Curso 4 de 8 - Linux e SQL

Módulo 1

Introdução aos sistemas operacionais

Sistema Operacional (OS): Interface entre o hardware do computador e o usuário. É responsável por fazer com que o computador funcione da forma mais eficiente possível e, ao mesmo tempo, por facilitar o seu uso.

Sistemas operacionais comuns:

- Windows
- MacOS
- Linux
- Android
- IOS
- Chrome OS

Hardware: Componentes físicos de um computador.

Sistema operacional legado: Sistema operacional desatualizado, mas que ainda é utilizado por algum motivo como o suporte de softwares específicos. A utilização desses sistemas não é segura devido a problemas de segurança derivados da falta de atualizações.

Sistema operacional em funcionamento

O trabalho de um sistema operacional é ajudar outros programas de computador a serem executados com eficiência, cuidando de todos os detalhes confusos relacionados ao controle do hardware do computador.

Ao ligar o computador: Ao apertar o botão de ligar e interagir com o hardware, o computador é inicializado e o sistema operacional é ativado.

Inicializar o computador significa que um microchip especial chamado BIOS está ativado (ou o UEFI em sistemas mais novos). Tanto o BIOS quanto o UEFI contêm instruções de inicialização que são responsáveis por carregar um programa especial chamado carregador de boot (inicialização). O carregador de boot é responsável por iniciar o sistema operacional.

Podem surgir vulnerabilidades neste processo, visto que o BIOS não é verificado pelo antivírus, estando vulnerável à infecção por malware.

Comunicação com o computador: Usuário -> Aplicativos -> Sistema operacional -> Hardware. Para concluir tarefas, o usuário usa aplicativos no computador. Ao fazer isso, o aplicativo envia uma solicitação para o sistema operacional, que interpreta essa solicitação e a direciona para o componente apropriado do hardware do computador. O hardware, então, enviará informações de volta ao sistema operacional, que também é enviado de volta ao aplicativo.

Aplicativo: Programa que executa uma tarefa específica

Alocação de recursos: O sistema operacional realiza a alocação e o gerenciamento dos recursos do computador para garantir que a capacidade limitada do sistema do computador seja usada onde é mais necessária e com mais eficiência.

Interface do usuário (UI)

Interface de usuário: Programa que permite ao usuário controlar as funções do sistema operacional.

Interface gráfica do usuário (GUI): Interface de usuário que usa ícones na tela para gerenciar diferentes tarefas no computador. Componentes básicos do GUI:

- **Menu inicial**
- **Barra de tarefas**
- **Área de trabalho com ícones e atalhos.**

Interface de linha de comando (CLI): Interface de usuário baseada em texto que usa comandos para interagir com o computador. Esses comandos se comunicam com o sistema operacional e executam tarefas como abrir programas. É mais flexível e poderosa do que uma GUI.

Glossário de revisão

Aplicativo: Um programa que executa uma tarefa específica
Basic Input/Output System (BIOS): Um microchip que contém instruções de carga para o computador e é predominante em sistemas mais antigos

Carregador de inicialização: Um software que inicializa o sistema operacional
Interface de Linha de Comando (CLI): Uma interface do usuário (IU) baseada em texto que usa comandos para interagir com o computador
Interface gráfica do usuário (GUI): Uma interface do usuário que usa ícones na tela para gerenciar diferentes tarefas no computador
Hardware: Os componentes físicos de um computador
Sistema operacional legado: Um sistema operacional desatualizado, mas que ainda está sendo usado
Sistema operacional (SO): A interface entre o hardware do computador e o usuário
Memória de acesso aleatório (RAM): Um componente de hardware usado para memória de curto prazo
Unified Extensible Firmware Interface (UEFI): Um microchip que contém instruções de carga para o computador e substitui o BIOS em sistemas mais modernos
Interface do usuário (IU): Um programa que permite que o usuário controle as funções do sistema operacional
Máquina virtual (VM): Uma versão virtual de um computador físico

Módulo 2

Tudo sobre o linux

Linux: Sistema operacional de código aberto. Linus Torvalds queria melhorar o sistema UNIX e torná-lo de código aberto e acessível a qualquer pessoa, introduzindo o kernel linux. Ao mesmo tempo, Richard Stallman começou a trabalhar no GNU, que também era um sistema baseado em UNIX. Ambos compartilhavam do objetivo de criar um software gratuito e aberto a qualquer pessoa. Ao juntar o GNU ao kernel de Linus, surge o Linux.

O Linux e muitos dos programas que vêm com o Linux são licenciados sob os termos da Licença Pública GNU, que permite que você os use, compartilhe e modifique livremente.

Componentes do Linux:

- **Usuário:** A pessoa que interage com o computador. O usuário é a primeira camada da arquitetura do linux, está iniciando tarefas ou os comandos que o sistema operacional executará.
- **Aplicativos:** Um programa que executa uma tarefa específica. Um exemplo de aplicativo no Linux é o editor de texto Nano.
- **Shell:** Interpretador de linha de comando, processa comandos e gera resultados.

