

CLOUD FUNDAMENTALS, ADMINISTRATION AND SOLUTION ARCHITECT

# AMAZON WEB services (AWS)

HENRIQUE POYATOS



04

## LISTA DE FIGURAS

Figura 4.1 – <i>Magic Quadrant for Cloud Infrastructure as a Service, Worldwide</i> .....	7
Figura 4.2 – Console de gerenciamento da AWS .....	10
Figura 4.3 – Painel do <i>Trusted Advisor</i> .....	16
Figura 4.4 – Console de gerenciamento da AWS .....	18
Figura 4.5 – Acesso aos grupos de recursos .....	19
Figura 4.6 – Sumário da situação atual dos serviços EC2 em uma determinada região .....	20
Figura 4.7 – Execução de uma instância EC2.....	20
Figura 4.8 – Passo 1, escolhendo de uma <i>Amazon Machine Image</i> (AMI) .....	21
Figura 4.9 – Passo 2, escolhendo de um tipo de instância EC2 .....	22
Figura 4.10 – Passo 3, configurando detalhes da instância EC2 .....	23
Figura 4.11 – Passo 4, adicionando dispositivo de armazenamento na instância EC2 .....	24
Figura 4.12 – Passo 5, escolhendo <i>tags</i> para a instância EC2 .....	25
Figura 4.13 – Passo 6, configurando o grupo de segurança da instância EC2 .....	25
Figura 4.14 – Passo 7, revisando antes de subir a instância EC2 .....	26
Figura 4.15 – Criando ou configurando o par de chaves para acesso SSH na instância EC2 .....	27
Figura 4.16 – Instância EC2 em execução.....	28
Figura 4.17 – Lista de instâncias EC2 na AWS.....	28
Figura 4.18 – Acesso SSH à instância EC2 utilizando o par de chaves.....	29
Figura 4.19 – Derrubando uma instância EC2 no console da AWS .....	29
Figura 4.20 – Identify and Access Management (IAM) .....	30
Figura 4.21 – Criação de usuário no IAM .....	31
Figura 4.22 – Criando usuário no IAM para uso em API, CLI e SDK .....	31
Figura 4.23 – Criação de grupo no IAM.....	32
Figura 4.24 – Criação do grupo 'CLIUsers' no IAM .....	32
Figura 4.25 – Finalização do grupo 'CLIUsers' no IAM .....	33
Figura 4.26 – <i>Tags</i> para o usuário 'hpoyatos' no IAM .....	33
Figura 4.27 – Resumo da criação do usuário no IAM.....	34
Figura 4.28 – Usuário criado com ID e chave de acesso no IAM.....	34
Figura 4.29 – Comando <i>aws configure</i> .....	35
Figura 4.30 – Anexar política ao grupo 'CLIUsers' .....	36
Figura 4.31 – Comando <i>aws ec2 describe-instances</i> .....	37
Figura 4.32 – Comando <i>aws ec2 start-instances</i> .....	37
Figura 4.33 – Comando <i>aws ec2 stop-instances</i> .....	38

## LISTA DE QUADROS

Quadro 4.1 – Tipos de instâncias e seus casos de uso .....	22
Quadro 4.2 – Tipos de EBS e seus casos de uso .....	23

EMANIP

**LISTA DE COMANDOS DE PROMPT DO SISTEMA OPERACIONAL**

Comando de prompt 4.1 – Comando para configurar profile no AWS CLI .....	35
Comando de prompt 4.2 – Comando para descrever as instâncias EC2 no AWS CLI .....	36
Comando de prompt 4.3 – Comando para iniciar instâncias EC2 no AWS CLI .....	37
Comando de prompt 4.4 – Comando para derrubar instâncias EC2 no AWS CLI ....	38

## SUMÁRIO

4 AMAZON WEB SERVICES (AWS) .....	6
4.1. Por que aprender AWS? .....	6
4.2. Certificações AWS.....	8
4.3. Alcance global da AWS .....	9
4.4. Tipos de interface com a AWS .....	10
4.5. <i>Core Design Architecture</i> .....	11
4.5.1. Princípios gerais .....	12
4.5.2. Excelência baseada nos cinco pilares.....	13
4.6. <i>Trust Advisor</i> .....	15
4.7. Serviços principais.....	16
4.8. <i>Hands on: Elastic Compute Cloud (EC2)</i> .....	18
4.8.1. Criando uma instância EC2.....	18
4.8.2. Automação de instância EC2 usando AWS CLI .....	30
CONCLUSÃO.....	39
REFERÊNCIAS .....	40

## 4 AMAZON WEB SERVICES (AWS)

Não é apenas de venda de livros que vive a Amazon! Brincadeiras à parte, faz muito tempo que a livraria virtual de Jeff Bezos não depende mais do hábito de leitura das pessoas para pagar por suas despesas de divórcio (ok, agora eu paro).

Na virada do novo milênio, a Amazon.com já era um dos maiores *websites* de comércio eletrônico do mundo, contando com uma quantidade astronômica de acessos por segundo e uma única falha poderia representar milhões de dólares em prejuízo por vendas não realizadas. Para ilustrar, em 2013 a Amazon.com ficou meros 40 minutos fora do ar e o prejuízo estimado foi de quase 5 milhões de dólares (TWENEY, 2013).

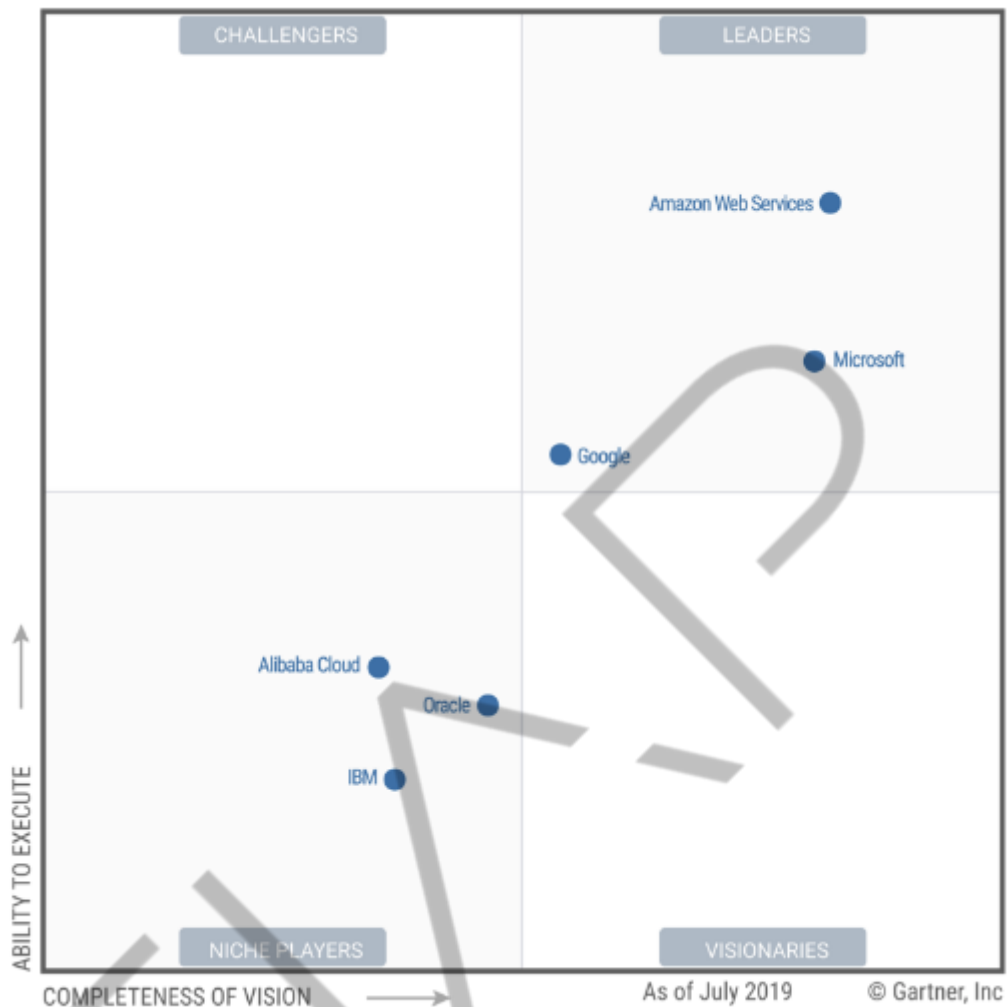
Por esse motivo, desde seus primórdios, a empresa investiu em uma sólida infraestrutura tecnológica, com alta disponibilidade, balanceamento de carga, redundância física, políticas de *disaster and recovery*. Além disso, desde muito cedo soube tirar proveito de seus dados, observando a análise de tráfego de seu website, dimensionando adequadamente sua infraestrutura para grandes eventos de compras, como a *Black Friday*.

Não demorou muito para a empresa perceber que essa grande infraestrutura e expertise também poderia se tornar um produto, e a plataforma AWS foi lançada em julho de 2002, embora tenha sido oficialmente relançada em março de 2006, mais parecida com os modelos de negócio e produtos que temos hoje em dia.

### 4.1. Por que aprender AWS?

- Embora improvável, talvez você esteja procurando razões pelas quais deva aprender AWS; e se for o caso, seguem alguns fortes motivos: em primeiro lugar, segundo a Canalys (2019), **praticamente um terço do mercado global de *cloud computing*** pertence hoje à Amazon, análise embasada aqui pela Gartner (2019), que, em seu **quadrante mágico do setor de *cloud computing***, aponta a Amazon como líder absoluta desse mercado.

Figure 1. Magic Quadrant for Cloud Infrastructure as a Service, Worldwide



Source: Gartner (July 2019)

Figura 4.1 – Magic Quadrant for Cloud Infrastructure as a Service, Worldwide

Fonte: Gartner (2019)

Como razão adicional, podemos destacar que profissionais que trabalham com *cloud computing*, e especialmente AWS, têm um grande impulso em suas carreiras em termos de salário: no relatório “IT Skills and Salary Report”, da Global Knowledge (2019), é possível observar que as certificações AWS aparecem com as mais apreciadas do mercado, resultando em um aumento significativo na remuneração de profissionais no mundo todo. Em 2020 tivemos destaque para o crescimento de demanda e valorização dos profissionais de cybersecurity, segundo Goodison (2020); em 2021, a tendência é a mesma.

## 4.2. Certificações AWS

Sim, existem várias certificações AWS, como duas para arquiteto de soluções (AWS Certified Solutions Architect Professional e AWS Certified Solutions Architect Associate), duas indicadas para o administrador de sistemas (AWS Certified SysOps Administrator Associate e AWS Certified DevOps Engineer Professional) e duas indicadas para desenvolvedores (AWS Certified Developer Associate e AWS Certified DevOps Engineer Professional), além de outras bem especializadas, como o desenvolvedor de habilidades para Alexa (AWS Certified Alexa Skill Builder Specialty) ou para especialistas em segurança (AWS Certified Security Specialty) (AWS, s.d., a)

Contudo, não convém colocar a carruagem na frente dos bois, comecemos pelo início: a certificação **AWS CCP Certified Cloud Practitioner** é considerada fundamental e destina-se a pessoas com conhecimentos e qualificações necessários para demonstrar eficazmente uma compreensão geral da Nuvem AWS, independentemente de funções técnicas específicas abordadas por outras AWS *certifications*. O interessante é que esse exame pode ser feito tanto em um centro de testes como no conforto e na segurança sanitária de sua residência por meio de um exame supervisionado on-line.

Segundo AWS (s.d., b), estas são as habilidades validadas pela certificação:

- Definir o que é a AWS Cloud e sua infraestrutura básica global.
- Descrever os princípios básicos da arquitetura da AWS Cloud.
- Descrever a proposta de valor da AWS Cloud.
- Descrever os principais serviços na plataforma da AWS e os casos de uso comuns.
- Descrever os aspectos básicos de segurança e conformidade dos produtos e serviços da AWS e o modelo de segurança compartilhada.
- Definir os modelos de faturamento, gerenciamento de contas e definição de preço.
- Identificar fontes de documentação ou assistência técnica (por exemplo, *whitepapers* ou tíquetes de suporte).



- Descrever as características básicas/principais de implantação e operação na Nuvem AWS.

A Amazon ainda recomenda que os candidatos ao exame possuam pelo menos seis meses de experiência com a AWS Cloud em qualquer função, inclusive nas áreas técnicas, gerenciais e financeiras, bem como de vendas e compras, isso em razão de algumas perguntas que caem no exame envolvendo muito o contexto profissional e qual seria a tomada de decisões mais adequada. O exame está disponível em bahasa, inglês, japonês, coreano e chinês simplificado, possui 65 perguntas de múltipla escolha e múltiplas respostas que devem ser respondidas em 90 minutos, custando atualmente cem dólares (AWS, s.d., b).

Para encorajá-lo a adquirir o AWS CCP, este capítulo foi estruturado de modo a abordar alguns dos conhecimentos que são exigidos nesse exame. Contudo, não tome este material como fonte única de informação; muito estudo, prática e simulados serão necessários para otimizar seu desempenho no exame. Bons estudos!

### 4.3. Alcance global da AWS

Segundo a AWS (s.d., d), o serviço hoje abrange globalmente 24 regiões, e cada uma delas é composta de dois ou mais *datacenters* (chamados “zonas de disponibilidade”). Na América do Sul, desde 2011 São Paulo é uma dessas regiões. O serviço conta com 77 zonas de disponibilidade, que é a quantidade de *datacenters* presentes em cada região; em São Paulo, por exemplo, há três zonas de disponibilidade (ou três *datacenters*). A estrutura é expandida com as chamadas localidades de borda ou pontos de presença, totalizando 216 deles. Um mapa interativo está disponível em: <<https://www.infrastructure.aws/>>.

Conforme já mencionado, existem no mínimo duas zonas de disponibilidade (*datacenters*) em cada região para garantir desempenho, tolerância a falhas e *disaster recovery*. Redundância é tudo!

Essas zonas de disponibilidade são importantes tanto do ponto de vista tecnológico, diminuindo a latência de rede ao hospedar os dados mais próximos dos usuários, quanto do ponto de vista legal. Para citar um exemplo real, existem leis brasileiras que impedem a hospedagem de dados financeiros fora do país. Uma

instituição financeira pode optar por um modelo em *cloud computing*, usufruindo dos benefícios de ser um cliente AWS, desde que se restrinja à região de São Paulo, cumprindo assim as obrigações legais inerentes dessa área de negócios.

Contudo, é importante observar que nem todos os serviços estão disponíveis em todas as regiões. Algumas delas são muito novas, outras podem não ter demanda para determinados serviços, logo há diferentes serviços disponíveis nas regiões AWS.

