

FIAP GRADUAÇÃO

TECNOLOGIA EM DESENVOLVIMENTO DE SISTEMAS

DevOps Tools & Cloud Computing

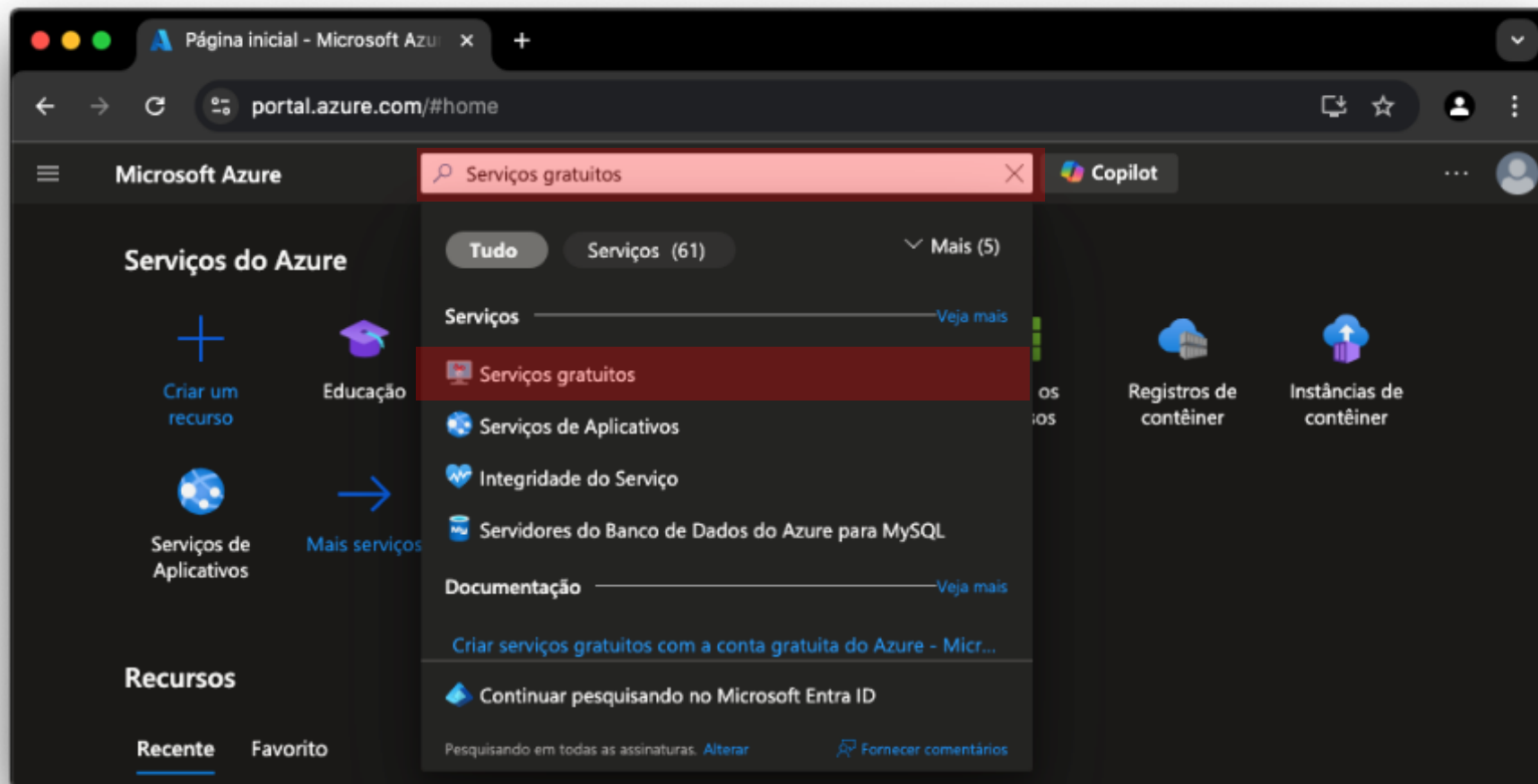
Preparação VM Linux Free – Azure CLI e Docker

PROF. JOÃO MENK profjoao.menk@fiap.com.br

PROF. Rafael Pereira profrafael.pereira@fiap.com.br

Criando a VM Linux Free - Azure

Vamos iniciar entrando no Portal da Azure e procurando por: **Serviços Gratuitos**



Criando a VM Linux Free - Azure

Existem vários Serviços Gratuitos na Azure, com uma cota diária ou mensal estipulada. Alguns somente pelo período de 12 meses e outros disponíveis sempre, desde que não zere seus créditos, pois a conta é desativada nessa condição

Serviços gratuitos

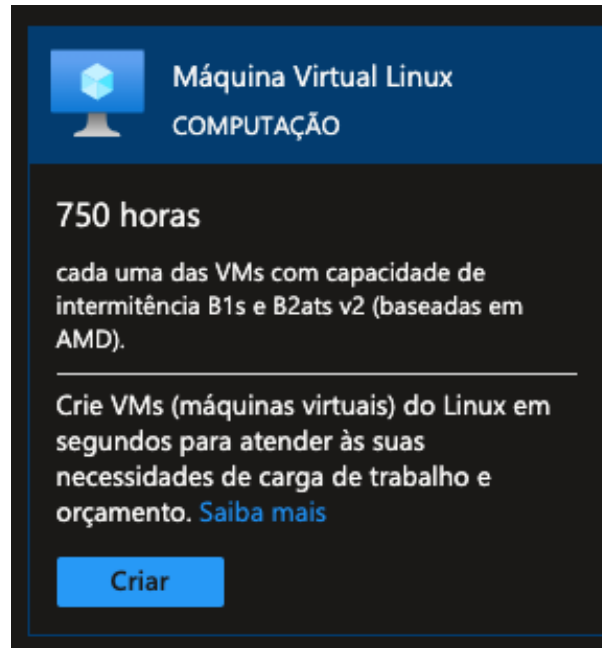
Serviços que incluem valores gratuitos mensais por 12 meses

Por 12 meses após a inscrição, novos clientes podem usar até o valor gratuito mensal especificado de cada um desses serviços sem ser cobrado. O uso do serviço é cobrado pelo valor de Pagamento Conforme o Uso depois que você atinge o **limite mensal**. Para saber mais, consulte as perguntas frequentes sobre a conta gratuita do Azure.