- **Filesystem Hierarchy Standard (FHS):** Componente do Linux que organiza os dados.
- **Kernel:** Componente do Linux que gerencia os processos e a memória (núcleo do sistema). Ele se comunica com o hardware para executar os comandos enviados pelo shell. Usa drivers para permitir que os aplicativos executem tarefas.
- **Hardware:** Componentes físicos de um computador.

Distribuições Linux

Distribuições: Diferentes versões do Linux, também chamadas de distros ou sabores do Linux. Diferentes distribuições contêm diferentes programas pré-instalados, interfaces de usuário e outros. As distribuições incluem:

- **Kernel linux**
- **Utilitários**
- **Sistema de gerenciamento de pacotes**
- **Instalador**

Principais Distros:

- **Red Hat (CentOS)**
- **Slackware (SUSE)**
- **Debian (Ubuntu e KALI LINUX)**

Kali Linux: Marca registrada da Offensive Security e é derivada do Debian. É uma distribuição de código aberto feita especificamente com o teste de penetração e a perícia digital em mente.

Teste de penetração: Ataque simulado que ajuda a identificar vulnerabilidades em sistemas, redes, sites, aplicativos e processos.

Ferramentas de teste de penetração no Kali Linux:

- **Metasploit:** Usado para procurar e explorar vulnerabilidades em máquinas
- **Burp Suite:** Ferramenta que ajuda a testar pontos fracos em aplicativos da web.
- **John the Ripper:** Ferramenta usada para adivinhar senhas.

Perícia digital: Processo de coleta e análise de dados para determinar o que aconteceu depois de um ataque

Ferramentas de perícia digital no Kali Linux:

- **tcpdump:** Analisador de pacotes de linha de comando usado para capturar o tráfego da rede.
- **Wireshark:** Ferramenta de interface gráfica usada para analisar tráfego de rede ao vivo e capturado.
- **Autopsy:** Ferramenta forense usada para analisar discos rígidos e smartphones.

Pacote: Software que pode ser combinado com outros pacotes para formar um aplicativo. Alguns pacotes podem ser grandes o suficiente para formar aplicativos por conta própria.

Os pacotes contêm os arquivos necessários para que um aplicativo seja instalado, que podem incluir Dependências, que são arquivos suplementares usados para executar um aplicativo.

Gerenciador de pacotes: Ferramenta que ajuda o usuário a instalar, gerenciar e remover pacotes ou aplicativos. O Linux usa vários gerenciadores de pacote, como o dpkg. Gerenciadores diferentes usam extensões de arquivos diferentes, tais como .rpm (Red Hat) e o .deb (Debian).

Ferramenta de gerenciamento de pacotes: Ferramenta que permite trabalhar com pacotes por meio do shell, mais comumente usada do que o gerenciador de pacote pois permite que o usuário execute tarefas básicas com mais facilidade, como a instalação de um novo pacote. Duas ferramentas notáveis são o Advanced Package Tool (APT) e o Yellowdog Updater Modified (YUM).

Recursos do laboratório Linux (Qwiklabs): Clicar em **Start Lab (Iniciar laboratório)** para abrir um terminal temporário, as instruções do laboratório serão movidas para o lado direito da tela. A caixa de diálogo contém o botão **End Lab (Terminar laboratório)** o **cronômetro** e o botão **Open Linux Console (Abrir console do Linux)**. Pode clicar em **Check my progress** para verificar o status de conclusão da tarefa ou receber uma dica.

O shell

Shell: Interpretador de linha de comando, ajuda o usuário a se comunicar com o sistema operacional por meio da linha de comando.

Comando: Uma instrução que diz ao computador que faça alguma coisa. O shell se comunica com o kernel para executar esses comandos.

Entrada padrão: Informações recebidas pelo sistema operacional pela linha de comando.

echo: Comando do linux que gera uma sequência de strings especificada.

Dados de string: Dados que consistem em uma sequência ordenada de caracteres.

Saída padrão: Informação retornada pelo sistema operacional por meio do shell.

Erro padrão: Mensagens de erro retornada pelo sistema operacional por meio do shell.

