**Este módulo abordará os seguintes tópicos:**

* **AWS Well-Architected Framework**
* **Confiabilidade e alta disponibilidade**
* **AWS Trusted Advisor**

**AWS WELL-ARCHITECTED FRAMEWORK**

A tradução de AWS Well-Architected Framework para o português é Framework Bem Arquitetado da AWS. Esse nome representa um conjunto de boas práticas para projetar infraestruturas na nuvem que sejam seguras, eficientes, resilientes e de alto desempenho

Arquitetura é como planejar e construir algo grande e organizado. Imagine uma cidade: para funcionar bem, ela precisa de ruas, prédios, energia e transporte, tudo pensado antes de ser construído. Nos sistemas de tecnologia, funciona do mesmo jeito!

Os **arquitetos de nuvem** são como planejadores que ajudam empresas a organizar sua tecnologia para que tudo funcione da melhor forma possível. Eles fazem três coisas importantes:

1. **Conversam com os chefes da empresa** para entender o que precisa melhorar.
2. **Garantem que a tecnologia ajude a empresa a crescer e alcançar seus objetivos.**
3. **Ajudam as equipes que estão colocando as ideias em prática**, para que tudo funcione direitinho.

Se um sistema for bem planejado, ele tem mais chances de dar certo e ajudar a empresa a ter sucesso.

O **AWS Well-Architected Framework** é um **guia** para projetar infraestruturas na nuvem que sejam:

* **Seguras** – Protegem dados e sistemas contra ameaças.
* **De alto desempenho** – Funcionam de maneira rápida e eficiente.
* **Resilientes** – Conseguem se recuperar de falhas e continuar operando.
* **Eficientes** – Utilizam recursos da melhor forma possível.

Esse framework oferece uma **abordagem consistente** para **avaliar e implementar** arquiteturas de nuvem, garantindo que sigam as **práticas recomendadas**. Essas recomendações foram desenvolvidas com base nas **lições aprendidas** durante **análises de arquiteturas de clientes** da AWS.

A AWS também disponibiliza a **AWS Well-Architected Tool**, uma ferramenta que ajuda a avaliar e aprimorar arquiteturas seguindo esse framework.

O **AWS Well-Architected Framework** é baseado em **seis pilares** que ajudam a criar infraestruturas na nuvem mais eficientes e seguras:

1. **Excelência operacional** – Melhora processos e operações para garantir eficiência.
2. **Segurança** – Protege dados e sistemas contra ameaças.
3. **Confiabilidade** – Garante que os sistemas funcionem corretamente e possam se recuperar de falhas.
4. **Eficiência de desempenho** – Otimiza o uso de recursos para melhor desempenho.
5. **Otimização de custos** – Ajuda a reduzir gastos desnecessários.
6. **Sustentabilidade** – Minimiza impactos ambientais da infraestrutura na nuvem.

Os primeiros cinco pilares foram introduzidos em **2015**, e o pilar de **Sustentabilidade** foi adicionado em **2021** para ajudar empresas a reduzir os impactos ambientais de suas operações na nuvem.

O pilar de **Identity and Access Management (IAM)** no **AWS Well-Architected Framework** trata da **gestão de credenciais e autenticação**, garantindo que o acesso aos sistemas seja seguro e controlado.

Como gerenciar credenciais e autenticação?

Credenciais são **senhas, tokens e chaves** que permitem acesso a sistemas na nuvem. Para evitar riscos, é essencial protegê-las com boas práticas de segurança.

Práticas recomendadas:

* **Definir requisitos claros** para o gerenciamento de identidade e acesso.
* **Proteger o usuário raiz** da conta AWS, evitando acessos desnecessários.
* **Usar autenticação multifator (MFA)** para aumentar a segurança.
* **Automatizar controles de acesso**, reduzindo erros humanos.
* **Integrar-se a um provedor de federação centralizado**, facilitando a gestão de usuários.
* **Aplicar requisitos de senha**, garantindo senhas fortes e seguras.
* **Alternar credenciais regularmente**, evitando acessos indevidos.
* **Realizar auditorias periódicas**, verificando possíveis vulnerabilidades.

Cada pilar do **AWS Well-Architected Framework** tem um conjunto de **princípios de design** e **práticas recomendadas**, que ajudam a criar arquiteturas seguras e eficientes.

O **pilar de Excelência Operacional** do **AWS Well-Architected Framework** é sobre garantir que os sistemas funcionem bem e tragam benefícios reais para o negócio. Ele ajuda a organizar e melhorar processos para que tudo seja mais eficiente.

Como isso funciona?

O foco principal é **executar e monitorar sistemas**, para que a empresa consiga entregar valor de forma contínua. Isso significa que não basta apenas rodar um sistema, é preciso sempre buscar melhorias.

