


Luciano Antonio Siqueira

# Certificação LPI-2

**4ª Edição**



ALTA BOOKS  
E D I T O R A  
Rio de Janeiro, 2014



Suba e me veja às vezes  
Então tentarei te alcançar  
Bem do fundo do meu mar  
Pois você vê eu aprendo

(Arnaldo Baptista)



<b>Prefácios</b>	<b>7</b>
<b>Introdução</b>	<b>9</b>
<b>Visão geral das mudanças nos exames LPIC nível 2</b>	<b>9</b>
<b>Tópico 200: Planejamento de capacidade</b>	<b>13</b>
200.1 Avaliar e Resolver Problemas no Uso de Recursos	14
200.2 Prever Necessidades Futuras de Recursos	21
<b>Tópico 201: O kernel Linux</b>	<b>31</b>
201.1 Componentes do kernel	32
201.2 Compilando um kernel Linux	35
201.3 Controlar o kernel em tempo real e solução de problemas	44
201.4 Gerenciamento de dispositivos udev	51
<b>Tópico 202: Início do sistema</b>	<b>57</b>
202.1 Personalizar o início do sistema SysV-init	58
202.2 Recuperação do sistema	62
202.3 Carregadores de boot alternativos	68
<b>Tópico 203: Sistema de arquivos e dispositivos</b>	<b>79</b>
203.1 Trabalhando com o sistema de arquivos Linux	80
203.2 Manutenção de um sistema de arquivo Linux	87
203.3 Criando e configurando opções de sistemas de arquivos	93
<b>Tópico 204: Administração avançada de dispositivos de armazenamento</b>	<b>105</b>
204.1 Configuração de RAID	106
204.2 Ajustar o acesso a dispositivos de armazenamento	109
204.3 Gerenciamento de volumes lógicos (LVM)	117
<b>Tópico 205: Configuração de rede</b>	<b>125</b>
205.1 Configurações básicas de rede	126
205.2 Configuração avançada de rede e resolução de problemas	132
205.3 Soluções de problemas de rede	144
<b>Tópico 206: Manutenção do sistema</b>	<b>151</b>
206.1 Compilar e instalar programas a partir do código fonte	152

206.2 Operações de backup	155
206.3 Informar usuários sobre questões relativas ao sistema	159
<b>Tópico 207: Servidor de Nomes de Domínio</b>	<b>165</b>
207.1 Configuração básica de um servidor DNS	166
207.2 Criação e manutenção de zonas de DNS	168
207.3 Segurança de DNS	177
<b>Tópico 208: Serviços Web</b>	<b>185</b>
208.1 Configuração básica do Apache	186
208.2 Configuração do Apache para HTTPS	195
208.3 Implementação do Squid como servidor Proxy	197
208.4 Implementação do Nginx como um servidor web e proxy reverso	198
<b>Tópico 209: Compartilhamento de arquivos</b>	<b>205</b>
209.1 Configurar um servidor Samba	206
209.2 Configurar um servidor NFS	212
<b>Tópico 210: Administração dos clientes de rede</b>	<b>219</b>
210.1 Configuração DHCP	220
210.2 Autenticação por PAM	223
210.3 Uso de cliente LDAP	227
210.4 Configurar um servidor OpenLDAP	231
<b>Tópico 211: Serviços de e-mail</b>	<b>237</b>
211.1 Utilização de servidores de email	238
211.2 Administração da entrega local de e-mail	243
211.3 Administrar entrega de e-mail remoto	245
<b>Tópico 212: Segurança do Sistema</b>	<b>253</b>
212.1 Configuração de roteador	254
212.2 Segurança de servidores FTP	262
212.3 Shell seguro (SSH)	263
212.4 Tarefas de segurança	268
212.5 Open VPN	270
<b>Objetivos LPIC2</b>	<b>275</b>
<b>Respostas</b>	<b>303</b>



Este livro destina-se a candidatos em busca da certificação LPI nível 2 (LPIC-2) e atende tanto às necessidades de profissionais que já trabalham com outros sistemas operacionais como as de profissionais de Linux em geral. Este projeto da Linux New Media, tão bem conduzida por Rafael Peregrino e pelo Claudio Bazzoli, foi desenvolvido pelo Luciano Siqueira, um profissional raro, capaz de aliar conhecimentos técnicos profundos com uma impressionante capacidade de comunicação. Como resultado, temos esta obra completa, abrangente e, ao mesmo tempo, que dá todas as condições para que um candidato se prepare para as provas de certificação LPIC-2.

O LPI – Linux Professional Institute Linux ([www.lpi.org](http://www.lpi.org)) promove e certifica habilidades essenciais em Linux e em tecnologias de Open Source por meio de provas abrangentes, de alta qualidade e independentes de distribuições Linux. O LPI foi criado em 1999 pela comunidade Linux como uma organização internacional sem fins lucrativos, com o objetivo ser reconhecido como o líder global na certificação de profissionais de Linux, promovendo o Linux e o movimento de Open Source.

