



Adm. e Segurança de Redes de Computadores

Docker
C.T. Informática
Prof. Vinícius Alves Hax



Antes

- Criando imagem com o Docker



Hoje

- Modos de rede do VirtualBox
- Configuração de rede no Linux

Principais modos de rede do VirtualBox

- Não conectado: Simula como se o cabo estivesse desconectado
- NAT: Modo padrão. Através de um NAT com a máquina física a máquina acessa a Internet mas não pode ser acessada
- Rede NAT: Mesmo que o anterior porém máquina virtual pode ser acessada (requer configuração específica)
- Bridged: Máquina virtual usa a placa de rede como se fosse um host comum


Principais modos de rede do VirtualBox (2)

- Rede interna: Rede somente entre as Vms
- Rede exclusiva de hospedeiro: Semelhante ao anterior mas o host participa da rede com uma placa de rede virtual
- Rede cloud: Permite conectar a VM local em um serviço de nuvem (VMs remotas)
- Driver genérico: Permite que o usuário escolha um driver diferente para a placa de rede (pouco usado)

Modos de rede (resumo)

Table 6.1. Overview of Networking Modes

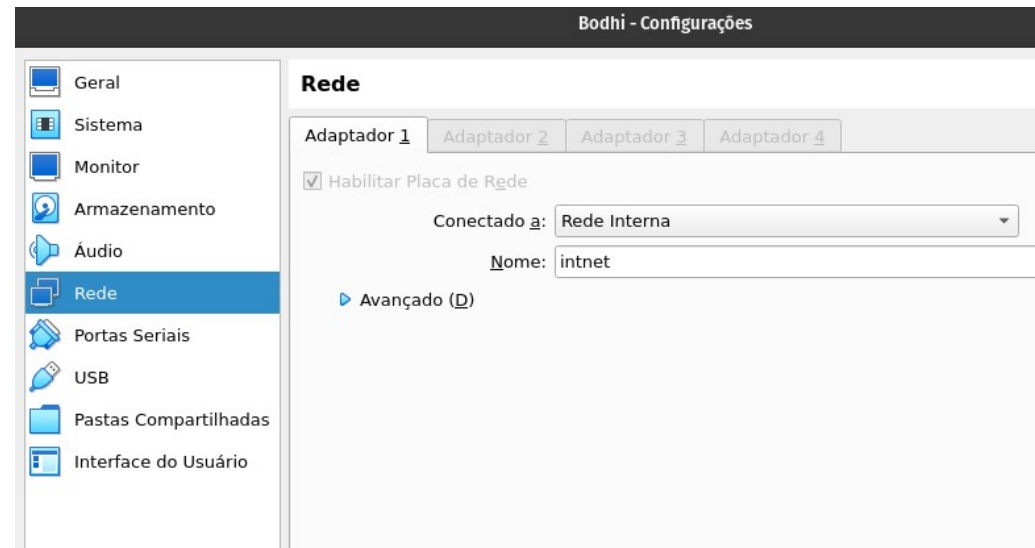
Mode	VM→Host	VM←Host	VM1↔VM2	VM→Net/LAN	VM←Net/LAN
Host-only	+	+	+	-	-
Internal	-	-	+	-	-
Bridged	+	+	+	+	+
NAT	+	Port forward	-	+	Port forward
NATservice	+	Port forward	+	+	Port forward



Para as atividades a seguir iremos usar o modo
“Rede Interna”. Esse modo permite testes
isolados do “mundo exterior”

Ativando o modo “Rede Interna”

- Selecione a máquina virtual e clique com o botão direito. Escolha “Configurações”
- Dentro de configurações escolha “Rede”



