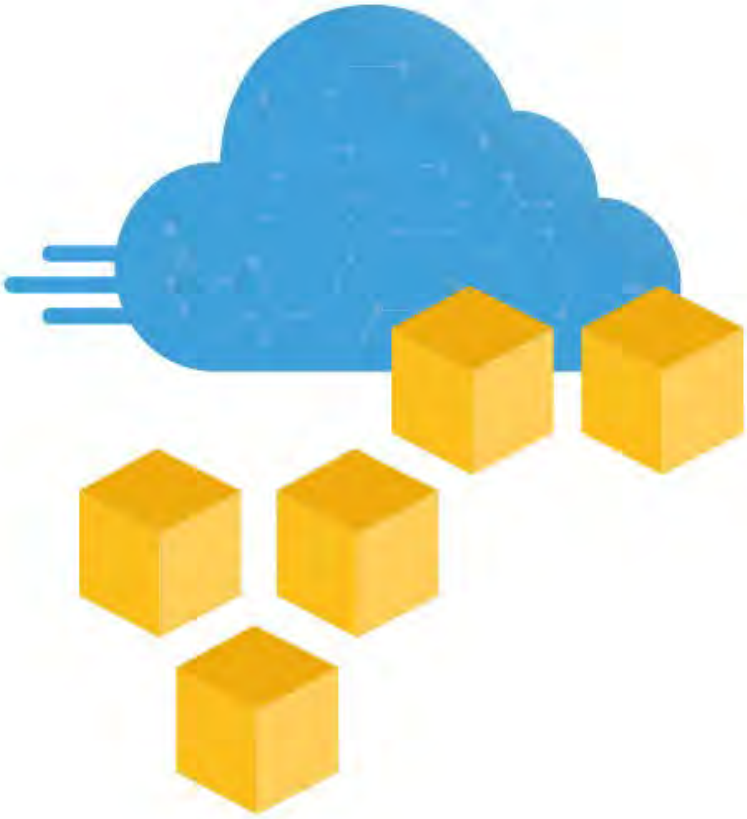
Amazon AWS

Descomplicando a computação na nuvem



Casa do Código

JONATHAN LAMIM ANTUNES

ISBN

Impresso e PDF: 978-85-5519-237-1

EPUB: 978-85-5519-238-8

MOBI: 978-85-5519-239-5

Você pode discutir sobre este livro no Fórum da Casa do Código: http://forum.casadocodigo.com.br/.

Caso você deseje submeter alguma errata ou sugestão, acesse http://erratas.casadocodigo.com.br.

SOBRE O AUTOR

Tudo começou em 2004, quando entrei para a escola técnica. Lá, estudei informática e aprendi sobre manutenção, redes de computadores e sistemas operacionais, mas o que me atraiu mesmo foram as matérias relacionadas à programação. Aprendi a programar usando Delphi, depois comecei a estudar JavaScript e HTML, e foi aí que me apaixonei por desenvolvimento para web.

Em 2005, concluí o curso técnico e me mudei do interior de Minas Gerais para o Espírito Santo, onde comecei a ter oportunidades de colocar em prática tudo o que já havia aprendido. Comecei então a escrever artigos sobre desenvolvimento web, dar aulas de programação e informática básica, e auxiliar alunos de uma escola particular durante as aulas no laboratório de informática.

Com o passar do tempo, fui me aprofundando nos estudos sobre desenvolvimento web, passei a colaborar com projetos open source e a visão foi se abrindo ainda mais. Quanto mais eu aprendia, mais eu queria ensinar e compartilhar. Já são mais de 300 artigos escritos,

muitas horas de aulas ministradas, várias palestras e hangouts, e ainda sinto que posso compartilhar muito mais conteúdo.

Hoje sou CEO da JLamim Tecnologia & Educação, onde trabalhamos com desenvolvimento web e mobile, cursos em diversas áreas, programas de mentoria, consultoria e SEO.

Para conhecer um pouco mais sobre o meu trabalho e meus artigos, veja os links:

- **Site:** http://www.jonathanlamim.com.br
- **Facebook:** https://www.facebook.com/JonathanLamimAntunes
- **Instagram:** http://instagram.com/jonathanlamim
- **Twitter:** http://twitter.com/jlamim

AGRADECIMENTOS

Este é o meu segundo livro, e mais uma vez tenho de agradecer a Deus por tudo que ele tem feito.

Mais uma vez deixo meus agradecimentos à editora Casa do Código e toda a sua equipe pelo excelente trabalho realizado em mais esta publicação.

Ao amigo Mikael Hadler, por ter aceitado o desafio de fazer a revisão técnica deste livro, contribuindo com o seu conhecimento para que o conteúdo das próximas páginas possa ser de fácil assimilação e aprendizado.

À minha esposa, Juliana, por ter tido ainda mais paciência durante todos esses meses em que eu trabalhei neste livro. Aos meus pais, irmãos, tios, padrinhos, amigos, por acreditarem no meu trabalho e me ajudarem a seguir em frente mesmo nos momentos em que o cansaço e o desânimo eram grandes.

Obrigado a você que está lendo esta obra, que acreditou que o conteúdo dela pode ser útil para o seu crescimento profissional.

SOBRE O LIVRO

Infraestrutura é um ponto muito importante quando se trata de grandes aplicações, pois é preciso pensar em escalabilidade, gerenciamento e, principalmente, os serviços necessários para o bom funcionamento dessas aplicações.

No decorrer deste livro, você encontrará informações e detalhes sobre diversos serviços da Amazon AWS. Eles possibilitarão a você hospedar e gerenciar facilmente aplicações dos mais variados tamanhos, fazendo uso de variados recursos e com um custo possível de ser controlado. Com a Amazon AWS, você pagará apenas pelos recursos que utilizar, o que torna o serviço ainda mais atraente e viável.

Se você precisa de uma infraestrutura robusta para rodar um aplicativo por tempo pré-determinado, então você pode alocar os recursos de infraestrutura necessários somente para esse período e, ao terminar, voltar para a infraestrutura inicial. Assim, você pagará o valor dessa infraestrutura adicional somente pelo período em que ela foi usada, sem a necessidade de ficar pagando um valor mais alto depois sem fazer uso dos recursos.

COMPOSIÇÃO DO LIVRO

Veja a seguir como está a composição do livro, os capítulos e o que é tratado em cada um.

Capítulo 1 — Introdução

Neste capítulo, falo em linhas gerais sobre o serviço da Amazon AWS, o cadastro, as regiões e o período de gratuidade que é dado aos novos usuários.

Capítulo 2 — Amazon S3

Amazon S3 (ou Amazon Simple Storage Service é um serviço de armazenamento de dados na nuvem. Seguro, escalável e com grande durabilidade, tornou-se uma das ferramentas mais utilizadas por desenvolvedores e equipes de TI para o armazenamento de arquivos, criptografados ou não.

Neste capítulo, veremos detalhes sobre o seu funcionamento, como configurar e usar os seus recursos para fazer upload de páginas estáticas de um site.

Capítulo 3 — Amazon S3 — Gerenciando operações com AWS SDK

O AWS SDK é uma ferramenta de integração utilizada no desenvolvimento de sistemas que fazem uso de recursos e serviços do AWS, e neste capítulo você verá exemplos de aplicação do AWS SDK para o AWS S3. Neste capítulo, veremos como usar o SDK em linguagens como NodeJS, Python e PHP para criar buckets e enviar arquivos.

Capítulo 4 — Amazon EC2 (Elastic Compute Cloud

O EC2 é o serviço que permite a você criar, na nuvem, instâncias de servidores virtuais usando diferentes configurações — isso inclui armazenamento, memória e processadores — e sistemas operacionais (Windows ou Unix. É em uma instância do EC2 que você vai colocar sua aplicação para funcionar.

Neste capítulo, veremos como trabalhar com o EC2, desde a configuração até o acesso via SSH.

Capítulo 5 — Amazon EC2 — Instalando o Apache, PHP e MySQL na instância

Neste capítulo vamos preparar a instância para hospedar nela um site feito com o CMS Wordpress. Vamos instalar passo a passo o Apache, o PHP e o MySQL na instância.

Capítulo 6 — Amazon EC2 — Instalando e configurado o Wordpress

Neste capítulo, realizaremos o passo a passo da instalação e configuração do Wordpress na instância configurada no capítulo 5.

Capítulo 7 — Amazon RDS (Relational Database Service)

O RDS é um serviço de banco de dados relacional disponibilizado pela Amazon que facilita o gerenciamento dos bancos de dados e tem como principais vantagens a escalabilidade e o autogerenciamento. Neste capítulo, veremos como criar e configurar instâncias de bancos de dados para usar em nossas aplicações.

Capítulo 8 — Amazon ElastiCache

O ElastiCache é um serviço da AWS para armazenamento de cache de memória na nuvem e que possui suporte para as bibliotecas de cache Redis e Memcached, permitindo um alto desempenho de gravação e leitura dos dados armazenados. Neste capítulo, veremos como configurar o ElastiCache e como utilizá-lo de maneira integrada ao Wordpress.

Capítulo 9 — Amazon Route 53

"O Amazon Route 53 é um serviço web de Domain Name System (DNS) altamente disponível e escalável". Neste capítulo, vamos ver como o Route 53 funciona e quais os processos necessários para obter o DNS de um site hospedado nos servidores da Amazon, e

como configurar domínios e subdomínios para que apontem para instâncias específicas do EC2.

Capítulo 10 — Amazon CloudFront

Em linhas gerais, o CloudFront é o serviço de CDN (Content Delivery Network, ou Rede de Distribuição de Conteúdo da Amazon. Ele tem a flexibilidade para se integrar com outros serviços da AWS oferecendo aos desenvolvedores e empresas uma entrega de conteúdo mais rápida e com grande facilidade de uso.

Neste capítulo, veremos como ele funciona e o passo a passo para configurarmos e usarmos o CloudFront.

Capítulo 11 — SES (Simple Email Service

O Amazon SES é um serviço para envio de e-mails utilizando o protocolo STMP, que pode ser escalado conforme a necessidade da aplicação e tem um custo pequeno se comparado a outros serviços do mesmo tipo. Neste capítulo, veremos como configurar e usar o SES para enviarmos e-mails através de nossas aplicações.

Capítulo 12 — Amazon SNS (Simple Notification Service

O SNS ou Simple Notification Service é o serviço da Amazon responsável por disparar notificações para usuários e endpoints. Neste capítulo, veremos como configurar e utilizar o SNS para o envio de notificações em nossas aplicações.

Capítulo 13 — Amazon CloudWatch

O Amazon CloudWatch é o serviço de monitoramento de recursos da AWS. Através dele, você poderá configurar diversos tipos de monitoramento de recursos, e até mesmo os gastos que sua

conta está gerando. Neste capítulo, veremos como configurar e utilizar o CloudWatch para monitorarmos as nossas instâncias e os gastos com a nossa conta.

Capítulo 14 — Gerenciamento de custos da conta

Neste capítulo, vamos conhecer cada parte do painel de gerenciamento de custos da conta, o seu funcionamento e configurações.

Apêndices

Este livro é complementado por 4 apêndices que servem como auxílio ao conteúdo dos 14 capítulos que serão estudados.

- Apêndice 1 — Instalando o AWS CLI e configurando as credenciais de acesso
- Apêndice 2 — Linux: comandos básicos
- Apêndice 3 — MySQL: comandos básicos
- Apêndice 4 — Como configurar o Apache para utilizar URLs amigáveis

DÚVIDAS & ERRATAS

Caso tenha dúvidas durante a leitura, você pode publicá-las no fórum da Casa do Código, em http://forum.casadocodigo.com.br/.

Caso você deseje submeter alguma errata ou sugestão, acesse http://erratas.casadocodigo.com.br

Boa leitura!

Sumário

Capítulo 1

INTRODUÇÃO

Esse é o ponto de partida da leitura e estudo sobre os serviços da Amazon AWS. Nesta introdução, você terá uma visão geral do serviço, como se cadastrar para poder utilizá-lo, aproveitar o período de gratuidade que é oferecido e vantagens em usar os serviços da Amazon AWS.

1.1 O QUE É A AWS?

AWS (ou Amazon Web Services) é uma plataforma de serviços na nuvem. Essa plataforma oferece soluções para armazenamento, redes e computação, em várias camadas. E o melhor de tudo, você pode administrar todos esses serviços através de uma interface web, ou também por APIs e linha de comando, depende do serviço que está utilizando no momento.

Dentre os mais variados serviços da AWS, os que mais se destacam são o EC2, que oferece servidores virtuais, e o S3, para armazenamento de arquivos. Além desses dois, todos os demais serviços da AWS funcionam muito bem juntos, pois foram projetados para poderem se integrar, de modo que você possa utilizar e administrar os mais variados recursos de infraestrutura da sua aplicação, de forma descomplicada e individualmente.

Uma característica importante dos serviços da AWS é que você paga somente pelo recurso usado; não há um valor mensal fixo. Seu custo será conforme a demanda, ou seja, se em um determinado

momento sua aplicação precisar de mais recursos, você os aloca e paga por eles enquanto estiver utilizando. Esse formato é conhecido como *pay-per-use*.

1.2 REGIÕES E ZONAS DE DISPONIBILIDADE

Outro ponto importante da AWS é a possibilidade de escolher a região onde vai utilizá-los. Dessa forma, você pode ter um tempo de resposta menor na sua aplicação. Hoje já temos datacenter da Amazon aqui no Brasil (que fica localizado em São Paulo), mas infelizmente os custos são maiores do que um datacenter em outra região.

Esses datacenters estão divididos por regiões (regions) e zonas de disponibilidade (*availability zone*):

- **Regiões:** é uma definição bem objetiva, pois o próprio nome já define. Representa a região do mundo na qual os datacenters estão localizados. Cada região possui dois ou mais datacenters, também chamados de zona de disponibilidade.
- **Zonas de disponibilidade:** é cada um dos datacenters, isolados, que existem dentro de uma região. Esses datacenters são independentes, mas com uma conexão bastante rápida para se comunicarem com datacenters AWS de outras regiões.

> É comum definir "zonas de disponibilidade" pelo termo AZ, pois é um acrônimo para AVAILABILITY ZONES.

Dentro de uma região existem diversos servidores responsáveis

por distribuir o conteúdo rapidamente, com uma baixa latência. Esses servidores recebem o nome de Pontos de Presença (*Edge Locations*).

Veja na tabela a seguir as regiões disponíveis e seus respectivos códigos:

Código	Região
us-east-1	US East (N. Virginia)
us-west-2	US West (Oregon)
us-west-1	US West (N. California)
eu-west-1	EU (Ireland)
eu-central-1	EU (Frankfurt)
ap-southeast-1	Asia Pacific (Singapore)
ap-northeast-1	Asia Pacific (Tokyo)
ap-southeast-2	Asia Pacific (Sydney)
ap-northeast-2	Asia Pacific (Seoul)
ap-south-1	Asia Pacific (Mumbai)
sa-east-1	South America (São Paulo)

1.3 VANTAGENS DA ARQUITETURA DA AWS

A arquitetura dividida em regiões e zonas de disponibilidade traz grandes vantagens para as aplicações instaladas nela. Como as regiões e zonas de disponibilidade se comunicam, você pode ter uma mesma aplicação rodando em diversas regiões e zonas diferentes. Dessa forma, o acesso será bem mais rápido, pois os servidores serão acionados conforme a localização dos usuários, usando a zona mais próxima dele e que possui o sistema instalado.

Você já teve problemas de indisponibilidade do servidor por conta de chuva, queda de árvore, falta de energia ou problemas de

hardware? Na AWS é possível evitar esse tipo de problema, pois como você pode ter a aplicação rodando em servidores diferentes dentro de uma região, quando um deles parar de responder, o outro entra em ação e faz a entrega das requisições para o usuário.

1.4 CADASTRO E USO GRATUITO

Além de ser bastante simples fazer o cadastro para utilizar a AWS, você ainda tem direito a um **nível de uso gratuito**. Esse nível de uso gratuito está disponível somente para novos clientes e é válido por 12 meses, a partir da data em que você se cadastrou.

Cadastro

Para se cadastrar e começar a usufruir dos recursos e serviços da AWS, acesse http://console.aws.amazon.com, e então crie uma nova conta.

Informe o e-mail e marque a opção *"I am a new user"*, e em seguida clique em *"Sign in using our secure server"*, conforme figura seguinte:

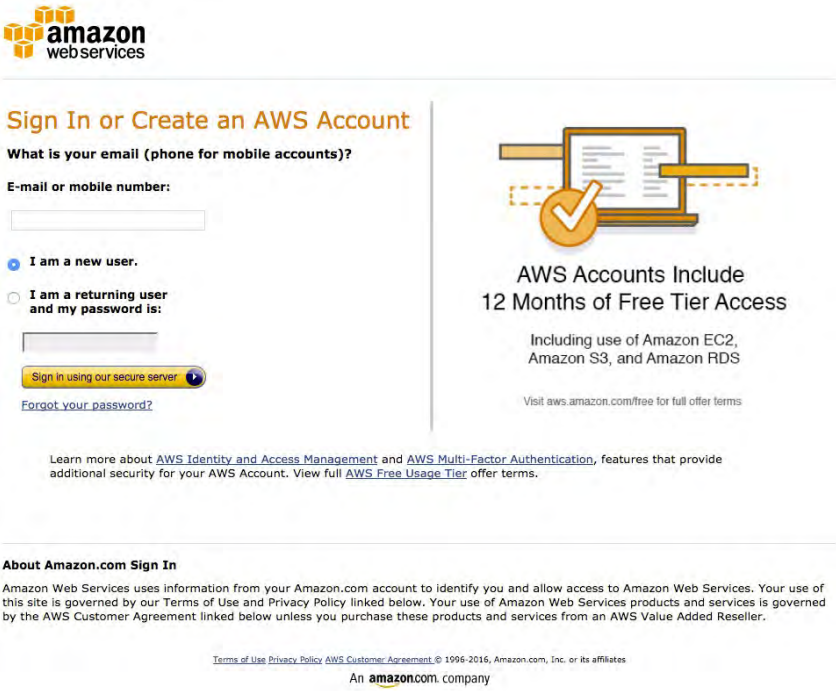


Figura 1.1: Criando uma conta na AWS — Passo 1

Em seguida, você deverá preencher o restante das informações no formulário e clicar em *"Create account"*.

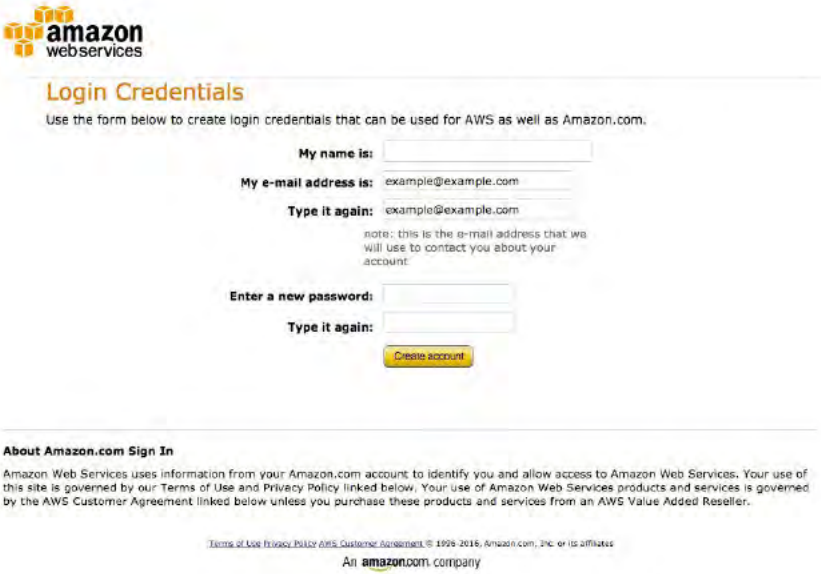


Figura 1.2: Criando uma conta na AWS — Passo 2

O restante do fluxo de pagamento é composto por formulários para que você forneça informações de cobrança e alguns outros dados. É bastante simples e não há dificuldades para concluir.

Uso gratuito

O uso gratuito é aplicado a vários serviços da AWS, cada um com suas características e limites. Veja a seguir algumas características dessa gratuidade para os principais serviços da AWS.

- **S3**
    - 5GB de armazenamento
    - 20.000 requisições GET
    - 2.000 requisições PUT
- **EC2**
    - 750 horas por mês de uso de instância t2.micro

Linux, RHEL ou SLES
    - 750 horas por mês de uso de instância Windows t2.micro

Por exemplo, execute 1 instância durante 1 mês inteiro ou 2 instâncias por duas semanas.

- **RDS**
    - 750 horas de uso de instância db.t2.micro Single-AZ do Amazon RDS
    - 20 GB de armazenamento de banco de dados: qualquer combinação de propósito geral (SSD) ou magnético
    - 20 GB para backups
    - 10.000.000 de E/S (Entrada e Saída)
- **Amazon CloudFront**
    - 50 GB de transferência de dados para fora
    - 2.000.000 Solicitações HTTP ou HTTPS
- **Amazon ElastiCache**
    - 750 horas por mês no cache.t2micro

Essa quantidade é suficiente para execução contínua durante o mês inteiro

Mais detalhes sobre o uso gratuito podem ser vistos em http://aws.amazon.com/pt/free/.

1.5 O AMBIENTE DE ADMINISTRAÇÃO

Após fazer o cadastro e se logar, a tela inicial do ambiente de administração, também chamado de console, deverá ser semelhante

à figura a seguir:

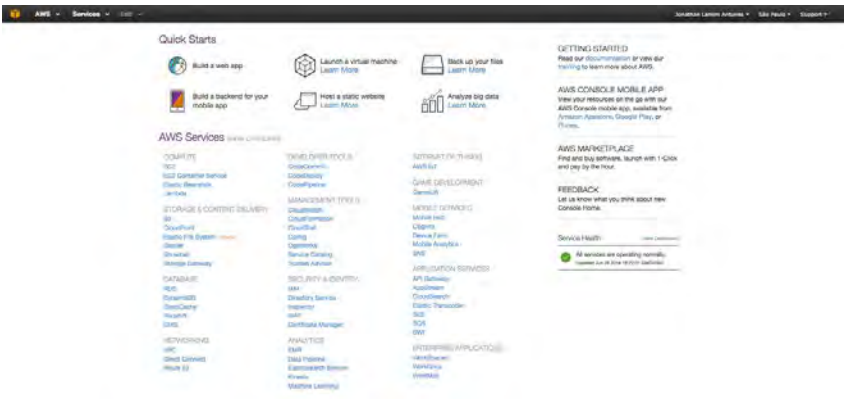


Figura 1.3: Tela principal do painel

Nessa tela, você tem acesso a todos os serviços disponíveis, suporte, configurações e atualização dos seus dados.

Clicando na opção *"SHOW CATEGORIES"*, você poderá filtrar a exibição dos serviços por categorias, o que facilita a localização do serviço a ser gerenciado.

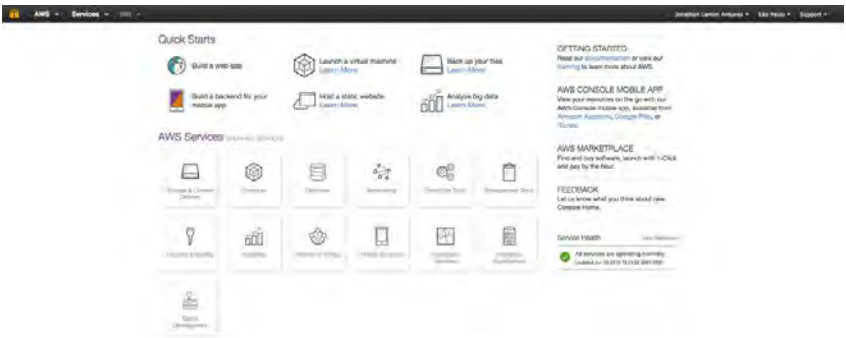


Figura 1.4: Tela principal do painel com filtro de categorias

Veja na próxima figura a aplicação do filtro para a categoria *"Storage & Content Delivery"*.

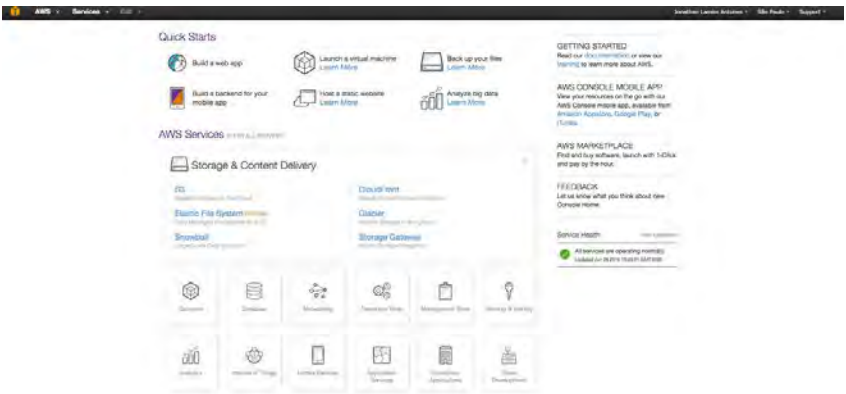


Figura 1.5: Tela principal do painel com os serviços filtrados por categoria

Personalizando a barra de menu do console

Além de você ter o acesso a partir da tela principal do console e do menu *"Services"*, é possível customizar o menu, criando atalhos para acessar as configurações de cada serviço. Para fazer isso, você deve clicar na opção *"Edit"* do menu, e na lista que se abrirá, clique e arraste o item desejado para a barra de menu. Faça isso com os serviços que você gerencia com mais frequência.

Figura 1.6: Barra de menu antes da criação dos atalhos

Figura 1.7: Barra de menu após a criação dos atalhos

Para remover um atalho da barra de menu, clique novamente em *"Edit"*, e então clique e arraste o item desejado para fora da barra.

1.6 CONSIDERAÇÕES FINAIS

Agora que você já tem uma ideia geral do que é a AWS, como funciona, já fez o seu cadastro e viu onde encontrar os acessos para os serviços, é hora de conhecer com mais detalhes alguns dos serviços. No próximo capítulo, você vai aprender a trabalhar com o S3, o serviço de armazenamento de arquivos na nuvem.

Links úteis

- **AWS Brasil:** http://aws.amazon.com/pt
- **FAQs:** http://aws.amazon.com/pt/faqs
- **Documentação:** http://aws.amazon.com/pt/documentation

Capítulo 2

AMAZON S3

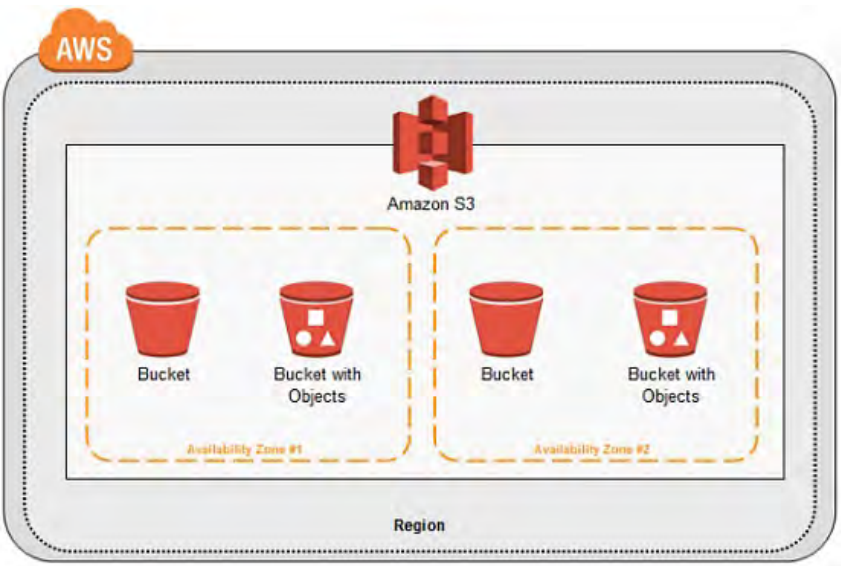


Figura 2.1: Arquitetura do Amazon S3

2.1 INTRODUÇÃO

Amazon S3 (ou Amazon Simple Storage Service) é um serviço de armazenamento de dados na nuvem. Seguro, escalável e com grande durabilidade, tornou-se uma das ferramentas mais utilizadas por desenvolvedores e equipes de TI para o armazenamento de arquivos, criptografados ou não.

O S3 possui um painel gráfico para gerenciamento do serviço, e também uma API que facilita muito a integração com outros

sistemas. Se você não precisa integrar com nenhum sistema, e vai subir manualmente os arquivos, pode utilizar o painel gráfico. Se precisa integrar com algum sistema, pode usar a API, que está disponível para diversas linguagens de programação, como Java, PHP, NodeJS, Python, entre outras.

Na estrutura do S3, você verá dois termos com bastante frequência:

- **Bucket** na tradução literal, significa *balde*, mas vamos trazer para o cenário deste livro, o que seria equivalente a uma pasta para armazenamento dos objetos.
- **Objetos** são os arquivos, ou seja, cada objeto armazenado corresponde a um arquivo, e junto com esse arquivo estarão armazenados os seus metadados.

> **Metadados:** são os dados de um arquivo, como por exemplo, o seu nome, tamanho e tipo.

A AWS limita o número de buckets por conta, e esse limite é de 100 buckets. Já o número de objetos em cada bucket é ilimitado, e cada objeto pode ter até 5TB de tamanho. Dentro de cada bucket, você pode criar várias pastas, então se começar a escalar esse limite de 100 buckets, você terá bastante espaço para armazenamento, de forma organizada.

2.2 CONSOLE DE GERENCIAMENTO

O console de gerenciamento é o ambiente que você acessa via browser através do login em http://console.aws.amazon.com, e que disponibiliza para você as funcionalidades necessárias para

gerenciar os buckets e os objetos armazenados.

Para acessar o S3, você pode ir pelo menu *Services*, ou então na tela principal do console, onde é exibida a lista de todos os serviços. De modo a agilizar o acesso, crie um atalho para o S3 na barra de menu, conforme foi ensinado no capítulo 1.

A tela principal do painel do S3 é a seguinte:

Figura 2.2: Tela principal do Amazon S3

Essa tela é exibida quando não há nenhum *bucket* criado. Ela traz informações sobre o S3 e o botão *Create Bucket*, que é acionado para a criação de uma nova pasta para armazenamento dos objetos.

2.3 O CICLO DE TRABALHO DO S3

O S3 possui um ciclo de trabalho bastante simples de ser compreendido. Ele é composto por 6 etapas, que vão desde o acesso ao serviço até a remoção de um objeto (arquivo, dado) e um bucket.

Veja na figura a seguir como é esse ciclo:

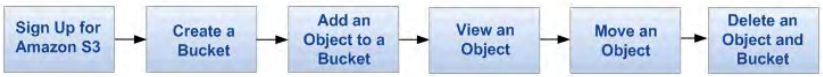


Figura 2.3: Ciclo de trabalho no S3

1. **Sign up for Amazon S3:** acesso ao serviço;
2. **Create a Bucket:** criação do bucket;
3. **Add an object to a Bucket:** adiciona um objeto ao bucket;
4. **View an Object:** visualização do objeto;
5. **Move an Object:** move o objeto dentro do bucket;
6. **Delete an Object and Bucket:** remove um objeto e o bucket.

Como disse, é um ciclo bastante simples e funciona tanto na interface gráfica quanto via API. Você aplicará esse ciclo agora, criando um bucket e gerenciando objetos dentro dele através da interface gráfica.

2.4 CRIANDO UM BUCKET

Para criar um bucket, você precisará definir um nome e uma região onde ele estará localizado. Dessas duas informações, a mais importante é a região, pois ela determinará a velocidade em que os arquivos serão acessados pelo usuário.

Se você está criando um bucket para disponibilizar arquivos para brasileiros, então o ideal é que este bucket tenha como região São Paulo. Assim a entrega será mais rápida, pois a proximidade entre os usuários e o datacenter é menor.

Se estiver criando um bucket em que os acessos aos arquivos não serão constantes, seria mais no nível de backup. Então você pode criar em uma outra região, de modo a obter custos menores. Uma dica para esse tipo de uso é criar o bucket em alguma região dos Estados Unidos, na qual o custo é menor.

Agora você criará o seu primeiro bucket. Clique no botão *Create Bucket*, e na tela que vai abrir, informe o nome de identificação desse bucket e a região em que ele deve estar, em seguida clique no botão *Create*.

> O nome do bucket deve ser definido com letras minúsculas, sem espaços e sem caracteres especiais.

Figura 2.4: Tela para criação de bucket

Após criar o bucket, a tela mudará e você passará a ver uma tela como a da figura a seguir:

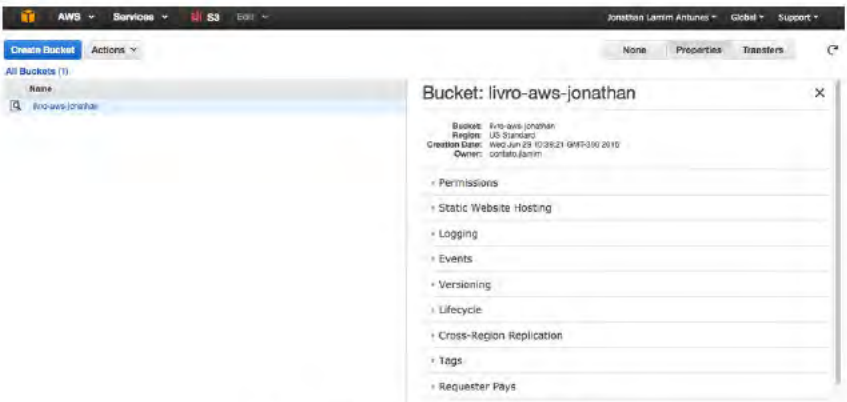


Figura 2.5: Tela após a criação de bucket

Do lado esquerdo, você tem a lista com os buckets criados, e logo acima dela o botão *Create Bucket* e o botão *Actions*, que possui acesso para operações de gerenciamento de um bucket. Do lado direito, você terá informações variadas sobre o bucket selecionado, tais como:

- **Permissions (permissões):** possibilita a configuração das permissões de acesso ao bucket, liberando assim o acesso para outros usuários. Ao criar o bucket, o seu usuário é o único que possui permissão para acessar os arquivos do bucket.
- **Static Website Hosting (hospedagem para site estático):** é basicamente um servidor de hospedagem para sites estáticos. Isso é possível pois cada bucket possui sua própria URL, e você pode armazenar em um bucket qualquer tipo de arquivo, então ao colocar arquivos HTML e imagens no bucket e ativar a opção, você terá um site estático funcionando.
- **Logging (log):** possibilita configurar o bucket para gerar log de atividades.

- **Events (eventos):** dispara notificações sempre que alguma operação é realizada no bucket.
- **Versioning (versionamento):** faz o controle de versão dos objetos do bucket, assim você pode manter um histórico dos objetos.
- **Lifecycle (ciclo de vida):** permite que você crie regras para manter o bucket atualizado e limpo, por exemplo, mover ou remover objetos automaticamente.
- **Cross-Region Replication (replicação para múltiplas regiões):** permite replicar os arquivos para outro bucket de maneira automática. Muito utilizado em conjunto com o *Versioning*.
- **Tags:** são usadas de modo a ajudar no entendimento dos relatórios de custo que são disponibilizados pela AWS.
- **Requester Pays (requisições de pagamento):** permite alterar o responsável pelos custos do bucket.
- **Transfer Acceleration (aceleração de transferência):** se ativado, permite que você acelere a transferência de dado tanto de entrada quanto de saída do S3.

Ao ativar um bucket, você pode assumir os custos dele ou passá-los para o solicitante. Ao passar o custo do bucket para o solicitante, você perde a possibilidade de acesso anônimo, pois vai ser através da autenticação que a Amazon saberá quem é o solicitante e cobrar dele os custos operacionais do bucket, como por exemplo, requisições e transferência de objetos.

2.5 MANIPULANDO OBJETOS NO BUCKET

Para manipular os objetos de um bucket, você precisa acessá-lo. Para isso, dê um duplo clique sobre o bucket criado, e então verá a tela a seguir:

Figura 2.6: Tela interna do bucket

Upload de objetos

Para fazer o upload, clique no botão *Upload* e, na tela que se abrirá, clique em *Add Files*. É permitido o upload múltiplo, o que significa que você pode enviar mais de um arquivo de cada vez. Selecione alguns arquivos e você verá uma alteração na tela.

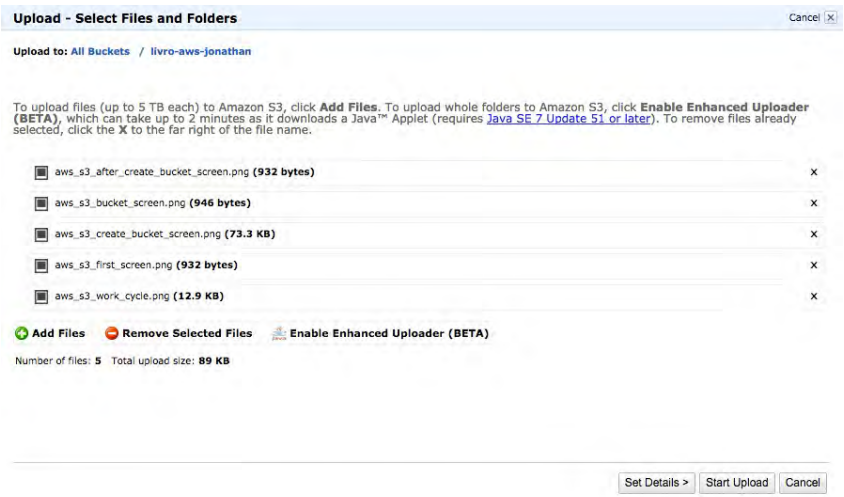


Figura 2.7: Tela para upload de arquivos

Após selecionar os arquivos, eles serão listados na tela que se abriu, e você ainda poderá removê-los antes de processar o upload

deles para o bucket. Basta clicar no *X* que fica à frente do arquivo desejado.

Se quiser remover múltiplos arquivos de uma única vez antes de processar o upload, então você deve manter a tecla *shift* pressionada e ir clicando sobre cada um dos arquivos da lista, e em seguida clicar no botão *Remove Selected Files*.

Logo a seguir dos botões, você poderá visualizar a quantidade de arquivos a serem enviados para o bucket e o tamanho total deles.

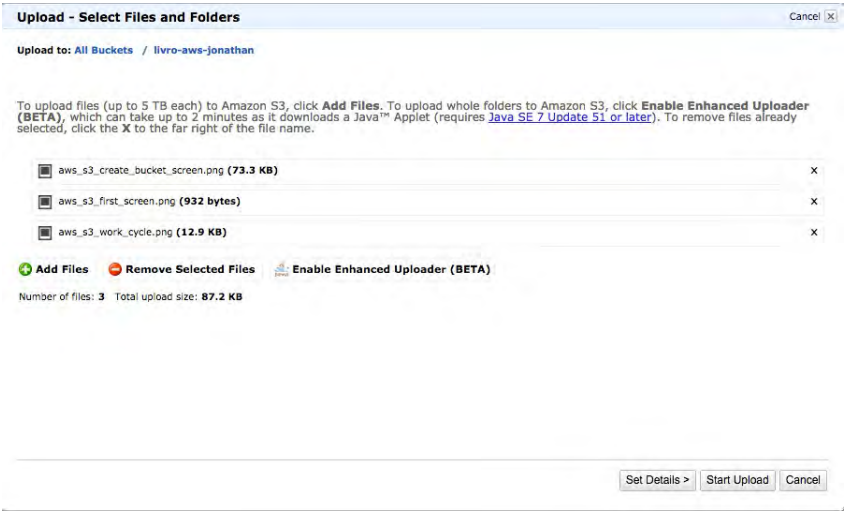


Figura 2.8: Tela para upload de arquivos com os arquivos adicionados

Com os arquivos definidos, basta clicar no botão *Start Upload*. A tela vai se fechar e você poderá ver o progresso do upload do lado direito da tela de gerenciamento do bucket, e do lado esquerdo a lista com os arquivos que já foram adicionados.

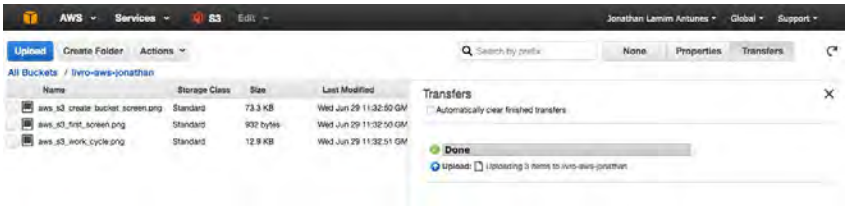


Figura 2.9: Tela interna do bucket com arquivos adicionados

Para gerenciar os arquivos do bucket, você pode clicar sobre eles com o botão direito do mouse e acionar as operações a partir do menu suspenso que se abrirá.

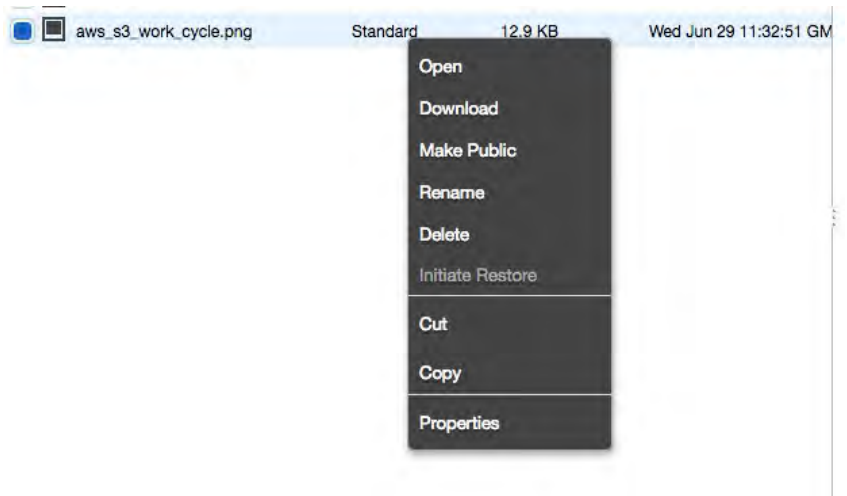


Figura 2.10: Menu dropdown para manipulação dos arquivos do bucket

A última opção desse menu é *Properties* e, se acionada, lhe permite visualizar informações mais detalhadas sobre o objeto, inclusive as permissões de acesso.

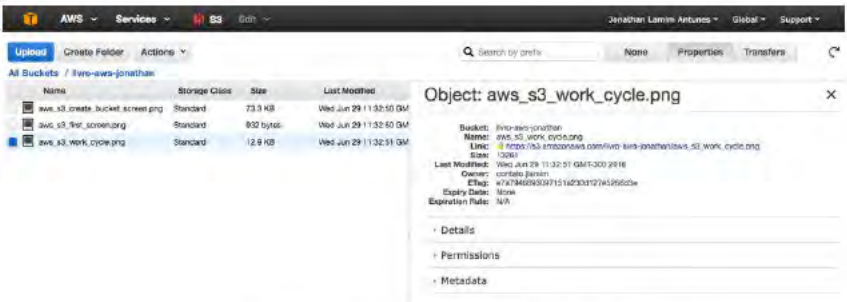


Figura 2.11: Propriedades de um objeto do bucket

- **Details:** permite a você definir o tipo de armazenamento e também criptografia do arquivo;
- **Permissions:** permite a você definir permissões de acesso;
- **Metadata:** permite a você adicionar outros metadados ao arquivo.