Falando em serviços, são dezenas deles. A AWS considera quatro seus *core services*, ou serviços principais: computação, armazenamento, redes e entrega de conteúdo e banco de dados.

#### 4.4. Tipos de interface com a AWS

Existem várias formas de interagir com os serviços e a infraestrutura providos pela AWS. A principal maneira, e que é utilizada inicialmente por todas as empresas, é o **AWS Management Console**, um console que pode ser acessado por meio de um navegador web ou um *mobile app* (Figura “Console de gerenciamento da AWS”).

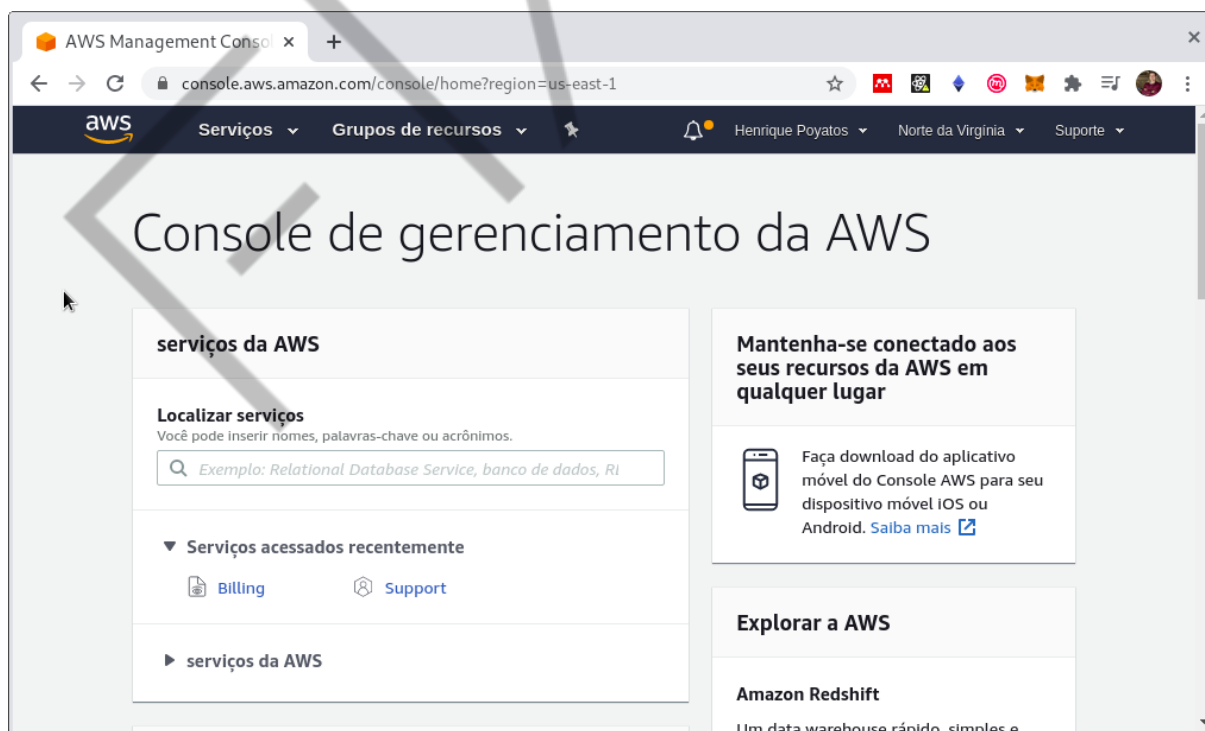


Figura 4.2 – Console de gerenciamento da AWS

Fonte: Elaborado pelo autor (2020)

Contudo, conforme a operação da empresa se torna mais robusta e passe a exibir mais elasticidade e flexibilidade (como alocação dinâmica de recursos), inevitavelmente serão necessárias outras interações que permitam uma maior automação. Uma delas é a **AWS Command Line Interface (CLI)**, um acesso feito por linha de comando, que é independente de linguagem de programação e permite automação por meio de scripts. Outra maneira é o **AWS Software Development Kit (SDK)**, que suporta diversas linguagens de programação (como Python) e permite integração com sistemas preexistentes.

Confira, ao final deste capítulo, um *hands on* no qual utilizamos o CLI para subir e derrubar uma instância no **Elastic Compute Cloud (EC2)**, o principal serviço da AWS.

#### 4.5. Core Design Architecture

Muitos equívocos são cometidos por empresas e profissionais inexperientes em *cloud computing*: podem ser infraestruturas que não primam pela segurança das informações ali suportadas, ou infraestruturas mal dimensionadas que subestimam o volume de acessos de seus usuários, ou o contrário: infraestruturas superestimadas que gastam recursos desnecessários, dando à empresa uma equivocada ideia de que *cloud computing* é algo muito caro (e que o ideal seria manter uma infraestrutura própria).

Tendo isso em mente, a empresa publica e atualiza regularmente o **AWS Well-Architected**, um guia de boas práticas (na verdade, um *whitepaper*) que auxilia arquitetos de nuvem a criar infraestruturas seguras, resilientes, eficientes e de alta performance para aplicações e cargas de trabalho (AWS, 2020). O documento está disponível em PDF (consulte as referências no fim do capítulo) ou gratuitamente em sua versão Kindle (apenas em inglês).

Baseado em cinco pilares (excelência operacional, segurança, confiabilidade, eficiência de performance e otimização de custos), ele fornece uma abordagem consistente para que clientes e parceiros avaliem arquiteturas e implementem designs que possam se expandir com o tempo. Este que começou como um único *whitepaper*

hoje é um framework que conta com [laboratórios práticos](#) e uma ferramenta de avaliação de aderência, o [AWS Well-Architected Tool](#).

#### 4.5.1. Princípios gerais

Vamos a alguns princípios gerais, segundo a AWS (2020):

**Pare de adivinhar suas necessidades de capacidade:** se você tomar uma decisão de baixa capacidade ao implantar uma carga de trabalho, poderá acabar ficando com recursos ociosos caros ou terá que lidar com as implicações de desempenho da capacidade limitada. Graças à *cloud computing* e sua capacidade elástica, isso não precisa mais ser um problema: você pode usar a quantidade necessária, ou a capacidade necessária, e dimensionar para cima e para baixo automaticamente (CLI, SDK e inteligência de negócios estão aí para isso!).

**Sistemas de teste na escala de produção:** em *cloud*, você pode criar uma escala de produção ambiente de teste sob demanda, conclua seu teste e desative os recursos! Como você paga apenas pelo ambiente de teste quando está em execução, pode simular seu ambiente ativo por uma fração do custo dos testes no local. Simule sempre com o tamanho real!

**Automatize para facilitar a experimentação arquitetônica:** a automação permite criar e replicar suas cargas de trabalho a baixo custo e evitar a despesa do esforço manual. Você pode acompanhar as alterações em sua automação, auditar o impacto e reverter para parâmetros anteriores quando necessário. (Parece que as aulas de *shell script* e linguagem de programação ganharam uma nova dimensão agora, não?)

**Permitir arquiteturas evolutivas:** em um ambiente tradicional, as decisões de arquitetura são frequentemente implementadas como eventos estáticos e únicos, com algumas versões principais de um sistema durante sua vida útil. Como a empresa e seu contexto continuam evoluindo, essas decisões iniciais podem prejudicar a capacidade de fornecer requisitos de negócios variáveis. Em *cloud*, a capacidade de automatizar e testar sob demanda reduz o risco de impacto das alterações no projeto e isso permite que os sistemas evoluam ao longo do tempo, para que as empresas

possam tirar proveito das inovações como uma prática padrão. Negócio e infraestrutura crescem juntos, de “mãos dadas”!

**Impulsione arquiteturas usando dados:** na *cloud*, você pode coletar dados sobre **como suas escolhas arquiteturais** afetam o comportamento de sua carga de trabalho. Isso lhe permite tomar decisões baseadas em fatos sobre como melhorar sua carga de trabalho. **Decisões baseadas em dados!** Sua infraestrutura de nuvem é código, para que você possa usar esses dados a fim de informar suas escolhas e melhorias na arquitetura ao longo do tempo.

**Melhore durante os “dias de jogo”:** teste o desempenho de sua arquitetura e de seus processos agendando regularmente “dias de jogo” para simular eventos em produção. Isso vai ajudá-lo entender onde as melhorias podem ser feitas e vai auxiliá-lo a desenvolver organizações.

#### 4.5.2. Excelência baseada nos cinco pilares

Ainda segundo a AWS (2020), para cumprir o **pilar da excelência operacional** é preciso estimar a capacidade de apoiar o desenvolvimento e executar cargas de trabalho de forma eficaz, obter *insights* sobre suas operações e melhorar continuamente processos e procedimentos de suporte para agregar valor aos negócios.

Trata-se de executar e monitorar sistemas para entregar valor ao negócio visando a melhoria contínua de processos e procedimentos. Destacam-se o gerenciamento e a automação de alterações, a reação a eventos de sistema e a definição de padrões de modo a gerenciar o dia a dia operacional de forma eficiente.

**Principais Serviços da AWS:** AWS Config, Amazon Cloudwatch, AWS CloudFormation.

O **pilar da segurança** abrange a capacidade de proteger dados, sistemas e ativos para aproveitar as tecnologias em nuvem a fim de melhorar sua segurança. Fornece uma visão geral dos princípios de design, práticas recomendadas e perguntas. Você pode encontrar orientações prescritivas sobre a implementação no Pilar de Segurança. O objetivo é proteger informações e sistemas, concentrando-se na confidencialidade e na integridade de dados, identificando e gerenciando privilégios

de forma eficiente, protegendo sistemas e estabelecendo controles para a detecção de eventos de segurança.

**Principais serviços da AWS:** IAM, Cloudtrail, WAF, KMS.

Ainda em relação à segurança, sete princípios devem ser observados:

- **Implemente uma base sólida de identidade:** implemente o princípio do menor privilégio e imponha a separação de tarefas com autorização apropriada para cada interação com os recursos da AWS. Centralize o gerenciamento de identidades e tenha como objetivo eliminar a dependência de credenciais estáticas de longo prazo.
- **Ativar rastreabilidade:** monitore, alerte e audite ações e alterações em seu ambiente em tempo real. Integre a coleção de logs e métricas aos sistemas para automaticamente investigar e agir.
- **Aplicar segurança em todas as camadas:** aplique uma abordagem de defesa em profundidade com vários controles de segurança. Aplique a todas as camadas (por exemplo, borda da rede, VPC, balanceamento de carga, cada instância e serviço de computação, sistema operacional, aplicativo e código).
- **Automatize as melhores práticas de segurança:** mecanismos de segurança automatizados baseados em software melhoram a sua capacidade de dimensionar com segurança, de forma mais rápida e econômica. Crie arquiteturas seguras, incluindo a implementação de controles definidos e gerenciado como código em modelos controlados por versão.
- **Proteja dados em trânsito e em repouso:** classifique seus dados em níveis de sensibilidade e use mecanismos, como criptografia, tokenização e controle de acesso, quando apropriado; todo cuidado é pouco!
- **Mantenha as pessoas afastadas dos dados:** use mecanismos e ferramentas para reduzir ou eliminar a necessidade de acesso direto ou processamento manual de dados. Isso diminui o risco de uso indevido ou modificação e erro humano ao manipular dados confidenciais. Nada de

credenciais demais: as pessoas devem acessar apenas o que precisam acessar!

- **Prepare-se para eventos de segurança:** prepare-se para um incidente, tendo políticas e processos de gerenciamento e investigação de incidentes alinhados à sua organização. Execute simulações de resposta a incidentes e use ferramentas com automação para aumentar sua velocidade de detecção, investigação e recuperação.

O **pilar da confiabilidade** se resume a evitar e se recuperar rapidamente de falhas, focando em configuração, requisitos de projetos, planejamento de recuperação e gestão de mudanças. Isso inclui a capacidade de operar e testar a carga de trabalho por todo o seu ciclo de vida.

**Principais serviços da AWS:** AWS CloudFormation, Amazon CloudWatch, S3, Glacier.

O **pilar da eficiência de desempenho** aborda o uso eficiente de recursos de TI e computação, selecionando os tipos e os tamanhos certos dos recursos, baseando-se nos requisitos de carga, performance de monitoramento na tomada de decisões fundamentais para manter a eficiência à medida que as necessidades de negócio evoluem.

**Principais serviços da AWS:** Autoscaling, ElastiCache, CloudWatch, CloudFront.

Por fim, o **pilar da otimização de custos** observa a importância de se evitar gastos desnecessários, compreendendo e controlando como o dinheiro está sendo gasto, seleção do número certo e mais adequado dos tipos de recursos, análise dos gastos ao longo do tempo e escalabilidade para atender às necessidades de negócios sem gastos excessivos.

**Principais serviços da AWS:** Tag resources, AWS Cost Explorer, Autoscaling.

#### 4.6. *Trust Advisor*

Além do AWS Well-Architected Tool, os clientes da AWS contam com uma excelente ferramenta chamada **Trust Advisor**, que pode dar recomendações de

performance, segurança e otimização de custos. Entretanto, é importante observar que, sem o devido suporte contratado, o *Trust Advisor* não faz a análise de todas as métricas que ele tem disponíveis. É uma excelente ferramenta cujo “poder de fogo” cresce na medida em que o suporte mais adequado é contratado.

