**COMPUTAÇÃO EM NUVEM**

AULA 4

Prof.ª Ana Paula Costacurta

**CONVERSA INICIAL**

No Tema 1, estudaremos o *Amazon Elastic Block Storage* (EBS), um serviço de armazenamento em bloco integrado com EC2. Conheceremos os tipos de volumes e os *Snapshots*,que são *backups* incrementais dos dados. Serão apresentados os serviços de dados disponibilizados pela AWS.

No Tema 2, veremos o *Amazon Simple Storage* (S3), um serviço de armazenamento que é apenas um repositório de arquivos, não é integrado com EC2. Estudaremos as várias categorias de classes de armazenamento do S3. Serão apresentados também os tipos de hospedagens web que são disponibilizados pelo S3.

No Tema 3, estudaremos *Amazon Relational Database Service* (RDS), um serviço de banco de dados relacional com possibilidade de criar instâncias sob demanda ou reservadas. Conheceremos as instâncias e classes de instâncias que podem conter vários Bancos de Dados (DB) e que são criadas em um ambiente separado na nuvem. Veremos as limitações para criação das instâncias. Serão apresentadas as características de cada tipo de instância e suas características. Também aprenderemos sobre os tipos de armazenamento das instâncias disponibilizadas pelo RDS.

No Tema 4, aprenderemos sobre *Amazon Elasticache*, um serviço de armazenamento em *cache* para melhorar o desempenho na utilização dos servidores de aplicativos. Veremos detalhes dos mecanismos oferecidos *ElastiCache* para *Redis* e *MenCached* e será realizada uma comparação entre os mecanismos.

No Tema 5, estudaremos o *Amazon CloudFront,* um serviço de entrega de conteúdo (CDN) para acelerar a entrega do conteúdo utilizando os pontos de presenças da AWS. Serão apresentadas as configurações básicas, as formas de distribuições, os mecanismos de segurança e formas de garantir a disponibilidades utilizadas pelo *Amazon CloudFront*.

**TEMA 1 – AMAZON ELASTIC BLOCK STORE (EBS)**

A EBS oferece serviço de armazenamento de blocos com alta *performance* e fácil de usar com Amazon EC2. Possui quatro tipos de volumes diferentes, podendo ajustar performance e tamanho sem interrupção. Os benefícios de utilizar o EBS são: performance para carga, facilidade de uso, disponibilidade e resiliência, escalabilidade, segurança e economia.

**1.1 TIPOS DE VOLUMES**

São divididos em duas categorias:

**1) SSD**: ideal para cargas de trabalho com operações de leitura e gravação frequentes com pequeno E/S, desempenho é IOPS. Recomenda-se grandes cargas de trabalho, como bancos de dados e volume de inicialização.

**2) HDD**: ideal para cargas de trabalho de *streaming*, medidas em MiB/s. Recomenda-se para cargas de trabalho intensiva, como MapReduce e processamento de log.

Na tabela 1, visualizamos a características de cada tipo de volume.

Tabela 1 – Características dos tipos de volume EBS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Unidades de estado sólido (SSDs)** | | **Unidades de disco rígido (HDD)** | |
| Tipo de volume | O1 | gp2  (padrão) | ST1 | SC1 |
| Casos de uso | Bancos de dados NoSQL e relacionais com alto consumo de E/S | Volumes de inicialização, aplicativos interativos de baixa latência e desenvolvimento e teste | *Big data*, *data warehouses*, processamento de *logs* | Dados “mais frios” (menos acessados), que exigem menos verificações por dia |
| Nome da API | IO1 | GP2 | ST1 | SC1 |
| Tamanho do volume | 4 GB a 16 TB | 1 GB a 16 TB | 500 GB a 16 TB | 500 GB a 16 TB |
| Máximo de IOPS por volume | 64.000 | 16.000 | 500 | 250 |
| Máximo de *throughput*por volume | 1.000 MB/s | 250 MB/s | 500 MB/s | 250 MB/s |
| Máximo de IOPS/instância | 80.000 | 80.000 | 80.000 | 80.000 |
| Taxa de transferência máxima/Instância | 2.375 MB/s | 2.375 MB/s | 2.375 MB/s | 2.375 MB/s |
| Preço | 0,125 USD/GB/mês  0,065 USD/IOPS provisionadas | 0,10 USD/GB/mês | 0,045 USD/GB/mês | 0,025 USD/GB/mês |
| Atributo de performance dominante | IOPS | IOPS | MB/s | MB/s |

Fonte: AWS Brasil, 2020.

**1.2 BACKUPS (SNAPSHOTS)**

Possibilidade de criação de *backup*de um ponto no tempo de um volume EBS que chamamos de *Snapshot*, pode ser utilizado para *backup* de dados ou base para novos volumes, são *backups* incrementais. Como pode ser realizado mesmo o volume em uso, somente são capturados dados do volume no momento que o comando é emitido. Se for realizado um *snapshot* de um volume criptografado automaticamente, o *snapshot* é criptografado também. A Figura 1 mostra o funcionamento de como é realizado o *snapshot* incremental.

