



Comandos Shell

Comparativo Profissional entre Linux, CMD e PowerShell (Básico ao Avançado)

Este documento técnico apresenta uma comparação detalhada dos principais comandos de linha utilizados nos três ambientes mais comuns da TI moderna: **Linux (Bash)**, **CMD** e **PowerShell**. Os comandos são agrupados em três níveis de complexidade — **básico, intermediário e avançado** — com **20 exemplos reais para cada nível**, permitindo uma transição mais fluida entre plataformas e um aprendizado mais prático e produtivo.

Linux Bash

O Bash é a shell padrão na maioria das distribuições Linux.

É um interpretador robusto, scriptável, e amplamente adotado para automação, gerenciamento de servidores e ambientes DevOps.

CMD (Prompt de Comando)

O CMD é uma ferramenta tradicional do Windows com origem no MS-DOS.

Ainda é utilizado para rotinas de administração básica, scripts .bat e diagnósticos rápidos.

PowerShell

O PowerShell é uma shell moderna baseada em objetos, orientada ao .NET. Multiplataforma e altamente scriptável, é a escolha padrão para administração moderna no Windows e está ganhando espaço em ambientes Linux.



Tabelas Comparativas por Nível

Nível Básico – Navegação, arquivos e ambiente

Ação	Linux (Bash)	CMD	PowerShell
1. Mostrar diretório atual	pwd	cd ou echo %cd%	Get-Location
2. Listar arquivos	ls	dir	Get-ChildItem
3. Listar arquivos detalhados	ls -l	dir	`Get-ChildItem
4. Entrar em pasta	cd pasta	cd pasta	cd pasta
5. Voltar uma pasta	cd ..	cd ..	cd ..
6. Voltar ao diretório raiz	cd /	cd \	cd \
7. Criar pasta	mkdir nova	mkdir nova	mkdir nova
8. Criar arquivo vazio	touch teste.txt	echo. > teste.txt	New-Item teste.txt
9. Ver conteúdo de arquivo	cat teste.txt	type teste.txt	Get-Content teste.txt
10. Renomear arquivo	mv a.txt b.txt	rename a.txt b.txt	Rename-Item a.txt b.txt
11. Apagar arquivo	rm a.txt	del a.txt	Remove-Item a.txt
12. Apagar pasta	rm -r pasta/	rmdir /s pasta	Remove-Item pasta -Recurse
13. Copiar arquivo	cp a.txt b.txt	copy a.txt b.txt	Copy-Item a.txt b.txt
14. Mover arquivo	mv a.txt pasta/	move a.txt pasta\	Move-Item a.txt pasta\
15. Limpar terminal	clear	cls	Clear-Host
16. Mostrar data	date	date /t	Get-Date
17. Mostrar hora	date / time	time /t	Get-Date
18. Sair da shell	exit	exit	exit
19. Mostrar usuário	whoami	whoami	whoami
20. Mostrar hostname	hostname	hostname	hostname

