



SITE RELIABILITY ENGINEERING

Um guia completo sobre boas práticas para SREs.

Autores

Este livro contou com quatro autores ao total. O conteúdo a seguir foi escrito em parceria por:



Pedro Tebaldi
Gestor BI



Dario B. Bestetti
CTO



Aldry Rocha
Gestora marketing

e sua diagramação realizada pela:



Camila Fernandes
Designer

Introdução

Devido ao aumento da complexidade dos sites, aplicativos e infraestruturas em nuvem, ter um cuidado especial com a arquitetura de um sistema e com o seu desenvolvimento é apenas uma pequena parte de todo o processo que envolve a entrega de soluções confiáveis para o usuário final. Após o lançamento de um novo aplicativo, é necessário atualizar, implementar novas funcionalidades e corrigir possíveis bugs.

De acordo com dados de mercado, estima-se que de 40% a 90% do custo total do produto seja gasto após o lançamento, envolvendo atividades de operação e manutenção. Além disso, algumas perguntas acabam surgindo neste momento, como: "será que meu sistema vai escalar?", "se eu aumentar a quantidade de usuários, será que tudo vai continuar funcionando da mesma forma?" ou "será que meu sistema crítico vai estar sempre no ar quando o usuário precisar acessar?".

E é aí que deve entrar em cena um profissional que entenda de infraestrutura e operações. O profissional que atua na engenharia de confiabilidade do ambiente é o Site Reliability Engineer (SRE). Sua função é aprimorar o design e a operação dos sistemas, assegurando que todos os serviços online operem de forma contínua, confiável e ágil. Ou seja, a confiabilidade torna-se o foco principal da sua atuação.

Como principal habilidade profissional de um SRE, destaca-se a capacidade de identificar quais são os processos que devem ser otimizados, trazendo, assim, maior automatização e desempenho ao software, removendo do sistema trabalhos manuais e repetitivos. Ao contrário dos administradores de sistemas, normalmente avessos ao risco, os engenheiros de confiabilidade adotam o risco de maneira controlada.



#1
princípio

A funcionalidade mais importante de qualquer sistema é a **confiabilidade**.

Sumário

Surgimento do SRE	07
Fundamentos do SRE	10
Atribuições do profissional	13
Métricas para SRE	16
Rotina do SRE	25
Quão confiável minha empresa deve ser?	29
Conclusão	31

Surgimento do SRE

O termo Site Reliability Engineering começou a ser usado em 2003, por meio de Ben Treynor Sloos, hoje vice-presidente de engenharia da Google nos EUA. O SRE nasceu no momento em que Ben teve de comandar uma equipe de operações com sete engenheiros de software e precisou desenvolver uma nova abordagem de gerenciamento de operação para todo o serviço dentro da plataforma do Google, fazendo uso de recursos eficientes e com foco na confiabilidade do sistema perante seu usuário.

A partir daí, a confiabilidade do sistema passou a ser o fator mais importante dos seus projetos e do gerenciamento dos seus times. O impacto é tão positivo que empresas como Amazon, Netflix e LinkedIn também passaram a adotar o conceito de SRE.

O SRE é tanto uma prática quanto uma função de trabalho, onde a engenharia suporta diretamente as operações de software.

No Google, o SRE é descrito por seu fundador como “o que acontece quando você pede a um engenheiro de software para projetar uma equipe de operações”.

No esquema abaixo, listamos algumas atividades realizadas no dia a dia de profissionais que atuam para aumentar a confiabilidade dos sistemas:

O QUE OS TIMES DE SRE FAZEM?

SREs desenvolvem soluções para projetar, construir e executar sistemas em grande escala, de forma **escalável, confiável e eficiente**.

Guiam a arquitetura dos sistemas, atuando na interseção entre desenvolvimento de software e engenharia de sistemas.

SREs desenvolvem uma cultura que busca tratar a engenharia de software como aliada da operação.

Trabalham com um espírito de pessimismo construtivo: **esperam pelo melhor, mas planejam-se para o pior**.