Três pontos importantes:

1. **Automatização de alterações**
   * Tudo que puder ser feito de forma automática deve ser considerado. Assim, menos tempo é gasto com tarefas repetitivas, e os erros humanos diminuem.
2. **Resposta a eventos**
   * Se algo inesperado acontecer, o sistema deve reagir rápido e da melhor maneira possível. Isso evita problemas grandes e mantém tudo funcionando.
3. **Definição de padrões para operações diárias**

* Ter regras e processos bem definidos ajuda a evitar confusão e facilita a gestão da tecnologia no dia a dia.

Resumo:

Esse pilar garante que a empresa tenha **processos organizados, automatizados e eficientes**, sempre buscando melhorias para agregar mais valor ao negócio

Os **princípios de design da excelência operacional** no **AWS Well-Architected Framework** ajudam a garantir que sistemas na nuvem sejam eficientes, seguros e sempre aprimorados. Aqui está um resumo claro e direto:

**Os cinco princípios de design da excelência operacional**

1. **Executar operações como código**
   * Tudo deve ser tratado como código: **infraestrutura, aplicações e procedimentos operacionais**.
   * Isso permite **automatizar respostas a eventos**, reduzindo erros humanos e garantindo consistência.
2. **Fazer alterações frequentes, pequenas e reversíveis**
   * Atualizações devem ser feitas em **pequenos incrementos**, para que possam ser revertidas sem impacto significativo.
   * Isso melhora a **agilidade e segurança** das mudanças.
3. **Refinar os procedimentos operacionais com frequência**
   * Sempre busque **melhorar processos** e adaptar procedimentos conforme a carga de trabalho evolui.
   * Realizar **testes regulares** garante que as equipes estejam preparadas e que os procedimentos sejam eficazes.
4. **Prever falhas**
   * Identifique **possíveis pontos de falha** e crie estratégias para **mitigá-los**.
   * Teste cenários de falha e valide se os procedimentos de resposta são eficientes.
   * Simulações regulares ajudam a equipe a se preparar para eventos inesperados.
5. **Aprender com eventos e falhas operacionais**

* Cada falha deve ser vista como uma **oportunidade de aprendizado**.
* Compartilhar lições aprendidas com toda a equipe ajuda a **melhorar continuamente** os processos.

**Resumo**

A **excelência operacional** garante que sistemas sejam **automatizados, seguros e sempre aprimorados**, permitindo que empresas **entreguem valor comercial** de forma eficiente.

O **pilar de Excelência Operacional** no **AWS Well-Architected Framework** se baseia em **quatro áreas principais** para garantir que os sistemas funcionem de forma eficiente e tragam valor para o negócio: **organização, preparação, execução e evolução**.

**Organização**

* **Definir prioridades** para focar no que é mais importante para o negócio.
* **Estruturar a organização** para dar suporte aos objetivos empresariais.
* **Criar uma cultura organizacional** que apoie os resultados desejados.

**Preparação**

* **Projetar a carga de trabalho** para que seja possível monitorar seu estado.
* **Reduzir defeitos e facilitar correções**, garantindo um fluxo eficiente para a produção.
* **Mitigar riscos de implantação**, evitando problemas antes que aconteçam.
* **Garantir que a equipe esteja pronta** para oferecer suporte à carga de trabalho.

**Execução**

* **Monitorar a integridade da carga de trabalho** para garantir que tudo funcione corretamente.
* **Acompanhar a integridade das operações**, identificando possíveis melhorias.
* **Gerenciar eventos operacionais**, garantindo respostas rápidas e eficazes.

**Evolução**

* **Melhorar continuamente as operações**, adaptando-se às mudanças do negócio.
* **Aprender com falhas e eventos operacionais**, garantindo que cada experiência contribua para melhorias futuras.
* **Coletar métricas** para medir o impacto das operações e ajustar estratégias conforme necessário.

**Resumo**

A **Excelência Operacional** garante que as operações sejam **bem estruturadas, monitoradas e aprimoradas continuamente**, permitindo que a empresa **se adapte às mudanças e melhore seus processos**.

O **pilar de Segurança** no **AWS Well-Architected Framework** garante que sistemas, informações e ativos sejam protegidos contra ameaças. Ele também ajuda a avaliar riscos e definir estratégias para reduzir problemas.

**Principais pontos**

1. **Proteger a confidencialidade e integridade dos dados**
   * Garantir que apenas pessoas autorizadas possam acessar informações.
   * Evitar alterações indevidas nos dados.
2. **Identificar e gerenciar quem pode fazer o quê**
   * Controlar permissões de usuários e serviços.
   * Usar **IAM (Identity and Access Management)** para definir acessos.
3. **Proteger sistemas**
   * Implementar medidas de segurança para evitar ataques.
   * Usar criptografia e autenticação forte.
4. **Estabelecer controles para detectar eventos de segurança**

* Monitorar atividades suspeitas.
* Criar alertas para responder rapidamente a ameaças.

