
Ansible Collection - LinuxHA

AnsibleGuy

Nov 28, 2023

USAGE

1	1 - Installation	3
2	2 - Config	5
3	3 - Basic	9
4	1 - Basic module arguments	11
5	Config	13
6	Raw	21
7	Status	23

Tip: Check out the repository on [GitHub](#)

Tip: Check out the repository on [GitHub](#)

1 - INSTALLATION

1.1 Ansible

See the [documentation](#) on how to install Ansible.

1.2 Dependencies

1.2.1 Install

You will have to install the LinuxHA packages on the target server:

- [LinuxHA](#) (packages: [corosync](#) and [pacemaker](#))
- [crm shell](#) (package: [crmsh](#))

```
sudo apt install corosync pacemaker crmsh
```

The [xmlltodict python module](#) is used to parse config!

It is only needed on the Ansible controller!

```
python3 -m pip install xmlltodict
```

1.2.2 Configure

After that - configure the basic cluster using the 'corosync.conf' file.

See: [Docs](#)

1.3 Collection

```
# stable version:
ansible-galaxy collection install ansibleguy.linuxha

# latest version:
ansible-galaxy collection install git+https://github.com/ansibleguy/collection_linuxha.
↪git
```

(continues on next page)

(continued from previous page)

```
# install to specific directory for easier development  
cd $PLAYBOOK_DIR  
ansible-galaxy collection install git+https://github.com/ansibleguy/collection_linuxha.  
↪git -p ./collections
```

Tip: Check out the repository on [GitHub](#)

2 - CONFIG

Docs: [corosync.conf](#)

2.1 Example

```
totem {
  version: 2
  cluster_name: clusterName

  knet_transport: udp
  transport: knet

  # generate key: corosync-keygen -k /etc/corosync/authkey_clusterName
  # all nodes in the cluster must share the same key
  keyfile: /etc/corosync/authkey_clusterName
  crypto_cipher: aes256
  crypto_hash: sha256
  secauth: yes

  link_mode: passive
  # passive, active, rr
  netmtu: 1500
  ip_version: ipv4
  # ipv4, ipv6, ipv4-6

  interface {
    ringnumber: 0
    bindnetaddr: 192.168.1.0
    # network-address of subnet
    mcastport: 5405
    # +2 for next cluster
  }
  interface {
    ringnumber: 1
    bindnetaddr: 10.0.0.0
    # network-address of subnet
    mcastport: 5405
    # +2 for next cluster
  }
}
```

(continues on next page)

```
}

logging {
  # debugging
  debug: off
  fileline: off
  function_name: off
  #timestamp: hires

  to_stderr: yes
  to_syslog: yes
  syslog_facility: daemon
  syslog_priority: info
  # alert, crit, debug (same as debug = on), emerg, err, info, notice, warning

  logger_subsys {
    subsys: QUORUM
    debug: off
  }
}

quorum {
  provider: corosync_votequorum
}

nodelist {
  node {
    name: node1
    nodeid: 1
    ring0_addr: 192.168.1.1
    # private ip
    ring1_addr: 10.0.0.1
    # public ip
  }
  node {
    name: node2
    nodeid: 2
    ring0_addr: 192.168.1.2
    # private ip
    ring1_addr: 10.0.0.2
    # public ip
  }
}

system {}
nozzle {}
```

2.2 Testing

```
root@lha01:~# corosync -f
> parse error in config: ...
```

2.3 Starting

```
root@lha01:~# systemctl enable corosync.service
root@lha01:~# systemctl enable pacemaker.service
root@lha01:~# systemctl start corosync.service
root@lha01:~# systemctl start pacemaker.service
```

