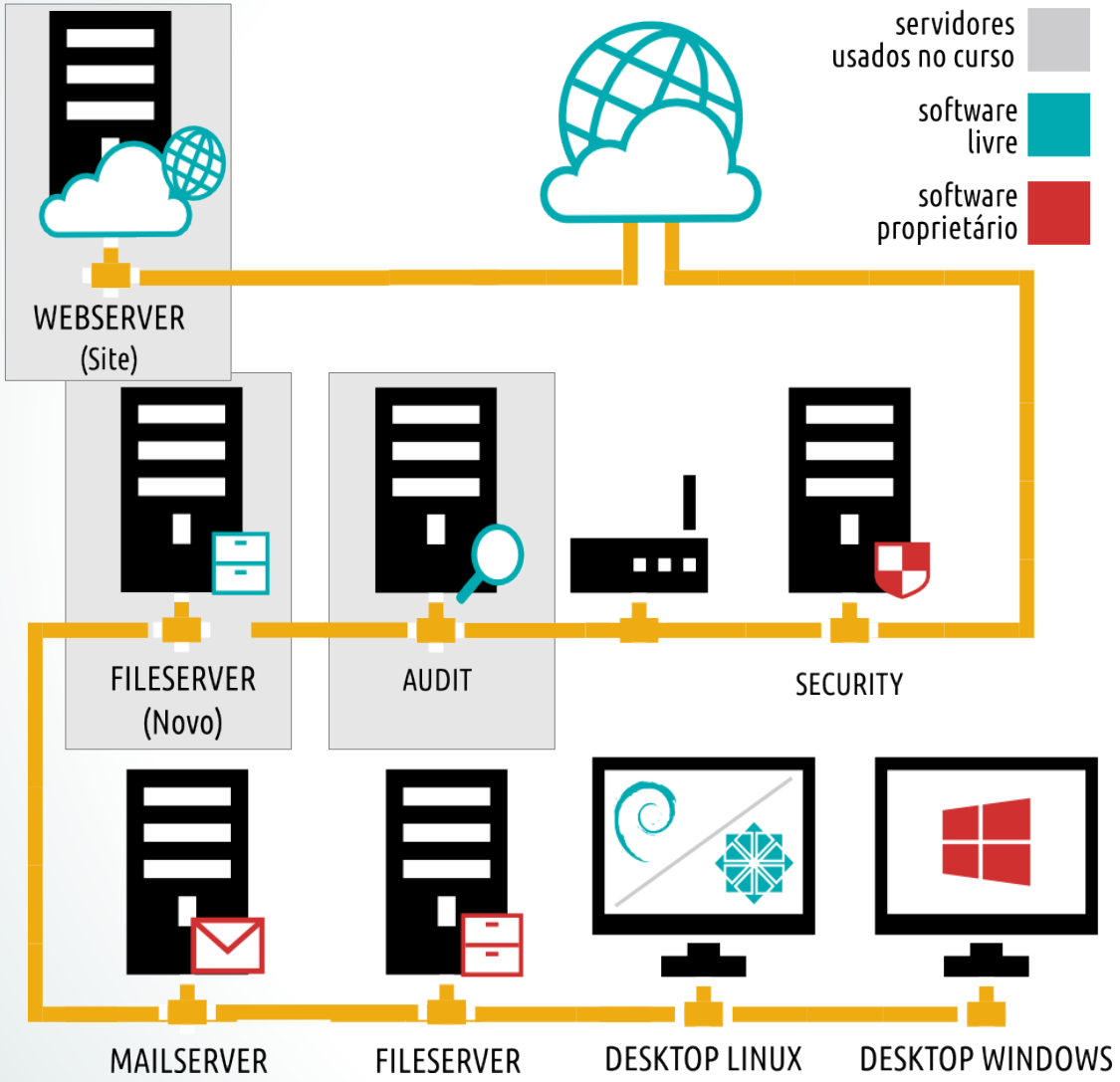

4LINUX
FREE SOFTWARE SOLUTIONS

Só na 4Linux você aprende MUITO MAIS!

Implementando Controle de Acesso ao Sistema com PAM

IT EXPERIENCE



Nesta Aula:

- FileServer (Samba4) – Local

  Acesso pelo VirtualBox

  SO: Debian Linux

Implementando Controle de Acesso ao Sistema com PAM

Objetivos:

- Introdução Teórica – PAM
  - Tipos, gerenciadores e controles do PAM
- Implementação Prática do PAM
  - Módulos: nologin, securetty, time, whell e limits

Implementando Controle de Acesso ao Sistema com PAM

Introdução Teórica – PAM

- Historicamente, cada programa possuía a sua própria maneira de autenticar usuários.
- Atualmente, vários sistemas utilizam um mecanismo de autenticação centralizado chamado Módulos de Autenticação Plugáveis **(Pluggable Authentication Modules - PAM).**
- Seu desenvolvimento inicial aconteceu em 1996 pela Sun Microsystems e atualmente é suportado nos sistemas **operacionais AIX, HP-UX, Solaris, Linux, FreeBSD, Mac OS X e NetBSD.**

Implementando Controle de Acesso ao Sistema com PAM

Introdução Teórica – Arquivos do PAM

- As bibliotecas PAM são configuradas no arquivo **/etc/pam.conf** ou nos arquivos dentro do diretório **/etc/pam.d/**. Vale ressaltar que não é possível utilizar as duas formas de configuração do LinuxPAM. Ou um ou outro, não os dois ao mesmo tempo.

- Os arquivos de configuração utilizam módulos normalmente localizados no diretório **/lib/security/** ou **/lib64/security/** e se comportam como objetos carregáveis dinamicamente.

Implementando Controle de Acesso ao Sistema com PAM

Introdução Teórica – Gerenciadores do PAM

➢O PAM trabalha com gerenciadores e controles, e cada tipo de módulo provê uma funcionalidade diferente dentro do sistema. Vamos comentar primeiro os gerenciadores:

- → **account**
- → **auth (authentication)**
- → **password**
- → **session**

Implementando Controle de Acesso ao Sistema com PAM

Introdução Teórica – Controles do PAM

Além dos gerenciadores, existem também os controles:

→ **required**: Checa a existência do módulo solicitado. Caso esse módulo falhe, somente depois de verificar se todos os módulos do mesmo tipo estão disponíveis é que o usuário será avisado;

→ **requisite**: Checa a existência do módulo solicitado e avisa o usuário imediatamente caso este módulo falhe.

Implementando Controle de Acesso ao Sistema com PAM

Introdução Teórica – Controles do PAM

→ **sufficient**: Somente a verificação do módulo é suficiente para a autenticação, desde que nenhum módulo marcado como required falhe.

→ **optional**: O sucesso ou a falha deste módulo não interfere no processo de autenticação.

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Maquina Datacenter

1 – Para começar verifique quais módulos do PAM estão instalados:

```
1# ls /lib/security
```

2 – Para descobrir se sua aplicação tem suporte no PAM, basta utilizar o comando "ldd".

```
2# ldd /bin/login | grep libpam
```

3 – Podemos verificar também quais serviços trabalham com o PAM

```
3# ls /etc/pam.d/
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo nologin

1 – Vamos fazer um teste simples que serve para bloquear usuários comuns. Abra o arquivo "/etc/pam.d/login" e visualize o módulo "pam_nologin.so" na linha 36:

```
1# vim +36 /etc/pam.d/login
```

```
36 auth       requisite  pam_nologin.so
```

2 – Para esse "módulo" funcionar ele necessita que o arquivo "nologin" esteja criado dentro do diretório /etc. Execute:

```
2# touch /etc/nologin
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo nologin

3 – Agora tente logar diretamente no terminal (Não utilize o comando su) com um dos usuários criados na Aula de Administração de Usuários:

< O login deverá ser bloqueado sem qualquer mensagem de retorno >

4 – Remova o arquivo para que possamos testar outros módulos do PAM:

```
1# rm /etc/nologin
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo securetty

1 – Vamos bloquear o acesso de root ao terminal do servidor sem o uso do PAM, apenas editando o arquivo /etc/securetty:

```
1# vim +31 /etc/securetty
```

```
31 #tty3
```

2 – Tente logar-se com o usuário root no TTY1 e em seguida no TTY3.

< O login deverá ser bloqueado SOMENTE na TTY3 >

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo time

1 – Para usar o PAM para gerenciar o login do root, vamos abrir o arquivo /etc/pam.d/login e descomentar a linha 68:

```
1# vim +68 /etc/pam.d/login
```

```
68 account   requisite  pam_time.so
```

2 – Vamos negar o login do root no terminal, editando o arquivo /etc/security/time.conf" e acrescentando na última linha:

```
1# vim /etc/security/time.conf
```

```
login;*;root;!Al0000-2400
```

< Tente logar em qualquer terminal >

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo time

Com relação as opções de acesso utilizadas no slide anterior:

login → Serviço que irá ser controlado

***** → Terminal

root → Usuário

Al0000-2400 → Dias e horários de filtragem.

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo time

O dia da semana é especificado em duas letras em inglês:

Mo → Segunda-Feira

Tu → Terça-Feira

We → Quarta-Feira

Th → Quinta-Feira

Fr → Sexta-Feira

Su → Sábado

É possível utilizar também os seguintes curingas:

wd → Finais de semana (somente sábados e domingos);

wk → Semana, Segunda-Feira a Sexta-Feira;

Al → Todos os dias;

! → Especifica uma exceção, em geral utilizado para negar uma regra.

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo time

3 – Para negar o acesso ao root via ssh abra arquivo de configuração do ssh /etc/ssh/sshd_config e inclua na linha 9:

```
1# vim +9 /etc/pam.d/sshd
    account    required    pam_time.so
```

4 – Completando a configuração abra o arquivo /etc/security/time.conf e adicione no final:

```
2# vim /etc/security/time.conf
    sshd;*;root;!Al0000-2400    < Tente logar com o root via ssh >
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo time

5 – Vamos restringir o acesso do usuário Harry que é analista ao horário comercial 08h as 18h de segunda a sexta:

```
1# vim /etc/security/time.conf
 sshd;*;harry.rosemberg|voce.sobrenome;!wd0000-2400
 login;*;harry.rosemberg|voce.sobrenome;!wd0000-2400
```

5 – Caso não seja um curso de final de semana alterar a data da máquina:

```
2# date MMDDYY
```

< Altere a data do sistema para um domingo e tente fazer um acesso >

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo whell

O módulo whell do PAM, permite limitar quais usuários poderão ter acesso a um determinado comando.

Vamos liberar apenas ao grupo de analistas que usem o comando su:

1 – Verificando quais usuários pertencem ao grupo analistas:

```
1# grep analistas /etc/group
```

2 - Abra o arquivo /etc/pam.d/su, descomente e edite a linha 15:

```
2# vim +15 /etc/pam.d/su
```

```
auth       required   pam_wheel.so group=analistas
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo whell

3 – Habilite os logs de su, descomente a linha 67 do arquivo login.defs:

```
3# vim +67 /etc/login.defs
```

SULOG_FILE /var/log/sulog

4 – Logue com o usuário annie.dee e tente executar o comando "su -"

```
4$ su -
su: Permissão negada
```

Faça o mesmo teste como usuário harry.rosemberg.

5 – Finalizando verifique o log do comando su:

```
5# tail -f -n2 /var/log/sulog
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo limits

- O módulo “pam_limits” é usado para limitar praticamente todos os recursos da máquina a cada usuário. Pode-se inclusive determinar o tempo de consumo de CPU e memória RAM.

1 – Verifique o módulo ativado no /etc/pam.d/login:

```
1# vim +77 /etc/pam.d/login
```

```
77 session    required   pam_limits.so
```

Implementando Controle de Acesso ao Sistema com PAM

Implementação prática do PAM – Módulo limits

2 – A configuração do módulo é feita no arquivo /etc/security/limits.conf:

sintaxe: <usuario/grupo> <tipo_de_limite> <recurso> <valor_do_limite>

3 – Vamos limitar o número de terminais consecutivos que um usuário pode utilizar.

```
1# vim /etc/security/limits.conf
```

@analistas hard maxlogins 2

4 – Para testar, logue 3 vezes com o usuário Harry Rosemberg por SSH ou Localmente;

Implementando Controle de Acesso ao Sistema com PAM

- Outro fator importante na questão da segurança é o chamado Fork Bomb!

- Ele nada mais é do que um processo capaz de criar sub-processos, em looping infinito, causando o congelamento do servidor.

Implementando Controle de Acesso ao Sistema com PAM

- Crie um exemplo conforme abaixo:

$ bomb () { bomb | bomb & } ; bomb

Para limitar o número de processos gerados por usuário, criando forks como o que aparece acima adicione a seguinte linha ao arquivo /etc/security/limits.conf

```
1# vim /etc/security/limits.conf
```

*** hard nproc 200**


4LINUX
FREE SOFTWARE SOLUTIONS