O programa de certificação profissional LPI é composto de três níveis de certificação (LPIC-1, LPIC-2 e LPIC-3), sendo que a certificação LPIC-1 tem como alvo profissionais junior e plenos, ao passo que a certificação LPIC-2 é orientada a profissionais mais experientes e líderes de equipes. Para que um candidato seja certificado no nível 2 já tenha obtido a certificação no nível 1.

No mundo de tecnologia, cada vez mais a certificação profissional é necessário que ele tem se tornado, uma vez é um indicativo claro e objetivo do conhecimento de uma pessoa a respeito de um determinado assunto, no nosso caso, o Linux. Obviamente, na hora de uma contratação, por exemplo, outros fatores também contam, mas o mais difícil de se avaliar é o conhecimento, já que as características pessoais e a experiência podem ser facilmente avaliadas com entrevistas, testes, referências, etc.

Assim, pode-se dizer que a certificação profissional acaba sendo uma ferramenta essencial tanto para quem contrata como para quem é contratado, garantindo que os candidatos tenham as habilidades necessárias e, conseqüentemente, sejam capazes de executar o que se espera deles. Dessa forma, garante-se um padrão de qualidade e facilita tanto a contratação como futuras promoções.

### **José Carlos Gouveia**

---

José Carlos Gouveia é Diretor Geral do Linux Professional Institute – LPI – da América Latina. Anteriormente, trabalhou por cinco anos para a SGI – Silicon Graphics – como Diretor Geral da América Latina, além de ter sido diretor geral da Novell, Platinum Technology, PeopleSoft e JDEdwards, tendo sido também diretor da Anderson Consulting (Accenture) e da Dun&Bradstreet Software e gerente da EDS. Gouveia é formado em Ciência da Computação pela Unicamp, com pós-graduação pela Unicamp e pela PUC-RJ.

## Prefácio por Jim Lacey



A certificação do Linux Professional Institute (LPI) é uma ótima maneira de demonstrar a empregadores e a potenciais clientes que você possui as habilidades e os conhecimentos necessários para trabalhar num ambiente Linux e de Código Aberto, além de ser uma excelente ferramenta para seu desenvolvimento profissional. O programa de certificação LPI é globalmente reconhecido por profissionais de TI, empresas e governos como a certificação Linux número um em todo o mundo. Esse programa vem continuamente sendo reconhecido por fabricantes como pré-requisito para seus próprios programas específicos de certificação em alto nível. São exemplos os programas de certificação do Ubuntu pela Canonical, HP, IBM, Novell e Oracle.

O Linux Professional Institute completa 10 anos em 2009, ano em que nós vamos ampliar nosso programa de certificação para satisfazer as necessidades de um mercado de TI em contínua transformação. Nós encontramos muitos desafios nos últimos dez anos e, apesar das reviravoltas da economia em 2008, vemos o aumento do número de exames aplicados ao redor do mundo. Esperamos que essa tendência positiva de certificações emitidas e exames realizados continue ao longo de 2009 e 2010, ao passo que profissionais de TI buscam atualizar seus conhecimentos e habilidades para permanecerem competitivos nesse período desafiador da economia.

Além disso, o panorama para os próximos anos no setor de Software Open Source como um todo é muito otimista. Grandes, pequenas e médias empresas, governos e outras organizações do setor público procuram novas oportunidades de negócios enquanto valorizam seu importante patrimônio pessoal. Nesse contexto, a tecnologia de Código Aberto oferece um instigante potencial em função das soluções versáteis disponibilizadas às revendas e do estável ciclo de atualizações de produtos.

Investir em tecnologia de Código Aberto é investir em pessoas, afirmação essa que sempre temos que ter em mente no Linux Professional Institute. É por isso que nossa mensagem de profissionalismo continua a ganhar apoio em todo mundo.

No LPI, nós nos guiamos pelas lições bem sucedidas do passado. Reduzindo a falta de capacitação que desafia a indústria, nosso instituto continuará a desempenhar um papel fundamental no futuro do Código Aberto. Os próximos anos serão cruciais para o LPI e nós queremos que você faça parte desse esforço em promover a contínua adoção do Linux e do Código Aberto por todo mundo!

### **Jim Lacey**

---

Jim Lacey é Presidente e CEO do Linux Professional Institute.



## Introdução

Muita coisa aconteceu desde a última vez que a Certificação LPI foi alterada. Alguns dos conteúdos que eram abordados encontravam pouca aplicação prática. Além disso, a organização dos tópicos não obedecia a uma ordenação lógica e em alguns pontos não havia distinção entre as provas da certificação nível 1 e as provas da certificação nível 2.

A revisão 3.0, além de eliminar alguns conteúdos ultrapassados e incluir novos conteúdos atualmente mais relevantes, procurou estabelecer focos ainda mais distintos entre a certificação nível 1 e a certificação nível 2.

