BROMS: Gestão Uniforme de um Parque Computacional Multi-Plataforma

António Coutinho António Luís Sousa Carlos Baquero Francisco Moura
José Pedro Oliveira José Orlando Pereira

Departamento de Informática, Universidade do Minho
Largo do Paço, 4709 Braga Codex

{ajc,als,cbm,fsm,jpo,jop}@di.uminho.pt
http://gsd.di.uminho.pt/

Resumo

O crescimento dos parques de máquinas pessoais levanta consideráveis problemas de administração, contrastando com o que o ocorre com recursos centralizados. Nenhuma das soluções existentes para o efeito apresenta um compromisso aceitável entre a liberdade de configuração que se espera de uma máquina pessoal e o controlo eficiente dos recursos resultante de uma gestão centralizada. Neste contexto propõe-se uma solução deste dilema através da coordenação de um sistema de *boot* remoto avançado com um conjunto de serviços de rede. A aplicação deste sistema à gestão e manutenção de laboratórios pedagógicos demonstrou que se pode assim criar um ambiente de ensino muito mais fiável e flexível do que o tradicional.

1 Introdução

Ao longo dos últimos vinte anos decorreram várias transformações radicais na organização dos recursos computacionais no seio das organizações. Uma das mais expressivas consistiu no crescimento dos recursos computacionais ao nível dos utilizadores ou grupos de utilizadores, a qual foi acompanhada de alguma erosão nas competências dos serviços computacionais centralizados.

Esta transformação foi sem dúvida importante para a disseminação dos meios de computação e veio conferir uma maior autonomia e dinamismo aos utilizadores. Nas fases pioneiras da instalação dos meios informáticos pessoais, para um grupo considerável de organizações, foi sendo negligenciada a necessidade crescente de apoio a um parque computacional fisicamente distribuído. Esta falta de preocupação ocorreu sempre que a iniciativa de instalação de pequenas plataformas ocorria por parte dos utilizadores, ficando estes com o ónus da administração das máquinas assim introduzidas.

Actualmente, porém, a ubiquidade deste tipo de configurações tem vindo a evidenciar a dificuldade estrutural em gerir racionalmente um parque descentralizado. Além disso, os sistemas operativos comuns para computadores pessoais são tradicionalmente voltados para a administração por parte dos utilizadores, herança da fase pioneira, não oferecendo suporte adequado a uma gestão uniforme e centralizada. Estes sistemas carecem também de protecção adequada contra actos erróneos de configuração, que possam ser involuntariamente cometidos pelos utilizadores. A título de exemplo veja-se como é comum a desconfiguração, muitas vezes irreversível, de sistemas Windows por parte de utilizadores.

No caso particular das instituições de ensino em que a formação em informática é um dos seus objectivos primários, a gestão de laboratórios pedagógicos é muitas vezes um campo de batalha no que respeita à manutenção da operacionalidade dos sistemas. Até porque para aprender é necessário experimentar e ao experimentar é inevitável desconfigurar quando tal não é expressamente protegido pelo sistema de exploração em uso.

Reside aqui a motivação para a procura de um sistema de gestão de parques computacionais descentralizados que permita um uso eficiente dos recursos humanos (em termos de técnicos de administração de sistemas) por forma a minimizar a sua alocação a tarefas repetitivas do tipo "configurar/testar/reconfigurar". Tarefas estas que se multiplicam pelos vários postos de trabalho onde ciclicamente vão surgindo os mesmos problemas.

A solução aqui proposta passa pela adaptação das plataformas compatíveis por forma a tratar o disco local como uma cópia potencialmente volátil de uma imagem que é gerida centralizadamente e que se adapta ao *hardware* específico de cada máquina. A solução de gestão irá também conduzir os utilizadores à manutenção dos seus dados num sistema remoto sobre controlo centralizado o que vem endereçar o problema da manutenção de uma disciplina de *backups* eficiente.

