

# Laboratório https

Grupo 8

4ºads-B

## 1. Criação da ec2

Criando uma ec2 com ip público:

▼ Configurações de rede [Informações](#)

VPC - obrigatório [Informações](#)

vpc-004ea393b4af985e1 (anime-match)  
10.0.0.0/24

↻

Sub-rede [Informações](#)

subnet-002bd2a3b1bfcd0ed sprint-public-subnet  
VPC: vpc-004ea393b4af985e1 Proprietário: 617823867544  
Zona de disponibilidade: us-east-1a Endereços IP disponíveis: 122 CIDR: 10.0.0.0/25

↻

[Criar nova sub-rede](#)

Atribuir IP público automaticamente [Informações](#)

Habilitar

Taxas adicionais se aplicam quando fora do limite de nível gratuito

Firewall (grupos de segurança) [Informações](#)

Um grupo de segurança é um conjunto de regras de firewall que controlam o tráfego para sua instância. Adicione regras para permitir que o tráfego específico alcance sua instância.

☐ Criar grupo de segurança

☒ Selecionar grupo de segurança existente

Grupos de segurança comuns [Informações](#)

Selecionar grupos de segurança

↻

public-sg sg-0a2ca1b3052a84170 ✕  
VPC: vpc-004ea393b4af985e1



[Comparar regras do grupo de segurança](#)

Os grupos de segurança que você adicionar ou remover aqui serão adicionados ou removidos em todas as suas interfaces de rede.


► Configuração avançada de rede


ID da instância

 i-02b08db595a286409 (https-lab)

1. Abra um cliente SSH.
2. Localize o arquivo de chave privada. A chave usada para executar esta instância é `lucas_key.pem`
3. Execute este comando, se necessário, para garantir que sua chave não fique visível publicamente.  
 `chmod 400 "lucas_key.pem"`
4. Conecte-se à sua instância usando sua IP público:  
 `34.235.132.211`

Exemplo:

 `ssh -i "lucas_key.pem" ubuntu@34.235.132.211`

 **Observação:** na maioria dos casos, o nome de usuário suposto está correto. No entanto, leia as instruções de uso da AMI para verificar se o proprietário da AMI alterou o nome de usuário da AMI padrão.

```
PS C:\Users\lukas\OneDrive\Documentos> ssh -i "lucas_key.pem" ubuntu@34.235.132.211
The authenticity of host '34.235.132.211 (34.235.132.211)' can't be established.
ED25519 key fingerprint is SHA256:0N0cxKMR7+0Ewbz2zzh4LSGIvVKRWuxRN5rmem6mk7M.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '34.235.132.211' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-1008-aws x86_64)
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/pro
```

System information as of Sat Jun 1 19:31:41 UTC 2024

```
System load:  0.65          Processes:            109
Usage of /:    23.2% of 6.71GB Users logged in:       0
Memory usage:  21%          IPv4 address for enX0: 10.0.0.77
Swap usage:    0%
```

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.  
See <https://ubuntu.com/esm> or run: `sudo pro status`

The list of available updates is more than a week old.  
To check for new updates run: `sudo apt update`

The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in `/usr/share/doc/*/copyright`.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.

To run a command as administrator (user "root"), use "`sudo <command>`".  
See "`man sudo_root`" for details.

```
ubuntu@ip-10-0-0-77:~$ |
```

## 2. Nginx.

Nginx instalado na máquina.

```
ubuntu@ip-10-0-0-77: $ sudo systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
   Active: active (running) since Sat 2024-06-01 19:34:54 UTC; 16s ago
     Docs: man:nginx(8)
  Process: 7516 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Process: 7518 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
 Main PID: 7519 (nginx)
    Tasks: 2 (limit: 1130)
   Memory: 1.7M (peak: 1.9M)
      CPU: 12ms
   CGroup: /system.slice/nginx.service
           └─7519 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─7520 "nginx: worker process"

Jun 01 19:34:54 ip-10-0-0-77 systemd[1]: Starting nginx.service - A high performance web server and a reverse proxy server...
Jun 01 19:34:54 ip-10-0-0-77 systemd[1]: Started nginx.service - A high performance web server and a reverse proxy server.
ubuntu@ip-10-0-0-77: $
```

### 3. Domínio.

Criando uma conta no no-ip.

## Crie Sua Conta No-IP

Email

lucas.lrodrigues@sptech.school

Senha

[Show password](#)

\*\*\*\*\*

Password strength: Strongest.