Montando nossa rede



PC Linux
IP: 192.168.0.1
Máscara: 255.255.255.0

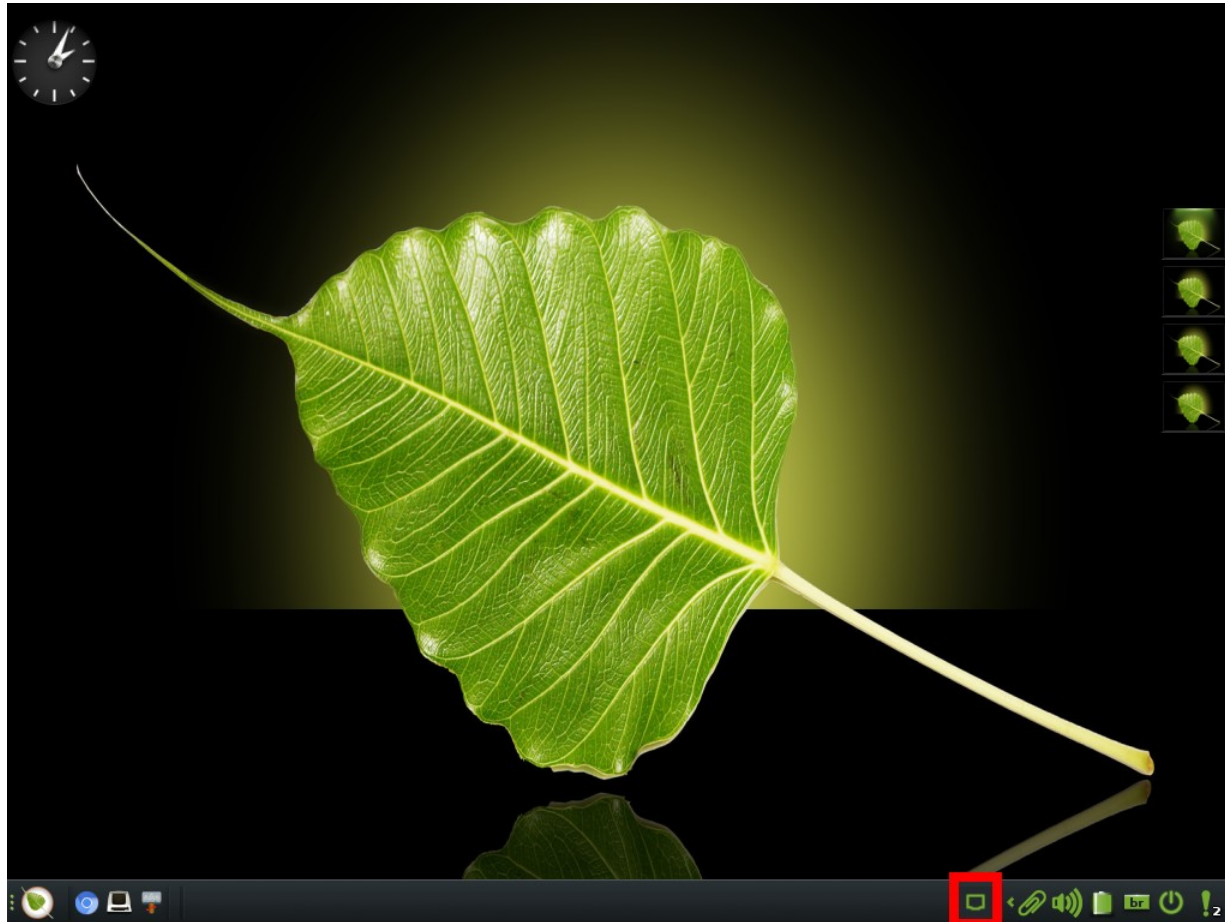
Ubuntu Server
IP: 192.168.0.2
Máscara: 255.255.255.0



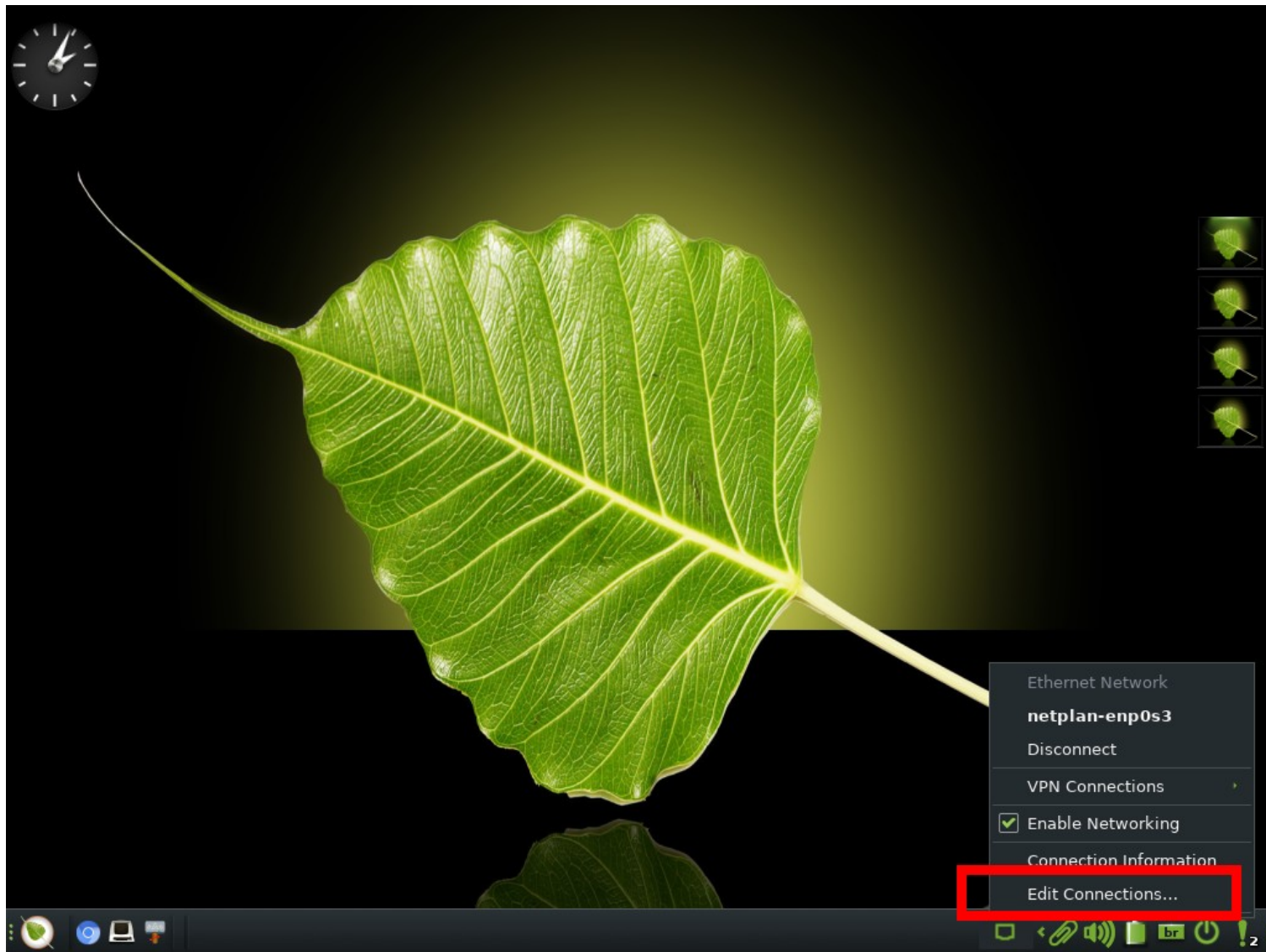
Configuração por GUI (*graphic user interface*)

- Vantagens
 - Não exige muitos conhecimentos técnicos
 - Interface verifica erros comuns no preenchimento
- Desvantagens
 - Muda bastante de sistema para sistema (principalmente no Linux)
 - Nem sempre permite parâmetros avançados

