



Relatório do Trabalho de Instalação I

Gabriel Luis de Assis Souza - 202110329
Higor Batista Fernandes - 202020469
Matheus Felipe Godoi Coutinho - 202110840
Robson Ferreira dos Santos Junior - 202120530

Grupo B - Turma 14 A

Lavras – MG

Junho de 2024

Resumo

Este relatório tem como objetivo descrever como foi realizada a instalação e a configuração de serviços de redes, sendo que foram disponibilizadas duas máquinas virtuais (VMs) para a instalação e a configuração de servidores Linux Debian. Nesse sentido, será apresentada nesta primeira parte os seguintes serviços de redes:

- Serviço de sincronização de hora
- Servidor Web

Como foi descrito anteriormente, foram disponibilizadas duas máquinas virtuais (VMs) para a realização desse trabalho. Desse modo, tais máquinas virtuais utilizadas serão identificadas abaixo:

- 192.168.1.3 (aluno@vm03)
- 192.168.1.4 (aluno@vm04)

Etapa 1

• Serviço de sincronização de hora

No primeiro momento, fizemos algumas pesquisas em alguns sites sobre os serviços nativos pelo Debian que foram sugeridos pelo roteiro. Nesse sentido, fizemos pesquisas a respeito do funcionamento dos serviços: `timcdatctl`, `timcsQ.cd` e `cmio.Q`.

A partir das pesquisas realizadas, discutimos qual seria a melhor alternativa para a sincronização de hora, sendo que uma das máquinas virtuais (VMs) deveria receber a hora certa de um dos servidores do projeto NTP.br e então servir essa hora certa à outra máquina virtual (VM). Desse modo, após a discussão, concluímos que o melhor serviço para a proposta dada seria o `cmio.Q`, uma vez que, este possui diversos benefícios que atendem a proposta, tais como: Sincronização com Múltiplas Fontes, Suporte para Servidor NTP, e além de apresentar alta precisão e confiabilidade para serviços de sincronização de hora.

Desse modo, implementamos tal serviço na 192.168.1.3 (aluno@vm03), e esta serviu a hora certa para a 192.168.1.4 (aluno@vm04). Os detalhes desta implementação serão descritos abaixo:

- Inicialmente, foi necessário realizar a instalação do `cmio.Q` na máquina 192.168.1.3 (aluno@vm03) com os seguintes comandos:

sudo apt install chrony

- Após isso, foi acessado a pasta onde estava o arquivo de configuração do cmio-Q com o seguinte comando:

cd /etc/chrony

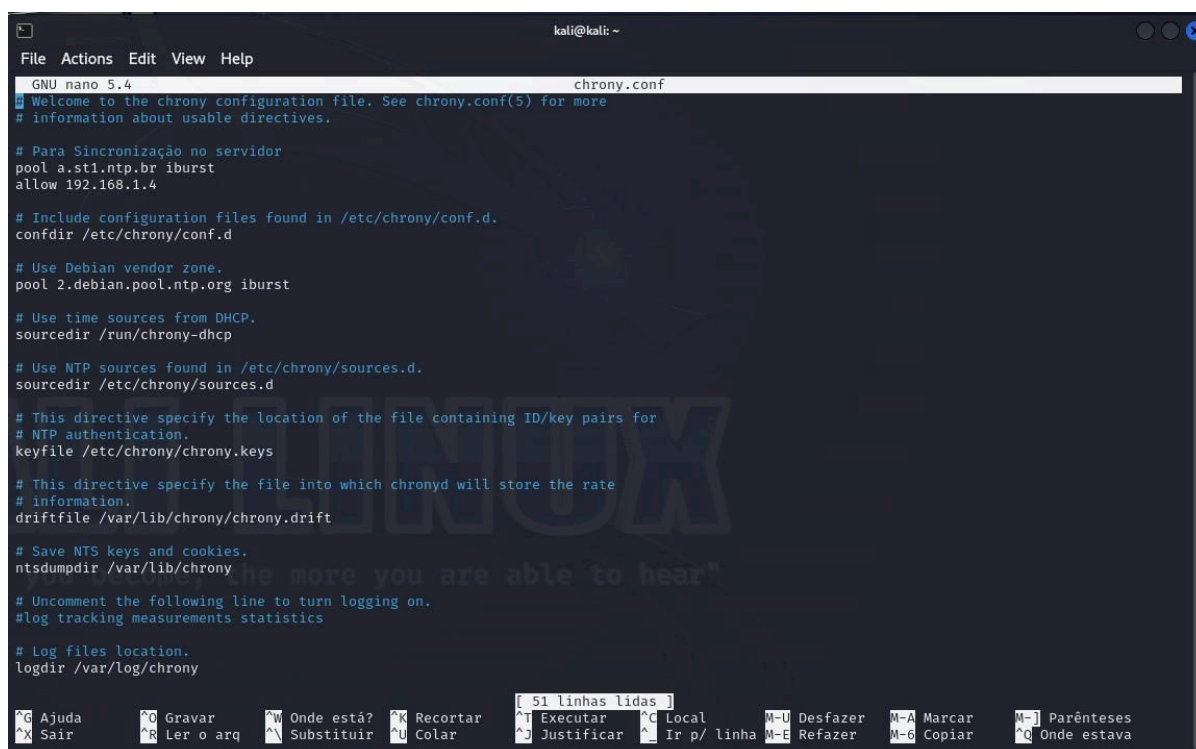
- Em seguida acessamos o arquivo para editá-lo, com o seguinte comando: **sudo nano chrony.conf**