2.4 Checking

```
root@lha01:~# systemctl status corosync.service
> corosync.service - Corosync Cluster Engine
>   Loaded: loaded (/lib/systemd/system/corosync.service; enabled; vendor preset:
↳enabled)
>   Active: active (running) since Sat 2023-04-01 16:30:46 CEST; 9min ago

root@lha01:~# systemctl status pacemaker.service
> pacemaker.service - Pacemaker High Availability Cluster Manager
>   Loaded: loaded (/lib/systemd/system/pacemaker.service; enabled; vendor preset:
↳enabled)
>   Active: active (running) since Sat 2023-04-01 16:31:05 CEST; 9min ago

root@lha01:~# crm status
> Cluster Summary:
>   * Stack: corosync
>   * Current DC: lha02 (version 2.0.5-ba59be7122) - partition with quorum
>   * Last updated: Sat Apr  1 16:31:39 2023
>   * Last change:  Sat Apr  1 16:31:11 2023 by hacluster via crmd on lha02
>   * 2 nodes configured
>   * 0 resource instances configured
>
> Node List:
>   * Online: [ lha01 lha02 ]
>
> Full List of Resources:
>   * No resources
```

Tip: Check out [the repository on GitHub](#)

3 - BASIC

3.1 Run once

Cluster-specific operations **must only be executed on ONE NODE AT A TIME!**

Such operations include:

- Resource actions
- most Node actions
- Configuration changes

You can achieve this easily by either:

- use the 'run_once: true' parameter (*dynamic*)

```
- hosts: linuxha
gather_facts: false
become: true
tasks:
  - name: Run once per cluster
    ansibleguy.linuxha.raw:
      cmd: 'to execute'
      run_once: true
```

- add an inventory-group of 'leader-nodes' that will be used as execution targets (*static but working for multi-cluster setup*)

inventory

```
---
all:
  hosts:
    # cluster 1
    node1:
    node2:
    # cluster 2
    node3:
    node4:
  children:
    linuxha:
      hosts:
```

(continues on next page)

(continued from previous page)

```
node1:
node2:
node3:
node4:
linuxha_leader:
  hosts:
    node1:
    node3:
```

playbook

```
- hosts: linuxha_leader
  gather_facts: false
  become: true
  tasks:
    - name: Run once per cluster
      ansibleguy.linuxha.raw:
        cmd: 'to execute'
```

Tip: Check out [the repository on GitHub](#)

1 - BASIC MODULE ARGUMENTS

TESTS: [Playbook](#)

4.1 All modules

Table 1: Definition

Parameter	Type	Re-quired	Default	Comment
debug	boolean	false	false	Enable debug output for module processing and crmsh

4.2 Config/Action modules

Table 2: Definition

Parameter	Type	Re-quired	Default	Comment
wait	boolean	false	false	Make crm wait for the cluster transition to finish (<i>for the changes to take effect</i>) after each processed line
force	boolean	false	false	Make crm proceed with applying changes where it would normally ask the user to confirm before proceeding. This option is mainly useful in scripts, and should be used with care. Force will also hide some output from you or even change the exit-code from 1 to 0 on a 'soft-failure'!

Tip: Check out the repository on [GitHub](#)

STATE: unstable

TESTS: [Playbook](#)

5.1 Definition

Table 1: Definition

Parameter	Type	Re-quired	Default	Aliases	Comment
raw	boolean	false	false	-	Return the full un-modified/-simplified config-dump
subset	list	false	['groups', 'locations', 'nodes', 'orders', 'primitives', 'properties', 'clones']	parse, sub	Provide one or multiple status-subsets to parse (<i>ignored if 'raw: true' is set</i>)

For basic parameters see: [Basic](#)

5.2 Info

Module to pull the current LinuxHA configuration.

5.3 Examples

```

- hosts: node1
gather_facts: false
become: true
tasks:
  - name: Example
    ansibleguy.linuxha.config:
      # raw: false
      # subset: ['groups', 'locations', 'nodes', 'orders', 'primitives', 'properties', 'clones']

  - name: Pulling current config
    ansibleguy.linuxha.config:
      register: lha_config

  - name: Showing config
    ansible.builtin.debug:
      var: lha_config.data

# {
#   "clones": {
#     "ANSIBLE_TEST_3_5": [
#       "ANSIBLE_TEST_3_4"
#     ]
#   },
#   "groups": {
#     "ANSIBLE_TEST_3_6": [
#       "ANSIBLE_TEST_3_1",
#       "ANSIBLE_TEST_3_2"
#     ]
#   },
#   "locations": {
#     "ANSIBLE_TEST_3_8": {
#       "node": "node2",
#       "role": "Started",
#       "rsc": "ANSIBLE_TEST_3_3",
#       "score": "INFINITY"
#     }
#   },
#   "nodes": {
#     "node1": 1,
#     "node2": 2
#   },
#   "orders": {
#     "ANSIBLE_TEST_3_7": {
#       "first": "ANSIBLE_TEST_3_3",
#       "first-action": "start",
#       "kind": "Mandatory",
#       "then": "ANSIBLE_TEST_3_1",
#       "then-action": "start"
#     }
#   },
# }