Visualizando o arquivo e obtendo o link

Para visualizar o arquivo, basta clicar sobre ele com o botão direito e escolher a opção *Open*. Ele será aberto em uma nova aba, e se você observar a URL, verá que ela possui um parâmetro de autenticação após o nome do arquivo.

Se tentar acessar essa URL sem passar o parâmetro, será gerado um erro. Esse erro ocorre porque o arquivo não está definido como público, logo é necessária autenticação para acessá-lo.

Figura 2.12: Erro de acesso por falta de parâmetro de autenticação

Para identificar se um arquivo é público ou não, basta verificar se existe um cadeado do lado esquerdo do link que aparece nas informações dele, do lado direito da tela, quando um arquivo é selecionado.

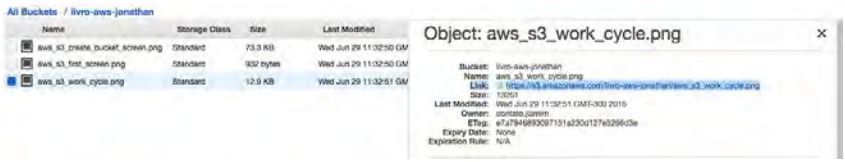


Figura 2.13: Link do arquivo

Se você quer tornar o seu arquivo público, removendo assim a necessidade de autenticação para visualização, basta clicar com o botão direito sobre ele e escolher a opção *Make Public* e, em seguida, confirmar. Após executar o procedimento, clique na opção *Properties*, que fica na parte superior direita da tela para visualizar novamente as informações do arquivo.

Onde está o link já não haverá mais o cadeado e sim um ícone azul, que identifica o arquivo como público. Dessa forma você pode compartilhar a URL dele à vontade, que não será exibido nenhum erro.

Como as requisições no S3 são do tipo `GET`, cada arquivo inserido em um bucket terá sua própria URL.

2.6 CRIANDO PASTAS DENTRO DE UM BUCKET

Para criar uma pasta dentro de um bucket, o processo é bastante simples. Clique no botão *Create Folder*. Em seguida, logo no início da lista de objetos do bucket, será adicionada uma linha com um campo que deve ser preenchido com o nome da pasta. Digite o nome e pressione a tecla *Enter* para que a pasta seja criada.

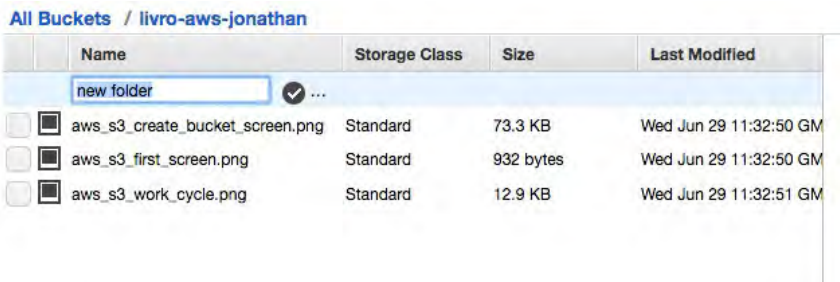


Figura 2.14: Criando uma pasta dentro do bucket

Para acessar a pasta, dê um duplo clique sobre ela na lista, e será aberta uma tela semelhante à que se abriu quando o bucket foi criado. A partir de agora, você pode inserir os arquivos diretamente dentro dessa pasta que acabou de criar, ou então na raiz do próprio bucket.

2.7 ENHANCED UPLOADER

O Enhanced Upload é uma ferramenta de upload que utiliza

Java para ser carregada. Ela permite que seja feito o upload de pastas e seus arquivos, diferente do upload tradicional, no qual você só pode fazer o upload de arquivos.

Para ativá-lo, basta clicar no botão *Enable Enhanced Uploader (BETA)* e aguardar que a tela seja atualizada. Pode ser que você precise fornecer as permissões para que a aplicação Java seja executada, então faça conforme as solicitações forem surgindo na tela.

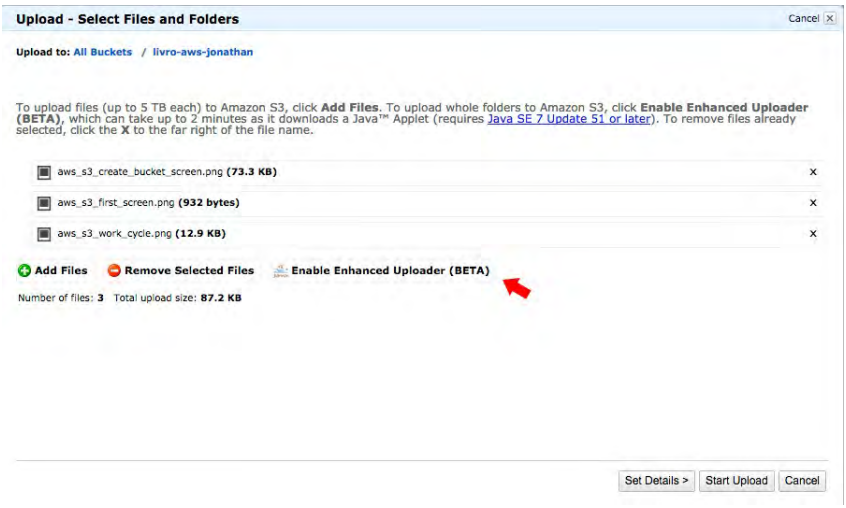


Figura 2.15: Enhanced Uploader (BETA)

Após ativar, não será mais exibida a janela padrão do browser para upload dos arquivos, mas sim uma janela personalizada, muito parecida com a padrão. O processo de seleção e continuidade do upload continua o mesmo, só aumenta a possibilidade de fazer o upload de uma ou mais pastas em vez de somente arquivos.

2.8 CONFIGURANDO O BUCKET PARA SALVAR OS LOGS

Os logs são uma maneira de saber o que acontece com o bucket, e após configurar, eles levam 24 horas para estarem disponíveis. Para fazer a configuração, você deverá navegar até a lista dos buckets, clicar sobre o bucket que deseja configurar o log e, em seguida, ir até a opção *Logging* localizada no painel do lado direito da tela.

Então, você deverá selecionar o checkbox **Enabled**, escolher o bucket alvo, que será o responsável por armazenar esses logs (é indicado criar um bucket somente para receber os logs), definir o prefixo e clicar em *Save*.

Os logs armazenados também serão objetos, e sua listagem será como a de um arquivo que foi enviado via upload.

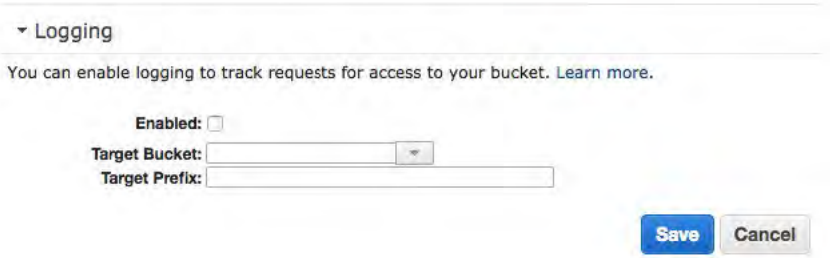


Figura 2.16: Criando uma pasta dentro do bucket

2.9 CRIANDO UM SITE ESTÁTICO COM S3

Criar um site estático com o S3 é bastante rápido e necessita de poucos passos. Para começar, crie um bucket para receber o site estático, em seguida crie 2 arquivos HTML, um chamado `index.html` e o outro `error.html` . Para ter um site estático dentro do S3, você vai precisar desses 2 arquivos, pelo menos.

Use os códigos a seguir, em seus respectivos arquivos, para poder concluir a criação das duas páginas.

- **index.html**

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>Site Estático com Amazon AWS - S3</title>
  </head>
<body>
  <h1>Site Estático com Amazon AWS - S3</h1>
  <p>Todo site estático precisa no S3 precisa que seus arquivos se
jam públicos.</p>
</body>
</html>
```

- **error.html**

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>Site Estático com Amazon AWS - S3</title>
  </head>
<body>
  <h1>Site Estático com Amazon AWS - S3</h1>
  <p>Desculpe, mas a página que está tentando acessar não existe.</
p>
</body>
</html>
```

Com esses arquivos e o bucket criados, é hora de fazer o upload deles para dentro do bucket. Repita os procedimentos vistos anteriormente para proceder com o upload.

Após concluir o upload, você precisará tornar os dois arquivos públicos. Para isso, selecione os arquivos e clique no botão *Actions*. No menu dropdown que se abrirá, escolha a opção *Make Public* e confirme. A partir desse momento, os arquivos já são públicos e você pode fazer a configuração final para ativar o site estático.

Volte para a lista de buckets e selecione o bucket criado para armazenar os arquivos do site estático. Do lado direito da tela, vá até a opção *Static Website Hosting* e escolha a opção *Enable website hosting*. Nela informe o nome do arquivo `index.html` para o campo *Index Document* e o arquivo `error.html` para o campo *Error Document*.

Você também pode configurar regras para redirecionamento na opção *Edit Redirection Rules*, de modo a fazer redirecionamentos das requisições. Entretanto, ficaremos com a abordagem mais simples, sem essas regras.

Feitas essas configurações clique em *Save*.

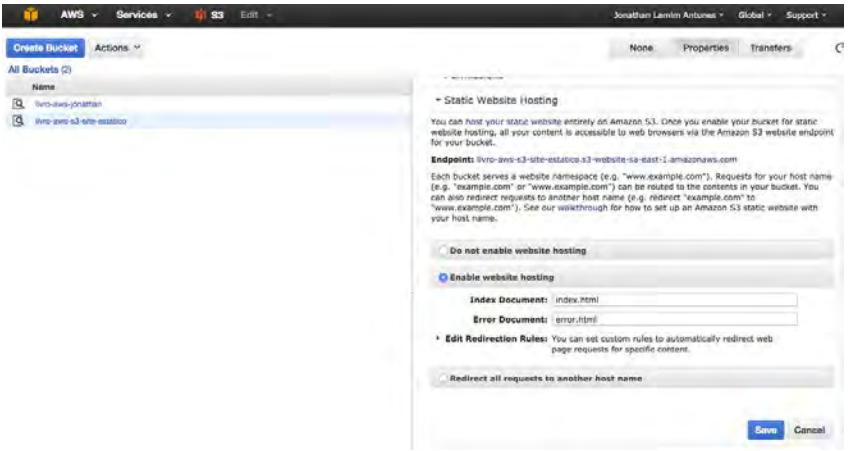


Figura 2.17: Configuração de site estático no S3

Para acessar o site estático, você deverá utilizar a URL informada junto das informações de *Static Website Hosting*, onde está definida como **Endpoint**. Esse será o link de acesso ao site estático.

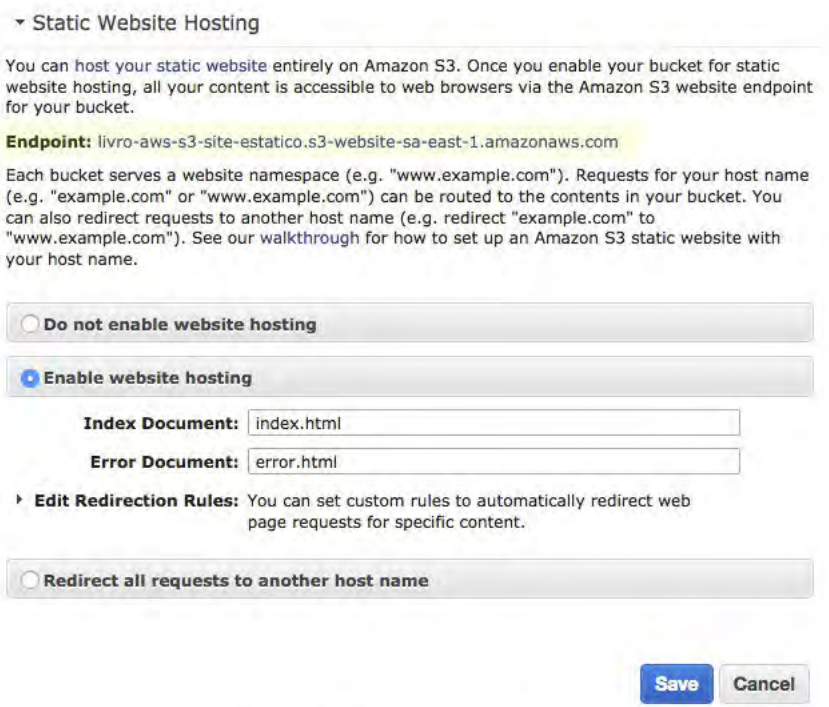


Figura 2.18: URL para acesso ao site estático

2.10 CONSIDERAÇÕES FINAIS

Neste capítulo, você foi apresentado ao Amazon S3, aprendeu a utilizar a interface gráfica para criar e gerenciar os buckets, e ainda criou um site estático usando os recursos do S3. No próximo capítulo, você vai ver como utilizar o S3 através das APIs que a Amazon disponibiliza.

Links úteis

- **Guia e iniciação:** https://docs.aws.amazon.com/AmazonS3/latest/gsg/GetStartedWithS3.html
- **Documentação:**

https://aws.amazon.com/documentation/s3/

Capítulo 3

AMAZON S3 - GERENCIANDO OPERAÇÕES COM AWS SDK

3.1 INTRODUÇÃO

O AWS SDK é uma ferramenta de integração usada no desenvolvimento de sistemas que fazem uso de recursos e serviços do AWS. Neste capítulo, você verá exemplos de aplicação do AWS SDK para o AWS S3.

Para otimizar o processo de aprendizado, vamos utilizar os exemplos de código fornecidos pela própria Amazon, que estão disponíveis no GitHub (https://github.com/awslabs). Será necessário que você possua um cliente Git instalado na sua máquina para que possa fazer a clonagem dos códigos de exemplo.

3.2 CONFIGURANDO UM USUÁRIO IAM PARA ACESSO AO S3

Todo acesso feito via API precisa de autenticação. Logo, é necessário criar um usuário e configurar as permissões de acesso e segurança.

Crie um usuário para que possa utilizar nos exemplos que serão vistos mais adiante, ainda neste capítulo.

Para criar o usuário e fazer as devidas configurações, acesse o console de gerenciamento e vá em **Services > IAM**.

Figura 3.1: Tela principal do AWS IAM

No menu lateral, selecione a opção *Users* e, em seguida, clique no botão *Create New User*.

Figura 3.2: Tela de usuários do AWS IAM

Informe o nome dos usuários que deseja criar e, depois, clique

em *Create*. Mantenha a opção **Generate an access key for each user** selecionada, para que uma chave de acesso seja criada automaticamente para cada um deles.

Figura 3.3: Tela de cadastro de usuários no AWS IAM

Após criar os usuários, você será direcionado para uma tela informativa onde estarão os detalhes de cada usuário que foi criado. Esses detalhes são: *Access Key ID* e *Secret Access Key*.

Faça o download das credenciais através do botão *Download Credentials*, que fica na parte inferior direita da tela, pois não haverá essa possibilidade depois, e essas credenciais serão utilizadas para acesso ao S3 através da API e dos SDKs.

Para sair dessa tela, basta clicar em *Close*, também na parte inferior direita da tela.

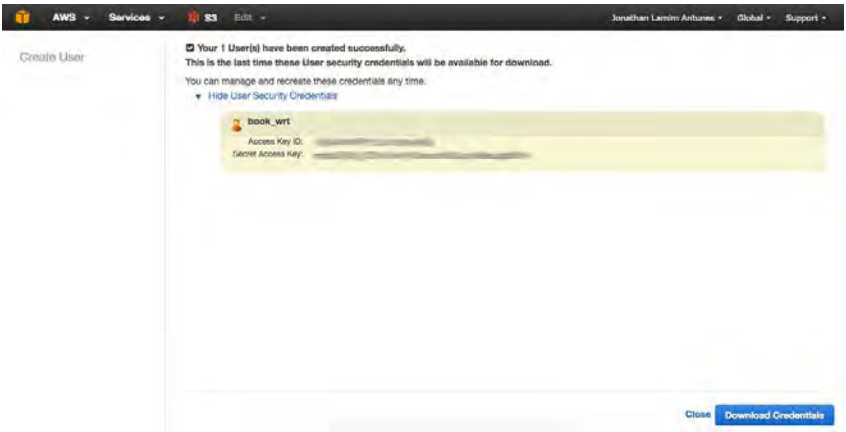


Figura 3.4: Tela de usuários após o cadastro no AWS IAM

Dê um duplo clique sobre o usuário criado para que possa prosseguir com as configurações. Em seguida, clique na aba *Permissions* e então em *Attach Policy*, para que possa definir as permissões de acesso para esse usuário.

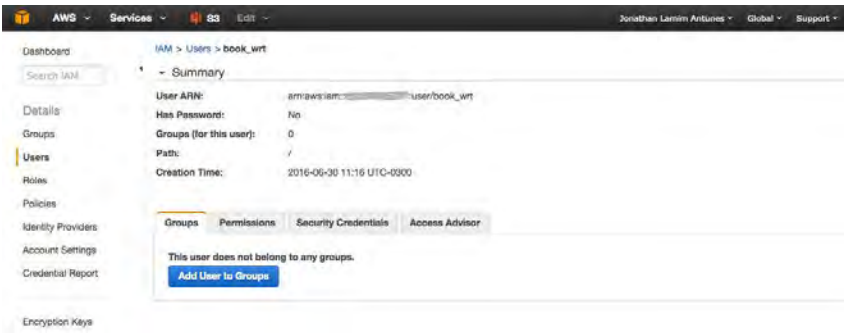


Figura 3.5: Tela de detalhes do usuário no AWS IAM

Existem muitos níveis de permissão já configurados e disponíveis para uso, mas como nesse momento vamos trabalhar apenas com o AWS S3, você vai utilizar o **AmazonS3FullAccess**, que permitirá acesso total aos recursos do S3.

Use a caixa de busca para filtrar os registros e facilitar a

localização. Busque pelo termo *S3*, conforme a seguir:

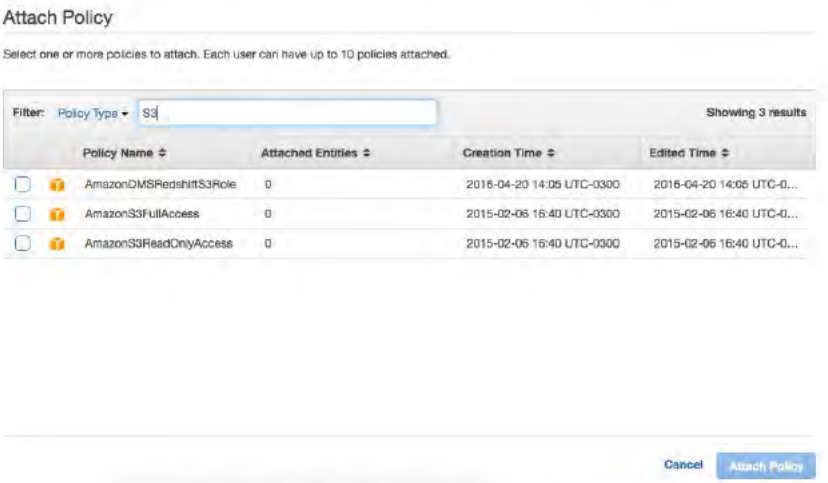


Figura 3.6: Tela de permissões do usuário no AWS IAM

Selecione o registro e clique em *Attach Policy*. Pronto, o usuário está criado e com permissão total para o S3, e agora já é possível utilizar a API e os SDKs.

Como o acesso aos recursos do S3 via API são autenticados, é necessário que você configure as credenciais do usuário no seu ambiente de desenvolvimento. Dê uma pausa na leitura do capítulo e vá até o **Apêndice 1** deste livro. Faça o processo de instalação do AWS CLI e a configuração das credenciais de acesso.

3.3 AWS SDK PARA NODEJS

Para fazer uso do AWS SDK para NodeJS, você precisará do NodeJS e do NPM instalados em sua máquina.

- NodeJS (http://nodejs.org)
- NPM (http://www.npmjs.com/package/npm)

Clonando o código de exemplo

Para clonar o código de exemplo, abra o terminal (ou o prompt de comando no Windows) e execute os comandos a seguir:

```
git clone https://github.com/awslabs/aws-nodejs-sample.git

cd aws-nodejs-sample

npm install
```

O comando `npm install` fará com que todas as dependências informadas no arquivo `package.json` sejam instaladas. Veja o `package.json` do exemplo:

```
{
    "dependencies": {
        "aws-sdk": ">= 2.0.9",
        "node-uuid": ">= 1.4.1"
    }
}
```

Após instalar as dependências, é hora de testar o código de exemplo. Para isso, execute o comando `node sample.js` no terminal.

Ao executar o comando citado, será criado um novo bucket no S3, e será criado um objeto dinamicamente. Esse objeto será um arquivo TXT com o famoso *Hello World*.

Veja adiante o código que foi utilizado para executar esse procedimento:

```
// Carrega as dependências 'aws-sdk' e 'node-uuid'
var AWS = require('aws-sdk');
var uuid = require('node-uuid');

// Cria um cliente S3
var s3 = new AWS.S3();

// Define o nome do bucket e o nome do arquivo a ser criado
var bucketName = 'node-sdk-sample-' + uuid.v4();
var keyName = 'hello_world.txt';
```

```
//Executa a criação do bucket
//Como parâmetro recebe o nome do bucket e uma função de callback
//que vai processar a criação do objeto dentro do bucket
s3.createBucket({Bucket: bucketName}, function() {
  //define as informações do objeto a ser criado
  var params = {Bucket: bucketName, Key: keyName, Body: 'Hello Wor
ld!'};
  //processa a criação do objeto no bucket
  s3.putObject(params, function(err, data) {
    if (err)
      console.log(err)
    else
      console.log("Successfully uploaded data to " + bucketName +
"/" + keyName);
  });
});
```

Veja a seguir um print com o resultado dessa execução na tela:

```
MacBook-Pro-de-Jonathan:aws-nodejs-sample jlamim$ node sample.js
Successfully uploaded data to node-sdk-sample-d2219de7-bf91-47f8-bb13-a3b9d02fdf47/hello_world.txt
```

Figura 3.7: Execução do exemplo do AWS SDK para NodeJS

3.4 AWS SDK PARA PYTHON

Para fazer uso do AWS SDK para Python, você precisará do Python, pip e Boto instalados em sua máquina.

- Python (https://www.python.org)
- pip (http://pip.readthedocs.org/en/latest)
- Boto (https://github.com/boto/boto)

Clonando o código de exemplo

Para clonar o código de exemplo, abra o terminal (ou o prompt de comando no Windows) e execute os comandos:

```
git clone https://github.com/awslabs/aws-python-sample.git
```

```
cd aws-python-sample
```

Para testar esse exemplo, execute o comando `python`

`s3_sample.py` . Ao executá-lo, será criado um novo bucket no S3 e um objeto dinamicamente. Também será exibida na tela a URL de acesso desse objeto. Em seguida, será solicitado para pressionar a tecla *Enter*, e então o objeto e o bucket serão removidos.

Veja adiante o código utilizado para esse procedimento:

```
# Importa as bibliotecas e SDKs necessários
import boto3
import uuid

# Instancia um novo cliente para o S3
s3client = boto3.client('s3')

# Define um nome para o bucket
bucket_name = 'python-sdk-sample-{}'.format(uuid.uuid4())
print('Creating new bucket with name: {}'.format(bucket_name))

# Cria o bucket
s3client.create_bucket(Bucket=bucket_name)

# Imprime na tela o nome dos buckets que estão no S3
list_buckets_resp = s3client.list_buckets()
for bucket in list_buckets_resp['Buckets']:
    if bucket['Name'] == bucket_name:
        print('(Just created) --> {} - there since {}'.format(
            bucket['Name'], bucket['CreationDate']))

# Define o nome do objeto
object_key = 'python_sample_key.txt'

# Imprime na tela o nome do objeto
print('Uploading some data to {} with key: {}'.format(
    bucket_name, object_key))

# Cria o objeto
s3client.put_object(Bucket=bucket_name, Key=object_key, Body=b'Hel
lo World!')

# Gera a URL de acesso ao objeto e exibe na tela
url = s3client.generate_presigned_url(
    'get_object', {'Bucket': bucket_name, 'Key': object_key})
print('\nTry this URL in your browser to download the object:')
print(url)
```

```
try:
    input = raw_input
except NameError:
    pass
input("\nPress enter to continue...")

# Agora executa os procedimentos usando recursos da API
print('\nNow using Resource API')

# Instancia um novo serviço
s3resource = boto3.resource('s3')

# Agora o bucket
bucket = s3resource.Bucket(bucket_name)

# E o objeto
obj = bucket.Object(object_key)

# Exibe na tela o nome do bucket, o objeto, o tamanho do objeto,
# o conteúdo do objeto e a data da última modificação
print('Bucket name: {}'.format(bucket.name))
print('Object key: {}'.format(obj.key))
print('Object content length: {}'.format(obj.content_length))
print('Object body: {}'.format(obj.get()['Body'].read()))
print('Object last modified: {}'.format(obj.last_modified))

# Remove os objetos do bucket
print('\nDeleting all objects in bucket {}.'.format(bucket_name))
delete_responses = bucket.objects.delete()
for delete_response in delete_responses:
    for deleted in delete_response['Deleted']:
        print('\t Deleted: {}'.format(deleted['Key']))

# Remove o bucket
print('\nDeleting the bucket.')
bucket.delete()
```

Veja a seguir um print com o resultado dessa execução na tela:

```
MacBook-Pro-de-Jonathan:aws-python-sample jlamim$ python s3_sample.py
Creating new bucket with name: python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b
(Just created) --> python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b - there since 2016-06-30 20:04:18+00:00
Uploading some data to python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b with key: python_sample_key.txt

Try this URL in your browser to download the object:
https://python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b.s3.amazonaws.com/python_sample_key.txt?AWSAccessKey
Id=AKIAIWNRYYUJLM6346MQ&Expires=1467320659&Signature=MsqNldq5UQcA4kG80WbSjbQOM%kc%3D

Press enter to continue...

Now using Resource API
Bucket name: python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b
Object key: python_sample_key.txt
Object content length: 12
Object body: Hello World!
Object last modified: 2016-06-30 20:04:20+00:00

Deleting all objects in bucket python-sdk-sample-4c9ccb1d-0072-41a1-bc42-34106bdbbc4b.
        Deleted: python_sample_key.txt

Deleting the bucket.
```

Figura 3.8: Execução do exemplo do AWS SDK para Python

3.5 AWS SDK PARA PHP

Para fazer uso do AWS SDK para PHP, você precisará do PHP 5.3 ou superior e do Composer instalados em sua máquina.

- PHP (http://php.net)
- Composer (http://getcomposer.org)

Clonando o código de exemplo

Para clonar o código de exemplo, abra o terminal (ou o prompt de comando no Windows) e execute os comandos a seguir:

```
git clone https://github.com/awslabs/aws-php-sample.git
```

```
cd aws-php-sample
```

```
composer install
```

O comando `composer install` fará com que todas as dependências necessárias para o exemplo sejam instaladas. Para testar o exemplo, execute o comando `php sample.php` .

Ao executar esse comando, será criado um novo bucket no S3 e 2 objetos dinamicamente, além de exibir na tela o conteúdo de cada um dos objetos. E por fim, serão removidos os objetos e o bucket. Veja adiante o código usado para esse procedimento:

```php
// Faz o include do SD usando o autoload do Composer
require 'vendor/autoload.php';

//Define os parâmetros de configuração para o cliente que serão in
stanciados
$s3 = new Aws\S3\S3Client([
    'version' => '2006-03-01',
    'region'  => 'us-west-2'
]);

// Define o nome do bucket
$bucket = uniqid("php-sdk-sample-", true);
echo "Creating bucket named {$bucket}\n";

// Cria o bucket
$s3->createBucket(['Bucket' => $bucket]);
$s3->waitUntil('BucketExists', ['Bucket' => $bucket]);

// Define o nome do objeto
$key = 'hello_world.txt';
echo "Creating a new object with key {$key}\n";

// Cria o objeto no bucket
$s3->putObject([
    'Bucket' => $bucket,
    'Key'    => $key,
    'Body'   => "Hello World!"
]);

// Faz o download do objeto para exibir o conteúdo dele na tela
echo "Downloading that same object:\n";
$result = $s3->getObject([
    'Bucket' => $bucket,
    'Key'    => $key
]);

// Exibe o conteúdo do objeto na tela
echo "\n---BEGIN---\n";
echo $result['Body'];
echo "\n----END----\n\n";

// Registra o wrapper para poder utilizar uma URL para criar um ob
jeto e obter
// seus dados
$s3->registerStreamWrapper();

// Define o nome do novo objeto
$key2 = 'hello_again_world.txt';
```

```
echo "Creating a second object with key {$key2} using stream wrapp
ers\n";

// Cria um novo objeto, agora usando uma URL do S3 para executar o
 procedimento
file_put_contents("s3://{$bucket}/{$key2}", 'Hello Again!');

// Faz o download do objeto para exibir o conteúdo dele na tela
echo "Downloading that same object:\n";
$data = file_get_contents("s3://{$bucket}/{$key2}");

// Exibe o conteúdo do objeto na tela
echo "\n---BEGIN---\n";
echo $data;
echo "\n----END----\n\n";

// Remove os objetos contidos no bucket
echo "Deleting all objects in bucket {$bucket}\n";
$batch = Aws\S3\BatchDelete::fromListObjects($s3, ['Bucket' => $bu
cket]);
$batch->delete();

// Remove o bucket
echo "Deleting bucket {$bucket}\n";
$s3->deleteBucket(['Bucket' => $bucket]);
```

Veja a seguir um print com o resultado dessa execução na tela:

```
MacBook-Pro-de-Jonathan:aws-php-sample jlamim$ php sample.php
Creating bucket named php-sdk-sample-5775801b0e8261.94586898
Creating a new object with key hello_world.txt
Downloading that same object:

---BEGIN---
Hello World!
----END----

Creating a second object with key hello_again_world.txt using stream wrappers
Downloading that same object:

---BEGIN---
Hello Again!
----END----

Deleting all objects in bucket php-sdk-sample-5775801b0e8261.94586898
Deleting bucket php-sdk-sample-5775801b0e8261.94586898
```

Figura 3.9: Execução do exemplo do AWS SDK para PHP

Caso encontre um erro semelhante ao da figura a seguir ao executar esse código, adicione `date_default_timezone_set('America/Sao_Paulo');` logo após o `require`, para que seja definido o timezone corretamente.

```
Fatal error: Uncaught exception 'Exception' with message 'DateTime::__construct(): It is not safe to rely on the s
ystem's timezone settings. You are *required* to use the date.timezone setting or the date_default_timezone_set()
function. In case you used any of those methods and you are still getting this warning, you most likely misspelled
 the timezone identifier. We selected the timezone 'UTC' for now, but please set date.timezone to select your time
zone.' in /Volumes/WebRepo/exemplos-livro-aws/aws-php-sample/vendor/aws/aws-sdk-php/src/Api/Parser/XmlParser.php:1
32
Stack trace:
#0 /Volumes/WebRepo/exemplos-livro-aws/aws-php-sample/vendor/aws/aws-sdk-php/src/Api/Parser/XmlParser.php(132): Da
teTime->__construct('2016-06-30T21:0...')
#1 /Volumes/WebRepo/exemplos-livro-aws/aws-php-sample/vendor/aws/aws-sdk-php/src/Api/Parser/XmlParser.php(36): Aws
\Api\Parser\XmlParser->parse_timestamp(Object(Aws\Api\TimestampShape), Object(SimpleXMLElement))
#2 /Volumes/WebRepo/exemplos-livro-aws/aws-php-sample/vendor/aws/aws-sdk-php/src/Api in /Volumes/WebRepo/exemplos-
livro-aws/aws-php-sample/vendor/aws/aws-sdk-php/src/Api/Parser/XmlParser.php on line 132
```

Figura 3.10: Possível erro causado pela configuração do timezone

3.6 CONSIDERAÇÕES FINAIS

Neste capítulo, você viu exemplos de aplicação da API do S3 em linguagens diferentes. No próximo capítulo, você vai aprender sobre o AWS EC2, que vai permitir criar instâncias de servidores virtuais com sistema operacional Linux e Windows.

Links úteis

- **AWS S3:** http://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html
- **AWS IAM (Identity and Access Management):** http://aws.amazon.com/iam
- **SDK para NodeJS:** https://aws.amazon.com/pt/sdk-for-node-js/
- **SDK para Python:** https://aws.amazon.com/pt/sdk-for-python/

- **SDK para PHP:** https://aws.amazon.com/pt/sdk-for-php/

Capítulo 4

AMAZON EC2 (ELASTIC COMPUTE CLOUD)

4.1 INTRODUÇÃO

O EC2 (ou Elastic Compute Cloud) é o serviço que permite você criar, na nuvem, instâncias de servidores virtuais usando diferentes configurações — isso inclui armazenamento, memória e processadores — e sistemas operacionais (Windows ou Unix). É em uma instância do EC2 que você colocará sua aplicação para funcionar.

Você pode criar no EC2 vários tipos de instâncias, sejam elas para uso geral, com baixa configuração ou mesmo com alta configuração, possuindo memória e processamento de alta capacidade. Tem também microinstâncias (que são as integrantes do período de uso gratuito que a Amazon oferece para novos usuários) e que possuem configurações limitadas.

Os benefícios do EC2 são:

- Recursos escaláveis;
- Controle total;
- Integração com outros serviços da Amazon;
- Confiabilidade;
- Segurança;
- Custo acessível;

- Facilidade na utilização.

Os conhecimentos sobre EC2 permitirão a você gerenciar a infraestrutura de servidor virtual onde sua aplicação estará. Então aproveite o conteúdo deste capítulo para conhecer e compreender o ecossistema do EC2.

4.2 TIPOS DE INSTÂNCIA

Como foi citado na introdução desse capítulo, o EC2 possui vários tipos de instâncias, que são:

- Uso geral;
- Otimizadas para computação;
- Otimizadas para memória;
- GPU;
- Otimizadas para armazenamento.

Veja a seguir mais detalhes sobre cada um desses tipos.

Uso geral

As instâncias de uso geral possuem capacidade limitada em relação aos demais tipos de instância e estão dividas em T2, M3 e M4.

T2

As instâncias de uso geral T2 são indicadas para projetos que não demandam de potência total da CPU de forma constante, como por exemplo: ambientes de desenvolvimento, servidores de compilação, repositórios de código, sites e aplicações web de baixo tráfego, microsserviços, experimentos iniciais de produtos, pequenos bancos de dados, entre outros.

Os recursos disponíveis dessa instância de uso geral T2 são:

- Processadores Intel Xeon de alta frequência com turbo de até 3,3 GHz;
- CPU com capacidade de intermitência, regida por créditos de CPU e desempenho de linha de base constante;
- Equilíbrio entre recursos de computação, memória e rede.

Veja na figura a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de T2:

Modelo	vCPU	Créditos de CPU/hora	Mem (GiB)	Armazenamento
t2.nano	1	3	0,5	Somente EBS
t2.micro	1	6	1	Somente EBS
t2.small	1	12	2	Somente EBS
t2.medium	2	24	4	Somente EBS
t2.large	2	36	8	Somente EBS

Figura 4.1: Informações sobre instâncias T2

M3

É uma linha de instâncias de uso geral a nível intermediário, com boa configuração, e indicada para projetos como: bancos de dados de pequeno e médio porte, tarefas de processamento de dados que exigem memória adicional, grupos de armazenamento em cache e servidores de back-end para SAP, Microsoft SharePoint, computação em cluster, aplicações empresariais, entre outros.

Os recursos disponíveis dessa instância de uso geral M3 são:

- Processadores Intel Xeon E5-2670 v2 (Ivy Bridge) de alta frequência ou processador Intel Xeon E5-2670 (Sandy Bridge) operando a 2,6 GHz;
- Armazenamento em instância baseado em SSD para alto desempenho de E/S;
- Equilíbrio entre recursos de computação, memória e rede.

Veja na figura seguinte uma tabela com informações sobre esses recursos, distribuídos nas variações de M3:

Modelo	vCPU	Mem (GiB)	Armazenamento em SSD (GB)
m3.medium	1	3,75	1 x 4
m3.large	2	7,5	1 x 32
m3.xlarge	4	15	2 x 40
m3.2xlarge	8	30	2 x 80

Figura 4.2: Informações sobre instâncias M3

M4

Considerada como a última geração das instâncias de uso geral, ela possui recursos equilibrados, e é indicada para projetos na mesma linha dos indicados para M3.

Os recursos disponíveis dessa instância de uso geral M4 são:

- Processadores Intel Xeon® E5-2676 v3 2,4 GHz (Haswell);
- Otimizado para EBS por padrão;

- Suporte a redes aperfeiçoadas;
- Equilíbrio entre recursos de computação, memória e rede.

Veja na figura a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de M4:

Modelo	vCPU	Mem (GiB)	Armazenamento em SSD (GB)	Largura de banda dedicada do EBS (Mbps)
m4.large	2	8	Somente EBS	450
m4.xlarge	4	16	Somente EBS	750
m4.2xlarge	8	32	Somente EBS	1.000
m4.4xlarge	16	64	Somente EBS	2.000
m4.10xlarge	40	160	Somente EBS	4.000

Figura 4.3: Informações sobre instâncias M3

Otimizadas para computação

São instâncias que possuem um maior desempenho e melhor custo benefício a nível preço/desempenho dentro das instâncias do EC2.

As instâncias otimizadas para computação têm seu uso indicado para: frotas de front-end de alto desempenho, servidores da web, processamento em lotes, dados analíticos distribuídos, aplicativos científicos e de engenharia de alto desempenho, veiculação de anúncios, jogos MMO, codificação de vídeo, entre outros.

Esse tipo de instância está divido em 2 grupos, o C3 e o C4, ambos indicados para as mesmas finalidades, mas com recursos diferentes.

Recursos do C3

- Processadores Intel Xeon E5-2680 v2 (Ivy Bridge) de alta frequência;
- Suporte aprimorado a redes;
- Suporte a clusters;
- Armazenamento de instâncias com respaldo de SSD.

Veja na figura a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de C3:

Modelo	vCPU	Mem (GiB)	Armazenamento em SSD (GB)
c3.large	2	3,75	2 x 16
c3.xlarge	4	7,5	2 x 40
c3.2xlarge	8	15	2 x 80
c3.4xlarge	16	30	2 x 160
c3.8xlarge	32	60	2 x 320

Figura 4.4: Informações sobre instâncias C3

Recursos do C4

- Processadores Intel Xeon E5-2666 v3 (Haswell) de alta frequência otimizados especificamente para o EC2;
- Otimizado para EBS por padrão;
- Capacidade de controlar configurações de estado C e P no tipo de instância c4.8xlarge;

- Suporte a redes aperfeiçoadas e clustering.

Veja na figura adiante uma tabela com informações sobre esses recursos, distribuídos nas variações de C4:

Modelo	vCPU	Mem (GiB)	Armazenamento	EBS dedicado Largura de banda (Mbps)
c4.large	2	3,75	Somente EBS	500
c4.xlarge	4	7,5	Somente EBS	750
c4.2xlarge	8	15	Somente EBS	1.000
c4.4xlarge	16	30	Somente EBS	2.000
c4.8xlarge	36	60	Somente EBS	4.000

Figura 4.5: Informações sobre instâncias C4

Otimizadas para memória

As instâncias otimizadas para memória possuem configurações robustas tanto de memória quanto de largura de banda, o que permite alto desempenho. Ela está dividida em X1 e R3.

X1

É considerada a instância de menor preço por GiB de RAM dentro no EC2, e possui os seguintes recursos:

- Processadores Intel Xeon E7-8880 v3 (Haswell) de alta frequência;
- Custo mais baixo por GiB de RAM;
- 1.952 GiB de memória de instância baseada em DDR4;
- Armazenamento SSD e com otimização para o EBS como padrão;

- Capacidade de controlar a configuração de estado C e estado P do processador.

Veja a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de X1:

Modelo	vCPU	Mem (GiB)	Armazenamento em SSD (GB)	Largura de banda dedicada do EBS (Mbps)
x1.32xlarge	128	1.952	2 x 1.920	10.000

Figura 4.6: Informações sobre instâncias X1

Seu uso é indicado para: execução de banco de dados na memória, mecanismos de processamento de big data, aplicações de computação de alto desempenho (HPC), entre outros.

> As instâncias X1 são certificadas pela SAP para executar o Business Warehouse on HANA (BW), o Data Mart Solutions on HANA, o Business Suite on HANA (SoH) e o Business Suite S/4HANA de próxima geração em um ambiente de produção na Nuvem AWS.

R3

A diferença entre a R3 e a X1 são os recursos, e com essa diferença de recursos as indicações de uso também variam.

Os recursos da R3 são:

- Processadores Intel Xeon E5-2670 v2 (Ivy Bridge) de

alta frequência;
- Armazenamento em SSD;
- Suporte aprimorado a redes.

Veja uma tabela com informações sobre esses recursos, distribuídos nas variações de R3:

Modelo	vCPU	Mem (GiB)	Armazenamento em SSD (GB)
r3.large	2	15.25	1 x 32
r3.xlarge	4	30,5	1 x 80
r3.2xlarge	8	61	1 x 160
r3.4xlarge	16	122	1 x 320
r3.8xlarge	32	244	2 x 320

Figura 4.7: Informações sobre instâncias R3

Seu uso é indicado para: bancos de dados de alto desempenho, caches de memória distribuídos, recursos analíticos de memória, montagem e análise de genoma, Microsoft SharePoint, aplicações empresariais, entre outros

GPU

As instâncias GPU oferecem recursos para o trabalho com gráficos e aplicações de GPU de um modo geral. Possui recursos bastante robustos, como:

- Processadores Intel Xeon E5-2670 (Sandy Bridge) de alta frequência;
- GPUs NVIDIA de alto desempenho, cada uma delas com 1.536 núcleos CUDA e 4 GB de memória de vídeo;

- Cada GPU apresenta um codificador de vídeo de hardware integrado projetado para suportar até oito streams de vídeo HD em tempo real (720p a 30 fps) ou até quatro streams de vídeo Full HD em tempo real (1080p a 30 fps);
- Suporte para captura e codificação de quadros de baixa latência, tanto do sistema operacional quanto de alvos específicos de renderização, possibilitando experiências de streaming interativo de alta qualidade.

Veja na figura a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de G2:

Modelo	GPUs	vCPU	Mem (GiB)	Armazenamento em SSD (GB)
g2.2xlarge	1	8	15	1 x 60
g2.8xlarge	4	32	60	2 x 120

Figura 4.8: Informações sobre instâncias G2

Seu uso é indicado para: streaming de aplicações 3D, aprendizagem de máquina, codificação de vídeo, cargas de trabalho de gráficos ou computação de GPU no lado do servidor, entre outros.

Otimizadas para armazenamento

Instâncias otimizadas para armazenamento são utilizadas em projetos nos quais o armazenamento de arquivos é alto, e necessita não só de espaço, mas também de velocidade de E/S para a transferência desses arquivos entre cliente e servidor (instância). Elas estão divididas em 2 grupos, I2 e D2.

I2

São instâncias de E/S elevada, em que o volume de dados trafegado é alto. Seus recursos são:

- Processadores Intel Xeon E5-2670 v2 (Ivy Bridge) de alta frequência;
- Armazenamento em SSD;
- Suporte para TRIM;
- Suporte aprimorado a redes;
- Alto desempenho de E/S aleatória.

Veja a seguir uma tabela com informações sobre esses recursos, distribuídos nas variações de I2:

Modelo	vCPU	Mem (GiB)	Armazenamento (GB)
i2.xlarge	4	30,5	1 x 800 SSD
i2.2xlarge	8	61	2 x 800 SSD
i2.4xlarge	16	122	4 x 800 SSD
i2.8xlarge	32	244	8 x 800 SSD

Figura 4.9: Informações sobre instâncias I2

Seu uso é indicado para: bancos de dados NoSQL, bancos de dados transacionais escaláveis, armazenamento de dados, sistemas de arquivos, entre outros.

D2

São as instâncias de armazenamento denso, ou seja, para armazenamento de uma grande massa de arquivos e dados. Oferece até 48TB de armazenamento local em HDD e alta taxa de

transferência.

Seus recursos são:

- Processadores Intel Xeon E5-2676v3 (Haswell) de alta frequência
- Armazenamento HDD
- Alto desempenho consistente no momento do lançamento
- Alta taxa de transferência de disco
- Compatíveis com as redes aperfeiçoadas do Amazon EC2

Veja a tabela com informações sobre esses recursos, distribuídos nas variações de D2:

Modelo	vCPU	Mem (GiB)	Armazenamento (GB)
d2.xlarge	4	30,5	3 x 2.000 HDD
d2.2xlarge	8	61	6 x 2.000 HDD
d2.4xlarge	16	122	12 x 2.000 HDD
d2.8xlarge	36	244	24 x 2.000 HDD

Figura 4.10: Informações sobre instâncias D2

Seu uso é indicado para: armazenamento de dados para processamento paralelo massivo (MPP), computação distribuída de MapReduce e Hadoop, sistemas de arquivo distribuídos, sistemas de arquivos de rede, aplicações de processamento de logs ou dados, entre outros.

4.3 INSTÂNCIAS RESERVADAS

As instâncias reservadas do EC2 são instâncias nas quais você tem a possibilidade de reservar a capacidade computacional do EC2 por um período que varia de 1 a 3 anos, e em troca a Amazon oferece um desconto no valor a ser pago. Esse desconto pode chegar a até 75% do valor/hora cobrado.

Elas são utilizadas para reduzir os custos e, em contrapartida, oferecem a confiabilidade de você poder criar instâncias reservadas sempre que necessário.

4.4 INSTÂNCIAS DEDICADAS

As instâncias dedicadas do EC2 são aquelas executadas em VPC, com hardware dedicado. Elas ficam isoladas no nível de hardware das outras instâncias. A aquisição de instâncias dedicadas pode ser feita no mesmo formato das instâncias reservadas, permitindo que você obtenha bons descontos ao adquirir uma instância por um longo prazo.

4.5 INSTÂNCIAS SPOT

As instâncias Spot são nada mais do que instâncias nas quais você pode propor uma configuração de capacidade computacional para ela, de modo a atender às necessidade de sua aplicação. A principal diferença entre instâncias spot e instâncias *on demand* é que a disponibilidade para início das operações de uma instância spot não é imediata.

DICA

Use um grupo de instâncias on demand para que você possa ter como garantia um nível mínimo de recursos para as aplicações, e os recursos adicionais que forem necessário você aloca em instâncias spot, conforme a necessidade surgir.

4.6 CONSOLE DE GERENCIAMENTO

O console de gerenciamento é o ambiente que você acessa via browser através do login em http://console.aws.amazon.com, e que disponibiliza para você as funcionalidades necessárias para gerenciar as instâncias.

Para acessar o EC2, você pode ir pelo menu *Services*, ou então na tela principal do console, onde é exibida a lista de todos os serviços. De modo a agilizar o acesso, crie um atalho para o EC2 na barra de menu, conforme foi ensinado no capítulo 1.

A tela principal do painel do EC2 é a seguinte:

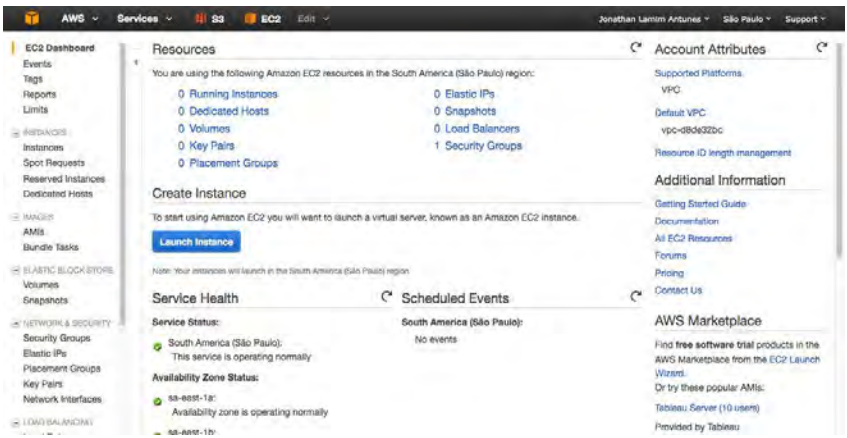


Figura 4.11: Tela principal do Amazon EC2

Essa tela traz diversas informações sobre o EC2, incluindo acesso aos mais variados recursos dele. No lado esquerdo da página, você tem um menu bem completo, com as funcionalidades do EC2 que você pode acessar e administrar, como por exemplo, a lista de instâncias (*Instances* > *Instances*), instâncias reservadas (*Reserved Instances*), entre outros.

Ao longo do capítulo e do livro, você verá mais informações e detalhes sobre o EC2.

4.7 CRIANDO UMA INSTÂNCIA

Agora que você já sabe como funciona o EC2 e o que são as instâncias, chegou a hora de criar sua primeira instância. Estando logado no console, e dentro do ambiente de gerenciamento do EC2, clique no botão *Launch Instance* para poder iniciar o processo de sua criação.

> Toda instância precisa ter a sua região definida, então verifique na barra de menu, no canto superior direito da tela, qual a região está selecionada. Utilize a região São Paulo para esse exemplo.

Passo 1 — Escolhendo a imagem do sistema operacional

O primeiro passo na criação da instância é escolher a imagem do sistema operacional que será utilizado na instância. A lista traz várias, inclusive algumas delas fazem parte do programa de gratuidade e estão marcadas com a tag *free tier eligible*, logo abaixo da logo do sistema operacional.

Nesse momento, vamos utilizar a imagem Amazon Linux AMI, que é a distribuição Linux customizada da Amazon. Para isso, clique no botão *Select* que aparece no final da linha correspondente à imagem que vamos usar (a primeira da lista).

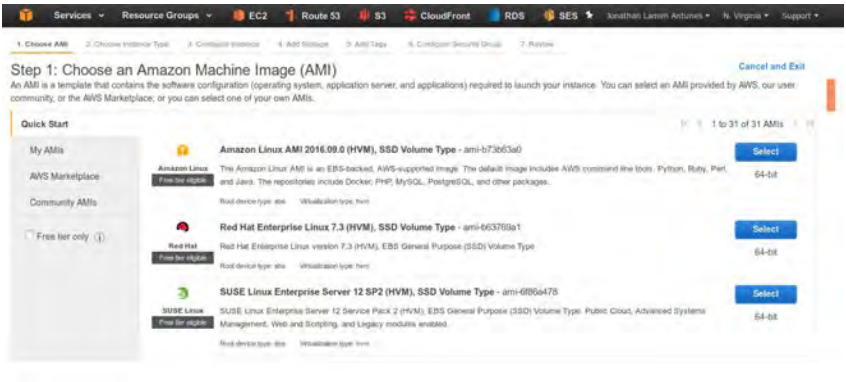


Figura 4.12: Tela de seleção da imagem do sistema operacional da instância

> Do lado esquerdo da tela, você pode ver um menu com opções para filtrar as imagens, podendo assim localizar imagens de outros sistemas operacionais diferentes, como por exemplo, o CentoOS 7, que você encontra em AWS Marketplace.

Passo 2 — Escolhendo o tipo de instância

Ao clicar em *Select*, você será redirecionado para outra tela, na qual escolherá o tipo de instância a ser utilizada. Por padrão, já vem selecionado o tipo `t2.micro`, que é uma instância de uso geral e faz parte do plano de gratuidade. É exatamente essa que você vai usar neste capítulo.

Como o tipo de instância já está selecionado, clique no botão *Next: Configure Instance Details* para dar sequência ao processo de

configuração da instância.

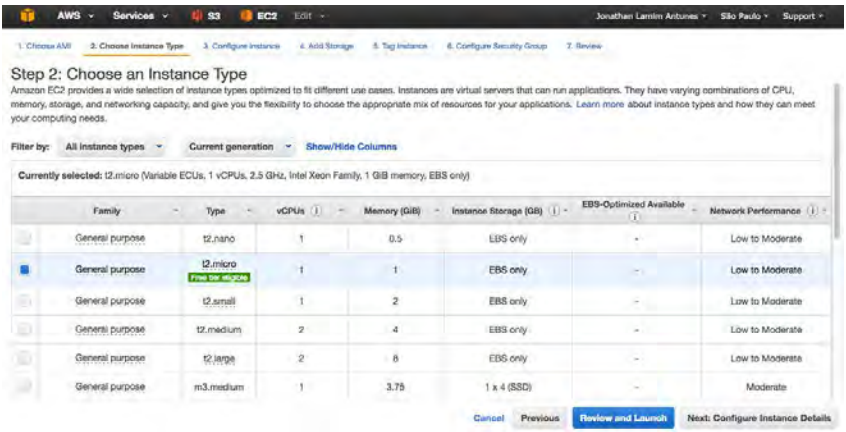


Figura 4.13: Tela de seleção do tipo da instância

Passo 3 — Configurando a instância

Nesse terceiro passo, você executará o processo de configuração da instância. Uma instância possui diversas variantes de configuração, como por exemplo, o número de instâncias (*number of instances*). Mantenha as configurações padrões da instância para que possamos utilizar a gratuidade durante esse estudo, e clique em *Next: Add Storage* para prosseguir.

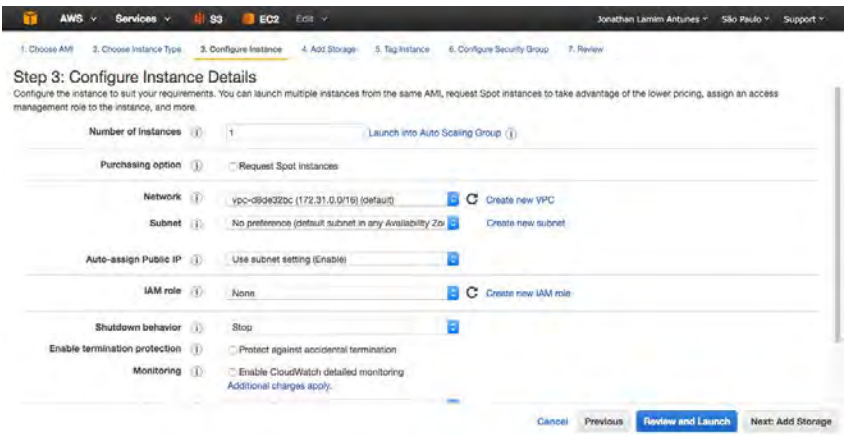


Figura 4.14: Tela de configuração da instância

Passo 4 — Definindo a configuração de armazenamento para a instância

Agora é o momento de definir qual será o espaço de armazenamento usado pela instância. Também não altere os valores padrões, mas saiba que você poderá fazer isso a qualquer momento após a finalização do processo de configuração da instância, escalando a sua configuração conforme a necessidade. Clique em *Next: Tag Instance* para prosseguir.

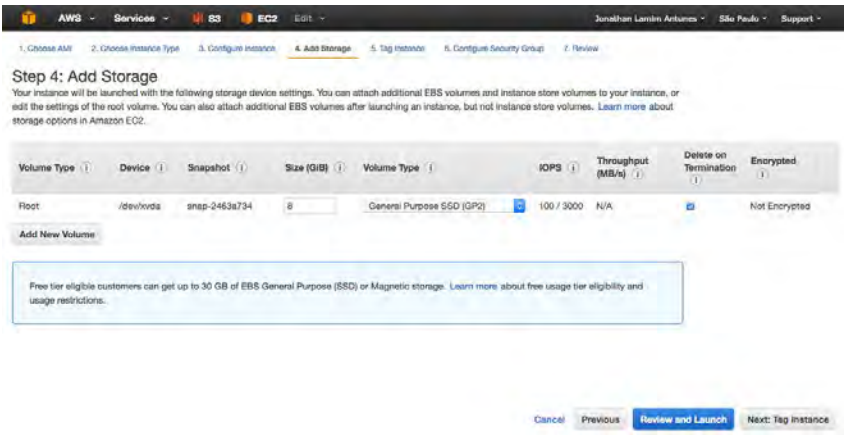


Figura 4.15: Tela de configuração do armazenamento da instância

Existe uma opção na configuração do armazenamento da instância que permite definir se o volume e o seu conteúdo serão removidos quando uma instância for excluída. Se a opção **Delete on Termination** estiver selecionada, ao excluir a instância, o volume e seu conteúdo serão excluídos. Mas se não estiver marcado, ela é removida, mas o volume e os dados continuam intactos.

Como esse volume pode ser associado a outras instâncias, e não somente à instância que está configurando, você pode usá-lo em uma outra instância para obter os dados contidos nele.

Passo 5 — Configurando uma tag de identificação para a instância

Essa tag é um nome de identificação, composto por chave e valor. A chave já vem preenchida como *Name* e você deverá informar o valor, que será o nome da instância que está sendo criada. Utilize o nome **MinhaPrimeiraInstancia** para o campo *Value* e, em seguida, clique em *Next: Configure Security Group*.

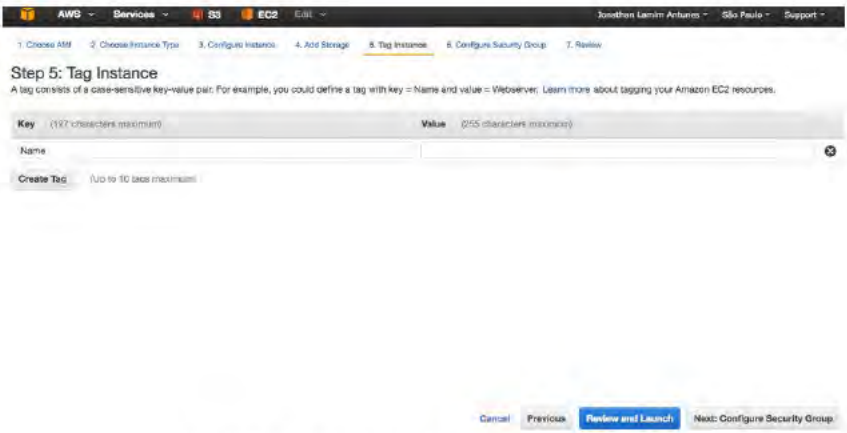


Figura 4.16: Tela de configuração da identificação da instância

Passo 6 — Configurando o grupo de segurança

Nessa etapa, você vai configurar as regras de firewall da instância pelo grupo de acesso, definindo quais portas ficarão abertas e quais são os IPs que podem acessar a instância.

Por padrão já vem criada uma regra para o acesso SSH, através da porta 22, mas ele libera o acesso a partir de qualquer IP e você precisa mudar isso. No combobox da opção **Source**, selecione a opção My IP, para que o acesso seja liberado apenas para o seu IP.

Feito isso, você criará 2 novas regras, clicando em *Add Rule*.

- **HTTP**
  - **Type:** HTTP
  - **Porta:** 80
  - **Source:** Anywhere
- **MySQL**
  - **Type:** MySQL/Aurora
  - **Porta:** 3306

- **Source:** My IP

Repare que, nesse grupo de configuração que você está criando, somente a regra para o HTTP permite o acesso irrestrito, as demais regras restringem o acesso ao seu IP. Isso é feito pois a porta 80 é usada para o acesso ao servidor web, através do browser.

Antes de finalizar, altere as informações de **Security Group Name** para *LivroAWSCap5* e de **Description** para *Grupo do Capítulo 5 do Livro AWS*. Para finalizar, clique em *Review and Lauch*.

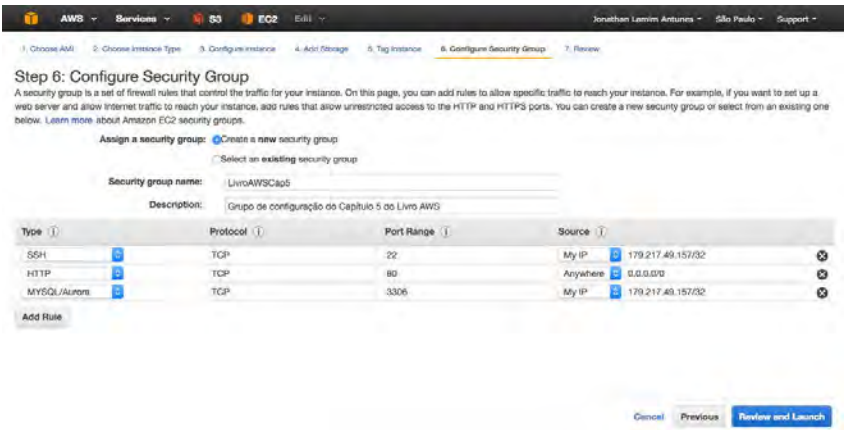


Figura 4.17: Tela de configuração do grupo de segurança

Passo 7 — Conclusão

Nesst passo, você verá todas as configurações e informações da instância, e caso alguma esteja errada, você poderá editar. Para prosseguir, clique em *Launch*.

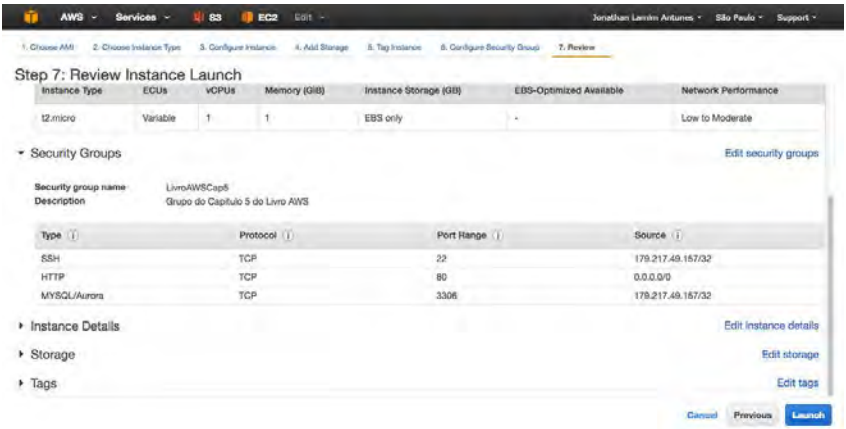


Figura 4.18: Tela de visualização das informações da instância

Será exibida uma tela solicitando a escolha ou criação de uma chave de acesso. Selecione a opção **Create a new key pair** e dê a ela o nome de *LivroAWSCap5KeyPair*. Em seguida, clique em *Download Key Pair* e salve o arquivo com o nome de `kp_livro_aws_cap_5.pem` em um local seguro no seu computador.

> O download da chave deve ser feito nesse momento, pois ao sair dessa tela, não será possível voltar e fazer o download dela.

Agora clique em *Launch Instances* para finalizar o processo.

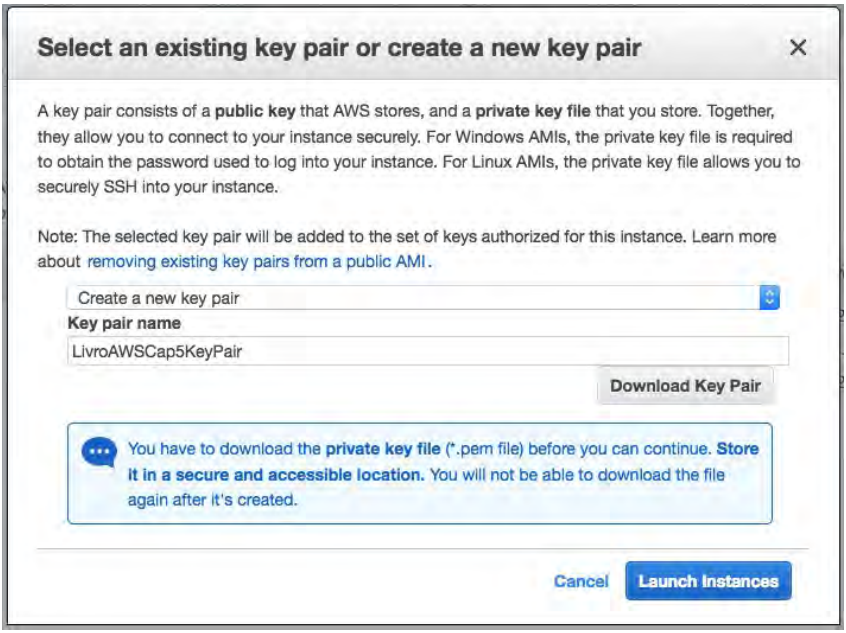


Figura 4.19: Tela de criação da chave de acesso

Após finalizar o processo, você verá uma tela parecida com a da figura seguinte, que trará informações sobre a instância.

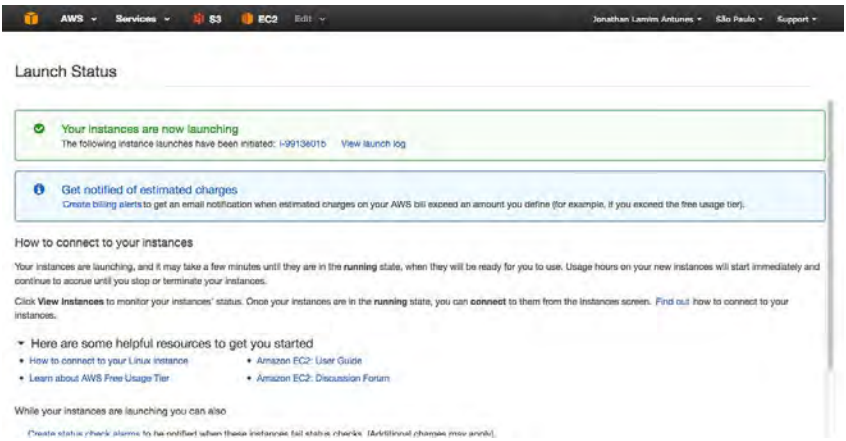


Figura 4.20: Tela pós-criação da instância

Clique em *View Instances* para visualizar as instâncias criadas em sua conta.

Na figura a seguir, você verá a lista com a instância que acabou de criar, e as informações sobre ela logo abaixo. Fique atento às informações marcadas em vermelho nela.

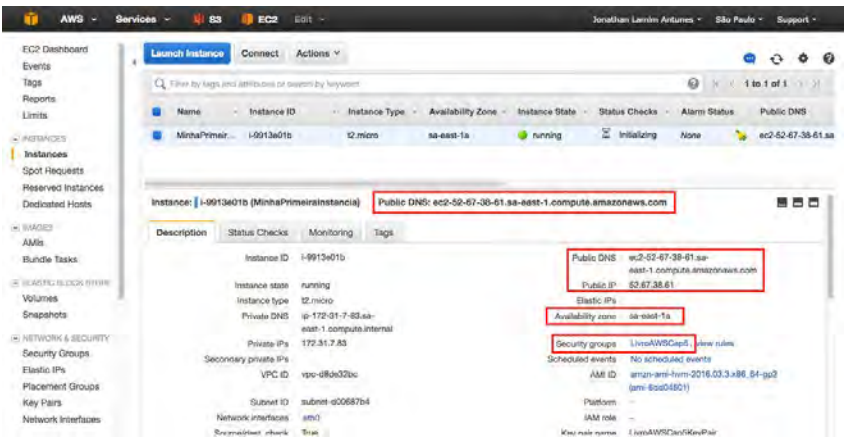


Figura 4.21: Tela de exibição das instâncias

Nesse momento, a instância ainda não está acessível, pois não temos uma aplicação configurada nela. Porém, fique tranquilo, que faremos isso mais adiante.

4.8 CONECTANDO À INSTÂNCIA

Existem 3 formas de se conectar uma instância: pelo console de gerenciamento, pelo terminal via SSH, e por SFTP.

Conectando à instância pelo console de gerenciamento

A Amazon possui um client SSH que pode ser acessado diretamente pelo console de gerenciamento. Para utilizá-lo, vá até a lista das instâncias, selecione a instância que acabou de criar e clique

no botão *Connect*. Ao clicar, aparecerá uma tela conforme a seguir:

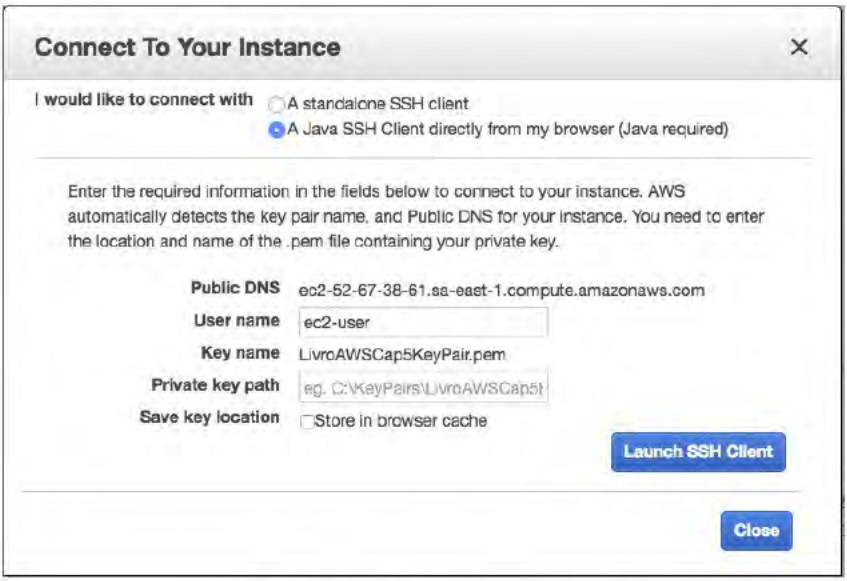


Figura 4.22: Tela de conexão SSH no console

Nessa tela, você tem 2 opções de acesso com suas respectivas configurações para acesso.

- **A standalone SSH client** — Nessa opção, será utilizado o cliente SSH que você tiver instalado em seu computador. Ao selecioná-la, serão exibidas instruções para que possa efetuar a conexão diretamente em seu cliente SSH.

- **A Java SSH Client directly from my browser (Java required)** — Nessa opção você vai utilizar o client SSH da Amazon, desenvolvido em Java, e que será usado em uma janela do browser. Ao selecioná-la, serão exibidas as informações de conexão.

Pode ser que, ao escolher essa opção, você precise confirmar algumas informações após clicar em *Launch SSH Client*. Essas informações são referentes a arquivos e diretórios que precisam ser criados. Basta ler as informações nas caixas de diálogo e ir fornecendo as informações conforme for solicitado.

Após escolher o cliente SSH que deseja utilizar, basta clicar no botão *Launch SSH Client* para poder abri-lo.

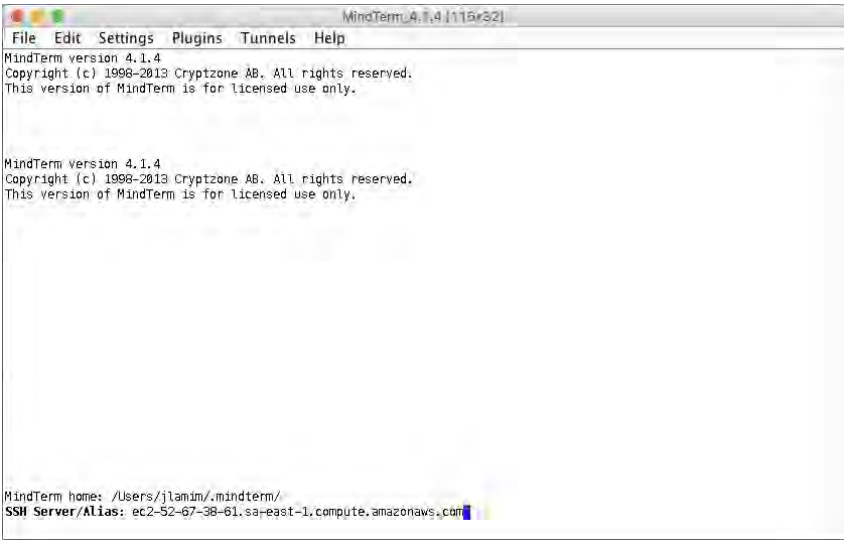


Figura 4.23: Cliente SSH da Amazon

Conectando à instância via SSH (terminal)

O acesso via SSH pode ser feito via terminal (usado em larga escala por usuários de Linux e Mac), ou então através de clientes de SSH (utilizados em ambiente Windows), como por exemplo o PuTTY, KiTTY e Terminals.

Também é possível acessar a instância via SSH usando o prompt do Windows.

Para acessar via SSH, você deve abrir o terminal, ou o prompt e digitar o seguinte comando:

```
ssh -i path/da/chave/de/acesso/kp_livro_aws_cap_5.pem ec2-user@ip-da-instancia
```

Onde:

- `path/da/chave/de/acesso/` : deve corresponder à pasta onde está a chave que foi gerada durante a criação da instância.
- `ip-da-instancia` : deve ser o IP da instância, que é informado nos detalhes logo abaixo da lista de instâncias.

O usuário `ec2-user` é o usuário padrão das instâncias do EC2, e não possui senha, pois a autenticação é feita através da chave de acesso.

Após executar o comando informado anteriormente, você deverá ver um tela semelhante à exibida a seguir:

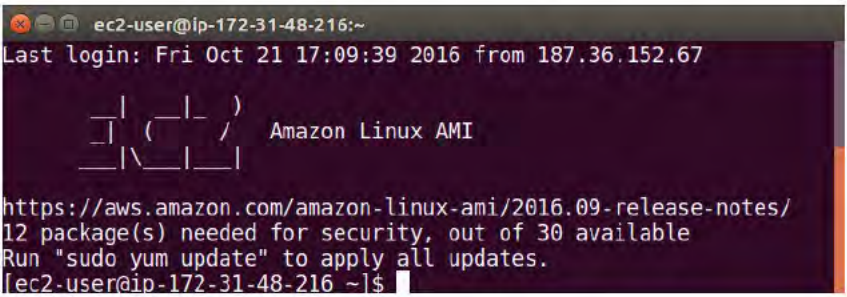


Figura 4.24: Conexão SSH via Terminal

A partir daí, é só utilizar os comandos do Linux para poder executar operações como: navegar por diretórios, criar diretórios e arquivos, instalar pacotes. Veja no **Apêndice 2** alguns comandos básicos do Linux.

Conectando à instância via SFTP

Para se conectar à instância via SFTP (bastante usado para a transferência de arquivos), você precisará de um cliente FTP instalado, da chave gerada na criação da instância (`LivroAWSCap5KeyPair.pem`) e dos dados de acesso (IP e usuário, que no caso é `ec2-user`). Neste livro, é utilizado o FileZilla como cliente FTP, e você pode fazer download dele acessando https://filezilla-project.org.

Após instalar, abra-o para que seja iniciado o processo de configuração. O primeiro passo é configurar um novo site para ser acessado, que nesse caso será a instância criada.

Vá até o menu **Arquivo** > **Gerenciador de Sites** para definir as informações de acesso à instância. Você deve inserir as informações conforme a figura seguinte, lembrando de que, em *Host*, você deve colocar o IP da sua instância.

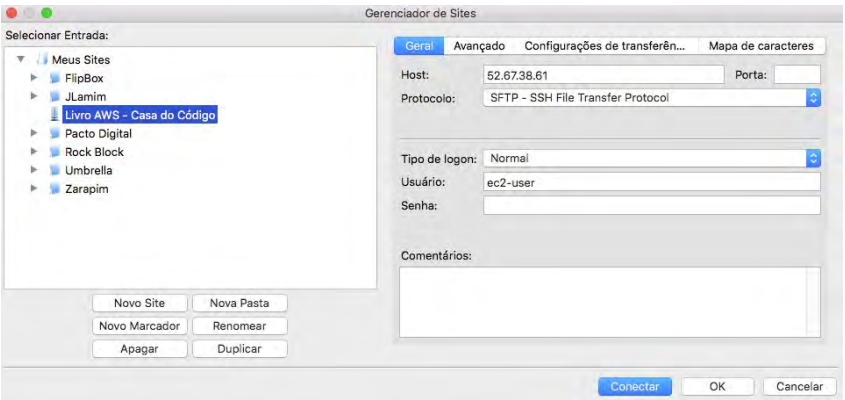


Figura 4.25: Conexão SFTP com FileZilla

Após preencher com as devidas informações, clique em *Ok* e acesse **Editar > Configurações** para poder instalar a chave de acesso à instância, pois ela substitui o uso da senha para a conexão.

Dentro das opções de configuração, acesse **Conexão > SFTP** e clique em *Adicionar arquivos de chave*. Nesse momento, você deverá selecionar o arquivo `LivroAWSCap5KeyPair.pem` e, em seguida, clicar em *Ok*.

> Pode ser que o FileZilla solicite a conversão do arquivo `.pem` para um arquivo `.ppk`. Ele faz a conversão automaticamente, basta que você confirme caso a caixa de diálogo de conversão seja exibida.

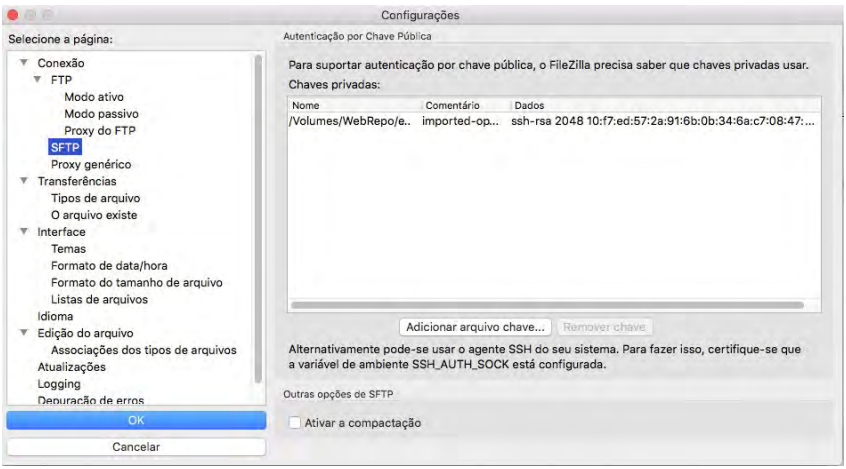


Figura 4.26: Conexão SFTP com FileZilla

Feito isso, você já pode se conectar à instância via SFTP, bastando selecionar a instância na lista de sites e clicar em *Conectar*. Logo após se conectar, você será direcionado para o diretório `/home/ec2-user`, que é o diretório do usuário que foi usado para a

conexão. Para ir até a raiz do servidor, basta remover o caminho do diretório da barra de endereço, deixando somente uma `/`.

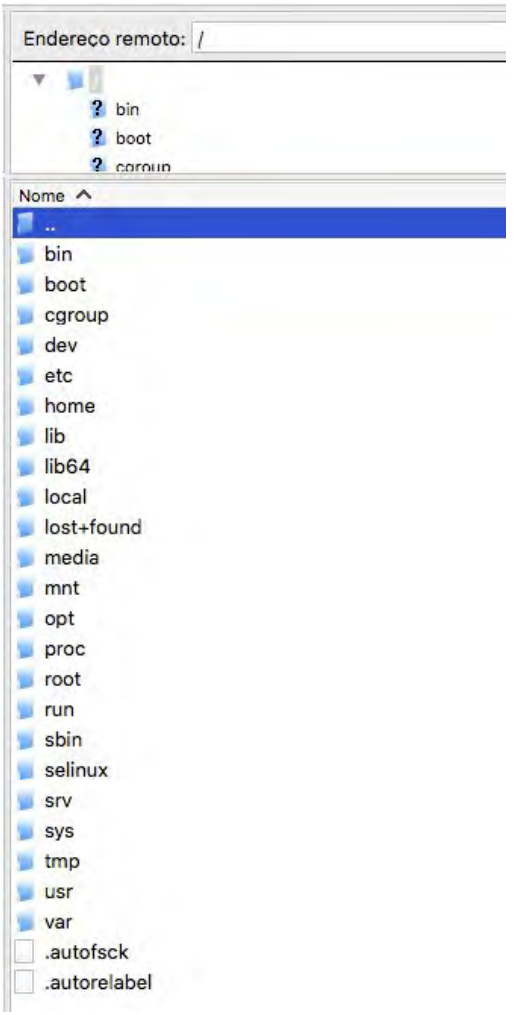


Figura 4.27: Árvore de diretórios da instância

4.9 CONSIDERAÇÕES FINAIS

Neste capítulo, você viu detalhes sobre o ecossistema do EC2, os tipos de instância, seus recursos e sugestões de uso. Além disso, você

aprendeu a criar e configurar uma instância dentro do EC2.

Nos próximos capítulos, você aprenderá a instalar o Apache, PHP e MySQL na sua instância, para que em seguida possa criar um site usando o Wordpress nessa instância.

Links úteis

- **Documentação do EC2:** http://aws.amazon.com/pt/ec2
- **Documentação sobre os tipos de instância:** http://aws.amazon.com/pt/ec2/instance-types

Capítulo 5

AMAZON EC2 — INSTALANDO O APACHE, PHP E MYSQL NA INSTÂNCIA

5.1 INTRODUÇÃO

Neste capítulo, você começará a preparar a sua instância para hospedar nela um site feito com o CMS Wordpress. Será feita a instalação do Apache, do MySQL e do PHP, passo a passo.

Conecte-se via SSH à instância para poder prosseguir com as instalações.

5.2 ATUALIZANDO OS PACOTES E DEPENDÊNCIAS INSTALADOS

O primeiro passo a ser realizado é a atualização dos pacotes e dependências do Linux. Para isso, você deverá executar o comando:

```
sudo yum update -y
```

O processo pode demorar um pouco dependendo de quantos pacotes e dependências precisarem ser atualizados. Ao final da execução, será exibida a mensagem `Concluído!` no terminal.

5.3 INSTALANDO O APACHE, O MYSQL E O PHP

Para instalar os pacotes do Apache, MySQL e PHP, execute o comando:

```
sudo yum install -y httpd24 php56 mysql55-server php56-mysqlnd
```

Como serão instalados 4 pacotes, e será necessário fazer o download de cada um deles, esse processo pode demorar mais do que o processo de atualização executado no primeiro passo. Mais uma vez, ao final da execução, será exibida a mensagem `Concluído!` no terminal.

Após a conclusão, execute o comando `sudo service httpd start` para inicializar o Apache, e assim ativar o servidor web.

5.4 CONFIGURANDO O APACHE PARA INICIAR JUNTO COM O SISTEMA

Anteriormente, você inicializou o Apache manualmente, mas caso sua instância seja reiniciada, o Apache permanecerá desativado. Para que ele passe a iniciar junto com a inicialização do sistema, execute o comando a seguir:

```
sudo chkconfig httpd on
```

Esse comando não dá nenhum retorno de execução na tela, como os anteriores, então para verificar se a configuração foi aplicada, execute o comando `chkconfig --list httpd` e veja se o retorno será algo semelhante ao apresentado a seguir:

```
[ec2-user ~]$ chkconfig --list httpd
httpd           0:off   1:off   2:on    3:on    4:on    5:on    6:
off
```

Se os níveis 2, 3, 4 e 5 estiverem marcados como `on` , é sinal de

que o comando para inicialização junto com o sistema foi executado com sucesso.

Para testar o servidor, acesse a instância pelo browser, utilizando o DNS público da usa instância. Será algo semelhante a `ec2-52-67-75-163.sa-east-1.compute.amazonaws.com` , em que o IP vai variar e talvez a região, pois dependerá da região onde você criou a instância.

Ao acessar via browser, o resultado deverá ser algo como a figura:

Figura 5.1: Teste do Apache no browser

5.5 AJUSTANDO AS PERMISSÕES DE ACESSO AO DIRETÓRIO WWW

Após instalar o Apache, será criado um diretório `www` dentro de `var` e, dentro dele, alguns outros. Para visualizar esses diretórios, execute o comando adiante:

```
ls -l /var/www
```

O retorno desse comando deve ser algo como:

```
drwxr-xr-x 2 root root 4096 Mar  7 22:32 cgi-bin
drwxr-xr-x 3 root root 4096 Jul 12 18:37 error
drwxr-xr-x 2 root root 4096 Mar  7 22:32 html
drwxr-xr-x 3 root root 4096 Jul 12 18:37 icons
drwxr-xr-x 2 root root 4096 Jul 12 18:37 noindex
```

Como pode ser visto, somente o usuário `root` tem permissão sobre esses diretórios. Para dar permissão ao usuário `ec2-user`, você deve criar um novo grupo e adicioná-lo a esse grupo, dando então permissão de acesso a ele.

Para criar o grupo `www`:

```
sudo groupadd www
```

E agora adicionar o usuário ao grupo:

```
sudo usermod -a -G www ec2-user
```

Após criar o grupo e adicionar o usuário, se desconecte da instância executando o comando `exit` e se reconecte novamente. Para verificar se o usuário `ec2-user` recebeu a permissão, execute o comando `groups` e veja se o retorno será como o seguinte:

```
ec2-user wheel www
```

Agora execute o comando `sudo chown -R root:www /var/www` para alterar a propriedade de `www` para `/var/www`. Em seguida, execute os comandos adiante para atualizar as permissões dos diretórios existentes.

```
sudo chmod 2775 /var/www
sudo find /var/www -type d -exec chmod 2775 {} \;
```

O último comando executado serve para que as permissões sejam aplicadas de maneira recursiva, atingindo os arquivos e diretórios que estiverem dentro de `/var/www`.

