lab-01-linux-permissions — Rubric

Lab 01: Linux Permissions & Access Control

Grading Rubric

Total Points: 100
Passing Score: 70/100
Certification Alignment: CompTIA Linux+ (XK0-005), A+ Core 2 (220-1102)

Performance Criteria Matrix

Category 1: Permission Configuration (30 points)

Criterion	Beginning (1)	Developing (2)	Proficient (3)	Advanced (4)
Numeric Notation	Cannot convert between numeric and symbolic	Converts with errors (e.g., 664 instead of 644)	Correctly converts common patterns (755, 644)	Converts complex patterns including special bits (1755, 2755)
chmod Command Execution	Incorrect syntax or does not execute	Executes with assistance or after multiple attempts	Executes correctly on first attempt	Uses both numeric and symbolic notation appropriately
Permission Troubleshooting	Cannot identify permission problems from ls output	Identifies problems but cannot explain why	Correctly identifies and explains permission gaps	Proactively identifies security implications

Scoring: - 1-4 points: Beginning - 5-8 points: Developing - 9-11 points: Proficient - 12 points: Advanced

Weight: 12 points × 2.5 = **30 points**

Category 2: Ownership Management (25 points)

Criterion	Beginning (1)	Developing (2)	Proficient (3)	Advanced (4)
chown Syntax	Incorrect syntax (e.g., `chown :group file`)	Correct syntax but incomplete (missing group)	Correctly changes owner and group	Understands recursive ownership changes and ACLs
Ownership vs. Permissions	Conflates ownership with permissions	Understands difference but struggles to apply	Correctly applies both independently	Explains interaction between ownership and effective permissions
Group Management	Does not use groups effectively	Uses groups but with incorrect permissions	Correctly configures group ownership and permissions	Implements multi-user access with groups

Scoring: - 3-5 points: Beginning - 6-8 points: Developing - 9-11 points: Proficient - 12 points: Advanced

Weight: 12 points × 2.08 = **25 points**

Category 3: Security Awareness (20 points)

Criterion	Beginning (1)	Developing (2)	Proficient (3)	Advanced (4)
Principle of Least Privilege	Uses 777 without considering security	Avoids 777 but applies overly permissive settings (775 on executables)	Applies minimum-necessary permissions	Explains security trade-offs and threat models
World-Writable Detection	Cannot identify security violations	Identifies violations but doesn't understand risk	Correctly identifies and remediates violations	Performs proactive security audits
Access Control Models	No awareness of DAC/MAC concepts	Basic awareness but cannot explain	Explains discretionary access control	Contrasts DAC with MAC/RBAC models

Scoring: - 3-5 points: Beginning - 6-8 points: Developing - 9-11 points: Proficient - 12 points: Advanced

Weight: 12 points × 1.67 = **20 points**

Category 4: Troubleshooting Methodology (15 points)

Criterion	Beginning (1)	Developing (2)	Proficient (3)	Advanced (4)
Diagnostic Process	Random trial-and-error	Follows checklist but skips steps	Systematic investigation (ls, stat, test)	Uses advanced tools (getfacl, lsattr)
Error Interpretation	Cannot interpret “Permission denied” errors	Identifies error but not root cause	Correctly traces error to specific misconfiguration	Predicts errors before execution
Documentation	No documentation of process	Incomplete or disorganized notes	Clear command history with explanations	Provides detailed before/after analysis

Scoring: - 3-5 points: Beginning - 6-8 points: Developing - 9-11 points: Proficient - 12 points: Advanced

Weight: 12 points × 1.25 = **15 points**

Category 5: Reflection & Analysis (10 points)

Criterion	Beginning (1-2)	Developing (3-4)	Proficient (5-6)	Advanced (7-8)
Question 1: chmod 777 Risks	Generic answer (“it’s bad”)	Mentions security but lacks specifics	Identifies attack vectors (tampering, escalation)	References industry standards (OWASP, CIS)
Question 2: chown Syntax	Incorrect or incomplete	Correct but superficial explanation	Explains owner vs. group implications	Discusses ACL alternatives
Question 3: Real-World Scenario	No scenario provided or unrealistic	Plausible but generic example	Specific, realistic scenario with justification	Includes compliance/regulatory context (PCI-DSS, HIPAA)

Scoring: - 3-6 points: Beginning - 7-12 points: Developing - 13-18 points: Proficient - 19-24 points: Advanced

Weight: 24 points × 0.42 = **10 points**

Automated Validation (Pass/Fail Gate)

Criteria: Must pass `test_solution.sh` with exit code 0

- [ ] `run.sh` has 755 permissions
- [ ] `run.sh` owned by appuser:appteam
- [ ] `data/` directory has 775 permissions
- [ ] `data/` directory owned by appuser:appteam
- [ ] No world-writable files exist
- [ ] Application executes successfully as appuser
- [ ] Log file created in data directory

Requirement: All checks must pass to receive credit for technical implementation.

