Treinamento: FreeBSD – Introdução e Prática

Instrutor: Danilo Perillo Chiacchio

Nessa Aula Vamos Aprender:

- **Entendendo as Permissões Especiais setuid, setgid e sticky bit.**

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões Especiais

- Vamos iniciar falando sobre a permissão especial sticky bit. Essa permissão permite que somente o proprietário do diretório e/ou arquivo possa removê-lo do sistema operacional;

- Essa permissão especial é default para o diretório /tmp:

```
root@freebsd01:/ # ls -lad /tmp/
drwxrwxrwt  8 root  wheel  512 Jun 28 17:27 /tmp/
```

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões Especiais

- Conforme demonstra a figura anterior, a permissão especial sticky bit é representada pela letra “t”. Essa permissão (sitcky bit) também pode ser atribuída com o comando “chmod”. A seguir, vamos criar um diretório chamado “lab02-sticky” dentro de /root/laboratorio e definir a permissão especial sticky bit:

```
root@freebsd01:~/laboratorio # ls -lad lab02-sticky/
drwxr-xr-x  2 root  wheel  512 Jun 29 00:04 lab02-sticky/
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # chmod 1755 lab02-sticky/
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad lab02-sticky/
drwxr-xr-t  2 root  wheel  512 Jun 29 00:04 lab02-sticky/
```

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões Especiais

- Isso significa que todos os arquivos/diretórios dentro de “lab02-sticky” somente poderão ser excluídos pelo seu respectivo usuário dono do arquivo/diretório. Caso algum usuário tente excluir algum arquivo que não seja de sua propriedade um erro será retornado para o mesmo;

- As permissões setuid e setgid (s-bit) são, basicamente, permissões especiais aplicadas para usuários e grupos de usuários com objetivo de conceder privilégios adicionais para usuários e grupos. A permissão s-bit suid bit (setuid) quando definida, permite que qualquer usuário execute determinado binário ou programa como se fosse o usuário dono;

- Já a permissão sgid bit (setgid) faz papel similar ao suid bit, porém com reflexo para grupo dono do arquivo/diretório.

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões Especiais

- Por exemplo:

O sendmail é um programa responsável pelo recebimento, transmissão e armazenamento de mensagens de correio eletrônico. Torna-se necessário para ele der executado com o s-bit ao usuário (setuid). Como o usuário em questão é o root, o sendmail será executado sempre com as credenciais desse usuário.

- Essas permissões (setuid e setgid) também podem ser gerenciadas através do comando “chmod”. Exemplos:

```
root@freebsd01:~/laboratorio # ls -lad script01.sh
-rw-r--r--  1 root  wheel  0 Jun 29 00:16 script01.sh
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # chmod 4644 script01.sh
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad script01.sh
-rwSr--r--  1 root  wheel  0 Jun 29 00:16 script01.sh
```

Tópico 4: Administração do Sistema Operacional

Trabalhando com Permissões Especiais

- Conforme demonstra a figura anterior, foi definido o s-bit para o usuário/proprietário do script “script01.sh”. A letra que representa o s-bit é o “s”. Quando aparece no formato “S”, significa que o usuário não possui a permissão de execute naquela posição. Para definir o s-bit para o grupo dono do script (setgid):

```
root@freebsd01:~/laboratorio # ls -lad script01.sh
-rwSr--r--  1 root  wheel  0 Jun 29 00:16 script01.sh
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # chmod 2644 script01.sh
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio #
root@freebsd01:~/laboratorio # ls -lad script01.sh
-rw-r-Sr--  1 root  wheel  0 Jun 29 00:16 script01.sh
```

Nota: Por razões de segurança, é recomendado que a permissão especial s-bit seja utilizada apenas com arquivos binários e não com arquivos de shell script, por exemplo.