5.6 TESTANDO O SERVIDOR WEB

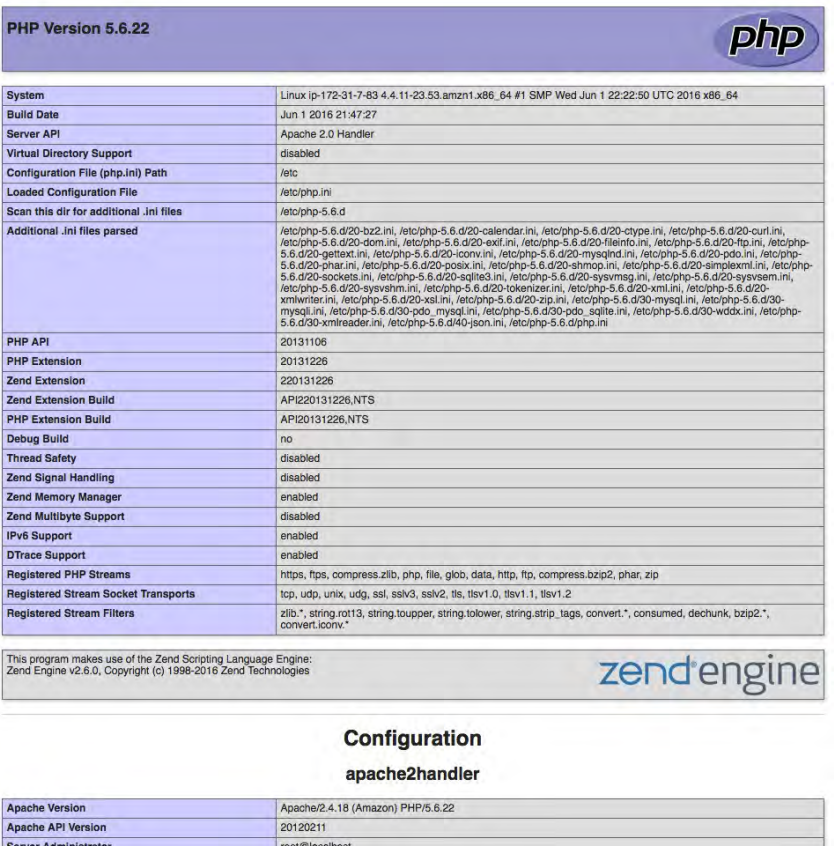
Agora que as configurações já foram aplicadas, você deve testar o servidor e verificar se está tudo funcionando. Para que você possa testar e, ao mesmo tempo, obter informações sobre as configurações do Apache, PHP e MySQL, você vai criar um arquivo

`phpinfo.php` dentro de `/var/www`, usando o comando:

```
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

Esse comando cria o arquivo `phpinfo.php`, contendo o código `<?php phpinfo(); ?>`.

Após executar o comando, volte ao browser e acesse a instância pelo DNS Público, só que dessa vez chamando o arquivo `phpinfo.php`. Na tela você deverá ver algo semelhante à figura a seguir:



PHP Version 5.6.22

System	Linux ip-172-31-7-83 4.4.11-23.53.amzn1.x86_64 #1 SMP Wed Jun 1 22:22:50 UTC 2016 x86_64
Build Date	Jun 1 2016 21:47:27
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php-5.6.d
Additional .ini files parsed	/etc/php-5.6.d/20-bz2.ini, /etc/php-5.6.d/20-calendar.ini, /etc/php-5.6.d/20-ctype.ini, /etc/php-5.6.d/20-curl.ini, /etc/php-5.6.d/20-dom.ini, /etc/php-5.6.d/20-exif.ini, /etc/php-5.6.d/20-fileinfo.ini, /etc/php-5.6.d/20-ftp.ini, /etc/php-5.6.d/20-gettext.ini, /etc/php-5.6.d/20-iconv.ini, /etc/php-5.6.d/20-mysqlnd.ini, /etc/php-5.6.d/20-pdo.ini, /etc/php-5.6.d/20-phar.ini, /etc/php-5.6.d/20-posix.ini, /etc/php-5.6.d/20-shmop.ini, /etc/php-5.6.d/20-simplexml.ini, /etc/php-5.6.d/20-sockets.ini, /etc/php-5.6.d/20-sqlite3.ini, /etc/php-5.6.d/20-sysvmsg.ini, /etc/php-5.6.d/20-sysvsem.ini, /etc/php-5.6.d/20-sysvshm.ini, /etc/php-5.6.d/20-tokenizer.ini, /etc/php-5.6.d/20-xml.ini, /etc/php-5.6.d/20-xmlwriter.ini, /etc/php-5.6.d/20-xsl.ini, /etc/php-5.6.d/20-zip.ini, /etc/php-5.6.d/30-mysql.ini, /etc/php-5.6.d/30-mysqli.ini, /etc/php-5.6.d/30-pdo_mysql.ini, /etc/php-5.6.d/30-pdo_sqlite.ini, /etc/php-5.6.d/30-wddx.ini, /etc/php-5.6.d/30-xmlreader.ini, /etc/php-5.6.d/40-json.ini, /etc/php-5.6.d/php.ini
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,NTS
PHP Extension Build	API20131226,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	enabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.6.0, Copyright (c) 1998-2016 Zend Technologies

Configuration

apache2handler

Apache Version	Apache/2.4.18 (Amazon) PHP/5.6.22
Apache API Version	20120211

Figura 5.2: Teste do servidor web

Caso você não consiga visualizar um resultado semelhante, ou seja exibido algum erro, então o processo de instalação não foi executado com sucesso. Para verificar se todos os pacotes necessários foram instalados, execute o comando `sudo yum list installed httpd24 php56 mysql55-server php56-mysqlnd` para obter o status dos pacotes que foram instalados logo no início deste capítulo.

Se algum dos pacotes anteriores não for exibido na lista, é porque ele não foi instalado e você deverá repetir a sua instalação. Para isso, utilize o comando `sudo yum install nome_do_pacote` .

5.7 INICIALIZANDO O MYSQL

Para inicializar o MySQL, execute o comando:

```
sudo service mysqld start
```

Com o MySQL inicializado, vamos executar o processo de instalação do `mysql_secure_installation` . Execute o seguinte comando para processar a instalação:

```
sudo mysql_secure_installation
```

Ao executar o comando, será solicitada a senha para o usuário `root` do MySQL. Por padrão, o usuário `root` não possui senha, então pressiona *Enter* para prosseguir. Será perguntado então se deseja definir a senha para o usuário `root` , digite **Y** e, em seguida, informe a senha.

A partir desse momento, serão feitas algumas perguntas. Veja na lista a seguir quais são e o que deverá responder:

- **Remover usuários anônimos:** Y
- **Desativar login remoto para o usuário `root` :** Y
- **Remover o banco de dados `test` e o acesso a ele:** Y
- **Recarregar a tabela de privilégios:** Y

Agora o MySQL está instalado e configurado de maneira segura, mas ainda falta um último passo, que é fazer com que ele seja inicializado junto com o sistema operacional, assim como foi feito com o Apache. Para isso, execute os comandos a seguir:

```
sudo chkconfig mysqld on
chkconfig mysqld --list
```

Como retorno deverá obter:

```
mysqld              0:não    1:não    2:sim    3:sim    4:sim    5:
sim    6:não
```

Se os níveis 2, 3, 4 e 5 estiverem marcados como `on` , é sinal de que o comando para inicialização junto com o sistema foi executado com sucesso e você concluiu a configuração do servidor web na sua instância.

5.8 CONSIDERAÇÕES FINAIS

Neste capítulo, você viu um passo a passo de como instalar e configurar o Apache, o MySQL e o PHP em uma instância, deixando-a pronta para receber uma aplicação web. No próximo campítulo, você vai aprender a instalar o Wordpress nessa instância.

Links úteis

- **Apache:** http://www.apache.org/
- **MySQL:** https://www.mysql.com/
- **PHP:** http://php.net/

Capítulo 6

AMAZON EC2 — INSTALANDO E CONFIGURANDO O WORDPRESS

Neste capítulo, você aprenderá a realizar a instalação do Wordpress em uma instância EC2, passo a passo. O Wordpress é o CMS para blog mais utilizado e que possui uma curva de conhecimento muito pequena para que seja instalado, por isso ele foi o escolhido.

Acesse a sua instância via SSH para que possa executar o passo a passo da instalação.

6.1 DOWNLOAD DA VERSÃO MAIS RECENTE DO WORDPRESS

Pelo comando a seguir, você fará o download da versão mais recente do Wordpress para o diretório de usuário da instância, que no caso é `ec2-user` .

```
wget https://wordpress.org/latest.tar.gz
```

É um processo relativamente rápido. Ao término, você deverá ter um resultado em sua tela parecido com o apresentado a seguir:

```
--2016-07-15 15:46:21--  https://wordpress.org/latest.tar.gz
Resolvendo wordpress.org (wordpress.org)... 66.155.40.250, 66.155.
40.249
Conectando-se a wordpress.org (wordpress.org)|66.155.40.250|:443..
. conectado.
A requisição HTTP foi enviada, aguardando resposta... 200 OK
Tamanho: 7773389 (7,4M) [application/octet-stream]
Salvando em: “latest.tar.gz”

latest.tar.gz       100%[===================>]   7,41M  1,92MB/s
  in 3,9s

2016-07-15 15:46:25 (1,92 MB/s) - “latest.tar.gz” salvo [7773389/7
773389]
```

Com o download concluído, é hora de descompactar o arquivo. Para isso, você deve utilizar o comando `tar -xzf latest.tar.gz` no terminal. Esse comando não exibe nenhum retorno na tela, então você precisará executar o comando `ls` para verificar se o arquivo foi descompactado. Se for exibida a informação como mostrada a seguir, é porque a descompactação do arquivo foi executada com sucesso.

```
latest.tar.gz  wordpress
```

6.2 CRIANDO O BANCO DE DADOS

Para que o Wordpress funcione corretamente, é preciso conectá-lo a um banco de dados MySQL. Sendo assim, você criará nesse passo o banco de dados e realizar a configuração dos arquivos no Wordpress.

> Veja no **Apêndice 3** alguns comandos básicos do MySQL

Se você executou corretamente todos os procedimentos do capítulo anterior, então a conexão com o banco de dados está sendo

iniciada automaticamente junto com o sistema operacional da instância. Caso não tenha aplicado a configuração para iniciar o MySQL automaticamente, execute o comando `sudo service mysqld start` antes de prosseguir.

Logando no MySQL

Para executar qualquer procedimento no MySQL, é preciso estar logado. Para fazer isso, na linha de comando, é simples, execute o comando `mysql -u root -p` . Ao ser solicitada a senha, informe a mesma que você configurou no capítulo anterior ao instalar o MySQL.

Após executar o comando, sua tela no terminal deverá exibir uma mensagem parecida com a mostrada a seguir:

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.5.46 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights
 reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current inpu
t statement.

mysql>
```

Se estiver vendo essa mensagem ou semelhante no terminal, é sinal de que a conexão foi feita e agora você já pode executar comandos do MySQL via terminal para manipular bancos de dados.

Veja no **Apêndice 3** alguns comandos básicos para utilizar o MySQL através da linha de comando.

Criando um usuário para acessar o banco de dados

Execute o comando a seguir para criar um novo usuário no banco de dados, ele se chamará `wordpress-user`.

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'senha_do_u
suario';
```

Após executar o comando, deverá ver a seguinte mensagem: `Query OK, 0 rows affected (0.00 sec)`. Se foi exibida, é sinal de que o usuário foi criado corretamente.

Criando o banco de dados e definindo os privilégios de acesso

Agora você deverá criar o banco de dados, que se chamará `wordpress-db`. Para isso, execute o comando a seguir:

```
CREATE DATABASE `wordpress-db`;
```

Em seguida, defina os privilégios desse usuário executando o comando:

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"loca
lhost";
```

E para finalizar, execute o comando `FLUSH PRIVILEGES;` para que os privilégios definidos sejam aplicados. Agora você já pode executar o comando `exit` para sair do ambiente de gerenciamento do MySQL no terminal.

6.3 CONFIGURANDO A CONEXÃO DO

WORDPRESS COM O BANCO DE DADOS

Com o banco de dados criado, é hora de atualizar as configurações de conexão no Wordpress. Acesse o diretório `wordpress` que está na raiz do seu usuário com o comando `cd wordpress` . Em seguida, execute o comando `cp wp-config-sample.php wp-config.php` para criar uma cópia do arquivo de configuração do Wordpress.

Concluída a cópia, execute o comando `'nano wp-config.php'` para abrir o arquivo de configuração no editor de texto dentro do terminal e atualizar as configurações de conexão. Ao abrir o arquivo no editor de texto, localize as linhas descritas a seguir, alterando o seu conteúdo conforme mostrado:

```
// ** MySQL settings - You can get this info from your web host **
 //
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress-db');

/** MySQL database username */
define('DB_USER', 'wordpress-user');

/** MySQL database password */
define('DB_PASSWORD', 'senha_que_voce_definiu');
```

No final do arquivo, você encontrará um grupo de constantes que definem chaves de segurança. Você deverá informar uma chave diferente para cada item, conforme mostrado a seguir:

- **Como está no arquivo**

```
define('AUTH_KEY',         'put your unique phrase here');
define('SECURE_AUTH_KEY',  'put your unique phrase here');
define('LOGGED_IN_KEY',    'put your unique phrase here');
define('NONCE_KEY',        'put your unique phrase here');
define('AUTH_SALT',        'put your unique phrase here');
define('SECURE_AUTH_SALT', 'put your unique phrase here');
define('LOGGED_IN_SALT',   'put your unique phrase here');
define('NONCE_SALT',       'put your unique phrase here');
```

- **Exemplo de como deve ficar**

```
define( 'AUTH_KEY',         't`DK%X:>xy|e-Z(BXb/f(Ur`8#~UzUQG-^_Cs
_GHs5U-&Wb?pgn^p8(2@}IcnCa|' );
define( 'SECURE_AUTH_KEY',  'D&ovlU#|CvJ##uNq}bel+^MFtT&.b9{UvR]g%
ixsXhGlRJ7q!h}XWdEC[BOKXssj' );
define( 'LOGGED_IN_KEY',    'MGKi8Br(&{H*~&0s;{k0<S(O:+f#WM+q|npJ-
+P;RDKT:~jrmgj#/-,[hOBk!ry^' );
define( 'NONCE_KEY',        'FIsAsXJKL5ZlQo)iD-pt??eUbdc{_Cn<4!d~y
qz))&B D?AwK%)+)F2aNwI|siOe' );
define( 'AUTH_SALT',        '7T-!^i!0,w)L#JK@pc2{8XE[DenYI^BVf{L:j
vF,hf}zBf883td6D;Vcy8,S)-&G' );
define( 'SECURE_AUTH_SALT', 'I6`V|mDZq21-J|ihb u^q0F }F_NUcy`l,=ob
Gtq*p#Ybe4a31R,r=|n#=]@]c #' );
define( 'LOGGED_IN_SALT',   'w<$4c$Hmd%/*]`Oom>(hdXW|0M=X={we6;Mpv
tg+V.o<$|#_}qG(GaVDEsn,~*4i' );
define( 'NONCE_SALT',       'a|#h{c5|P &xWs4IZ20c2&%4!c(/uG}W:mAvy
<I44`jAbup]t=]V<`}.py(wTP%%' );
```

> Não utilize essas chaves do exemplo. Gere chaves fortes para que a segurança seja ainda maior.

Após editar, salve o arquivo e saia do editor.

6.4 COPIANDO OS ARQUIVOS DO WORDPRESS PARA O DIRETÓRIO PRINCIPAL DO SERVIDOR

Agora que as configurações já foram realizadas, é hora de mover os arquivos do diretório `wordpress` para o diretório `/var/www/html` do servidor. Para isso, execute o comando `mv * /var/www/html/` no terminal.

Para verificar se os arquivos foram movidos com sucesso, execute o comando `cd /var/www/html` e, em seguida, `ls` . Se for exibido algo como o conteúdo a seguir, é sinal de que os arquivos

foram copiados com sucesso.

```
index.php    readme.html      wp-blog-header.php    wp-config-samp
le.php  wp-includes        wp-login.php     wp-signup.php
license.txt  wp-activate.php  wp-comments-post.php  wp-content
        wp-links-opml.php  wp-mail.php      wp-trackback.php
phpinfo.php  wp-admin         wp-config.php         wp-cron.php
        wp-load.php        wp-settings.php  xmlrpc.php
```

Use o DNS público da instância para verificar se está sendo aberta a página de configuração do Wordpress no browser. Ao acessar, deverá ser exibido algo como a figura a seguir:

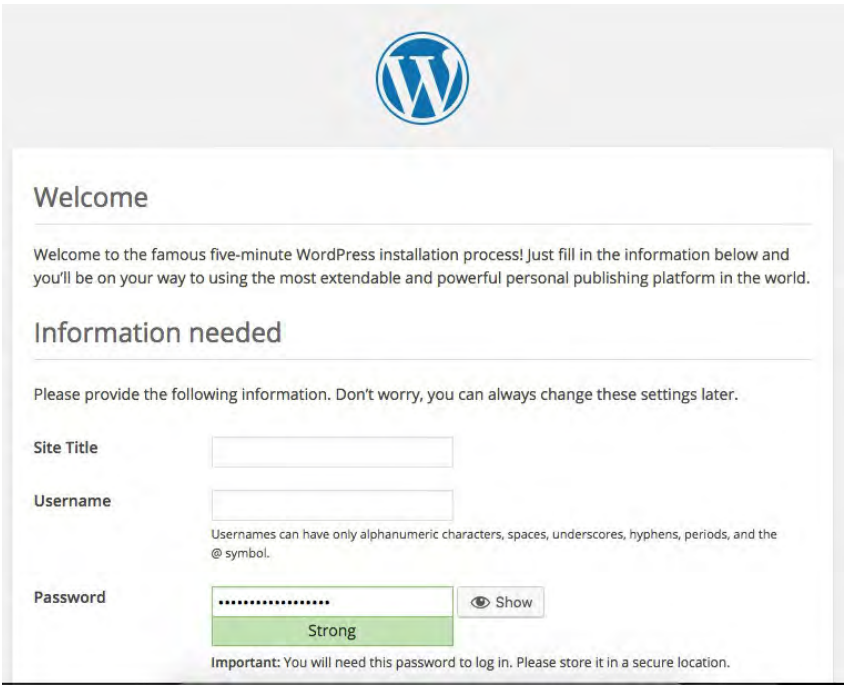


Figura 6.1: Tela de configuração do Wordpress

Preencha os campos solicitados e, em seguida, acesse o DNS público novamente. Você deverá ver algo como:

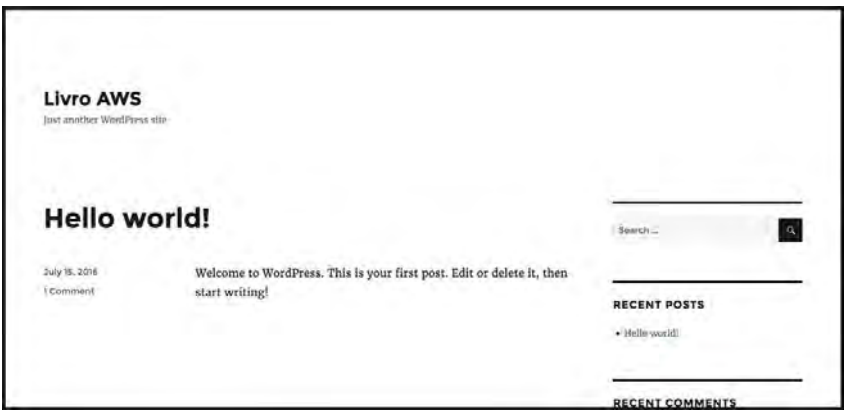


Figura 6.2: Tela inicial do Wordpress após configuração

6.5 INSTALANDO O WORDPRESS VIA SFTP

Nos passos anteriores, você aprendeu a instalar o Wordpress na sua instância usando o terminal, sem precisar de nenhuma transferência de arquivo da sua máquina para a instância. Um procedimento bastante comum — usado geralmente em hospedagens compartilhadas — é fazer o download do Wordpress para a máquina local, e em seguida enviar via FTP para o servidor.

Para realizar esse processo, você deverá estar com os arquivos do Wordpress na sua máquina local e conectado à instância via FTP. Localize os arquivos locais e envie para a instância, no diretório `/var/www/html` . Feito isso, você precisará criar o usuário e o banco de dados para se conectar. Então, siga os passos apresentados neste capítulo para executar esses procedimentos.

Com os arquivos já copiados para instância e o banco de dados configurados, basta você acessar o DNS público da instância que o Wordpress lhe guiará pelo processo de configuração. Ele vai solicitar as informações de conexão com o banco de dados e as configurações solicitadas no processo que foi realizado anteriormente.

6.6 CONSIDERAÇÕES FINAIS

Neste capítulo, você aprendeu a fazer uma instalação do Wordpress em uma instância EC2 de maneira prática e rápida. Nos próximos capítulos, você vai aprender sobre outros serviços da AWS, e vai aplicar esses conhecimentos nessa instalação do Wordpress que acabou de fazer.

Links úteis

- **Wordpress:** http://wordpress.org

Capítulo 7

AMAZON RDS (RELATIONAL DATABASE SERVICE)

7.1 INTRODUÇÃO

O RDS (ou Relational Database Service) é um serviço de banco de dados relacional disponibilizado pela Amazon que facilita o gerenciamento dos bancos de dados. Ele tem como principais vantagens a escalabilidade e o autogerenciamento.

Nos capítulos 5 e 6, você montou uma instância com Linux, Apache, MySQL e PHP, e instalou nela o Wordpress. Da forma como foi feito, toda a estrutura de banco de dados está localizada na instância e o seu gerenciamento é totalmente manual.

Para aplicações pequenas, usar o banco de dados direto na instância do EC2 não vai gerar problemas de gerenciamento, mas aplicações maiores vão gerar bastante trabalho manual de gerenciamento desse banco de dados.

Ao utilizar um banco de dados autogerenciável no RDS, você ampliará as possibilidade de automatização de tarefas.

Ao utilizar um banco de dados no RDS, você poderá:

- Fazer o gerenciamento de forma automática;

- Automatizar as rotinas de backup;
- Fazer replicações automáticas do banco de dados;
- Atualizar o software de banco de dados automaticamente;
- Monitorar o banco de dados via CloudWatch.

Atualmente, o RDS permite a criação de banco de dados Aurora, MySQL, MariaDB, PostgreSQL, Oracle (EE, SE, SE One, SE Two) e SQL Server (Express, Web, SE, EE). Essas opções possibilitam o uso dele para os mais diversos tipos e portes de aplicação.

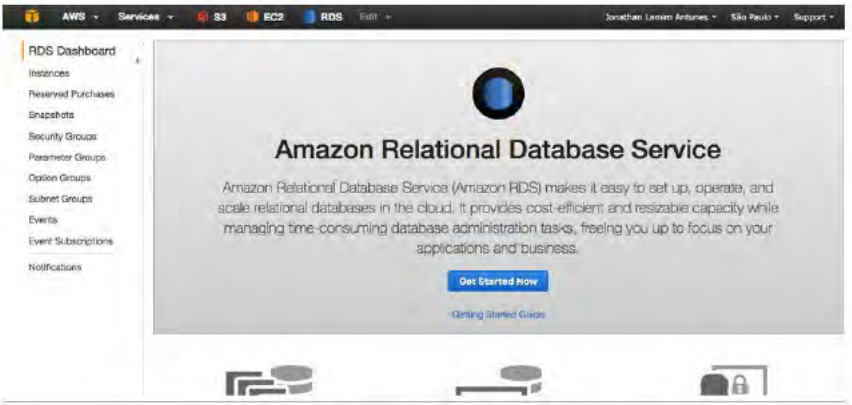


Figura 7.1: RDS Dashboard

7.2 CRIANDO UM BANCO DE DADOS NO RDS

O RDS também trabalha com o conceito de instâncias. Sendo assim, para cada banco de dados criado você terá uma nova instância no RDS.

Para criar uma nova instância, clique no menu *Instances* do lado esquerdo, e depois em *Launch DB Instance*. Em seguida, selecione o tipo de banco de dados **MySQL**, e clique *Select* para prosseguir.

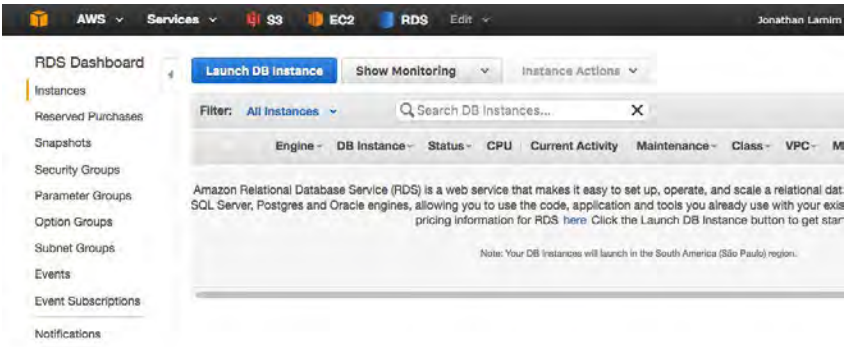


Figura 7.2: Criando um banco de dados no RDS

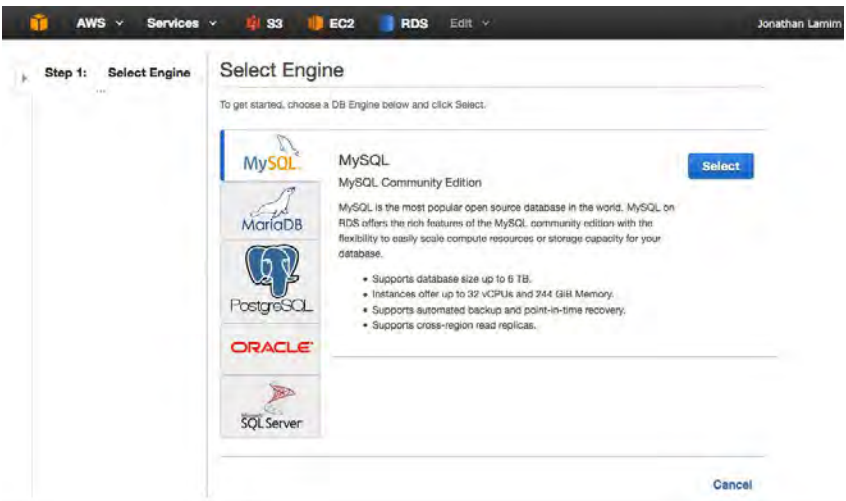


Figura 7.3: Selecionando um tipo de banco de dados no RDS

Logo após você escolher o tipo de banco de dados, será aberta uma tela onde você poderá escolher se vai utilizar o banco de dados em ambiente de produção (Production) ou em ambiente de desenvolvimento (Dev/Test).

Uma diferença importante a ser observada entre essas duas opções é que se escolher o uso para ambiente de desenvolvimento, poderá utilizar dentro do programa de gratuidade. Mas escolhendo

o ambiente de produção, pagará pelo seu uso.

Optando pelo ambiente de produção, você terá mais recursos disponíveis, como por exemplo a criação de réplicas do banco de dados em zonas diferentes (Deploy Multi-AZ) e o aumento da velocidade de operações de entrada e saída, através do IOPS.

> Para verificar os custos do RDS, acesse https://aws.amazon.com/rds/pricing/.

Escolha a opção **Dev/Test** para que não tenha custos durante os estudos, e clique em *Next Step*.

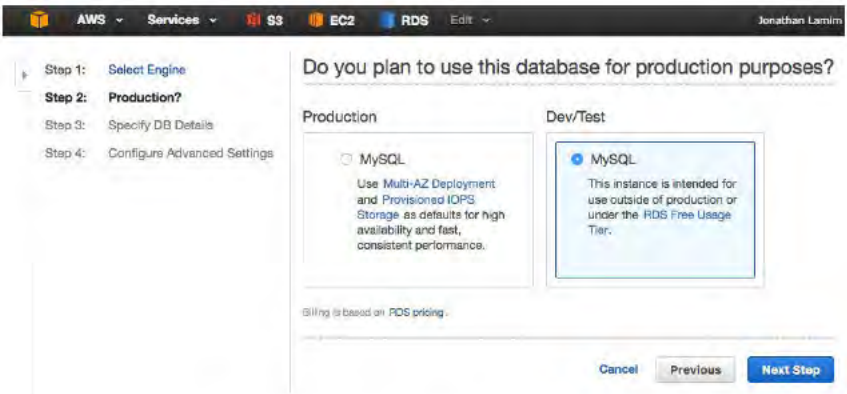


Figura 7.4: Selecionando o plano do banco de dados

Agora é hora de configurar os detalhes do banco de dados MySQL. Nesse momento, você vai definir as seguintes informações:

- **DB Engine Version:** é a versão do banco de dados que você vai utilizar. Para esse exemplo, mantenha a versão que já vier selecionada.
- **DB Instance Class:** é o tipo da instância (micro, small,

medium, large, xlarge) que será criada. Selecione a instância `db.t2.micro` para se manter no plano de gratuidade.

- **Multi-AZ Deployment:** é a criação de réplicas do banco de dados em zonas diferentes. Selecione `No` para se manter no plano de gratuidade.
- **Storage Type:** é o tipo de armazenamento que será usado. Selecione `Magnetic` para se manter no plano de gratuidade.
- **Alocated Storage:** é o espaço em disco destinado para a instância. Mantenha o valor mínimo já informado, que é de 5GB.
- **DB Instance Identifier:** é a identificação única da instância. Use `livroaws-wp` .
- **Master Username:** é o usuário master que acessará a instância. Use `wordpress_user` .
- **Master Password:** é a senha do usuário master. Escolha uma senha segura e repita no campo seguinte, **Confirm Password**.

Clique em *Next Step* para prosseguir com as configurações do banco de dados.

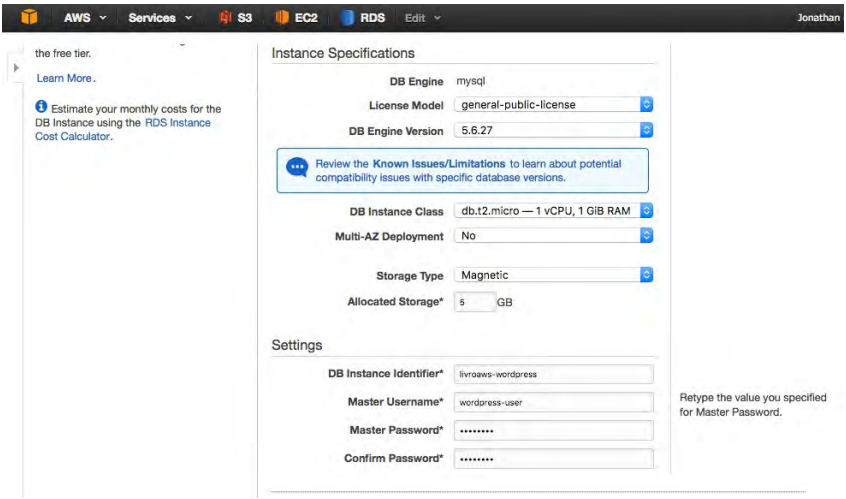


Figura 7.5: Detalhes do banco de dados

Para as próximas informações a serem configuradas, mantenha os valores padrões para *Network & Security*, exceto para a opção **VPC Security Group(s)**, em que você deverá selecionar a opção `LivroAWSCap5`, que é o grupo de configurações de acesso que foi definido no capítulo 5.

Ao definir o *Security Group* como `LivroAWSCap5`, você permitirá que todas instâncias EC2 pertencentes ao grupo tenham permissão de conexão ao RDS. Em **Database Name**, coloque `wordpress_db`, para definir o nome do banco de dados que será criado. Mantenha as demais informações conforme os padrões que já vierem preenchidos.

Para concluir, clique em *Launch DB Instance*. Aguarde enquanto a instância é criada e as configurações aplicadas.

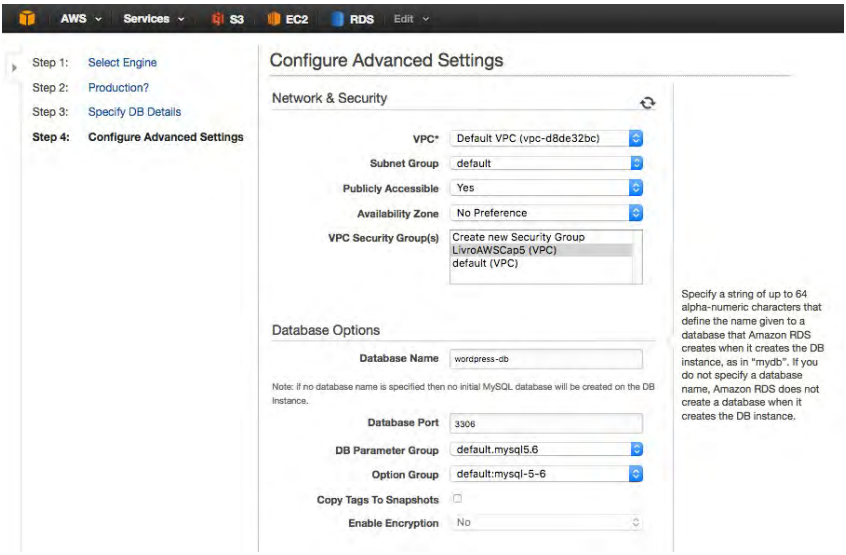


Figura 7.6: Detalhes do banco de dados

Acessando a lista de instâncias do RDS logo após concluir a criação, você perceberá que o status estará como `backing-up` e permanecerá por alguns minutos. Aguarde até que o status mude para `available`, para poder utilizá-la.

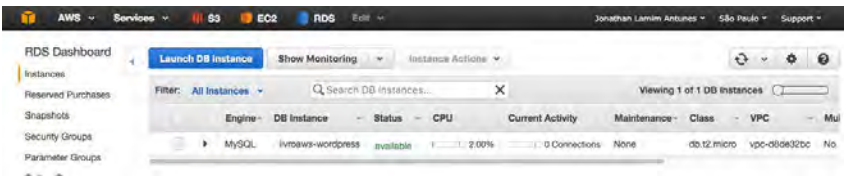


Figura 7.7: Instância pronta para uso

7.3 TESTANDO A CONEXÃO COM O BANCO DE DADOS

Para testar a conexão com o banco de dados, você pode utilizar algum SGBD que permita acesso remoto, ou então acessar através do terminal, via conexão SSH.

Acessando via terminal

Conecte-se via SSH à instância EC2 criada no capítulo 5 e execute o comando de conexão com o `mysql` a seguir:

```
mysql -h <host> -u <user> -p
```

- `host` : é o endpoint da instância que você encontra na lista de instâncias.
- `user` : é o usuário que foi criado no RDS. Nesse exemplo, é `wordpress_user` .

> Veja mais sobre comandos básicos do MySQl no **Apêndice 3**.

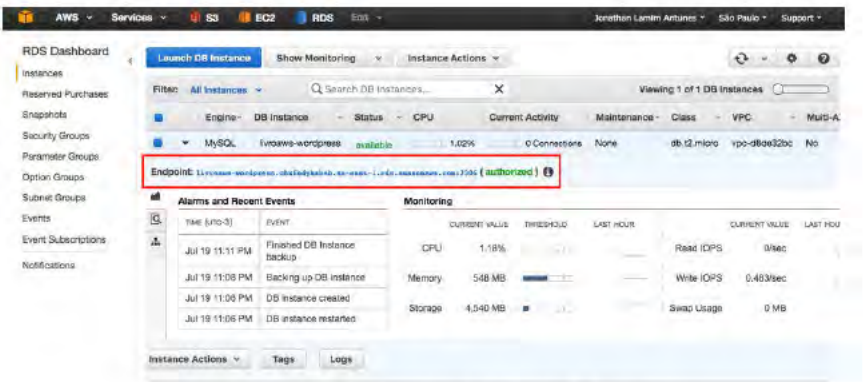


Figura 7.8: Informações da instância do RDS

Se tiver um retorno semelhante ao apresentado a seguir, é porque a conexão foi efetuada com sucesso.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 21
Server version: 5.6.27-log MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights
 reserved.
```

```
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current inpu
t statement.

mysql>
```

Caso você não consiga se conectar e tenha um erro parecido com os apresentados a seguir, verifique os seguintes pontos:

```
ERROR 2003 (HY000): Can't connect to MySQL server on <host> (110)

