend uniqueness enforcement.

By default, Triton pools segregates Triton instances by the Triton backend(s) used by models loaded. Disabling this segregation enables leases with models with differing

Triton backend requirements to be collocated. The mixing of Triton backends can lead to runtime out-of-memory errors.

`--instance-quota|-q`

Specifies the maximum allocatable quota per Triton instance in the pool as an integer. This value limits the number of leases which can be assigned to a single Triton instance in the pool based. Leases deployed into the pool must specify the amount of quota the consume and will only be place on Triton instances with sufficient remaining quota. Specifying a quota larger than is physically available can lead to resource exhaustion errors and server crashes. When the provided value is outside the configured server limits, the pool creation request will fail. Value is required and must be a value greater than zero.

`--instances|-c`

Specifies the minimum and maximum number of Triton instances allowed to exist in the pool. Expected format: `&lt;minimum&gt;,&lt;maximum&gt;` where `&lt;minimum&gt;` and/or `&lt;maximum&gt;` can be replaced with `*` to use the configured server default value. When not provided the configured server defaults will be used.

`--triton-container`

Specifies the Triton container image to be used for all Triton instances in the pool. `&lt;image-name&gt;` must be in the allowed list of Triton container images managed by the TMS administrator.

`--triton-resources`

Specifies the hardware resources to allocate to the Triton server for this lease.

Expected format:
`cpu=&lt;count&gt;,gpu=&lt;count&gt;,repository-size=&lt;memory&gt;,system-memory=&lt;memory&gt;,shared-memory=&lt;memory&gt;`
, where `&lt;count&gt;` is expected to be a positive integer, and `&lt;memory&gt;` is expected to be a positive number followed by Ki, Mi, or Gi to indicate the amount of memory.

When not provided, configured server defaults are used.

`<name>`

Unique name of the pool used to reference the pool when creating leases which make use of the pool. Pool names can contain only alphanumeric, hyphen, and underscore characters. Pool names are case insensitive and must not conflict with any existing, active pool. Value is required, maximum allowed size is 512 characters, and minimum size is 8 characters.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Pool Delete

`tmsctl pool delete (<triton-pool-name>|<triton-pool-name>) [(--target|-t):<target>] [(--porcelain|-z)]`

An error code will be returned when no default exists and (–target|-t) has not been specified.

Options

`<triton-pool-name>` Unique identifier of the pool. Represented as 32 character UUID.

`<triton-pool-name>` Unique name of the pool.

Pool names can contain only alphanumeric, hyphen, and underscore characters. Pool names are case insensitive. Maximum allowed size is 1024 bytes.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Pool List

tmsctl pool list [(--verbose|-v)] [(--target|-t):<target>] [(--porcelain|-z)]

An error code will be returned when no default exists and (–target|-t) has not been specified.

Options

--verbose|-v

Whether to produce more verbose details about the pools.

--porcelain|-z

Formats output in an easy-to-parse format for scripts. See Common Options.

--target|-t

Determines the Triton Management Service to connect to. See Common Options.

Pool Status

tmsctl pool status (<triton-pool-name>|<triton-pool-id>) [(--verbose|-v)] [(--target|-t):<target>] [(--porcelain|-z)]

An error code will be returned when no default exists and (–target|-t) has not been specified.

Options

--verbose|-v

Whether to produce more verbose details about the pools.

<triton-pool-id>

Unique identifier of the pool. Represented as 32 character UUID.

`<triton-pool-name>`

Unique name of the pool. Pool names can contain only alphanumeric, hyphen, and underscore characters. Pool names are case insensitive. Maximum allowed size is 1024 bytes.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Allowlist

Allowlist Add

`tmsctl allowlist add <image> [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and adds a Triton container image to the Triton allowlist.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`<image>`

Container image to add the list of allowed Triton container images.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Allowlist List

`tmsctl allowlist list [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and lists the Triton container images new leases are allowed to be created with.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Allowlist Remove

`tmsctl allowlist rm <image> [(--target|-t):<target>] [(--porcelain|-z)]`

Connects to `<target>` and removes a Triton container image from the Triton allow-list.

When no default `<target>` exists and (`--target|-t`) has not been specified, an error will occur.

Options

`<image>`

Container image to remove from the list of allowed Triton container images.

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

`--target|-t`

Determines the Triton Management Service to connect to. See Common Options.

Target

Target Add

`tmsctl target add [--force] [--set-default] <name> <url>`

Adds a new `<target>` to the set of configured Triton Management Services.

`<url>` is required to be prefixed with “http://” or “https://”.

When `<target>` already exists in the list of configured Triton Management Services, an error will occur unless `--force` is specified.

Options

`--force`

Allows for the replacement of an existing configured Triton Management Service, when specified, with `<target>` and `<url>`.

`--set-default`

Sets as default target for future commands which require a connection to Triton Management Service.

Target List

`tmsctl target list [(--porcelain|-z)]` **`tmsctl targets [(--porcelain|-z)]`**

Lists all configured Triton Management Server targets.

Options

`--porcelain|-z`

Formats output in an easy-to-parse format for scripts. See Common Options.

Target Remove

`tmsctl target rm [--force] <name>`

Removes `<target>` from the set of configured Triton Management Services.

If the target is the default, it is not removed unless the `--force` flag is used.

Options

`--force`

Removes `<target>` regardless if it has been set to default or not.

Target Set

`tmsctl target set <name>`

`tmsctl target <target>`

Sets `<target>` as the default target for future commands which require a connection to Triton Management Service.

`<target>` must have already been added to list of possible targets.

see `tmsctl target add` for additional details.

Configuring tmsctl

On startup, `tmsctl` will read a configuration if name `.tmsctlconfig` from the user's home directory. This file contains information about any named targets (see the target command) as well as options to configure the output of `tmsctl`.

The format of `.tmsctlconfig` is not guaranteed to be stable and this point and may change in the future. For now, it is a JSON file.

Configuring Output Colors

By default, `tmsctl` outputs everything in the default colors of the console. This can be changed by adding an entry named `"console"` at the top level of the configuration file and setting is `"enable-colors"` property to `"true"`. The example configuration below will tell `tmsctl` to enable colors with its default color scheme:

```
{
"console": {
"enable-colors": "true"
}
}
```

If the default `tmsctl` color scheme does not work well with your preferred terminal settings, you can customize the set of colors that `tmsctl` will use. When colors are enabled, `tmsctl` will read the additional properties from the `"console"` object to control text color:

- `"color"`: used for most output.
- `"emphasis"`: used for lines that add emphasis (e.g. lease IDs in `tmsctl lease create`).
- `"error"`: used for errors
- `"header"`: used for header lines.
- `"understated"`: used for output that can often be ignored.
- `"warning"`: used for warnings.

In addition to the above, some option can have `"-back"` added to it to control the background color of the corresponding entry. The options that support `"-back"` are `"emphasis"`, `"error"`, `"warining"` and `"understated`”.

Allowed values for colors are those listed by the .NET class (ConsoleColor) [https://learn.microsoft.com/en-us/dotnet/api/system.consolecolor]. Colors must be provided in lower case, with words separated by a `-`. For example, to use `ConsoleColor.DarkGreen`, you would specify `"dark-green"` in the configuration file.