- Para buscar a sincronização no servidor NTP.br, dentro do arquivo foi adicionado:

pool a.st1.ntp.br iburst

- Além disso, foi adicionado o IP da máquina virtual (VMs), 192.168.1.4 (aluno@vm04), para ser cliente e receber a sincronização, a partir do seguinte comando:

allow 192.168.1.4



```

GNU nano 5.4                                chrony.conf
Welcome to the chrony configuration file. See chrony.conf(5) for more
# information about usable directives.

# Para Sincronização no servidor
pool a.st1.ntp.br iburst
allow 192.168.1.4

# Include configuration files found in /etc/chrony/conf.d.
confdir /etc/chrony/conf.d

# Use Debian vendor zone.
pool 2.debian.pool.ntp.org iburst

# Use time sources from DHCP.
sourcedir /run/chrony-dhcp

# Use NTP sources found in /etc/chrony/sources.d.
sourcedir /etc/chrony/sources.d

# This directive specify the location of the file containing ID/key pairs for
# NTP authentication.
keyfile /etc/chrony/chrony.keys

# This directive specify the file into which chronyd will store the rate
# information.
driftfile /var/lib/chrony/chrony.drift

# Save NTS keys and cookies.
ntsdumpdir /var/lib/chrony

# Uncomment the following line to turn logging on.
#log tracking measurements statistics

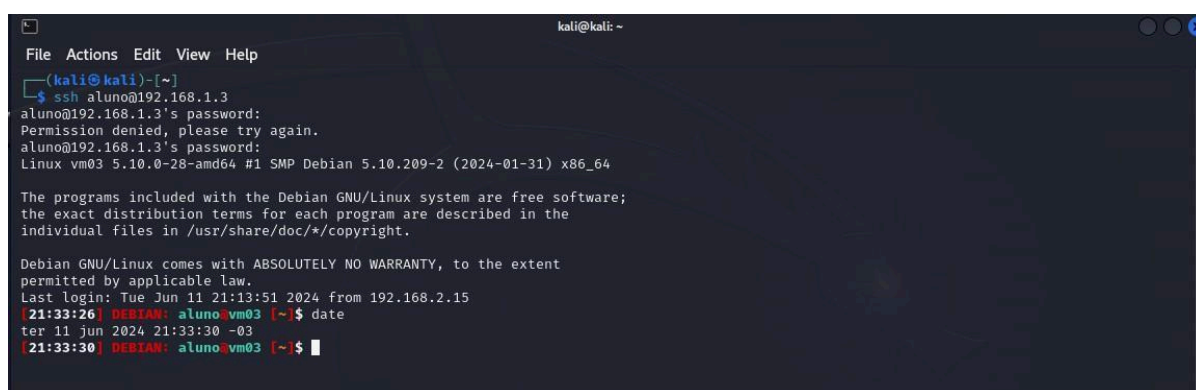
# Log files location.
logdir /var/log/chrony

```

Figura A: Arquivo de configuração do cmio-Q, a qual fica evidente a edição do arquivo para sincronização de hora das máquinas virtuais (VMs)