Esse pilar garante que a empresa tenha **proteção eficiente**, reduzindo riscos e mantendo a segurança dos sistemas e dados.

Os **princípios de design de segurança** no **AWS Well-Architected Framework** ajudam a proteger sistemas e dados contra ameaças. Aqui estão os sete princípios principais:

1. **Implementar uma base de identidade forte**
   * Aplicar o **privilégio mínimo**, garantindo que cada usuário tenha apenas as permissões necessárias.
   * Separar responsabilidades e **centralizar o gerenciamento de privilégios** para evitar credenciais de longo prazo.
2. **Habilitar a rastreabilidade**
   * Monitorar e auditar **ações e alterações** no ambiente em tempo real.
   * Integrar **logs e métricas** para responder automaticamente a eventos de segurança.
3. **Aplicar segurança em todas as camadas**
   * Implementar **defesa em profundidade**, protegendo desde a rede até cada instância e aplicativo.
   * Usar **Virtual Private Cloud (VPC), sub-redes e balanceadores de carga** para reforçar a segurança.
4. **Automatizar as práticas recomendadas de segurança**
   * Criar **arquiteturas seguras** e definir controles como código.
   * Automatizar processos para **escalar com segurança e eficiência**.
5. **Proteger dados em trânsito e ociosos**
   * Classificar dados por **nível de confidencialidade** e aplicar **criptografia, tokenização e controle de acesso**.
   * Garantir que dados estejam protegidos tanto **durante a transmissão** quanto **armazenados**.
6. **Manter as pessoas longe dos dados**
   * Reduzir o acesso direto a dados sensíveis para evitar erros humanos.
   * Criar **ferramentas e mecanismos** que eliminem a necessidade de manipulação manual.
7. **Preparar-se para eventos de segurança**

* Ter um **processo de gerenciamento de incidentes** bem definido.
* Realizar **simulações de resposta a incidentes** e usar automação para **detecção, investigação e recuperação rápidas**.

Esses princípios garantem que sistemas na nuvem sejam **seguros, monitorados e preparados para lidar com ameaças**.

O **pilar de Confiabilidade** no **AWS Well-Architected Framework** garante que sistemas funcionem corretamente e de forma consistente, mesmo diante de falhas. Ele ajuda a criar **cargas de trabalho resilientes**, que se recuperam rapidamente para atender às necessidades do negócio.

**Principais tópicos**

1. **Projeto de sistemas distribuídos**
   * Evita pontos únicos de falha, garantindo que o sistema continue operando mesmo se uma parte falhar.
   * Usa **múltiplas zonas de disponibilidade** para aumentar a estabilidade.
2. **Planejamento de recuperação**
   * Define estratégias para restaurar serviços rapidamente após falhas.
   * Implementa **backup e replicação** para proteger dados e garantir continuidade.
3. **Tratamento de alterações**

* Garante que mudanças no sistema sejam feitas de forma segura e controlada.
* Usa **implantação gradual e testes automatizados** para evitar impactos negativos.

Esse pilar garante que sistemas sejam **estáveis, resilientes e preparados para lidar com falhas**, reduzindo interrupções e melhorando a experiência dos usuários.

Os **princípios de design de confiabilidade** no **AWS Well-Architected Framework** garantem que sistemas possam se recuperar de falhas e operar de forma consistente. Aqui estão os cinco princípios principais:

1. **Recuperar-se automaticamente de falhas**
   * Monitorar indicadores de desempenho e configurar **recuperação automatizada** quando um limite for violado.
   * Permitir **notificação e rastreamento automáticos** de falhas para resposta rápida.
2. **Testar procedimentos de recuperação**
   * Simular falhas e validar se os procedimentos de recuperação funcionam corretamente.
   * Usar **automação** para recriar cenários de falha e corrigir problemas antes que aconteçam.
3. **Dimensionar horizontalmente para aumentar a disponibilidade**
   * Substituir um **recurso grande** por **vários menores**, distribuindo solicitações entre eles.
   * Reduzir o impacto de um **único ponto de falha** no sistema.
4. **Parar de adivinhar a capacidade**
   * Monitorar **demanda e uso do sistema** para ajustar recursos automaticamente.
   * Automatizar a **adição ou remoção de recursos** conforme necessário.
5. **Gerenciar alterações na automação**

* Usar **automação** para aplicar mudanças na infraestrutura de forma segura.
* Garantir que alterações sejam feitas sem comprometer a estabilidade do sistema.

Esses princípios ajudam a criar sistemas **resilientes, escaláveis e preparados para lidar com falhas**, garantindo **alta disponibilidade e desempenho confiável**.