Tendo sido avançados apenas alguns dos tópicos da solução explorada, dar-se-á curso à sua apresentação na secção 3, precedida por uma panorama geral das propostas existentes. Na secção 4 é descrita a aplicação desta solução à gestão do parque computacional do Departamento de Informática da Universidade do Minho (DI/UM) e a sua integração na infra-estrutura de rede com protecção de domínios por *firewalls*. A secção 5 conclui esta exposição.

2 Panorama dos Sistemas Existentes

2.1 Sistemas "Chave na Mão"

A gestão de parques computacionais depende em primeiro lugar do tipo de utilizadores para o qual é disponibilizado, pelo que é necessário averiguar quais os requisitos em cada caso, avaliando seguidamente, em que medida cada uma das soluções existentes satisfaz ou não esses requisitos.

Em primeiro lugar consideram-se equipamentos tais como terminais de ponto-de-venda, quiosques informativos ou quiosques Internet em que a capacidade de configuração dos recursos por parte dos utilizadores finais deve ser extremamente reduzida, ou mesmo nula.

Neste caso, pretende-se assegurar o bom funcionamento do sistema, mesmo na presença de utilizadores potencialmente hostis, proporcionar uma actualização centralizada do *software* e assegurar uma fácil substituição de postos danificados, por exemplo, por clonagem.

As máquinas utilizadas em infra-estruturas de rede, desempenhando diversos serviços, como *proxy* e *firewalls*, são um exemplo semelhante, em que uma configuração pode ser clonada para rapidamente actualizar toda a infra-estrutura.

2.2 Sistemas de Acesso Público

Em segundo lugar temos os recursos computacionais disponibilizados em locais públicos tais como instituições de ensino ou cibercafés em que:

- é disponibilizado um conjunto de recursos homogéneos, tanto de *hardware* como de *software*, num número relativamente elevado de postos de trabalho;
- deve ser permitido que cada utilizador possa instalar recursos particulares de *software* e de informação durante uma sessão de trabalho, sem que isso degrade o desempenho e segurança do sistema para outros utilizadores, ou se traduza numa violação da sua privacidade ou um encorajamento à pirataria de *software*;
- as consequências de acções maliciosas, intencionais ou não, como é o caso da introdução de vírus, bem como os danos resultantes da falta de preparação dos utilizadores ou de erros no *software* utilizado, devem limitar-se à sessão e ao utilizador correspondente.

No caso de utilização em locais como escolas, sobretudo em ambiente de salas de aula, é ainda necessário assegurar uma área de trabalho a cada indivíduo, respeitando a sua privacidade e a integridade dos dados. O acesso a essa área deve também ser independente do posto de trabalho que o indivíduo tem ao seu dispor numa dada altura.

Cada aula necessita ainda recursos de *software* específicos, administrados autonomamente em relação ao resto do sistema, o que deve ser possível sem que isso implique a reserva de recursos para além do tempo em que são de facto utilizados ou comprometa a segurança do sistema.

2.3 Sistemas Personalizados

Em ambientes com necessidades informáticas mais diferenciadas, como em sectores de investigação, desenvolvimento e criatividade o padrão de utilização é diferente, pois:

- a existência de um posto de trabalho fixo torna irrelevante a capacidade dos diversos utilizadores migrarem entre diferentes postos de trabalho;
- existe responsabilização dos indivíduos pelos recursos à sua disposição, o que torna improvável a danificação voluntária dos recursos de *software*;
- torna-se mais importante que cada posto de trabalho possa ter uma configuração diferenciada, por exemplo para evitar pagar múltiplas licenças para um produto que apenas um utilizador necessita.

Se por um lado se torna menos necessária a gestão centralizada como uma medida para impedir que sejam causados danos aos recursos, esta torna-se mais difícil como consequência da maior diversidade de ambientes de trabalho. Há que considerar que embora haja menos hipóteses de se danificar *software*, quando isso acontece como consequência de erros, enganos ou vírus, os custos do tempo de baixa são superiores uma vez que os postos de trabalho não são indiferenciados.

Embora também seja difícil efectuar uma gestão centralizada dos recursos de *software*, esta é mais importante para tarefas como actualização de versões, tanto para assegurar a interoperabilidade dos ficheiros produzidos por vários utilizadores como também para superar falhas conhecidas que comprometam a segurança ou a confiabilidade.