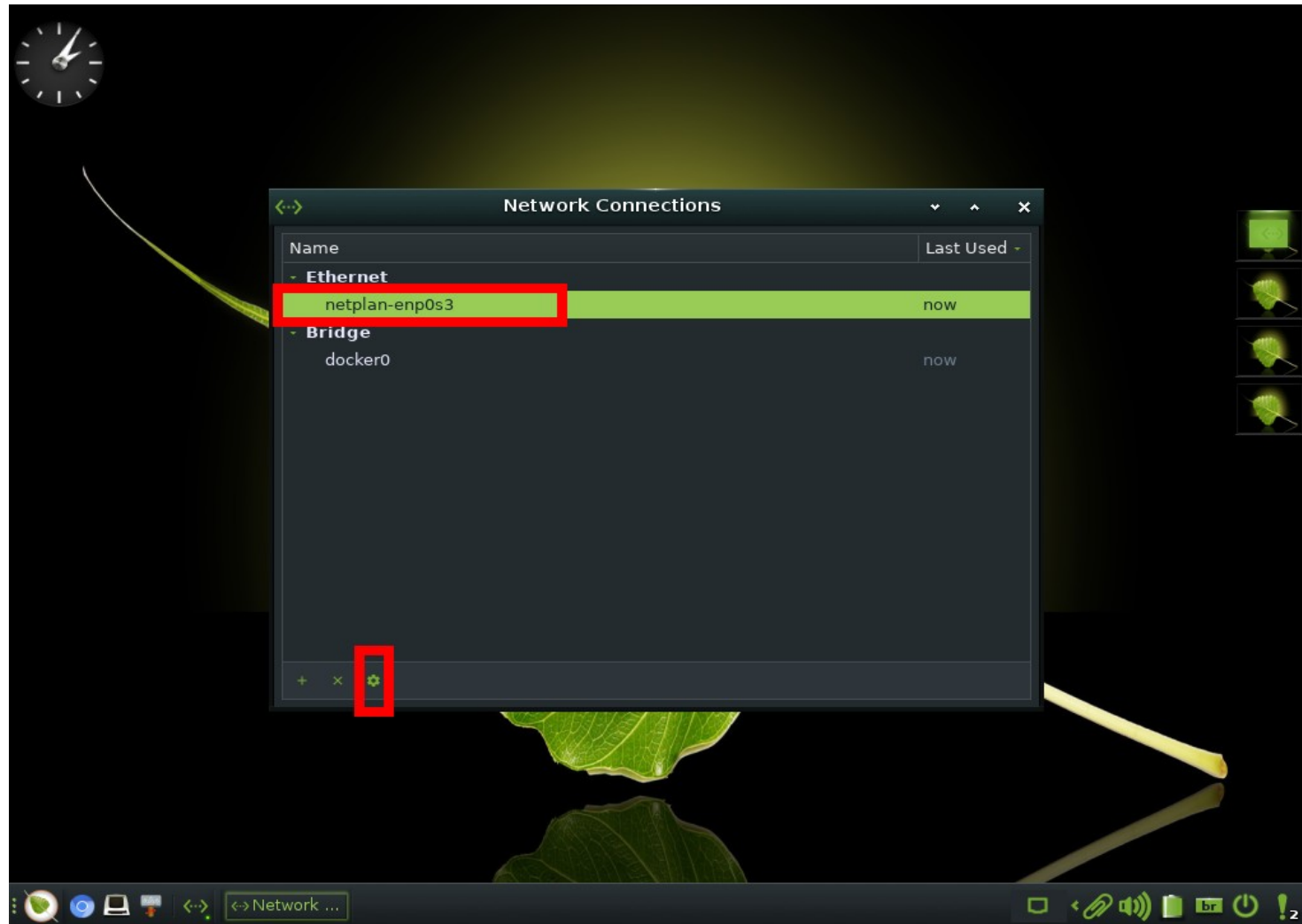
Alterando o IP no Bodhi Linux



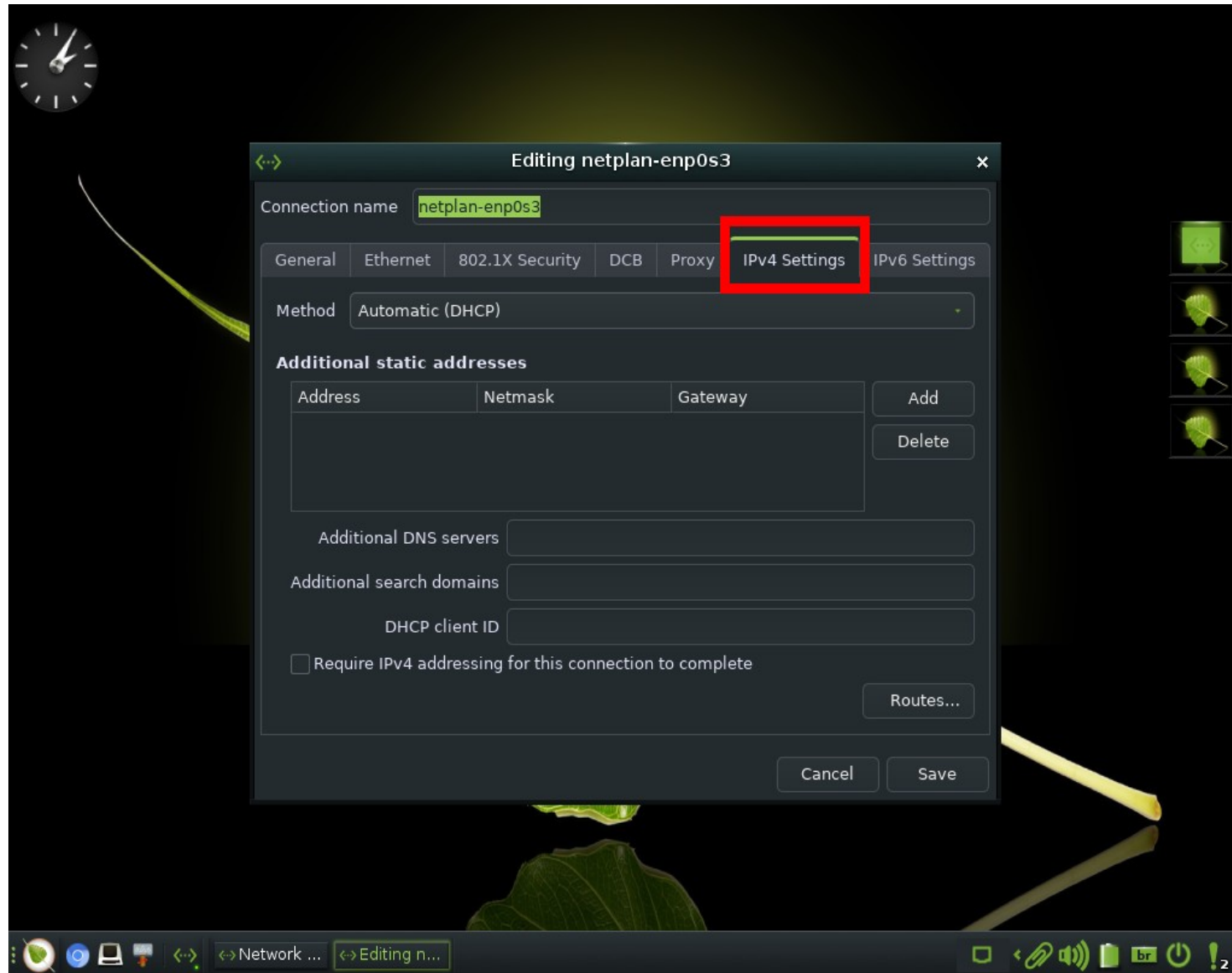
Alterando o IP no Bodhi Linux



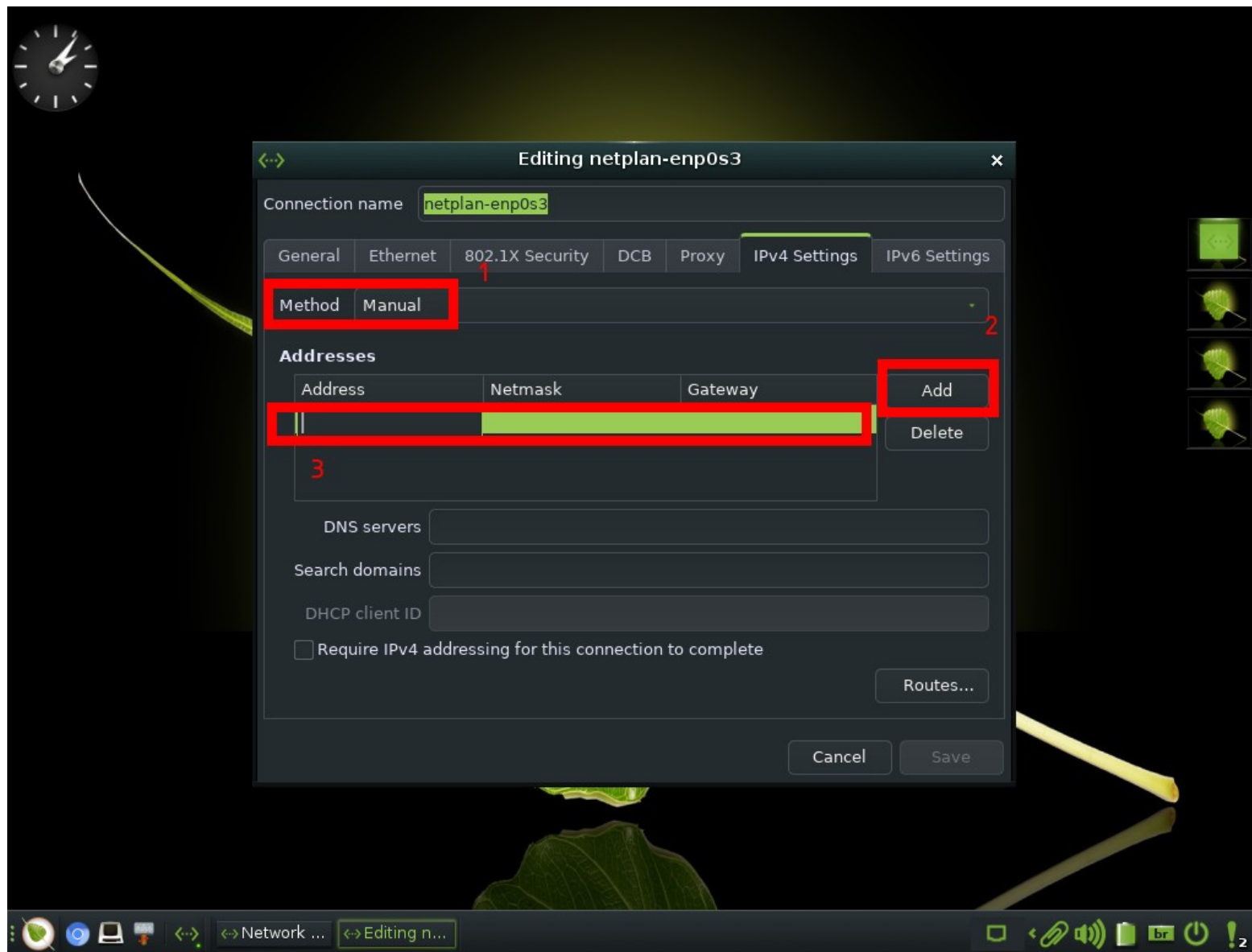
Alterando o IP no Bodhi Linux



Alterando o IP no Bodhi Linux

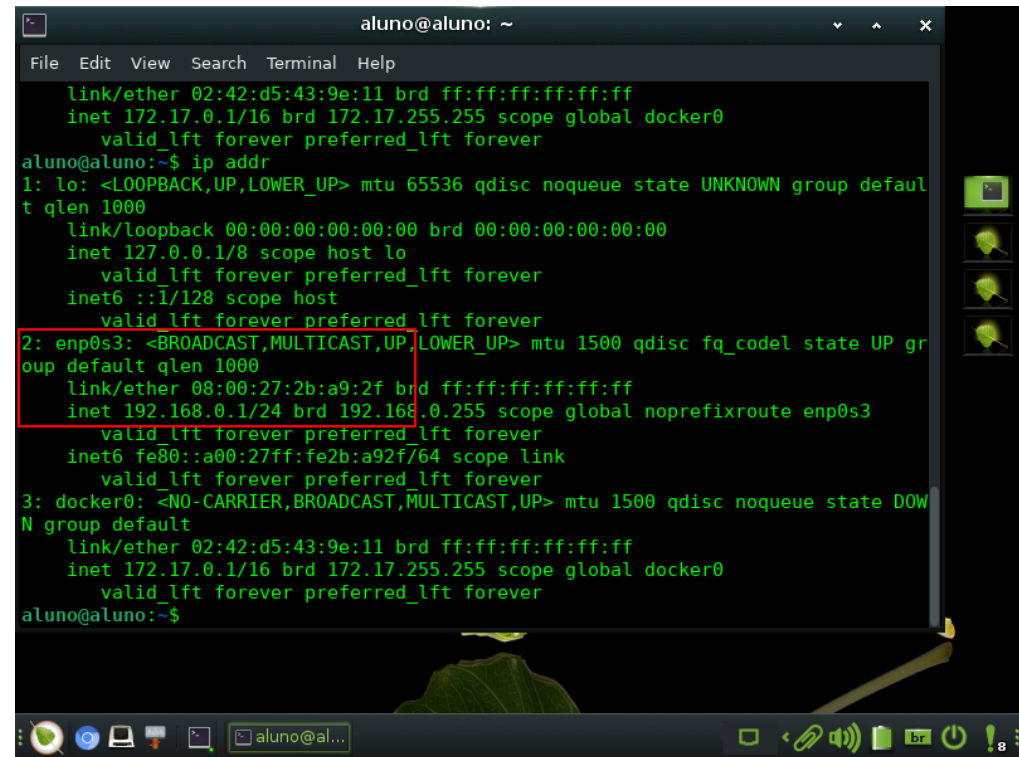


Alterando o IP no Bodhi Linux



Testando a configuração

- No terminal
 - \$ ip addr
(identifique a placa de rede correta para verificar se o IP foi atribuído)



```
aluno@aluno: ~  
File Edit View Search Terminal Help  
link/ether 02:42:d5:43:9e:11 brd ff:ff:ff:ff:ff:ff  
inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0  
valid_lft forever preferred_lft forever  
aluno@aluno:~$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000  
    link/ether 08:00:27:2b:a9:2f brd ff:ff:ff:ff:ff:ff  