- Por fim, foi necessário reiniciar o cmio-Q com o seguinte comando: `sudo systemctl restart chrony.service`
- Dessa forma, fizemos a verificação da data e da hora na 192.168.1.3 (aluno@vm03) e na 192.168.1.4 (aluno@vm04), a qual já apareciam corretamente utilizando o comando:

`date`



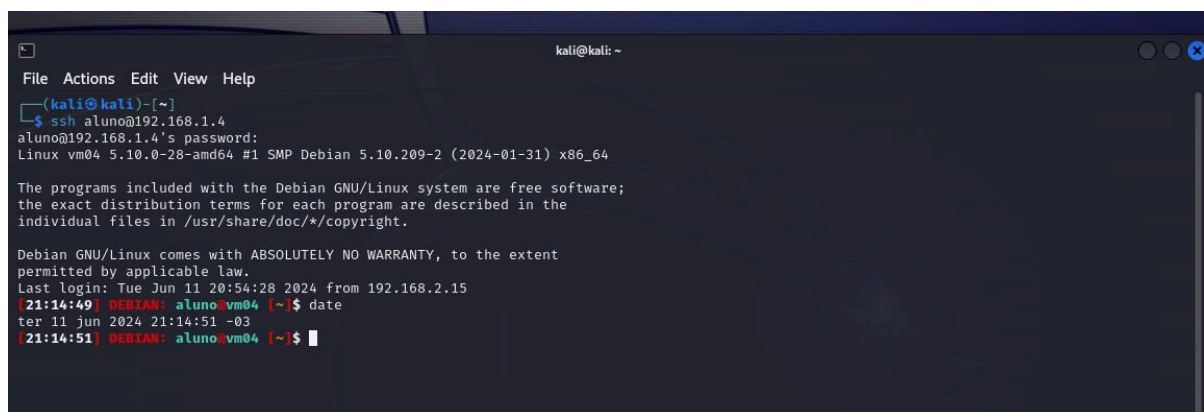
```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ssh aluno@192.168.1.3
aluno@192.168.1.3's password:
Permission denied, please try again.
aluno@192.168.1.3's password:
Linux vm03 5.10.0-28-amd64 #1 SMP Debian 5.10.209-2 (2024-01-31) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jun 11 21:13:51 2024 from 192.168.2.15
21:33:26 DEBIAN: aluno@vm03 [~]$ date
ter 11 jun 2024 21:33:30 -03
21:33:30 DEBIAN: aluno@vm03 [~]$
  
```

Figura B: Verificação da data e hora da 192.168.1.3 (aluno@vm03), sendo que esta deveria receber a hora certa de um dos servidores do projeto NTP.br



```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ ssh aluno@192.168.1.4
aluno@192.168.1.4's password:
Linux vm04 5.10.0-28-amd64 #1 SMP Debian 5.10.209-2 (2024-01-31) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Jun 11 20:54:28 2024 from 192.168.2.15
21:14:49 DEBIAN: aluno@vm04 [~]$ date
ter 11 jun 2024 21:14:51 -03
21:14:51 DEBIAN: aluno@vm04 [~]$
  
```

Figura C: Verificação da data e hora da 192.168.1.4 (aluno@vm04), sendo que esta recebe a data e hora certa a partir da 192.168.1.3 (aluno@vm03)

Etapa 2

• Servidor Web

Inicialmente, fizemos algumas pesquisas a respeito de como fazer a instalação do servidor Web em uma das máquinas virtuais (VMs). Desse modo,

precisaríamos colocar um servidor Web que continha uma página (html, php, etc.) com os itens: nomes dos integrantes do grupo; relatório de atividades desenvolvidas para a conclusão do presente trabalho.

Além disso, o servidor deverá funcionar ainda como um repositório de trabalhos executados na disciplina e ser capaz de atender aos usuários por meio dos protocolos HTTP e HTTPS. Assim sendo, após nossas pesquisas encontramos uma maneira de realizar tal proposta de modo que todos os itens acima fossem atendidos.

