Site Reliability Engineer (SRE) Interview Preparation Guide

This repository is an attempt to consolidate useful resources for Site Reliability Engineer (SRE) interview preparation.

Contributing

Please take a look at the [contribution guidelines](CONTRIBUTING.md) first. Contributions are always welcome!

Basics

- [ ] Simple: [What happens when you type in 'www.cnn.com' in your browser?](https://syedali.net/2013/08/18/what-happens-when-you-type-in-www-cnn-com-in-your-browser)
- [ ] Detailed: [What happens when you type google.com into your browser's address box and press enter?](https://github.com/alex/what-happens-when)

Linux

- [ ] [What every SRE should know about GNU/Linux shell related internals: file descriptors, pipes, terminals, user sessions, process groups and daemons](https://biriukov.dev/docs/fd-pipe-session-terminal/0-sre-should-know-about-gnu-linux-shell-related-internals-file-descriptors-pipes-terminals-user-sessions-process-groups-and-daemons)

Boot Process

- [ ] [An introduction to the Linux boot and startup processes](https://opensource.com/article/17/2/linux-boot-and-startup) - [ ] [What happens when we turn on computer?](https://www.cdn.geeksforgeeks.org/what-happens-when-we-turn-on-computer) - [ ] [What happens when we turn on computer?](https://leetcode.com/discuss/interview-question/125107/What-happens-when-we-turn-on-computer)
- [ ] [From Power up to login prompt](http://www.scott-a-s.com/files/linux_boot.pdfh)

Filesystem

- [ ] [Understanding Inodes](https://syedali.net/2015/02/08/understanding-inodes)
- [ ] [Understand UNIX / Linux Inodes Basics with Examples](https://www.thegeekstuff.com/2012/01/linux-inodes)
- [ ] [Understanding proc filesystem](https://syedali.net/2013/08/20/understanding-proc-filesystem) - [ ] [Common Mount Options](https://syedali.net/2015/01/06/common-mount-options) - [ ] [Understanding Linux filesystems: ext4 and beyond](https://opensource.com/article/18/4/ext4-filesystem)

Kernel

- [ ] [Explain the basics of Linux kernel](http://learnlinuxconcepts.blogspot.com/2014/03/explain-basics-of-linux-kernel.html) - [ ] [Kernel Space and User Space](http://learnlinuxconcepts.blogspot.com/2014/02/kernel-space-and-user-space.html) - [ ] [Linux Kernel Process Management](http://learnlinuxconcepts.blogspot.com/2014/03/process-management.html) - [ ] [Linux Addressing](http://learnlinuxconcepts.blogspot.com/2014/02/linux-addressing.html)
- [ ] [Linux Kernel Memory Management](http://learnlinuxconcepts.blogspot.com/2014/02/linux-memory-management.html)
- [ ] [STACK AND HEAP](http://learnlinuxconcepts.blogspot.com/2014/02/stack-and-heap.html)
- [ ] [Paging and Segmentation](http://learnlinuxconcepts.blogspot.com/2014/02/paging-and-segmentation.html)
- [ ] [Linux Kernel System Calls](http://learnlinuxconcepts.blogspot.com/2014/02/system-calls.html) - [ ] [The Virtual Filesystem](http://learnlinuxconcepts.blogspot.com/2014/10/the-virtual-filesystem.html) - [ ] [Concurrency and Race Conditions](http://learnlinuxconcepts.blogspot.com/2014/07/concurrency-and-race-conditions.html)
- [ ] [Memory Leak](https://stackoverflow.com/questions/312069/the-best-memory-leak-definition) - [ ] [What is a kernel Panic?](http://learnlinuxconcepts.blogspot.com/2014/07/what-is-kernel-panic.html) - [ ] [Book about the linux kernel](https://0xax.gitbooks.io/linux-insides/content)

Troubleshooting

- [ ] [Linux troubleshooting tools](https://syedali.net/2013/08/20/linux-troubleshooting-tools)
- [ ] [Linux Performance Analysis in 60,000 Milliseconds](https://medium.com/netflix-techblog/linux-performance-analysis-in-60-000-milliseconds-accc10403c55)
- [ ] [strace](https://www.dedoimedo.com/computers/strace.html)
- [ ] [lsof](https://www.dedoimedo.com/computers/lsof.html)
- [ ] [Linux system debugging](https://www.dedoimedo.com/computers/linux-system-debugging-super.html)
- [ ] [SaaS where users can test their Linux troubleshooting skills](https://sadservers.com)

