OSCP LINUX PRACTICE EXAM (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://oscplinux.examzify.com

Copyright © 2026 by Examzify - A Kaluba Technologies Inc. product.

ALL RIGHTS RESERVED.

No part of this book may be reproduced or transferred in any form or by any means, graphic, electronic, or mechanical, including photocopying, recording, web distribution, taping, or by any information storage retrieval system, without the written permission of the author.

Notice: Examzify makes every reasonable effort to obtain accurate, complete, and timely information about this product from reliable sources.

SAMPLE

Table of Contents

Copyright 1

Table of Contents 2

Introduction 3

How to Use This Guide 4

Questions 5

Answers 8

Explanations 10

Next Steps 15

SAMPLE

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

SAMPLE

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

SAMPLE

Questions

SAMPLE

1. When following a symbolic link, whose permissions determine access?

A. The symlink's permissions

B. The directory's permissions

C. The target file's permissions

D. The owner's permissions

2. What is the effect of echo test > file.txt 2>&1?

A. Redirects stdout to file.txt; stderr to screen

B. Redirects both stdout and stderr to file.txt

C. Redirects stdout to file and leaves stderr unchanged

D. Redirects input from file.txt

3. In a find command using -exec, what placeholder represents the current file?

A. {}

B. {}%

C. @

D. @@

4. In a diff output, what does a line starting with '<' denote?

A. It corresponds to a line from the updated file

B. It marks a line to be deleted

C. It indicates a line from the original file

D. It indicates a context line unchanged

5. What is another name for a tar.gz file?

A. tarball

B. archive

C. tar.gz

D. bundle

SAMPLE

6. Process substitution in shells allows feeding the output of multiple commands into another command. Which command demonstrates this technique?

A. It allows feeding the output of multiple commands into another command using <(command)

B. It speeds up command execution by caching results

C. It merges two command outputs into a single file

D. It duplicates a command's output across multiple processes

7. Which option do you add to netstat to resolve hostnames to IP addresses in the output?

A. -p

B. -n

C. -a

D. -t

8. What type of content does /etc primarily store?

A. Global configuration files

B. User data

C. Binary executables

D. System logs

9. In Bash arithmetic, what does echo $((a++)) print?

A. It prints an error.

B. The value after the increment.

C. It prints nothing.

D. The original value of a before the increment.

10. What does the loop for ((i=0;i<5;++i)); do echo $i; done print?

A. It runs forever.

B. It prints 1 through 5.

C. It prints 0 through 4, each on a new line.

D. It prints 0 through 4, on a single line.

SAMPLE

Answers

SAMPLE

1. C
2. B
3. A
4. C
5. A
6. A
7. B
8. A
9. D
10. C

SAMPLE

Explanations

SAMPLE

1. When following a symbolic link, whose permissions determine access?

A. The symlink's permissions

B. The directory's permissions

C. The target file's permissions

D. The owner's permissions

A symbolic link is just a pointer to another file, so the access decision is governed by the permissions of the file it points to, not by the link itself. The kernel ignores the link's own permission bits when following it. To reach the target you also need proper traversal rights on all directories along the path, but once the path resolves, the target file's read, write, and execute bits determine whether you can access it. In short, access is controlled by the target file's permissions.

2. What is the effect of echo test > file.txt 2>&1?

A. Redirects stdout to file.txt; stderr to screen

B. Redirects both stdout and stderr to file.txt

C. Redirects stdout to file and leaves stderr unchanged

D. Redirects input from file.txt

This is about how shell I/O redirection works. The command first sends stdout to file.txt with > file.txt. Then 2>&1 redirects stderr to wherever stdout is currently going. Since stdout is already going to file.txt, stderr is redirected there as well. So both stdout and stderr end up in file.txt. If the order were reversed, the result could be different, but with this left-to-right processing, both streams go to the file.

