

Módulo 1 - Aulas 1 e 2

Tarefas

No final deste módulo você deve submeter em um ÚNICO arquivo PDF os seguintes prints:

- Atividade 1.1: passo 9.
- Atividade 1.2: passo 9.
- Atividade 1.3: passo 4.
- Atividade 1.4: passo 9.
- Atividade 1.5: passo 8.

Regras para a elaboração do documento:

1. Antes de cada Print, adicione obrigatoriamente uma frase explicativa que sinalize do que se trata o print. A inserção de prints sem a devida frase explicativa será considerada como tentativa de atrapalhar a correção do instrutor e será penalizada a critério do instrutor. Exemplo:

Print da atividade 1.1: [Imagem com o Print]

2. Os Prints devem ser tirados da TELA CHEIA. Quando tirados da VM da AWS, devem ser capturados **obrigatoriamente** da tela cheia clicando no botão "Screenshot" → "Take screenshot" da barra de ferramentas do Hypervisor da AWS.
3. Insira **somente a quantidade de Prints solicitados por atividade** usando exclusivamente 1 página por print. **A página do documento onde você vai inserir o Print deve estar com a orientação no modo PAISAGEM** para termos melhor aproveitamento do espaço. Ou seja, seu documento deverá ter a mesma quantidade de páginas que a quantidade do total de Prints! A inserção de prints desnecessários será considerada como tentativa de atrapalhar a correção do instrutor e será penalizada com nota 0.

4. Apresente Prints legíveis e com tamanho correto para fácil leitura. O envio de prints com letras minúsculas poderá ser considerado como tentativa de atrapalhar a correção do instrutor e será penalizada a critério do instrutor.

Atividade 1.1 – Comparando integridade de arquivos de texto no Kali Linux

Nesta atividade, vamos explorar alguns comandos do Kali Linux para comparar arquivos de texto, que previamente foram copiados, e descobrir se houve manipulação mediante o cálculo da diferença entre eles.

Vamos começar inicializando nosso Kali Linux via RDP ao IP Kali: 192.168.98.40, com usuário “aluno” e senha “rnipesr”.

1. Clique no botão superior esquerdo “Aplicativos” → “Aplicações habituais” → “Acessórios” → “Editor de Texto” para abrir o Mousepad e insira o seguinte texto:

```
 Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam eget  
 ligula eu lectus lobortis condimentum. Aliquam nonummy auctor massa.  
 Pellentesque habitant morbi tristique senectus et netus et malesuada  
 famas ac turpis egestas. Nulla at risus. Quisque purus magna, auctor  
 et, sagittis ac, posuere eu, lectus. Nam mattis, felis ut adipiscing.
```

2. Salve o texto na pasta “Arquivo → Salvar como → Documentos” com o nome “Texto.txt”. Feche o Mousepad.
3. Abra o Terminal Emulator (presente no meu inferior da Área de Trabalho) e execute o seguinte comando (com senha “rnipesr”) para ser super usuário:

```
 └─(aluno㉿kali)-[~]  
└─$ sudo -i  
[sudo] senha para aluno:
```

4. Navegue até a pasta “Documentos” e veja que o arquivo “Texto.txt” está presente com o seguinte comando:

```
└──(root㉿kali)-[~]
└─# cd /home/aluno/Documentos/

└──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
Texto.txt
```

5. Copie o arquivo "Texto.txt" no arquivo "Texto_copia.txt" na mesma pasta e veja que realmente foi copiado:

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# cp Texto.txt Texto_copia.txt

└──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
Texto_copia.txt  Texto.txt
```

6. Verifique se há diferença entre os arquivos:

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# diff Texto.txt Texto_copia.txt
```

Veja que o comando mostrado acima não tem retorno, ou seja, não há diferença entre os arquivos.

