Laboratório - Familiarizando-se com o Linux Shell

# Apresentação

Neste laboratório, você usará a linha de comando do Linux para gerenciar arquivos e pastas e realizar algumas tarefas administrativas básicas.

Parte 1: Noções básicas do Shell

Parte 2: Copiando, Excluindo e Movendo Arquivos

# Equipamentos recomendados

=   Máquina virtual Security Workstation.

# Instruções

## Parte 1: Noções básicas do Shell

O shell é o termo usado para se referir ao interpretador de comandos no Linux. Também conhecido como Terminal, Linha de Comando e Prompt de Comando, o shell é uma maneira muito poderosa de interagir com um computador Linux.

### Etapa 1: Acesse a linha de comando

a.     Faça logon na Security Workstation VM como **analyst** , usando a senha **cyberops**. A conta **analyst** é usada como exemplo de conta de usuário neste laboratório.

b.     Para acessar a linha de comando, clique no ícone de **terminal** localizado no Dock, na parte inferior da tela da VM. O emulador de terminal é aberto.



### Etapa 2: Exibir páginas manuais na linha de comando.

É possível exibir a ajuda da linha de comando usando o comando **man**. Uma página de man, abreviação de página de manual, é uma documentação integrada dos comandos do Linux. Uma página de man fornece informações detalhadas sobre um determinado comando e todas as suas opções disponíveis.

a.     Para saber mais sobre a página do manual, digite:

[analyst@secOps ~]$ **man man**

#### Pergunta:

Cite algumas seções que estão incluídas em uma página de manual.

***Área de Resposta***

b.     Digite **q** para sair da página man.

c.     Useo o comando **man** para saber mais sobre o comando **cp**

[analyst@secOps ~]$ **man cp**

#### Pergunta:

Qual é a função do **comando cp**?

***Área de Resposta***

Que comando você usaria para descobrir mais informações sobre o comando **pwd** ? Qual é a função do comando **pwd** ?

***Área de Resposta***

### Etapa 3: Criar e mudar de diretórios.

Nesta etapa, você usará o comando para mudança de diretório (**cd**), para criar um diretório (**mkdir**), e listar um diretório (**ls**)

**Nota**: Um diretório é uma outra palavra para pasta. Os termos diretório e pasta são usados como sinônimos neste laboratório.

a.     Digite **pwd** no prompt.

[analyst@secOps ~]$ **pwd**

/home/analyst

#### Pergunta:

Qual é o diretório atual?

***Área de Resposta***

b.     Navegue no diretório **/home/analyst** se não for seu diretório atual. Digite **cd /home/analyst**

[analyst @secOps/] $ **cd /home/analyst/**

c.     Digite **ls -l** no prompt de comando para listar os arquivos e pastas que estão na pasta atual. Em relação a lista, a opção **-l** exibe o tamanho do arquivo, permissões, propriedade, data de criação e muito mais.

[analyst@secOps ~]$ **ls –l**

total 20

drwxr-xr-x 2 analyst analyst 4096 Mar 22  2018 Desktop

drwxr-xr-x 3 analyst analyst 4096 Apr  2 14:44 Downloads

drwxr-xr-x 9 analyst analyst 4096 Jul 19  2018 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar 21  2018 second\_drive

-rw-r--r-- 1 analyst analyst  255 Apr 17 16:42 space.txt

d.     No diretório atual, use o comando **mkdir** para criar três novas pastas: **cyops\_folder1**, **cyops\_folder2**, e **cyops\_folder3**. Digite **mkdir cyops\_folder1** e pressione **Enter**. Repita estas etapas para criar **cyops\_folder2** e **cyops\_folder3**.

[analyst @secOps ~] $ **mkdir cyops\_folder1**

[analyst @secOps ~] $ **mkdir cyops\_folder2**

[analyst @secOps ~] $ **mkdir cyops\_folder3**

[analyst @secOps ~] $

e.     Digite **ls -l** para verificar se as pastas foram criadas:

[analyst@secOps ~]$ **ls –l**

total 32

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:01 cyops\_folder1

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:02 cyops\_folder2

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:02 cyops\_folder3

drwxr-xr-x 2 analyst analyst 4096 Sep 26  2014 Desktop

drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads

drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar  3 15:56 second\_drive

-rw-r--r-- 1 analyst analyst  254 Aug 16 13:38 space.txt

f.      Digite **cd /home/analyst/cyops\_folder3** no prompt de comando e pressione **Enter**.

[analyst @secOps ~] $ **cd /home/analyst/cyops\_folder3**

[analyst @secOps cyops\_folder3] $

#### Pergunta:

Em que pasta você está agora?