O **pilar de Eficiência de Desempenho** no **AWS Well-Architected Framework** garante que os recursos computacionais sejam usados de forma eficiente para atender às necessidades do sistema, sem desperdício.

**Principais tópicos**

1. **Seleção dos tipos e tamanhos certos de recursos**
   * Escolher instâncias e serviços adequados com base na carga de trabalho.
   * Ajustar configurações para otimizar desempenho e custo.
2. **Monitoramento e desempenho**
   * Acompanhar métricas para identificar gargalos e oportunidades de melhoria.
   * Usar ferramentas como **Amazon CloudWatch** para monitoramento contínuo.
3. **Tomar decisões embasadas**

* Adaptar a infraestrutura conforme a demanda muda.
* Implementar práticas que garantam eficiência sem comprometer a qualidade.

Esse pilar ajuda a manter **alto desempenho** sem desperdício de recursos, garantindo que a empresa possa escalar e evoluir de forma sustentável.

Os **princípios de design de eficiência de desempenho** no **AWS Well-Architected Framework** ajudam a otimizar o uso de recursos computacionais sem desperdício. Aqui estão os cinco principais:

1. **Democratizar tecnologias avançadas**
   * Usar serviços gerenciados na nuvem, como **bancos de dados NoSQL, machine learning e transcodificação de mídia**, sem precisar gerenciar infraestrutura complexa.
   * Permite que equipes foquem no **desenvolvimento de produtos**, em vez de gastar tempo com provisionamento e manutenção.
2. **Ter alcance global em minutos**
   * Implantar sistemas em **múltiplas regiões da AWS** para reduzir latência e melhorar a experiência do usuário.
   * Garante **disponibilidade e desempenho** sem custos excessivos.
3. **Usar arquiteturas serverless**
   * Elimina a necessidade de gerenciar servidores, reduzindo a carga operacional.
   * Serviços **serverless** escalam automaticamente e podem **reduzir custos**, pois operam conforme a demanda.
4. **Experimentar com mais frequência**
   * Testar diferentes **tipos de instâncias, armazenamento e configurações** para encontrar a melhor opção.
   * Permite ajustes rápidos e melhora a eficiência.
5. **Considerar a afinidade mecânica**

* Escolher tecnologias que se alinhem ao objetivo do sistema.
* Exemplo: analisar padrões de acesso a dados antes de definir **bancos de dados ou armazenamento**.

Esses princípios garantem que os sistemas sejam **eficientes, escaláveis e adaptáveis**, permitindo que empresas usem recursos de forma inteligente.

O **pilar de Otimização de Custos** no **AWS Well-Architected Framework** ajuda a evitar gastos desnecessários, garantindo que os recursos sejam usados de forma eficiente.

**Principais tópicos**

1. **Compreender e controlar onde o dinheiro está sendo gasto**
   * Monitorar custos e identificar áreas de desperdício.
   * Usar ferramentas como **AWS Cost Explorer** para análise detalhada.
2. **Selecionar o número correto de tipos de recursos**
   * Escolher instâncias e serviços adequados para a carga de trabalho.
   * Ajustar configurações para otimizar desempenho e custo.
3. **Analisar gastos ao longo do tempo**
   * Acompanhar tendências de consumo e ajustar estratégias conforme necessário.
   * Implementar práticas de **Cloud Financial Management**.
4. **Scaling eficiente**

* Dimensionar recursos conforme a demanda, sem gastos excessivos.
* Usar **Auto Scaling** para ajustar capacidade automaticamente.

Esse pilar garante que os sistemas sejam **financeiramente sustentáveis**, permitindo que empresas cresçam sem desperdício.

Os **princípios de design de otimização de custos** no **AWS Well-Architected Framework** ajudam a reduzir gastos desnecessários e melhorar a eficiência financeira na nuvem. Aqui estão os cinco principais:

1. **Implementar gerenciamento financeiro da nuvem**
   * Investir em **programas, processos e conhecimento** para otimizar custos.
   * Criar estratégias para **melhorar a eficiência financeira** da organização.
2. **Adotar um modelo de consumo**
   * Pagar **apenas pelos recursos necessários**, ajustando conforme a demanda.
   * Evitar gastos baseados em **previsões imprecisas**.
3. **Medir a eficiência geral**
   * Avaliar **resultados comerciais** e os custos associados.
   * Usar métricas para entender **ganhos e oportunidades de redução de custos**.
4. **Parar de gastar dinheiro em trabalho pesado indiferenciado**
   * Deixar que a **AWS gerencie infraestrutura**, como servidores e racks.
   * Focar no **desenvolvimento de produtos e serviços**, em vez de manutenção de TI.
5. **Analisar e atribuir despesas**

* Identificar **custos exatos** e atribuí-los a **proprietários de cargas de trabalho**.
* Medir o **retorno sobre investimento (ROI)** e otimizar recursos.

