

Chmod + dicas

Autor: sc4rf4c3 <bhangelo@hotmail.com>

Data: 25/05/2006

Chmod + dicas

By sc4rf4c3@post.com

Usando e entendendo permissões no Linux

Resolvi montar essa pequena dica, sei que já existem várias, porém vou colocar algumas dicas do *chmod*, porque onde trabalho fizeram o favor de setar um *chmod -R 777* na pasta /, aí o servidor ficou com permissão errada para todos os arquivos, tive que setar as permissões arquivo por arquivo comparando com outra máquina com mesmo sistema Slackware, pois não podíamos desligar o servidor.

Exemplos de permissões nos arquivos:

```
drwxr-xr-x 19 root  root    4096 Jan  5 16:56 ../
-rwxr-xr-x  1 root  root    38812 Mar  9 2003 MAKEDEV*
-rw-r--r--  1 root  root    1162 Mar  9 2003 README.MAKEDEV
lrwxrwxrwx  1 root  root      4 Oct 15 2003 X0R -> null
crw-rw----  1 root  root    10, 175 Mar 25 2001 agpgart
crw-rw----  1 root  root      1, 10 Mar 25 2001 aio
drwxr-xr-x  2 root  root    4096 Jun 11 2002 amiraid/
crw-r--r--  1 root  root    10, 134 Jun  7 1996 apm_bios
crw-rw----  1 root  root      36,  8 Mar 25 2001 arpd
drwxr-xr-x  2 root  root    4096 May 19 2002 ataraid/
crw-rw----  1 root  sys     10,  3 Jul 17 1994 atibm
crw-rw----  1 root  root    10, 186 Mar 25 2001 atomicps
crw-rw--w-  1 root  sys     14,  4 Jul 18 1994 audio
crw-rw--w-  1 root  sys     14, 20 Jul 18 1994 audio1
brw-r-----  1 root  disk    29,  0 Feb 15 1995 aztcd
crw-r--r--  1 root  root    10, 128 May 24 1996 beep
drwxr-xr-x  2 root  root    36864 Jun  3 2002 cciss/
lrwxrwxrwx  1 root  root      8 Oct 15 2003 cdrom -> /dev/hdd
brw-r-----  1 root  disk    24,  0 Jul 18 1994 cdu535
crw-rw----  1 root  root     67,  0 Mar 25 2001 cfs0
```

O primeiro caractere da string indica o tipo de arquivo. Se for "d" representa um diretório, se for "-" equivale a um arquivo.

- d: diretório
- b: arquivo de bloco
- c: arquivo especial de caractere

- p: canal
- s: socket
- -: arquivo normal

É necessário ter um certo cuidado com as permissões. Por exemplo, do que adianta o usuário ter permissão de gravação se ele não tem permissão de leitura habilitada? Ele poderá ler o arquivo para poder modificá-lo? Não! De certo, isso tem utilidade em arquivos de log.

Fazendo associação com as letras r, w, x e o caractere -, vamos entender cada uma:

- r: significa permissão de leitura (read);
- w: significa permissão de gravação (write);
- x: significa permissão de execução (execution);
- -: significa permissão desabilitada.

Atenção:

- ---: nenhuma permissão;
- r--: permissão de leitura;
- r-x: leitura e execução;
- rw-: leitura e gravação;
- rwx: leitura, gravação e execução.

Vamos supor que o arquivo estava assim:

```
crwxrwxrwx 1 root disk 37, 128 Dec 11 1995 nht0
```

Dessa forma ele está errado. Sua forma correta é:

```
crw-r----- 1 root disk 37, 128 Dec 11 1995 nht0
```

Usei o comando:

```
# chmod 640 nht0
```

Alterando as permissões de crwxrwxrwx para crw-r-----. Note que sempre as letras iniciais não mudam, ok?

O comando chmod -R seta as permissões no diretório e nos arquivos dentro dele, cuidado com isso, foi o que fizeram aqui no meu servidor.

Use assim:

```
$ chmod [número] [nome do arquivo]
```

Exemplo:

chmod 755 apache

Comando Número Permissão

chmod	000	-----
chmod	400	r-----
chmod	444	r--r--r--
chmod	600	rw-----
chmod	620	-rw--w----
chmod	640	-rw-r-----
chmod	644	rw-r--r--
chmod	645	-rw-r--r-x
chmod	646	-rw-r--rw-
chmod	650	-rw-r-x---
chmod	660	-rw-rw----
chmod	661	-rw-rw---x
chmod	662	-rw-rw--w-
chmod	663	-rw-rw--wx
chmod	664	-rw-rw-r--
chmod	666	rw-rw-r--
chmod	700	rwX-----
chmod	750	rwXr-x---
chmod	755	rwXr-xr-x
chmod	777	rwXrwXrwX
chmod	4711	-rws--x--x

Outro detalhe, os símbolos:

- u: usuário
- g: grupo
- o: outros
- a: todos

Para mostrar como essa combinação é feita, vamos supor que você deseje adicionar permissão de gravação no arquivo teste.txt para um usuário. Então o comando a ser digitado será:

\$ chmod u+w teste.txt

O "u" indica que a permissão será dada a um usuário. O sinal de adição (+) indica que está sendo adicionada a permissão e "w" indica que a permissão que está sendo dada é de gravação.

Caso você queira dar permissões de leitura e execução ao seu grupo, o comando será:

\$ chmod g+rw teste.txt

vamos supor que o arquivo teste.old deverá estar com todas as permissões disponíveis para o grupo. Podemos usar então:

\$ chmod g=rwx teste.txt

Usar o chmod com valores numéricos é algo bastante prático. Em vez de usar letras como símbolos para cada permissão, usam-se números. Se determinada permissão é habilitada, atribui-se valor 1, caso contrário, atribui-se valor 0. Sendo assim, a string de permissões r-xr----- na forma numérica fica 101100000. Essa combinação de 1 e 0 é um número binário.

Mas temos ainda que acrescentar a forma decimal (ou seja, números de 0 a 9). Para isso, observe a tabela abaixo:

Permissão Binário Decimal		
---	000	0
--x	001	1
-w-	010	2
-wx	011	3
r--	100	4
r-x	101	5
rw-	110	6
rwX	111	7

Pra quem está começando agora vai ser um pouco complicado entender de uma vez isto, porém se você possui uma máquina com *Linux*, treine.

<http://www.vivaolinux.com.br/dica/Chmod-+-dicas>

[Voltar para o site](#)