Nível Intermediário – Diagnóstico, manipulação e rede

Ação	Linux (Bash)	CMD	PowerShell
1. Exibir IP	ip a ou ifconfig	ipconfig	ipconfig
2. Testar conexão (ping)	ping google.com	ping google.com	Test-Connection google.com
3. Ver processos	ps aux	tasklist	Get-Process
4. Matar processo por PID	kill 1234	taskkill /PID 1234 /F	Stop-Process -Id 1234
5. Ver uso de disco	df -h	wmic logicaldisk get size	Get-PSDrive
6. Ver variáveis de ambiente	env	set	Get-ChildItem Env:
7. Criar variável temporária	VAR=valor	set VAR=valor	\$VAR = "valor"
8. Ver portas abertas	netstat -tuln	netstat -an	Get-NetTCPConnection
9. Criar script simples	nano script.sh	notepad script.bat	notepad script.ps1
10. Executar script	./script.sh	script.bat	.\script.ps1
11. Redirecionar saída	ls > lista.txt	dir > lista.txt	Get-ChildItem > lista.txt
12. Concatenar arquivos	cat a.txt b.txt > c.txt	copy a.txt + b.txt c.txt	Get-Content a.txt, b.txt > c.txt
13. Procurar em arquivo	grep texto arquivo.txt	find "texto" arquivo.txt	Select-String texto arquivo.txt
14. Contar linhas/bytes	wc -l arquivo.txt	<i>não disponível nativamente</i>	(Get-Content arquivo.txt).Length
15. Histórico de comandos	history	<i>não disponível</i>	Get-History
16. Mostrar sistema	uname -a	systeminfo	Get-ComputerInfo
17. Ver serviços ativos	systemctl list-units	sc query	Get-Service
18. Agendar tarefa	crontab -e	schtasks /create ...	Register-ScheduledTask
19. Ver permissões	ls -l	icacls arquivo.txt	Get-Acl arquivo.txt
20. Mostrar usuários do sistema	cut -d: -f1 /etc/passwd	net user	Get-LocalUser



Nível Avançado – Scripts, automação e administração

Ação	Linux (Bash)	CMD	PowerShell
1. Executar como root	sudo comando	runas /user:Admin cmd	Start-Process powershell -Verb RunAs
2. Compactar arquivos (zip)	zip arquivo.zip a.txt	powershell Compress-Archive	Compress-Archive
3. Extrair arquivos	unzip arquivo.zip	tar -xf via PS	Expand-Archive
4. Monitorar recursos	top / htop	<i>não disponível</i>	`Get-Process
5. Ver logs do sistema	journalctl	eventvwr	Get-EventLog
6. Executar múltiplos comandos	cmd1 && cmd2	cmd1 && cmd2	; ou &&
7. Executar em segundo plano	comando &	<i>não aplicável</i>	Start-Job
8. Loop em arquivos	for i in *.txt; do ...; done	for %i in (*.txt) do ...	foreach (\$i in *.txt) { ... }
9. Ler linha por linha	while read l; do ...; done	for /F %%i in (arquivo.txt) do ...	`Get-Content arquivo.txt
10. Automatizar backup	rsync -av pasta/ destino/	<i>uso limitado</i>	Copy-Item -Recurse
11. Ver tarefas agendadas	crontab -l	schtasks /query	Get-ScheduledTask
12. Executar scripts em lote	bash script.sh	call script.bat	& "script.ps1"
13. Alterar permissões	chmod 755 arquivo	icacls arquivo /grant ...	Set-Acl
14. Listar usuários com permissão	getfacl arquivo	icacls arquivo.txt	Get-Acl
15. Ver uso de memória	free -h	<i>não disponível</i>	Get-Counter "\\Memory\\Available MBytes"
16. Usar alias para comando	alias l='ls -la'	<i>não nativo</i>	Set-Alias
17. Baixar arquivo da web	wget URL / curl URL	<i>precisa do PowerShell</i>	Invoke-WebRequest
18. Script com parâmetros	\$1, \$2	%1, %2	\$args[0], \$args[1]
19. Criação de funções	function nome() { }	<i>limitado</i>	function Nome { }
20. Autenticar via SSH	ssh usuario@host	<i>precisa do OpenSSH</i>	ssh usuario@host

Dominar os comandos básicos, intermediários e avançados nos ambientes de linha de comando — seja no **Linux Bash**, no tradicional **CMD** ou no moderno **PowerShell** — é uma habilidade essencial para qualquer profissional de TI. Este conhecimento permite:

- **Automatizar tarefas repetitivas**
- **Administrar sistemas locais e remotos**
- **Desenvolver scripts reutilizáveis e robustos**
- **Transitar com segurança entre diferentes plataformas**

Com a crescente convergência entre sistemas operacionais (como WSL no Windows e PowerShell no Linux), **ter fluência em múltiplas shells** se tornou não apenas uma vantagem, mas uma exigência.

EducaCiência FastCode para a comunidade