

NAME

containerd-config.toml – configuration file for containerd

SYNOPSIS

The **config.toml** file is a configuration file for the containerd daemon. The file must be placed at **/etc/containerd/config.toml** or specified with the **--config** option of **containerd** to be used by the daemon. If the file does not exist at the appropriate location or is not provided via the **--config** option containerd uses its default configuration settings, which can be displayed with the **containerd config(1)** command.

DESCRIPTION

The TOML file used to configure the containerd daemon settings has a short list of global settings followed by a series of sections for specific areas of daemon configuration. There is also a section for **plugins** that allows each containerd plugin to have an area for plugin-specific configuration and settings.

FORMAT**version**

The version field in the config file specifies the config's version. If no version number is specified inside the config file then it is assumed to be a version 1 config and parsed as such. Please use version = 2 to enable version 2 config as version 1 has been deprecated.

root The root directory for containerd metadata. (Default: "/var/lib/containerd")

state The state directory for containerd (Default: "/run/containerd")

plugin_dir

The directory for dynamic plugins to be stored

[grpc] Section for gRPC socket listener settings. Contains the following properties:

- **address** (Default: "/run/containerd/containerd.sock")
- **tcp_address**
- **tcp_tls_cert**
- **tcp_tls_key**
- **uid** (Default: 0)
- **gid** (Default: 0)
- **max_recv_message_size**
- **max_send_message_size**

[ttrpc]

Section for TTRPC settings. Contains properties:

- **address** (Default: "")
- **uid** (Default: 0)
- **gid** (Default: 0)

[debug]

Section to enable and configure a debug socket listener. Contains four properties:

- **address** (Default: "/run/containerd/debug.sock")
- **uid** (Default: 0)
- **gid** (Default: 0)
- **level** (Default: "info") sets the debug log level. Supported levels are: "trace", "debug", "info", "warn", "error", "fatal", "panic"
- **format** (Default: "text") sets log format. Supported formats are "text" and "json"

[metrics]

Section to enable and configure a metrics listener. Contains two properties:

- **address** (Default: "") Metrics endpoint does not listen by default
- **grpc_histogram** (Default: false) Turn on or off gRPC histogram metrics

disabled_plugins

Disabled plugins are IDs of plugins to disable. Disabled plugins won't be initialized and started.

required_plugins

Required plugins are IDs of required plugins. Containerd exits if any required plugin doesn't exist or fails to be initialized or started.

[plugins]

The plugins section contains configuration options exposed from installed plugins. The following plugins are enabled by default and their settings are shown below. Plugins that are not enabled by default will provide their own configuration values documentation.

- **[plugins."io.containerd.monitor.v1.cgroups"]** has one option **no_prometheus** (Default: **false**)
- **[plugins."io.containerd.service.v1.diff-service"]** has one option **default**, a list by default set to **["walking"]**
- **[plugins."io.containerd.gc.v1.scheduler"]** has several options that perform advanced tuning for the scheduler:
 - **pause_threshold** is the maximum amount of time GC should be scheduled (Default: **0.02**),
 - **deletion_threshold** guarantees GC is scheduled after n number of deletions (Default: **0** [not triggered]),
 - **mutation_threshold** guarantees GC is scheduled after n number of database mutations (Default: **100**),
 - **schedule_delay** defines the delay after trigger event before scheduling a GC (Default **"0ms"** [immediate]),
 - **startup_delay** defines the delay after startup before scheduling a GC (Default **"100ms"**)
- **[plugins."io.containerd.runtime.v2.task"]** specifies options for configuring the runtime shim:
 - **platforms** specifies the list of supported platforms
 - **sched_core** Core scheduling is a feature that allows only trusted tasks to run concurrently on cpus sharing compute resources (eg: hyperthreads on a core). (Default: **false**)

- **[plugins."io.containerd.service.v1.tasks-service"]** has performance options:
 - **blockio_config_file** (Linux only) specifies path to blockio class definitions (Default: ""). Controls I/O scheduler priority and bandwidth throttling. See blockio configuration (<https://github.com/intel/goresctrl/blob/main/doc/blockio.md#configuration>) for details of the file format.
 - **rdt_config_file** (Linux only) specifies path to a configuration used for configuring RDT (Default: ""). Enables support for Intel RDT, a technology for cache and memory bandwidth management. See RDT configuration (<https://github.com/intel/goresctrl/blob/main/doc/rdt.md#configuration>) for details of the file format.
- **[plugins."io.containerd.grpc.v1.cri".containerd]** contains options for the CRI plugin, and child nodes for CRI options:
 - **default_runtime_name** (Default: "runc") specifies the default runtime name
- **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes]** one or more container runtimes, each with a unique name
- **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.]** a runtime named **<runtime>**
- **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes..options]** options for the named **<runtime>**, most important:
 - **BinaryName** specifies the path to the actual runtime to be invoked by the shim, e.g. **"/usr/bin/runc"**

oom_score

The out of memory (OOM) score applied to the containerd daemon process (Default: 0)