ERROR 2005 (HY000): Unknown MySQL server host <host> (0)
```

- Nas configurações do **Security Group**, se o acesso para o MySQL está definido corretamente, pois pode ser que o seu IP não tenha permissão de acesso.
- Se o endereço do host está correto.
- Se a instância do RDS já acabou de ser montada e está liberada para uso.

Acessando remotamente via SGBD

Você também pode acessar remotamente um banco de dados do RDS através de SGBDs, como o MySQL Workbench, o SequelPro e o RazorSQL. Para isso, você utilizará as mesmas credenciais de acesso (host, usuário e senha) do acesso via terminal.

7.4 CONECTANDO O WORDPRESS COM O RDS

No capítulo anterior, você instalou o Wordpress na instância do EC2 e fez a conexão dele com o MySQL instalado na própria instância. Agora você vai alterar essa conexão para que ele passe a

utilizar a instância de banco de dados do RDS, podendo assim fazer uso de outros recursos da AWS (que veremos nos próximos capítulos) para otimizar o funcionamento da aplicação instalada.

Como você já está conectado ao MySQL do RDS no terminal, basta executar o comando `exit;` para que se desconecte e possa executar comandos para gerenciamento de arquivos e da instância. Acesse o diretório `html` onde está a instalação do Wordpress usando o comando `cd /var/www/html` . Em seguida, abra o arquivo `wp-config.php` com o comando `nano wp-config.php` .

Edite o arquivo atualizando os dados de conexão do banco de dados. Substitua `localhost` pelo endereço do host da instância do RDS e o nome do banco, o usuário e senha pelos que você configurou no início do capítulo.

Ao concluir a instalação, saia do editor e abra o DNS público da instância do Wordpress no browser. Você verá que ele carrega a tela para que complete a instalação em vez de carregar diretamente a página principal da aplicação.

Isso ocorre porque você está usando um banco de dados vazio. Então, o próprio Wordpress considera que deve executar o restante da configuração para que funcione corretamente, criando assim a estrutura de banco de dados.

7.5 MANTENDO O RDS SEGURO

Ao criar a instância no RDS, você utilizou o mesmo **Security Group** da instância do Wordpress no EC2. Com isso, a instância tem as mesmas permissões de acesso que a instância no EC2, o que não é legal.

Se o RDS é uma instância de banco de dados, então não há a necessidade de ter a porta 80 para acesso HTTP liberada, e nem a

porta 22 para acesso via SSH. Precisamos apenas da porta 3306 liberada, que é a porta do MySQL.

Para melhorar a segurança da sua instância RDS, você vai atualizar as configurações de segurança, criando um novo grupo chamado `WP_RDS`. Ele vai permitir somente o acesso de uma instância do EC2 e o seu (a partir do seu IP) à instância do RDS.

Criando um novo Security Group

Acesse a opção *Security Groups* no menu lateral e clique em *Create Security Group* para começar a fazer essas configurações. Para o campo **Security Group Name**, informe `WP_RDS` e, para o campo **Description**, coloque `Grupo WP RDS`.

Na aba **Inboud**, clique em *Add Rule* para poder inserir a primeira regra de segurança que deve ser aplicada a esse grupo.

- **Type:** MySQL
- **Source:** Custom

Ao escolher a opção `Custom` para o campo **Type**, você deverá informar o **security group** utilizado pela instância do EC2, que é `LivroAWSCap5`. Comece digitando `sg` para que as opções se abram e você possa escolher o grupo correto.

Fazendo isso, somente as instâncias do EC2 que estiverem no grupo `LivroAWSCap5` poderão acessar a instância do RDS que estiver associada a esse grupo que está sendo criado, e você não terá mais o acesso remoto.

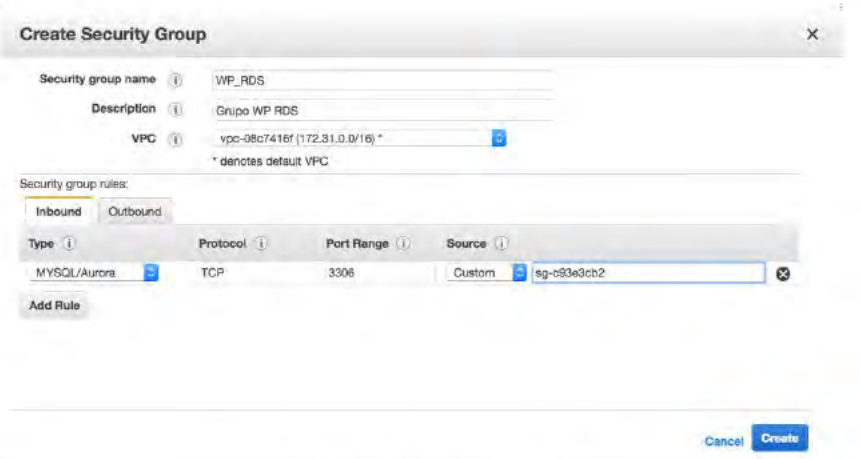


Figura 7.9: Security Group — WP_RDS

Para liberar o acesso remoto à instância do RDS nesse novo grupo, crie uma nova regra permitindo o acesso para o seu IP.

- **Type:** MySQL
- **Source:** My IP

Preenchidos os campos, clique em *Create*.

Atualizando o *Security Group* na instância do RDS

Agora que o novo *Security Group* já foi criado, você deve voltar ao RDS para atualizar a instância. Na lista de instâncias do RDS, selecione a instância criada, clique em *Instance Actions* e escolha a opção *Modify*.

No grupo de configuração *Network & Security*, altere o *Security Group* para `WP_RDS` . No final da tela, selecione a opção *Apply Immediately* para aplicar as configurações imediatamente. Em seguida, clique em *Continue*, e depois em *Modify DB Instance* para executar a mudança.

O processo de alteração pode demorar um pouco para ser executado. Nos detalhes da instância, é possível ver o status da execução.

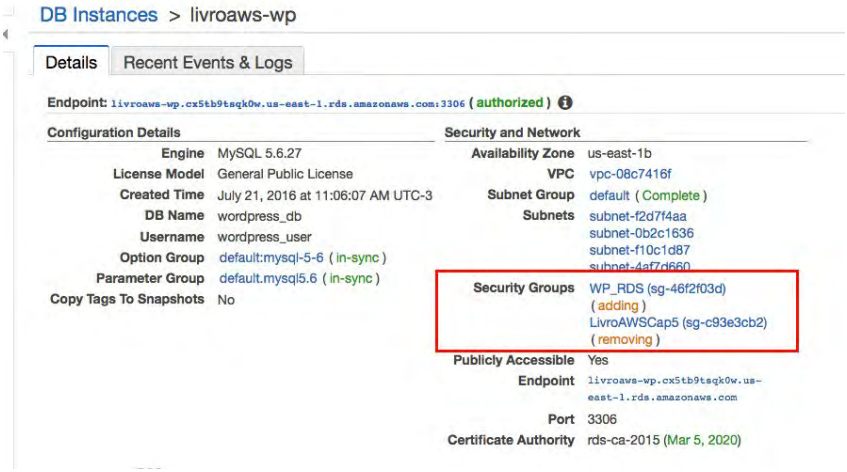


Figura 7.10: Alteração de Security Group na instância do RDS

7.6 FAZENDO BACKUPS DO BANCO DE DADOS

Ao criar uma instância no RDS, você tem a possibilidade de configurar o backup automático, que é realizado diariamente. Você pode definir o período de retenção e também a janela de backup, que é a hora em que ele deve ser executado e quanto tempo essa janela durará.

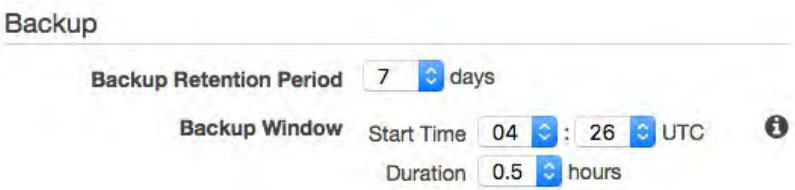


Figura 7.11: Backup automático da instância do RDS

Você pode fazer também esse backup de forma manual, que é conhecido no RDS como **snapshot**. Para realizar esses *snapshots*, você deve selecionar a instância, clicar em *Instance Actions*, e em seguida escolher a opção *Take Snapshot*.

Informe o nome para identificar o *snapshot*, e clique em *Take Snapshot* para executá-lo.

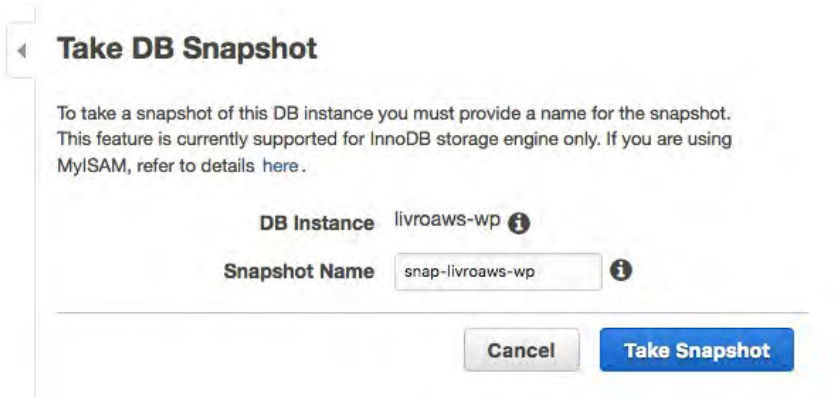


Figura 7.12: Fazendo um snapshot da instância do RDS

O processo pode demorar um pouco, e você pode acompanhá-lo na lista de *snapshots* que pode ser acessada pelo menu lateral na opção *Snapshots*.

7.7 RESTAURANDO UM BACKUP DO BANCO DE DADOS

Para restaurar um backup, você deve selecionar a instância e clicar em *Instance Actions* e, em seguida, escolher a opção *Restore to Point in Time*.

Então, você deverá selecionar o ponto de restauração, preencher as informações seguintes do mesmo modo como se estivesse criando uma nova instância, e clicar em *Launch DB Instance*.

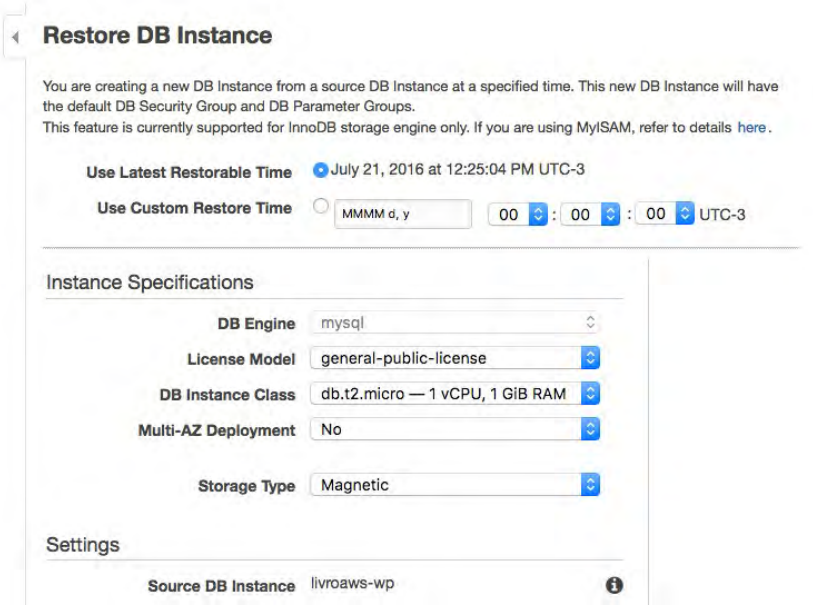


Figura 7.13: Restaurando um backup da instância do RDS

Restaurando um snapshot

Para restaurar um snapshot, você deve selecionar o snapshot e clicar em *Restore Snapshot*. Então, você deverá selecionar o ponto de restauração, preencher as informações seguintes do mesmo modo como se estivesse criando uma nova instância, e clicar em *Restore DB Instance*.

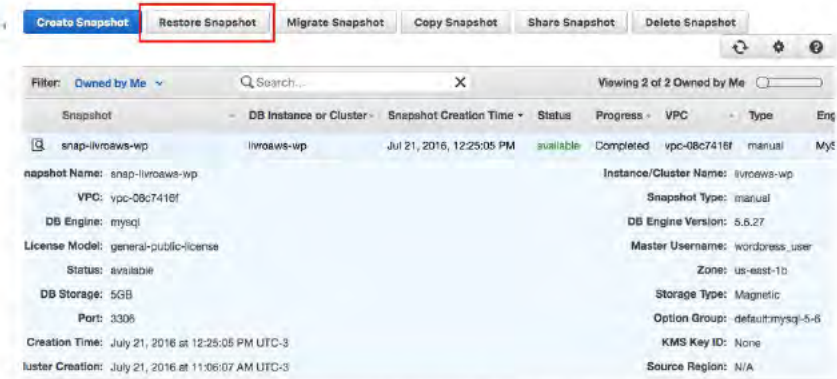


Figura 7.14: Restaurando um snapshot da instância do RDS

7.8 CONSIDERAÇÕES FINAIS

Neste capítulo, você aprendeu a trabalhar com o RDS, configurá-lo de maneira segura e permitindo acesso remoto. Além disso, você conectou o Wordpress instalado na instância do EC2 com o banco de dados da instância do RDS.

Você pode usar o RDS para armazenar o banco de dados de diversas aplicações, e essas não precisam necessariamente estar no EC2, podem estar fora do ambiente da AWS. Vários serviços da AWS trabalham de forma independente, o que permite o uso para diversos tipos de aplicações, como por exemplo: SNS, CodeDeploy, DinamoDB, S3, entre outros.

Nos próximos capítulos, você verá outros serviços e também algumas integrações, entre esses serviços.

Links úteis

- **Documentação do RDS:** https://aws.amazon.com/documentation/rds/
- **Plano de gratuidade do RDS:** https://aws.amazon.com/rds/free/

- **Custos do RDS:** https://aws.amazon.com/rds/pricing/

Capítulo 8

AMAZON ELASTICACHE

8.1 INTRODUÇÃO

O ElastiCache é um serviço da AWS para armazenamento de cache de memória na nuvem e que possui suporte para as bibliotecas de cache Redis e Memcached, permitindo um alto desempenho de gravação e leitura dos dados armazenados. Ele também está enquadrado no programa de gratuidade da Amazon, o que vai permitir utilizá-lo durante os estudos do livro sem nenhuma despesa.

Veja a seguir o que a documentação oficial diz sobre o uso do ElastiCache:

"O Amazon ElastiCache pode ser usado para aumentar bastante a latência e o processamento de cargas de trabalho de aplicativos que exigem muita leitura (como redes sociais, jogos, compartilhamento de mídia e portais de FAQ) ou cargas de trabalho com grande quantidade de cálculos (como mecanismo de recomendação). O cache melhora o desempenho dos aplicativos armazenando partes críticas de dados na memória para oferecer acesso de baixa latência. As informações em cache podem incluir os resultados de consultas a banco de dados com intensa E/S ou os resultados de cálculos computacionalmente intensivos."

Imagine um projeto de um portal de conteúdo, que possui uma publicação média de 20 postagens diárias, e uma média de 10 mil

acessos diários e 300 acessos simultâneos. Sem o uso do cache, o volume de entrada e saída de dados será muito alto, o que necessitará de um servidor com alto desempenho, aumentando assim o custo.

Ao utilizar o cache, você faz com que todo esse conteúdo acessado não precise ser processado por completo a cada acesso, reduzindo assim o volume de operações de entrada e saída no banco de dados, e também pode trabalhar com um servidor menos robusto, reduzindo o custo.

O uso de cache traz diversos benefícios para a aplicação, como por exemplo:

- Redução de operações de entrada e saída no banco de dados;
- Redução de custo por possibilitar operar com servidor menos robusto;
- Melhoria no desempenho da aplicação.

Se sua aplicação faz uso de informações que não sofrem alteração frequentemente, mas possui um volume alto de acessos, o uso do cache vai ser muito importante para que a aplicação possa funcionar bem, sem travamentos ou demora para carregar as informações. É para isso que temos o ElastiCache.

8.2 CRIANDO UM SECURITY GROUP PARA O ELASTICACHE

Antes de criar um cluster para o armazenamento do cache, é importante criar um *Security Group* específico, para que estejam liberadas somente as portas necessárias.

Nesse caso, crie um *Security Group* para o cluster de cache, conforme já foi feito anteriormente, com as seguintes informações:

- **Security Group Name:** WP_ElastiCache
- **Description:** Cache para o Wordpress

Regras de seguranças (Inbound)

- **Type:** Custom TCP Rule
- **Port Range:** 11211
- **Source:** Custom (aponte para o security group `LivroAWSCap5`)

Esta regra vai permitir que somente instâncias do EC2 se conectem com o cluster do ElasticCache associado a esse *Security Group*.

> As regras INBOUND são regras para entrada, ou seja, o que é configurado nela serve para filtrar as conexões de entrada.

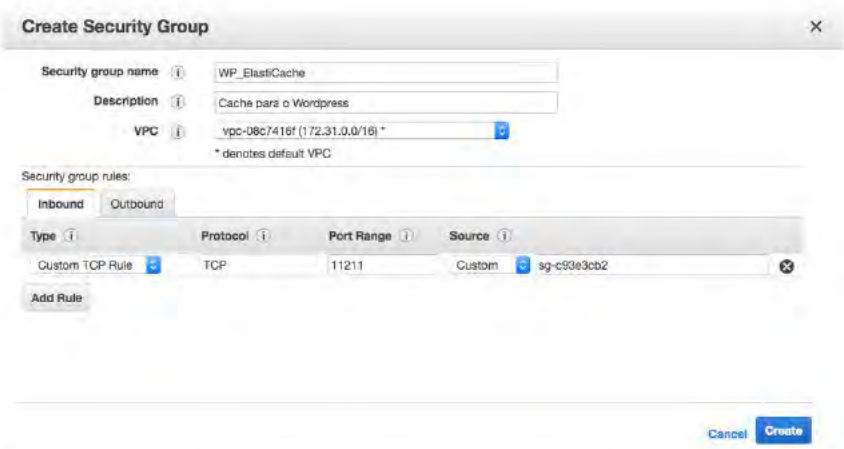


Figura 8.1: ElastiCache Security Group

8.3 CRIANDO O CLUSTER DE CACHE

Agora que o *Security Group* já está criado, é hora de criar o cluster para começar a armazenar o cache. Acesse o menu superior *Services* e, em seguida, escolha a opção *ElastiCache*. Depois, clique em *Get Started Now* para iniciar o processo de criação.

Figura 8.2: ElastiCache Dashboard

O processo consiste em quatro etapas e, na primeira, você deve escolher qual a engine vai utilizar. Como já foi dito, você tem como opções o Memcached e o Redis. Selecione o Memcached e clique em *Next*.

O passo seguinte é a definição de detalhes do cluster. Altere somente as informações para **Cluster Name**, **Node Type** e **Number of Nodes**.

- **Cluster Name:** WP-Cache-Cluster
- **Node Type:** cache.t2.micro (essa escolha vai mantê-lo no plano de gratuidade)
- **Number of Nodes:** 1

Para prosseguir, clique em *Advanced Memcached settings*. Selecione então o **Security Group**, que foi criado para a uso no

ElastiCache, chamado *WP_ElastiCache*. Para finalizar, clique em *Create*.

Figura 8.3: ElastiCache - Criando o cluster

Vai levar alguns minutos para que o cluster seja liberado. Enquanto isso, seu status será *creating*. Quando ele estiver pronto para uso, o status mudará para *available*.

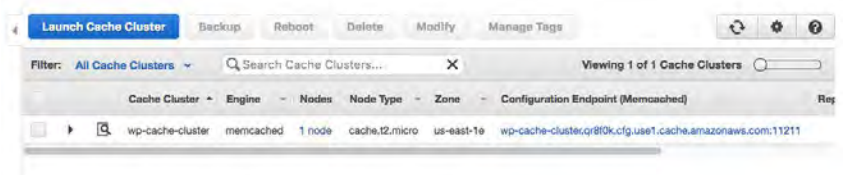


Figura 8.4: ElastiCache Clusters

8.4 TESTANDO O FUNCIONAMENTO DO CLUSTER DE CACHE

A maneira mais simples e rápida de você testar o funcionamento do cache é através do terminal, usando *telnet*. Estando conectado à instância do EC2 via SSH, instale o telnet com o comando `sudo yum install telnet` .

Após instalar, acesse o cluster via telnet com o comando

`telnet <endpoint_do_cluster> 11211` . Após executar o comando, deverá ver uma mensagem semelhante a esta no terminal:

```
Trying 172.31.41.134...
Connected to wp-cache-cluster.qr8f0k.cfg.use1.cache.amazonaws.com.
Escape character is '^]'.
```

Para testar, execute o comando `set teste 0 0 8` para poder gravar um texto com a chave teste, que não expira e com um tamanho de 8 bytes. Em seguida, digite `livroaws` e pressione a tecla *Enter*. O retorno do comando será `STORED` .

Para verificar se `livroaws` foi gravado na chave teste, digite `get teste` e pressione *Enter*. O retorno deverá ser `livroaws END` .

Se esse foi o retorno, então é sinal de que o cluster está funcionando. Para sair do telnet, digite `quit` e pressione *Enter*.

8.5 EXCLUINDO UM CACHE CLUSTER

Excluir um cache cluster é bastante simples. Basta ir até a lista de clusters através do menu lateral **Cache Clusters**, selecionar o cluster que deseja excluir e clicar em *Delete*. Será iniciado então o processo de remoção e, em alguns minutos, estará concluído.

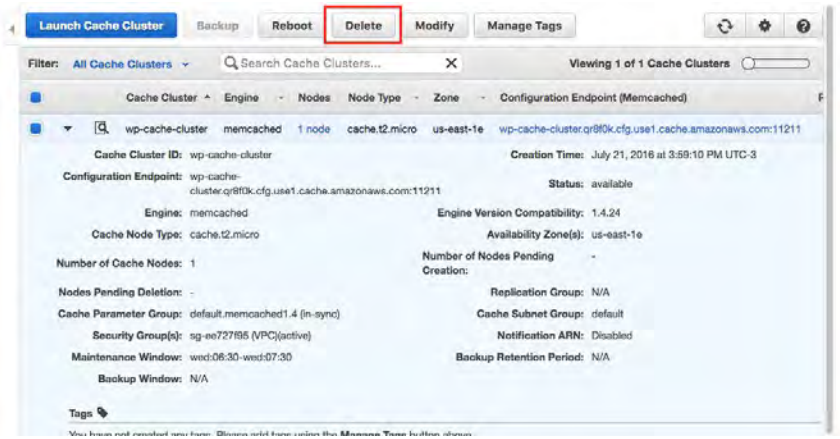


Figura 8.5: Excluindo um cluster

8.6 INSTALANDO O MEMCACHED NO EC2

O cluster que acabamos de criar vai utilizar o Memcached. Então, será necessário fazer sua instalação na instância do EC2 que está rodando o Wordpress.

Acesse a instância via SSH pelo terminal, como já foi feito outras vezes nos capítulos anteriores, e execute o comando a seguir para executar a instalação:

```
sudo yum install php-memcached
```

Após executar o comando, é necessário reiniciar o Apache. Então execute o comando `sudo service httpd restart` .

Feito isso, o Memcached já estará funcionando no servidor e você poderá iniciar o processo de configuração do Wordpress para utilizá-lo.

8.7 INTEGRANDO O WORDPRESS COM O ELASTICACHE

Para integrar o Wordpress com o ElastiCache, você pode usar o plugin **W3 Total Cache (W3TC)**. Faça a instalação e ativação desse plugin no Wordpress e, em seguida, execute o processo de configuração conforme descrito a seguir.

Configurando os endpoints do cluster do ElastiCache no W3TC

Após a criação do cluster no ElastiCache, você terá disponível um endpoint que será usado na configuração do plugin.

Esse endpoint é encontrado na lista de cluster do ElastiCache, conforme mostra a figura a seguir:

Figura 8.6: ElastiCache Clusters

Copie o endpoint, incluindo a porta que aparece no final do endereço e volte para as configurações do W3TC no Wordpress.

Vá até o menu lateral e escolha a opção *Performance* e, em seguida, *General Settings*. Localize o bloco *Page Cache*, ative o cache e defina o método de cache (Page Cache Method) como **Memcached** e, depois, salve as configurações.

Volte ao menu lateral e escolha a opção *Page Cache*. Na página de configuração do cache de páginas, você poderá configurar a forma como deseja que o cache seja utilizado. Mas não vamos entrar em detalhes sobre essas configurações, vamos focar na configuração da conexão do W3TC com o ElastiCache.

Ainda na página de configuração, localize o bloco de

configurações **Advanced**. Nele você encontrará um campo chamado **Memcached hostname:port / IP:port**. Cole nesse campo o endpoint do cluster do ElastiCache.

Após colar, clique no botão testar para verificar se está tudo correto com a comunicação.

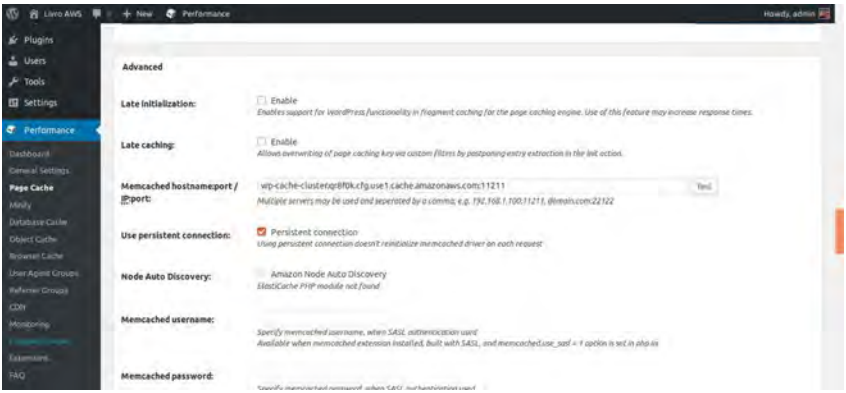


Figura 8.7: ElastiCache Clusters

Para confirmar as configurações, clique em *Save all settings*.

Você poderá utilizar o ElastiCache para vários configurações de cache no Wordpress, para minificação de arquivos, entre outros, tudo de forma simples e semelhante à que foi configurada para o cache de páginas.

8.8 CONSIDERAÇÕES FINAIS

Viu como é fácil criar um clustar para fazer o cache das aplicações e ainda integrar o Wordpress para poder fazer uso desse recurso? Com o uso do cache, você pode melhorar muito a performance das suas aplicações e ainda reduzir seus custos mensais na AWS.

Lembre-se sempre dos benefícios que foram apresentados no

início do capítulo em relação ao uso de cache:

- Redução de operações de entrada e saída no banco de dados;
- Redução de custo por possibilitar operar com servidor menos robusto;
- Melhoria no desempenho da aplicação.

Links úteis

- **Documentação:** http://aws.amazon.com/pt/elasticache
- **Documentação para desenvolvedores:** http://aws.amazon.com/pt/documentation/elasticache
- **Redis:** http://redis.io
- **Memcached:** http://www.memcached.org

Capítulo 9

AMAZON ROUTE 53

9.1 INTRODUÇÃO

"O Amazon Route 53 é um serviço web de Domain Name System (DNS) altamente disponível e escalável". Ele foi projetado para ser um serviço de confiança, com um baixo custo e de fácil configuração por parte dos desenvolvedores. Com ele, você pode fazer o roteamento dos usuários para a sua aplicação, traduzindo o nome do domínio para o endereço IP de uma instância do EC2, ou mesmo do S3.

Além de permitir o roteamento dos usuários para a aplicação, ele ainda permite também o registro de domínios `.com` . E ao adquirir domínios pelo Route 53, a configuração de DNS deles é feita automaticamente.

No decorrer deste capítulo, veremos como fazer a configuração do Route 53 para um domínio, que é o recurso mais utilizado.

Importante

O Route 53 não faz parte do plano de gratuidade da Amazon, portanto cada *Hosted Zone* criada no Route 53 tem um custo de US$ 0,50 mensais. Não é muito, mas caso você não queira ter custos enquanto aprende a usar os recursos da AWS, pode utilizar serviços de DNS como o do Registro.br, por exemplo, que são gratuitos.

9.2 CRIANDO UM ROTEAMENTO NO ROUTE 53

Para criar um roteamento no Route 53, você deverá acessar o painel da AWS. Em seguida, acesse o serviço do Route 53 no menu *Services*.

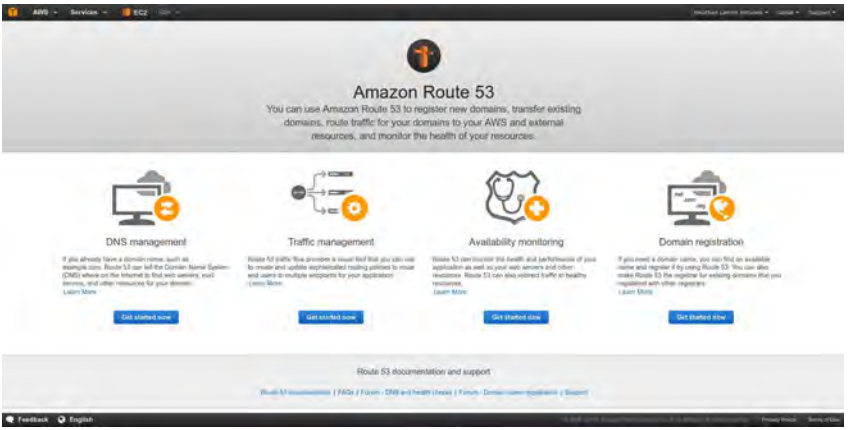


Figura 9.1: Tela inicial do Route 53

Na tela principal, clique em *Get started now* para ser direcionado à tela de criação das *Hosted Zones*.

O que são as Hosted Zones?

Hosted Zones são nada mais nada menos do que as zonas de roteamento de um domínio. Para cada domínio que você queira rotear com o Route 53, será necessário criar uma nova *Hosted Zone*.

Criando uma Hosted Zone

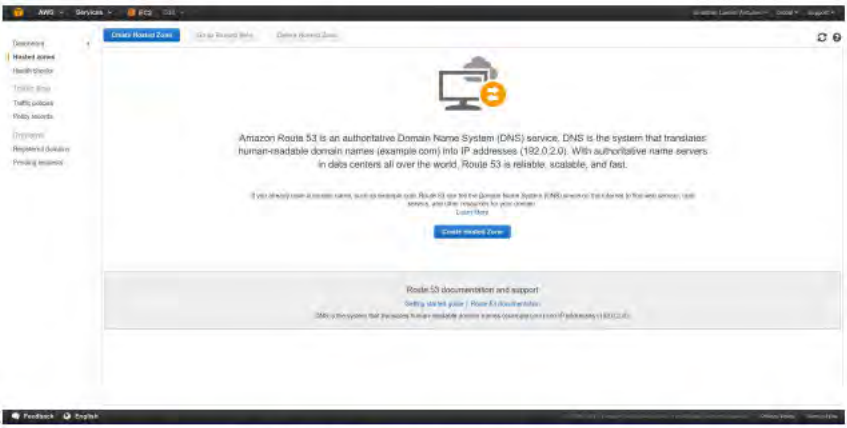


Figura 9.2: Tela inicial para iniciar a criação de uma Hosted Zone

Na tela inicial das *Hosted Zones*, clique em *Create Hosted Zone* para iniciar o processo de criação e configuração.

Em seguida, você será direcionado para a tela na qual deverá preencher as seguintes informações:

- **Domain Name:** nome do domínio (example.com).
- **Comment:** um comentário sobre a hosted zone, mas seu preenchimento não é obrigatório.
- **Type:** é o tipo de roteamento que será feito. Utilize *Public Hosted Zone* para que o domínio seja configurado como público e permita o acesso via browser aos usuários.

Em seguida, basta clicar em *Create* para finalizar a primeira etapa do processo de criação da *Hosted Zone*.

Figura 9.3: Tela de lançamento dos dados da Hosted Zone

Ao concluir a criação da *Hosted Zone*, serão exibidos os DNSs do domínio para que você possa configurá-los no Registro.br, ou em outro serviço onde tenha adquirido o domínio.

Figura 9.4: Informações de DNS

A lista de DNSs possui 4 endereços, e geralmente são necessários apenas 2 deles para configurar o domínio. Você pode escolher qualquer um destes, mas vou dar uma dica interessante:

> Antes de definir quais os endereços utilizar, abra o terminal e execute um ping para cada um dos endereços de DNS do domínio. Veja quais possuem o menor tempo de resposta e utilize-os.

Configurando o Record Set

O Record Set é a representação da configuração que você vai fazer. É por meio dele que você pode rotear subdomínios e, até mesmo, o acesso com ou sem `www` para o seu domínio.

Certifique-se de estar na tela com a lista de DNSs da sua *Hosted Zone* e, então, clique em *Create Record Set* para iniciar a criação de um Record Set para o domínio.

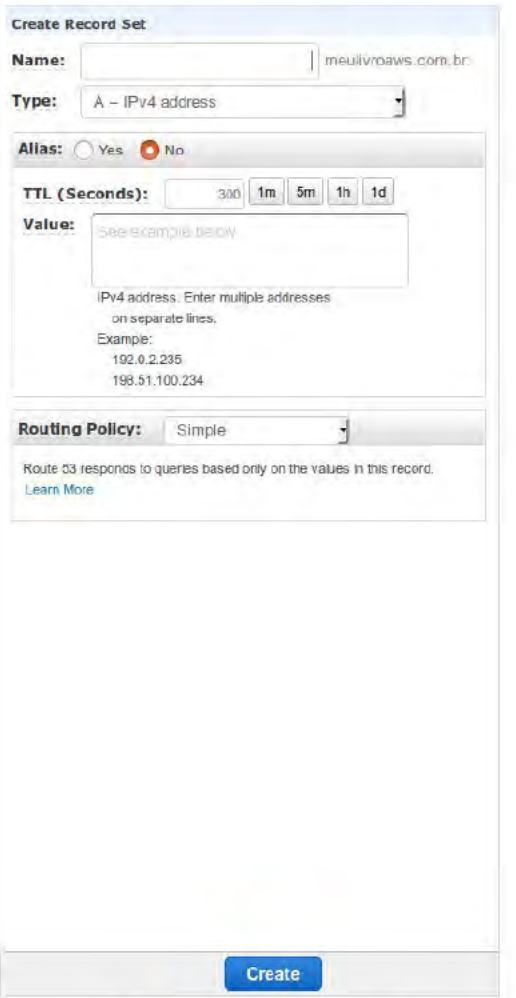


Figura 9.5: Criando um Record Set

No formulário que será aberto, preencha os campos conforme instruções a seguir:

- **Name:** pode ser usado tanto para subdomínio quanto para informar o uso do `www`. Nesse caso, utilize `www`.
- **Type:** é o tipo de Record Set que deve ser criado. Nesse caso, use o tipo A.

- **Value:** é o IP da instância que deverá responder ao roteamento.

Clique em *Create* para salvar as informações do novo Record Set.

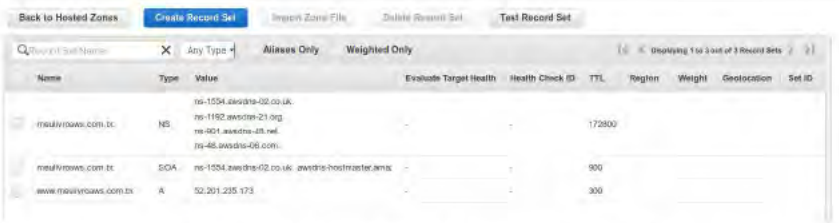


Figura 9.6: Informações de DNS com o novo Record Set

Agora vamos criar um outro Record Set para o acesso sem o `www` . Repita os passos anteriores, deixando apenas o campo **Name** sem ser preenchido.

Após concluir a criação do Record Set, sua lista de Record Sets deve ser semelhante à da figura a seguir:

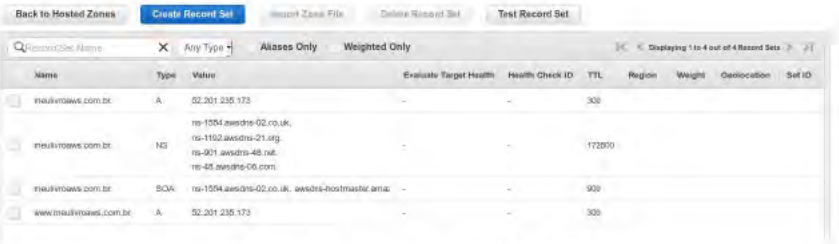


Figura 9.7: Informações finais da Hosted Zone

9.3 APONTANDO UM BUCKET DO S3 EM UMA HOSTED ZONE

Nos capítulos 2 e 3, falamos sobre o S3, que é o serviço para armazenamento de arquivos estáticos. É possível adicionar buckets do S3 em uma *Hosted Zone*, fazendo com que o acesso ao bucket

possa acontecer a partir de um subdomínio.

Se você já tem um bucket criado, ótimo, pode seguir neste capítulo. Entretanto, se não criou o bucket no capítulo 2, então faça uma pausa para criá-lo e volte em seguida para continuar a configuração dele na *Hosted Zone*.

> **IMPORTANTE**
>
> Para fazer a conexão entre o bucket do S3 e um Record Set, é preciso que o bucket tenha o mesmo nome do site.

Configurando o Record Set para o bucket do S3

Acesse a *Hosted Zone* do domínio criado e clique em *Create Record Set* para iniciar o processo de configuração. No campo **Name**, você deverá informar o subdomínio que desejar. É muito comum nesses casos usar o subdomínio `s3` ou `static` . Vamos utilizar `s3` para esse exemplo. Para o campo **Type**, você deve escolher o tipo *A*.

Agora vamos utilizar outros campos que não foram vistos nos primeiros Record Sets configurados. No campo **Alias**, marque a opção *Yes* e informe o endpoint do bucket criado no S3. Esse campo é o responsável por habilitar a ligação entre os serviços.

Feito isso, é só clicar em *Create* para concluir o processo.

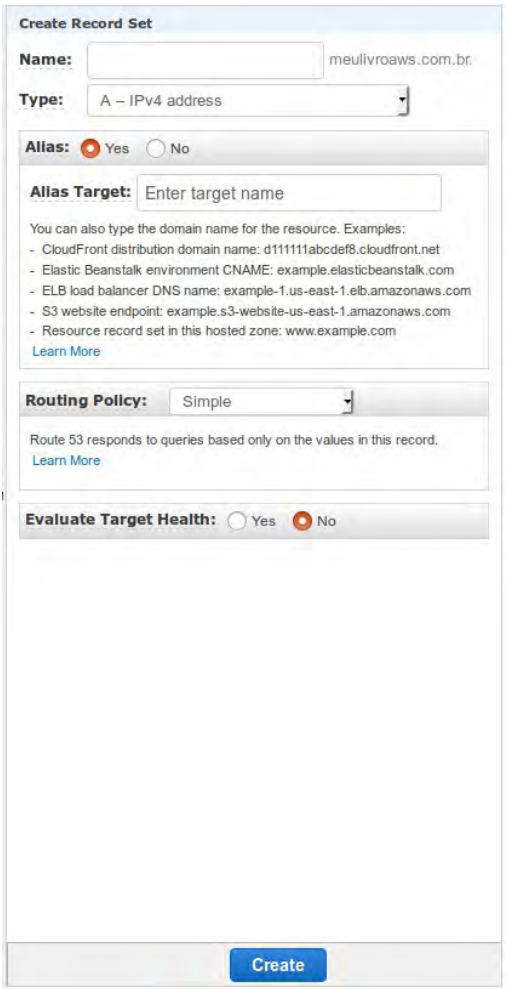


Figura 9.8: Formulário para criação do Record Set para ligação com o Bucket no S3

9.4 CONSIDERAÇÕES FINAIS

Neste capítulo, você aprendeu a fazer o roteamento de domínios usando o Route 53. De maneira simples e rápida, você configurou o roteamento de um domínio e obteve os endereços de DNS para configurar no painel do serviço onde adquiriu o domínio.

Links úteis

- **Documentação do Route 53:** https://aws.amazon.com/pt/route53/
- **Documentação para desenvolvedores:** https://aws.amazon.com/pt/documentation/route53/

Capítulo 10

AMAZON CLOUDFRONT

Figura 10.1: Tela inicial do Route 53

10.1 INTRODUÇÃO

Em linhas gerais, o CloudFront é o serviço de CDN (Content Delivery Network, ou Rede de Distribuição de Conteúdo) da Amazon. Ele tem a flexibilidade para se integrar com outros serviços da AWS, oferecendo aos desenvolvedores e empresas uma entrega de conteúdo mais rápida e com grande facilidade de uso.

Duas das características mais relevantes do CloudFront são a baixa latência e a alta velocidade utilizada no processo de transferência dos arquivos requisitados. Na sequência deste capítulo, você verá como integrar o CloudFront com um bucket do S3 para provisão de arquivos estáticos, e também como integrá-lo com o Route 53 para ter uma URL amigável para provisão dos arquivos.

10.2 INTEGRANDO UM BUCKET DO S3 COM O

CLOUDFRONT

Acesse o painel do S3 e crie um bucket (conforme foi feito no capítulo 2) e chame-o de `meulivroaws` . Esse bucket deverá ser configurado para receber um site estático.

Após criar o bucket, é hora de integrá-lo ao CloudFront. Para isso, acesse o painel do CloudFront.

Figura 10.2: Tela inicial do CloudFront

> **IMPORTANTE**
>
> O CloudFront não faz parte do programa de gratuidade, logo, todos os exemplos relacionados ao CloudFront descritos neste livro gerarão custos. Mas fique tranquilo, pois esses custos são mínimos.

Clique no botão *Create Distribution* para iniciar o processo de configuração do CloudFront.

Figura 10.3: Passo 1 da configuração do CloudFront

Clique em *Get Started* na opção **Web**.

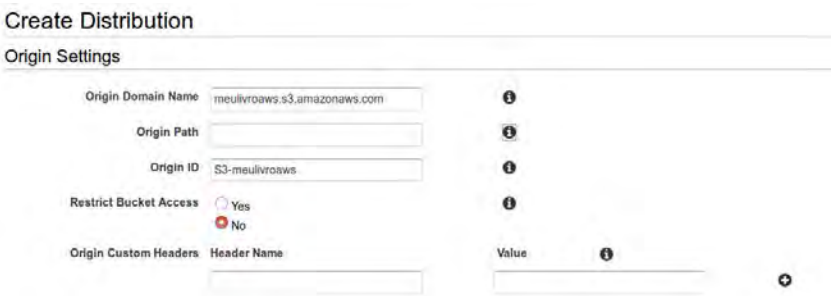


Figura 10.4: Passo 2 da configuração do CloudFront

Em seguida, é hora de começar a preencher as configurações mais específicas do CloudFront. Vamos começar por *Origin Settings*, preenchendo o campo **Origin Domain Name** com o nome do bucket que foi criado e mantendo os demais campos com os valores padrões.

Na sequência, vamos para as configurações de *Default Cache Behavior Settings*. Não altere essas configurações, mantenha o padrão.

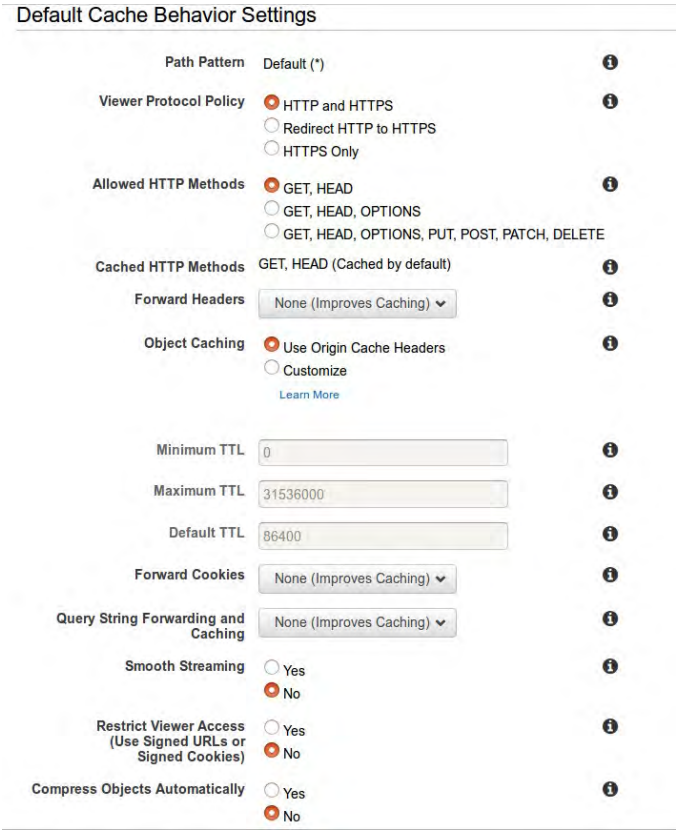


Figura 10.5: Passo 2 da configuração do CloudFront

Em seguida, para *Default Settings*, informe em **CNAMEs** o subdomínio *static.meulivroaws.com.br* e mantenha as demais configurações com os valores padrões.

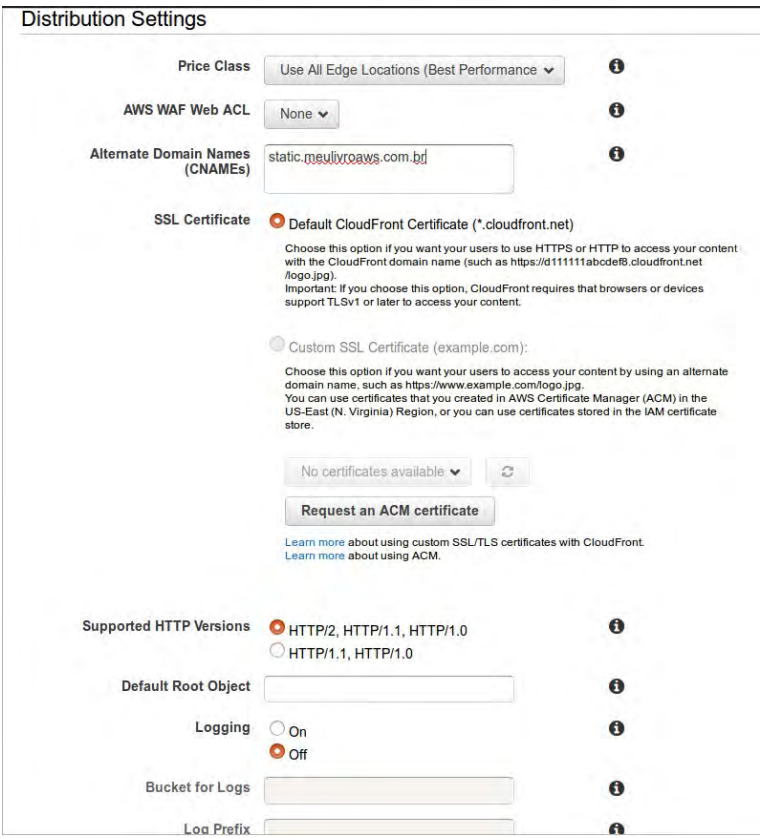


Figura 10.6: Passo 2 da Configuração do CloudFront

Feito isso, basta clicar em *Create Distribution* para concluir o processo.

Figura 10.7: Lista de distribuições do CloudFront

O processo de conclusão da criação da distribuição pode demorar um pouco devido à necessidade de replicar as configurações para os servidores.

10.3 INTEGRANDO O CLOUDFRONT COM O ROUTE 53

Agora que já criamos a distribuição no CloudFront, é hora de configurar uma URL amigável para fazer a provisão de conteúdo, e faremos isso no Route 53. Volte ao Route 53 e acesse a *Hosted Zone* que criamos no capítulo anterior para que possamos criar um novo Record Set para ela.

Clique em *Create Record Set* para iniciar o processo de configuração do Record Set para a distribuição do CloudFront. Para o campo **Name**, informe *static*. Mantenha o campo **Type** como *A*, e **Alias** selecione *Yes*. No **Alias Target**, escolha a distribuição do CloudFront que acabou de ser criada.

Salve o novo Record Set clicando em *Create*.

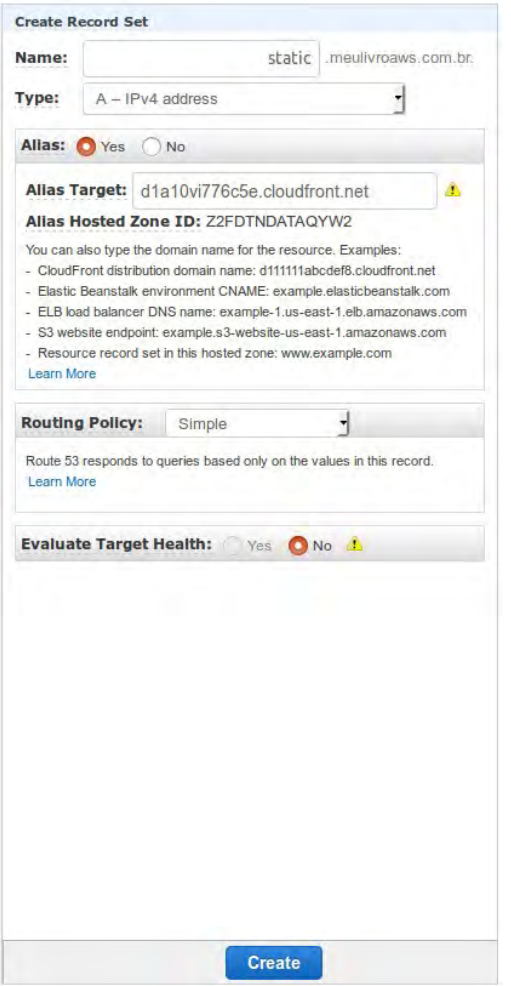


Figura 10.8: Criando o Record Set

A configuração está pronta e basta aguardar a propagação do DNS para que ela esteja acessível.

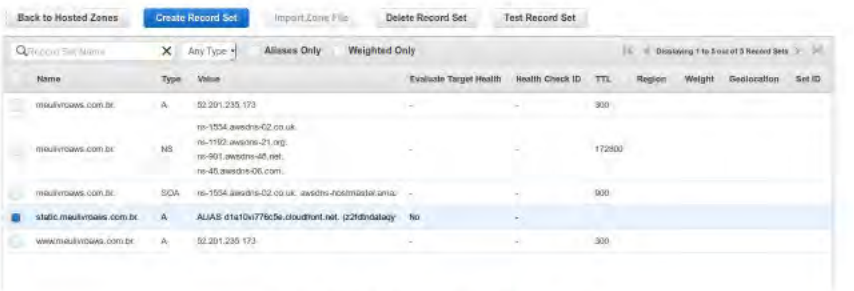


Figura 10.9: Tela inicial do CloudFront

10.4 CONSIDERAÇÕES FINAIS

O CloudFront funciona como uma CDN particular para o seu domínio. Com ele, você pode combinar os serviços (como fizemos neste capítulo) para prover conteúdo através de URLs amigáveis, fazendo uso de cache e de toda a velocidade de entrega que o CloudFront possui.

Os custos do CloudFront são baixos, e a cobrança é feita baseada no volume de dados que foram requisitados e transferidos para fora dos pontos de presença do CloudFront. Se você controlar bem o cache dos arquivos no S3, vai conseguir otimizar bastante os custos com CloudFront.

Links úteis

- **Documentação:** https://aws.amazon.com/pt/cloudfront/
- **Documentação para desenvolvedores:** https://aws.amazon.com/pt/documentation/cloudfront/

Capítulo 11

AMAZON SES (SIMPLE EMAIL SERVICE)

11.1 INTRODUÇÃO

O Amazon SES é um serviço para envio de e-mails utilizando o protocolo STMP, que pode ser escalado conforme a necessidade da aplicação. Ele tem um custo pequeno se comparado a outros serviços do mesmo tipo.

Com o SES, você pode enviar até 2 mil e-mails diariamente dentro de uma instância do EC2. Se sua aplicação precisar enviar uma quantidade maior do que essa, você paga o adicional, sendo este no valor de US$ 0,10 por cada mil e-mails excedentes que forem enviados. Ou seja, com US$ 1,00, você pode enviar 10 mil e-mails.

Além do custo acessível, o SES é muito fácil de ser configurado e usado, não sendo necessários conhecimentos muito avançados de programação. Neste capítulo, vamos ver como configurar e disparar e-mails usando o SES.

11.2 ACESSANDO O SES ATRAVÉS DO CONSOLE DE GERENCIAMENTO

Acesse o console de gerenciamento da sua conta na Amazon AWS e no menu *Services*, e então acesse a opção *SES*.

Se você ainda não estiver utilizando o SES, a tela exibida será semelhante à da figura a seguir:

Figura 11.1: Tela inicial do SES

Do lado esquerdo, você tem acesso ao menu do SES e, do lado direito, links para documentação e configurações.

Opções do menu

- **Domains:** você poderá adicionar e validar domínios para o envio de e-mails.
- **Email Addesses:** você poderá adicionar e validar endereços de e-mails que serão utilizados nos disparos.
- **Sending Statistics:** você poderá visualizar as estatísticas de envio dos e-mails. Nessa tela, você terá gráficos com diversas informações sobre os e-mails disparados.
- **SMTP Settings:** você poderá criar suas credenciais (Host, usuário e senha) para o envio de e-mails usando SMTP.
- **Supression List Removal:** você poderá adicionar e-

mails à "Supression List" da Amazon, que serve para identificar endereços de e-mail que têm problema no recebimento das mensagens, gerando rejeição — para mais informação, acesse https://docs.aws.amazon.com/console/ses/suppression-list.

- **Cross-Account Notification:** você pode configurar notificações para contas que o autorizaram a enviar e-mails usando uma identidade (host, usuário e senha do SMTP) que não é de sua propriedade — para mais informações, acesse https://docs.aws.amazon.com/console/ses/sending-auth/delegate-notifications-overview.

- **Rules Sets:** você pode criar regras para o recebimento de mensagens dos domínios verificados — para mais informação, acesse https://docs.aws.amazon.com/console/ses/receiving-email.

- **IP Address Filters:** você pode criar filtros de e-mails para aceitar ou rejeitar um IP específico ou um intervalo de IPs — para mais informação, acesse https://docs.aws.amazon.com/console/ses/receiving-email/ip-address-filters.

Sandbox: o ambiente de testes do SES

O SES possui um modo **Sandbox** — que já vem habilitado por padrão — em que você é limitado a fazer envios somente para e-mails validados e de sua propriedade, e com um máximo de 200 e-mails por dia.

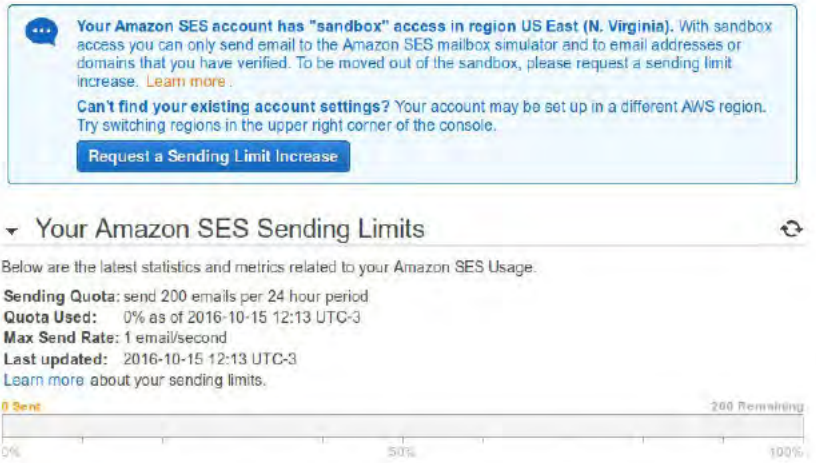


Figura 11.2: Sending Statistics

Para sair do modo **Sandbox**, você não precisa fazer nenhuma mudança nas configurações, nem no console de gerenciamento, nem em seu código. Basta solicitar o aumento do limite de envio através do formulário que pode ser acessado clicando em *Request a Sending Limit Increase*.

As limitações do SES

Como já foi visto anteriormente, o SES possui algumas limitações relacionadas a quantidade de e-mails por dia, número de disparos por segundo, entre outras. Veja a seguir algumas dessas limitações:

Mensagens

- O tamanho máximo para as mensagens é de 10MB, incluindo anexos.
- Aceita cabeçalhos de e-mail seguindo a RFC 822.
- Aceita diversos tipos de arquivos anexados, exceto os apresentados na figura a seguir:

.ade	.fxp	.mag	.msc	.prg	.url
.adp	.gadget	.mam	.msh	.reg	.vb
.app	.hlp	.maq	.msh1	.scf	.vbe
.asp	.hta	.mar	.msh2	.scr	.vbs
.bas	.inf	.mas	.mshxml	.sct	.vps
.bat	.ins	.mat	.msh1xml	.shb	.vsmacros
.cer	.isp	.mau	.msh2xml	.shs	.vss
.chm	.its	.mav	.msi	.sys	.vst
.cmd	.js	.maw	.msp	.ps1	.vsw
.com	.jse	.mda	.mst	.ps1xml	.vxd
.cpl	.ksh	.mdb	.ops	.ps2	.ws
.crt	.lib	.mde	.pcd	.ps2xml	.wsc
.csh	.lnk	.mdt	.pif	.psc1	.wsf
.der	.mad	.mdw	.plg	.psc2	.wsh
.exe	.maf	.mdz	.prf	.tmp	.xnk

Figura 11.3: Lista de extensões de arquivos não suportadas nos anexos

Envio e recebimento

- No modo **Sandbox**, os e-mails ou domínio utilizados em *From*, *Source*, *Sender* e *Return-Path* devem estar validados. Além disso, a opção *Reply-To* não está disponível.
- No modo de **Produção**, todos os itens citados anteriormente estão disponíveis e não tem necessariamente que ter os endereços de e-mail ou domínio validados.
- O número máximo de destinatários (recipients) por e-mail enviado é de 50. Considera-se destinatário: *To*,*Cc* ou *Bcc*.
- Você pode ter no máximo 1000 identidades verificadas, entre e-mail e domínio, por região do AWS.
- No modo **Sandbox**, são enviados no máximo 200 e-mail por dia, com um limite de 1 envio por segundo.

11.3 ENVIANDO E-MAILS COM O SES

Validando um endereço de e-mail

Para enviar um e-mail utilizando o SES no modo **Sandbox**, é preciso que o endereço tanto do remetente quanto do destinatário estejam validados. Sendo assim, vamos inserir 2 e-mails válidos na lista para podermos fazer os disparos.

Na tela principal do SES, vá até o menu lateral e acesse a opção *Email Addresses*. Em seguida, clique no botão *Verify a New Email Address*.

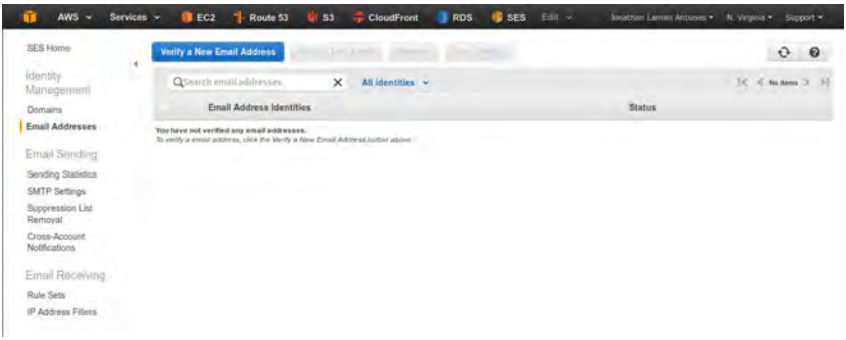


Figura 11.4: Validando um endereço de e-mail

Em seguida, informe o endereço de e-mail que deseja validar e clique em *Verify This Email Address*.

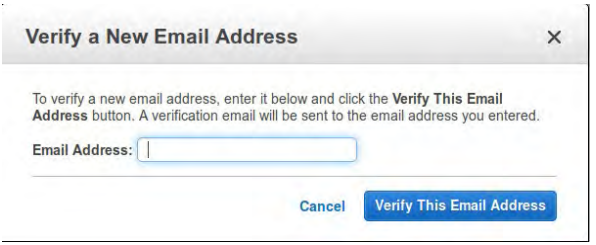


Figura 11.5: Validando um endereço de e-mail

O processo inicial de validação é rápido e, ao ser concluído, você deverá ver a seguinte mensagem na tela:

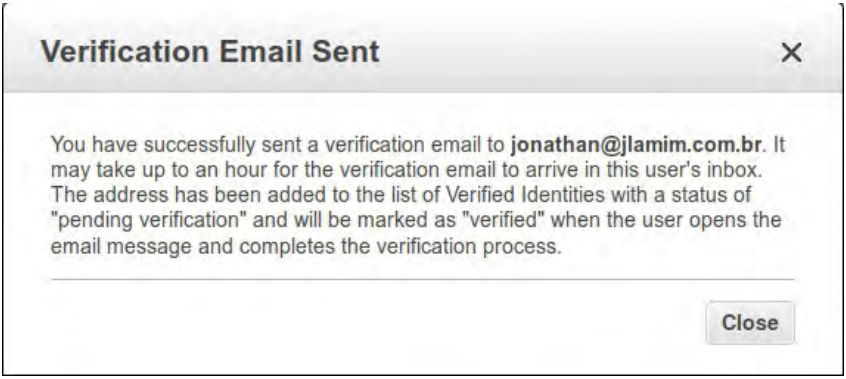


Figura 11.6: Mensagem de validação do endereço de e-mail

Ao fechar a exibição da mensagem, você verá que o e-mail para o qual você solicitou a validação já aparece na lista, mas com o status *pending verification*.

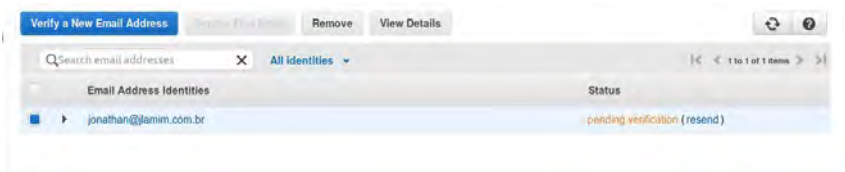


Figura 11.7: Lista de e-mails validados

Isso acontece porque você ainda precisa acessar o e-mail informado para poder confirmar que ele existe.

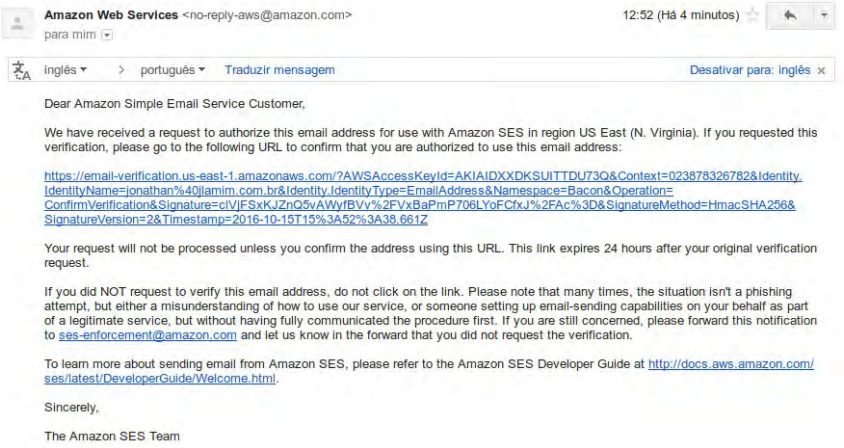


Figura 11.8: Mensagem de validação enviada por e-mail

Ao clicar no link, o processo final de validação será executado e você será redirecionado para uma tela fora do console, informando que a validação foi executada com sucesso.

Figura 11.9: Mensagem de validação executada com sucesso

Após confirmar o e-mail, o status dele na lista deverá passar de *pending verification* para *verified*. Então, você poderá fazer os primeiros envios usando o modo **Sandbox**.

Figura 11.10: Lista de e-mails validados

Enviando um e-mail de teste

Para testarmos se o SES está mesmo funcionando corretamente, vamos usar um recurso nativo do console, no qual só vamos precisar preencher um formulário e disparar o e-mail. Na lista de e-mails validados, selecione um deles (caso tenha validado mais de um) e clique no botão *Send a Test Email*.

Será aberta uma caixa de diálogo com um formulário, na qual você deverá informar o destinatário (To), o assunto (Subject) e o conteúdo da mensagem (Body). Repare que o remetente já está preenchido e não pode ser alterado, pois ele é o e-mail que já está validado e que você selecionou para executar o teste.

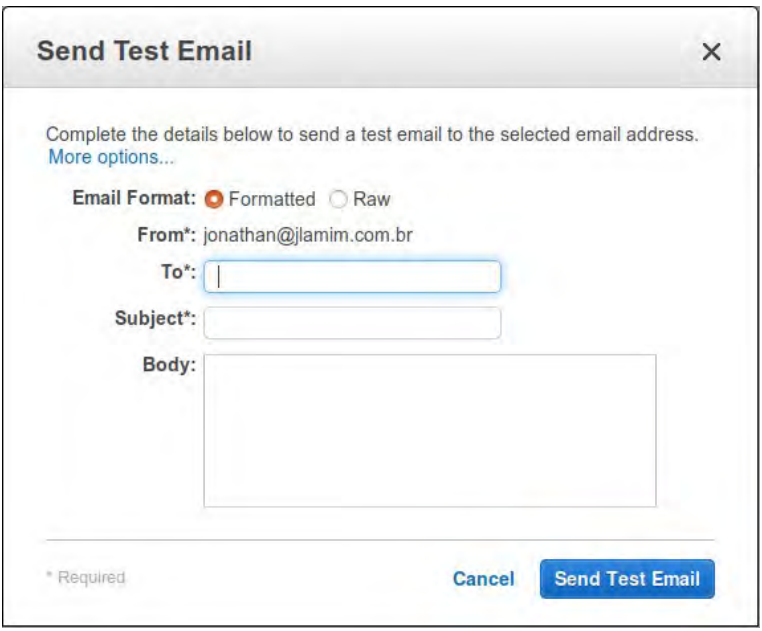


Figura 11.11: Formulário para o envio do e-mail de teste

Após preencher o formulário, clique no botão **Send Test Email** e aguarde o processamento do envio. Ao finalizar, verifique sua caixa de e-mail para ter certeza de que recebeu a mensagem.

Se recebeu, é sinal de que o SES está funcionando corretamente e você já pode começar a utilizá-lo, seja no modo **Sandbox** para novos testes, ou em modo **Produção**.

Como criar as credenciais para o envio de e-mails pelo protocolo SMTP

Para fazer o envio de e-mails com o SES usando o protocolo SMTP, é necessário criar as credenciais. Para isso, acesse a opção *SMTP Settings* no menu lateral do console do SES.

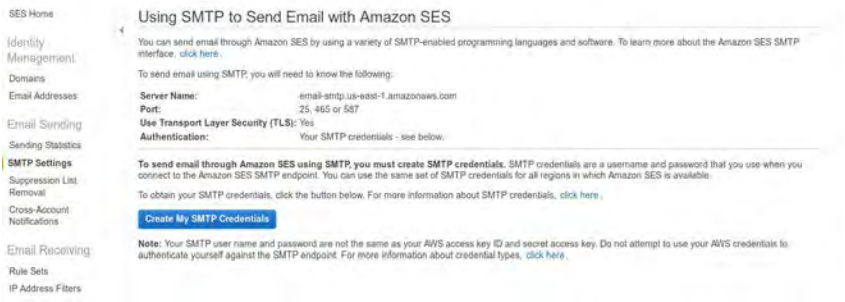


Figura 11.12: Tela inicial do SMTP Settings

Para iniciar o processo de criação das credenciais, clique no botão *Create My SMTP Credentials*. Você será redirecionado para a tela de criação das credenciais, que contém somente um campo: **IAM User Name**. Esse campo já vem preenchido automaticamente, e é aconselhado que o seu valor não seja alterado.

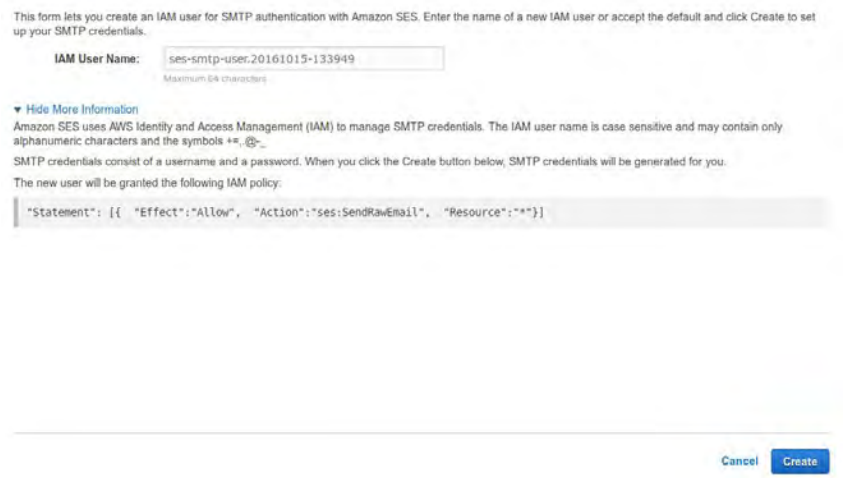


Figura 11.13: Tela de criação da credencial

Clique no botão *Create* para prosseguir com a criação das credenciais. Ao final do processo, você será direcionado para a tela com as informações da credencial que acabou de ser criada e também um botão para fazer o download do arquivo com as

configurações da credencial.

Como você não terá acesso às informações da credencial que foram criadas após sair da tela de exibição, é recomendado que faça o download do arquivo, que está em formato CSV.

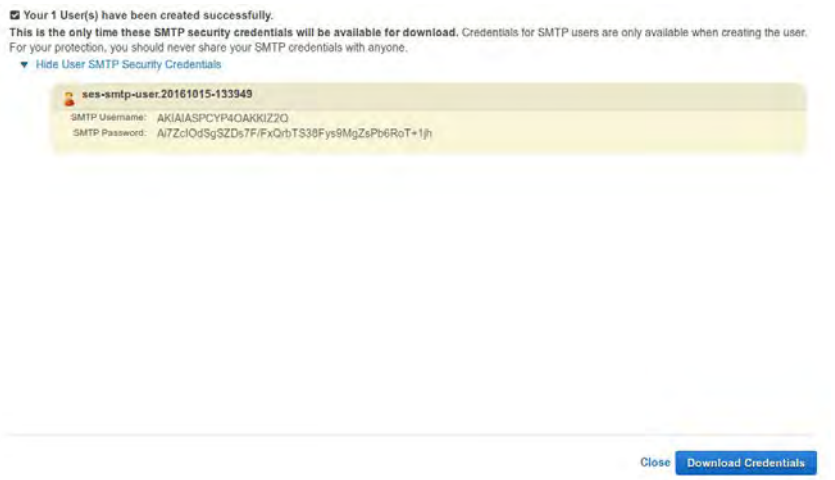


Figura 11.14: Tela com as informações da credencial

O arquivo CSV é bem simples e fácil de ser compreendido. Na primeira linha, ele traz o nome dos campos e, na segunda, os valores.

