

## Hypervisors e Máquinas Virtuais

1. Em que tipo de situações é vantajoso usar a plataforma de nuvem para implantação de aplicações?

R: A computação nuvem permite armazenar arquivos, software e servidores em um ambiente virtual, que é acessado em qualquer dispositivo com conexão à internet. Quando se necessita de infraestrutura isolada (para recuperação de desastres ou armazenamento de backups). Quando se deseja reduzir custos de infraestrutura (desde que não haja restrições regulatórias). Com a computação nuvem é possível aumentar ou diminuir os recursos das soluções utilizadas, sob a demanda.

2. Compare os chamados Hypervisors Tipo 1 e Tipo 2.

R: A virtualização **do tipo completa** fornece dois **tipos de hypervisor**. O **tipo 1**, chamado **de bare-metal** e o **tipo 2**, chamado **de hosted**. O **hypervisor do tipo bare-metal** interage diretamente com o hardware **da** máquina física. Ele é **completamente independente do sistema operacional do host**.

3. Como executar um servidor em uma máquina virtual pode ser vantajoso comparado a uma máquina física?

R: Com a máquina virtual, o usuário tem diversos benefícios financeiros, como corte de gastos com aquisição de novos hardwares, gastos com instalações, espaço físico, manutenção, consumo de eletricidade entre outros que podem vir a custar algum dinheiro.

4. Como é o uso de uma máquina virtual em termos de upgrade de hardware e backup?

R: Passar um ambiente de processamento de uma máquina virtual para outra (como na restauração de backup ou em upgrade) é bem mais ágil. Como o ambiente visto pelas aplicações instaladas na máquina virtual é o de um mesmo hardware virtual em qualquer instalação, ao se fazer um upgrade ou uma restauração de backup, a necessidade de alterações é minimizada.

5. Como virtualização de processamento e computação em nuvem se relacionam?

R: A real diferença entre os dois conceitos é que a virtualização que se trata da manipulação do software sob o hardware, durante isso a computação em nuvem é um “defeito” desse processo.

6. Que são nuvens Privadas, Públicas e Híbridas?

R: No modelo privado, a disponibilidade de recursos depende do seu investimento exclusivo

na estrutura, o que pode sair mais caro. Já na nuvem híbrida, essa adequação de recursos depende de como seu sistema foi arquitetado e em quais servidores internos ou remotos está a demanda para ajuste.

7. Qual a diferença entre IaaS, PaaS e SaaS?

R: A diferença entre IaaS, PaaS e SaaS é que a IaaS tem como benefício a economia de gastos e aquisição de hardware, apenas isso, pois PaaS e SaaS tem as mesmas funções que no caso seria, a forma de disponibilizar softwares e soluções de tecnologia através da internet, sendo assim a empresa não precisa instalar e atualizar softwares e hardware.

8. Que significa “on premisses”?

R: É aquele em que todo o processo é feito dentro de uma máquina própria ligada à uma estrutura interna.

9. Que significa IaaS?

R: Oferece os recursos de armazenamento, rede e computação necessários para o ambiente corporativo.

10. Que significa PaaS?

R: já que é uma tecnologia voltada, principalmente, para desenvolvedores e profissionais de criação de sistemas.

11. Que significa SaaS?

R: É uma forma de cloud computing que oferece aos usuários finais uma aplicação em nuvem por meio de um navegador da Internet, incluindo a infraestrutura de TI e plataformas subjacentes dela.

12. Por que o uso das plataformas de cloud terceirizadas podem ser vantajosas?

R : Esta é a solução que está por trás de empresas de sucesso, são inúmeros exemplos sendo que todas as empresas gigantes investem em cloud computing, seja para atender melhor e aumentar o número de clientes ou para executar tarefas internas de organizações e planejamento.

13. De onde vem a redução de custos no modelo de cloud?

R: Ao ver do consumidor de serviços de nuvem, a redução de custos vem do pagamento apenas pelo tempo usado, pelo menor tempo necessário para implantação de serviços, menor pessoal, maior foco nos negócios da empresa e pela redução da necessidade de investimentos.