Com *Amazon Data Lifecycle Manager,* é possível realizar os *backups* de dados de forma simples e automatizada, os *snapshots* são limpos regularmente quando é criada uma política de ciclo de vida.

**1.3 SERVIÇO DE DADOS**

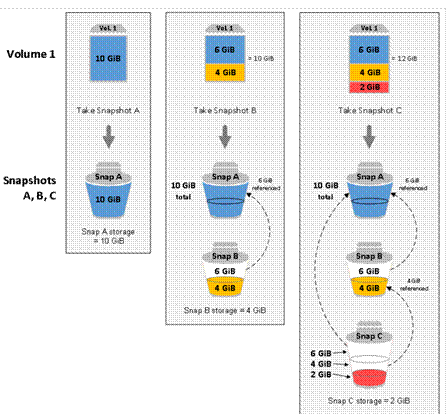
A *Amazon* EBS tem disponível os seguintes serviços de dados:

**1) Volumes elásticos**: facilidade de adaptar facilmente os volumes conforme a necessidade. É possível aumentar dinamicamente, ajudar performance e alterar tipo de volume sem ter que deixar o volume indisponível.

**2) Criptografia**: são utilizadas as chaves de cliente (CMK) do AWS *Key Management Service* (KMS) para criação e volumes criptografados, eliminando, assim, a necessidade de criar uma infraestrutura a parte para gerenciamento de chaves.

**3) Restauração rápida**: é possível criar um volume a partir de um *Snapshot*que já está configurado para inicialização.

Figura 1 – Funcionamento de *Snapshot* Incremental

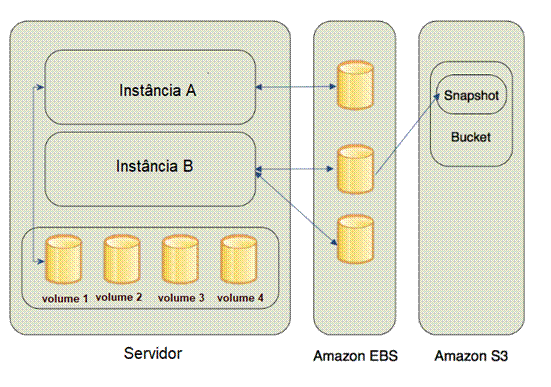


Fonte: AWS Brasil, 2020.

**TEMA 2 – AMAZON SIMPLE STORAGE SERVICE (S3)**

É o serviço de armazenamento de dados na nuvem. Diferente do *Amazon*EBS, o *Amazon* S3 não permite conectar com *Amazon* EC2. Na Figura 2, podemos ver as duas opções de armazenamento e a instância.

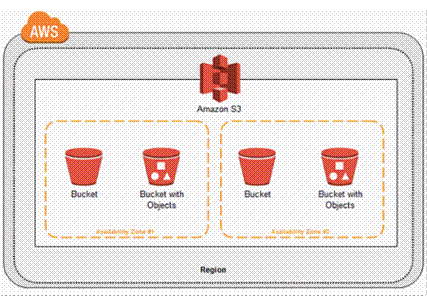
Figura 2 – Opções de armazenamento



Fonte: AWS Brasil, 2020.

O *Amazon*S3 é apenas um repositório remoto de armazenamento de arquivos. Na figura 3, podemos ver a estrutura do*Amazon* S3.

Figura 3 – Estrutura*Amazon* S3



Fonte: AWS Brasil, 2020.

Quando falamos em *Amazon* S3, dois termos aparecem com bastante frequência:

**1) *Bucket*(balde):** um *container* para objetos que são armazenados.

**2) Objetos:**metadados (conjunto de nome e valor) e dados das entidades que são armazenadas, a identificação em um *bucket*é feita por meio de uma chave (nome) e um ID.

**2.1 CLASSES DE ARMAZENAMENTO**

O *Amazon* S3 possui tem várias categorias de armazenamento para casos de uso diferenciados. Podemos subdividir essas categorias em quatro grandes categorias de armazenamento:

**1) S3 *Standard***: armazenamento de dados com muito baixo custo com alta disponibilidade, é o mais utilizado e caso você não atribua nenhuma classe de armazenamento quando realizar o upload, esse será por padrão selecionado.

**2) Acesso desconhecido ou dinâmico**: armazenamento para otimização de custos, funcionando em dois níveis: otimizado e custo baixo. Foi projetado para automaticamente mover os dados para o nível econômico sem impactos. A Classe S3 *Intelligent-Tiering* monitora os padrões de acesso infrequentes e automaticamente move para acesso frequente acaso o dado seja acessado, sem ter taxa de recuperação quando os objetos são movidos entre níveis.