Glossário de revisão

Aplicativo: Um programa que executa uma tarefa específica
Bash: O shell padrão na maioria das distribuições Linux
CentOS: Uma distribuição de código aberto que está intimamente relacionada à Red Hat
CPU (Central Processing Unit, unidade central de processamento): O processador principal de um computador, que é usado para realizar tarefas gerais de computação em um computador
Comando: Uma instrução que diz ao computador para fazer algo
Forense digital: A prática de coletar e analisar dados para determinar o que aconteceu após um ataque
Diretório: Um arquivo que organiza onde outros arquivos são armazenados
Distribuições: As diferentes versões do Linux
Caminho do arquivo: O local de um arquivo ou diretório
Padrão de hierarquia do sistema de arquivos (FHS): O componente do sistema operacional Linux que organiza os dados
Interface gráfica do usuário (GUI): Uma interface do usuário que usa ícones na tela para gerenciar diferentes tarefas no computador
Disco rígido: Um componente de hardware usado para memória de longo prazo
Hardware: Os componentes físicos de um computador
Hardware interno: Os componentes necessários para o funcionamento do computador
Kali Linux ™: Uma distribuição de código aberto do Linux que é amplamente usada no setor de segurança
Kernel: O componente do sistema operacional Linux que gerencia os processos e a memória
Linux: Um sistema operacional de código aberto
Pacote: Um software que pode ser combinado com outros pacotes para formar um aplicativo

Gerenciador de pacotes: Uma ferramenta que ajuda os usuários a instalar, gerenciar e remover pacotes ou aplicativos
Parrot: Uma distribuição de código aberto que é comumente usada para segurança
Teste de penetração (pen test): Um ataque simulado que ajuda a identificar vulnerabilidades em sistemas, redes de computadores, sites, aplicativos e processos
Dispositivos periféricos: Componentes de hardware que são conectados e controlados pelo sistema do computador
Memória de acesso aleatório (RAM): Um componente de hardware usado para memória de curto prazo
Red Hat® Enterprise Linux® (também chamado simplesmente de Red Hat neste curso)**:** Uma distribuição do Linux baseada em assinatura criada para uso corporativo
Shell: O interpretador da linha de comando
Erro padrão: Uma mensagem de erro retornada pelo sistema operacional por meio do shell
Entrada padrão: Informações recebidas pelo sistema operacional por meio da linha de comando
Saída padrão: Informações retornadas pelo sistema operacional por meio do shell
Dados de String: Dados que consistem em uma sequência ordenada de caracteres
Ubuntu: Uma distribuição de código aberto e fácil de usar que é amplamente utilizada em setores de segurança e outros
Usuário: a pessoa que interage com um computador

Módulo 3

Sistema de arquivos do Linux

Tarefas de linha de comando de um analista de segurança:

- **Trabalhar com logs de servidor**
- **Navegar, gerenciar e analisar arquivos remotamente**
- **Verificar e configurar o acesso de usuários e grupos**
- **Dar autorização e definir permissões de arquivos**

Bash: Shell padrão na maioria das distribuições Linux.

Comando: Instrução que diz ao computador o que fazer.

Argumento: Informação específica necessária para um comando. Alguns comandos usam vários argumentos. Todos os comandos e argumentos diferenciam letras maiúsculas de minúsculas!

Diretório raiz: O diretório de nível mais alto no Linux. O FHS é um sistema hierárquico que se ramifica a partir de um diretório raiz. É designado por uma única barra (/). Os outros subdiretórios se ramificam de outros subdiretórios, se distanciando cada vez mais da pasta raiz, sendo representados por uma série de barras (exemplo: /home/iagobgc/Downloads).

Comandos de navegação e leitura:

- **pwd:** Imprime o diretório de trabalho na tela.
- **whoami:** Exibe o nome do usuário atual
- **ls:** Exibe os nomes dos arquivos e diretórios no diretório de trabalho atual.
- **cd:** Navega entre diretórios. Pode usar “cd ..” para subir um nível na estrutura do arquivo (voltar um diretório).
- **cat:** Exibe o conteúdo de um arquivo
- **head:** Exibe o começo de um arquivo, as 10 primeiras linhas por padrão. Pode usar “-n número” para especificar o número de linhas a serem exibidas a partir do começo.
- **tail:** Faz o oposto do head, exibindo apenas o final do arquivo, sendo as 10 últimas como padrão.
- **less:** Exibe uma página do arquivo de cada vez. É possível usar alguns controles durante a leitura usando o teclado, tais como: **barra de espaço** (avança uma página), **b** (retroceder uma página), **seta para baixo** (avançar uma linha), **seta para cima** (retroceder uma linha), **q** (Sair)

Diretórios padrão da FHS:

- **/home:** Cada usuário tem seu próprio diretório pessoal.
- **/bin:** Significa “binary” (binário) e contém binários e outros executáveis. Executáveis são arquivos que contêm uma série de comandos que o computador precisa seguir para executar programas e realizar funções.
- **/etc:** Armazena arquivos de configuração do sistema
- **/tmp:** Armazena arquivos temporários. Comumente usado por atacantes porque qualquer pessoa no sistema pode modificar os dados nesses arquivos.
- **/mnt:** Significa “montagem”, armazena mídia como unidades USB e discos rígidos.

O comando “man hier” exibe informações sobre os diretórios.

Gerencie o conteúdo do arquivo no Bash

grep: Pesquisa um arquivo específico e retorna todas as linhas do arquivo que contém uma String específica. **Exemplo:** (**grep string arquivo.txt**).