#2
princípio

Nosso monitoramento não
decide nossa confiabilidade;
nosso usuários sim.

Fundamentos de SRE

A Engenharia de confiabilidade de sistemas é uma abordagem da engenharia de software para as operações de TI. Para gerenciar sistemas, as equipes de SRE fazem uso de softwares, visando solucionar problemas e automatizar tarefas operacionais.

Com times de SRE, tarefas que antes eram responsabilidade das equipes de operação, muitas vezes realizadas até de forma manual, passam a ser realizadas por um engenheiro ou equipes de operações que utilizam software ou automação para solucionar problemas e gerenciar sistemas de produção.

As práticas de SRE tornam-se grandes aliadas dos times que buscam criar sistemas escaláveis e altamente confiáveis. Qualquer tarefa que um administrador de sistemas faria manualmente, o SRE a realizará através de software. Ou seja, é sempre preferível que os sistemas sejam programados para realizar atividades de forma automatizada.

A abordagem da engenharia de confiabilidade dos sistemas ajuda as equipes a alcançar o equilíbrio entre lançar novas funcionalidades de forma mais rápida e garantir que esses sistemas sejam mais estáveis e mais confiáveis para os usuários finais.

Um SRE busca maior confiabilidade e menor risco de falhas não apenas no momento presente, mas também ao longo do tempo, conforme esse sistema cresce. Ele também facilita o trabalho das equipes que estão migrando suas operações para o desenvolvimento nativo em nuvem.

SRES TRATAM A OPERAÇÃO COMO UM PROBLEMA DE ENGENHARIA DE SOFTWARE:

Contratam profissionais que saibam **desenvolver automações**.

Usam **software para realizar tarefas** normalmente executadas por administradores de sistemas.

Projetam arquiteturas de serviço mais **confiáveis e operáveis** desde o início.

#3

princípio

Softwares bem estruturados podem levar a uma **disponibilidade de 99.9%**.

Operações bem estruturadas → **99.99%**

Negócios bem estruturados → **99.999%**

Atribuição do profissional

As atividades mais importantes da abordagem de SRE são a padronização e a automação. Os engenheiros de confiabilidade devem estar sempre em busca de uma maneira de aprimorar e automatizar tarefas operacionais.

Ao observar que um problema se repete, deve-se criar uma solução automatizada. Com isso, o SRE se assegura de que o trabalho operacional não ocupe mais da metade do seu tempo de trabalho. O equilíbrio entre trabalho operacional e desenvolvimento é vital para a abordagem de SRE.

Outro ponto central na abordagem desenvolvida pelo Google é o foco na melhoria contínua. Os profissionais de SRE não apenas "desenvolvem" a automação e restauram os serviços com falhas, mas também buscam garantir que os incidentes não aconteçam novamente. Uma das atividades-chave realizadas é o "postmortem", um relatório que permite identificar a causa

raiz do problema, ou mais de uma, para produzir um plano de ação e resolvê-lo. Essa abordagem ajuda a evitar trabalhar "apagando incêndios" e a buscar soluções mais definitivas para o aumento da confiabilidade.

O perfil ideal para um candidato ao cargo de SRE é um engenheiro de software com boa experiência em administração de sistemas escaláveis, com conhecimentos em codificação e automação. Esse profissional também pode trabalhar para eliminar gargalos de desempenho, isolar falhas usando os padrões de disjuntor¹ e antepara², criando runbooks³ e automatizando os processos operacionais diários.

Um SRE também deve se preocupar com diferentes aspectos de um serviço: arquitetura da aplicação e suas dependências; instrumentação e monitoração; acompanhamento de métricas; planejamento de capacidade; gerenciamento de mudanças; resposta a emergências e desempenho do sistema.

Idealmente um único profissional não deve atuar em mais de dois eventos por turno de plantão para evitar sobrecarga, pois será preciso tempo para responder e corrigir interrupções de serviço, iniciar a documentação do postmortem com análise de causa raiz e registrar bugs resultantes.

