Capítulo 16 - Rsyslog

16.1. Objetivos

•Entender o funcionamento do Rsyslog;

•Utilizar as facilidades, níveis e destinos do Rsyslog;

•Configurar o Rsyslog para fins de debug;

•Ativar gravação de Logs remotos.

16.1. Introdução teórica

A necessidade de registro das atividades dos usuários e serviços dos sistemas é, notoriamente, muito importante para Administradores de Sistemas. A norma NBR ISO/IEC 27002 recomenda no item 10.10.1 as seguintes características de um sistema de logs:

1.Identificação dos usuários;

2. Datas e horários de entrada e saída de terminais;

3. Hostname ou endereço IP, para serviços acessados via rede;

4.Registro das tentativas de acesso aceitos e rejeitados.

16.1. Organização do Rsyslog

Cada linha do arquivo /etc/rsyslog.conf é organizada contendo o seguinte conteúdo:

```
# facilidade.nível  destino
```

Vamos explicar o que vem a ser cada um desses itens:

•**facilidade** - É usada para especificar que tipo de programa está enviando a mensagem.

•**nível** - Especifica o nível de gravidade da mensagem.

•**destino** - Especifica para onde deve ser mandada a mensagem de log.

16.1. Facilidades do Syslog

auth - Mensagens de segurança/autorização.

authpriv - Mensagens de segurança/autorização (privativas).

cron - Serviços de agendamento (cron e at).

daemon - Outros serviços do sistema que não possuem facilidades específicas.

ftp - Serviço de ftp do sistema.

kern - Mensagens do kernel.

lpr - Subsistema de impressão.

local{0-7} - Reservados para uso local.

mail - Subsistema de e-mail.

news - Subsistema de notícias da USENET.

security - Sinônimo para a facilidade auth.

rsyslog - Mensagens internas geradas pelo syslog.

user - Mensagens genéricas de nível do usuário.

uucp - Subsistema de UUCP.

* - *Confere com todas as facilidades.*

16.2. Níveis

emerg	*- O sistema está inutilizável.*
alert	*- Uma ação deve ser tomada imediatamente para resolver o problema.*
crit	*- Condições críticas.*
err	*- Condições de erro.*
warning	*- Condições de alerta.*
notice	*- Condição normal, mas significante.*
info	*- Mensagens informativas.*
debug	*- Mensagens de depuração.*
*	*- Confere com todos os níveis.*
none	*- Nenhuma prioridade.*
error	*- Sinônimo para o nível err.*
panic	*- Sinônimo para o nível emerg.*
warn	*- Sinônimo para o nível warning.*

16.3. Destinos

arquivo - O Rsyslog enviará os logs para um arquivo. Essa opção é a mais comum.

| - O Rsyslog enviará os logs através de um pipe. Muito usado para redirecionar logs à um terminal.

@ - Com a arroba, o Rsyslog enviará seus logs para um computador remoto, utilizando hostname ou endereço IP.

user1,user2 Especificando o usuário, o Rsyslog enviará a mensagem para os usuários especificados. Múltiplos usuários são separados por vírgula.

** - Com o asterisco, o Rsyslog enviará os logs para todos os usuários logados no momento, através do comando "wall".*

16.4. Arquivos importantes

Logs de controle do kernel (comando dmesg): /var/log/messages
Logs de depuração de daemons: /var/log/daemon.log
Logs utilizados pelo comando last: /var/log/wtmp
Logs utilizados pelo comando lastb: /var/log/btmp
Log utilizado pelo comando lastlog: /var/log/lastlog
Logs utilizados pelo comando w e who: /var/run/utmp

16.5. Prática dirigida

16.5.1. Verificando os logs

Instale o pacote do rsyslog:

```
# aptitude install rsyslog
```

1) Edite o arquivo de configuração do rsyslog, e coloque ative a opção de Logs do cron:

```
# vi /etc/rsyslog.conf
cron.*                                  /var/log/cron.log
```

2) Vamos criar uma política de rsyslog que possibilite registrar todos os acontecimentos:

```
*.* /var/log/tudo.log
```

