



Criando uma pequena rede doméstica com o Samba

Uma rede na varanda

Mesmo quem tem uma pequena rede doméstica poderá algum dia querer centralizar os serviços de armazenamento de arquivos ou impressão. O Samba é tudo o que você precisa para isso – e não custa nada.

POR MARKUS KLIMKE

Imagine a seguinte situação: num escritório de uma pequena empresa, com um número reduzido de funcionários, há reclamações sobre a pouca flexibilidade no trabalho com os computadores. Um funcionário reclama que seus arquivos não podem ser acessados por ninguém a não ser o chefe, um segundo precisa que todos no departamento dele tenham acesso a uma planilha, um terceiro reclama que a impressora funciona somente de vez em quando e um quarto pede acesso a outros computadores além do dele. Se possível aos computadores com Linux, com Windows® e, talvez, também aos Macs.

Nesses casos não tem xepa: o administrador do sistema precisa de faro fino para encontrar uma infraestrutura adequada e que consiga atender às expectativas de seus clientes – ou seja, os usuários da rede. É fácil complicar desnecessariamente a situação, estabelecendo interações altamente complexas entre o servidor e os clientes. Vejamos: Kerberos para autenticação, uma criptografia SSL/TLS sobre certificados caríssimos, um *backend* LDAP parrudo, servidores Linux e Windows em operação interativa, sala climatizada, segurança na porta, café das antilhas

numa cafeteira robotizada e... opa, pra que tudo isso? Será que tudo isso vale a pena numa rede pequena?

O relacionamento do número de servidores adaptados para os clientes também deve ser especialmente equilibrado. Nesse caso é imprescindível um bom planejamento e uma solução clara. Este tutorial mostra um exemplo de como um administrador pode montar uma rede funcional e confiável em um pequeno escritório, em uma república de estudantes ou em casa.

Aprendendo a sambar

O *Samba* [1] é um intermediário entre dois (ou três) mundos. Ele oferece, numa rede, diferentes recursos – como impressão e armazenamento de arquivos – de maneira centralizada. Esses recursos podem ser acessados por máquinas rodando Linux, Windows ou Mac OS que estejam ligados na rede, desde que reconheçam o protocolo *SMB*, a base sobre a qual o servidor Samba foi desenvolvido. O protocolo *SMB* (*Server Message Block* – Bloco de Mensagens do Servidor) foi desenvolvido pela IBM e pela Microsoft® para a troca de arquivos dentro de uma rede. Os desenvolvedores do Samba não podiam contar com nenhuma documentação da Microsoft®, portanto valeram-se de fer-

ramentas de captura de tráfego de rede como a *Ethereal* [2] para observar como as estações Windows conversavam entre si. A partir desse monitoramento puderam, por engenharia reversa, implementar sua versão de um servidor que dialoga pelo dialeto *SMB* – o servidor Samba, hoje na versão 3. A versão 4 está em fase de desenvolvimento e espera-se que em breve ponha o pé na estrada.

Mesmo tendo como núcleo o protocolo *SMB*, basicamente o Samba pode oferecer mais do que somente um protocolo de comunicação: se utilizado como servidor, ele coloca à disposição a possibilidade de compartilhamento de dados e informações na rede. Tanto o Linux quanto o Windows® e o Mac OS devem interagir de maneira semelhante.

Construindo o carro alegórico

Praticamente todas as distribuições Linux oferecem o pacote Samba. As “caixinhas” de servidor da SUSE e da Red Hat, chamadas *SLES* (*SUSE Linux Enterprise Server*) e *RHEL* (*Red Hat Enterprise Linux*), incluem uma interface gráfica para administração do Samba. Entretanto, o suporte técnico prestado por essas empresas a seus produtos é, obviamente, pago. Lançaremos mão, aqui, de uma solução mais “crua”, porém

Listagem 1: Samba-enredo

```

01 [global]
02  workgroup = REPUBLICATINGA
03  server string = Samba 3.0.14a
04  passdb backend = smbpasswd:/etc/samba/smbpasswd
05  username map = /etc/samba/smbusers
06  name resolve order = wins bcast hosts
07  printcap name = CUPS
08  printing = CUPS
09  logon drive = H:
10  logon script = scripts\logon.bat
11  logon path = \\chefe\profile\%U
12  domain logons = Yes
13  preferred master = Yes
14  wins support = Yes
15  add user script = /usr/sbin/useradd -m '%u'
16  delete user script = /usr/sbin/userdel -r '%u'
17  add group script = /usr/sbin/groupadd '%g'
18  delete group script = /usr/sbin/groupdel '%g'
19  add user to group script = /usr/sbin/usermod -G '%g' '%u'
20  add machine script = /usr/sbin/useradd -s /bin/false -d /var/lib/nobody '%u'
21
22 [homes]
23  comment = Pastas pessoais
24  valid users = %S
25  read only = No
26  browseable = No
27
28 [printers]
29  comment = Gráfica Tinga
30  path = /var/spool/samba
31  printable = Yes
32  guest ok = Yes
33  browseable = No
34
35 [netlogon]
36  comment = Netlogon
37  path = /servidor/netlogon
38  valid users = %U
39  read only = No
40
41 [profile]
42  comment = Perfis de usuário
43  path = /servidor/profiles
44  valid users = %U
45
46  create mode = 0600
47  directory mode = 0700
48  writable = yes
49  browsable = no
50
51 [geral]
52  comment = Depósito da República Tinga
53  path = /servidor/geral
54  read only = No

```

ainda eficaz: instalaremos o Samba em um PC com Linux recém-instalado e o configuraremos manualmente.