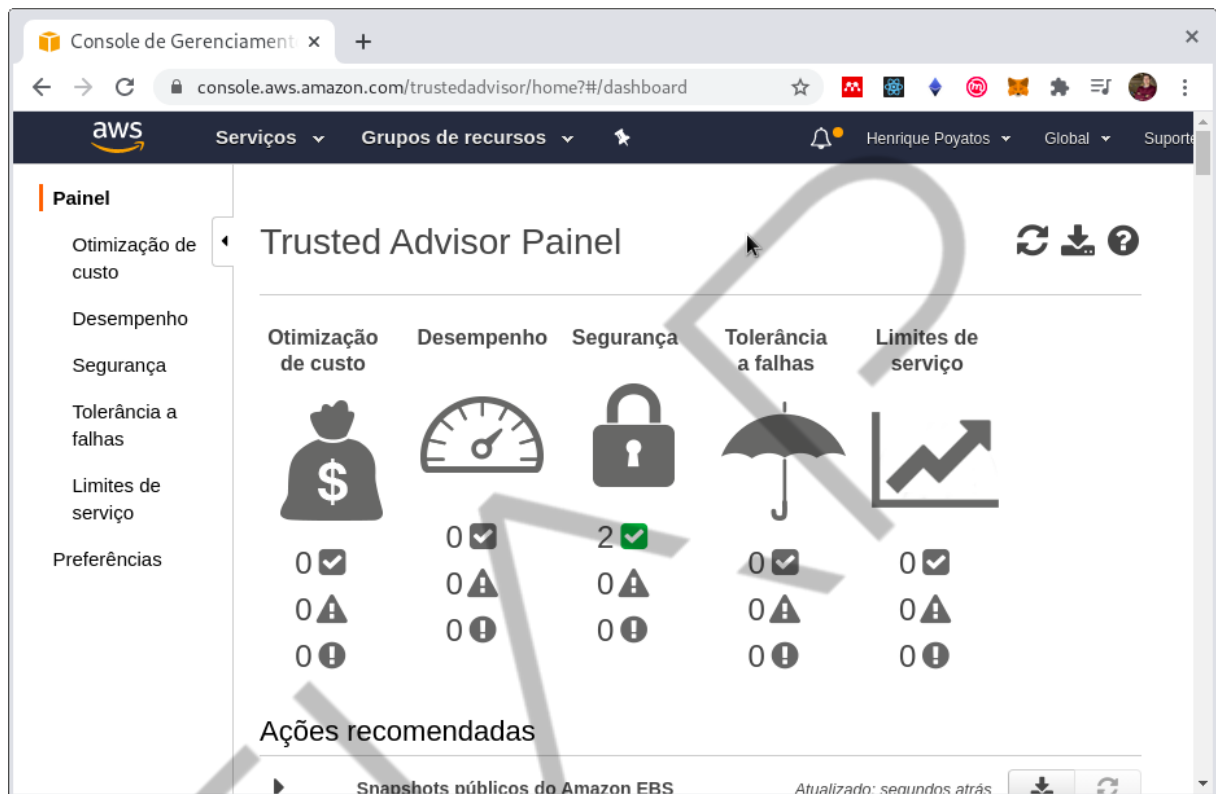


Figura 4.3 – Painel do *Trusted Advisor*

Fonte: Elaborado pelo autor (2020)

#### 4.7. Serviços principais

Vamos abordar brevemente os quatro principais serviços da AWS (dentre dezenas de outros!):

- **EC2:** *Elastic Compute Cloud*, é o serviço mais famoso da AWS. Trata-se da possibilidade de subir máquinas virtuais Windows ou Linux com configurações variadas de forma segura, altamente escalável para os mais diversos usos: Big Data, ERP, CRM, aplicações em escala, *e-commerce*.
- **Elastic Beanstalk:** plataforma como serviço *deploy* de aplicações web. Ele é integrado com o Git e IDEs de mercado (Dev). Faz provisionamento automático, seguro, escalável e automático (equivalente ao Google App Engine).



- **Lambda:** serviço de função como serviço (*Function as a service* – FaaS). É cobrado por execução, diferentemente dos outros modelos cuja cobrança é por hora. Suporta diversas linguagens de programação, como Python e NodeJS, você hospeda uma função e, de forma *servless*, a Amazon faz a alocação de recursos para que esta rode de maneira adequada. O serviço é altamente integrável com outros da AWS (Equivalente ao Google Cloud Functions).
- **ECS:** Gestão de Containers Docker, escalável, de baixo custo, seguro e fácil de se integrar com outros serviços da AWS.

Falando um pouco mais sobre o EC2, o Amazon Elastic Compute Cloud é um serviço que disponibiliza capacidade computacional segura e redimensionável na nuvem. Provavelmente, é como muitas empresas tiveram o seu primeiro contato com *cloud computing*, além de ser o serviço que possibilitou o surgimento de todo um ecossistema mundial de *startups*, especialmente na área de tecnologia; afinal, o modelo de negócios baseado em *cloud* permite que a infraestrutura tecnológica cresça proporcionalmente ao negócio, sem grandes e proibitivos investimentos iniciais em infraestrutura.

Quanto à modalidade de gastos, a mais comum é **on demand**, ou seja, sob demanda, você paga aquilo que usa (*pay as you go*), preços por hora e com a possibilidade de *free tier* (degustação gratuita).

Existe ainda a possibilidade de fazer uma **reserva** de infraestrutura, que pode ser de 1 ou 3 anos, cujos descontos podem chegar até 75%, com parte do pagamento à vista e outra parte mensalmente. Trata-se do modelo ideal para projetos sólidos cuja necessidade de infra é certa por um longo período.

Outra modalidade é a de *spot* ou leilão. É interessante observar que a AWS possui um uso de infraestrutura bastante flutuante, contando com grande capacidade ociosa durante algumas horas do dia ou períodos do ano. O cliente pode definir o preço que está disposto a pagar pela capacidade ociosa da AWS, e se o preço for aceito pela empresa, as instâncias solicitadas serão provisionadas.

Por fim, existe ainda a possibilidade de ter um ou mais **servidores dedicados** na AWS. Cobrados por hora, os descontos podem chegar a 70%.

## 4.8. Hands on: Elastic Compute Cloud (EC2)

Nenhum capítulo sobre AWS que se preze seria completo sem um *hands on* a respeito do “carro-chefe” dentre os produtos da AWS, o Elastic Compute Cloud ou EC2. A seguir, não veremos apenas como subir uma instância no serviço, mas de que maneira se conectar a ele e como automatizar sua operação por meio de linha de comando AWS (AWS CLI).

### 4.8.1. Criando uma instância EC2

Como é possível observar na Figura “Console de gerenciamento da AWS”, em primeiro lugar é preciso acessar o console de gerenciamento da AWS em um navegador web devidamente autenticado no serviço.

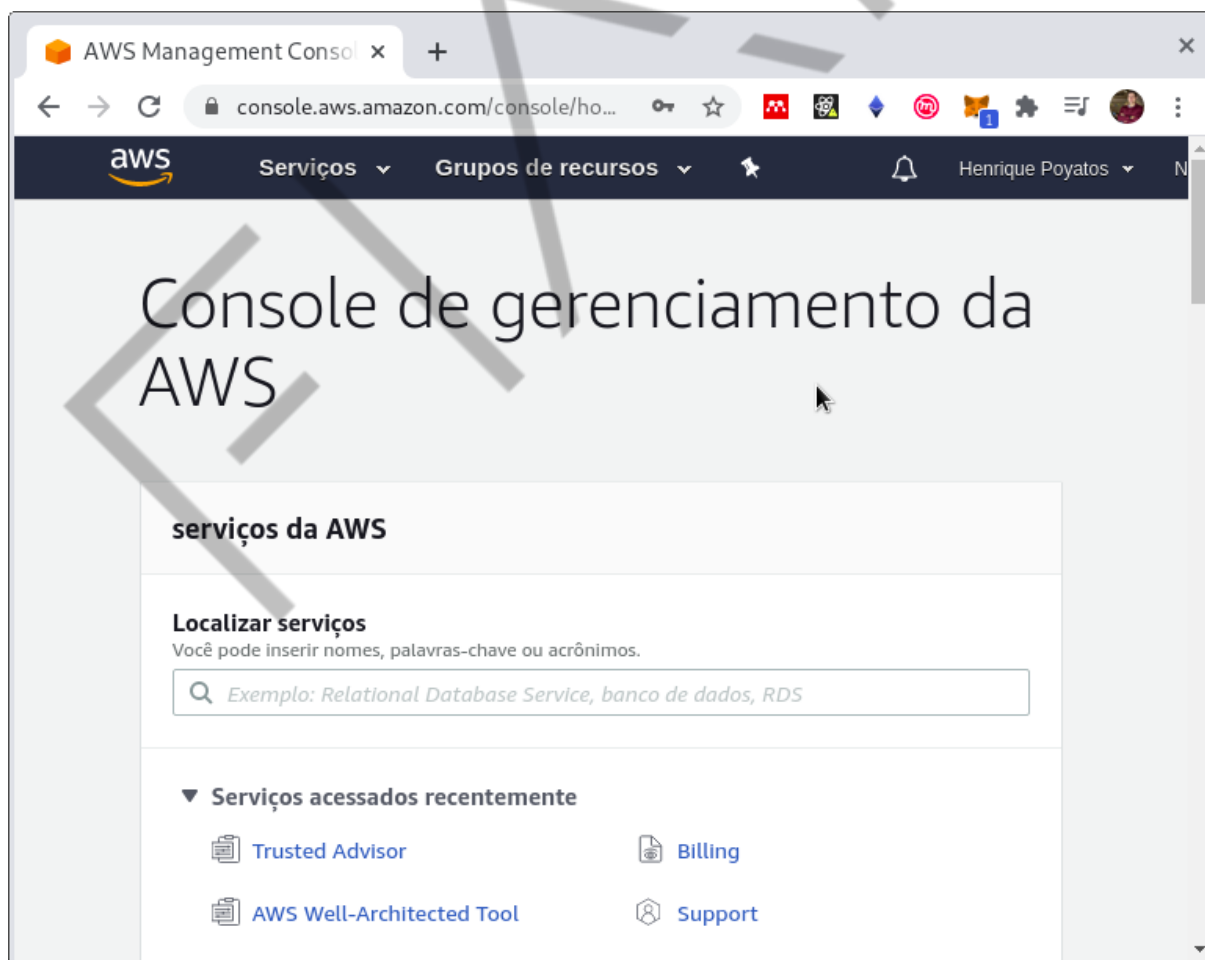


Figura 4.4 – Console de gerenciamento da AWS  
Fonte: Elaborado pelo autor (2020)

Uma vez que há dezenas de serviços diferentes, realizar a busca pelo serviço desejado na barra “Localizar serviços” é sempre uma boa opção. Ao clicar na barra, contudo, os principais serviços são exibidos e não é nenhuma surpresa que “EC2” esteja no topo da lista (Figura “Acesso aos grupos de recursos”).

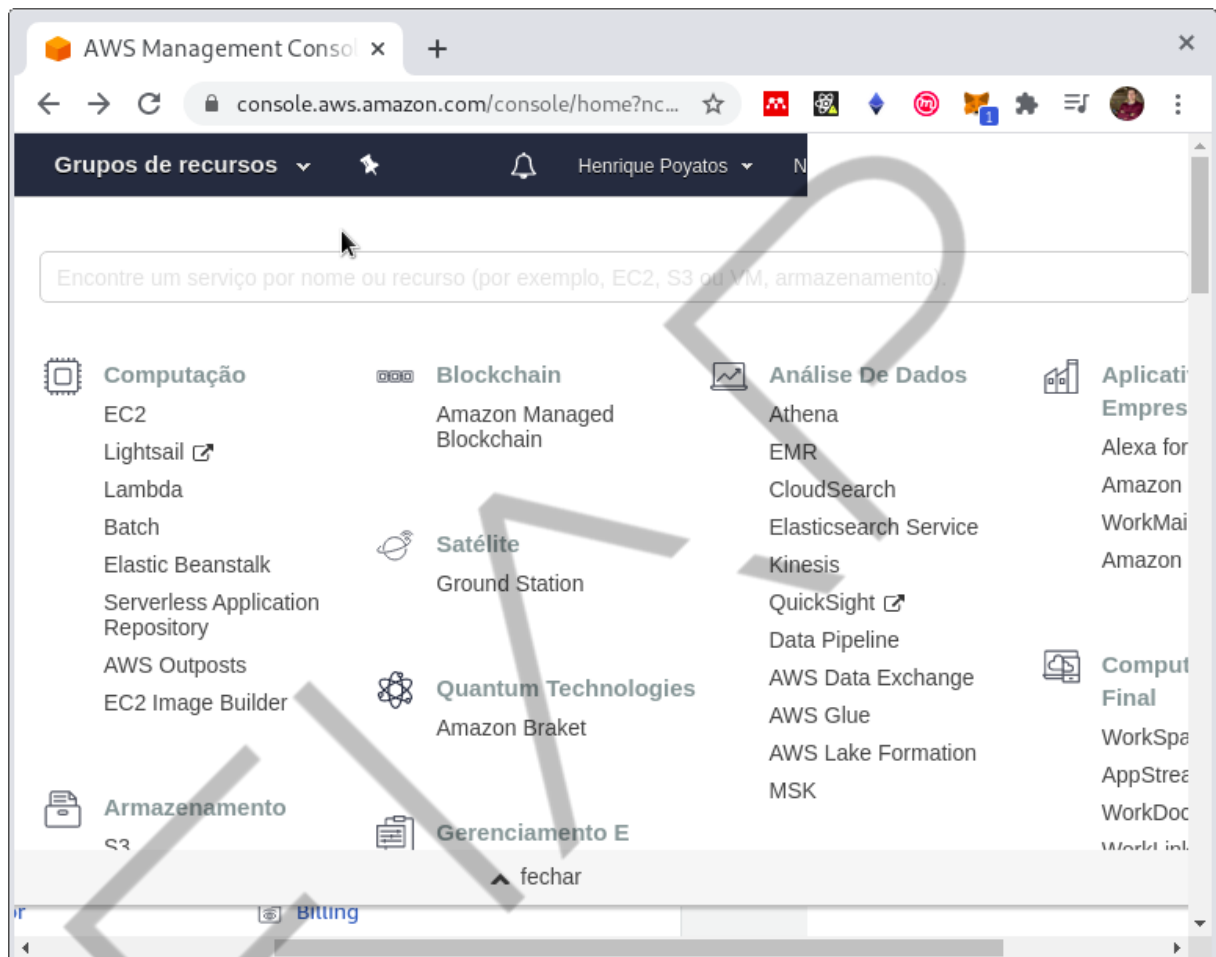


Figura 4.5 – Acesso aos grupos de recursos  
Fonte: Elaborado pelo autor (2020)

Ao entrar no console específico do serviço EC2, a primeira coisa que podemos ver é um sumário dos recursos EC2 que estão sendo usados naquele momento e naquela região em específico. Não há muito a se observar quando entramos pela primeira vez (Figura “Sumário da situação atual dos serviços EC2 em uma determinada região”).