Piping (|): Envia uma saída padrão de um comando como entrada padrão para outro comando para processamento adicional. **Exemplo:** (**ls /home/analyst/reports | grep users**) -> Está filtrando os arquivos e diretórios que serão mostrados pelo ls. Serve para juntar vários comandos.

find: Procura diretórios e arquivos que atendam aos critérios especificados. É possível adicionar diversos critérios ao comando find, tais como string específicas, determinado tamanho de arquivo ou última vez em que foi modificado. Após o comando find, o primeiro argumento representa onde a pesquisa será feita (**find /home/iagobgc**), após isso, você precisa indicar os critérios da pesquisa por meio de uma option, que são especificações que começam com um "**-**" que modificam o comportamento de um comando (**find /home/iagobgc -option**).

Tipos de option:

- **-name e iname:** Encontra nomes de arquivos ou diretórios que contenham uma string específica. "-name" distingue letras maiusculas de minúsculas e "-iname" não. A string específica deve ser inserida entre asteriscos e aspas após a option. Exemplo: (**find /home/iagobgc -name "**updates**"**). Também pode ser usado para pesquisar extensões de arquivo colocando "*extensão" no lugar da string especificada. Exemplo: (**find /home/iagobgc -name "*txt"**)
- **-mtime:** Procura por arquivos modificados recentemente. Após a option, a procura em dias deve ser colocada após "-" para "menos dias" e "+" para mais dias. Exemplo: (**find /home/iagobgc -mtime -3**) retorna todos os arquivos e diretórios modificados nos últimos 3 dias.
- **-not:** Pode ser utilizado antes de outra option (como -name e -name) e serve para ignorar a especificação colocada. Ou seja, mostrará tudo o que tem no diretórios, menos o que foi especificado.

mkdir: Cria um novo diretório. Utiliza 1 argumento que indica o nome do diretório que será criado, exemplo: (**mkdir logs**) - cria um diretório chamado "logs".

rmdir: Remove ou exclui um diretório. Utiliza 1 argumento que indica o nome do diretório que se quer excluir, exemplo: (**rmdir logs**) - exclui o diretório "logs".

touch: Cria um novo arquivo. Utiliza 1 argumento que indica o nome do arquivo criado e a sua extensão (txt, chc, exe, etc…), exemplo: (**touch emails.txt**) - cria um arquivo de texto chamado “emails”.

rm: Remove ou exclui um arquivo. Utiliza 1 argumento que indica o nome do arquivo que se quer excluir e a sua extensão, exemplo: (**rm emails.txt**) - exclui o arquivo de texto chamado “emails”.

mv: Move um arquivo ou diretório para um novo local. Utiliza 2 argumentos, o primeiro indica o arquivo a ser movido e o segundo a localização para onde será movido, exemplo: (**mv logs.txt /home/iagobgc/Documentos**) - manda o arquivo “logs.txt” para a pasta Documentos. Também pode ser usado para renomear o arquivo ao colocar o nome alternativo no lugar do segundo argumento, exemplo: (**mv logs.txt registros.txt**) - muda o nome do arquivo “logs” para “registros”.

cp: Copia um arquivo ou diretório em um novo local. Utiliza 2 argumentos, o primeiro indica o arquivo a ser copiado e o segundo o local onde ele será copiado, exemplo: (**cp receitas.txt /home/iagobgc/Documentos**) - copia o arquivo “receitas” para o diretório “Documentos”.

nano: Comando referente a um programa de edição de arquivos. Utiliza 1 argumento que se refere ao nome do arquivo que se quer editar, exemplo: (**nano trabalho.txt**) - abre a edição do arquivo de texto “trabalho”. Também pode ser utilizado para criar arquivos da mesma forma que editando, mas utilizando o nome de um arquivo que não existe no diretório atual.

Redirecionamento de saída padrão: Os operadores de colchete (**>**) e (**>>**) também podem ser usados para redirecionar a saída padrão. Quando usados com o **echo**, eles podem enviar a saída do comando para um arquivo, ao invés da tela. A diferença é que (**>**) sobrescreve o arquivo existente e (**>>**) adiciona o conteúdo ao final do arquivo. Exemplo: (**echo “29/01/2024” >> datas.txt**) - adiciona a string especificada no arquivos “datas.txt”.

Autenticação e autorização de usuários

Permissão: Tipo de acesso concedido a um arquivo ou diretório. Estão relacionadas a autorização.

Autorização: Conceito de conceder acesso a recursos específicos em um sistema. Permite limitar o acesso a arquivos ou diretórios específicos.

Permissões no Linux:

- **Read (r):** Permissões de leitura em arquivos significam que o conteúdo pode ser lido. Em um diretório, significa que o usuário pode ler todos os arquivos do diretório.
- **Write (w):** Permissões de gravação em arquivos significam que o conteúdo pode ser modificado. Em um diretório, significa que novos arquivos podem ser criados nesse diretório.
- **Execute (x):** Permissões de execução em arquivos significam que o arquivo pode ser executado se for um arquivo executável. Em diretórios, permitem que os usuários entrem em um diretório e acessem seus arquivos.