7. Abra o arquivo "Texto_copia.txt":

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# nano Texto_copia.txt
```

8. Modifique a primeira letra "L" por "P":

Porem ipsum dolor sit amet, consectetur adipiscing elit. Etiam eget ligula eu lectus lobortis condimentum. Aliquam nonummy auctor massa. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Nulla at risus. Quisque purus magna, auctor et, sagittis ac, posuere eu, lectus. Nam mattis, felis ut adipiscing.

Salve o arquivo modificado com "Ctrl+X", "s" e "Enter"

9. Agora, verificaremos a diferença entre os arquivos (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
└─(root㉿kali)-[/home/aluno/Documentos]
  └─# diff Texto.txt Texto_copia.txt
    1c1
    < Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam eget
      ligula eu lectus lobortis condimentum. Aliquam nonummy auctor massa.
      Pellentesque habitant morbi tristique senectus et netus et malesuada
      fames ac turpis egestas. Nulla at risus. Quisque purus magna, auctor
      et, sagittis ac, posuere eu, lectus. Nam mattis, felis ut adipiscing.
      \ Nenhum caractere de nova linha no final do arquivo
    ---
    > Porem ipsum dolor sit amet, consectetur adipiscing elit. Etiam eget
      ligula eu lectus lobortis condimentum. Aliquam nonummy auctor massa.
      Pellentesque habitant morbi tristique senectus et netus et malesuada
      fames ac turpis egestas. Nulla at risus. Quisque purus magna, auctor
      et, sagittis ac, posuere eu, lectus. Nam mattis, felis ut adipiscing.
```

A saída do comando mostrado acima é:

- **1c1**: Indica que a diferença começa na linha 1 do arquivo original (Texto.txt).
- <: Indica a linha no arquivo original (Texto.txt). Neste caso, a linha original começa com "Lorem ipsum dolor sit amet..."
- ---: Divide a linha original da linha modificada.
- >: Indica a linha no arquivo modificado (Texto_copia.txt). Neste caso, a linha modificada começa com "Porem ipsum dolor sit amet..."
- Portanto, a diferença encontrada está na primeira linha do arquivo, onde a palavra "Lorem" foi substituída por "Porem".

10. Não apague os arquivos usados. Feche o Mousepad e o Terminal.

Parabéns! Agora, você sabe como verificar se um arquivo de texto copiado foi modificado ou não!

Atividade 1.2 – Comparando integridade de arquivos genéricos no Kali Linux

Nesta atividade, vamos explorar alguns comandos do Kali Linux para comparar arquivos genéricos (Word, PDF, imagens, etc), que previamente foram copiados, e descobrir se houve manipulação mediante o cálculo da diferença entre eles.

Vamos começar inicializando nosso Kali Linux via RDP ao IP Kali: 192.168.98.40, com usuário "aluno" e senha "rnpesr".

1. Abra o Terminal e insira o seguinte comando para abrir o programa "Kolourpaint" com o usuário "aluno":

```
└──(aluno㉿kali)-[~]
└─$ kolourpaint
```

2. No programa "kolourpaint", selecione algumas das cores e fique à vontade para pintar no quadro branco com a Caneta usando diferentes cores.
3. Ao acabar de pintar, clique em "Arquivo" → "Salvar como" com o nome "Imagen1" na pasta "Documentos" e clique em "Salvar".
4. Repita o passo anterior salvando com o nome "Imagen1_copia".
5. Ainda no programa "kolourpaint", clique em alguma das cores e pinte somente um ponto a mais na figura criada no passo 4.
6. Clique em "Arquivo" → "Salvar como" com o nome "Imagen2" e clique em "Salvar". Feche o programa "kolourpaint".
7. Volte ao Terminal e verifique que os arquivos de imagem estão presentes:

```
└──(aluno㉿kali)-[~]
└─$ cd /home/aluno/Documentos

└──(aluno㉿kali)-[~/Documentos]
└─$ ls
Imagen1_copia.png  Imagen1.png  Imagen2.png  Texto_copia.txt  Texto.txt
```

8. Agora, vamos comparar os arquivos (que são iguais) "Imagen1.png" e "Imagen1_copia.png":

```
└──(root㉿kali)-[ ~/Documentos]
└─# cmp Imagen1.png Imagen1_copia.png
```

Veja que o comando mostrado acima não tem retorno, ou seja, não há diferença entre os arquivos.

