Day-to-Day Device Operations

A console-first guide for operators and engineers · Checked against tacctl 0.2.25

Contents

0. The tacctl console 2

1. Operators and engineers 2

1.1 Find a device 2

1.2 Connect to a device 2

1.3 Check a device that isn't behaving (operator) 3

1.4 Look at authentication logs (operator) 3

1.5 See which devices have configuration drift (operator) 4

2. Engineers 4

2.1 Bring devices in 4

2.2 The daily change routine 5

3. Common problems 6

4. Quick reference 6

A quick guide to the tacctl tasks that operators and engineers do every day, all from the tacctl console. **Section 1** is for operators and engineers. **Section 2** is for engineers only. For superuser tasks, see *Superuser Daily Operations*.

> **Example names used on this page** Server: `tacacs.example.net` · Scope: `site-a` · Devices: `sw1`, `sw2`, `sw3` (Cisco), `rtr1`, `rtr2`, `jsw1` (Junos), `pdu1` (WTI) · You: `alice`
>
> **Read-only users:** you can use the console, `device list`, `device show`, `device notice list`, `ssh` and `passwd`. The checks, logs and drift views in Section 1 need the **operator** tier. In the console, `help` lists exactly what your role can run.

0. The tacctl console

Log in to the tacctl server with ssh (`ssh alice@tacacs.example.net`). You land at the console prompt, `tacacs>`. Type commands **without** the `tacctl` prefix. `Tab` completes, `?` (or `--help` after a command) shows what comes next, `help` lists what your tier can run, `man <command>` opens the manual page there, `passwd` changes your password and `exit` leaves. *Getting Started* §1 covers the console in full.

---

1. Operators and engineers

1.1 Find a device

```
device list                       # every device in your scopes
device list core                  # anything matching "core" (name, address, vendor or
scope)
device list --scope site-a        # one site or scope
device list --vendor juniper
device list --probe               # (operator) add REACH: does each device answer on ssh
right now
device show sw1                   # full details for one device
```

1.2 Connect to a device

```
ssh sw1                           # interactive session
ssh sw1 -- show version           # run one command and come back
ssh rtr1 -- show interfaces terse
```

You are logged in as yourself. `exit` on the device brings you back to the console.

If ssh prints a **pinned and an offered fingerprint** and refuses, the host key has changed. Do not work around it: see *Connecting to Devices* §5.

1.3 Check a device that isn't behaving (operator)

```
device check sw1        # scope, last seen, reachable, host key vs. pin, SNMP name/
location/vendor
device check sw1 --sources   # also test the addresses the device sends AAA from
device notice list      # open issues, each with the command that fixes it
```

`device check` connects from the tacctl server. If the server has no route to the device's management port, a timeout may be a false alarm.

With `--sources`, `device check` also connects to each address the device sends its AAA requests from (other than its registered address) and compares its ssh host key with the pinned one, without logging in. A warning that an address is unreachable (it may be by design) or presents another host key goes to an engineer (*Engineer Guide* §3.1).

1.4 Look at authentication logs (operator)

```
log failures               # failed logins in the last 24 hours
log search alice           # one user's (or a keyword's) entries; TACACS+: last 7 days
log tail 50                # the 50 most recent entries
log tail -f                # follow live; Ctrl-C to stop
log accounting 50          # the 50 most recent accounting records (commands run on
devices)
```

Narrow it down (0.2.24). `log tail`, `log search`, `log failures` and `log accounting` take filters that combine (an entry must pass all of them), in any place on the line:

```
log failures --since -1h                    # failed logins in the last hour
log search alice --since yesterday --until today
log accounting 20 --user alice --device sw1   # alice's last 20 records on sw1
log accounting --type change --since "2026-11-04 08:00"
log accounting 50 --json                    # one JSON object per line, for jq
```

Filter	Shows
`--since <time>`, `--until <time>`	Entries from, and up to, that time
`--user <name>`	Entries of that user, as the log writes the name
`--device <name or address>`	Entries of that device or enrolled host (of your scopes), or of an address
`--type login|change|command`	Accounting only: one kind of record
`--json`	One JSON object per entry, no headings

- A time is `YYYY-MM-DD`, `YYYY-MM-DD HH:MM` or `YYYY-MM-DD HH:MM:SS` (local time), `now`, `today`, `yesterday`, or an offset such as `-1h`, `-30m` or `2 days ago`. Any other form is a usage error, as is a `--since` later than the `--until`.

- The filters apply before a count: `log accounting 5 --user alice` is alice's last five records. `log tail -f` applies them to new entries.
- A name that is neither a device nor a host you may see is refused. A log that names no device shows nothing for `--device` and says so. A filter that leaves nothing prints `No matching entries.`
- Each accounting record now names the device that sent it (`"Device":"sw1"`; the address when the device is not registered), and `<`, `>` and `&` show as typed. Records written before 0.2.24 have no `Device`.
- Operators pass the filters through sudo, `log failures` included.