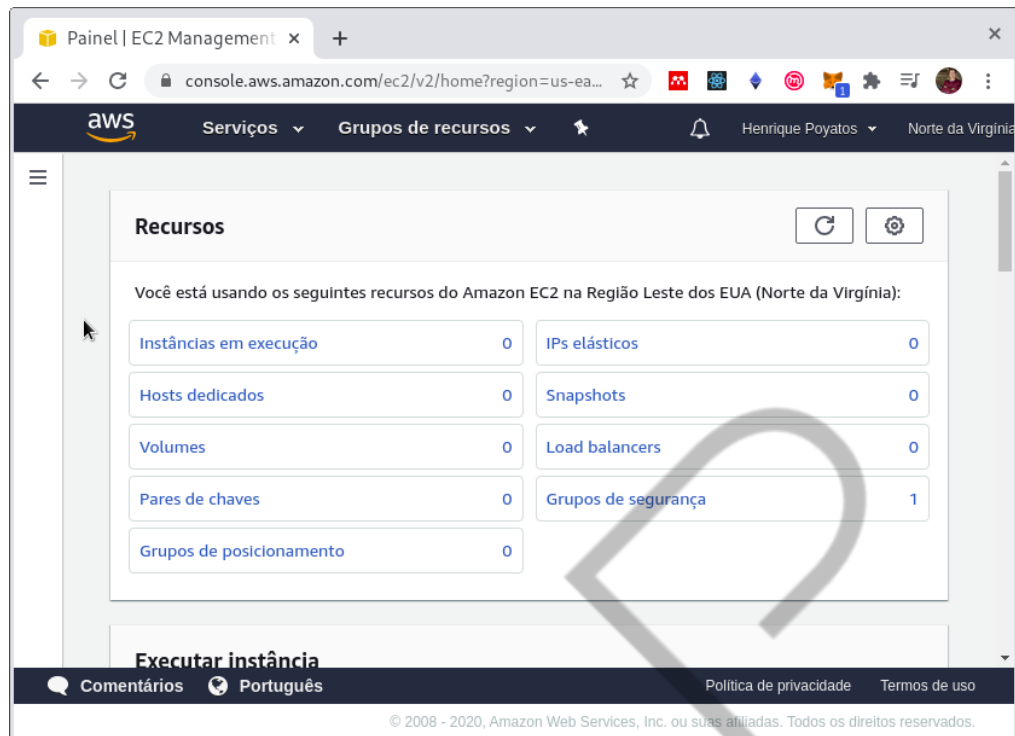


Figura 4.6 – Sumário da situação atual dos serviços EC2 em uma determinada região  
Fonte: Elaborado pelo autor (2020)

Mais abaixo na mesma tela, temos um botão de “Executar instância” e é exatamente isso que queremos fazer (Figura “Execução de uma instância EC2”).

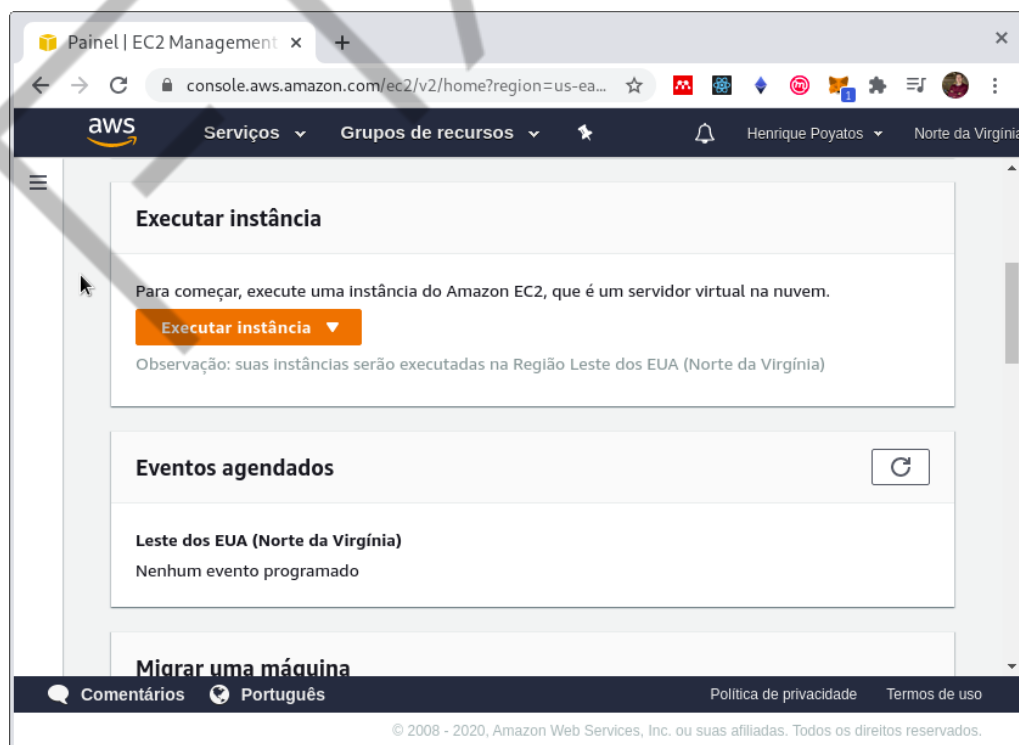


Figura 4.7 – Execução de uma instância EC2  
Fonte: Elaborado pelo autor (2020)

O primeiro passo para a criação de uma instância EC2 na AWS é a escolha de uma imagem para essa VM, o que a empresa chama de *Amazon Machine Image* (AMI). É nesse momento que podemos fazer uma escolha de arquitetura (selecionando entre x86 e processadores ARM, por exemplo) e sistema operacional. Há várias opções em sistemas operacionais Linux, como Red Hat Enterprise Linux (RHEL), SUSE Linux Enterprise Server, Ubuntu Server e uma distribuição específica da Amazon para a AWS chamada **Amazon Linux**. Nesse mesmo passo, há a possibilidade de optar por algumas AMIs com Windows Server, além de AMIs criadas pela comunidade, ou seja, cujo fornecedor não é a Amazon. Por fim, é possível importar sua própria AMI, recurso muito útil caso a VM seja fruto de uma migração (Figura “Passo 1, escolhendo de uma *Amazon Machine Image* (AMI)”).

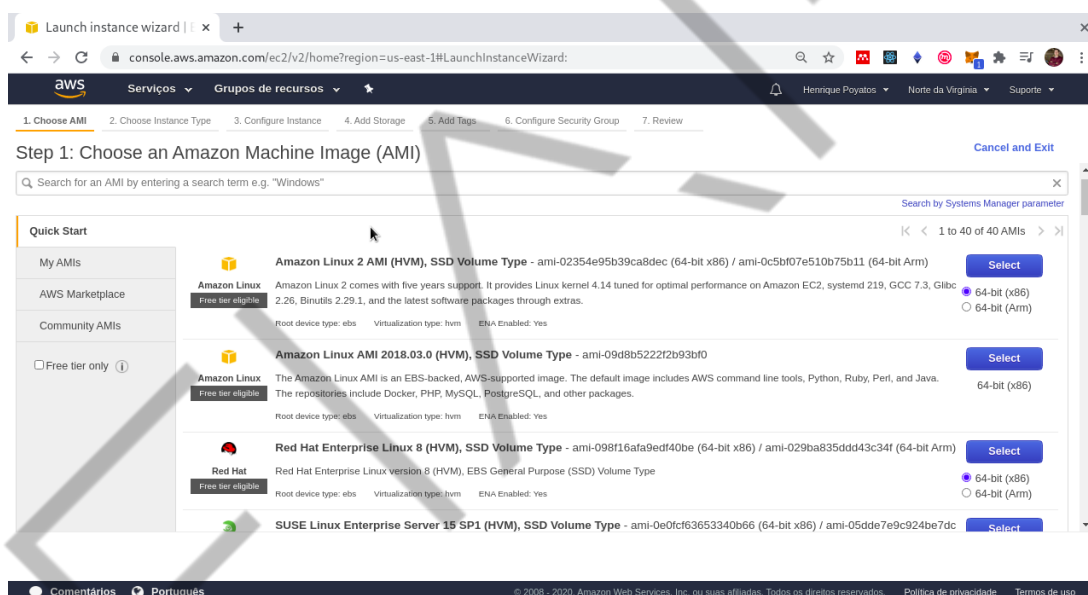


Figura 4.8 – Passo 1, escolhendo uma *Amazon Machine Image* (AMI)

Fonte: Elaborado pelo autor (2020)

O segundo passo é escolher o tipo de instância que será criado, e existem diversas configurações para os mais diferentes propósitos, como é possível acompanhar no Quadro “Tipos de instância e seus casos de uso”.

Tipos de instância	Descrição
T2, T3, T3a, A1, M6g, M5, M5a, M5n e M4	Instâncias de uso geral que possuem um equilíbrio de recursos de computação, memória e rede, que podem ser usados para diversas cargas de trabalho. Ideais para aplicativos que utilizam esses recursos em proporções iguais, como servidores da Web e repositórios de código.
C4, C5, C5a, C5n e C6g	Ideais para aplicativos vinculados à computação que se beneficiam de processadores de alto desempenho, como servidores da Web de alto desempenho, transcodificação de mídia ou servidor de jogos dedicados.

R6g, R5, R5a, R5n, R4, X1e, X1 e z1d	Otimizadas de memória são projetadas para fornecer desempenho rápido para cargas de trabalho que processam grandes conjuntos de dados na memória, como bancos de dados <i>in memory</i> , por exemplo. Ideais para Big Data que precise de resposta rápida.
P3, P2, Inf1, G4, G3 e F1	Instâncias de computação aceleradas usam aceleradores de <i>hardware</i> , ou coprocessadores, para executar funções, como cálculos de número de ponto flutuante, processamento de gráficos ou correspondência de padrões de dados, mais eficientemente do que é possível no software em execução nas CPUs, tornando-as ideais para <i>Machine Learning</i> .
I3, I3en, D2 e H1	São otimizadas para fornecer dezenas de milhares de operações de E/S aleatórias de baixa latência por segundo (IOPS) para aplicativos, ou seja, bancos de dados de altíssimo desempenho.

Quadro 4.1 – Tipos de instância e seus casos de uso

Fonte: AWS (s.d.)

É claro que, quanto maior a demanda por recursos, maior será a estimativa de gastos (Figura “Passo 2, escolhendo um tipo de instância EC2”).

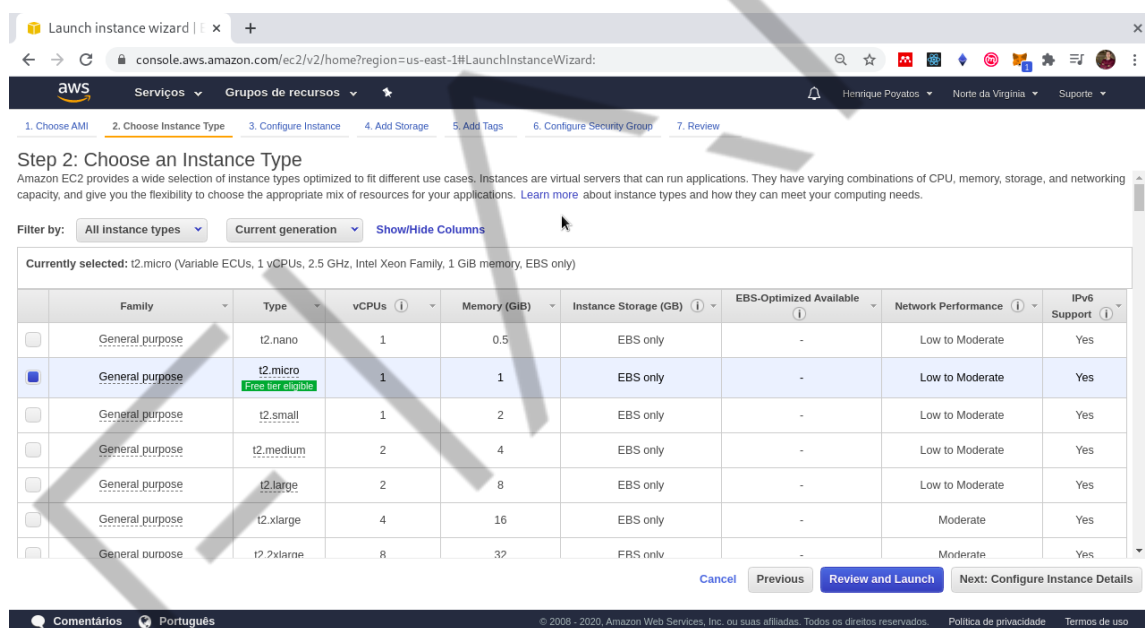


Figura 4.9 – Passo 2, escolhendo um tipo de instância EC2

Fonte: Elaborado pelo autor (2020)

O terceiro passo permite a configuração de detalhes da instância, como personalização da VPC (*Virtual Private Cloud*), sub-redes, entre outras possibilidades (Figura “Passo 3, configurando detalhes da instância EC2”).

**Launch instance wizard**

console.aws.amazon.com/ec2/v2/home?region=us-east-...

**Step 3: Configure Instance Details**

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Number of instances** 1 [Launch into Auto Scaling Group](#)

**Purchasing option** ☐ Request Spot instances

**Network** vpc-a3eb1ade (default) [Create new VPC](#)

**Subnet** No preference (default subnet in any Availability Zone) [Create new subnet](#)

**Auto-assign Public IP** Use subnet setting (Enable)

**Placement group** ☐ Add instance to placement group

**Capacity Reservation** Open

**IAM role** None [Create new IAM role](#)

**Shutdown behavior** Stop

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)

Comentários Português Política de privacidade Termos de uso

© 2008 - 2020, Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados.

Figura 4.10 – Passo 3, configurando detalhes da instância EC2

Fonte: Elaborado pelo autor (2020)

O quarto passo (vide Figura “Passo 4, adicionando dispositivo de armazenamento na instância EC2”) é o momento de configurar o espaço de armazenamento da instância, o seu “disco”; observe que esses volumes de armazenamento são chamados pela empresa de EBS (*Elastic Block Store*) e são considerados um serviço de apoio ao EC2, logo, sempre que a instância precisar de mais volumes de armazenamento, é possível contratar mais no serviço de EBS e “plugar” na instância em execução. Esses EBS são classificados em tipos diferentes destinados a propósitos distintos, como é possível observar no Quadro “Tipos de EBS e seus casos de uso”.

Tipos de EBS	Descrição
SSD (gp2)	Drives de estado sólido (SDD) destinados para uso geral, como <i>webserver</i> s.
SSD Prov IOPS (io1)	Drives de estado sólido (SDD), otimizados para operações de entrada e saída, sendo ideais para bancos de dados.
<i>Throughput Optimized</i> HDD (st1)	Disco magnético com alta taxa de transferência, ideal para uso em <i>Big Data</i> .
<i>Cold</i> HDD (sc1)	Disco magnético tradicional (HDD), com taxas de gravação e leitura mais baixas, destinado para <i>backups/archives</i> . Sua vantagem é o custo mais baixo.

Quadro 4.2 – Tipos de EBS e seus casos de uso

Fonte: AWS (s.d., f)



Segundo a AWS (s.d., f), o serviço de EBS também conta com o recurso de *snapshots* nesses volumes de armazenamento, salvando o estado atual dos dados, podendo servir não apenas de *backup*, mas como um excelente *template* na criação de novas instâncias.

Observe ainda, no último campo à direita da Figura “Passo 4, adicionando dispositivo de armazenamento na instância EC2”, a possibilidade de encriptação dos dados armazenados (o chamado *encryption on rest*).

