at - Agendamento de Tarefas

at é uma ferramenta de agendamento de tarefas que permite que tarefas sejam executadas no seu sistema. O **batch** é como **at**, mas executa as tarefas únicas quando a carga do sistema permite, todos os comando e funções de **at** podem ser replicados pelo **batch** com os mesmos argumentos.

Comandos Relacionados

Os seguintes comandos são incluídos para monitorar o comando **at**:

- **at** - O programa usado para enviar uma tarefa única.
- **atq** - O programa usado para revisar a fila de trabalhos enviados com **at**
- **atrm** - O programa usado para remover trabalhos da fila **at**.

Uso do comando at

O comando **at** pode ser emitido para permitir que você escreva um script personalizado ou apontar para um script pré-construído. Além disso, as especificações de tempo permitido pode aumentar a complexidade.

O argumento *-m* pode ser adicionado a qualquer comando que diga ao daemon para enviar o usuário no sistema local avisando sobre a conclusão da tarefa.

Agendando um Script Existente

Este exemplo dirá ao aplicativo **at** para executar o script */home/user/testscript.sh* às 16h de hoje e envie um e-mail ao usuário após conclusão.

```
user@darkstar$ at 16:00 -m -f /home/user/testscript.sh
```

Como você pode ver, o *-f* argumento significa **arquivo**, pois aponta para o arquivo que será executado.

Agendando uma nova ação

Como visto abaixo, quando nenhum arquivo é especificado, o aplicativo **at** abre um prompt a solicitar que o usuário insira o novo script.

```
user@darkstar$ at 19:40 -m
warning: commands will be executed using /bin/sh
at> ping -c 4 www.google.com
at> echo $?
at> <EOT>
```

```
job 4 at Sun Dec 30 19:40:00 2012
```

O usuário precisará digitar a sequência de comandos que serão emitidos dentro do prompt, após a conclusão você pode fechar o prompt com a combinação de teclas Ctrl+d

O comando acima instruiu o **at** para pingar o google 4 vezes às 19:40, retornar o status do aplicativo e enviar os resultados por e-mail para o usuário.

Certifique-se de anotar o id do job, é o único identificador visível para o job e deve ser referenciado se você deseja remover o job mais tarde.

Revisando Tarefas Em filas

O comando **atq** lista os job pendentes do usuário. A saída é o número do job, data, hora, fila e nome de usuário.

```
user@darkstar$ atq
4 Sun Dec 30 19:40:00 2012 a user
```

Uma vez que o número da tarefa tenha sido anotado, você pode revisar os comandos incluído no trabalho emitindo o comando *at -c {jobnumber}*, e o exemplo é mostrado abaixo.

```
user@darkstar$ at -c 4
#!/bin/sh
# atrun uid=1000 gid=1000
# mail user 1
umask 22
....Environment Details....
cd /home/user || {
   echo 'Execution directory inaccessible' >&2
   exit 1
}
ping -c 4 www.google.com
echo $?
```

Removendo Tarefas Em filas

O comando **atrm** remove trabalhos da fila pendente, não apresenta qualquer confirmação de que o trabalho é removido, por isso pode ser aconselhável executar **atq** após **atrm** para confirmar que o trabalho foi removido.

```
user@darkstar$ atrm 4
```

Mais recursos

- at manual page
- BrunoLinux
- http://sathyaphoenix.wordpress.com/2009/01/18/using-at-command-to-schedule-jobs-in-linux/

Sources

- Original escrito por mfillpot
- Tradução Português por MacgyverPT (Miguel Rosa)

howtos, task scheduling, author mfillpot
translated pt, macgyverpt

From:
https://docs.slackware.com/ - **SlackDocs**

Permanent link:
https://docs.slackware.com/pt-br:howtos:software:at

Last update: **2020/11/19 10:09 (UTC)**