14. Que é uma imagem?

R: A imagem de máquina virtual é um modelo que contém uma configuração de software, incluindo um sistema operacional. No caso da AWS, por exemplo, diversas imagens são homologadas e oferecidas como produto ao cliente de modo a poupar tempo para estabelecer uma máquina virtual

em funcionamento.

15. Que é “Escalabilidade Vertical”?

R: É baseada na expansão de uma rede a partir da adição de mais energia e memória à unidade de processamento principal do sistema.

16. Quais as características do EC2 da AWS?

R: Servidores virtuais que podem ser criados e contratados para uso através da Internet, várias configurações de capacidade de CPU, memória armazenamento e redes para suas instâncias informações seguras de login para suas instâncias usando pares de chave (a AWS armazena a chave pública e você armazena a chave privada em um lugar seguro).

17. Quais são as características do Fargate e do Elastic Beanstalk da AWS?

R: É um mecanismo de computação sem servidor para contêineres que funciona tanto com o Amazon Elastic Container Service quanto com o Amazon Elastic Kubernetes Service.

18. Quais são as características do ECS e do EKS da AWS?

R: **Características de ECS -**

Uma opção sem servidor **com o AWS Fargate.**

Integração **com o AWS Identity and Access Management (IAM)**

Orquestração de contêineres gerenciada pela **AWS**

**Características de EKS -**

é um software de código aberto que permite implantar e gerenciar aplicações containerizadas em grande escala.

19. O que é virtualização de storage?

R: Virtualização de armazenamento é o processo de agrupar o espaço de armazenamento físico usando software que representa a aparência de um unico dispositivo de armazenamento virtual.

20. Quais as características do S3 da AWS?

R: Você pode replicar objetos para um ou mais buckets de destino na mesma Região da **AWS** ou em outra diferente para obter latência reduzida, conformidade, segurança, recuperação de desastres e outros casos de uso.

Controle de Versão, Git e GitHub

21. O que é CI/CD?

R: Nada mais é que um método para entregar aplicações com frequência aos clientes.

22. Por que o controle de versões é ferramenta importante quando se adota o CI/CD?

R: Ao se implantar uma filosofia de CI/CD, ferramentas de controle de versão (como o Git) são importantes pois há:

- a. exigência de gerenciar uma grande quantidade de documentos (códigos-fonte, manuais, documentação de programas, scripts para implantação de infraestrutura etc.)
- b. grande quantidade de pessoas utilizando os documentos o que, às vezes, leva a versões conflitantes de um mesmo documento
- c. necessidade de agilidade de atualização de documentos, na medida que o desenvolvimento ocorre e que muitas vezes inclui a necessidade de recuperar versões antigas de documentos.

23. Que é o Git?

R: De longe, do Sistema de controle de versão moderno mais usado no mundo hoje é o Git. O Git é um projeto de código aberto maduro e com manutenção ativa desenvolvido em 2005 por Linus Torvalds, o famoso criador do kernel do sistema operacional Linux.

24. Qual a diferença entre um controle de versões centralizado (CVCS) e distribuído (DVCS)?

R: Em um sistema de controle de versões centralizado (CVCS), as bases de dados e documentos gerenciados estão em um equipamento centralizado, que controla todas as modificações que ocorrem. Em um sistema de controle de versões distribuído (DVCS), as bases de dados e documentos gerenciados são replicadas para os usuários e, eventualmente, sincronizadas com o servidor centralizado, atualizando as informações sobre as modificações que ocorreram.

25. Qual o objetivo de um “git clone”?

R: É usado sobretudo para apontar para um repositório existente e fazer um **clone** ou cópia deste repositório no novo diretório.

26. O que é um “staging area” no Git?

R: No GIT há uma terceira opção: o staging. É basicamente um espaço temporário onde você determina quais mudanças serão adicionadas.

27. O que é um “commit” no Git?

R: Estruturas de um cronograma.

28. Que é o GitHub?

R: Em um nível elevado, o GitHub é um site e serviço baseado em nuvem que ajuda os desenvolvedores a armazenar e gerenciar o seu código, rastreando e controlando também as alterações nos seus códigos.

29. Por que é importante ser um usuário hábil de Git e GitHub?

R: o **GIT** possui repositórios, e não somente um único local com o histórico de seu trabalho.