**3) Acessos infrequentes**: indicado para *workloads* e objetos maiores que 128KB e sem uso pelo menos 30 dias, caso não seja utilizado por 30 dias será cobrado o valor de 30 dias. O custo mais baixo cobra uma taxa de recuperação e o valor mínimo cobrado é de 128KB, porém, com penalidades financeiras para acesso antes de 30 dias. As classes S3 *Standard*-IA e S3 *One Zone* – IA estão disponíveis em acesso de milissegundos.

**4) Gladier**: armazenamento de dados históricos por um longo período de tempo. Custo baixo, porém, com penalidades financeiras para acesso antes de 90 dias. As classes S3 *Glacier* e S3 *Glacier Deep Archive* não podem ser acessados em tempo real, sendo necessário restaurar para poder acessar.

**2.2 HOSPEDAGEMWEB**

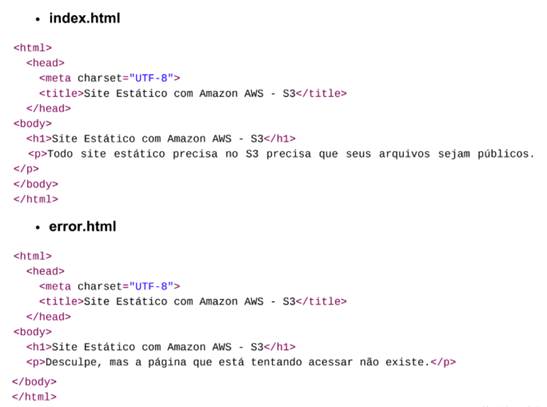
No *Amazon* S3, é possível hospedar um site, que podem ser criados basicamente dos seguintes tipos:

**1) Sites simples**:contêm um servidor *Content Management System* (CMS), que é um sistema de gerenciamento de conteúdo. Como sites de aplicativos comuns, podemos citar: *WordPress* e *Joomla*. São sites com tráfego médio a baixo, como site de marketing, sites de conteúdos e *blogs*.

**2) Aplicativos da web de página única**:precisam de uma carga em navegador, por meio de HTML, *JavaScript* e CSS, que são pré-carregados no navegador.

**3) Sites estáticos**: são siteseconômicos, que disponibilizam HTML, *JavaScript,* imagens, vídeos e outros arquivos. Para iniciar a criação de um *site*, estático é necessário criar apenas dois arquivos: index.html e outro error.html. Geralmente são sites pessoais ou de marketing. A imagem mostra o exemplo de código de cada um dos arquivos.

Figura 4 – Exemplo básico de código dos arquivos de sites estáticos



Fonte: Antunes, 2018.

**TEMA 3 – AMAZON RELATIONAL DATABASE SERVICE (RDS)**

Serviço webde banco de dados relacional gerenciado, utilizado para configuração, gerenciamento e dimensionamento de banco de dados relacional na AWS. Usando o RDS pode-se optar por usar instância de banco de dados reservadas ou sob demanda. É possível fazer *backups*, detecção de falhas, *patches* e recuperação. Apesar de oferecer um serviço gerenciado, não fornece acesso *Shell*e restringe alguns procedimentos e tabelas. O *Amazon*RDS oferece os produtos: MySQL, MariaDB, PostgreSQL, *Oracle* e MicrosoftSQL.

**3.1 Instâncias**

A instância *Amazon*RDS é um ambiente separado na nuvem. Uma instância pode conter vários bancos de dados (DB) e ser gerenciada pelas mesmas ferramentas que já são utilizadas para banco de dados autônomos. É possível criar até dez instâncias de banco de dados, com algumas limitações:

1) Com licença incluída até dez de cada edição SQL *Server*e dez *Oracle*.

2) 40 MySQL, MariaDB ou PostgreSQL.

3) Sem licença incluída até 40 *Oracle* quando a organização possui a licença.

A instância de DB possui um identificador que é utilizado com parte do nome do DNS, por exemplo, DB com identificado db1 e identificador da região 123456789012, o DNS será db1.123456789012.us-east-1.rds.amazonaws.com.

**3.2 CLASSE DE INSTÂNCIAS**

A capacidade de computação e memória da instância é determinada pela classe da instância. A Amazon RDS oferece três tipos de classes de instâncias: Padrão, com memória otimizada e desempenho ampliado. A seguir, veremos em detalhes cada tipo de classe:

**1) Padrão**: são instâncias para uso geral, são: M5 e M4.

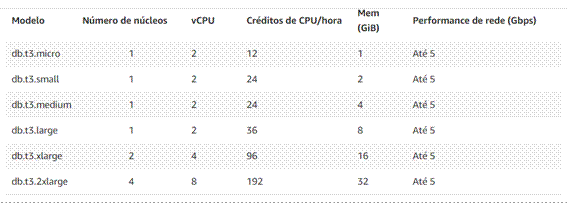
**2) Memória otimizada**: são instâncias para uso intensivo de memória, são: R5, R4, X1e, X1 e Z1d.