    inet 192.168.0.1/24 brd 192.168.0.255 scope global noprefixroute enp0s3  
        valid_lft forever preferred_lft forever  
    inet6 fe80::a00:27ff:fe2b:a92f/64 scope link  
        valid_lft forever preferred_lft forever  
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default  
    link/ether 02:42:d5:43:9e:11 brd ff:ff:ff:ff:ff:ff  
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0  
        valid_lft forever preferred_lft forever  
aluno@aluno:~$
```


Configurando a rede via cli (*command line interface*)

```
# ip addr add IP/NETMASK dev INTERFACE
```


Exemplo:


```
# ip addr add 192.168.0.1/24 dev eth1
```

Lembrando que /24 equivale a uma máscara de rede 255.255.255.0, por exemplo

/8 = 255.0.0.0

/16 = 255.255.0.0

- 
- Reinicie o computador
 - Verifique o ip:
\$ ip addr
 - O que aconteceu com o endereço IP?

- 
- Reinicie o computador
 - Verifique o ip:
\$ ip addr
 - O que aconteceu com o endereço IP?
 - O comando ip addr atribui o IP de forma temporária (principalmente para testes)
 - Para mudar o IP de forma permanente temos um outro caminho ...

Utilizando netplan

- O netplan é a maneira padrão de muitas distribuições Linux atuais de configurar a rede
- Seus arquivos ficam, por padrão, na pasta /etc/netplan

```
$ cd /etc/netplan
```

```
$ ls
```

Utilizando netplan (2)

- Os arquivos são utilizados na ordem e um com número maior sobrescreve o de número menor
 - Ex: o arquivo 01-network-manager-all.yaml sobrescreve o arquivo 00-installer.config.yaml
- O formato utilizado é o YAML, uma espécie de alternativa ao XML. Nele, a indentação identifica que um objeto está dentro do outro
 - A indentação padrão é dada por espaço duplo

Utilizando netplan (3)

- Faça uma cópia de segurança do arquivo (sempre uma boa prática)

```
$ sudo cp 01-network-manager-all.yaml  
01.yaml.bkp
```

```
$ sudo nano 01-network-manager-all.yaml
```



Configuração manual mínima

network:

version: 2

renderer: NetworkManager

ethernets:

enp0s3:

dhcp4: no

addresses: [192.168.0.2/24]

Testando a configuração

- Testando a sintaxe
`$ sudo netplan try`
- Aplicando a configuração (se não houver erros)
`$ sudo netplan apply`
- Verificando o IP
`$ sudo ip addr`

Referências

- Modos de rede do VirtualBox
<https://www.virtualbox.org/manual/ch06.html>
- Comando IP
<https://www.certificacaolinux.com.br/comando-linux-ip/>
- Usando o netplan
<https://www.bosontreinamentos.com.br/linux/como-configurar-endereco-ip-estatico-no-linux-ubuntu-18-04-com-netplan/>