SEG12 - Semana 1 - Sessão 6

Francisco Marcelo, Marcelo Karam e Felipe Scarel

06-08-2018

Segurança básica e procedimentos operacionais

As atividades desta sessão serão realizadas na máquina virtual *Client_Linux*.

1) Identificando senhas fracas

Uma das formas de verificar se o seu sistema atende às recomendações básicas de segurança é utilizar os programas "quebradores" de senha, ou *password crackers*. Neste exercício, utilizaremos um desses programas para mostrar seu funcionamento.

1. Obtenha e instale o *password cracker* John the Ripper, ou simplesmente `john`.

```
# apt-get install john
```

2. Crie o arquivo `/root/dicionario.txt` com uma lista de senhas. Caso considere necessário, acrescente palavras que julgue impróprias para uso em senhas. Por exemplo:

```
123456
1234
rnpesr
senha
abacate
```

3. Rode o *password cracker* com o comando `# john -wordlist=/root/dicionario.txt -rules /etc/shadow`.

```
# john -wordlist=/root/dicionario.txt -rules /etc/shadow
Created directory: /root/.john
Loaded 5 password hashes with 5 different salts (crypt, generic crypt(3) [?/64])
Press 'q' or Ctrl-C to abort, almost any other key for status
123456           (aluno2)
senha            (marcelo)
abacate          (aluno3)
rnpesr           (root)
rnpesr           (aluno)
5g 0:00:00:01 100% 3.676g/s 70.58p/s 352.9c/s 352.9C/s 123456..Abacate9
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

4. Veja o resultado da verificação com o comando `# john -show /etc/shadow`.

```
# john -show /etc/shadow
root:rnpesr:16842:0:99999:7:::
aluno:rnpesr:16842:0:99999:7:::
aluno2:123456:17752:0:99999:7:::
marcelo:senha:17752:0:99999:7:::
aluno3:abacate:17752:0:99999:7:::

5 password hashes cracked, 0 left
```

2) Descobrindo a funcionalidade do bit SGID em diretórios

A utilidade do SUID e SGID foi vista desde a sessão de aprendizagem 1. Execute a sequência de comandos e depois responda as seguintes perguntas:

1. Crie o grupo `corp` e defina-o como grupo secundário do seu usuário.

```
# groupadd corp
# usermod -a -G corp aluno
# groups aluno
aluno : aluno cdrom floppy sudo audio dip video plugdev netdev bluetooth corp
```

2. Entre no sistema a partir da sua conta e crie um diretório chamado `dir_corp`.

```
$ mkdir dir_corp
$ ls
dir_corp
```

3. Verifique a qual grupo pertence o diretório criado no passo acima. Modifique-o para que passe a pertencer ao grupo `corp` e mude a sua permissão para `2755`.

```
$ chgrp corp ~/dir_corp/
$ chmod 2755 ~/dir_corp/
$ ls -ld dir_corp/
drwxr-sr-x 2 aluno corp 4096 Ago  9 19:15 dir_corp/
```

4. Crie, no seu diretório *home* um arquivo chamado `arq1`. Em seguida, mude para o diretório criado no segundo item e crie um arquivo chamado `arq2`.

```
$ pwd
/home/aluno
$ touch arq1
$ touch dir_corp/arq2
```

5. Verifique os grupos aos quais pertencem os arquivos criados no item anterior. Você saberia explicar por que os arquivos pertencem a grupos distintos, embora tenham sido criados pelo mesmo usuário?

```
$ ls -ld arq1
-rw-r--r-- 1 aluno aluno 0 Ago  9 19:19 arq1
$ ls -ld dir_corp/arq2
-rw-r--r-- 1 aluno corp 0 Ago  9 19:19 dir_corp/arq2
```

   O arquivo criado no diretório `/home/aluno/dir_corp/` possui o mesmo grupo dono de seu diretório-pai, pois o mesmo está com o bit SGID definido — isso faz com que qualquer arquivo criado dentro dele tenha o mesmo grupo dono que o próprio diretório, independente do usuário que o tenha criado. Já o arquivo criado no diretório `/home/aluno/` tem o mesmo grupo primário do usuário que o criou, já que este diretório não tem o bit SGID definido.

6. Quais as vantagens desse esquema?

   Esse recurso é útil em diretórios compartilhados, nos quais diversos usuários criam arquivos que precisam ter permissão de escrita e/ou leitura para todos os usuários do grupo do diretório.

3) Obtendo informações sobre os recursos computacionais

1. Vimos, no texto teórico, que uma das importantes funções de um administrador de sistemas é acompanhar o uso dos recursos computacionais de sua instituição. Discuta com o seu colega quais comandos vistos em todo o módulo podem auxiliar na coleta desse tipo de informação.

   Diversos comandos podem ser utilizados para verificar o uso dos recursos computacionais, dentre os quais podemos destacar: `df`, `du`, `ps`, `top`, `htop`, `free`, `vmstat`, `iostat`, `lsof`, etc.

4) Controlando os recursos dos usuários

Um dos grandes desafios de um administrador de sistema, nos tempos atuais, é controlar a ocupação do espaço em disco do seu sistema — aplicações do tipo P2P (*peer-to-peer*), por exemplo, são consumidoras vorazes desse tipo de recurso.

1. Que medidas podem ser tomadas para controlar a ocupação de disco de forma automática?

   A instalação e configuração de *quotas* de disco para usuários é uma excelente maneira de implementar controles nesse sentido.