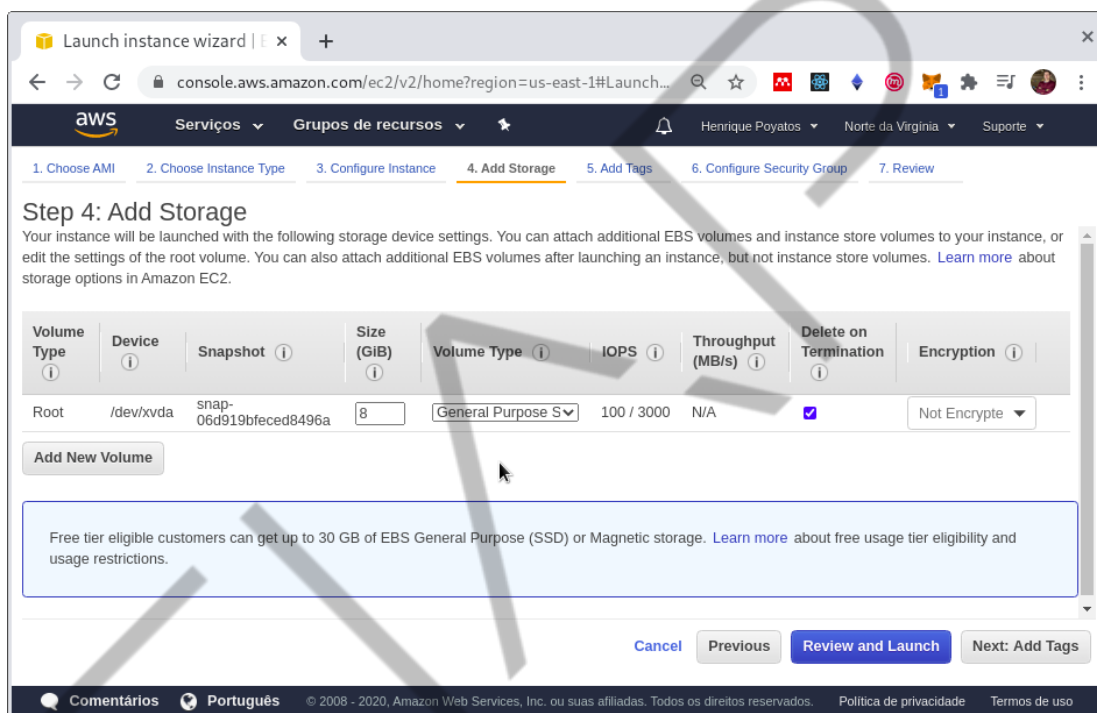


Figura 4.11 – Passo 4, adicionando dispositivo de armazenamento na instância EC2

Fonte: Elaborado pelo autor (2020)

Como podemos observar na Figura “Passo 5, escolhendo *tags* para a instância EC2”, é possível adicionar *tags* para identificar melhor suas instâncias. A forma de organizar é muito específica de organização para organização, mas é comum o uso de *tags* com o propósito da instância, o nome de seus serviços (apache2, mysql etc.) ou o nome dos projetos que suportam.



Launch instance wizard | x

console.aws.amazon.com/ec2/v2/home?region=...

Serviços Grupos de recursos Henrique Poyatos Norte da Virgínia Suporte

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key	Value	Instances	Volumes
Name	Webserver	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Add another tag (Up to 50 tags maximum)

Cancel Previous Review and Launch Next: Configure Security Group

Comentários Português Política de privacidade Termos de uso

© 2008 - 2020, Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados.

Figura 4.12 – Passo 5, escolhendo *tags* para a instância EC2  
Fonte: Elaborado pelo autor (2020)

O sexto passo é o momento de configurar o acesso a essa instância, trata-se de fato da configuração do firewall; embora seja possível realizar as configurações de segurança da forma tradicional no Linux (ou Windows), o console EC2 da AWS traz uma opção prática e simples de se configurar (Figura “Passo 6, configurando o grupo de segurança da instância EC2”).

Launch instance wizard | x

console.aws.amazon.com/ec2/v2/home?region=us-east-1#LaunchIns...

Serviços Grupos de recursos Henrique Poyatos Norte da Virgínia Suporte

1. Choose AMI 2. Choose Instance Type 3. Configure Instance 4. Add Storage 5. Add Tags 6. Configure Security Group 7. Review

### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group ☐ Select an existing security group

Security group name: launch-wizard-1

Description: launch-wizard-1 created 2020-08-06T22:06:26.793-03:00

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTPS	TCP	443	Custom 0.0.0.0/0, ::0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0, ::0	e.g. SSH for Admin Desktop

Add Rule

**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known

Cancel Previous Review and Launch

Comentários Português © 2008 - 2020, Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados. Política de privacidade Termos de uso

Figura 4.13 – Passo 6, configurando o grupo de segurança da instância EC2  
Fonte: Elaborado pelo autor (2020)

No sétimo e último passo, ilustrado na Figura “Passo 7, revisando antes de subir a instância EC2”, é apresentado um sumário com toda a configuração prévia da instância para uma breve revisão antes que esta seja executada pela primeira vez. Revise com atenção antes de prosseguir.

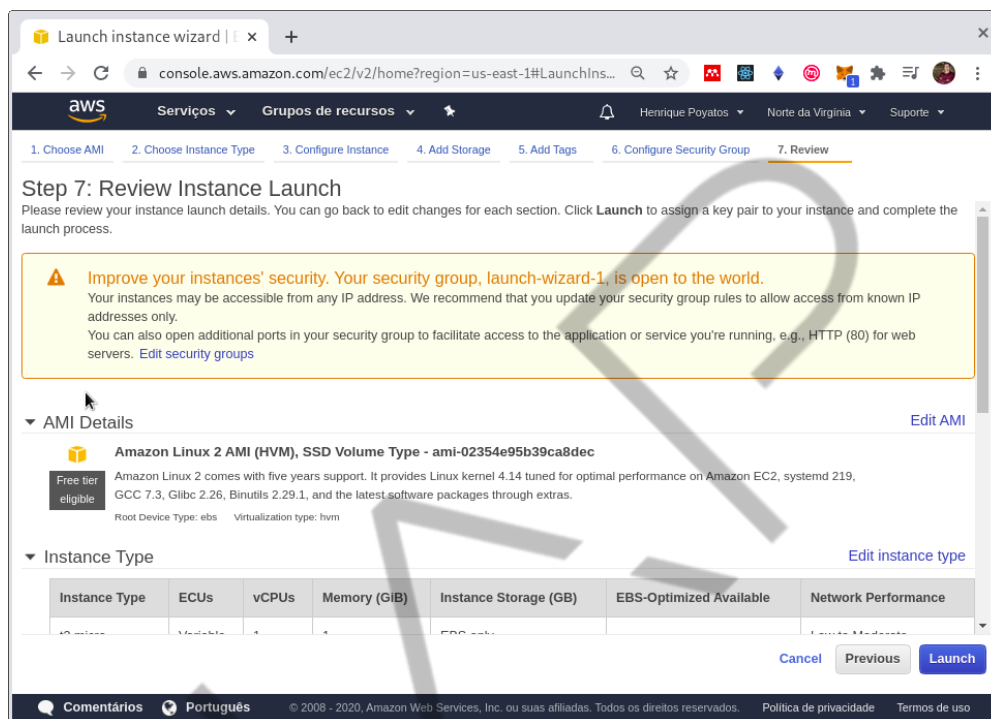


Figura 4.14 – Passo 7, revisando antes de subir a instância EC2

Fonte: Elaborado pelo autor (2020)

Por fim, é necessário definir o par de chaves que será utilizado para acessar a instância. Chamamos de par de chaves pois a técnica usada aqui são as chaves assimétricas ou chaves público-privada, por meio das quais os *hosts* criptografam e descriptografam sua comunicação, combinando sua própria chave privada e a chave pública do outro *host*. É possível criar um par de chaves ou utilizar um par preexistente, fica “a gosto do freguês” (Figura “Criando ou configurando o par de chaves para acesso SSH na instância EC2”).

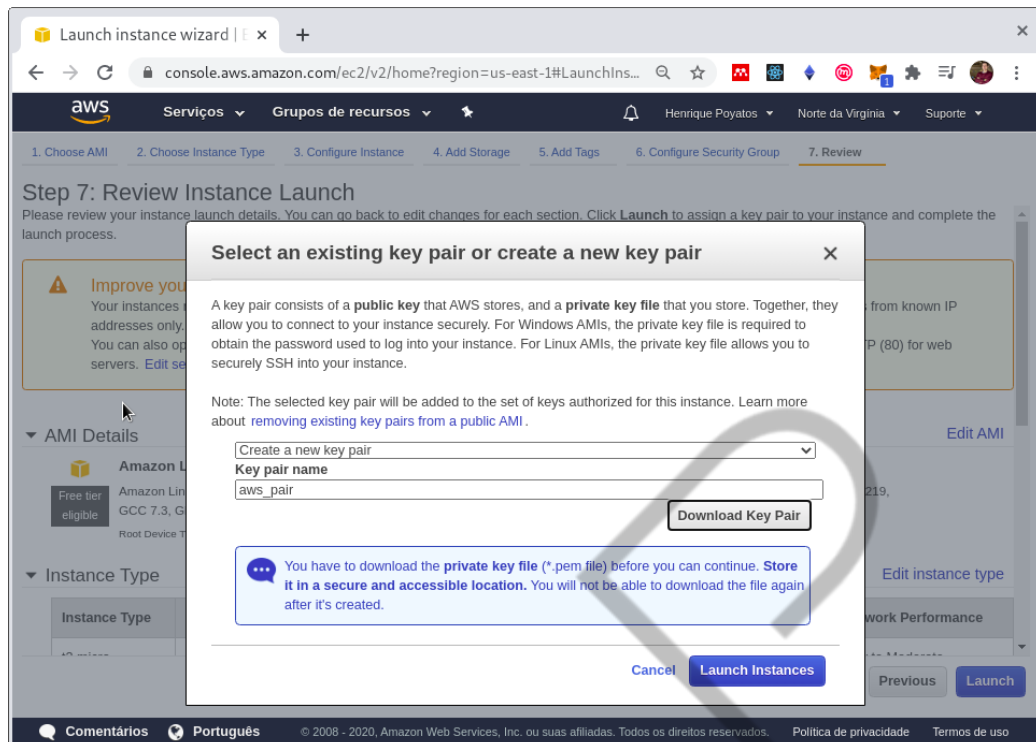


Figura 4.15 – Criando ou configurando o par de chaves para acesso SSH na instância EC2  
Fonte: Elaborado pelo autor (2020)

Na Figura “Instância EC2 em execução”, podemos ver o final do processo, com a instância devidamente inicializada. Repare que, ao longo do procedimento, havia sempre um “atalho” (o botão “*Review and Launch*” estava sempre presente no canto inferior direito) que permitia subir a instância com as configurações dos seis passos de maneira padronizada.

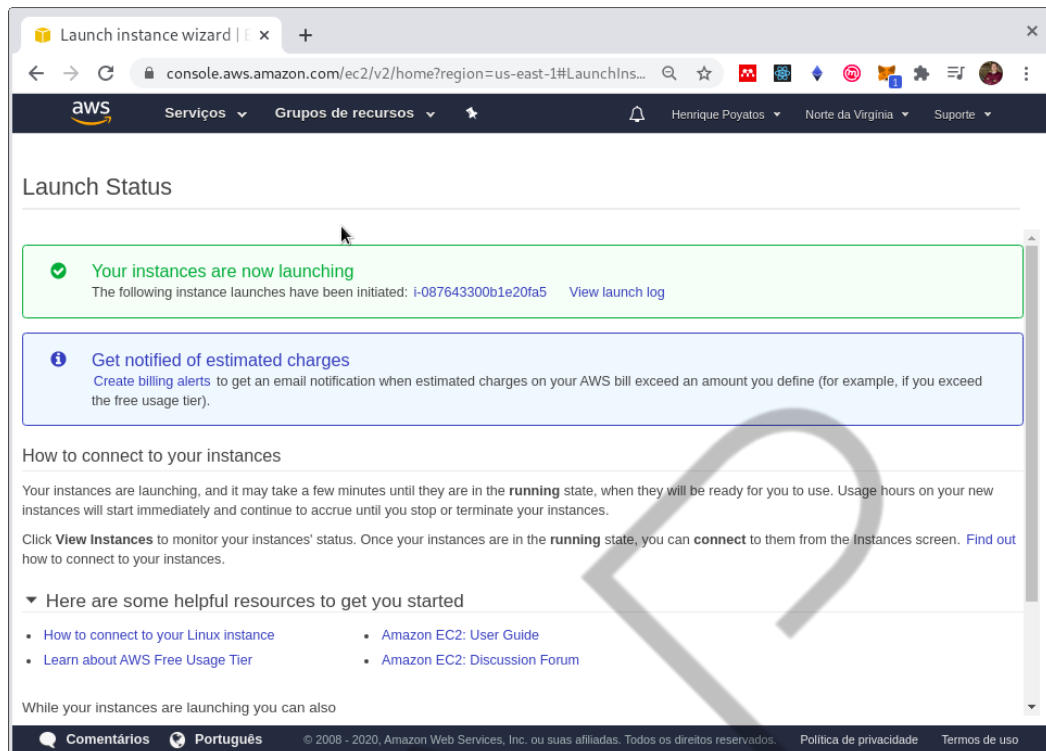


Figura 4.16 – Instância EC2 em execução  
Fonte: Elaborado pelo autor (2020)

Na listagem de instâncias no console de gerenciamento EC2, é possível visualizar as principais informações das VMs disponíveis e, conforme observado na Figura “Lista de instâncias EC2 na AWS”, a instância “webserver” está em funcionamento.

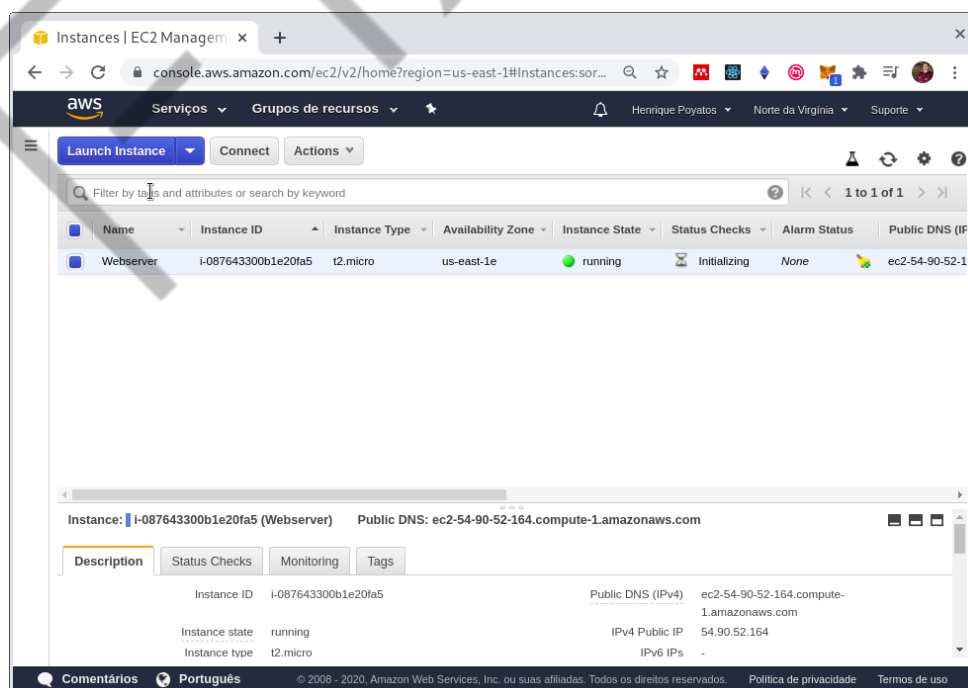


Figura 4.17 – Lista de instâncias EC2 na AWS  
Fonte: Elaborado pelo autor (2020)

Baixando o arquivo .PEM no momento da criação das chaves e seguindo as instruções da AWS, é simples realizar um acesso SSH à instância, conforme pode ser observado na Figura “Acesso SSH à instância EC2 utilizando o par de chaves”. O procedimento usando Putty no Windows também é de fácil configuração.

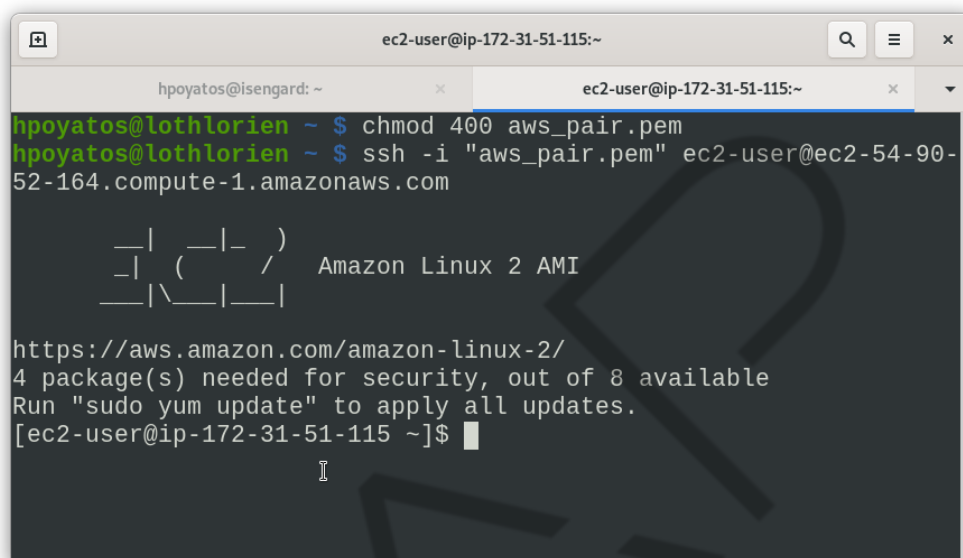


Figura 4.18 – Acesso SSH à instância EC2 utilizando o par de chaves  
Fonte: Elaborado pelo autor (2020)

Nas listagens de instâncias no console EC2 na AWS, com o botão direito do mouse é fácil realizar ações importantes, como derrubar a instância em execução (Figura “Derrubando uma instância EC2 no console da AWS”).

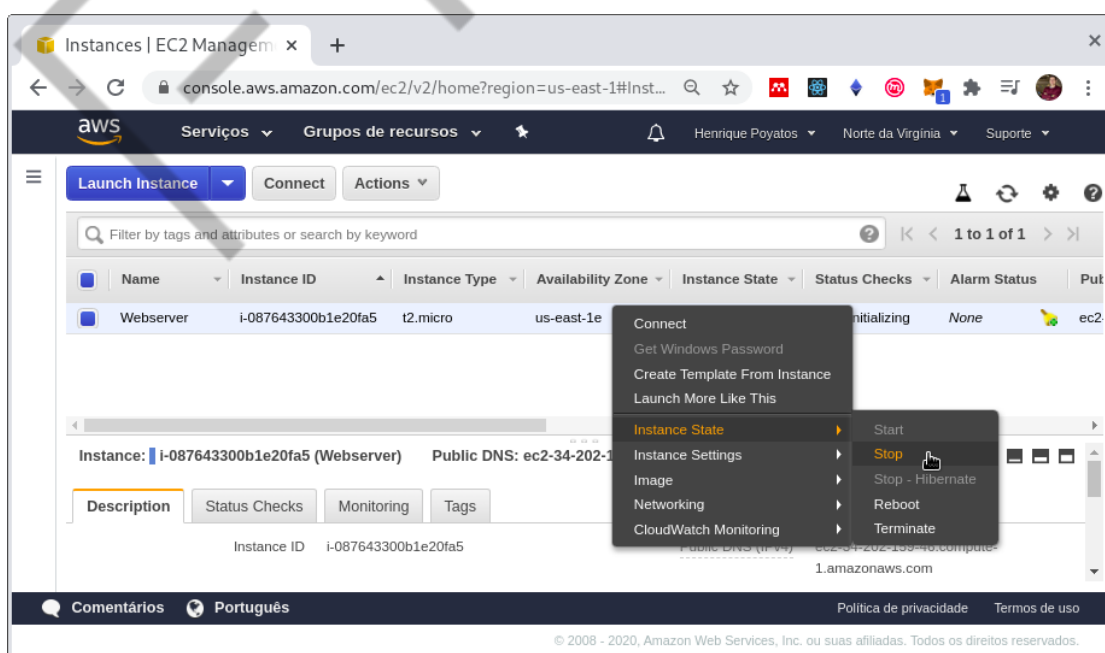


Figura 4.19 – Derrubando uma instância EC2 no console da AWS  
Fonte: Elaborado pelo autor (2020)

#### 4.8.2. Automação de instância EC2 usando AWS CLI

Vejamos agora uma das coisas mais importantes para um profissional especializado em *cloud computing*: a automação das operações de *cloud* utilizando linha de comandos ou uma API que a integre aos processos da empresa.

O primeiro passo é realizar a instalação do utilitário AWS. Ele pode ser instalado em Windows, macOS, Linux e Docker, e como pré-requisito é necessário ter a linguagem Python pré-instalada. Verifique a documentação oficial para mais detalhes: <[https://docs.aws.amazon.com/pt\\_br/cli/latest/userguide/cli-chap-install.html](https://docs.aws.amazon.com/pt_br/cli/latest/userguide/cli-chap-install.html)>.

O segundo passo é a criação de credenciais a serem utilizadas pelo AWS CLI na linha de comando. É necessário acessar <<https://console.aws.amazon.com/iam/home#/home>> e selecionar o gerenciamento de acesso de “Usuários” no menu esquerdo (vide Figura “Identity and Access Management (IAM)”).

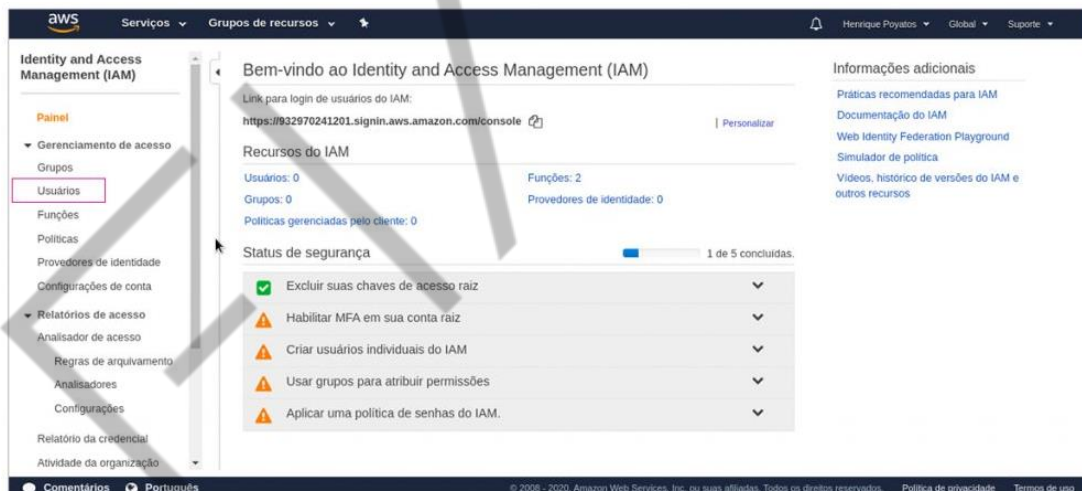


Figura 4.20 – Identity and Access Management (IAM)

Fonte: Elaborado pelo autor (2020)

Clique em “Adicionar usuário” (Figura “Criação de usuário no IAM”).

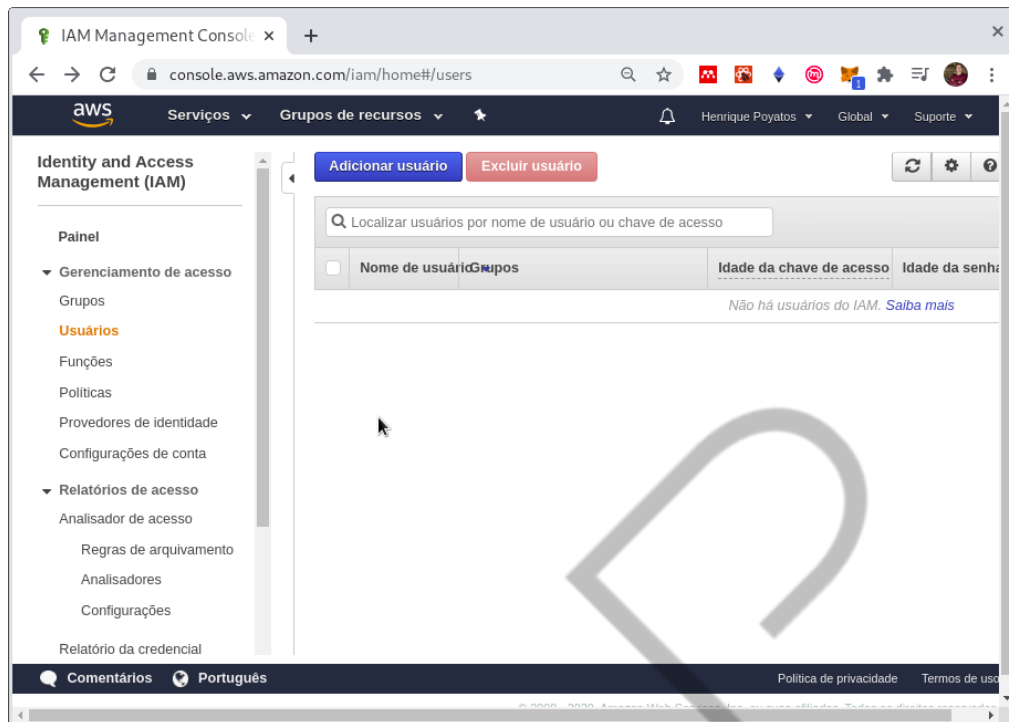


Figura 4.21 – Criação de usuário no IAM  
Fonte: Elaborado pelo autor (2020)

Como podemos observar na Figura “Criando usuário no IAM para uso em API, CLI e SDK”, além do nome do usuário, é importante marcar a opção “Acesso programático”, que habilitará o usuário justamente para um acesso via API, CLI ou SDK da AWS.

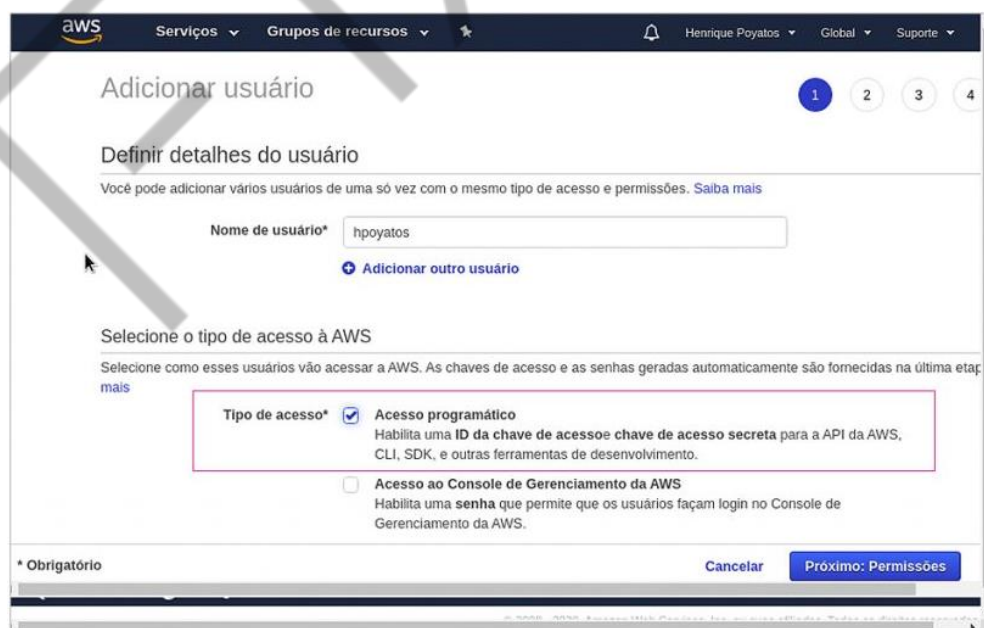


Figura 4.22 – Criando usuário no IAM para uso em API, CLI e SDK  
Fonte: Elaborado pelo autor (2020)



Como não havia grupos de usuários no IAM previamente criados, faz-se necessário criar um, como mostra a Figura “Criação de grupo no IAM”.

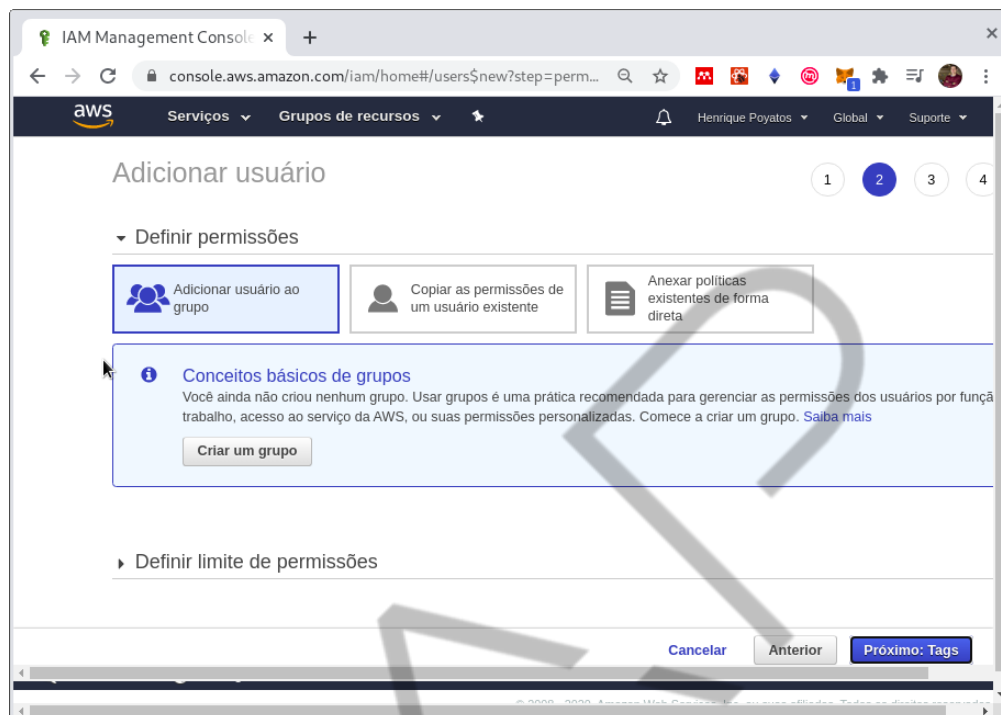


Figura 4.23 – Criação de grupo no IAM  
Fonte: Elaborado pelo autor (2020)

No exemplo em questão, o grupo será criado com o nome “CLIUsers” (Figura “Criação do grupo ‘CLIUsers’ no IAM”).

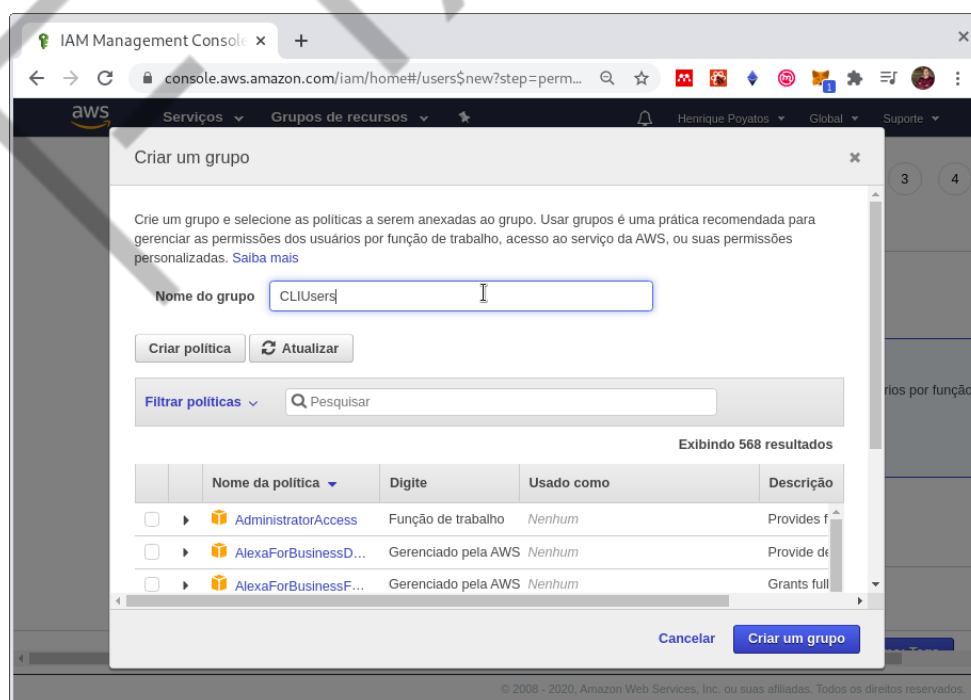


Figura 4.24 – Criação do grupo “CLIUsers” no IAM  
Fonte: Elaborado pelo autor (2020)



Finalizada a criação do grupo, o usuário “hpoyatos” estará associado a ele logo em seguida (Figura “Finalização da criação do grupo ‘CLIUsers’ no IAM”).

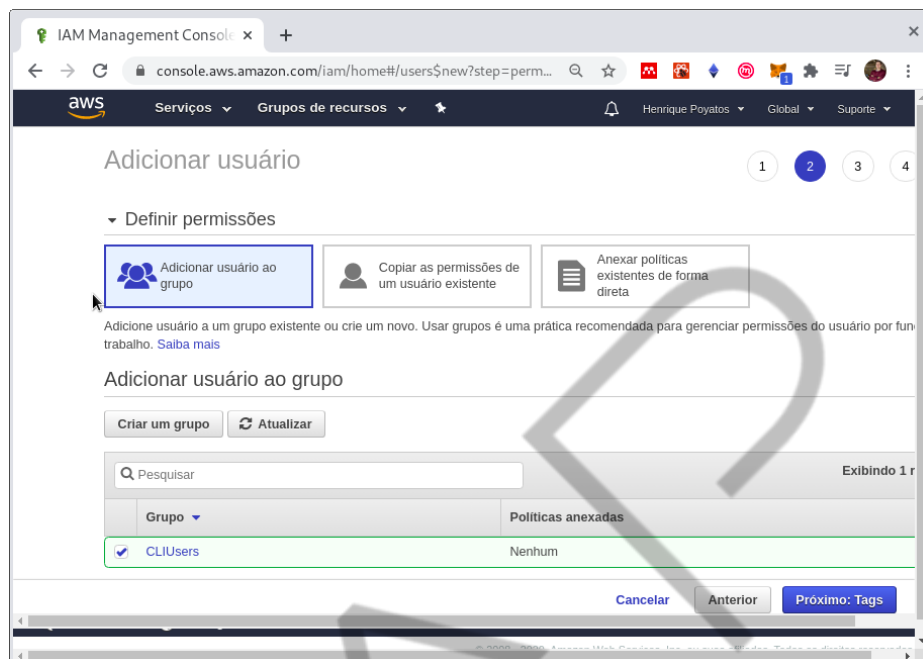


Figura 4.25 – Finalização da criação do grupo ‘CLIUsers’ no IAM  
Fonte: Elaborado pelo autor (2020)

Assim como acontecia com a instância, *tags* podem ser utilizadas no usuário para aprimorar a localização em uma busca, portanto preencha como julgar adequado (Figura “Tags para o usuário ‘hpoyatos’ no IAM”).

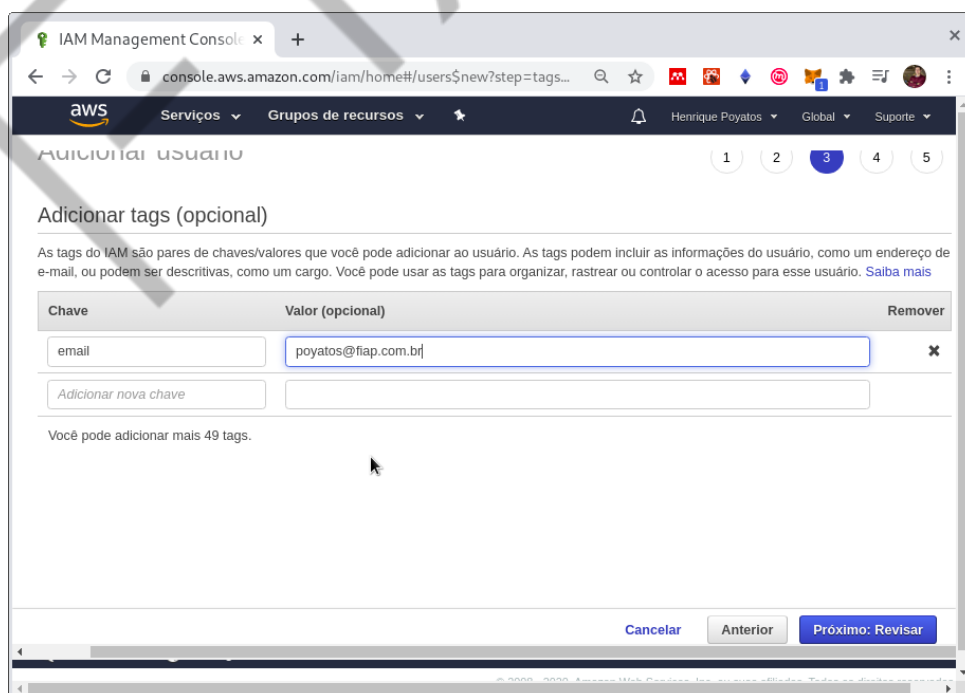


Figura 4.26 – Tags para o usuário “hpoyatos” no IAM  
Fonte: Elaborado pelo autor (2020)

Revise os dados e clique em “Adicionar usuário” (Figura “Resumo da criação do usuário no IAM”);

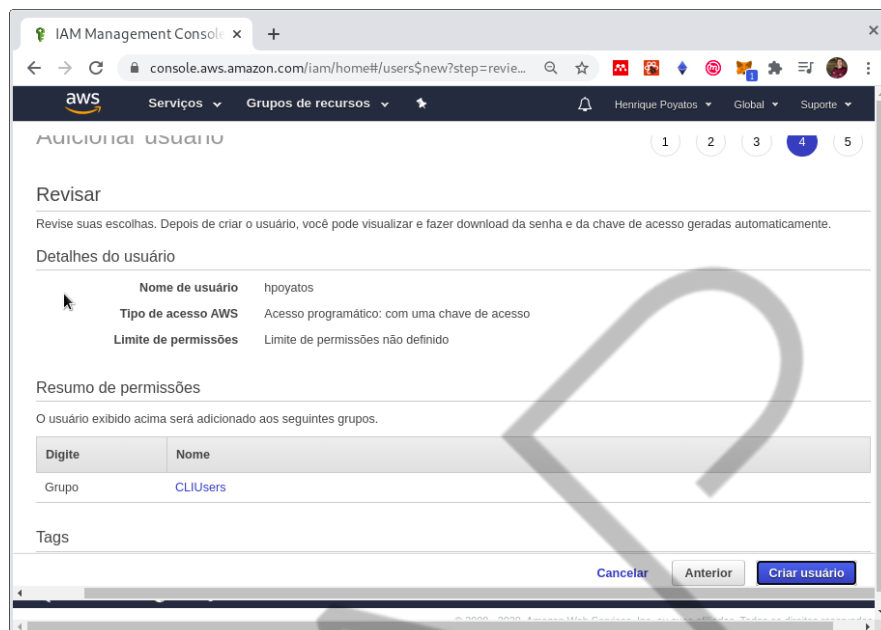


Figura 4.27 – Resumo da criação do usuário no IAM  
Fonte: Elaborado pelo autor (2020)

Na Figura “Usuário criado com ID e chave de acesso no IAM”, observamos que o usuário “hpoyatos” foi gerado com êxito, e o console AIM retorna duas informações importantes: o **ID da chave de acesso** e a **Chave de acesso secreta** (visível apenas ao clicar no link “Exibir”), é com eles que configuramos o utilitário **aws cli** para o devido acesso via linha de comando.

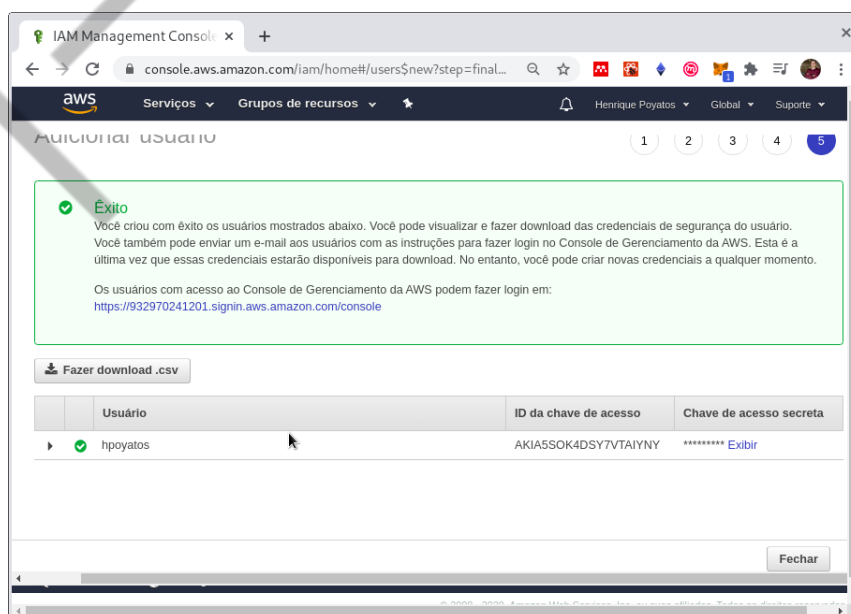


Figura 4.28 – Usuário criado com ID e chave de acesso no IAM  
Fonte: Elaborado pelo autor (2020)

No terminal de comandos com o AWS devidamente instalado, rode o comando.

```
$ aws configure --profile <profile>
```

Comando de prompt 4.1 – Comando para configurar profile no AWS CLI

Fonte: Elaborado pelo autor (2020)

Observe, na Figura “Comando aws configure”, que as duas informações são solicitadas, assim como a região da AWS preferencial e o modo preferencial de exibir informações de saída no aws cli (que pode ser em texto puro ou json).

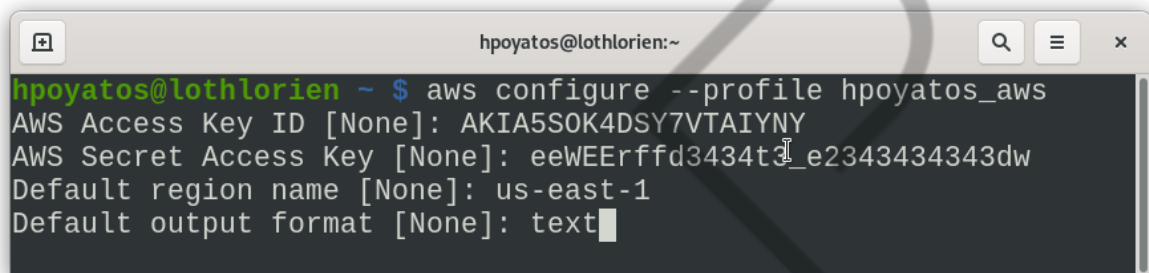


Figura 4.29 – Comando aws configure

Fonte: Elaborado pelo autor (2020)

Antes de rodar qualquer comando interessante usando o **aws cli**, já adianta que, se algum comando for tentado neste momento, teremos mensagens de erro como retorno, isso porque um erro foi cometido ao criar o grupo “CLIUsers” lá atrás, na Figura “Finalização do grupo ‘CLIUsers’ no IAM”: nenhuma política foi associada ao grupo em sua criação. Logo, o usuário “hpoyatos”, ainda que com as credenciais adequadas, não possui permissão para fazer nada! Então retorne ao gerenciador de grupo e associe a política apropriada: no exemplo, dei plenos poderes EC2 com a política “AmazonEC2FullAccess”. Repare, na Figura “Anexar política ao grupo ‘CLIUsers’”, que existem políticas menos permissivas, logo, trate-as de maneira adequada, afinal, grandes poderes trazem grandes responsabilidades, como já dizia o grande pensador Ben Parker.

