

Turma de PB ABR 2025 | Programa de Bolsas _ DevSecOps



Data da entrega da atividade	2025
Data final Apresentação	2025
Assunto	Atividade de Linux
Time Compass	Thiago Geremias De Oliveira, Yurisay Josefina Escudero Guevara, Davi Santos Cardoso Da Silva, e Patricia Maria Moura dos Santos

Atividade de Linux

Etapa 1: Configuração do Ambiente

- Criar uma VPC com com 2 sub-redes públicas e 2 privadas.
- Criar uma instância EC2 na AWS, com Sistema Operacional Ubuntu ou Amazon Linux 2023, na VPC criada e na sub-rede pública;

Etapa 2: Configuração do Servidor Web

- Subir um servidor Nginx na EC2;
- Criar uma página simples em html que será exibida dentro do servidor Nginx.

Etapa 3: Script de Monitoramento + Webhook

- Criar um script que verifique a cada 1 minutos se o site está disponível, ou seja se ele está rodando normalmente, caso a aplicação não esteja funcionando, o script deve envio uma notificação via algum desses canais, Discord, Telegram ou Slack, informando da indisponibilidade do serviço.
- O script deve armazenar os logs da sua execução em um local no servidor, por exemplo: /var/log/meu_script.log

Etapa 4: Testes e Documentação

- Testar a implementação.
- Fazer a documentação explicando o processo de instalação do Linux no Github.
- * Cuidado com dados que podem comprometer a segurança.

Desafios Bônus:

- A configuração da EC2 com o Nginx, página html e scripts de monitoramento são injetados automaticamente via User Data, para inicializarem junto com a máquina.
- Criar um arquivo Cloudformation que inicialize todo o ambiente.

Projeto: Configuração de Servidor Web com Monitoramento

Objetivo: Desenvolver e testar habilidades em Linux, AWS e automação de processos através da configuração de um ambiente de servidor web monitorado.

Etapa 1: Configuração do Ambiente

Tarefas:

1. Criar uma **VPC** na AWS com:
 - a. 2 **sub-redes públicas** (para acesso externo).
 - b. 2 **sub-redes privadas** (para futuras expansões).
 - c. Uma **Internet Gateway** conectada às sub-redes públicas.
2. Criar uma **instância EC2** na AWS:
 - a. Escolher uma **AMI baseada em Linux** (Ubuntu/Debian/Amazon Linux).
 - b. Instalar na **sub-rede pública** criada anteriormente.
 - c. Associar um **Security Group** que permita tráfego HTTP (porta 80) e SSH (porta 22, opcional).
3. **Acessar a instância via SSH** para realizar configurações futuras.

Etapa 2: Configuração do Servidor Web

Tarefas:

1. Instalar o servidor **Nginx** na EC2.
2. Criar uma **página HTML simples** para ser exibida pelo servidor.
3. Configurar o **Nginx para servir a página** corretamente.

□ Dicas para os alunos:

- **Personalizar a página** com informações sobre o projeto.
- Criar um **serviço systemd** para garantir que o Nginx reinicie automaticamente se parar.

Etapa 3: Monitoramento e Notificações

Tarefas:

1. Criar um script em Bash ou Python para monitorar a disponibilidade do site.
2. O script deve:
 - a. Verificar se o site responde corretamente a uma requisição HTTP.
 - b. Criar logs das verificações em `/var/log/monitoramento.log`.
 - c. Enviar uma notificação via Discord, Telegram ou Slack se detectar indisponibilidade.
3. Configurar o script para **rodar automaticamente a cada 1 minuto** usando cron ou systemd timers.

□ Dicas para os alunos:

- Usar `curl` no Bash ou `requests` no Python para testar a resposta do site.
- Configurar um **bot do Telegram** ou **webhook do Discord/Slack** para receber alertas.

Etapa 4: Automação e Testes

Tarefas:

1. **Testar** a implementação:
 - a. Verificar se o site está acessível via navegador.
 - b. Parar o Nginx e verificar se o script detecta e envia alertas corretamente.
2. Criar uma **documentação no GitHub** explicando:
 - a. Como configurar o ambiente.
 - b. Como instalar e configurar o servidor web.
 - c. Como funciona o script de monitoramento.
 - d. Como testar e validar a solução.

□ Dicas para os alunos:

- Explicar os comandos usados na documentação.
- Compartilhar prints dos testes.

Desafios Bônus (Opcional)

Para alunos que desejam se aprofundar mais:

1. **Automação com User Data:**
 - a. Configurar a EC2 para já iniciar com Nginx, HTML e script de monitoramento via **User Data**.
2. **Criação de um template CloudFormation ou Terraform:**
 - a. Criar um **arquivo YAML do CloudFormation** que provisiona toda a infraestrutura automaticamente.
3. **Monitoramento avançado:**
 - a. Enviar estatísticas para **CloudWatch** e configurar **alarmas na AWS**.