**3) Desempenho ampliado**: são instâncias para uso total da CPU, são: T3 e T2.

**3.3. INSTÂNCIAS T3**

São instâncias de uso geral com capacidade de intermitência, oferecem equilíbrio entre recursos de computação, memória e rede. Ideais para trabalhos com picos temporários de utilização. Possuem processador Intel (AVX, AVX2 ou Turbo) de 2.5 GHz com otimização para EBS. Na figura 5, podemos ver as características dos modelos das instâncias do Tipo T3.

Figura 5 – Características dos modelos de instância T3

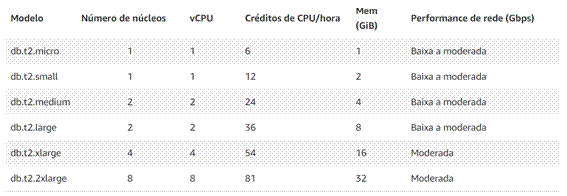


Fonte: AWS Brasil, 2020.

**3.4 INSTÂNCIA T2**

São instâncias de uso geral com capacidade de intermitência, sendo uma boa escolha para microsserviços e banco de dados de teste e preparação. Possuem processador Intel (AVX ou Turbo) até 3.3 GHz. Na figura 6, podemos ver as características dos modelos das instâncias do tipo T2.

Figura 6 – Características dos modelos de Instâncias T2

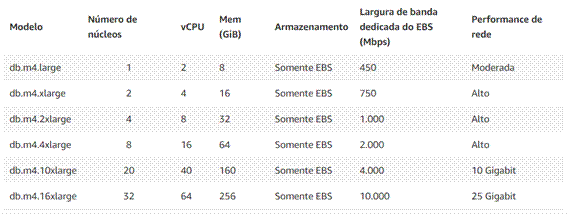


Fonte: AWS Brasil, 2020.

**3.5. INSTÂNCIAS M4**

São instâncias com equilíbrio entre recursos de computação, memória e rede com otimização para EBS, sendo uma boa escolha para banco de dados de médio tamanho e aplicativos de código aberto ou empresariais. Possuem processador Intel Xeon E5-2676 V3 (AVX, AVX2 ou Turbo) até 2.4 GHz. Na figura 7, podemos ver as características dos modelos das instâncias do tipo M4.

Figura 7 – Características dos modelos de instâncias M4

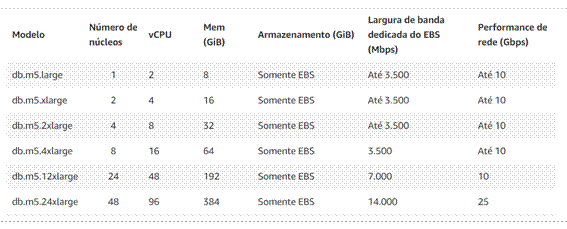


Fonte: AWS Brasil, 2020.

**3.6 INSTÂNCIAS M5**

São instâncias de uso geral com melhor performanceque a M4. Oferecem equilíbrio entre recursos de computação, memória e rede, sendo uma boa escolha para banco de dados. Possuem processador Intel Xeon Platinun 8175 (AVX, AVX2 ou Turbo) até 2.5 GHz com otimização para EBS. Na figura 8, podemos ver as características dos modelos das instâncias do tipo M5.

Figura 8 – Características dos modelos de instâncias M5

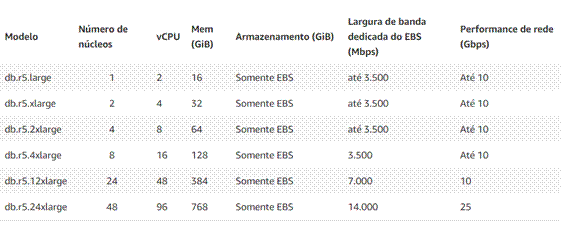


Fonte: AWS Brasil, 2020

**3.7 INSTÂNCIAS R5**

São instâncias otimizadas que disponibilizam 5% por vCPU a mais de memória, 10% por GiB de aprimoramento de preço e 20% de aumento de performance que as R4, até 768 GiB de memória. Possuem processador Intel Xeon Platinun 8000 (AVX, AVX2 ou Turbo) até 3.1 GHz com otimização para EBS. Na figura 9, podemos ver as características dos modelos das instâncias do tipo R5.

Figura 9 – Características dos modelos de instâncias R5

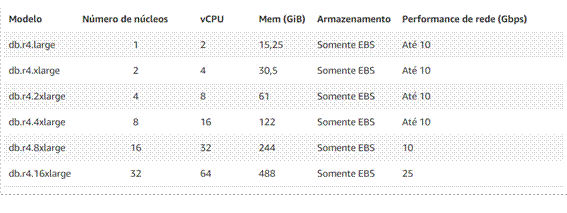


Fonte: AWS Brasil, 2020.