Serviço	Categoria	Limite Gratuito	Detalhes
Máquina Virtual do Windows	COMPUTAÇÃO	750 horas	cada uma das VMs com capacidade de intermitência B1s e B2ats v2 (baseadas em AMD). Crie VMs (máquinas virtuais) do Windows em segundos para atender às suas necessidades de carga de trabalho e orçamento. Saiba mais
Máquina Virtual Linux	COMPUTAÇÃO	750 horas	cada uma das VMs com capacidade de intermitência B1s e B2ats v2 (baseadas em AMD). Crie VMs (máquinas virtuais) do Linux em segundos para atender às suas necessidades de carga de trabalho e orçamento. Saiba mais
Azure Managed Disks	ARMAZENAMENTO	64 GB x 2	(P1) unidades de estado sólido armazenamento SSD, além de instantâneo de 1 GB e 2 milhões de operações de E/S. Obtenha um armazenamento em bloco de alto desempenho e durável para Máquinas Virtuais do Azure com gerenciamento simplificado. Saiba mais
Armazenamento de Blob do A...	ARMAZENAMENTO	5 GB	Hot block de armazenamento localmente redundante (LRS) com 20.000 operações de leitura e 10.000 de gravação. Use o armazenamento de objeto amplamente escalonável para qualquer tipo de dados não estruturados. Saiba mais
Arquivos do Azure	ARMAZENAMENTO	100GB	de transações LRS otimizadas, 1 arquivos quentes e frios. 2 milhões de operações de leitura, lista e outras operações de arquivo. Migre para o armazenamento de arquivos simples, distribuído, de plataforma cruzada sem alterar o código. Saiba mais
Key Vault	SEGURANÇA	10.000 transações	Chaves RSA de 2048 bits ou operações secretas, Camada padrão.
Codificação de Serviços de ML	MEDIA	20 minutos de saída	Codificação padrão do arquivo de fonte de áudio ou de vídeo do codificador.
Banco de Dados do Azure para...	BANCOS DE DADOS	750 horas	de servidor flexível—instância B1MS com capacidade de disparo, armazenamento de 32
Banco de dados do Azure para...	BANCOS DE DADOS	750 horas	de servidor flexível—instância B1MS com capacidade de disparo, armazenamento de 32
Azure Cosmos DB	BANCOS DE DADOS	Taxa de transferência provisionada de 400 RU/s	Com 25 GB de armazenamento

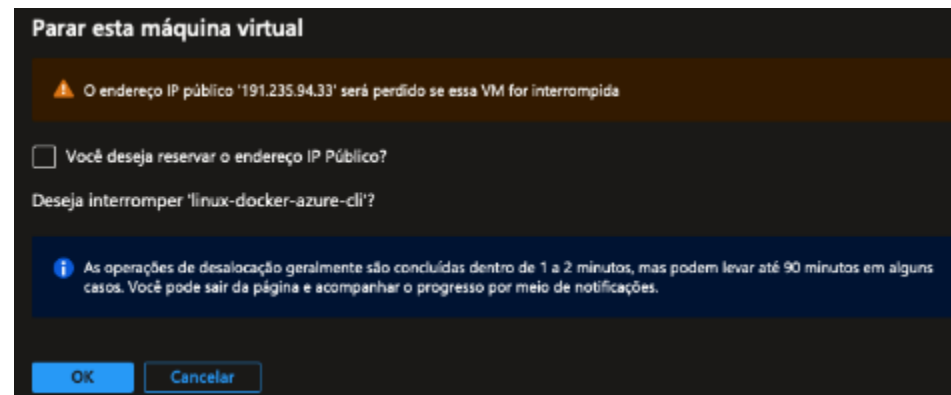
Criando a VM Linux Free - Azure

Vamos escolher **Máquina Virtual Linux**



Essa VM terá:

- ✓ 2 CPUs e 1GB RAM
- ✓ SSD Premium de 60GB
- ✓ IP Dinâmico



Criando a VM Linux Free - Azure

Nessa opção temos bem menos escolhas para definir as propriedades de nossa Máquina Virtual em nuvem, pois muitos recursos utilizados são pré-definidos

Microsoft Azure

Pesquisar recursos, serviços e documentos (G+/I)

Copilot

Página inicial > Serviços gratuitos >

Criar uma máquina virtual

Básico Marcas Revisar + criar

Crie uma máquina virtual que execute Linux ou Windows. Selecione uma imagem do Azure Marketplace ou use sua própria imagem personalizada. Conclua as guias Noções básicas e, em seguida, Revisar + criar para provisionar uma máquina virtual com parâmetros padrão ou revise cada guia para personalização completa. [Saiba mais](#)

Essa assinatura pode não estar qualificada para implantar VMs de determinados tamanhos em determinadas regiões.

Detalhes do projeto

Selecione a assinatura para gerenciar os custos e os recursos implantados. Use grupos de recursos como pastas para organizar e gerenciar todos os seus recursos.

Assinatura * p0841 - Azure para Estudantes

Grupo de recursos * rg-linux-free

[Criar novo](#)

Detalhes da instância

Nome da máquina virtual * linux-docker-azure-dl

Região * (South America) Brazil South

Imagem * AlmaLinux OS 9 - x64 Gen2

< Anterior Avançar: Marcas > Revisar + criar

[Enviar comentários](#)

Criando a VM Linux Free - Azure

Nessa opção temos bem menos escolhas para definir as propriedades de nossa Máquina Virtual em nuvem, pois muitos recursos utilizados são pré-definidos

The screenshot shows the 'Criar uma máquina virtual' (Create a virtual machine) page in the Microsoft Azure portal. The page is in Portuguese and displays the following configuration options:

- Tamanho (Size):** Standard_B2ats_v2 - 2 vcpus, 1 GiB memória (R\$ 56,25/mês) (serviços gratuitos).
- Conta de administrador (Administrator account):**
 - Tipo de Autenticação (Authentication type):** Senha (Password) is selected.
 - Nome de usuário (Username):** adminx
 - Senha (Password):** [Redacted]
 - Confirmar senha (Confirm password):** [Redacted]
- Regras de portas de entrada (Inbound ports rules):**
 - Portas de entrada públicas (Public inbound ports):** Permitir portas selecionadas (Allow selected ports) is selected.
 - Selecione as portas de entrada (Select inbound ports):** SSH (22)

At the bottom, there are navigation buttons: Anterior, Avançar: Marcas, and Revisar + criar. A link for 'Enviar comentários' (Send comments) is also visible.

