Treinamento: FreeBSD – Introdução e Prática

Instrutor: Danilo Perillo Chiacchio

Nessa Aula Vamos Aprender:

- **Trabalhar com Permissões POSIX para arquivos e diretórios.**

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões

- Após aprender o básico sobre gerenciamento de arquivos e diretórios, é de fundamental importância entender as permissões contidas no sistema de arquivos que tem o papel de controlar o que usuários e grupos poderão fazer no sistema. Basicamente, esse controle está sobre o processo de leitura (read), escrita (write) e execução (bit execute, no caso de programas);

- Cada item do sistema (arquivos, diretórios, etc) possuem determinadas permissões dentro do sistema de arquivos;

- Podemos visualizar essas informações através da figura a seguir:

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões

```
root@freebsd01:~ # ls -la
total 8944
drwxr-xr-x   3 root  wheel      512 Jun 21 00:36 .
drwxr-xr-x  18 root  wheel     1024 Jun 23 07:39 ..
-rw-r--r--   2 root  wheel      989 Jun 10 08:26 .cshrc
-rw-------   1 root  wheel     9163 Jun 22 06:14 .history
-rw-r--r--   1 root  wheel      149 Mar 24 23:11 .k5login
-rw-------   1 root  wheel       36 Jun 20 14:08 .lesshst
-rw-r--r--   1 root  wheel      297 Mar 24 23:11 .login
-rw-r--r--   2 root  wheel      254 Mar 24 23:11 .profile
drwxr-xr-x   2 root  wheel      512 Jun  9 23:46 laboratorio
-rwxr--r--   1 root  wheel       62 Jun 10 08:20 teste.sh
-rw-------   1 root  wheel  9072640 Jun 21 00:36 troff.core
```

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões

- Conforme podemos observar na figura, a saída do comando “ls -la” possuem várias informações em formato de colunas. Vamos tomar como base para explicação a linha correspondente ao diretório chamado **“laboratorio”**:

  - **drwxr-xr-x** = Permissoões na qual serão aplicadas ao usuários, grupos de usuários e outros usuários. O primeiro digito (d, nesse exemplo) serve para especificar o tipo de objeto. Nesse contexto “d” significa diretório e “-” trata-se de um arquivo. Os próximos três digitos (rwx) representam as permissões que são atribuídas ao usuário dono do arquivo. Os próximos três digitos (r-x) representam as permissões que são atribuídas ao grupo dono do arquivo. Os últimos três digitos (r-x) representam as permussões que são atribuídas aos demais (outros) usuários. Mais detalhes sobre o significado de cada permissão serão abordados no tópico a seguir;

  - **1** = Número de links que o arquivo/diretório possui;

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões

- **root** = Usuário dono do arquivo/diretório;
- **wheel** = Grupo dono do arquivo/diretório;
- **512** = Tamanho do arquivo/diretório, em bytes;
- **Jun 9 23:46** = Mês/Dia/Hora da última modificação do arquivo/diretório;
- **laboratorio** = Nome do arquivo/diretório.

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- Conforme mencionado anteriormente, as permissões são atribuídas aos objetos (arquivos, diretórios, etc) presentes no sistema de arquivos. Elas (as permissões) podem ser leitura (read), escrita (write) e execução (execute) e podem ser atribuídas de forma conjunta a um objeto do sistema de arquivos;

- As permissões podem ser atribuídas de duas formas, literal ou octal. Em seu formato literal, elas são definidas através de letras. Exemplo:

r = read (leitura);
w = write (escrita);
x = execute (execução).

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- Já no formato octal, as permissões são expressas através de números decimais. Exemplo:

0 = Sem leitura, sem escrita e sem execução;
1 = Somente execução;
2 = Sem leitura, com escrita e sem execução;
3 = Sem leitura, com escrita e com execução;
4 = Somente leitura;
5 = Com leitura e com execução;
6 = Com leitura, com escrita e sem execução;
7 = Permissão total, ou seja, com leitura, com escrita e com execução.

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- A seguir uma correlação do número de cada permissão e seu valor corresponde na forma literal:

Código Númerico	Efeito
0	- - -
1	- - x
2	- w -
3	- w x
4	r - -
5	r – x
6	r w -
7	r w x

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- O gerenciamento das permissões pode ser feito através do comando **“chmod”**. O uso do comando “chmod” possibilita o trabalho com 4 bits, onde cada bit é responsável por um tipo de permissão. Exemplo:

Owner	Group	Other
r w x	r - x	r - x
4+2+1	4+0+1	4+0+1
7	5	5

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- **Nota:** O bit anterior ao bit utilizado para definir as permissões do usuário dono (owner) é utilizado para definir as permissões especiais e geralmente esse bit é omitido.

A seguir, exemplos de como manusear as permissões de arquivos/diretórios:

```
root@freebsd01:~/laboratorio # touch lab01.txt
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # mkdir lab01-dir
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad lab01.txt
-rw-r--r--  1 root  wheel  0 Jun 28 23:07 lab01.txt
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad lab01-dir/
drwxr-xr-x  2 root  wheel  512 Jun 28 23:07 lab01-dir/
```

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- Conforme demonstra a figura anterior, o arquivo “lab01.txt” e o diretório “lab01-dir” foram criados;
- Através do comando “ls” podemos visualizar quais permissões foram atribuídas ao arquivo e ao diretório;
- O arquivo “lab01.txt” possui as permissões 644 (rw-r - -r - -) e o diretório “lab01-dir” possui as permissões 755 (rwxr-xr-x). É importante ressaltar que essas permissões são as padrões para o sistema e são atribuídas com base na chamada de sistema “umask”.

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- **Nota:** Umask é o acrônimo para “User Mask”. Trata-se de uma chamada de sistema (System Call) utilizada para especificar quais permissões serão utilizadas para criação de arquivos e diretórios dentro do sistema. Em outras palavras, a umask define quais permissões serão “revogadas” do usuário dono, grupo dono ou outros usuários. O valor padrão de umask para o FreeBSD é “022”, onde:

  0 = Nenhuma permissão do usuário dono (user) será revogada, ou seja, acesso “full” concedido;

  2 = Permissões de escrita e execução serão revogadas para o grupo dono;

  2 = Permissões de escrita e execução serão revogadas para os demais usuários (outros).

- **O valor para umask pode ser alterado no arquivo de configuração /etc/login.conf**

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- Alterando a permissão do arquivo “lab01.txt” para somente leitura para todos:

```
root@freebsd01:~/laboratorio # ls -lad lab01.txt
-rw-r--r--  1 root  wheel  0 Jun 28 23:07 lab01.txt
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # chmod 444 lab01.txt
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad lab01.txt
-r--r--r--  1 root  wheel  0 Jun 28 23:07 lab01.txt
```

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- Alterando a permissão do diretório “lab01-dir” para que somente o usuário dono do diretório tenha acesso “full” ao diretório:

```
root@freebsd01:~/laboratorio # ls -lad lab01-dir/
drwxr-xr-x  2 root  wheel  512 Jun 28 23:07 lab01-dir/
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # chmod u=rwx,g=,o= lab01-dir/
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad lab01-dir/
drwx------  2 root  wheel  512 Jun 28 23:07 lab01-dir/
```

Tópico 4: Administração do Sistema Operacional

Entendendo as Permissões (Read, Write and eXecute)

- **Importante: É importante ressaltar que a maioria das permissões pode ser atribuída tanto para arquivos quanto para diretórios.**