1 ROLE RESPONSIBILITIES

Hardening at round start. Eviction and patching once a threat hunter hands off a confirmed foothold. Restoring a broken service without weakening it. Stripping the tooling red team's playbook depends on, ahead of any foothold.

2 ORDER OF OPERATIONS

Round start: Firewall, user cleanup, package updates, service config, red team tooling audit (below), in that order, per host.

Nonessential services: Start shut down. Restart only once validated.

On handoff: Confirm account, process chain, hole, and scope before starting. Ask if anything is missing.

Evict: Kill session, reset/remove account, remove persistence.

Patch: Fix the actual vulnerability, not just the artifact. Rotate anything possibly compromised.

Protect the fix: Harder to undo than it was to apply.

3 DENY RED TEAM: SAFE TO STRIP

Confirm nothing scored depends on it, then log the removal like any other change.

- **Python 2/3, if unused:** Ansible's default connection plugin needs a Python interpreter on the managed node; most red team post-ex and staging scripts assume one too. `apt remove --purge python3` (or `chmod 000` the binary and `apt-mark hold` python3 to keep it pinned) breaks both.
- **Perl, Ruby, other unused interpreters:** same idea, lower prevalence. Rarely needed by a scored app, often needed by an older exploit or pivot script.
- **Compilers (gcc, cc, make, build-essential):** rarely needed at runtime. Blocks on-box compilation of a privilege-escalation exploit or custom implant.
- **netcat/ncat/socat, if unused:** the default pivot and reverse-shell relay. Check startup scripts and cron before removing.
- **SSH client (not server) on hosts that never SSH out:** removes a lateral-movement path, leaves inbound access alone.
- **curl/wget, if a service never calls out:** rarely safe to pull, but closes the fetch-and-execute pattern on a locked-down box.

4 DENY RED TEAM: PROTOCOLS & AGENTS

- **WinRM/PSRemoting:** transport for Ansible's `winrm` connection and common Empire/C2 lateral movement. Disable if unused: `Disable-PSRemoting -Force`, block 5985/5986 at the host firewall.
- **Admin shares (`C$`, `ADMIN$`):** what psexec/wmiexec ride in on. Scope to your management host with a firewall rule instead of a blanket disable if you use them yourself.
- **PowerShell v2:** installable alongside v5+, lacks v5's logging; attackers downgrade to it on purpose. `Disable-WindowsOptionalFeature -Online -FeatureName MicrosoftWindowsPowerShellV2`.
- **Outbound access, generally:** cutting default outbound after hardening (Firewall Reference) kills fetch-and-execute, Ansible included.
- Needed by a scored service? Narrow it, don't cut it, and note why in the tracking template.

5 EVICTION COMMANDS

- *Linux:* `kill -9 [PID]`. *Windows:* `Stop-Process -Id [PID] -Force`.
- Reset a stolen account's password immediately. Delete an attacker-created account.
- Remove persistence: cron (`crontab -e`, `/etc/cron.d`), scheduled tasks, unfamiliar `authorized_keys`, any added service/startup entry.
- `userdel -r [user]` (Linux) / `Remove-LocalUser` (Windows): remove an attacker-created account and its home directory.
- Log what you found before deleting it.

6 PATCHING COMMANDS

- Update the vulnerable package: `apt update && apt upgrade -y` (Debian), Windows Update / `DISM` (Windows).
- Close the specific exposed port or service.
- Fix the misconfiguration, not just its symptom.
- `debsums -c`: verify installed package files against known-good hashes (Debian), to confirm a patched binary is actually clean.

7 SSH & REMOTE ACCESS HARDENING

- `/etc/ssh/sshd_config`: `PermitRootLogin no`, `PasswordAuthentication no` once keys are confirmed working, `AllowUsers [list]`.
- `systemctl restart sshd` after any config change; test a second session before closing the first.
- `fail2ban`: auto-ban repeated failed logins. `apt install fail2ban`, configure a jail for sshd.
- Windows: NLA required for RDP, restrict Remote Desktop Users group membership.

8 FIREWALL REFERENCE

- `ufw allow ssh`, `ufw allow [port]/tcp`, `ufw enable`. (Linux)
- `firewall-cmd --permanent --add-service=http` then `--reload` (RHEL family).
- Windows: `New-NetFirewallRule`, or `netsh advfirewall firewall add rule` for the legacy syntax.
- Disallow unnecessary outbound after hardening a host; most legitimate services do not need free outbound access.

9 PROTECTING A FIX

- `chattr +i [file]`: makes a restored Linux file immutable, even to root. Remove with `chattr -i` before intentional edits.
- `icacls` on Windows: lock down write access tighter than default.
- Re-run the account/group baseline after any fix.

10 BACKUP & RESTORE

- `rsync -avz [src] [dest]`: fast, incremental-friendly copy. (Linux)
- `robocopy [src] [dest] /MIR`: mirrors a directory tree. (Windows)
- `vssadmin list shadows`: check for existing Volume Shadow Copies on Windows before assuming nothing is recoverable.
- Always restore to a new location first if time allows, then swap in, rather than overwriting a possibly-still-needed original.

11 VERIFY

- Re-run the file-integrity baseline against the same paths.
- Confirm the scored service is up and responding correctly.
- Watch for the same account, address, or pattern returning.

12 PATCH MANAGEMENT

- `apt-mark hold [package]`: pin a package version after patching, so an unrelated update does not silently reintroduce a fixed issue.
- `yum versionlock [package]` (RHEL family): same idea.
- Confirm the patched version actually took: `dpkg -l [package]` or `rpm -q [package]` after any update.
- A restart may be required for some patches (kernel, some services) to actually apply; check before assuming a patch is live.

13 WEB & DATABASE HARDENING

- Disable directory listing: `autoindex off;` (Nginx) or `Options -Indexes` (Apache).
- Remove default/sample content shipped with the install; it is the first thing a scanner checks.
- Review database user privileges: an application account should never have superuser rights on the database it uses.
- Confirm the database is not listening on a public interface unless it genuinely needs to be: `bind-address` in MySQL, `listen_addresses` in PostgreSQL.

14 KERNEL & OS-LEVEL HARDENING

- `sysctl net.ipv4.ip_forward`: should be 0 unless the host is genuinely a router.
- `sysctl kernel.randomize_va_space`: confirm ASLR is enabled (value 2).

- Disable unused kernel modules that expand attack surface, if you know the host does not need them.
- Windows: disable SMBv1 (`Disable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol`) and LLMNR via Group Policy, unless still needed.

15 SANITY CHECKS AFTER ANY CHANGE

- Re-test the service from a second session before closing the first, especially after an SSH or RDP config change.
- Confirm the firewall change did not also block something legitimate.
- Time-stamp the change in the tracking template immediately, not after moving to the next task.

16 TASKS VS. SERVICES

- `systemctl list-units --type=service` (Linux) or `Get-Service` (Windows): what is running now.
- Cross-reference against what the host should run.
- Keep a known-good config or backup of anything hardened, in a name and location that is not the obvious first guess.

17 ONE-LINERS

- `find / -mmin -30 -type f 2>/dev/null`: files touched in the last 30 minutes, a fast recent-change sweep. (Linux)
- `Get-ChildItem C:\-Recurse | Where LastWriteTime -gt (Get-Date).AddMinutes(-30)`: same idea. (Windows)
- `systemctl show [service] -p ExecStart`: confirm exactly what a service launches, useful when a unit file looks tampered with.
- `netstat -tulnp | grep LISTEN`: quick view of everything actually listening right now.