A certificação nível 1 procura abordar todos os aspectos que envolvem a configuração e a manutenção de uma máquina local conectada à rede. Já a certificação nível 2 tem por objetivo geral a configuração e a manutenção de um ambiente de servidor. Apesar das mudanças, prevalece a política do LPI de abordar somente as ferramentas tradicionais de um sistema GNU/Linux, independente de distribuição. A seguir, está a visão geral das modificações nessa nova revisão da prova, fornecida pelo próprio LPI. ●

## Visão geral das mudanças nos exames LPIC nível 2



A nova revisão dos objetivos para as provas LPIC nível 2, válida a partir de abril de 2009, levou as provas para a versão 3.0. Essa é a segunda revisão completa dos objetivos, que padroniza a versão para o mundo todo. No âmbito geral, o LPI antecipou o ciclo de cinco anos para revisões completas. Por volta de cada dois anos e meio, os objetivos serão modificados para refletir as possíveis mudanças do Linux. A próxima versão do LPIC-1 será a 3.5, que refletirá essa revisão parcial.

Além dessas revisões principais, haverá adendos incluídos numa média trimestral, com o intuito de esclarecer pontos e detalhes dos exames. Esses adendos não alteram a versão da prova, pois têm apenas o intuito de esclarecer a cobertura da prova para organizadores de cursos e livros.

### Os novos pesos

O peso total de cada prova foi estabelecido em 60. Isso significa que, salvo em provas com perguntas “beta” para fins de desenvolvimento do exame, cada prova terá exatamente 60 questões. Portanto, a indicação de peso 3 em um determinado objetivo indica que haverá três questões sobre o tema na prova (exceto, novamente, no caso de haver questões beta para fins de desenvolvimento dos exames).

## Numeração dos objetivos

A numeração dos objetivos é passível de dúvida em função de sua falta de linearidade. Por isso, os prefixos *1.* e *2.* foram descartados nessa revisão. Em todos os momentos em que numerações como *1.xxx.y* ou *2.xxx.y* aparecem, o fazem para citar os objetivos antigos.

## Redução de conteúdo duplicado

Em versões anteriores dos objetivos da certificação LPI, alguns tópicos eram abordados tanto nos exames do nível 1 quanto nos exames do nível 2. Em alguns casos, o mesmo conteúdo aparecia em diferentes provas dentro do mesmo nível de certificação. A atualização dos objetivos buscou reduzir as ocorrências de conteúdo duplicado em diferentes provas ou objetivos.

Contudo, algumas tecnologias – como DNS – são importantes nos dois níveis de certificação e estão distribuídos nos locais apropriados. Por exemplo, na certificação nível 1, a abordagem sobre o DNS está restrita à configuração do cliente do serviço. Na certificação nível 2, a abordagem passa para configuração e segurança de servidores DNS.

## Versões de programas

Quando apropriado, as versões específicas de programas são mostradas nos objetivos. Por exemplo, a abordagem do Kernel 2.4 foi descartada para priorizar a versão 2.6. As questões relacionadas ao ReiserFS limitam-se à versão 3 do sistema de arquivos e o servidor Bind 8.x não é mais abordado na prova.

## Alterações de conteúdo

Enquanto o foco da certificação nível 1 foi mais direcionado para o uso e administração de um sistema Linux local, a maioria dos serviços de rede e demais tarefas administrativas foram movidas para a certificação nível 2.

Além disso, os tópicos que tratam do kernel foram reunidos na certificação nível 2. Além disso, tópicos mais avançados, antes abordados na certificação nível 1, foram trazidos para a certificação nível 2. Por exemplo, a compilação de programas a partir do código fonte agora é abordada na LPIC-2. Temas relativos a segurança e solução de problemas ganharam muito mais foco nesse nível.

## Solução de problemas e Segurança

A cobertura dos conteúdos relativos a solução de problemas e segurança aumentou muito. Em muitos casos, pesos originalmente marcados como 1 ou 2 passaram a 5. Com isso, o próprio tópico específico para solução de problemas tornou-se um dos mais importantes na prova.



## Gerenciamento de dispositivos udev

O sistema udev foi incluído para garantir que os candidatos compreendam como é a detecção e gerenciamento de dispositivos em sistemas GNU/Linux modernos.

## LVM

A abordagem ao LVM – *Logical Volume Management* – foi ampliada. Agora, o tema conta com seu próprio objetivo.

## IMAP/POP

A prova 202 agora aborda servidores IMAP/POP. São abordados os mais comuns em ambientes GNU/Linux: o Courier e Dovecot. ●

## Como usar este livro



Este livro está organizado segundo o programa de conteúdos oficiais para a Certificação LPI-2. Dessa forma, o candidato encontrará exatamente os temas que são abordados nos exames de certificação, na profundidade que é exigida para a prova. Além disso, a sequência dos assuntos corresponde à sequência em que serão abordados na prova. Essa estrutura auxilia o candidato a manter o foco naquilo que é importante para a prova, mas sem deixar de lado a coerência e a consistência do texto.

