Cheatography

Linux General Purpose Cheat Sheet Cheat Sheet

by r_tempest via cheatography.com/208223/cs/44554/

Firewalld

Show all zones	firewall-cmd --get-zones
Show all zones and settings	firewall-cmd --list-all-zones
Allow port at zone	firewall-cmd --permanent --zone= *<zone>* --add-port=*<port>*/*<protocol>*
Reload settings	firewall-cmd --reload
Allow port from specific network	firewall-cmd --permanent --zone= *<zone>* --add-rich-rule='rule family="ipv4" source address="*<network>*" port protocol="*<protocol>*" port="*<port>*" accept'
Remove port from zone	firewall-cmd --permanent --zone= *<zone>* --remove-port=*<port>*/*<protocol>*
Remove port from specific network	firewall-cmd --permanent --remove-rich-rule='rule family="ipv4" source address="*<network>*" port protocol="*<protocol>*" port="*<port>*" accept
Allow traffic between interface	firewall-cmd --permanent --zone= *<zone>* --add-rich-rule='rule family="ipv4" source address="*<interface>*" accept'

After making any modifications to firewall rules, it is necessary to reload the rules for the changes to take effect.

File, Folders and Permissions management

Remove file	rm -f *<file>*
Remove folder and its contents	rm -fr *<folder>*
Recursively remove files matching regex patterns	find *<directory>* -regex '*<extendend_regex>*' -exec rm -f {} \;
Recursively remove files by extension	find *<directory>* -iname '*.*<extension>*' -exec rm -f {} \;
Recursively remove files older than N days	find *<directory>* -type f -mtime +*<number_of_days>* -exec rm -f {} \;
Cut or rename files or folders	mv *<sources files/folder> <destination folder/file>*
Copy files	cp *<source files> <destination folder>*
Copy folders	cp -r *<folders> <destination folder>*
Change permission of file/folder	chmod *<owner_octal><group_octal><other_octal> <file/folder>*
Change permission of folder (Recursively)	chmod -R *<owner_octal><group_octal><other_octal> <folder>*
Change owner user and group of file/folder	chown *<user>*:*<group> <file/folder>*
Change owner user and group of folder (Recursively)	chown -R *<user>*:*<group> <folder>*

[1] Check the octal permissions system at **Octal Permissions** session

[2] You can set the maximum search depth when using `find` to remove files with the `-maxdepth N` option. This helps prevent unwanted recursive deletions. For example: `find <di rec tor y> -maxdepth 1 ....`

Networking

Check port communication (telnet)	telnet *<address>* *<port>*
Check port communication (netcat)	nc -zv *<address>* *<port>*
Open dummy port (netcat)	nc -lp *<port>*
Open TLS/SSL Port (openssl)	openssl s_server -accept *<port>* -cert *<certificate_file>* -key *<key_file>*
Test connection to TLS/SSL Port(openssl)	openssl s_client -connect *<address>*:*<port>*
Show TCP IPV4 Listening ports	ss -ltnp4 | grep *<port>*
Show TCP IPV6 Listening ports	ss -ltpn6 | grep *<port>*
Show all connection and ports	ss -putona
Show interfaces	ip a
Show routes	ip route
Show DNS properties	nslookup *<DNS>*
Show ip information	dig -x *<ip>* +all
Show route to destination	traceroute *<ip>*

Octal Permissions	
Read	4
Write	2
Execute	1

For multiple permissions, just add the octal values. Example: read(4) and execute(1) = 5.

More about it: Linux file permissions explained

Remote Access	
Connect via SSH using password	ssh -p *<port> <username>@<hostname/ip>*
Copy file from local to remote using SCP	scp -P *<port> <local_file_path> <username>@<hostname/ip>:<destination folder/file>*
Copy file from remote to local using SCP	scp -P *<port> <username>@<hostname/ip>:<file_path> <destination folder/file>*
Copy file from local to remote using rsync	rsync -avz -e "ssh -p *<port>*" *<local file/folder> <username>@<hostname/ip>:<destination folder/file>*
Copy file from remote to local using rsync	rsync -avz -e "ssh -p *<port>*" *<username>@<hostname/ip>:<folder/file> <destination folder/file>*

[1] Prefer using rsync instead of SCP since it will be deprecated soon. Check references for more information.

[2] For key authentication with SSH, SCP or RSYNC add a `-i` **`<ke y_f ile>`** after the SSH or SCP command. I.e: `ssh -i /home/ foo /k e y.rsa ....`

References

OpenSSH SCP deprecation in RHEL 9

User Management	
Add user	useradd -s *<shell>* -d *<user_home>* -m -U *<username>*
Add User to Group	usermod -aG *<group_name> <user_name>*
Remove user from group	deluser *<group_name> <user_name>*
Delete User	userdel -f -r *<user_name>*
Add group	groupadd *<group_name>*
Delete group	groupdel *<group_name>*
List user groups	groups *<user_name>*