3. In a find command using -exec, what placeholder represents the current file?

A. {}

B. {}%

C. @

D. @@

The current file is inserted where {} appears in a -exec action. For every file find matches, the command after -exec runs with {} replaced by that file's path. The action must be terminated with \; (or you can use \+ to pass multiple files at once). Example: find . -type f -name ".txt" -exec cat {} \; will print the contents of each matching text file. The other tokens you might have seen aren't recognized as the placeholder in this context, so they won't substitute the current file. Always use {} to reference the file being processed.*

SAMPLE

4. In a diff output, what does a line starting with '<' denote?

A. It corresponds to a line from the updated file

B. It marks a line to be deleted

C. It indicates a line from the original file

D. It indicates a context line unchanged

In a side-by-side diff, the left side represents the original file. A line that starts with a less-than sign is a line that exists in the original file at that spot. It shows what was there before the change, and you compare it to the corresponding line on the right (the updated file) to see what was added, removed, or changed. If there's no matching line on the right, that original line was deleted; if there's a different line on the right, that's what replaced it.

5. What is another name for a tar.gz file?

A. tarball

B. archive

C. tar.gz

D. bundle

A tarball is the best term because it refers to a tar archive, which groups many files into a single file. When that tar archive is compressed with gzip, you get a .tar.gz (or .tgz) file, and people still call it a tarball. The word "archive" is correct but generic, while "tar.gz" is just the filename extension. "Bundle" isn't standard terminology. So the familiar, widely used name for a tar.gz file is a tarball.

6. Process substitution in shells allows feeding the output of multiple commands into another command. Which command demonstrates this technique?

A. It allows feeding the output of multiple commands into another command using <(command)

B. It speeds up command execution by caching results

C. It merges two command outputs into a single file

D. It duplicates a command's output across multiple processes

Process substitution lets you feed the output of commands into another command by making each command's output appear as a file-like path. Using the syntax <(command) creates a read-only handle that the receiving command can open as if it were a file. When you pass multiple such paths to a single command, it can read each command's output as separate inputs without creating temporary files. This is exactly what the example shows: feeding the outputs of multiple commands into one command via <(command). The other choices describe caching, redirecting outputs to a single file, or duplicating output across processes, which do not capture the technique of turning command outputs into file-like inputs for another command.

SAMPLE

7. Which option do you add to netstat to resolve hostnames to IP addresses in the output?

A. -p

B. -n

C. -a

D. -t

Netstat can resolve addresses to hostnames by default, showing friendly names instead of raw IPs. To display the numeric IP addresses in the output, you use the numeric option, which disables name resolution. This makes the addresses appear as IPs directly, which is why this option is the best fit. The other options serve different purposes: one shows the owning process, one lists all sockets, and one restricts output to TCP connections, none of which change how hostnames are resolved.

8. What type of content does /etc primarily store?

A. Global configuration files

B. User data

C. Binary executables

D. System logs

The main idea here is where system-wide settings live in the filesystem. /etc is the place for global configuration files that tell the OS and services how to behave. These are the files that configure startup, networking, services, and application settings, often in plain text. For example, files like /etc/ssh/sshd_config or /etc/fstab define SSH behavior and filesystem mounting, respectively. User data sits elsewhere, typically in /home, binaries live in /bin or /usr/bin, and logs are stored in /var/log. So /etc primarily stores global configuration files.

9. In Bash arithmetic, what does echo $((a++)) print?

A. It prints an error.

B. The value after the increment.

C. It prints nothing.

D. The original value of a before the increment.

Post-increment behavior in Bash arithmetic expansion returns the current value of the variable, then increments it. So echo $((a++)) prints the original value of a before the increment, and a is increased afterward. For example, if a starts as 3, the command prints 3 and a becomes 4 after. This is why the original value is what gets printed, not the incremented value, and there's no error or silence. If you want the incremented value to print, you'd use a pre-increment: echo $((++a)) prints the new value.

SAMPLE

10. What does the loop for ((i=0;i<5;++i)); do echo $i; done print?

A. It runs forever.

B. It prints 1 through 5.

C. It prints 0 through 4, each on a new line.

D. It prints 0 through 4, on a single line.

This uses a C-style for loop in Bash. It starts with i=0, checks i<5 before each iteration, and after each iteration runs the increment ++i. The body prints the current value of i. Since the increment happens after the body, the values seen and printed are 0, 1, 2, 3, 4. When i becomes 5, the condition i<5 is false, so the loop stops. Echo prints each value on its own line, giving 0 through 4, each on a new line.

SAMPLE

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

https://oscplinux.examzify.com

We wish you the very best on your exam journey. You've got this!

SAMPLE