Ao final de cada tópico foram colocadas 10 perguntas correspondentes aos temas abordados no tópico em questão, totalizando 130 questões no livro. Com a finalidade de familiarizar o candidato, as perguntas foram formuladas com o mesmo formato em que apareceram nos exames de certificação. As respostas para as perguntas de todos os tópicos encontram-se no final do livro.

Professores e escolas podem se beneficiar da adoção desse livro. Conteúdos densos são abordados de forma objetiva e coesa, o que facilita o ensino e a preparação de aulas.

A leitura do livro não dispensa a experimentação prática, devendo, assim, ser acompanhada. Dado o grande volume de assuntos abordados, a utilização das ferramentas e conceitos demonstrados são muito importantes para fixação, principalmente para quem os está vendo pela primeira vez. ●



## Introdução à quarta edição

Esta nova edição do livro corresponde ao exame de certificação LPIC-2 versão 4.0, válida a partir de novembro de 2013. Como em atualizações anteriores, houve uma redistribuição de alguns conteúdos, atualizações, remoções e inclusões de temas.

Inicialmente, há um novo tópico, *Planejamento de Capacidade*, que aborda conceitos e ferramentas utilizados para avaliar e dimensionar recursos de sistema. Na realidade atual, onde a oferta sob demanda de recursos de máquina e computação distribuída tornaram-se comuns, trata-se de um assunto de grande relevância para o profissional.

Em contrapartida, o antigo tópico 213 — Solução de Problemas — foi removido. Seu assunto não é menos importante, mas tornou-se a repetição de diversos assuntos já abordados em outros tópicos.

Alguns tópicos conhecidos permanecem, mas perderam um pouco da importância no exame. Compilar o kernel do Linux, por exemplo, ainda é exigido, mas não com a mesma ênfase.

Das inclusões pontuais, merecem destaque os carregadores de boot *Grub2* e *Syslinux*, a configuração de dispositivos iSCSI, um maior enfoque na LDAP e a inclusão do servidor Web *Nginx*. Este último cresceu vertiginosamente nos últimos anos, tornando-se um forte competidor ao Apache. Por esse motivo, agora ambos são os protagonistas do tópico *Serviços Web*.

Temas como *udev*, *OpenVPN*, etc. foram remanejados para capítulos mais apropriados. Todos os objetivos oficiais para a Certificação LPIC-2 versão 4.0 podem ser encontrados no apêndice do livro, traduzidos para o português. ●



Peso total do tópico  
na prova: 8

# Tópico 200:

## Planejamento de capacidade

Principais temas abordados:

- Monitorar recursos de hardware e rede
- Prever necessidades futuras de recursos



## 200.1 Avaliar e Resolver Problemas no Uso de Recursos

### Peso 6

A infraestrutura de computação atual apresenta uma tendência semelhante àquela dos primórdios da computação, onde máquinas de grande capacidade eram utilizadas por equipamentos de pouca ou nenhuma capacidade computacional. A principal diferença, além da óbvia evolução geral da tecnologia, está na maneira que essas máquinas de maior capacidade são compartilhadas.

Hoje, os recursos compartilhados – processamento, memória, armazenamento, transmissão de dados – não são necessariamente utilizados diretamente por um usuário num terminal, mas utilizados por outras máquinas (provavelmente *máquinas virtuais*) cujos recursos são redimensionados sob demanda.

Essa oferta flexível de recursos torna possível alocar a quantidade certa de capacidade computacional a cada necessidade. Para isso, é muito importante conseguir identificar onde há subutilização de recursos e onde os recursos não são suficientes para determinadas tarefas.

Existem diversas ferramentas Linux para monitorar a utilização de recursos e também são úteis para identificar gargalos em servidores mais simples e computadores domésticos.

### Carga de Entrada/Saída

Alguns comandos para monitoramento da atividade do processador e da comunicação entre dispositivos são parte do pacote `sysstat`. O pacote `sysstat` está disponível nas principais distribuições e pode ser instalado com o comando `apt-get install sysstat` ou `yum install sysstat`. Ele contém ferramentas para monitorar o uso e performance do sistema, que podem ser utilizadas diretamente ou agendadas para gerar relatórios. Destacam-se dois desses comandos: `iostat` e `sar`.

O comando `iostat` gera um relatório com estatística de uso da CPU e de entrada/saída de dispositivos e partições. Para exibir um relatório simplificado sobre a utilização de CPU, dispositivos e partições desde o último início do sistema, basta executar `iostat` sem argumentos:

```
# iostat
Linux 3.2.0-4-amd64 (debian) 11/24/13  _x86_64_ (1 CPU)

avg-cpu:  %user  %nice  %system  %iowait  %steal  %idle
           0.11   0.05   0.14    0.44    0.01   99.25

Device    tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
vda       4.10     55.34        21.75        525551     206521
sda       0.02     0.08         0.00         800        0
sdb       0.02     0.08         0.00         800        0
dm-0     5.49     52.30        19.89        496613     188884
dm-1     0.60     0.54         1.86         5172       17628
```