Portanto, implementamos tal serviço na 192.168.1.4 (aluno@vm04) e os detalhes desta implementação serão descritos abaixo:

- De início foi utilizado o seguinte comando para sincronizar os arquivos de índice e pacotes da fonte:

`sudo apt update`

- Logo em seguida foi usado o seguinte comando para instalar o apache na máquina:

`sudo apt install apache2`

- Após isso, usamos o seguinte comando para inicializar o servidor Web:

`sudo service apache2 start`

- Para checar o status e se foi inicializado corretamente usamos o comando: `sudo service`

`apache2 status`

- Em seguida foi necessário realizar a alteração da página web com o seguinte comando, vale destacar que para acessar o arquivo "index.html" e conseguir editá-lo foi usado o sudo para acessar com as permissões de administrador

`sudo nano index.html`



```
GNU nano 5.4 index.html
<!DOCTYPE html>
<html>
  <head lang="pt">
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
  </head>
  <body>
    <h1>Trabalhos de Redes - UFLA</h1>
    <p>Matheus Felipe Godoi Coutinho - 202110840 - 14A </p>
    <p>Gabriel Luis de Assis Souza - 202110329 - 14A </p>
    <p>Higor Batista Fernandes - 202020469 - 14A </p>
    <p>Robson Ferreira dos Santos Junior - 202120530 - 14A </p>
  </body>
</html>
```

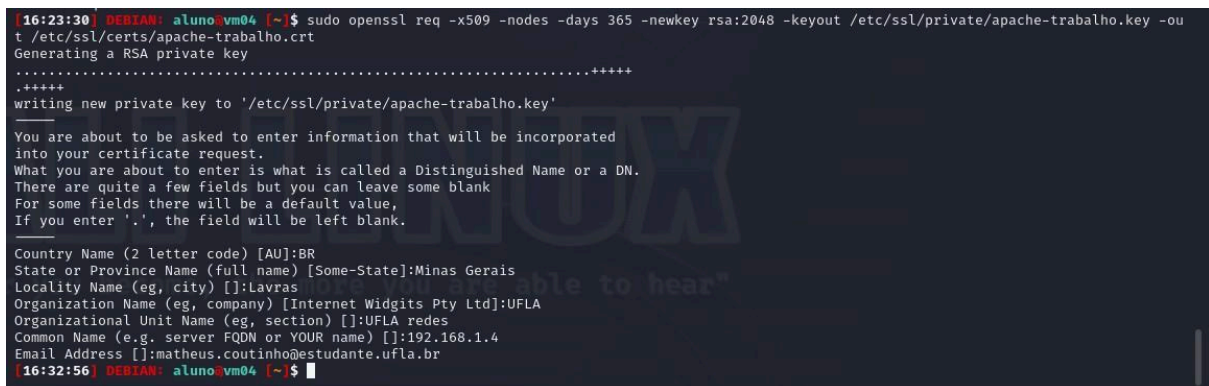
Figura D: Alteração do arquivo index.html da 192.168.1.4 (aluno@vm04) com os nomes dos integrantes do grupo

- Em seguida, foi realizado a instalação do servidor HTTPs e HTTP, a partir do seguinte comando:

apt-get install apache2 openssl build-essential -y

- Após isso, devemos criar uma chave de certificado:

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-trabalho.key -out /etc/ssl/certs/apache-trabalho.crt



```
[16:23:30] DEBIAN: aluno@vm04 [~]$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-trabalho.key -out /etc/ssl/certs/apache-trabalho.crt
Generating a RSA private key
.....+++++
writing new private key to '/etc/ssl/private/apache-trabalho.key'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:BR
State or Province Name (full name) [Some-State]:Minas Gerais
Locality Name (eg, city) []:Lavras
Organization Name (eg, company) [Internet Widgits Pty Ltd]:UFLA
Organizational Unit Name (eg, section) []:UFLA redes
Common Name (e.g. server FQDN or YOUR name) []:192.168.1.4
Email Address []:matheus.coutinho@estudante.ufla.br
[16:32:56] DEBIAN: aluno@vm04 [~]$
```

Figura E: Criação da chave de certificado para o servidor HTTPs e HTTP

- Depois rodamos estes dois comandos a seguir, a fim de habilitar as configurações no apache e no site recém-criado:

sudo a2enmod ssl
sudo a2ensite default-ssl.conf

```
16:16:34] DEBIAN: aluno@vm04 [~]$ sudo a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
```