Criando a VM Linux Free - Azure

Nessa opção temos bem menos escolhas para definir as propriedades de nossa Máquina Virtual em nuvem, pois muitos recursos utilizados são pré-definidos

The screenshot shows the 'Criar uma máquina virtual' (Create a virtual machine) page in the Azure portal, specifically for the Linux Free tier. The page is in Portuguese and shows the 'Revisar + criar' (Review + create) step. A green banner at the top indicates 'Validação aprovada' (Validation approved). The 'Preço' (Price) section shows a price of 0.0770BRL/hora for the 1 X Standard B2ats v2 VM size. The 'TERMO' (Terms) section contains a disclaimer about the free offer. At the bottom, there is a 'Nome' (Name) field with the value 'Joao Mene' and a 'Criar' (Create) button highlighted with a red box.

Microsoft Azure

portal.azure.com/#create/microsoft.freeaccountvirtualmachine-linux

Pesquisar recursos, serviços e documentos (G+/I)

Copilot

Página inicial > Serviços gratuitos >

Criar uma máquina virtual

Validação aprovada

Básico Marcas Revisar + criar

O custo fornecido abaixo é uma estimativa e não o preço final. Para todas as suas necessidades de preços, use a calculadora de preços. [Calculadora de preços](#)

Preço

1 X Standard B2ats v2
por Microsoft
[Termos de uso](#) [Política de privacidade](#)

Créditos de assinatura se aplicam

0.0770BRL/hora
Preços de outros tamanhos de VM

TERMO

Ao clicar em "Criar", eu (a) concordo com os termos legais e as políticas de privacidade associadas às ofertas do Marketplace listadas acima; (b) autorizo a Microsoft a cobrar minha forma de pagamento atual pelas taxas associadas às ofertas, com a mesma frequência de cobrança que minha assinatura do Azure e (c) concordo que a Microsoft possa compartilhar minhas informações de contato, de uso e de transações com os provedores das ofertas para fins de suporte, cobrança e outras atividades transacionais. A Microsoft não fornece direitos para ofertas de terceiros. Veja os [Termos do Azure Marketplace](#) para obter mais detalhes.

Nome: Joao Mene

< Anterior Avançar > **Criar**

Baixar um modelo para automação Enviar comentários

Criando a VM Linux Free - Azure

Aguarde a criação do nosso Servidor Virtual e clique em **Ir para o Recurso**

The screenshot shows the Microsoft Azure portal interface. The browser address bar displays the URL: `portal.azure.com/#view/HubsExtension/DeploymentDetailsBlade/~/overview/id/%2Fsu...`. The page title is "CreateVm-almalinux.almalinux-x86_64-9-gen2-20240928145747 | ...". The left sidebar contains a search bar and a list of navigation items: "Visão Geral" (selected), "Entradas", "Saídas", and "Modelo". The main content area shows a green checkmark icon and the text "A implantação foi concluída". Below this, there are details about the deployment: "Nome da implantação:...", "Assinatura: pl0841 - Az...", "Grupo de recursos: rg-l...", "Hora de início: 2...", and "ID de Correlação: 6". There are also sections for "Detalhes de implantação" and "Próximas etapas", which include links for "Configurar desligamento automático", "Monitorar dependências de rede, desempenho e integridade da VM", and "Executar um script dentro da máquina virtual". On the right side, there are sections for "Gerenciamento de Custos" and "Microsoft Defender for Cloud". At the bottom, there is a button labeled "Ir para o recurso" which is highlighted with a red box, and another button labeled "Criar outra VM".

Criando a VM Linux Free - Azure

01) Navegue até a **opção Rede** e Configuração de rede

02) Abra as portas que mais iremos utilizar em nossos exercícios:

80

8080

3000

5000

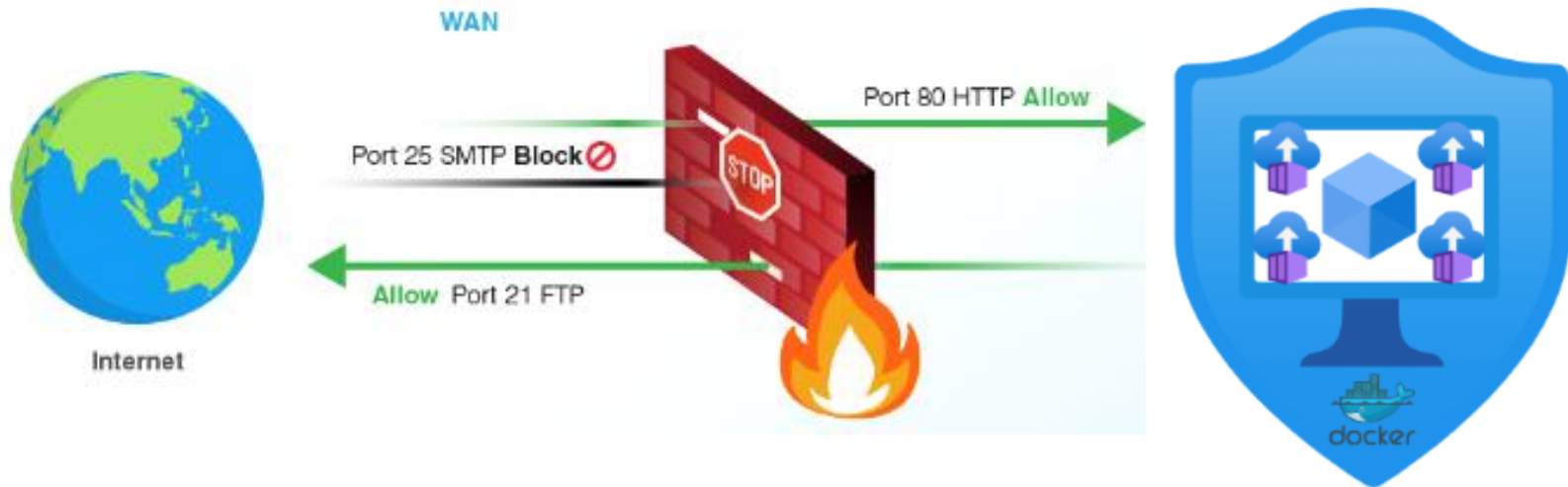
443 (Serviço HTTPS)

Regras de portas de entrada (9)						
300	SSH	22	TCP	Qualquer	Qualquer	Allow
310	Porta_8080	8080	Qualquer	Qualquer	Qualquer	Allow
320	Porta_80	80	Qualquer	Qualquer	Qualquer	Allow
330	Porta_3000	3000	Qualquer	Qualquer	Qualquer	Allow
340	Porta_5000	5000	Qualquer	Qualquer	Qualquer	Allow
350	Porta_443	443	TCP	Qualquer	Qualquer	Allow

Criando a VM Linux Free - Azure

03) Lembre-se sempre:

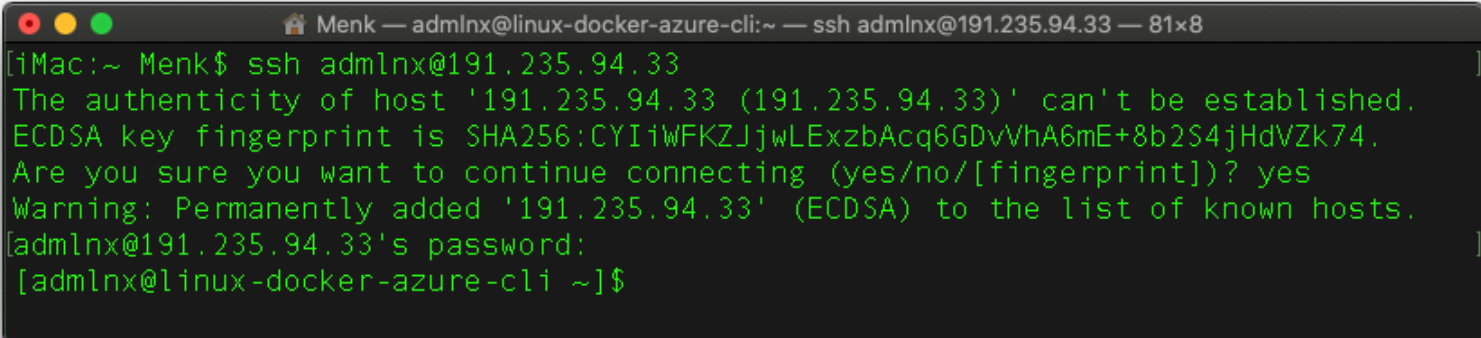
Qualquer Aplicativo ou API que rodar nesse Servidor Virtual deve-se verificar se a porta de escuta está liberada para podermos acessar externamente



Instalando alguns Softwares

Agora que temos nosso Servidor Virtual operante e com configurações iniciais executadas vamos instalar algumas ferramentas necessárias

1) Acesse sua VM através do terminal da sua máquina local com **SSH**

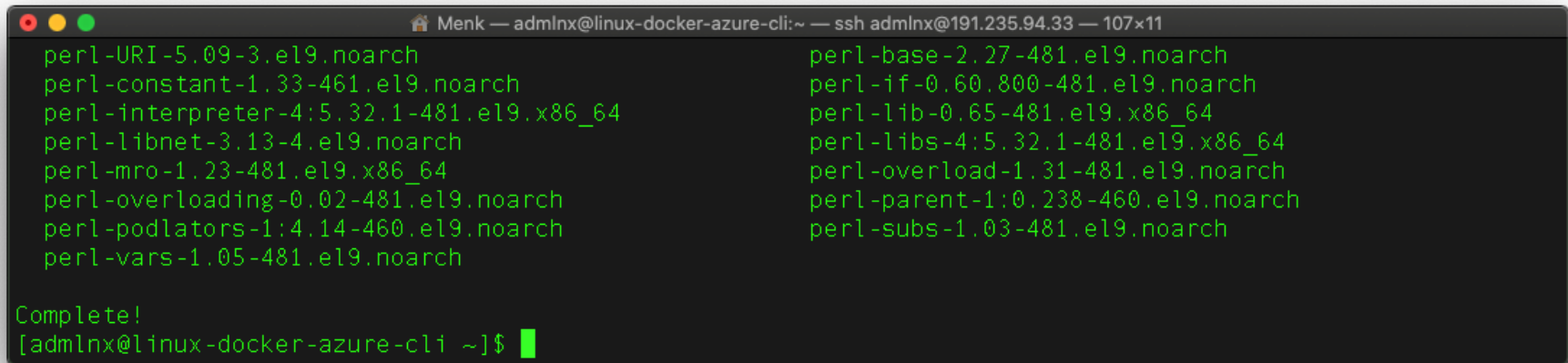
A terminal window titled 'Menk' with a subtitle 'admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 81x8'. The window shows the process of establishing an SSH connection to a remote host. The user enters 'ssh admlnx@191.235.94.33'. The terminal displays a warning about the host's authenticity, showing the ECDSA key fingerprint. The user responds with 'yes'. The terminal then shows the password prompt and the user's input, followed by the successful connection to the remote host.

```
iMac:~ Menk$ ssh admlnx@191.235.94.33
The authenticity of host '191.235.94.33 (191.235.94.33)' can't be established.
ECDSA key fingerprint is SHA256:CYIiWFKZJjwLExzbAcq6GDvVhA6mE+8b2S4jHdVZk74.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '191.235.94.33' (ECDSA) to the list of known hosts.
[admlnx@191.235.94.33's password:
[admlnx@linux-docker-azure-cli ~]$
```

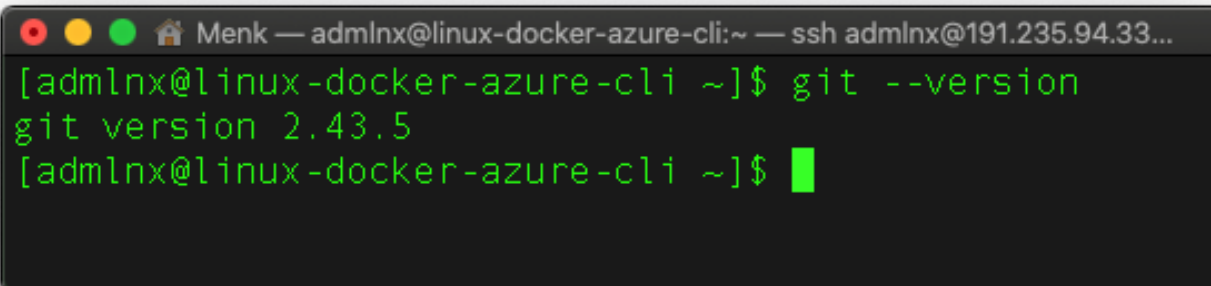
Instalando o Git

Para instalar o Git execute o comando abaixo

```
sudo yum install git -y
```

A terminal window titled 'Menk' with the prompt 'admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 107x11'. The output of the command 'sudo yum install git -y' is displayed in two columns. The first column lists various perl-related packages being installed, including perl-URI, perl-constant, perl-interpreter, perl-libnet, perl-mro, perl-overloading, perl-podlators, and perl-vars. The second column lists additional perl-related packages, including perl-base, perl-if, perl-lib, perl-libs, perl-overload, perl-parent, and perl-subs. The output ends with 'Complete!' and the prompt '[admlnx@linux-docker-azure-cli ~]\$'.

```
git --version
```

A terminal window titled 'Menk' with the prompt 'admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33...'. The output of the command 'git --version' is displayed as 'git version 2.43.5'. The prompt '[admlnx@linux-docker-azure-cli ~]\$' is shown at the bottom.