Na porção `avg-cpu` está o relatório referente à utilização do CPU. Os campos são:

- *%user*  
Exibe o percentual de uso da CPU durante a execução no nível de usuário (aplicações).
- *%nice*  
Exibe o percentual de uso da CPU durante a execução no nível de usuário com prioridade modificada.
- *%system*  
Exibe o percentual de uso da CPU durante a execução no nível de sistema (kernel).
- *%iowait*  
Exibe o percentual de tempo em que a CPU estava ociosa aguardando resposta de entrada/saída de disco.
- *%steal*  
Exibe o percentual de tempo que a CPU virtual estava ociosa devido ao hipervisor estar servindo outra CPU virtual.
- *%idle*  
Exibe o percentual de tempo em que a CPU estava ociosa e não aguardava por entrada/saída de disco.

Abaixo das informações de CPU é exibida a utilização dos dispositivos de armazenamento. Os campos são:

- *Device*  
O nome do dispositivo ou partição localizado em `/dev`.
- *tps*  
O número de transferências por segundo realizadas no dispositivo. Uma transferência é uma operação de entrada ou saída, mas mais de uma operação pode ser incluída numa transferência. O tamanho de uma transferência é indeterminado.
- *kB\_read/s* e *kB\_wrtn/s*  
Quantidade de dados lidos e gravados no dispositivo por segundo.
- *kB\_read* e *kB\_wrtn*  
Quantidade de dados lidos e gravados no dispositivo no intervalo.

Para gerar o relatório em intervalos de tempo específicos, é utilizada a opção `-d`. Assim, para um intervalo de dois segundos:



```
# iostat -d 2
Linux 3.2.0-4-amd64 (debian) 11/24/13 _x86_64_ (1 CPU)

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
vda                 3.98     53.73        21.12        525551     206577
sda                 0.02     0.08         0.00         800         0
sdb                 0.02     0.08         0.00         800         0
dm-0                5.33     50.77        19.32        496613     188940
dm-1                0.58     0.53         1.80         5172        17628

Device:            tps    kB_read/s    kB_wrtn/s    kB_read    kB_wrtn
vda                 0.00     0.00         0.00         0           0
sda                 0.00     0.00         0.00         0           0
sdb                 0.00     0.00         0.00         0           0
dm-0                0.00     0.00         0.00         0           0
dm-1                0.0      0.00         0.00         0           0
```

A primeira listagem mostra o uso desde o último início do sistema e as listagens seguintes o uso nos últimos dois segundos.

Todas as informações são tomadas de arquivos no diretório `/proc/e/sys/`, portanto, é necessário que esses diretórios estejam montados e seus conteúdos disponíveis.

O comando `sar` é utilizado para gerar relatórios com as mesmas informações do `iostat`. As informações lidas pelo `sar` são aquelas armazenadas por agendamentos de outros comandos do pacote `sysstat`. Portanto, é importante verificar se o serviço de coleta de dados será ativado no início do sistema. No Debian, o arquivo `/etc/default/sysstat` deve conter a linha `ENABLED=true`. Em distribuições que usam `systemd` (como Fedora), o serviço deve ser ativado para início automático com `systemctl enable sysstat.service`.

Os arquivos com os dados coletados são armazenados em `/var/log/sysstat/` ou `/var/log/sa/`, cada arquivo corresponde a um dia. Para ler as informações de um arquivo específico, utiliza-se a opção `-f`:

```
sar -f /var/log/sysstat/sa24
```

Os relatórios podem ser gerais ou para uma área específica do sistema:

- *sar*  
Sem opções, exibe dados de uso da CPU
- *sar -b*  
Resumo da entrada/saída de dados.
- *sar -d*  
Entrada e saída de dados, por dispositivo.
- *sar -B*  
Informação sobre a paginação feita pelo sistema.
- *sar -r*  
Utilização da memória RAM.

*sar -S*

Utilização do espaço de swap.

*sar -n DEV*

Estatísticas de tráfego de dados em todas interfaces de rede do sistema.

Diversas outras informações podem ser geradas pelo `sar`. Em `man sar` estão listadas todas as opções possíveis de relatórios.

Os comandos do pacote `sysstat` não são os únicos que podem exibir relatórios de uso de recursos no Linux. Comandos tradicionais do Linux também oferecem opções de inspeção e análise.

Semelhante ao `sar -n DEV`, o comando `netstat -i` lista as interfaces de rede e suas respectivas estatísticas de tráfego. O comando `netstat -s` exibe estatísticas de tráfego para cada protocolo de rede.

Já o `vmstat` é praticamente uma alternativa ao `iostat`. Ele informa sobre atividade de processos, memória, paginação, entrada/saída de blocos, falhas, discos e `cpu`.

O primeiro relatório informa as médias desde o último início do sistema. Relatórios adicionais fornecem informações de amostras num intervalo de tempo. Relatórios de processos e memórias são gerados instantaneamente.

