MACA – GUIA DE INSTALAÇÃO E CONFIGURAÇÃO

Gustavo Motta

Versão: 3.2.2

São Paulo, setembro de 2005

APRESENTAÇÃO

O objetivo deste manual é mostrar como configurar e instalar os componentes do MACA em ambientes LINUX e Windows. O objetivo do MACA é prover os serviços de autenticação de usuários, de gerência de sessões e de autorização de acesso para uma variedade de aplicações implementadas em diferentes plataformas, a partir de uma API padronizada. Os serviços do MACA são disponibilizados a partir da configuração e instalação dos seguintes componentes:

- **MACA CS**: oferece os serviços de autenticação de usuários, de gerência de sessões e de autorização de acesso;
- **MACA CS Monitor**: módulo administrativo para monitorar e controlar as sessões ativas de usuários e estatísticas de atividade do MACA CS;
- **MACA AD WEB**: módulo para gerência de contas de usuários, que permite a administradores de contas a criação, a alteração, a exclusão e a consulta a contas de usuários cadastrados no MACA. Adicionalmente, permite a gerência de senhas e a atribuição/exclusão de papéis de usuários;
- **MACA AD**: módulo para administração da política de autorização de acesso, que permite a administradores de autorizações a criação, a alteração, a exclusão e a consulta a recursos, papéis e autorizações.

As instruções para configuração e instalação destes componentes são apresentadas a partir do capítulo 2. O capítulo 1 faz uma breve apresentação do MACA e de sua arquitetura de software para facilitar o entendimento da relação entre os seus componentes.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

SUMÁRIO

APRESENTAÇÃO 2

SUMÁRIO 4

LISTA DE FIGURAS 7

LISTA DE TABELAS 9

1 INTRODUÇÃO 11

1.1 Arquitetura de Software do MACA 11
1.1.1 Base de Informações de Gerência de Segurança 12
1.1.2 MACA AD e MACA AD WEB 13
1.1.3 MACA CS 13
1.1.4 Implementação 13

2 MACA CS 15

2.1 Requisitos de Software 15
2.1.1 Serviço OpenLDAP – Instalação e Configuração 16
Habilitando o LDAP sobre TLS/SSL 25
Configurações Úteis 27
2.1.2 Serviço HTTP – Instalação e Configuração 27
2.1.3 Maquina Virtual Java – Instalação e Configuração 28
2.2 MACA CS – Instalação e Configuração 28
2.2.1 Configurações Úteis 31
Política de Senhas 31
Arquivos de Log 31
Usuários e Sessões 31
Definido Contextos Customizados 31
2.2.2 Configurando o MACA CS para Acessar o serviço LDAP via TLS/SSL 32
2.2.3 Configurando o MACA CS para Comunicação com os Clientes via IIOP sobre TLS/SSL 33
2.2.4 Configuração do MACA CS com Balanceamento de Carga e Tolerância a Falhas 36

3 MACA CS MONITOR 39

3.1 Requisitos de Software 39
3.1.1 Serviço JSP Tomcat – Instalação e Configuração 39
3.2 MACA CS Monitor – Instalação e Configuração 45
3.2.1 Configurando o MACA CS Monitor para Acessar o serviço MACA CS via IIOP sobre TLS/SSL 49

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

4 MACA AD WEB 53

4.1 MACA AD Web – Instalação e Configuração 53
4.1.1 Configurações Úteis 58
Política de Senhas 58
Definindo um Gerador de Senhas Customizado 58
Redefinição do Esquema de Dados da Conta de Usuários 59
Entendendo o Esquema Padrão 59
Definido Lookups para Atributos do Esquema de Dados 62
4.1.2 Configurando o MACA AD Web para Acessar o serviço LDAP via TLS/SSL 63
4.1.3 Configurando o MACA AD Web para Acessar o serviço MACA CS via IIOP sobre TLS/SSL 65

5 MACA AD 70

5.1 MACA AD – Instalação e Configuração 70
5.1.1 Configurando o MACA AD para Acessar o serviço LDAP via TLS/SSL 73
5.1.2 Configurando o MACA AD para Acessar o serviço MACA CS via IIOP sobre TLS/SSL 75

6 PROPRIEDADES DE CONFIGURAÇÃO 79

6.1.1 maca_ca.properties 79
6.1.2 maca_cs.properties 83
6.1.3 maca_bean.properties 85

7 DEFINIÇÕES E ACRÔNIMOS 86

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

LISTA DE FIGURAS

Figura 1 – Arquitetura de Software do MACA 12

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

LISTA DE TABELAS

Tabela 1 – Configuração dos arquivos de log – usado no MACA CS 79
Tabela 2 – Configuração da política de senhas– usado no MACA CS e MACA AD Web 80
Tabela 3 – Configuração do LDAP– usado no MACA CS, MACA AD Web e MACA AD .. 81
Tabela 4 – Configuração de segurança do LDAP– usado no MACA CS,
MACA AD Web e MACA AD 82
Tabela 5 – Miscelânea 82
Tabela 6 – Configuração de usuários e sessões 83
Tabela 7 – Configuração dos arquivos de log 83
Tabela 8 – Miscelânea 84
Tabela 9 – Configuração para o serviço de nomes 84
Tabela 10 – Configuração pelos módulos MACA AD Web e MACA CS Monitor 85

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

1 Introdução

O objetivo do MACA (***M**iddleware* de **A**utenticação e **C**ontrole de **A**cesso) é prover os serviços de autenticação de usuário e controle de acesso para aplicações legadas ou em desenvolvimento, independente de plataforma e linguagem de programação, através de uma API padronizada.

O MACA implementa um modelo de autorização contextual para o controle de acesso baseado em papéis (CABP) definido pelo NIST. O CABP tem características adequadas para definição e administração viável de políticas de acesso, particularmente em aplicações corporativas emergentes, que demandam um controle com granularidade fina para um grande número de usuários e recursos.

A arquitetura de software do MACA baseia-se em padrões de processamento aberto e distribuído a fim de alcançar interoperabilidade e portabilidade para segurança. Adota o serviço de diretórios LDAP como uma base de informações de gerenciamento de segurança (BIGS); o CORBA *Security Service* (CSS) para autenticação de usuários e o serviço de decisão para acesso a recursos (RAD – *Facility*), do CORBA *horizontal facilities*, como soluções de *middleware* para autenticação de usuários e solicitação de autorizações de acesso, respectivamente, por parte das aplicações clientes. Esta solução viabiliza a administração da política de autorização e o controle de acesso de modo unificado e consistente, a partir de diferentes sistemas, em plataformas e linguagens de programação distintas, mas de forma padronizada.

Os componentes principais integram esta arquitetura são: o servidor de autenticação e controle de acesso (MACA CS); o módulo para monitorar e controlar as sessões dos usuários (MACA CS MONITOR); o módulo para administração da política de autorização de acesso (MACA AD); o módulo para gerência de contas de usuários (MACA AD WEB); e aplicações clientes que solicitam autorizações de acesso e autenticação de usuários. Este manual descreve passo-a-passo o como configurar e instalar cada um destes módulos. A fim de facilitar seu uso, a subseção seguinte descreve a arquitetura do software do MACA.

1.1 Arquitetura de Software do MACA

A arquitetura onde os componentes do MACA estão implementados é apresentada na Figura 1. É um modelo cliente-servidor multicamada com os seguintes componentes principais: um servidor LDAP, encarregado de manter a base de gerência de informações de segurança; os módulos para administração da política de autorização de acesso (MACA AD) e para gerência de contas de usuários (MACA AD WEB); um servidor de segurança (MACA CS), encarregado de oferecer serviços de autenticação de usuário, de decisão de acesso a recursos e finalmente as aplicações cliente que requisitam estes serviços de segurança.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

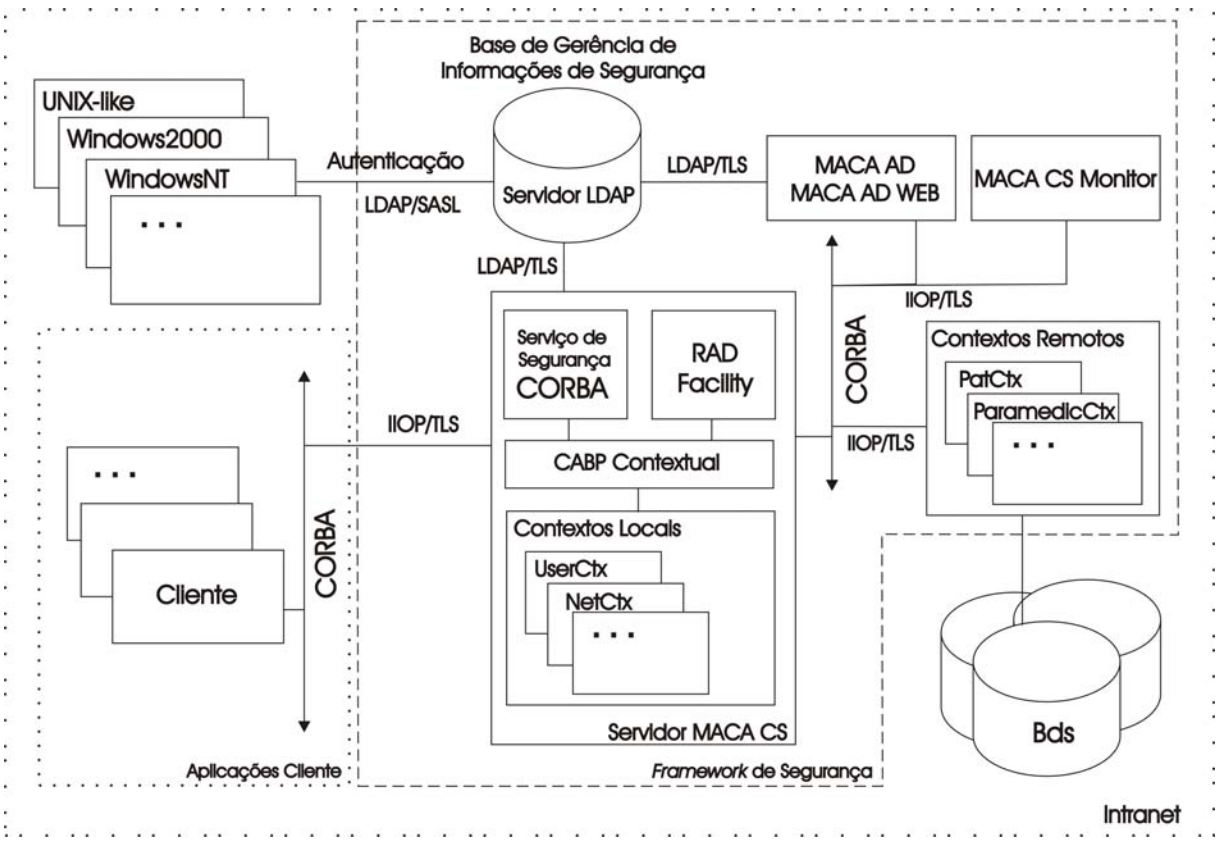


Figura 1 – Arquitetura de Software do MACA

Nesta arquitetura, as aplicações cliente são protegidas pela *intranet* da instituição que as mantém. Isto é, a *intranet* limita o acesso apenas para terceiros confiáveis através de conexões seguras (uma rede privada virtual, por exemplo). Entretanto, tanto dos acessos externos, quanto os internos, são regulados pela política de segurança armazenada na BIGS e estabelecida pela instituição mantenedora das aplicações.

1.1.1 Base de Informações de Gerência de Segurança

A BIGS (Figura 1) mantém num servidor LDAP configurações de segurança, tais como, autorizações de acesso, papéis, representações dos recursos protegidos e dos usuários, dados para autenticação, relacionamentos usuários-papéis, papéis-autorizações, etc. Todo acesso à BIGS deve ser realizado através do protocolo LDAP sobre TLS (*Transport Layer Security*) para assegurar confidencialidade e integridade na comunicação. Adicionalmente, TLS pode assegurar autenticação mútua entre clientes LDAP e o servidor LDAP. Ademais, autenticação via sistemas operacionais e gerenciadores de bancos de dados mais populares podem efetuadas com segurança junto ao servidor LDAP através do protocolo SASL. Esta solução viabiliza a autenticação unificada de usuários numa organização, independente de sistema operacional ou aplicação.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

1.1.2 MACA AD e MACA AD WEB

A administração da política de segurança é realizada na BIGS com o MACA AD (Figura 1) e apenas por usuários privilegiados através de interações seguras e com autenticação e controle de acesso adequados. De modo similar, o MACA AD WEB permite a gerência de contas de usuários via *web* com flexibilidade para definição dos atributos armazenados em cada conta no LDAP. O servidor LDAP deve ser protegido fisicamente e todos os acesso não locais devem ser desabilitados. A localização unificada das políticas de segurança facilita a administração, mas introduz uma sobrecarga adicional, visto que todos os clientes LDAP usam um único servidor. Ademais, um servidor central introduz um único ponto de falha e abre oportunidades de ataques para colocá-lo fora de serviço. Como LDAP é um serviço de diretório distribuído e a maioria das implementações disponíveis têm mecanismo de réplica automático, é viável construir um servidor logicamente centralizado e tolerante a falhas, embora fisicamente distribuído e redundante.

1.1.3 MACA CS

Cabe ao servidor de segurança (Figura 1) oferecer autenticação, autorização e controle de acesso às aplicações clientes, dentre outros serviços de segurança. O RAD – *Facility* oferece interfaces padronizadas que permitem o controle de acesso detalhado, ao nível da aplicação, mas de uma forma em que a lógica do controle de acesso é separada da lógica da aplicação, com transparência em relação ao modelo de decisão efetivamente implementado. Este *framework* é adequado para suportar o modelo de autorização do MACA, pois prevê o tratamento dos fatores dinâmicos que influenciam a lógica de autorização e possibilita a combinação de diferentes políticas de controle de acesso. O Serviço de Segurança CORBA oferece interface padrão para autenticação de usuários e faz o controle de acesso transparente para as operações definidas nos objetos CORBA. O módulo "CABP Contextual" implementa o modelo de autorização do MACA, sendo utilizado como política de autorização de acesso pelo CSS e pelo RAD – *Facility*. Todas as interações entre objetos CORBA cliente e o servidor de segurança devem ocorrer via IIOP (*Internet Inter-ORB Protocol*) sobre TLS.

1.1.4 Implementação

Os módulos MACA CS e MACA AD foram implementados em Java e o MACA AD WEB foi desenvolvido em Java/JSP. O MACA AD WEB permite a um administrador de contas criar, consultar, atualizar e remover contas usuários. O MACA AD permite a um administrador de autorizações estabelecer políticas de acesso através da definição de papéis, recursos e autorizações. O módulo "CABP Contextual" implementa o serviço de autenticação de usuários com *login* e senha, o gerenciador de sessões de usuários e o serviço de autorização de acesso. Ainda provê um interpretador para regras para autorização contextuais. Tais regras permitem que a política de acesso seja estabelecida com base em variáveis ambientais que denotam informações sobre usuário corrente, data/hora e local do

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

acesso, e outras que podem ser livremente programadas e incorporadas para especificação de políticas de autorização mais complexas. Os contextos locais são implementados em Java e compartilham o mesmo espaço de endereçamento do módulo “CABP Contextual”. Contextos remotos são acessados por clientes Java/CORBA via IIOP sobre TLS. Contextos são bibliotecas dinâmicas carregadas em tempo de execução através do mecanismo de extensão de Java.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

2 MACA CS

O MACA CS requer para o seu funcionamento a instalação prévia ou a disponibilidade de um conjunto de outros componentes de software, a saber:

1. Serviço de diretórios LDAP OpenLDAP;
2. Serviço HTTP, tais como o Apache, o Tomcat ou o MS-IIS;
3. Maquina virtual Java.

A seção 2.1 traz as instruções para efetuar as instalações e configurações prévias necessárias para o correto funcionamento do MACA CS. A seção 2.2 mostra como instalar e configurar o MACA CS. Recomenda-se que todas as instalações dos componentes do MACA (MACA CS, MACA CS Monitor, MACA AD Web e MACA AD) sejam configuradas inicialmente sem utilizar TLS/SSL, ou seja, sem aplicar as configurações recomendadas nas subseções 2.2.2, 2.2.3, 3.2.1, 4.1.2, 4.1.3, 5.1.1 e 5.1.2. Estas configurações poderão ser feitas posteriormente.

Antes de iniciar a instalação do MACA CS, baixe o pacote de instalação da versão escolhida do MACA no *site* http://maca.sourceforge.net/index.html#Downloads e o descompacte com o seguinte comando no LINUX:

```
gunzip -c maca_deployment_VERSION.tgz | tar xf -
```

Caso esteja no MS-Windows, utilize o utilitário WinRar, disponível no *site* http://www.rarlab.com/, para descompactá-lo.

2.1 Requisitos de Software

As instruções para instalação contidas neste manual visam facilitar a instalação dos componentes de software necessários ao MACA CS, mas em nenhuma circunstância substituem a documentação de instalação original destes componentes. Deste modo, não há garantias de que tais instruções estejam completas e que funcionarão para quaisquer versões de sistemas operacionais. A documentação original deve sempre ser consultada e estudada. A instalação recomendada neste manual usou as seguintes versões de produtos:

- Linux 2.4.27-2-386 #1 Thu Jan 20 10:55:08 JST 2005 i686 GNU/Linux;
- OpenLDAP 2.2.26;
- OpenSSL 0.9.7g;

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

- BerkeleyDB 4.1.25 (ou gdbm 1.8.3);

Os limites mínimos de versões quando não indicados neste texto devem ser consultados diretamente nos manuais originais dos componentes requeridos.

2.1.1 Serviço OpenLDAP – Instalação e Configuração

O OpenLDAP é um componente de instalação opcional na maioria das distribuições do sistema operacional LINUX. Há pelo menos uma versão de distribuição para o sistema operacional MS-Windows, disponível em http://www.ilex.fr/download/install-openldap-windows.exe, com instalação automática, incluindo as configuração de TLS. Certifique-se de que a versão do OpenLDAP disponível para o seu sistema operacional é uma versão estável 2.0 ou superior. Isto porque o MACA CS requer a versão 3 do protocolo LDAP, somente disponível a partir do OpenLDAP 2.0.

Caso o seu sistema operacional não tenha o OpenLDAP disponível ou se for desejado instalar uma versão mais recente do produto, você poderá obtê-lo, juntamente com o guia do administrador contendo instruções de instalação no *site* do projeto http://www.openldap.org. É recomendável instalar uma versão estável do OpenLDAP obtida diretamente deste *site*. Isto porque as versões de instalação disponíveis nos sistemas operacionais, em geral, não estão compiladas para suportar o protocolo LDAP sobre TLS (*Transport Layer Security*). A utilização do TLS é essencial para assegurar a confidencialidade, a autenticação e a integridade das trocas de informações entre o OpenLDAP e seus clientes. A utilização do LDAP sobre TLS, embora desejável, não é um requisito para funcionamento do MACA e seus componentes.