***Área de Resposta***

**Observação**: no prompt **[analyst @secOps ~] $** acima: O símbolo til **~** representa o diretório pessoal do usuário atual. Neste exemplo, o diretório home do usuário atual é **/home/analyst**. Após o comando **cd /home/analyst/cyops\_folder3**, o diretório home do usuário atual agora é **/home/analyst/cyops\_folder3**.

**Nota**: **$** (cifrão) indica privilégio de usuário regular. Se uma ‘**#’** (hashtag ou sinal de libra) é exibido no prompt, indica privilégio elevado (**usuário root**).

**Nota**: Emboraesses símbolos, convenções e conceitos principais permaneçam os mesmos, o prompt de uma janela de terminal é altamente personalizável no Linux. Portanto, a estrutura de prompt vista na VM CyberOps Workstation provavelmente será diferente do prompt em outras instalações do Linux.

**Desafio:** Digite o comando **cd ~** e descreva o que acontece.

#### Pergunta:

Por quê isso aconteceu?

***Área de Resposta***

g.     Use o comando **mkdir** para criar uma nova pasta chamada **cyops\_folder4** **dentro da pasta cyops\_folder3**:

[analyst @secOps ~] $ **mkdir /home/analyst/cyops\_folder3/cyops\_folder4**

[analyst @secOps ~] $

h.     Use o comando **ls -l** para verificar a criação da pasta.

analyst @secOps ~] $ **ls —l /home/analyst/cyops\_folder3**

total 4

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:04 cyops\_folder4

i.      Até este ponto, temos usado *caminhos completos ou absolutos.* Caminho absoluto é o termo usado quando se refere a caminhos que sempre começam no diretório raiz (/). Também é possível trabalhar com *caminhos relativos.* Caminhos relativos reduzem a quantidade de texto a ser digitado. Para entender caminhos relativos, devemos entender o**.** e **..** (ponto e ponto duplo) diretórios. No diretório **cyops\_folder3**, emita um **ls —la**:

analyst @secOps ~] $ **ls —la /home/analyst/cyops\_folder3**

total 12

drwxr-xr-x  3 analyst analyst 4096 Aug 16 15:04 .

drwxr-xr-x 20 analyst analyst 4096 Aug 16 15:02 ..

drwxr-xr-x  2 analyst analyst 4096 Aug 16 15:04 cyops\_folder4

A opção **-a** diz a **ls** para mostrar todos os arquivos. Repare no **. e** **..** listagens mostradas por **ls**. Essas listagens são usadas pelo sistema operacional para rastrear o diretório atual (.) e o diretório pai (..) Você pode ver o uso do **.** e **..** ao usar o comando **cd** para alterar diretórios. Usando o comando **cd** para alterar o diretório para o **.** diretório incorre em nenhuma alteração de diretório visível como o **.** aponta para o próprio diretório atual.

j.      Altere o diretório atual para /home/analyst/cyops\_folder3:

[analyst @secOps ~] $ **cd /home/analyst/cyops\_folder3**

[analyst @secOps cyops\_folder3] $

k.     Digite **cd .**

[analyst @secOps cyops\_folder3] $ **cd .**

[analyst @secOps cyops\_folder3] $

#### Pergunta:

O que acontece?

***Área de Resposta***

l.      Alterando o diretório para o arquivo **..** , irá mudar para o diretório que está um nível acima. Este diretório também é conhecido como *diretóriopai*. Digite **cd ..**

[analyst @secOps cyops\_folder3] $ **cd ..**

[analyst @secOps ~] $

#### Pergunta:

O que acontece?

***Área de Resposta***

m.   Qual seria o diretório atual se você emitiu o cd **..** comando em [analyst @secOps/] $?

***Área de Resposta***

Qual seria o diretório atual se você emitiu o **cd..** comando em [analyst @secOps home] $?

***Área de Resposta***

Qual seria o diretório atual se você emitiu o cd **..** comando em [analyst @secOps/] $?

***Área de Resposta***

### Etapa 4: Saídas de redirecionamento.

Outro poderoso operador de linha de comando no Linux é conhecido como *redirect*. Representado pelo símbolo **>** , este operador permite que a saída de um comando seja redirecionada para algum local diferente da janela do terminal atual (o padrão).

a.     Use o comando **cd** para mudar para o diretório **/home/analyst/** (**~)** :

[analyst @secOps/] $ **cd /home/analyst/**

[analyst @secOps ~] $

b.     Use o comando **echo** para fazer eco de uma mensagem. Como nenhuma saída foi definida, echo irá enviar para a janela do terminal atual:

analyst @secOps ~] $ **echo Esta é uma mensagem ecoada para o terminal por echo.**

Esta é uma mensagem ecoada para o terminal pelo eco.

c.     Use o operador **>** para redirecionar a saída do eco para um arquivo de texto em vez de para a tela:

analyst @secOps ~] $ **echo Esta é uma mensagem ecoada para o terminal por echo. > some\_text\_file.txt**

Nenhuma saída foi mostrada.