```

(continues on next page)

(continued from previous page)

```
# "primitives": {
#   "ANSIBLE_TEST_3_1": {
#     "class": "ocf",
#     "params": {
#       "ip": "127.100.1.2",
#       "nic": "lo"
#     },
#     "provider": "heartbeat",
#     "type": "IPAddr2"
#   },
#   "ANSIBLE_TEST_3_2": {
#     "class": "ocf",
#     "params": {
#       "ip": "127.100.1.3",
#       "nic": "lo"
#     },
#     "provider": "heartbeat",
#     "type": "IPAddr2"
#   },
#   "ANSIBLE_TEST_3_3": {
#     "class": "ocf",
#     "params": {
#       "ip": "127.100.1.4",
#       "nic": "lo"
#     },
#     "provider": "heartbeat",
#     "type": "IPAddr2"
#   },
#   "ANSIBLE_TEST_3_4": {
#     "class": "ocf",
#     "operations": {
#       "monitor": {
#         "interval": "5s",
#         "on-fail": "restart",
#         "timeout": "60"
#       }
#     },
#     "params": {
#       "host_list": "1.1.1.1 8.8.8.8"
#     },
#     "provider": "pacemaker",
#     "type": "ping"
#   }
# },
# "properties": {
#   "cluster-infrastructure": "corosync",
#   "cluster-name": "debian",
#   "dc-version": "2.0.5-ba59be7122",
#   "have-watchdog": false,
#   "last-lrm-refresh": 1680443090,
#   "stonith-enabled": false
# }
```

(continues on next page)

```
# }

- name: Pulling only a subset of the current config
  ansibleguy.linuxha.config:
    subset: ['properties', 'primitives']
    register: lha_config_subset

- name: Showing config subset
  ansible.builtin.debug:
    var: lha_config_subset.data

# {
#   "primitives": {
#     "ANSIBLE_TEST_3_1": {
#       "class": "ocf",
#       "params": {
#         "ip": "127.100.1.2",
#         "nic": "lo"
#       },
#       "provider": "heartbeat",
#       "type": "IPAddr2"
#     },
#     "ANSIBLE_TEST_3_2": {
#       "class": "ocf",
#       "params": {
#         "ip": "127.100.1.3",
#         "nic": "lo"
#       },
#       "provider": "heartbeat",
#       "type": "IPAddr2"
#     },
#     "ANSIBLE_TEST_3_3": {
#       "class": "ocf",
#       "params": {
#         "ip": "127.100.1.4",
#         "nic": "lo"
#       },
#       "provider": "heartbeat",
#       "type": "IPAddr2"
#     },
#     "ANSIBLE_TEST_3_4": {
#       "class": "ocf",
#       "operations": {
#         "monitor": {
#           "interval": "5s",
#           "on-fail": "restart",
#           "timeout": "60"
#         }
#       },
#       "params": {
#         "host_list": "1.1.1.1 8.8.8.8"
#       }
#     }
#   }
# }
```

(continues on next page)