Esses princípios garantem que os sistemas sejam **financeiramente sustentáveis**, permitindo que empresas cresçam sem desperdício.

**CONFIABILIDADE E ALTA DISPONIBILIDADE**

A **confiabilidade** é a capacidade de um sistema de funcionar corretamente sempre que necessário. Isso envolve **hardware, firmware e software**, e qualquer falha em um desses componentes pode afetar a disponibilidade do sistema.

**Principais conceitos**

* **Confiabilidade é estatística**: Como "tudo falha, o tempo todo", é importante medir a confiabilidade com métricas como o **Tempo Médio Entre Falhas (MTBF)**, que calcula o tempo total de serviço dividido pelo número de falhas.
* **Todos os componentes importam**: Um sistema é composto por várias partes, e todas precisam funcionar corretamente para garantir a confiabilidade.
* **Exemplo do carro**: Se a ignição falhar, o carro não pode ser usado, tornando-o **indisponível**. Se isso acontecer repetidamente, ele não será considerado **confiável**.

**Alta disponibilidade**

A **alta disponibilidade** garante que um sistema continue operando mesmo diante de falhas. Isso pode ser alcançado com:

* **Redundância**: Ter componentes duplicados para evitar interrupções.
* **Monitoramento**: Detectar falhas rapidamente e acionar respostas automáticas.
* **Recuperação rápida**: Implementar estratégias para restaurar serviços sem impacto significativo.

Esses conceitos são essenciais para garantir que sistemas sejam **estáveis, resilientes e preparados para lidar com falhas**.

As métricas de confiabilidade ajudam a medir o desempenho de um sistema e sua capacidade de se manter disponível. Aqui estão as principais:

1. **Tempo Médio Até a Falha (MTTF)**
   * Mede o tempo médio que um sistema funciona antes de falhar.
   * Exemplo: Se um aplicativo opera por **96 horas** antes de falhar, seu MTTF é **96 horas**.
2. **Tempo Médio Para Reparo (MTTR)**
   * Mede o tempo médio necessário para corrigir uma falha e restaurar o sistema.
   * Exemplo: Se o reparo leva **72 horas**, o MTTR é **72 horas**.
3. **Tempo Médio Entre Falhas (MTBF)**

* Calculado como **MTTF + MTTR**, indicando o tempo total entre falhas.
* Exemplo: Se o MTTF é **96 horas** e o MTTR é **72 horas**, o MTBF será **168 horas (1 semana)**.

Essas métricas ajudam a prever falhas e melhorar a **confiabilidade** do sistema, garantindo que ele funcione de forma consistente

A **disponibilidade** mede o tempo em que um sistema está funcionando corretamente em relação ao tempo total. Ela é expressa como uma **porcentagem de tempo de atividade** ao longo de um período, geralmente um ano.

**Principais conceitos**

* **Cálculo da disponibilidade**:
  + **Disponibilidade = Tempo normal de operação / Tempo total**
  + Exemplo: Se um sistema opera corretamente por **99,9% do tempo**, significa que ele pode ter até **8 horas e 45 minutos de indisponibilidade por ano**.
* **Número de 9s**:
  + **99%** → Até **3 dias e 15 horas** de indisponibilidade por ano.
  + **99,9%** → Até **8 horas e 45 minutos** de indisponibilidade por ano.
  + **99,99%** → Até **52 minutos** de indisponibilidade por ano.
  + **99,999%** → Até **5 minutos** de indisponibilidade por ano.
* **Impacto das falhas**:
* Qualquer falha em **hardware, software ou infraestrutura** reduz a disponibilidade.
* Interrupções podem ser **programadas (manutenção)** ou **não programadas (falhas inesperadas)**.

A alta disponibilidade é essencial para garantir que sistemas críticos permaneçam operacionais com **mínima interrupção**.

A **alta disponibilidade** garante que um sistema continue operando mesmo diante de falhas, minimizando o tempo de inatividade e reduzindo a necessidade de intervenção humana.

**Principais características**

* **Resistência a falhas**: O sistema pode sofrer degradação parcial sem ficar indisponível.
* **Tempo de inatividade reduzido**: Serviços essenciais são restaurados rapidamente, geralmente em menos de **1 minuto**.
* **Recursos compartilhados**: Componentes trabalham juntos para manter a operação contínua.
* **Combinação de software e hardware**: Usa tecnologias abertas para garantir estabilidade e recuperação rápida.

**Como garantir alta disponibilidade**

* **Redundância**: Ter componentes duplicados para evitar interrupções.
* **Monitoramento ativo**: Detectar falhas rapidamente e acionar respostas automáticas.
* **Recuperação automatizada**: Implementar estratégias para restaurar serviços sem impacto significativo.