¹ Disjuntor é um padrão de design usado para detectar falhas e aplicar a lógica de evitar que uma falha se repita constantemente, durante a manutenção, falha temporária do sistema externo ou dificuldades inesperadas do sistema.

² O padrão Bulkhead é um tipo de design de aplicativo que tolera falhas. Em uma arquitetura bulkhead, os elementos de um aplicativo são isolados em pools para que, se um falhar, os outros continuem a funcionar.

³ Um runbook é um guia detalhado de "como fazer" para concluir uma tarefa ou procedimento comumente repetido no processo de operações de TI de uma empresa. Runbooks são criados para fornecer a todos na equipe - novos ou experientes - o conhecimento e as etapas para resolver um determinado problema com rapidez e precisão.

“

***100% é uma confiabilidade alvo errada
por basicamente qualquer parâmetro***

”

Benjamin Treynor Sloss

Vice Presidente de Engenharia 24x7 do Google

Métricas de SRE

Toda a abordagem de engenharia de confiabilidade, desenvolvida pelo Google, é baseada em uma cultura de análise de dados. Um SRE vai desenvolver e implementar soluções de acordo com as métricas definidas pelo time.

Por meio das práticas de SRE, as equipes determinam quais funcionalidades novas podem ser lançadas e em que momento. Isso é realizado fazendo uso do **Error Budget**, que é o orçamento para falhas e pode ser gasto para aplicar mudanças no sistema. A confiabilidade de um sistema será medida por indicadores de nível de serviço (SLI) e objetivos de nível de serviço (SLO).

O **SLI** é uma métrica determinada a partir de aspectos específicos dos níveis de serviços oferecidos. Como principais indicadores podemos citar latência de solicitação, disponibilidade, taxa de erros e capacidade do sistema. O **SLO** é baseado no intervalo ou valor desejado (meta) para um nível de serviço com base no SLI.

SLOs

Os SLOs fazem parte de um conjunto de medidas que visam aumentar a qualidade do gerenciamento do serviço. Esse conjunto de medidas ajuda a entender quais comportamentos realmente importam, como medi-los e de que forma os avaliar para que o serviço tenha um nível de qualidade aceitável.

Escolher as métricas apropriadas garante que as ações corretas sejam realizadas caso algo dê errado. Isso traz maior confiança para o time de Site Reliability Engineering sobre o que é importante para o pleno funcionamento do serviço, além de resultar em uma experiência positiva para o usuário final.

Para que os SLOs sejam realmente válidos, algumas diretrizes devem ser seguidas: todos os stakeholders precisam aprovar os SLOs como adequados ao produto; as pessoas responsáveis

por garantir que eles sejam atingidos precisam concordar que a meta é factível de ser atingida em circunstâncias normais; a empresa deve se comprometer a utilizá-los como norte para a tomada de decisões e também é necessário que exista um processo de constante aprimoramento dos SLOs.

Caso esses pontos não sejam seguidos, os SLOs tornam-se apenas mais um KPI não utilizado, acabando por não ter relevância no processo de tomada de decisão.

Como criar SLOS

Para criar os SLOs é preciso levar em consideração o que ele deve cobrir. Sua principal missão é garantir um alto nível de confiabilidade para os usuários finais. Para isso, é preciso saber como o seu usuário utiliza seu sistema e quais jornadas dos usuários são as mais críticas.

Caso o sistema seja complexo, o melhor caminho para criar SLOs é realizar a organização dos componentes de sistema em categorias como resposta/solicitação, armazenamento, pipeline de dados e assim por diante, especificando SLIs para cada uma dessas categorias. É indicado tentar manter os SLOs sempre o mais simples possível, apenas na quantidade suficiente para fornecer uma boa cobertura dos atributos do sistema, garantindo assim o sucesso da sua implementação.

PERGUNTAS QUE AJUDAM A DEFINIR O SLO:

Como os usuários fazem [uso de suas aplicações](#)?