Termos do Serviço e Política de Privacidade

☒ eu concordo com os [Termos de Serviço](#) e a [Política de Privacidade](#).  
Também concordo que somente criarei uma conta gratuita.

Inscrição nas comunicações por e-mail

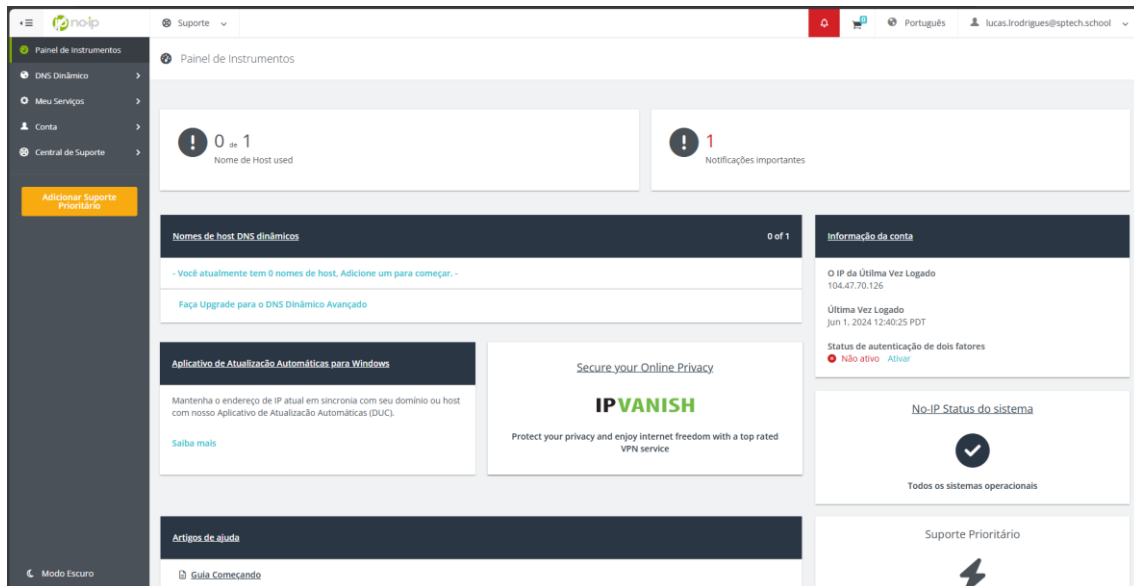
☒ Desbloqueie ofertas exclusivas e insights de ponta. Junte-se ao No-IP Inner Circle agora!

Free Sign Up



Cadastre-se com o Google

©2024 • No-IP.com • Todos os Direitos Reservados. [Política de Privacidade](#) & [Termos de Serviço](#)



Adicionando um novo hostname.

## + Create a Hostname

### Nome de Host ⓘ

The name may not be greater than 19 characters.

### Domínio ⓘ

### Tipo de Registro

- ☒ DNS Host (A) ⓘ
- ☐ AAAA (IPv6) ⓘ
- ☐ DNS Alias (CNAME) ⓘ
- ☐ Web Redirect ⓘ

[Gerencie](#) seu registro de Round Robin, TXT, SRV e DKIM

### IPv4 Endereço ⓘ

### Wildcard ⓘ

[Compre Enhanced](#)

para ativar a opção Wildcard

### MX Registros

[+ Adicionar Registros MX](#)

Cancelar

Create Hostname with DDNS Key

Criar hostname

Criar hostname					Pesquisar...	
Nome de Host	Last Update	IP / Alvo	Type	DDNS Key		
lab-https.ddns.net	Jun 1, 2024 12:47 PDT	54.159.209.54	A	<a href="#">Create DDNS Key</a>	<a href="#">Modificar</a>	x

#### 4. Certbot

Instalando o certbot.

```
ubuntu@ip-10-0-0-77:~$ sudo apt install certbot python3-certbot-nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-acme python3-certbot python3-configargparse python3-icu python3-josepy python3-parsedatetime python3-rfc3339
Suggested packages:
  python-certbot-doc python-certbot-apache python-acme-doc python-certbot-nginx-doc
The following NEW packages will be installed:
  certbot python3-acme python3-certbot python3-certbot-nginx python3-configargparse python3-icu python3-josepy
  python3-parsedatetime python3-rfc3339
0 upgraded, 9 newly installed, 0 to remove and 0 not upgraded.
Need to get 1097 kB of archives.
After this operation, 5699 kB of additional disk space will be used.
```