Instalando o NANO

O nano é um editor de texto muito utilizado para distribuições Linux sem interface gráfica. Vamos instalar esse editor para caso seja necessário

```
sudo yum install nano -y
```

```
Menk — adminx@linux-docker-azure-cli:/spring-h2-maven — ssh adminx@191.235.94.33 — 155x29
[adminx@linux-docker-azure-cli spring-h2-maven]$ sudo yum install nano -y
Last metadata expiration check: 1:21:45 ago on Sat Sep 28 18:52:23 2024.
Dependencies resolved.
=====
Package                                Architecture      Version           Repository        Size
=====
Installing:
 nano                                  x86_64            5.6.1-5.el9      baseos            690 k
Transaction Summary
=====
Install 1 Package

Total download size: 690 k
Installed size: 2.7 M
Downloading Packages:
nano-5.6.1-5.el9.x86_64.rpm                                                    472 kB/s | 690 kB  00:01
-----
Total                                                                           345 kB/s | 690 kB  00:02

Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
```

Instalando a Azure CLI

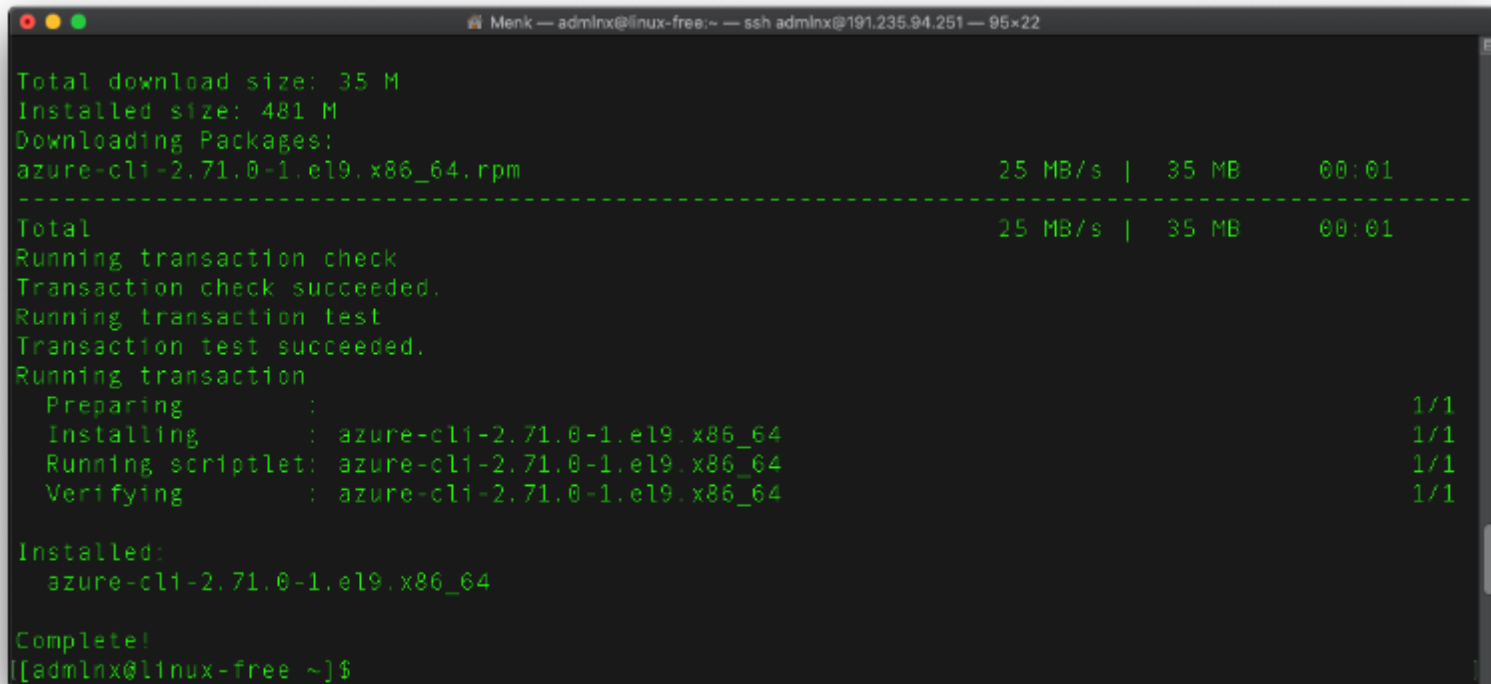
1) Rode os comandos abaixo no terminal para instalar a Azure CLI

```
sudo rpm --import https://packages.microsoft.com/keys/microsoft.asc
```

```
sudo dnf install -y https://packages.microsoft.com/config/rhel/9.0/packages-microsoft-prod.rpm
```

```
sudo dnf install azure-cli -y
```

No futuro mantenha sua versão sempre atualizada com o comando abaixo:
az upgrade



```
Menk — adminx@linux-free:~ — ssh adminx@191.235.94.251 — 95x22

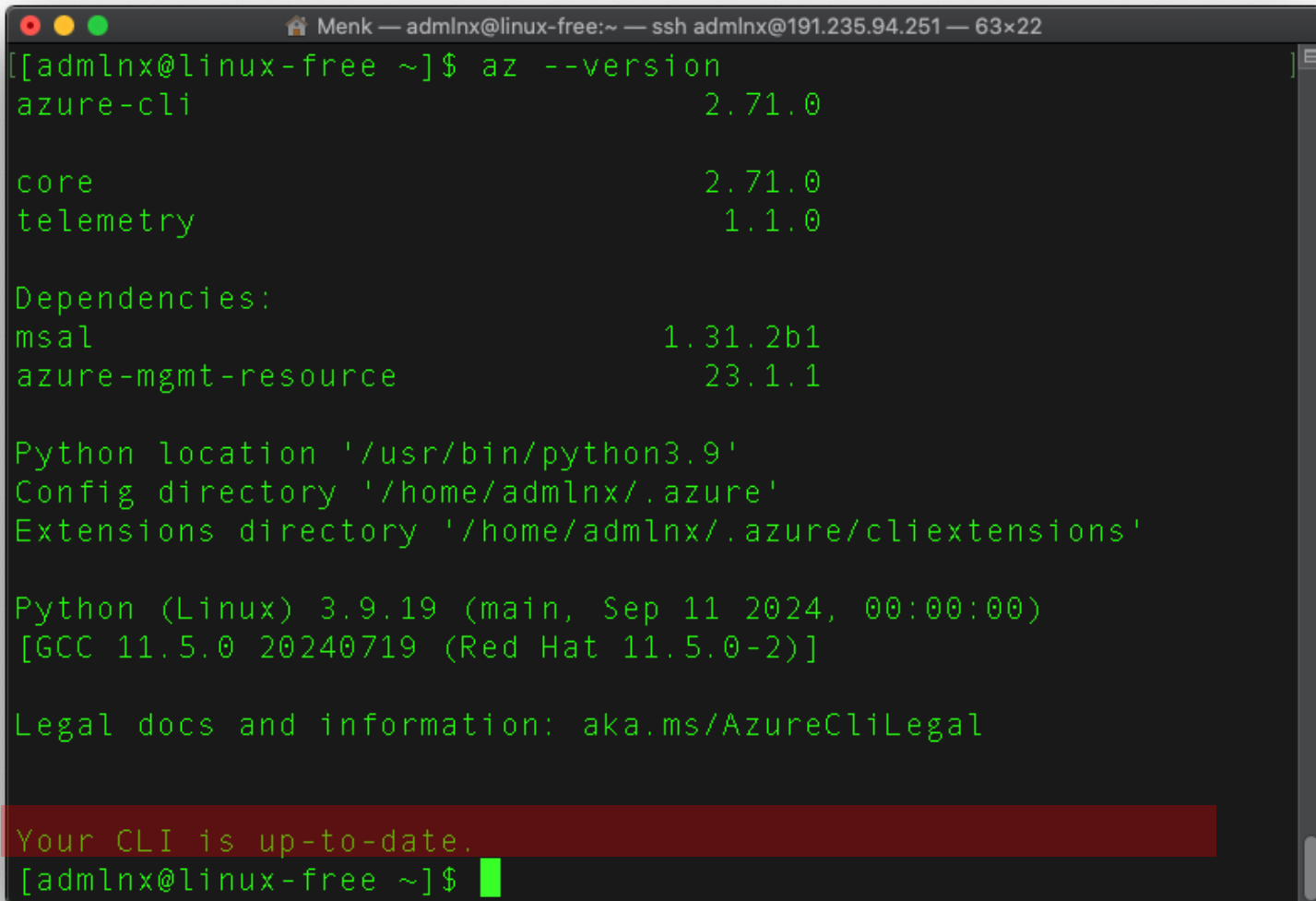
Total download size: 35 M
Installed size: 481 M
Downloading Packages:
azure-cli-2.71.0-1.el9.x86_64.rpm                25 MB/s | 35 MB    00:01
-----
Total                                           25 MB/s | 35 MB    00:01
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :                                1/1
  Installing     : azure-cli-2.71.0-1.el9.x86_64  1/1
  Running scriptlet: azure-cli-2.71.0-1.el9.x86_64  1/1
  Verifying      : azure-cli-2.71.0-1.el9.x86_64  1/1

Installed:
  azure-cli-2.71.0-1.el9.x86_64

Complete!
[[adminx@linux-free ~]]$
```

Instalando a Azure CLI

2) Verifique o sucesso da instalação com o comando
az --version



```
Menk — admlnx@linux-free:~ — ssh admlnx@191.235.94.251 — 63x22
[[admlnx@linux-free ~]$ az --version
azure-cli                        2.71.0

core                            2.71.0
telemetry                       1.1.0

Dependencies:
msal                            1.31.2b1
azure-mgmt-resource            23.1.1

Python location '/usr/bin/python3.9'
Config directory '/home/admlnx/.azure'
Extensions directory '/home/admlnx/.azure/cliextensions'

Python (Linux) 3.9.19 (main, Sep 11 2024, 00:00:00)
[GCC 11.5.0 20240719 (Red Hat 11.5.0-2)]

Legal docs and information: aka.ms/AzureCliLegal

Your CLI is up-to-date.
[admlnx@linux-free ~]$
```


Instalando o Docker

Ainda no Terminal de controle do Servidor Virtual vamos instalar o Docker (sem interface gráfica)

01) Execute os comandos abaixo para realizar a instalação

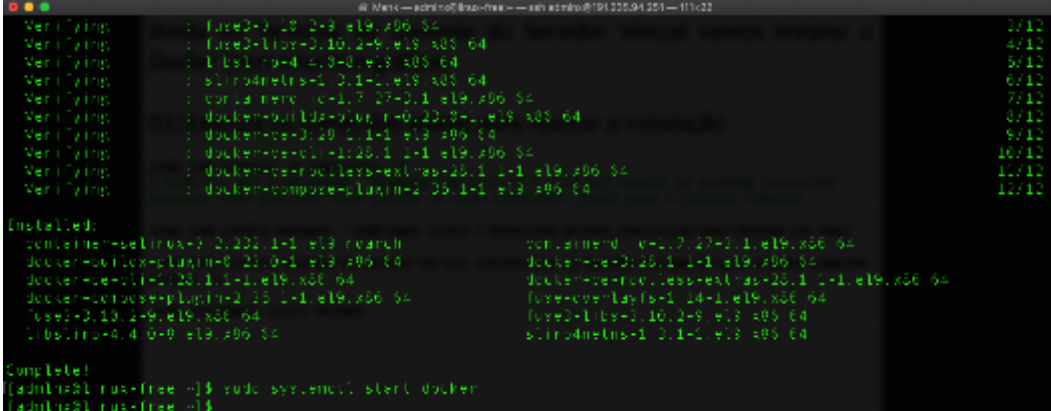
```
sudo yum install -y yum-utils -y
```

Pode ser que a mensagem: "Package yum-utils-4.3.0-13.el9.noarch is already installed" apareça. Caso positivo esse pacote já está instalado, passe para o próximo comando

```
sudo yum-config-manager --add-repo https://download.docker.com/linux/rhel/docker-ce.repo
```

```
sudo yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin -y
```

```
sudo systemctl start docker
```