Distribuições como SUSE e Red Hat distribuem o Samba em um único pacote RPM que acomoda todos os componentes a serem instalados. Instalar o Samba com esse pacote completo, com um `rpm -ivh samba-xxx.rpm`, não deverá causar nenhum dano ao sistema existente. Quem usa o Gentoo pode instalar um Samba já meio configurado, bastando empregar as *USE Flags*. O comando `USE="cups pam quotas readline winbind"` emerge samba dá conta do recado.

Afinando a cuíca

Antes de colocar a mão na massa, precisamos criar um arquivo de configuração para o servidor Samba, mais precisamente no diretório `/etc/samba/smb.conf`. Após a instalação dos pacotes, possivelmente esse arquivo já existirá e trará em seu interior numerosos e úteis exemplos muito bem comentados. Não destrua, apague nem modifique esse arquivo! Ele é uma fonte de consulta valiosíssima! Em vez disso, renomeie-o para algo como `/etc/samba/smb.conf.orig` ou coisa que o valha. Depois, crie um novo arquivo `/etc/samba/smb.conf` e digite nele o código da [listagem 1](#).

O arquivo de configuração divide-se em diferentes seções, algumas fixas e outras auto-nomeadas. Uma delas é a seção `[global]`, para configurações globais. As opções detalhadas utilizadas na configuração serão explicadas e descritas no decorrer deste tutorial. O que precisamos, para começar, é que pelo menos um arquivo de configuração com o básico esteja presente para que o Samba funcione.

Quem tiver pressa de conhecer detalhes sobre a configuração do arquivo `smb.conf`, suas seções e opções, pode consultar a página de manual: com o comando `man smb.conf`. Mesmo que você não seja um

dos “apressadinhos”, aconselhamos fortemente uma leitura atenta desse manual. É deveras esclarecedor.

Para verificar se você digitou tudo certo, o Samba oferece a ferramenta `testparm`, que mostra erros dentro do arquivo de configuração. Essa ferramenta é bastante útil para encontrar erros sempre que criamos ou modificamos alguma coisa. Use-a!

Esquentando os tamborins

Os clientes Linux e Windows® devem interagir em um domínio. O domínio é administrado por um computador com Linux e Samba instalados. A administração de domínios significa que o servidor Samba disponibiliza o nome do computador e os endereços IP, os nomes dos usuários e suas senhas, bem como as permissões. As conexões dentro da rede com Samba precisam saber de onde e para onde devem seguir. Nesse caso, é necessária uma resolução de nomes dentro do domínio, de forma que o nome de rede de um computador possa ser convertido em seu endereço IP. Há várias maneiras de fazer isso, sendo mais comum a operação de um DNS (*Domain Name Server* – Servidor de Nome de Domínio). Contudo, isso significa um gasto extra em instalação e início de operações. Porém, o Samba oferece suporte para WINS (*Windows Internet Naming Service*), cuja resolução de nomes ocorre de maneira fácil e prática com a ajuda do arquivo `/etc/hosts`.

Suponhamos que em uma rede fictícia devam operar um cliente Linux, um cliente Windows® e uma impressora laser (ver **figura 1**). O servidor deve se chamar *chefe* e os clientes Linux e Windows® *lin01* e *win01*, respectivamente; a impressora foi instalada na cozinha. Todos esses aparelhos têm uma placa de rede e mantêm um endereço IP pelo qual podem ser acessados na rede. Como resultado, para que o servidor Samba possa resolver os

endereços de IP pelo WINS em nomes e vice-versa, eles precisam constar do arquivo `/etc/hosts`:

```
[...]
192.168.100.1   chefe
192.168.100.10 lin01
192.168.100.20 win01
192.168.100.30 cozinha
[...]
```

Edite esse e os demais arquivos deste tutorial com seu editor de textos preferido. Para editar esses arquivos, bem como desempenhar as tarefas que veremos mais adiante, precisamos possuir permissão de *root*. A maneira mais fácil é iniciar uma sessão como usuário *root*. Se você possuir o programa *sudo* instalado e souber usá-lo, executar comandos como *root* é bem mais simples (para qualquer dúvida: `man sudo`).