A alta disponibilidade é essencial para sistemas críticos, garantindo que eles permaneçam operacionais mesmo em situações adversas.

A **disponibilidade** de um sistema depende de três fatores principais:

1. **Tolerância a falhas**
   * O sistema deve ter **redundância integrada** para continuar operando mesmo que alguns componentes falhem.
   * Isso envolve hardware especializado que detecta falhas e muda automaticamente para um componente redundante.
   * Importante lembrar que esse modelo não cobre falhas de software, que são as mais comuns.
2. **Dimensionamento**
   * A capacidade do sistema de **acomodar aumentos na demanda** sem precisar de grandes mudanças no design.
   * Embora não garanta disponibilidade, um bom dimensionamento evita sobrecarga e melhora a estabilidade.
3. **Capacidade de recuperação**

* O sistema deve ser capaz de **restaurar serviços rapidamente** após falhas ou desastres.
* Isso inclui processos e políticas para garantir que dados não sejam perdidos e que a recuperação seja eficiente.

Melhorar a disponibilidade pode gerar custos adicionais, então é essencial **equilibrar investimento e benefício**.

**AWS TRUSTED ADVISOR**

O **AWS Trusted Advisor** é uma ferramenta que ajuda a analisar e otimizar seu ambiente na AWS. Ele verifica sua infraestrutura e fornece recomendações para melhorar **segurança, desempenho, resiliência e otimização de custos**.

**Principais funcionalidades**

* **Identificação de riscos**: Avalia sua arquitetura e aponta áreas que precisam de melhorias.
* **Otimização de custos**: Identifica recursos não utilizados ou subutilizados para reduzir gastos.
* **Melhoria de segurança**: Verifica configurações para evitar vulnerabilidades.
* **Aumento de desempenho**: Analisa o uso dos serviços para garantir eficiência.
* **Resiliência**: Ajuda a garantir que sua infraestrutura possa se recuperar de falhas rapidamente.

O Trusted Advisor está disponível no **AWS Management Console** e pode ser acessado por clientes com planos de suporte **Business, Enterprise On-Ramp ou Enterprise.**

O **AWS Trusted Advisor** é uma ferramenta que analisa sua infraestrutura na AWS e fornece **orientações em tempo real** para otimizar o ambiente com base em práticas recomendadas. Ele verifica **cinco categorias** principais:

1. **Otimização de custos**
   * Identifica **recursos não utilizados ou subutilizados** para reduzir gastos.
   * Recomenda **capacidade reservada** para maior eficiência.
2. **Desempenho**
   * Analisa **Service Limits** e throughput provisionado.
   * Verifica **instâncias sobrecarregadas** para garantir melhor desempenho.
3. **Segurança**
   * Identifica **lacunas de segurança** e sugere melhorias.
   * Analisa **permissões de acesso** e recomenda práticas seguras.
4. **Tolerância a falhas**
   * Garante **alta disponibilidade** com auto scaling, health checks e backup.
   * Recomenda implantações **multi-AZ** para minimizar interrupções.
5. **Limites de serviço**

* Alerta sobre serviços próximos ao **limite de uso (80%)**.
* Os dados podem levar até **24 horas** para refletir mudanças.

Essa ferramenta está disponível no **AWS Management Console** e é acessível para clientes com planos de suporte **Business, Enterprise On-Ramp ou Enterprise**.

O **AWS Trusted Advisor** fornece recomendações para melhorar a infraestrutura na nuvem. No painel do seu amigo, as áreas de **otimização de custos** e **limites de serviço** parecem estar corretas, mas há pontos a serem analisados em **segurança, desempenho e tolerância a falhas**.

**Interpretação das recomendações**

1. **Segurança**
   * Verifique se há **permissões excessivas** em IAM.
   * Confirme se **grupos de segurança** não estão permitindo acesso irrestrito.
   * Ative **criptografia** para proteger dados sensíveis.
2. **Desempenho**
   * Analise **Service Limits** para garantir que os recursos não estejam sobrecarregados.
   * Ajuste **instâncias subutilizadas** para melhorar eficiência.
   * Verifique se há **gargalos de rede** impactando a performance.
3. **Tolerância a falhas**

* Habilite **Auto Scaling** para garantir disponibilidade em caso de aumento de demanda.
* Configure **multi-AZ deployments** para evitar interrupções.
* Implemente **backups e recuperação automática** para minimizar impactos de falhas.

Essas ações ajudarão seu amigo a **fortalecer a segurança, melhorar o desempenho e aumentar a resiliência** do ambiente na AWS.

**AWS Well-Architected Framework e seus Pilares**

O **AWS Well-Architected Framework** fornece diretrizes para projetar infraestruturas eficientes na nuvem, baseado em **seis pilares**:

1. **Excelência Operacional** – Executar e monitorar sistemas para otimizar operações e agregar valor comercial.
2. **Segurança** – Proteger informações, sistemas e ativos, garantindo controle de acessos e monitoramento de ameaças.
3. **Confiabilidade** – Garantir que um sistema funcione corretamente e consiga se recuperar de falhas.
4. **Eficiência de Desempenho** – Usar recursos de forma otimizada, garantindo escalabilidade e bom desempenho.
5. **Otimização de Custos** – Evitar gastos desnecessários e garantir o uso eficiente da infraestrutura.
6. **Sustentabilidade** – Minimizar impacto ambiental por meio de práticas sustentáveis na nuvem.

**Confiabilidade e Alta Disponibilidade**

* **Confiabilidade** mede a capacidade do sistema de operar corretamente ao longo do tempo, levando em conta **tempo médio entre falhas (MTBF)** e **tempo médio para reparo (MTTR)**.
* **Alta disponibilidade** reduz o tempo de inatividade, permitindo que um sistema continue operando mesmo diante de falhas.
* **Fatores que influenciam a disponibilidade:** **tolerância a falhas** (redundância), **dimensionamento** (capacidade de escalar) e **capacidade de recuperação** (rapidez na restauração).

**AWS Trusted Advisor**

* **Ferramenta que fornece recomendações em tempo real** para melhorar a infraestrutura AWS.
* Analisa cinco áreas:

1. **Otimização de Custos** – Identifica recursos não utilizados para reduzir gastos.
2. **Desempenho** – Monitora limites de serviços e throughput.
3. **Segurança** – Aponta vulnerabilidades e sugere melhorias.
4. **Tolerância a Falhas** – Indica práticas para garantir alta disponibilidade.
5. **Limites de Serviço** – Alerta sobre recursos próximos ao limite de uso.

Esse conjunto de práticas e ferramentas ajuda a **garantir eficiência, segurança e confiabilidade na nuvem**, permitindo que empresas operem de forma otimizada.

**ESTUDO DE CASO**

**Tecnologias usadas no Fly and Snap:**

1. **Amazon S3 (Simple Storage Service)** ☁️
   * Serve para **armazenar imagens** de forma segura e escalável.
   * Após o processamento, os arquivos são enviados para um **bucket do Amazon S3**, onde ficam disponíveis para consulta e uso.
   * O **serviço de imagens** recebe uma notificação sobre os arquivos e usa os dados de voo para calcular a **orientação 3D** e a **localização** de cada imagem.
2. **Amazon EC2 (Elastic Compute Cloud)**
   * Usado para **processar as imagens** antes de armazená-las.
   * A **máquina de ingestão** compacta os arquivos e os transfere via **FTP** para uma **instância do Amazon EC2**, onde os dados são analisados e organizados.
   * Também é usado para rodar o **banco de dados relacional (RDBMS)**, que armazena informações sobre as imagens.
3. **IAM (Identity and Access Management)**
   * Controla **quem pode acessar** os recursos da AWS.
   * Garante que apenas usuários autorizados possam gerenciar os **buckets do S3**, as **instâncias do EC2** e o **banco de dados**.
   * Ajuda a proteger credenciais e aplicar **autenticação multifator (MFA)** para maior segurança.
4. **RDBMS (Relational Database Management System)**
   * Usado para **organizar e armazenar informações** sobre as imagens.
   * O banco de dados relacional roda no **Amazon EC2** e contém **links para os arquivos no Amazon S3**.
   * Ele guarda dados como **carimbo de data/hora**, **localização GPS** e **orientação 3D** das imagens.
5. **FTP (File Transfer Protocol)**

* Usado para **transferir arquivos** entre sistemas.
* A **máquina de ingestão** compacta os arquivos e os envia via **FTP** para o **Amazon EC2**, onde são processados.
* Depois do processamento, os arquivos são **salvos em fita** para backup e armazenados por um provedor terceirizado.

Resumo do fluxo:

1. **Aeronave captura imagens e dados de navegação**.
2. **Imagens são armazenadas e enviadas via FTP para processamento no EC2**.
3. **Banco de dados RDBMS no EC2 organiza as informações das imagens**.
4. **Arquivos são armazenados no Amazon S3 para acesso futuro**.
5. **IAM controla o acesso seguro a todos os recursos**.

Esse sistema garante que as imagens sejam **precisas, organizadas e seguras**, permitindo que empresas usem esses dados para mapas, planejamento urbano e outras aplicações.

O **Show and Sell** é a parte da arquitetura da AnyCompany que cuida da **experiência do cliente no site**, permitindo que eles visualizem, personalizem e comprem produtos.