#### Pergunta:

Isso é esperado? Explique.

***Área de Resposta***

d.     Observe que, mesmo que o arquivo **some\_text\_file.txt** não existisse, antes do comando echo, ele foi criado automaticamente para receber a saída gerada pelo **echo**. Use o comando **ls -l** para verificar se o arquivo foi realmente criado:

[analyst @secOps ~] $ **ls —l some\_text\_file.txt**

-rw-r--r-- 1 analyst analyst   50 Feb 24 16:11 some\_text\_file.txt

e.     Use o comando **cat** para exibir o conteúdo do arquivo de texto **some\_text\_file.txt** :

[analyst @secOps ~] $ **cat some\_text\_file.txt**

Esta é uma mensagem ecoada no terminal por eco.

f.      Use o operador **>** novamente para redirecionar uma saída de eco diferente do echo para o arquivo de texto **some\_text\_file.txt** :

analyst @secOps ~] $ **echo Esta é uma mensagem DIFERENTE,** mais uma vez ecoou para o terminal por echo. **> some\_text\_file.txt**

g.     Mais uma vez, use o comando **cat** para exibir o conteúdo do arquivo de texto **some\_text\_file.txt** :

[analyst @secOps ~] $ **cat some\_text\_file.txt**

Esta é uma mensagem DIFERENTE, mais uma vez ecoada para o terminal pelo eco.

#### Pergunta:

O que aconteceu com o arquivo de texto? Explique.

***Área de Resposta***

### Etapa 5: Redirecionar e anexar a um arquivo de texto.

a.     Semelhante ao operador **>**, o operador **>>** também permite redirecionar dados para arquivos. A diferença é que **>>** acrescenta dados ao final do arquivo referido, mantendo o conteúdo atual intacto. Para anexar uma mensagem ao some\_text\_file.txt, execute o comando abaixo:

[analyst @secOps ~] $echo Esta é outra linha de texto. Ele será anexado ao arquivo de saída. >> some\_text\_file.txt

b.     Use o comando **cat** para exibir novamente o conteúdo do arquivo de texto **some\_text\_file.txt**:

[analyst @secOps ~] $ **cat some\_text\_file.txt**

Esta é uma mensagem DIFERENTE, mais uma vez ecoada para o terminal pelo eco.

Esta é outra linha de texto. Será ANEXADA ao arquivo de saída.

#### Pergunta:

O que aconteceu com o arquivo de texto? Explique.

***Área de Resposta***

### Etapa 6: Trabalhe com arquivos ocultos no Linux.

a.     No Linux, arquivos com nomes que começam com um '.' (ponto único) não são mostrados por padrão. Enquanto os arquivos de pontos não têm mais nada de especial sobre eles, eles são chamados de arquivos ocultos por causa desse recurso. Exemplos de arquivos ocultos são **.file5**, **.file6**, **.file7**.

**Nota**: Não confunda arquivos de pontos com o indicador de diretório atual “**.**” símbolo. Nomes de arquivos ocultos começam com um ponto (ponto), seguido por mais caracteres enquanto o diretório ponto é um diretório oculto composto por apenas um único ponto.

b.     Use **ls -l** para exibir os arquivos armazenados no diretório home do analista.

[analyst@secOps ~]$ **ls –l**

#### Pergunta:

Quantos arquivos são exibidos?

***Área de Resposta***

c.     Use o comando **ls -la** para exibir todos os arquivos no diretório home do analista, incluindo os arquivos ocultos.

[analyst@secOps ~]$ **ls –la**

#### Perguntas:

Quantos arquivos mais são exibidos do que antes? Explique.

***Área de Resposta***

É possível ocultar diretórios inteiros adicionando um ponto antes de seu nome também? Há algum diretório na saída de ls -la acima?

***Área de Resposta***

Dê três exemplos de arquivos ocultos mostrados na saída de **ls -la** acima.

***Área de Resposta***

d.     Digite o comando **man** ls no prompt para saber mais sobre o comando **ls**.

[analyst@secOps ~]$ **man ls**

e.     Use a tecla de seta para baixo (uma linha de cada vez) ou a barra de espaço (uma página de cada vez) para rolar a página para baixo e localizar a opção **-a** usada acima e ler sua descrição para se familiarizar com o comando **ls -a**.

## Parte 2: Copiando, Excluindo e Movendo Arquivos

### Etapa 1: Copiando arquivos

a.     O comando **cp** é usado para copiar arquivos em torno do sistema de arquivos local. Ao usar **cp**, uma nova cópia do arquivo é criada e colocada no local especificado, deixando o arquivo original intacto. O primeiro parâmetro é o arquivo de origem e o segundo é o destino. Execute o comando abaixo para copiar **some\_text\_file.txt** do diretório home para a pasta **cyops\_folder2**:

[analyst @secOps ~] $ **cp some\_text\_file.txt cyops\_folder2/**

Identifique os parâmetros no comando cp acima.

