ACL's Linux POSIX

João Paulo Ozório
Leandro Pereira
Paulo César Ferreira

O que são ACL's

- O sistema de permissões do Linux é considerado muito seguro e simples de utilizar.
- Cada arquivo ou diretório pode pertencer a apenas um usuário e um grupo.
- Com o crescimento do Linux no segmento de servidor de arquivos, pode existir a necessidade do uso de permissões um pouco mais especificas.
- As ACL's nos permite garantir direitos a mais de um usuário/grupo em um mesmo arquivo, não usando necessariamente os direitos gerais (permissão para outros usuários).
- ACL's POSIX interagem de forma complexa com as permissões nativas do Unix.

Permissões UNIX

- O Unix tem uma lista de usuários /etc/passwd e uma lista de grupos /etc/group;
- Cada usuário pertence a um grupo;
- Dentro de cada lista, não existe qualquer hierarquia entre usuários ou entre grupos;
- Não há grupos de grupos, nem usuários-líderes por exemplo;
- Normalmente, é criado um grupo primário para cada usuário Unix;
- Cada usuário ainda pode ser incluído em outros grupos, que serão seus grupos secundários.

ACL's POSIX

- A mais importante extensão das ACLs POSIX é a atribuição de usuários e grupos adicionais.

- As ACLs POSIX estendem (não substituem) as permissões tradicionais. Cada arquivo/diretório continua tendo o usuário-dono e o grupo-dono 'originais'.

- Cada arquivo e diretório têm sua própria ACL. A transmissão de permissões é possível, mas tem de ser explicitamente solicitada, e atuará apenas sobre arquivos criados dali para diante.

Permissões UNIX

- Usuário-dono do arquivo
- Grupo-dono do arquivo
- Permissões do usuário-dono sobre o arquivo
- Permissões do grupo-dono sobre o arquivo
- Permissões dos "outros" usuários sobre o arquivo
- Permissões especiais

r	permissão de leitura (bit mais significativo)
w	permissão de escrita
x	permissão de execução (para programas e scripts)

ACL's POSIX

- As ACLs POSIX são suportadas nativamente no Linux no kernel 2.6. As distribuições mais novas, que já trazem essa versão de kernel, têm suporte direto a ACL.

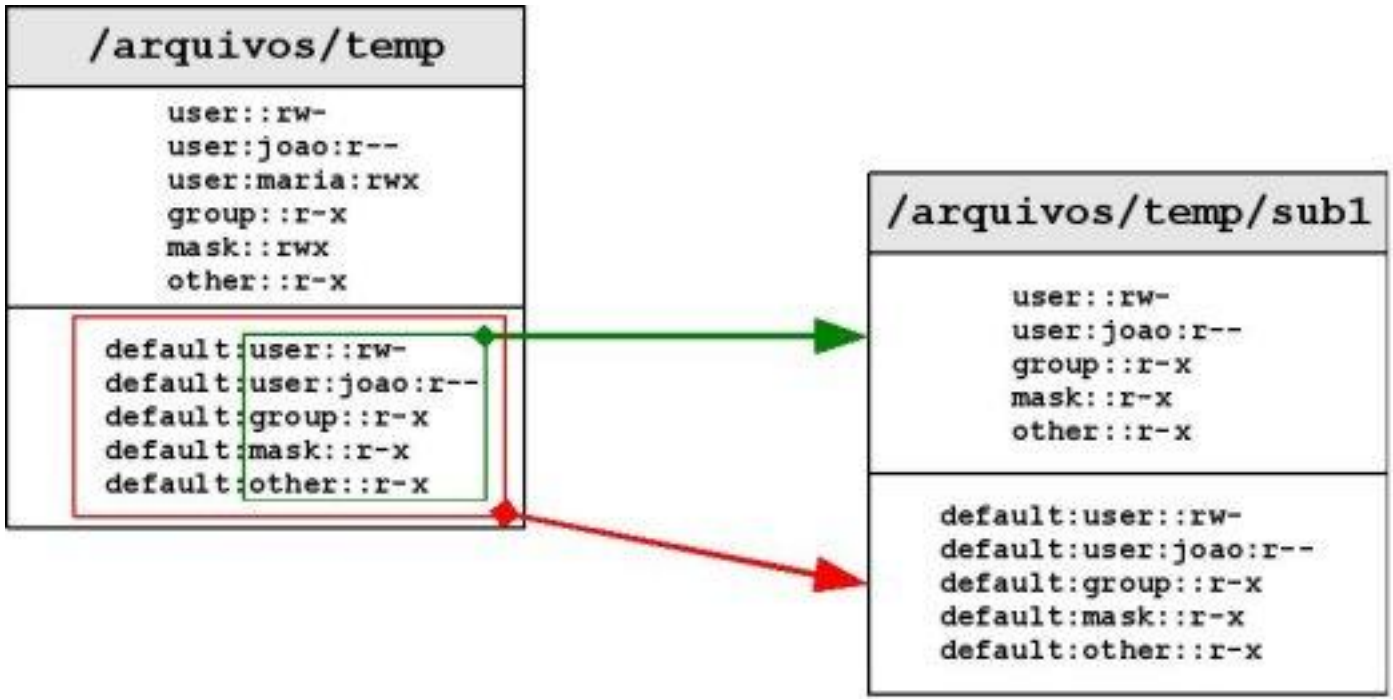
- Como outros programas, além do kernel, precisam ser atualizados para o suporte a ACLs, aconselha-se a usar distribuições com suporte a ACL ao invés de tentar incluir esse suporte manualmente.

ACL's Padrões

- Em diretórios, podemos usar um recuso conhecido como **ACLs Padrões**, neste caso teremos que:

- **Diretórios** herdam as ACLs padrões configuradas no diretório imediatamente acima;
- **Arquivos** herdam as ACLs configuradas como ACLs padrões no diretório imediatamente acima;

ACL's Padrões



ACL's POSIX

- Quantidade máxima de ACLs suportada pelos sistemas de arquivos.

Sistema de Arquivos	Quantidade de ACL
XFS	25
Ext2, Ext3	32
ReiserFS, JFS	8191

Tempo de acesso

Sistema de Arquivos	Sem ACL	Com ACL
Ext2	9	1743
Ext3	10	3804
ReiserFS	9	6165
XFS-256	14	7531
XFS-512	14	14
JFS	13	13

Algoritmo

- Se o usuário é o dono e tem permissões, o acesso é permitido;
- Se a ACL contém um usuário e ele tem as permissões necessárias, ele irá para a máscara de entrada (explicado abaixo).
  - Caso contrário, o acesso é negado;
- Se o usuário faz parte do grupo dono do arquivo, ou se o usuário faz parte de um grupo que contém uma entrada ACL, então:
  - Se as entradas contém as permissões necessárias, ele irá para a máscara de entrada;
  - Caso contrário, o acesso é negado.

Algoritmo

- Se a solicitação não se encaixar nas permissões acima, então:
  - Se as permissões para outros usuários tiver as permissões necessárias, o acesso é permitido;
  - Caso contrário, o acesso é negado.

- Checando a máscara de acesso:
  - Se a máscara de acesso contém as permissões necessárias, o acesso é permitido;
  - Caso contrário, o acesso é negado.

Atributos Estendidos

- O Linux mantém uma estrutura de dados chamada inode de tamanho fixo com as informações dos arquivos tais como tais como o dono, permissões e sua localização.
- Cada inode tem um atributo i_file_acl que se não é zero, aponta para um bloco do sistema de arquivos que contém todos atributos estendidos daquele inode

Suporte a ACLs no Kernel

- A maioria das distribuições atuais tornam fácil a utilização de ACLs POSIX.
- Devido a presença das ferramentas de manipulação de ACLs como parte do sistema padrão.
- Comando para verificar a presença de ACLs POSIX em seu kernel:

```
[root@localhost:~]# cat /boot/config-$(uname –r) | grep _ACL
CONFIG_ EXT2_FS_POSIX_ACL=y
CONFIG_ EXT3_FS_POSIX_ACL=y
CONFIG_ EXT4_FS_POSIX_ACL=y
CONFIG_REISERFS_FS_POSIX_ACL=y
CONFIG_JFS_POSIX_ACL=Y
CONFIG_XFS_POSIX_ACL=y
CONFIG_[OCFS2, BTRFS, TMPFS]_POSIX_ACL=y
```

Ferramentas de manipulação de ACLs POSIX

- As ferramentas de manipulção de ACLs POSIX são **setfacl** e **getfacl**, presentes nas distribuições modernas.

```
[root@localhost:~]# setfacl –version
setfacl 2.2.49
```

```
[root@localhost:~]# gettfacl –version
getfacl 2.2.49
```

```
Instalando no Ubuntu10.04:
[root@localhost:~]# apt-get install acl
```

Montagem do sistema de arquivos

- Apesar de seu kernel apresentar suporte a ACLs, seu sistema de arquivos deve ser montado para habilitar esse recurso.
- Para montar:

/arquivos = partição/sistema de arquivos
[root@localhost:~]# mount -o remount,acl /arquivos

- Caso seu computador possua partição saparada para o diretório **/arquivos**.

ACLs Mínimas

- Usando getfacl.

```
#Foi criado um arquivo temp para teste.

[root@localhost:~]# cd /arquivos/
[root@localhost:~]# getfacl temp
# file: temp          informações sobre o diretório(nome)
# owner: root         informações sobre o diretório(dono)
# group: root         informações sobre o diretório(grupo)
user::rwx             permissão para usuário
group::r-x            permissão para grupo
other::r-x            permissão para outros
```

ACLs Mínimas

- Usando setfacl para manipular as permissões tradicionais

#Foi criado um arquivo temp no diretório **/arquivos** para teste.

[root@localhost:~]# cd /arquivos/
[root@localhost:~]# setfacl –m user::rw temp
[root@localhost:~]# ls –ld temp para ver as permissões
drw-r-xr-x 2 root root 1024 2010-12-10 08:35 temp/

ACLs Mínimas

- Apresentam o conjunto mínimo de permissões exigido por um sistema de arquivos UNIX.

```
#Foi criado um arquivo temp para teste.

[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw
group::r-x
other::r-x
```

ACLs Extendidas

- Modificando as ACLs de um diretório ou arquivo, por meio da ferramenta **setfacl**, estamos criando ACLs Extendidas.

```
#Foi criado um arquivo temp para teste.

[root@localhost:~]# setfacl –m user:paulo:r-x temp
[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw-
user:paulo:r-x    permissão para ler e executar
group::r-x
mask::r-x
other::r-x
```

Estrutura

- **user:: :** ACL **do dono**
- **group:: :** ACL **do grupo**
- **other:: :** ACL **de outros**
- **mask:: :** ACL **de máscara**
- **user:paulo: :** ACL **de usuário (usuário paulo neste caso)**
- **group:inf: :** ACL **de grupo (grupo inf neste caso)**

- **Diferença:**
- ACL do grupo = grupo do arquivo;
- ACL de grupo = grupo manualmente especificado;

ACLs de máscara

- Ao criar uma ACL de **user** ou **group**, a ACL de máscara é automaticamente ajustada para permitir as permissões descritas nas ACLs[**user** e **group**].

```
#Exemplo.
user : paulo : r-x
mask: : r-x

#Criando uma nova ACL.
[root@localhost:~]# setfacl –m user:joao:rwx temp
[root@localhost:~]# getfacl temp
...
user:paulo:r-x
user:joao:rwx
Group::r-x
mask::rwx .
...
```

ACLs de máscara

- Função:
- Configurar o nível de privilégio que as ACLs**[user** e **group]** terão sobre o diretório.

ACLs de máscara

- Para incluir ACLs, sem modificar a ACL de máscara(**-n**)

```
#Incluindo ACLs com -n
[root@localhost:~]# setfacl –m user:paulo :r--: temp
[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw-
user:paulo:r--
group::r-x
group::r-x
other::r-x
```

ACLs de máscara

- Para incluir ACLs, sem modificar a ACL de máscara(**-n**)

```
#Incluindo ACLs com -n
[root@localhost:~]# setfacl -n -m user:joao :rwx: temp
[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw-
user:paulo:r--
user:joao:rwx       #effective:r-x
group::r-x
group::r-x
other::r-x
```

ACLs de máscara

- Setando máscara:

```
#Setando máscara
[root@localhost:~]# setfacl –m mask::r--temp
[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw-
user:paulo:r-x      #effective:r--
group::r-x           #effective:r--
mask::r---
other::r-x
```

Excluindo ACL

- Para excluir uma ACL:

```
#Excluindo ACL
[root@localhost:~]# setfacl –x user:joao,user:paulo:: temp
[root@localhost:~]# getfacl temp
# file: temp
# owner: root
# group: root
user::rw-
group::r-x
other::r-x
```

Referências

- http://wiki.sintectus.com/bin/view/GrupoLinux/ArtigoACLPosix
- http://epx.com.br/artigos/aclposix.php
- http://www.suse.de/~agruen/acl/linux-acls/online/#tab:acl_entries
- http://br-linux.org/artigos/dicas_acl.htm