CHECKLIST

RHEL SSH Compliance Hardening Checklist

A practical checklist for assessing, tuning, and validating Red Hat Enterprise Linux SSH configuration against compliance requirements before production use.

RHEL SSH Compliance Hardening Checklist

Use this checklist to review, implement, and validate SSH hardening on Red Hat Enterprise Linux before production use. It is designed to help you confirm who can log in, how they authenticate, and which cryptographic and session controls are enforced.

---

1) Policy and scope

- [ ] Identify the compliance standard or internal policy that applies to this host.
- [ ] Confirm whether the system is subject to additional requirements for MFA, logging, session timeout, or restricted administrative access.
- [ ] Define the approved administrative access model:
- [ ] Direct SSH access
- [ ] Jump host / bastion access
- [ ] Role-based access
- [ ] Break-glass access
- [ ] Document any approved exceptions for legacy tools, automation, or vendor support.
- [ ] Verify there is a rollback plan and an alternate access path before making changes.

---

2) Configuration inventory

- [ ] Review the active SSH daemon configuration file.
- [ ] Check for included snippets or drop-in files that may override main settings.
- [ ] Record the current effective values for all security-relevant SSH settings.
- [ ] Confirm the SSH service is managed by standard system service controls.
- [ ] Validate that the configuration source is tracked in configuration management.

---

3) Authentication controls

- [ ] Confirm whether password authentication is allowed by policy.
- [ ] If passwords are not allowed, set and verify password login is disabled for SSH.
- [ ] Confirm direct root login is prohibited or restricted according to policy.
- [ ] Verify only approved authentication methods are enabled.
- [ ] Confirm public key authentication is supported and operational if required.
- [ ] If MFA is required, verify the SSH path integrates with the approved MFA method.
- [ ] Validate that emergency access procedures are documented and tested.

Review questions

- [ ] Can any account still authenticate with a password when policy forbids it?
- [ ] Can root connect directly when policy requires administrative escalation instead?
- [ ] Are there any alternate auth paths exposed through included config files or PAM integration?

---

4) User and group restrictions

- [ ] Define the approved set of users or groups allowed to use SSH.
- [ ] Restrict access using user or group controls where required.
- [ ] Confirm service accounts are not unintentionally permitted interactive SSH access.
- [ ] Verify access rules match the organization?s administration model.
- [ ] Document any break-glass accounts and review their controls.

Review questions

- [ ] Are only the intended operators able to log in?
- [ ] Are shared accounts avoided or tightly controlled?
- [ ] Are temporary access grants reviewed and removed on schedule?

---

5) Cryptographic controls

- [ ] Confirm the approved cipher, key exchange, and MAC requirements for the environment.
- [ ] Verify any explicitly configured crypto settings match the baseline.
- [ ] Check that the selected algorithms are compatible with approved clients and automation.
- [ ] Test connection negotiation from representative client systems.
- [ ] Record any crypto-related exceptions and compensating controls.

Review questions

- [ ] Are weak or deprecated algorithms excluded where policy requires it?

- [ ] Do managed clients still connect successfully with the approved crypto set?
- [ ] Are vendor tools or legacy scripts dependent on older negotiation settings?

---

6) Session and timeout behavior

- [ ] Confirm the required idle timeout or session disconnect policy.
- [ ] Verify any SSH session timeout settings are configured as intended.
- [ ] Test that idle sessions disconnect after the approved period.
- [ ] Confirm banner, login notice, or session warning text is present if required.
- [ ] Validate that terminal sessions behave consistently for interactive and automated use.

Review questions

- [ ] Does the session remain open longer than policy allows?
- [ ] Do automation jobs require an exception for timeouts, and is that exception documented?
- [ ] Are users informed of monitoring or authorized-use conditions at login if required?

---

7) Logging and auditability

- [ ] Confirm successful and failed SSH authentication attempts are logged.
- [ ] Verify the log level is sufficient for audit review and incident response.
- [ ] Ensure logs are forwarded to the approved monitoring or SIEM platform if required.
- [ ] Check that time synchronization is functioning so logs can be correlated accurately.
- [ ] Validate log retention aligns with policy.
- [ ] Confirm administrators know where to find SSH-related events during an investigation.

Review questions

- [ ] Can you trace who logged in, when, and from where?
- [ ] Are failed logins visible and searchable?
- [ ] Are logs protected from unauthorized modification or deletion?

---

8) Operational compatibility checks

- [ ] Test login from all approved administrative client types.
- [ ] Test automation jobs that depend on SSH.
- [ ] Test any jump host or bastion workflow used in production.
- [ ] Verify privileged escalation still works after login.
- [ ] Confirm service owners understand any user experience changes.
- [ ] Identify and remediate workarounds that would bypass the standard access path.

Review questions

- [ ] Did the change break a required admin workflow?
- [ ] Did any team create an informal workaround after testing failed?
- [ ] Is the approved access path still practical for operations?

---

9) Validation before production use

- [ ] Check the SSH daemon configuration for syntax errors.
- [ ] Confirm the service loads the intended configuration successfully.

- [ ] Verify the effective runtime state matches the documented baseline.
- [ ] Test root login behavior against policy.
- [ ] Test password authentication behavior against policy.
- [ ] Test allowed-user or allowed-group restrictions.
- [ ] Test idle timeout and disconnect behavior.
- [ ] Verify logs show successful and failed connection attempts.
- [ ] Confirm approved administrative clients can still connect.
- [ ] Validate any crypto restrictions from the client side, not only the server side.

Minimum evidence to retain

- [ ] Configuration snapshot
- [ ] Validation command output or test results
- [ ] Approval reference or change ticket
- [ ] Exception record, if applicable
- [ ] Rollback confirmation or restoration steps

---

10) Change control and rollback

- [ ] Review the planned SSH changes with the system owner.
- [ ] Confirm the maintenance window and notification path.
- [ ] Ensure a second access method is available during implementation.
- [ ] Take a backup of the current configuration before changes.
- [ ] Apply changes in a controlled sequence.
- [ ] Reload or restart the SSH service only after validation.
- [ ] Re-test access immediately after the change.
- [ ] Restore the previous configuration if validation fails.

Rollback triggers

- [ ] Approved administrators cannot log in.
- [ ] Automation jobs fail unexpectedly.
- [ ] Logging is incomplete or missing.
- [ ] The host cannot be accessed through the documented recovery path.

---

11) Exception handling

- [ ] Document any deviation from the baseline.
- [ ] State the business justification for the exception.
- [ ] Define compensating controls.
- [ ] Set an expiration or review date for the exception.
- [ ] Assign an owner for remediation.
- [ ] Reassess whether the exception can be removed after upgrades or workflow changes.

Common exception examples

- [ ] Legacy automation that still requires password authentication
- [ ] Third-party tools that do not support modern crypto settings
- [ ] Temporary root access for emergency recovery with documented approval

---

12) Final compliance sign-off

- [ ] SSH access matches the documented policy.

- [ ] Authentication methods are approved and tested.
- [ ] Root access is restricted as required.
- [ ] User or group access limits are enforced.
- [ ] Crypto settings are aligned with the baseline.
- [ ] Session controls are working.
- [ ] Logging and monitoring capture the required events.
- [ ] Operational compatibility has been confirmed.
- [ ] Evidence has been stored in the change record.
- [ ] The system owner has approved the final state.

Quick review summary

Area	Status	Notes
Policy alignment		
Authentication		
User/group restrictions		
Cryptography		
Sessions/timeouts		
Logging/auditability		
Compatibility testing		
Rollback readiness		
Exceptions documented		

Recommended evidence bundle

Keep the following with the change record:

- Current and final SSH configuration snapshots
- Validation output from test logins
- Log samples showing success and failure events
- Exception approvals and compensating controls
- Rollback plan and recovery access confirmation

Notes

- Compliance is stronger when you validate behavior, not just file contents.
- Tight settings should never remove your ability to recover access safely.
- Reassess SSH posture whenever authentication, client tooling, or policy requirements change.