(continued from previous page)

```

#         "provider": "pacemaker",
#         "type": "ping"
#     }
# },
# "properties": {
#     "cluster-infrastructure": "corosync",
#     "cluster-name": "debian",
#     "dc-version": "2.0.5-ba59be7122",
#     "have-watchdog": false,
#     "last-lrm-refresh": 1680443090,
#     "stonith-enabled": false
# }
# }

- name: Pulling current config in raw-format
  ansibleguy.linuxha.config:
    raw: true
  register: lha_raw_config

- name: Showing raw-config
  ansible.builtin.debug:
    var: lha_raw_config.data

# {
#   "admin_epoch": "0",
#   "cib-last-written": "Sat Apr  8 16:33:40 2023",
#   "configuration": {
#     "constraints": {
#       "rsc_location": {
#         "id": "ANSIBLE_TEST_3_8",
#         "node": "node2",
#         "role": "Started",
#         "rsc": "ANSIBLE_TEST_3_3",
#         "score": "INFINITY"
#       },
#       "rsc_order": {
#         "first": "ANSIBLE_TEST_3_3",
#         "first-action": "start",
#         "id": "ANSIBLE_TEST_3_7",
#         "kind": "Mandatory",
#         "then": "ANSIBLE_TEST_3_1",
#         "then-action": "start"
#       }
#     },
#     "crm_config": {
#       "cluster_property_set": {
#         "id": "cib-bootstrap-options",
#         "nvpair": [
#           {
#             "id": "cib-bootstrap-options-have-watchdog",
#             "name": "have-watchdog",
#             "value": "false"

```

(continues on next page)

(continued from previous page)

```
#         },
#         {
#             "id": "cib-bootstrap-options-dc-version",
#             "name": "dc-version",
#             "value": "2.0.5-ba59be7122"
#         },
#         {
#             "id": "cib-bootstrap-options-cluster-infrastructure",
#             "name": "cluster-infrastructure",
#             "value": "corosync"
#         },
#         {
#             "id": "cib-bootstrap-options-cluster-name",
#             "name": "cluster-name",
#             "value": "debian"
#         },
#         {
#             "id": "cib-bootstrap-options-stonith-enabled",
#             "name": "stonith-enabled",
#             "value": "false"
#         },
#         {
#             "id": "cib-bootstrap-options-last-lrm-refresh",
#             "name": "last-lrm-refresh",
#             "value": "1680443090"
#         }
#     ]
# }
# },
# "nodes": {
#     "node": [
#         {
#             "id": "1",
#             "uname": "node1"
#         },
#         {
#             "id": "2",
#             "uname": "node2"
#         }
#     ]
# },
# "resources": {
#     "clone": {
#         "id": "ANSIBLE_TEST_3_5",
#         "primitive": {
#             "class": "ocf",
#             "id": "ANSIBLE_TEST_3_4",
#             "instance_attributes": {
#                 "id": "ANSIBLE_TEST_3_4-instance_attributes",
#                 "nvpair": {
#                     "id": "ANSIBLE_TEST_3_4-instance_attributes-host_list",
#                     "name": "host_list",
```

(continues on next page)

(continued from previous page)

```

#         "value": "1.1.1.1 8.8.8.8"
#     }
# },
#     "operations": {
#         "op": {
#             "id": "ANSIBLE_TEST_3_4-monitor-5s",
#             "interval": "5s",
#             "name": "monitor",
#             "on-fail": "restart",
#             "timeout": "60"
#         }
#     },
#     "provider": "pacemaker",
#     "type": "ping"
# },
#     "group": {
#         "id": "ANSIBLE_TEST_3_6",
#         "primitive": [
#             {
#                 "class": "ocf",
#                 "id": "ANSIBLE_TEST_3_1",
#                 "instance_attributes": {
#                     "id": "ANSIBLE_TEST_3_1-instance_attributes",
#                     "nvpair": [
#                         {
#                             "id": "ANSIBLE_TEST_3_1-instance_attributes-ip
→ ",
#                             "name": "ip",
#                             "value": "127.100.1.2"
#                         },
#                         {
#                             "id": "ANSIBLE_TEST_3_1-instance_attributes-nic
→ ",
#                             "name": "nic",
#                             "value": "lo"
#                         }
#                     ]
#                 },
#                 "provider": "heartbeat",
#                 "type": "IPaddr2"
#             },
#             {
#                 "class": "ocf",
#                 "id": "ANSIBLE_TEST_3_2",
#                 "instance_attributes": {
#                     "id": "ANSIBLE_TEST_3_2-instance_attributes",
#                     "nvpair": [
#                         {
#                             "id": "ANSIBLE_TEST_3_2-instance_attributes-ip
→ ",
#                             "name": "ip",

```

(continues on next page)