#### Pergunta:

Quais são os arquivos de origem e destino? (use caminhos completos para representar os parâmetros)

***Área de Resposta***

b.     Use o comando **ls** para verificar se **some\_text\_file.txt** está agora em **cyops\_folder2:**

[analyst @secOps ~] $ **ls cyops\_folder2/**

some\_text\_file.txt

c.     Use o comando **ls** para verificar se **some\_text\_file.txt** também está no diretório home:

[analyst@secOps ~]$ **ls –l**

total 36

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:01 cyops\_folder1

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:11 cyops\_folder2

drwxr-xr-x 3 analyst analyst 4096 Aug 16 15:04 cyops\_folder3

drwxr-xr-x 2 analyst analyst 4096 Sep 26  2014 Desktop

drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads

drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar  3 15:56 second\_drive

-rw-r--r-- 1 analyst analyst  142 Aug 16 15:09 some\_text\_file.txt

-rw-r--r-- 1 analyst analyst  254 Aug 16 13:38 space.txt

### Etapa 2: Excluindo Arquivos e Diretórios

a.     Use o o comando **rm** para remover arquivos Execute o comando abaixo para remover o arquivo **some\_text\_file.txt** do diretório home.  
  
O comando ls é então usado para mostrar que o arquivo **some\_text\_file.txt** foi removido do diretório home:

[analyst @secOps ~] $ **rm some\_text\_file.txt**

[analyst@secOps ~]$ **ls –l**

total 32

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:01 cyops\_folder1

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:11 cyops\_folder2

drwxr-xr-x 3 analyst analyst 4096 Aug 16 15:04 cyops\_folder3

drwxr-xr-x 2 analyst analyst 4096 Sep 26  2014 Desktop

drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads

drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar  3 15:56 second\_drive

-rw-r--r-- 1 analyst analyst  254 Aug 16 13:38 space.txt

b.     No Linux, os diretórios são vistos como um tipo de arquivo. Como tal, o comando **rm** também é usado para excluir diretórios, mas a opção **-r** (recursiva) deve ser usada. Observe que todos os arquivos e outros diretórios dentro de um determinado diretório também são excluídos ao excluir um diretório pai com a opção -r. Execute o comando abaixo para excluir a pasta **cyops\_folder1** e seu conteúdo:

[analyst @secOps ~] $ **rm —r cyops\_folder1**

[analyst@secOps ~]$ **ls –l**

total 28

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:11 cyops\_folder2

drwxr-xr-x 3 analyst analyst 4096 Aug 16 15:04 cyops\_folder3

drwxr-xr-x 2 analyst analyst 4096 Sep 26  2014 Desktop

drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads

drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar  3 15:56 second\_drive

-rw-r--r-- 1 analyst analyst  254 Aug 16 13:38 space.txt

### Etapa 3: Movendo arquivos e diretórios

a.     Mover arquivos funciona de forma semelhante à cópia de arquivos. A diferença é que mover um arquivo o remove de seu local original. Use os comandos **mv** para mover arquivos pelo sistema de arquivos local. Como os comandos **cp**, o comando mv também requer parâmetros de origem e destino. Execute o comando abaixo para mover o **some\_text\_file.txt** de **/home/analyst/cyops\_folder2** de volta para o diretório home:

[analyst @secOps ~] $ **mv cyops\_folder2/some\_text\_file.txt.**

[analyst @secOps ~] $ **ls —l cyops\_folder2/**

total 0

[analyst @secOps ~] $ **ls —l /home/analyst/**

total 32

drwxr-xr-x 2 analyst analyst 4096 Aug 16 15:13 cyops\_folder2

drwxr-xr-x 3 analyst analyst 4096 Aug 16 15:04 cyops\_folder3

drwxr-xr-x 2 analyst analyst 4096 Sep 26  2014 Desktop

drwx------ 3 analyst analyst 4096 Jul 14 11:28 Downloads

drwxr-xr-x 8 analyst analyst 4096 Jul 25 16:27 lab.support.files

drwxr-xr-x 2 analyst analyst 4096 Mar  3 15:56 second\_drive

-rw-r--r-- 1 analyst analyst  142 Aug 16 15:11 some\_text\_file.txt

-rw-r--r-- 1 analyst analyst  254 Aug 16 13:38 space.txt

#### Pergunta:

Que comando você usou para realizar a tarefa?

***Área de Resposta***

# Reflexão

Quais são as vantagens de usar a linha de comando do Linux?

***Área de Resposta***

*Fim do documento*