

REINALDO MONTEIRO COTRIM

Implantação do sistema CACIC - Configurador Automático e Coletor de  
Informações Computacionais na UESC – Universidade Estadual de Santa  
Cruz

LAVRAS  
MINAS GERAIS - BRASIL  
2007

REINALDO MONTEIRO COTRIM

Implantação do sistema CACIC - Configurator Automático e Coletor de  
Informações Computacionais na UESC – Universidade Estadual de Santa  
Cruz

Monografia apresentada ao departamento de  
Ciência da Computação da Universidade  
Federal de Lavras, como parte das exigências  
do curso de Pós-Graduação *Lato*  
*Sensu* em Administração de Redes Linux, para  
obtenção do título de especialista em Redes  
Linux

Orientador  
Prof. Joaquim Quinteiro Uchôa

LAVRAS  
MINAS GERAIS - BRASIL  
2007

REINALDO MONTEIRO COTRIM

Implantação do sistema CACIC - Configurator Automático e Coletor de  
Informações Computacionais na UESC – Universidade Estadual de Santa  
Cruz

Monografia apresentada ao departamento de  
Ciência da Computação da Universidade  
Federal de Lavras, como parte das exigências  
do curso de Pós-Graduação *Lato*  
*Sensu* em Administração de Redes Linux, para  
obtenção do título de especialista em Redes  
Linux

LAVRAS  
MINAS GERAIS - BRASIL  
2007

## **Dedicatória**

Dedico essa monografia a Deus, a minha família, a minha namorada, aos meus amigos e a todos que direta ou indiretamente contribuíram para essa conquista.

## **Agradecimentos**

A Deus, por ser a luz guia nos momentos mais difíceis.

Aos meus pais, José e Maria, por terem propiciado toda estrutura para eu chegar até aqui.

Aos meus irmãos, José Maria, José Carlos e Reginaldo, pelo apoio e carinho fraterno.

À minha namorada, Patrícia, por seu amor e incentivo para eu poder seguir em frente.

Ao professor Joaquim Quinteiro Uchôa pela orientação e apoio, transmitindo seu conhecimento.

Aos meus amigos, todos eles, que com amizade colaboraram direta e indiretamente para a conclusão de mais essa etapa.

## Resumo

O presente trabalho apresenta a implementação do sistema CACIC - Configurador Automático e Coletor de Informações Computacionais. A implantação desta ferramenta de inventário eletrônico de *software* e *hardware* tem o objetivo de auxiliar setores da Universidade Estadual de Santa Cruz a controlar os recursos de informática. Neste documento são apresentados detalhes de configuração, utilização dos recursos da ferramenta CACIC e resultados obtidos com a implementação da mesma.

## Sumário

1 - Introdução.....	1
2 - Revisão bibliográfica .....	5
2.1 - O CACIC.....	5
2.1.1 - Módulo agente .....	6
2.1.2 - Módulo gerente .....	7
2.1.3 - Módulo Super-Gerente.....	7
2.1.4 - Requisitos Mínimos de Hardware e Software.....	8
2.2 - Servidor apache .....	10
2.3 - A linguagem PHP.....	12
2.4 - O banco de dados MySQL.....	15
2.4 - A UESC.....	17
3 - Material e método .....	19
3.1 - Implantação.....	19
3.2 - Obtendo o CACIC.....	19
3.3 - Instalação do Módulo Gerente.....	20
3.3.1 - Banco de dados.....	20
3.3.2 - Servidor WEB com suporte a PHP.....	22
3.3.3 - Servidor de FTP.....	22
3.3.4 - A interface WEB.....	24
3.4 - Configuração do Módulo Gerente.....	25
3.5 - Instalação dos agentes .....	27
4 - Resultados e discussão.....	30
5 - Conclusão .....	36
6 - Bibliografia.....	37

## Índice de Figuras

Configuração do Agente.....	26
Login do administrador do sistema CACIC.....	26
Menu de configuração.....	27
Instalação do agente.....	29
Coleta de Hardware.....	34
Unidades de disco.....	35

# 1 - Introdução

A UESC – Universidade Estadual de Santa Cruz atualmente conta com um parque computacional de aproximadamente mil e duzentas máquinas, distribuídas em um *campus* muito extenso. Esse fato dificulta de forma expressiva o controle de *software* e *hardware* de toda universidade, fazendo-se necessário a implantação de um sistema que possa auxiliar nesse trabalho.

A existência de diversas plataformas de *hardware* e *software*, somados com a complexidade de gerenciamento dos mais diversos sistemas, aplicações e serviços de uma rede heterogênea, tornam a tarefa de gerenciamento corporativo desafiadora (CARVILHE, 2000).

O CACIC - Configurador Automático e Coletor de Informações Computacionais, considerado a primeira experiência consolidada de *software* livre desenvolvido e disponibilizado com segurança pelo setor público é um sistema eletrônico de inventário, capaz de fornecer um diagnóstico preciso da rede e disponibilizar informações como o número de equipamentos e sua distribuição nos mais diversos órgãos, os tipos de *software* utilizados e configurações de *hardware*.

O CACIC possibilita a manutenção periódica dos computadores ao permitir que problemas corriqueiros como a sobrecarga de espaço em disco, *software* de execução duvidosa e o pleno funcionamento do anti-vírus sejam diagnosticados com antecipação. Também pode fornecer informações patrimoniais e a localização física

dos equipamentos, ampliando o controle do parque computacional e a segurança na rede.

Um detalhe importante é que ele está baseado na (*General Public License*) GPL. Isto significa que é possível usá-lo de forma livre, sem gastos com licenças e tendo acesso irrestrito ao código fonte. A GNU *General Public License* (Licença Pública Geral), foi criada por Richard Stallman movido pela idéia de combater as licenças de utilização, que paulatinamente restringiam a liberdade dos usuários. Essa inspiração que em 1984 incentivou Stallman a criar a *Free Software Foundation*, com objetivo de estabelecer uma nova filosofia de produção de *software*: a filosofia do *Software Livre* (FERREIRA, 2006).

*Software Livre* é o *software* disponibilizado, de forma gratuita ou não, com as premissas de liberdade de instalação; plena utilização; possibilidade de modificações/aperfeiçoamentos para necessidades específicas; distribuição da forma original ou modificada, com ou sem custos. Contrapõe-se ao modelo *software* proprietário onde o usuário de *software* não tem permissão para redistribuí-lo nem alterar seu funcionamento para ajustar-se às suas necessidades (RIBEIRO, 2006).