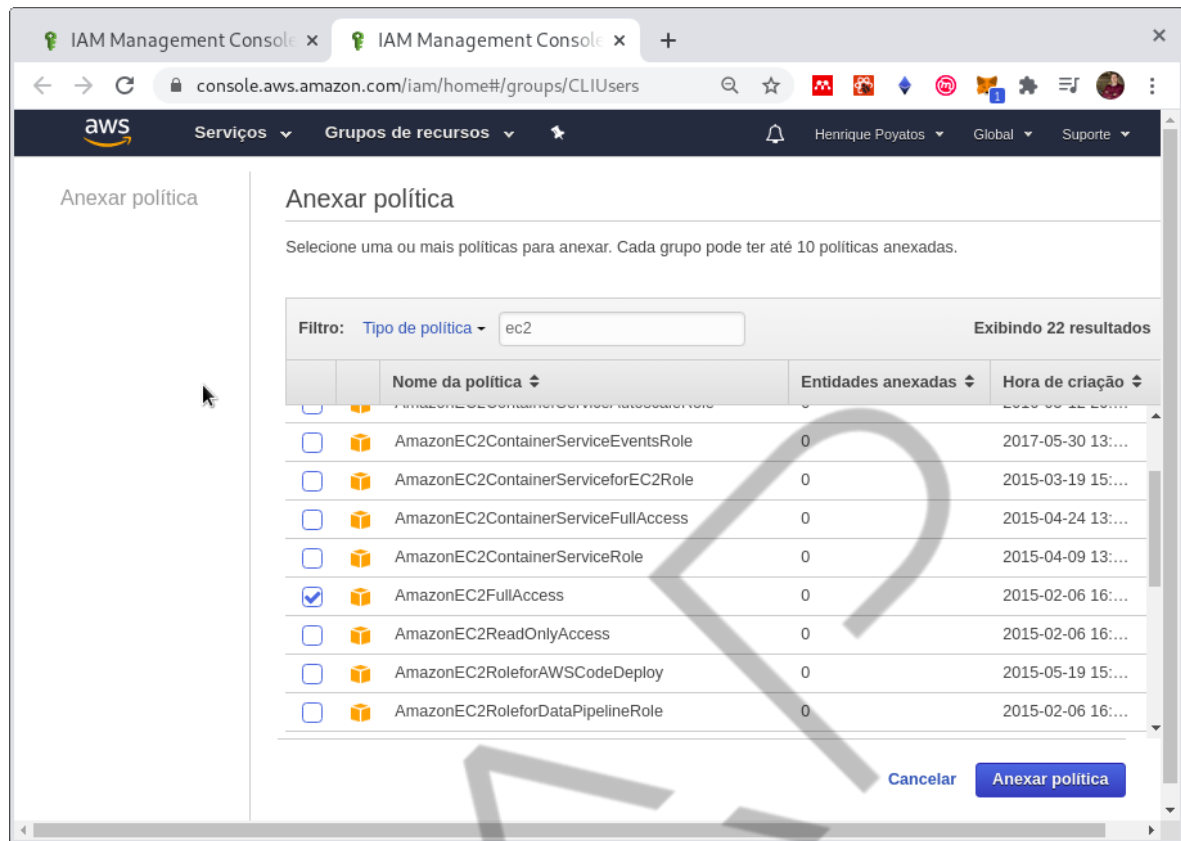


Figura 4.30 – Anexar política ao grupo “CLIUsers”

Fonte: Elaborado pelo autor (2020)

Como pode ser observado na documentação oficial, o utilitário AWS é poderosíssimo. Vamos começar listando ou descrevendo as instâncias disponíveis em nossa conta EC2, conforme o Comando de prompt “Comando para descrever as instâncias EC2 no AWS CLI” e a Figura “Comando aws ec2 describe-instances”.

```
$ aws ec2 describe-instances --profile <profile>
```

Comando de prompt 4.2 – Comando para descrever as instâncias EC2 no AWS CLI

Fonte: Elaborado pelo autor (2020)

```

hpooyatos@lothlorien ~ $
hpooyatos@lothlorien ~ $ aws ec2 describe-instances --profile hpooyatos_aws
RESERVATIONS 932970241201 r-0afc2e45d359538a2
INSTANCES 0 x86_64 False True xen ami-02354e95b39ca8dec i-087643300b1e20fa5 t2.micro aws_pair 2020-08-07T01:17:47.000Z ip-172-31-51-115.ec2.internal 172.31.51.115 /dev/xvda ebs True User initiated (2020-08-07 01:20:53 GMT) subnet-cab25afb hvm vpc-a3eb1ade
BLOCKDEVICEMAPPINGS /dev/xvda
EBS 2020-08-07T01:11:47.000Z True open attached vol-030d214f7e9018de6
CAPACITYRESERVATIONSPECIFICATION open
CPUOPTIONS 1 1
HIBERNATIONOPTIONS False
METADATATOPTIONS enabled 1 optional applied
MONITORING disabled
NETWORKINTERFACES interface 06:d7:bc:35:66:e1 eni-0096e5c4d3e85e3d0 932970241201 ip-172-31-51-115.ec2.internal 172.31.51.115 True in-use subnet-cab25afb vpc-a3eb1ade
ATTACHMENT 2020-08-07T01:11:47.000Z eni-attach-046e0e94050d648ce True 0 attached
GROUPS sg-07cf098d728c811bd launch-wizard-1
PRIVATEIPADDRESSES True ip-172-31-51-115.ec2.internal 172.31.51.115
PLACEMENT us-east-1e default
SECURITYGROUPS sg-07cf098d728c811bd launch-wizard-1
STATE 80 stopped
STATEREASON Client.UserInitiatedShutdown Client.UserInitiatedShutdown: User initiated shutdown

```

Figura 4.31 – Comando aws ec2 describe-instances

Fonte: Elaborado pelo autor (2020)

Como o id da instância em mãos (neste exemplo, “i-087643300b1e20fa5”), obtido com o comando `aws ec2 describe-instances` ou mesmo na listagem de instâncias no console EC2 no navegador web, podemos iniciar a instância via linha de comando, de acordo com o Comando de prompt “Comando para iniciar instâncias EC2 no AWS CLI” e a Figura “Comando aws ec2 start-instances”.

```
$ aws ec2 start-instances --instance-ids <ids das instâncias>
```

Comando de prompt 4.3 – Comando para iniciar instâncias EC2 no AWS CLI

Fonte: Elaborado pelo autor (2020)

```

hpooyatos@lothlorien ~ $ aws ec2 start-instances --instance-ids i-087643300b1e20fa5 --profile hpooyatos_aws
STARTINGINSTANCES i-087643300b1e20fa5
CURRENTSTATE 0 pending
PREVIOUSSTATE 80 stopped
hpooyatos@lothlorien ~ $
hpooyatos@lothlorien ~ $
hpooyatos@lothlorien ~ $

```

Figura 4.32 – Comando aws ec2 start-instances

Fonte: Elaborado pelo autor (2020)

Com a mesma facilidade que a instância foi levantada, ela pode ser derrubada. O Comando de prompt “Comando para derrubar instâncias EC2 no AWS CLI” e a Figura “Comando aws ec2 stop-instances” mostram o processo contrário. Não deixe

de verificar no console EC2 em seu navegador ou via acesso a SSH a instância respondendo (ou deixando de responder). Observe que, uma vez acesso via linha de comando, scripts que verificam a carga de acessos e criam ou levantam novas instâncias e as derrubam quando não são mais necessárias, ou que levantem instâncias em um determinado horário se tornam quase “brincadeira de criança”.

```
$ aws ec2 stop-instances --instance-ids <ids das instâncias>
```

Comando de prompt 4.4 – Comando para derrubar instâncias EC2 no AWS CLI  
Fonte: Elaborado pelo autor (2020)

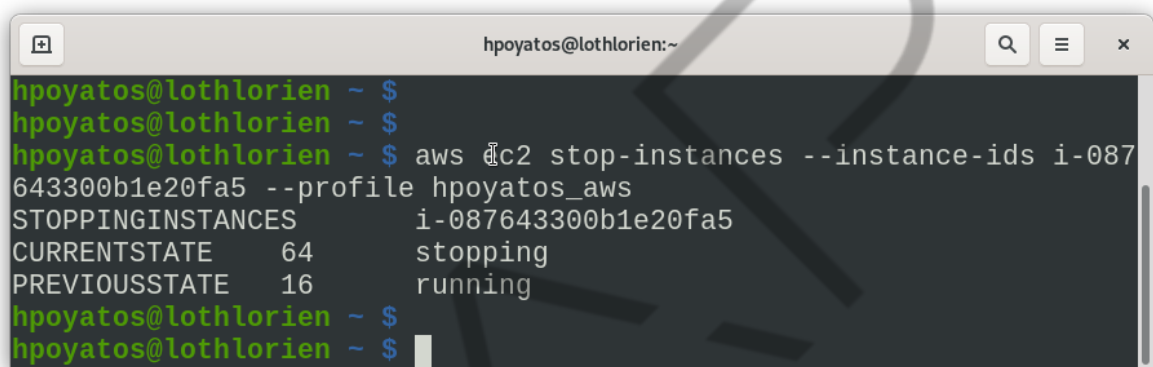
A screenshot of a terminal window titled 'hpoyatos@lothlorien:~'. The terminal shows a series of commands and their outputs. The first three lines are prompts: 'hpoyatos@lothlorien ~ \$', 'hpoyatos@lothlorien ~ \$', and 'hpoyatos@lothlorien ~ \$'. The fourth line is the command 'aws ec2 stop-instances --instance-ids i-087643300b1e20fa5 --profile hpoyatos\_aws'. The output shows 'STOPPINGINSTANCES i-087643300b1e20fa5', 'CURRENTSTATE 64 stopping', and 'PREVIOUSSTATE 16 running'. The final two lines are prompts: 'hpoyatos@lothlorien ~ \$' and 'hpoyatos@lothlorien ~ \$'.

Figura 4.33 – Comando aws ec2 stop-instances  
Fonte: Elaborado pelo autor (2020)

## CONCLUSÃO

A **Amazon Web Services (AWS)** é líder absoluta de mercado, resultado de vários anos de prestação de um serviço robusto, confiável e seguro. Por esse motivo, é indispensável para qualquer profissional que queira se especializar em *cloud computing* e dominar os principais produtos, como realizar planejamento, orçamentação, configuração e automação nesse ambiente. Tornar-se um profissional especializado em AWS é sinônimo de boas oportunidades profissionais e remuneração, em qualquer parte do mundo, então não deixe de absorver tais conhecimentos.

## REFERÊNCIAS

AWS. **AWS Certification.** [s.d.] a. Disponível em: <<https://aws.amazon.com/pt/certification/>>. Acesso em: 29 jul. 2020.

AWS. **AWS Certified Cloud Practitioner.** [s.d.] b. Disponível em: <<https://aws.amazon.com/pt/certification/certified-cloud-practitioner/>>. Acesso em: 29 jul. 2020.

AWS. **AWS Command Line Interface.** [s.d.] c. Disponível em: <<https://docs.aws.amazon.com/cli/latest/userguide/cli-chap-welcome.html>>. Acesso em: 7 ago. 2020.

AWS. **Infraestrutura global.** [s.d.] d. Disponível em: <<https://aws.amazon.com/pt/about-aws/global-infrastructure/>>. Acesso em: 29 jul. 2020.

AWS. **Netflix Case Study.** [s.d.] e. Disponível em: <<https://aws.amazon.com/pt/solutions/case-studies/netflix-case-study/>>. Acesso em: 29 jul. 2020.

AWS. **Recursos do Amazon EBS.** [s.d.] f. Disponível em: <[https://aws.amazon.com/pt/ebs/features/#SSD-backed\\_volumes\\_.28IOPS-intensive.29](https://aws.amazon.com/pt/ebs/features/#SSD-backed_volumes_.28IOPS-intensive.29)> Acesso em: 7 ago. 2020.

AWS. **Tipos de instância do Amazon EC2.** [s.d.] g. Disponível em: <<https://aws.amazon.com/pt/ec2/instance-types/>> Acesso em: 7 ago. 2020.

AWS. **AWS Well-Architected Framework.** 2020. Disponível em: <<https://aws.amazon.com/pt/architecture/well-architected/>>. Acesso em: 29 jul. 2020.

CANALYS. **Cloud market share Q4 2019 and full-year 2019.** 2019. Disponível em: <[https://www.canalys.com/static/press\\_release/2020/Canalys---Cloud-market-share-Q4-2019-and-full-year-2019.pdf](https://www.canalys.com/static/press_release/2020/Canalys---Cloud-market-share-Q4-2019-and-full-year-2019.pdf)>. Acesso em: 29 jul. 2020.

GARTNER. **Magic Quadrant for Cloud Infrastructure as a Service, Worldwide.** 2019. Disponível em: <<https://pages.awscloud.com/Gartner-Magic-Quadrant-for-Infrastructure-as-a-Service-Worldwide.html>>. Acesso em: 29 jul. 2020.

GLOBAL KNOWLEDGE. **2020IT Skills and Salary Report.** 2019. Disponível em: <<https://www.globalknowledge.com/us-en/content/salary-report/it-skills-and-salary-report/#reports>>. Acesso em: 29 jul. 2020.

GOODISON, D. **The 15 Top-Paying IT Certifications for 2020: Global Knowledge.** 2020. Disponível em: <<https://www.crn.com/slide-shows/cloud/the-15-top-paying-it-certifications-for-2020-global-knowledge>>. Acesso em: 29 jul. 2020.

TWENEY, D. **Amazon website goes down for 40 minutes, costing the company \$5 million.** 2013. Disponível em: <<https://venturebeat.com/2013/08/19/amazon-website-down/>>. Acesso em: 10 ago. 2020.