```
IAM User Name,Smtp Username,Smtp Password
"ses-smtp-user.20161015-133949",AKIAIASPCYP4OAKKIZ2Q,Ai7ZcIOdSgSZD
s7F/FxQrbTS38Fys9MgZsPb6RoT+1jh
```

As informações de portas, host, TLS e autenticação, você encontra na tela principal das configurações de SMTP.

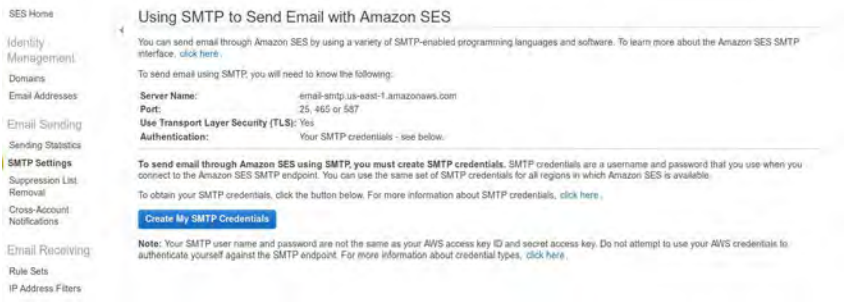


Figura 11.15: Tela inicial do SMTP Settings

Como enviar um e-mail via SMTP com SES, PHP e PHPMailer

Nas próximas linhas, você vai ver como utilizar SES, SMTP, PHP e PHPMailer para criar uma rotina de envio de e-mails. No servidor local, devemos criar um diretório chamado `php-ses-smtp`, onde armazenaremos todo o código da nossa aplicação.

Dentro do diretório `php-ses-smtp`, crie um arquivo chamado `index.php` que será o responsável por conter toda a codificação do envio de e-mail. Antes de começarmos a escrever o código, precisamos fazer a instalação da biblioteca PHPMailer, e faremos isso via Composer, usando a linha de comando a seguir:

```
composer require phpmailer/phpmailer
```

Após executar esse comando pelo terminal, estando dentro do diretório criado, você deverá ver uma mensagem semelhante à exibida a seguir:

```
Loading composer repositories with package information
Updating dependencies (including require-dev)
  - Installing phpmailer/phpmailer (v5.2.16)
    Downloading: 100%

phpmailer/phpmailer suggests installing league/oauth2-google (Need
ed for Google XOAUTH2 authentication)
Writing lock file
```

```
Generating autoload files
```

Feito isso, é hora de abrir um editor de textos, ou o software que você utiliza para escrever seus códigos, e editar o arquivo `index.php` que você criou. Veja a seguir o código completo, todo comentado, e lembre-se de substituir as informações de autenticação do SMTP pelas que você configurou.

```
<?php

require_once "vendor/autoload.php";

//Instanciamos o objeto do PHPMailer para podermos utilizá-lo
//E aplicamos as devidas configurações
$mail = new PHPMailer;

//Ativamos o modo Debug do SMTP.
$mail->SMTPDebug = 3;

//Definimos que o envio do email será pelo protocolo SMTP
$mail->isSMTP();

//Definimos o nome do host (obtido em SMTP Settings no console da
AWS)
$mail->Host = "server_name_do_ses";

//Definimos se deve ser feita a autenticação no SMTP ou não
$mail->SMTPAuth = true;

//Definimos o nome de usuário e senha para a autenticação
$mail->Username = "username_da_credencial_criada";
$mail->Password = "password_da_credencial_criada";

//Definimos o modo segurança para o SMTP (no SES é necessário usar
 o modo TLS)
$mail->SMTPSecure = "tls";

//Definimos a porta que será utilizada para envio ( no SES as port
as disponíveis são 25, 465 e 587)
$mail->Port = 587;

//Definimos nome e email do remetente (que deve ser o que está val
idado no SES)
$mail->From = "jonathan@jlamim.com.br";
$mail->FromName = "Jonathan Lamim Antunes";
```

```
//Definimos nome e email do destinatário (o nome do destinatário é
 opcional)
//Caso esteja usando o modo Sandbox do SES, o email do destinatári
o deve ser um
//dos emails validados
$mail->addAddress("jonathan@jlamim.com.br", "Jonathan Lamim");

//Definimos se o conteúdo do email será no formato HTML ou texto p
lano
//Nesse caso será HTML
$mail->isHTML(true);

//Definimos o assunto e o corpo do email
$mail->Subject = "Enviando um email via SMTP com SES, PHP e PHPMai
ler";
$mail->Body = "<i>Corpo do email utilizando tags <b>HTML</b>.</i>"
;
$mail->AltBody = "Texto alternativo caso o email não seja renderiz
ado no formato HTML";

//Processamos o envio do email e verificamos o status
if(!$mail->send())
{
    //Em caso de erro exibe a mensagem de erro
    echo "Falha no envio: " . $mail->ErrorInfo;
}
else
{
    //Em caso de envio com sucesso, exibe a mensagem de sucesso
    echo "Email enviado com sucesso.";
}
```

Para testar esse código, você pode acessar pelo browser através do seu localhost, que teria uma URL semelhante a http://localhost/php-ses-smtp. Outra possibilidade é você fazer o teste através do terminal, executando o comando a seguir, estando dentro do diretório `php-ses-smtp` :

```
php index.php
```

Após executar o teste, verifique a sua caixa de e-mails e veja se recebeu corretamente.

Como enviar um e-mail via SMTP com SES e Python

Vamos ver agora um exemplo de envio de e-mail com SES e SMTP usando um script na linguagem Python 2. Crie um arquivo chamado `ses-smtp.py` em sua máquina e coloque nele o código a seguir. Lembre-se de substituir as informações de autenticação do SMTP pelas que você configurou.

```
#!/usr/bin/python
# -*- coding: iso-8859-1 -*-
import os, sys

#Importamos a biblioteca de envio de email por SMTP
import smtplib
from email.MIMEMultipart import MIMEMultipart
from email.MIMEText import MIMEText

#Definimos os dados do SMTP
username_smtp = "username_da_credencial_criada"
password_smtp = "password_da_credencial_criada"
porta_smtp = 587
host_smtp = "server_name_do_ses"

#Definimos os emails do remetente e do destinatário
fromaddr = "jonathan@jlamim.com.br"
toaddr = "jonathan@jlamim.com.br"

#Definimos a estrutura da mensagem, informando o remetente, o dest
inatário e o assunto
msg = MIMEMultipart()
msg['From'] = fromaddr
msg['To'] = toaddr
msg['Subject'] = "Enviando um email via SMTP com SES e Python"

#Definimos o corpo da mensagem
body = "Lorem ipsum dolor sit amet, consectetur adipisicing elit,
sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
 Ut enim ad minim veniam, quis nostrud exercitation ullamco labori
s nisi ut aliquip ex ea commodo consequat."
msg.attach(MIMEText(body, 'plain'))

#Concluímos as configurações do envio
server = smtplib.SMTP(host_smtp, porta_smtp)
server.starttls()
server.login(username_smtp, password_smtp)
text = msg.as_string()

#Processamos o envio e exibimos mensagem de status
```

```
try:
    server.sendmail(fromaddr, toaddr, text)
    print "Email enviado com sucesso"
except SMTPException:
    print "Não foi possível enviar o email"

server.quit()
```

Para testar esse código, acesse o terminal, entre no diretório onde salvou o arquivo e execute o comando a seguir:

```
python index.py
```

11.4 CONSIDERAÇÕES FINAIS

Viu como é fácil utilizar o SES para enviar e-mails a partir de suas aplicações? Melhor que essa facilidade é o baixo custo para envio desses e-mails.

Através de um processo rápido de configuração, você já tem disponível as credenciais de acesso para poder fazer o envio usando o protocolo SMTP a partir da linguagem de programação que você tiver mais domínio.

Links úteis

- **Documentação:** https://aws.amazon.com/pt/ses/
- **Documentação para desenvolvedores:** https://aws.amazon.com/pt/documentation/ses/
- **RFC 822:** http://www.ietf.org/rfc/rfc0822.txt
- **PHPMailer:** https://github.com/PHPMailer/PHPMailer

Capítulo 12

AMAZON SNS (SIMPLE NOTIFICATION SERVICE)

O SNS (ou Simple Notification Service) é o serviço da Amazon responsável por disparar notificações para usuários e endpoints.

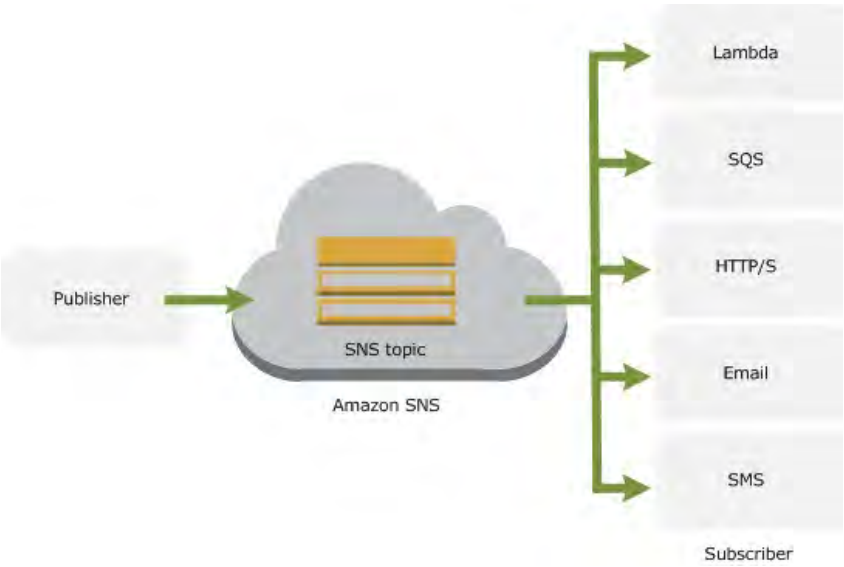


Figura 12.1: Fluxo de funcionamento do SNS

No SNS, os usuários que receberão essas notificações são organizados por tópicos, como acontece em fóruns de discussão, por exemplo. O usuário se inscreve em um tópico e, a partir de então, ele receberá notificações somente daquele tópico, ou seja, somente o conteúdo de interesse dele.

Atualmente, o SNS tem suporte para o envio de notificações por e-mail, SMS, HTTP/HTTPS, SQS e push para Android, iOS e Kindle Fire.

Seu custo é bastante baixo, por exemplo:

- **Mobile Push:** primeiro milhão é gratuito, e depois 0,50 USD por cada milhão enviado.
- **SMS:** 100 primeiros são gratuitos, e depois 0,75 USD por cada 100 enviados.
- **Email:** 1000 primeiros são gratuitos, e depois 2 USD por cada 100.000 enviados.

Soma-se a esses custos a cobrança pela quantidade de dados que serão trafegados para o disparo, que variam conforme a região e a quantidade de dados. Mas vale lembrar de que se estiver utilizando os tópicos SNS nas mesmas regiões dos servidores do EC2, não haverá cobrança de tráfego.

12.1 ACESSANDO O SNS ATRAVÉS DO CONSOLE DE GERENCIAMENTO

Acesse o console de gerenciamento da sua conta na Amazon AWS e no menu *Services*, e então acesse a opção *SNS*. Se você ainda não estiver usando o SNS, a tela exibida será semelhante à da figura a seguir:

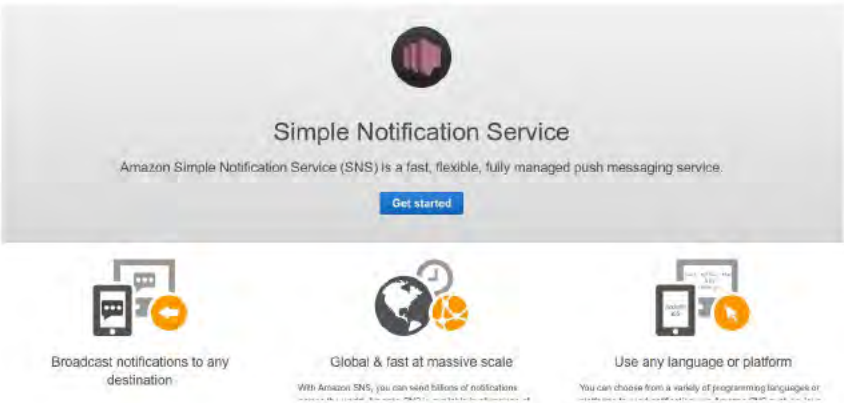


Figura 12.2: Tela inicial do SNS

Ao clicar em *Get started*, você verá a tela principal, com o menu de operação do SNS.

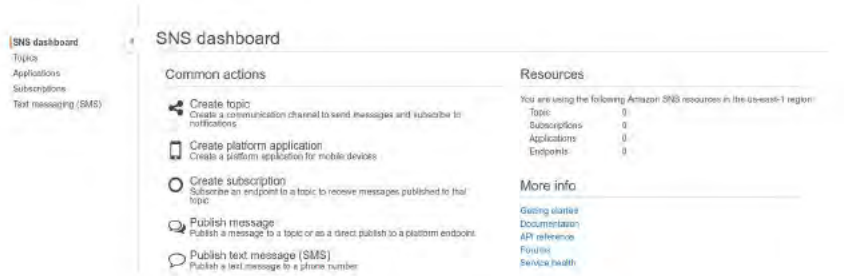


Figura 12.3: Tela principal do SNS

Nessa tela, você poderá controlar as operações comuns do SNS:

- Criação de tópicos;
- Criação de plataformas de aplicação;
- Adicionar usuários aos tópicos;
- Publicar mensagens e SMS.

E no menu lateral, você terá acesso às operações de maneira agrupada, para cada situação de uso do SNS.

Outras formas de acesso ao SNS

Além de poder acessar via interface gráfica, pelo console, você poderá acessar o SNS através:

- Da linha de comando (CLI);
- Da AWS Tools para Windows PowerShell;
- Das SDKs de desenvolvimento;
- Da API de consulta do SNS.

12.2 ENVIANDO MENSAGEM PARA UM TÓPICO ATRAVÉS DO CONSOLE

Para enviar uma mensagem em um tópico, é preciso primeiramente criá-lo. Para isso, acesse a opção *Create topic* no dashboard do SNS. Na tela que se abrirá, você terá dois campos para o preenchimento:

- **Topic name:** nome do tópico contendo no máximo 256 caracteres alfanuméricos, hífen e/ou underscores, sem espaços.
- **Display name:** nome utilizado para identificar o tópico no momento em que um usuário for se inscrever nele, é o nome de exibição. Não pode ter mais de 10 caracteres.

Figura 12.4: Tela de cadastro do tópico

Após preencher os campos, clique em *Create topic* para concluir o processo.

Você será redirecionado para uma tela com informações sobre o tópico que acabou de criar, e poderá então criar as subscrições e fazer os disparos das mensagens.

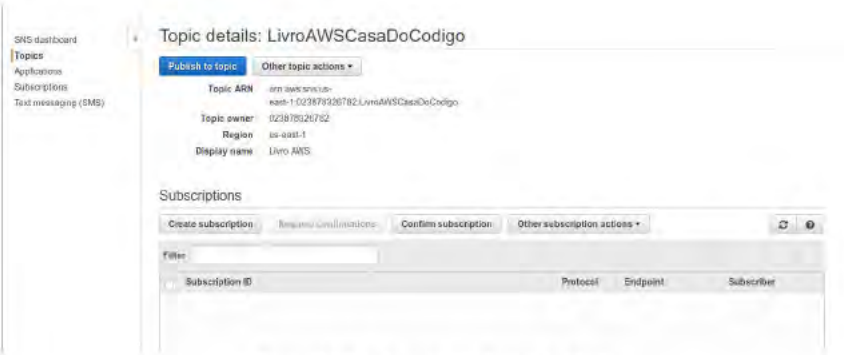


Figura 12.5: Tela após o cadastro do tópico

- **Topic ARN:** é o identificador do tópico que a API utiliza para o envio das mensagens.

> ARN significa **Amazon Resource Name**

- **Topic Owner:** é o código da conta de quem criou o tópico.
- **Region:** região onde o tópico foi criado.

Atenção para região onde criará o tópico, pois a escolha da região pode influenciar nos custos do serviço, conforme foi citado no início do capítulo.

- **Display Name:** nome de exibição do tópico.

Agora que o tópico foi criado, é o momento de adicionar usuários a ele, para que então possamos enviar mensagens. Clique no botão *Create Subscription* para inserir as inscrições para o tópico que acabamos de gerar.

Será aberta uma tela para que você defina o protocolo a ser usado para essa subscrição e o endpoint. O formato do endpoint vai variar conforme o protocolo escolhido. Os protocolos disponíveis são:

- HTTP e HTTPS
- Email e Email-JSON
- Amazon SQS
- Application
- AWS Lambda
- SMS

Selecione o protocolo **Email** para dar sequência ao cadastro do endpoint e informe o seu e-mail. Em seguida, clique em *Create subscription* para finalizar.

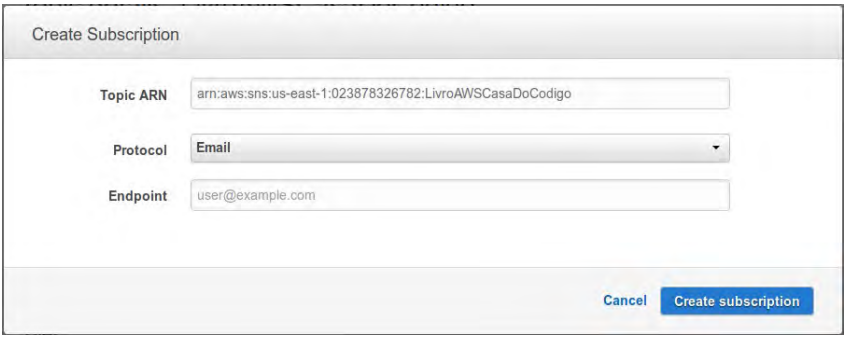


Figura 12.6: Tela de cadastro de inscrito no tópico

Após a conclusão do cadastro, ele será dado como *PendingConfirmation*. Esse status será modificado após você acessar o seu e-mail e confirmar a inscrição no tópico.

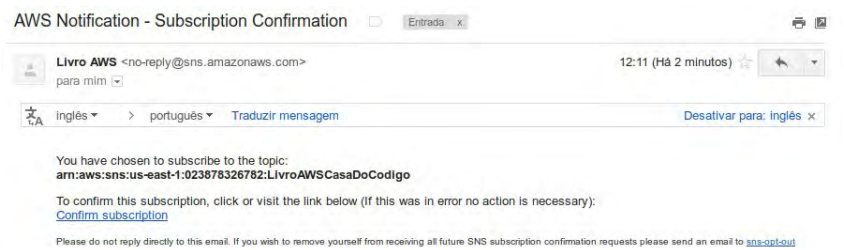


Figura 12.7: E-mail de confirmação da inscrição no tópico

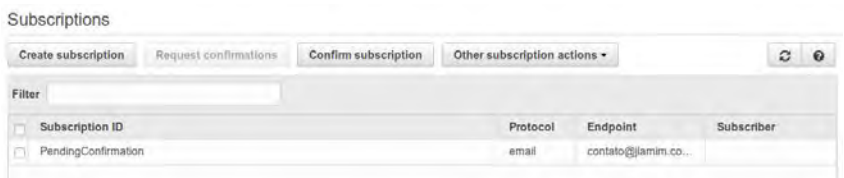


Figura 12.8: Lista de inscritos

Após clicar no link de confirmação da inscrição, uma tela semelhante à figura a seguir deverá ser exibida:

Figura 12.9: Inscrição confirmada

Com a confirmação executada, a lista de inscritos no tópico sofre alteração, pois passa a exibir o ARN da inscrição e o código do *Subscriber*.

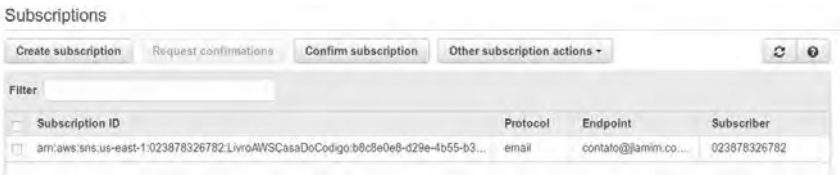


Figura 12.10: Lista de inscritos confirmados

Você pode cadastrar quantos inscritos quiser, com vários protocolos diferentes para um mesmo tópico. No momento em que você publicar alguma mensagem nele, todos receberão, independente do protocolo que foi usado para o cadastro.

Para testar o envio para múltiplos protocolos, crie mais uma inscrição, dessa vez usando o protocolo *SMS*. Informe o seu número de celular usando a seguinte formatação: `+55 xx xxxxxxxxx` .

Após concluir o cadastro, clique no botão *Publish to topic* que fica localizado no topo da tela. Você será redirecionado para a tela

de publicação de mensagens no tópico, onde deverá informar o assunto, o formato da mensagem, a mensagem e o TTL (Time to Live).

> TTL é o tempo, em segundos, para disparo das mensagens.

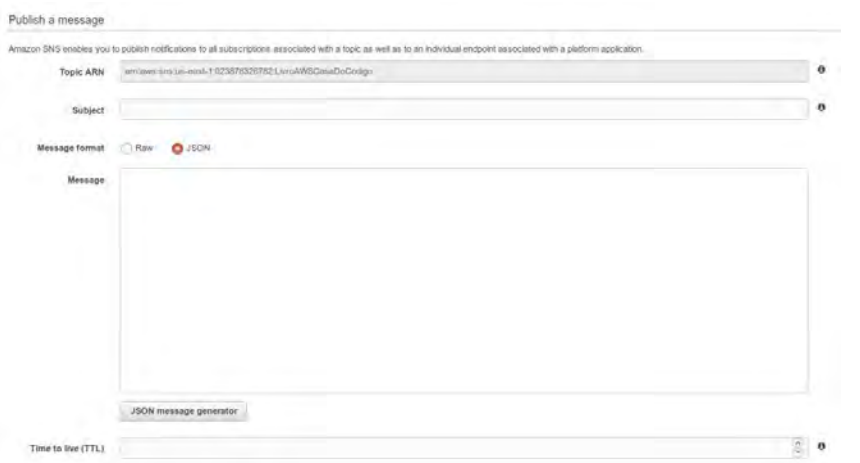


Figura 12.11: Tela de cadastro de publicação no tópico

Para auxiliar no processo de criação da estrutura da mensagem, clique no botão *JSON message generator*. Na tela que será aberta, informe as configurações para o envio da mensagem.

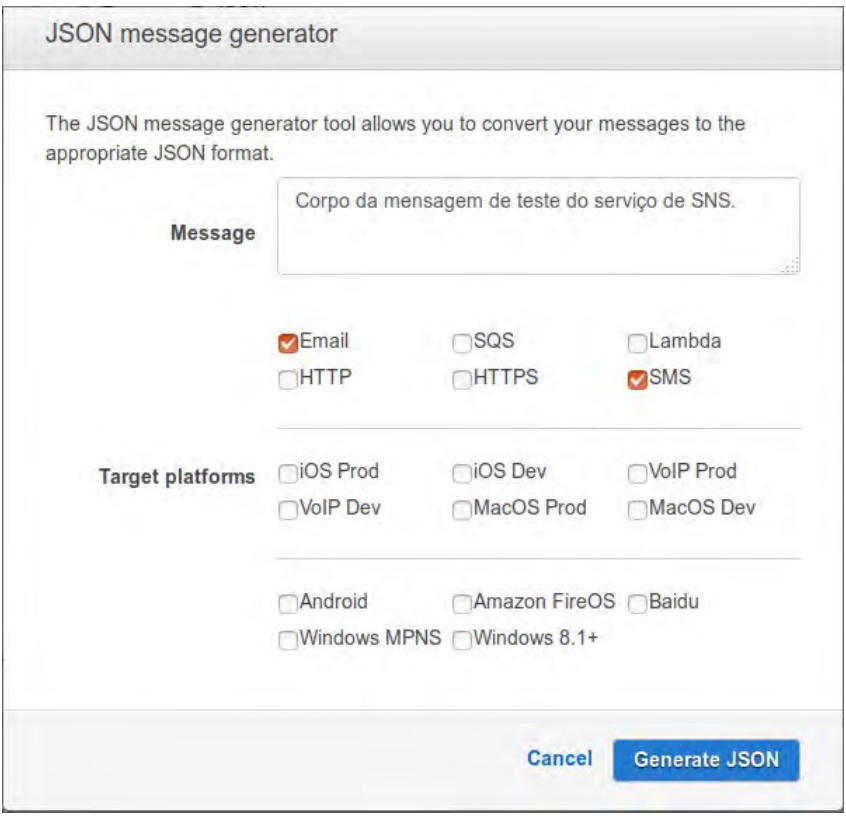


Figura 12.12: JSON message generator

Preencha os campos do formulário e, em seguida, clique em *Publish message* para disparar. Você será redirecionado para a tela de detalhes do tópico com a mensagem informativa do status do disparo da mensagem.

Verifique o seu e-mail e celular para se certificar do recebimento da mensagem.

Figura 12.13: Mensagem recebida no e-mail

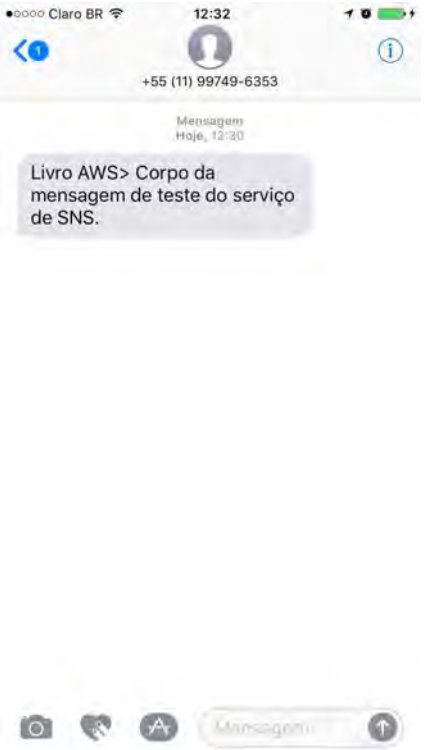


Figura 12.14: Mensagem recebida no celular

12.3 ENVIANDO MENSAGEM PARA UM TÓPICO ATRAVÉS DA API PARA PYTHON

Também é possível fazer o envio de mensagens para os tópicos,

além de criá-los e gerenciá-los usando a API. No código a seguir, foi implementado o envio de mensagem para o tópico que foi criado anteriormente neste capítulo.

O código foi escrito em Python e está utilizando a library `boto3` — biblioteca usada em Python para acesso às funcionalidades das APIs dos serviços da Amazon AWS. Assim como temos SDKs para PHP, Java e outras linguagens, para Python usamos a `boto3`.

> Para mais informações sobre boto3, acesse http://boto3.readthedocs.io.

Na primeira parte do código, são importadas as bibliotecas `boto3` e `json`. Na segunda parte, são definidas as variáveis de configuração que são necessárias para o uso da API do SNS.

No terceiro bloco, temos mais 2 variáveis, que são criadas para receber os retornos do boto3 e disponibilizar os métodos para a execução do envio da mensagem. E no quarto, o JSON da mensagem que será enviada.

Por fim, a mensagem é enviada usando `topic.publish`, armazenando o retorno na variável `response`, que exibimos na tela na linha seguinte.