30. Que é um “branch” no Git?

R: Um **branch no Git** é simplesmente um ponteiro móvel para um desses commits.

31. Que é um “pull” no Git?

R: O comando git pull é usado para buscar e baixar conteúdo de repositórios remotos e fazer a atualização imediata ao repositório local para que os conteúdos sejam iguais.

32. Que é um “merge” no Git?

R: Um comando.

33. Compare um “reset” e um “revert” no Git.

R: O git revert, reverte as modificações de commit antigo, e assim que ele reverte, ele cria um novo commit novo, ou seja, ele não modifica nenhum dos commits anteriores. O git reset, muda o “ponteiro” da branch está apontado.

34. Que é o CodeCommit na AWS?

R: É um serviço de controle de origem gerenciado seguro e altamente dimensionável que hospeda repositórios privados do Git. Ele torna mais fácil para as equipes colaborarem com segurança no código com contribuições criptografadas em trânsito e em repouso.

## Serviços, Bancos de Dados e API Gateway

35. Cite dois serviços de Bancos de Dados oferecidos pela AWS.

R: O modelo de banco de dados mais antigo e popular é adequado para consultas e operações complexas (que retornam conjuntos de dados, com uso de operações booleanas). Bancos de dados não relacionais são adequados para informações do tipo chave:valor em esquemas flexíveis (além de tabelas, documentos, grafos etc.). São modelos de grande desempenho e escalabilidade e operam bem com consultas predominantemente de leitura/escrita simples (GetItem/PutItem).

36. Que são chave de partição e ordenação em um banco de dados no DynamoDB?

O DynamoDB armazena dados como grupos de atributos, conhecidos como *itens*. Os itens são semelhantes às linhas ou registros de outros sistemas de banco de dados. O DynamoDB armazena e recupera cada item com base no valor da chave primária, que deve ser exclusivo. Os itens são distribuídos entre unidades de armazenamento de 10 GB cada, chamadas partições (armazenamento físico interno do DynamoDB). Cada tabela tem uma ou mais partições, conforme mostrado na ilustração a seguir. Para obter mais informações, consulte Partições e distribuição de dados.

37. Qual a vantagem de se usar o API gateway na frente de aplicações na nuvem?

R: Através dele, pode-se criar um único ponto de entrada para diversos tipos de serviço. Além disso, o API Gateway pode ser integrado a segurança da AWS de modo a permitir o uso de autenticação e controle de permissões com facilidade.

38. Qual a principal diferença entre uma solicitação de método tipo GET e tipo POST por um browser?

R: O GET é feito com o browser enviando os dados da solicitação na sua URL, ou seja, no cabeçalho. O POST é feito com o browser enviando seus dados no corpo (body) da solicitação.

39. Que são API de tipo REST-GET e REST-POST?

R: O API Gateway atende diversos métodos de acesso através dos protocolos HTTP e REST. O REST é, atualmente, o protocolo mais popular. Ele apresenta diversos métodos, os mais comuns são o GET e POST. O GET é feito para que o cliente solicite um recurso que é enviado na resposta da solicitação que, normalmente, é feita através de dados na sua URL.

40. Que é CORS?

R: CORS é sua ferramenta para permitir requisições de domínios diferentes ao back-end de seu sistema baseado em regras de acesso.

41. Qual a função oferecida pelo serviço Cognito na AWS?

R: O Amazon Cognito fornece autenticação, autorização e gerenciamento de usuários para suas aplicações Web e móveis.

42. Como é a interação de uma aplicação com o Cognito para autenticação?

R: A AWS possui um serviço desses, chamado Cognito, a quem você pode optar por delegar todo esse processo de autenticação. Este artigo, tem a intenção de propor uma integração de uma aplicação usando React com o AWS Cognito.

43. O que é autenticação por Social Id oferecida pelo serviço Cognito na AWS?

R: O Cognito pode delegar a autenticação para um chamado "Provedor de Identidade" (há vários: iPhones, Facebook, Instagram, Amazon, Google etc.). Essa é uma tendência vantajosa para usuários (pois reduz o número de cadastros que ele precisa fazer) e sistemas (que terceirizam a

guarda das senhas e reduz muito os procedimentos de recuperação de senhas perdidas).

