# k1.2 — What Differs Across Distros (and Why It Matters) 

Even though Linux systems feel similar, the *pain points* usually come from a handful of predictable differences. Learn these once, and you'll be able to "translate" smoothly between Debian/Ubuntu and RHEL-like systems (Rocky/Alma).

---

## 1) Package management (how you install/update software)

This is the **#1 day-to-day difference**.

- **Debian/Ubuntu:** `apt` (packages are typically `.deb`)
- **RHEL family:** `dnf` (packages are typically `.rpm`)

**Why you care:**

- Commands differ (`apt install` vs `dnf install`).
- Package *names* sometimes differ (e.g., dev headers, utilities, service names).
- Default repo contents and versions can vary a lot (important for PHP/Node).

---

## 2) Release cadence & "stability vs freshness"

Distros make different tradeoffs:

- **Debian stable / RHEL-like:** prioritize **predictability** and long support windows.
- **Ubuntu LTS:** also stable, but often ships **newer** versions than Debian stable.

**Why you care:**

- You might not get the newest PHP/Node from default repos on a “stable” distro.
- You’ll frequently choose between:
  - *distro packages* (stable, integrated), vs
  - *vendor repos / language version managers* (newer, more moving parts)

---

# 3) Security frameworks & defaults (MAC systems)

Two common “why is this blocked?” systems:

- **SELinux** (very common/enforced on RHEL family)
- **AppArmor** (common on Ubuntu)

**Why you care:**

- Your config can look correct, permissions can look correct, and it still fails.
- Troubleshooting often means:
  **service logs + system logs + security policy** (not just “chmod it”)

---

# 4) Firewall tooling (same concept, different interface)

The *concept* is universal: allow only what you need.

But tooling differs:

- **Ubuntu commonly:** UFW (friendly frontend)
- **RHEL family commonly:** firewalld

**Why you care:**

- Opening ports (like 80/443) is expressed differently.
- Guides online often assume one tool; you’ll learn the underlying idea so you can adapt.

---

# 5) Defaults, layout conventions, and “where is the config?”

Most Linux systems follow similar standards (especially for `/etc`, `/var`, logs), but defaults still vary:

- Different default config paths *per service*
- Different default module layouts (e.g., Nginx includes, PHP-FPM pool configs)
- Different service names in systemd

**Why you care:**

- You'll learn to *locate* configuration and logs reliably, rather than memorize one distro's exact path.

---

# A useful "translation mindset" 

When moving between distros, ask the same questions every time:

1. **How do I install it here?** (APT vs DNF, package names)
2. **What's the service name?** (`systemctl status …`)
3. **Where are the config + logs?** (`/etc/...`, `/var/log/...`, `journalctl`)
4. **Is a security framework blocking it?** (SELinux/AppArmor indicators)

---

# Tiny practice prompt (optional)

Tell me which family you want to *start* with in examples later:

1. Debian/Ubuntu
2. RHEL-like (Rocky/Alma)
3. Keep everything dual-written (both), whenever it matters

---

Revision #1
Created 2026-04-04 23:25:40 UTC by art10m
Updated 2026-04-04 23:25:54 UTC by art10m