Weight: Pass/Fail gate (required for Proficient or higher)

Overall Score Interpretation

Score Bands

Range	Grade	Interpretation	Certification Readiness
90-100	Advanced (A)	Exceeds expectations; demonstrates mastery with advanced concepts	Ready for Linux+ Domain 2.0
80-89	Proficient (B)	Meets all learning objectives; minor areas for improvement	Ready for A+ Core 2 Domain 2.0
70-79	Developing (C)	Meets basic objectives but needs reinforcement	Needs additional practice
Below 70	Beginning (F)	Does not meet minimum competency; requires remediation	Not ready for certification exam

---

Individual Task Scoring Breakdown

Task 1: Investigation (10 points)

- Correct observation table: 5 points
- Checkpoint Question 1 (two problems identified): 5 points

Task 2: Permissions (10 points)

- Correct chmod command: 5 points
- Checkpoint Question 2 (ownership issue): 5 points

Task 3: Ownership (15 points)

- Correct chown syntax: 7 points
- Verification and testing: 5 points
- Understanding ownership vs. permissions: 3 points

Task 4: Data Directory (15 points)

- Correct ownership configuration: 5 points
- Correct permission configuration (775): 5 points

- Successful application execution: 5 points

Task 5: Security Review (10 points)

- World-writable file check: 5 points
- Checkpoint Question 3 (least privilege): 5 points

Task 6: Syntax Practice (10 points)

- Numeric to symbolic conversions (3 questions × 2 pts): 6 points
- Symbolic to numeric conversions (2 questions × 2 pts): 4 points

Reflection Questions (30 points)

- Question 1 (chmod 777 risks): 10 points
- Question 2 (chown syntax): 8 points
- Question 3 (real-world scenario): 12 points

Total: 100 points

LMS Import Format (CSV)

```
Criterion,Beginning,Developing,Proficient,Advanced,Weight
"Permission Configuration - Numeric Notation",1,2,3,4,2.5
"Permission Configuration - chmod Execution",1,2,3,4,2.5
"Permission Configuration - Troubleshooting",1,2,3,4,2.5
"Ownership Management - chown Syntax",1,2,3,4,2.08
"Ownership Management - Ownership vs Permissions",1,2,3,4,2.08
"Ownership Management - Group Configuration",1,2,3,4,2.08
"Security Awareness - Least Privilege",1,2,3,4,1.67
"Security Awareness - World-Writable Detection",1,2,3,4,1.67
"Security Awareness - Access Control Models",1,2,3,4,1.67
"Troubleshooting - Diagnostic Process",1,2,3,4,1.25
"Troubleshooting - Error Interpretation",1,2,3,4,1.25
"Troubleshooting - Documentation",1,2,3,4,1.25
"Reflection - Question 1",2,4,6,8,0.42
"Reflection - Question 2",2,4,6,8,0.42
"Reflection - Question 3",2,4,6,8,0.42
```

Accommodations & Modifications

For Students with Extended Time

- Allow 60 minutes instead of 45 minutes

- No changes to grading criteria

For Students Requiring Scaffolding

- Provide completed observation table for Task 1
- Reduce reflection questions to 2 instead of 3
- Adjust total to 90 points (scale to 100)

For Advanced Students (Differentiation)

- Require extension activities (sticky bit, ACLs) for Advanced tier
- Add bonus question on SELinux contexts (+10 points extra credit)

Feedback Templates

Beginning Performance Feedback

```
You demonstrated foundational knowledge of Linux permissions but need additional
practice with:

- Converting between numeric and symbolic notation
- Understanding the relationship between ownership and permissions
- Applying security best practices (avoiding 777)

Recommended Resources:

- CompTIA Linux+ Study Guide, Chapter 5
- Linux Journey: File Permissions module
- Practice Labs: Repeat Lab 01 with different scenarios

Next Steps: Schedule remediation session with instructor before attempting Lab 02.
```