(continued from previous page)

```

#                                     "value": "127.100.1.3"
#                                     },
#                                     {
#                                     "id": "ANSIBLE_TEST_3_2-instance_attributes-nic
↪",
#                                     "name": "nic",
#                                     "value": "lo"
#                                     }
#                                     ]
#                                     },
#                                     "provider": "heartbeat",
#                                     "type": "IPAddr2"
#                                     }
#                                     ]
#                                     },
#                                     "primitive": {
#                                     "class": "ocf",
#                                     "id": "ANSIBLE_TEST_3_3",
#                                     "instance_attributes": {
#                                     "id": "ANSIBLE_TEST_3_3-instance_attributes",
#                                     "nvpair": [
#                                     {
#                                     "id": "ANSIBLE_TEST_3_3-instance_attributes-ip",
#                                     "name": "ip",
#                                     "value": "127.100.1.4"
#                                     },
#                                     {
#                                     "id": "ANSIBLE_TEST_3_3-instance_attributes-nic",
#                                     "name": "nic",
#                                     "value": "lo"
#                                     }
#                                     ]
#                                     },
#                                     "provider": "heartbeat",
#                                     "type": "IPAddr2"
#                                     }
#                                     }
#                                     },
#                                     "crm_feature_set": "3.6.1",
#                                     "dc-uuid": "1",
#                                     "epoch": "418",
#                                     "have-quorum": "1",
#                                     "num_updates": "12",
#                                     "update-client": "cibadmin",
#                                     "update-origin": "node1",
#                                     "update-user": "root",
#                                     "validate-with": "pacemaker-3.5"
#                                     }
# }

```

Tip: Check out the repository on GitHub

STATE: unstable

TESTS: [Playbook](#)

Docs: [crm-shell](#)

6.1 Definition

Table 1: Definition

Parameter	Type	Re-quired	Default	Aliases	Comment
cmd	string	true	-	com- mand, c	Raw command to pass to crm-shell
fail	boolean	false	true	f	Fail module if command fails

For basic parameters see: [Basic](#)

6.2 Info

Will pass any command to ‘crm-shell’.

Warning: THERE IS NO CLIENT-SIDE CONFIG VALIDATION!

6.3 Examples

```
- hosts: node1
gather_facts: false
become: true
tasks:
  - name: Example
    ansibleguy.linuxha.raw:
      cmd: 'to execute'
      # fail: true # Fail module if command fails
```

(continues on next page)

(continued from previous page)

```
# force: false
# wait: false
# debug: false

- name: Pulling raw running-config
  ansibleguy.linuxha.raw:
    cmd: 'configure show'
    register: lha_cnf

- name: Showing config
  ansible.builtin.debug:
    var: lha_cnf.stdout_lines

- name: Disabling stonith
  ansibleguy.linuxha.raw:
    cmd: 'configure property stonith-enabled=false'

- name: Adding resource
  ansibleguy.linuxha.raw:
    cmd: 'configure primitive vip1 IPAddr2 params ip=10.15.12.1 nic=eno1'

- name: Pulling raw status
  ansibleguy.linuxha.raw:
    cmd: 'status bynode'
    register: lha_status

- name: Showing status
  ansible.builtin.debug:
    var: lha_status.stdout_lines
```

Tip: Check out the repository on [GitHub](#)

STATE: unstable

TESTS: [Playbook](#)

7.1 Definition

Table 1: Definition

Parameter	Type	Re-quired	Default	Aliases	Comment
detailed	boolean	false	false	detail	Return a more detailed status parsed from XML output
subset	list	false	['cluster', 'nodes', 'resources', 'operations']	parse, sub	Provide one or multiple status-subsets to parse (<i>ignored if 'detailed: true' is set</i>)
time_format	string	false	%Y-%m-%d %H:%M:%S	t_fmt	Modify the datetime format used to parse timestamps (<i>ignored if 'detailed: true' is set</i>)

For basic parameters see: [Basic](#)

7.2 Info

Get parsed cluster-status.

If 'detailed: false' (*default*) is set - the output from 'crm status full' is parsed.

Else the output from 'crm status xml' is used.