Networking

- [ ] [The Internet explained from first principles](https://explained-from-first-principles.com/internet)

- [ ] [Network protocols for anyone who knows a programming language](https://www.destroyallsoftware.com/compendium/network-protocols?share_key=97d3ba4c24d21147)
- [ ] [Introduction to Linux interfaces for virtual networking](https://developers.redhat.com/blog/2018/10/22/introduction-to-linux-interfaces-for-virtual-networking)
- [ ] [Multi-tier load-balancing with Linux](https://vincent.bernat.ch/en/blog/2018-multi-tier-loadbalancer)
- [ ] [Introduction to modern network load balancing and proxying](https://blog.envoyproxy.io/introduction-to-modern-network-load-balancing-and-proxying-a57f6ff80236)
- [ ] [Load Balancing Algorithms](https://syedali.net/2013/08/22/load-balancing-algorithms)

Containers

- [ ] [Introduction to Docker and Containers](http://container.training/intro-selfpaced.yml.html)
- [ ] [Containers Patterns](https://l0rd.github.io/containerspatterns)
- [ ] [Docker Container Anti Patterns](https://blog.couchbase.com/docker-container-anti-patterns/)
- [ ] [Anti-Patterns When Building Container Images](https://jpetazzo.github.io/2021/11/30/docker-build-container-images-antipatterns

)

Kubernetes

- [ ] [Deploying and Scaling Microservices with Docker and Kubernetes](http://container.training/kube-selfpaced.yml.html)
- [ ] [Demystifying the Kubernetes Iceberg](https://asankov.dev/blog/2022/05/15/demystifying-the-kubernetes-iceberg-part-1) - [ ] [What happens when ... Kubernetes edition!](https://github.com/jamiehannaford/what-happens-when-k8s/blob/master/README.md)
- [ ] [Kubernetes Production Patterns](https://github.com/gravitational/workshop/blob/master/k8sprod.md) - [ ] [Kubernetes production best practices](https://learnk8s.io/production-best-practices) - [ ] [A Guide to the Kubernetes Networking Model](https://sookocheff.com/post/kubernetes/understanding-kubernetes-networking-model)
- [ ] [47 Things To Become a Kubernetes Expert](https://ymmt2005.hatenablog.com/entry/k8s-things)
- [ ] [Kubernetes Best Practices 101](https://github.com/diegolnasc/kubernetes-best-practices)
- [ ] [15 Kubernetes Best Practices Every Developer Should Know](https://spacelift.io/blog/kubernetes-best-practices)
- [ ] [THE KUBERNETES NETWORKING GUIDE](https://www.tkng.io)

- [ ] [The life of a DNS query in Kubernetes](https://www.nslookup.io/learning/the-life-of-a-dns-query-in-kubernetes

)

Infrastructure as code / Configuration management

- [ ] [Terraform](https://learn.hashicorp.com/terraform)
- [ ] [A Comprehensive Guide to Terraform](https://blog.gruntwork.io/a-comprehensive-guide-to-terraform-b3d32832baca)
- [ ] [Ansible](https://github.com/leucos/ansible-tuto)
- [ ] [Getting Started With Terraform on AWS](https://spacelift.io/blog/terraform-tutorial)
- [ ] [Google Cloud: Best practices for using Terraform](https://cloud.google.com/docs/terraform/best-practices-for-terraform)

Databases

- [ ] [Things You Should Know About Databases](https://architecturenotes.co/things-you-should-know-about-databases)
- [ ] [7 Database Paradigms](https://youtu.be/W2Z7fbCLSTw)
- [ ] [CAP theorem](https://en.wikipedia.org/wiki/CAP_theorem)
- [ ] [Evolutionary Database Design](https://martinfowler.com/articles/evodb.html)
- [ ] [ACID vs BASE in Databases](https://medium.com/geekculture/acid-vs-base-in-databases-1bcad774da26
) - [ ] [Understanding Database Sharding](https://www.digitalocean.com/community/tutorials/understanding-database-sharding)
- [ ] [Database Replication](https://galeracluster.com/library/documentation/tech-desc-introduction.html#database-replication)
- [ ] [SQL vs. NoSQL Database: When to Use, How to Choose](https://towardsdatascience.com/datastore-choices-sql-vs-nosql-database-ebec24d56106)
- [ ] [How do database indexes work?](https://planetscale.com/blog/how-do-database-indexes-work)
- [ ] [Redis Explained](https://architecturenotes.co/redis)
- [ ] [Database Sharding Explained](https://architecturenotes.co/database-sharding-explained)