## Processamento Serverless

44. O que é processamento Serverless?

R: Serverless é um modelo de desenvolvimento nativo em nuvem para criação e execução de aplicações sem o gerenciamento de servidores.

45. Que é processamento em containers?

R: Os containers proporcionam o isolamento de processo, que permite configurar o uso da CPU e da memória de maneira detalhada para melhor utilização dos recursos de computação.

46. Qual a vantagem de se usar o Lambda?

R: A grande vantagem em usar Lambda é diminuir a quantidade de código necessária para escrita de algumas funções.

47. O que é o boto3?

R: O boto3 é fornecido com "waiters", que pesquisam automaticamente alterações de status predefinidas nos recursos da AWS.

48. O que é necessário para iniciar a execução de uma função Lambda?

R: Insira a função LAMBDA. Pressione F2 para editar o texto e impedir a inserção automática de referência de célula. Por exemplo: Para criar a função LAMBDA, selecione OK.

49. O que é necessário para uma função Lambda poder acessar um serviço AWS?

R: É necessário que a função Lambda tenha uma política ("role" ou "função") em JSON especificando o tipo de acesso permitindo para a operação a ser executada em cada recurso. A política pode especificar o proprietário do recurso, a sua região, o tipo de recurso e qual o tipo de acesso permitido. É importante permitir apenas o suficiente para a execução da função.

50. Que tipo de processamento são comumente usados em lambda?

R: Ele usa uma arquitetura de **processamento** de dados paralelos controlada por eventos, ideal para workloads que precisam de mais de um derivativo de dados de um objeto.

51. Que acontece se um programa em lambda entra em loop e não termina nunca?

R: Dadas uma descrição de um **programa** e uma **entrada** finita, decida **se o programa** termina de rodar **ou** rodará indefinidamente.

52. Que acontece se mais de uma instância de um programa em lambda é chamada para executar?

R: Atingido o tempo limite configurado, a máquina virtual termina a execução do programa. Um alerta de log pode ser enviado para registrar a ocorrência. É uma situação indesejada pois o tempo de processamento afeta o custo do uso do lambda.

53. Como é possível acessar um recurso da AWS de dentro de um programa em lambda?

R: A AWS disponibiliza uma biblioteca (uma API para Python, JS, Go, Ruby e C#) com classes e métodos que permitem interagir com a maioria dos serviços da AWS com programas em lambda chamada de BOTO3. Como o BOTO3, é possível criar objetos que manipulam com o S3, DynamoDB, EC2 etc.

54. Como um programa em lambda recebe valores do evento que o iniciou e como retorna valores para ele?

R: O lambda recebe valores, na forma de um documento JSON, em um parâmetro denominado event. Esse parâmetro normalmente inclui: dados sobre o evento, informações de autenticação (token JWT, por exemplo) e dados sobre a solicitação a ser atendida. Os valores retornados são, muitas vezes, opcionais e estão na forma de documento JSON.

55. Que são “step function” na AWS?

R: coordenando aplicações distribuídas e microserviços utilizando workflows visuais.

## Monitoração, Segurança e Automação

56. Cite algumas ferramentas de monitoração de custos da AWS?

R: Serviço para orquestração e gerenciamento de um conjunto de funções Lambda que interagem e precisam se coordenar para serem executadas.

57. Cite algumas ferramentas de monitoração de desempenho da AWS?

R: Cost Explorer, Budgets e Cost Report.

58. Como funcionam o controle de permissões na AWS?

R: Para atribuir permissões a usuários federados, você pode criar uma entidade denominada função e definir permissões para a função

59. Como as permissões são descritas na AWS?

R: A entidade principal é o usuário que faz as solicitações no console. Em geral, para realizar uma ação, você deve ter somente a ação correspondente incluída em uma política. Para criar



um usuário, você precisa de **permissão** para chamar a ação CreateUser .

60. O que são permissões baseadas em recursos na AWS?

R: As **permissões** no nível do **recurso** referem-se à capacidade de usar ARNs para especificar **recursos** individuais em uma política.

61. O que é Infrastructure as Code – IaC?

R: Tem como objetivo automatizar o provisionamento da infraestrutura de **TI**.