The 'detailed' output might get updated in the future! (*different modes to filter data and so on*)

7.3 Examples

```

- hosts: node1
gather_facts: false
become: true
tasks:
  - name: Example
    ansibleguy.linuxha.status:
      # subset: ['cluster', 'nodes', 'resources', 'operations']
      # time_format: '%Y-%m-%d %H:%M:%S'

  - name: Pulling status
    ansibleguy.linuxha.status:
      register: lha_status

  - name: Showing status
    ansible.builtin.debug:
      var: lha_cnf.data

# {
#   "cluster": {
#     "changed_node": "node1",
#     "changed_time": "2023-04-02 15:49:30"
#     "changed_user": "root",
#     "changed_via": "cibadmin",
#     "dc": "node1",
#     "node_count": 2,
#     "resource_count": 7,
#     "updated": "2023-04-02 19:29:41"
#     "version": "2.0.5-ba59be7122"
#   },
#   "nodes": {
#     "node1": {
#       "resources": {
#         "a2": {
#           "resource": {
#             "class": null,
#             "provider": "systemd",
#             "type": "apache2.service"
#           },
#           "state": "Started"
#         },
#         "ng": {
#           "resource": {
#             "class": null,
#             "provider": "systemd",
#             "type": "nginx.service"
#           },
#           "state": "Started"
#         },
#         "pingGW": {
#           "resource": {

```

(continues on next page)

(continued from previous page)

```

#         "class": "ocf",
#         "provider": "pacemaker",
#         "type": "ping"
#     },
#     "state": "Started"
# },
#     "test2": {
#         "resource": {
#             "class": "ocf",
#             "provider": "heartbeat",
#             "type": "IPaddr2"
#         },
#         "state": "Started"
#     }
# },
#     "status": "online"
# },
#     "node2": {
#         "resources": {
#             "a2": {
#                 "resource": {
#                     "class": null,
#                     "provider": "systemd",
#                     "type": "apache2.service"
#                 },
#                 "state": "Started"
#             },
#             "pingGW": {
#                 "resource": {
#                     "class": "ocf",
#                     "provider": "pacemaker",
#                     "type": "ping"
#                 },
#                 "state": "Started"
#             },
#             "test1": {
#                 "resource": {
#                     "class": "ocf",
#                     "provider": "heartbeat",
#                     "type": "IPaddr2"
#                 },
#                 "state": "Started"
#             }
#         },
#         "status": "online"
#     }
# },
#     "resources": {
#         "a2": {
#             "nodes": {
#                 "node1": "Started",
#                 "node2": "Started"

```

(continues on next page)

(continued from previous page)

```
#         },
#         "resource": {
#             "class": null,
#             "provider": "systemd",
#             "type": "apache2.service"
#         }
#     },
#     "ng": {
#         "nodes": {
#             "node1": "Started"
#         },
#         "resource": {
#             "class": null,
#             "provider": "systemd",
#             "type": "nginx.service"
#         }
#     },
#     "pingGW": {
#         "nodes": {
#             "node1": "Started",
#             "node2": "Started"
#         },
#         "resource": {
#             "class": "ocf",
#             "provider": "pacemaker",
#             "type": "ping"
#         }
#     },
#     "test1": {
#         "nodes": {
#             "node2": "Started"
#         },
#         "resource": {
#             "class": "ocf",
#             "provider": "heartbeat",
#             "type": "IPAddr2"
#         }
#     },
#     "test2": {
#         "nodes": {
#             "node1": "Started"
#         },
#         "resource": {
#             "class": "ocf",
#             "provider": "heartbeat",
#             "type": "IPAddr2"
#         }
#     }
# },
# "operations": {
#     "node1": {
#         "a2": {
```

(continues on next page)