CI/CD

- [ ] [7 Pipeline Design Patterns for Continuous Delivery](https://www.singlestoneconsulting.com/blog/7-pipeline-design-patterns-for-continuous-delivery)
- [ ] [CI/CD patterns](https://continuousdelivery.com/implementing/patterns)
- [ ] [Six Strategies for Application

Deployment](https://thenewstack.io/deployment-strategies)

Clouds

- [ ] [The Open Guide to Amazon Web Services](https://github.com/open-guides/og-aws)
- [ ] [Learning Azure](https://docs.microsoft.com/en-us/learn/azure/)
- [ ] [Hands-On Training with GCP](https://cloud.google.com/training/badges)

Programming

Python

- [ ] [Python Basics](https://pythonbasics.org/)
- [ ] [Python For Everyone](https://www.py4e.com/)
- [ ] [Complete Python Tutorial](https://www.scaler.com/topics/python/)

Go (Golang)

- [ ] [A tour of Go](https://tour.golang.org)
- [ ] [Go by Example](https://gobyexample.com)
- [ ] [Go Tutorials & Examples](https://gosamples.dev)
- [ ] [Learn Go with Tests](https://quii.gitbook.io/learn-go-with-tests/)
- [ ] [Getting up and running with Go](http://www.golangprograms.com)
- [ ] [Effective Go](https://golang.org/doc/effective_go.html)
- [ ] [Go Design Patterns](https://github.com/tmrts/go-patterns)
- [ ] [Go Memory Management](https://povilasv.me/go-memory-management)
- [ ] [Style Guide](https://google.github.io/styleguide/go/guide)
- [ ] [Style Decisions](https://google.github.io/styleguide/go/decisions) - [ ] [Best Practices](https://google.github.io/styleguide/go/best-practices) - [ ] [50 Shades of Go: Traps, Gotchas, and Common Mistakes for New Golang Devs](https://devs.cloudimmunity.com/gotchas-and-common-mistakes-in-go-golang)

Big O Notation, Algorithms and Data Structures

- [ ] [AlgoExperts](https://www.algoexpert.io)
- [ ] [Hacking a Google Interview – Handout 1](http://courses.csail.mit.edu/iap/interview/Hacking_a_Google_Interview_Handout_1.pdf) - [ ] [Hacking a Google Interview – Handout 2](http://courses.csail.mit.edu/iap/interview/Hacking_a_Google_Interview_Handout_2.pdf)
- [ ] [Hacking a Google Interview – Handout 3](http://courses.csail.mit.edu/iap/interview/Hacking_a_Google_Interview_Handout_3.pdf)

System design

- [ ] [SystemsExpert course from AlgoExpert](https://www.algoexpert.io/se/product)
- [ ] [System Design 101](https://github.com/ByteByteGoHq/system-design-101) - [ ] [Grokking the System Design

Interview](https://www.educative.io/collection/5668639101419520/5649050225344512)
- [ ] [The System Design Primer](https://github.com/donnemartin/system-design-primer)
- [ ] [Crack the System Design Interview](https://www.puncsky.com/blog/2016/02/14/crack-the-system-design-interview) - [ ] [System design interview for IT companies](https://github.com/checkcheckzz/system-design-interview) - [ ] [Web Architecture 101](https://medium.com/storyblocks-engineering/web-architecture-101-a3224e126947)
- [ ] [What's in a Production Web Application?](https://web.archive.org/web/20210106095747/http://stephenmann.io/post/whats-in-a-production-web-application)
- [ ] [Distributed systems](http://book.mixu.net/distsys/single-page.html)
- [ ] [Failover](https://blog.alexewerlof.com/p/failover)
- [ ] [Monoliths, Service Architecture, and Microservices](https://architecturenotes.co/granularity-of-systems)