Developing Performance Feedback

```
You successfully completed the technical requirements but have opportunities to deepen
understanding:

- Explore advanced permission models (ACLs, SELinux)
- Practice troubleshooting methodology without hints
- Strengthen security analysis in reflection questions

Recommended Resources:

- RHEL Documentation: File Permissions and ACLs
- CompTIA Linux+ Practice Exams: Domain 2.0

Next Steps: You are ready to proceed to Lab 02 (SSH Key Management).
```

Proficient Performance Feedback

```
Excellent work! You demonstrated strong command of:

- Permission and ownership configuration
- Security best practices
- Systematic troubleshooting methodology

To achieve Advanced level:

- Explore special permissions (setuid, setgid, sticky bit)
- Research mandatory access control (SELinux, AppArmor)
- Practice explaining concepts to peers (teaching reinforces mastery)

Next Steps: Proceed to Lab 02 or attempt extension activities.
```

Advanced Performance Feedback

```
Outstanding performance! You demonstrated mastery of Linux permissions and security
concepts. Your work shows certification-ready competency.

Suggested Challenges:

- Configure SELinux contexts for /opt/secure_app
- Implement role-based access control with sudo policies
- Audit a real system for permission misconfigurations

Next Steps: You are ready for CompTIA Linux+ Domain 2.0 exam questions. Proceed to Lab
02.
```

Rubric Revision History

Version	Date	Changes	Rationale
1.0	2026-10-06	Initial release	Aligned to CompTIA Linux+ XK0-005 objectives

---

Rubric Usage Notes for Instructors

Holistic vs. Analytic Grading

Analytic Approach (Recommended for Certification Prep): - Score each criterion individually - Sum weighted scores for total - Provides detailed feedback on specific weaknesses

Holistic Approach (Faster for Large Classes): - Read entire submission - Assign overall tier (Beginning/Developing/Proficient/Advanced) - Use tier midpoint as score (e.g., Proficient = 84/100)

Inter-Rater Reliability

To ensure consistency across multiple graders:

1. Calibrate with 3 sample submissions (one per tier)
2. Discuss edge cases (e.g., “Is 770 acceptable instead of 775?”)
3. Document borderline decisions for reference

Partial Credit Guidelines

- **Command Syntax Errors:** Award 50% credit if concept is correct (e.g., `chmod 745` instead of `755`)
- **Reflection Questions:** Award proportional credit for partially complete answers
- **Automated Validation Failures:** If student demonstrates understanding in reflection but fails technical validation, consider manual review (may award up to 70% if process is documented)

Assessment Metadata

Bloom’s Taxonomy Coverage: - Remember: 15% - Understand: 20% - Apply: 35% - Analyze: 20% - Evaluate: 10%

Certification Objective Mapping: - Linux+ Domain 2.3: 70% - A+ Core 2 Domain 2.6: 30%

Cognitive Load: Moderate (Zone of Proximal Development for intermediate students)

Accessibility: Screen reader compatible (text-based terminal output)

Grading Workflow Checklist

- [ ] Run automated validation: `bash test_solution.sh`
- [ ] Review observation table completeness
- [ ] Check command syntax in student notes
- [ ] Read and score reflection questions using model answers
- [ ] Calculate weighted total score
- [ ] Assign performance tier (Beginning/Developing/Proficient/Advanced)
- [ ] Provide tier-appropriate feedback using templates
- [ ] Record score in LMS
- [ ] Return handout to student within 48 hours

Estimated Grading Time: 10 minutes per submission (with rubric)

Export Formats

This rubric is available in:

- **Markdown** (this file) — Human-readable reference
- **CSV** (above section) — LMS import (Canvas, Blackboard, Moodle)
- **JSON** — Automated grading tools
- **PDF** — Student distribution

Contact & Support

Curriculum Questions: curriculum@labforgestudio.com
Technical Issues: support@labforgestudio.com
Rubric Feedback: rubrics@labforgestudio.com

Document ID: LAB01-RUBRIC-v1.0
Last Updated: 2026-10-06
Next Review Date: 2027-04-01