3) Reinicialize o daemon do Rsyslog:

```
# invoke-rc.d rsyslog restart
```

Dica Red Hat:

```
# service rsyslog restart
```

4) Verifique o arquivo /var/log/tudo.log

```
# cat /var/log/tudo.log
```

16.5.1. Logs Centralizados

1) Primeiramente, é necessário que o servidor seja habilitado para receber os logs de outras máquinas, para isto, acrescente o parâmetro -r:

```
# vim /etc/default/rsyslog
```

2) Modifique o conteúdo do arquivo, acrescentando o parâmetro:

```
SYSLOGD="-r"
```

Descomente também as linhas no arquivo /etc/rsyslog.conf do servidor:

Tem que ficar assim:

```
$ModLoad imtcp
$InputTCPServerRun 514
$ModLoad imtcp
$InpuTCPServerRun 514
```

Reinicialize o serviço Rsyslod:

```
# invoke-rc.d rsyslog stop
# invoke-rc.d rsyslog start
```

3) Nas estações, é preciso alterar o arquivo /etc/rsyslog.conf, inserindo a linha que indica quais arquivos serão enviados e qual o servidor de logs:

```
*.*                    @ENDERECO_IP
```

Se a máquina servidor tiver IP = 192.168.1.100, a linha fica:

```
*.* @192.168.1.100
```

4) Reinicialize o serviço Rsyslod:

```
# invoke-rc.d rsyslog stop
# invoke-rc.d rsyslog start
```

Dica LPI: O candidato será questionado sobre como configurar um log remoto. Podem ser cobradas também, regras para adicionar os níveis, facilidades e destino, fique atento aos exercícios.

5) Certifique-se de que a porta está disponível para conexões remotas:

```
# nestat -putan | grep 514
```

6) Visualize a atividade do servidor de log na rede:

```
# mii-tool
# tcpdump -i ethX -X -n -vv src IP_DE_ORIGEM and dst IP_DE_DESTINO
```

16.5.1. Rotação de Logs

Com o tempo, os logs podem ocupar todo o espaço disponível na partição. Por isso, devemos configurar corretamente a política de rotação dos logs, ou seja, durante quanto tempo os logs serão armazenados no seu computador.

Para isso, abra o arquivo /etc/logrotate.conf:

```
# vim /etc/logrotate.conf
```

```
#Definindo rotação de logs semanalmente
weekly

# Manter os logs de 4 semanas
rotate 4

# Criar um arquivo novo para cada rotação de logs
create

# Descomente caso queira compactar os logs em formato .gz
compress

# Todo arquivo dentro deste diretório será considerado
como uma configuração de log rotate.

include /etc/logrotate.d
# Configurações para wtmp e btmp
/var/log/wtmp {
 missingok
 monthly
 create 0664 root utmp
 rotate 1
}

/var/log/btmp {
 missingok
 monthly
 create 0664 root utmp
 rotate 1
}

# system-specific logs may be configured here
```

1) Crie uma configuração de logrotate:

```
# vim /etc/logrotate.d/errors
```

2) Inclua no arquivo o seguinte conteúdo:

```
/var/log/*.err /var/log/*.info {
                        daily
                        size 5M
                        sharedscripts
                        postrotate
                        /usr/bin/pkill -1 syslogd
                        endscript
                        rotate 5
}
```

• /var/log/*.err /var/log/*.info - Todos os logs com a extensão err e info.

• daily - O sistema de logs será diário.

• size 5M - Faz o rotate quando o arquivo alcançar 5M.

• sharedscripts - Marca o inicio do bloco de comandos.

• postrotate - Efetua os scripts após aplicar rotate aos arquivos.

• /usr/bin/pkill -1 syslogd - Envia sinal 1 ao processo syslog.

• endscript - Encerra o bloco de comandos.

•rotate 5 - Aplica rotate aos arquivos 5 vezes.

1) Redirecione todos os logs para o arquivo teste.err:

```
# cat /var/log/* >> /var/log/teste.err
```

Execute o comando acima até ultrapassar os 5MB estabelecidos na política de logrotate.

2) Agora, execute o logrotate manualmente:

```
# logrotate -f /etc/logrotate.conf
```

3) Redirecione o arquivo teste.err.1 para o arquivo teste.err:

```
# cat /var/log/teste.err.1 >> /var/log/teste.err
```

4) Acione o logrotate:

```
# logrotate -f /etc/logrotate.conf
```

Refaça os teste até atingir o limite de 5 "backlogs".