**3.8 INSTÂNCIAS R4**

São instâncias otimizadas para utilização intensiva de memória com preços melhores que as R3 (geração anterior), que disponibilizam 5% por vCPU a mais de memória, 10% por GiB de aprimoramento de preço e 20% de aumento de performance que as R4, até 768 GiB de memória DDR4. Possuem processador Intel Xeon E-2686 v4 Broadwell (AVX, AVX2 ou Turbo) até 2.3 GHz com otimização para EBS. Na figura 10, podemos ver as características dos modelos das instâncias do tipo R4.

Figura 10 – Características dos modelos de instâncias R4

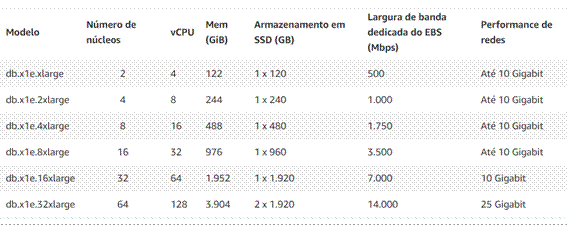


Fonte: AWS Brasil, 2020.

**3.9 INSTÂNCIAS X1E**

São instâncias otimizadas para alta performance de banco de dados com preços melhores entre todas as instâncias da *Amazon*RDS. Possuem processador Intel Xeon E7-8880 v3 Haswell (AVX ou AVX2) até 2.3 GHz com otimização para EBS e somente a x1e.16xlarge e a x1e.32xlarge têm Intel turbo. Na figura 11, podemos ver as características dos modelos das instâncias do tipo X1e:

Figura 11 – Características dos modelos de instâncias X1e

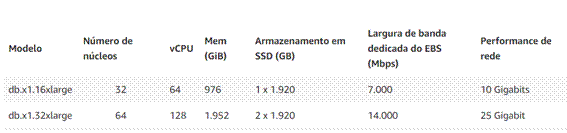


Fonte: AWS Brasil, 2020.

**3.10 INSTÂNCIAS X1**

São instâncias otimizadas para aplicativos de grande escala ou empresariais que ficam em memória, oferecem um melhor preço entre as instâncias da Amazon RDS. Possuem processador Intel Xeon E7-8880 v3 *Has well*(AVX, AVX2 ou Turbo) até 2.3 GHz com otimização para EBS. Na figura 12, podemos ver as características dos modelos das instâncias do tipo X1.

Figura 12 – Características dos modelos de instâncias X1

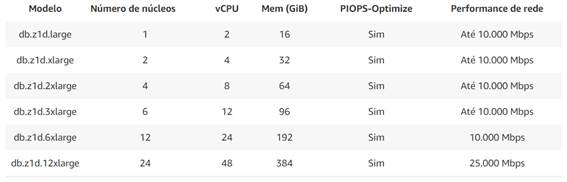


Fonte: AWS Brasil, 2020.

**3.11 INSTÂNCIAS Z1D**

São instâncias otimizadas que oferecem frequências até 4.0GHz em todos os núcleos de forma sustentada. Possuem alta performance e grande quantidade de memória, sendo ótimas para banco de dados relacionais de alto custo de licenciamento por núcleo. Possuem processado IntelXeon Scalable personalizado (AVX, AVX2 ou Turbo) até 4.0 GHz com otimização para EBS. Na figura 13, podemos ver as características dos modelos das instâncias do tipo Z1d.

Figura 13 – Características dos modelos de instâncias Z1d



Fonte: AWS Brasil, 2020.

**3.12 ARMAZENAMENTO DA INSTÂNCIA**

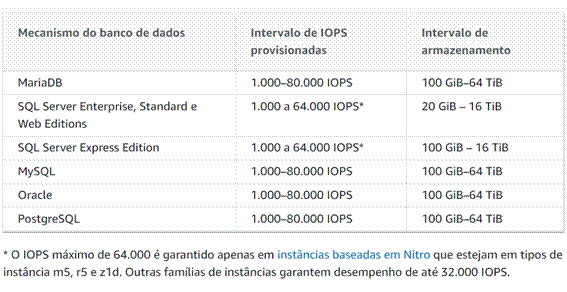
O *Amazon* RDS disponibiliza três tipos de armazenamento: SSD geral, SSD IOPS e magnético. É possível criar instâncias de DB MySQL, MariaDB, *Oracle* e PostgreSQL com capacidade de até 64TiB de armazenamento. DB SQL *Server* com capacidade até 16TiB de armazenamento. A seguir uma descrição resumida dos três tipos de armazenamento.

**1) SSD geral**: são armazenamentos econômicos, com capacidades entre 20 GiB-64 TiB para instâncias MariaDB, MySQL, *Oracle* e PostgreSQl e 20 GiB-16 Tib para SQL *Server*. Com três IOPS para cada GiB de desempenho de E/S, sendo 100 IOPS o mínimo. O desempenho aumenta conforme o tamanho do armazenamento.

**2) SSD IOPS**: são armazenamentos para atender cargas intensivas de E/S, oferece desempenho previsível e latência baixa. Na figura 14, mostramos os intervalos de IPOS para cada tamanho de armazenamento e mecanismo de DB.

**3) Magnéticos**: armazenamentos utilizados para retrocompatibilidade. Para qualquer novo armazenamento, o ideal é utilizar SSD geral ou SDD IOPS. O armazenamento magnético não permite escalar armazenamento com mecanismos DB do SQL Server e também não tem compatibilidade com escalabilidade automatizada de armazenamento. O limite é de 3 TiB e máximo 1.000 IOPS.

Figura 14 – Intervalo de armazenamento e intervalo de IOPS



Fonte: AWS Brasil, 2020.

**TEMA 4 – AMAZON ELASTICACHE**

O desempenho do servidor de aplicativo pode ser melhorado com a utilização de armazenamento em *cache*. O *Amazon ElastiCache* é um serviço de armazenamento em *cache*, pode ser gerenciado totalmente os armazenamentos de dados na memória. Com possibilidade de configurar, executar e escalar os dados na memória com recuperação de armazenamento de alto *throughput* e baixa latência. Na Figura 15, podemos observar o funcionamento do *ElastiCache*.

Dados que estão armazenados em memória *cache* podem ser entregues com muito mais rapidez. O *Amazon ElastiCache* oferece dois mecanismos: Redis, para aplicativos em tempo real, e *MenCached*, para aplicativos com uso intensivo de dados

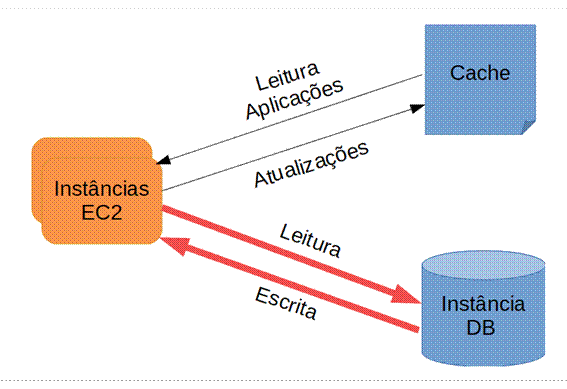
Os benefícios de utilização do *Amazon ElastiCache* são: performanceextrema, gerenciado e escalonável.

**4.1 *ELASTICACHE* PARA REDIS**

*Remote Dictionary Server* (Redis) é uma opção popular para armazenamento em cache que combina velocidade, simplicidade e versatilidade para suportar aplicativos em tempo real como jogos, comércio eletrônico, saúde, serviços financeiros e IOT.

Quando utilizado o Redis, todos os dados estão em memória principal do servidor, com isso eliminando a necessidade de acesso a discos. Oferece um conjunto avançado de recursos que faz com que seu uso seja eficaz para uma ampla variedade de utilizações.

Figura 15 – Funcionamento ElastiCache



Possui uma variedade de estruturas de dados como: *String*, Listas, Conjuntos, Conjuntos ordenados, *Hashes*, *Bitmaps*, *HyperLogLogs*. Com a simplificação do código que é necessário fornece simplicidade e facilidade de uso.

Os usos mais populares da *ElastiCache* para Redis são:

1) Armazenamento em *cache*;

2) *Chat*, sistema de mensagens e filas;

3) Placares de jogos;

4) Armazenamento de sessões;

5) *Streaming* de mídia avançada;

6) Dados geoespaciais;

7) *Machine learning.*

O Redis suporta as linguagens *Python*, *Java*, PHP, *Perl*, *Go*, *Ruby*, C/C#/C++, *Java Script* e Node.js.

**4.2 ELASTICACHE PARA MENCACHED**

Serviço de armazenamento de valor-chave e mantém os dados em memória. É uma boa escolha para utilizar como *cache*ou *DataStore*, com alta performance, utilizado em aplicativo em tempo real para web, aplicativos *mobile*, jogos e comércio eletrônico. Os casos de uso mais populares são armazenamento em cache e de sessão.

Os benefícios de utilizar o *ElastiCache* para *MenCached* são:

**1) Performance extrema**: como um *datastore* em memória e *cache* para suporte aos aplicativos com tempo de resposta de menos de um milissegundo.

**2) Seguro e protegido**: com suporte a *Amazon* VPC e pode ser isolado o *cluster*.

**3) Compatível com *MenCached***: sistema de armazenamento largamente adotado.

**4) Facilmente escalável**: até 20 nós e 12,7 TiB por *cluster*.

**5) Totalmente gerenciado**: não é necessário executar as tarefas de gerenciamento, *ElastiCache* monitora para manter as cargas de trabalho.

**6) Descoberta automática**: com *Auto Discovery* simplifica a maneira do aplicativo conectar-se ao *cluster* do *MenCached*.

O *MenCached* suporta as linguagens e protocolos: *Python*, *Java*, PHP, C/C#/C++, *Perl*, *Go*, *Ruby*, *JavaScript,* Node.js, Protocolo ASCII, Protocolo binário e Protocolos TCP e UDP.

**4.3 COMPARAÇÃO ENTRE REDIS E *MENCACHED***

O *MenCached* prioriza a simplicidade e o Redis possui um sofisticado conjunto de recursos. Na Tabela 2, são apresentadas as diferenças entre os dois mecanismos, com os principais requisitos para a escolha da melhor solução para as necessidades do uso pretendido.

Tabela 2 – Comparação entre *MenCached* e Redis

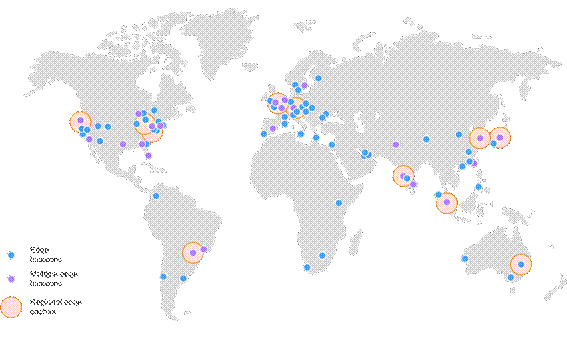
|  |  |  |  |
| --- | --- | --- | --- |
| **Requisito** | **Descrição** | ***MenCached*** | **Redis** |
| Latência de menos de milissegundos | Leitura de dados na memória mais rapidamente | Sim | Sim |
| Facilidade de uso do desenvolvedor | Quantidade de código mínima para integração com aplicativo | Sim | Sim |
| Particionamento de dados | Distribuição de dados em vários nós | Sim | Sim |
| Suporta várias linguagens de programação | Vários idiomas suportados | Sim | Sim |
| Estruturas de dados avançadas | Listas, conjuntos, conjuntos classificados, *hashes*, matrizes de *bits* e *hiperloglogs* | - | Sim |
| Arquitetura *multithread* | Utilização de vários núcleos de processamento, aumentando a capacidade de computação | Sim | - |
| Instantâneos | Manter dados com um instantâneo pontual para arquivamento ou recuperação | - | Sim |
| Replicação | Criar réplicas de um primário | - | Sim |
| Transações | Execução de grupo de comandos com operação isolada ou atômica | - | Sim |
| *Pub*/*Sub* | Publicação/Assinatura para utilização em sala de bate papo e comentários em tempo real | - | Sim |
| *Script* Lua | A linguagem Lua é poderosa, eficiente, leve e incorporável, podendo aumentar desempenho | - | Sim |
| Suporte geoespacial | Comandos especiais para dados geoespaciais em tempo real em escala | - | Sim |

Fonte: AWS Brasil, 2020.

**TEMA 5 – AMAZON CLOUDFRONT**

É um serviço de rede de entrega de conteúdo (CDN) para entregas de dados com segurança, baixa latência e velocidade alta de transferência. O *Amazon CloudFront* possibilita a aceleração da entrega do conteúdo se aproveitando dos pontos de presenças. Os conteúdos são obtidos na origem e armazenados em caches próximos aos usuários finais. Na figura 16, podemos verificar os pontos de presença disponibilizados pelo Amazon CloudFront.

Figura 16 – Pontos de presença do *Amazon CloudFront*



Fonte: AWS Brasil, 2020.

Crédito: KPPWC/Shutterstock.

**5.1 CONFIGURAÇÃO BÁSICA**

Para utilizar o *Amazon CloudFront* com uma configuração básica, é necessário realizar alguns procedimentos:

1) Os objetos da versão original devem ser armazenados em um *Bucket* da *Amazon*S3.

2) Conteúdos de *download* devem ser distribuídos como texto ou gráfico.

3) Objetos acessíveis a todos.

4) O nome de domínio para os objetos, por exemplo, http://d111111abcdef8.cloudfront.net/image.jpg.

5) Objetos devem ser mantidos por padrão de 24 horas nos pontos de presença.

**5.2 DISTRIBUIÇÃO**

É necessário criar uma distribuição e escolher as configurações desejadas para utilizar o *Amazon CloudFront* para distribuição de conteúdo.

As definições necessárias são:

**1) Origem de conteúdo**: a partir de qual origem o *CloudFront* irá obter os arquivos para realizar a distribuição.

**2) Acesso**: se deseja liberar acesso a todos ou restringir para alguns.

**3) Segurança**: se deseja que seja utilizado HTTPS para acesso ao conteúdo.

**4) Encaminhamento de *Cookie* ou *string* de consulta**.

**5) Restrição geográfica**: se deseja impedir acesso de alguma localização ao o conteúdo.

**6) Registros de acesso**: criação de *logs* para gravar as atividades de quem visualizou o conteúdo.

**5.3 SEGURANÇA**

O *Amazon CloudFront* pode ser utilizado como porta de entrada para o aplicativo e infraestrutura, garantindo a segurança do seu conteúdo critico, dados e infraestrutura.

Fornece criptografia SSL/TLS e HTTPS, e os recursos avançados de SSL são habilitados de forma automática. Os certificados AWS *Certificate Manager* (ACM) facilitam a utilização do certificado customizado e é gratuito. Os recursos avançados de SSL são: conexões HTTPS de meia/ponte total, grampeamento OCSP, *tickets* de sessão, *Perfect Forward Secrecy*, TLS *Protocol Enforcements* e criptografia em nível de campo.

Existe a possibilidade de restrição do acesso ao conteúdo com a utilização de vários recursos, podendo impedir acesso por regiões geográficas e deixar o conteúdo ser distribuído por meio do *CloudFront*.

Compatível com as regulamentações de segurança: PCI-DSS Nível 1 e ISO 9001, ISO 27001, SOC (1, 2 e 3), garantindo a segurança da entrega dos dados importantes que são confidenciais.

**5.4 DISPONIBILIDADE**

Por causa de picos de tráfegos, é possível utilizar o *Amazon CloudFront* para armazenamento em *cache* do conteúdo nos pontos de presenças ao redor do mundo reduzindo a carga da origem e com isso aumenta a disponibilidade do aplicativo.

Podem ser habilitadas várias origens para redundância da arquitetura. Pode ser configurado para fornecer o conteúdo de uma origem de *backup* quando a origem principal não estiver disponível com a utilização do recurso nativo chamado*failover* *de origem*.

**FINALIZANDO**

No Tema 1, aprendemos sobre *Amazon* EBS um serviço de armazenamento em bloco utilizado com integração com EC2. Conhecemos as duas categorias dos tipos de volumes: SSD e HDD. As unidades de estados sólido (SDD) têm dois tipos de volumes para alto consumo de E/S: o1 para banco de dados e gp2 para volumes de inicialização. As unidades de disco rígido (HDD) têm dois tipos de volumes para trabalho de *stremming*: sti1 dados quentes e sc1 dados frios. Aprendemos sobre os *Snapshots* para criação de *backups* incrementais, com possibilidade de utilização do *Amazon Data Lifecycle* *Manager* para automatização. Conhecemos os serviços de dados sendo eles: Volumes Estáticos, Criptografia e Restauração Rápida.

No Tema 2, estudamos sobre *Amazon* S3, um serviço de armazenamento que é apenas um repositório de arquivos. Dois termos importantes foram estudados: *Bucket* e Objetos. Conhecemos as quatro grandes classificações de armazenamento: S3 *Standart*, acesso desconhecido ou dinâmico, Acesso infrequente e *Gladier*. Vimos a possibilidade de realizar hospedagens web de sites, basicamente criados de três tipos: simples, página única ou estáticos.

No Tema 3, aprendemos sobre *Amazon* RDS, um serviço de banco de dados relacional com possibilidade de criar de instâncias sob demanda ou reservadas. As instâncias RDS podem conter vários Bancos de Dados (DB) e são criadas em um ambiente separado dentro da nuvem. Apresentamos a limitação de criação de dez instâncias tendo algumas restrições conforme o mecanismo utilização: MySql, MariaDB, PostgreSQL, *Oracle* e MicrosoftSQL. As instâncias são divididas em três classes: Padrão (M5, M4), memória otimizada (R5, R4, X1e, X1, Z1d) e desempenho ampliado (T3 e T2). Conhecemos as características de cada tipo de instâncias. Estudamos as formas de armazenamento da instância em que foram vistos três tipos de armazenamento: SSD geral, SSD IOPS e magnéticos.

No Tema 4, estudamos o *Amazon ElastiCache*, um serviço de armazenamento em cache para melhorar o desempenho da utilização dos servidores de aplicativos. O *Amazon ElastiCache* possui dois mecanismos: Redis e *MenCached*. O *ElastiCache* para Redis possui um sofisticado conjunto de recursos e o *ElastiCache*para *MenCached* prioriza a simplicidade. Foi realizada uma comparação entre Redis e *MenCached*apresentados os requisitos de cada um dos mecanismos.

No Tema 5, aprendemos sobre o *Amazon CloudFront* um serviço de entrega de conteúdo (CDN) com objetivo de acelerar a entrega dos conteúdos, utilizando os pontos de presenças disponibilizados da AWS. Conhecemos as configurações básicas para utilização do *Amazon ClouFront* e as configurações para realizar a distribuição do conteúdo. Conhecemos a forma de realização da segurança com utilização de criptografia, certificado AWS *Certificate Manger* (ACM) de forma gratuita e possibilidade de restrição de acesso aos dados de regiões geográficas. O *Amazon CloudFront* pode ser a porta de entrada, realizado a distribuição dos conteúdos pelos pontos de presença, tornando a aplicação com alta disponibilidade e com redundância de arquitetura.

**REFERÊNCIAS**

ANTUNES, J. L. **Amazon AWS** : descomplicando a computação em nuvem. Casa do Código: Brasport, 2018.

AWS BRASIL. Disponível em: <https://aws.amazon.com/pt>. Acesso em: 28 out. 2020.