9. Agora, vamos comparar os arquivos (que são diferentes) "Imagen1.png" e "Imagen2.png" (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
└──(aluno㉿kali)-[~/Documentos]
  └─$ cmp Imagem1.png Imagem2.png
    Imagem1.png e Imagem2.png são diferentes: byte 58, linha 3
```

O comando mostrado acima conta com os seguintes parâmetros (no seu caso pode haver diferença na quantidade de bytes):

- **Imagen1.png Imagem2.png são diferentes:** Isso indica que há uma diferença entre os dois arquivos.
- **byte 58:** Indica que a diferença foi encontrada no byte número 58 dos arquivos.
- **line 3:** Indica que a diferença foi encontrada na linha 3 dos arquivos. No contexto de arquivos binários, "linha" pode se referir a um número de byte específico, não necessariamente a uma linha de texto visível.

10. Feche o Terminal. Não apague os arquivos usados!

Parabéns! Agora, você sabe como verificar se um arquivo genérico (como imagens) copiado foi modificado ou não!

Atividade 1.3 – Verificando integridade de arquivos com função Hash no Kali Linux

Nesta atividade, vamos explorar o uso do comando MD5SUM do Kali Linux para comparar arquivos genéricos (Word, PDF, imagens, etc), que previamente foram copiados, e descobrir se houve manipulação mediante o cálculo da diferença entre eles.

Vamos começar inicializando nosso Kali Linux via RDP ao IP Kali: 192.168.98.40, com usuário "aluno" e senha "rnpesr".

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
└──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Navegue até a pasta “Documentos” com o seguinte comando:

```
└──(root㉿kali)-[~]
└─# cd /home/aluno/Documentos/
```

3. Verifique que os arquivos criados nas Atividades 1.1 e 1.2 estão presentes:

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
Imagem1_copia.png  Imagem1.png  Imagem2.png  Texto_copia.txt  Texto.txt
```

4. Calcule o Hash com o algoritmo MD5 de todos os arquivos. Os resultados dos caracteres hexadecimais serão diferentes na sua tentativa (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# md5sum Imagem1.png
ffee4a00274267c6dfd27f0607f6b345  Imagem1.png

└──(root㉿kali)-[/home/aluno/Documentos]
└─# md5sum Imagem1_copia.png
ffee4a00274267c6dfd27f0607f6b345  Imagem1_copia.png

└──(root㉿kali)-[/home/aluno/Documentos]
└─# md5sum Imagem2.png
702f2024904fd727d975210b692d9032  Imagem2.png

└──(root㉿kali)-[/home/aluno/Documentos]
└─# md5sum Texto.txt
e21678c114a62e8293e9f87bbd0b9235  Texto.txt

└──(root㉿kali)-[/home/aluno/Documentos]
└─# md5sum Texto_copia.txt
52ffef02ffe8f762147bc72b53e3bb0f  Texto_copia.txt
```

Os resultados do passo 4 são:

- **Imagen1.png e Imagem1_copia.png:** Ambos os arquivos têm o mesmo hash MD5: ffee4a00274267c6dfd27f0607f6b345. Isso indica que os conteúdos dos arquivos são idênticos.
- **Imagen2.png:** O hash MD5 de Imagem2.png é: 702f2024904fd727d975210b692d9032. Isso significa que o conteúdo de Imagem2.png é diferente do conteúdo de Imagem1.png.
- **Texto.txt e Texto_copia.txt:** Ambos os arquivos têm diferentes hashes MD5. Texto.txt tem o hash MD5: e21678c114a62e8293e9f87bb0b9235. Texto_copia.txt tem o hash MD5: 52ffef02ffe8f762147bc72b53e3bb0f. Isso indica que os conteúdos dos arquivos são diferentes..

5. Apague os arquivos usados até agora:

```
└──(root㉿kali)-[/home/aluno/Documentos]
└─# rm *
zsh: sure you want to delete all 5 files in /home/aluno/Documentos
[yn]? y

└──(root㉿kali)-[/home/aluno/Documentos]
└─# ls
```

6. Feche o Terminal.

Parabéns! Agora, você sabe como comparar arquivos com funções Hash.

Atividade 1.4 – Cifrando texto para proteger a confidencialidade no Kali Linux

Nesta atividade, vamos explorar os algoritmos ROT13 e a Cifra de César no Kali Linux para preservar a confidencialidade de textos.

Vamos começar inicializando nosso Kali Linux via RDP ao IP Kali: 192.168.98.40, com usuário “aluno” e senha “rnipesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnipesr”) para ser super usuário:

```
└─(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. A cifra ROT13 (rotate by 13 places) é uma cifra de substituição que substitui cada letra do alfabeto por aquela que está 13 posições à frente ou atrás no alfabeto. É uma cifra simétrica, o que significa que cifrar e decifrar usando ROT13 são processos idênticos. Agora, aplicaremos o algoritmo ROT13 em cima de uma mensagem de texto:

```
└─(root㉿kali)-[~]
└─# echo "Hackers do bem - Fundamental" | rot13
Unpxref qb orz - Shaqnzragny
```

3. Em seguida, decifre o texto cifrado:

```
└─(root㉿kali)-[~]
└─# echo "Unpxref qb orz - Shaqnzragny" | rot13
Hackers do bem - Fundamental
```

4. A Cifra de César no Kali Linux pode ser usada para criptografar ou descriptografar mensagens, permitindo que o usuário especifique a quantidade de posições a ser deslocada no alfabeto. Aqui está um exemplo simples de como fazer isso em um script Python. Navegue até a pasta “Documentos” com o seguinte comando:

```
└─(root㉿kali)-[~]
└─# cd /home/aluno/Documentos/
```

5. Crie o seguinte arquivo:

```
└─(root㉿kali)-[/home/aluno/Documentos]
└─# nano cesar_cipher.py
```

6. Insira o seguinte texto:

```
def caesar_cipher(text, shift):
    result = ""
    for char in text:
        if char.isalpha():
            shift_amount = shift % 26
            if char.islower():
                shifted = ord(char) + shift_amount
                if shifted > ord("z"):
                    shifted -= 26
                result += chr(shifted)
            else:
                shifted = ord(char) + shift_amount
                if shifted > ord("Z"):
                    shifted -= 26
                result += chr(shifted)
        else:
            result += char
    return result

def main():
    text = input("Digite o texto a ser cifrado/descifrado: ")
    shift = int(input("Digite a quantidade de posições a ser deslocada: "))
    encrypted_text = caesar_cipher(text, shift)
    print("Texto cifrado/descifrado:", encrypted_text)

if __name__ == "__main__":
    main()
```

A função percorre cada caractere (char) no texto de entrada. Se o caractere for uma letra, ele verifica se é maiúscula ou minúscula e realiza o deslocamento no alfabeto de acordo. Se não for uma letra, o caractere é mantido inalterado.

7. Aperte "Ctrl+X", "S" e "Enter" para salvar o arquivo.

8. Agora, use o script para cifrar o seguinte texto:

```
└─(root㉿kali)-[/home/aluno/Documentos]
└─# python3 cesar_cipher.py
Digite o texto a ser cifrado/descifrado: Hackers do bem - Fundamental
Digite a quantidade de posições a ser deslocada: 10
Texto cifrado/descifrado: Rkmuobc ny low - Pexnkwoxdkv
```

O resultado é o texto cifrado "Rkmuobc ny low - Pexnkwoxdkv".