#### 5. De volta ao nginx.

Adicionando o server name ao arquivo “/etc/nginx/sites-available/default”.

```
server {
    listen 80 default_server;
    listen [::]:80 default_server;

    root /var/www/html;

    index index.html index.htm index.nginx-debian.html;

    server_name lab-https.ddns.net;

    location / {
        # First attempt to serve request as file, then
        # as directory, then fall back to displaying a 404.
        try_files $uri $uri/ =404;
    }

    # pass PHP scripts to FastCGI server
    #
    #location ~ \.php$ {
    #    include snippets/fastcgi-php.conf;
    #
    #    # With php-fpm (or other unix sockets):
    #    fastcgi_pass unix:/run/php/php7.4-fpm.sock;
    #    # With php-cgi (or other tcp sockets):
```

Carregando o nginx.

```
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ sudo systemctl reload nginx
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ |
```

```
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ |
```

## 6. De volta ao Certbot.

Criando o certificado https.

```
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ sudo certbot --nginx -d lab-https.ddns.net
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Enter email address (used for urgent renewal and security notices)
(Enter 'c' to cancel): lucas.lrodrigues@sptech.school

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.4-April-3-2024.pdf. You must agree in
order to register with the ACME server. Do you agree?
-----
(Y)es/(N)o: y

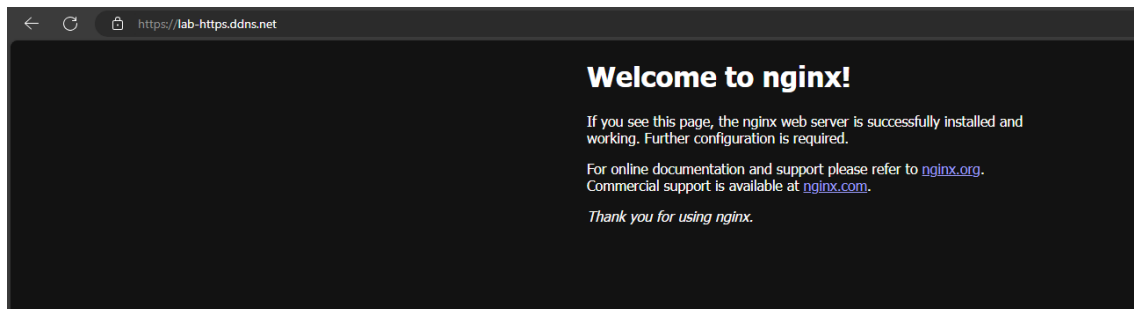
-----
Would you be willing, once your first certificate is successfully issued, to
share your email address with the Electronic Frontier Foundation, a founding
partner of the Let's Encrypt project and the non-profit organization that
develops Certbot? We'd like to send you email about our work encrypting the web,
EFF news, campaigns, and ways to support digital freedom.
-----
(Y)es/(N)o: n
Account registered.
Requesting a certificate for lab-https.ddns.net

Successfully received certificate.
Certificate is saved at: /etc/letsencrypt/live/lab-https.ddns.net/fullchain.pem
Key is saved at: /etc/letsencrypt/live/lab-https.ddns.net/privkey.pem
This certificate expires on 2024-08-30.
These files will be updated when the certificate renews.
Certbot has set up a scheduled task to automatically renew this certificate in the background.

Deploying certificate
Successfully deployed certificate for lab-https.ddns.net to /etc/nginx/sites-enabled/default
Congratulations! You have successfully enabled HTTPS on https://lab-https.ddns.net

-----
If you like Certbot, please consider supporting our work by:
* Donating to ISRG / Let's Encrypt: https://letsencrypt.org/donate
* Donating to EFF: https://eff.org/donate-le
-----
ubuntu@ip-10-0-0-77:/etc/nginx/sites-available$ |
```

## 7. Teste de acesso.



## 8. Configuração NGINX.

Configuração criada pelo Certbot no arquivo default.

```
listen [::]:443 ssl ipv6only=on; # managed by Certbot
listen 443 ssl; # managed by Certbot
ssl_certificate /etc/letsencrypt/live/lab-https.ddns.net/fullchain.pem; # managed by Certbot
ssl_certificate_key /etc/letsencrypt/live/lab-https.ddns.net/privkey.pem; # managed by Certbot
include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

}
```