Figura F: Habilitação das configurações do apache para o site criado

- Após isso, reiniciamos o serviço para aplicar as mudanças destacadas a partir do seguinte comando:

systemctl reload apache2

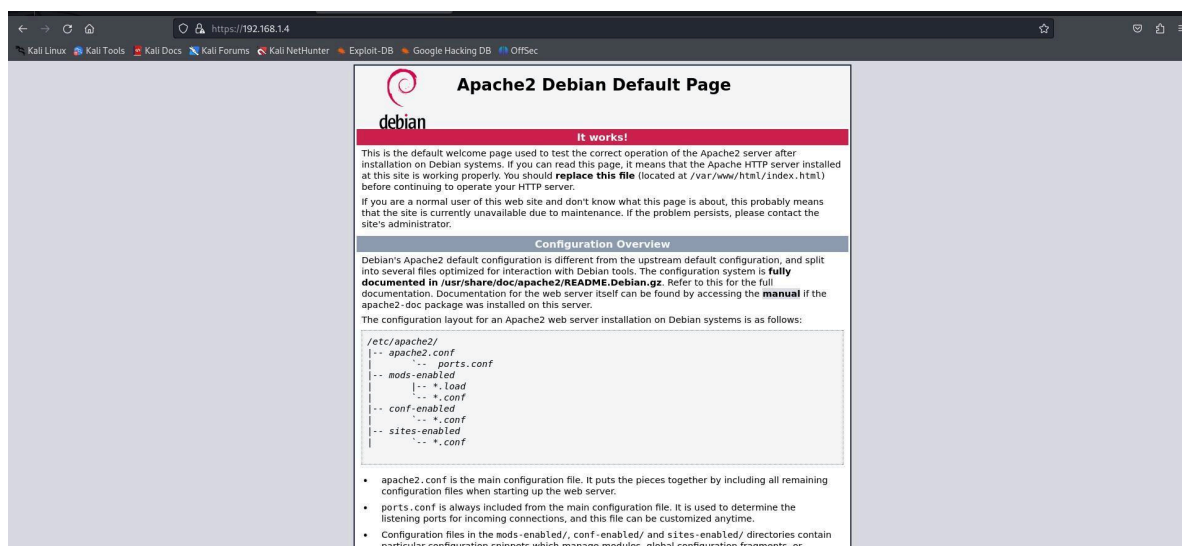
- A partir disso, verificamos então as mudanças destacadas a partir do seguinte comando:

sudo service apache2 status

```
16:41:30] DEBIAN: aluno@vm04 [~]$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2024-04-29 16:41:30 -03; 10s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 11458 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 11462 (apache2)
      Tasks: 55 (limit: 1115)
     Memory: 9.3M
        CPU: 55ms
    CGroup: /system.slice/apache2.service
            └─11462 /usr/sbin/apache2 -k start
              11463 /usr/sbin/apache2 -k start
              11464 /usr/sbin/apache2 -k start

abr 29 16:41:30 vm04 systemd[1]: Starting The Apache HTTP Server ...
abr 29 16:41:30 vm04 apachectl[11461]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.0.1
abr 29 16:41:30 vm04 systemd[1]: Started The Apache HTTP Server.
```

Figura G: Verificação dos status do The Apache HTTP Server



Apache2 Debian Default Page

debian

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Debian systems. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Debian's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Debian tools. The configuration system is **fully documented in `/usr/share/doc/apache2/README.Debian.gz`**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Debian systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|   |-- ports.conf
|-- mods-enabled
|   |-- *.load
|   |-- *.conf
|-- conf-enabled
|   |-- *.conf
|-- sites-enabled
|   |-- *.conf
```

- `apache2.conf` is the main configuration file. It puts the pieces together by including all remaining configuration files when starting up the web server.
- `ports.conf` is always included from the main configuration file. It is used to determine the listening ports for incoming connections, and this file can be customized anytime.
- Configuration files in the `mods-enabled/`, `conf-enabled/` and `sites-enabled/` directories contain particular configuration snippets which manage modules, global configuration fragments, or other site-specific configuration.

Figura H: Teste da página Default do Apache para o site Web

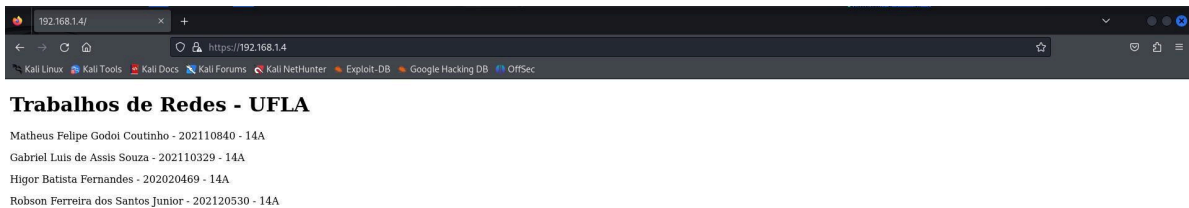


Figura I: Site funcional com os nomes dos integrantes do Grupo B

- Após a criação do site funcional com os nomes dos integrantes do Grupo B, foi criado um repositório no servidor Web com a utilização do Git. Nesse sentido, tornou-se possível acessar todos os documentos criados para disciplina através do site.
- Link do repositório criado no github: [matheusfgcz/Trabalhos-de-Redes-Trabalhos da disciplina Redes de Computadores da UFLA \(github.com\)](https://github.com/matheusfgcz/Trabalhos-de-Redes-Trabalhos-da-disciplina-Redes-de-Computadores-da-UFLA)