System design examples

- [ ] [Designing WhatsApp](http://highscalability.com/blog/2022/1/3/designing-whatsapp.html)
- [ ] [Designing Uber](http://highscalability.com/blog/2022/1/25/designing-uber.html) - [ ] [Designing Tinder](http://highscalability.com/blog/2022/1/17/designing-tinder.html) - [ ] [Designing Instagram](http://highscalability.com/blog/2022/1/11/designing-instagram.html) - [ ] [Designing Netflix](http://highscalability.com/blog/2021/12/13/designing-netflix.html)

Monitoring

- [ ] [SLOs & You: A Guide To Service Level Objectives](https://www.circonus.com/2018/07/a-guide-to-service-level-objectives) - [ ] [Setting up Service Monitoring — The Why's and What's](https://amitosh.medium.com/the-whys-and-what-s-of-setting-up-service-monitoring-cc1c165ee088)
- [ ] [How NOT to Measure Latency](https://youtu.be/lJ8ydIuPFeU)
- [ ] [The four Golden Signals of Kubernetes monitoring](https://sysdig.com/blog/golden-signals-kubernetes)

Prometheus

- [ ] [Introduction to Prometheus](https://training.promlabs.com/training/introduction-to-prometheus/training-overview/introduction)
- [ ] [Prometheus Relabeling Training](https://training.promlabs.com/training/relabeling/training-overview/prerequisites)
- [ ] [Avoid These 6 Mistakes When Getting Started With Prometheus](https://promlabs.com/blog/2022/12/11/avoid-these-6-mistakes-when-getting-started-with-prometheus)
- [ ] [A Deep Dive Into the Four Types of Prometheus

Metrics](https://www.timescale.com/blog/four-types-prometheus-metrics-to-collect) - [ ] [How Prometheus Querying Works](https://www.timescale.com/blog/how-prometheus-querying-works-and-why-you-should-care)
- [ ] [PromQL Cheat Sheet](https://promlabs.com/promql-cheat-sheet)

Processes

- [ ] [The practical guide to incident management](https://incident.io/guide)
- [ ] [Incident Response](https://response.pagerduty.com)
- [ ] [Postmortems](https://postmortems.pagerduty.com)
- [ ] [Runbooks](https://www.transposit.com/devops-blog/itsm/what-makes-a-good-runbook)
- [ ] [Identifying and tracking toil using SRE principles](https://cloud.google.com/blog/products/management-tools/identifying-and-tracking-toil-using-sre-principles)
- [ ] [Building SRE from Scratch](https://medium.com/ibm-garage/building-sre-from-scratch-485e23985bbd) - [ ] [SRE at Google: Our complete list of CRE life lessons](https://cloud.google.com/blog/products/devops-sre/sre-at-google-our-complete-list-of-cre-life-lessons)
- [ ] [Incident Management vs. Incident Response - What's the Difference?](https://rootly.io/blog/incident-management-vs-incident-response-what-s-the-difference)
- [ ] [Practical Guide to SRE: Using SLOs to Increase Reliability](https://rootly.io/blog/practical-guide-to-sre-using-slos-to-increase-reliability) - [ ] [Practical Guide to SRE: Automating On-Call](https://rootly.io/blog/practical-guide-to-sre-automating-on-call) - [ ] [Going from Zero to SRE](https://www.squadcast.com/blog/going-from-zero-to-sre) - [ ] [An Incident Command Training Handbook](https://blog.danslimmon.com/2019/06/24/an-incident-command-training-handbook)
- [ ] [Howie guide to post-incident investigations](https://www.jeli.io/howie/welcome)
- [ ] [Rundown of LinkedIn's SRE practices](https://www.srepath.com/rundown-of-linkedins-sre-practices) - [ ] [Rundown of Uber's SRE practice](https://www.srepath.com/rundown-of-uber-sre-practice)
- [ ] [SRE in the Real World](https://blog.relyabilit.ie/sre-in-the-real-world) - [ ] [SRE Engagement Models](https://certomodo.substack.com/p/sre-engagement-models) - [ ] [SRE Checklist](https://github.com/bregman-arie/sre-checklist)
- [ ] [Why bother with SLI and SLO?](https://blog.alexewerlof.com/p/why-bother-with-sli-and-slo)
- [ ] [The System Resiliency Pyramid](https://www.codereliant.io/the-system-resiliency-pyramid)