2.4 Cenários Unix

Em termos de sistemas Unix, existem dois cenários possíveis de administração de sistemas: servidores com clientes leves e grupos de estações de trabalho. Estes dois cenários não são mutuamente exclusivos, sendo inclusivé bastante comum a sua coexistência nas organizações.

A utilização de um servidor com um conjunto de clientes leves (i.e. terminais X-Windows) é a solução preferível em termos de administração, permitindo aparentemente satisfazer a totalidade dos requisitos acima apresentados. No entanto, a experiência demonstra que:

- se trata de uma solução não comportável em termos técnicos nem económicos para grupos numerosos de postos de trabalho;
- a instalação de variados pacotes de *software* geridos autonomamente (e.g. por um docente para um curso específico) incorre frequentemente em compromisso da segurança e colisões entre pacotes incompatíveis;
- a necessidade de proporcionar acesso remoto a utilizadores no servidor permite explorar mais facilmente eventuais brechas na segurança;
- a liberdade de cada utilizador está demasiadamente restrita pelas escolhas dos administradores de sistema.

Na prática, esta solução tende a degenerar para uma situação em que existem vários servidores para distribuir a carga e permitir algum autonomia na administração.

Por outro lado, a utilização de estações de trabalho coordenadas utilizando servidores de contas, áreas de trabalho e *software* instalado resolvem em grande parte os problemas da solução centralizada. Com efeito, a utilização do processador de cada estação alivia o servidor central, que pode inclusivé ser desdobrado por tarefa. Além disso, ao correr localmente, o X-Windows remove da rede o tráfego correspondente, o que pode significar mais de 90% da carga de uma rede local típica.

No entanto, este modelo tem ainda algumas limitações. Por exemplo, a gestão de versões de *software* instalado é ainda problemática, uma vez que a actualização do *software* de base das estações de trabalho tem que ser sincronizada.

A liberdade dos utilizadores tem também que ser tão restringida quanto em sistemas centralizados, de modo a garantir que qualquer utilizador tenha acesso a qualquer estação de trabalho.

Repare-se que no caso contrário, em que se admite que os utilizadores possam modificar o sistema de base da sua estação, rapidamente se perde a capacidade de diferentes utilizadores comutarem livremente entre postos de trabalho.

2.5 Cenários Windows

Em termos de sistemas Windows existem também dois cenários distintos:

- postos completamente autónomos ou agregados em redes relativamente *ad hoc*, como resultado da sua história como sistema para computadores pessoais;
- redes de estações de trabalho organizadas de um modo muito semelhante às já descritas em ambientes Unix.

No primeiro caso, cada utilizador tem plena liberdade para instalar *software* e modificar a configuração do seu posto de trabalho. Este facto impossibilita uma gestão centralizada dos produtos instalados. Em ambientes escolares, esta organização incorre necessariamente em custos bastante elevados em manutenção e na frequente indisponibilidade de postos de trabalho.

No segundo caso, embora se consigam vantagens substanciais em termos de administração centralizada, subsistem algumas dificuldades, tal como o difícil compromisso entre liberdade do utilizador e gestão centralizada. Neste caso, a decisão deve pender para o lado da liberdade uma vez que a tradição dos sistemas Windows como sistemas de uso pessoal faz com que os utilizadores ofereçam muito maior resistencia a

uma evolução para menos liberdade de configuração, por exemplo através da utilização do *Zero Administration Kit*.

Isto leva a que seja mais difícil tornar os utilizadores independentes das máquinas. Tecnicamente é até impossível qualquer tipo de restrição de utilização da máquina porque o sistema operativo não o permite ou porque o *software* de aplicação não o tolera.

No entanto há claras vantagens em ter um sistema que permita a re-instalação personalizada e rápida de estações de trabalho mas que possibilite a retenção de estado entre re-instalações, inclusivé com personalizações diversas e em locais diferentes.

3 Concretização

3.1 Caracterização do Sistema

O sistema de *boot* remoto que serviu de base a este estudo foi desenvolvido numa universidade suiça [3], tendo como objectivo principal resolver os problemas de administração dos laboratórios pedagógicos dessa instituição. De facto, um caso particularmente difícil de tratar é o dos computadores utilizados por alunos, que normalmente não se encaixam no perfil de utilizadores cuidadosos, exacerbando os problemas anteriormente citados.

Os objectivos mínimos que se prentende alcançar são os seguintes:

- reduzir ao mínimo o trabalho de administração das máquinas;
- impedir que os utilizadores possam, acidental ou intencionalmente, alterar a configuração da máquina de um modo que prejudique os utilizadores seguintes;
- permitir que cada máquina possa servir para vários sistemas operativos, podendo cada utilizador comutar livremente entre eles;
- criar um ambiente em que os dados de cada utilizador residam num servidor centralizado, e não em cada máquina, permitindo que lhe estejam sempre acessíveis qualquer que seja o laboratório, posto de trabalho, ou sistema operativo em que esteja de momento a trabalhar;
- autenticar os utilizadores antes até de carregar o sistema operativo, de modo a que só as pessoas autorizadas possam fazer seja o que for nessa máquina.

O aspecto mais inovador deste sistema é a relação entre o *boot* remoto e o disco local das máquinas clientes. De facto os sistemas baseados em *bootproms* são muitas vezes desprovidos de disco. Isto garante que os computadores são sempre re-iniciados num estado conhecido, uma vez que não têm estado local persistente. Essas soluções tem no entanto problemas a nível de eficiência uma vez que sobrecarregam a rede, principalmente se se tratar de sistemas operativos com memória virtual, que neste caso precisam de fazer *swap* remoto.

Na solução aqui estudada, os clientes têm disco, mas este é tratado como se fosse uma área de armazenamento volátil. Assim o programa de *boot* pode re-particionar e formatar o disco de cada vez que a máquina é ligada, adaptando-o a diferentes configurações e sistemas operativos. Deste modo, resolve-se o problema de ineficiência, uma vez que se usa o disco local, por exemplo no caso do *swap*, sem se perder as vantagens de ter um sistema inicial sempre em bom estado, já que todas as alterações que se façam aos dados presentes no disco só persistem até ao próximo *reboot*.

As principais características deste sistema são as seguintes:

- cada posto de trabalho tem uma *bootprom* que inicia uma sequência de arranque permitindo ao utilizador a escolha do sistema operativo que pretende;
- quando cada posto de trabalho arranca, aparece completamente "limpo", como se tivesse acabado se ser instalado, perfeitamente configurado e livre de vírus;
- cada computador tem um disco, mas este funciona apenas como armazenamento temporário, uma vez que todos os dados dos utilizadores residem num servidor remoto;
- cada utilizador pode usar qualquer dos postos de trabalho, independentemente do sistema operativo escolhido, e ter sempre acesso à sua área de trabalho residente num servidor;
- só podem utilizar os postos de trabalho os utilizadores que tenham conta no servidor.

Atingem-se portanto todos os objectivos anteriormente enunciados.

3.2 Instalação do *Software* nos Clientes.

A instalação do *software*, incluindo o sistema operativo que irá ser usado nos clientes, faz-se uma só vez para cada conjunto de máquinas semelhantes. Num dos clientes instala-se o sistema e o software pretendido, tendo o cuidado de instalar as partes mais volumosas em armazenamento remoto, onde poderão ser partilhadas por todos os clientes. Estando tudo devidamente instalado e configurado, comprime-se todo o conteúdo do disco, criando uma *imagem* desse sistema. Esse ficheiro é então copiado para o servidor de *boot* e será fornecido a todas as máquinas cujos utilizadores pretendam utilizar esse sistema. Isto significa que para todas as outras máquinas não chega sequer a haver instalação convencional de sistema ou outro *software*, uma vez que recebem à partida uma cópia exacta da instalação padrão. Obviamente que o número de licenças deverá ser compatível com o número de máquinas onde o *software* é replicado.

Torna-se evidente que o tamanho da imagem criada vai reflectir-se na eficiência global do sistema, uma vez que terá muitas vezes que ser transmitida pela rede e descomprimida pelos clientes. Convirá, portanto, que esta seja o mais pequena possível. No caso do sistema Linux isto é fácil de conseguir, dado que as suas grandes possibilidades de configuração permitem fazer um sistema em que quase tudo é importado de rede. No caso de sistemas Windows, a situação é diferente. Apesar de quase todas as aplicações terem opções de instalação em rede, verifica-se que insistem em instalar dezenas de megabytes nas directorias básicas do sistema, fazendo assim aumentar o tamanho da imagem.

3.3 Mecanismo de *Boot*

Numa primeira fase a *bootprom* faz um pedido de serviço DHCP/BOOTP a que responde o servidor de *boot*. Nessa resposta é indicado qual o programa que servirá de ambiente *pre-boot*. Esse programa é então carregado por TFTP e executado. As suas capacidades incluem uma linguagem interpretada que permite a realização de *scripts* simples, a possibilidade de criar e manusear partições e ficheiros em disco, interagir com o utilizador e o uso da API da *bootprom* para operações de carregamento de ficheiros adicionais.

Nesta fase, o utilizador depara-se com um menu que lhe permite escolher entre várias opções de *boot*. Tipicamente tratar-se-á de escolher entre sistemas operativos diferentes, ou optar entre um sistema "novo" ou "usado". No primeiro caso trata-se de um sistema virgem, como se acabado de instalar, e pressupõe a necessidade de carregar pela rede a respectiva imagem e descomprimi-la antes do *boot* propriamente dito. O segundo caso só funciona se numa sessão anterior se tiver instalado o sistema pretendido, bastando agora voltar a utilizá-lo, sem se voltar ao estado inicial. Esta hipótese é perigosa excepto se o utilizador actual for a mesma pessoa que o utilizou anteriormente, pelo que deve ser desencorajada em todos os outros casos.

No que diz respeito à necessidade de carregar a imagem pela rede, há uma optimização importante a considerar. Este sistema precisa de um espaço temporário no disco onde vai guardando a imagem comprimida que lhe chega através da rede. Só quando estiver completa é que esta é descomprimida para a partição adequada. A optimização referida consiste simplesmente em manter no disco a imagem comprimida. Assim, em cada *boot* desse sistema operativo só será necesserário carregar uma nova cópia da imagem se a cópia actual tiver sido corrompida, ou se a original no servidor tiver sido alterada. Deste modo a única operação que normalmente é necessário efectuar para obter um sistema "novo" é a descompressão da imagem que já existe no disco. A área do disco onde são guardadas estas imagens comprimidas é simplesmente o espaço que sobra das partições definidas pelo utilizador, que por isso deverão ser menores do que o tamanho total do disco.

3.4 Adaptação das Imagens às Máquinas

Recordando que os sistemas são instalados nas diversas máquinas a partir da descompressão de uma imagem comum, torna-se fácil perceber que poderão surgir problemas se as máquinas em causa forem muito diferentes umas das outras. Na realidade estes problemas não são tão generalizados quanto poderiam parecer, uma vez que questões como o processador e as dimensões da memória e do disco não interferem com as imagens, excepto no caso de serem insuficientes para o sistema que se pretende usar. Os componentes que podem dar problemas são os periféricos que exigem controladores específicos. Incluem-se neste caso as placas de rede, vídeo e de som.

Do acima exposto resulta que este método se aplica melhor a conjuntos homogéneos de máquinas. Isto é fácil de conseguir quando se trata de comprar um conjunto de máquinas novas. No caso de máquinas já existentes será boa ideia adquirir pelo menos placas de vídeo e de rede iguais para todas. Apesar dos problemas apontados, este método pode conviver com conjuntos heterogéneos de máquinas em certas

condições, por exemplo, quando o sistema a usar é o Linux. De facto, este sistema por ser flexível permite que uma única imagem se adapte a uma enorme variedade de máquinas. Com o Windows 95, o caso é diferente uma vez que o sistema *Plug & Play* tenta reconfigurar a máquina toda sempre que detecta uma pequena alteração de *hardware*, pedindo acesso ao CD de instalação mesmo para componentes que já estão instalados e exigindo vários *reboots*. Mesmo assim consegue-se controlar parte deste comportamento através de manipulações do *registry*.

Resumindo, em máquinas homogéneas qualquer dos sistemas arranca sem problemas, por seu turno máquinas heterogéneas causam frequentemente problemas aos sistemas operativos Microsoft. Mesmo no caso de máquinas heterogéneas existiria sempre a hipótese extrema de ter uma imagem diferente para cada cliente, ou de isolar os ficheiros que precisam de ser diferentes. A solução mais realista é conseguir agrupar as máquinas em subconjuntos homogéneos tendo uma imagem para cada um.

3.5 Variações do Método

O método descrito é bastante extremo, uma vez que se destina principalmente a laboratórios pedagógicos, onde temos realisticamente que admitir que um computador, depois de ser usado, poderá estar completamente impróprio para uso por outra pessoa. Basta pensar em vírus e em todas as armadilhas que se podem deixar num sistema sobre o qual se teve controlo completo. Por isso o método referido acentua a possibilidade de se descomprimir uma imagem nova em cada *reboot*.

Em ambientes mais benignos, podem-se propôr soluções mais moderadas. Um destes casos é o de uma estação de trabalho de um investigador ou funcionário de uma empresa. Uma vez que esse posto é sempre usado pela mesma pessoa, a sua re-instalação completa só será necessária se algo correr muito mal. Pode-se também reservar uma partição no disco que se deixa intacta mesmo quando se volta a instalar o sistema. Neste caso o utilizador pode perfeitamente administrar a sua máquina como quiser, sabendo que se por acaso a puser num estado pouco recomendável, pode num instante restaurar o estado inicial, preservando os dados que tenha guardado na partição persistente. Este cenário é também mais tolerante aos problemas levantados por máquinas heterogéneas, pois se a re-instalação for rara, será mais tolerável fazer uns *reboots* para o Windows se autoconfigurar.

Este método pode também ser usado para *clonagem* rápida de máquinas iguais, mesmo que nunca mais sejam re-instaladas.

4 O Caso DI/UM

4.1 Situação Anterior

Na Universidade do Minho existe uma entidade centralizada, o Centro de Informática (CI), que disponibiliza contas para os alunos, com direito a endereço de correio electrónico e a páginas WWW pessoais. Dispõe ainda de diversas salas equipadas com terminais X-Windows ou com computadores pessoais.

Além disso, o DI possui os seus próprios laboratórios pedagógicos, embora anteriormente fossem à partida classificados ou como laboratórios Unix ou como laboratórios Windows. Estes laboratórios estavam apenas acessíveis durante as aulas. Mesmo assim, no caso dos laboratórios Windows, a liberdade oferecida a cada utilizador permitia alterações na configuração do posto de trabalho, o que por vezes afectava a aula seguinte. Estas alterações introduziam elevados custos de manutenção, obrigando mesmo a re-instalações frequentes de todo o *software* existente.

Mesmo no caso de aulas dadas em terminais X-Windows ligados directamente a máquinas do CI, surgiam problemas, nomeadamente devidos ao facto do servidor estar a ser usado simultaneamente por outros utilizadores que nada tinham a ver com a aula em curso. A colocação de servidores nos laboratórios e o isolamento da rede por uma *firewall* beneficiou a qualidade das aulas mas originou um custo adicional na administração dos servidors, face à existência de contas em várias máquinas.

4.2 Arquitectura da Rede de Laboratórios do DI

Actualmente quase todos os laboratórios do DI podem funcionar como sistemas de acesso público, organizando-se da forma que se pode observar na Figura 1, onde se destacam:

- o servidor de *boot*;
- os servidores de *software*;
- o servidor de áreas de trabalho;
- os *proxies* de HTTP/FTP.

4.2.1 Servidores de *Boot*

Cada um dos grupos disciplinares do DI tem requisitos diferentes quanto ao *software* necessário à prossecução dos seus objectivos pedagógicos, o que levanta

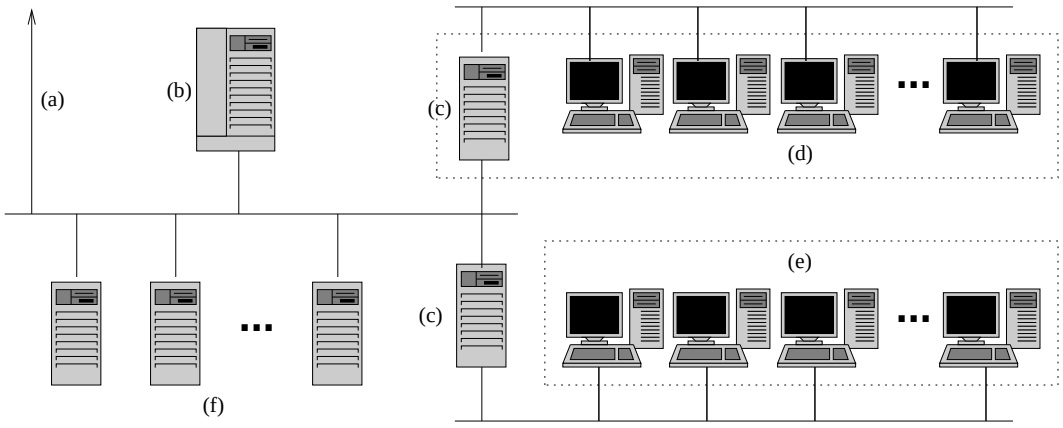

Figura 1: Arquitectura da rede de laboratórios de ensino do DI/UM. (a) Acesso ao exterior; (b) servidor de contas e áreas de trabalho; (c) servidores de *boot*, funcionando também como *firewalls* e *proxies* HTTP/FTP; (d) sala de aula, com o servidor à disposição do docente, para controlo de recursos; (e) sala aberta, com o servidor protegido; (f) servidores de *software* atribuídos aos grupos disciplinares.

problemas no controlo de versões e na gestão do servidor de *boot* que passaria a ser administrado por diversas de pessoas.

De forma a satisfazer os requisitos de um sistema de acesso público e evitar um potencial problema de administração, distribuiu-se a cada grupo disciplinar um servidor. É neste que deve ser instalado o *software* necessário para assegurar o correcto funcionamento das aulas desse grupo disciplinar e que será posteriormente disponibilizado para qualquer dos laboratórios. Desta forma reduz-se o tamanho das imagens que o servidor de *boot* deve enviar para cada um dos clientes.

Cada utilizador pode durante a sua permanência num determinado posto de trabalho instalar e desinstalar os recursos de *software* que pretender, não compromentendo a configuração do sistema para o próximo utilizador dessa máquina.

A minimização das consequências de acções maliciosas é conseguida limitando a possibilidade de estabelecimento de ligações com outras máquinas através da internet. Cada laboratório está isolado numa intranet, cujas ligações possíveis são efectuadas através de uma *firewall* que só permite acesso aos servidores de *software*, áreas de trabalho e *proxy* HTTP/FTP do DI.

4.2.2 Servidor de Áreas de Trabalho

Um componente importante da rede de laboratórios do DI é o servidor de áreas de trabalho e respectivo *backup*. Este servidor alberga as áreas de cada um dos utilizadores dos laboratórios do DI.

A criação das áreas de trabalho é efectuada de uma forma automática, utilizando os cartões magnéticos de estudante e uma base de dados com o nome e número de cada um dos alunos da universidade. Neste processo além da criação de um perfil de utilizador é gerado um termo de responsabilidade em que o aluno se compromete a utilizar correctamente os recursos disponibilizados e uma *password* que apesar de gerada aleatoriamente é probabilisticamente parecida com uma palavra portuguesa, para ser mais fácil de memorizar.

O protocolo utilizado pelo servidor para disponibilizar as áreas de trabalho para os diversos clientes é o SMB. Com este protocolo a importação de uma área de trabalho é validada com um utilizador e respectiva *password* restringindo o acesso, contrariamente ao que seria possível obter com NFS que não permite restringir o acesso a um determinado utilizador mas apenas a um conjunto de máquinas.

4.3 Tecnologia Usada para os Servidores

Todos os servidores referidos estão a correr o sistema operativo Linux. Este sistema tem todas as vantagens comuns ao sistemas Unix e algumas que lhe são específicas. No primeiro caso podem-se destacar a sua eficiência, fiabilidade e flexibilidade, em particular no que diz respeito à administração remota.

No que diz respeito às vantagens específicas do sistema Linux, a maior é a sua natureza *Open Source* [1], que advém do facto de o sistema operativo, bem como todos os utilitários e sistema de desen-

volvimento serem construídos num ambiente aberto à colaboração de todos os interessados, sempre com divulgação pública do código fonte usado na sua programação. Este modelo de desenvolvimento livre e distribuído têm-se tornado especialmente importante nos últimos anos, tendo-se demonstrado que é uma forma particularmente eficaz de conjugar os esforços de grandes equipas de pessoas, de modo a produzir produtos de *software* de grande qualidade [2]. Nos últimos meses, a relevância deste modelo tem chegado às grandes empresas de informática que começam a usar os seus resultados, bem como a contribuir para o seu progresso. Destas podem-se citar algumas: Netscape, Corel, Sun Microsystems, Compaq, Intel e IBM. Mesmo os principais afectados por este modelo de desenvolvimento já se aperceberam da sua importância [4].

Outra vantagem do sistema Linux é a de ser gratuito. Assim o custo total do *software* usado nos servidores de áreas de trabalho, servidores de *boot* e firewalls é nulo. Por sua vez o equipamento usado consiste em computadores pessoais, de média gama e marca branca, o que constitui também uma grande economia.

Um exemplo da fiabilidade dos servidores é bem evidenciado na sua disposição física. Encontram-se numa sala fechada, desprovidos de monitor, e toda a sua administração é feita remotamente. O último *reboot* ocorreu quando se levaram as máquinas para o seu lugar definitivo, e desde aí não foi necessário que ninguém voltasse a entrar nessa sala. Nesse período, o servidor de *boot* serviu milhares de sessões, e o servidor de áreas armazenou a informação de várias centenas de alunos, sem que tivessem surgido quaisquer situações anómalas.

5 Conclusões

A experiência recolhida com a aplicação desta tecnologia a um ambiente de ensino permite já uma visão minimamente sólida sobre as potencialidades deste sistema de administração. Constituindo os laboratórios de ensino um ambiente razoavelmente homogéneo em termos de *hardware*, torna-se aqui particularmente fácil e rápido proceder a adaptações da instalação, sobre todo o parque de máquinas. De facto, uma vez assimilada a nova abordagem à administração torna-se rapidamente impensável o recuo para outros modelos.

A nossa experiência pessoal apontou também a existência de vantagens consideráveis na adopção de servidores Linux como parte da infra-estrutura de apoio ao sistema, o que para além de outros factores propicía claras vantagens económicas.

Para além do caso testado, é fácil antever uma muito promissora aplicabilidade a outros ambientes, como seja na gestão de cibercafés, na configuração de sistemas de apoio à produção, onde é particularmente crítico o tempo de paragem para reconfiguração, e de um modo geral na administração de parques de máquinas significativos. Contudo, importa realçar que a eficiência do sistema como um todo, não advem da simples adopção de um mecanismo de *boot* remoto mas sim da sua integração numa arquitectura de rede e serviços, correctamente delineada. Assim, cada um dos casos será sempre um caso e acima de tudo uma interessante tarefa de engenharia.

Referências

[1] Bruce Perens. Open Source Definition. WWW: http: //opensource.org/ osd.html, June 1997.

[2] Eric Raymond. The Cathedral and the Bazaar. WWW: http: //www.earthspace.net/ ẽsr/writings/cathedral-bazaar/, June 1997.

[3] Marc Vuilleumier Stückelberg, Sandro Viale, and David Clerc. Remote-boot HOWTO. http: //cuiwww.unige.ch/ info/pc/remote-boot/howto.html.

[4] Vinod Valloppillil and Josh Cohen. The "Halloween" Papers. WWW: http: //opensource.org/ halloween.html, October 1998.