machine-info

- Description
- Options
  - Machine Information
  - Custom Fields
- Example
- See Also

systemd

machine-info

5

machine-info

Local machine information file

`/etc/machine-info`

Description

The `/etc/machine-info` file contains machine metadata.

The format of `machine-info` is a newline-separated list of environment-like shell-compatible variable assignments, ignoring comments and empty lines. It is possible to source the configuration from shell scripts, however, beyond mere variable assignments no shell features are supported, allowing applications to read the file without implementing a shell compatible execution engine. See `os-release(5)` for a detailed description of the format.

`/etc/machine-info` contains metadata about the machine that is set by the user or administrator. The settings configured here have the highest precedence. When not set, appropriate values may be determined automatically, based on the information about the hardware or other configuration files. It is thus completely fine for this file to not be present.

You may use `hostnamectl(1)` to change the settings of this file from the command line.

Options

The following machine metadata parameters may be set using `/etc/machine-info`:

Machine Information

`PRETTY_HOSTNAME=`
A pretty human-readable UTF-8 machine identifier string. This should contain a name like `Lennart's Laptop` which is useful to present to the user and does not suffer by the syntax limitations of internet domain names. If possible, the internet hostname as configured in `/etc/hostname` should be kept similar to this one. Example: if this value is `Lennart's Computer` an Internet hostname of `lennarts-computer` might be a good choice. If this parameter is not set, an application should fall back to the Internet hostname for presentation purposes.

`ICON_NAME=`
An icon identifying this machine according to the XDG Icon Naming Specification. If this parameter is not set, an application should fall back to `computer` or a similar icon name.

`CHASSIS=`
The chassis type. Currently, the following chassis types are defined: `desktop`, `laptop`, `convertible`, `server`, `tablet`, `handset`, `watch`, and `embedded`, as well as the special chassis types `vm` and `container` for virtualized systems that lack an immediate physical chassis.

Note that most systems allow detection of the chassis type automatically (based on firmware information or suchlike). This setting should only be used to override a misdetection or to manually configure the chassis type where automatic detection is not available.

`DEPLOYMENT=`
Describes the system deployment environment. One of the following is suggested: `development`, `integration`, `staging`, `testing`, `production`.

`LOCATION=`
Describes the system location if applicable and known. Takes a human-friendly, free-form string. This may be as generic as `Berlin, Germany` or as specific as `Left Rack, 2nd Shelf`.

`TAGS=`
A colon-separated list of tags attached to this machine. Tags are labels used to classify and group machines for management purposes, for example to identify the role a machine plays in a deployment (`company.role=webserver`, `company.role=database`), the fleet or organizational unit it

belongs to, or any other administrator-defined attribute. Example: `TAGS=company.role=webserver:company.division=frontend:company.location=berlin`.

Each individual tag must be 1…255 characters long and may consist only of the ASCII alphanumeric characters, `-`, `.` and `=`. The first character may not be `-`, `.` or `=`, and the last character may not be `-` or `.` (unless it takes the parameterized form, see below).

A tag may optionally be parameterized with a value, in the form `key=value`. The first `=` separates the key from the value; any further `=` characters are part of the value. The key (the part before the first `=`) follows the same restrictions as an unparameterized tag, in particular it may not be empty and may not end in `-` or `.`. The value (the part after the first `=`) may be empty and is otherwise unrestricted within the allowed character set. Example: `TAGS=company.role=webserver:company.env=production:company.location=berlin`. In interfaces that assign tags, the same key may not be assigned more than one value: `company.role=webserver:company.role=database` is refused (but a key may coexist with the corresponding unparameterized tag, e.g. `company.role:company.role=webserver`). When tags are read from the config file, invalid tags are ignored, and if the same key is nonetheless assigned more than once, the assignment specified last takes effect.

It is recommended for tags to be namespaced. In particular, tags given by an organization or company should be prefixed with some identifier for the organization before the first dot. Personal or purely local tags do not need to be namespaced.

The configured tags may be matched against with the `ConditionMachineTag=` and `AssertMachineTag=` unit settings, see `systemd.unit(5)` for details. They may be queried and changed with the `tags` command of `hostnamectl(1)`. Tags may also be declared in `machine-tags.d/` drop-in directories, see `machine-tags.d(5)`, and applied with `hostnamectl tags --apply`.

`HARDWARE_VENDOR=`
Specifies the hardware vendor. If unspecified, the hardware vendor set in DMI or `hwdb(7)` will be used.

`HARDWARE_MODEL=`
Specifies the hardware model. If unspecified, the hardware model set in DMI or `hwdb(7)` will be used.

`HARDWARE_SKU=`
Specifies the hardware SKU (Stock-Keeping Unit). If unspecified, the hardware SKU set in DMI will be used.

`HARDWARE_VERSION=`
Specifies the hardware version. If unspecified, the hardware version set in DMI will be used.

Custom Fields

Operating system vendors and systemd administrators may extend the file format and introduce new fields. It is highly recommended to prefix new fields with an OS-specific name in order to avoid name clashes. Applications reading this file must ignore unknown fields.

Example: `_LOCAL_EMERGENCY_CONTACT="user@example.org"`.

Example

```
PRETTY_HOSTNAME="Lennart's Tablet"
ICON_NAME=computer-tablet
CHASSIS=tablet
DEPLOYMENT=production
TAGS=room=kitchen:company.location=berlin:company.role=personal
```

See Also

`systemd(1)`, `os-release(5)`, `hostname(5)`, `machine-id(5)`, `hostnamectl(1)`, `machine-tags.d(5)`, `systemd-hostnamed.service(8)`

□□□ #1
□ □□□ □□□ 2026-10-31 14:46:52 UTC
□ □□□ □□□ 2026-10-31 14:46:52 UTC