Resume

- [ ] [SRE Complete Resume Writing

Guide](https://rootly.com/blog/sre-complete-resume-writing-guide)

Interview

SRE interview process

- [ ] [How to hire talent](https://syedali.net/2014/04/01/how-to-hire-talent)
- [ ] [Recruitment process for a Google job (SRE, Site Reliability Engineer)](https://web.archive.org/web/20220328124724/http://lambda-startup.com/recruitment-process-for-a-google-job-sre-site-reliability-engineer)

Interview Questions

- [ ] [A collection of questions to practice with for SRE interviews](https://github.com/michael-kehoe/sre-interview)
- [ ] [SRE Interview Questions](https://syedali.net/engineer-interview-questions) - [ ] [Sysadmin Test Questions](https://github.com/trimstray/test-your-sysadmin-skills) - [ ] [Kubernetes job interview questions](https://enterprisersproject.com/article/2019/2/kubernetes-job-interview-questions how-prepare)
- [ ] [DevOps Guide](https://github.com/Tikam02/DevOps-Guide)
- [ ] [Questions I ask in SRE interviews](https://dev.to/logan/questions-i-ask-in-sre-interviews-a9j)
- [ ] [DevOps Roadmap: Learn to become a DevOps Engineer or SRE](https://roadmap.sh/devops)
- [ ] [The Must-Know Terraform Interview Questions](https://devopsknowledge.hashnode.dev/the-must-know-terraform-interview-quest ions)

Blogposts

- [ ] [SRE Interviews in Silicon Valley](http://blog.marc-seeger.de/2015/05/01/sre-interviews-in-silicon-valley)
- [ ] [Preparing the SRE interview](https://blog.balthazar-rouberol.com/preparing-the-sre-interview) - [ ] [How to Get Into SRE](https://blog.alicegoldfuss.com/how-to-get-into-sre) - [ ] [My Job Interview at Google](https://catonmat.net/my-job-interview-at-google) - [ ] [Path to Site Reliability Management](https://danrl.com/srm)
- [ ] [Becoming a Site Reliability Engineer](https://www.tik.dev/blog/becoming-an-sre)
- [ ] [How I get a job at Google as SRE](https://fabrizio2210.medium.com/how-i-get-a-job-at-google-as-sre-83d44aef7859)
- [ ] [Become A DevOps Engineer in 2023: [Detailed Guide]](https://devopscube.com/become-devops-engineer)
- [ ] [How to Get an SRE Role](https://certomodo.substack.com/p/how-to-get-an-sre-role)

Books

SRE books

- [ ] [Site Reliability Engineering](https://sre.google/sre-book/table-of-contents) - [ ] [The Site Reliability Workbook](https://sre.google/workbook/table-of-contents) - [ ] [Seeking SRE](https://books.google.ru/books?id=tmhqDwAAQBAJ) - [ ] [Building Secure and Reliable Systems](https://sre.google/books/building-secure-reliable-systems)
- [ ] [Implementing Service Level Objectives](https://learning.oreilly.com/library/view/implementing-service-level/9781492076803)

Linux

- [ ] [Linux Kernel Development (3rd Edition)](https://www.amazon.com/Linux-Kernel-Development-Robert-Love/dp/0672329468) - [ ] [UNIX and Linux System Administration Handbook (5th Edition)](https://www.amazon.com/UNIX-Linux-System-Administration-Handbook/dp/0134277554)
- [ ] [Linux Pocket Guide, 3rd Edition](http://shop.oreilly.com/product/0636920040927.do)

Networking

- [ ] [TCP/IP Illustrated, Volume 1](https://www.amazon.com/TCP-Illustrated-Protocols-Addison-Wesley-Professional/dp/0321336313)

Troubleshooting and Performance

- [ ] [Systems Performance: Enterprise and the Cloud](https://www.amazon.com/Systems-Performance-Enterprise-Brendan-Gregg/dp/0133390098)
- [ ] [Systems Performance, 2nd Edition](https://www.informit.com/store/systems-performance-9780136820154?ranMID=24808)

Courses

- [ ] [Site Reliability Engineering: Measuring and Managing Reliability](https://www.coursera.org/learn/site-reliability-engineering-slos) - [ ] [School of SRE](https://linkedin.github.io/school-of-sre)