Qual [jornada eles percorrem](#) no aplicativo?

Quais são as [partes da infraestrutura](#) de que a jornada depende?

O que os [usuários esperam realizar](#) na aplicação?

SLIs

Os Service Level Indicators (SLIs) são medidas quantitativas (métricas) criadas cuidadosamente para definir algum aspecto do nível de serviço fornecido aos usuários finais. Servem para medir como está indo o cumprimento dos objetivos definidos previamente. Normalmente são documentadas em acordos de nível de serviço (SLAs), para saber se os objetivos de nível de serviço (SLOs) foram alcançados. É importante que as empresas tenham claro quais as medições fornecidas por software são, de fato, importantes para o pleno funcionamento do sistema.

Ter muitos SLIs pode distrair e impedir que os engenheiros de confiabilidade prestem atenção suficiente aos indicadores de desempenho mais críticos. Ao ter claro quais métricas devem ser revisadas, os SREs conseguem se concentrar melhor em onde estão as funções são mais importantes para o usuário.

Tipos de SLI

Os SLIs podem ser divididos basicamente em duas categorias:

1. Baseados em requisições

Esse tipo de SLI serve para medir o número de requisições válidas ou completas no serviço versus o número total de requisições do sistema. Geralmente depende de um tempo específico e ajuda o provedor de serviços a identificar possíveis obstáculos no atendimento aos clientes.

Por exemplo, um serviço com 5.000 requisições bem-sucedidas, em um total de 7.000, pode ser considerado um sistema com uma taxa de sucesso de 71%. Você pode comparar o SLI ao seu SLO: se o seu objetivo era atingir 70% de todas as requisições atendidas com sucesso, pode considerar que o objetivo foi alcançado, caso contrário, o SLO não foi atingido.

2. Baseado em janela

Os SLIs baseados em janelas são a medida do limite de confiabilidade de um serviço em um período de tempo específico. Isso ajuda a sua empresa a medir o desempenho dos serviços durante um intervalo de tempo específico ou em diferentes horários do dia para comparação.

A fórmula utilizada para esse indicador é **SLI = período de sucesso/total de períodos**. Como exemplo, podemos citar uma empresa que mede períodos de alto tráfego durante a hora do almoço. Nesse caso, é possível determinar quantos períodos de tempo têm requisições bem-sucedidas em relação ao total de períodos durante o horário de pico de tráfego. É essencial começar as medições pelo mais simples e importante aos olhos do usuário e depois evoluir visando aumentar cada vez mais a qualidade do sistema.

$$SLI = \left[\frac{\text{eventos bons}}{\text{eventos válidos}} \right] \times 100\%$$

Como criar SLIs

Para a criação de bons SLIs, é preciso seguir alguns passos específicos que visam garantir que as métricas medidas são as corretas para o aprimoramento da experiência do usuário. As etapas são:

1. Definir os SLOs

É preciso saber quais são seus SLOs (seus objetivos) para saber o que deve ser esperado com os SLIs (seus indicadores). Os SLIs dependem diretamente de SLOs bem definidos para que se possa medir o sucesso do cumprimento dos objetivos do sistema.

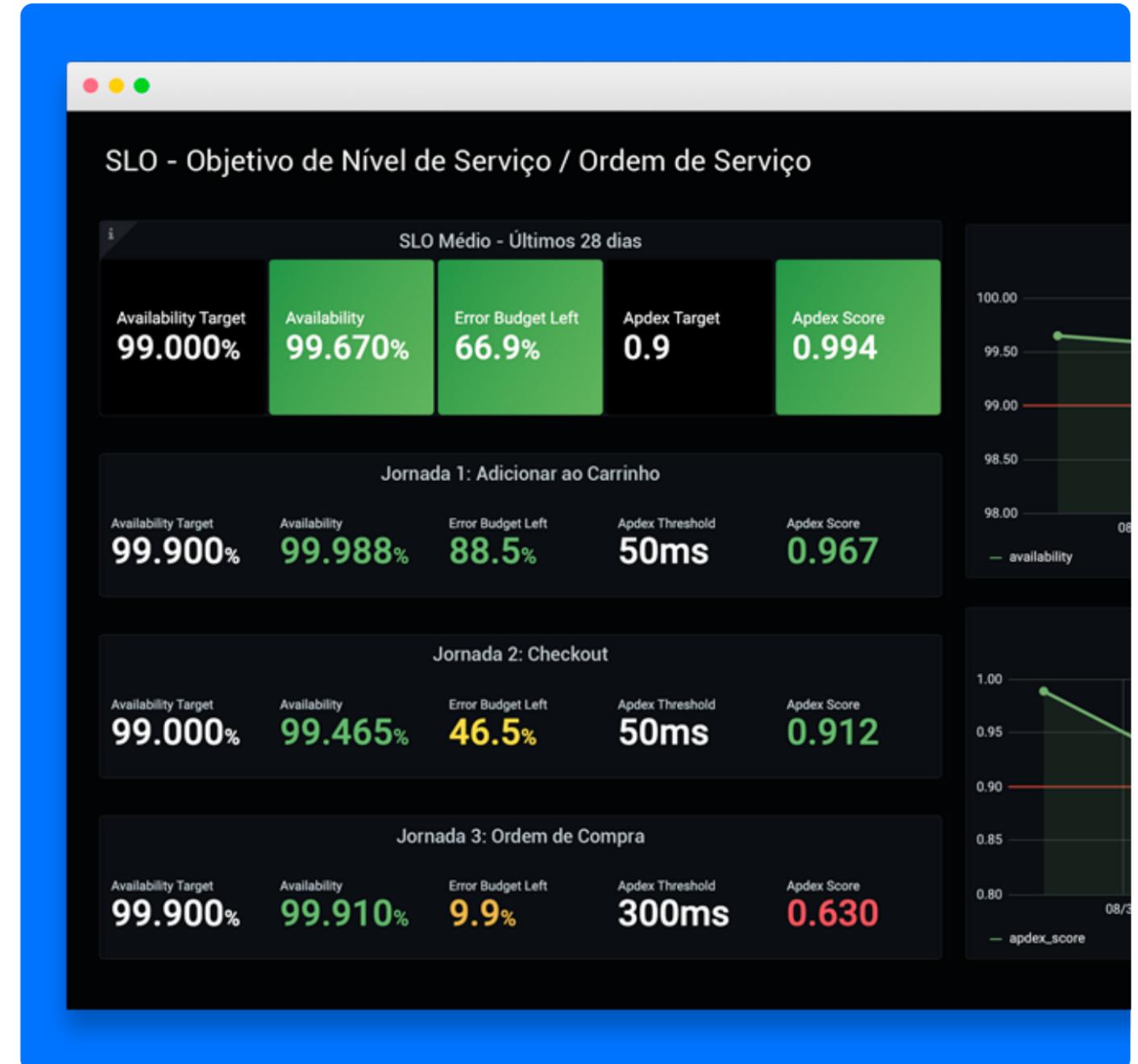
2. Definir métricas para cada SLI

Ter claro quais as métricas serão utilizadas para medir cada SLI em relação aos SLOs é muito importante. Já existem vários serviços em nuvem que possibilitam a personalização de opções para SLOs específicos. Podendo, por exemplo, rastrear SLIs de disponibilidade e latência diretamente em sua interface de nuvem ou criar seus próprios.

3. Alterar SLIs para mudanças

As alterações podem ocorrer dentro de um sistema com base em quantos usuários ou solicitações o sistema processa, quão alta é a latência e se o serviço atende consistentemente aos SLOs predefinidos.

Tente atualizar seus SLIs em resposta a essas alterações para mantê-los alinhados com os SLOs atuais. Por exemplo, você pode atingir seu SLO de redução da latência em 15% ao longo de um mês. Depois de atingir esse SLO, você pode definir um novo objetivo para reduzir ainda mais a latência e editar suas métricas de SLI para rastrear o novo objetivo. Sempre visando o aperfeiçoamento dos sistemas e o aumento da confiabilidade dos mesmos, não somente no presente, mas ao longo do tempo conforme o sistema cresce.



Error budget

O orçamento de erro (error budget), dentro do contexto de Site Reliability Engineering (SRE), serve como uma estrutura para gerenciamento de riscos, visando trazer equilíbrio entre o time de desenvolvimento e os SREs. Em linhas gerais, é usado para medir o quanto de risco a empresa está disposta a tolerar em um serviço. Vale ressaltar que quanto menor o orçamento disponível para error budget, menor será a capacidade da empresa em lançar novos recursos em produção.

O error budget deve ser “gasto” pela equipe de desenvolvimento ao lançar uma funcionalidade nova. Tendo como base o SLO e o orçamento de erro disponível, a equipe deve determinar se o lançamento é viável ou não. Vale destacar que não se deve esperar 100% de confiabilidade, mas que as falhas sejam planejadas e negociadas dentro de um percentual aceitável.

ALGUNS PONTOS CONSIDERÁVEIS PARA QUE O ERROR BUDGET FUNCIONE BEM:

- Contar com **SLOs adequados e aprovados pelas partes envolvidas** na empresa (time técnico e time de negócio);
- Tendo esgotado o orçamento de erro, algo deve ser feito para **restaurar a estabilidade do sistema**;
- Os responsáveis por garantir que o serviço atenda o SLO **devem estar de acordo que seja possível alcançá-lo** em circunstâncias normais;
- A empresa ter o comprometimento de **utilizar o error budget para a tomada de decisões e oficializar isso** por meio de uma política de orçamento de erros;
- Caso o orçamento de erro esgote-se para o período estabelecido, **deve-se cumprir o não lançamento de novos releases**.

Nível de Confiabilidade	Janela máxima de indisponibilidade		
	por ano	por trimestre	por 30 dias
90%	36.5 dias	9 dias	3 dias
95%	18.25 dias	4.5 dias	1.5 dias
99%	3.65 dias	21.6 horas	7.2 horas
99.5%	1.83 dias	10.8 horas	3.6 horas
99.9%	8.76 horas	2.16 horas	43.2 minutos
99.95%	4.38 horas	1.08 horas	21.6 minutos
99.99%	52.6 minutos	12.96 minutos	4.32 minutos
99.999%	5.26 minutos	1.30 minutos	25.9 segundos

Como calcular o error budget?

Para a definição da métrica, os times de produto e SRE devem se reunir para criar o orçamento de erros, tendo como base o objetivo de nível de serviço (SLO). Nessa métrica deve estar claro e objetivo o quanto de tempo, dentro do período estabelecido, que o serviço pode ficar indisponível ou não confiável. Como a métrica é definida em conjunto com os times, isso ajuda a diminuir a tensão sobre negociações para a definição de qual nível de risco se deve assumir em uma atualização de serviço.

Por exemplo, se for definido que o objetivo (SLO) é ter 99% dos eventos válidos com sucesso, se tem 1% de margem de erro. Ou seja, se uma API recebe 1.000.000 requisições, com um SLO de disponibilidade de 99%, o orçamento é de 10.000 erros de requisição durante esse período de tempo.

SIMPLIFICANDO SLI, SLO E ERROR BUDGET

SLI

O que é “bom”?

SLO

O quão “bom” precisamos ser?

Error budget

O quão bom nós atingimos?
Ou inversamente, quão ruim nós atingimos?

Rotina do SRE

Ser engenheiro de confiabilidade é ter seu tempo dividido entre execução de tarefas operacionais e novos desenvolvimentos. De acordo com as práticas recomendadas do Google, o SRE não deve passar mais de 50% do seu tempo trabalhando nas operações, isso deve ser monitorado para garantir que a porcentagem não seja ultrapassada. O resto do tempo deve ser gasto em atividades de desenvolvimento, como criação de novas funcionalidades, escalonamento do sistema e implementação da automação.

O engenheiro de confiabilidade é responsável pela disponibilidade, latência, desempenho, eficiência, gerenciamento de mudanças, monitoração, resposta a emergências e planejamento de capacidade de seu(s) serviço(s).

A seguir, detalharemos um pouco mais cada uma dessas atividades, dentro do contexto da rotina do SRE.

1. Monitoramento

O primeiro procedimento é simplesmente visualizar sua aplicação como ela é, e para isso é extremamente importante em um primeiro momento definir seus Níveis de Serviço de maneira que você consiga encaixar e metrificar a sua aplicação corretamente.

Sem um bom monitoramento não é possível saber se o serviço está funcionando. Portanto, monitorar a aplicação permite dar visibilidade à confiabilidade do software para seus usuários. Com isso, as saídas válidas da aplicação, sejam elas alertas, tickets ou logs, tornam-se matérias-primas para tomada de decisões no futuro. Um bom caminho para começar a monitoração é iniciar pelos *The Four Golden Signals* (os quatro sinais de ouro), que são latência, tráfego, erros e saturação.

2. Planejamento de capacidade

Aqui é importante se planejar para o crescimento orgânico - aumento na adoção e uso do produto por clientes - e se preparar para o crescimento inorgânico - saltos repentinos de demanda, como campanhas de marketing e lançamentos. Outro ponto relevante é relacionar a infraestrutura básica com capacidade do serviço, sempre buscando ter um pouco de capacidade em excesso para atingir as metas de confiabilidade.

É importante destacar que capacidade em excesso se reflete em maior custo. Um SRE deve primeiro entender a demanda para, posteriormente, conseguir prever e provisionar, podendo até modificar o software em função disso. O caminho do meio, quando se trata de capacidade, para a infraestrutura dos serviços é o melhor a ser seguido. Buscando o equilíbrio para não estourar nem o orçamento e nem a capacidade do sistema.

3. Resposta a incidentes

Acontecendo o incidente, a primeira coisa é não entrar em pânico, o mundo não vai acabar. A reação mais adequada é focar em mitigar o problema, para depois depurá-lo e resolvê-lo.

Embora você não possa evitar 100% dos incidentes, é bom realizar o postmortem para saber o que fazer quando ocorrerem falhas. O postmortem permite documentar o incidente, entender todas as causas que contribuíram para que o incidente tenha ocorrido e criar ações efetivas que previnam/reduzam o risco e o impacto de uma recorrência no futuro.

Esse processo de postmortem define o momento certo para declarar um incidente - quando ocorrer uma falha visível para

o usuário, quando a degradação do serviço estiver acima de um limite pré-determinado, quando houver perda de dados de cliente ou quando o tempo de resolução atingir um limite, são alguns exemplos.

4. Gerenciamento de mudanças

A maioria dos problemas de indisponibilidade estão relacionados às mudanças nos ambientes/sistemas em produção. Diante disso, rollouts progressivos, fast fail, e rollback podem ajudar a impactar um número menor de usuários quando mudanças ruins acontecem. Remover humanos do processo de lançamento também reduz a chance de falhas de uma mudança. Com isso, aumenta a velocidade de entrega de funcionalidades e confiança/segurança do ambiente.

5. Cultura

O primeiro paradigma que precisa ser quebrado ao trabalhar com times de SRE é parar de buscar culpados e entender que as falhas acontecem e sempre acontecerão, não importa o quanto você invista nisso. Uma cultura de busca por culpados produz uma equipe que não relata os problemas, gerando, assim, um ciclo negativo.

Para desenvolver um time de SRE é importante contratar bons engenheiros de software e bons engenheiros de sistemas. Além disso, toda a equipe deve saber programar, pois o profissional não irá apenas realizar trabalho manual, pois metade do seu tempo será utilizado para automatizar tarefas. Outro ponto importante da cultura é empoderar o time de SREs para que eles consigam aplicar o error budget, previamente acordado, sem precisar consultar ninguém.