```
File Actions Edit View Help
GNU nano 5.4
<!DOCTYPE html>
<html>
  <head lang="pt">
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
  </head>
  <body>
    <h1>Trabalhos de Redes - UFLA</h1>
    <p>Matheus Felipe Godoi Coutinho - 202110840 - 14A </p>
    <p>Gabriel Luis de Assis Souza - 202110329 - 14A </p>
    <p>Higor Batista Fernandes - 202020469 - 14A </p>
    <p>Robson Ferreira dos Santos Junior - 202120530 - 14A </p>
    <p><a href="TrabalhoDeRedesSlides.pdf" target="_blank">Slides do trabalho</a></p>
    <p><a href="trabalho-cisco.pkt" target="_blank">Trabalho packet tracer 1</a></p>
    <p><a href="" target="_blank">Relatorio do trabalho Etapa 1</a></p>
  </body>
</html>
```

Figura J: Arquivo html alterado para possibilitar o download dos trabalhos da disciplina

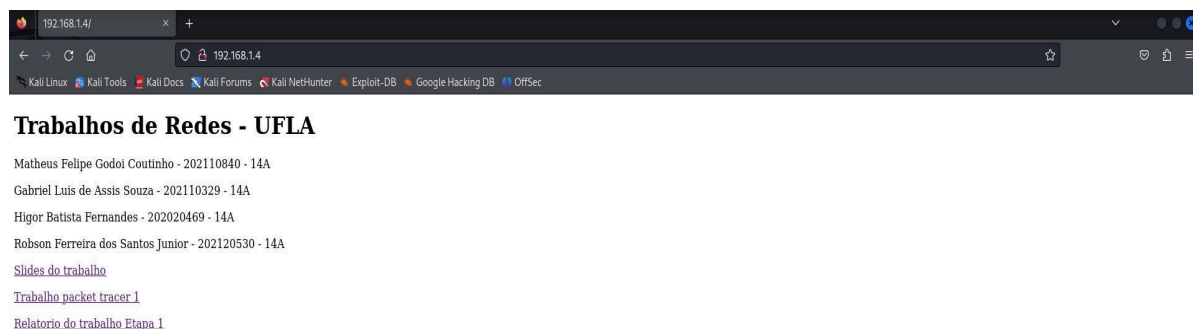


Figura K: Site funcional com os nomes dos integrantes do Grupo B e os links dos trabalhos da disciplina



Figura L: Slides da apresentação abertos a partir do site criado

- **Obs:** Fomos orientados a fazer uma modificação opcional pelo monitor da Disciplina (Daniel) a fazer o redirecionamento automático do acesso do HTTP ao HTTPS, tendo em vista que o HTTPS possui criptografia, tornando-o uma alternativa mais segura.

Referências Bibliográficas

- **1 Etapa)**

- [O que é o Chrony, sua importância e como instalar - DevOps Mind](#)
- [chrony – Introduction \(chrony-project.org\)](#)

- **2 Etapa)**

- [Como Instalar o Servidor Web Apache no Ubuntu 20.04 | DigitalOcean](#)
- [4 maneiras para enviar projetos para um servidor remoto \(youtube.com\)](#)
- [Tutorial - Enable HTTPS on Apache \[Step by step \] \(techexpert.tips\)](#)