Tipos de proprietários:

- **Usuário (u):** É o proprietário do arquivo. Quando se cria um arquivo, você se torna o proprietário dele, mas a propriedade pode ser alterada.
- **Grupo (g):** Consiste em vários usuários, sendo uma das formas de gerenciar um ambiente multiusuário. Cada usuário faz parte de um determinado grupo.
- **Outros (o):** Podem ser considerados todos os usuários do sistema, ou seja, qualquer pessoa com acesso ao sistema pertence a esse grupo.

No Linux, as permissões dos arquivos e diretórios são representadas por uma string de 10 caracteres. Exemplo: "**drwxrwxrwx**" - o primeiro caractere indica o **tipo** de arquivo (no exemplo, um diretório), o segundo, terceiro e quarto caracteres indicam as **permissões do usuário**, o quinto, sexto e sétimo caracteres indicam as **permissões do grupo**, o oitavo, nono e décimo caracteres indicam as **permissões dos outros** (todos os usuários do sistema).

Caso alguma char da string esteja faltando, significa que aquela permissão não está concedida para o tipo de proprietário, exemplo: "**-rw-rw-rw-**" - a permissão de execução do arquivo não foi concedida para nenhum tipo de proprietário.

ls -l: Exibe as permissões dos arquivos e diretórios.

ls -a: Exibe arquivos ocultos.

ls -la: Exibe as permissões dos arquivos e diretórios, incluindo os ocultos.

chmod: Altera as permissões em arquivos e diretórios. Significa "modo de mudança". Cada alteração deve ser separada por vírgula, exemplo: (**chmod u+x,g-w,g=r teste.txt**) - Usa-se operadores matemáticos para adicionar ou retirar permissões. No exemplo, "u+x" está adicionando a permissão de execução ao tipo de proprietário "usuário", já em "g-w" está retirando a permissão de gravação do tipo de proprietário "grupo", e em "g=r" está retirando todas as permissões existentes para o tipo de proprietário "outros" e atribuindo apenas a permissão.

Usuário root (ou super usuário): É um usuário com privilégios elevados para modificar o sistema. Possuem nenhuma restrição, podem criar, modificar ou excluir qualquer arquivo e executar qualquer programa. Apenas usuários com privilégio de root podem adicionar novos usuários. Usuários normais podem ser adicionados ao root temporariamente.

Problemas em executar comandos como root:

- **Riscos de segurança**
- **Fácil de cometer erros irreversíveis**
- **Responsabilidade**

sudo: Comando que concede temporariamente permissões elevadas a usuários específicos. Permite a execução de comandos como um usuário elevado sem precisar entrar e sair de outra conta. Nem todos os usuários podem utilizar o sudo, pois devem ter acesso por meio de um arquivo de configuração chamado **sudoers**.

useradd: Adiciona um usuário ao sistema. Apenas usuários com privilégio de root podem usar o comando. Exemplo: (**sudo adduser iaguinho2**). Também é possível utilizar a option "**-m**" para criar o diretório pessoal junto com o usuário, exemplo: (**sudo useradd -m iaguinho2**) - cria o diretório "iaguinho2" na pasta /home para o novo usuário.

É possível utilizar outros argumentos para adicionar o usuário a algum grupo, o "**-g**" define o grupo padrão do usuário criado, exemplo: (**sudo adduser -g seguranca iaguinho2**) - cria o usuário "iaguinho2" e atribui a seu grupo padrão o endereço "seguranca". O argumento "**-G**" define grupos adicionais, exemplo: (**sudo adduser -G financas,administracao iaguinho2**) - cria o usuário "iaguinho2" e o adiciona aos grupos existentes "financas" e "administracao".

userdel: Remove um usuário do sistema. Apenas usuários com privilégio de root podem usar o comando. Exemplo: (**sudo userdel iaguinho2**). Para apagar o usuário juntamente com os arquivos do seu diretório, é necessário usar a option "**-r**", exemplo: (**sudo userdel -r iaguinho2**) - apaga o usuário "iaguinho2" e todos os arquivos do seu diretório pessoal.

usermod: Modifica as contas de usuários já existentes. As mesmas opções "**-g**" e "**-G**" do comando "**useradd**" podem ser usadas com usermod com usuários já existentes. Para alterar o grupo principal, utiliza-se a option "**-g**", exemplo: (**sudo usermod -g executivo iaguinho2**) - altera o grupo principal de iaguinho2 para "executivo".

Para adicionar o usuário a um grupo adicional, usa-se a option "**-G**" juntamente com a "**-a**" para anexar o usuário a um grupo existente, exemplo: (**sudo**

usermod -a -G marketing iaguinho2) - adiciona o usuário “iaguinho2” ao grupo adicional “marketing”.