RESUMO DE ATIVIDADES DO SRE:



Métricas e monitoramento

SLO, SLI, Error Budget, Dashboards & Análises.



Planejamento de capacidade

Previsão, Gestão de Demanda & Performance.



Gestão de mudanças

Lançamento, Mudanças em Produção & Automação.



Resposta de incidentes

Plantão, Análises & Postmortem.



Cultura

Treinamento & Melhoria Contínua.

Quão confiável minha empresa deve ser?

Uma dica importante que a equipe do Google dá sobre qual a janela de tempo utilizar para o orçamento de erros é sempre buscar períodos iguais. Eles mesmos utilizam janelas de 28 dias. Segundo eles, este período, além de representar 4 semanas corridas, é a memória curta do usuário para erros.

Dentro deste período, por exemplo, uma disponibilidade de 99.9% significa uma parada de 43,2 minutos durante o período inteiro. Portanto, cuidado ao buscar SLOs com disponibilidades altíssimas, como 99.999%, que significaria 24,2 segundos apenas de indisponibilidade durante o período inteiro.

De acordo com Benjamin Treynor Sloss, Vice-presidente de Engenharia 24x7 do Google, *"100% é uma confiabilidade alvo errada por basicamente qualquer parâmetro"*. Isso porque não permite flexibilidade, barra a inovação e simplesmente custa caro demais, mesmo para as maiores empresas.

Em linhas gerais, SLOs com disponibilidades de 99.9% para cima, são para grandes empresas. SLOs de 99.999% são para pouquíssimas empresas, como Google, Netflix, Uber, etc. Para um cenário mais realista, SLOs de 99% já começam ser interessantes e possibilita espaço para inovações e experimentações.

Você pode começar com um SLO até mesmo menor, digamos 97% ou 98% e medir por algum tempo como seus sistemas se comportam e a partir daí, gradualmente aumentar o SLO para 99% ou mesmo acima disto. Porém, é importante que estes objetivos estejam realmente alinhados com a área de negócios da organização.



SLOs de 100% não permitem flexibilidade, **barram a inovação e simplesmente custam caro demais**, mesmo para as maiores empresas como Google, Netflix, Uber, etc.

Conclusão

O SRE é um profissional (ou um time de profissionais) focado no desenvolvimento e entrega de novas e melhores funcionalidades aos sistemas, além de buscar constantemente garantir a não repetição de erros e bugs apresentados anteriormente. Ou seja, é uma forma de mitigar falhas que impactam os usuários de um sistema.

Aplicações baseadas em nuvem, que possuem um grande volume de usuários e que o tempo de resposta é crítico para o sucesso do negócio, demandam serviços de SRE altamente especializados para garantir boas experiências finais aos usuários.

A OpServices está preparada para ajudar sua empresa na jornada de tornar suas aplicações e ambientes de sustentação com um alto grau de confiabilidade e performance, de acordo com as expectativas de suas unidades de negócios e clientes.

CONHEÇA NOSSOS **SERVIÇOS** DE **CONSULTORIA** PARA **SRE**:



Treinamento
de equipes



Consultoria e
Implementação



Alocação de
Profissional

Sobre a OpServices

Empresa especializada em **monitoramento de ambientes complexos de TI e visualização de dados**, com mais de 20 anos de mercado. É reconhecida por desenvolver soluções e serviços avançados de acordo com as necessidades específicas de cada cliente.

Grendene

LEAR

marisa

RENNER

INSTITUTO BUTANTAN
A serviço da vida

Zaffari HIPERMERCADOS **BOURBON**

MARISTA
COLÉGIOS | UNIDADES SOCIAIS

Klabin

digi+

Banrisul

SantaCruz

MOREFREE
liberdade para seu negócio

Conheça nossas redes sociais: [f](#) [in](#) [▶](#)