O presente trabalho apresenta a implantação do CACIC na rede da UESC. A escolha do CACIC foi estimulada pela necessidade do setor de manutenção desta universidade em obter relatórios do parque computacional, para um melhor e mais ágil atendimento aos usuários.

A maioria dos chamados de atendimento feitos no *HelpDesk* da UESC são destinados à manutenção de computadores ou rede. Boa

parte deste trabalho poderia ser feito de forma preventiva, reduzindo os atendimentos em caráter de urgência, e eliminando o problema antes que ele aconteça. Para isso seria necessário que a equipe do setor tivesse um inventário atualizado e detalhado da situação dos computadores instalados na rede.

Foi por essa motivação que a idéia de usar um *software* de inventário surgiu e o CACIC foi o escolhido pela sua capacidade de fornecer um diagnóstico preciso na coleta de informações sobre *hardware* e *software*, e também pelo fato de ser livre, poupando a universidade de gastos com licenças.

O capítulo 2 deste trabalho faz uma revisão bibliográfica de todas as tecnologias usadas nesse projeto, apresentando o CACIC de forma geral, incluindo os módulos agente, gerente e super gerente. Além disso, nesse capítulo são mostrados os requisitos mínimos de *software* e *hardware* para instalação do sistema. O capítulo dois mostra ainda o servidor web apache e suas funcionalidades juntamente com a linguagem PHP e o banco de dados MySQL.

O capítulo 3 faz uma apresentação sobre os materiais e métodos usados no projeto, falando sobre a implantação do sistema e explicado como se pode obter o CACIC. Neste capítulo também é descrito passo a passo o processo de instalação do servidor agente que se subdivide em instalação do banco de dados, instalação do servidor web com suporte a PHP, instalação do servidor FTP e interface web. Ainda neste capítulo é mostrado como se deve configurar o módulo gerente e a

instalação do módulo agente do CACIC.

No capítulo 4 é feita uma discussão sobre os resultados obtidos com a implantação do sistema. No capítulo 5 é apresentada uma conclusão sobre o trabalho realizado

## 2 - Revisão bibliográfica

### 2.1 - O CACIC

O CACIC - Configurador Automático e Coletor de Informações Computacionais, primeiro *Software* Público do Governo Federal, resultado do Consórcio de Cooperação entre a SLTI - Secretaria de Logística Tecnologia da Informação, do Ministério do Planejamento, Orçamento e Gestão - MPOG e a DATAPREV - Empresa de Tecnologia e Informações da Previdência Social, desenvolvido pelo Escritório Regional da DATAPREV no Espírito Santo (DATAPREV, 2005). É um sistema de inventário eletrônico capaz de buscar dados sobre *hardware* e *software*. O CACIC tem seu funcionamento baseado em três módulos: Agente, Gerente e Super-gerente.

Segundo (DATAPREV, 2005) as funções do CACIC são:

- Coletar informações sobre os componentes de *hardware* instalados em cada computador e disponibilizá-las aos administradores de sistemas.
- Alertar os administradores de sistemas quando forem identificadas alterações na configuração dos componentes de *hardware* de cada computador.
- Coletar diversas informações sobre os *software* instalados em cada computador e disponibilizá-las aos administradores de sistemas.
- Identificar diretórios compartilhados considerados inseguros e aplicar as

restrições de segurança necessárias.

- Coletar informações de Patrimônio (tombo, localização, etc.) de cada computador e disponibilizá-las aos administradores de sistemas.
- Alertar os administradores de sistemas quando forem identificadas alterações na localização física do computador.

### **2.1.1 - Módulo agente**

O módulo agente é um programa, compilado, que fica permanentemente ativo na estação de trabalho do usuário Windows, podem ser executados nas versões 95 OSR2, 98, 98 SE, ME, NT, 2000 e XP. Esse módulo é responsável por coletar, com uma frequência definida pelo gerente, os dados de *hardware*, *software* e rede do microcomputador. Após cada coleta, os dados são enviados para o gerente (DATAPREV, 2005). O agente CACIC para GNU/Linux encontra-se ainda em desenvolvimento. Escrito em Perl, o projeto foi iniciado por Yves Junqueira a partir das discussões da comunidade. Deverá contar com interface gráfica construída em Python. A versão disponível para testes ainda não é recomendada para utilização “em produção” (DATAPREV, 2006).

### **2.1.2 - Módulo gerente**

O módulo gerente é composto de um conjunto de *software* que devem ser instalados em um servidor conhecido como LAMP: Linux, Apache, MySQL e PHP e que trabalham integrados com o objetivo de administrar os módulos agentes que estão instalados nas estações de trabalho dos usuários. O módulo-gerente recebe os dados coletados dos agentes, organiza-os e disponibiliza em forma de relatórios e consultas, através da interface web existente. Além disso, a partir do módulo-gerente é possível configurar algumas características de comportamento dos módulos agentes (DATAPREV, 2005).

### **2.1.3 - Módulo Super-Gerente**

O módulo super-gerente é composto de um conjunto de *software* que devem ser instalados em um servidor e que trabalha integrando as informações consolidadas dos diversos módulos gerentes instalados na rede. O super-gerente possui uma visão global de todo o parque computacional instalado e distribuído pela organização (DATAPREV, 2005). O uso deste módulo não será abordado neste trabalho, visto que sua utilização só é útil para grandes corporações que utilizem mais de um módulo gerente em sua rede.

## 2.1.4 - Requisitos Mínimos de *Hardware* e *Software*

Segundo (DATAPREV, 2006) os requisitos mínimos de *hardware* para instalação são.

### Modulo Gerente

- Computador Pentium/AMD 500MHz.
- 128MB de memória RAM (desejável: 256MB).
- 15MB disponível em disco rígido (para banco de dados, páginas web de administração do sistema e disponibilização de agentes para atualização das estações).
- Interface de rede.

### Modulo Agente

- 32 MB de memória RAM.
- 2,5MB de espaço disponível em disco rígido.
- Interface de rede.

Segundo (DATAPREV, 2006) os requisitos mínimos de *software* para instalação são.

### Modulo Gerente

- Debian GNU/Linux versão Sarge (Recomendado).
- Ambientes de desenvolvimento C e Perl.
- Servidor web Apache-1.3.22 ou superior.

- Interpretador PHP4.
- Servidor de banco de dados MySQL-3.23.46 ou superior.
- Servidor de FTP.
- Postfix ou Sendmail.
- Pacotes para instalação do CACIC.

#### Modulo Agente para Windows

- MS-Windows 95 OSR2 ou superior.

#### Modulo Agente para Linux

- Sistema operacional GNU/Linux.
- Ambiente de desenvolvimento Perl.
- O utilitário lspci<sup>1</sup>.
- O dmidecode<sup>2</sup>.

---

<sup>1</sup> **dmidecode** fornece informações sobre o *hardware* como descrito na BIOS. É usado como uma ferramenta por outros programas da detecção de *hardware*

<sup>2</sup> Comando linux utilizado para listar os dispositivos PCI.

## **2.2 - Servidor apache**

O servidor Apache (*Apache server*) é o mais bem sucedido servidor web livre. Foi criado em 1995 por Rob McCool, então funcionário do NCSA (*National Center for Supercomputing Applications*), Universidade de Illinois. Numa pesquisa realizada em dezembro de 2005, foi constatado que a utilização do Apache supera 60% nos servidores ativos no mundo (WIKIPEDIA, 2007). Quando Rob deixou o NCSA, o desenvolvimento foi interrompido, assim muitos desenvolvedores buscaram personalizar sua própria versão do NCSA ou adicionar mais características para atender as suas necessidades. Neste momento começa a história do Apache com Brian Behlendorf e Cliff Skolnick abrindo uma lista de discussão para interessados no desenvolvimento, conseguindo espaço em um servidor doado pela HotWired e trocando *patches* corrigindo problemas, adicionando recursos e discutindo idéias com outros desenvolvedores e *hackers* interessados neste projeto (FOCALINUX, 2007) .

O Apache *Server* é um *software* livre, o que significa que qualquer um pode estudar ou alterar seu código-fonte, além de poder utilizá-lo gratuitamente. É graças a essa característica que o *software* foi, e continua sendo, melhorado ao passar dos anos. Graças ao trabalho muitas vezes voluntário de vários desenvolvedores, o Apache continua sendo o servidor web mais usado no mundo.

Além de estar disponível para o Linux e para outros sistemas

operacionais baseados no Unix, o Apache também conta com versões para o Windows, para o Novell Netware e para o OS/2, o que o torna uma ótima opção para rodar em computadores obsoletos (desde que este atenda aos requisitos mínimos de *hardware*).

O servidor Apache é capaz de executar código em PHP, Perl, *Shell Script* e até em ASP e pode atuar como servidor FTP, HTTP, entre outros. Sua utilização mais conhecida é a que combina o Apache com a linguagem PHP e o banco de dados MySQL.

A exigência de *hardware* do Apache depende de sua aplicação, mas um PC Pentium com 64 MB de memória RAM é capaz de executá-lo tranquilamente em um ambiente corporativo pequeno. No entanto, quando se trata de um site na internet, é interessante ter máquinas tão poderosas quanto o que exige o nível de acesso.

Algumas características do Servidor Apache segundo (FOCALINUX, 2007) :

- Possui suporte a scripts cgi usando linguagens como Perl, PHP, Shell Script, ASP, etc.
- Suporte a autorização de acesso podendo ser especificadas restrições de acesso separadamente para cada endereço/arquivo/diretório acessado no servidor.
- Autenticação requerendo um nome de usuário e senha válidos para acesso a alguma página/sub-diretório/arquivo (suportando criptografia via Crypto e MD5).

- Negociação de conteúdo, permitindo a exibição da página Web no idioma requisitado pelo Cliente Navegador.
- Suporte a tipos mime.
- Personalização de *logs*.
- Mensagens de erro.
- Suporte a virtual *hosting* (é possível servir 2 ou mais páginas com endereços/ portas diferentes através do mesmo processo ou usar mais de um processo para controlar mais de um endereço).
- Suporte a IP virtual *hosting*.
- Suporte a *name* virtual *hosting*.
- Suporte a *proxy* e redirecionamentos baseados em URLs para endereços Internos.
- Suporte a criptografia via SSL, Certificados digitais.
- Módulos DSO (*Dynamic Shared Objects*) permitem adicionar/remover funcionalidades e recursos sem necessidade de recompilação do programa.

## **2.3 - A linguagem PHP**

PHP (um acrônimo recursivo para "PHP: *Hypertext Preprocessor*") é uma linguagem de programação de computadores interpretada, livre e muito utilizada para gerar conteúdo dinâmico na web. Apesar de ser uma linguagem de fácil aprendizado e de uso para

pequenos *scripts* dinâmicos simples, o PHP é uma linguagem poderosa orientada a objetos.

A linguagem surgiu por volta de 1994, como um subconjunto de *scripts* Perl criados por Rasmus Lerdof, com o nome PHP/FI (*Personal Home Page Tools/Forms Interpreter*). Com as adições de Zeev Suraski e Andi Gutmans, dois programadores israelitas pertencentes ao *Technion*, o Instituto Israelita de Tecnologia, que reescreveram o *parser*, era lançada em 1997 a PHP 3, primeira versão estável e parecida com a linguagem atual. Ao reescrever o *parser*, foi criado a *Zend Engine*, que é mantido oficialmente pela empresa *Zend* em conjunto com a comunidade PHP. Em Maio de 2000 veio a público a versão 4, e em Julho de 2004, a versão 5, onde a principal mudança foi uma nova API para orientação a objetos provida pela *Zend Engine 2*.

Trata-se de uma linguagem extremamente modularizada, o que a torna ideal para instalação e uso em servidores web. Diversos módulos são criados no repositório de extensões PECL (*PHP Extension Community Library*) e alguns destes módulos são introduzidos como padrão em novas versões da linguagem. É muito parecida, em tipos de dados, sintaxe e mesmo funções, com a linguagem C e com a C++. Pode ser, dependendo da configuração do servidor, embutida no código HTML.

Além disso, destaca-se a extrema facilidade com que PHP lida com servidores de base de dados, como MySQL, Firebird, PostgreSQL, Microsoft SQL Server e Oracle.

Existem versões do PHP disponíveis para os seguintes

sistemas operacionais: Windows, Linux, FreeBSD, Mac OS, OS/2, AS/400, Novell Netware, RISC OS, IRIX e Solaris (WIKIPEDIA, 2006).

Principais características do PHP segundo (INFOWESTER, 2007a).

- Código Aberto: Todo o código fonte do PHP está disponível.
- Custo Zero: O PHP é gratuito. Basta ir ao site oficial e fazer o *download*.
- Multiplataforma: O PHP pode rodar sobre o Unix, Linux, Windows, etc.
- Eficiência: O PHP consome poucos recursos do servidor, permitindo que programas complexos sejam desenvolvidos, sem que isto implique em grande demora na sua execução. Além disso, o PHP como módulo nativo do servidor WEB, evita chamadas externas, o que o torna ainda mais eficiente.
- Acesso a Bancos de Dados: Pode-se acessar diretamente os principais bancos de dados utilizados atualmente e qualquer banco de dados do mercado por meio de ODBC.
- Processamento de Imagens: O PHP pode criar imagens dinamicamente e enviá-las ao *browser* do usuário.
- Além destas características, existe ainda a capacidade de ler informação do padrão XML, processamento de arquivos (leitura e gravação, tanto no formato texto quanto binário), a manipulação

de variáveis complexas, a utilização de funções e classes e geração de código *JavaScript*, ou outro qualquer para processamento no lado cliente, a manipulação de *e-mails*, o gerenciamento de documentos PDF e muitas outras características que tornam o PHP uma linguagem realmente potente e indicada para a construção de sites dinâmicos.

## **2.4 - O banco de dados MySQL**

O MySQL é um dos sistemas de gerenciamento de banco de dados mais populares que existe e, por ser otimizado para aplicações web, é amplamente utilizado na internet. É muito comum encontrar serviços de hospedagem de sites que oferecem o MySQL e a linguagem PHP, justamente porque ambos trabalham muito bem em conjunto.

Outro fator que ajuda na popularidade do MySQL é sua disponibilidade para praticamente qualquer sistema operacional, como Linux, FreeBSD (e outros sistemas baseados em Unix), Windows e Mac OS X. Além disso, o MySQL é um *software* livre (sob licença GPL), o que significa que qualquer um pode estudá-lo ou alterá-lo conforme a necessidade (INFOWESTER, 2007b).

Entre as características técnicas do SGBD MySQL, estão segundo (INFOWESTER, 2007b):

- Alta compatibilidade com linguagens como PHP, Java, Python, C#,

Ruby e C/C++.

- Baixa exigência de processamento (em comparação com outros SGBD).
- Vários sistemas de armazenamento de dados (*database engine*), como MyISAM, MySQL Cluster, CSV, Merge, InnoDB, entre outros.
- Recursos como *transactions* (transações), conectividade segura, indexação de campos de texto, replicação, etc.
- Instruções em SQL, como indica o nome.

O MySQL surgiu na Suécia pelas mãos de três colegas: Allan Larsson, David Axmark e Michael Monty Widenius. Trabalhando com base de dados, eles sentiram a necessidade de fazer determinadas conexões entre tabelas e usaram o mSQL para isso. Porém, não demorou para perceberem que essa ferramenta não lhes atendia conforme o necessário e passaram a trabalhar em uma solução própria. Surgiu então o MySQL, cuja primeira versão foi lançada no ano de 1996.

Um fato importante a ser destacado sobre o MySQL é que esse SGBD também possui uma licença comercial, isto é, paga. A MySQL AB, empresa que o desenvolve e que o distribui, oferece suporte diferenciado a quem estiver disposto a pagar por isso (INFOWESTER, 2007b).

## **2.4 – A UESC**

A Universidade Estadual de Santa Cruz teve sua origem nas escolas isoladas criadas no eixo Ilhéus/Itabuna, na década de 60. Em 1972, resultante da iniciativa das lideranças regionais e da Comissão Executiva do Plano da Lavoura Cacaueira (CEPLAC), as escolas isoladas (Faculdade de Direito de Ilhéus, Faculdade de Filosofia de Itabuna, e Faculdade de Ciências Econômicas de Itabuna) congregaram-se, formando a Federação das Escolas Superiores de Ilhéus e Itabuna - FESPI. Reunidas em Campus, na Rodovia Ilhéus/Itabuna, no município de Ilhéus, os estabelecimentos de ensino foram ganhando maturidade e competência, criando as condições para pleitear o "status" de Universidade. Mantida, entretanto, por uma fundação de natureza privada, o acesso as seus cursos tornava-se particularmente difícil, considerada à realidade regional. Assim, a Federação reorientou-se no sentido de tornar-se uma fundação pública e em 1991, o governo do estado da Bahia incorporou a FESPI, escola particular, ao quadro das escolas públicas de 3º grau da Bahia. (UESC, 2007)

A Universidade Estadual de Santa Cruz oferece 26 cursos de graduação entre bacharelados e licenciaturas, podem citar: Medicina, Engenharia de Produção e Sistemas, Direito, Enfermagem, Ciência da Computação, Educação Física, entre outros. Em nível de pós-graduação a Uesc oferece cursos de especializações (Gestão Participativa; Direito Processual Civil; Economia de Empresas; Economia das Sociedades

Cooperativas, entre outros), cursos de mestrado (Desenvolvimento Regional e Meio Ambiente; Sistemas Aquáticos Tropicais; Cultura & Turismo, etc.) e um curso de doutorado (Genética e Biologia Molecular).

Do ponto de vista organizacional, a Uesc é composta de departamentos que possuem atribuições administrativas e acadêmicas e os colegiados dos cursos que são responsáveis pelas tarefas didático-pedagógicas. A universidade também é composta por órgãos suplementares (Biblioteca, Hospital Veterinário, Prefeitura do Campus etc.), pró-reitorias (Administrativa, Pesquisa, Extensão e Graduação), gerências (Recursos Humanos, Financeiro, Informática, etc.) e setores (de Pessoal, Importação, Materiais, Patrimônio, Compras, etc.). A alta administração da UESC é exercida pelos Conselhos, tendo a Reitoria como órgão executivo das decisões desses Conselhos.

## **3 - Material e método**

### **3.1 – Implantação**

A implantação do CACIC no parque computacional da UESC tem como meta melhorar o controle dos seus recursos de informática. Sua implantação foi estimulada pelos requisitos que o *software* atende tecnicamente, além de ser um *software* livre que não acarretará em custos com licenças para a universidade.

### **3.2 - Obtendo o CACIC**

O primeiro passo para a implantação do CACIC é fazer o cadastro no site oficial para obter uma cópia do *software*. A partir daí, com o cadastro feito, é possível ter acesso a toda documentação e pacotes de instalação. A distribuição do CACIC é feita de forma gratuita pelo governo federal e seu suporte é dado pela comunidade formada pelos criadores e usuários, contando com fóruns de discussão e um *wiki* para desenvolvimento da documentação e tutoriais.

### **3.3 - Instalação do Módulo Gerente**

Foi utilizado para instalação do Módulo Gerente um Intel Celeron CPU 1.80GHz com 256MB de memória RAM e HD de 40 GB. Esta máquina está disponível para a coleta das informações enviadas pelos agentes instalados nos computadores da rede. O Sistema Operacional adotado foi o Debian GNU/Linux versão Sarge que é o recomendado pela documentação oficial para a configuração do Módulo Gerente.

#### **3.3.1 - Banco de dados**

A instalação do banco de dados foi feita seguindo o manual, onde é recomendado instalar o servidor e o cliente MySQL através do comando apt-get. O apt-get é uma interface simples de linha de comando para fazer o *download* de pacotes e instalá-los. O Debian e os sistemas baseados nele utilizam esse comando.

```
shell# apt-get install mysql-client mysql-server
```

Depois do banco de dados já instalado é necessário que se atribua uma senha de *root* ao servidor. Essa senha é atribuída com o seguinte comando:

```
shell# /usr/bin/mysqladmin -u root password '<SENHA>'
```

O MySQL é usado para rodar a base de dados do servidor gerente que é responsável pelo gerenciamento dos agentes. A criação dessa base de dados se dá por meio de um *script* que é encontrado no diretório criado quando se descompacta o arquivo *cacic-gerente-Jun2005.tar.gz*. Dentro da pasta criada se encontra o *script* chamado *criar\_banco\_cacic.sql*, esse *script* cria a estrutura de tabelas depois que é criada a base de dados com o nome *cacic*.

Para criar a base de dados com o nome *cacic* é utilizado o comando.

```
shell# echo "CREATE DATABASE cacic;" | mysql -u root \ -p<SENHA>
```

A estrutura de tabelas é criada com a chamada do *script* *criar\_banco\_cacic.sql* que é executado com o comando.

```
shell# mysql -u root -p<SENHA> -D cacic < criar_banco_cacic.sql
```

Com a base de dados criada e toda a estrutura de tabelas é necessário a criação de um usuário administrador do gerente CACIC.

```
shell# echo "INSERT INTO usuarios (nm_usuario_acesso, te_senha, \
id_grupo_usuarios) VALUES ('admin', PASSWORD('<SENHA_ADM>'),'2');" \
| mysql -u root -p<SENHA_ROOT_MYSQL> -D cacic
```

### **3.3.2 - Servidor WEB com suporte a PHP**

Um dos serviços mais importantes para o funcionamento do Módulo gerente é o servidor web. Para seu funcionamento é necessário instalar um servidor Apache juntamente com suporte PHP e módulos para o perfeito funcionamento do sistema. Para a instalação do servidor Apache e seus módulos foi utilizado o comando apt-get.

```
shell# apt-get install apache php4-dev php4 php4-imap php4-mysql php4-gd
```

Com esse comando foram instalados os seguintes pacotes: Apache, mod\_php4, PHP4, php4-dev e os módulos PHP para IMAP, MySQL e GD.

### **3.3.3 - Servidor de FTP**

É necessário a configuração de um servidor FTP na máquina gerente, a sua principal função é a atualização dos agentes instalados nos

computadores que serão monitorados pelo CACIC. O servidor escolhido para esse fim foi o ProFTPD instalado com o comando apt-get.

```
shell# apt-get install proftpd
```

As entradas a seguir no final do arquivo `/etc/proftpd.conf` são necessárias para medidas de segurança. Elas transformam o diretório *home* do usuário no diretório raiz do servidor FTP.

```
DefaultRoot ~
```

A configuração no comando seguinte permite que o usuário que será usado pelo CACIC para baixar os pacotes via FTP não tenha um *shell* válido no sistema operacional.

```
RequireValidShell off
```

É necessário a adição de um usuário que é usado pelo CACIC para os *downloads* de *updates* dos agentes. No exemplo a seguir, é mostrado o comando utilizado para realizar essa tarefa.

```
shell# adduser --shell /bin/false --home /var/www/ftpcacic ftpcacic
```

Por fim, é criado um diretório chamado agentes dentro de ftpcacic. Ele é usado para armazenar os arquivos do agente:

```
shell# mkdir -p /var/www/ftpcacic/agentes
```

### **3.3.4 - A interface WEB**

A interface web é construída em PHP e está compactada no arquivo cacic-gerente-Jun2005.tar.gz esse arquivo quando descompactado gera uma pasta chamada cacic2.

```
shell# tar -zxvf cacic-gerente-Jun2005.tar.gz
```

Depois da descompactação a pasta cacic2 deve ser copiada para o diretório raiz do servidor web, e então deve ser alterada a sua propriedade de permissões para o usuário apache do sistema operacional do servidor, no caso do Debian esse usuário é o www-data.

```
shell# cp -R cacic-gerente/cacic2 /var/www/  
shell# chown -R www-data /var/www/cacic2
```

### **3.4 - Configuração do Módulo Gerente**

Neste passo é estabelecida uma configuração para o funcionamento do módulo gerente. São inseridos dados sobre a instituição, dados comportamentais sobre o agente e informações de rede dos computadores monitorados. A primeira parte para a configuração do módulo gerente é a configuração da interface web do CACIC, alterando as seguintes linhas do arquivo `/var/www/cacic2/include/config.php`.

```
$usuario_bd = "root"; //Usuário usado para acesso ao bd cacic  
$senha_usuario_bd = "<SENHA>"; //Senha usada para acesso ao banco de  
dados do CACIC
```

Em seguida é só acessar em uma máquina com interface gráfica, o diretório `/cacic2` via navegador.

```
http://<IP_DO_GERENTE>/cacic2/
```

Se toda a instalação ocorrer bem, aparecerá uma tela como a da Figura 1. Observando no topo da janela a versão do CACIC e o nome da instituição. Como o gerente ainda não foi configurado, aparecerá "Dataprev - ESES.P". De qualquer maneira, essa informação indica que o banco de dados do CACIC foi acessado e consultado corretamente.

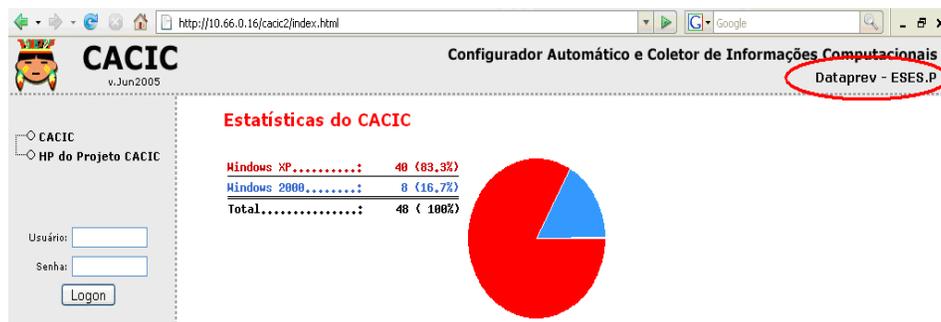


Figura 1: Configuração do Agente

Para fazer o *login* é só inserir o usuário admin juntamente com a sua senha escolhida na inserção do usuário no banco MySQL. Isso é mostrado na Figura 2.



Figura 2: Login do administrador do sistema CACIC

Com o *login* feito é permitido o acesso ao menu de configuração, como mostra a Figura 3, acessando esse menu é possível as mais diversas ações administrativas. Entre elas configurar o módulo gerente, configurar as ações dos agentes, cadastrar redes além da exibição de relatórios sobre sua rede monitorada.



Figura 3: Menu de configuração

### 3.5 - Instalação dos agentes

O módulo agente é o componente de *software*, compilado que fica permanentemente ativo na estação de trabalho do usuário. A

instalação dos agentes é necessária para que o servidor gerente tenha acesso aos dados de configuração das máquinas na rede. Esse módulo em execução é responsável por coletar dados relativos a *hardware*, *software*, redes e patrimônio, manter o módulo gerente atualizado em relação às informações dos computadores monitorados e enviar alertas e notificações para o módulo gerente quando ocorrerem mudanças de *hardware* ou patrimônio.

Para a instalação do agente CACIC no Windows basta executar o programa `chkcacic.exe`, devendo estar este devidamente configurado para o servidor onde o módulo gerente está instalado. O programa de instalação apresenta uma interface gráfica através da qual o usuário deve indicar o IP do servidor do gerente e a pasta em que será instalado o agente CACIC no cliente. A interface do programa de instalação do módulo agente pode ser vista na Figura 4.

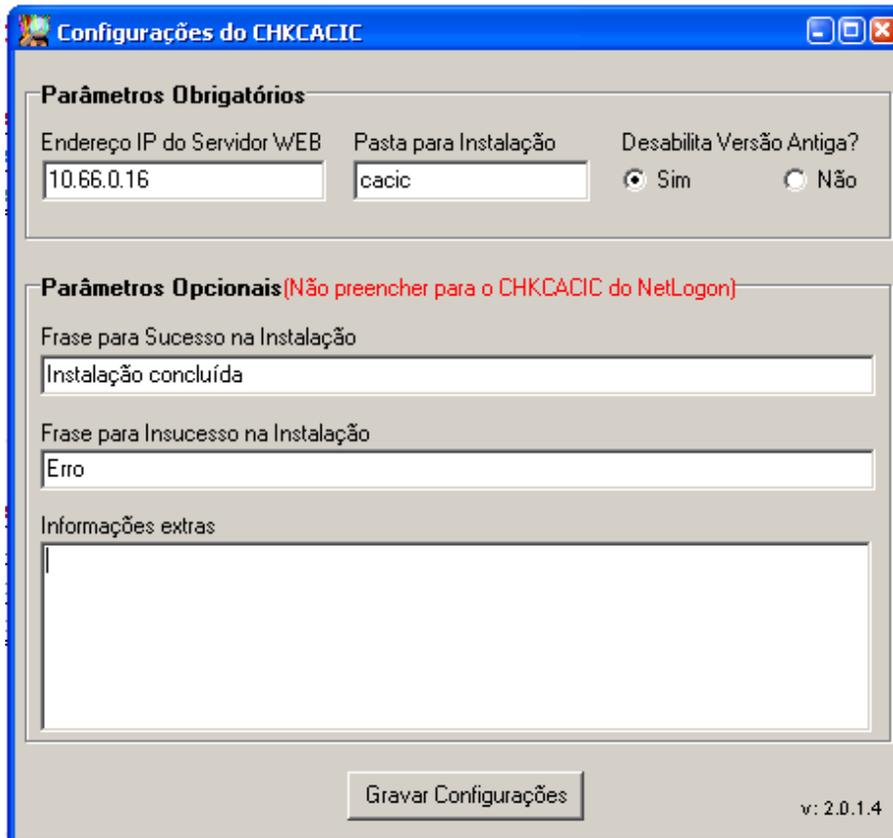


Figura 4: Instalação do agente

## 4 - Resultados e discussão

A escolha do CACIC foi motivada principalmente pela sua condição de ser um *software* livre e esse fato foi determinante. A compra de um *software* proprietário demandaria um custo maior e o desenvolvimento de outro *software*, que atendesse às necessidades do setor de manutenção, seria inviável, visto que, além do custo alto, levaria mais tempo para implantação.

Não só o fato de ser livre, mas também as funcionalidades do CACIC atendiam às necessidades do setor de manutenção, que era a realização atualizada de um inventário de *hardware* e *software* nos computadores da universidade, e de apresentar individualmente a localização física de cada máquina através de seu controle de patrimônio.

Para o funcionamento do módulo gerente foi preciso a configuração de um servidor, com tecnologias livres utilizando LAMP (LAMP, um acrônimo para Linux, Apache, MySQL e PHP). A farta documentação encontrada facilitou também o trabalho com a utilização dessas ferramentas contribuindo para um maior aprofundamento no estudo e configuração das mesmas.

A implantação dos agentes foi uma das etapas com mais dificuldades encontradas, talvez pelo fato da documentação do CACIC não abordar muitos detalhes sobre esse tema. Mesmo assim as dúvidas foram sanadas através do fórum de discussão da comunidade CACIC. Foi possível perceber que o agente só faz corretamente a coleta se a rede

onde ele agir estiver cadastrada no banco de dados do módulo gerente. O funcionamento do agente no Windows 98 também não foi satisfatório visto que o usuário poderia interromper o processo de coleta do agente a qualquer momento, prejudicando a atualização dos dados. A solução encontrada foi atualizar gradativamente, junto com a instalação dos agentes, os sistemas operacionais de forma a proteger o processo de coleta das ações de um usuário comum no sistema.

Aproximadamente cento e cinquenta computadores já estão com o agente de coletas do CACIC instalado. A meta é atingir todo o *campus* da universidade gradativamente. Entre os setores contemplados nessa primeira etapa estão alguns setores da torre administrativa e dois laboratórios de informática, respectivamente um do curso de história e outro do curso de filosofia. Com esse total de computadores funcionando o agente do CACIC foi possível obter resultados positivos da ferramenta.

As coletas estão trazendo relatórios atualizados diariamente. Através destas coletas é possível avaliar as funcionalidades do CACIC de forma prática. Os relatórios emitidos pelo módulo de coleta de informação de patrimônio auxiliam na localização física de determinada máquina na rede. Essas informações são cadastradas no momento da instalação do agente na máquina monitorada. Um exemplo prático do uso desse serviço é a localização de máquinas que estejam congestionando por algum motivo o tráfego na rede. Com auxílio de *softwares* para controle de banda é possível analisar o tráfego utilizado na rede e identificar essas máquinas pelo IP, mas não pela localização física, e com

a ajuda desse módulo do CACIC esse tipo de identificação está sendo realizada.

Através do módulo de coleta de patrimônio é gerado um relatório do número de máquinas por setor. Essa funcionalidade poderá ser aplicada para o inventário anual realizado pelo setor de patrimônio da UESC onde cada máquina no CACIC estará identificada pelo seu número de tombo e localização.

Esse tipo de informação sobre localização gera uma informação errada se a máquina for mudada de setor. Uma solução encontrada para esse problema foi trabalhar junto com o setor de patrimônio que é responsável pela movimentação de bens na universidade. Essa parceria prevê que cada vez que uma máquina for remanejada de setor essa informação deve ser transmitida ao setor de manutenção que é responsável pelo gerenciamento do CACIC. Feito isso os dados de localização patrimonial então são alterados no agente instalado na máquina e automaticamente são atualizados na base de dados do servidor gerente.

Até o presente momento há uma grande satisfação com o desempenho do CACIC. Apesar do seu uso não estar sendo feito em todo o *campus* ele já tem atendido positivamente com os resultados descritos acima. A meta esperada é atingir até o final do ano de 2007 a instalação dos agentes coletores do CACIC em todo *campus*. Este trabalho só não está sendo feito de forma mais rápida porque é realizado em conjunto com o setor de patrimônio que paralelamente está fazendo um inventário

gradativo à medida que os agentes são instalados. Esse inventário está registrando o tombo da máquina sua configuração e setor de lotação. No momento em que é instalando os agentes esses dados sobre a localização também estão sendo registrados no módulo coleta de patrimônio do agente.

O módulo de coleta de informação de *hardware* tem ajudado de forma positiva na busca de informações sobre o *hardware* instalado nos computadores, tais como memória, placa de vídeo, placa de rede, discos rígidos, etc. Essas informações são bastante úteis para saber a característica física de cada computador. É possível também através desse módulo ser informado por *e-mail* de qualquer alteração feita no *hardware* dos computadores. Esse tipo de informação é bastante útil visto que qualquer alteração de configuração de *hardware* só é permitida ser feita pelo setor de manutenção. A figura 5 mostra o exemplo de uma coleta de *hardware* feita em uma máquina da rede.

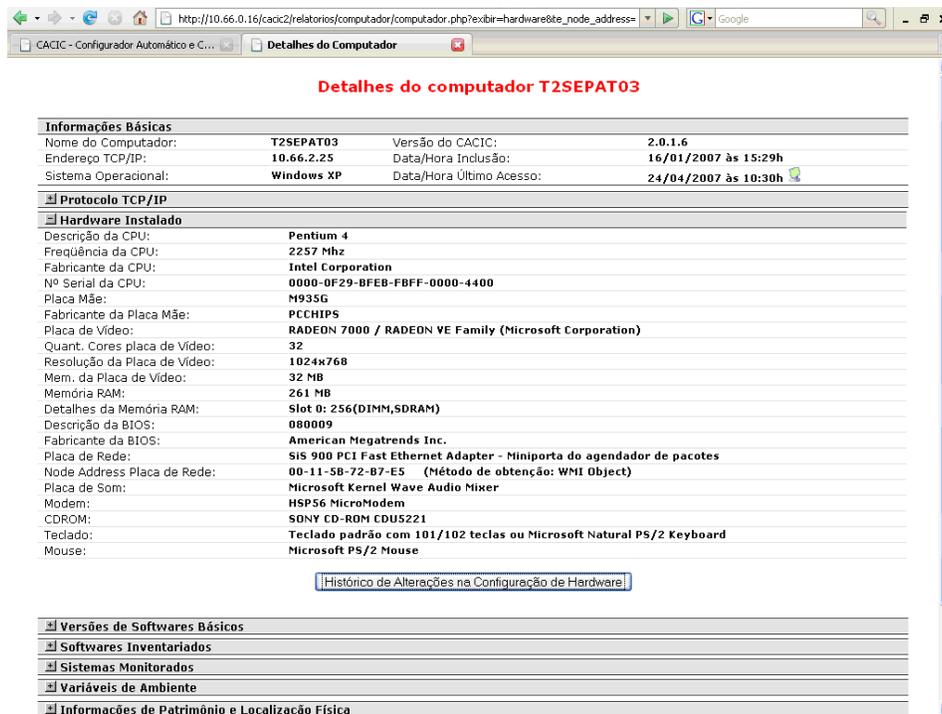


Figura 5: Coleta de Hardware

Um histórico da configuração do *hardware* também é armazenado no banco de dados, esse tipo de informação pode ser útil para saber quais peças já foram trocadas em cada máquina. O status de uso do disco rígido de cada máquina também pode ser acompanhado, essa informação tem ajudado bastante na manutenção de máquinas de laboratórios de informática, que geralmente precisam de manutenção periódica para limpeza de disco. A figura 6 mostra um relatório onde é apresentado o status do disco com trinta e dois por cento de uso indicando também o sistema de arquivos usado.

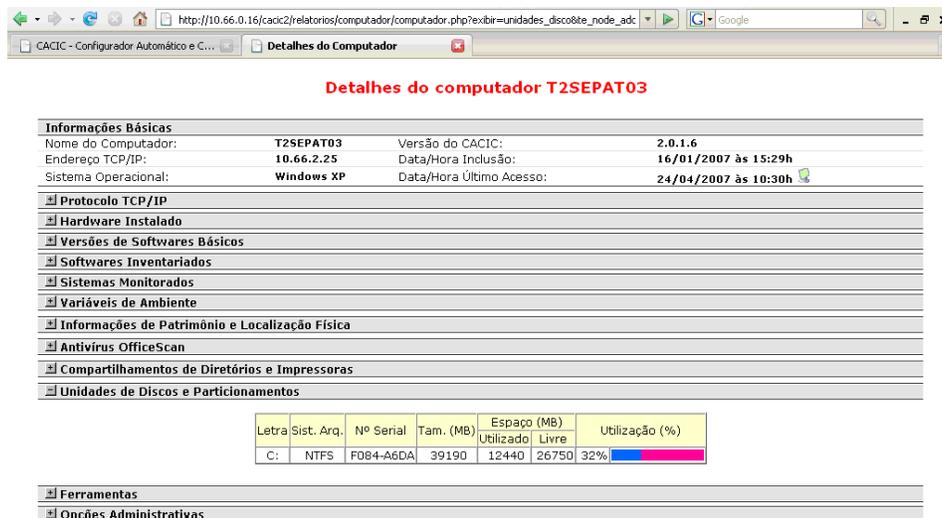


Figura 6: Unidades de disco

O módulo de coleta de informações de *software* permite que sejam coletadas informações sobre a versão de diversos programas e também acompanhar se estes são licenciados ou não. Evitando problemas com pirataria de *software*, e preservando a universidade de sofrer algum tipo de processo judicial. Este módulo também tem ajudado a detectar programas que não são permitidos na universidade como jogos por exemplo.

## 5 - Conclusão

Com a implantação do sistema de inventário CACIC na rede da UESC, mesmo que ainda em caráter parcial, já se pode visualizar sua contribuição para o gerenciamento da mesma. As máquinas onde já se encontram instalados os agentes estão constantemente atualizando suas informações depositadas na base de dados do módulo gerente, já existem estudos e propostas sobre tecnologias que possam trabalhar juntamente com o CACIC de forma a melhorar o uso dos recursos de informática.

Pode-se concluir também que esse trabalho irá contribuir fundamentalmente não só para o setor de informática na universidade mas também para áreas de controle administrativo como o setor de patrimônio que com essa ferramenta irá automatizar seu processo de inventário coletando dados sobre os computadores da universidade.

A realização deste trabalho foi um desafio bastante gratificante. Onde foi possível empregar os conhecimentos adquiridos durante o curso de especialização. Temas como configuração de servidores linux e conceitos de rede puderam ser vistos de forma prática ajudando a ampliar estes conhecimentos aplicando a bagagem teórica aprendida nas disciplinas. Os resultados obtidos atenderam as necessidades e expectativas do projeto satisfazendo os motivos de sua implementação dando espaço para melhorias e projetos futuros.

## 6 – Bibliografia

(CARVILHE, 2000) CARVILHE, José Luís Vieira. A utilização de tecnologias WEB em sistemas de gerência corporativa. 2000. 102 f. Monografia (Especialização em Sistemas Distribuídos) - Pontifícia Universidade Católica do Paraná, Curitiba.

(DATAPREV, 2005) DATAPREV. Manual de implantação sistema de inventário CACIC Parte I – Introdução ao sistema. 2005.

(DATAPREV, 2006) DATAPREV. Manual de implantação sistema de inventário CACIC Parte II – Instalação do gerente em sistemas Debian GNU/Linux Instalação dos agentes. 2006.

(FERREIRA, 2006) Ferreira, V.. Aplicação do *Software* livre nas Instituições de Ensino Federal da Cidade de Jataí/GO. Monografias do Curso ARL, América do Sul, 0 4 09 2006.

(FOCALINUX, 2007) FOCALINUX. *Apache* [on-line]. Disponível na Internet via www. url:<http://focalinux.cipsga.org.br/guia/avancado/ch-s-apache.htm>. Arquivo capturado em 05 de janeiro de 2007.

(INFOWESTER, 2007) INFOWESTER. *Conhecendo o Servidor Apache* [on-line]. Disponível na Internet via www. url: <http://www.infowester.com/servapach.php>. Arquivo capturado em 05 de janeiro de 2007.

(INFOWESTER, 2007a) INFOWESTER. *Linguagem PHP* [on-line]. Disponível na Internet via www. url: <http://www.infowester.com/php.php>. Arquivo capturado em 05 de janeiro de 2007.

(INFOWESTER, 2007b) INFOWESTER. *Banco de dados MySQL e PostgreSQL* [on-line]. Disponível na Internet via www. url: <http://www.infowester.com/postgremysql.php>. Arquivo capturado em 15 de janeiro de 2007.

(RIBEIRO, 2006) Ribeiro, D.. *Software Livre na Administração Pública. Estudo de caso sobre adoção do SAMBA na Auditoria Geral do Estado de Minas Gerais. Monografias do Curso ARL, América do Sul, 2 9 08 2006.*

(UESC, 2007) UESC. *Universidade Estadual de Santa Cruz* [on-line]. Disponível na Internet via www. url: [http://www.uesc.br/a\\_uesc/](http://www.uesc.br/a_uesc/). Arquivo capturado em 18 de maio de 2007.

(WIKIPEDIA, 2006) WIKIPEDIA. *PHP* [on-line]. Disponível na Internet via www. url: <http://pt.wikipedia.org/wiki/PHP>. Arquivo capturado em 10 de dezembro de 2006.

(WIKIPEDIA, 2007) WIKIPEDIA. *Servidor Apache* [on-line]. Disponível na Internet via www. url: [http://pt.wikipedia.org/wiki/Servidor\\_Apache](http://pt.wikipedia.org/wiki/Servidor_Apache). Arquivo capturado em 05 de janeiro de 2007.