Os computadores registrados no arquivo `/etc/hosts` devem estar disponíveis na rede com os nomes definidos. Para testar, numa linha de comando do servidor Samba digite:

```
ping nome do computador
```

Veja se o comando `ping` comprova (ou não) que existe uma máquina com esse nome na rede e que ela está respondendo. Caso algum computador não responda,

o arquivo `/etc/hosts` pode conter um erro. Mas atenção! O problema também pode ser, simplesmente, que o cabo de rede do computador que não responde está desconectado, quebrado ou aquela máquina está desligada. Numa rede, as possibilidades de problemas crescem exponencialmente.

Concentração na avenida

O Samba – infelizmente, diga-se de passagem – possui um banco de dados de usuários diferente do usado pelo Linux. O sistema operacional usa o arquivo `/etc/passwd` para armazenar os nomes dos usuários, seu shell de login, seu número de usuário e grupo (UID e GID) e até sua senha. Já o Samba guarda seus usuários e senhas no arquivo `/etc/samba/smbpasswd`, cuja sintaxe é bem diferente. Se pudéssemos criar apenas usuários no Samba para que o servidor funcionasse tudo estaria bem. Mas, infelizmente, temos que criar usuários no Samba e no Linux para que a coisa funcione.

Para sincronizar as senhas no Linux e no Samba sem que seja preciso criar cada usuário duas vezes, o Samba pode trabalhar com alguns bancos de dados e possui diferentes formas de conexão a eles. A dobradinha mais usada hoje em dia é formada pelo Samba na linha de frente e o OpenLDAP na marcação. Entretanto, tanto essa como

Listagem 2: Ala das baianas

```
01 Account Operators (S-1-5-32-548) -> -1
02 Administrators (S-1-5-32-544) -> -1
03 Backup Operators (S-1-5-32-551) -> -1
04 Sala de Estar (S-1-5-21-4173815429-3277392822-1971295425-3007) -> salaestar
05 Domain Admins (S-1-5-21-4173815429-3277392822-1971295425-512) -> root
06 Domain Guests (S-1-5-21-4173815429-3277392822-1971295425-514) -> nobody
07 Domain Users (S-1-5-21-4173815429-3277392822-1971295425-513) -> users
08 Guests (S-1-5-32-546) -> -1
09 Power Users (S-1-5-32-547) -> -1
10 Print Operators (S-1-5-32-550) -> -1
11 Replicators (S-1-5-32-552) -> -1
12 System Operators (S-1-5-32-549) -> -1
13 Users (S-1-5-32-545) -> -1
```

outras soluções (usuários no MySQL, por exemplo) se destacam fortemente pela sua complexidade.

A solução mais simples nem sempre é a pior – especialmente para redes pequenas. Um dos métodos é usar o arquivo `smbpasswd` já citado (**listagem 1**, linha 4), que dá conta do recado de forma eficiente para poucos usuários. Contudo, o sistema Samba relaciona os usuários do arquivo `smbpasswd` com os já cadastrados em `/etc/passwd`. O “duplo cadastro” é necessário, portanto – mas não é nada que assuste numa rede doméstica ou numa pequena empresa com poucos usuários. Lembre-se, apenas, de que o usuário tem que ser criado **primeiro** no arquivo `/etc/passwd` e, só depois, no arquivo `smbpasswd`. Se assim não for, o usuário não conseguirá se conectar ao sistema, ou conseguirá mas não terá acesso aos recursos – pastas compartilhadas e impressoras.

Por padrão, o administrador do Linux é o usuário `root`. Como ele obviamente já existe no Linux, vamos criar um `root` no Samba para que ele seja o administrador também. Digite:

```
smbpasswd -a root
```

Esse comando cadastra o `root` no banco de dados de usuários do Samba. Por motivos de segurança, o comando pede em seguida a senha correspondente, que pode (e deve!) ser diferente daquela do `root` local no Linux. O usuário adicionado dessa forma no banco de dados de usuário do Samba é o administrador Windows®, que será necessário mais tarde para a configuração dos clientes Windows®. Para que ele seja reconhecido também com o nome de *administrador* nas máquinas com Windows (e não como `root`), deve ser feito um mapeamento dos nomes de usuário – para quem não sabe, no Windows o superusuário chama-se *administrador* (ou *administrator*, no Windows em

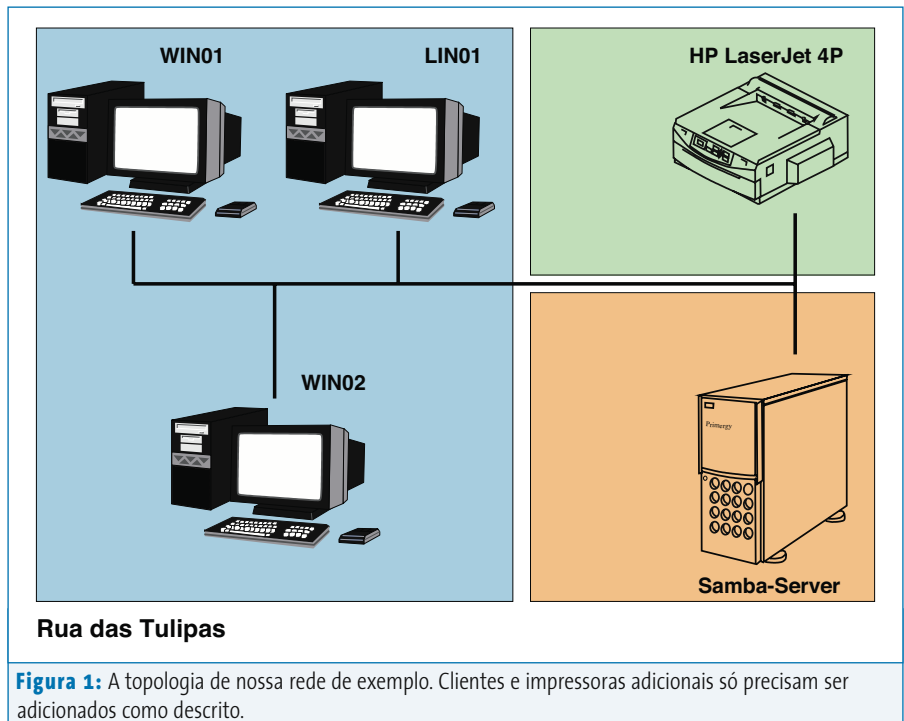


Figura 1: A topologia de nossa rede de exemplo. Clientes e impressoras adicionais só precisam ser adicionados como descrito.

inglês). Caso ainda não esteja disponível, crie o arquivo `/etc/samba/smbusers` (**listagem 1**, linha 5) e, com seu editor de textos favorito, insira nele a linha `root = Administrator`.

Conforme mencionado, a ferramenta `smbpasswd` utiliza os usuários já cadastrados no arquivo `/etc/passwd` do Linux. Para um novo usuário entrar no Samba, é necessário primeiro criá-lo também no Linux. É possível fazê-lo rapidamente com o comando:

```
useradd joseusuario -m -G users
```

Para verificar se o usuário foi realmente criado no Linux, use o comando `vipw`. O novo usuário pode ser encontrado no final da lista, e a partir de agora é um membro do grupo `users`. Para ele foi criado (com a opção `-m`) um diretório pessoal em `/home/joseusuario`. Ainda faltam as senhas. De certa forma, o plural aqui está correto, já que há uma clara diferença entre os bancos de dados individuais do Linux e do Samba. Para definir a senha do senhor Zé Usuário no Linux, digite:

```
passwd joseusuario
```

A senha é gravada em `/etc/passwd`. Agora falta gravar a senha do Samba:

```
smbpasswd -a joseusuario
```

Pronto! O usuário e a senha estão cadastrados em `/etc/samba/smbpasswd`. Na prática, o que conseguimos com esse trio de comandos é que nosso amigo Zé Usuário pode “se logar” tanto no Windows® quanto no Linux com o mesmo usuário.

Início do desfile

Para acessar qualquer recurso, como ler e gravar dados dentro do servidor ou imprimir nas impressoras da rede, é necessário definir permissões, chamadas de *direitos* para os usuários do Samba. Também é possível agrupar esses usuários em *grupos* e conceder os direitos ao grupo todo.

O Samba não possui uma administração de grupos própria e usa o recurso do próprio Linux em `/etc/group` – ainda bem, senão seria mais um item para ad-

ministrar em duplicata! Como nem tudo são flores, também neste caso é necessário um mapeamento, mais precisamente entre os grupos do Linux e do Windows®. Os grupos servem como cápsulas dos usuários. Normalmente podemos definir qual direito conceder a cada usuário. Mas também é possível definir os mesmos direitos a um grupo composto por vários deles. A composição do grupo é arbitrária, ao gosto do administrador.

O mapeamento entre os grupos padrão do Windows® e os do Linux é essencial para que as estações de trabalho pensem que o Samba é na verdade um servidor Windows NT. Para conseguir a façanha, os comandos apropriados são:

```
net groupmap modify ntgroup="Domain Admins" 2
unixgroup=root
net groupmap modify ntgroup="Domain Users" 2
unixgroup=users
net groupmap modify ntgroup="Domain Guests" 2
unixgroup=nobody
```

Muitos administradores dividem a rede por espaços físicos. Numa pequena empresa, por exemplo, que funcione numa casa

antiga, o departamento de vendas está na sala, o financeiro na despensa (embaixo da escada e com um pilar no meio do cômodo) e o “patrão” no quarto. Se sua empresa for parecida, que tal atribuir um grupo a cada cômodo da casa? Isso pode ser útil quando o PC desse cômodo é utilizado por mais do que uma pessoa.

Outro exemplo, mais prático: o administrador de uma república de estudantes cria, com o comando `groupadd salaestar` um grupo no Linux para o computador instalado na sala. Depois, cria o mapeamento correspondente para o Samba com:

```
net groupmap add ntgroup="Sala de Estar" 2
unixgroup=salaestar type=d
```

Em uma república de estudantes, uma concepção de segurança exagerada pode deixar os moradores com os nervos à flor da pele; em contrapartida, quando se trata de uma pequena empresa, faz muito sentido separar “o financeiro” de outros departamentos por meio de grupos. É preciso atentar para o fato de que os grupos do Windows® não diferenciam

letras maiúsculas e minúsculas, enquanto o Linux faz essa diferenciação. Além disso, o Linux não tolera espaços nos nomes de grupo.

Um teste para comprovar se o mapeamento está correto pode ser feito com o comando `net groupmap list | sort`. A listagem 2 mostra a saída do comando. Ela mostra que os grupos do Windows® indicam por meio de uma seta seus grupos mapeados do Linux. As colunas criptografadas de números com um `S` no início são os SIDs (identificadores de segurança). Eles são comparáveis à resolução de nomes de um DNS, que pega o endereço IP e devolve o nome do computador correspondente – e vice-versa. Em nosso caso, a resolução é entre os SIDs e os grupos.

Para adicionar um usuário a um grupo recém-criado (por exemplo, o *Vaca*, da Engenharia Civil do CEFET, se mudou para a nossa república), utilize o comando:

```
usermod -G salaestar vaca
```

O administrador pode verificar se a inscrição do usuário foi realizada com sucesso em qualquer grupo com o comando `vigr`. Sob o grupo `salaestar` somente o usuário *vaca* deve ser encontrado.

Listagem 3: Apuração

```
01 Domain=[REPUBLICATINGA] OS=[Unix] Server=[Samba 3.0.14a]
02
03      Sharename      Type      Comment
04      ---            --        ---
05      netlogon        Disk      Netlogon
06      geral           Disk      Depósito da República Tinga
07      IPC$            IPC       IPC Service (Samba 3.0.14a)
08      ADMIN$          IPC       IPC Service (Samba 3.0.14a)
09      grafica         Printer   Gráfica Tinga
10
11 Domain=[REPUBLICATINGA] OS=[Unix] Server=[Samba 3.0.14a]
12
13      Server          Comment
14      ---            ---
15      CHEFE           Samba 3.0.14a
16
17      Workgroup       Master
18      ---            ---
19      REPUBLICATINGA   CHEFE
```

Mestre-sala e porta-bandeira

Com o que foi feito até agora, a base já foi preparada e o servidor Samba pode entrar em operação. O Samba é iniciado como daemon em segundo plano. Dependendo da distribuição, o nome do script inicial em `/etc/init.d` pode variar. O script do SUSE se chama `smb`; no Gentoo e no Debian toma o nome de `samba`. Na versão utilizada no Gentoo e no Debian, o administrador inicia o servidor `samba` com o comando `/etc/init.d/samba start`. Além do daemon `smbd` do Samba, ele inicia também um daemon `nmbd`, que serve para a resolução de nomes. Porém, para que ela seja

executada de maneira correta, é necessário adicionar ainda mais um item em `/etc/nsswitch.conf`:

```
hosts: files wins
```

Essa alteração (no `/etc/nsswitch.conf`) não é obrigatoriamente necessária no servidor – afinal, ele é o próprio o servidor WINS e não precisaria saber de si mesmo para que a resolução de nomes funcionasse. Entretanto, a resolução WINS deve ser ativada também no servidor, para manter a coerência entre as configurações de todas as máquinas. Além disso, é fundamental que um cliente Linux dependa somente de um único serviço do `/etc/nsswitch.conf` para encontrar todos os computadores e impressoras na rede.

Muito importante também é um outro daemon: o daemon Winbind possibilita o mapeamento de usuários e grupos entre Linux e Windows. Da mesma forma que qualquer outro serviço, inicia-se também com um script inicial (`/etc/init.d/winbind start`). Se tal script inicial não existir, pode-se iniciar o Winbind também diretamente do console com `winbindd` e, eventualmente, finalizá-lo como root com `killall winbindd`.

Para verificar se os *daemons* necessários estão no ar, use os comandos:

```
ps aux | grep smb
ps aux | grep nmbd
ps aux | grep winbind
```

Pode-se testar se o Samba também reage corretamente às pesquisas com o seguinte comando, emitido no próprio servidor:

```
smbclient -L localhost -U%
```

O resultado para o nosso exemplo é apresentado na **listagem 3**. Para conectar-se ao diretório pessoal do Zé Usuário, digite:

O galpão do carnavalesco

O CUPS (*Common Unix Print System*, Sistema de Impressão Comum Unix [3]) é uma das formas mais populares para enviar comandos para uma impressora. Um dos pontos fortes é o sistema *Foomatic* [4]. O Foomatic é um banco de dados com drivers de várias impressoras para o CUPS. Aqueles que querem operar impressoras exóticas ou muito antigas também podem desenvolver seus próprios drivers, porém, a maioria deles pode ser encontrada no Foomatic.

Antes de tudo, o CUPS precisa ser instalado. Este pacote está disponível em quase todas as distribuições Linux, sendo que o SUSE o separa em pacotes de cliente, servidor, desenvolvedor e drivers. Além dos pacotes de desenvolvedor, vale a pena instalar também o pacote *foomatic-filters*. Não é necessário instalar os pacotes de desenvolvedor e de servidor nos clientes Linux. Nesse caso, basta um comando `rpm -ivh foomatic-xxx.rpm` (*xxx* é a versão do pacote, o número exato depende da distribuição). Os usuários do Gentoo executam a instalação definindo a *USE Flag* `samba`, compilam o CUPS com `emerge cups`, e em seguida instalam o banco de dados Foomatic com `USE="ppds" emerge foomatic`.

O arquivo de configuração do CUPS é especialmente bem-documentado. Neste exemplo, os itens para iniciar da primeira vez serão assim:

```
ServerName chefe
DocumentRoot /usr/share/cups/docs
User lp
Group lp
SystemGroup lp
LogLevel info
Port 631
```

```
smbclient //chefe/zeusuario -U zeusuario
```

lembrando que *chefe* é o nome, na rede, do servidor Samba. Já para ver o diretório pessoal do Vaca, digite:

```
smbclient //chefe/vaca -U vaca
```

O comando solicita a senha do usuário. Após a autenticação, o administrador tem acesso ao servidor. Se for um usuário comum, terá acesso à sua área pessoal e às pastas compartilhadas no servidor que pertençam a seu grupo.

```
<Location />
Order Deny,Allow
Deny From All
Allow From 127.0.0.1
Allow From 192.168.100.*
</Location>
```

```
<Location /admin>
AuthType Basic
AuthClass System

Order Deny,Allow
Deny From All
Allow From 127.0.0.1
Allow From 192.168.100.1
</Location>
```

Com esta configuração inicia-se o CUPS:

```
/etc/init.d/cups start
```

As diretivas de permissão servem em primeiro lugar para o acesso geral à impressora configurada com o CUPS. Em segundo lugar, possibilitam o acesso à interface web para configuração, permitindo-o exclusivamente ao próprio servidor. Além da interface web, o CUPS pode ser administrado e configurado com as ferramentas de console. Certamente a opção mais simples é a interface web, que o administrador acessa em um navegador com o endereço `http://chefe:631`. Por motivos de segurança, o CUPS solicita um usuário que, a menos que seja especificado de outra forma na configuração, é o root.

Na interface web é possível adicionar novas impressoras por meio de um assistente, que especifica a conexão para elas e escolhe os respectivos *drivers*.

A seção `[homes]` da configuração do Samba na **listagem 1** (até a linha **22**) não terá nenhum local específico sem a opção `path`, na qual os diretórios pessoais são encontrados. Essa é uma característica do Samba pois, caso nada seja especificado, os diretórios podem ser acessados por padrão com `//nome do servidor/nome do usuário`, o que possibilita ao usuário do Linux o acesso a seu diretório pessoal. No Microsoft Windows® essa operação é feita primeiramente com uma conexão do cliente Windows® no

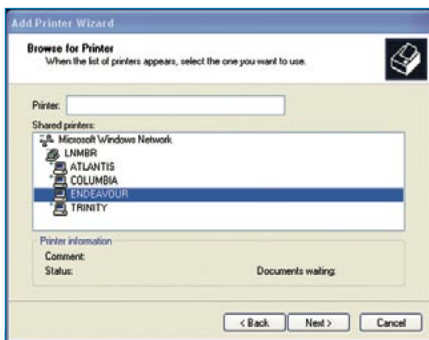


Figura 2: É fácil adicionar uma impressora ao cliente Windows®. De acordo com a configuração do Samba e do CUPS, o Assistente solicitará a escolha de um driver ou utilizará o do servidor CUPS.

domínio Samba. Esse procedimento está descrito no quadro **Tem gringo no Samba**, à página 52

Passistas

Além dos arquivos pessoais em seu diretório *home*, pode-se querer de vez em quando trocar arquivos na rede. Para isso, são necessárias recursos aos quais todos os usuários têm acesso. Esses recursos estão definidos na **listagem 1** a partir da linha 50. O diretório que deverá conter os arquivos compartilhados localiza-se abaixo de `/servidor/geral` e, logicamente, precisa estar com as permissões de acesso (no Linux!) liberadas apropriadamente. Além disso, o administrador precisa limitar o acesso ao grupo *users* e permitir o acesso de gravação no diretório. Tudo isso terá a seguinte cara:

```
mkdir -p /servidor/geral
chown -R root:users /servidor/geral
chmod -R ug+rw,or+x-w /servidor/geral
```

Agora, o administrador e o grupo *users* terão acesso de gravação. Todos os outros usuários, tanto sob Windows® quanto sob Linux, terão acesso negado. Um usuário terá acesso a essa habilitação com o comando:

```
smbclient //chefe/gemeinsam -U usuário
```

Da mesma forma é o comportamento com habilitações para o acesso à rede e aos perfis (**listagem 1**, a partir da linha 35) que precisam ser administrados de acordo com os mesmos direitos de acesso. Dessa forma, o acesso à rede apresentará uma área na qual ficarão os scripts de logon. O próprio script será determinado na opção `logon script` em `/etc/samba/smb.conf` (ver **listagem 1**, linha 10). Com essa opção, um script é especificado com relação a um caminho na seção `[netlogon]`. O script será copiado em um logon no cliente Windows® e listado ali. Servem para os scripts de logon arquivos de lote (*batch*) exclusivos com a extensão `.bat`. Eles devem ser criados no Windows® para que não haja conflitos de codificação de caracteres (o Linux usa UTF-8 e o Windows® um dialeto de UTF-16). Como exemplo, esse script pode ter o seguinte conteúdo:

```
net use s: \\chefe\geral
```

No caso de logon, a habilitação utilizada comumente será incluída sob o *drive S:* e será utilizável na área de trabalho do Windows®.

Os perfis são diretórios nos quais o Windows® armazena as configurações de cada usuário. Entre elas estão os ícones na área de trabalho, as configurações de impressoras e correio eletrônico e outros. O caminho no qual estas configurações devem ser armazenadas é definido pelo administrador com a opção `logon path` (ver **listagem 1**, linha 11). Caso ele não seja definido, o Samba grava os perfis no diretório pessoal do usuário. Para guardá-los de maneira centralizada, é necessário que uma habilitação para os perfis esteja disponível (**listagem 1**, a partir da linha 41). A opção na linha 44 permite o

acesso somente aos usuários que o Samba conhece. Isso ocorre por meio de variáveis de ambiente disponibilizadas pelo Samba (mais detalhes na página de manual do `smb.conf`). As linhas 45 e 46 têm grande importância para o acesso: os perfis criados nelas somente serão acessíveis por um único usuário, ou seja, aquele ao qual o perfil pertence.

O porta-estandarte

Antes que seja possível imprimir na rede de maneira centralizada, vamos precisar de um servidor de impressão. Para saber como implementar um, veja o quadro **O galpão do carnavalesco**.

O cliente Linux necessita somente do pacote `cups-client`. O arquivo de configuração é o `/etc/cups/client.conf`, no qual a opção `ServerName` – normalmente comentada para não ter efeito – indica qual o servidor de impressão. Em nosso exemplo:

```
ServerName chefe
```

O serviço de impressão não precisa ser iniciado no cliente. Os comandos gerais de impressão no pacote do cliente ajudam a encontrar o servidor. Isso vale também para comandos de impressão como `lpr -Pnome_impressora festa_de_embaio`.

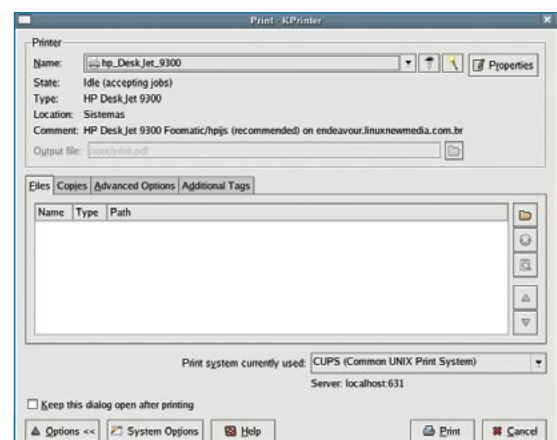


Figura 3: O programa de impressão *kprinter* é muito fácil de operar e oferece todas as possibilidades de configuração para diferentes sistemas de impressão. Ele também converte dados para o formato Postscript quando se tratar de um texto em ASCII.

pdf. O comando do serviço de impressão LPRNG emula o servidor CUPS. Os programas de impressão do KDE, como o *kprinter*, oferecem uma opção de serviços de impressão, entre os quais pode se escolher também o CUPS (ver **figura 3**).

Em contrapartida, um usuário do Windows® somente pode adicionar uma impressora se o cliente estiver casado com o domínio (veja no quadro **Tem gringo no Samba**). Caso isso ocorra, o usuário pode escolher a impressora por meio da opção *Impressoras e Aparelhos de Fax* do menu Iniciar e, depois, *Adicionar Impressora* (ver **figura 2**). Caso o servidor Samba não encontre nenhum *driver* para a impressora, o programa de instalação da impressora solicitará um. Se quiser manter os drivers atuais dos clientes, pode-se incluir essa configuração explicitamente com a opção `use client drivers = Yes` na seção `[printers]` do arquivo de configuração do Samba.

Apoteose

Quem quiser adicionar computadores à rede da **figura 1** poderá fazer isso num piscar de olhos. No caso do Windows®,

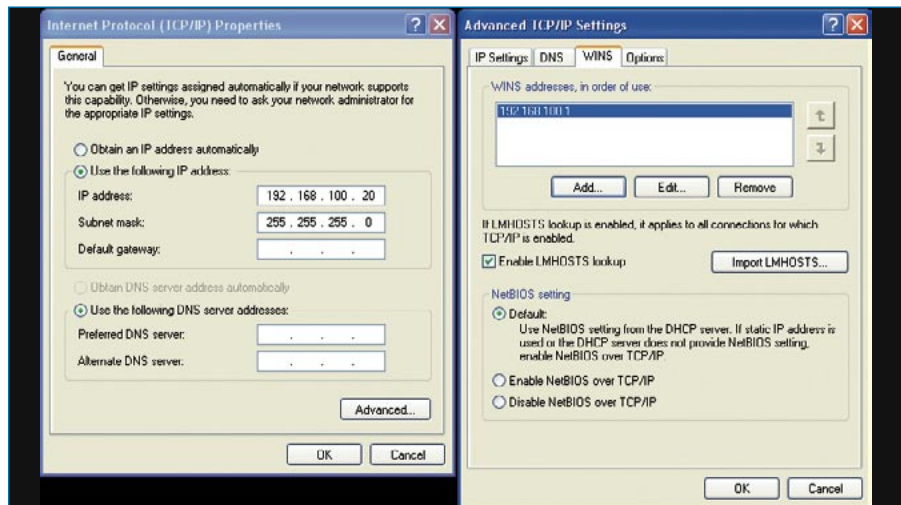


Figura 4: Para se comunicar em rede, o computador precisará de um endereço IP. O cadastro do servidor WINS serve a resolução de nomes pelo Samba. Para que o computador acesse a Internet, indica-se um *gateway* acessível.

é necessário que os novos computadores a adicionar estejam disponíveis no servidor Samba em `/etc/hosts` para que o WINS possa resolvê-los na rede. Para Linux, deve-se configurar a resolução de nomes em `/etc/nsswitch.conf`, conforme descrito anteriormente. Um cliente Windows® encontrará seu novo lar, que também está descrito no quadro “Tem

gringo no Samba”. Como última dica, o próprio Samba possui uma interface web para sua configuração. Ela se chama SWAT (*Samba Web Administration Tool*) e, embora não a tenhamos usado em nosso tutorial, ele facilita em muito a administração de um servidor Samba. Para saber mais sobre ela, visite o site oficial [1] do Samba. ■

Tem gringo no Samba

Antes que um computador possa ser adicionado a um domínio da rede, é necessário que este tenha acesso à rede local. Esse acesso é possível após o cadastro de um endereço IP e de um servidor WINS para resolução de nomes. O administrador precisa informar sua senha para entrar na interface gráfica correspondente para essa configuração. Pelo painel de controle, item conexões de rede, verifica-se se a placa de rede foi reconhecida – ou seja, se seu driver foi corretamente instalado. A caixa de diálogo permite ajustar a conexão TCP/IP à rede; para isso cadastre corretamente o endereço IP e o servidor WINS. Veja a **figura 4**.

Para adicionar o computador ao domínio REPUBLICATINGA, clique com o botão direito do mouse na *Área de Trabalho* e escolha a opção *Propriedades*. Atenção: você precisa de direitos de administrador para isso. A aba *Nome do Computador* contém o botão *Alterar* e novamente um campo *Nome do Computador*, no qual aparece o nome, na rede, dessa estação de trabalho (no exemplo, `win01`). Ao clicar em *Domínio* e *Nome do Domínio*, o Windows® procura o caminho para o servidor Samba. O novo computador é cadastrado, com a opção `add machine script`, da **listagem 1**, no arquivo `smbpasswd`, e o adiciona dessa forma ao domínio.

A partir de agora é possível acessar o cliente Windows com o usuário cadastrado no servidor Samba. Na *Área de Trabalho*, as conexões de rede mostram um drive `H:`, o diretório pessoal (`/home`) do usuário. Todos os dados armazenados ali serão enviados direto para o servidor Samba.



Lantejoulas, plumas e tecidos

O material usado em nosso desfile foi:

- `net-fs/samba-3.0.14a`
- `net-print/cups-1.1.23-r1`
- `net-print/foomatic-3.0.2`

SOBRE O AUTOR

Markus Klimke é engenheiro de sistemas no Centro de Processamento de Dados da Universidade Técnica de Hamburgo-Hamburgo, na Alemanha, e é responsável pela integração de sistemas.

INFORMAÇÕES

- [1] Samba (em inglês): www.samba.org
- [2] Etherreal (em inglês): www.ethereal.com
- [3] CUPS (em inglês): www.cups.org
- [4] Impressão em Linux (em inglês): www.linuxprinting.org