(continued from previous page)

```

#         "2023-04-02 15:44:50": {
#             "change": "2023-04-02 15:44:50",
#             "migration_threshold": "1000000",
#             "operation": "probe",
#             "operation_id": "128",
#             "rc": "0",
#             "status": "ok",
#             "time_exec": "0ms",
#             "time_queue": "0ms"
#         }
#     },
#     "pingGW": {
#         "2023-04-02 15:48:13": {
#             "change": "2023-04-02 15:48:13",
#             "migration_threshold": "1000000",
#             "operation": "start",
#             "operation_id": "143",
#             "rc": "0",
#             "status": "ok",
#             "time_exec": "10102ms",
#             "time_queue": "0ms"
#         }
#     },
#     "test1": {
#         "2023-04-02 15:33:44": {
#             "change": "2023-04-02 15:33:44",
#             "migration_threshold": "1000000",
#             "operation": "stop",
#             "operation_id": "112",
#             "rc": "0",
#             "status": "ok",
#             "time_exec": "17ms",
#             "time_queue": "0ms"
#         }
#     }
# },
#     "node2": {
#         "a2": {
#             "2023-04-02 15:44:51": {
#                 "change": "2023-04-02 15:44:51",
#                 "migration_threshold": "1000000",
#                 "operation": "probe",
#                 "operation_id": "102",
#                 "rc": "0",
#                 "status": "ok",
#                 "time_exec": "0ms",
#                 "time_queue": "0ms"
#             }
#         }
#     }
# }

```

(continues on next page)

(continued from previous page)

```
- name: Pulling detailed status
  ansibleguy.linuxha.status:
    detailed: true
    register: lha_status_detailed

- name: Showing detailed status
  ansible.builtin.debug:
    var: lha_status_detailed.data

# {
#   "node_history": {
#     "node": {
#       "name": "node1",
#       "resource_history": {
#         "id": "ANSIBLE_TEST_2_1",
#         "migration-threshold": "1000000",
#         "operation_history": {
#           "call": "245",
#           "exec-time": "21ms",
#           "last-rc-change": "Sun Apr  2 20:59:51 2023",
#           "last-run": "Sun Apr  2 20:59:51 2023",
#           "queue-time": "0ms",
#           "rc": "0",
#           "rc_text": "ok",
#           "task": "start"
#         },
#       },
#       "orphan": "false"
#     }
#   },
#   "nodes": {
#     "node": [
#       {
#         "expected_up": "true",
#         "id": "1",
#         "is_dc": "true",
#         "maintenance": "false",
#         "name": "node1",
#         "online": "true",
#         "pending": "false",
#         "resources_running": "1",
#         "shutdown": "false",
#         "standby": "false",
#         "standby_onfail": "false",
#         "type": "member",
#         "unclean": "false"
#       },
#       {
#         "expected_up": "true",
#         "id": "2",
#         "is_dc": "false",
```

(continues on next page)

(continued from previous page)

```

#         "maintenance": "false",
#         "name": "node2",
#         "online": "true",
#         "pending": "false",
#         "resources_running": "0",
#         "shutdown": "false",
#         "standby": "false",
#         "standby_onfail": "false",
#         "type": "member",
#         "unclean": "false"
#     }
# ]
# },
# "resources": {
#     "resource": {
#         "active": "true",
#         "blocked": "false",
#         "failed": "false",
#         "failure_ignored": "false",
#         "id": "ANSIBLE_TEST_2_1",
#         "managed": "true",
#         "node": {
#             "cached": "true",
#             "id": "1",
#             "name": "node1"
#         },
#     },
#     "nodes_running_on": "1",
#     "orphaned": "false",
#     "resource_agent": "ocf::heartbeat:IPaddr2",
#     "role": "Started"
# },
# },
# "summary": {
#     "cluster_options": {
#         "maintenance-mode": "false",
#         "no-quorum-policy": "stop",
#         "stonith-enabled": "false",
#         "stop-all-resources": "false",
#         "symmetric-cluster": "true"
#     },
#     "current_dc": {
#         "id": "1",
#         "name": "node1",
#         "present": "true",
#         "version": "2.0.5-ba59be7122",
#         "with_quorum": "true"
#     },
#     "last_change": {
#         "client": "cibadmin",
#         "origin": "node1",
#         "time": "Sun Apr  2 20:59:51 2023",
#         "user": "root"
#     }
# }

```

(continues on next page)

(continued from previous page)

```
# },
#   "last_update": {
#     "time": "Sun Apr  2 20:59:55 2023"
#   },
#   "nodes_configured": {
#     "number": "2"
#   },
#   "resources_configured": {
#     "blocked": "0",
#     "disabled": "0",
#     "number": "1"
#   },
#   "stack": {
#     "type": "corosync"
#   }
# },
#   "version": "2.0.5"
# }
```