Siga os seguintes passos para instalação e configuração do OpenLDAP para funcionamento com o MACA CS no sistema operacional LINUX[1]:

1. Baixe o pacote de instalação da versão escolhida do OpenLDAP no *site* http://www.openldap.org/software/download;
2. Verifique a versão do núcleo de seu sistema operacional digitando o seguinte comando:

   ```
   uname -a
   ```

   A versão do núcleo do LINUX deverá ser 2.4.x ou superior;
3. Após baixar o arquivo, ele dever ser descompactado com o seguinte comando:

[1] A instalação no MS-Windows é autoexplicativa, mas é obrigaria que a configuração do OpenLDAP seja feita tal como indicada a partir do passo 13 desta seção.

```
gunzip -c openldap-VERSION.tgz | tar xf -
```

4. Em seguida, mude para o diretório recém criado com o comando:

```
cd openldap-VERSION
```

5. Obtenha os privilégios de *root* se ainda não os obteve executando o comando:

```
su -
```

6. Agora, é preciso executar o utilitário de configuração da instalação do OpenLDAP para especificar a utilização do TLS (opcional) e também para verificar se o sistema operacional tem todos os pacotes de software necessários para compilação/instalação do OpenLDAP. Para executar a configuração, entre com o seguinte comando:

```
./configure --with-tls --prefix=/usr/local --sysconfdir=/etc
--localstatedir=/var --enable-referrals
```

A opção `--with-tls` prepara a configuração de compilação para utilização do TLS e requer que as bibliotecas do pacote OpenSSL estejam disponíveis no sistema operacional. Caso a configuração ocorra sem erros, pule os dois passos seguintes (7 e 8) e vá direto para o passo 9. Caso o configure acuse erro pela ausência do OpenSSL, siga as instruções do passo 7. Caso o configure acuse erro pela ausência das bibliotecas do BerkeleyDB, siga as instruções do passo 8. Em vez de utilizar o BerkeleyDB, pode-se configurar o OpenLDAP para usar o pacote de banco de dados *gdbm*. No caso, siga as instruções do passo 9. **O OpenLDAP é mais estável com o gdbm**.

Caso o comando de configuração continue a acusar a falta das bibliotecas do BerkeleyDB, mesmo após a sua instalação bem sucedida no passo 8, então, execute os comandos a seguir. Cada bloco de linhas abaixo deve ser executado como uma única linha de comando no prompt:

```
LD_LIBRARY_PATH=/usr/local/BerkeleyDB.VERSION/lib; export
LD_LIBRARY_PATH

env CPPFLAGS="-I/usr/local/BerkeleyDB.VERSION/include"
LDFLAGS="-L/usr/local/BerkeleyDB.VERSION/lib" ./configure
--with-tls --prefix=/usr/local --sysconfdir=/etc
--localstatedir=/var --enable-referrals
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Se tudo ocorreu bem, vá para o passo 10.

7. Caso o pacote OpenSSL não esteja disponível, ele poderá ser obtido no site http://www.openssl.org. Baixe o pacote de instalação da versão estável mais recente do OpenSSL (até o momento, a 0.9.7g) e execute os seguintes comandos para instalá-lo:

```
gunzip -c openssl-VERSION.tar.gz | tar xf -
cd openssl-VERSION
./config --prefix=/usr/local --openssldir=/usr/local/openssl
make
make test
make install
```

Volte para o passo 6 e execute o comando de configuração.

8. Caso o pacote BerkeleyDB não esteja disponível, ele poderá ser obtido no site http://www.sleepycat.com/download/index.shtml. Baixe o pacote de instalação da versão estável mais recente do BerkeleyDB (até o momento, a 4.1.25) e execute os seguintes comandos para instalá-lo:

```
gunzip -c db-VERSION.tar.gz | tar xf -
cd db-VERSION
cd build_unix
../dist/configure
make
make install
```

Volte para o passo 6 e execute o comando de configuração.

9. Caso o pacote *gdbm* não esteja disponível, ele poderá ser obtido no site ftp://ftp.gnu.org/gnu/gdbm/. Baixe o pacote de instalação da versão estável mais recente do OpenSSL (até o momento, a 1.8.3) e execute os seguintes comandos para instalá-lo:

```
gunzip -c gdbm-VERSION.tar.gz | tar xf -
cd gdbm-VERSION
./configure
make
make test
make install
```

Volte para o passo 6 e execute o comando de configuração.

10. Com a configuração realizada com sucesso, continue executando a instalação do OpenLDAP com os comandos abaixo:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
make depend
make
make test
make install
```

11. Com a instalação bem sucedida, deve-se localizar os diretórios onde os binários e os arquivos de configuração do OpenLDAP estão. De acordo com a configuração fornecida anteriormente, os binários estarão em

    ```
    /usr/local/libexec/
    ```

    e os arquivo de configuração em

    ```
    /etc/openldap/
    ```

12. Agora, inicie o serviço LDAP executando o seguinte comando:

    ```
    /usr/local/libexec/slapd -f /etc/openldap/slapd.conf
    ```

    A opção `-f` indica para o servidor do OpenLDAP onde encontrar os arquivos de configuração. Verifique se o serviço foi iniciado corretamente digitando

    ```
    ps -ef | grep slapd
    ```

13. Agora, encerre o serviço LDAP executando o seguinte comando:

    ```
    kill -INT `cat /var/slapd.pid`
    ```

    ou então usando o PID obtido com o comando `ps`

    ```
    kill -9 <pid>
    ```

14. Com o serviço LDAP corretamente instalado e pronto para funcionar, pode-se iniciar os passos para configurá-lo de acordo com as necessidades do MACA CS. Mude para o diretório de configuração do OpenLDAP com o comando:

    ```
    cd /etc/openldap/
    ```

15. Copie o arquivo `maca_deployment_VERSION/ldap.confs/incor.schema` do diretório de distribuição do MACA para o diretório `/etc/openldap/schema/`

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

16. Edite o arquivo `/etc/openldap/slapd.conf` fazendo as seguintes alterações.

    Abaixo da linha

    ```
    include         /etc/openldap/schema/core.schema
    ```

    inclua o arquivo com os esquemas requeridos pelo MACA CS acrescentando as linhas seguintes,

    ```
    include         /etc/openldap/schema/cosine.schema
    include         /etc/openldap/schema/nis.schema
    include         /etc/openldap/schema/inetorgperson.schema
    include         /etc/openldap/schema/incor.schema
    ```

    caso ainda não estejam presentes. Grave o arquivo e saia para o prompt para testar a nova configuração iniciando o serviço do OpenLDAP em modo de depuração para verificar se há algum erro. Para isto, basta executar o seguinte comando:

    ```
    /usr/local/libexec/slapd -d 256 -f /etc/openldap/slapd.conf
    ```

    Caso tudo ocorra bem, a seguinte saída é esperada:

    ```
    bdb_initialize: Sleepycat Software: Berkeley DB 4.1.25: (Dec...
    bdb_db_init: Initializing BDB database
    slapd starting
    ```

    Pressione `control-c` para encerrar o serviço. Caso o OpenLDAP acuse erros porque há classes ou atributos no `incor.schema` que já estão definidos, então edite o arquivo `/etc/openldap/schema/incor.schema` e comente a definição destas classes ou atributos e tente novamente iniciar o serviço;

17. Tudo correndo bem, edite novamente o arquivo `/etc/openldap/slapd.conf` fazendo as seguintes alterações:

    Modifique a diretiva `suffix` para refletir o domínio de sua organização. Por exemplo, no Instituto do Coração (InCor) da Faculdade de Medicina da USP, o domínio do DNS é `incor.usp.br`, logo o sufixo de seu serviço LDAP `dc=incor,dc=usp,dc=br` e corresponde à raiz da árvore de informações do LDAP. Portanto, altere o sufixo conforme indicado

abaixo, substituindo os componentes de domínio apresentados pelos componentes de domínio de sua organização:

```
suffix          "dc=organizacao,dc=com,dc=br"
```

Depois, modifique o nome do usuário *root* do serviço, que tem privilégios de administrador do diretório. Neste caso, altere o sufixo presente na propriedade `rootdn` para refletir o sufixo definido na propriedade anterior. Assim o valor da propriedade modifica-se para:

```
rootdn          "cn=Manager,dc=organizacao,dc=com,dc=br"
```

Para fins de teste e instalação, a senha do *root* definida na propriedade `rootpw` pode permanecer. Entretanto, é altamente recomendável a sua mudança após a instalação e configuração, definido-a no formato SSHA. Para isto, no prompt digite o comando:

```
slappasswd
```

Entre com uma nova senha, pressione enter, repita a senha e pressione enter de novo. Este utilitário retornará para você o *hash* da senha, como no exemplo abaixo:

```
{SSHA}nr9qBfUOeQqYoXuo2RMoLmMCaN3P61Ez
```

Pegue este valor retornado e defina-o como valor da propriedade `rootpw`, conforme ilustrado a seguir:

```
rootpw          {SSHA}nr9qBfUOeQqYoXuo2RMoLmMCaN3P61Ez
```

Assim, a senha do *root* não fica mais armazenada em texto limpo, que permite uma leitura direta;

18. O próximo passo é configurar os índices de banco de dados que o OpenLDAP deverá manter. A não definição destes índices implica em grandes perdas de desempenho por parte do MACA CS. Ainda editando o arquivo `/etc/openldap/slapd.conf`, faça o seguinte:

    Abaixo da linha

    ```
    index   objectClass     eq
    ```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Acrescente as seguintes linhas com valores para propriedade `index`:

```
index userPassword pres
index uid,cn,sn,givenname,mail,ou,employeetype eq,sub,pres
index employeenumber,uidnumber,gidnumber,member eq,pres
index incorroledn,incorresourcedn eq,sub,pres
index incorprivilege,incorprivilegetype,incorauthorizationtype eq,pres
```

Estas linhas de configurações de índice podem ser obtidas no arquivo `maca_deployment_VERSAO/ldap.confs/index.conf` do diretório de distribuição do MACA

19. Para finalizar esta configuração inicial, é preciso definir a política de acesso interna do servidor LDAP. Basicamente, o que se vai fazer é desabilitar o acesso anônimo, dando direitos de leitura/consulta apenas para usuários autenticados, que podem alterar apenas a própria conta. Também são definidos privilégios de leitura, escrita e consulta para os administradores do diretório LDAP. Para efetuar estas configurações, acrescente no final do arquivo `/etc/openldap/slapd.conf`, ainda em edição, as linhas abaixo. Antes, porém, remova quaisquer listas de controle de acesso que eventualmente estejam definidas. **Não esqueça de substituir nestas linhas todas as ocorrências do sufixo `dc=organizacao,dc=com,dc=br` pelo sufixo definido anteriormente na propriedade `suffix`. Não deixe espaços entre as vírgulas!**

```
########################################
# Listas para controle de acesso - MACA #
########################################

access to attr=userpassword,passwordHistory,passwordExpirationTime,passwordExpWarned,passwordMustChange,passwordAllowChangeTime
  by self write
  by group/incorgroup/member="cn=Administrador de Contas,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by * compare

access to attr=passwordlocked,passwordLockTime,passwordUnlockTime
  by group/incorgroup/member="cn=Administrador de Contas,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by * compare

access to dn.subtree="ou=people,dc=organizacao,dc=com,dc=br"
  by self write
  by group/incorgroup/member="cn=Administrador de Contas,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by users read

access to dn.subtree="ou=groups,dc=organizacao,dc=com,dc=br"
  by group/incorgroup/member="cn=Administrador de Contas,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by group/incorgroup/member="cn=Administrador de Autorizacao,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by users read

access to dn.subtree="ou=resources,dc=organizacao,dc=com,dc=br"
  by group/incorgroup/member="cn=Administrador de Autorizacao,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by users read

access to dn.subtree="ou=authorizations,dc=organizacao,dc=com,dc=br"
  by group/incorgroup/member="cn=Administrador de Autorizacao,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" write
  by users read

access to dn.subtree="dc=organizacao,dc=com,dc=br"
  by group/incorgroup/member="cn=Administrador de Contas,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" read
  by group/incorgroup/member="cn=Administrador de Autorizacao,cn=administrador,cn=papeis,ou=groups,dc=organizacao,dc=com,dc=br" read
  by users read
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Estas linhas de configurações de controle de acesso podem ser obtidas no arquivo `maca_deployment_VERSAO/ldap.confs/acls.conf` do diretório de distribuição do MACA

20. Grave o arquivo `/etc/openldap/slapd.conf` e saia para o prompt de comando a fim de testar a nova configuração iniciando o serviço do OpenLDAP em modo de depuração. Para isto, basta executar o seguinte comando:

```
/usr/local/libexec/slapd -d 256 -f /etc/openldap/slapd.conf
```

Caso tudo ocorra bem, a seguinte saída é esperada:

```
bdb_initialize: Sleepycat Software: Berkeley DB 4.1.25: (Dec...
bdb_db_init: Initializing BDB database
slapd starting
```

21. Pressione `control-c` para encerrar o serviço e reinicie o OpenLDAP no modo normal, conforme indicado abaixo:

```
/usr/local/libexec/slapd -f /etc/openldap/slapd.conf
```

22. Agora, OpenLDAP está pronto para receber a carga com o banco de dados inicial para funcionamento dos componentes do MACA. Este banco de dados está no formato textual LDIF (veja a seção 7.3 em http://www.openldap.org/doc/admin21/dbtools.html para saber mais sobre o formato). Antes da carga, faça o seguinte:

Copie o arquivo com o banco de dados `maca_deployment_VERSAO/ldap.confs/maca_inicial.ldif` do diretório de distribuição do MACA para o diretório do OpenLDAP `/etc/openldap/`;

Edite este arquivo e substitua todas as ocorrências do texto `organizacao` pelo texto corresponde à organização no nome do sufixo que foi definido anteriormente na propriedade `suffix`. Repita esta operação para o texto `com`, se for o caso. Não se esqueça de procurar apenas por palavras inteiras no momento da troca! Por exemplo, se `organização` fosse substituído por `incor`, e `com` por `gov`, o primeiro registro do arquivo ficaria assim:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
dn: dc=incor,dc=gov,dc=br
objectClass: top
objectClass: dcobject
objectClass: organization
o: dc=incor,dc=gov,dc=br
dc: incor
```

Usando o arquivo modificado, proceda a carga no banco do OpenLDAP executando o comando abaixo:

```
ldapadd -f /etc/openldap/maca_inicial.ldif -x
-D "cn=Manager,dc=organizacao,dc=com,dc=br" -W
```

Não esqueça de substituir o sufixo `dc=organizacao,dc=com,dc=br` pelo sufixo definido na propriedade `suffix`. **Caso ocorra algum erro**, corrija o problema no arquivo LDIF. Antes de repetir o comando anterior para adicionar o banco, é preciso remover às entradas que eventualmente foram carregadas executando o comando abaixo:

```
ldapdelete -r -v -x -D "cn=Manager,dc=organizacao,dc=com,dc=br"
-W "dc=organizacao,dc=com,dc=br"
```

Não esqueça de substituir o sufixo `dc=organizacao,dc=com,dc=br` pelo sufixo definido na propriedade `suffix`. Agora repita o comando para adicionar as entradas.

23. Com o banco carregado, verifique o seu conteúdo executando o comando de busca abaixo:

```
ldapsearch -x -D "cn=Manager,dc=organizacao,dc=com,dc=br" -W
-LLL -s sub -b "dc=organizacao,dc=com,dc=br"  "(uid=*)"
```

Não esqueça de substituir o sufixo `dc=organizacao,dc=com,dc=br` pelo sufixo definido na propriedade `suffix`. O resultado obtido deverá ser o seguinte:

```
dn: uid=nome.sobrenome,ou=people,dc=organizacao,dc=com,dc=br
objectClass: top
objectClass: person
objectClass: incorperson
objectClass: inetorgperson
objectClass: passwordobject
uid: nome.sobrenome
givenName: Nome Sobrenome
cn: Nome
sn: Sobrenome
mail: nome.sobrenome@organizacao.com.br
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
employeeNumber: 00000000000
userPassword:: bWFjYQ==
```

Esta entrada no diretório corresponde ao usuário com privilégios de administrador do MACA. Fazendo o login no MACA AD WEB, este usuário poderá criar contas de usuários, incluindo usuários com papéis de administradores. Posteriormente, este usuário poderá ser excluindo da base ou ser renomeado por um outro administrador de contas.

Com este último passo, o servidor OpenLDAP está minimamente configurado para suportar o MACA.

Habilitando o LDAP sobre TLS/SSL

A habilitação do TLS/SSL no OpenLDAP vai permitir a confidencialidade e a integridade da comunicação entre clientes e servidor. Esta habilitação é altamente recomendada em ambientes de produção porque protege senhas em trânsito durante a autenticação de usuários. Para configurar o LDAP sobre TLS/SSL, certifique-se que o pacote OpenSSL esteja instalado e que o OpenLDAP tenha sido compilado com a opção TLS (ver item 6). Para habilitação, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*);
2. Mude para o diretório onde o OpenLDAP foi instalado, em geral, o diretório `/etc/openldap/`;
3. Crie nele um subdiretório denominado `certificados` com o comando abaixo e mude-se para ele:

```
mkdir ./certificados
cd ./certificados
```

4. Agora, crie uma requisição de certificado para o serviço OpenLDAP com o comando abaixo:

```
openssl req -new > new.cert.csr
```

Ele fará uma série de perguntas, que devem ser respondidas segundo o modelo a seguir:

```
Using configuration from /usr/share/ssl/openssl.cnf
Generating a 1024 bit RSA private key
..................++++++
.............++++++
writing new private key to 'privkey.pem'
Enter PEM pass phrase:<entre com a senha da chave privada>↵
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
Verifying password - Enter PEM pass phrase: <repita a senha anterior>↵
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:BR↵
State or Province Name (full name) [Some-State]:<sigla do seu estado>↵
Locality Name (eg, city) []:<nome da sua cidade>↵
Organization Name (eg, company) [Internet Widgits Pty Ltd]: <sua empresa>↵
Organizational Unit Name (eg, section) []:<nome do seu setor>↵
Common Name (eg, YOUR name) []:<entre com o DNS do servidor LDAP!>↵
Email Address []:<entre com o e-mail do administrador do OpenLDAP>↵

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:<entre com uma senha>↵
An optional company name []:<entre o nome de sua empresa>↵
```

Com isto, o comando cria uma requisição de certificado (arquivo `new.cert.csr`) e a chave privada correspondente (arquivo `privkey.pem`);

5. Com esta requisição, pode-se solicitar um certificado digital para alguma Autoridade Certificadora (AC) pública ou existente no mercado privado. Para fins de teste, porém, pode-se criar um certificado auto assinado com o comando abaixo:

```
openssl x509 -in new.cert.csr -out ldap.cert -req
-signkey privkey.pem -days 365
```

Entre com senha da chave privada para permitir a criação do certificado. Este comando cria o certificado no arquivo `ldap.cert`;

6. Agora, edite o arquivo /etc/openldap/slapd.conf e defina as seguintes propriedades:

```
#######################################################################
# Habilita SSL
#######################################################################

TLSCertificateFile /etc/openldap/certificados/ldap.cert
TLSCertificateKeyFile /etc/openldap/certificados/privkey.pem
```

Grave o arquivo e saia para o prompt de comando;

7. Agora, reinicie o serviço LDAP executando os seguintes comandos:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
kill -INT `cat /var/slapd.pid`
/usr/local/libexec/slapd -f /etc/openldap/slapd.conf -h "ldaps:///"
```

É necessário entrar com a senha da chave privada para ativa o TLS/SSL.

A opção `-h` indica para o servidor do OpenLDAP habilitar o serviço exclusivamente no modo LDAPS (LDAP sobre TLS/SSL), disponível na porta 636. O serviço LDAP normal, quando habilitado, fica disponível na porta 389. Para habilitar ambas os modos (não recomendável), a opção `-h` deve ser na seguinte forma:

```
-h "ldap:/// ldaps:///"
```

Configurações Úteis

Alguns valores default de propriedades do OpenLDAP devem ser modificados ou acrescentados no arquivo `/etc/openldap/slapd.conf` de acordo com a necessidade, sendo os mais comuns os seguintes:

- Alterar o default de 500 da propriedade `sizelimit` para 5000. Esta propriedade indica o número máximo entradas retornando numa consulta;
- Sendo o LDAP um serviço crítico, do qual um grande número de outros serviços e aplicações são dependentes, é recomendável configurar outros servidores/serviços LDAP para atuarem como réplicas. Com isto é possível obter maior tolerância a falhas e balanceamento de carga. As instruções de como configurar o mecanismo de réplica do OpenLDAP são detalhadas em http://www.openldap.org/doc/admin21/replication.html.

2.1.2 Serviço HTTP – Instalação e Configuração

O MACA CS requer a disponibilidade no servidor em que executa de um serviço HTTP padrão. Se o MACA CS for instalado num ambiente LINUX, poderá ser usado tanto o serviço HTTP do Apache, quanto o serviço HTTP do Tomcat. No caso do Windows NT/2000, pode-se usar o Tomcat ou o serviço HTTP do MS-IIS.

A instalação de um destes serviços é bastante simples, motivo pelo qual não a apresentaremos aqui. Tampouco o MACA CS requer configurações especiais neles. As versões destes serviços que já foram utilizadas com sucesso com o MACA CS são apresentadas na lista abaixo, bem como os *sites* onde são encontradas instruções de instalação. É importante instalar versões estáveis destes produtos.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

- Apache HTTP Server 1.3.23 (Unix) – http://httpd.apache.org;
- Tomcat 4.0.6 (Unix/Windows NT/2000) – http://jakarta.apache.org/tomcat/index.html;
- MS-IIS 5.0 (Windows NT/2000) - http://www.microsoft.com

2.1.3 Maquina Virtual Java – Instalação e Configuração

O MACA CS requer a disponibilidade do ambiente de execução Java (JRE) versão 4.0 ou superior. O MACA CS foi exaustivamente testado em ambientes LINUX (2.4.18-4GB) e Windows 2000 (5.0.2195 *Service Pack* 3) com a máquina virtual *Java™ 2 Runtime Environment, Standard Edition (build 1.4.1_01-b01* e build *1.5.0_04-b05)*. Os ambientes de execução Java podem ser obtidos no site http://java.sun.com/j2se/downloads.html. Recomenda-se efetuar a instalação com as configurações padrão para o sistema operacional escolhido. As instruções de instalação para diversos sistemas operacionais estão em site http://java.sun.com/j2se/1.5.0/download.jsp. Não esquecer de definir a variável de ambiente JAVA_HOME, especificando o local de instalação do JRE. Você pode defini-la na sua sessão com o seguinte comando no Linux:

```
JAVA_HOME=<path do JRE>; export JAVA_HOME
```

2.2 MACA CS – Instalação e Configuração

A instalação e configuração básica do MACA CS é bastante simples, considerando-se que os softwares e serviços requisitados por ele estão disponíveis. O MACA CS pode ser instalado no mesmo servidor em que o OpenLDAP foi instalado ou numa outra máquina. Entretanto, o serviço HTTP e a máquina virtual Java têm de estar disponíveis no mesmo servidor do MACA CS. O desempenho é melhor quando todos estes serviços são instalados numa mesma máquina.

Para instalar e configurar o MACA CS, proceda da seguinte forma:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Copie todo o diretório `maca_deployment_VERSAO/maca_cs/` do diretório de distribuição do MACA para o local de sua preferência no servidor onde o MACA CS está sendo instalado. No LINUX, recomenda-se copiá-lo abaixo do diretório `/usr/local/`, considerado o diretório escolhido nestas instruções;
3. Mude o diretório corrente para `/usr/local/maca_cs/bin`;

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

4. Edite o arquivo `maca_ca.properties` e efetue as seguintes alterações:

   Troque o valor da propriedade `maca_ca.ldap.server` para o nome DNS ou endereço IP onde o servidor OpenLDAP foi instalado. Caso seja no mesmo servidor, mantenha o valor `127.0.0.1`;

   Troque o valor da propriedade `maca_ca.ldap.rootDN` para o nome do sufixo escolhido (valor da propriedade `suffix`) durante a instalação do OpenLDAP. **Atenção, não deixe espaços em branco entre as vírgulas e use o caracter de *escape* \ antes do sinal de igual (\=)**;

5. Salve o arquivo;

6. Edite o arquivo `maca_cs.properties` e efetue as seguintes alterações:

   Troque o valor da propriedade `maca_cs.IOROutputDir` para caminho do diretório raiz do serviço HTTP. Se for o servidor Apache, o diretório padrão é o `/usr/local/httpd/htdocs`. Com o MS-IIS, o caminho padrão é o `/Inetpub/wwwroot`. Certifique-se do diretório correto foi definido para a propriedade;

7. Salve o arquivo;

8. Certifique-se se a variável de ambiente JAVA_HOME esta corretamente definida e acrescente o status executável aos arquivos de script `maca_cs_start.sh` e `maca_cs_stop.sh` com os comandos abaixo:

```
chmod a+x maca_cs_start.sh
chmod a+x maca_cs_stop.sh
```

9. Para iniciar o serviço do MACA CS, execute o comando abaixo:

```
./maca_cs_start.sh
```

   A seguinte saída é apresentada se tudo ocorrer bem:

```
   MACA 3.2.2 - Middleware de AutenticaþÒo e Controle de Acesso
   Copyright (c) 2000-2005 FundaþÒo Zerbini/Instituto do CoraþÒo - HC.FMUSP Arquivo
Recurso /maca_ca.properties carregado
Arquivo ./maca_ca.properties carregado
Arquivo com esquema de usuarios '' nao localizado ou inexistente: esquema default carregado
LDAP server: 127.0.0.1
Users root : ou=people,dc=organizacao,dc=com,dc=br
Arquivo ./maca_cs.properties carregado
Arquivo ./orb.properties carregado
        JacORB V 1.4.1, www.jacorb.org
        (C) Gerald Brose, FU Berlin/XTRADYNE Technologies, July 2002
[ POA RootPOA - ready ]
[ POA pa_POA - ready ]
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
[ POA ado_POA - ready ]
[ POA monitor_POA - ready ]
[ POA creds_POA - ready ]
pa IOR:000000000000003649444C3A6F6D672E6F72672F536563757269747 94C6576656C322F507
2696E636970616C41757468656E74696361746F723A312E3000000000000002000000000000000740
00102000000000F3230302E3134342E37332E3130360000B79000000000001A5374616E646172644
96D706C4E616D652F70615F504F412F706100000000000020000000000000008000000004A4143000
0000001000000 1C000000000001000100000001050100010001010900000000105010001000000010
000002C000000000000000100000001000000 1C000000000010001000000010501000100010109 0
000000105010001
ado IOR:000000000000003849444C3A6F6D672E6F72672F44665265736F7572636541636365737 3
4465636973696F6E2F41636365737344656369736973696F6E3A312E3000000000020000000000000074
00010200000000F3230302E3134342E37332E3130360000B79000000000001C5374616E64617264
496D706C4E616D652F61646F5F504F412F61646F000000020000000000000008000000004A414300
0000000100000 1C000000000001000100000001050100010001010900000000105010001000000001
0000002C000000000000000100000001000000 1C000000000010001000000010501000100010109
0000000105010001
monitor IOR:000000000000001849444C3A6D6163615F63732F4D6F6E69746F723A312E30000000
0002000000000000007C000102000000000F3230302E3134342E37332E3130360000B790000000000
00245374616E64617264496D706C4E616D652F6D6F6E69746F725F504F412F6D6F6E69746F720000
0002000000000000000800000000 4A414300000000010000001C000000000001000100000001050 1
0001000101090000000105010001000000010000002C000000000000000100000001000000 1C0000
0000000100010000000105010001000101090000000105010001
Contexto dtCtx[contexts.DateTimeContext] carregado.
MACA_CS iniciado.
ServiceStopper: escutando na porta 828
```

Verifique se os arquivos `maca_cs.iors`, `monitor.ior`, `pa.ior` e `ado.ior` foram criados no diretório raiz do serviço HTTP. Caso estejam e não sejam arquivos antigos, preexistentes, então o MACA CS foi iniciado com sucesso;

Observação: o serviço do MACA CS somente é iniciado/encerrado corretamente se o diretório corrente for o `maca_cs/bin/`, para que ele possa acessar os arquivos de configuração e as bibliotecas (presentes em `maca_cs/lib/`);

Atenção: o serviço do MACA CS deve ser reiniciado sempre que o OpenLDAP for reiniciado!

10. Para encerrar o serviço do MACA CS, execute o comando abaixo:

```
./maca_cs_stop.sh
```

Quando o serviço é encerrado normalmente (sem usar o comando `kill`/`killall` ou matando uma tarefa no Windows), os arquivos criados pelo MACA CS no diretório raiz do serviço HTTP são removidos. O comandos `kill`/`killall` somente devem ser usados quando o MACA CS não consegue ser encerrando normalmente com o comando anterior.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

2.2.1 Configurações Úteis

Algumas configurações default do MACA CS são bastante simples para funcionamento em ambientes de produção. Nesta seção, recomendamos algumas configurações úteis.

Política de Senhas

No arquivo `maca_ca.properties`, recomenda-se definir as seguintes propriedades:

```
maca_ca.passwordPolicy.passwordCheckSyntax=true
maca_ca.passwordPolicy.passwordMustChange=true
```

Veja a Tabela 2 com a descrição destas e de outras propriedades para a política de senhas. **Atenção, as propriedades definidas aqui DEVEM ser as mesmas definidas para o MACA AD Web (ver seção 4.1.1).**

Arquivos de Log

No arquivo `maca_ca.properties`, recomenda-se definir as seguintes propriedades:

```
maca_ca.log.LogFileCreationFrequency=M
maca_ca.log.LogMaxDirSize=1000
```

Veja a Tabela 1 com a descrição destas e de outras propriedades para arquivos de log.

Usuários e Sessões

No arquivo `maca_cs.properties`, recomenda-se definir as seguintes propriedades:

```
maca_cs.user.cache=true
maca_cs.user.maxNumber=500
maca_cs.user.maxSessionsPerUser=25
maca_cs.ldap.userAccountAttrList=givenname, employeenumber, mail
```

Veja a Tabela 6 com a descrição destas e de outras propriedades usuários e sessões.

Definido Contextos Customizados

O MACA CS permite a criação de classes que implementam contextos customizados, dependentes de ambientes organizacionais específicos. Para realizar isto, basta implementar a interface `Context.java` disponível no diretório maca_deployment_VERSAO/maca_cs/contextos/src/contexts do diretório de distribuição do MACA. Neste diretório há um exemplo (classe `DateTimeContextExample.java`) de classe que com uma implementação desta interface. Para torná-la disponível, compile esta classe e coloque o seu código objeto

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

(`DateTimeContextExample.class`) disponível no CLASS_PATH da máquina virtual utilizada pelo MACA CS. Outra forma de torná-la disponível é gerar um arquivo `.jar` com o código objeto da classe e colocá-lo no diretório `$JAVA_HOME/jre/bin/ext/`. Depois, basta incluir o nome do contexto, com respectiva classe, na propriedade `maca_cs.contexts` (ver Tabela 8). Exemplos de utilização de contextos são mostrados no manual "MACA Administrativo - Manual do Usuário", integrante do diretório de distribuição do MACA.

2.2.2 Configurando o MACA CS para Acessar o serviço LDAP via TLS/SSL

Antes de configurar o MACA CS para acessar o serviço LDAP sobre TLS/SSL (LDAPS), certifique-se que o LDAPS está habilitado. Neste caso, obtenha do responsável pelo serviço o certificado digital da Autoridade Certificadora (AC) que assinou o certificado instalado para ativação do LDAPS. Este certificado é dispensável se o certificado instalado para ativação do LDAPS foi assinado por uma AC confiável. Caso o certificado instalado para ativação do LDAPS seja auto assinado (não usa uma AC), ele deve ser obtido. Nas instruções para habilitação do LDAPS no OpenLDAP neste manual, um certificado auto assinado é criado no arquivo `/etc/openldap/certificados/ldap.cert`.

Para configurar o MACA CS, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a configuração será realizada;
2. Mude o diretório corrente para `/usr/local/maca_cs/bin`;
3. Copie o certificado `/etc/openldap/certificados/ldap.cert` para o diretório corrente (`/usr/local/maca_cs/bin`);
4. Agora, é preciso criar um *keystore* para armazenar os certificados confiáveis para o MACA CS, no caso o certificado `ldap.cert`. Para isto, execute o comando a seguir e entre com os dados solicitados conforme o modelo:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias ldap -file ./ldap.cert -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha para o keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado e armazena o certificado importado `ldap.cert`.

5. Agora, edite o arquivo `maca_ca.properties` e efetue as seguintes alterações:

   Troque o valor da propriedade `maca_ca.ldap.server` para o nome DNS ou endereço IP onde o servidor OpenLDAP foi instalado, acrescido da porta `:636` do serviço LDAPS. Caso seja servidor local, coloque o valor `127.0.0.1:636`;

   Acrescente as seguintes propriedades:

```
##########################
# Configuracoes do LDAPS #
##########################

maca_ca.ldap.ssl=true
maca_ca.ldap.ssl.console=true
maca_ca.ldap.ssl.keystore=./ca_keystore
maca_ca.ldap.ssl.keystorepwd=<senha do keystore>
```

6. Salve o arquivo;
7. Reinicie o MACA CS com os seguintes comandos:

```
./maca_cs_stop.sh
./maca_cs_start.sh
```

   Agora, o MACA CS se conectará com o servidor OpenLDAP com o protocolo LDAPS na porta 636.

2.2.3 Configurando o MACA CS para Comunicação com os Clientes via IIOP sobre TLS/SSL

O MACA CS pode ser configurado para permitir ou obrigar que seus clientes se conectem a ele através do protocolo IIOP sobre TLS/SSL. Adicionalmente, o MACA CS pode ser configurando para exigir a autenticação de clientes através de certificados digitais, como pré-requisito para o estabelecimento de conexões IIOP sobre TLS/SSL. Deste modo, é possível configurar o MACA CS para aceitar conexões apenas de clientes confiáveis. Neste contexto, não confundir clientes, programas, com usuários do MACA CS, pessoas, que se autenticam por meio de senhas.

Para configurar o MACA CS, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Mude o diretório corrente para `/usr/local/maca_cs/bin`;

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

3. Crie nele um subdiretório denominado `certificados` com o comando abaixo e mude-se para ele:

```
mkdir ./certificados
cd ./certificados
```

4. Agora, crie um certificado auto assinado para o serviço do MACA CS com o comando abaixo entrando os valores solicitados de acordo com o modelo apresentado:

```
$JAVA_HOME/bin/keytool -genkey -alias maca_cs -keyalg RSA -validity 365 -keystore ./maca_cs_keystore↵
Enter keystore password: <entre com a senha da chave privada> ↵
Enter keystore password: <repita a senha anterior> ↵
What is your first and last name?
   [Unknown]: <entre com o DNS completo do seu servidor> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade, ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <maca_cs>
(RETURN if same as keystore password): ↵
```

Isto cria um certificado auto assinado e sua respectiva chave privada no arquivo `/usr/local/maca_cs/bin/certificados/maca_cs_keystore`. Para requisição de um certificado para emissão por uma AC confiável, veja os detalhes em http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html.

O arquivo `maca_cs_keystore` é um tipo de banco de dados que armazena a chave privada e o certificado, que é público. É necessário exportar este certificado para fornecê-los a todos os clientes que desejam se conectar com o serviço MACA CS via IIOP sobre TLS/SSL. Para exportá-lo, execute o seguinte comando:

```
$JAVA_HOME/bin/keytool -export -alias maca_cs -file maca_cs.cer
-keystore ./maca_cs_keystore
```

Após a execução deste comando, o arquivo `maca_cs.cer` armazenará o certificado a ser distribuído para os clientes do MACA CS;

5. Mude o diretório corrente para `/usr/local/maca_cs/bin`, edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
####################################
# Configuracoes para IIOP sobre SSL #
####################################

jacorb.security.support_ssl=on

jacorb.security.ssl.server.supported_options=20
jacorb.security.ssl.server.required_options=0

jacorb.security.keystore=./certificados/maca_cs_keystore
jacorb.security.keystore_password=<senha da chave privada>

jacorb.ssl.socket_factory=org.jacorb.security.ssl.sun_jsse.SSLSocketFactory
jacorb.ssl.server_socket_factory=org.jacorb.security.ssl.sun_jsse.SSLServerSocketFactory
```

6. Salve o arquivo;

7. Reinicie o MACA CS com os seguintes comandos:

```
./maca_cs_stop.sh
./maca_cs_start.sh
```

Agora, o MACA CS, além de aceitar conexões de clientes usando o protocolo IIOP, aceitará também conexões usando IIOP sobre TLS/SSL. **Para evitar a possibilidade de senhas de usuários trafegarem sem proteção na rede, deve-se obrigar que o cliente use IIOP sobre TLS/SSL**. Neste caso, basta alterar a propriedade `jacorb.security.ssl.server.required_options` para o seguinte valor:

```
jacorb.security.ssl.server.required_options=20
```

Assim, o MACA CS só permitirá conexões IIOP sobre TLS/SSL e esta é a configuração mínima recomendável!

A configuração do MACA CS para exigir a autenticação de clientes também é simples. Basta alterar a propriedade `jacorb.security.ssl.server.required_options` para o seguinte valor:

```
jacorb.security.ssl.server.required_options=40
```

Com isto, o MACA CS vai exigir que o cliente envie o seu certificado digital e que este seja um certificado confiável. Caso o certificado do cliente seja auto assinado ou emitido por uma AC não conhecida, este certificado auto assinado ou o certificado da AC deverão ser incluídos no *keystore* de certificados confiáveis do MACA CS. O comando abaixo mostra como importar tais certificados:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias <nome cliente ou
da AC> -file ./<nome do arquivo com certificado> -keystore
./ca_keystore
```


www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Note que o arquivo do *keystore* é o mesmo (`./ca_keystore`) usado para a configuração do LDAPS no MACA CS, embora não precise ser. Considerando que os eventuais certificados considerados confiáveis pelo MACA CS foram importados, deve-se acrescentar as seguintes propriedades no arquivo `orb.properties`:

```
javax.net.ssl.trustStore=./ca_keystore
javax.net.ssl.trustStorePassword=<senha do keystore>
```

Reiniciando o MACA CS, ele somente aceitará conexões IIOP sobre TLS/SSL de clientes autenticados, isto é, clientes que possuem certificados digitais confiáveis.

Mais detalhes nas configurações do JacOrb versão 1.4.1, ORB utilizado pelo MACA CS, podem ser obtidas em http://www.jacorb.org/documentation.html.

2.2.4 Configuração do MACA CS com Balanceamento de Carga e Tolerância a Falhas

A utilização do MACA CS com balanceamento de carga e tolerância a falhas requer a utilização do software de fonte aberto denominado Balance versão 2.33-1. O balance é um *proxy* TCP com mecanismos de balanceamento de carga e tolerância a falhas que pode ser obtido no *site* http://balance.sourceforge.net.

O balance deve ser instalado num servidor que atuará como proxy dos serviços HTTP dos servidores onde estarão instalados o instâncias do MACA CS. Ou seja, haverá um conjunto de n (n > 1) servidores, cada um deles com uma instância do servidor OpenLDAP e outra instância do servidor MACA CS instalados e em operação. Um dos servidores OpenLDAP atuará como mestre para atualizações e os demais serão replicas atualizadas automaticamente pelo mestre (veja o *link* http://www.openldap.org/doc/admin21/replication.html para saber como configurar o OpenLDAP com réplicas). Cada uma das n instâncias do MACA CS ira publicar os IORs com endereços para acesso pelos clientes aos seus serviços através do protocolo HTTP. Pegue-se agora o Balance, uma solução para balanceamento de carga e tolerância a falhas para serviços HTTP e interponha entre os clientes e cada uma das instâncias dos n servidores HTTP onde os arquivos com os IORs do MACA CS estão publicados. Quando um cliente faz a requisição http://proxy.organizacao.com.br/maca_cs.iors, o Balance vai escolher, dentre os n servidores, aquele que está em condições para atender a solicitação do usuário. A partir deste momento, o cliente se conecta ao servidor MACA CS correspondente ao IOR publicado no arquivo `maca_cs.iors`. Pode-se concluir que esta solução assegura para os clientes do MACA CS, transparentemente, disponibilidade e escalabilidade. Disponibilidade porque, quando um servidor está fora de serviço, outro é selecionado para atender a requisição do usuário transparentemente. Escalável porque, se um servidor MACA CS é capaz de atender m usuários

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

simultâneos, e se o Balance distribui as requisições igualmente entre os servidores, então este *cluster'* de servidores é capaz de atender em média n*m usuários simultâneos.

Para configurar o MACA CS com balanceamento de carga e tolerância a falhas faça o seguinte:

1. Obtenha os privilégios de administrador (*root*) no servidor que atuará como *proxy*;
2. Baixe o pacote de instalação `balance-2.33-1.i386.rpm` do Balance no *site* http://sourceforge.net/project/showfiles.php?group_id=6355 e proceda à instalação executando o seguinte comando;