[cgroup]

Section for Linux cgroup specific settings

- **path** (Default: "") Specify a custom cgroup path for created containers

[proxy_plugins]

Proxy plugins configures plugins which are communicated to over gRPC

- **type** (Default: "")
- **address** (Default: "")

timeouts

Timeouts specified as a duration

imports

Imports is a list of additional configuration files to include. This allows to split the main configuration file and keep some sections separately (for example vendors may keep a custom runtime configuration in a separate file without modifying the main **config.toml**). Imported files will overwrite simple fields like **int** or **string** (if not empty) and will append **array** and **map** fields. Imported files are also versioned, and the version can't be higher than the main config.

stream_processors

- **accepts** (Default: "[]") Accepts specific media-types
- **returns** (Default: "") Returns the media-type

- **path** (Default: "") Path or name of the binary
- **args** (Default: []) Args to the binary

EXAMPLES

Complete Configuration

The following is a complete **config.toml** default configuration example:

```
version = 2

root = "/var/lib/containerd"
state = "/run/containerd"
oom_score = 0
imports = ["/etc/containerd/runtime_*.toml", "./debug.toml"]

[grpc]
  address = "/run/containerd/containerd.sock"
  uid = 0
  gid = 0

[debug]
  address = "/run/containerd/debug.sock"
  uid = 0
  gid = 0
  level = "info"

[metrics]
  address = ""
  grpc_histogram = false

[cgroup]
  path = ""

[plugins]
  [plugins."io.containerd.monitor.v1.cgroups"]
    no_prometheus = false
  [plugins."io.containerd.service.v1.diff-service"]
    default = ["walking"]
  [plugins."io.containerd.gc.v1.scheduler"]
    pause_threshold = 0.02
    deletion_threshold = 0
    mutation_threshold = 100
    schedule_delay = 0
    startup_delay = "100ms"
  [plugins."io.containerd.runtime.v2.task"]
    platforms = ["linux/amd64"]
    sched_core = true
  [plugins."io.containerd.service.v1.tasks-service"]
    blockio_config_file = ""
    rdt_config_file = ""
```

Multiple Runtimes

The following is an example partial configuration with two runtimes:

```

[plugins]

[plugins."io.containerd.grpc.v1.cri"]

[plugins."io.containerd.grpc.v1.cri".containerd]
  default_runtime_name = "runc"

[plugins."io.containerd.grpc.v1.cri".containerd.runtimes]
[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.runc]
  privileged_without_host_devices = false
  runtime_type = "io.containerd.runc.v2"

[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.runc.options]
  BinaryName = "/usr/bin/runc"

[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.other]
  privileged_without_host_devices = false
  runtime_type = "io.containerd.runc.v2"

[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.other.options]
  BinaryName = "/usr/bin/path-to-runtime"

```

The above creates two named runtime configurations - named **runc** and **other** - and sets the default runtime to **runc**. The above are used *solely* for runtimes invoked via CRI. To use the non-default "other" runtime in this example, a spec will include the runtime handler named "other" to specify the desire to use the named runtime config.

The CRI specification includes a <https://github.com/kubernetes/cri-api/blob/de5f1318aede866435308f39cb432618a15f104e/pkg/apis/runtime/v1/api.proto#L476>, which will reference the named runtime.

It is important to note the naming convention. Runtimes are under **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes]**, with each runtime given a unique name, e.g. **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.runc]**. In addition, each runtime can have shim-specific options under **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.<runtime>.options]**, for example, **[plugins."io.containerd.grpc.v1.cri".containerd.runtimes.runc.options]**.

The **io.containerd.runc.v2** runtime is used to run OCI-compatible runtimes on Linux, such as runc. In the example above, the **runtime_type** field specifies the shim to use (**io.containerd.runc.v2**) while the **BinaryName** field is a shim-specific option which specifies the path to the OCI runtime.

For the example configuration named "runc", the shim will launch **/usr/bin/runc** as the OCI runtime. For the example configuration named "other", the shim will launch **/usr/bin/path-to-runtime** instead.

BUGS

Please file any specific issues that you encounter at <https://github.com/containerd/containerd>.

AUTHOR

Phil Estes estesp@gmail.com (<mailto:estesp@gmail.com>)

SEE ALSO

ctr(8), containerd-config(8), containerd(8)