**Como funciona o Show and Sell?**

1. **Visualização do produto**
   * Os clientes acessam o site e podem ver **imagens e vídeos** dos produtos.
   * As imagens estão disponíveis em **vários formatos**, como mapas de visualização em grande escala.
   * O site usa **Elastic Load Balancing** com **HTTPS** para garantir segurança e desempenho.
   * Os **ativos estáticos** (imagens e vídeos) são armazenados em um **bucket do Amazon S3**.
2. **Personalização do produto**
   * Os clientes podem escolher um **local no mapa** e visualizar sua paisagem urbana em vídeo.
   * Também podem selecionar o **tamanho físico do mapa**, o **esquema de cores** (branco, monocromático ou colorido) e até adicionar **LEDs** para criar mapas iluminados.
   * O serviço **Mapping** verifica se há imagens disponíveis para o local escolhido.
3. **Processo de compra**
   * Se o cliente gostar da demonstração, ele pode **fazer um pedido**.
   * O pagamento é feito com **cartão de crédito**, mas a AnyCompany **não processa nem armazena** essas informações.
   * O processamento do pagamento é feito por um **provedor terceirizado certificado em conformidade com PCI**.
4. **Produção e envio**

* Após a confirmação do pagamento, o site instrui o serviço **Order** a enviar o pedido para produção.
* Os pedidos e detalhes dos clientes são registrados no **banco de dados Show and Sell**, que é um **RDBMS baseado no Amazon EC2**.
* Para iniciar uma **demonstração em vídeo** ou a **impressão do pedido**, o serviço **Order** coloca uma mensagem na **fila de Produção**.
* O serviço **Render** indica quando um vídeo de demonstração está disponível.
* O serviço **Order** também acompanha o **status do pedido** e registra as mudanças no banco de dados.
* Os clientes podem **rastrear seus pedidos** durante a manufatura e ver quando foram enviados. A entrega é feita por um **serviço terceirizado Dispatch**.

Resumo do fluxo:

1. **Clientes visualizam e personalizam produtos no site**.
2. **Pagamento é processado por um provedor terceirizado**.
3. **Pedido é registrado no banco de dados e enviado para produção**.
4. **Fila de Produção gerencia vídeos de demonstração e impressão**.
5. **Clientes podem rastrear seus pedidos até a entrega**.

Esse sistema garante que os clientes tenham uma **experiência fluida e segura**, desde a visualização do produto até a entrega final.

O **Make and Ship** é a parte da arquitetura da AnyCompany que cuida da **produção e envio dos produtos**.

**Como funciona o Make and Ship?**

1. **Criação dos modelos 3D**
   * A AnyCompany usa uma tecnologia própria para gerar **modelos 3D** a partir de fotografias e vídeos.
   * O serviço **Render** é uma frota de **instâncias g2.2xlarge do Amazon EC2**, que recebe pedidos da **fila de Produção** e cria os modelos 3D.
   * Esses modelos são armazenados em um **bucket do Amazon S3**.
2. **Geração de vídeos de demonstração**
   * O serviço **Render** também usa os modelos 3D para criar **vídeos de voo**, permitindo que os clientes vejam uma demonstração do produto antes da compra.
   * Esses vídeos são armazenados em um **bucket do Amazon S3 separado**.
   * A equipe exclui **demonstrações antigas** uma vez por ano, mas mantém os modelos para projetos futuros.
3. **Processo de impressão**
   * Quando um cliente faz um pedido, uma **mensagem é colocada na fila de impressão** com um **link para o modelo 3D**.
   * A equipe de **Make and Ship** tem **quatro impressoras 3D** que imprimem modelos detalhados e de alta resolução.
   * Uma **máquina de condutores impressos on-premises** recebe os pedidos da fila de impressão e os envia para a próxima impressora disponível.
4. **Atualização do status do pedido**

* Em cada estágio do processo, as **atualizações de status** dos pedidos são publicadas na **fila de status do pedido**.
* O site da AnyCompany consome essa fila e mostra o **histórico de pedidos** para os clientes.
* O **condutor de impressão** envia uma **atualização final** quando o pedido estiver **concluído, aprovado no controle de qualidade e pronto para envio**.

Resumo do fluxo:

1. **Render gera modelos 3D e os armazena no Amazon S3**.
2. **Render cria vídeos de demonstração e os salva em outro bucket do S3**.
3. **Pedidos são colocados na fila de impressão com links para os modelos 3D**.
4. **Impressoras 3D produzem os modelos físicos**.
5. **Fila de status do pedido mantém os clientes informados sobre o progresso**.
6. **Condutor de impressão confirma quando o pedido está pronto para envio**.

Esse sistema garante que os produtos sejam **precisos, bem produzidos e entregues com eficiência**.