Podem ser fornecidos ao `vmstat` o intervalo de atualização e o número de repetições de cada informa. Por exemplo, gerar informes do últimos dois segundos, dez vezes:

```
$ vmstat 2 10
```

procs	memory				swap		io		system			cpu		
r b	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id	wa
0 0	49264	121396	27796	2229236	0	0	53	51	252	238	5	3	91	1
0 0	49264	121296	27804	2229268	0	0	0	20	512	678	1	1	99	0
0 0	49264	121172	27812	2229276	0	0	0	6	518	660	1	1	98	1
0 0	49264	121048	27812	2229276	0	0	0	0	479	651	1	1	98	0
0 1	49264	120768	27812	2229376	0	0	0	62	501	681	1	1	99	0
0 0	49264	120768	27812	2229396	0	0	0	2	562	723	1	1	98	1
0 0	49264	120644	27812	2229396	0	0	0	0	449	612	0	1	99	0
0 0	49264	120520	27820	2229396	0	0	0	18	530	687	1	1	98	0
0 0	49264	120396	27820	2229396	0	0	0	0	464	633	1	1	99	0
0 1	49264	120272	27820	2229496	0	0	0	60	517	689	1	1	98	0

O modo de exibição padrão do `vmstat` é o modo VM, cujos campos e subcampos são detalhados a seguir:

- **procs**

r: Número de processo executáveis (executando ou aguardando execução).

b: Número de processos dormindo ininterruptíveis.

- **memory**

swpd: Total de memória virtual utilizada.

free: Total de memória livre.

buff: Total de memória utilizada como buffer.

cache: Total de memória utilizada como cache.

inact: Total de memória inativa (opção -a).

active: Total de memória ativa (opção -a).

- **swap**

si: Quantidade de memória carregada do disco (por segundo).

so: Quantidade de memória movida para o disco (por segundo).

- **io**

bi: Blocos recebidos de um dispositivo de bloco (blocos/s).

bo: Blocos enviados para um dispositivo de bloco (blocos/s).

- **system**

in: O número de interrupções por segundo, incluindo o clock.

cs: O número de mudanças de contexto por segundo.

- **cpu**, percentuais do total do tempo da CPU.

us: Tempo gasto executando código não-kernel (tempo de usuário, incluindo tempo nice).

sy: Tempo gasto executando código do kernel (tempo do sistema).

id: Tempo ocioso.

wa: Tempo gasto aguardando por entrada/saída.

st: Tempo roubado de uma máquina virtual.

O vmstat oferece outros modos de exibição. Um deles é modo disco, iniciado com a opção -d:

```
$ vmstat -d
disk-----reads-----writes-----IO-----
total    merged    sectors    ms    total    merged    sectors    ms    cur    sec
loop0      0         0         0     0     0 0     0 0     0 0
loop1      0         0         0     0     0 0     0 0     0 0
loop2      0         0         0     0     0 0     0 0     0 0
loop3      0         0         0     0     0 0     0 0     0 0
loop4      0         0         0     0     0 0     0 0     0 0
loop5      0         0         0     0     0 0     0 0     0 0
loop6      0         0         0     0     0 0     0 0     0 0
loop7      0         0         0     0     0 0     0 0     0 0
sda 306268 19859 15038540    2268632 180738    96537    12733376 11353793 0 1230
sr0      0         0         0     0     0     0     0 0     0 0
```

Os campos do modo disco indicam:

- **Reads**

total: Total de leituras com sucesso

merged: Leituras agrupadas (resultando em uma E/S)

sectors: Setores lidos com sucesso

ms: Milissegundos gastos lendo



- **Writes**

total: Total de escritas completadas com sucesso

merged: Escritas agrupadas (resultando em uma E/S)

sectors: Setores escritos com sucesso

ms: Milissegundos gastos escrevendo

- **IO**

cur: Entrada/saída em progresso

s: Segundo gastos com E/S

Para exibir informações de uma partição específica, basta iniciar no modo partição informando seu caminho com a opção -p:

```
$ vmstat -p /dev/sda1
sda1      reads  read sectors  writes  requested writes
          120428    5559754      52566    4181088
```

Os campos no modo partição representam:

- reads: Número total de leituras realizadas nessa partição
- read sectors: Total de setores lidos para a partição
- writes : Número total de escritas realizadas nessa partição
- requested writes: Número total de requisições de escrita feitas nessa partição

## Carga de processamento

Se o objetivo é inspecionar processos e atividade dos usuários, os comandos a seguir são indicados:

- ps

Mostra os processos ativos de maneira detalhada.

- top

Monitora continuamente os processos, mostrando informações como uso de memória e CPU de cada processo. A tecla [H] fornece ajuda sobre o uso do programa. Pode ser usado para alterar a prioridade de um processo.

- pstree

Mostra processos ativos em formato de árvore genealógica (processos filhos ligados aos respectivos processos pais).

- w

Mostra quais usuários estão utilizando o sistema e suas atividades em execução.