- Caso não coloque a opção “-a”, o -G irá substituir todos os grupos adicionais do usuário pelos especificados.

Existem outras options para o usermod, tais como: “**-d**” para mudança de diretório inicial, “**-l**” para mudança do nome de login e “**-L**” para bloquear a conta. Exemplo: (**sudo usermod -d /home/diretorio_iaguinho2 iaguinho2**) - Muda o diretório pessoal de “iaguinho2” para “/home/diretorio_iaguinho2”.

chown: Muda as propriedades de um arquivo ou diretório. Pode ser usado para mudar o usuário proprietário de algum arquivo ou diretório, exemplo: (**sudo chown iaguinho2 arquivo.txt**) - muda o usuário proprietário do arquivo para “iaguinho2”. Também é possível alterar o grupo proprietário ao inserir dois pontos (**:**), exemplo: (**sudo chown :segurança arquivo.txt**) - mudou o grupo proprietário do arquivo para “segurança”.

groupdel: Apaga algum grupo no sistema. Toda vez que um usuário novo é criado, um grupo de mesmo nome também é criado, tendo apenas o usuário novo como membro. É uma boa prática apagar os grupos não mais utilizados. Exemplo: (**sudo groupdel novo_usuario**).

passwd: Altera a senha do usuário. Exemplo: (**sudo passwd usuario**).

su: Muda o usuário do terminal temporariamente sem ter que sair da sessão, exemplo: (**su - usuario2**). Utiliza-se “exit” para sair do usuário temporário.

Obter ajuda no Linux

Como o Linux é de código aberto, ele se tornou uma comunidade global de usuários que contribuem com frequência. Esta comunidade é extremamente valiosa para todos os usuários, pois é possível encontrar respostas para as tarefas do dia a dia apenas pesquisando na internet. Outra fonte confiável é o Unix & Linux Exchange.

man: Exibe informações sobre outros comandos e como eles funcionam, exemplo: (**man usermod**) - mostra o manual do comando “usermod”.

whatis: Exibe uma descrição de um comando de forma breve, exemplo: (**whatis tail**) - descreve o que o comando “tail” faz.

apropos: Pesquisa nas descrições dos manuais de comandos por uma string específica, ou seja, procura por comandos que possuam a palavra especificada em seus manuais. Exemplo: (**apropos password**) - mostra todos os comandos que possuem a palavra “password” em seus manuais. Também é possível utilizar a option “**-a**” para procurar por duas strings, exemplo: (**apropos -a change password**).

Glossário de revisão

Caminho absoluto do arquivo: O caminho completo do arquivo, que começa na raiz
Argumento (Linux): Informações específicas necessárias para um comando
Autenticação: O processamento de verificação da identidade de alguém
Autoridade: O conceito de concessão de acesso a recursos específicos em um sistema
Bash: O shell padrão na maioria das distribuições Linux
Comando: Uma instrução que diz ao computador para fazer algo
Caminho do arquivo: O local de um arquivo ou diretório
Filesystem Hierarchy Standard (FHS) (Padrão de hierarquia do sistema de arquivos): O componente do sistema operacional Linux que organiza os dados
Filtragem: Seleção de dados que correspondem a uma determinada condição
nano: um editor de arquivos de linha de comando disponível por padrão em muitas distribuições do Linux
Opções: Entrada que modifica o comportamento de um comando
Permissões: O tipo de acesso concedido a um arquivo ou diretório
Princípio do privilégio mínimo: O conceito de conceder apenas o acesso e a autorização mínimos necessários para concluir uma tarefa ou função
Caminho relativo do arquivo: Um caminho de arquivo que começa no diretório atual do usuário
Diretório raiz: O diretório de nível mais alto no Linux
Usuário raiz (ou superusuário): Um usuário com privilégios elevados para modificar o sistema
Entrada padrão: Informações recebidas pelo sistema operacional por meio da linha de comando
Saída padrão: Informações retornadas pelo sistema operacional por meio do shell

Módulo 4

Introdução à SQL e bancos de dados

Banco de dados: Coleção organizada de informações ou dados. Geralmente relacionado a planilhas.

Planilha VS Banco de dados:

- **P:** Projetada para um único usuário ou uma pequena equipe
- **BD:** Projetado para ser acessado por várias pessoas simultaneamente
- **BD:** Armazena montes massivos de dados
- **BD:** Realiza tarefas complexas ao acessar dados

Banco de dados relacional: Banco de dados estruturado contendo tabelas que se relacionam entre si. Cada tabela contém campos de informação, que são representados por colunas em uma planilha. Abaixo das colunas ficam as linhas, que representam os dados específicos relacionados às colunas.

Exemplo da tabela "**funcionarios**" que possui as informações: **id_funcionario**, **funcionario**, **departamento** e **id_maquina**:

id_funcionario	**funcionario**	**departamento**	**id_maquina**
00001	Alberto	Desenvolvimento	00024
00002	Ana	Cibersegurança	00088

Chave primária: Coluna em que cada linha tem uma entrada exclusiva. Serve para identificar a tabela. Não deve ter valores duplicados nem valores nulos ou vazios. Exemplo: (**id_funcionario**).