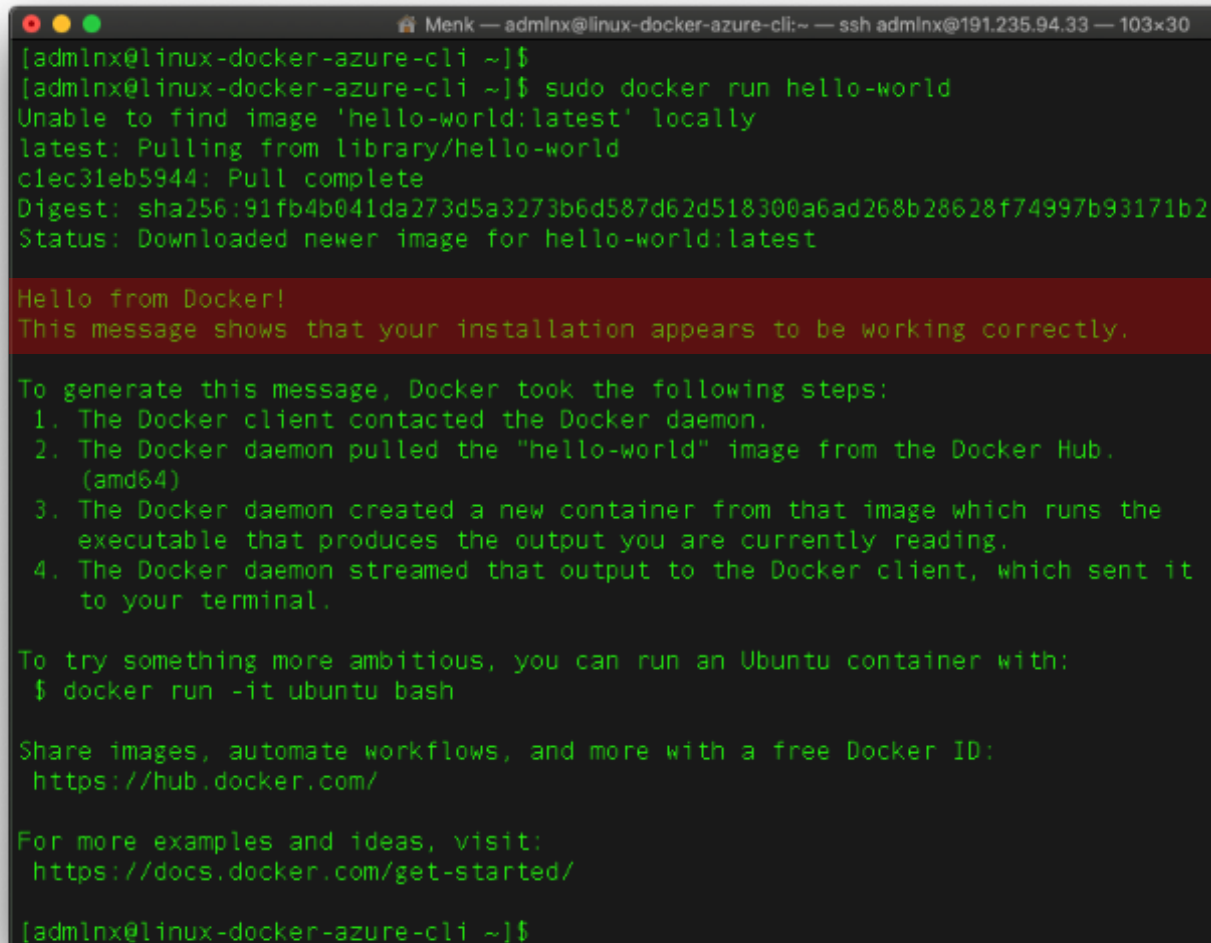
```
Verifying : fuse2-libs-2.9-9.el9.x86_64                                17/12
Verifying : fuse2-libs-2.10.2-9.el9.x86_64                            47/12
Verifying : libslirp-4.4.0-8.el9.x86_64                              55/12
Verifying : slirp4netns-1.2.1-1.el9.x86_64                           63/12
Verifying : containerd-1.7.27-1.el9.x86_64                           71/12
Verifying : docker-buildx-plugin-0.20.0-1.el9.x86_64                 83/12
Verifying : docker-ce-20.10.1-1.el9.x86_64                          91/12
Verifying : docker-ce-cli-20.10.1-1.el9.x86_64                      103/12
Verifying : docker-ce-rootless-extras-20.10.1-1.el9.x86_64          111/12
Verifying : docker-compose-plugin-2.25.1-1.el9.x86_64               123/12
Installed:
  containerd-1.7.27-1.el9.x86_64
  docker-buildx-plugin-0.20.0-1.el9.x86_64
  docker-ce-20.10.1-1.el9.x86_64
  docker-ce-cli-20.10.1-1.el9.x86_64
  docker-ce-rootless-extras-20.10.1-1.el9.x86_64
  docker-compose-plugin-2.25.1-1.el9.x86_64
  fuse2-2.10.2-9.el9.x86_64
  libslirp-4.4.0-8.el9.x86_64
  slirp4netns-1.2.1-1.el9.x86_64
  containerd-1.7.27-1.el9.x86_64
  docker-ce-20.10.1-1.el9.x86_64
  docker-ce-rootless-extras-20.10.1-1.el9.x86_64
  fuse2-libs-2.10.2-9.el9.x86_64
  slirp4netns-1.2.1-1.el9.x86_64

Completed!
[admin@21 ru5-free ~]$ sudo systemctl start docker
[admin@21 ru5-free ~]$
```

Instalando o Docker

02) Teste a instalação com o comando abaixo

`sudo docker run hello-world`



```
Menk — adminx@linux-docker-azure-cli:~ — ssh adminx@191.235.94.33 — 103x30
[adminx@linux-docker-azure-cli ~]$
[adminx@linux-docker-azure-cli ~]$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
c1ec31eb5944: Pull complete
Digest: sha256:91fb4b041da273d5a3273b6d587d62d518300a6ad268b28628f74997b93171b2
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

[adminx@linux-docker-azure-cli ~]$
```

Instalando o Docker

03) Para evitarmos ter que executar sempre com o comando **sudo** do Linux, vamos incluir o nosso usuário no grupo de usuário do Docker

```
Menk — admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 120x6
[admlnx@linux-docker-azure-cli ~]$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.47/containers/json": dial unix /var/run/docker.sock: connect: permission denied
[admlnx@linux-docker-azure-cli ~]$
```

sudo usermod -aG docker admlnx

```
Menk — admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 81x5
[admlnx@linux-docker-azure-cli ~]$ sudo usermod -aG docker admlnx
[admlnx@linux-docker-azure-cli ~]$
[admlnx@linux-docker-azure-cli ~]$
```

Após esse comando, saia do terminal e entre novamente para efetivar

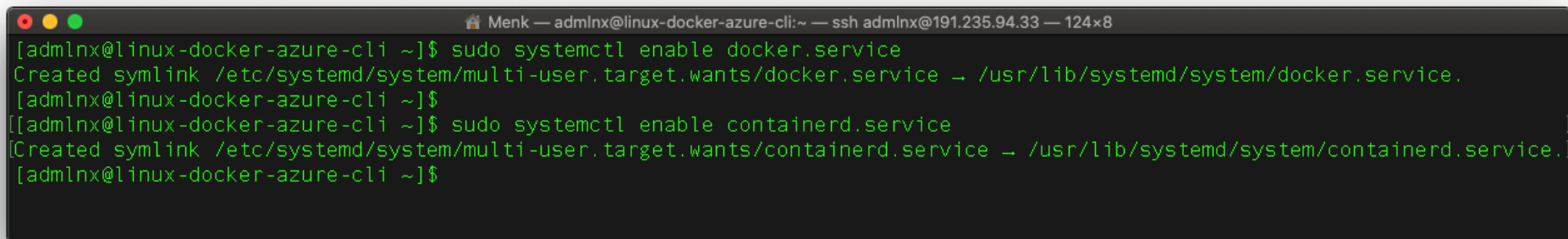
```
Menk — admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 70x5
[admlnx@linux-docker-azure-cli ~]$ docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
[admlnx@linux-docker-azure-cli ~]$
```

| Instalando o Docker

04) Para evitarmos ter que iniciar sempre o serviço do Docker vamos inclui-lo no início do sistema. Execute os seguintes comandos no terminal

```
sudo systemctl enable docker.service
```

```
sudo systemctl enable containerd.service
```

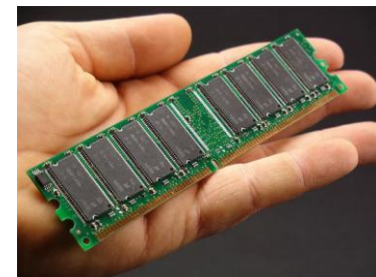
A terminal window with a dark background and green text. The window title bar shows 'Menk — admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 124x8'. The terminal content shows two commands being executed: 'sudo systemctl enable docker.service' and 'sudo systemctl enable containerd.service'. Each command is followed by a confirmation message: 'Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.' and 'Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.' respectively. The prompt '[admlnx@linux-docker-azure-cli ~]\$' is visible at the start of each line.

```
Menk — admlnx@linux-docker-azure-cli:~ — ssh admlnx@191.235.94.33 — 124x8
[admlnx@linux-docker-azure-cli ~]$ sudo systemctl enable docker.service
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service → /usr/lib/systemd/system/docker.service.
[admlnx@linux-docker-azure-cli ~]$
[[admlnx@linux-docker-azure-cli ~]$ sudo systemctl enable containerd.service
Created symlink /etc/systemd/system/multi-user.target.wants/containerd.service → /usr/lib/systemd/system/containerd.service.
[admlnx@linux-docker-azure-cli ~]$
```

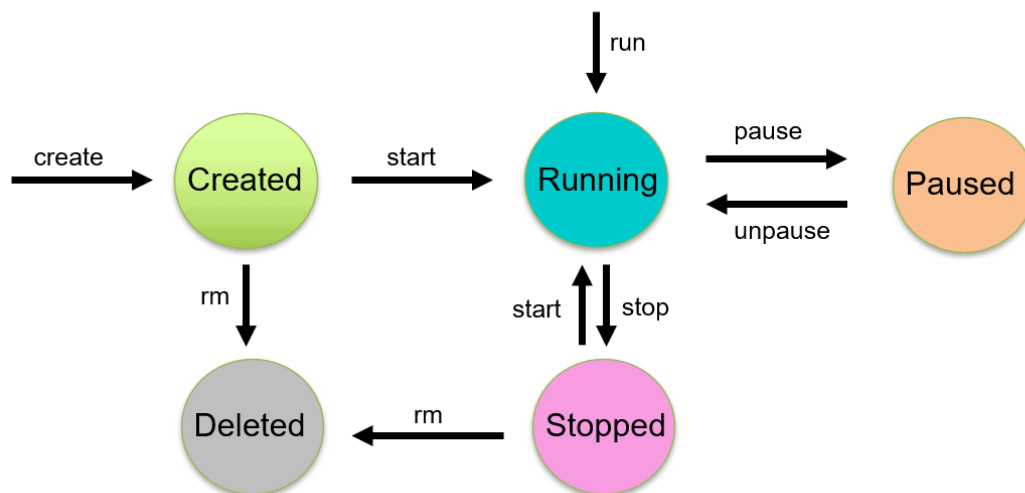
Testando o Servidor Virtual

FIAP

Vamos cuidar da nossa memória RAM, pois temos apenas 1GB...



A cada teste que fizer ou projeto finalizado, **remova o container, imagem e dependências (Volume, Network etc)**



| Testando o Servidor Virtual



Vamos fazer o seguinte exercício para testar nosso Servidor Virtual criando Containers e acessando-os remotamente

1) Rode o Container

```
docker container run --name nginx -d -p 80:80 nginx
```

2) Acesse seu Web Browser em **IPdaVM:80** e verifique se o servidor funciona

Ex.: 191.235.94.34:80

3) Após o Teste bem sucedido apague o Container e a Imagem

```
docker container rm -f nginx  
docker image rm nginx
```

REMOVENDO CONTAINERS, IMAGENS, VOLUMES



A opção `rm` pode ser usada em outros objetos do Docker como Imagens e Volumes também. Vamos aprender uma forma simples de limpar os nossos recursos criados no Docker

O comando abaixo encerra a execução de todos os Containers no Host
`docker container stop $(docker container ls -aq)`



O comando abaixo remove todos os Containers do Host
`docker container rm $(docker container ls -aq)`

O comando abaixo remove todas as Imagens no Host
`docker image rm $(docker image ls -aq)`



O comando abaixo remove todas os Volumes no Host
`docker volume rm $(docker volume ls -q)`



REMOVENDO CONTAINERS, IMAGENS, VOLUMES

O comando abaixo remove:

- ✓ Todos os Containers parados
- ✓ Todas as Redes não usadas
- ✓ Todos os Volumes não usados
- ✓ Todas as Imagens sem nenhum Container associado
- ✓ Todo o Cache Build pendente
- ✓ Não pede confirmação



`docker system prune -a -f --volumes`

```
Menk — -bash — 76x8
iMac:~ Menk$ docker system prune --all --force --volumes
Deleted build cache objects:
puu9ykhgum7fmd4a18o1fencf
t70qo51dz462l28jk5oowlztm

Total reclaimed space: 0B
iMac:~ Menk$
```


Copyright © 2025 Prof. João Carlos Menk

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor)