E-mail direcionado ao departamento de arquitetura SOLVIMM

Bom dia!

Conforme solicitado, segue projeção para soluções aplicáveis sob os conceitos e tecnologias Amazon Web Servisse, visando atender as demandas previamente identificadas junto ao Cliente **Notícias da Nuvem.**

Objetivo: Propor soluções customizadas, necessárias para migrar o serviço sob a plataforma WordPress hoje hospedado no provedor LocalWeb, para **AWS** considerando o aporte de novas funcionalidades e recursos disponíveis nesta plataforma. Agregando recursos diferenciados como: backup, redundância, auto escalonamento, performance, consumo sob demanda, segurança e monitoramento.

Considerando que a atual plataforma de hospedagem do cliente não dispõe de recursos tecnológicos voltados para o mínimo contingenciamento e segurança, optamos por reconstruir toda a infra baseada única e exclusivamente nas recomendações de melhores práticas e topologia indicada na documentação da AWS para este senário. Desta forma seguiremos o desenho para referência de arquitetura na integra conforme imagem abaixo:

Diagrama

Descrição gerada automaticamente

Referencias: Documentação de melhores práticas disponibilizadas pela AWS.

<https://github.com/aws-samples/aws-refarch-wordpress>

Criação de conta AWS em nome do respectivo cliente, devidamente configurada de acordo diretivas financeiras e objetivos alinhados, ativação de métricas via CloudWhatch incluindo alertas de segurança com base no faturamento e serviços habilitados.

Detalhamento do fluxograma estrutural:

01 – Os usuários da plataforma terão respostas para resolução de nome diretamente do serviço de DNS AWS (Route 53). Em seguida acesso ao conteúdo imediatamente disponível em cache via CDN que utilizando recursos das zonas ed-location mais próximas, melhorando a experiencia de navegação.

02 – Dentro da região escolhida (us-east-1), teremos uma camada Internet Gateway redirecionando todas as entradas e saída dentro da VPC padrão da conta.

03 – Sub-redes públicas (IP externo) para acesso, seguro a infraestrutura aplicando o objeto NAT gateway (para acesso via SSH).

04 – A estrutura será replicada em duas AZ (Zonas de disponibilidades) devidamente gerenciadas através da ativação do serviço Application load balancer que fará toda a distribuição de carga e ativação das instâncias previamente configuradas através de modelos EC2, automatizadas via aplicação do recurso de Auto Scaling.

05 – App subnet. Instância WordPress diretamente conectada ao net gateway para acesso administrativo, aplication load Balancer e auto Scallin para o devido monitoramento, gerenciamento e automações.

06 – Data Subnet. Objeto Meam cached para ganho de performance para consulta direta em cahce de dados recorrentes diminuindo a carga de processamento e consultas diretas ao banco de dados.

07 – Banco de dados padrão Amazon RDS (Aurora) com recursos nativos de master e read replica aumentando a segurança e contingenciamento para estrutura dos dados.

08 – EFS Mount Target o qual será o ponto de montagem do storage para armazenamento dos arquivos da plataforma WordPress.

09 - Storage para as imagens do site agregando os recursos de backup, possibilidade de gerenciamento de ciclo de vida das informações através do Elastic File System possibilitando o crescimento sob demanda e utilizações de áreas de dados com menor custo para dados infrequentes.

Instâncias gerenciadas:

Type: c5.xlarge | vCPU: 04 | Memory: 08

Instance Storage: EBS only - EBS-Optimized: Yes - Net Performace: Up to 10 Gigabit

Inst-001: 4CPU – 8GB RAM (Estágio A).

Inst-002: 4CPU – 8GB RAM (Estágio B).

Inst-003: 4CPU – 8GB RAM (Estágio C).

Inst-004: 4CPU – 8GB RAM (Estágio D).

Inst-005: 4CPU – 8GB RAM (Estágio D).

Estágios operacionais (Auto Scaling):

Estágio A - CPU e MEMÓRIA com redução aproximada em torno de **30%** para ajustar ao período ocioso e de baixa demanda.

Estágio B – CPU e MEMÓRIA para atender período de pico imprevisíveis com margem para que o sistema não corra o risco de parar gerando a perda de **20%** dos visitantes. (Inst-002 iniciado a 92% de consumo dos recursos da Inst-001 via Auto Scaling).

Estágio C – CPU e MEMÓRIA para atender período de pico imprevisíveis com margem para que o sistema não corra o risco de parar gerando a perda de **20%** dos visitantes. (Inst-003 / Inst-004 iniciado a 92% de consumo dos recursos da Inst-002 via Auto Scaling).

Estágio D – CPU e MEMÓRIA para atender o crescimento previsto na ordem de 50% dos visitantes no período de final de ano. (Inst-005 iniciada no período definido como período de maior demanda de 10/11 a 31/12 via Auto Scaling).

**Instância principal** (Inst-001): Wordpress on Ubuntu 18.04 - Root Device Type: ebs Virtualization type: hvm Hourly Software Fees: $0.20 per hour.on c5.xlarge instance. Additional taxes or fees may apply.

**Instâncias de escalonamentos** e redundância conforme demanda programada (Inst-002, Inst-003, Inst-004, Inst-005).

Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type

Type: c5.xlarge | vCPU: 04 | Memory: 08 | Instance Storage: EBS only

EBS-Optimized: Yes - Net Performace: Up to 10 Gigabit

Processo de migração provedor LocalWeb para AMAZON WEB SERVICE:

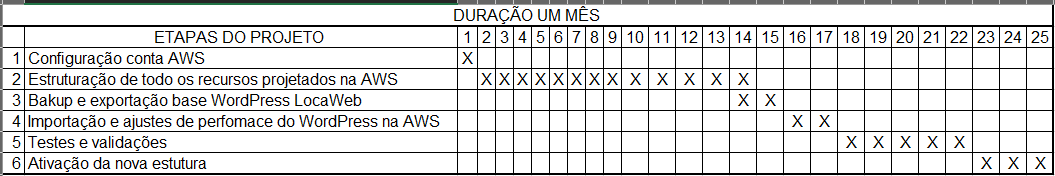
Transferência do domínio (Route 53), configuração do ip público, redirecionamento de registro cname para aws. Junto ao respectivo provedor de registro.

Migração do banco de dados: via painéis administrativos WordPress, com ativação do pluguin All-in-on Migration para exportação do banco de dados do site, repetindo este processo no destino com objetivo de importar os dados.

Monitoramento do Ambiente via cloudwatch:

Configurar painéis com métricas padrões para: Faturamento, EC2, EFS, EVENTOS, LOGS, RDS, S3 e ESTADOS.

Cronograma de execução:



<https://trello.com/b/ic5jFzYU/desafio-devops>

<https://gitlab.com/brandaonetwork/desafio-devops>

Atenciosamente,

Jorge Brandão Junior

Candidato.