Chave estrangeira: É uma coluna em uma tabela que é uma chave primária em outra tabela. Diferente das chaves primárias, as chaves estrangeiras podem ter valores vazios e duplicados. Ela permite conectar duas tabelas.

SQL (Linguagem de consulta estruturada): É uma linguagem de programação usada para criar, interagir e solicitar informações de um banco de dados. Os analistas de segurança geralmente usam SQL para encontrar informações relevantes para apoiar decisões relacionada à segurança cibernética.

Consulta (Query): Solicitação de dados de uma tabela de banco de dados ou de uma combinação de tabelas.

Filtragem SQL VS filtragem Linux: A filtragem do Linux se concentra no gerenciamento de arquivos e diretórios em um sistema, enquanto a filtragem do SQL se concentra na manipulação de dados estruturados em bancos de dados.

Para trabalhar com o SQL, é possível acessá-lo de várias interfaces diferentes, como a linha de comando do Linux. Tanto o SQL quanto o Linux permitem que você filtre dados específicos, mas o SQL oferece as vantagens de estruturar os dados e permitir que você junte dados de várias tabelas.

Consultas SQL

SELECT: Indica quais colunas devem ser retornadas.

FROM: Indica qual tabela consultar.

Exemplo de consulta: (**SELECT id_funcionario, id_maquina FROM funcionarios;**) - Irá retornar os dados das colunas "**id_funcionario**" e "**id_maquina**" da tabela "**funcionarios**". É possível selecionar todas as colunas ao inserir um "*****" após o select, inferindo que o usuário está solicitando todos os dados de todas as colunas, exemplo: (**SELECT * FROM funcionarios;**).

ORDER BY: Ordena os registros retornados por um consulta com base em uma coluna ou colunas especificadas. Pode ser feito de forma ascendente ou descendente.

Para usar de forma crescente, basta colocar o **ORDER BY** no final da consulta, seguido pela coluna de referência, exemplo: (**SELECT id_funcionario, departamento, id_maquina FROM funcionarios ORDER BY departamento**) - irá retornar as colunas "**id_funcionario**", "**id_maquina**" e "**departamento**" da tabela funcionarios, mas sequenciadas pela coluna "**departamento**".

Para utilizar de forma decrescente, é só adicionar a palavra DESC após o ORDER BY, no final do script. Exemplo: (**SELECT id_funcionario, departamento, id_maquina FROM funcionarios ORDER BY departamento DESC;**) - Irá ordenar a consulta a partir da tabela "**departamento**" e de forma decrescente.

Para ordenar por várias colunas, basta separá-las por vírgula após o ORDER BY, exemplo: (**ORDER BY departamento, id_maquina;**).

Filtragem: Selecionar dados que correspondam a uma determinada condição.

Operadores: Símbolo ou palavra-chave que representa uma operação.

WHERE: Indica a condição para a filtragem. Após a o where, a condição é representada pelo uso de operadores. Para encontrar dados específicos em uma coluna, é possível utilizar o operador de igual a **(=)** para encontrar apenas os dados com a String determinada. Exemplo: (**WHERE estado = "SP"**) ou de forma completa: (**SELECT * FROM brasil WHERE estado = "SP"**).

É possível usar o sinal de porcentagem **(%)** como um caractere curinga para caracteres não especificados. Exemplo: “**Lago%**” - retornaria todos os registros que começam com “**Lago**”, como “**Lago-sul**” e “**Lago-norte**”. Quando a porcentagem (%) é usada, não é possível utilizar o operador de igual (=), então utiliza-se o **LIKE**. Também é possível limitar o número de caracteres coringa usando “_” para cada caractere, exemplo: (**WHERE quadra LIKE “Q_”**) - Irá retornar Q1, Q2, Q3....

LIKE: Operador usado com WHERE para pesquisar um padrão em uma coluna. Exemplo: (**WHERE animais LIKE “Tubarão%”**) - retornaria apenas os animais que começam com “Tubarão”.

Mais filtros SQL

Tipos de dados comuns:

- **String:** Sequência ordenada de caracteres, podendo ser números, letras ou símbolos. Um exemplo de String seria um nome de usuário.

- **Numeric:** Dados que consistem em números, como uma contagem de tentativas de login.

- **Date and time:** Dados que representam uma data ou hora.

Operadores para dados numéricos e de data e hora:

- **=** (igual)
- **>** (maior que)
- **<** (menor que)
- **<>** (não igual a)
- **>=** (maior ou igual a)
- **<=** (menor ou igual a)

Exemplo: (**SELECT * FROM log_in_attempts WHERE time > “18:00”;**) - Retorna todas as colunas da tabela “**log_in_attempts**” filtradas pelo horário, que deve ser maior do que “**18:00**”.

BETWEEN: Operador que filtra números ou datas dentro de um intervalo geralmente utilizado após o **WHERE**, utiliza a palavra “**AND**” para separar o intervalo de tempo, exemplo: (**SELECT * FROM machines WHERE OS_patch-date BETWEEN “2021-03-01” AND “2021-09-01”;**) - irá retornar uma lista de todas as máquinas atualizadas entre as datas especificadas.

AND: Operador que especifica que ambas as condições devem ser atendidas simultaneamente, exemplo: (**SELECT * FROM machines WHERE operating_system = 'OS 1' AND email_client = 'Email Client 1';**) - irá retornar todas as colunas da tabela "**operating_system**", mas irá apenas retornar os dados que atendem a ambas as condições impostas, no caso, "**OS 1**" e "**Email Client 1**".

OR: Operador que especifica que qualquer condição especificada pode ser atendida, exemplo: (**SELECT * FROM machines WHERE operating_system = 'OS 1' OR operating_system = 'OS 3';**) - irá retornar todas as máquinas que possuam os sistema '**OS 1**' ou '**OS 3**'.

NOT: Operador que nega uma condição, exemplo: (**SELECT * FROM machines WHERE NOT operating_system = 'OS 3';**) - irá retornar todas as máquinas que contenham qualquer sistema operacional, menos o '**OS 3**'.

Junções SQL

Uma forma de especificar para o SQL a coluna que estou me referindo quando ela existe em duas tabelas é adicionar o nome da tabela, seguida de um ponto, antes do nome da coluna, exemplo: (**funcionarios.id_funcionario**) - retorna apenas a coluna "**id_funcionario**" da tabela "**funcionarios**". Outro exemplo: (**maquinas.id_funcionario**) - retorna a coluna "**id_funcionario**" da tabela "maquinas".

INNER JOIN: Retorna linhas correspondentes em uma coluna especificada que existe em mais de uma tabela, exemplo: (**SELECT nome, escritorio, sistema_operacional FROM funcionarios INNER JOIN maquinas ON funcionarios.id_funcionarios = maquinas.id_funcionarios;**) - primeiro eu selecionei todas as colunas que eu gostaria de retornar (de ambas as tabelas) e escolhi a primeira coluna que será mostrada (com o **FROM**), depois eu utilizei o "**INNER JOIN maquinas**" para juntar a tabela "**maquinas**" a tabela "**funcionarios**". Por fim, especifiquei em qual coluna basear o INNER JOIN, no caso, a coluna "**id_funcionarios**".

Tipos de junções externas:

- **LEFT JOIN:** Retorna todos os registros da primeira tabela, mas apenas retorna as linhas da segunda tabela que coincidem com uma coluna especificada.

- **RIGHT JOIN:** Retorna todos os registros da segunda tabela, mas apenas retorna as linhas da primeira tabela que correspondem a uma coluna especificada.
- **FULL OUTER JOIN:** Retorna todos os registros de todas as tabelas.

Funções de agregação: São funções que realizam um cálculo sobre vários pontos de dados e retornam o resultado do cálculo. Os dados reais não são retornados. Para usar uma função de agregação, coloque a palavra-chave para ela após a palavra-chave `SELECT` e, em seguida, entre parênteses, indique a coluna na qual deseja realizar o cálculo.

Tipos de funções de agregação:

- **COUNT:** Retorna um único número que representa o número de linhas retornadas da consulta.
- **AVG:** Retorna um único número que representa a média dos dados numéricos em uma coluna.
- **SUM:** Retorna um único número que representa a soma dos dados numéricos em uma coluna.

Glossário de revisão

Banco de dados: Uma coleção organizada de informações ou dados
Dados de data e hora: Dados que representam uma data e/ou hora
Operador exclusivo: Um operador que não inclui o valor de comparação
Filtragem: Selecionar dados que correspondam a uma determinada condição
Chave estrangeira: Uma coluna em uma tabela que é uma chave primária em outra tabela
Operador de inclusão: Um operador que inclui o valor de comparação
Geração de registros: Um registro A de eventos que ocorrem nos sistemas de uma organização
Dados numéricos: Dados compostos por números
Operador: Um símbolo ou palavra-chave que representa uma operação
Chave primária: Uma coluna em que cada linha tem uma entrada única
Consulta: Uma solicitação de dados de uma tabela do banco de dados ou de uma combinação de tabelas
Banco de dados relacional: Um banco de dados estruturado que contém tabelas relacionadas entre si
Dados String: Dados que consistem em uma sequência ordenada de caracteres
SQL (Linguagem de consulta estruturada): Uma linguagem de programação usada para criar, interagir e solicitar informações de um banco de dados

Sintaxe: As regras que determinam o que é estruturado corretamente em uma linguagem de computador
Caractere curinga: Um caractere especial que pode ser substituído por qualquer outro caractere

FINALIZADO!