Example: a user reports "I can't log in to sw2".

1. Run `log search <user>`.
2. Match the log line:

Log line	Meaning
`failed to validate the user …`	Wrong password
`bad secret detected for ip …`	The device's TACACS+ key is wrong. Hand it to an engineer.
No line at all	The request never reached the server. Hand it to an engineer.

1.5 See which devices have configuration drift (operator)

```
device config list --differs     # devices whose last pull differs from the standard
device list --differs            # the same devices, in the device table
device config list --failed      # devices whose last pull failed
device config list --never       # devices never read
```

Report these to an engineer.

---

2. Engineers

Everything here works on the devices of **your own scopes**. Other scopes' devices are refused. Each task links to the step-by-step section of *Engineer Guide: Device Changes*.

2.1 Bring devices in

Task	At the console	Details
Onboard a new address (the usual way)	`device enroll 10.10.0.5`	Engineer Guide §2.2

Task	At the console	Details
Enroll every unregistered device that already logs in	`device discover`, then `device enroll --discovered`	Engineer Guide §2.2
Register a device you know	`device add sw3 10.10.0.3 --vendor cisco --pull`	Engineer Guide §2.1
Bring in a blank Junos device	`device onboard 10.10.0.31 --login admin --name jsw1 --check`	Engineer Guide §2.3
Print the walkthrough to paste by hand	`device config show sw3`	Engineer Guide §4.1

> Every registration shows the device's ssh key fingerprint. **Compare it with the device's console** before you accept it (*Connecting to Devices* §5).

2.2 The daily change routine

```
device config pull --scope site-a --diff     # read the scope, then show what differs
device config list --differs                 # which devices differ
device config diff sw2                       # what differs on one device, section by section
device config plan sw2                       # the fix: report, paste region, checks, back-out
device config apply rtr2 --check             # Junos: the device's own commit check; commits nothing
device config apply rtr2                     # Junos: type the name back, commit confirmed, proof
device config pull sw2 --diff                # after a Cisco paste: confirm the fix
```

How each vendor takes the fix:

Vendor	You do	Details
Cisco	Paste the region by hand, with a second session open	Engineer Guide §5.5
Junos	Let tacctl apply it under `commit confirmed`	Engineer Guide §5.1
WTI	Type the menu checklist. No paste, no rollback.	Engineer Guide §5.6

For several Junos devices, check the scope first (`device config apply --scope site-a --check`), apply one device to review and prove the change, then apply the rest (Engineer Guide §5.1; what a shape and a canary are: §5.2 and §5.3).

> **`PENDING until <time>`** on a Junos apply means: **type nothing on that device until the time passes.** Any `commit` would confirm the change.

If a change goes wrong: `device config plan <name> --backout` prints the back-out. If you are locked out completely, ask a superuser for a break-glass hold (*Break Glass*). `breakglass list` shows the holds in your scopes.

---

3. Common problems

Problem	What to do	Who
A device shows `unconfigured`	Its address isn't covered by a scope. Ask a superuser to add it.	Superuser
ssh says the host key changed	Check the key on the console, then pin the verified key: `device hostkey <name> set SHA256:<fp>` (*Connecting to Devices* §5)	Engineer (own scopes) or superuser
You can't see a device	It is outside your scopes. Ask a superuser for access.	Superuser
A login is refused on a device	Run `log search <user>` and `log failures` (see §1.4).	Operator
`needs pull (syntax unknown)` on a Cisco device	Run `device ios-syntax <name> ios12` (IOS 12.x) or `modern`, then pull again.	Engineer
An old Cisco device won't connect	Run `device legacy-ssh <name> enable`.	Engineer
A pull fails with `timed out`	Check reachability with `device check <name>`.	Operator
A pull or ssh is refused: break-glass held	The server isn't answering that device on purpose. See `breakglass list`.	—
Apply says `failed: calls itself '<x>'`	The device's hostname doesn't match the registry. Check you have the right device.	Engineer

---

4. Quick reference

Task	Command	Who
Find a device	`device list <text>`	Everyone
Connect	`ssh <name>`	Everyone
Run one command	`ssh <name> -- <command>`	Everyone
Change password	`passwd`	Everyone
Health check	`device check <name>`	Operators
Failed logins	`log failures` / `log search <user>`	Operators

Task	Command	Who
Devices with drift	`device config list --differs`	Operators
Onboard a device	`device enroll <address>`	Engineers
Read configurations	`device config pull <name>` or `--scope <scope>`	Engineers
What differs	`device config diff <name>`	Engineers
How to fix it	`device config plan <name>`	Engineers
Fix it (Junos)	`device config apply <name>`	Engineers
Back-out	`device config plan <name> --backout`	Engineers

For more detail, type `help <command>`, `<command> --help` or `man <command>` at the console.

See also: Operator Field Guide · Engineer Guide: Device Changes · Troubleshooting · `man tacctl`