9. Finalmente, use o script para decifrar o texto cifrado com deslocamento de -10 unidades (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
└─(root㉿kali)-[/home/aluno/Documentos]
└─# python3 cesar_cipher.py
Digite o texto a ser cifrado/descifrado: Rkmuobc ny low -
Digite a quantidade de posições a ser deslocada: -10
Texto cifrado/descifrado: Hackers do bem - Fundamental
```

O resultado é o texto original decifrado!

10. Apague o arquivo cesar_cipher.py:

```
└─(root㉿kali)-[/home/aluno/Documentos]
└─# rm cesar_cipher.py

└─(root㉿kali)-[/home/aluno/Documentos]
└─# ls
```

11. Feche o Terminal.

Parabéns! Agora você sabe implementar o ROT13 e a Cifra de César no Kali Linux.

Atividade 1.5 – Controlando fluxo de trabalho via linha de comando no Kali Linux

Nesta atividade, vamos explorar a ferramenta de gerenciamento de tarefas na linha de comando chamada Taskwarrior.

Vamos começar inicializando nosso Kali Linux via RDP ao IP Kali: 192.168.98.40, com usuário “aluno” e senha “rnpesr”.

1. Abra o Terminal e execute o seguinte comando (com senha “rnpesr”) para ser super usuário:

```
└──(aluno㉿kali)-[~]
└─$ sudo -i
[sudo] senha para aluno:
```

2. Crie um diretório para armazenar os dados do Taskwarrior:

```
└──(root㉿kali)-[~]
└─# mkdir -p ~/.task
```

3. Crie um arquivo de configuração básico dentro desse diretório:

```
└──(root㉿kali)-[~]
└─# echo 'data.location=~/task' > ~/.taskrc
```

4. Vamos adicionar 3 tarefas simples:

```
└──(root㉿kali)-[~]
└─# task add Comprar leite
Created task 1.
```

```
└──(root㉿kali)-[~]
└─# task add Comprar cafe
Created task 2.
```

```
└──(root㉿kali)-[~]
└─# task add Comprar salgados
Created task 3.
```

5. Liste as tarefas (a idade da tarefa "Age" será diferente na sua tentativa):

```
└──(root㉿kali)-[~]
└─# task list

ID Age Description      Urg
1 2min Comprar leite    0
2 2min Comprar cafe     0
3 1min Comprar salgados 0

3 tasks
```

6. Marque a tarefa 2 como urgente e liste novamente as tarefas:

```
└─(root㉿kali)-[~]
└─# task 2 modify +urgent
Modifying task 2 'Comprar cafe'.
Modified 1 task.

└─(root㉿kali)-[~]
└─# task list

ID Age Tags      Description          Urg
 2 2min urgent   Comprar cafe       0.8
 1 3min          Comprar leite     0
 3 2min          Comprar salgados  0

3 tasks
```

7. Coloque o prazo de 31 de dezembro de 2036 para a tarefa 3 e liste novamente as tarefas:

```
└─(root㉿kali)-[~]
└─# task 3 modify due:2036-12-31
Modifying task 3 'Comprar salgados'.
Modified 1 task.

└─(root㉿kali)-[~]
└─# task list

ID Age Tags      Due        Description          Urg
 3 3min          2036-12-31 Comprar salgados  2.4
 2 4min urgent   Comprar cafe       0.8
 1 4min          Comprar leite     0

3 tasks
```

Veja que um valor de urgência é atribuído para cada tarefa.

8. Finalmente, configure a tarefa 1 como concluída e veja que ela é apagada ao listar as tarefas (**NÃO SE ESQUEÇA DE PRINTAR ESTE PASSO!**):

```
└─(root㉿kali)-[~]
└─# task 1 done
Completed task 1 'Comprar leite'.
Completed 1 task.
You have more urgent tasks.

└─(root㉿kali)-[~]
└─# task list

ID Age Tags Due Description Urg
2 4min 2026-12-31 Comprar salgados 2.4
1 4min urgent Comprar cafe 0.8

2 tasks
```

9. Feche o Terminal. Parabéns! Agora, você conhece os principais comandos de gestão de fluxo de trabalho via linha de comando com a ferramenta Taskwarrior!