```
import boto3
import json

aws_access_key = "SUA_ACCESS_KEY"
aws_secret_access = "SUA_SECRET_ACCESS_KEY"
region = "REGIAO_DO_TOPICO_CRIADO"
arn_topic = "ARN_DO_TOPICO_CRIADO"

sns = boto3.resource('sns',region_name=region,aws_access_key_id=aw
s_access_key,aws_secret_access_key=aws_secret_access)
```

```
topic = sns.Topic(arn_topic)

message = json.dumps({"default": "Mensagem enviada pela API para P
ython, usando boto3","email": "Mensagem enviada pela API para Pyth
on, usando boto3","sms": "Mensagem enviada pela API para Python, u
sando boto3"})

response = topic.publish(Subject='PythonAPI',MessageStructure='jso
n',Message=message)

print response
```

12.4 USANDO O SNS PARA ENVIO DE PUSH NOTIFICATION

Como já foi dito, é possível enviar notificações para dispositivos móveis através do SNS. Sabe-se que cada sistema operacional móvel possui uma plataforma e formato específico para envio dessas notificações.

No Google usa-se o FCM (Firebase Cloud Messaging), e na Apple o APNS (Apple Push Notification Service). E até mesmo a Amazon possui a ADM (Amazon Device Messaging) para o Kindle Fire.

Com essas particularidades de cada plataforma, acaba sendo necessário escrever o código de integração para cada uma delas, o que torna o processo de escrita e manutenção de código mais massivo. O SNS foi construído para facilitar esse trabalho e permitir que, através de um código-fonte único, seja possível enviar as notificações para Android, iOS e Kindle Fire.

Para usar esse recurso, é necessário ter conhecimento sobre mobile push e programação mobile, além de necessitar de contas ativas no Google e na Apple para poder obter as chaves necessárias.

Neste livro, vamos abordar o processo de configuração das aplicações dentro do console do SNS, para que, a partir daí, você possa integrar o envio de notificações com qualquer linguagem de programação e sistema operacional.

Para criar uma aplicação, você deve ir até o menu lateral do console do SNS e clicar em *Applications*. E em seguida clique em *Create platform application*.

A quantidade de campos a serem preenchidos vai variar conforme a plataforma escolhida no campo *Push notification platform*. Mas a critério de exemplo, vamos criar uma aplicação para enviar notificações para dispositivos que usem Android, ou seja, escolher a opção GCM.

O console da Amazon ainda mostra como GCM, mas a Google mudou a nomenclatura para FCM.

Preencha os campos **Application Name** com o nome da aplicação e **API Key** com a chave da sua aplicação, que deve ser criada primeiramente no console do Firebase Cloud Messaging.

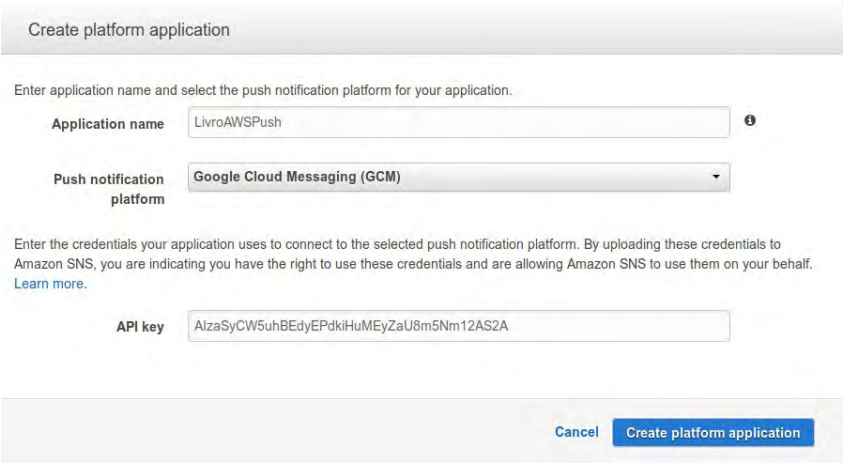


Figura 12.15: Tela de criação de aplicação para envio de push notification

Após criada a aplicação, você terá uma tela semelhante à exibida a seguir:

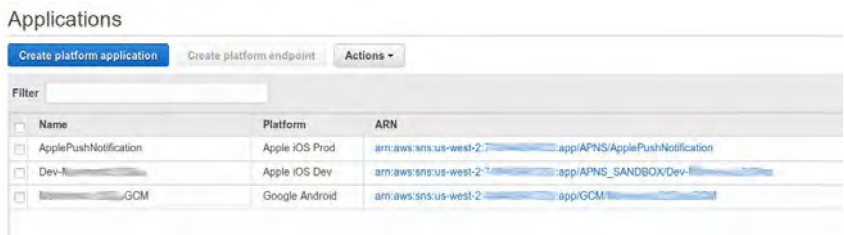


Figura 12.16: Lista de aplicações criadas

Clicando no link do ARN, você será direcionado para a tela na qual serão exibidas as informações sobre a aplicação e os endpoints, ou seja, usuários da aplicação que vão receber as notificações.

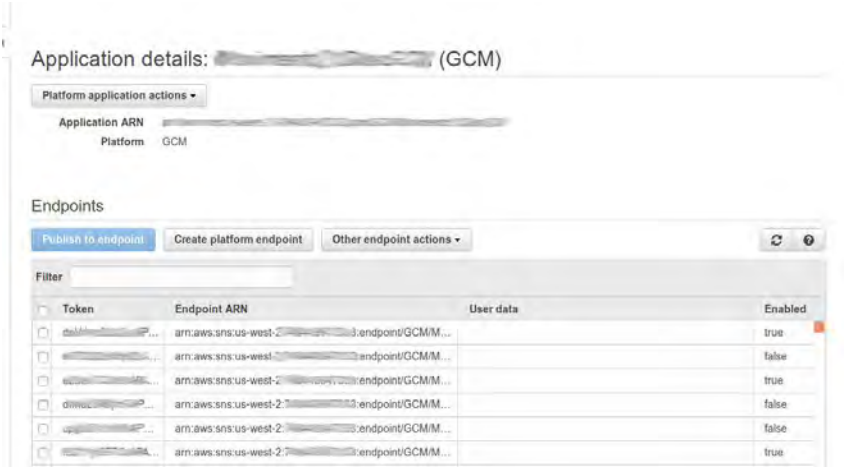


Figura 12.17: Lista dos endpoints cadastrados para a aplicação

Para adicionar manualmente um endpoint, você deve clicar no botão *Create platform endpoint*, então preencher o formulário que será aberto e, em seguida, clicar em *Add endpoint* para finalizar.

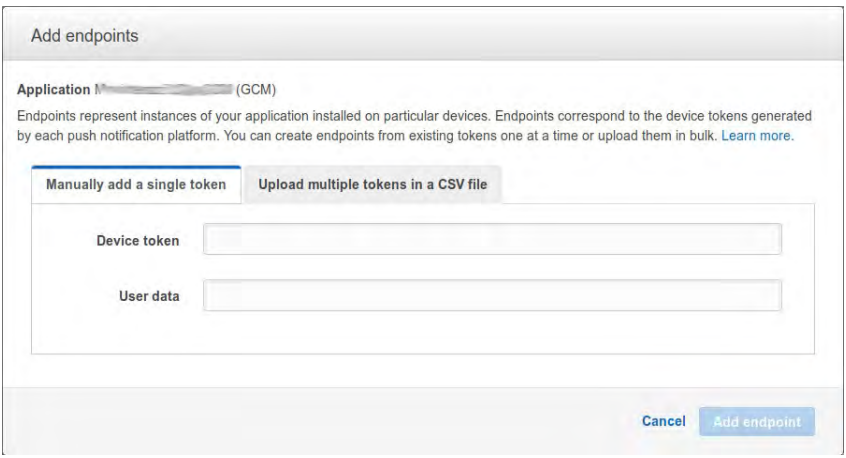


Figura 12.18: Formulário para cadastrar um endpoint

A forma mais comum de se cadastrar endpoints para aplicações no SNS é usando a API para a linguagem específica da sua aplicação, pois você precisará informar o token do dispositivo do usuário.

12.5 CONSIDERAÇÕES FINAIS

Viu como é fácil configurar e utilizar o SNS para enviar notificações para usuários tanto via e-mail quanto push? Com a API do SNS, você tem todas essas possibilidades, podendo implementá-las em várias linguagens. Vale a pena conferir mais detalhes na documentação.

Links úteis

- **Documentação:** https://aws.amazon.com/pt/sns/
- **Documentação para desenvolvedores:** https://aws.amazon.com/pt/documentation/sns/
- **boto3:** http://boto3.readthedocs.io
- **Firebase Cloud Messaging:** https://firebase.google.com/docs/cloud-messaging/
- **Apple Push Notification Service:** http://goo.gl/E3N8I4

Capítulo 13

AMAZON CLOUDWATCH

O Amazon CloudWatch é o serviço de monitoramento de recursos da AWS. Através dele, você poderá configurar diversos tipos de monitoramento de recursos, e até mesmo os gastos que sua conta está gerando.

Dentre os serviços que podem ser monitorados estão as instâncias do EC2, os volumes do EBS e bancos de dados do RDS. Neste capítulo, falaremos sobre o funcionamento do CloudWatch e o monitoramento de instâncias do EC2 e gastos com a conta.

13.1 COMO FUNCIONA O CLOUDWATCH

Para entender como o CloudWatch funciona, vamos usar um exemplo bem simples. Imagine que você possui uma aplicação rodando e quer ser notificado em 2 momentos: quando o uso da CPU for superior a 50% e quando for superior a 85%. Para isso, você precisará criar 2 métricas de monitoramento, que dispararão 2 alarmes, um para cada métrica.

No CloudWatch, uma **métrica** é a informação com possibilidade de monitoramento, que no caso do exemplo é o uso da CPU, mas pode ser o estado do disco ou mesmo o status da instância. Além da *métrica*, o CloudWatch tem os **alarmes**, que são os avisos relacionados à métrica de monitoramento configurada.

Esses alarmes executarão alguma ação automática, que pode ir

desde reiniciar a instância, ou até mesmo aumentar o número de servidores disponíveis para a aplicação. Após configurados os alarmes, você poderá visualizar os gráficos de monitoramento de cada métrica, e assim ter uma visão geral de como está o monitoramento.

13.2 COMO CRIAR ALARMES DE MONITORAMENTO DO EC2

Com base nesse exemplo, vamos criar os alarmes para o monitoramento do uso da CPU. Acesse o menu *Services* e escolha em seguida *CloudWatch*. Você deverá ver uma tela semelhante à figura a seguir:

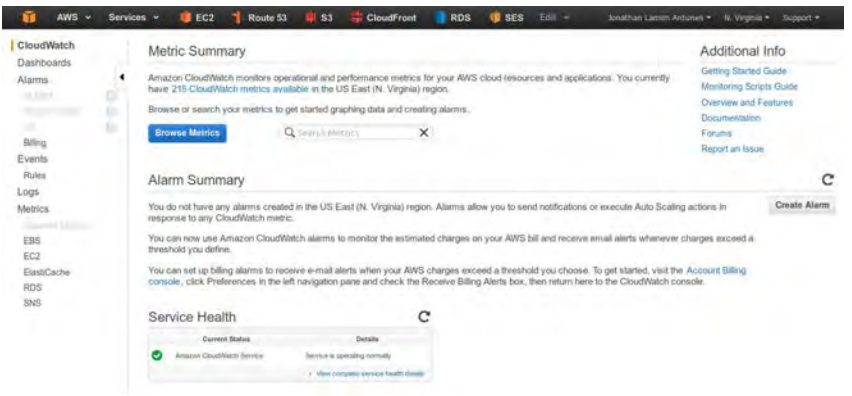


Figura 13.1: CloudWatch Dashboard

Essa é a tela principal do CloudWatch. Nela você tem informações sobre os alarmes e métricas criados, links para informações adicionais e documentação, e do lado esquerdo o menu principal do CloudWatch.

Vá até o menu lateral e clique na opção *Alarms*. Você deverá ver uma tela como a da figura seguinte:

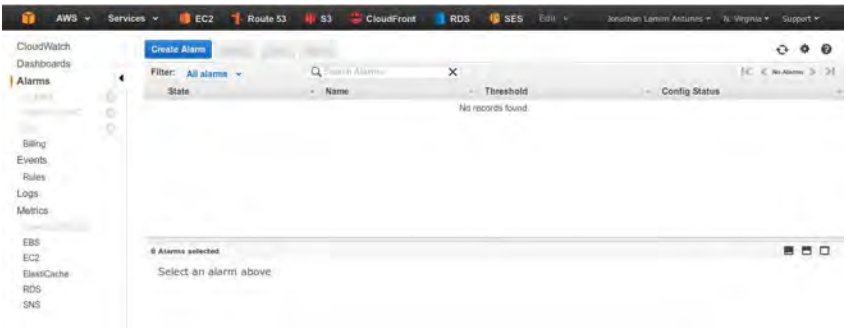


Figura 13.2: Lista de alarmes

A lista de alarmes vai estar vazia, pois até o momento não foi criado nenhum alarme de monitoramento. Faremos isso agora.

Clique no botão *Create Alarm* para iniciar o processo de criação do alarme, que é divido em 2 etapas. A primeira é a de seleção da métrica que deverá ser monitorada. Para atender ao nosso exemplo, vamos monitorar o uso da CPU da nossa instância no EC2. Selecione então a opção **Per-Instance Metrics** para continuar.

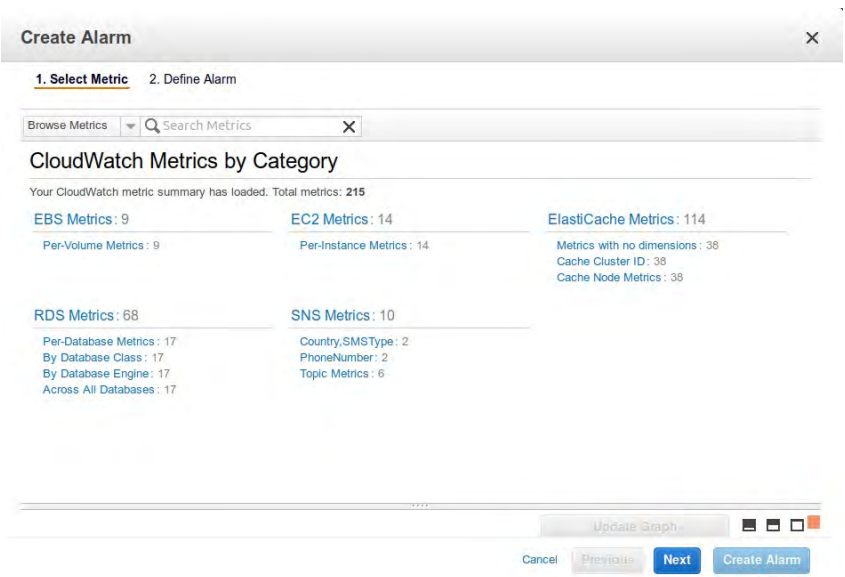


Figura 13.3: Criação do alarme — Lista de métricas

Em seguida, você deverá selecionar a métrica que deseja usar para monitoramento. No nosso caso vamos, escolher a opção **CPUUtilization**. Selecione o checkbox dessa métrica e, depois, clique em *Next*.

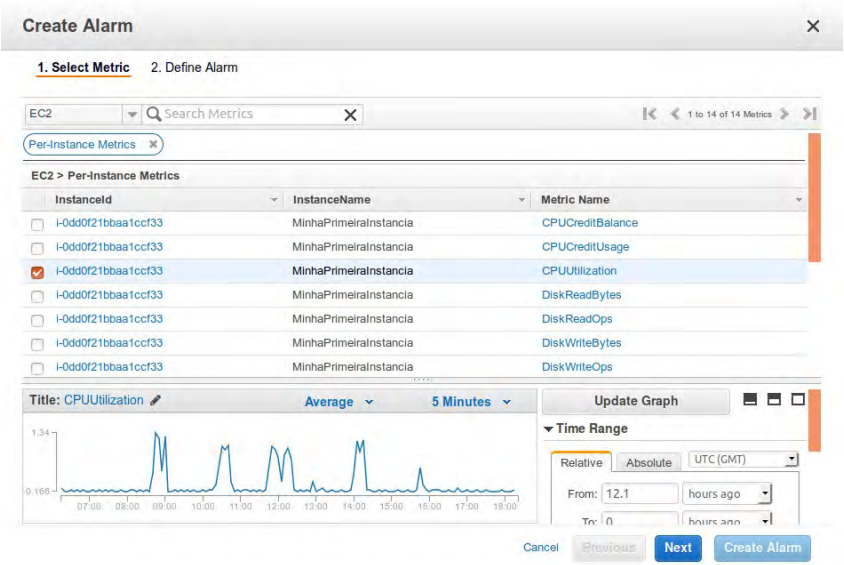


Figura 13.4: Criação do alarme — Seleção da métrica

Assim finaliza-se a primeira etapa da criação do alarme. Na segunda etapa, é o momento de configurar as definições do alarme e as ações que deverão ser executadas quando a métrica informada for atingida.

Os 2 primeiros campos, **Name** e **Description**, são campos de identificação do seu alarme. Utilize um nome que facilite a sua identificação, como por exemplo, **CPU50Utilization**, que já deixa claro que é um alarme para os 50% de uso da CPU.

Para o campo **Whenever - is**, vamos informar 50 (não é necessário colocar o sinal de %), que é o percentual que queremos usar para o momento em será disparado o alarme. Na seção *Actions*

é que será configurada a ação desse alarme, que no caso do nosso exemplo enviará um e-mail.

No campo **Whenever this alarm**, selecione *State is ALARM*. E no campo **Send notification to**, clique em *New list* para poder informar o e-mail que vai receber o aviso.

Como os avisos por e-mail utilizam tópicos do SNS para disparo, você deverá informar o nome do tópico e o e-mail. Para o nome do tópico, informe **CloudWatchAlarms** e, para **Email list**, use um e-mail válido que você tenha acesso.

Se quiser criar mais de uma notificação para o mesmo alarme, basta clicar no botão + *Notification*, que um novo bloco será adicionado para que possa configurá-lo. Além do botão + *Notification*, ainda existem os botões + *AutoScaling Action* e + *EC2 Action*.

O + *AutoScaling Action* é usado para fazer o escalonamento automático da instância. Já o + *EC2 Action* é para executar uma operação no EC2 (recover, stop, terminate, reboot).

Do lado direito da tela, existem 2 campos: **Period** e **Statistic**. O *Period* é responsável por definir o intervalo de tempo do monitoramento. Como o CloudWatch faz parte do programa de gratuidade, deixe selecionada a opção *5 Minutes* para que você possa testar sem custo.

O *Statistic* é o tipo de estatística que será exibida para o alerta. Você pode escolher entre Average, Minimun, Maximun, Sum e Data Samples. Mantenha a opção *Average* selecionada para esse exemplo.

Para concluir, clique em *Create Alarm*.

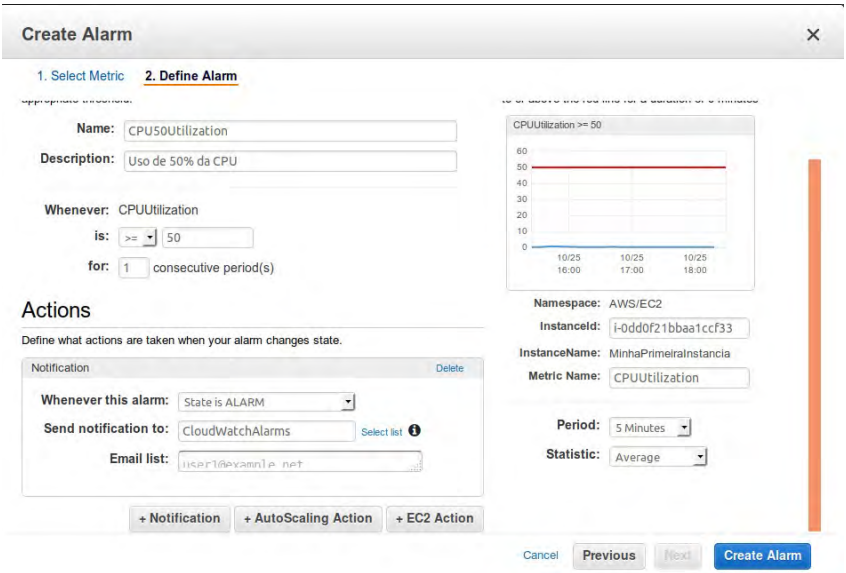


Figura 13.5: Criação do alarme — Configuração do alarme

Como você criou um tópico no SNS para disparar a notificação do alarme, será necessário confirmar o e-mail informado. Então uma tela como a da figura a seguir será exibida.

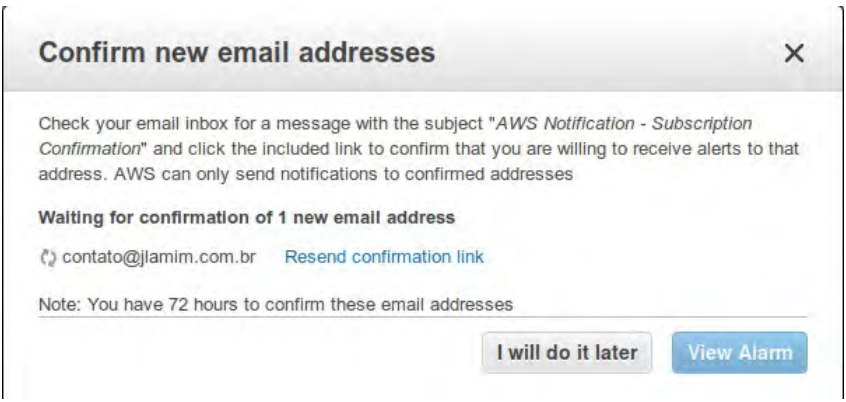


Figura 13.6: Criação do alarme — Confirmação do e-mail

Você pode clicar em *I will do it later* para sair da tela e voltar para a lista de alarmes e depois confirmar o e-mail, ou então abrir

sua caixa de e-mail e confirmá-lo antes de fechar a tela. Se optar pela segunda opção, segundos após confirmar o e-mail, o botão *View Alarm* será habilitado.

> O prazo para confirmação do endereço de e-mail é de 72 horas, fique atento.

Feito isso, a sua lista de alarmes deverá ficar semelhante à da figura:

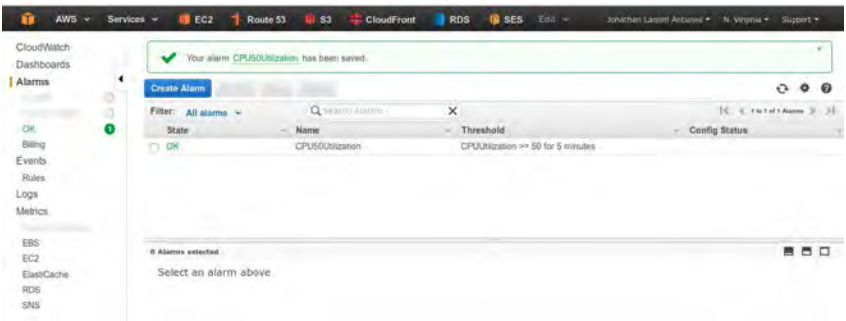


Figura 13.7: Lista de alarmes

Como nosso exemplo vai usar 2 alarmes, repita o processo descrito anteriormente para configurar o segundo alarme, que deverá disparar quando a CPU atingir 85% de uso. Quando for definir a notificação, você não precisará criar outro tópico. Bastará selecionar na lista a opção *CloudWatchAlarms*.

Após configurados os alarmes, é só ficar de olho no e-mail. Assim, sempre que uma métrica for atingida, o alarme será acionado e a ação configurada será executada, e no caso do nosso exemplo, o e-mail de notificação será enviado.

Os diferentes status de um alarme

Um alarme do CloudWatch pode ter 3 status diferentes:

- **ALARM** — Significa que há algum problema no monitoramento e que esse deve ser analisado e resolvido.
- **INSUFICIENT** — Significa que os dados da métrica não são suficientes para o monitoramento. É comum que métricas recém-criadas apareçam com esse status, então atualize a página e veja se o status muda para OK. Se não mudar, verifique as configurações do alarme para ter certeza de que todas as informações necessárias para o monitoramento foram informadas corretamente.
- **OK** — Significa que o monitoramento está correto, tudo dentro dos parâmetros especificados.

13.3 COMO CRIAR ALARMES DE MONITORAMENTO DOS GASTOS DA CONTA

Assim como é importante criar os alarmes para monitorar o funcionamento dos serviços utilizados, é importante monitorar os gastos da sua conta. Para fazer a configuração dos alarmes de monitoramento dos custos pelo painel do CloudWatch, você precisará atualizar algumas configurações no console de gerenciamento de gastos.

Ele pode ser acessado pelo endereço https://console.aws.amazon.com/billing, ou pelo menu pessoal *Billing & Cost Management*.

Figura 13.8: Console de gerenciamento de gastos

Na página principal, você verá informações sobre os gastos mensais da sua conta e, do lado esquerdo, o menu principal desse console de gerenciamento. Para poder criar os alarmes, você deve acessar a opção *Preferences* no menu lateral do console de gerenciamento de gastos, selecionar a opção *Receive Billing Alerts* e clicar em *Save preferences*.

Ao atualizar as preferências, conforme descrito, você estará habilitando o monitoramento dos gastos da conta e poderá então criar os alarmes no CloudWatch.

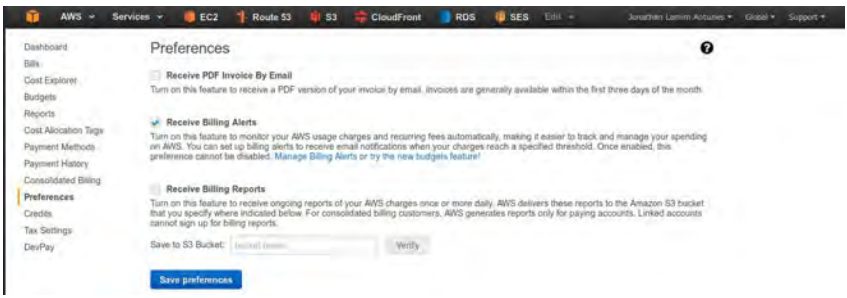


Figura 13.9: Console de gerenciamento de gastos — Preferências

Volte para o CloudWatch para podermos então configurar um

alarme para o monitoramento de gastos da conta.

No console do CloudWatch, vá até o menu lateral e selecione a opção *Billing*. Como não temos nenhum alarme criado para monitorar os gastos, você verá informações, links e um botão *Create Alarm*. Clique nesse botão para iniciarmos o processo de configuração do alarme para os gastos da conta.

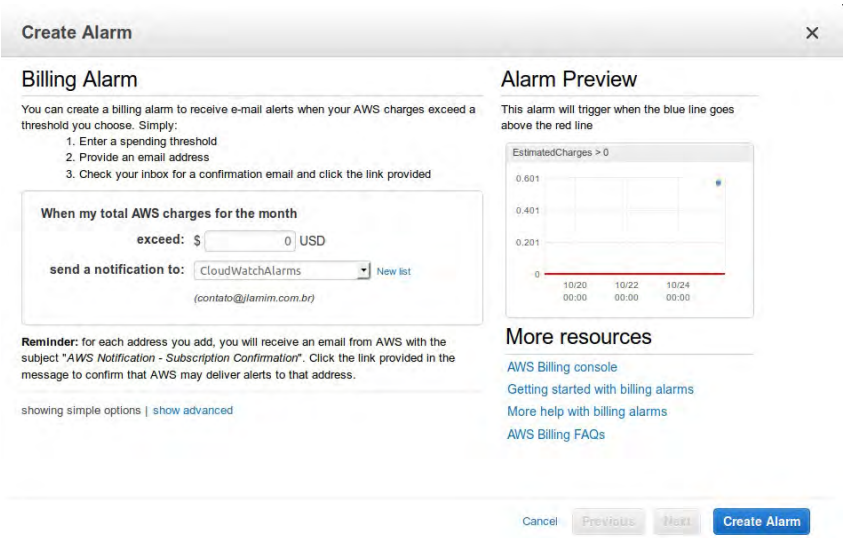


Figura 13.10: CloudWatch — Alarme de gastos da conta

É uma configuração bem simples de ser feita, pois diferente do alarme de monitoramento do EC2 (que você tem várias opções de monitoramento), para os gastos você monitora o valor do gasto. Ou seja, define um limite e, quando esse limite for excedido, o alarme é disparado.

Como você deve estar utilizando os recursos que se enquadram no programa de gratuidade, é quase certo que esse alarme não será disparado. Mas vamos configurá-lo mesmo assim, usando um valor baixo, de 1 dólar.

No campo **send a notification to**, mantenha selecionada a opção *CloudWatchAlarms*, e em seguida clique em *Create Alarm*. A partir desse momento, os gastos da conta já estão sendo monitorados.

Existem outras possibilidades de configuração para o monitoramento de gastos que você pode configurar criando um alarme através do botão localizado na página principal do CloudWatch, tendo acesso a essas outras possibilidades.

O processo é semelhante ao que foi feito para configurar o monitoramento de uso da CPU no EC2, em 2 etapas. Na primeira, você deverá escolher a opção **Billing Metrics** para então poder visualizar todas as opções de monitoramento disponíveis. Escolha a opção desejada e siga com o processo de configuração clicando em *Next*. Os demais passos são iguais aos já vistos anteriormente.

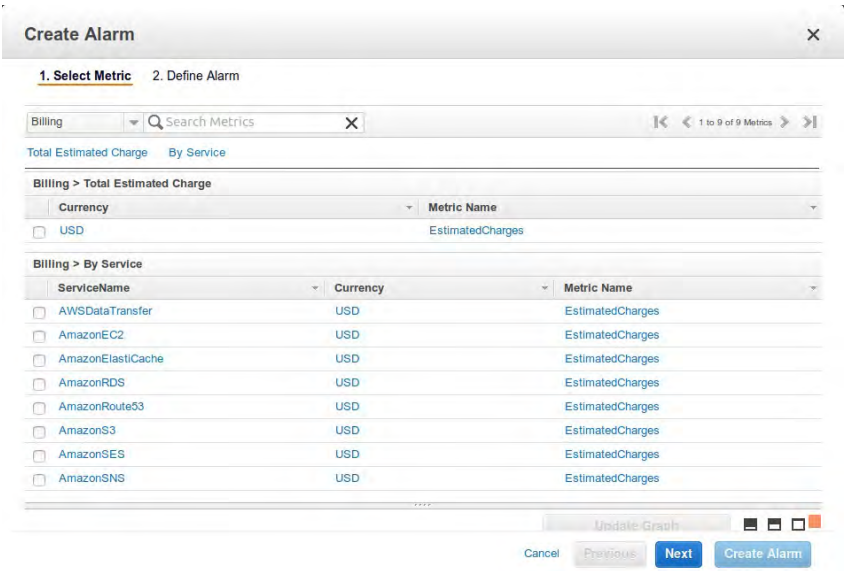


Figura 13.11: CloudWatch — Alarme de gastos da conta

13.4 CONSIDERAÇÕES FINAIS

Viu como é fácil configurar e utilizar o CloudWatch para ser notificado quando alguma métrica for atingida? Através dos alarmes do CloudWatch, você ganha uma tranquilidade em relação ao que está sendo utilizado de recurso na sua aplicação e consegue mantê-la em pleno funcionamento.

Se algo sair errado, você pode corrigir manualmente, ou mesmo deixar alguma configuração automática para ser realizada.

Links úteis

- **Documentação:** https://aws.amazon.com/pt/cloudwatch/
- **Documentação para desenvolvedores:** https://aws.amazon.com/pt/documentation/cloudwatch/

Capítulo 14

GERENCIAMENTO DE CUSTOS DA CONTA

Agora que você já aprendeu a utilizar diversos serviços da Amazon AWS, chegou a hora de aprender a gerenciar os custos da sua conta. No capítulo anterior, vimos como configurar alertas de monitoramento dos gastos, e neste capítulo veremos mais informações sobre o console de gerenciamento.

Para começarmos, acesse o console de gerenciamento de gastos pelo menu pessoal *Billing & Cost Management*, ou então pelo link https://console.aws.amazon.com/billing.

14.1 DASHBOARD (PAINEL)

Ao acessar o console, o Dashboard é a primeira tela exibida. Nela você terá informações sobre os gastos atuais da conta, gráfico de gastos e tabela dos serviços utilizados que estão dentro do programa de gratuidade.

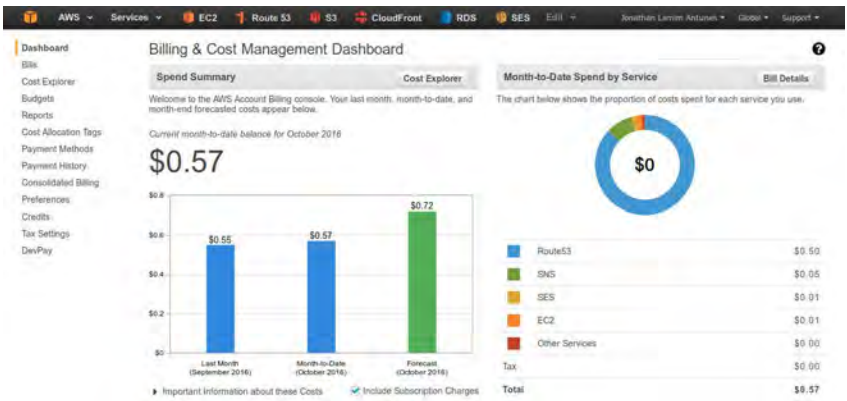


Figura 14.1: Dashboard do console de gerenciamento de gastos

Figura 14.2: Dashboard do console de gerenciamento de gastos

14.2 BILLS (FATURAS)

Acessando o menu *Bills*, você terá acesso às informações da fatura do mês corrente ou de qualquer outro mês que selecionar na parte superior da tela, logo abaixo do título. Poderá ainda exportar essas informações em formato CSV, ou então imprimir.

Na primeira tabela, é exibido um sumário com os custos totais dos serviços da AWS e, na segunda tabela, são exibidos detalhes

desses custos. Para visualizar os detalhes, basta você clicar sobre o serviço que deseja conferir os detalhes.

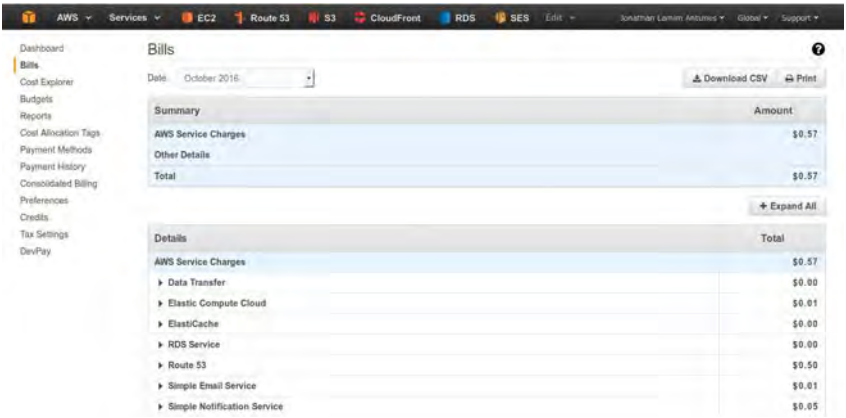


Figura 14.3: Console de gerenciamento de gastos — Bills

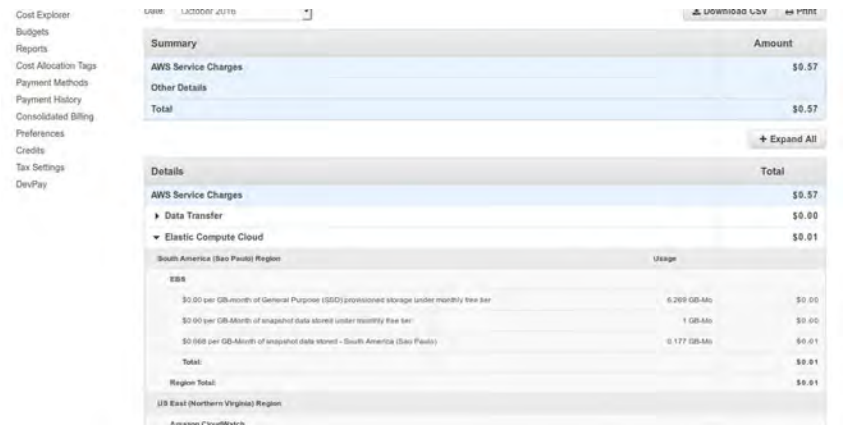


Figura 14.4: Console de gerenciamento de gastos — Bills

14.3 COST EXPLORER (EXPLORADOR DE CUSTOS)

O Cost Explorer é uma opção em que você pode obter relatórios e dados analíticos sobre os gastos com os serviços, e pode ser acessado pela opção *Cost Explorer* no menu lateral. Quando você

cria a conta, esse recurso fica desabilitado. Para habilitá-lo, basta clicar no botão *Enable Cost Explorer*.

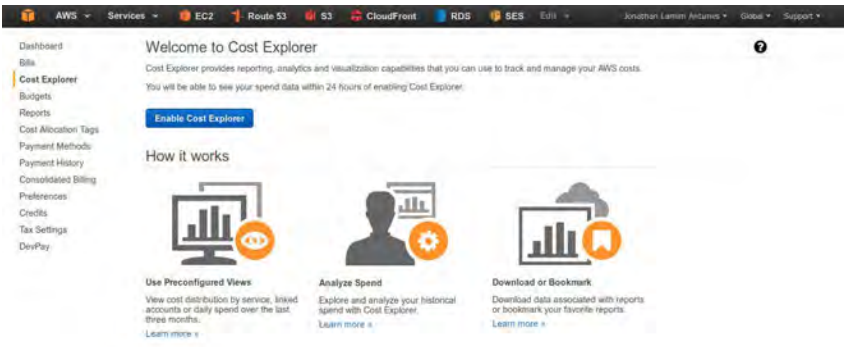


Figura 14.5: Console de gerenciamento de gastos — Cost Explorer

Figura 14.6: Console de gerenciamento de gastos — Cost Explorer

As informações começarão a ser exibidas 24 horas após a ativação e você verá uma tela semelhante à da figura a seguir, na qual aparecem os gráficos e opções para montar o seu relatório, como filtros (filtering), agrupamento (grouping) e período de tempo (time range).

Figura 14.7: Console de gerenciamento de gastos — Cost Explorer

Ao clicar no botão *New Report*, você poderá criar um novo relatório, com as informações que você quiser e estiverem disponíveis. Veja a seguir um relatório montado para verificar os gastos com os serviços do Route 53 durante o ano de 2016.

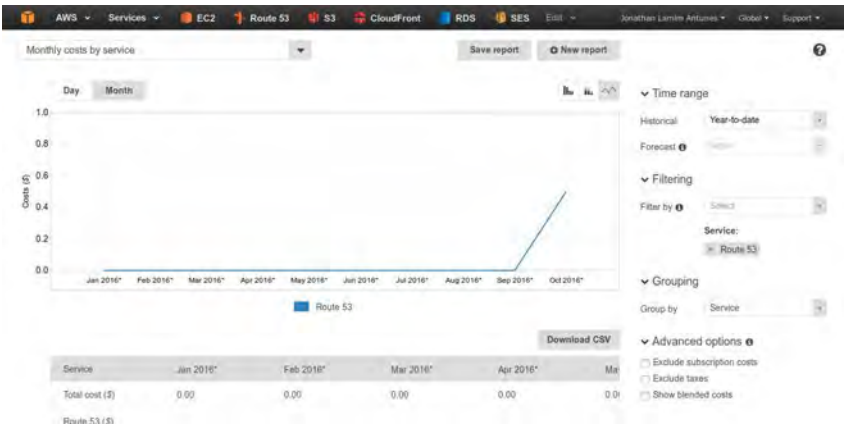


Figura 14.8: Console de gerenciamento de gastos — Cost Explorer

Para conseguir essas informações, o tipo de relatório escolhido foi o **Monthly costs by service**, com o período de tempo **Year-to-date** (ano todo), filtrando somente informações do serviço **Route 53** (filtering) e agrupando por serviço (grouping).

Após criar o relatório, você pode simplesmente fazer o download do arquivo CSV com as informações clicando em *Download CSV*, ou então salvar o relatório para poder analisá-lo posteriormente pelo botão *Save report*.

Após salvar o relatório, você não poderá mais fazer alterações em suas configurações. Se isso for necessário, você deverá excluir o relatório atual e criar um novo.

14.4 BUDGETS (ORÇAMENTOS)

As informações de budget são acessadas pelo menu lateral, na opção *Budget*. Os budgets são o orçamento mensal que você define para a sua conta, tanto a nível de custos gerais quanto para serviços específicos. Junto com os orçamentos, você pode definir notificações por e-mail para o momento em que os valores atingirem um determinado nível.

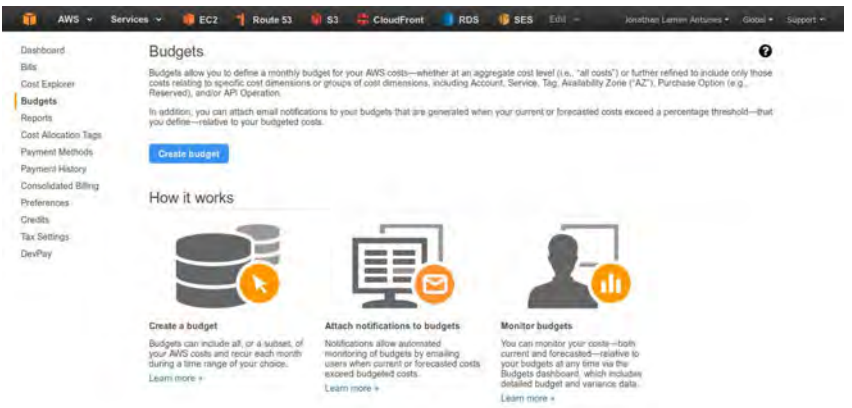


Figura 14.9: Console de gerenciamento de gastos — Budgets

Como você está usando os recursos disponíveis no programa de gratuidade, você não terá gasto quase nenhum. Então, crie um budget para o mês atual com valor de 1 dólar.

Para criar um budget, clique no botão *Create budget* e preencha o formulário com as informações desse budget que está criando.

- **Select cost or usage:** opção onde você define se quer definir o budget por custo ou uso.
- **Name:** nome de identificação do budget.
- **Period:** período de duração do budget.
- **Start date:** data inicial.
- **End date:** data final.
- **Amount:** valor total do budget.
- **Include costs related to:** definição dos custos que devem estar associados a esse budget.

Na sequência do formulário, estão as informações para notificação, que são de preenchimento opcional. Elas são muito semelhantes às informações de notificação que foram utilizadas quando foram criados os alarmes no CloudWatch.

- **Notify me when:** você define quando a notificação deve ser enviada.
- **Email contacts:** lista de e-mails que deve receber a notificação.
- **SNS topic ARN:** código ARN do tópico do SNS que será utilizado para disparar a notificação.

Preenchidos os campos, clique em *Create* para finalizar.

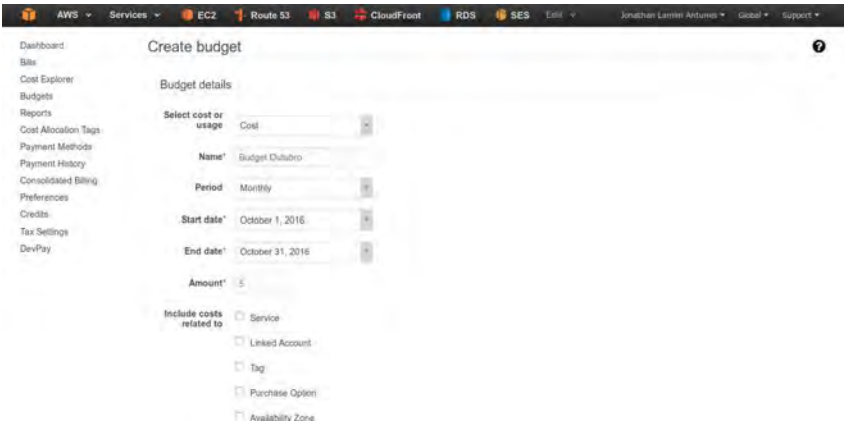


Figura 14.10: Console de gerenciamento de gastos — Budgets

Após criar o budget, você o verá na lista de budgets.

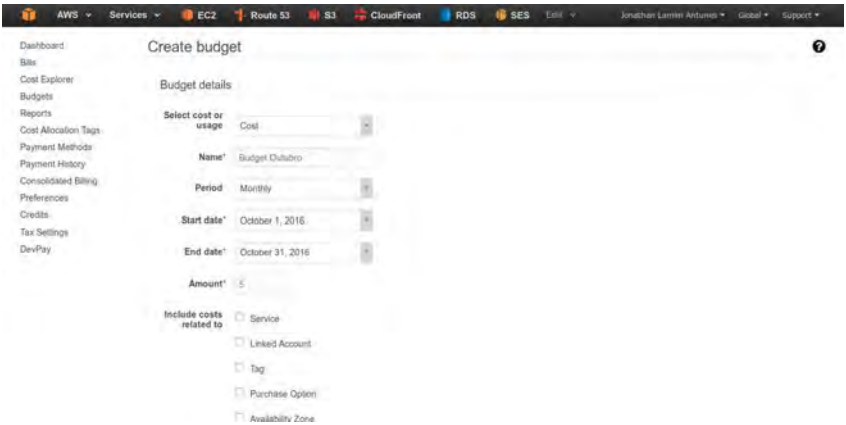


Figura 14.11: Console de gerenciamento de gastos — Budgets

14.5 REPORTS (RELATÓRIOS)

Acessada por meio do menu lateral na opção *Reports*, é onde você pode criar relatórios personalizados para uma melhor análise dos gastos com os serviços da AWS.

Figura 14.12: Console de gerenciamento de gastos — Reports

Na parte superior da tela, tem um botão *Create report*, que permite a criação de relatórios personalizados. Já na parte inferior, existem links para relatórios já montados pela própria Amazon, com informações de uso de instâncias do EC2.

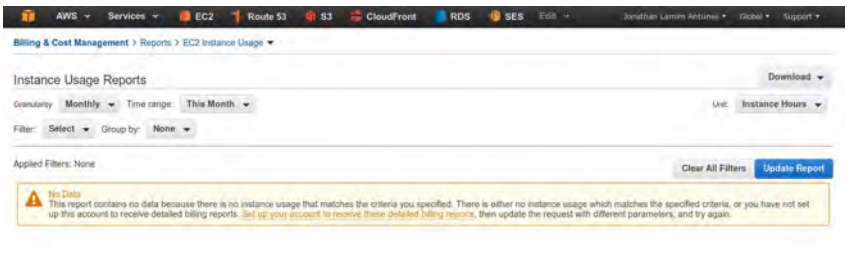


Figura 14.13: Console de gerenciamento de gastos — EC2 Instance Usage Report

Se você optar por criar um relatório personalizado, vai precisar preencher um formulário com informações que vão compor o relatório. Essa criação é feita em três etapas:

1. **Select Content** — Onde você selecionará o conteúdo do relatório.
    - **Report name:** nome do relatório.
    - **Time unit:** unidade de tempo do relatório.
    - **Include:** opção para incluir informações adicionais sobre

os *resource IDs*.

- **Enable support for..:** definição de qual serviço terá suporte.
- **Report includes:** informações sobre o que será incluído no relatório.

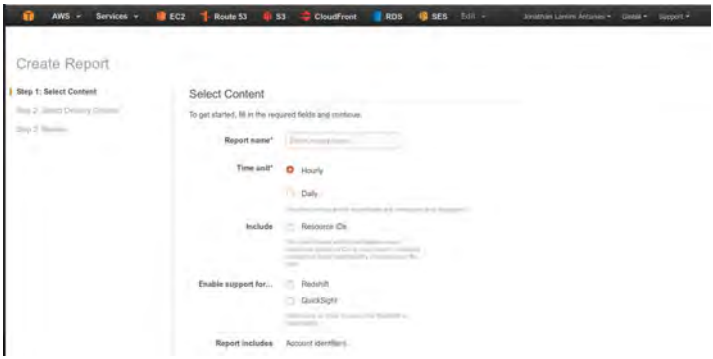


Figura 14.14: Console de gerenciamento de gastos — Relatório personalizado (etapa 1)

2. **Select Delivery Options** — Onde você definirá as opções de entrega do relatório.
    - **S3 Bucket:** nome do bucket do S3 onde o relatório será armazenado. Se não tiver um bucket criado, vá até o S3 e crie um para poder usar no armazenamento dos relatórios.
    - **Report path prefix:** prefixo usado no path do relatório.
    - **Compression:** modo de compressão que deverá ser utilizado na geração do arquivo.

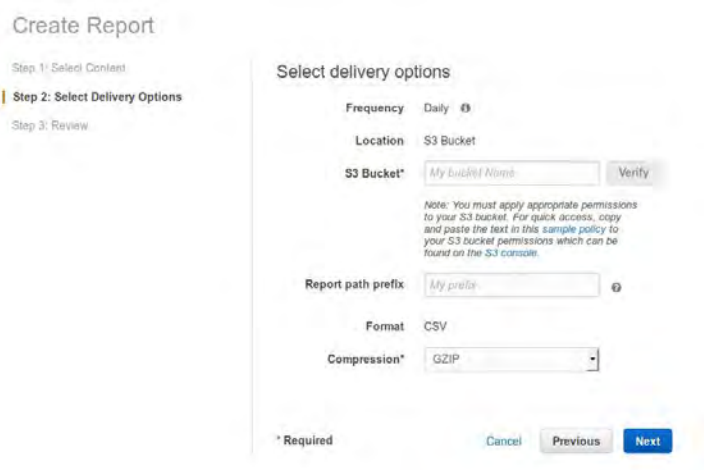


Figura 14.15: Console de gerenciamento de gastos — Relatório personalizado (etapa 2)

3. **Review** — Etapa onde você pode verificar os dados informados nas etapas 1 e 2, e se estiver tudo correto, clicar em *Review and Complete* para concluir.

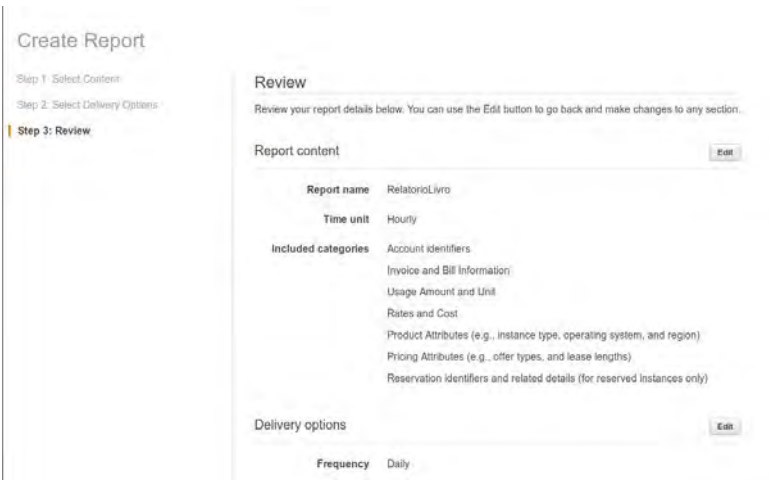


Figura 14.16: Console de gerenciamento de gastos — Relatório personalizado (etapa 3)

Feito isso, você verá os relatórios personalizados na tela principal da opção *Reports*.

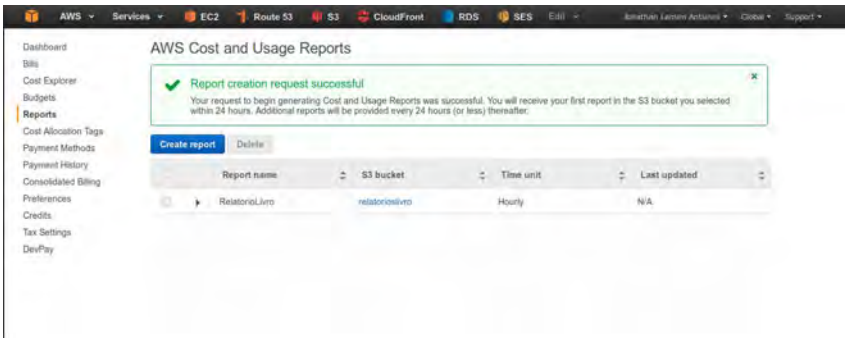


Figura 14.17: Console de gerenciamento de gastos — Reports criados

O primeiro relatório será gerado e disponibilizado em até 24 horas. Você poderá acessá-lo diretamente pelo bucket no S3.

14.6 COST ALLOCATION TAGS(TAGS DE ALOCAÇÃO DE CUSTOS)

Acessado através do menu lateral na opção *Cost Allocation Tags*, esse recurso serve para que você possa usar tags de alocação de custos para categorização e controle de seus custos na AWS. Quando você aplicar tags aos seus recursos AWS (como instâncias do Amazon EC2 ou buckets do Amazon S3) e ativar as tags, a AWS gera um relatório de alocação de custos com o seu uso e os custos agregados por suas tags ativas.

Você pode aplicar marcas que representam categorias de negócios (tais como centros de custo, nomes de aplicativos, ou proprietários) para organizar seus custos em vários serviços.

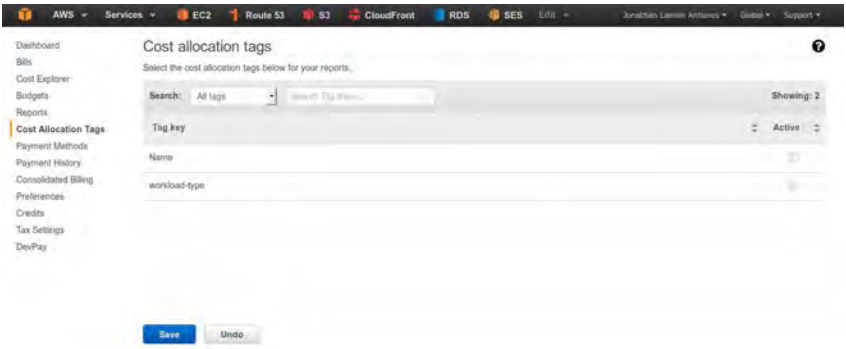


Figura 14.18: Console de gerenciamento de gastos — Cost Allocation Tags

14.7 PAYMENT METHODS (MÉTODOS DE PAGAMENTO)

É a área na qual você configura os métodos de pagamento, que no caso é feito via cartão de crédito. O acesso a essa área é feito pela opção *Payment Methods* no menu lateral.

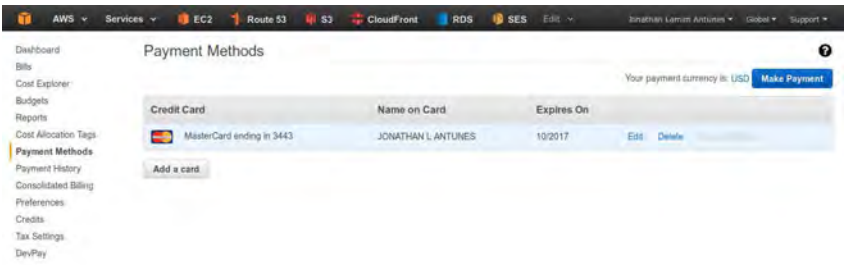


Figura 14.19: Console de gerenciamento de gastos — Payment Methods

Para cadastrar outro cartão, basta clicar no botão *Add a card* e preencher o formulário.

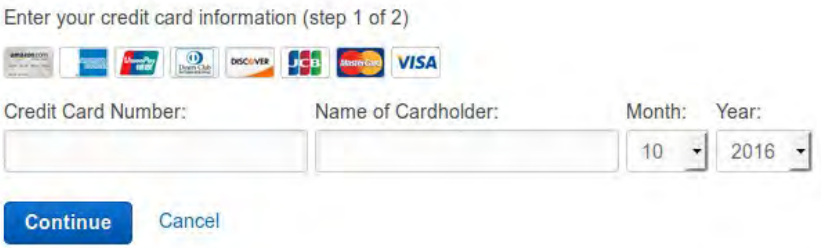


Figura 14.20: Console de gerenciamento de gastos — Payment Methods (credit card)

O botão *Make Payment* vai levar para o histórico de pagamentos, e veremos mais detalhe sobre ele a seguir.

14.8 PAYMENT HISTORY (HISTÓRICO DE PAGAMENTOS)

Nesta seção, você poderá ver todo o histórico de pagamentos que já foram efetuados e filtrá-los por data. Para acessar a seção, use a opção *Payment History* no menu lateral.

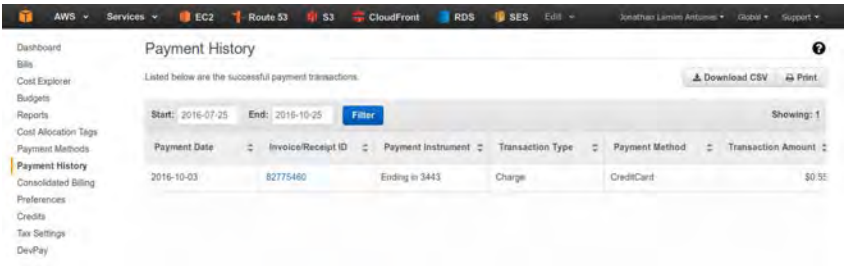


Figura 14.21: Console de gerenciamento de gastos — Payment History

Se clicar no link do Receipt ID, você poderá fazer o download do recibo de pagamento.

14.9 CONSOLIDATED BILLING

(FATURAMENTO CONSOLIDADO)

Se você gerencia diversas contas da AWS, essa opção é bastante interessante, pois ela permite que você receba um relatório de faturamento consolidado, contendo detalhes das contas AWS gerenciadas por você. Para acessar a seção, vá até o menu lateral e clique em *Consolidated Billing*.

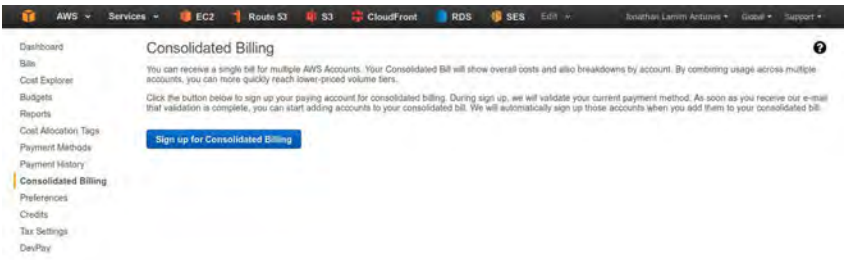


Figura 14.22: Console de gerenciamento de gastos — Consolidated Billing

Você precisa se inscrever para ter acesso aos recursos de faturamento consolidado. Para isso, basta clicar em *Sign up for Consolidated Billing* e aguardar o e-mail de confirmação.

Após a Amazon validar os seus dados, você já poderá adicionar novas contas para esse faturamento consolidado.

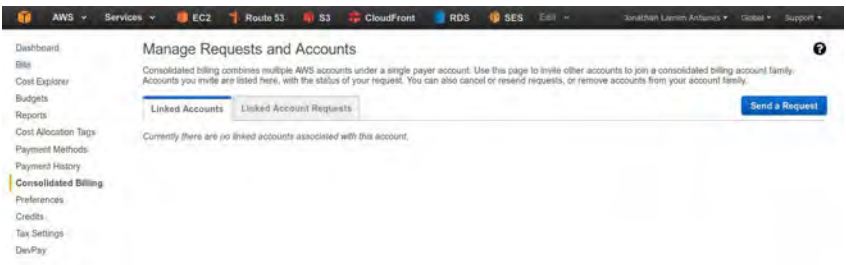


Figura 14.23: Console de gerenciamento de gastos — Consolidated Billing

Para adicionar uma nova conta à lista, basta clicar em *Send a Request* e informar o e-mail do titular da conta. Então, clique em

Send e aguarde até que a requisição seja aceita.

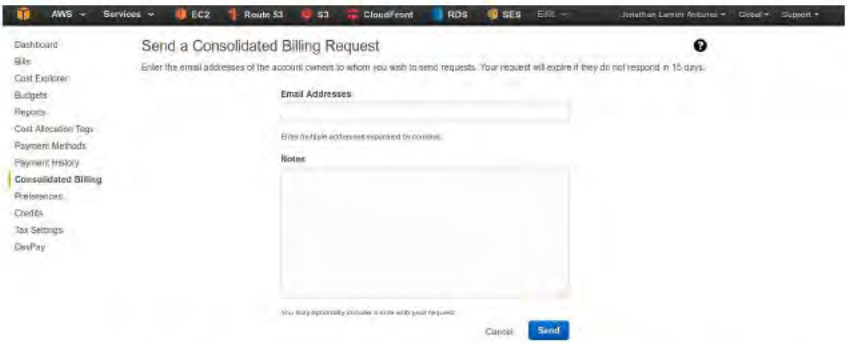


Figura 14.24: Console de gerenciamento de gastos — Consolidated Billing

14.10 PREFERENCES (PREFERÊNCIAS)

Essa tela foi vista no capítulo anterior, mas de forma bem objetiva. Acessada através da opção *Preferences* do menu lateral, ela serve para aplicar ativar/desativar alguns recursos na sua conta.

- **Receive PDF Invoices By Email:** se marcada, você receberá uma cópia, por e-mail, do recibo da fatura de cada mês.
- **Receive Billing Alerts:** se marcada, permite que você crie alertas de monitoramento através do CloudWatch.
- **Receive Billing Reports:** se marcada, você terá à disposição em um bucket do S3 relatórios de faturamento.

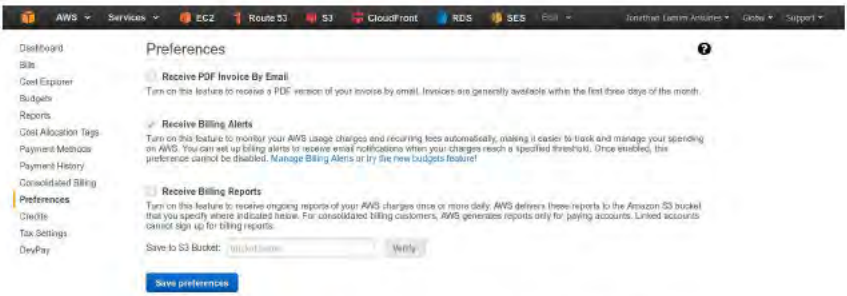


Figura 14.25: Console de gerenciamento de gastos — Preferences

14.11 CREDITS (CRÉDITOS)

Na seção Credits, é onde você poderá fazer uso de promo codes que lhe darão créditos para utilizar em serviços da AWS. Acessada pela opção *Credits* no menu lateral, ela possui 2 campos a serem preenchidos:

- **Promo code:** o campo onde você vai informar o código promocional.
- **Security Check:** que você deve preencher com os caracteres exibidos na imagem logo acima do campo.

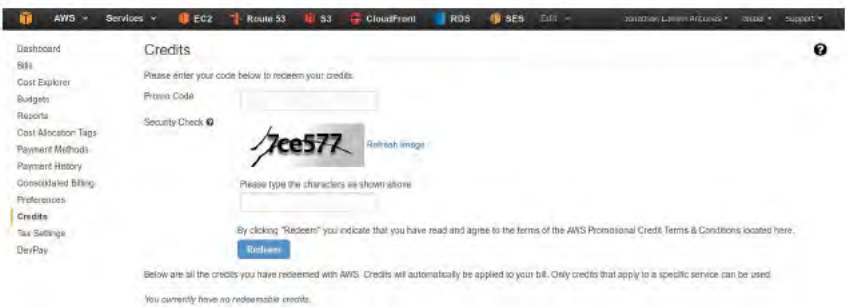


Figura 14.26: Console de gerenciamento de gastos — Credits

14.12 TAX SETTINGS (CONFIGURAÇÕES FISCAIS)

Nesta seção, que pode ser acessada pelo menu lateral pela opção *Tax Settings*, você poderá fazer configurações fiscais da sua conta, e abrir chamados para que sua conta tenha acesso ao serviço pago de suporte da AWS.

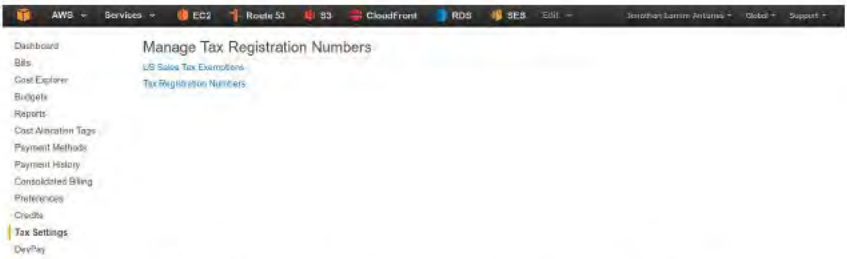


Figura 14.27: Console de gerenciamento de gastos — Tax Settings

14.13 DEVPAY

O Amazon DevPay é um serviço de pagamento da Amazon que facilita a implementação de rotinas de cobrança. Faz uso do Amazon Payments para processar os pagamentos permitindo o uso da infraestrutura de pagamentos confiável da Amazon.

No momento da escrita deste livro, o DevPay não estava aceitando a criação de novas contas. Então não entraremos em detalhes sobre ele.

14.14 CONCLUSÃO

Embora com muitos recursos, o console de gerenciamento de gastos da AWS é bem simples de ser utilizado. Ele permite que você tenha uma visão bastante simplificada e rápida dos gastos da sua conta.

Links úteis

- **Documentação:**

https://aws.amazon.com/pt/documentation/account-billing/

Capítulo 15

APÊNDICE 1 — INSTALANDO O AWS CLI E CONFIGURANDO AS CREDENCIAIS DE ACESSO

O AWS CLI é uma ferramenta de linha de comando para gerenciar os serviços no AWS. Através dela, você pode controlar múltiplos serviços do AWS, sem precisar se logar no console pelo browser.

15.1 INSTALANDO O AWS CLI

O AWS CLI está disponível para Windows, Mac e Linux e o seu processo de instalação é bastante simples.

Windows

Faça o download do instalador conforme a configuração do seu sistema operacional (32 bits ou 64 bits).

- **32 bits:** https://s3.amazonaws.com/aws-cli/AWSCLI32.msi
- **64 bits:** https://s3.amazonaws.com/aws-cli/AWSCLI64.msi

Após o download, execute o instalador seguindo as instruções que ele passar. Ao concluir a instalação, faça um teste verificando a versão que foi instalada. Para realizar o teste, abra o prompt de comando e digite nele o seguinte comando:

```
aws --version
```

O resultado deverá ser algo semelhante a `aws-cli/1.7.36 Python/2.7.9 Windows/7` . Vale lembrar que a versão do Python e do Windows variarão conforme a configuração do seu ambiente.

Mac, Linux ou Unix

Para Mac, Linux ou Unix, vamos utilizar a instalação via linha de comando. Você precisará ter o Python instalado em sua máquina, e ele deve estar em versão superior à 2.6.5 ou 3.3.

A instalação do Python, caso você não o tenha instalado, pode ser feita através do comando a seguir:

```
sudo apt-get install Python
```

Execute no terminal os comandos a seguir para realizar o processo de download, descompactação e instalação, respectivamente:

```
curl "https://s3.amazonaws.com/aws-cli/awscli-bundle.zip" -o "awsc
li-bundle.zip"

unzip awscli-bundle.zip

sudo ./awscli-bundle/install -i /usr/local/aws -b /usr/local/bin/a
ws
```

Após concluir a instalação, faça um teste executando o comando `aws --version` , e veja se o retorno será algo semelhante a `aws-cli/1.10.43 Python/2.7.10 Darwin/15.5.0 botocore/1.4.33` . Vale lembrar que as versões vão variar conforme o seu sistema operacional e o do Python.

15.2 CONFIGURANDO AS CREDENCIAIS DE ACESSO VIA AWS CLI

Antes de começar a configurar, abra o arquivo com as credenciais do usuário que você criou no início do capítulo 2 deste livro, pois nele estão as informações que serão usadas nessa configuração.

Com o arquivo aberto, acesse o terminal (ou prompt de comando no Windows) e execute o comando `aws configure`. Esse comando criará uma configuração padrão para uso da API e também do AWS CLI.

Ao executar, serão solicitadas as seguintes informações:

- **AWS Access Key ID:** chave de acesso constante no arquivo de credenciais;
- **AWS Secret Access Key:** chave secreta constante no arquivo de credenciais;
- **Default region name:** região padrão para uso;
- **Default output format:** formato de retorno dos dados.

> Para a configuração padrão, não preencha a informação para DEFAULT OUTPUT FORMAT; dê apenas um *Enter*.

Feito isso, a credencial padrão estará criada e você poderá executar as operações através da API e do AWS CLI, na qual o usuário configurado possui permissão.

15.3 LINKS ÚTEIS

- **AWS** **CLI:**

http://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html

CAPÍTULO 16

APÊNDICE 2 — LINUX: COMANDOS BÁSICOS

Ao utilizar uma instância com sistema operacional Linux no EC2, você precisará executar alguns comandos para realizar tarefas via terminal, como por exemplo:

- Manipular arquivos e diretórios;
- Alterar permissão de arquivos e diretórios;
- Verificar logs;
- Fazer instalação de pacotes e softwares;
- Fazer download e descompactação de arquivos.

Veja a seguir uma série de links com detalhes sobre cada um dos itens citados.

16.1 MANIPULANDO ARQUIVOS E DIRETÓRIOS

Veja no tutorial do link a seguir comandos fundamentais para uso no Linux:

http://bit.ly/Fundamentos-do-sistema-Linux-comandos-do-Linux

16.2 MANIPULANDO PERMISSÕES

Veja no tutorial do link a seguir como fazer a manipulação de permissões em arquivos usando o comando `chmod` :

http://bit.ly/Utilizando-o-chmod-parar-mudar-permissoes-em-arquivos

16.3 VERIFICANDO LOGS

Veja no tutorial do link a seguir como fazer a verificação de log no Linux:

http://bit.ly/SysLog-Sistema-de-log-do-Linux

16.4 INSTALANDO PACOTES

Veja no tutorial do link a seguir como instalar pacotes `.tar.gz` e `.tar.bz2` no Linux:

http://bit.ly/Instalando-pacotes-tar-gz-e-tar-bz2

16.5 COMPACTAÇÃO E DESCOMPACTAÇÃO DE ARQUIVOS

Veja no tutorial do link a seguir como fazer a compactação de arquivos no Linux:

http://bit.ly/Compactacao-de-Arquivos

Capítulo 17

APÊNDICE 3 — MYSQL: COMANDOS BÁSICOS

Neste apêndice, você verá alguns comandos básicos que lhe auxiliarão no uso dos recursos do MySQL via terminal. Alguns desses comandos você já deve ter visto no capítulo 5, mas a maioria deles não.

> Todas as operações a seguir serão executadas no terminal, então conecte-se à sua instância via SSH.

17.1 INSTALANDO O MYSQL

Para instalar o MySQL, é preciso poucos comandos. O primeiro deles é `sudo yum -y install mysql-server` , e o segundo é `sudo yum -y install mysql` .

O `mysql-server` é o servidor do banco de dados, e o `mysql` cliente que permitirá manipular o banco de dados pela linha de comando. Após executar esses 2 comandos, você deve executar `sudo service mysqld start` , que é o responsável por inicializar o `mysql` na instância.

É padrão do MySQL não definir uma senha para o usuário

`root` , mas, por questões de segurança, você deve defini-la. Para isso, execute o comando `mysqladmin -u root password` . Será solicitada uma senha, e em seguida que você repita essa senha.

Pronto, o MySQL está instalado e com senha para o usuário `root` definida.

17.2 CONECTANDO O MYSQL VIA TERMINAL

Para se conectar ao MySQL pelo terminal, execute o comando `mysql -u root -p` e, em seguida, informe a senha. Sua tela no terminal deverá ficar semelhante ao apresentado a seguir:

```
$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14
Server version: 5.5.46 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights
 reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current inpu
t statement.

mysql>
```

A partir desse momento, você está conectado ao MySQL e pode gerenciar os seus bancos de dados diretamente pelo terminal.

> Não se esqueça que todo comando do MySQL que for executar precisa conter um ";" no final, caso contrário não será executado.

17.3 CRIANDO UM BANCO DE DADOS E UM USUÁRIO

Ao criar um banco de dados, é preciso informar qual usuário terá acesso e quais serão as permissões. É recomendado que não seja usado o usuário `root` para acesso ao banco de dados a partir de uma aplicação, então os comandos a seguir criarão um novo banco de dados e um novo usuário para acessá-lo.

O código para criar o banco de dados:

```
create database livro;
```

Agora definir os privilégios e o usuário:

```
grant all privileges on livroaws.* to 'livroaws-user'@'localhost'
identified by 'senha_do_usuario';

flush privileges;

exit;
```

Pronto, o banco de dados `livroaws` foi criado e o usuário `livroaws-user` também. E pelo comando `exit;` , você se desconectou do cliente MySQL no terminal.

17.4 MANIPULANDO O BANCO DE DADOS

Conecte-se ao banco de dados `livroaws` com o usuário que acabou de criar, o `livroaws-user` . Para isso, utilize o comando `mysql -u livroaws-user -p` e informe a senha que você criou para esse usuário.

Após se conectar, execute o comando `show databases;` , para exibir todos os bancos de dados criados até o momento. Você deverá ter um retorno parecido com o apresentado a seguir:

```
show databases;
+--------------------+
```

```
| Database           |
+--------------------+
| information_schema |
| livroaws           |
| mysql              |
| performance_schema |
| wordpress-db       |
+--------------------+
5 rows in set (0.00 sec)
```

Execute o comando `connect livroaws` para se conectar o banco de dados `livroaws` e poder executar operações e manipular os dados dentro dele. Após executar o comando, deverá ver algo como apresentado a seguir:

```
Connection id:    15
Current database: livroaws
```

A partir desse momento, você já pode rodar queries para criação de tabelas, pesquisa de dados, entre outras, dentro do banco `livroaws`.

Criando uma tabela dentro do banco de dados

Para criar uma tabela dentro do banco de dados `livroaws`, você deverá executar o comando a seguir. Ele vai criar uma tabela chamada `capitulos`, que conterá os campos `id` e `titulo`.

```
create table capitulos (id integer auto_increment, titulo varchar(
100), primary key(id));
```

Listando as tabelas de um banco de dados

Para listar as tabelas de um banco de dados, você deve executar o comando `show tables;`. O retorno deverá ser o seguinte:

```
+--------------------+
| Tables_in_livroaws |
+--------------------+
| capitulos          |
+--------------------+
```

Exibindo detalhes de uma tabela

Para exibir os detalhes de uma tabela, ou seja, a sua estrutura, execute o comando `describe capitulos` .

```
+--------+-------------+------+-----+---------+----------------+
| Field  | Type        | Null | Key | Default | Extra          |
+--------+-------------+------+-----+---------+----------------+
| id     | int(11)     | NO   | PRI | NULL    | auto_increment |
| titulo | varchar(100)| YES  |     | NULL    |                |
+--------+-------------+------+-----+---------+----------------+
```

Inserindo um registro na tabela `capitulos`

Para inserir um registro, você deverá utilizar o comando a seguir:

```
insert into capitulos (titulo) values ('Livro AWS - Amazon EC2');
```

Listando registros

Para listar os registros, execute o comando `select` , como apresentado a seguir:

```
select * from capitulos;
```

O retorno deverá ser:

```
+----+-----------------------+
| id | titulo                |
+----+-----------------------+
|  1 | Livro AWS - Amazon EC2 |
+----+-----------------------+
1 row in set (0.00 sec)
```

Esses são apenas alguns comandos básicos do MySQL que você pode utilizar no terminal. Consulte a documentação do MySQL para obter a lista completa de comandos, em http://dev.mysql.com/doc/refman/5.7/en/mysql.html

Capítulo 18

APÊNDICE 4 — CONFIGURAÇÃO DO APACHE PARA UTILIZAR URLS AMIGÁVEIS

Sabemos que o uso de URLs amigáveis é muito importante para a indexação dos sites em motores de busca. Por isso, vamos ver neste apêndice como fazer essa configuração.

É um processo rápido e fácil de ser executado, e você precisará apenas estar logado via SSH à instância EC2 na qual deseja aplicar a configuração.

18.1 ACESSANDO A INSTÂNCIA VIA SSH

Para acessar a sua instância via SSH, você pode usar a linha de comando a seguir, substituindo *nome-da-chave* pelo nome do seu arquivo que contém a chave de acesso ao servidor, e também substituindo *endpoint-da-instancia* pelo endereço do endpoint da instância.

Para executar a linha de comando, basta pressionar *Enter* após digitá-la.

```
ssh -i "nome-da-chave.pem" ec2-user@endpoint-da-instancia
```

18.2 CRIANDO UM ARQUIVO PARA TESTE

Antes de iniciar o processo de configuração, vamos criar um arquivo que nos auxiliará na verificação do status do `mod_rewrite` e no teste de funcionamento das URLs amigáveis.

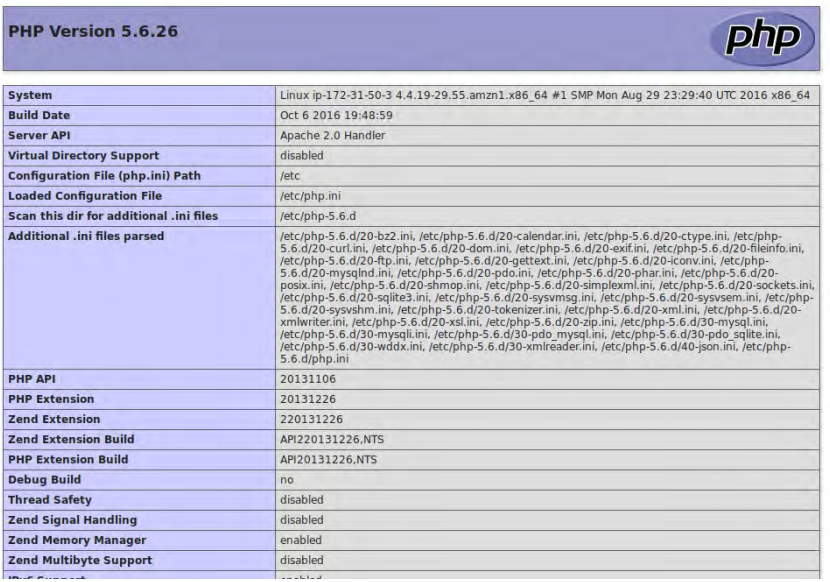
Para criar o arquivo `phpinfo.php` , execute o comando a seguir no terminal. Ele vai criar um arquivo no diretório `/var/www/html` contendo a função PHP para exibir na tela as configurações do servidor.

```
echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

Feito isso, acesse o endpoint da sua instância pelo browser conforme mostrado a seguir:

```
http://endpoint-da-instancia/phpinfo.php
```

O resultado deverá ser algo semelhante a:



PHP Version 5.6.26

System	Linux ip-172-31-50-3 4.4.19-29.55.amzn1.x86_64 #1 SMP Mon Aug 29 23:29:40 UTC 2016 x86_64
Build Date	Oct 6 2016 19:48:59
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php-5.6.d
Additional .ini files parsed	/etc/php-5.6.d/20-bz2.ini, /etc/php-5.6.d/20-calendar.ini, /etc/php-5.6.d/20-ctype.ini, /etc/php-5.6.d/20-curl.ini, /etc/php-5.6.d/20-dom.ini, /etc/php-5.6.d/20-exif.ini, /etc/php-5.6.d/20-fileinfo.ini, /etc/php-5.6.d/20-ftp.ini, /etc/php-5.6.d/20-gettext.ini, /etc/php-5.6.d/20-iconv.ini, /etc/php-5.6.d/20-mysqlnd.ini, /etc/php-5.6.d/20-pdo.ini, /etc/php-5.6.d/20-phar.ini, /etc/php-5.6.d/20-posix.ini, /etc/php-5.6.d/20-shmop.ini, /etc/php-5.6.d/20-simplexml.ini, /etc/php-5.6.d/20-sockets.ini, /etc/php-5.6.d/20-sqlite3.ini, /etc/php-5.6.d/20-sysvmsg.ini, /etc/php-5.6.d/20-sysvsem.ini, /etc/php-5.6.d/20-sysvshm.ini, /etc/php-5.6.d/20-tokenizer.ini, /etc/php-5.6.d/20-xml.ini, /etc/php-5.6.d/20-xmlwriter.ini, /etc/php-5.6.d/20-xsl.ini, /etc/php-5.6.d/20-zip.ini, /etc/php-5.6.d/30-mysql.ini, /etc/php-5.6.d/30-mysqli.ini, /etc/php-5.6.d/30-pdo_mysql.ini, /etc/php-5.6.d/30-pdo_sqlite.ini, /etc/php-5.6.d/30-wddx.ini, /etc/php-5.6.d/30-xmlreader.ini, /etc/php-5.6.d/40-json.ini, /etc/php-5.6.d/php.ini
PHP API	20131106
PHP Extension	20131226
Zend Extension	220131226
Zend Extension Build	API220131226,NTS
PHP Extension Build	API20131226,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled

Figura 18.1: Resultado da execução de phpinfo()

18.3 ATIVANDO O MOD_REWRITE DO APACHE

Para ativar o `mod_rewrite` do Apache, você deverá executar o comando a seguir no terminal:

```
sudo a2enmode
```

Para verificar se o `mod_rewrite` foi realmente ativado, volte ao browser, na tela com as informações do servidor, e procure por `mod_rewrite` . Se ele estiver ativo, serão exibidas informações sobre ele.

18.4 AJUSTANDO AS CONFIGURAÇÕES

Após ativar o `mod_rewrite` , é preciso que você faça algumas alterações no arquivo de configuração do Apache para que o servidor responda corretamente às requisições feitas através de URLs amigáveis.

Para isso, abra o arquivo `httpd.conf` , conforme linha de comando a seguir:

```
sudo nano /etc/httpd/conf/httpd.conf
```

Ao abrir o arquivo, localize a linha `<Directory "/var/www/html">` e então localize a configuração `AllowOverride None` que aparece na sequência. Substitua `None` por `All` , e salve o arquivo de configuração.

Para aplicar essa alteração nas configurações, é necessário reiniciar o Apache. Então execute a linha de comando a seguir:

```
sudo service httpd restart
```

Pronto, as configurações foram executadas e agora é hora de testar.

18.5 TESTANDO AS URLS AMIGÁVEIS

Para testar o funcionamento das URLs amigáveis, vamos precisar criar um arquivo `.htaccess`. Execute a sequência de comandos a seguir para acessar o diretório `www` no servidor, e criar o arquivo `.htaccess`.

```
cd /var/www/html
sudo nano .htaccess
```

Após executar o segundo comando, a tela de edição do arquivo será aberta no terminal. Então você deverá colocar o seguinte código no arquivo `.htaccess`:

```
RewriteEngine on
RewriteBase /
RewriteRule ^phpinfo$ phpinfo.php [L]
```

Esse código permitirá que o arquivo `phpinfo.php` seja executado sem a necessidade de informar sua extensão, apenas utilizando `phpinfo` junto do endpoint da sua instância.

```
http://endpoint-da-instancia/phpinfo
```

> A configuração do arquivo `.htaccess` vai variar conforme sua aplicação. Essa que utilizamos anteriormente é básica, somente para testarmos o funcionamento das configurações aplicadas.