Estes comandos, especialmente em conjunto com os comandos de estatísticas de uso, podem ajudar a identificar possíveis gargalos prejudicando o desempenho do sistema. Por exemplo, se os comandos `iostat` ou `vmstat` mostrarem que muito do tem-

po da CPU é gasto apenas aguardando requisições de entrada/saída de dispositivo, é possível que exista alguma falha no dispositivo ou no processo está executando gerando as requisições.

Processos aguardando por requisições de E/S ficam no estado chamado *dormindo ininterruptível*. Esses processos podem ser identificados pela letra **D** na saída do comando `ps -ely`:

```
# ps -ely|grep "^D"
D    0  18    2  0  80    0    0    0  -   ?      00:00:00 kswapd0
```

As opções `-ely` determinam a exibição dos processos em formato longo, mais detalhado. A primeiro campo indica o estado do processo e o terceiro campo indica seu número PID, entre outros. Num sistema sem problemas, é normal que eventualmente um ou outro processo esteja muito rapidamente no estado *D*. Num sistema com problemas de E/S, o processo responsável pode ser rastreado dessa forma.

Para investigar quem ou quais processos estão utilizando um determinado arquivo, é usado o comando `lsuf`. Na medida que, em ambientes Unix, praticamente todos os recursos podem ser acessados a partir de arquivos ou pseudo-arquivos, o `lsuf` é uma ferramenta de investigação bastante poderosa.

Por exemplo, é possível listar todos os arquivos abertos por um determinado processo utilizando o comando:

```
lsuf -p PID
```

Neste código, `PID` representa o PID do programa em questão. Para verificar todos os arquivos sendo utilizados por processos disparados por um usuário em particular, utiliza-se a opção `-u usuário`. Para checar quais processos estão utilizando um arquivo, basta executar o comando `lsuf` fornecendo como argumento o caminho completo para o arquivo:

```
# lsuf /dev/net/tun
COMMAND      PID    USER   FD  TYPE DEVICE SIZE/OFF  NODE NAME
openvpn      661  openvpn  5u  CHR  10,200    0t0  9859 /dev/net/tun
qemu-syst    20649  qemu    24u  CHR  10,200    0t0  9859 /dev/net/tun
```

A saída deste comando mostra várias informações úteis sobre o(s) processo(s) a acessar o arquivo `/dev/net/tun` (interface de rede TUN). Este uso é especialmente útil para checar processos bloqueando dispositivos ou pontos de montagem.

Outro comando útil para identificar o consumo geral de recursos da máquina é o uptime:

```
# uptime
13:31:00 up 36 days, 21:03,  3 users,  load average: 0.05, 0.19, 1.27
```



É exibido quanto tempo desde que o sistema foi ligado pela última vez e a quantidade de usuários atualmente no sistema. Os valores finais, *load average*, referem-se a média de carga do sistema nos últimos 1, 5 e 15 minutos, respectivamente. As médias são calculadas a partir do total de processos que estão rodando ou ininterruptíveis no intervalo de tempo em questão. Um processo rodando é aquele utilizando ou aguardando para utilizar o processador. Um processo ininterruptível está tentando acessar um dispositivo de entrada/saída, como um disco rígido.

Os valores devem ser interpretados de acordo com o número de processadores presentes no sistema. Numa máquina com apenas um processador, uma média próxima a 1 significa que a praticamente todo momento havia um processo rodando ou ininterruptível. Já numa máquina com quatro processadores, a média próxima a 1 significa que o fato ocorreu em apenas um dos processadores. Portanto, para supor que havia um processo rodando ou ininterruptível nos quatro processadores, um valor próximo a 4 deve ser a média.

## 200.2 Prever Necessidades Futuras de Recursos



### Peso 2

Um método eficiente para prever se a capacidade instalada será suficiente para atender demandas futuras é analisar a curva ascendente em gráficos de utilização de recursos. Existem algumas ferramentas Linux para criação de gráficos a partir do monitoramento de máquinas e redes que auxiliam nessa tarefa.

Existem diversas ferramentas Linux para monitorar a utilização de recursos e também são úteis para identificar gargalos em servidores mais simples e computadores domésticos.

### Collectd

O collectd é um pequeno daemon (serviço) escrito em C para ganho de performance. Ele lê diversas estatísticas de sistema e atualiza arquivos RRD, criando-os se necessário. Como o daemon não precisa iniciar toda vez que precisa atualizar os arquivos, torna-se muito rápido e discreto no sistema. As estatísticas são muito detalhadas e os arquivos atualizados a cada dez segundos. O collectd está disponível nos repositórios de todas as principais distribuições Linux.



## RDD

RRD é o acrônimo de Round Robin Database. O RRD é um sistema para armazenar e exibir dados em série (como largura de banda de rede, temperatura do ambiente, carga média do servidor, etc). Ele armazena os dados de um modo muito compacto que não aumenta ao longo do tempo. Pode ser utilizado por scripts ou por interfaces que consultam o dispositivo e montam uma interface gráfica para o usuário.