```
rpm -i balance-2.33-1.i386.rpm
```

Após a instalação, verifique se a instalação foi bem sucedida digitando o comando:

```
balance –help
```

Certifique-se que a porta 80 do proxy está livre, se desejar que os clientes usem a porta padrão do serviço HTTP.

3. Considerando que há 3 instâncias do MACA CS (`maca_cs1.organizacao.com.br`, `maca_cs2.organizacao.com.br` e `maca_cs3.organizacao.com.br`) em operação, instaladas de modo convencional, execute o comando balance com os seguintes parâmetros:

```
balance http maca_cs1.organizacao.com.br:http maca_cs2.organizacao.com.br:http maca_cs3.organizacao.com.br:http
```

Com esta configuração o Balance distribui a carga das requisições HTTP entre os três servidores. Caso um deles fique indisponível, a carga é distribuída para os servidores restantes. O manual do comando Balance traz outras opções de configuração, que podem ser alteradas interativamente, sem precisar interromper o serviço.

Atenção, com esta solução, os clientes do MACA CS devem acessar os endereços dos serviços do MACA CS no arquivo `maca_cs.iors` através do *proxy* com a URL http://proxy.organizacao.com.br/maca_cs.iors. Veja os detalhes da programação dos clientes no manual "MACA Cliente - Guia do Programador". **Para os clientes MACA CS Monitor, MACA AD Web e Maca AD, as propriedades `maca_ca.maca_cs.url` e**

`maca_bean.maca_cs.url` devem se referir ao servidor do proxy na URL.

Limitações da solução: quando um servidor fica fora de serviço, as sessões abertas nele também ficam indisponíveis e o usuário é obrigado a efetuar um novo login. O procedimento de login deve recuperar um novo IOR via *proxy*, de modo a obter o endereço de um servidor MACA CS disponível.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

3 MACA CS Monitor

O MACA CS Monitor requer para o seu funcionamento a instalação prévia ou a disponibilidade do servidor JSP Tomcat. A seção 3.1 traz as instruções para efetuar a instalação e configuração do Tomcat necessárias para o correto funcionamento do MACA CS Monitor. A seção 3.2 mostra como instalar e configurar o MACA CS Monitor.

3.1 Requisitos de Software

As instruções para instalação contidas neste manual visam facilitar a instalação dos componentes de software necessários ao MACA CS Monitor, mas em nenhuma circunstância substituem a documentação de instalação original destes componentes. Deste modo, não há garantias de que tais instruções estejam completas e que funcionarão para quaisquer versões de sistemas operacionais. A documentação original que deve ser consultada e estudada sempre. A instalação apresentada neste manual usou as seguintes versões de produtos:

- Linux 2.4.27-2-386 #1 Thu Jan 20 10:55:08 JST 2005 i686 GNU/Linux ou Windows 2000 (5.0.2195 *Service Pack* 3);
- Java™ 2 Software Development Kit, Standard Edition (build 1.4.1);
- Tomcat 4.0.6;

Os limites mínimos de versões quando não indicados neste texto devem ser consultados diretamente nos manuais originais dos componentes requeridos.

3.1.1 Serviço JSP Tomcat – Instalação e Configuração

O Tomcat pode ser instalado tanto no sistema operacional Linux, quanto no sistema operacional Windows 2000. Ele deve ser instalado no mesmo servidor onde o MACA CS Monitor funcionará. Recomenda-se que este servidor não seja o mesmo em que o MACA CS está funcionando, embora possam ser instalados numa mesma máquina. O Tomcat requer a instalação prévia do Java SDK (*Software Development Kit*) versão 1.4.x (Não funciona no Java JDK 1.5.x). Não confundir com o JRE, que instala apenas o ambiente execução para aplicações Java.

Os arquivos e as instruções de instalação do Tomcat podem ser obtidos no *site* do projeto http://jakarta.apache.org/tomcat/index.html. A seguir apresentamos as instruções gerais para instalação e configuração do Tomcat de acordo com as necessidades do MACA CS Monitor:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Baixe o pacote de instalação da versão 1.4.x do Java SDK no *site* http://java.sun.com/j2se/1.4.2/download.html para a versão do seu sistema operacional;
3. Instale o Java SDK de acordo com as recomendações específicas do para o sistema operacional, encontradas em http://java.sun.com/j2se/1.4.2/install.html;

   Após a instalação, verifique se há uma variável de ambiente denominada `JAVA_HOME` com valor igual ao caminho absoluto do diretório onde o Java SDK foi instalado. Caso não exista é necessário criar ou atualizar a variável `JAVA_HOME` com este valor:

   ```
   JAVA_HOME=<path do Java SDK>; export JAVA_HOME
   ```

4. Baixe o pacote de instalação da versão 4.0.6 do Tomcat (binários) no *site* http://archive.apache.org/dist/jakarta/tomcat-4/archive/v4.0.6/bin/ para a a opção do seu sistema operacional. Baixe o arquivo `jakarta-tomcat-4.0.6.exe` se a instalação for no Windows NT/2000 ou o `jakarta-tomcat-4.0.6.tar.gz` se desejar instalar no Linux;
5. Após baixar o arquivo, ele dever ser descompactado com o seguinte comando no Linux (Para instalar no Windows NT/2000, basta executar o arquivo `jakarta-tomcat-4.0.6.exe` e efetuar a instalação padrão):

   ```
   gunzip -c jakarta-tomcat-4.0.6.tar.gz | tar xf -
   ```

6. Após a instalação, verifique se há uma variável de ambiente denominada `CATALINA_HOME` com valor igual ao caminho absoluto do diretório onde o Tomcat 4.0.6 foi instalado, por exemplo `/usr/local/jakarta-tomcat-4.0.6`. Caso não exista é necessário criar ou atualizar a variável `CATALINA_HOME` com este valor (Linux):

   ```
   CATALINA_HOME=<path do Tomcat>; export CATALINA_HOME
   ```

   Atenção, não colocar a barra "/" no final do nome do diretório.

7. Para iniciar o Tomcat, mude-se para o diretório onde ele foi instalado e execute o seguinte comando (Linux):

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
cd $CATALINA_HOME/bin
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
startup
```

Após iniciá-lo, verifique se ele foi iniciado corretamente acessando a seguinte URL num *web browser* de sua preferência, que exibirá a página inicial do Tomcat:

http://127.0.0.1:8080/index.html

Certifique-se, porém, que a porta local 8080 não esteja sendo utilizada e não se esqueça de atualizar o nome do servidor na URL;

8. Para encerrar o Tomcat, mude-se para o diretório onde ele foi instalado e execute o seguinte comando (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
```

9. Agora, com o Tomcat disponível, é preciso configurá-lo para permitir o correto funcionamento do MACA CS Monitor. Duas alterações são necessárias:

   - Configurar o Tomcat para aceitar o protocolo HTTPS, ou seja, HTTP sobre SSL para assegurar confidencialidade e integridade na comunicação entre cliente e o servidor HTTP;
   - Modificar as portas do protocolo HTTP e HTTPS para as portas padrão 80 e 443;

10. Para configurar o SSL, faça o seguinte. Mude-se para o diretório de configuração do Tomcat executando o comando:

```
cd $CATALINA_HOME/conf
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

11. Execute o comando a seguir para criar um certificado digital auto assinado do serviço HTTPS do Tomcat (para mais detalhes, veja o *link* http://jakarta.apache.org/tomcat/tomcat-4.0-doc/ssl-howto.html), entrando com os valores recomendados:

```
$JAVA_HOME/bin/keytool –genkey -alias tomcat -keyalg RSA -validity
365 ↵
Enter keystore password: changeit ↵
Enter keystore password: changeit ↵
What is your first and last name?
   [Unknown]: <entre com o DNS completo do seu servidor> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade,
ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <tomcat>
(RETURN if same as keystore password): ↵
```

12. O certificado é criado por default no diretório *home* do usuário corrente. Para verificar sua criação no Linux, execute o comando a seguir:

```
ls -l –a $HOME
```

Um arquivo chamando `.keystore` deverá aparecer na lista. No Windows 2000, este arquivo é criado no diretório "`C:\Documents and Settings\<usuario>`" Para verificar o certificado execute o comando (Linux):

```
$JAVA_HOME/bin/keytool -list -keystore $HOME/.keystore -storepass
changeit
```

A seguinte saída deverá ser apresentada:

```
Keystore type: jks
Keystore provider: SUN

Your keystore contains 1 entry

tomcat, May 26, 2003, keyEntry,
Certificate fingerprint (MD5): 98:B7:93:2C:E1:F8:DF:36:C3:53:DF:01:D3:11:A3:7D
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

No Windows 2000, digite:

```
"%JAVA_HOME%/bin/keytool" -list -keystore "C:\Documents and
Settings\<usuario>\.keystore" -storepass changeit
```

13. Com o certificado criado, basta editar o arquivo `$CATALINA_HOME/conf/server.xml` e efetuar as seguintes configurações:

Descomentar a entrada do conector "*Define an SSL HTTP/1.1 Connector on port 8443*" para que fique da seguinte forma (alterações em negrito e sublinhadas):

```
<!-- Define an SSL HTTP/1.1 Connector on port 8443 -->
<!-- -->
<Connector className="org.apache.catalina.connector.http.HttpConnector"
           port="8443" minProcessors="5" maxProcessors="75"
           enableLookups="true"
           acceptCount="10" debug="0" scheme="https" secure="true">
  <Factory className="org.apache.catalina.net.SSLServerSocketFactory"
           clientAuth="false" protocol="TLS"/>
</Connector>
<!--  -->
```

Grave o arquivo e saia para o prompt de comando;

14. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./ shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

15. Após reiniciá-lo, verifique se o HTTPS foi iniciado corretamente acessando a seguinte URL num *web browser* de sua preferência, que exibirá a página inicial do Tomcat:

https://127.0.0.1:8443/index.html

Certifique-se, porém, de que a porta local do 8443 já não está sendo utilizada e não se esqueça de atualizar o nome do servidor na URL. Note que o protocolo utilizado é o HTTPS. O seu *browser* pode apresentar uma janela de alerta afirmando que o certificado do servidor não é confiável.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Aceite o certificado, pois o alerta é dado porque o certificado é auto assinado, não sendo emitido por uma entidade certificadora conhecida do *browser*;

16. Para finalizar a configuração do Tomcat, basta alterar as portas padrão para os serviços HTTP e HTTPS, conforme instruções abaixo:

Edite o arquivo `$CATALINA_HOME/conf/server.xml` e efetue as seguintes configurações:

Modifique a entrada do conector "*Define a non-SSL HTTP/1.1 Connector on port 8080*" para que fique da seguinte forma (alterações em negrito e sublinhadas):

```
<!-- Define a non-SSL HTTP/1.1 Connector on port 8080 -->
<Connector className="org.apache.catalina.connector.http.HttpConnector"
           port="80" minProcessors="5" maxProcessors="75"
           enableLookups="true" redirectPort="443"
           acceptCount="10" debug="0" connectionTimeout="60000"/>
<!-- Note : To disable connection timeouts, set connectionTimeout value to -1 -->
```

Modifique a entrada do conector "*Define an SSL HTTP/1.1 Connector on port 8443*" para que fique da seguinte forma (alterações em negrito e sublinhadas):

```
<!-- Define an SSL HTTP/1.1 Connector on port 8443 -->
<!-- -->
<Connector className="org.apache.catalina.connector.http.HttpConnector"
           port="443" minProcessors="5" maxProcessors="75"
           enableLookups="true"
           acceptCount="10" debug="0" scheme="https" secure="true">
  <Factory className="org.apache.catalina.net.SSLServerSocketFactory"
           clientAuth="false" protocol="TLS"/>
</Connector>
<!--  -->
```

Grave o arquivo e saia para o prompt de comando;

17. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

18. Após reiniciá-lo, verifique se o HTTPS foi iniciado corretamente acessando a seguinte URL num *web browser* de sua preferência, que exibirá a página inicial do Tomcat:

    https://127.0.0.1/index.html

    Certifique-se, porém, de que as portas locais do HTTP (80) e do HTTPS (443) já não estão sendo utilizadas e não se esqueça de atualizar o nome do servidor na URL. Se tudo correu bem, então o Tomcat está pronto para executar o MACA CS Monitor.

3.2 MACA CS Monitor – Instalação e Configuração

A instalação e configuração básica do MACA CS Monitor é bastante simples, considerando-se que os softwares e serviços requisitados por ele estão disponíveis. O MACA CS Monitor deve ser instalado no mesmo servidor em que o JSP Tomcat foi instalado.

Para instalar e configurar o MACA CS Monitor, proceda da seguinte forma:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Copie o arquivo `maca_deployment_VERSAO/maca_cs_monitor/maca_cs_monitor.war` do diretório de distribuição do MACA para o diretório `$CATALINA_HOME/webapps/`;
3. Copie o arquivo `maca_deployment_VERSAO/maca_cs/lib/jacorb.jar` do diretório de distribuição do MACA para o diretório `$CATALINA_HOME/lib/`;
4. Copie o arquivo `maca_deployment_VERSAO/maca_cs/bin/jacorb.properties` do diretório de distribuição do MACA para o diretório `$JAVA_HOME/jre/lib/`;
5. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

6. Mude para o diretório `$CATALINA_HOME/webapps/`. Note que foi criando um subdiretório com o mesmo nome do arquivo `maca_cs_monitor.war`, sem a extensão. Isto porque o este arquivo traz este diretório compactado. Sempre que o Tomcat é iniciado, ele verifica se já existe subdiretórios com os nomes dos arquivos `.war` sem a extensão. Caso não existam, ele descompacta os arquivos e cria os subdiretórios. Estes subdiretórios contêm uma aplicação JSP com todos os recursos necessários à sua execução;

7. Agora é preciso efetuar uma única configuração na aplicação MACA CS Monitor para que ela possa ser utilizada. Para isto, edite o arquivo `$CATALINA_HOME/webapps/maca_cs_monitor/config/maca_bean.properties` e faça o seguinte:

   Altere o valor da propriedade `maca_bean.maca_cs.url` para a URL com o nome do servidor onde o MACA CS foi instalado, e capaz de recuperar os arquivos `maca_cs.iors`, `monitor.ior`, `pa.ior` e `ado.ior` que foram criados no diretório raiz do serviço HTTP. Por exemplo, o valor deve ser alterado para:

   ```
   maca_bean.maca_cs.url=http\://macacs.organizacao.com.br
   ```

   Caso haja mais de um servidor MACA CS disponível, a lista de URLs de cada um deles deve ser colocada separada por ponto-e-vírgula (;), conforme ilustrado abaixo:

   ```
   maca_bean.maca_cs.url=http\://macacs1.organizacao.com.br; http\://macacs2.organizacao.com.br
   ```

   Atenção, não deixe espaços em branco entre os ponto-e-vírgulas e use o caracter de *escape* \ antes do sinal de dois pontos (\:);

8. Grave o arquivo em edição e verifique se a URL abaixo retorna o conteúdo do arquivo maca_cs.iors:

   http://<DNS do servidor do MACA CS>/maca_cs.iors

   O *browser* deve retornar algo parecido com a saída a seguir,

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
paIOR:000000000000003649444C3A6F6D672E6F72672F53656375726974794C6576656C322F5072696E636970616C41757468656E74696361746F72
3A312E3000000000000000200000000000000074000102000000000F3230302E3134342E37332E3130360000B7B000000000001A5374616E6461726449
6D706C4E616D652F70615F504F412F706100000000000200000000000000800000000004A414300000000010000001C0000000000010001000000105
0100010001010900000001050100010000000010000002C0000000000000001000000010000001C0000000000010001000000010501000100010109000
00000105010001;adoIOR:000000000000003849444C3A6F6D672E6F72672F44665265736F7572636541636365737344656369736973696F6E2F41636365
7373446563697369 6F6E3A312E3000000000020000000000000074000102000000000F3230302E3134342E37332E3130360000B7B000000000001C53
74616E64617264496D706C4E616D652F61646F5F504F412F61646F000000020000000000000080000000004A414300000000010000001C0000000000
0100010000000105010001000101090000000105010001000000010000002C0000000000000001000000010000001C0000000000010001000000010501
0100010001010900000000105010001;monitorIOR:000000000000001849444C3A6D6163615F63732F4D6F6E69746F723A312E3000000000020000000
000000007C0001020000000000F3230302E3134342E37332E3130360000B7B000000000024537461 6E64617264496D706C4E616D652F6D6F6E69746F
725F504F412F6D6F6E69746F72000000020000000000000080000000004A414300000000010000001C0000000000010001000000010501000100010101
0900000001050100010000000010000002C0000000000000001000000010000001C0000000000010001000000010501000100010109000000001050100
01;
```

mas numa única linha. Com isto, o MACA CS Monitor está pronto para funcionar, bastando reiniciá-lo para que esta configuração tenha efeito;

9. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

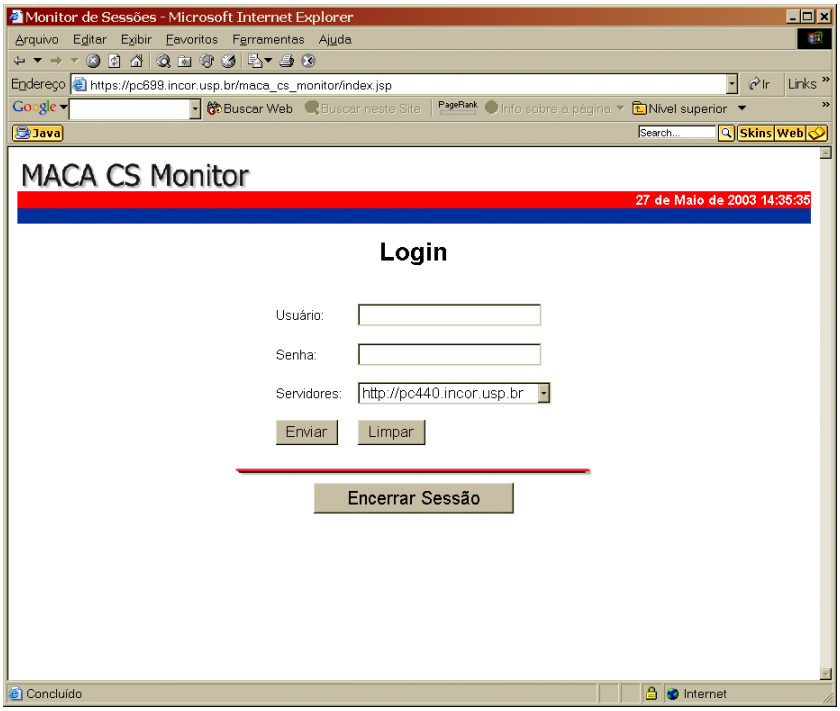
No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

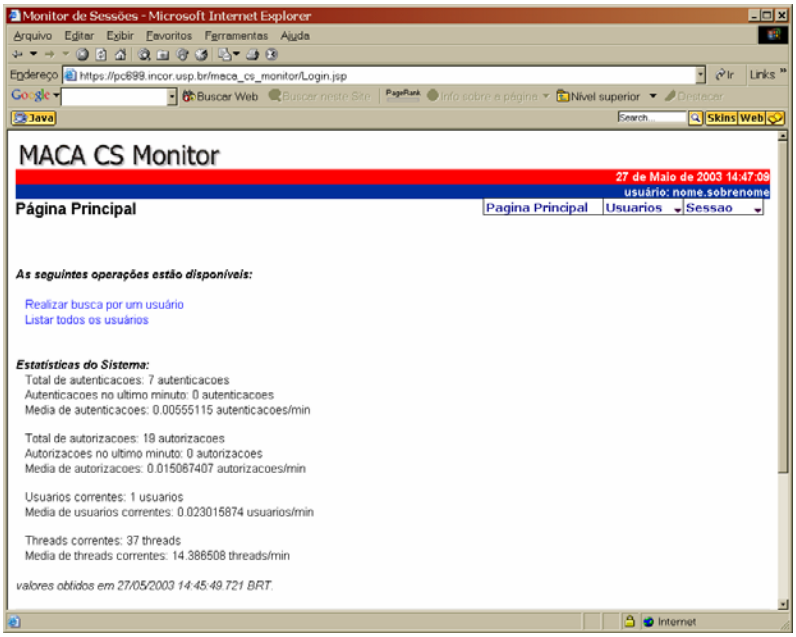
10. Agora, para acessar o MACA CS Monitor, entre com a URL a seguir no seu *browser*:

https://<DNS do servidor do MACA CS Monitor>/maca_cs_monitor/

A seguinte página deverá aparecer:



11. Entre com o login do usuário padrão carregado do arquivo `maca_deployment_VERSAO/ldap.confs/maca_inicial.ldif` durante a instalação do OpenLDAP. O usuário padrão tem o login **`nome.sobrenome`** e a senha **`maca`**. Após o login, a página abaixo é exibida com informações estatísticas do servidor MACA CS selecionado para conexão.



A partir desta página, administradores do MACA CS poderão listar os usuários ativos, isto é, usuários com sessões abertas no momento. Para cada usuário, o administrador poderá limpar seu cache ou forçar o encerramento de uma sessão. Pelo menos um usuário aparecerá, no caso, o próprio administrador do MACA CS. Administradores do MACA CS são todos os usuários com os papéis de "Administrador de Contas" ou "Administrador de Autorizacao", ou seus descendentes.

O usuários `nome.sobrenome` possui ambos os papéis atribuídos. Com todas estas páginas em funcionamento, fica encerrada a instalação do MACA CS Monitor.

Sempre que sair do MACA CS Monitor, não esquecer de encerrar a sessão!

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

3.2.1 Configurando o MACA CS Monitor para Acessar o serviço MACA CS via IIOP sobre TLS/SSL

O MACA CS Monitor pode ser configurado para permitir ou obrigar que a conexão com o servidor MACA CS seja através do protocolo IIOP sobre TLS/SSL. Adicionalmente, o MACA CS pode ser configurando para permitir a sua autenticação como cliente, através de um certificado digital, junto ao servidor MACA CS.

Para configurar o MACA CS Monitor, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Mude o diretório corrente para `$CATALINA_HOME/webapps/maca_cs_monitor/config/`;
3. Caso o certificado do servidor MACA CS seja auto assinado ou emitido por uma AC não confiável, é necessário criar um *keystore* para armazenar tais certificados. Caso as instruções da seção 2.2.3 tenham sido seguidas, obtenha com o responsável o certificado auto assinado criado disponível em `/usr/local/maca_cs/bin/certificados/maca_cs.cer` e o copie para o diretório corrente (`$CATALINA_HOME/webapps/maca_cs_monitor/config/`);
4. Agora execute o comando a seguir e entre com os dados solicitados conforme o modelo para importar o certificado:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias maca_cs -file ./maca_cs.cer -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha para o keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado e armazena o certificado importado `maca_cs.cer`.

5. Edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
#####################################
# Configuracoes para IIOP sobre SSL #
#####################################

jacorb.security.support_ssl=on

jacorb.security.ssl.client.supported_options=20
jacorb.security.ssl.client.required_options=0

jacorb.ssl.socket_factory=org.jacorb.security.ssl.sun_jsse.SSLSocketFactory
jacorb.ssl.server_socket_factory=org.jacorb.security.ssl.sun_jsse.SSLServerSocketFactory

javax.net.ssl.trustStore=<diretorio do tomcat>/webapps/maca_cs_monitor/config/ca_keystore
javax.net.ssl.trustStorePassword=<senha do keystore>
```

Estas propriedades configuram o cliente para aceitar conexões IIOP sobre TLS/SSL, mas não o obrigam a realizar. Isto é, se o servidor MACA CS estiver configurado para obrigar conexões IIOP sobre TLS/SSL, então o cliente as realizará. Por outro lado, caso o MACA CS não obrigue tais conexões, o cliente fará uma conexão IIOP convencional. Para obrigar o cliente a realizar somente conexões IIOP sobre TLS/SSL, então altere o valor da propriedade `jacorb.security.ssl.client.required_options` para

```
jacorb.security.ssl.client.required_options=20
```

Esta configuração, entretanto, não habilita a autenticação do cliente junto ao MACA CS através de certificados digitais (ver item 9 a seguir para saber como configurar);

6. Salve o arquivo;
7. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

8. Agora, acesse o MACA CS Monitor entrando com a URL a seguir no seu *browser* para testar a nova configuração:

https://<DNS do servidor do MACA CS Monitor>/maca_cs_monitor/

9. Para habilitar a autenticação do cliente, é preciso criar para ele um certificado digital, que poderá ser auto assinado, ou emitido por uma AC confiável para o MACA CS. Para criar um certificado auto assinado, execute o comando abaixo entrando os valores solicitados de acordo com o modelo apresentado:

```
$JAVA_HOME/bin/keytool -genkey -alias cliente -keyalg RSA -validity 365 -keystore ./cliente_keystore↵
Enter keystore password: <entre com a senha da chave privada> ↵
Enter keystore password: <repita a senha anterior> ↵
What is your first and last name?
   [Unknown]: <entre com o DNS completo da estação do cliente> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade, ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <cliente>
(RETURN if same as keystore password): ↵
```

Isto cria um certificado auto assinado e sua respectiva chave privada no arquivo `$CATALINA_HOME/webapps/maca_cs_monitor/config/cliente_keystore`. Para requisição de um certificado para emissão por uma AC confiável, veja os detalhes em http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html.

O arquivo `cliente_keystore` é um tipo de banco de dados que armazena a chave privada e o certificado, que é público. É necessário exportar este certificado para fornecê-los a todos os servidores MACA CS que desejam aceitar conexões deste cliente através do IIOP sobre TLS/SSL. Para exportá-lo, execute o seguinte comando:

```
$JAVA_HOME/bin/keytool -export -alias cliente -file cliente.cer
-keystore ./cliente_keystore
```

Após a execução deste comando, o arquivo `cliente.cer` armazenará o certificado a ser fornecido para os servidores do MACA CS com os quais o cliente deseja autenticar;

10. Com o certificado do cliente criado, agora edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
jacorb.security.keystore=<diretorio do tomcat>/webapps/maca_cs_monitor/config/cliente_keystore
jacorb.security.keystore_password=<senha da chave privada>
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Depois altere o valor da propriedade `jacorb.security.ssl.client.supported_options` para

```
jacorb.security.ssl.client.supported_options=40
```

Grave o arquivo.

Agora o cliente está apto a se autenticar junto ao serviço MACA CS. **ATENÇÃO, não esquecer que o MACA CS deve ter sido configurado para exigir a autenticação de clientes e que o certificado deste cliente deve ser um certificado confiável para o MACA CS.**

11. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

12. Agora, acesse o MACA CS Monitor entrando com a URL a seguir no seu *browser* para testar a nova configuração:

https://<DNS do servidor do MACA CS Monitor>/maca_cs_monitor/

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

4 MACA AD Web

O MACA AD Web requer para o seu funcionamento a instalação prévia ou a disponibilidade do servidor JSP Tomcat. A instalação e configuração do Tomcat para o MACA AD Web deve seguir as mesmas instruções presentes na seção 3.1 vista anteriormente. Em princípio, MACA AD Web e o MACA CS Monitor podem ser instalados num mesmo servidor Tomcat. Portanto, a seção 4.1 mostra como instalar e configurar o MACA AD Web, considerando que o Tomcat já está instalado.

4.1 MACA AD Web – Instalação e Configuração

A instalação e configuração básica do MACA AD Web é bastante simples, considerando-se que os softwares e serviços requisitados por ele estão disponíveis. O MACA AD Web deve ser instalado no mesmo servidor em que o JSP Tomcat foi instalado.

Para instalar e configurar o MACA AD Web, proceda da seguinte forma:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Copie o arquivo `maca_deployment_VERSAO/maca_ad_web/maca_ad_web.war` do diretório de distribuição do MACA para o diretório `$CATALINA_HOME/webapps/`;
3. Copie o arquivo `maca_deployment_VERSAO/maca_cs/lib/jacorb.jar` do diretório de distribuição do MACA para o diretório `$CATALINA_HOME/lib/`, caso ele ainda não esteja lá;
4. Copie o arquivo `maca_deployment_VERSAO/maca_cs/bin/jacorb.properties` do diretório de distribuição do MACA para o diretório `$JAVA_HOME/jre/lib/`, caso ele ainda não esteja lá;
5. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

6. Mude-se para o diretório `$CATALINA_HOME/webapps/`. Note que foi criando um subdiretório com o mesmo nome do arquivo `maca_ad_web.war`, sem a extensão. Isto porque o este arquivo traz este diretório compactado. Sempre que o Tomcat é iniciado, ele verifica se já existe subdiretórios com os nomes dos arquivos `.war` sem a extensão. Caso não existam, ele descompacta os arquivos e cria os subdiretórios. Estes subdiretórios contêm uma aplicação JSP com todos os recursos necessários à sua execução;

7. Agora é preciso efetuar as configurações da aplicação MACA AD Web para que ela possa ser utilizada. Para isto, edite o arquivo `$CATALINA_HOME/webapps/maca_ad_web/config/maca_bean.properties` e faça o seguinte:

   Altere o valor da propriedade `maca_bean.maca_cs.url` para a URL com o nome do servidor onde o MACA CS foi instalado, e capaz de recuperar os arquivos `maca_cs.iors`, `monitor.ior`, `pa.ior` e `ado.ior` que foram criados no diretório raiz do serviço HTTP. Por exemplo, o valor deve ser alterado para:

   ```
   maca_bean.maca_cs.url=http\://macacs.organizacao.com.br
   ```

   Este parâmetro de configuração no MACA AD Web só aceita uma URL, diferentemente do MACA CS Monitor, onde este parâmetro aceita uma lista de URLs. **Atenção, use o caracter de *escape* \ antes do sinal de dois pontos (`\:`)**;

   Grave o arquivo;

8. Agora, edite o arquivo `$CATALINA_HOME/webapps/maca_ad_web/config/maca_ca.properties` e faça o seguinte:

   Altere o valor da propriedade `maca_ca.maca_cs.url` para a URL com o nome do servidor onde o MACA CS foi instalado, e capaz de recuperar os arquivos `maca_cs.iors`, `monitor.ior`, `pa.ior` e `ado.ior` que foram

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

criados no diretório raiz do serviço HTTP. Por exemplo, o valor deve ser alterado para:

```
maca_ca.maca_cs.url=http\://macacs.organizacao.com.br
```

Este parâmetro de configuração deve ter o mesmo valor que o parâmetro `maca_bean.maca_cs.url`, do arquivo `maca_bean.properties` definido anteriormente. **Atenção, use o caracter de *escape* \ antes do sinal de dois pontos (\:)**;

Altere o valor da propriedade `maca_ca.ldap.server` para o endereço (DNS ou IP) do servidor onde o servidor OpenLDAP foi instalado. Por exemplo, o valor deve ser alterado para:

```
maca_ca.ldap.server=<nome do servidor OpenLDAP>.organizacao.com.br
```

Troque o valor da propriedade `maca_ca.ldap.rootDN` para o nome do sufixo escolhido (valor da propriedade `suffix`) durante a instalação do OpenLDAP. Por exemplo, o valor deve ser alterado para:

```
maca_ca.ldap.rootDN=dc\=<nome organizacao>,dc\=com,dc\=br
```

Atenção, não deixe espaços em branco entre as vírgulas e use o caracter de *escape* \ antes do sinal de igual (\=);

Grave o arquivo;

9. Verifique se a URL abaixo retorna o conteudo do arquivo maca_cs.iors:

   http://<DNS do servidor do MACA CS>/maca_cs.iors

   O *browser* deve retornar algo parecido com a saída abaixo,

```
paIOR:000000000000003649444C3A6F6D672E6F72672F53656375726974794C6576656C322F5072696E636970616C41757468656E74696361746F72
3A312E3000000000000000200000000000000074000102000000000F3230302E3134342E37332E3130360000B7B000000000001A5374616E6461726449
6D706C4E616D652F70615F504F412F706100000000000200000000000000080000000004A414300000000010000001C000000000001000100000001
0100010001010900000001050100010000000010000002C000000000000000010000000100000001C000000000001000100000001050100010001010900
00000105010001;adoIOR:000000000000003849444C3A6F6D672E6F72672F44665265736F7572636541636365737344656369736F6E2F41636365
7373446563697369 6F6E3A312E3000000000020000000000000074000102000000000F3230302E3134342E37332E3130360000B7B000000000001C53
74616E64617264496D706C4E616D652F61646F5F504F412F61646F000000020000000000000080000000004A414300000000010000001C0000000000
0100010000000105010001000101090000000105010001000000010000002C000000000000000010000000100000001C000000000001000100000001
0100010001010900000001050100010000000010000002C000000000000000010000000100000001C000000000001000100000001050100010001010900
000000007C0001020000000000F3230302E3134342E37332E3130360000B7B000000000002453 74616E64617264496D706C4E616D652F6D6F6E69746F
725F504F412F6D6F6E69746F72000000020000000000000008000000004A414300000000010000001C000000000001000100000001050100010001010
0900000001050100010000000010000002C000000000000000010000000100000001C000000000001000100000001050100010001010900000001050100
01;
```

mas numa única linha. Com isto, o MACA AD Web está pronto para funcionar, bastando reiniciá-lo para que esta configuração tenha efeito;

10. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

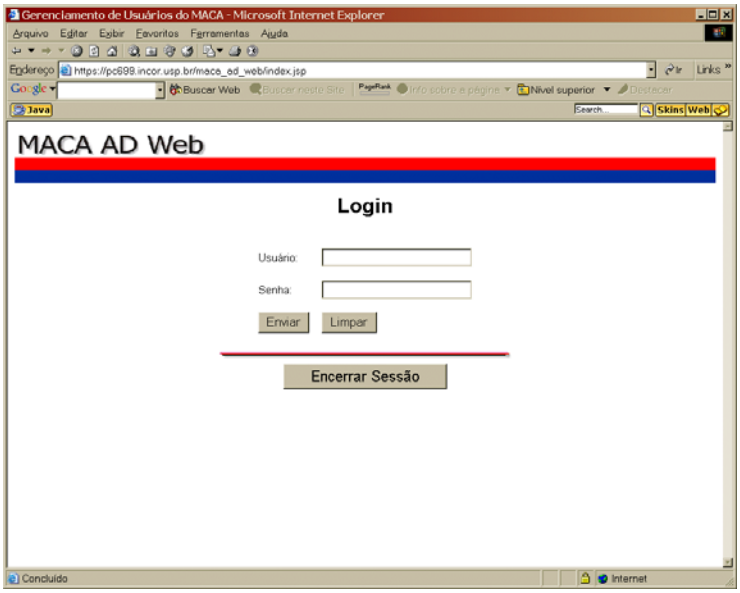
No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

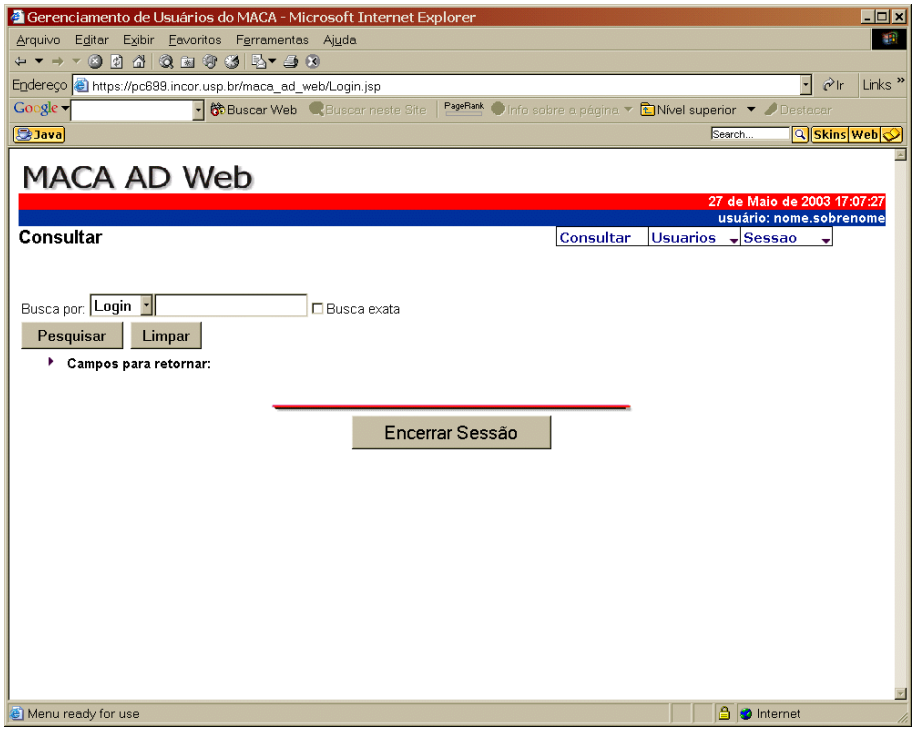
11. Agora, para acessar o MACA AD Web, entre com a URL a seguir no seu *browser*:

https://<dns do servidor do maca ad web>/maca_ad_web/

A seguinte página deverá aparecer:



12. Entre com o login do usuário padrão carregado do arquivo `maca_deployment_VERSAO/ldap.confs/maca_inicial.ldif` durante a instalação do OpenLDAP. O usuário padrão tem o login **`nome.sobrenome`** e a senha **`maca`**. Após o login, a página de consulta a contas de usuários do MACA é exibida, conforme ilustrado abaixo:



A partir desta página, administradores de contas do MACA poderão criar, alterar, remover e consultar contas de usuários. Adicionalmente, o MACA AD Web permitirá a gerência de senhas e a atribuição/exclusão de papéis de usuários;

Agora, use o MACA AD Web para criar uma conta para você ou para a pessoa que será o administrador do MACA. Para que esta conta tenha os privilégios completos de administrador, conceda a ela os papéis “Administrador de Contas” e “Administrador de Autorizacao”. A senha criada para novos usuários é igual ao login do usuário. Após a criação desta conta, ela poderá ser usada para alterar/remover a conta padrão `nome.sobrenome`.

Não se esqueça de alterar as senhas das contas dos administradores

Sempre que sair do MACA AD Web, não esquecer de encerrar a sessão!

Mais detalhes na utilização do MACA AD Web são encontrados no manual "MACA Administrativo - Manual do Usuário", integrante do diretório de distribuição do MACA.

4.1.1 Configurações Úteis

Algumas configurações *default* do MACA AD Web são bastante simples para funcionamento em ambientes de produção. Nesta seção, recomendamos algumas configurações úteis.

Política de Senhas

No arquivo `maca_ca.properties`, recomenda-se definir as seguintes propriedades:

```
maca_ca.passwordPolicy.passwordCheckSyntax=true
maca_ca.passwordPolicy.passwordMustChange=true
maca_ca.passwordPolicy.passwordFactory=maca_ca.BasicPasswordFactory
```

A última propriedade especifica uma classe pré-definida do MACA CS para geração de senhas que substitui a classe *default*, que gera a senha inicial igual ao login do usuário. A classe acima especificada gera senhas aleatórias com 6 caracteres e a envia por e-mail para o usuário (valor da propriedade `mail`). Quando se define esta classe para a propriedade maca_ca.passwordPolicy.passwordFactory, deve-se também definir as seguintes propriedades no arquivo `maca_ca.properties`:

```
maca_ca.mail=<e-mail do administrador do MACA AD Web que enviará o
e-mail para o usuário com a senha dele>
mail.smtp.host=<DNS ou IP do servidor smtp que enviará o e-mail para
o usuário com a senha dele>
```

Veja a Tabela 2 com a descrição destas e de outras propriedades para a política de senhas.

DEFININDO UM GERADOR DE SENHAS CUSTOMIZADO

O MACA AD Web permite a criação de uma classe customizada para geração de senhas iniciais. Para realizar isto, basta implementar a interface `PasswordFactory.java` disponível no diretório maca_deployment_VERSAO/maca_ad_web/extensions/src/maca_ca do diretório de distribuição do MACA. Neste diretório há um exemplo (classe `BasicPasswordFactoryExample.java`) de classe que com uma implementação dela. Para torná-la disponível, compile esta classe e coloque o seu código objeto (`BasicPasswordFactoryExample.class`) disponível no CLASS_PATH da máquina virtual que é utilizada pelo Tomcat onde o MACA AD Web funciona. Outra forma de torná-la disponível é gerar um arquivo `.jar` com o código objeto da classe e colocá-lo no diretório `$JAVA_HOME/jre/bin/ext/`. Depois, basta mudar o valor da propriedade

`maca_ca.passwordPolicy.passwordFactory` para o nome completo da classe implementada e reiniciar o Tomcat.

Redefinição do Esquema de Dados da Conta de Usuários

O MACA AD Web é instalado com um esquema de dados pré-definido que especifica quais atributos farão parte da conta de um usuário. Este esquema padrão obriga o armazenamento do login, nome completo, primeiro nome, sobrenome, CPF e e-mail. Ele é definido no padrão XML e está armazenado no arquivo `$CATALINA_HOME/webapps/maca_ad_web/config/esquema_usuario.xml`. O MACA permite a extensão do esquema padrão num novo arquivo XML, que passa a valer com a redefinição da propriedade `maca_ca.user.schema` no arquivo `maca_ca.properties` (ver Tabela 5). Assim, as informações constantes na conta dos usuários podem ser customizadas de acordo com a necessidade de cada organização onde o MACA será instalado.

Entendendo o Esquema Padrão

O esquema padrão de contas do MACA segue a estrutura do XML apresentado a abaixo:

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<maca_ca version="3.2">
  <schema>
    <user>
      <ldapentry dn="">
        <attribute name="objectClass" visible="false" required="true" multivalue="true">
          <value><![CDATA[top]]></value>
          <value><![CDATA[person]]></value>
          <value><![CDATA[inetorgperson]]></value>
          <value><![CDATA[incorperson]]></value>
          <value><![CDATA[passwordobject]]></value>
        </attribute>
        <tab name="Identificacao">
          <attribute name="uid" displayname="Login" required="true" maxlen="30"></attribute>
          <attribute name="givenName" displayname="Nome" required="true" maxlen="100" case="firstUpper"></attribute>
          <attribute name="cn" displayname="Prenome" required="true" maxlen="20" case="firstUpper"></attribute>
          <attribute name="sn" displayname="Sobrenome" required="true" maxlen="20" case="firstUpper"></attribute>
          <attribute name="employeeNumber" displayname="CPF" required="true" maxlen="12"></attribute>
          <attribute name="mail" displayname="E-mail" required="true" multivalue="true" maxlen="50"></attribute>
        </tab>
      </ldapentry>
    </user>
  </schema>
</maca_ca>
```

O entendimento e a extensão deste esquema padrão exige o conhecimento prévio dos esquemas de dados LDAP definidos no diretório `/etc/openldap/schema` no servidor onde o OpenLDAP foi instalado. O esquema XML anterior define uma entrada de conta de usuário (elemento **`ldapentry`**) como um conjunto de atributos (elemento **`attribute`**). O elemento **`tab`** serve para agrupar atributos afins em pastas no momento da exibição pelo MACA AD Web. Um atributo pode ter um conjunto de valores default definido pelo elemento **`value`** e é caracterizado por um conjunto de parâmetros[2]. O único parâmetro obrigatório para um atributo é o `name`, que necessariamente deve corresponder a um atributo definido nos esquemas LDAP (definidos no diretório

[2] Chamados de atributos na nomenclatura XML, mas que aqui serão chamados de parâmetros para não confundir com o elemento "attribute" que está sendo definido.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

/etc/openldap/schema). O atributo objectClass é obrigatório e especifica as classes de objetos LDAP (definidas no diretório /etc/openldap/schema) que a conta do usuário tem. Cada classe de objetos LDAP permite ou obriga o armazenamento de um conjunto de atributos LDAP (definidos no diretório /etc/openldap/schema). Por exemplo, a classe de objetos person do LDAP está definida porque é ela quem permite (e obriga) o armazenamento dos atributos cn e sn, primeiro nome e sobrenome, respectivamente. As informações sobre quais classes de objetos LDAP estão disponíveis, com quais atributos, estão nos esquemas de dados do diretório /etc/openldap/schema. A lista a seguir descreve cada um dos parâmetros que podem ser definidos para os elementos do tipo **attribute** no esquema de dados das contas de usuários do MACA:

- name: indica o nome do atributo LDAP a ser armazenado na conta do usuário. Para cada atributo definido, deve existir pelo menos uma classe de objetos especificada no atributo objectClass que permite o seu armazenamento;

- displayname: especifica o nome em português que será exibido como rótulo do atributo no cadastro da conta do usuário no MACA AD Web;

- visible: valor booleano que especifica se o atributo é exibido no cadastro do MACA AD Web ou não. O valor default é true, isto é, o preenchimento é opcional;

- required: valor booleano que especifica se o preenchimento do atributo é obrigatório ou não. O valor default é false, isto é, visível;

- requiredby: especifica o nome da classe de objetos do LDAP que obriga o preenchimento de um valor para o atributo. Por exemplo, os atributos uidNumber e gidNumber e homeDirectory são requeridos pela classe posixaccount do LDAP. Logo, ou se define o estes atributos como obrigatórios com o parâmetro required, ou se especifica o parâmetro requiredby="posixaccount" em cada um deles. Neste último caso, o MACA AD Web, antes de gravar um conta verifica se estes atributos estão vazios, se estiverem, então ele retira a classe posixaccount do atributo objectClass, permitindo, portanto, a gravação da conta sem estes atributos. Ou seja, o preenchimento em conjunto deste atributos fica opcional;

- allowedby: especifica o nome da classe de objetos do LDAP que permite o preenchimento de um valor para o atributo. No caso da classe posixaccount, o atributo loginShell é opcional. Sempre que os atributos obrigatórios da classe

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

posixaccount tiverem parâmetro requiredby definido (conforme item anterior), os atributos opcionais (caso do loginShell) devem ter o parâmetro allowedby="posixaccount" também definido. Isto somente é dispensado quando os parâmetros requeridos por uma classe de objetos são definidos como obrigatórios com o parâmetro required;

- maxlen: especifica o número máximo de caracteres do atributo;

- multivalue: valor booleano que especifica se o atributo admite o armazenamento de múltiplos valores ou não. O valor default é false, isto é, só permite o armazenamento de um valor;

- case: especifica se, antes do armazenamento, os caracteres do valor do atributo serão convertidos para letras maiúsculas (upper), minúsculas (lower) ou se apenas a primeira letra de cada palavra será convertidas para maiúscula, ficando as demais letras minúsculas (firstUpper);

- samevalueof: especifica o nome do atributo cujos valores são usados para preencher o valor deste campo no momento da gravação. Por exemplo, se o atributo uidNumber deve ter sempre o mesmo valor do atributo employeeNumber então deve-se definir o parâmetro samevalueof="employeeNumber" no atributo uidNumber. O parâmetro samevalueof ainda permite a definição do valor do atributo combinado com uma *string* constante. Por exemplo, se o valor do atributo homeDirectory deve ser igual ao valor atributo uid, prefixado pelo *string* /home/, então deve-se definir o parâmetro samevalueof="/home/'uid'" no atributo homeDirectory. No momento da gravação, o padrão 'uid' é substituído pelo valor do atributo uid;

- replacementregex: especifica uma expressão regular usada para localizar um padrão nos *string* do valor do atributo para substituir pelo valor do parâmetro replacementvalue. Por exemplo, a definição dos parâmetros replacementregex="usp" e replacementvalue="usp.br" substitui todas as ocorrências do *string* usp pelo *string* usp.br no momento de gravação do atributo. Para trocar apenas a primeira ocorrência, basta especificar o parâmetro replacementtype="first". O valor default é replacementtype="all".

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

O arquivo `$CATALINA_HOME/webapps/maca_ad_web/config/esquema_usuario_estendido.xml` apresenta uma versão estendida do esquema padrão. Para torná-lo o esquema corrente, altere a propriedade `maca_ca.user.schema` no arquivo `maca_ca.properties` para o seguinte valor

```
maca_ca.user.schema=<diretorio do tomcat>/webapps/maca_ad_web/config/esquema_usuario_estendido.xml
```

e reinicie o Tomcat. **Atenção, é OBRIGATÓRIO alterar esta propriedade para definir o mesmo esquema no `maca_ca.properties` do servidor MACA CS.**

DEFINIDO LOOKUPS PARA ATRIBUTOS DO ESQUEMA DE DADOS

O esquema de dados em XML permite a definição de *lookups* para validação e consulta em bancos de dados relacionais de valores para preenchimento de atributos armazenados nas contas de usuários do MACA. Suponha que exista uma tabela do recursos humanos de uma organização com o nome e o CPF dos funcionários e que o MACA AD Web somente deve permitir o cadastramento de usuários que são funcionários. Para estabelecer este *lookup* é preciso:

1. Editar o arquivo com o esquema de dados escolhido e acrescentar um elemento de dados XML especificado uma conexão com o banco de dados desejado. O exemplo a seguir mostra uma conexão para um banco de dados de recursos humanos num gerenciador Oracle:

```
<urls>
  <databaseurl name="rh"
               url="jdbc:oracle:thin:@bd.organizacao.com.br:1521:rh"
               driver="oracle.jdbc.driver.OracleDriver"
               username="<login do usuario do banco>"
               password="<senha do usuario do banco>">
  </databaseurl>
    ...
</urls>
```

   É importante frisar que o driver do banco da dados (um arquivo `.jar`) tem de estar no *class path* da máquina virtual Java usado pelo Tomcat, ou no diretório `$CATALINA_HOME/lib/`. Caso contrário, o MACA AD Web não conseguirá se conectar com o banco de dados;

2. Ainda no arquivo, acrescentar os *lookups* como consultas ao banco de dados através do seguinte elemento XML, que ilustra o exemplo de *lookup* à tabela de funcionários:

```
<lookups>
  <lookup name="funcionarios"
          databaseurlname="rh"
          query="select ID, Nome
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
                    from TAB_FUNC
                    where Situacao='Ativo'"
                case="upper">
    </lookup>
    ...
</lookups>
```

O parâmetro `case` converte toda a consulta para letras maiúsculas e é opcional. O default mantém as letras inalteradas.

3. Para concluir, é preciso agora acrescentar os parâmetros de *lookup* no atributo `employeeNumber`, conforme ilustrado a serguir:

```
<attribute name="employeeNumber" displayname="CPF" required="true" maxlen="12"
           lookupname="funcionarios" lookupupdatefield="ID" lookupdisplayfield="Nome"
           lookupchecked="true">
</attribute>
```

Em destaque, aparecem os parâmetros adicionados. `lookupname` especifica o nome do *lookup* utilizado. `lookupupdatefield` especifica qual campo da tabela é o usado para validar o preenchimento do atributo em questão. Ou seja, é o valor da coluna `ID` da consulta que será usado para validar o atributo LDAP `employeeNumber`. `lookupdisplayfield` especifica o nome da coluna pela qual se poderá fazer consultas para localizar um `ID` (CPF). `lookupchecked` é um valor booleano que, quando `true`, exige a presença do valor do atributo `employeeNumber` como um valor da coluna `ID` para permitir a gravação do registro da conta do usuário. Desta forma, somente são gravadas contas de usuários que são funcionários. O valor default deste parâmetro é `false`.

4. Após a definição dos *lookups*, salve o arquivo de esquema e reinicie o Tomcat. Não esquecer de copiar os drivers dos gerenciadores de bancos de dados utilizados para o diretório `$CATALINA_HOME/lib/`.

4.1.2 Configurando o MACA AD Web para Acessar o serviço LDAP via TLS/SSL

Antes de configurar o MACA AD Web para acessar o serviço LDAP sobre TLS/SSL (LDAPS), certifique-se que o LDAPS está habilitado. Neste caso, obtenha do responsável pelo serviço o certificado digital da AC que assinou o certificado instalado para ativação do LDAPS. Este certificado é dispensável se o certificado instalado para ativação do LDAPS foi assinado por uma AC confiável. Caso o certificado instalado para ativação do LDAPS seja auto assinado (não usa uma AC), ele deve ser

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

obtido. Nas instruções para habilitação do LDAPS no OpenLDAP neste manual, um certificado auto assinado é criado no arquivo `/etc/openldap/certificados/ldap.cert`.

Para configurar o MACA AD Web, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a configuração será realizada;
2. Mude o diretório corrente para `$CATALINA_HOME/webapps/maca_ad_web/config/`;
3. Copie o certificado disponível em `/etc/openldap/certificados/ldap.cert` para o diretório corrente (`$CATALINA_HOME/webapps/maca_ad_web/config/`);
4. Agora, é preciso criar ou atualizar o *keystore* para armazenar os certificados confiáveis para o MACA AD Web, no caso o certificado `ldap.cert`. Para isto, execute o comando a seguir e entre com os dados solicitados conforme o modelo:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias ldap -file ./ldap.cert -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha do keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado e armazena o certificado importado `ldap.cert`.

5. Agora, edite o arquivo `maca_ca.properties` e efetue as seguintes alterações:

Troque o valor da propriedade `maca_ca.ldap.server` para o nome DNS ou endereço IP onde o servidor OpenLDAP foi instalado, acrescido da porta `:636` do serviço LDAPS. Caso seja servidor local, coloque o valor `127.0.0.1:636`;

Acrescente as seguintes propriedades:

```
############################
# Configuracoes do LDAPS #
############################
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
maca_ca.ldap.ssl=true
maca_ca.ldap.ssl.console=true
maca_ca.ldap.ssl.keystore=<diretorio do tomcat>/webapps/maca_ad_web/config/ca_keystore
maca_ca.ldap.ssl.keystorepwd=<senha do keystore>
```

6. Salve o arquivo;
7. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

8. Agora, acesse o MACA AD Web entrando com a URL a seguir no seu *browser* para testar a nova configuração:

https://<dns do servidor do maca ad web>/maca_ad_web/

Agora, o MACA AD Web se conectará com o servidor OpenLDAP com o protocolo LDAPS na porta 636.

4.1.3 Configurando o MACA AD Web para Acessar o serviço MACA CS via IIOP sobre TLS/SSL

O MACA AD Web pode ser configurado para permitir ou obrigar que a conexão com o servidor MACA CS seja através do protocolo IIOP sobre TLS/SSL. Adicionalmente, o MACA CS pode ser configurando para permitir a sua autenticação como cliente, através de um certificado digital, junto ao servidor MACA CS.

Para configurar o MACA AD Web, siga os seguintes passos:

1. Obtenha os privilégios de administrador (*root*), conforme o sistema operacional onde a instalação será realizada;
2. Mude o diretório corrente para `$CATALINA_HOME/webapps/maca_ad_web/config/`;
3. Caso o certificado do servidor MACA CS seja auto assinado ou emitido por uma AC não confiável, é necessário criar um *keystore* para armazenar tais certificados. Caso as instruções da seção 2.2.3 tenham sido seguidas, obtenha com o responsável o certificado auto assinado criado disponível em

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

`/usr/local/maca_cs/bin/certificados/maca_cs.cer` e o copie para o diretório corrente (`$CATALINA_HOME/webapps/maca_ad_web/config/`);

4. Agora execute o comando a seguir e entre com os dados solicitados conforme o modelo para importar o certificado:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias maca_cs -file ./maca_cs.cer -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha para o keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado e armazena o certificado importado `maca_cs.cer`.

5. Edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
#####################################
# Configuracoes para IIOP sobre SSL #
#####################################

jacorb.security.support_ssl=on

jacorb.security.ssl.client.supported_options=20
jacorb.security.ssl.client.required_options=0

jacorb.ssl.socket_factory=org.jacorb.security.ssl.sun_jsse.SSLSocketFactory
jacorb.ssl.server_socket_factory=org.jacorb.security.ssl.sun_jsse.SSLServerSocketFactory

javax.net.ssl.trustStore=<diretorio do tomcat>/webapps/maca_ad_web/config/ca_keystore
javax.net.ssl.trustStorePassword=<senha do keystore>
```

Estas propriedades configuram o cliente para aceitar conexões IIOP sobre TLS/SSL, mas não o obrigam a realizar. Isto é, se o servidor MACA CS estiver configurado para obrigar conexões IIOP sobre TLS/SSL, então o cliente as realizará. Por outro lado, caso o MACA CS não obrigue tais conexões, o cliente fará uma conexão IIOP convencional. Para obrigar o cliente a realizar somente conexões IIOP sobre TLS/SSL, então altere o valor da propriedade `jacorb.security.ssl.client.required_options` para

```
jacorb.security.ssl.client.required_options=20
```


www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Esta configuração, entretanto, não habilita a autenticação do cliente junto ao MACA CS através de certificados digitais (ver item 9 a seguir para saber como configurar);

6. Salve o arquivo;
7. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

8. Agora, acesse o MACA AD Web entrando com a URL a seguir no seu *browser* para testar a nova configuração:

https://<dns do servidor do maca ad web>/maca_ad_web/

Para habilitar a autenticação do cliente, é preciso criar para ele um certificado digital, que poderá ser auto assinado, ou emitido por uma AC confiável para o MACA CS. Para criar um certificado auto assinado, execute o comando abaixo entrando os valores solicitados de acordo com o modelo apresentado:

```
$JAVA_HOME/bin/keytool -genkey -alias cliente -keyalg RSA -validity 365 -keystore ./cliente_keystore↵
Enter keystore password: <entre com a senha da chave privada> ↵
Enter keystore password: <repita a senha anterior> ↵
What is your first and last name?
   [Unknown]: <entre com o DNS completo da estação do cliente> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade, ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <cliente>
(RETURN if same as keystore password): ↵
```

Isto cria um certificado auto assinado e sua respectiva chave privada no arquivo `$CATALINA_HOME/webapps/maca_ad_web/config/cliente_keystore`. Para

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

requisição de um certificado para emissão por uma AC confiável, veja os detalhes em http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html.

O arquivo `cliente_keystore` é um tipo de banco de dados que armazena a chave privada e o certificado, que é público. É necessário exportar este certificado para fornecê-los a todos os servidores MACA CS que desejam aceitar conexões deste cliente através do IIOP sobre TLS/SSL. Para exportá-lo, execute o seguinte comando:

```
$JAVA_HOME/bin/keytool -export -alias cliente -file cliente.cer
-keystore ./cliente_keystore
```

Após a execução deste comando, o arquivo `cliente.cer` armazenará o certificado a ser fornecido para os servidores do MACA CS com os quais o cliente deseja autenticar;

9. Com o certificado do cliente criado, agora edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
jacorb.security.keystore=<diretorio do tomcat>/webapps/maca_ad_web/config/cliente_keystore
jacorb.security.keystore_password=<senha da chave privada>
```

Depois altere o valor da propriedade `jacorb.security.ssl.client.supported_options` para

```
jacorb.security.ssl.client.supported_options=40
```

Grave o arquivo.

Agora o cliente está apto a se autenticar junto ao serviço MACA CS. **ATENÇÃO, não esquecer que o MACA CS deve ter sido configurado para exigir a autenticação de clientes e que o certificado deste cliente deve ser um certificado confiável para o MACA CS.**

10. Reinicie o serviço do Tomcat executando os comandos (Linux):

```
cd $CATALINA_HOME/bin
./shutdown.sh
./startup.sh
```

No Windows NT/2000, faça assim:

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
cd %CATALINA_HOME%\bin
shutdown
startup
```

11. Agora, acesse o MACA CS Monitor entrando com a URL a seguir no seu *browser* para testar a nova configuração:

https://<dns do servidor do maca ad web>/maca_ad_web/

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

5 MACA AD

O MACA AD é um aplicação Java convencional e requer para o seu funcionamento a instalação prévia ou a disponibilidade do *Java™ 2 Runtime Environment (JRE), Standard Edition (build 1.4.x)* ou superior estável. A instalação e configuração do JRE 1.4.x deve seguir as recomendações da seção 2.1.3. O MACA AD funciona MS-Windows NT/2000 e requer um ambiente gráfico de janelas para funcionar no sistema operacional Linux, seguindo os mesmos requisitos de núcleo especificados para o MACA CS.

5.1 MACA AD – Instalação e Configuração

A instalação e configuração básica do MACA AD é bastante simples, considerando-se que os softwares e serviços requisitados por ele estão disponíveis. Para instalar e configurar o MACA AD, proceda da seguinte forma:

1. Copie o conteúdo do diretório `maca_deployment_VERSAO/maca_ad/` do diretório de distribuição do MACA para um diretório de sua preferência e torne este diretório o diretório corrente;
2. Agora é preciso efetuar as configurações da aplicação MACA AD para que ela possa ser utilizada. Para isto, edite o arquivo `maca_ca.properties` e faça o seguinte:

   Altere o valor da propriedade `maca_ca.maca_cs.url` para a URL com o nome do servidor onde o MACA CS foi instalado, e capaz de recuperar os arquivos `maca_cs.iors`, `monitor.ior`, `pa.ior` e `ado.ior` que foram criados no diretório raiz do serviço HTTP. Por exemplo, o valor deve ser alterado para:

   ```
   maca_ca.maca_cs.url=http\://macacs.organizacao.com.br
   ```

   Atenção, use o caracter de *escape* \ antes do sinal de dois pontos (\:);

   Altere o valor da propriedade `maca_ca.ldap.server` para o endereço (DNS ou IP) do servidor onde o servidor OpenLDAP foi instalado. Por exemplo, o valor deve ser alterado para:

   ```
   maca_ca.ldap.server=<nome do servidor OpenLDAP>.organizacao.com.br
   ```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Troque o valor da propriedade `maca_ca.ldap.rootDN` para o nome do sufixo escolhido (valor da propriedade `suffix`) durante a instalação do OpenLDAP. Por exemplo, o valor deve ser alterado para:

```
maca_ca.ldap.rootDN=dc\=<nome organizacao>,dc\=com,dc\=br
```

Atenção, não deixe espaços em branco entre as vírgulas e use o caracter de *escape* \ antes do sinal de igual (\=);

Grave o arquivo;

Observação: os arquivos de configuração `maca_ca.properties` e `orb.properties` também estão incluídos como recursos dentro do arquivo `maca_ad-VERSAO-app.jar`. Portanto, eles podem ser alterados dentro deste arquivo utilizando o WinZip. Neste caso, não há necessidade dos arquivos de propriedades externos, que **DEVEM** ser removidos do diretório, pois têm precedência na definição de propriedades (sobrepõem os arquivos internos);

3. Verifique se a URL abaixo retorna o conteudo do arquivo maca_cs.iors:

   http://<DNS do servidor do MACA CS>/maca_cs.iors

   O *browser* deve retornar algo parecido com a saída abaixo,


```
paIOR:000000000000003649444C3A6F6D672E6F72672F53656375726974794C6576656C322F5072696E636970616C41757468656E74696361746F72
3A312E3000000000000002000000000000007400010200000000F3230302E3134342E37332E3130360000B7B000000000001A5374616E6461726449
6D706C4E616D652F70615F504F412F706100000000000200000000000000080000000004A414300000000010000001C0000000000010001000000105
0100010001010900000001050100010000000010000002C0000000000000001000000010000001C0000000000010001000000010501000100010109000
00000105010001;adoIOR:000000000000003849444C3A6F6D672E6F72672F44665265736F7572636541636365737344656369736973696F6E2F41636365
737344656369736973696F6E3A312E3000000000020000000000000074000102000000000F3230302E3134342E37332E3130360000B7B000000000001C53
74616E64617264496D706C4E616D652F61646F5F504F412F61646F000000020000000000000080000000004A414300000000010000001C0000000000
0100010000000105010001000101090000000105010001000000010000002C0000000000000001000000010000001C0000000000010001000000010501
0100010001010900000000105010001;monitorIOR:000000000000001849444C3A6D6163615F63732F4D6F6E69746F723A312E3000000000002000000
000000007C0001020000000000F3230302E3134342E37332E3130360000B7B000000000002453746164E64617264496D706C4E616D652F6D6F6E69746F
725F504F412F6D6F6E69746F72000000020000000000000080000000004A414300000000010000001C0000000000010001000000010501000100010
0900000001050100010000000010000002C0000000000000001000000010000001C0000000000010001000000010501000100010109000000001050100
01;
```


   mas numa única linha. Com isto, o MACA AD está pronto para funcionar;

4. Inicie o MACA AD executando os comandos (Linux):

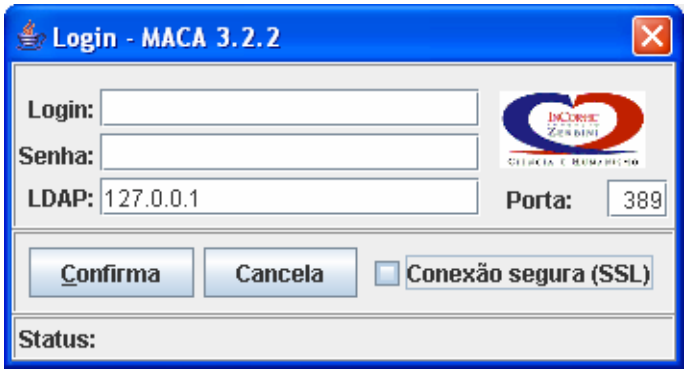
   ```
   $JAVA_HOME/bin/java –jar maca_ad-VERSAO-app.jar
   ```

   No Windows NT/2000, faça assim:

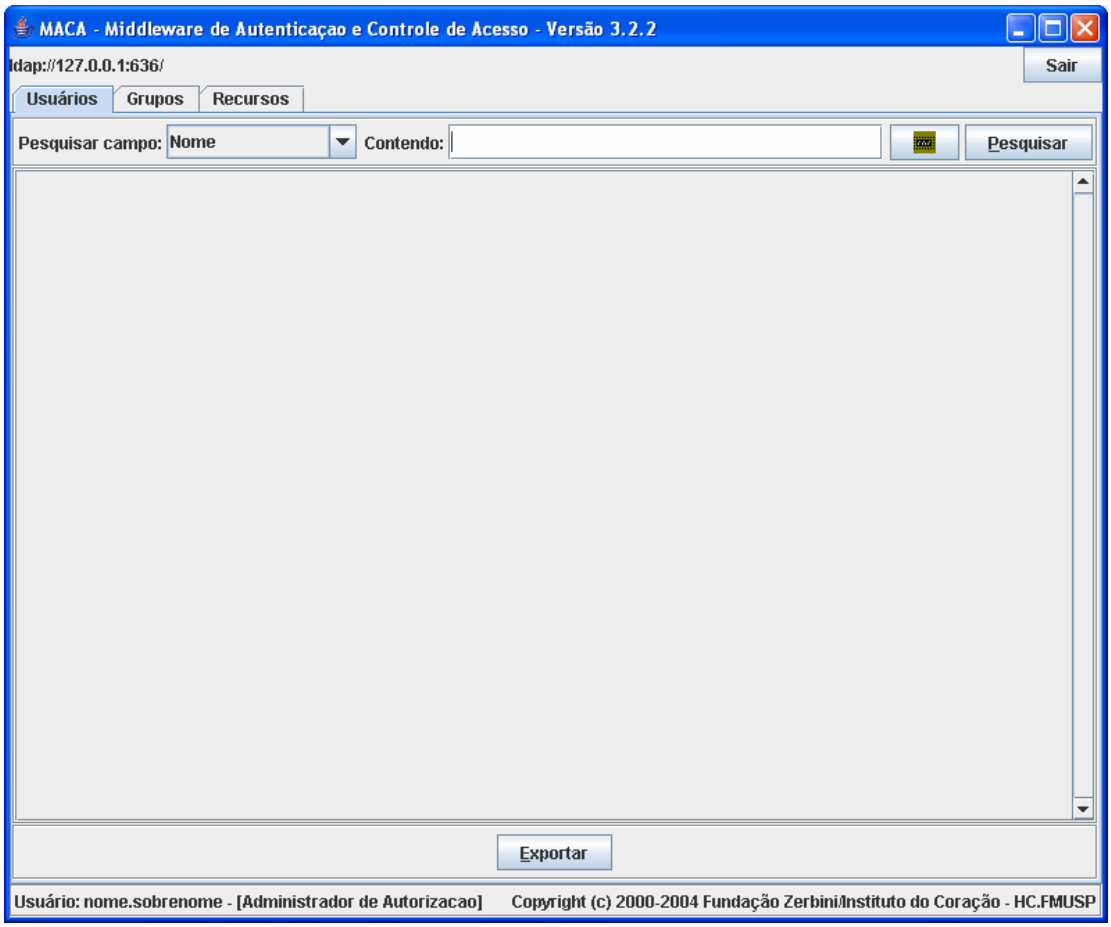
   `"%JAVA_HOME%\bin\java" –jar maca_ad-VERSAO-app.jar` *ou simplesmente*
   `java –jar maca_ad-VERSAO-app.jar` *ou ainda*
   clique com botão esquerdo do mouse sobre o arquivo `maca_ad-VERSAO-app.jar`

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

A seguinte janela deverá aparecer:



5. Entre com o login do usuário padrão carregado do arquivo `maca_deployment_VERSAO/ldap.confs/maca_inicial.ldif` durante a instalação do OpenLDAP. O usuário padrão tem o login **`nome.sobrenome`** e a senha **`maca`**. Após o login, a pasta de consulta a contas de usuários do MACA é exibida, conforme ilustrado abaixo:



Com esta aplicação, os administradores de autorização do MACA poderão administrar a política de autorização de acesso, que permite a criação, a alteração, a remoção e a consulta a recursos, papéis e autorizações.

Mais detalhes na utilização do MACA AD são encontrados no manual "MACA Administrativo - Manual do Usuário", integrante do diretório de distribuição do MACA.

5.1.1 Configurando o MACA AD para Acessar o serviço LDAP via TLS/SSL

Antes de configurar o MACA AD para acessar o serviço LDAP sobre TLS/SSL (LDAPS), certifique-se que o LDAPS está habilitado. Neste caso, obtenha do responsável pelo serviço o certificado digital da AC que assinou o certificado instalado para ativação do LDAPS. Este certificado é dispensável se o certificado instalado para ativação do LDAPS foi assinado por uma AC confiável. Caso o certificado instalado para ativação do LDAPS seja auto assinado (não usa uma AC), ele deve ser obtido. Nas instruções para habilitação do LDAPS no OpenLDAP neste manual, um certificado auto assinado é criado no arquivo `/etc/openldap/certificados/ldap.cert`.

Para configurar o MACA AD Web, siga os seguintes passos:

1. Mude-se para o diretório onde o MACA AD foi instalado;
2. Copie o certificado disponível em `/etc/openldap/certificados/ldap.cert` para o diretório corrente;
3. Agora, é preciso criar ou atualizar o *keystore* para armazenar os certificados confiáveis para o MACA AD, no caso o certificado `ldap.cert`. Para isto, execute o comando a seguir e entre com os dados solicitados conforme o modelo:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias ldap -file ./ldap.cert -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha do keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

No Windows NT/2000, entre

```
"%JAVA_HOME%/bin/keytool" -import -trustcacerts -alias ldap -file ./ldap.cert -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha do keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
```

```
        SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado ou atualizado e armazena o certificado importado `ldap.cert`.

Agora, edite o arquivo `maca_ca.properties` e acrescente as seguintes propriedades:

```
#########################
# Configuracoes do LDAPS #
#########################

maca_ca.ldap.ssl=true
maca_ca.ldap.ssl.console=false
maca_ca.ldap.ssl.keystore=./ca_keystore
maca_ca.ldap.ssl.keystorepwd=<senha do keystore>
```

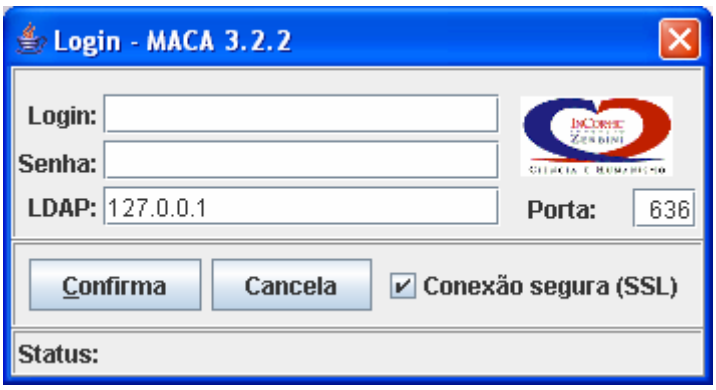
4. Salve o arquivo;
5. Inicie o MACA AD executando os comandos (Linux):

```
$JAVA_HOME/bin/java –jar maca_ad-VERSAO-app.jar
```

No Windows NT/2000, faça assim:

`"%JAVA_HOME%\bin\java" –jar maca_ad-VERSAO-app.jar` *ou simplesmente*
`java –jar maca_ad-VERSAO-app.jar` *ou ainda*
clique com botão esquerdo do mouse sobre o arquivo `maca_ad-VERSAO-app.jar`

A seguinte janela deverá aparecer:



Note que já vem com o SSL habilitado.

5.1.2 Configurando o MACA AD para Acessar o serviço MACA CS via IIOP sobre TLS/SSL

O MACA AD pode ser configurado para permitir ou obrigar que a conexão com o servidor MACA CS seja através do protocolo IIOP sobre TLS/SSL. Adicionalmente, o MACA CS pode ser configurando para permitir a sua autenticação como cliente, através de um certificado digital, junto ao servidor MACA CS.

Para configurar o MACA AD, siga os seguintes passos:

1. Mude-se para o diretório onde o MACA AD foi instalado;
2. Caso o certificado do servidor MACA CS seja auto assinado ou emitido por uma AC não confiável, é necessário criar um *keystore* para armazenar tais certificados. Caso as instruções da seção 2.2.3 tenham sido seguidas, obtenha com o responsável o certificado auto assinado criado disponível em `/usr/local/maca_cs/bin/certificados/maca_cs.cer` e o copie para o diretório corrente;
3. Agora execute o comando a seguir e entre com os dados solicitados conforme o modelo para importar o certificado:

```
$JAVA_HOME/bin/keytool -import -trustcacerts -alias maca_cs -file ./maca_cs.cer -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha para o keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

No Windows NT/2000, entre

```
"%JAVA_HOME%/bin/keytool" -import -trustcacerts -alias maca_cs -file ./maca_cs.cer -keystore ./ca_keystore↵
Enter keystore password: <entre com uma senha para o keystore>↵
Owner: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD, O=InCor, L=Sao Paulo, ST=SP, C=BR
Issuer: EMAILADDRESS=gustavo.motta@incor.usp.br, CN=pc440.incor.usp.br, OU=SPD,O=InCor, L=Sao Paulo, ST=SP, C=BR
Serial number: 0
Valid from: Wed May 28 10:52:33 BRT 2003 until: Thu May 27 10:52:33 BRT 2004
Certificate fingerprints:
         MD5:  D6:FC:AF:CC:E0:F7:50:89:BA:A4:C4:85:6A:0C:C9:6A
         SHA1: 40:67:07:E1:72:4B:60:78:A5:37:04:0F:57:10:4E:72:66:EB:45:36
Trust this certificate? [no]:  yes↵
Certificate was added to keystore
```

Note que o arquivo `ca_keystore` foi criado ou atualizado e armazena o certificado importado `maca_cs.cer`.

4. Edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
#####################################
# Configuracoes para IIOP sobre SSL #
#####################################
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
jacorb.security.support_ssl=on

jacorb.security.ssl.client.supported_options=20
jacorb.security.ssl.client.required_options=0

jacorb.ssl.socket_factory=org.jacorb.security.ssl.sun_jsse.SSLSocketFactory
jacorb.ssl.server_socket_factory=org.jacorb.security.ssl.sun_jsse.SSLServerSocketFactory

javax.net.ssl.trustStore=./ca_keystore
javax.net.ssl.trustStorePassword=<senha do keystore>
```

Estas propriedades configuram o cliente para aceitar conexões IIOP sobre TLS/SSL, mas não o obrigam a realizar. Isto é, se o servidor MACA CS estiver configurado para obrigar conexões IIOP sobre TLS/SSL, então o cliente as realizará. Por outro lado, caso o MACA CS não obrigue tais conexões, o cliente fará uma conexão IIOP convencional. Para obrigar o cliente a realizar somente conexões IIOP sobre TLS/SSL, então altere o valor da propriedade `jacorb.security.ssl.client.required_options` para

```
jacorb.security.ssl.client.required_options=20
```

Esta configuração, entretanto, não habilita a autenticação do cliente junto ao MACA CS através de certificados digitais (ver item 6 a seguir para saber como configurar);

5. Salve o arquivo;
6. Inicie o MACA AD executando os comandos (Linux):

```
$JAVA_HOME/bin/java –jar maca_ad-VERSAO-app.jar
```

No Windows NT/2000, faça assim:

```
"%JAVA_HOME%\bin\java" –jar maca_ad-VERSAO-app.jar ou simplesmente
java –jar maca_ad-VERSAO-app.jar                   ou ainda
clique com botão esquerdo do mouse sobre o arquivo maca_ad-VERSAO-app.jar
```

7. Para habilitar a autenticação do cliente, é preciso criar para ele um certificado digital, que poderá ser auto assinado, ou emitido por uma AC confiável para o MACA CS. Para criar um certificado auto assinado, execute o comando abaixo entrando os valores solicitados de acordo com o modelo apresentado:

```
$JAVA_HOME/bin/keytool -genkey -alias cliente -keyalg RSA -validity 365 -keystore ./cliente_keystore↵
Enter keystore password: <entre com a senha da chave privada> ↵
Enter keystore password: <repita a senha anterior> ↵
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

```
What is your first and last name?
   [Unknown]: <entre com o DNS completo da estação do cliente> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade, ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <cliente>
(RETURN if same as keystore password): ↵
```

No Windows NT/2000, faça assim:

```
"%JAVA_HOME%/bin/keytool" -genkey -alias cliente -keyalg RSA -validity 365 -keystore ./cliente_keystore↵
Enter keystore password: <entre com a senha da chave privada> ↵
Enter keystore password: <repita a senha anterior> ↵
What is your first and last name?
   [Unknown]: <entre com o DNS completo da estação do cliente> ↵
What is the name of your organizational unit?
   [Unknown]: <entre com o nome do setor onde você trabalha> ↵
What is the name of your organization?
   [Unknown]: <entre com o nome da empresa onde você trabalha> ↵
What is the name of your City or Locality?
   [Unknown]: <entre com o nome da cidade onde você trabalha> ↵
What is the name of your State or Province?
   [Unknown]: <entre com a sigla do estado onde você trabalha> ↵
What is the two-letter country code for this unit?
   [Unknown]: BR ↵
Is CN=organizacao.com.br, OU=setor, O=organizacao, L=cidade, ST=estado, C=BR correct?
   [no]: yes ↵
Enter key password for <cliente>
(RETURN if same as keystore password): ↵
```

Isto cria um certificado auto assinado e sua respectiva chave privada no arquivo `cliente_keystore`. Para requisição de um certificado para emissão por uma AC confiável, veja os detalhes em http://java.sun.com/j2se/1.4.2/docs/tooldocs/solaris/keytool.html.

O arquivo `cliente_keystore` é um tipo de banco de dados que armazena a chave privada e o certificado, que é público. É necessário exportar este certificado para fornecê-los a todos os servidores MACA CS que desejam aceitar conexões deste cliente através do IIOP sobre TLS/SSL. Para exportá-lo, execute o seguinte comando:

```
$JAVA_HOME/bin/keytool -export -alias cliente -file cliente.cer
-keystore ./cliente_keystore
```

No Windows NT/2000, faça assim:

```
"%JAVA_HOME%/bin/keytool" -export -alias cliente -file cliente.cer
-keystore ./cliente_keystore
```


www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Após a execução deste comando, o arquivo `cliente.cer` armazenará o certificado a ser fornecido para os servidores do MACA CS com os quais o cliente deseja autenticar;

8. Com o certificado do cliente criado, agora edite o arquivo `orb.properties` e acrescente as seguintes propriedades:

```
jacorb.security.keystore=./cliente_keystore
jacorb.security.keystore_password=<senha da chave privada>
```

Depois altere o valor da propriedade `jacorb.security.ssl.client.supported_options` para

```
jacorb.security.ssl.client.supported_options=40
```

Grave o arquivo.

Agora o cliente está apto a se autenticar junto ao serviço MACA CS. **ATENÇÃO, não esquecer que o MACA CS deve ter sido configurado para exigir a autenticação de clientes e que o certificado deste cliente deve ser um certificado confiável para o MACA CS.**

9. Inicie o MACA AD executando os comandos (Linux):

```
$JAVA_HOME/bin/java –jar maca_ad-VERSAO-app.jar
```

No Windows NT/2000, faça assim:

`"%JAVA_HOME%\bin\java" –jar maca_ad-VERSAO-app.jar` *ou simplesmente* `java –jar maca_ad-VERSAO-app.jar` *ou ainda clique com botão esquerdo do mouse sobre o arquivo* `maca_ad-VERSAO-app.jar`

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

6 Propriedades de Configuração

6.1.1 maca_ca.properties

As propriedades do arquivo **maca_ca.properties** estão disponíveis para os módulos MACA CS, MACA AD Monitor e MACA AD.

Tabela 1 – Configuração dos arquivos de log – usado no MACA CS

Propriedade	Default	Descrição
maca_ca.log.LogFileCreationFrequency	D	Define a freqüência de criação de arquivos de log. Os valores possíveis são: H – Hourly, D – Daily, W – Weekly, M – Monthly, Y – Yearly e N – None. Por exemplo, de for escolhida a opção M, um novo arquivo de log será criado ao final de cada mês. O arquivo antigo é renomeado anexando-se a data e hora ao nome do arquivo, que passa a ter a extensão '.old'. O valor default é D.
maca_ca.log.LogFileMaxLength	0	Tamanho máximo de arquivo de log em Mbytes. Quando o log atual alcança este limite, um novo arquivo é criado e o arquivo antigo é renomeado para anexar a data e a hora correntes ao seu nome, seguido da extensão '.old'.; O valor 0 (ou negativo) indica que não há limite máximo e é o valor default.
maca_ca.log.LogMaxDirSize	250	Tamanho máximo, em Mbytes, do total de arquivos de log renomeados para a extensão '.old' no diretório para armazenamento de logs (propriedade maca_cs.log.Path). Quando o limite é ultrapassado, os arquivos mais antigos são removidos de modo que tamanho máximo do total de arquivos de log renomeados seja inferior a este limite. O valor 0 (ou negativo) indica que não há limite máximo.
maca_ca.log.LogFileMaxLengthFreq	5	Freqüência (em minutos) com que será feita a verificação do tamanho máximo individual do arquivo (propriedade maca_ca.log.LogFileMaxLength). O seu valor deve ser > 0 e a freqüência default é de 5 minutos.
maca_ca.log.LogMaxDirSizeFreq	60	Freqüência (em minutos) com que será feita a verificação do tamanho máximo do total de arquivos de log renomeados (propriedade maca_ca.log.LogMaxDirSize). O seu valor deve ser > 0 e a freqüência default é de 60 minutos.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Tabela 2 – Configuração da política de senhas– usado no MACA CS e MACA AD Web

Propriedade	Default	Descrição
maca_ca.passwordPolicy.passwordStorageScheme	SHA	Indica o algoritmo usado pelo MACA para codificação da senha no servidor LDAP. Os algoritmos atualmente suportados são o SHA, SHA-1, MD5 e CLEAR, que armazena a senha limpa, sem codificação.
maca_ca.passwordPolicy.passwordCheckSyntax	false	Indica se a sintaxe da senha será verificada antes que uma nova senha seja definida. Quando 'true', o MACA verificará o tamanho mínimo (maca_ca.passwordPolicy.passwordMinLength) para senhas e se a senha não possui valores comuns, como o nome do usuário.
maca_ca.passwordPolicy.passwordMinLength	6	Número mínimo de caracteres permitido para uma senha quando a propriedade maca_ca.passwordPolicy.passwordCheckSyntax está ativada.
maca_ca.passwordPolicy.passwordExp	false	Indica se a senha expira depois de decorrido um dado número de segundos. Quando a senha expira, o usuário é obrigado a trocar de senha.
maca_ca.passwordPolicy.passwordMaxAge	8640000	Número de segundos usado para definir a data e hora para expiração da senha do usuário quando a propriedade maca_ca.passwordPolicy.passwordExp está ativada. O valor *default* corresponde a 100 dias (em segundos). Esta propriedade é usada apenas como referência, sugerindo uma data padrão. Porém, a data real de expiração da senha pode ser personalizada na conta de cada usuário pelo administrador de contas.
maca_ca.passwordPolicy.passwordWarning	86400	Tempo em segundos antes da data de expiração da senha do usuário, após o qual o usuário será alertado de que sua senha irá expirar. O valor *default* corresponde a 10 dias.
maca_ca.passwordPolicy.passwordKeepHistory	false	Indica se o histórico das últimas senhas deve ser mantido.
maca_ca.passwordPolicy.passwordInHistory	6	Quantidade senhas antigas mantidas no histórico do usuário.
maca_ca.passwordPolicy.passwordChange	true	Indica se o usuário pode mudar a própria senha.
maca_ca.passwordPolicy.passwordMustChange	false	Indica que o usuário deve modificar sua senha na primeira autenticação ou quando a senha é "resetada" pelo administrador.
maca_ca.passwordPolicy.passwordMinAge	0	Interstício em segundos antes do qual o usuário não poderá mudar a própria senha.
maca_ca.passwordPolicy.passwordFactory	maca_ca.SimplePasswordFactory	Nome da classe para geração de senhas iniciais do MACA. A classe default gera senhas iguais ao login do usuário. A outra opção é usar a classe 'maca_ca.BasicPasswordFactory', que gera senhas aleatórias com 6 caracteres e a envia por E-mail para o usuário. Ver propriedade `maca_ca.mail` na Tabela 5

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Tabela 3 – Configuração do LDAP– usado no MACA CS, MACA AD Web e MACA AD

Propriedade	Default	Descrição
maca_ca.ldap.product	`Openldap`	Nome produto que implementa o protocolo LDAP. Outro valor possível é o "iplanet".
maca_ca.ldap.server	`127.0.0.1`	Nome (endereço IP ou DNS) do servidor LDAP. Deve ser modificado quando o servidor LDAP estiver funcionando numa máquina diferente da máquina onde o MACA CS ou o MACA AD executam. Pode ser sufixada com o número da porta, quando necessário, por exemplo: 127.0.0.1:636 para conexões TLS/SSL.
maca_ca.ldap.rootDN	`dc=root`	Informa o DN raiz do servidor LDAP. Deve ser modificado para o DN utilizado na sua instituição, por exemplo: dc=organizacao,dc=com,dc=br.
maca_ca.ldap.searchableFields	`uid, givenname, cn, sn, mail, employeenumber`	Usado pelo MACA AD Web, indica por quais atributos LDAP um administrador de contas pode efetuar consultas.
maca_ca.ldap.returningFields	`uid, givenname, cn, sn, mail, employeenumber`	Usado pelo MACA AD Web, indica quais atributos LDAP podem retornar numa consulta.
maca_ca.ldap.filterFields		Usado pelo MACA AD Web, indica por quais atributos LDAP a consulta realizada por um administrador de contas será filtrada de forma compulsória. De outro modo, o MACA AD Web só irá permitir que o administrador consulte contas de usuários cujos atributos (presentes na propriedade `maca_ca.ldap.filterFields`) são um subconjunto dos mesmos atributos da conta do próprio administrador. Por exemplo, se esta propriedade tem o atributo `st` (unidade da federação) definido, então o administrador de contas somente poderá consultar contas de usuários cujo atributo `st` tenha o mesmo valor que o seu próprio `st`, ou um subconjunto do mesmo no caso de atributos multivalorados. Logo, um administrador com st igual a SP só poderia consultar usuários de SP, etc.
maca_ca.ldap.groupsRDN	`ou=groups`	Informa o RDN, a partir do DN raiz, onde estão localizados os grupos de 'papéis' e de 'organograma' no servidor LDAP.
maca_ca.ldap.peopleRDN	`ou=people`	Informa o RDN, a partir do DN raiz, onde estão localizadas as entradas dos registros de usuários no servidor LDAP.
maca_ca.ldap.roleRDN	`cn=papeis`	Informa o RDN, a partir do RDN dos grupos, onde estão localizados os grupos de 'papéis' no servidor LDAP.
maca_ca.ldap.orgChartRDN	`cn=organograma`	Informa o RDN, a partir do RDN dos grupos, onde estão localizados os grupos de 'organograma' no servidor LDAP.
maca_ca.ldap.resourcesRDN	`ou=resources`	Informa o RDN, a partir do DN raiz, onde estão localizadas as entradas que representam os recursos protegidos no servidor LDAP.
maca_ca.ldap.authorizationsRDN	`ou=authorizations`	Informa o RDN, a partir do DN raiz, onde estão localizadas as entradas que armazenam as autorizações no servidor LDAP.
maca_ca.ldap.groupObjectClass	`incorgroup`	Nome da classe de objetos LDAP que possui atributos de agrupamento.
maca_ca.ldap.groupMemberAttribute	`member`	Nome do atributo que armazena os DNs dos membros de um grupo.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Tabela 4 – Configuração de segurança do LDAP– usado no MACA CS, MACA AD Web e MACA AD

Propriedade	Default	Descrição
maca_ca.ldap.ssl	`false`	Indica se as conexões entre o servidor LDAP e o MACA CS ou MACA AD são seguras (usam o protocolo SSL) ou não.
maca_ca.ldap.ssl.keystore	`maca_ca_cacerts`	Nome do arquivo onde são armazenados os certificados das autoridades certificadoras de confiança do MACA.
maca_ca.ldap.ssl.keystorepwd		Senha para autorização de acesso ao conteúdo do arquivo indicado na propriedade 'maca_ca.ldap.ssl.keystore'. O valor default é nulo, o que obriga o usuário a entrar explicitamente com a senha sempre que necessário.
maca_ca.ldap.ssl.console	`True`	Indica o tipo de interface com o usuário que o MACA usa para obter senha e a opção sobre a confiabilidade de um certificado de servidor. A console de linha de comando é a usada por default. Quando falso, é usada uma interface gráfica baseada em janelas.
maca_ca.ldap.ssl.factory.socket	`maca_ca.ssl.AutomaticSocketFactory`	Construtor de sockets SSL customizado para o maca que permite o tratamento adequado de certificados de servidores emitidos por autoridades certificadoras não conhecidas do ambiente Java.

Tabela 5 – Miscelânea

Propriedade	Default	Descrição
maca_ca.maca_cs.url	`http://127.0.0.1`	URL HTTP onde o servidor MACA CS foi iniciado (ou algum proxy) para recuperação dos arquivos com endereços do CORBA IOR. Usada nos módulos MACA AD Web e MACA AD.
maca_ca.mail		E-mail do administrador do MACA e deve ser definido sempre que a propriedade `maca_ca.passwordPolicy.passwordFactory` for definida para 'maca_ca.BasicPasswordFactory' (ver Tabela 2). Usada no módulo MACA AD Web.
maca_ca.user.schema		Especifica o nome completo ou relativo do arquivo XML que especifica o esquema de dados para a conta de usuários armazenada no servidor LDAP. (ver seção 4.1.1)
maca_ca.user.cache.length	`10`	Número de posições no cache de autorizações de acesso, por usuário. Isto é, fica no cache do usuário os resultados dos últimos "maca_ca.user.cache.length" acessos. Quando o usuário encerra todas as sessões abertas, o cache é limpo. Usada no módulo MACA CS.

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

6.1.2 maca_cs.properties

As propriedades do arquivo **maca_cs.properties** estão disponíveis exclusivamente para o módulo MACA CS.

Tabela 6 – Configuração de usuários e sessões

Propriedade	Default	Descrição
maca_cs.user.cache	`false`	Ativa (true) ou desativa (false) o cache de autorizações de acesso para os usuários do MACA CS.
maca_cs.user.maxNumber	`0`	Número máximo de usuários simultâneos. Utilizado para evitar sobrecargas no sistema, limitando o número máximo de usuários simultâneos àquele adequado ao ambiente operacional onde está instalado. O valor zero ou negativo indica que não há limite máximo.
maca_cs.user.maxSessionsPerUser	`0`	Número máximo de sessões simultâneas por usuário. Utilizado para evitar sobrecargas no sistema, limitando o número máximo de sessões simultâneas por usuário àquele adequado ao ambiente operacional onde está instalado. O valor zero ou negativo indica que não há limite máximo.
maca_cs.ClientCallbackRefreshTime	`5`	Intervalo em minutos para o MACA CS realizar callbacks nos cliente para verificar queda ou inatividade (veja as seções 2.5 ou 3.5 do MACA Cliente – Guia do Programador para saber como definir um *callback*).
maca_cs.LoginTimeout	`60`	Tempo em minutos para encerramento por inatividade da sessão de um usuário.
maca_cs.ldap.userAccountAttrList		Informa a lista de nomes de atributos da conta do usuário no servidor LDAP cujos valores poderão ser recuperados através da interface Credentials (veja as seções 2.3.2 ou 3.3.2 do MACA Cliente – Guia do Programador para saber como obtê-los). Os nomes de atributos são separados por vírgulas. Por exemplo, para recuperar o nome completo do usuário e sua matrícula, deve-se incluir esta propriedade com os seguintes valores: `givenName, employeeNumber`

Tabela 7 – Configuração dos arquivos de log

Propriedade	Default	Descrição
maca_cs.log.Path		Nome do caminho onde os arquivos de log do MACA CS serão gravados. Se não for informado, o MACA CS usa o console como saída padrão (default).
maca_cs.log.StatisticsRefreshTime	`5`	Freqüência em minutos com que o log de estatísticas de atividade do MACA CS será atualizado.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

Tabela 8 – Miscelânea

Propriedade	Default	Descrição
maca_cs.IOROutputDir	.	Indica onde salvar os arquivos com o IOR dos serviços de autenticação (pa.ior) e de autorização de acesso (ado.ior). Em geral, dever ser o diretório raiz do servidor HTTP onde o MACA CS está executando.
maca_cs.contexts	`dtCtx,contexts.DateTimeContext`	Informa configuração das extensões dos contextos. A sintaxe é a seguinte: <nome_do_contexto,nome_da_classe_do_contexto><;nome_do_contexto, nome_da_classe_do_contexto>*. Define o contexto de datas como default. Os contextos de usuários (`userCtx`) e do LDAP (`ldapCtx`) são predefinidos.
maca_cs.service.stop.port	`8375`	Informa a porta que o MACA CS usa para o seu serviço de encerramento – maca_cs_shutdown.jar.

Tabela 9 – Configuração para o serviço de nomes

Propriedade	Default	Descrição
maca_cs.naming.useService	`false`	Indica se o MACA CS deverá ser iniciado utilizando o serviço de nomes CORBA.
maca_cs.naming.rootContext	`maca_cs.ctx`	Indica o nome do contexto raiz do MACA CS no serviço de nomes CORBA.
maca_cs.naming.authenticatorsSubContext	`authenticators.ctx`	Indica o subcontexto, abaixo do contexto raiz do MACA CS, onde se encontram os nomes objetos de autenticação disponíveis através do MACA CS.
maca_cs.naming.PrincipalAuthenticator	`pa.server`	Nome do objeto que implementa a interface "PrincipalAuthenticator", abaixo do subcontexto "maca_cs.ctx/`authenticators.ctx/`".
maca_cs.naming.authorizersSubContext	`authorizers.ctx`	Indica o subcontexto, abaixo do contexto raiz do MACA CS, onde se encontram os nomes objetos de autorização de acesso disponíveis através do MACA CS.
maca_cs.naming.AccessDecisionObject	`ado.server`	Nome do objeto que implementa a interface "AccessDecisionObject", abaixo do subcontexto "maca_cs.ctx/`authorizers.ctx/`".

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

6.1.3 maca_bean.properties

Tabela 10 – Configuração pelos módulos MACA AD Web e MACA CS Monitor

Propriedade	Default	Descrição
maca_bean.maca_cs.url	`http://127.0.0.1`	URL onde localizar os arquivos com os IORs dos serviços de autenticação e de controle de acesso do MACA CS. Deve obedecer o seguinte formato: `http://<DNS ou IP do servidor MACA CS>`.
maca_bean.web.session.timeout	`30`	Tempo em minutos para o timeout por inatividade da sessão web de um usuário do MACA AD Web ou MACA CS Monitor.

www.incor.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

7 Definições e Acrônimos

- **ADO**: Access Decision Object;
- **API**: Application Programming Interface;
- **BIGS**: Base de Informações de Gerenciamento de Segurança;
- **CABP**: Controle de Acesso Baseado em Papéis;
- **CORBA**: Common Object Request Broker Architecture;
- **CSS**: CORBA Security Service;
- **HC.FMUSP**: Hospital das Clínicas da Faculdade de Medicina da Universidade de São Paulo;
- **IDL**: Interface Definition Language;
- **JSP**: Java Server Pages;
- **JRE**: Java Run-time Environment;
- **SDK**: Software Development Kit;
- **InCor**: Instituto do Coração;
- **IOR**: Interoperable Object Reference;
- **LDAP**: Lightweight Directory Access Protocol;
- **LDIF**: LDAP Data Interchange Format
- **MACA AD**: módulo para administração das políticas de autorização e controle de acesso e gerência de contas de usuários;
- **MACA CS**: servidor CORBA do MACA para autenticação de usuários e controle e acesso;
- **MACA**: *Middleware* de Autenticação e Controle de Acesso;
- ***Middleware***: serviço propósito geral que se situa entre plataformas e aplicações;

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP

- **NIST**: National Institute of Standards and Technology;
- **OMG**: Object Management Group;
- **ORB**: Object Request Broker;
- **RAD – Facility**: Resource Access Decision Facility;
- **SASL**: Simple Authentication and Security Layer;
- **SGBD**: Sistema Gerenciador de Bancos de Dados;
- **TLS:** Transport Layer Security;
- **XML**: *eXtensible Markup Language*;

www.**incor**.usp.br
Av. Dr. Enéas de Carvalho Aguiar, 44 – 05403-000 São Paulo – SP