O `collectd` pode monitorar praticamente qualquer aspecto do sistema. Sua arquitetura de plugins o torna muito personalizável, o que especialmente útil para uso em equipamentos limitados, como roteadores. A lista completa de plugins pode ser vista em [https://collectd.org/wiki/index.php/Table\\_of\\_Plugins](https://collectd.org/wiki/index.php/Table_of_Plugins). A configuração geral do `collectd` e a ativação e configuração de seus plugins é feita no arquivo `/etc/collectd/collectd.conf`.

As informações colhidas podem ser utilizadas para encontrar gargalos de performance (análise de performance) e prever a carga futura do sistema (planejamento de capacidade). Não é o foco do `collectd` gerar os gráficos, mas apenas recolher as informações e gravá-las em arquivos RRD. Por padrão, os arquivos são armazenados no diretório `/var/lib/collectd/`, mas isso pode ser modificado no arquivo `/etc/collectd/collection.conf`. Essas informações podem ser transmitidas via rede por plugins específicos e consultadas por programas especializados, como o `rrdtool`.

Contudo, operar diretamente com os dados RRD pode ser muito trabalhoso. Existem interfaces que tornam a tarefa de monitorar e gerar os gráficos mais simples, como a interface web Cacti.

## Cacti

O Cacti é uma interface completa para o RRDTool. Ele armazenam todas as informações necessárias para gerar gráficos e preenchê-los com dados de em um banco de dados MySQL. A interface é feita em PHP. Além de manter gráficos, fontes de dados e os arquivos *round robin* no banco de dados, o cacti também é responsável pela coleta dos dados.

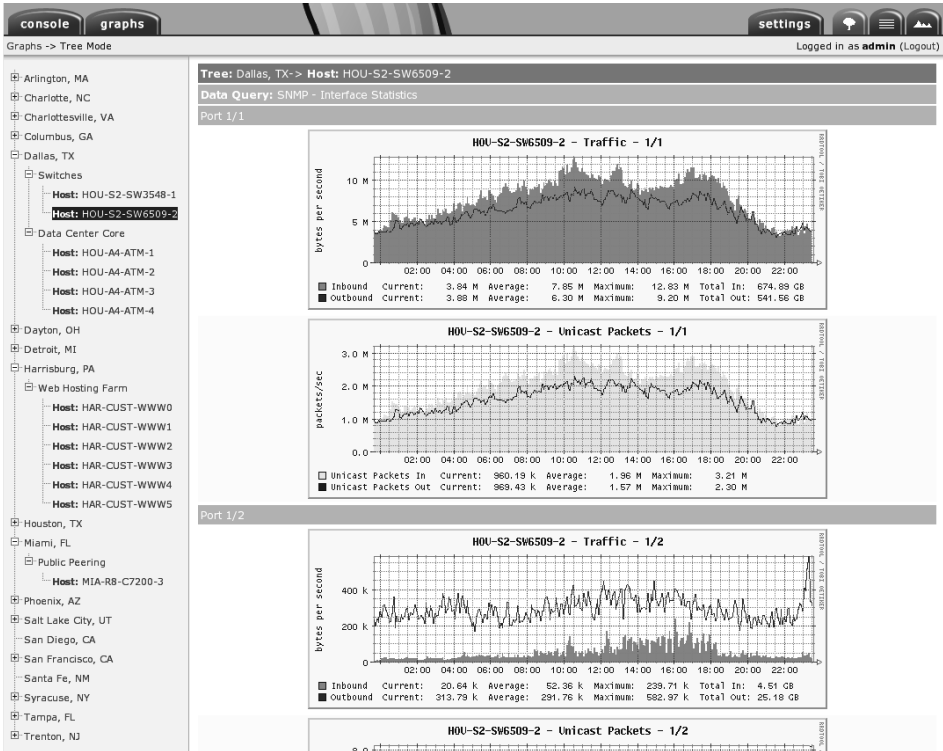


Figura 1. A interface Web do Cacti facilita a coleta e criação de dados RRD.

Na máquina onde foi realizada a instalação, a interface do Cacti poderá ser aberta pelo navegador Web no endereço <http://localhost/cacti>. O Cacti também oferece suporte s SNMP para criação de gráficos com MRTG.

## MRTG

O MRTG (*Multi Router Traffic Grapher*) é uma ferramenta para monitorar a carga de tráfego em conexões de rede. O MRTG cria páginas HTML contendo imagens com uma representação em tempo real do tráfego.

O MRTG utiliza o protocolo SNMP (*Simple Network Management Protocol*), que via de regra está disponível no roteador da rede, possibilitando monitorar todo o tráfego passando pelo roteador.

O MRTG é um programa simples e está disponível nos repositórios de todas as principais distribuições. Depois de instalado, é necessário editar o arquivo `/etc/mrtg/mrtg.cfg`. A seguir, um exemplo mínimo do conteúdo de `mrtf.cfg`: